

# **HTTP(S) Mailbox**

June 2004  
Product Revision Level 1.0.2



## Contents

---

Contents .....	2
<b>Introducing HTTP(S) Mailbox.....</b>	<b>3</b>
Overview – HTTP(S) Mailbox.....	3
How to perform Initial Setup – HTTP(S) Mailbox .....	4
How to conduct a Loopback Test – HTTP(S) Mailbox.....	5
Access from a Browser – HTTP(S) Mailbox .....	6
Access from a Programmatic Client – HTTP(S) Mailbox .....	10
<b>Appendix – HTTP(S) Mailbox.....</b>	<b>15</b>
Common Errors – HTTP(S) Mailbox .....	15
Glossary – HTTP(S) Mailbox.....	16

# INTRODUCING HTTP(S) MAILBOX

This guide contains information needed to use the HTTP(S) Mailbox on the Covisint Connect messaging hub. With the HTTP(S) Mailbox described in this guide, you can use a Web browser or programmatic HTTP(S) client to exchange information with your trading partners through the Covisint messaging hub. To use the HTTP(S) Mailbox, you should have prior knowledge of the HTTP(S) protocol, Web servers and TCP/IP networking basics. The loopback test implies the basic knowledge of EDI document standards. The intended audience for this document includes integration developers, network engineers and customer connection support staff from Covisint Connect trading partners and third-party organizations. Please contact [connectsupport@covisint.com](mailto:connectsupport@covisint.com) with any issues.

## Overview – HTTP(S) Mailbox

---

The HTTP(S) Mailbox is a component of the HTTP(S) Connector. It allows trading partners with standard HTTP(S) client software or a Web browser to perform the following tasks:

- Connect to the Covisint Connect messaging hub using unidirectional (to Covisint) TCP/IP link over the public Internet, VPN tunnel or ANX/ENX networks.
- Submit messages to be processed by the Covisint messaging hub.
- Retrieve messages from their Covisint Connect Mailbox.
- Perform secure communications over SSL (HTTPS), if required.

On the inbound flow to Covisint, the HTTP(S) Mailbox accepts the client posts, extracts the message data and routing information and sends it to the Covisint's Document Recognition Service. The HTTP(S) Mailbox acknowledges all successful posts by sending a tracking ID back to the client.

On the outbound flow from Covisint, the HTTP(S) Mailbox is able to return the mailbox directory and any of the available messages queued on the mailbox database table. All outbound data is returned as the body of the HTTP(S) response. That is why, from a network perspective, the HTTP(S) mailbox requires an inbound-only TCP/IP connection - from the trading partner to Covisint Connect.

## How to perform Initial Setup – HTTP(S) Mailbox

### Obtain a User ID and Password from Covisint

Please contact [connectsupport@covisint.com](mailto:connectsupport@covisint.com) to get your user ID and password. Your trading partner profile and relationships will be established when you register with Covisint.

### Determine Type and Attributes Of Network Connection

Before connecting to the HTTP(S) Mailbox, some necessary communication requirements must be satisfied. The following table lists important HTTP(S) Mailbox connectivity information:

Network	IP Address	Port
Public Internet	64.37.249.63 (messaging.covisint.com)	443
ANX	206.18.241.63	443
ENX	To be determined, contact <a href="mailto:connectsupport@covisint.com">connectsupport@covisint.com</a>	TBD
VPN	To be determined, contact <a href="mailto:connectsupport@covisint.com">connectsupport@covisint.com</a>	TBD

When using the public Internet, secure HTTP over SSL (HTTPS) is a mandatory requirement. For a highly secure VPN connection, plain HTTP should be used since double encryption reduces communication channel performance. When using secure ANX or ENX networks, you may request HTTPS for additional connection-level security, though it is not mandated.

### Testing Credentials and Connection

The easiest way to test your connection is to launch a Web browser and attempt to get your HTTP(S) Mailbox directory:

- 1 Open the browser and enter **GET** service URL (see section **HTTP(S) Mailbox: Access From a Browser** for details).
- 2 Supply your valid credentials when prompted.
- 3 The Mailbox directory should appear on the browser screen, as shown in **Figure 3**.

# How to conduct a Loopback Test – HTTP(S) Mailbox

The loopback test is a process of sending a test document to and from yourself. There are two ways to execute this task.

### **Test with the Roundtrip Document**

- 1 Prepare a valid EDI business document with your ID as both sender and receiver.

**Note:**

**For the clarity of the test, the business document should be an atomic EDI transaction with a single ISA, GS and ST segment.**

- 2 Save this document to the local file.
- 3 Open the browser and enter post service URL (see **HTTP(S) Mailbox: Access From a Browser** for details).
- 4 Supply your valid credentials when asked.
- 5 Submit the prepared file to the HTTP Mailbox.
- 6 Allow several seconds for message processing.
- 7 Enter **GET** service URL into the browser (see **HTTP(S) Mailbox: Access From a Browser** for details).
- 8 Supply your valid credentials when asked.

**Note:**

**The Mailbox directory should appear on the browser screen, as shown in Figure 3.**

- 9 Click the **GET** link on the message entry to save the content of the message to the local file.
- 10 Run a binary comparison utility against the initial and the loopback files.

### **Test Using Explicit Roundtrip Routing Data**

- 1 Take any valid EDI business document and save it to the file.

**Note:**

**For the clarity of the test, the business document should be an atomic EDI transaction with a single ISA, GS and ST segment.**

- 2 Open the browser and enter **post** service URL (see **HTTP(S) Mailbox: Access From a Browser** for details).
- 3 Supply your valid credentials when prompted.
- 4 Submit the prepared file to the HTTP(S) Mailbox. Enter **From ID, From ID Qualifier, To ID, To ID Qualifier, Message Type, and Message Format** into the corresponding optional fields of the **post** form.

**Note:**

**Use your partner ID for both From ID and To ID fields. Use your EDI Interchange ID Qualifier, for both From ID Qualifier and To ID Qualifier fields. Enter Message type as a three-digit numeric value that identifies EDI Transaction Set Control Number (transaction type), e.g. 830. Enter Message Format as a predefined string value. For the regular EDI document, that value would be X12.**

- 5 Please follow steps 5-10 from the **Test with the Roundtrip Document** above.

The loopback test is successful when you get the initial document back unchanged.

## Access from a Browser – HTTP(S) Mailbox

### Post Message from a Web Browser

- 1 Open a Web browser and enter the following URL into the address box:  
***HTTP(S)://host:port/invoke/HTTPConnector.Mailbox/post***
- 2 Enter your username and password if prompted.
- 3 The browser should display the HTTP(S) Mailbox Receptor form, as shown in **Figure 1**.

**Note:**

This form allows you to pick the message payload from a local file and specify the routing data explicitly (optional).

Figure 1: HTTP(S) Mailbox Receptor Form

- 4 Enter the full path to the file into the **File to post** input box or click the **[Browse...]** button to find the file.
- 5 Enter optional routing data into the text boxes **From ID, From ID Qualifier, To ID, To ID Qualifier, Message type, Message Format, Standard Version, Customer Tracking Number**. Use the syntax described in the paragraph **How To Conduct the Loopback Test** (second method).

**Note:**

The HTTP(S) Mailbox required From ID, To ID, Message Type and Message Format to turn on explicit routing. The rest of the parameters are optional. Please contact [connectsupport@covisint.com](mailto:connectsupport@covisint.com) to determine the set of possible predefined values for the above-mentioned parameters.

- 6 Click the **[Post file to mailbox]** button.
- 7 Wait for the message to be passed to the HTTP(S) Mailbox (usually from several seconds to several minutes).

**Note:**

The HTTP(S) mailbox has reasonable message size limitations. It is a good practice not to post messages larger than 10 megabytes. Please use the FTP Mailbox to transfer larger messages. If there is a requirement to pass messages larger than 10 megabytes over HTTP(S) Mailbox, then use message chunking. Please contact [connectsupport@covisint.com](mailto:connectsupport@covisint.com) to do this.

- 8 The HTTP(S) Mailbox post service parses the form data and routes the message. The HTTP(S) Mailbox Receptor Web form is returned back to the browser for further posts. This time it contains a table that has the detailed information about recent posts, see [Figure 2](#).

### HTTP(S) Mailbox Receptor

Username: widget

File to post.....

\_\_\_\_\_ Optional Explicit Routing \_\_\_\_\_

From id (required).....

From id qualifier.....

To id (required).....

To id qualifier.....

Message type (required).....

Message format (required)....

Standard version .....

Customer tracking number.....

\_\_\_\_\_ Recent posts: 3 \_\_\_\_\_

File	Posted On	Size	Tracking id
sample_dellior	10/29/03 11:29:38 CST	4745	PO28000007
Single862_1.xml	10/29/03 11:29:17 CST	876	PO28000006
test.doc	10/29/03 11:28:52 CST	4737	PO28000005

Figure 2: Receptor Form Updated With Recent Posts Table

Once the web form has been initialized, repeat steps 4 – 8 for any number of subsequent posts. A new entry is added to the recent Post Table every time a successful post is performed.

## Get Mailbox Directory as a Web Page

- 1 Open the Web browser and enter the following URL into the address box:

***http(s)://host:port/invoke/HTTPConnector.Mailbox/get***

***or***

***http(s)://host:port/invoke/HTTPConnector.Mailbox/get?action=html\_dir***

- 2 Enter your username and password if prompted.
- 3 The browser should display the HTTP(S) Mailbox Directory, as shown in **Figure 3**.

The directory contains references to the messages that have not been retrieved from the message box yet. Every message has an internal retrieval counter on the Covisint Connect messaging hub. Every time the message gets retrieved, the corresponding counter is incremented by one. The value of the retrieval counter for messages on the regular mailbox directory equals zero.

### HTTP(S) Mailbox Directory

Username: **widget**  
Messages in mailbox: **10**

Message Id	Action	Type	Size	Created On
M0000282	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:07 EDT
M0000283	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:07 EDT
M0000284	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:07 EDT
M0000285	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:08 EDT
M0000286	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:08 EDT
M0000287	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:08 EDT
M0000288	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:08 EDT
M0000289	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:09 EDT
M0000290	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:10 EDT
M0000291	<a href="#">view/get</a>	DELFOR	4737	07/25/03 12:17:10 EDT

Figure 3: HTTP(S) Mailbox Directory

## View Mailbox Message Using Mailbox Directory

- 1 Use **view** hyperlink on the mailbox directory to display the corresponding message in the browser window.

**Note:**

The view functionality is introduced for reference and debugging purposes only. If you click on the view hyperlink, it does not change the value of the retrieval counter. Since HTTP(S) Mailbox doesn't know payload type and encoding for the particular message, it returns the message data stream as text/plain content with iso\_8859\_1 charset, in most cases viewable in the browser. However, in case of binary or non-ASCII payload, all of the characters will not be displayed correctly.

**DO NOT SAVE THE MESSAGE USING THE VIEW HYPERLINK.**



- 2 When finished viewing, click the browser **Back** button to return to the Mailbox Directory.

### **Get Mailbox Message Using Mailbox Directory**

- 1 Use **GET** hyperlink to retrieve the message from the HTTP(S) Mailbox and save it to your local disk.

When you click on this link you are prompted with the *open/save* dialog window. Select *save* and use the suggested filename to save the message. The filename is created using the mailbox message ID and extension **.mbx**. After you save the message, the retrieval counter will be incremented by one.

- 2 You can continue to save messages one after another until you reach the end of the mailbox directory table.
- 3 Click the **Refresh** button on the browser after saving messages.

The updated mailbox directory table should not contain entries for messages that have already been retrieved.

### **Get Full Mailbox Directory as a Web Page**

- 1 Open the Web browser and enter the following URL into the address box:

***http(s)://host:port/invoke/HTTPConnector.Mailbox/get?action=full\_html\_dir***

- 2 Enter your username and password if prompted.

The browser should display a so-called full HTTP(S) Mailbox Directory that contains references to all messages currently available on the Mailbox Database table, regardless of the retrieval counter value. The Mailbox Directory table is extended with the Covisint internal tracking ID, last retrieval timestamp and retrieval counter.

#### **Note:**

**The full directory feature is introduced mostly for debugging and troubleshooting purposes.**

### **Get Mailbox Directory For Specified Time Range**

You can specify a time range while querying both regular and full mailbox directories. To do this, add two extra parameters with timestamp values to the URL string: "**since**" and "**until**". The timestamp format is "yyyy-MM-dd\_HH:mm:ss\_z." See example below:

`http(s)://host:port/invoke/HTTPConnector.Mailbox/get?action=html_dir&since=2003-10-30_11:23:-5_EST&until=2003-10-31_02:10:00_EST`

If the "**since**" timestamp is omitted, the directory contains all transactions submitted before the "**until**" timestamp. If the "**until**" timestamp is omitted, the directory contains all transactions submitted after the "since" timestamp.

# Access from a Programmatic Client – HTTP(S) Mailbox

---

### ***Post Message from a Programmatic Client***

HTTP(S) Mailbox **post** service accepts an HTTP(S) POST request from any programmatic client that supports the HTTP 1.1 protocol specification. The message payload should be contained in the body of the post. The content type setting on the HTTP header should have a value of *application/octet-stream*. The internal Covisint tracking ID is returned as a small XML snippet in the body of the HTTP response, formed in response to the customer's post request.

The URL string to use with the programmatic client:

***http(s)://host:port/invoke/HTTPConnector.Mailbox/post***

Request header requirement:

***Content-type: application/octet-stream***

Response header to expect:

***Content-type: text/xml***

Response body to expect:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <TrackingId>a117a21b-ef33-12b7-1432</TrackingId>
</Response>
```

## ***Post Message with Explicit Routing Data***

As an additional feature, the post service allows you to perform so-called channel-driven routing, which controls the message routing process. Regularly, the routing data is extracted from the message body during the document recognition process and doesn't require any customer intervention. However, if you want to override the routing data contained in the message body itself, or to specify routing data explicitly, e.g. for the message that doesn't contain any routing data at all, you can do that by adding three extra parameters. Do this either to the HTTP request header or to the URL request string (whichever is easier to implement - your choice).

### ***Pass routing data in the HTTP request header***

The URL string to use:

***http(s)://host:port/invoke/HTTPConnector.Mailbox/post***

Request header requirement:

***Content-type:*** application/octet-stream  
***From-party-id:*** Covisint-defined Partner ID  
***From-id-qualifier:*** Covisint-defined Partner ID Interchange Qualifier  
***To-party-id:*** Covisint-defined Partner ID  
***To-id-qualifier:*** Covisint-defined Partner ID Interchange Qualifier  
***Message-type:*** Covisint-defined Message Type  
***Message-format:*** Covisint-defined Message Format  
***Standard-version:*** Covisint-defined version of document standard  
***Customer-tracking-no:*** Control number or any other tracking number

### ***Pass routing data in the URL request string***

The URL string to use:

***http(s)://host:port/invoke/HTTPConnector.Mailbox/post?from\_party\_id=aaaa&from\_id\_qualifier=bb&to\_party\_id=cccc&to\_id\_qualifier=dddd&message\_type=eeee&message-format=ffff&standard\_version=gggg&customer\_tracking\_no=hhhh***

Request header requirement:

***Content-type: application/octet-stream***

In both cases of channel-driven routing, the data that is contained in the HTTP header or URL string overrides the result of the Covisint document recognition process.

## ***Get Mailbox Directory from a Programmatic Client***

The HTTP(S) Mailbox service returns the directory of the virtual mailbox in a simple XML format as a response to the HTTP **GET** request from the programmatic client. The XML directory contains references to the messages that have not been retrieved from the message box yet. The value of the retrieval counter for such messages is less than or equal to zero.

The URL string to use:

***http(s)://host:port/invoke/HTTPConnector.Mailbox/get?action=xml\_dir***

Response header to expect:

***Content-type: text/xml***

Response body:

```
<?xml version="1.0"?>
<MailboxDir>
  <Msg>
    <MailboxId>M0000062</MailboxId>
    <Size>14343</Size>
    <Type>830</Type>
    <TimeCreated>06/06/03 16:09:24 EDT</TimeCreated>
  </Msg>
  <Msg>
    <MailboxId>M0000063</MailboxId>
    <Size>14343</Size>
    <Type>830</Type>
    <TimeCreated>06/06/03 16:09:25 EDT</TimeCreated>
  </Msg>
  <Msg>
    <MailboxId>M0000066</MailboxId>
    <Size>14343</Size>
    <Type>830</Type>
    <TimeCreated>06/06/03 16:09:25 EDT</TimeCreated>
  </Msg>
</MailboxDir>
```

If the mailbox is empty, there are no `<Msg>` elements in the directory document.

### ***Get Full Mailbox Directory from a Programmatic Client***

This is the same as the regular mailbox directory function but it returns the full mailbox XML directory that contains references to all messages currently available in the mailbox. The XML response document is extended with the tracking ID, last retrieval timestamp and retrieval counter. The full directory feature is introduced mostly for debugging and troubleshooting purposes.

The URL string to use:

***http(s)://host:port/invoke/HTTPConnector.Mailbox/get?action=full\_xml\_dir***

Response header to expect:

***Content-type: text/xml***

Response body:

```
<?xml version="1.0"?>
<MailboxDir>
  <Msg>
    <MailboxId>M0000004</MailboxId>
    <TrackingId>8609a21b-985a-11d7-8432</TrackingId>
    <Size>2999</Size>
    <Type>830</Type>
    <TimeCreated>06/05/03 15:25:26 EDT</TimeCreated>
    <TimeRetrieved>06/11/03 17:11:50 EDT</TimeRetrieved>
    <RetCount>14</RetCount>
  </Msg>
  <Msg>
    <MailboxId>M0000028</MailboxId>
    <TrackingId>84a2818d-985a-11d7-8432</TrackingId>
    <Size>14343</Size>
    <Type>830</Type>
    <TimeCreated>06/06/03 16:07:53 EDT</TimeCreated>
    <TimeRetrieved>06/11/03 13:28:03 EDT</TimeRetrieved>
    <RetCount>3</RetCount>
  </Msg>
  <Msg>
    <MailboxId>M0000030</MailboxId>
    <TrackingId>8609b2b2-985a-11d7-8432</TrackingId>
    <Size>14343</Size>
    <Type>830</Type>
    <TimeCreated>06/06/03 16:07:57 EDT</TimeCreated>
    <TimeRetrieved>06/11/03 16:06:36 EDT</TimeRetrieved>
    <RetCount>2</RetCount>
  </Msg>
</MailboxDir>
```

If the mailbox is empty, there are no `<Msg>` elements in the directory document.

### **Get Mailbox Message by ID**

You can specify a time range when you query both regular and full mailbox directories. Please use the same syntax as described above for the browser access mode.

### **Get Mailbox Message by ID**

The HTTP(S) Mailbox service returns a message with a specified Mailbox ID as a response to the **GET** request from the programmatic client. The response header carries the Mailbox ID and internal tracking ID for the message. The retrieval is performed regardless of the retrieval counter's value. This feature allows you to execute multiple retrieval attempts in case of an intermittent failure.

The regular message retrieval procedure is to get the HTTP(S) Mailbox directory first (see above). Then, the actual message can be retrieved using the desired message Mailbox ID.

The URL string to use:

***http(s)://host:port/invoke/HTTPConnector.Mailbox/get?action=msg&id=M0000030***

Response header to expect:

***Content-type:*** application/octet-stream

***Mailbox-id:*** M0000030

***Tracking-id:*** P028002121

***Message-type:*** DELFOR

Response body:

The binary stream with the message payload or web-server **403 - Service error** if payload is null or if no message found.

The retrieval counter is incremented by one for every retrieval attempt.

## APPENDIX – HTTP(S) MAILBOX

### Common Errors – HTTP(S) Mailbox

To indicate an error response, HTTP(S) Mailbox always uses standard HTTP response codes with detailed error descriptions passed in the HTTP format as the response body. See the most common error situations below.

Error Description	HTTP Status Code	HTTP Status Message	Response Body
No error	200	OK	~
Access denied	403	Forbidden	"Access denied" error as the webMethods IS HTML error report
Unknown service or resource in the URL string	403	Forbidden	"Unknown Service Exception" as the webMethods IS HTML error report
Message does not exist or payload is empty	403	Forbidden	403 Forbidden - Service Error - Message does not exist or payload is empty
Invalid action for "get" specified	403	Forbidden	403 Forbidden - Service Error - Not implemented
Common mailbox failure	403	Forbidden	403 Forbidden - Service Error - HTTP mailbox failure <details>
Common Web server error	500	Server Error	~

## Glossary – HTTP(S) Mailbox

---

The following terms are used in the guide:

**ANX** - Advanced Network Exchange (formerly Automotive Network Exchange) used throughout North America. ANX is a TCP/IP network comprised of trading partner subscribers, certified service provider and network points allowing for efficient and secure electronic communications among subscribers, with only a single connection.

**EDI** - Electronic Data Interchange, a company-to-company electronic exchange of business data using industry standards. EDI is currently used to electronically transmit product information, prices, promotions, product authorizations, purchase orders, invoices, and payments.

**ENX** - European Network Exchange, a private IP data network used by the European automotive industry.

**HTTP** - Hypertext Transfer Protocol, an application-level protocol standard (defined in RFC2616) used on IP networks, such as the Internet, to transfer Web content between computers.

**HTTP(S) Connector** - A messaging hub component that provides access to messaging hub services using the HTTP(S) protocol.

**HTTP(S) Mailbox** - A sub-component of the HTTP(S) Connector that provides access to messaging hub services in mailbox mode using the HTTP(S) protocol.

**ISA, GS, ST** - EDI standard definitions for the specific parts of a business document.

**Programmatic HTTP(S) Client** - A software component in custom applications that implements the HTTP and SSL protocol RFCs and uses them to communicate with the Web (HTTP) servers in automatic mode.

**RFC** - A Request for Comments, a format document from the Internet Engineering Task Force (IETF) that is the result of committee drafting and subsequent review by interested parties. Some RFCs are informational in nature.

**SSL** - Secure Socket Layer protocol that provides data encryption, server authentication, message integrity and optional client authentication for TCP/IP connections.

**TCP** - A transport-level protocol widely used on IP networks, including the Internet.

**VPN** - Virtual Private Network, a virtual private network is a way to use a public telecommunication infrastructure, such as the Internet, to provide remote offices or individual users with secure access to their organization's network.