

REST Initiator Transport Adaptor

- Overview
 - Outgoing message representations
 - Raw FIX message
 - JSON
 - Body of XmlData (213) tag
 - Using secure and non-secure connections
 - Message validation
 - Error handling
 - Logging
 - Configuration example #1 (Enable extra logging levels)
 - Configuration example #2 (Log to separate file)
 - Configuration
 - Configuration parameters
 - TransportLayer.TransportAdapters
 - TransportLayer.RestOutTA.Description
 - TransportLayer.RestOutTA.DllName
 - TransportLayer.RestOutTA.Type
 - TransportLayer.RestOutTA.ClientID
 - TransportLayer.RestOutTA.LogCategory
 - TransportLayer.RestOutTA.Protocol
 - TransportLayer.RestOutTA.ServerHost
 - TransportLayer.RestOutTA.ServerPort
 - TransportLayer.RestOutTA.URI
 - TransportLayer.RestOutTA.SendingTimeout
 - TransportLayer.RestOutTA.SendingMaxAttempts
 - TransportLayer.RestOutTA.ConversionMethod
 - TransportLayer.RestOutTA.ResendMessageOnServerError
 - TransportLayer.RestOutTA.ErrorCodesForUndeliveredEvent
 - Proxy connection parameters
 - TransportLayer.RestOutTA.ProxyHost
 - TransportLayer.RestOutTA.ProxyPort
 - TransportLayer.RestOutTA.ProxyLogin
 - TransportLayer.RestOutTA.ProxyPassword
 - Validation parameters
 - TransportLayer.RestOutTA.ValidateFIXMessage
 - Security parameters
 - TransportLayer.RestOutTA.HTTPS.PrivateKey
 - TransportLayer.RestOutTA.HTTPS.Certificate
 - TransportLayer.RestOutTA.HTTPS.PrivateKeyPassword
 - TransportLayer.RestOutTA.authHeader
 - TransportLayer.RestOutTA.authValue
 - Configuration example
 - Monitoring Rest Initiator Transport Adapter with FIXICC
- Routing
- Troubleshooting
 - Host not found
 - Description
 - Solution
 - Send to Client failed
 - Description
 - Solution
 - Validation error
 - Description
 - Solution
 - Wrong security options
 - Description
 - Solution

Overview

This document contains a description of [FIXEdge](#)'s REST Initiator Transport Adaptor and common steps required to configure and use it.

FIXEdge REST initiator adaptor is a module in FIXEdge which sends FIX messages from the [Business Layer](#) to the destination system as an HTTP(S) REST POST request.

The adaptor has the following features:

- Send Raw FIX message
- Send FIX message in JSON format
- Send XmlData (213) tag data
- Secure connection support
- Proxy server support
- Message validation

Outgoing message representations

An outgoing HTTP message body can be formed in several modes:

- Raw FIX message with SOH as a delimiter
- JSON representation of FIX messages tags and values
- content of tag XmlData (213) of the FIX message

The message conversion type is specified with the `TransportLayer.RestOutTA.ConversionMethod` parameter.

Raw FIX message

Example of raw FIX message (pipe symbol stands for the SOH symbol):

```
8=FIX.4.2|9=153|35=D|49=BLP|56=SCHB|34=1|50=30737|97=Y|52=20000809-20:20:
50|11=90001008|1=10030003|21=2|55=TESTA|54=1|38=4000|40=2|59=0|44=30|47=I|60=20000809-18:20:32|10=061|
```

JSON

Example of JSON message (contains repeating groups):

```
{ "35": "E", "50": "30737", "116": "OMS1", "52": "20150101-01:01:01.080", "66": "List1", "394": "3", "68": "2", "73": [ { "11": "0003", "67": "1", "100": "DSMD", "55": "MSFT", "54": "1", "38": "100" }, { "11": "0004", "67": "2", "100": "DSMD", "55": "IBM", "54": "2", "38": "200" } ] }
```

Session level tags: BeginString(8), BodyLength(9), SenderCompID(49), TargetCompID(56) and CheckSum(10) are not inserted in output JSON body.

Body of XmlData (213) tag

The tags XmlData (213) and XmlDataLen (212) can appear in any message type. One of the common ways to use it is to create an [XML message \(e.g. non-FIX MessageType \(n\)\)](#) as a FIX message wrapper over a non-FIX message. The Rest Adapter can extract the data from the XmlData (213) and send it to the external services.

The example of exporting data from tag XmlData (213) (FIX message 35=n contain another FIX message 35=AE (marked orange) within XmlData (213)):

Input (from BL)	Output (sent to destination system)
<pre>8=FIX.4.4 9=259 35=n 49=0 56=0 34=1 52=20200317-10:23:55.068 212=201 213=8=FIX .4.4 9=168 35=E 49=0 56=RESTAdapter 34=1 50=30737 66=List1 116=OMS1 52=20191101-11:11:01.080 394=3 68=2 73=2 11=0003 67=1 100=DSMD 55=11 54=1 38=100 11=0004 67=2 100=DSMD 55=12 54=2 38=200 10=235 10=219 </pre>	<pre>8=FIX.4.4 9=168 35=E 49=0 56=RESTAdapter 34=1 50=30737 66=List1 116=OMS1 52=20191101-11:11:01.080 394=3 68=2 73=2 11=0003 67=1 100=DSMD 55=11 54=1 38=100 11=0004 67=2 100=DSMD 55=12 54=2 38=200 10=235 </pre>

Using secure and non-secure connections

The Transport Adapter supports secure (HTTPS) and non-secure (HTTP) connections, for details [see](#).

Message validation

Message validation is configured by using `TransportLayer.RestOutTA.ValidateFIXMessage`.

Example1: validation is off:

```
TransportLayer.RestOutTA.ValidateFIXMessage = No
```

Example2: a message is not sent, OnUndeliveredEvent on BL is generated:

```
TransportLayer.RestOutTA.ValidateFIXMessage = ValidateAndReject
```

Example of log record:

```
2020-02-12 12:18:11,971 UTC   WARN   [RestInitiatorClient2] 140246545049344 FIX validation error: Incorrect integer value in tag 67: 'check'[Group tag=73, Entry #=0] in message New Order - List (E) with sequence number 3.
2020-02-12 12:18:11,971 UTC   ERROR   [RestInitiatorClient2] 140246545049344 Error in sendToClient: Incorrect integer value in tag 67: 'check'[Group tag=73, Entry #=0] in message New Order - List (E) with sequence number 3.
```

For details [see](#).

Error handling

Error information about HTTP error codes is saved to the log. See [logging section](#).


A user can handle errors with the [Business Layer Engine](#) in case of error responses from the destination system (for example receiving 400 or 500 error codes).

An [OnUndeliveredMessageEvent](#) is generated and an undelivered message is transferred to FIXEdge Business Layer for further processing. In the example below, a rejected message from `RestInitiatorClient` will be sent to FIX session :

```
<OnUndeliveredMessageEvent>
  <Source>
    <Client Name="RestInitiatorClient" />
  </Source>
  <Action>
    <Send><FixSession SenderCompID="FE" TargetCompID="SC" /></Send>
  </Action>
</OnUndeliveredMessageEvent>
```

Logging

The transport adapter uses category **RestOutTA** for logging REST events.

 This category cannot be re-configured

The category of the other actions can be defined by the user in order to track Transport Adapters events from different adapters.

List of logged actions:

- Adapter initialization
- Parsing information, validation errors
- Configuration parameters
- Sent and received messages

Additional information about logging configuration can be found here:

- [FIXEdge logs format](#)
- [How to redirect FIX Antenna and/or FIXEdge logging to Syslog instead of files](#)
- [How to divide different categories and severities of log files into different files in the Logging section](#)

Configuration example #1 (Enable extra logging levels)

The following configuration example shows how to enable `Note`, `Debug` and `Trace` level of REST events; specifies logging category for `RestOutTA1` as `RestInitiatorClient1` and enable `Note`, `Debug` and `Trace` level of `RestOutTA1` adapter.

FIXEdge.properties

```
TransportLayer.RestOutTA1.LogCategory = RestInitiatorClient1
Log.RestOutTA.Device = File
Log.RestOutTA.DebugIsOn = true
Log.RestOutTA.TraceIsOn = true
Log.RestOutTA.NoteIsOn = true
Log.RestInitiatorClient1.Device = File
Log.RestInitiatorClient1.DebugIsOn = true
Log.RestInitiatorClient1.TraceIsOn = true
Log.RestInitiatorClient1.NoteIsOn = true
```

FIXEdge log example for this configuration:

FIXEdge.log

```
2019-10-29 13:34:04,676 UTC INFO [RestOutTA] 10012 Log Category is registered to FileLogger
Category name: RestOutTA
File name: '../FIXEdge1/log/FixEdge.log'
File record format: %date{ISO8601} %timezone %level%tablevel [%logger] %thread %message
Create unique file name: off
Recreate file on restart: off
Flush on each record: on
Time zone: UTC
Log rotate is on:
size limit is 104857600
File Limit: 10
Rotate directory: ../FIXEdge1/log
Note: Filename is shared with other categories
2019-10-29 13:34:04,677 UTC DEBUG [RestInitiatorClient1] 10012 Setting up HTTP session with
jsonplaceholder.typicode.com:80 endpoint
2019-10-29 13:34:04,677 UTC DEBUG [RestInitiatorClient1] 10012 Timeout for client 'RestInitiatorClient1'
= 60.
2019-10-29 13:34:04,677 UTC INFO [CC_Layer] 10012 Client RestInitiatorClient1 has logged in
2019-10-29 13:34:04,677 UTC INFO [RestInitiatorClient1] 10012 REST Initiator TA v.0.0.0.1 started.
2019-10-29 13:34:04,677 UTC INFO [TransportLayer] 10012 Module 'TransportLayer' version 0.1.1.5 was
loaded.
2019-10-29 13:34:04,677 UTC INFO [CC_Layer] 10012 Admin REST API is disabled
2019-10-29 13:34:04,677 UTC INFO [CC_Layer] 10012 Control Centre XML Sockets Monitor is started
2019-10-29 13:34:04,677 UTC INFO [CC_Layer] 10012 Control Centre Layer is initialized
2019-10-29 13:34:04,677 UTC INFO [CC_Layer] 10012 FixEdge pid file '../FIXEdge1/log/FixEdge.pid'.
2019-10-29 13:34:04,678 UTC INFO [XmlSockImpl] 12532 Log Category is registered to FileLogger
```

Configuration example #2 (Log to separate file)

The following configuration example shows how to log Transport Adapter events into a separate log file `FIXEdge1/log/RestOutTA.log`:

FIXEdge.properties

```
TransportLayer.RestOutTA1.LogCategory = RestInitiatorClient1
Log.RestOutTA.Device = File
Log.RestOutTA.DebugIsOn = true
Log.RestOutTA.TraceIsOn = true
Log.RestOutTA.NoteIsOn = true
Log.RestOutTA.File.Name = FIXEdge1/log/RestOutTA.log
Log.RestInitiatorClient1.Device = File
Log.RestInitiatorClient1.DebugIsOn = true
Log.RestInitiatorClient1.TraceIsOn = true
Log.RestInitiatorClient1.NoteIsOn = true
Log.RestInitiatorClient1.File.Name = FIXEdge1/log/RestOutTA.log
```

RestOutTA.log example

RestOutTA.log

```
2019-10-29 13:40:07,830 UTC INFO [RestInitiatorClient1] 18488 Log Category is registered to FileLogger
Category name: RestInitiatorClient1
File name: '../FIXEdge1/log/RestOutTA.log'
File record format: %date{ISO8601} %timezone %level%tablevel [%logger] %thread %message
Create unique file name: off
Recreate file on restart: off
Flush on each record: on
Time zone: UTC
Log rotate is on:
  size limit is 104857600
  File Limit: 10
  Rotate directory: ../FIXEdge1/log
2019-10-29 13:40:07,830 UTC DEBUG [RestInitiatorClient1] 18488 Setting up HTTP session with
jsonplaceholder.typicode.com:80 endpoint
2019-10-29 13:40:07,830 UTC DEBUG [RestInitiatorClient1] 18488 Timeout for client 'RestInitiatorClient1'
= 60.
2019-10-29 13:40:07,831 UTC INFO [RestInitiatorClient1] 18488 REST Initiator TA v.0.0.0.1 started.
```



Configuration


See [How to configure adapters in FIXEdge](#) for a guide on how to enable Transport Adapter functionality in FIXEdge.


Using proxy server feature

The Transport Adapter supports proxy server configuration for cases when a direct connection from the FIXEdge server to the destination system is forbidden.

A host, port, user and password for proxy connection should be configured in `FIXEdge.properties` file, for details, [see](#).

Property	Description	Required	Values/Examples
Configuration parameters			
TransportLayer. TransportAdapters	The list of user-defined Transport Adapter names. The several instances should be separated by a comma  The following configuration will show configuration parameters for an adapter with the name TransportLayer.RestOutTA , i.e. <code>TransportLayer.TransportAdapters = TransportLayer.RestOutTA</code>	No	
TransportLayer. RestOutTA. Description	The description of the adapter. This information is shown in Monitoring API	Yes	REST Initiator Client
TransportLayer. RestOutTA.DIIName	Path to Transport Adapter library	Yes	Linux: <ul style="list-style-type: none">libREST_Initiator_TA-gcc44-MD-x64.so Windows: <ul style="list-style-type: none">bin/REST_Initiator_TA-vc10-MD-x64.dll
TransportLayer. RestOutTA.Type	Transport Adapter library type  deprecated parameter	No	Supported values: <ul style="list-style-type: none">DLL
TransportLayer. RestOutTA.ClientID	User defined id/name for referencing to Transport Adapters in Business Logic configuration	Yes	RestInitiatorClient

TransportLayer. RestOutTA. LogCategory	Transport adaptor log category.	Yes	RestInitiatorClient
TransportLayer. RestOutTA. Protocol	Connection protocol	Yes	Supported values: <ul style="list-style-type: none"> • HTTP - for non-secure connections • HTTPS - for secure connections
TransportLayer. RestOutTA. ServerHost	The destination host or IP address for sending messages	Yes	
TransportLayer. RestOutTA. ServerPort	The Port for the destination host	Yes	
TransportLayer. RestOutTA. URI	URI for HTTP request. The resulted URI will be composed as: <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;">  <Protocol>://<ServerHost>:<ServerPort><URI> </div>	Yes	/messages
TransportLayer. RestOutTA. SendingTimeout	Time (in seconds) for waiting for the response. After this time the adapter tries next time to start a connection. The maximum value may be limited by Operating System.	No	Default value = 0 - no timeout
TransportLayer. RestOutTA. SendingMaxAttempts	Number of connection attempts	No	Default value = 2
TransportLayer. RestOutTA. ConversionMethod	The conversion method is the way the outgoing message is converted	Yes	Supported values: <ul style="list-style-type: none"> • Raw - Raw FIX message with SOH as a delimiter • NumericTagValueMapping - JSON representation of FIX messages tags and values • WrapInXmlMessage - content of tag XmlData (213) of the FIX message
TransportLayer. RestOutTA. ResendMessageOn ServerError	Controls undelivered message resending in the case of an error received from the server. <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;">  This parameter does not control resending in the case of network errors. </div>	Yes	Default value = False
TransportLayer. RestOutTA. ErrorCodesForUndeliveredEvent	String parameter with a comma-separated list of error codes that generates an OnUndeliveredMessageEvent event and prevents resending.	Yes	Default value = 503 (Service Unavailable), 504 (Gateway Timeout) The list can be empty.
Proxy connection parameters			
TransportLayer. RestOutTA. ProxyHost	Proxy host	No	
TransportLayer. RestOutTA. ProxyPort	Proxy port	Conditional Mandatory if ProxyHost is set	
TransportLayer. RestOutTA. ProxyLogin	Proxy user name	No	

TransportLayer. RestOutTA. ProxyPassword	Proxy user password	Conditional Mandatory if ProxyLog in is set	
Validation parameters			
TransportLayer. RestOutTA. ValidateFIXMessage	Enables/disables validation of FIX message <div style="border: 1px solid #ccc; padding: 5px; width: fit-content;">  The validation isn't applied for raw conversion method. I.e. <code>TransportLayer.RestOutTA.ConversionMethod = Raw</code> </div>	No	Supported values: <ul style="list-style-type: none"> • No - the validation is disabled. Default • ValidateAndReject - validation is enabled. The Adapter doesn't send a message to BL if the message doesn't pass validation. • ValidateAndWarn - validation is enabled. The Adapter sends a message to BL even if the message doesn't pass validation. The validation error warning message is logged in this case
Security parameters			
TransportLayer. RestOutTA.HTTPS. PrivateKey	path to the private key file	Conditional Mandatory if Protocol = HTTPS	
TransportLayer. RestOutTA.HTTPS. Certificate	path to certificate key file	Conditional Mandatory if Protocol = HTTPS	
TransportLayer. RestOutTA.HTTPS. PrivateKeyPassword	password for encrypted private key file	No	
TransportLayer. RestOutTA. authHeader	The HTTP header name for authentication. All outgoing requests will contain the HTTP header with the value of the parameter.	No	
TransportLayer. RestOutTA. authValue	Authentication value in HTTP header	No	

Configuration example

A typical configuration for windows

FIXEdge.properties

```
TransportLayer.TransportAdapters = TransportLayer.RestOutTA1
TransportLayer.RestOutTA1.Description = RESTInitiatorClient1
TransportLayer.RestOutTA1.DllName = bin/REST_Initiator_TA-vc10-MD-x64.dll
TransportLayer.RestOutTA1.Type = DLL
TransportLayer.RestOutTA1.ClientID = RestInitiatorClient1
TransportLayer.RestOutTA1.LogCategory = RestInitiatorClient1
TransportLayer.RestOutTA1.Protocol = HTTP
TransportLayer.RestOutTA1.ServerHost = jsonplaceholder.typicode.com
TransportLayer.RestOutTA1.ServerPort = 80
#TransportLayer.RestOutTA1.ProxyHost =
#TransportLayer.RestOutTA1.ProxyPort =
#TransportLayer.RestOutTA1.ProxyLogin =
#TransportLayer.RestOutTA1.ProxyPassword =
TransportLayer.RestOutTA1.URI = /posts
TransportLayer.RestOutTA1.SendingTimeout = 60
TransportLayer.RestOutTA1.SendingMaxAttempts = 2
TransportLayer.RestOutTA1.ConversionMethod = NumericTagValueMapping
TransportLayer.RestOutTA1.ValidateFIXMessage = ValidateAndWarn
Log.RestOutTA.Device = File Console
Log.RestOutTA.DebugIsOn = true
Log.RestOutTA.TraceIsOn = true
Log.RestOutTA.NoteIsOn = true
Log.RestOutTA.File.Name = FIXEdgel/log/RestOutTA.log
Log.RestInitiatorClient1.Device = File
Log.RestInitiatorClient1.DebugIsOn = true
Log.RestInitiatorClient1.TraceIsOn = true
Log.RestInitiatorClient1.NoteIsOn = true
Log.RestInitiatorClient1.File.Name = FIXEdgel/log/RestOutTA.log
```

Example of a configuration block with security parameters:

FIXEdge.properties

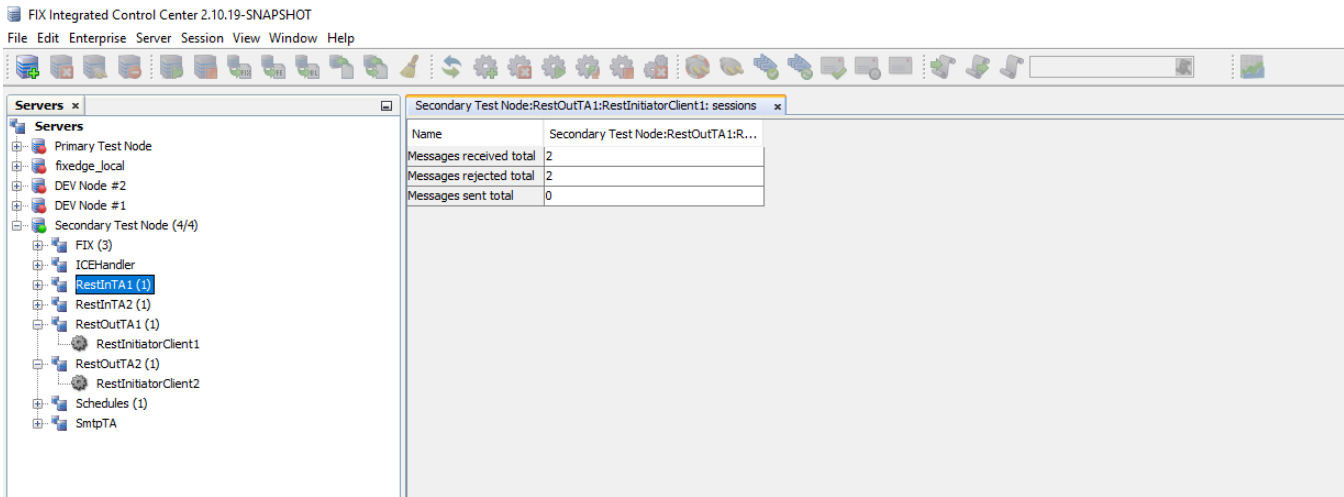
```
TransportLayer.RestOutTA1.Protocol = HTTPS
TransportLayer.RestOutTA1.HTTPS.PrivateKey = FIXEdgel/conf/TestRESTAPI.key
TransportLayer.RestOutTA1.HTTPS.Certificate = FIXEdgel/conf/TestRESTAPI.crt
TransportLayer.RestOutTA1.authHeader = apikey
TransportLayer.RestOutTA1.authValue = QWERTY0123456789
```

Monitoring Rest Initiator Transport Adapter with FIXICC

REST Initiator Transport Adapter is displayed in [FIXICC](#) in a tree in the "servers" list at the left. In the example below, RestOutTA1 is an adapter name and ResInitiatorClient1 is a ClientID. Messages statistics are shown at the right -

- total number of sent messages
- total number of received messages
- total number of rejected messages

On the picture below the configuration with two Rest Initiator adapters is shown (RestOutTA1 and RestOutTA2).



Routing

The Transport Adapter can be accessed in BL via ClientIDparameter. See the example of BL_Config.xml rule below (ClientID=RestInitiatorClient1):

BL_Config.xml

```
<Rule>
  <Source>
    <FixSession SenderCompID="FIXCLIENT1" TargetCompID="FIXEDGE" />
  </Source>
  <Action>
    <Send Name="RestInitiatorClient1" />
  </Action>
</Rule>
```

Troubleshooting

In all troubleshooting cases, first check error information in the REST TA logfile (defined by Log.*.File.Name parameter) and the FIXEdge.log.

To check that the Adapter has started correctly, search for the string "REST Initiator". An example of the output:

```
REST Initiator TA v.0.0.0.1 started
```

Host not found


Description

Message is not sent, adapter is initiated correctly, user gets error in REST TA log file:

```

❗ 2019-10-24 06:46:21,622 UTC DEBUG [RestInitiatorClient1] 27532 sendRequest attempt 1 of 2
2019-10-24 06:46:21,629 UTC ERROR [RestInitiatorClient1] 27532 Client 'RestInitiatorClient1' could not send POST request to the server (2)
'jsonplaceholder.typicode.com'. Reason: Host not found: jsonplaceholder.typicode.com.
2019-10-24 06:46:21,629 UTC DEBUG [RestInitiatorClient1] 27532 sendRequest attempt 2 of 2
2019-10-24 06:46:21,630 UTC ERROR [RestInitiatorClient1] 27532 Client 'RestInitiatorClient1' could not send POST request to the server (2)
'jsonplaceholder.typicode.com'. Reason: Host not found: jsonplaceholder.typicode.com.
2019-10-24 06:46:21,631 UTC WARN [RestInitiatorClient1] 27532 sendToClient failed. Sending message reject.
2019-10-24 06:46:21,639 UTC DEBUG [RestInitiatorClient1] 27532 sendToClient finished
```

In FIXEdge.log:

 2019-10-24 06:46:21,631 UTC ERROR [RestOutTA] 27532 Error in sendToClient: Client 'RestInitiatorClient1' could not send POST request to the server (2) 'jsonplaceholder.typicode.com'. Reason: Host not found: jsonplaceholder.typicode.com.


Solution

A possible reason for this error is the wrong host. Check that the host is available (ping<ServerHost>). If the host is available, contact Support. Otherwise, correct the TransportLayer.RestOutTA.ServerHostparameter value in adapter configuration.

Send to Client failed

Description

Message is not sent, adapter is initiated correctly, user gets error in REST TA log file:

 2019-10-24 07:17:05,706 UTC DEBUG [RestInitiatorClient1] 24408 sendToClient started
2019-10-24 07:17:05,706 UTC DEBUG [RestInitiatorClient1] 24408 sendRequest attempt 1 of 2
2019-10-24 07:17:06,146 UTC ERROR [RestInitiatorClient1] 24408 REST server returned an error response status: 404
2019-10-24 07:17:06,146 UTC WARN [RestInitiatorClient1] 24408 sendToClient failed. Sending message reject.
2019-10-24 07:17:06,146 UTC DEBUG [RestInitiatorClient1] 24408 sendToClient finished

Solution

A possible reason is wrong URI. If the URI value is correct, contact Support. Otherwise, correct the TransportLayer.RestOutTA.URlparameter value in adapter configuration.

Validation error

Description

User gets validation error, but message is sent successfully:

 2019-10-24 08:19:25,663 UTC DEBUG [RestInitiatorClient1] 31516 sendToClient started
2019-10-24 08:19:25,663 UTC WARN [RestInitiatorClient1] 31516 FIX validation error: Field value 'FUT' does not meet ValBlock dictionary conditions in tag SecurityType (167) in message Execution Report (8) with sequence number 3.
2019-10-24 08:19:25,663 UTC DEBUG [RestInitiatorClient1] 31516 sendRequest attempt 1 of 2
2019-10-24 08:19:26,108 UTC TRACE [RestInitiatorClient1] 31516 Message sent successfully.
2019-10-24 08:19:26,108 UTC DEBUG [RestInitiatorClient1] 31516 sendToClient finished

User gets validation error, message is rejected:

 2019-10-24 08:22:27,526 UTC DEBUG [RestInitiatorClient1] 34024 sendToClient started
2019-10-24 08:22:27,526 UTC WARN [RestInitiatorClient1] 34024 FIX validation error: Field value 'FUT' does not meet ValBlock dictionary conditions in tag SecurityType (167) in message Execution Report (8) with sequence number 2.
2019-10-24 08:22:27,526 UTC WARN [RestInitiatorClient1] 34024 sendToClient failed. Sending message reject.
2019-10-24 08:22:27,526 UTC DEBUG [RestInitiatorClient1] 34024 sendToClient finished

Solution

This validation error means that the FIX message has incorrect tag values for message structure. In the first case, the message is sent successfully because ValidateFIXMessage = ValidateAndWarn is set up. In the second case, the message is rejected because ValidateFIXMessage = ValidateAndReject is set up. Possible actions:

- correct message
- update dictionary
- turn off validation: TransportLayer.RestOutTA.ValidateFIXMessage = No).

Wrong security options

Description

Message is not sent, adapter is initiated correctly, user gets error in log:

```
! 2019-10-24 08:36:44,231 UTC DEBUG [RestInitiatorClient1] 22820 sendToClient started
2019-10-24 08:36:44,232 UTC DEBUG [RestInitiatorClient1] 22820 sendRequest attempt 1 of 2
2019-10-24 08:36:44,436 UTC ERROR [RestInitiatorClient1] 22820 Client 'RestInitiatorClient1' could not send POST request to the server (2)
'jsonplaceholder.typicode.com'. Reason: SSL Exception: error:140770FC:SSL routines:SSL23_GET_SERVER_HELLO:unknown protocol.
2019-10-24 08:36:44,436 UTC DEBUG [RestInitiatorClient1] 22820 sendRequest attempt 2 of 2
2019-10-24 08:36:44,561 UTC ERROR [RestInitiatorClient1] 22820 Client 'RestInitiatorClient1' could not send POST request to the server (2)
'jsonplaceholder.typicode.com'. Reason: SSL Exception: error:140770FC:SSL routines:SSL23_GET_SERVER_HELLO:unknown protocol.
2019-10-24 08:36:44,561 UTC WARN [RestInitiatorClient1] 22820 sendToClient failed. Sending message reject.
2019-10-24 08:36:44,561 UTC DEBUG [RestInitiatorClient1] 22820 sendToClient finished
```

or:

```
! 019-10-30 06:23:10,941 UTC DEBUG [RestInitiatorClient1] 24840 Setting up HTTP session with jsonplaceholder.typicode.com:443 endpoint
2019-10-30 06:23:10,941 UTC DEBUG [RestInitiatorClient1] 24840 Timeout for client 'RestInitiatorClient1' = 60.
2019-10-30 06:23:10,941 UTC INFO [RestInitiatorClient1] 24840 REST Initiator TA v.0.0.0.1 started.
2019-10-30 06:23:25,490 UTC DEBUG [RestInitiatorClient1] 27036 sendToClient started
2019-10-30 06:23:25,490 UTC DEBUG [RestInitiatorClient1] 27036 sendRequest attempt 1 of 2
2019-10-30 06:23:25,660 UTC ERROR [RestInitiatorClient1] 27036 REST server returned an error response status: 400
2019-10-30 06:23:25,660 UTC WARN [RestInitiatorClient1] 27036 sendToClient failed. Sending message reject.
2019-10-30 06:23:25,660 UTC DEBUG [RestInitiatorClient1] 27036 sendToClient finished
```

Solution

A possible reason for this error is that the security parameters are not correct or do not correspond to destination system connection properties. Try to change `TransportLayer.RestOutTA.Protocol` or `TransportLayer.RestOutTA.ServerPort` parameters.