

Travelex Insurance Services

Integration Guide

for Retail Plans

Version 2.04  
Authored By: TIS Development

Date created: 8/7/2008

Date modified: 7/11/2018

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Comments |
| 1.0 | 8/7/2008 | Zul Dewsi | Web-service API documentation |
| 1.1 | 7/20/2011 | Margo Catton | Web-service Product Updates |
| 1.2 | 9/7/2012 | Margo Catton | CreatePolicy 04/2009 Web Service added |
| 1.3 | 9/11/2014 | Troy Petry | Update Travel Basic, Travel Select & Travel Max product codes. |
| 1.31 | 4/26/2016 | Troy Petry | Update connection standards. SSL is no longer accepted. Client must use TLS. |
| 1.32 | 5/16/16 | Troy Petry | Remove OTPL plans, update strProduct samples |
| 1.4 | 1/23/2017 | Ann Cannavo | Add Travel Solve product codes. |
| 1.5 | 10/02/2017 | Ann Cannavo | New Product List and Requirements |
| 2.0 | 5/1/2018 | Troy Petry | Connection requirements and cardholder data tokenization. Removal of CreatePolicy042009. CreatePolicy052018 added. |
| 2.01 | 6/26/2018 | Troy Petry | Update the test url. |
| 2.02 | 7/3/2018 | Troy Petry | Update GetPaymentConfiguration Location number parameter name. |
| 2.03 | 7/11/2018 | Troy Petry | Update required parameters for GetPaymentConfiguration method. |
| 2.04 | 1/28/2020 | Troy Petry | Destination country codes must be sent as ISO alpha-3 values. |

# Table of Contents

[1. Web Service Integration Quick Start 4](#_Toc519092550)

[2. Web Service Integration Overview 6](#_Toc519092551)

[2.1 Technology Compatibility 6](#_Toc519092552)

[2.2 Security Model 7](#_Toc519092553)

[2.2.1 User/Password/Location# 7](#_Toc519092554)

[2.2.2 TLS 7](#_Toc519092555)

[2.3 Production / Test Environments 7](#_Toc519092556)

[2.3.1 Test 7](#_Toc519092557)

[2.3.2 Production 7](#_Toc519092558)

[2.4 Support 7](#_Toc519092559)

[3. Web Service Details 9](#_Toc519092560)

[3.1 Special Processing Rules 9](#_Toc519092561)

[3.2 Cardholder Information Tokenization (TSEP JavaScript) 9](#_Toc519092562)

[3.2.1 Implementation Steps 9](#_Toc519092563)

[3.2.2 Sessions 10](#_Toc519092564)

[3.2.3 Event Handling 11](#_Toc519092565)

[3.3 GetRate052005 11](#_Toc519092566)

[3.4 GetPaymentConfiguration 13](#_Toc519092567)

[3.5 CreatePolicy052018 13](#_Toc519092568)

[3.6 Web Service Calling Methodologies 17](#_Toc519092569)

[ Form POST 17](#_Toc519092570)

[ SOAP 1.1 17](#_Toc519092571)

[3.7 Error Handling 18](#_Toc519092572)

[Appendix A: Constants 19](#_Toc519092573)

[Destinations 19](#_Toc519092574)

[Cruise Lines 19](#_Toc519092575)

[Tour Operators 19](#_Toc519092576)

[Airlines 19](#_Toc519092577)

[US States 19](#_Toc519092578)

[Appendix B: Tokenization Activity Diagram 20](#_Toc519092579)

[Appendix C: Parameters 21](#_Toc519092580)

[Appendix D: TIS API Exception Handling 27](#_Toc519092581)

[Formatting / Field Validation Errors 27](#_Toc519092582)

[Business Rule Validation Errors 28](#_Toc519092583)

[Operational Errors 29](#_Toc519092584)

[Typical Web Service Integration Setup Errors 30](#_Toc519092585)

[Appendix D: TSEP Event Handling 31](#_Toc519092586)

[Appendix E: Webservice xml response 36](#_Toc519092587)

# Web Service Integration Quick Start

Here is a quick summary of what steps are required to implement this integration strategy:

*Step 1*. Review this document to understand the data you need to send to our API.

*Step 2*. Create the travel insurance quoting and enrolment pages based on the information needed by our web services. The majority of this information, if not all, is typically collected by the normal purchase pages for a travel booking.

*Step 3.* Add validation logic to the pages to check for valid enrollment data.

See Appendix D: TIS API Exception Handling for a list of the validation logic that needs to be added.

Examples of validation logic include, but are not limited to:

* Maximum trip cost limit
* Maximum trip length
* Valid e-mail address
* Coverage options / plan type

*Step 4.*

a. Add the TSEP JavaScript to your credit card payment page. This code will pass cardholder information to TIS’ card processor directly from the user’s browser and return a token that will be used as payment within the CreatePolicy052018 service call.

b. Create the code to access our web service API.

c. Unit test the logic with common business scenarios.

d. Create and execute a test plan that includes execution of typical scenarios.

The following scenarios are an example of test scenarios that were part of an actual test plan:

|  |  |
| --- | --- |
| **Scenario** | **Test Group** |
| 1 | All products. |
| 2 | Individual coverage options for each product. |
| 3 | Additional coverages (combination of coverages) for each product |
| 4 | Scenarios where there are a) one traveler, b) more than one traveler, c) minors traveling with adults |
| 5 | Scenarios where there are multiple trip costs, including trip costs greater than the maximum trip cost allowed for the product. |
| 6 | Trip length is within maximum trip length limit for each product. |

Invalid scenarios must be checked by your web site and disallowed before sending through order. The following scenarios are an example of test cases that must be attempted, checked and handled:

* Exceeding maximum trip length.
* Exceeding maximum trip cost.
* Invalid Email address

See Appendix D: TIS API Exception Handling for a list of the validation logic that needs to be added.

The test plan should include a results table that can easily be verified by your team and by TIS during the formal acceptance phase. A sample test table is shown below:

| Test | Web Service used | Product | Coverage Type | Traveler DOB | Trip Cost | Depart Date | Return Date | Policy Cost | Email Address |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | CreatePolicy052018 | TBB | TC,ADD | 1/1/1970  1/1/1975 | 1000  500 | 12/1/2020 | 12/30/2020 |  |  |
| 2 | CreatePolicy052018 | TSB | TC,ADVS,MU | 1/1/1970 | 1000 | 12/1/2020 | 12/30/2020 |  |  |
| 3 | GetRate052005 | TBB | TC | 1/1/1955 | 1000 | 12/5/2020 | 12/15/2020 |  |  |
| 4 | GetRate052005 | TSB | TC,CFR | 1/1/1973 | 700 | 12/1/2020 | 12/30/2020 |  |  |

*Step 5.* Once sufficient testing has completed, initiate a formal acceptance phase with TIS. Contact Technology Solutions to begin the test phase. For the test phase, supply the static ip addresses of the computers from which you will be consuming the web services. During this setup phase, you will get your unique userid and password.

*Step 6.* Once testing has completed successfully, contact Technology Solutions for final business approval.

*Step 7*. Launch your new web site!

# Web Service Integration Overview

Travelex Insurance Services offers standard web services that allow for seamless integration between your web site and ours. The web services provide technology-independent integration capabilities that are loosely coupled.

The three web services currently offered and supported are:

1. **GetRate052005** – This web service returns a policy quote. Collect mandatory quoting data and call this web service to get a quote on the order.
2. **GetPaymentConfiguration** – This web service returns a TSYS URL and manifest for use with the TSEP JavaScript on your credit card payment page.
3. **CreatePolicy052018** – This web service is used for policy enrolment.
   1. Collect the details of the travel protection plan,
   2. verify the data,
   3. get a TSYS URL with manifest from TIS “GetPaymentConfiguration” method,
   4. include the URL and manifest within the TSEP JavaScript on the credit card payment page.
      1. Once the user enters their card information within the payment page, the JavaScript calls TSYS and receives a token that must be used as payment.
   5. Submit enrollment information with the token as payment by invoking this web service.

This service returns policy details including the policy confirmation number and benefit details. The returned policy details can be incorporated into a confirmation web page to the user.

See **Section 3.2 Cardholder Information Tokenization (TSEP JavaScript)** and **Appendix B: Tokenization Activity Diagram** for more information about credit card tokenization and TSEP.

## Technology Compatibility

The Travelex web-services were implemented with the Visual Studio.Net product. The web-services include support for:

* SOAP 1.1
* Form POST

The web-services API is described in the WSDL service description and can be found at <https://api.travelexinsurance.com/TIServices.asmx>

## Security Model

### User/Password/Location#

CreatePolicy052018 web service requires userid, password, and Location# to be passed into the API. These parameters are used for authentication in our system.

### TLS

All calls to the web-service API must be made over TLS 1.2.

SSL and early versions of TLS are not available due to PCI compliance requirements and data security concerns.

## Production / Test Environments

Two environments are available for use to aid in your use of the Travelex Web Service API.

### Test

The Test environment provides an area for partners to test their web service implementations before implementing in production.

The URL for web services and its service description (WSDL) in Test environment is:   
<https://api-test.travelexinsurance.com/TIServices.asmx>

An IP Address must be provided to Travelex for security purposes to hit the Travelex test environment.

### Production

The production environment has live data and must **not** be used during development or testing. This environment has TLS 1.2 implementation to secure your data.

The secured URL for web services and its WSDL in a Production environment is: <https://api.travelexinsurance.com/TIServices.asmx>

## Support

If you have questions on this API or are experiencing technical difficulties with this API, please call Business Solutions at 1–888-457-4602 option 4 for assistance.

Have the following information available when calling Business Solutions:

* Account Number
* Web-Service API used (e.g. CreatePolicy052018)
* Calling convention used (e.g. SOAP)
* Data being sent (parameter list)
* Date and Time when experiencing technical difficulty
* Error reported, if any.

# Web Service Details

This section gives detailed information about our API. This API includes the basic travel insurance products that are offered. For information about custom products, please contact the Technology Solutions team.

## Special Processing Rules

* Available to US Residents only.
* Policy documents, rates, and upgrades are traveller residency specific.
* Upgrades are not available for all states. Please see rating sheets.
* RVD is not available for travel to specific destinations: Ireland, Mexico, Jamaica, and Costa Rica.
* A Trip Cost of at least $1 must be insured for Travel Basic and Travel Select.
* **Kids Included Pricing for Travel Select only:** Kids age 17 and under are covered when accompanied by a covered adult. This does not apply to the cost of upgrades. Kids will be charged the appropriate per person rate for upgrades.
* Need to provide separate fields on your website to capture individual trip cost by traveller for quoting and purchase. Capturing a total trip cost that is split between travellers or one trip cost field for all travellers assuming each is the same is not accepted.
* A valid email address is required as the Confirmation of Coverage (COC) will be delivered via email only.

## Cardholder Information Tokenization (TSEP JavaScript)

**Overview**

“TSYS Secur-ePayment (TSEP) is a hosted field payment solution that helps developers provide additional security to a merchant’s website. They are plug-in fields in the merchant’s e-Commerce web site that can only communicate with the TSYS Host using field level encryption and returns a TSYS token. TSEP fields are customized, secure, input elements. When a user enters a card number in this field, the card number is converted into a token. The token can be used to send a payment request, instead of using the credit card number.” (TSYS Acquiring Solutions, 2018)

### Implementation Steps

1. Add the script tag to the <head> of the html credit card payment page. The src attribute noted below as “*[TSYS URL including Manifest]”* is obtained by using the TIS **GetPaymentConfiguration** method.  
   <script src="*[TSYS URL including Manifest]*"></script>

**Sample**

<script src="<https://stagegw.transnox.com/transit-tsep-web/jsView/00000000001201?ABCDEF01234567890987654321FEDCBAABCDEF01234567890987654321FEDCBA>"></script>

1. Add the function tsepHandler() to the <head> of the html webpage. This function needs to be used to handle various events from the TransIT host  
     
   function tsepHandler(eventType, event){  
   }
2. Add the div tags to the body of the webpage.  
     
   <div id="tsep-cardNumDiv"></div> (mandatory)

<div id="tsep-datepickerDiv"></div> (mandatory)

<div id="tsep-cvv2Div"></div> (optional)

**Sample**

<html>

<head>

<script src="<https://stagegw.transnox.com/transit-tsep-web/jsView/00000000001201?ABCDEF01234567890987654321FEDCBAABCDEF01234567890987654321FEDCBA>"></script>

<script>

function tsepHandler(eventType, event){

}

</script>

</head>

<body>

<form method="POST" id="payment-form">

<div id="tsep-cardNumDiv"></div>

<div id="tsep-datepickerDiv"></div>

<div id="tsep-cvv2Div"></div>

</form>

</body>

</html>

### Sessions

TSEP functionality is session based. Once the fields are displayed and once the request for the token is made by those fields, the session is no longer valid. A success or decline response is then received. A new request call needs to be made if there is a timeout.

The client application needs to be configured in a way that once the request for the token is made, a new call is made to get the new TSEP (PAN) fields, which would then be a new session. If a token request has already been made a response of HideTSEP001 is received.

**Session Error**

Indicates TSEP fields have already been used to make a request for a token. A new request for the TSEP fields needs to be made for the next attempt.

{"responseCode":"TSEP001","status":"FAILURE","message":"Encryption Failure","tsepToken":null,"maskedCardNumber":null,"cardType":null,"transactionID":null,"expirationDate":null,"cvv2":null}

### Event Handling

See Appendix D: TSEP Event Handling.

## GetRate052005

This method returns a quote for a Travelex policy. The production service description can be found at: <https://api.travelexinsurance.com/TIServices.asmx?op=GetRate052005>

The URL for web services and its service description (WSDL) in the Test environment is: <https://api-test.travelexinsurance.com/TIServices.asmx>

**For more information on each parameter, see Appendix C: Parameters.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Data Type** | **Format** | **Example** |
| strLocation \* | A unique code to specify the agency through which the sale was made. | String | Variable Characters Max - 7 | 00-0000 |
| strDepartureDate \* | Trip departure date | String | Date -mm/dd/yyyy | 01/03/2020 |
| strReturnDate \* | Trip return date | String | Date -mm/dd/yyyy | 01/05/2020 |
| strProduct \* | Special code that identifies the travel insurance product purchased. | String | 3 to 5 Characters with the following constraints:  STB – Travel Basic  STS – Travel Select  FIB – Flight Insure  FIPB – Flight Insure Plus | STB |
| strForm \* | Product Form Number, a special code that identifies the version of the product. | String | Variable Characters  Max – 25  TBB-1117 Travel Basic TSB-1117 Travel Select FIB-1117 Flight Insure FIPB-1117 Flight Insure Plus | TBB-1117 |
| strCoverageType \* | Special code that identifies the specific travel insurance coverage(s) selected.  Note: The following coverages can be combined with a comma and no spaces:  CFR, MU, ADD, ADVS  See Available Coverages on the right.  Coverage is not available for all states, please check Rate Sheet for details | String | **Available Coverages:**  **Travel Basic**  TC = Trip Cancellation  ADD= AD&D Common – Carrier Air Only  **Travel Select:**  TC = Trip Cancellation  CFR = Cancel for Any Reason  MU= Medical Upgrade  ADD= AD&D Common – Carrier Air Only  ADVS= Adventure Sports  **Flight Insure / Plus**  PK = Flight Insure Plus  FO = Flight Insure | TC,ADD  TC,ADVS  PK |
| IntFacPremium\* | The Flight Accident Coverage Premium amount.  Not applicable for STB or TSB | Integer | Number | 300000 |
| intNumTravelers \* | Total number of travelers for the policy (1 – 10) | Integer | Non-Negative Number | 2 |
| strDOB()\* | An array of the traveler DOB. First value is for the Primary Traveler | String | Date -mm/dd/yyyy | 05/05/1965 |
| intTripCost()\* | An array of the trip costs for travelers. First value is for the Primary Traveler | Integer | Money | 1000 |
| strState \* | State or Province for primary traveler's address.  This must match a fixed set of values defined by Travelex. | String | 2 Characters.  See Appendix A | NE |
| strCountry \* | Country of residence. This must match a fixed set of values defined by Travelex. | String | See Appendix A | United States |
| strCollisionStartDate() | An array of start dates for each rental vehicle damage. Can include any multiple of days within travel dates. | String | Date -mm/dd/yyyy | 01/03/2020 |
| strCollisionEndDate() | An array of end dates for each rental vehicle damage. Can include any multiple of days within travel dates. | String | Date -mm/dd/yyyy | 01/05/2020 |

**\*** These parameters are mandatory to get a policy rate. The rest can be passed as an empty string (“”) or integer zero (0) for String and Integer parameters respectively.

## GetPaymentConfiguration

This method returns a TSYS URL and Manifest for use with TSEP JavaScript on your credit card payment page.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Description** | **Data Type** | **Format** | **Example** |
| Location \* | A unique code to specify the agency through which the sale was made. | String | Variable Characters  Max - 7 | 00-0000 |
| UserID \* | Unique UserID supplied by Travelex Insurance. | String | Variable Characters  Max - 15 | myusername |
| Password \* | Unique password supplied by Travelex Insurance. | String | Variable Characters  Max - 150 | mypassword |
| ProductFormNumber\* | Special code that identifies the travel insurance product purchased. | String | Variable Characters  Max – 25 | TBB-1117 |
| HostUrl | The url from which the TSEP call will be made. | String | Variable Characters  Max – 50 | Company.com |

## CreatePolicy052018

This method creates a Travelex policy and returns Policy# and other important Policy details. The service description can be found at

<https://api.travelexinsurance.com/TIServices.asmx?op=CreatePolicy052018>

The URL for web services and its service description (WSDL) in Test environment is: <https://api-test.travelexinsurance.com/TIServices.asmx>

**For more information on each parameter, see Appendix C: Parameters.**

| **Name** | **Description** | **Data Type** | **Format** | **Example** |
| --- | --- | --- | --- | --- |
| strUserID \* | Unique UserID supplied by Travelex | String | Variable Characters Max - 15 | myusername |
| strPassword \* | Unique password supplied by Travelex | String | Variable Characters Max - 150 | mypassword |
| strBrokerLocation | If applicable, a unique code to specify the broker of the sale. | String | Variable Characters Max - 7 | 00-0000 |
| strAgencyLocation \* | A unique code to specify the agency through which the sale was made. | String | Variable Characters Max - 7 | 00-0000 |
| strAgentCode | A unique code to specify the agent through which the sale was made. | String | Variable Characters Max - 10 | Agent99 |
| strGroupID | Not applicable. | String | Empty string |  |
| strInvoiceNumber | Invoice Number created off partner system | String | Variable Characters Max - 15 | 1234567 |
| dblTotalPolicyCost | Not applicable. | String | Enter Zero | 0 |
| strDepartureDate \* | Trip departure date | String | Date -mm/dd/yyyy | 03/20/2020 |
| strReturnDate \* | Trip return date | String | Date -mm/dd/yyyy | 04/21/2020 |
| strPurchaseDate | Policy purchase date | String | Date -mm/dd/yyyy | 02/19/2020 |
| strProduct \* | Special code that identifies the travel insurance product purchased. | String | 3 to 5 Characters with the following constraints:  STB – Travel Basic  STS – Travel Select  FIB – Flight Insure  FIPB – Flight Insure Plus | STB |
| strForm \* | Product Form Number, a special code that identifies the version of the product. | String | Variable Characters Max - 25 | TBB-1117 |
| strCoverageType \* | Special code that identifies the specific travel insurance coverage(s) selected.  Note: The following coverages can be combined with a comma and no spaces:  CFR, MU, ADD, ADVS  See Available Coverages on the right.  Coverage is not available for all states, please check Rate Sheet for details | String | **Available Coverages:**  **Travel Basic**  TC = Trip Cancellation  ADD= AD&D Common – Carrier Air Only  **Travel Select:**  TC = Trip Cancellation  CFR = Cancel for Any Reason  MU= Medical Upgrade  ADD= AD&D Common – Carrier Air Only  ADVS= Adventure Sports  **Flight Insure / Plus**  PK = Flight Insure Plus  FO = Flight Only | TC,ADD  TC,ADVS  PK |
| intFacPremium | The Flight Accident Coverage Premium amount.  Not applicable for STB and STS | Integer | Money | 300000 |
| intNumTravelers \* | Total number of travelers for the policy (1-10) | Integer | Non-Negative Number | 2 |
| strFirstName() \* | An array of the first names of each traveler. First value is for the Primary Traveler | String | Variable Characters Max - 50 | Don |
| strLastName() \* | An array of the last names of each traveler. First value is for the Primary Traveler | String | Variable Characters Max - 50 | Traveler |
| strDOB() \* | An array of the traveler DOB. First value is for the Primary Traveler | String | Date - mm/dd/yyyy | 05/27/1970 |
| tripCost () \* | An array of the trip costs for travelers. First value is for the Primary Traveler | Decimal | Money | 1200 or 1200.00 |
| strAddress1 \* | Address line 1 for the primary traveler's address | String | Variable Characters  Max - 255 | 1111 Smith Avenue |
| strAddress2 | Address line 2 for the primary traveler's address | String | Variable Characters  Max - 255 | Apt #232 |
| strCity \* | City for primary traveler's address | String | Variable Characters Max - 255 | York |
| strState \* | State or Province for primary traveler's address  This must match a fixed set of values defined by Travelex. | String | 2 Characters.  See Appendix A for State Codes | NE |
| strZip \* | Zip code for primary traveler's address | String | 5 or 9 digit zip code. Format: ##### or #####-#### | 68136 |
| strCountry \* | Country of residence for primary traveler.  This must match a fixed set of values defined by Travelex. | String | See Appendix A | United States |
| strPhone \* | Phone number for primary traveler | String | (xxx)xxx-xxxx | (402)653-9871 |
| strFax | Fax number for primary traveler | String | (xxx)xxx-xxxx | (402)653-9870 |
| strEmail \* | Email address for primary traveler | String | A valid email address | don@yahoo.com |
| strCollisionStartDate() | An array of start dates for each rental vehicle damage. Can include any multiple travel periods within travel dates. | String | Date - mm/dd/yyyy | 01/03/2020 |
| strCollisionEndDate() | An array of end dates for each rental vehicle damage. Can include any multiple travel periods within travel dates. | String | Date - mm/dd/yyyy | 01/05/2020 |
| strBeneficiary | Any beneficiary information for the policy | String | Variable Characters Max – 1000 | This is the beneficiary detail |
| strPaymentType \* | The payment type code specifies how the policy was paid. | String | Variable Characters:  VI = Visa  MC = MasterCard  DS = Discover  AM = American Express  ACC = On Account | VI |
| tokenCCNumber \* | Token from TSEP JavaScript call to TSYS. | String | Valid token | Heuu8AS5yjm20762 |
| strMaskedCardNumber | The last 4 digits from the credit card. | String | String length 4 digits for Externally Authorized credit cards (i.e. card authorized outside of TIS systems).  Enter Empty String (“”) for all others. | 4111 |
| strCCCardHolderName \* | Credit card holders name | String | Variable Characters Max - 40 | Don Traveler |
| strCCExpirationMonth \* | Expiration month | String | 2 digit month | 04 |
| strCCExpirationYear \* | Expiration year | String | 4 digit year | 2021 |
| strCCAuthorizationNumber | Authorization number for externally authorized credit cards (i.e. card authorized outside of TIS systems) | String | Authorization number if the card was externally authorized, otherwise send and Empty string (“”). | 213658 |
| strExternallyAuthorized | Indicates the credit card was externally authorized (i.e. card authorized outside of TIS systems) | String | “EA” for externally authorized credit cards, otherwise send and Empty string (“”). | EA |
| strCheckNumber | Not applicable. | String | Empty string |  |
| strCruiseLine | Cruise line. Defaults to "none". | String | This must match a fixed set of values defined by Travelex. See Appendix A. | Orient Lines |
| strTourOperator | Tour-Operator. Defaults to "none". | String | This must match a fixed set of values defined by Travelex. See Appendix A. | Abc Tours |
| strAirLine | Airline. Defaults to "none". | String | This must match a fixed set of values defined by Travelex. See Appendix A. | Air 2000 |
| strDestination | Trip destination. | String | This must match a fixed set of ISO alpha-3 country values. See Appendix A. | USA |
| strFlightNumber | Flight Number | String | Variable Characters Max- 100 | AZ12345 |
|  |  |  |  |  |

\* These parameters are mandatory to create a policy. The rest can be passed as an empty string (“”) or integer zero (0) for String and Integer parameters respectively. Please refer to Appendix C: Parameters for more information.

## Web Service Calling Methodologies

There are 2 calling methodologies for web services:

### Form POST

Sending a Form POST request simply requires all parameters containing the name/value pairs in a form to be submitted by the method ‘Post.

### SOAP 1.1

Sending a SOAP POST request is the recommended way of calling web services. SOAP is a lightweight protocol for exchange of information in a decentralized, distributed environment. It is an XML based protocol that consists of three parts: an envelope that defines a framework for describing what is in a message and how to process it, a set of encoding rules for expressing instances of application-defined data types, and a convention for representing remote procedure calls and responses. This methodology also allows to do better error handling using SOAP fault tag.

Please visit [https://api.travelexinsurance.com/TIServices.asmx](https://api.travelexinsurance.com/TIServices.asmx?op=CreatePolicy052018) for additional online documentation and examples of how to invoke web-services.

## Error Handling

SOAP handling is the preferred mechanism for interacting with web services APIs as better error handling can be performed using the SOAP fault tag.

Any errors returned by TIS web service will start with a string “Error”.

# Appendix A: Constants

The following lists are to be used for the dropdown lists on your web site.

## Destinations

Please see a list of constants in the file “**travelex\_supported\_destinations.txt**”

## Cruise Lines

Please see a list of constants in the file “**travelex\_supported\_cruiselines.txt**”

## Tour Operators

Please see a list of constants in the file “**travelex\_supported\_touroperators.txt**”

## Airlines

Please see a list of constants in the file “**travelex\_supported\_airlines.txt**”

## US States

Please see a list of constants in the file “**travelex\_supported\_US\_states.txt**”

# Appendix B: Tokenization Activity Diagram

The Tokenization diagram below is available as a separate document upon request.



# Appendix C: Parameters

This table summarizes the CreatePolicy052018 web service API with examples of values for the multiple travel insurance plan types. It gives an indication of whether or not the field value is always required (mandatory), never required (optional), or whether other factors determine if it is required or not (varies).

**Note**: **Optional** parameter means a value is not forced but a minimum value is required to consume an API.

For string parameters it is an empty string (“”). For integer/double parameters, it is an integer zero (0).

| **Parameter Name** | **Description** | **Travel Basic** | **Travel Select** | **Flight Insure** | **Flight Insure Plus** |
| --- | --- | --- | --- | --- | --- |
| strUserID | User Name | Mandatory | Mandatory | Mandatory | Mandatory |
| strPassword | Password | Mandatory | Mandatory | Mandatory | Mandatory |
| strBrokerLocation | Broker location ID | Optional | Optional | Optional | Optional |
| strAgencyLocation | Agency location ID | Mandatory  (##-####) | Mandatory  (##-####) | Mandatory  (##-####) | Mandatory  (##-####) |
| strAgentCode | Agent Code | Optional | Optional | Optional | Optional |
| strGroupID | Group ID | N/A  Enter empty string (“”) | N/A  Enter empty string (“”) | N/A  Enter empty string (“”) | N/A  Enter empty string (“”) |
| strInvoiceNumber | Invoice Number off Partner System | Optional | Optional | Optional | Optional |
| dblTotalPolicyCost | Total policy cost.  Minimum value entered should be 0 | N/A  Enter Zero (0) | N/A  Enter Zero (0) | N/A  Enter Zero (0) | N/A  Enter Zero (0) |
| strDepartureDate | Trip departure date | Mandatory | Mandatory | Mandatory | Mandatory |
| strReturnDate | Trip return date | Mandatory | Mandatory | Mandatory | Mandatory |
| strPurchaseDate | Policy purchase date | Optional | Optional | Optional | Optional |
| strProduct | Product type | STB | STS | FIB | FIPB |
| strForm | Product Form Number, a special code that identifies the version of the product. | TBB-1117 | TSB-1117 | FIB-1117 | FIPB-1117 |
| strCoverageType | Coverage type code  Note: Upgrades are not available for all states | TC,  ADD | TC,  CFR,  MU,  ADD,  ADVS | FO | PK |
| intFacPremium | The Flight Accident Coverage Premium amount | N/A  Enter Zero (0) | N/A  Enter Zero (0) | 300000  500000  1000000 | 300000  500000  1000000 |
| intNumTravelers | Total number of travelers for the policy | Mandatory,  Numerical value > 0 | Mandatory,  Numerical value > 0 | Mandatory  Numerical value  > 0 | Mandatory,  Numerical value  > 0 |
| strFirstName() | String array of the first names of each traveler. First value is for the Primary Traveler | Mandatory | Mandatory | Mandatory | Mandatory |
| strLastName() | String array of the last names of each traveler. First value is for the Primary Traveler | Mandatory | Mandatory | Mandatory | Mandatory |
| strDOB() | String array of the traveler DOB. First value is for the Primary Traveler | Mandatory | Mandatory | Mandatory | Mandatory |
| tripCost() | Decimal array of the trip costs for travelers. First value is for the Primary Traveler | Mandatory | Mandatory | Mandatory | Mandatory |
| strMedUpgrade() | String array of medical-upgrades for travelers. First value is for the Primary Traveler | N/A  Enter empty string (“”) | N/A  Enter empty string (“”) | N/A  Enter empty  String (“”) | N/A  Enter empty string (“”) |
| strAddress1 | Address line 1 for the primary traveler's address | Mandatory | Mandatory | Mandatory | Mandatory |
| strAddress2 | Address line 2 for the primary traveler's address | Optional | Optional | Optional | Optional |
| strCity | City for primary traveler's address | Mandatory | Mandatory | Mandatory | Mandatory |
| strState | Postal state code for primary traveler's address | Mandatory | Mandatory | Mandatory | Mandatory |
| strZip | Zip code for primary traveler's address | Mandatory | Mandatory | Mandatory | Mandatory |
| strCountry | Country of residence for primary traveler | Mandatory | Mandatory | Mandatory | Mandatory |
| strPhone | Phone number for primary traveler | Mandatory | Mandatory | Mandatory | Mandatory |
| strFax | Fax number for primary traveler | Optional | Optional | Optional | Optional |
| strEmail | email address for primary traveler | Mandatory | Mandatory | Mandatory | Mandatory |
| strCollisionStartDate() | An array of start dates for each rental vehicle damage. Can include any multiple travel periods within the travel dates. | Date - mm/dd/yyyy | Date - mm/dd/yyyy | N/A  Enter empty  String (“”) | N/A  Enter empty string (“”) |
| strCollisionEndDate() | String array of end dates for each rental vehicle damage. Can include any multiple travel periods within the travel dates. | Date - mm/dd/yyyy | Date - mm/dd/yyyy | N/A  Enter empty  String (“”) | N/A  Enter empty string (“”) |
| strBeneficiary | Any beneficiary information for the policy | Optional | Optional | Optional | Optional |
| strPaymentType | The payment type code | Mandatory | Mandatory | Mandatory | Mandatory |
| tokenCCNumber\* | Token from TSEP JavaScript call to TSYS. | N/A Enter Empty String (“”) for Externally Authorized credit cards.  Mandatory for all others | N/A Enter Empty String (“”) for Externally Authorized credit cards.  Mandatory for all others | N/A Enter Empty String (“”) for Externally Authorized credit cards.  Mandatory for all others | N/A Enter Empty String (“”) for Externally Authorized credit cards.  Mandatory for all others |
| strMaskedCardNumber\* | The last 4 digits of the credit card number. | Mandatory for Externally Authorized credit cards.  N/A Enter Empty String (“”) for all others. | Mandatory for Externally Authorized credit cards.  N/A Enter Empty String (“”) for all others. | Mandatory for Externally Authorized credit cards.  N/A Enter Empty String (“”) for all others. | Mandatory for Externally Authorized credit cards.  N/A Enter Empty String (“”) for all others. |
| strCCCardHolderName | Credit card holders name | Mandatory | Mandatory | Mandatory | Mandatory |
| strCCExpirationMonth | Expiration month | Mandatory | Mandatory | Mandatory | Mandatory |
| strCCExpirationYear | Expiration year | Mandatory | Mandatory | Mandatory | Mandatory |
| strCCAuthorizationNumber | Credit card authorization number. Mandatory if credit card payment type is Externally Authorized. | Optional | Optional | Optional | Optional |
| strExternallyAuthorized | Is Credit-card Externally Authorized? | Optional | Optional | Optional | Optional |
| strCheckNumber | Check Number | Optional | Optional | Optional | Optional |
| strCruiseLine | Cruise line. | Optional | Optional | Optional | Optional |
| strTourOperator | Tour-Operator | Optional | Optional | Optional | Optional |
| strAirLine | Airline | Optional | Optional | Optional | Optional |
| strDestination | Trip destination. | Required | Required | Required | Required |
| strFlightNumber | Flight Number | Optional | Optional | Optional | Optional |

# Appendix D: TIS API Exception Handling

There are four categories of exceptions:

* Formatting / Field Validation Errors  
  These errors are formatting related errors that need to be caught before submitting the data to the Travelex web service.
* Business Rule Validation Errors  
  These errors are business rules that need to be caught before submitting the data to the Travelex web service.
* Operational Errors  
  These errors are normal errors that can occur during order processing and need to be handled by your system.
* Typical Web Service Integration Setup Errors  
  These errors are typical when working with our web services for the first time and should not be received after working through the initial setup.

## Formatting / Field Validation Errors

| **Error Message** | **Description** | **Solution** |
| --- | --- | --- |
| "Coverage Type is not valid." | The strCoverageType parameter does not match any Coverage. | For the product, enter the valid coverage type values. |
| "Expiration date is invalid."  “Expiration date month is invalid.”  “Expiration date year is invalid.” | Invalid credit-card expiration date. | Combination of strCCExpirationMonth and strCCExpirationYear parameters  Expiration month should be 2 digits (e.g. 09), Expiration year should be 4 digits (e.g. 2010) |
| "You have passed an invalid Product." | The strProduct parameter does not match any product. | Pass a valid product in the strProduct parameter.  Example: TBB |
| "You have passed an invalid Coverage." | The strCoverageType parameter does not match any Coverage types. | Pass a valid coverage type for the product in the strCoverageType parameter. |
| "City is required." | Empty string is passed for City | strCity parameter |
| "State is not valid." | Invalid State passed for the Country used. | Combination of strState and strCountry parameters |
| "Zip Code is required." | Empty string is passed for Zip | strZip parameter |
| “Card holder name is required." | (self explanatory) | strCCCardHolderName parameter |
| "Invalid Date of Birth." | Traveler date-of-birth is invalid. | strDOB parameter. Use this format: mm/dd/yyyy |

## Business Rule Validation Errors

The following error messages reflect business rule validation for the offered travel insurance products. These exceptions need to be caught on your system before submitting the data to the Travelex web service.

| **Error Message** | **Description** | **Typical Cause / Solution** |
| --- | --- | --- |
| "Departure date must be before Return date." | (self explanatory) | Combination of strDepartureDate and strReturnDate parameters |
| "Departure date must be today or greater. Return date must be on or after departure date." | (self explanatory) | Combination of strDepartureDate and strReturnDate parameters |
| "Trip Length exceeds maximum number of days for the selected product. Maximum Trip Length for product - [ProductName]: [MaxDayCutoff], Departure Date: [DepartureDate] Return Date: [ReturnDate] | (self explanatory) | Combination of strProduct, strDepartureDate and strReturnDate parameters |
| "Coverage type is not valid for this product." | (see above) | Combination of strProduct and StrCoverageType parameters |
| "You have passed an Invalid Location Number." | Invalid Location Number passed for your agency | strAgencyLocation or strBrokerLocation parameter |
| “You are currently not configured to sell this product. Please contact our Customer Service Department at 1-888-457-4602." | The agency with the Location# passed is not configured to sell the Product passed. | Combination of strProduct and (strLocation or strAgencyLocation or strBrokerLocation) parameters |
| "Trip-cost exceeds the maximum limit of [MaxTripCost]" | TripCost passed is greater than the maximum Tripcost for the Product used. | Combination of strProduct and intTripCost parameters |
| Invalid Payment Type. | (self explanatory) | strPaymentType |
| "You are currently not configured to use [payment-type] payment type." | (self explanatory) | strPaymentType |

## Operational Errors

|  |  |  |
| --- | --- | --- |
| **Error Message** | **Description** | **Solution** |
| "Error: Credit card was declined. Authorization failed." | An unexpected error occurred when attempting to authorize the credit card. This is the message returned from our credit card processor and can be for the following reasons:   * Insufficient funds. * Invalid credit card number. * Invalid expiration date for the credit card number. | * Combination of strCCNumber , strCCExpirationMonth and strCCExpirationYear parameters * Collect different credit card information and re-submit the order. |
| "Error: The system is experiencing a heavy load now and cannot complete the operation. Please try again." | An unexpected system error occurred during the processing of the policy order. | * Re-submit the order at a later time. |
| “Error: Cannot find server”  When server is down, you will get no reply. | The server is currently unavailable due to maintenance. | * Re-submit the order at a later time. |

## Typical Web Service Integration Setup Errors

| **Error Message** | **Description** | **Typical Causes / Solution** |
| --- | --- | --- |
| <?xml version="1.0" encoding="utf-8" ?>  <string xmlns="**http://www.travelex-insurance.com/**"> **Error: Login failed.** </string> | The userid, password combination is not valid or setup for the location number. | * Verify login data sent.   OR   * Verify that login information has been set up in test environment. |
| <?xml version="1.0" encoding="utf-8" ?>  <string xmlns="**http://www.travelex-insurance.com/**"> **Error: Object reference not set to an instance of an object.** </string> | An unexpected system error occurs when processing the order. | Typical Causes:   * Mandatory fields do not contain data. * Optional fields contain invalid data. * Missing fields.   Solution:   * Verify that all mandatory fields have data. * Verify that all optional fields have the correct null data. Integer fields must have a value. String fields can be blank. * Verify that there are no missing fields. |
| <?xml version="1.0" encoding="utf-8" ?>  <string xmlns="**http://www.travelex-insurance.com/**">**Error: writing policy exception: |Index was outside the bounds of the array.**</string> | An unexpected system error occurs when processing an array in the API. | Typical Causes:   * The array for traveler information is improperly formed. * The array for RVD information is improperly formed.   Solution:   * Verify that array is properly formed. |
| 403: Access Forbidden | Access has not been granted by the TIS Test Environment for the IP address of the computer that is attempting to communicate with it. | * Contact TIS Business Solutions with the ip address to be permitted access to the test environment. |

# Appendix D: TSEP Event Handling

The response from the TransIT host are events. All the events will be sent to "tsepHandler()" as input parameters. The first parameter is *eventType* and is a string indicating the type of event. Second parameter is the actual event object.

Each event type will have an attribute and will help determine the exact event and its description. Possible events generated are:

|  |  |  |  |
| --- | --- | --- | --- |
| **Event type** | **Description** | **Attribute** | **Value** |
| FocusOnEvent | The Focus On event occurs when a TSEP field gets focus. | fieldName | **event.fieldName** – The name of the TSEP field where the event occurred.  Possible values:   * tsep-cardNum * tsep-datepicker * tsep-cvv2 |
| FocusOutEvent | The Focus Out event occurs when a TSEP field is about to lose focus. | fieldName | **event.fieldName** – The name of the TSEP field where the event occurred.  Possible values:   * tsep-cardNum * tsep-datepicker * tsep-cvv2 |
| BlurEvent | The Blur event occurs when a TSEP field loses focus. | fieldName | **event.fieldName** – The name of the TSEP field where the event occurred.  Possible values:   * tsep-cardNum * tsep-datepicker * tsep-cvv2 |
| FieldValidationErrorEvent | Field Validation Error event will occur when any of the TSEP fields fail required validation. | responseCode  status  message  fieldName | **event.responseCode** – The response code of the field validation failure.  Possible values:   * [TSEPERR101](javascript:toggleBlock('129858'))   Length of the CVV is incorrect. Allowable length: Min -3 , Max -4  {"responseCode":"TSEPERR101","status":"FAIL","message":"CVV must be at least 3 characters long","fieldName":"tsep-cvv2"}   * [TSEPERR102](javascript:toggleBlock('129863'))   The date entered is not in MM/YYYY format.  {"responseCode":"TSEPERR102","status":"FAIL","message":"Date must be in the form MM/YYYY","fieldName":"tsep-datepicker"}   * [TSEPERR103](javascript:toggleBlock('129864'))   The date entered is in past.  {"responseCode":"TSEPERR103","status":"FAIL","message":"Expiry date cannot be in the past","fieldName":"tsep-datepicker"}   * [TSEPERR104](javascript:toggleBlock('129865'))   An invalid month value is in the date field.  {"responseCode":"TSEPERR104","status":"FAIL","message":"Expiry date month can only be betweeen 1 to 12","fieldName":"tsep-datepicker"}   * [TSEPERR105](javascript:toggleBlock('129866'))   The card number entered is less than 13 digits.  {"responseCode":"TSEPERR105","status":"FAIL","message":"CardNumber must be at least 13 characters long","fieldName":"tsep-cardNum"}  **event.status** – The status of the field validation. It will always have the value **FAIL**.  **event.message** – A brief description of the field validation error.  **event.fieldName** – The name of the TSEP field where the event occurred.  Possible values:   * tsep-cardNum * tsep-datepicker * tsep-cvv2 |
| ErrorEvent | Error event will occur when any error is generated by the host. | responseCode  status  message | **event.responseCode** – The response code of the host validation failure. It will always have the value [TSEPERR911](javascript:toggleBlock('129854')).   * Incorrect Manifest * Domain URL recorded for the user and the request domain does not match * Device ID send in the request is incorrect (not TSEP)     {"responseCode":"TSEPERR911","status":"FAIL","message":"Authentication Failed"}  **event.status** – The status of the host validation. It will always have the value **FAIL**.  **event.message** – A brief description of the host validation error. |
| TokenEvent | A Token event will occur when a token response is received from the host. | responseCode  status  message  tsepToken  maskedCardNumber  cardType  transactionID  expirationDate  cvv2 | **event.responseCode** – The response code of the tokenization request. A successful token creation will always have the value **A0000**. Any other code indicates a failure.  **event.status** – The status of the token request.  Possible values:   * PASS * FAIL * FAILURE   **event.message** – A brief description of the status of the token request.  **event.tsepToken** – A token for the card number entered, if the token request is successful.  **event.maskedCardNumber** – The last four digits of card number entered, if the token request is successful.  **event.cardType** – The card type of the card number entered, if the token request is successful.  Possible values:   * V - Visa * M - Mastercard * X - American Express * R - Discover, CUP, Diners * J - JCB   **event.transactionID** – The unique TSYS transaction ID of the request, if token request is successful.  **event.expirationDate** – The card expiration date entered by the card holder.  **event.cvv2** – The card CVV value entered by the card holder. |

# Appendix E: Webservice xml response

The CreatePolicy052018 webservice returns an xml response (like below) after successfully creating the policy in the TIS system:

