Application Gateway Specification Document

Global eTelecom, Inc.

12/13/2011

Version 1.6

Introduction:

This document serves as a technical guide for transmitting merchant applications to GETI and details the communication method, application request specifications, and response specifications. Its purpose is to provide software developers with the necessary information to create an interface for processing merchant applications.

Overview:

The Application Gateway is designed to accommodate various input requirements. This allows for the development of a single interface that can be easily configured to handle many different scenarios.

The Application Gateway uses web services to present distributed methods for integration into client applications, and an interface with the Application Gateway can be developed with any programming language that can consume a web service.

Extensible Markup Language (XML) is used to send data packet requests to the Authorization Gateway and to receive a response back. Simple Object Access Protocol (SOAP) is used for XML message exchange over HTTPS. The Application Gateway also employs a custom SOAP header for authentication information.

Contents

Connection Method

GETI supports connection via secure (https) webservice using SOAP. SOAP is a simple XML-based protocol to let applications exchange information over HTTP.
The webservice address used for certification and testing is as follows:

https://demo.eftchecks.com/webservices/AppGateway.asmx

A username and password for certification will be provided upon request.

NOTE: A live webservice address, user name, and password will be supplied upon successful certification.

SUBMISSION

The Application Gateway has been designed for fast and easy integration with your existing system. Simply create an xml data packet that conforms to the NewMerchAppBPSchemaWithGC.xdr and pass it to the Application Gateway for processing. To accomplish this the Authorization Gateway provides 2 web methods; one for certification and one for production. In addition, each web method contains a custom SOAP header used for authentication.

SOAP Header

The SOAP header contains the following fields:

UserName	String	Username provided by GETI for authorization.
Password	String	Password provided by GETI for authorization.

Web Methods

A definition of the web methods can be found below. Each web method contains a hyperlink to a sample SOAP request and response.

- **BoardCertificationMerchant_ACH**
    - **URL:(**https://demo.eftchecks.com/webservices/AppGateway.asmx?op=BoardCertificationMerchant_ACH)
    - **Description:** This method will process an **ACH** merchant application and return a detail success or failure response. This method is used during interface testing and certification.
    - **Input:** Accepts an XML string called a DataPacket that must conform to the NewMerchAPP_ACH.XSD Application Schema (https://demo.eftchecks.com/webservices/Schemas/App/NewMerchApp_ACH.xsd)

***NOTE – ISO ID 9999 is to be used for all Certification Testing**

  - Sample XML File can be found here (https://demo.eftchecks.com/webservices/Schemas/App/Example/NewMerchAppSample_ACH.xml)
  - **Output:** Outputs an XML string.

- BoardMerchant_ACH
  - **Description:** This method will process an **ACH** merchant application and return a detail success or failure response.
  - **Input:** Accepts an XML string called a DataPacket that must conform to the NewMerchAPP_ACH.XSD Application Schema (https://demo.eftchecks.com/webservices/Schemas/App/NewMerchApp_ACH.xsd)
  - **Output:** Outputs an XML string.

- BoardCertificationMerchant_Check21
  - Url:(https://demo.eftchecks.com/webservices/AppGateway.asmx?op=BoardCertificationMerchant_Check21)
  - **Description:** This method will process a **Check21** merchant application and return a detail success or failure response. This method is used during interface testing and certification.
  - **Input:** Accepts an XML string called a DataPacket that must conform to the NewMerchAPP_Check21.XSD Application Schema (https://demo.eftchecks.com/webservices/Schemas/App/NewMerchApp_Check21.xsd)

    ***NOTE – ISO ID 9999 is to be used for all Certification Testing**
  - Sample XML File can be found here (https://demo.eftchecks.com/webservices/Schemas/App/Example/NewMerchAppSample_Check21.xml)
  - **Output:** Outputs an XML string.

- BoardMerchant_Check21
  - **Description:** This method will process a **Check21** merchant application and return a detail success or failure response.
  - **Input:** Accepts an XML string called a DataPacket that must conform to the NewMerchAPP_Check21.XSD Application Schema (https://demo.eftchecks.com/webservices/Schemas/App/NewMerchApp_Check21.xsd)
  - **Output:** Outputs an XML string.

- BoardCertificationMerchant_Gift
  - Url:(https://demo.eftchecks.com/webservices/AppGateway.asmx?op=BoardCertificationMerchant_Gift)

  - **Description:** This method will process a **Gift** merchant application and return a detail success or failure response. This method is used during interface testing and certification.
  - **Input:** Accepts an XML string called a DataPacket that must conform to the NewMerchAPP_Gift.XSD Application Schema (https://demo.eftchecks.com/webservices/Schemas/App/NewMerchApp_Gift.xsd)

    ***NOTE – ISO ID 9999 is to be used for all Certification Testing**
  - Sample XML File can be found here (https://demo.eftchecks.com/webservices/Schemas/App/Example/NewMerchAppSample_Gift.xml)
  - **Output:** Outputs an XML string.

- **BoardCertificationTerminal_ACH**
  - **Description:** This method will process a terminal application to add a terminal to an EXISTING merchant location and return a detail success or failure response. This method is used during interface testing and certification.
  - **Input:**
    - GETI LocationID as Integer
    - Accepts an XML string called a DataPacket that must conform to the NewTermApp_ACH.XSD Application Schema (https://demo.eftchecks.com/webservices/Schemas/App/NewTermApp_ACH.xsd)
  - **Output:** Outputs an XML string.

- **BoardTerminal_ACH**
  - **Description:** This method will process a terminal application to add a terminal to an EXISTING merchant location and return a detail success or failure response
  - **Input:**
    - GETI LocationID as Integer
    - Accepts an XML string called a DataPacket that must conform to the NewTermApp_ACH.XSD Application Schema (https://demo.eftchecks.com/webservices/Schemas/App/NewTermApp_ACH.xsd)
  - **Output:** Outputs an XML string.

- **BoardCertificationTerminal_Check21**
  - **Description:** This method will process a Check21 terminal application to add a terminal to an EXISTING merchant location and return a detail success or failure response. This method is used during interface testing and certification.
  - **Input:**
    - GETI LocationID as Integer

        - Accepts an XML string called a DataPacket that must conform to the NewTermApp_Check21.XSD Application Schema (https://demo.eftchecks.com/webservices/Schemas/App/NewTermApp_Check21.xsd)
    - **Output:** Outputs an XML string.

- BoardTerminal_Check21
    - **Description:** This method will process a Check21 terminal application to add a terminal to an EXISTING merchant location and return a detail success or failure response. **Input:**
        - GETI LocationID as Integer
        - Accepts an XML string called a DataPacket that must conform to the NewTermApp_Check21.XSD Application Schema (https://demo.eftchecks.com/webservices/Schemas/App/NewTermApp_Check21.xsd)
    - **Output:** Outputs an XML string.

- UploadCertificationSupportingDocs
    - **Description:** This method will upload PDF byte arrays of the signed Merchant application as well as voided check and any other supporting documents that need to be attached. This method is used during interface testing and certification.
    - **Input:** Accepts a Merchant ID as an Integer called MerchantID and a PDF document as a Byte array called DataPacket.
    - **Output:** NONE

- UploadSupportingDocs
    - **Description:** This method will upload PDF byte arrays of the signed Merchant application as well as voided check and any other supporting documents that need to be attached.
    - **Input:** Accepts a Merchant ID as an Integer called MerchantID and a PDF document as a Byte array called DataPacket.
    - **Output:** NONE

- **RetrieveCertificationMerchantStatus**
    - **Description:** This method will process a merchant ID and return a detailed merchant status. This method is used during interface testing and certification.
    - **Input:** Accepts a merchant id as an integer.
    - **Output:** Outputs an XML string.

- **RetrieveMerchantStatus**
    - **Description:** This method will process a merchant ID and return a detailed merchant status.
    - **Input:** Accepts a merchant id as an integer.

  - **Output:** Outputs an XML string.

- **RequestCertificationCheckLimitIncrease**
  - **Description:** This method will request a check limit increase for a specified Terminal ID.
  - **Input(s):**
    - Requires a Terminalid as an integer.
    - Requires a RequestedCheckLimit as a decimal
  - **Output:** Outputs an XML string.
  - **Usage**: After request, use RetrieveCertificationMerchantStatus to see if the check limit increase was approved, and to retriever your new MID number to input into the physical terminal.

- **RequestCheckLimitIncrease**
  - **Description:** This method will request a check limit increase for a specified Terminal ID.
  - **Input(s):**
    - Requires a Terminalid as an integer.
    - Requires a RequestedCheckLimit as a decimal
  - **Output:** Outputs an XML string.
  - **Usage**: After request, use RetrieveMerchantStatus to see if the check limit increase was approved, and to retriever your new MID number to input into the physical terminal.

- **RequestCertificationBankAccountChange**
  - **Description:** This method will request a bank account change for a Location ID.
  - **Input(s):**
    - Requires a LocationID as an integer.
    - Requires a NewRoutingNumber as a string
    - Requires a NewAccountNumber as string
  - **Output:** Outputs an XML string with IssueID as integer.
  - **Usage**: After request, use `UploadCertificationIssueSupportingDocs` to upload signed merchant bank change request as pdf.

- **RequestBankAccountChange**
  - **Description:** This method will request a bank account change for a Location ID.
  - **Input(s):**
    - Requires a LocationID as an integer.
    - Requires a NewRoutingNumber as a string
    - Requires a NewAccountNumber as string
  - **Output:** Outputs an XML string with IssueID as integer.
  - **Usage**: After request, use `UploadIssueSupportingDocs` to upload signed merchant bank change request as pdf.

- **RequestCertificationMerchantCancellation**
  - **Description:** This method will request a merchant cancellation.
  - **Input(s):**
    - Requires a MerchantID as an integer.
    - Requires a CancellationReason as CancellationReason Enum
  - **Output:** Outputs an XML string with IssueID as integer.

- **RequestMerchantCancellation**
  - **Description:** This method will request a merchant cancellation.
  - **Input(s):**
    - Requires a MerchantID as an integer.
    - Requires a CancellationReason as CancellationReason Enum
  - **Output:** Outputs an XML string with IssueID as integer.

RESPONSE

Each web method in the Application Gateway will return an XML string and detail the success or failure of the submission. If the application is accepted the following XML response will be returned

The Application Gateway XML response may contain the following elements:

- **STATUS:** Will contain a text description of the overall status for the application. Status will either be “Approved” or “Failed”.
- **MESSAGE:** Will contain a summary of the success or failure of the merchant application.
- **APP_DATA:** Contains the detailed Merchant, Location, and Terminal elements.
  - **Merchant:** Contains the following attributes:
    - **ID:** Contains the numerical ID for the merchant.
    - **Name:** Contains the text name of the merchant.
    - **Type:** Contains the element type. (Merchant)
    - **Active:** Contains a 1. Indicating that the merchant is active.
    - **CrossRefID:** Contains a user defined ID for cross referencing the merchant.
  - **Location:** Contains the following attributes:
    - **ID:** Contains the numerical ID for the location
    - **Name:** Contains the text name of the location.
    - **Type:** Contains the element type. (Location)
    - **Active:** Contains a 1. Indicating that the location is active.
    - **CrossRefID:** Contains a user defined ID for cross referencing the location.
  - **Terminal:** Contains the following attributes:

- **ID:** Contains the numerical ID for the terminal.
- **Name:** Contains the text name of the terminal.
- **Type:** Contains the element type. (Terminal)
- **Active:** Contains a 1. Indicating that the terminal is active.
- **CrossRefID:** Contains a user defined ID for cross referencing the terminal.
- **ManualEntry:** Contains a type used for internal purposes.

Response Message - Example Success Response

```
<?xml version="1.0" encoding="utf-8"?><RESPONSE
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><STATUS>Approved</STATUS><MESSAGE
>1 merchant(s) created.
0 merchant(s) not created due to errors.

-----------------------
Merchants Created:
Test Merchant ACH 1 (ISO ID: 9999, CrossRef: 261407, Status: AppApprovedandActivated)

</MESSAGE><APP_DATA><Merchant ID="16" Name="Test Merchant ACH 1" Type="Merchant"
Active="1" CrossRefID="261407"><POC1 FirstName="John" LastName="Doe " UserName="JDoe -
MERCH5" Password="83782K4Q" /><Location ID="21" Name="Test Merchant " Type="Location"
Active="1" CrossRefID="261407" ACHName="TESTMERCHANT"><POC1 FirstName="John"
LastName="Doe" UserName="JDoe5" Password="02K6X2NN" /><Terminal ID="111150"
Name="Lipman Nurit 3000-01 (111150) " Type="Terminal" Active="1" CrossRefID="41680"
ManualEntry="N" MID="101-111150-609" /><Terminal ID="111151" Name="Hypercom T7+-02
(111151) " Type="Terminal" Active="1" CrossRefID="53317" ManualEntry="N" MID="101-111151-
609" /><Terminal ID="111152" Name="Hypercom T4100-03 (111152) " Type="Terminal"
Active="1" CrossRefID="296223" ManualEntry="N" MID="101-111152-609" /><Terminal
ID="111153" Name="Lipman Nurit 3000-04 (111153) " Type="Terminal" Active="1"
CrossRefID="296228" ManualEntry="N" MID="101-111153-609" /><Terminal ID="111154"
Name="Hypercom T4100-05 (111154) " Type="Terminal" Active="1" CrossRefID="296225"
ManualEntry="N" MID="101-111154-609"
/></Location></Merchant></APP_DATA><VALIDATION_MESSAGE><RESULT>Passed</RESU
LT><SCHEMA_FILE_PATH /></VALIDATION_MESSAGE></RESPONSE>
```

Exceptions

If an error occurs within the Application Gateway the XML string response will detail the reason for the error within an Exception element. The Exception element will NOT be present if an error did not occur.

EXCEPTION Element - Example as a child of the RESPONSE element

<?xml version="1.0" encoding="utf-8"?><RESPONSE xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"><VALIDATION_MESSAGE><RESULT>Failed</RESULT><SCHEMA_FILE_PATH>http://localhost/GETI.eMagnus.WebServices/schemas/app/NewMerchApp_ACH.xsd</SCHEMA_FILE_PATH><VALIDATION_ERROR LINE_NUMBER="1" LINE_POSITION="138"><SEVERITY>Error</SEVERITY><MESSAGE>The required attribute 'merchReturnFee' is missing.</MESSAGE></VALIDATION_ERROR></VALIDATION_MESSAGE></RESPONSE>

The Exception element will contain the following elements.

- **MESSAGE or VALIDATION_MESSAGE:** Contains text information about the exception.

SAMPLE CODE

Note: please add **https://demo.eftchecks.com/webservices/appgateway.asmx** as a web reference to your project.

```
using System;

namespace AppGatewaySample
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine(BoardCertificationMerchant());
        }


        public static string BoardCertificationMerchant()
        {
            //This function will get the Certification Terminal Settings for Terminal 1010.


            //Create variable to hold Authorization Gateway Response
            string myAppGatewayResponse;


            //Create an instance of the Authorization Gateway
            com.eftchecks.demo.AppGateway myAppGateway = new com.eftchecks.demo.AppGateway();


            //Create an instance of the Authorization Header
            com.eftchecks.demo.RemoteAccessHeader myAppHeader = new
com.eftchecks.demo.RemoteAccessHeader();


            //Populate the Auth Header with the User Name, Password, and Terminal ID
            myAppHeader.UserName = "myUserNameGoesHere";
            myAppHeader.Password = "myPasswordGoesHere";


            //Apply the Auth Header to the Auth Gateway
            myAppGateway.RemoteAccessHeaderValue = myAppHeader;


            String myDatapacket = "<xml>";
            //Get the Certification Terminal Settings from the Authorization Gateway
            myAppGatewayResponse = myAppGateway.BoardCertificationMerchant_ACH(myDatapacket);


            //Create a new XML Document for the Certification Terminal Settings
            System.Xml.XmlDocument myAppResponse = new System.Xml.XmlDocument();


            //Load the Certification Terminal Settings XML into an XML Document

            myAppResponse.LoadXml(myAppGatewayResponse);


            //Return the Certification Terminal Settings

            return myAppResponse.OuterXml.ToString();


        }


    }

}
```

Contact Information

For questions or to receive certification and live username/passwords and URLs please contact:

Integration Department
Integaration@globaletelecom.com

Document History

Version Number	Modification Date	Modification
1.01	7/8/2008	Added web methods UploadCertificationSupportingDocs and RetrieveCertificationMerchantStatus
1.2	8/4/2008	Added web method UploadSupportingDocs and RetrieveMerchantStatus and links to the published XSD and example XML.
1.3	6/5/2009	Added new BoardMethods for both ACH and Check21. Added Maintenance methods for Check Increase, Bank Account Change, Merchant

		Cancellation. Corrected Response Example to match responses.
1.4	11/17/2009	Added web methods for adding terminals to existing merchant locations.
1.5	4/20/2011	Added web methods for boarding gift merchants
1.6	12/13/2011	Added sample code

Global eTelecom, Inc.

877-454-3835
www.GlobaleTelecom.com
www.CheckTraining.com
Sales@GlobaleTelecom.com