E-mail Server Administration Guide

Genesys Engage Email 8.5.2

10/1/2024

Table of Contents

E-Mail Server Administration Guide

This guide provides information for administrators regarding E-mail Server. In addition to the topics on this page, there is also information on the following:

- How E-mail Server uses JavaMail properties.
- The types of notification of delivery status and message disposition.
- How to customize the e-mails from external resources.
- How to set up Microsoft Azure or Gmail mailboxes for OAuth 2.0 authorization.
- How to split the To and CC email recipient list and generate an interaction (an individual email message) for each unique recipient in the list.

See also information, applying to both UCS and E-mail Server, on mixing IPv6 and IPv4 and on running the server as a Windows Service with TLS.

Limitations

- Attachments—There is no limit on the size of attachments to e-mails. You can use the `maximum-msg-size` option to limit the overall size of incoming messages (that is, the total size of all message parts, including the body and any attachments).
- UCS:
  - E-mail Server 8.1.2 can work only with UCS 8.1.1 or later (however UCS 8.1.1 can work with any version of E-mail Server).
  - E-mail Server 8.1.3 or later requires UCS 8.1.3 or later.
  - E-mail Server 8.5.x requires UCS 8.1.4 or later.
- For optimal performance, Genesys recommends that you use no more than 25 mailboxes with each instance of E-mail Server.

Note on deleting interactions in strategies

In its requests to UCS, E-mail Server provides parameters for tenant ID, Interaction type, Interaction subtype, status, and parent ID.

Therefore, when E-mail Server updates threaded interactions in UCS, the parent interaction must still exist in the UCS database. For example, in the case of a chat interaction and a chat transcript being sent, the parent must not be deleted before E-mail Server successfully sends the transcript.

In versions prior to 8.1.400.10, when E-mail Server sent an e-mail, it incorrectly updated the corresponding interaction in the UCS database. This incorrect update prevented statistics from being computed correctly.

List of Attached Files

Starting in release 8.1.0, inbound e-mails can include an attached data type `_AttachmentFileNames`, which contains a list of the names of files attached to the inbound e-mail.

Handling Unparsable E-Mails

If E-mail Server is unable to parse an incoming e-mail, it creates a new e-mail interaction (a "wrapping message") with the following characteristics:

- The header is the same as the header of the original, unparsable e-mail.
  - If the header of the original e-mail is unparsable, the subject of the new interaction is `Unknown subject.`
  - If the From address of the original e-mail is not valid, the From address of the new interaction is `unknown@<default_domain>`, where <default_domain> is the domain specified by the `default_domain` configuration option of the E-mail Server application.
- The text of the new interaction is `Error encountered during preprocessing of this message + <reason_for_failure> + Original Incoming Email is attached to this Email.`
- The original e-mail is attached to the new e-mail.
- The new e-mail has an attached key-value pair, whose key is `_WrappingMessageReason` and whose value is a text string that describes the reason for creating the wrapping message.

Connection Information

This page documents general connection information for E-mail Server.

Connecting to a Proxy Server

Starting in release 8.5.1, E-mail Server can connect to a SOCKS or HTTP proxy server. To do this, you must create a section called **[proxy]** in the E-mail Server Application object, containing the following options:

useProxy
Default value: false
Valid values: true, false

Any value other than `true` means that no proxy will be used.

port
Default value: No default value
Valid values: Positive integer value smaller than 65535.

If the port number is absent or invalid, an `IllegalArgumentException` is thrown explaining the error.

host
Default value: No default value
Valid values: Alphanumeric string

Name or IP address of the proxy's host. Examples: `192.168.15.28`, `myProxyHost`. If a bad value is provided or E-mail Server is for some reason not able to connect to the specified host on the specified port, a `MailConnectException` is thrown. If no host is provided, an `IllegalArgumentException` is thrown.

user
Default value: No default value
Valid values: Alphanumeric string

Optional user name to authenticate on the proxy. If this option is present, **password** must also be present.

password
Default value: No default value
Valid values: Alphanumeric string

Optional password to authenticate on the proxy. If this option is present, **user** must also be present.

socksVersion
Default value: 5
Valid values: Alphanumeric string

For SOCKS proxy only: version of the SOCKS proxy being used. Only version 5 is supported. This

option is not needed for an HTTP proxy.

Connecting to Exchange Server with EWS

Starting in release 8.1.4, E-mail Server can connect to an Exchange Server running Exchange Web Services (EWS). By connecting to the corporate server using an HTTP connection, you gain flexibility in getting through the firewall, as HTTP ports are often already opened.

To do this, use the following options settings:

pop-client Section

Option Name	Setting
type	ews (new possible value in 8.1.4)
folder-path (new in 8.1.4)	(empty) The key must be present.
port	The port used for EWS. Common values are 80 for unsecured connections and 443 for secured connections
server	EWS url (see "Finding the EWS URI" below).
mailbox	User's adress, for example, JeffP@contoso.com

smtp Section

Option Name	Setting
server-type (new in 8.1.4)	ews
server	EWS url (see "Finding the EWS URI" below). For example, https://owa.example.com/ews/exchange.asmx

> **Important**
>
> The `pop-connection-security` and `smtp-connection-security` options have no effect when used with EWS. Secured or non-secured connection are used depending on the server's configuration. The connection is automatically done and negotiated using Apache's HTTP client, which handles TLS negotiation if needed.

Finding the EWS URI

Most of the time the EWS is published together with the OWA: If the OWA-URL is for example https://owa.example.com/owa, EWS is available at https://owa.example.com/ews/exchange.asmx. The EWS-URL can be tested in any browser (except Internet Explorer). The request should be forwarded to https://example.com/ews/Services.wsdl and a WSDL should be sent to the browser.

Specifying the **From** Header

When using the `Forward` object in a routing strategy, or any method that can specify a user to go in

the **From** header, the corporate e-mail server might refuse to send the e-mail. To avoid this, you can configure the corporate server to allow sending e-mails on behalf of another user.

Here is an example for Microsoft Exchange using PowerShell:

```
Add -ADPermission "Bruce Wayne" -User "gotham\selinaK" -Extendedrights "Send As"
```

This allows selinaK to send e-mails on behalf of the user "Bruce Wayne."

Customizing E-mail Server

List of Forbidden Headers

E-mail Server does not allow certain headers to be added to outbound e-mail. These excluded headers are listed in the file `com/genesyslab/icc/emailserver/ForbiddenHeaders.properties`, contained in `esj.jar`, which is normally located in `\GCTI\eServices 8.1.0\E-mail Server Java\<E-Mail Server Application name>\lib`.

You can modify this list by extracting `ForbiddenHeaders.properties` to the `<E-Mail Server Application name>` directory, then editing the content. E-mail Server will then use this modified file. If there is no such file in the `<E-Mail Server Application name>` directory, E-mail Server uses the one in `esj.jar`.

Customizing the Format of External Resource E-mails

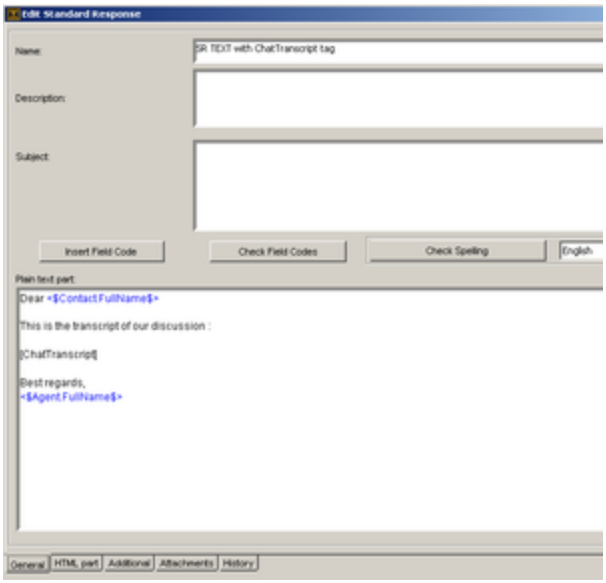
E-mails from external resources are received in plain text format. You can customize the format in which these e-mails are presented.

Chat Transcripts

Ordinarily when including the transcript of a chat session in an e-mail, the transcript is appended to the end of the e-mail. You can modify this as follows.

8.5.102.x and Earlier

Prior to release 8.5.103.x, you can place the tag `[ChatTranscript]` where you want the transcript to appear. You can do this in the plain text or the HTML version or both. The following figure shows an example (click to enlarge).



Chat Transcript Tag in Standard Response

8.5.103.x and Later

Starting with release 8.5.103.x, you can use the new tag `<Genesys_Chat_Transcript>...</Genesys_Chat_Transcript>` to indicate the position of the transcript in the email.

If you do not use the new tag, the transcript email works as in previous versions.

If you do use the new tag, the transcript displays

- The end user's local timestamp
- The chat start time, duration, and subject
- The name and size of any files transmitted by chat

Also, if you use the new tag, note the following:

- In the email, the transcript appears in place of the delimiters `<Genesys_Chat_Transcript>` and `</Genesys_Chat_Transcript>`. Within those delimiters, you can define the preferred timestamp format between the delimiters `<Timestamp>` and `</Timestamp>`, using any format defined in SimpleDateFormat (JavaSE-7). The following figures show a plain text standard response and the resulting email text (click to enlarge).

Transcript Tag in Standard Response 8.5.103

- 

Sample Plaintext Transcript Email 8.5.103

- If the timestamp format is invalid, the default timestamp format is `yyyy-MM-dd'T'HH:mm:ss`, and the error message `Invalid Timestamp format provided in the Chat Transcript` is logged (full description in next item).
- The two error codes introduced in release 8.5.103.x are as follows:
  - `30001|STANDARD|CHAT_DATE_FORMAT_PARSE_ERROR|%sInvalid Timestamp format provided in the Chat Transcript.%s`
  - `30002|STANDARD|CHAT_UNMARSHALL_ERROR|%sFailed to unmarshall the chat transcript XML.%s`. When this error occurs, the chat transcript is not sent.

Starting with release 8.5.103.x, the HTML version of the e-mails that send chat transcripts supports Cascading Style Sheets (CSS) and the configuration must include the style tag (`<style type="text/css">...</style>`). See CSS for HTML Chat Transcripts for details.

JavaMail Properties

E-mail Server uses the JavaMail API library 1.5.1. JavaMail can make use of numerous properties, which are documented at the following locations:

- Environment properties: http://www.oracle.com/technetwork/java/javamail-1-149769.pdf (Appendix A: Environment Properties)
- JavaMail Session properties: https://javamail.java.net/nonav/docs/api/
- JavaMail Session properties for IMAP: https://javamail.java.net/nonav/docs/api/com/sun/mail/imap/package-summary.html
- JavaMail Session properties for POP3: https://javamail.java.net/nonav/docs/api/com/sun/mail/pop3/package-summary.html

These properties are treated in different ways in eServices, depending on

- Whether they are set internally by E-mail Server.
- Whether they can be modified by users.

These two parameters define three different categories of property:

- Set internally and not user-modifiable
- Set internally and user-modifiable
- Not set internally and user-modifiable

The next three sections list the properties in each category and describe how to set the ones in user-modifiable categories.

Set Internally, Not User-Modifiable

- mail.pop3.class
- mail.imap.class

Set Internally, User-Modifiable

- mail.debug
- mail.pop3.host
- mail.pop3.user
- mail.pop3.port

- mail.pop3.connectiontimeout
- mail.pop3.timeout
- mail.pop3.socketFactory.class
- mail.pop3.socketFactory.fallback
- mail.pop3.socketFactory.port
- mail.pop3.socks.host
- mail.pop3.socks.port
- mail.imap.host
- mail.imap.user
- mail.imap.port
- mail.imap.connectiontimeout
- mail.imap.timeout
- mail.imap.socketFactory.class
- mail.imap.socketFactory.fallback
- mail.imap.socketFactory.port
- mail.imap.socks.host
- mail.imap.socks.port
- mail.smtp.socks.host
- mail.smtp.socks.port

You can modify these using existing configuration options in the **[pop-client]** and **[smtp-client]** sections, as shown in the following table. In this table, <protocol> is either POP3 or IMAP; for example, mail.<protocol>.timeout covers `mail.pop3.timeout` and `mail.imap.timeout`.

JavaMail Properties Controlled by Configuration Options:

JavaMail Property	Configuration Option
mail.debug	enable-debug
mail.<protocol>.connectiontimeout	connect-timeout
mail.<protocol>.timeout	protocol-timeout
mail.<protocol>.user	mailbox
mail.<protocol>.host	server
mail.<protocol>.port mail.<protocol>.socketFactory.port	port
mail.<protocol>.socketFactory.class mail.<protocol>.socketFactory.fallback	enable-ssl

See the eServices Options Reference for complete information on these options.

Not Set Internally, User-Modifiable

Any of the properties not listed in the two preceding sections can be modified by creating options in E-mail Server's `pop-client` section. The option name is the property name. For the value, see the JavaMail documentation listed above.

> **Important**
>
> Do not use this method to modify the properties, listed in the preceding section, that are controlled by configuration options.

Here is an example of adding an option to modify a JavaMail property: Some POP3 servers do not properly implement TOP, an optional POP command. This can create conflicts between the results of the TOP and RETR commands, which in turn can prevent E-mail Server from parsing the retrieved e-mail. To prevent these conflicts, you can create an option that invokes JavaMail's mail.pop3.disabletop property. The option name is `mail.pop3.disabletop`, it must be in the pop-client section, and its value must be `true`. E-mail Server then does not use TOP to retrieve messages, only RETR.

Delivery Status Notification and Message Disposition Notification

Outbound e-mails can include a request for a return message indicating whether and how the original e-mail was delivered. In Genesys eServices, you do this using the Send Email object in a routing strategy, as described in the Universal Routing 8.1 Reference Manual. The return message is of one of the following three types:

- If delivery fails: InboundNDR
- If delivery succeeds:
  - InboundReport
  - InboundDisposition

These types are represented as attribute values of the Interaction Subtype Business Attribute in Configuration Manager. E-mail Server assigns the return message to the appropriate type, and UCS stores it as a child of the outbound e-mail that contained the request.

The following sections describe each type and its contents.

InboundNDR

If one of the SMTP servers involved in the transport of the original e-mail fails to deliver it, E-mail Server submits the return message to the system with subtype InboundNDR (NDR stands for non-delivery report). There are two ways that E-mail Server can detect an inbound e-mail as an NDR:

RFC 3464

For this way, both of the following must be true:

- The e-mail conforms to RFC 3464, which means that it includes information about delivery status.
- That information indicates that delivery failed for at least one recipient.

In this case, E-mail Server submits the e-mail to the system with attached data.

Message Parts

For this way, all of the following must be true:

1. The e-mail contains either a message/rfc822 part or a message/rfc822-headers part.
2. The part referred to in (1) contains a Message-ID header.
3. One of the following must be true:

- The e-mail's From field contains one of the values specified in the E-mail Server `ndr-senders-list` option (the default is `mailer-daemon,postmaster,mmdf`).
- The `Message-ID` header referred to in (2) matches the message ID of some interaction already stored in the UCS database.

In this case, E-mail Server submits the e-mail to the system with no particular attached data. The subtype InboundNDR indicates the failure of delivery; the message itself contains no additional information.

Structured Information

When E-mail Server attaches data to the inbound interaction, it is of two kinds. The first kind, structured information, is listed in the following table.

Key	Possible Values	Description
_DSNInfo.RecipientCount	Any integer	Number of recipient addresses covered in this DSN
_DSNInfo.Recipient1.Recipient	Any string	Recipient address
_DSNInfo.Recipient1.Action	delayed delivered expanded failed relayed	Action applied for this recipient
_DSNInfo.Recipient2.Recipient	Same as _DSNInfo.Recipient1.Recipient	
_DSNInfo.Recipient2.Action	Same as _DSNInfo.Recipient1.Action	
_DSNInfo.RecipientN.Recipient	Same as _DSNInfo.Recipient1.Recipient	
_DSNInfo.RecipientN.Action	Same as _DSNInfo.Recipient1.Action	

In RecipientN in the last two rows of the preceding table, N is the value of `_DSNInfo.RecipientCount:` the number of recipients covered in this InboundReport.

> **Important**
>
> A non-delivery report may arrive even if the outbound e-mail did not request it. If it does, E-mail Server still submits it to the system with the subtype InboundNDR.

Raw Information

The second kind of information that E-mail Server attaches is raw information. That is, all information included in the reply e-mail's header is attached as key-value pairs, with the key name formed by prefixing `_DSNRawInfo` to the field name used in the reply. Some examples are:

```
_DSNRawInfo.Reporting-MTA
_DSNRawInfo.RecipientCount
_DSNRawInfo.Recipient1.Original-Recipient
_DSNRawInfo.Recipient1.Action
_DSNRawInfo.Recipient1.Status
_DSNRawInfo.Recipient1.Remote-MTA
```

`_DSNRawInfo.Recipient2.Final-Recipient`

For details, see RFC 1894.

InboundReport

You request delivery status notification (DSN) by selecting the Delivery status notification box in a Send Email routing object. The reply to this request receives the subtype InboundReport. This reply conforms with RFC 1894, and includes, as attached data:

- The structured information listed in Table 21 on page 203.

  Note that if the _DSNInfo.RecipientN.Action key has a value of failed, E-mail Server assigns the reply the subtype InboundNDR, not InboundReport.

- The raw information contained in the keys whose names start with _DSNRawInfo, as described in the previous section.

> **Important**
>
> Depending on the implementation of the SMTP servers involved, there are the following possibilities:
>
> - Individual DSN messages can be generated, each one related to one or several recipient addresses.
> - A single DSN message can be generated, related to one or several recipient addresses.

InboundDisposition

You request message disposition notification (MDN, also called read receipt) by selecting the `Message disposition notification` box in a `Send Email` routing object. The reply to this request receives the subtype `InboundDisposition`. This reply conforms with RFC 3798, and includes, as attached data, the information listed in the following table.

Key	Possible Values	Description
_MDNInfo.ActionMode	manual-action automatic-action	Mode of the action applied to the e-mail
_MDNInfo.DispositionType	displayed deleted	What was done with the e-mail
_MDNInfo.SendingMode	MDN-sent-manually MDN-sent-automatically	How the message disposition notification is being sent
_MDNInfo.Recipient	Any string	Recipient address covered by this

Key	Possible Values	Description
		message disposition notification

This reply is sent as long as all of the following conditions are met:

- It was requested in the outbound e-mail. This is independent of whether a delivery status notification was also requested.
- Delivery succeeded.
- Either of the following:
  - The recipient agreed to send the read receipt.
  - The recipient mailer was configured to automatically send read receipts.

In addition to the structured information listed in the preceding table, all information included in the reply e-mail's header is attached as key-value pairs, with the key name formed by prefixing `_MDNRawInfo` to the field name used in the reply. Some examples are:

- `_MDNRawInfo.Disposition`
- _MDNRawInfo.Final-Recipient
- _MDNRawInfo.Original-Message-ID
- _MDNRawInfo.Reporting-UA

For details, see RFC 3798.

Customizing the Format of External Resource E-mails

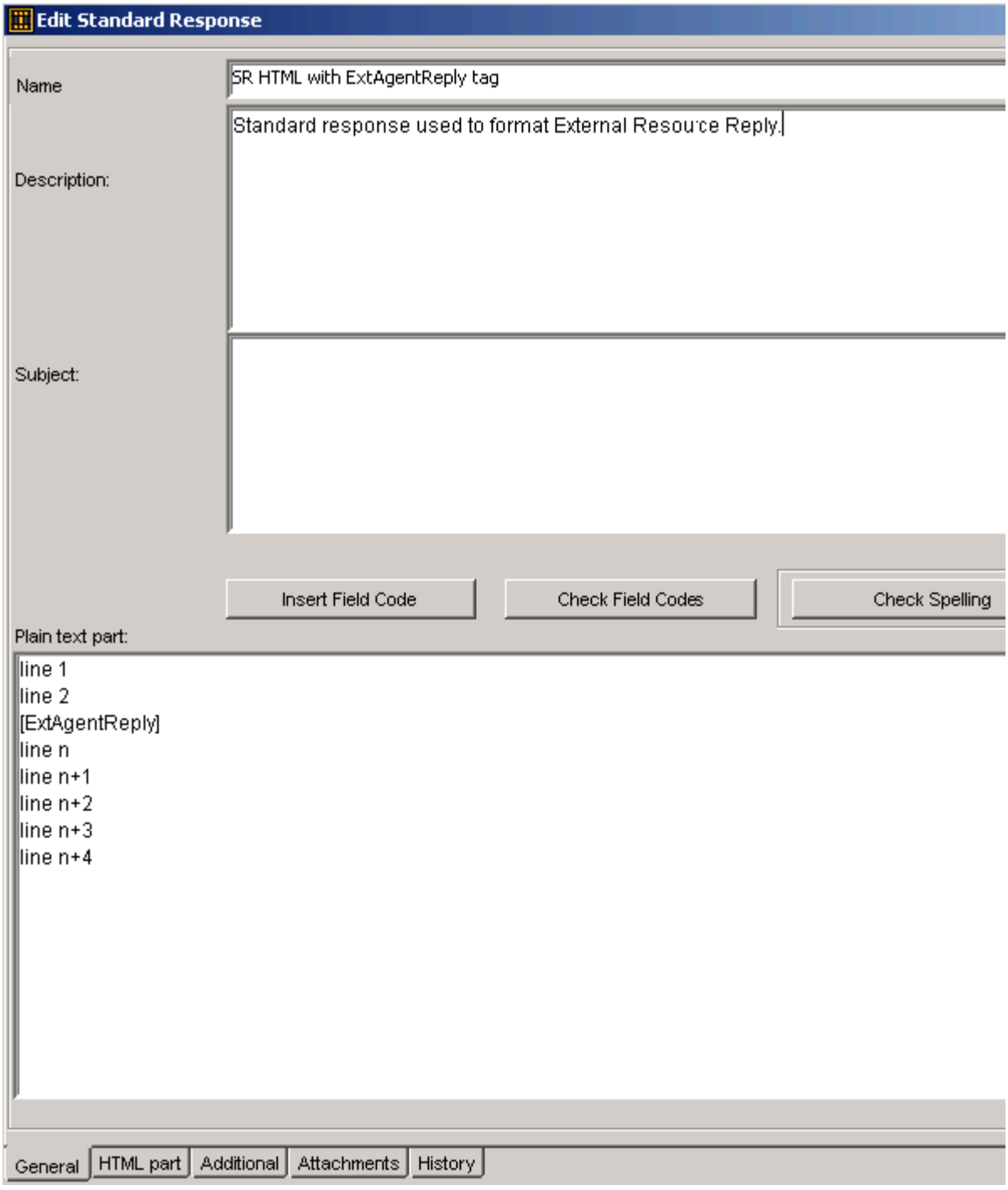
E-mails from external resources are received in plain text format. To customize the format in which these e-mails are presented, you must:

Create New Standard Response

Create a New Standard Response in Knowledge Manager

This standard response will be used to customize e-mails built by the `ResponseFromExtResource` service.

1. Create a new standard response in Knowledge Manager containing the tag [ExtAgentReply]. The following figure shows a sample plain-text version.



Standard Response for External Reply: General Tab

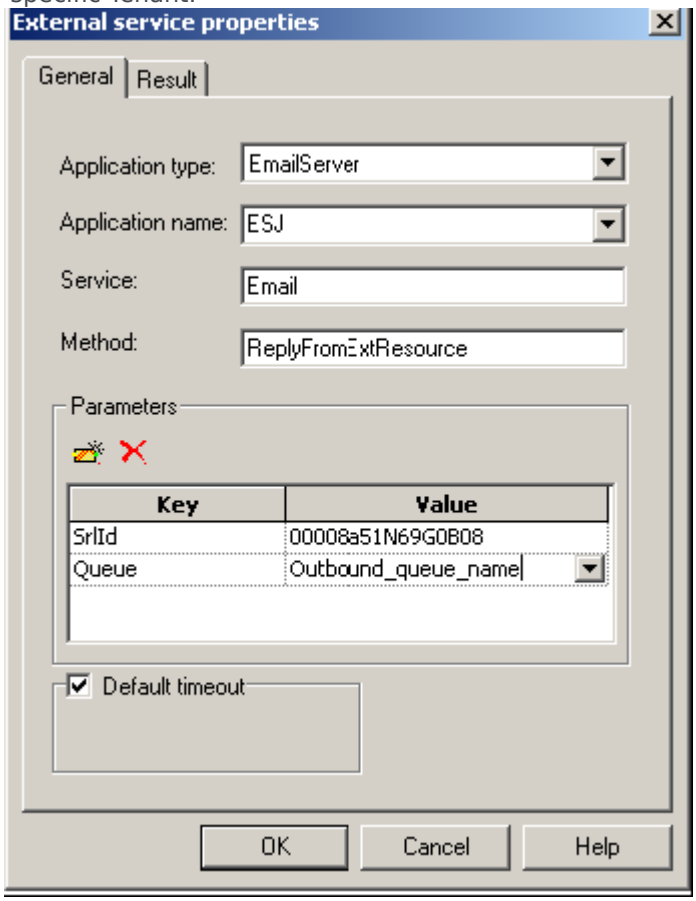
2. Optionally, use the HTML tab to create an HTML version.

Update the Strategy

Update the Strategy that Sends the Standard Response

You must add the standard response ID to the strategy. This provides the strategy with the standard response you wish to use for responses from external resources.

1. Log in to Interaction Routing Designer and open your strategy for editing.
2. In the strategy, create an `External Service` block to call the E-mail Server's `CreateReplyFromExtResource` service. See the figure below for an example.
3. On the `General` tab, enter the following values:
   `Application type: EmailServer`
   `Application name:` <name of your E-mail Server application>
   `Service: Email`
   `Method: ReplyFromExtResource`
4. In the `Parameters` section add two entries: `SrlId` and `Queue`.
5. For the `SrlId` parameter, the value is the ID of the standard response you created for external resource responses. You will locate the ID in the next procedure (Find the Standard Response ID).
6. For the `Queue` parameter, the value is the name of the queue upon which the strategy to send e-mails is loaded. The name of this queue can be found in Configuration Manager in the `Scripts` folder under the specific Tenant.



External Service Block Properties

> **Important**
>
> Additional optional parameters can be added. Refer to the Reply Email From External Resource block description in the Universal Routing 8.1 Reference Manual for more information.

7. Save and reload the modified strategy.

Find the Standard Response ID

Locate the standard response ID

This ID is needed to update the strategy to allow for the customization of e-mails built from the ReplyFromExtResource service.

1. In Configuration Manager, locate the standard response created in the Create New Standard Response procedure. Under your tenant, navigate to `Resources -> Business Attributes -> Category Structure-> Attribute Values.`
2. Locate your standard response and open it. The standard response ID appears in the `Name` field but it is not possible to copy it from here.
3. Click the `Annex` tab.
4. In the `General` section double-click the `Id` option.
5. The option value is the standard response ID. Copy the option value.

CSS for HTML Chat Transcripts

Starting with release 8.5.103, the HTML version of the e-mails that send chat transcripts can include CSS information.

> **Important**
>
> Tags <Genesys_Chat_Transcript>...</Genesys_Chat_Transcript>, <Timestamp>...</Timestamp>, and <style type="text/css">...</style> are mandatory for CSS support.

The following figure shows an example of the output.

Sample HTML Transcript Email

And following is the code that produced the example. You can use it as a template for the text of an HTML standard response.

```
Dear <$ Contact.FirstName+" "+Contact.LastName $>.

Please find below the transcript of the chat session that you initiated on <$
Interaction.DateCreated $>.
Note: If there are problems on how this message is displayed, please view it in a web browser.


Sincerely,
Customer Support
---

<Genesys_Chat_Transcript><Timestamp>yyyy/MM/dd, hh:mm:ss
aaa</Timestamp></Genesys_Chat_Transcript>

<style type="text/css">
```

```
#outlook a { padding: 0; } /* Force Outlook to provide a "view in browser" menu link. */
.transcript {
font-family:arial, helvetica, sans-serif;
overflow-y: auto;
font-size: 13px;
line-height: 18px;
padding: 24px;
}
/* IF YOU WANT TO USE DARK THEME UN COMMENT BELOW 6 LINES
.transcript.cx-theme{background-color: #33383d}
.transcript.cx-theme .message-text {color:#fdfdfd}
.transcript.cx-theme .bubble {background-color: #222529}
.transcript.cx-theme .bubble-arrow {fill:#222529}
.transcript.cx-theme .name {color:#fdfdfd}
.transcript.cx-theme .message .time {color: #aaa}
*/
/* IF YOU WANT TO USE DARK THEME COMMENT BELOW 6 LINES AND UNCOMMENT ABOVE 6 LINES */
.transcript.cx-theme{background-color:#fdfdfd}
.transcript.cx-theme .message {border-bottom:1px solid #eee}
.transcript.cx-theme .message-text {color:#000}
.transcript.cx-theme .bubble {background-color:#F2F4F7}
.transcript.cx-theme .bubble-arrow {fill:#F2F4F7}
.transcript.cx-theme .name {color:#222}
.transcript.cx-theme .message .time {color:#aaa}
.transcript .notice a {
color: #75A8FF!important;
}
.transcript .message {
min-height: 48px;
margin: 5px 0;
box-sizing: border-box;
position: relative;
}
.transcript .message .bubble {
padding: 10px 15px;
padding: 8px;
}
.transcript .message .bubble-arrow {
position: absolute;
top: 16px;
}
.transcript .message .name {
font-size: 13px;
font-weight: bolder;
padding-right: 5px;
}
.transcript .message .time {
display: block;
margin-top: 5px;
}
.transcript .message .message-text {
word-wrap: break-word;
white-space: pre-line;
}
@-moz-document url-prefix() {
white-space: pre-wrap;
}
.transcript .message .message-text > p {
margin: 10px 0 0 0;
}
.transcript .message .avatar-wrapper {
width: 48px;
position: absolute;
```

```
top: 2px;
}
.transcript .message .avatar {
height: 48px;
width: 48px;
}
.transcript .message > p.NewTextBubble {
-webkit-animation: NewTextBubble 0.5s 1;
animation: NewTextBubble 0.5s 1;
}
.transcript .message.system {
color: black;
background: none;
padding: 8px 0;
}
.transcript .message.system .bubble {
padding: 0;
background-color: transparent;
}
.transcript .message.system .message-text {
font-style: italic;
}
.transcript .message.system .name {
display: none;
}
.transcript .message.system .avatar-wrapper {
display: none;
}
.transcript .message.system .time {
color: #ccc;
}
.transcript .message.system .bubble-arrow {
display: none;
}
.transcript .message.them {
border: 0;
padding: 6px 0;
text-align: left;
}
.transcript .message.them .bubble {
margin-left: 53px;
text-align: left;
}
.transcript .message.them .bubble-arrow {
left: 46px;
}
.transcript .message.them .bubble-arrow .right {
display: none;
}
.transcript .message.them .avatar-wrapper {
left: 0;
}
.transcript .message.you {
border: 0;
padding: 6px 0;
text-align: right;
}
.transcript .message.you .bubble {
margin-right: 53px;
text-align: left;
}
.transcript .message.you .bubble-arrow {
right: 46px;
```

```
}
.transcript .message.you .bubble-arrow .left {
display: none;
}
.transcript .message.you .avatar-wrapper {
right: 0;
}
.transcript .message.injected {
border: 0;
margin: 6px 0;
padding: 8px;
}
.transcript .message.injected .bubble,
.transcript .message.injected .bubble-arrow,
.transcript .message.injected .avatar-wrapper {
display: none;
}
.cx-img-map {
background-repeat: no-repeat;
}
.preset-blue .cx-img-map,.cx-img-map.preset-blue {
background-image: url("<your-url>");
}
.px48 .cx-img-map,.px48.cx-img-map{
height: 48px;
width: 48px;
}
.px48 .cx-img-map.avatar-customer,.cx-img-map.px48.avatar-customer{
background-position: -160px 0px;
}
.px48 .cx-img-map.avatar-agent,.cx-img-map.px48.avatar-agent{
background-position: -160px -144px;
}
</style>
```

Setting up Microsoft Azure mailboxes for OAuth 2.0 authorization

Starting with version 8.5.107.06, E-mail Server supports the OAuth 2.0 authorization access to Microsoft Exchange Online API for Office 365 with the IMAP and EWS protocols. Starting with version 8.5.202.02, OAuth 2.0 support is extended to POP3 and SMTP protocols.

Starting with version 8.5.205.03, OAuth 2.0 support is extended to the Graph API with client secret only.

To set up Microsoft Azure mailboxes using the OAuth 2.0 authorization access:

1. Create a Microsoft Azure application.
2. Configure Genesys E-mail Server.
3. Configure Proxy.

Creating a Microsoft Azure application

Starting with version 8.5.205.03, E-Mail Server can support the Graph API with Client Credentials Grant Flow.

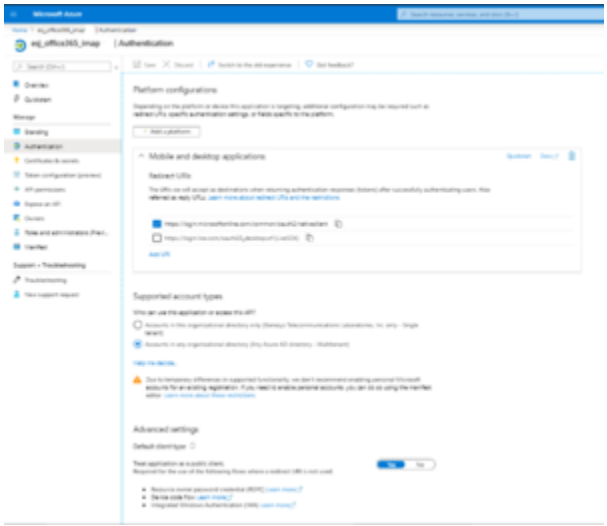
OAuth defines the following grant types: authorization code, implicit, resource owner password credentials, and client credentials. The Genesys solution uses resource owner password credentials.

1. Follow Steps 1-8 as described in this documentation to register an Azure public client application for the mailbox(es) that will be accessed by Genesys E-mail Server. Note that a single Azure application can support all mailboxes for the same company.
   Note: The Azure client application does not need to be public if it is for Graph.
2. In step 6, for entering the name in Supported account types:
   - Select **Accounts in any organizational directory (Any Azure AD directory – Multitenant)**.
     Note: Single-tenant accounts, **Accounts in this organizational directory only**, are also supported.
   - Leave the **Redirect URI** empty (as well as in Step 7).
3. Copy and paste the Application (client) ID and the Directory (tenant) ID into a text document for insertion during the E-mail Server configuration.

After the application is created, it should look similar to this (click to expand it):

Where **esj_office365_imap** is the Azure application name.

If you open the **Supported account types** and **Redirect URIs**, it should look similar to this:



The **Application ID URI** should be empty:

Adding Application Permissions

Read through the article Permissions and consent in the Microsoft identity platform endpoint to learn about permissions and consent.

Graph

Set the application permissions for graph as given in this document Configure permissions for Microsoft Graph. Ensure that the application has the following application permissions with an admin consent grant:

(Click to expand)

We recommend to read through this document Limiting application permissions to specific Exchange Online mailboxes to learn how to limit app access to specific mailboxes.

Legacy protocols

For other protocols (EWS, IMAP, POP3, and SMTP), you might want to consult this document, Configure permissions for Microsoft Graph, although it focuses on getting permissions for the Graph API and you must configure application permissions for Office 365 Exchange Online.

For IMAP, POP3, and EWS, the application must have the following permissions granted by the company's administrator, depending on the email protocols used:

(Click to expand)

Since early 2020, Microsoft stopped exposing API permissions for IMAP, POP3, and SMTP under Office 365 Exchange Online, although it still supports those protocols and continues exposing the **User.*** and **EWS.*** permissions. Based on the article, Azure Active Directory app manifest, users can edit the Azure App manifest to configure an app's attribute.

Users can manually edit the `requiredResourceAccess` attribute by adding **IMAP.AccessAsUser.All**, **POP.AccessAsUser.All**, and **SMTP.Send**. The following example shows the related part (`requiredResourceAccess`) in the manifest of an Azure app. The app has all 5 permissions. As Microsoft continues to expose the **User.*** and **EWS.*** permissions under Office 365 Exchange Online, users can add these two permissions in the Azure UI. The `requiredResourceAccess` attribute contains two items. Then, the user can manually add the items that represent **IMAP.AccessAsUser.All**, **POP.AccessAsUser.All**, and **SMTP.Send** under the same `resourceAccess` of the same `resourceAppId`. The ID values for **IMAP.AccessAsUser.All** and other protocols could be different for different apps. The Exchange Admin must find it out.

(Click to expand)

Important

- For the EWS protocol, the Azure application must have the following permissions:
  User.Read

  EWS.AccessAsUser.All
- For the IMAP protocol, the Azure application must have the following permissions:
  User.Read

  IMAP.AccessAsUser.All
- For the POP3 protocol, the Azure application must have the following permissions:
  User.Read

  POP3.AccessAsUser.All
- For the SMTP protocol, the Azure application must have the following permissions:
  SMTP.Send

Setting up a mailbox

The mailboxes has the following special settings in the company's system:

- Multifactor authentication is disabled on the mailboxes for the legacy protocols (POP3, IMAP, SMTP, and EWS).
- The IMAP protocol is enabled (if IMAP is used).
- The POP3 protocol is enabled (if POP3 is used).
- Nothing is needed for the EWS protocol.

Configuring Genesys E-mail Server

To configure E-mail Server:

- Set the JavaMail property **mail.<type>.auth.mechanisms** (where *<type>* can be **ews**, **imap**, **pop3**, or **smtp**) to XOAUTH2. (To disable OAuth 2.0, remove the JavaMail property.)

> **Important**
>
> The Graph protocol only supports oAuth2.

- Add this JavaMail property for a Microsoft Office 365/Outlook POP3 mailbox pop-client:

```
mail.pop3.auth.xoauth2.two.line.authentication.format=true
```

- Configure options in the **smtp-client** section.
- Configure options in the **pop-client** section.

Configuring the **smtp-client** section

Configure the following configuration options:

- directory-id—Specify the Directory (tenant) ID of the registered Microsoft Azure application for the Office 365 mailbox.
- tenant-authority—Specify the authority server of the registered Microsoft Azure application for the Office 365 mailbox. For Office 365, the default configuration value is `https://login.microsoftonline.com/`.
- client-id—Specify the Client ID of the registered Microsoft Azure application for the Office 365 mailbox.
- scope—Specify the access token scope. For Office 365, the default configuration value is:
  - `https://outlook.office.com/.default` for the legacy protocols (POP3, IMAP, SMTP, and, EWS).
  - `https://graph.microsoft.com/.default` for the GRAPH protocol.
- token-expiry-margin-time— Specify the amount of time an SMTP connection for the EWS/SMTP type remains connected before its access token expires and the server closes the connection.
- password— If the type is Graph, this password is used for the Azure application secret.

Configuring the **pop-client** section

Configure the following configuration options:

- directory-id—Specify the Directory (tenant) ID of the registered Microsoft Azure application for the Office 365 mailbox in the corresponding POP client.
- tenant-authority—Specify the authority server of the registered Microsoft Azure application for the Office 365 mailbox in the corresponding POP client. For Office 365, the default configuration value is `https://login.microsoftonline.com/`.
- client-id—Specify the Client ID of the registered Microsoft Azure application for the Office 365 mailbox in the corresponding POP client.

- scope—Specify the access token scope. For Office 365, the default configuration value is:
  - `https://outlook.office.com/.default` for the legacy protocols (POP3, IMAP, SMTP, and, EWS).
  - `https://graph.microsoft.com/.default` for the GRAPH protocol.
- password— If the type is Graph, this password is used for the Azure application secret.

Configuring Proxy

E-mail Server implements Microsoft identity platform and OAuth 2.0 Resource Owner Password Credentials. It sends the client identification and user's credentials to the Microsoft Identity Platform (IDP) to request an access token. If a proxy has been used to access Office 365, you must have additional access to the Microsoft IDP (`https://login.microsoftonline.com/`, default https port is 443) and keep the same credentials if they were created previously.

To comply with RFC 6749, the Microsoft IDP validates the resource owner credentials by accessing the resource owner.

Setting up Gmail mailboxes for OAuth 2.0 authorization

For basic authentication, Genesys E-mail Server can access Gmail mailboxes using the IMAP, POP3, and SMTP protocols.

Starting with version 8.5.201.05, E-mail Server supports the OAuth 2.0 authorization access to Gmail mailboxes with the IMAP and SMTP protocols. Starting with version 8.5.202.02, OAuth 2.0 support is extended to POP3 and SMTP protocols.

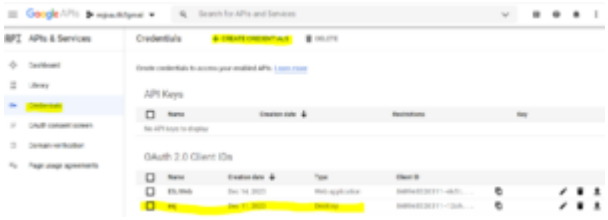
To set up Gmail mailboxes using the OAuth 2.0 authorization access:

1. Create a Google application.
2. Configure Genesys E-mail Server.

Creating a Google application

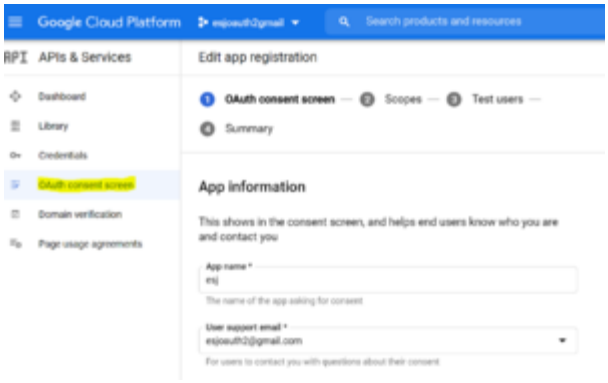
For OAuth 2.0 authorization access to Gmail mailboxes with IMAP, POP3, and SMTP protocols, create a Google application in the Google platform. (In our example, the **esj** account is created.)

1. Follow this Google documentation to configure the application. The main configuration points are included in this procedure.
2. Select **Desktop** as an application type. E-mail Server uses "Manual copy/paste" as the redirect method.



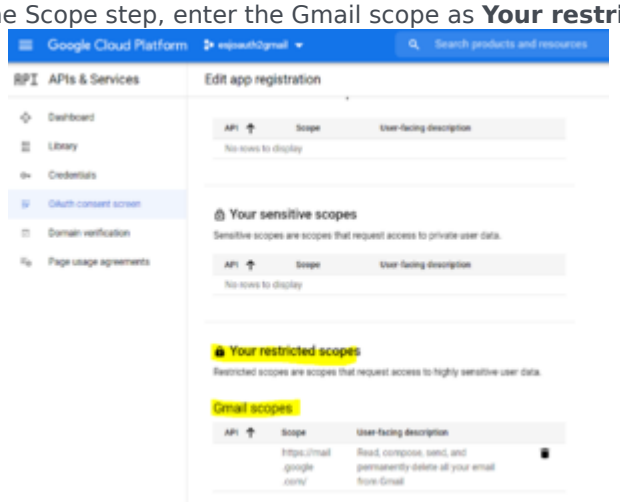
(Click to expand)

3. Download the **client_id** and **client_secret** by clicking the Download arrow of your Google application. These values are required to get the access token and to configure Genesys E-mail Server.
4. On the OAuth consent screen, add your **App information** (App name, User support email). For example:



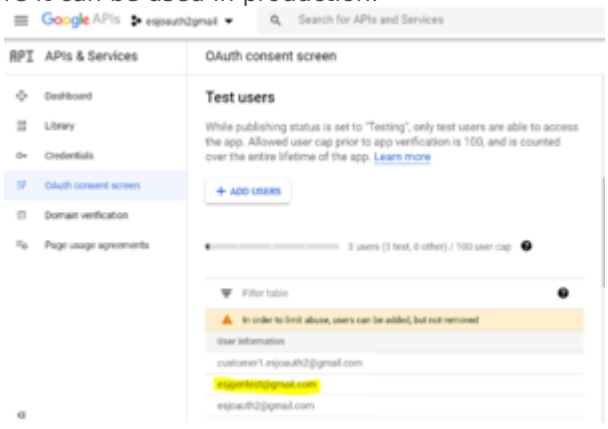
(Click to expand)

5. In the Scope step, enter the Gmail scope as **Your restricted scopes**.



(Click to expand)

6. (Optional) Add test users. This step is for the testing phase. You can add an existing or new mailbox that the E-mail Server can access as a user. The application must be published after the testing phase before it can be used in production.



(Click to expand)

7. Get a refresh token manually. Follow the steps as described in this Google documentation to get the OAuth 2.0 refresh token:
   - **Step 1**: Generate a code verifier and challenge. (Note: A refresh token can be acquired without this

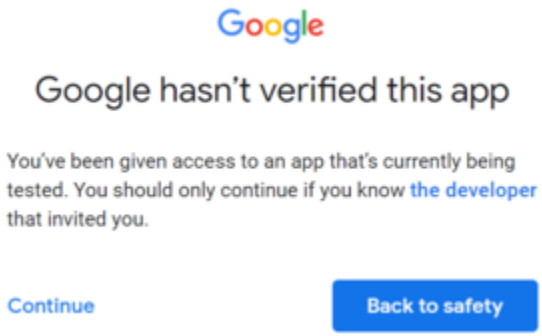
step.)

- **Step 2**: Send a request to Google's OAuth 2.0 server. In a web browser, enter the following as a URL, replacing *<your client id>* with your application client ID:

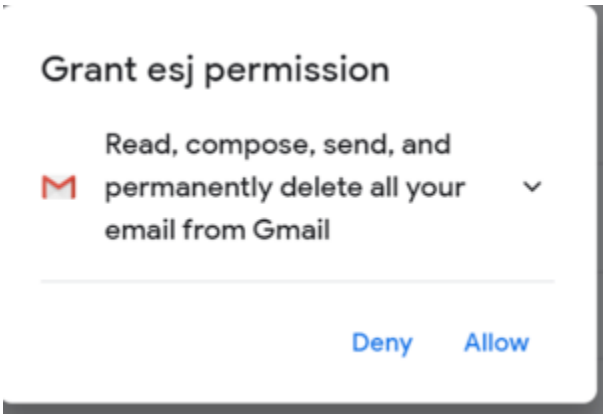
  ```
  https://accounts.google.com/o/oauth2/
  auth?scope=https://mail.google.com/&redirect_uri=urn:ietf:wg:oauth:2.0:oob&response_type=code&clien
  client id>
  ```

  Note that **redirect_uri** and **response_type** values cannot be changed.

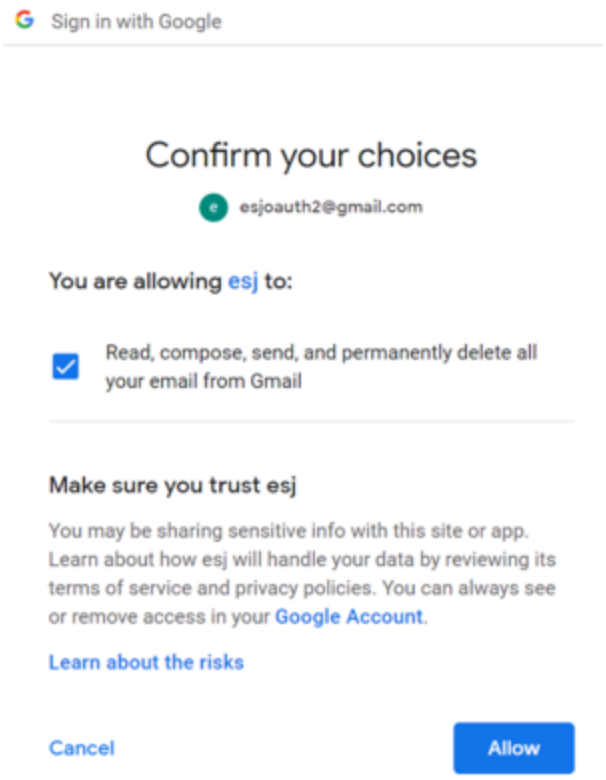
- **Step 3**: Google prompts the user for consent. You will be prompted to sign in with a mailbox if it was not signed in. In the test phase, if you have multiple mailboxes, only the mailboxes that have been added as Test Users can access the application. This may change after the application is published. After signing in with mailbox credentials, there will be an alert in the test phase. Click **Continue**:



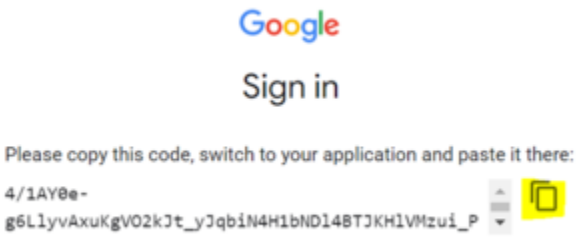
Click **Allow**:



Click **Allow**:



The Authorization Code is displayed. Copy the Code by clicking **Copy Icon**:



- **Step 4**: Exchange authorization code for refresh and access tokens.
  The Authorization Code acquired in Step 3 can be used to exchange for OAuth 2.0 access and refresh tokens within 10 minutes (after you received the authorization code) by means of the following command:

  ```
  curl -d "code=<your authorization
  code>&grant_type=authorization_code&redirect_uri=urn:ietf:wg:oauth:2.0:oob&client_id=<your
  client_id>&client_secret=<your client secret>" -X POST
  https://oauth2.googleapis.com/token
  ```

  Replace *<your authorization code>*, *<your client_id>*, and *<your client_secret>* with the actual values of your application. Keep the rest as is.

  Here is an example of the response:

  ```
  {
  ```

"access_token": "ya29.a0AfH6S7ztJBShUiXC-z7P_-**********",

"expires_in": 3599,

"refresh_token": "1//06LxPX1ZCgYIARAAGAYSNwF**********",

"scope": "https://mail.google.com/",

"token_type": "Bearer"

}

The Bearer **access_token**, which can be used to access the mailbox in IMAP/POP3/SMTP, expires every 3600 seconds. The **refresh_token** can be used to get a new Bearer token. After the application is published, the refresh code will only expire under the conditions listed in this Google document.

Configuring Genesys E-mail Server

To configure E-mail Server:

- Set the JavaMail property **mail.<type>.auth.mechanisms** (where *<type>* can be **imap**, **pop3**, and **smtp**) to X0AUTH2. (To disable OAuth 2.0, remove the JavaMail property.)
- Configure options in the **smtp-client** section.
- Configure options in the **pop-client** section.
- Configure options in the **secret-<*secretName*>** section.

Configuring the **smtp-client** section

Configure the following configuration options:

- client-id—Specify the Client ID of the Google application.
- password—Specify the refresh_token of the SMTP account.
- secret—Specify the secretName of the secret-*<secretName>* section to associate with the Google application secret.
- tenant-authority—Specify the valid Google authority server, which is a case-insensitive string that contains "google".

Configuring the **pop-client** section

Configure the following configuration options:

- client-id—Specify the Client ID of the Google application.
- password—Specify the refresh_token of the Gmail mailbox.

- secret—Specify the secretName of the secret-<*secretName*> section to associate with the Google application secret.
- tenant-authority—Specify the valid Google authority server, which is a case-insensitive string that contains "google".

Configuring the **secret-<*secretName*>** section

- password—Specify the client_secret value of the registered Google account.

Limitations

- During the test phase, a refresh token expires in 7 days.
- During the test phase, the refresh token may stop working if the Gmail mailbox is not used for a few days.

Splitting To and CC email recipients

Starting with version 8.5.108.00, E-mail Server supports splitting the **To** and **CC** email recipient list and generating an interaction (an individual email message) for each unique recipient in the list. When a POP client processes an email to the mailbox, E-mail Server copies the same email message into multiple copies based on the recipient list. As a result, one email can generate multiple interactions. All of the copied email messages will have the same message body and headers.

This feature is enabled by setting the copy-message-to-recipients configuration option to `true`. When this feature is disabled (**copy-message-to-recipients** is set to `false`, by default), E-mail Server creates only one interaction per email message and submits that interaction.

Additional configuration options are added to this feature:

- copy-message-with-header - (Optional) to specify the name of the header to be added to the InteractionAttributes for each copy of the inbound email made when the **copy-message-to-recipients** option is set to `true`.
- inbound-msg-copy-limit - (Optional) to specify the maximum number of interactions that can be created by copying the same inbound email when the **copy-message-to-recipients** option is set to `true`.

Configure shared mailbox for Microsoft Exchange Online

Overview

ESJ allows to use Microsoft Exchange Online shared mailboxes for the following protocols:

- Inbound: IMAP, EWS, GRAPH
- Outbound: SMTP, EWS, GRAPH

Configuration specifics for ESJ

Inbound e-mails

- **IMAP**, **EWS** configuration for shared mailboxes:
  - for **pop-client/mailbox** option, specify the shared mailbox e-mail address.
  - for **pop-client/service-account** option, specify the e-mail address with delegated *Full Access* to the shared mailbox.
- **GRAPH** configuration for shared mailboxes:
  - for **pop-client/mailbox** option, specify the shared mailbox e-mail address.
  - **pop-client/service-account** option, specify an empty string (only client credential grant is supported for GRAPH).

If Auto Acknowledge or Auto Response functionality is planned to be used, for **pop-client** options, for the **address** and **default-from-address** options, specify the the same value as in **mailbox** option.

Outbound e-mails

- Configuration for shared mailboxes for all supported protocols (IMAP, EWS, GRAPH):
  - for **smtp-client/user** option, specify e-mail address with delegated *Full Access* to the shared mailbox.
- Additionally, **GRAPH** configuration for shared mailboxes:
  - **smtp-client/allow-delegate** option must be set to `true`.

The *FROM* address in outbound emails must have the value of shared mailbox. The *FROM* address for outbound emails can be assigned either by agent (using the configuration of Business Attributes E-mail Accounts) or by a strategy in *Send* block.

Configuration specifics for Microsoft Exchange Online

The following configuration must be provided:

1. For shared mailbox go to the **Delegation** tab, open **Read and manage (Full Access)**.
2. Add the user that are to be used in **pop-client/service-account** and **smtp-client/user**.