DELORME

Developer Guide for IPC Outbound

V 1.2
Updated: 10/21/2014

Contents

Revision History

Date	Version	Description	Author
2011-12-07	0.0.1	Draft Version Ready	EBS
2012-01-17	0.0.5	Added Revision History	EBS
2012-02-17	0.0.6	Change timestamp description from milliseconds to seconds	EBS
2012-03-09	1.0.0	Locked down the final revision of the interface	CAR
2012-04-04	1.0.1	Updated retry back off schema and added pre-canned messages.	EBS
2012-04-20	1.0.2	Edited	BG
2013-05-21	1.0.3	Added new message codes and examples	EBS, SBK
2013-06-11	1.1	Added pingback feature and updated examples	AJA
2013-08-05	1.1.1	Added message code for Canned messages	SBK
2014-06-11	1.1.2	Merged in “Device to Device Messaging” document	MDG
2014-07-30	1.1.3	Added GenericBinary message code	MDG
2014-10-21	1.2	Added EncryptedPinpoint message code	MDG

Introduction

IPC Outbound is an inReach data push service for professional and enterprise inReach customers. The service uses web standards to relay messages sent from an inReach to a web service. Each message is represented as an **Event** using an efficient JSON schema. JSON is a text based open standard for representing simple data structures. The messages are pushed to the web service using HTTP.

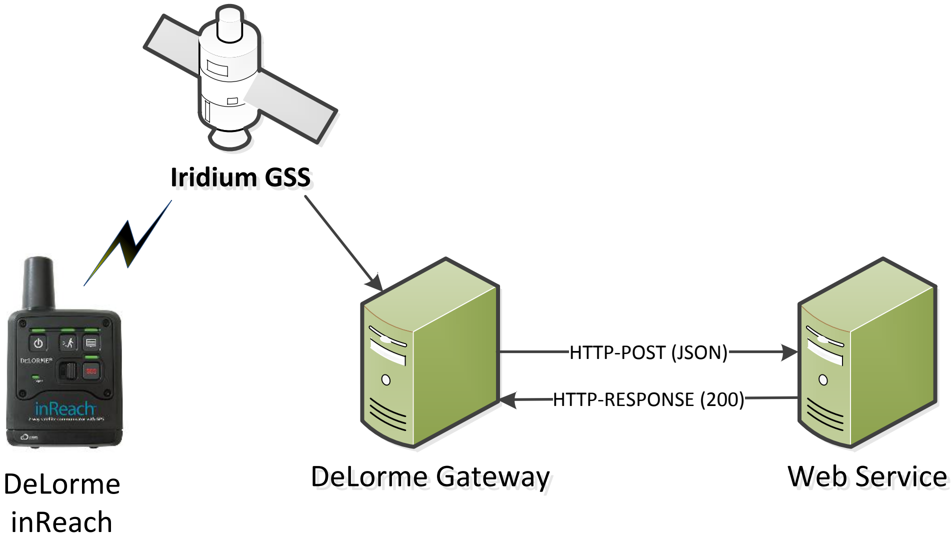
How It Works

The DeLorme data push service requires end users to setup a web service to handle incoming HTTP-POST requests from the DeLorme gateway. The post requests occur when messages are received from registered inReach devices. The messages are queued in the DeLorme gateway and then relayed to the web service. The relayed content contains information about the inReach message that was sent.

The relay period is near real-time. This means that as soon as the message has been received from the Iridium GSS (Gateway SBD Subsystem) it will be relayed to the end user's web service.

Every HTTP session should have a response from the web service. Successful sessions should respond with code 200. Anything else will be considered a failed delivery. When a failure occurs, the message is queued and then retried. Repeated failures will cause the system to wait longer between tries. Please see Failed Delivery for more information about the retry schema.

The image below is a simplified diagram of the request and expected response.



Security

All HTTP-POST data can be encrypted using HTTPS (Hypertext Transfer Protocol Secure). HTTPS is the use of HTTP over an encrypted connection. To use this service, the server administrator must create a signed public key certificate for the web server. The certificate should be signed by a trusted authority. All certificates are validated.

Failed Delivery

If the initial delivery attempt fails, the HTTP-POST will be retried. Subsequent failures will result in the service waiting longer between retries. The service will retry after 2, 4, 8, 16, 32, 64, and 128 seconds. Repeated attempts will continue every 128 seconds. During this time the messages will be queued. Upon success the service will reset to “near real-time.”

The system administrator can opt in to receive notifications when repeated failures occur by filling in the ECTC (Enterprise Customer Technical Contact) information in the enterprise.delorme.com site. The IPC Outbound service will send an email and/or SMS notification to the ECTC when the queue contains a message older than 2 hours or has exceeded 1000 messages.

Service Rules

IPC Outbound allows professional and enterprise customers the choice of whether or not to assign a device to a user. Special rules have been developed around the how services function when users are not assigned to devices. These rules include how message relay, device synchronization and GEOS work. Below is a table that defines which services are enabled when a user is assigned to the device.

User Assignment Rules

Service	Assigned	Not Assigned
Message Delivery	Enabled	Disabled
Device Synchronization	Enabled	Disabled
GEOS	Enabled	Disabled

Message Delivery Service

The DeLorme enterprise website has a robust set of services for sending and receiving SMS and SMTP messages to the recipients of an inReach message. Users can opt out of automatic message delivery by not assigning a device.

Message Synchronization

The DeLorme inReach utilizes an extremely optimized protocol for Over-the-Air communications. The optimizations compress the number of bytes used per message. A key feature of the protocol is the employment of synchronized data. Synchronizing data allows the protocol to encode a long free text message to 1 byte or less.

IPC Outbound can make synchronization transparent when a user is assigned to the device. The service will automatically make the translation from the encoded value to the contact, message or puck message button.

This feature can be disabled by not assigned a user to the device. With this configuration, IPC Outbound will not translate the encoded values. Instead, it will pass values in the appropriate fields. For example, instead of receiving an IPC Outbound message addressed to john.smith@email.com. The address will be the number 24. It is up to the receiving service to make the translation.

GEOS

Customers may opt to utilize GEOS Search and Rescue. Please note that a user must be assigned to the device to enable GEOS.

JSON Refresher

The IPC Outbound request (HTTP-POST) will contain content that describes a message using JSON. Below is a short refresher on the JSON data format.

Structures in JSON are either name/value pairs or an ordered list of values. A value can be a string, number, object, array, true, false or null. String values are wrapped in double quotation marks (“example”). Object values begin with a left brace ({) and end with a right brace (}). Arrays begin with a left square bracket ([) and end with a right square bracket (]).

Below is an example of JSON. The names are the value types. For example the value for **String** is the string, “This is a string”. The value of **Number** is the number 12345 (notice the missing quotation marks). The value of **ObjectArray** is an array of objects. Each object contains a **name** and **value**.

```
{
  "String" : "This is a string",
  "Number" : 12345,
  "Boolean" : true,
  "ObjectArray" : [
    { "name" : "one" },
    { "name " : "two" }
  ]
}
```

Please refer to RFC 4627 at http://tools.ietf.org/html/rfc4627 for more details.

Event Schema

The root object contains the string “Version” and an “Events” array of objects. The version is used to distinguish between different iterations of the object. Updates to the JSON schema will be reflected in the version string.

The “Events” array can contain one or more event objects from different inReach devices. An event is defined as a single transmission of data from an inReach to the DeLorme gateway. Each event is linked to a specific inReach device by the device's International Mobile Equipment Identity or IMEI. An IMEI is a unique 15-digit identifier for each inReach. In the example, there is one event from inReach device 100000000000001. Events contain information about telemetry, type, status, and free text. More descriptive definitions of the values in the event object can be found below.

Name	Type	Description
Version	String	The current schema version. Changes to this schema will be reflected in this name/value pair.
Events	Array	An array of event objects. An event is defined as a single transmission of data from an inReach to the DeLorme gateway.

The event object has the following structure.

Name	Type	Description
imei	String	The inReach IMEI. An IMEI is a number that uniquely identifies each inReach device. It is usually 15 digits. For this implementation it is expressed as a string.
messageCode	Number	The type of message. Definitions are in the Message Codes Table on the next page.
freeText	String	The text that was sent by the inReach. This can be empty if nothing was sent.
timeStamp	Number	The time that the message was created. This is in milliseconds since EPOCH (January 1, 1970).
pingbackReceived	Number	The time that the pingback request was received on IPCInbound. This is in milliseconds since EPOCH (January 1, 1970).
pingbackResponded	Number	The time that the pingback response was constructed on IPCOutbound. This is in milliseconds since EPOCH (January 1, 1970).
addresses	Array	An array of addresses.
point	Object	Describes the location of the message. Values are filled with 0 when there is no location information.
status	Object	inReach status information
payload	String	Base64 encoded binary data.

The address array contains objects with the following structure.

Name	Type	Description
address	String	An address can be a SMS phone number, email address or synchronized contact. Please see Message Synchronization for more information about synchronized contacts.

The point object contains the following structure.

Name	Type	Description
latitude	Number	The latitude that the message originated from in decimal degrees. Calculated using the WGS84 datum.
longitude	Number	The longitude that the message originated from in decimal degrees. Calculated using the WGS84 datum.
altitude	Number	The height in meters.

gpsFix	Number	No Fix = 0, 2D Fix = 1, 3D Fix = 2, 3D Fix+ = 3
course	Number	True Course over ground expressed in degrees.
speed	Number	Speed over ground in kilometers per hour.

The status object contains the following structure.

Name	Type	Description
autonomous	Number	A value of 1 indicates the message originated from an inReach. 0 indicates the message originated from a paired device and was then sent by the inReach.
lowBattery	Number	Indication that the inReach battery is below 20%. A value of 1 indicates that the inReach battery is low. Otherwise the value 0 means that the battery is not low.
intervalChange	Number	The change in the SOS or tracking interval in seconds. 0 indicates that the interval has not change. Anything greater than 0 indicates a change in interval. (600 = 10 minutes)

Message Codes Table

Code	Name	Description
0	Position Report	Drops a breadcrumb while tracking.
1	Reserved	Reserved for later use.
2	Locate Response	Position for a locate request.
3	Free Text Message	Message containing a free-text block.
4	Declare SOS	Declares an emergency state.
5	Reserved	Reserved for later use.
6	Confirm SOS	Confirms an unconfirmed SOS.
7	Cancel SOS	Stops a SOS event.
8	Reference Point	Shares a non-GPS location.
9	Check In	Response to a scheduled or unscheduled check-in.
10	Start Track	Begins a tracking process on the server.
11	Track Interval	Indicates changes in tracking interval.
12	Stop Track	Ends a tracking process on the server.
13	Unknown Index	Used when the device receives a message from the server addressed to a synced contact identifier that is not on the device.
14	Puck Message 1	Sends the first of three inReach message button events.
15	Puck Message 2	Sends the second of three inReach message button events.
16	Puck Message 3	Sends the third of three inReach message button events.
17	Map Share	Sends a message to the shared map.
20	Mail Check	Sent to determine if any messages are queued for the device.
21	Am I Alive	Sent when the device needs to determine if it is active. This message is automatically replied to by the DeLorme server to indicate the current status of the device.
24 - 63	Pre-defined Message	The index for a text message that is synchronized with the server.
64	Encrypted Binary	An encrypted binary Earthmate message.*
65	Pingback Message	A pingback response message (initiated through IPCInbound).
66	Generic Binary	An uninterpreted binary message.

67	EncryptedPinpoint	A fully-encrypted inReach message.*
3099	Canned Message	A Quicktext message, potentially edited by the user.

*Encryption options require the customer to supply its own gateway. For details, see https://files.delorme.com/support/inreachwebdocs/Encryption_Administration_Guide.pdf

Example Event

Below is an example of a single request. The request **Version** is 2.0 and the **Events** array contains one event.

The event is a text message, **messageCode 3**, with the message, “**On my way.**” The event was created on Tuesday, December 13, 2011 at 8:56AM UTC. The message is addressed to two recipients. The first is to the SMS phone number **2075752244** and the second is to the email address **support@delorme.com**.

The message originated from latitude **43.807** and longitude **-70.16355**. At which point the device was moving northeast at **50 kilometers per hour**.

The **lowBattery** status is 1. This indicates that the battery is below 20%.

```
{
   "Version": "2.0",
   "Events": [
      {
         "imei": "100000000000001",
         "messageCode":  3,
         "freeText":  "On my way.",
         "timeStamp":  1323784607376,
         "addresses": [
            { "address": "2075752244" },
            { "address": "support@delorme.com" }
         ],
         "point":  {
            "latitude":  43.8078653812408,
            "longitude":  -70.1636695861816,
            "altitude":  45,
            "gpsFix":  2,
            "course":  45,
            "speed":  50
         },
         "status":  {
            "autonomous":  0,
            "lowBattery":  1,
            "intervalChange":  0
         }
      }
   ]
}
```

Web Service

DeLorme uses HTTP-POST to relay inReach data traffic. You must set up a HTTP 1.1 web service to receive this data.

Request Content

The content of each HTTP-POST is a JSON object. The object describes one or more inReach events. The content type is application/json and the length of the content is equal to the number of bytes in the JSON object. Below are examples.

POST / xxxx.xxx HTTP/1.1
Content-Type: application/json
Content-Length: The length in bytes of the JSON object

Example Free Text Content

```
{"Version": "2.0", "Events": [{"imei":"100000000000001", "messageCode":3, "freeText":"Hello World",
"timeStamp": 1323784607376, "pingbackReceived":null,  "pingbackResponded":null, "addresses":[],
"point":{ "latitude":43.8078653812408, "longitude":-70.1636695861816, "altitude":45, "gpsFix":2,
"course":45, "speed":50 }, "status":{ "autonomous":0, "lowBattery":0 , "intervalChange":0}}]}
```

Example Binary Content

```
{"Version":"2.0","Events":[{"payload":"VGhlIGZvcmNlIGlzIHN0cm9uZyB3aXRoIHRoaXMgb25lLg==","imei
":"300234010571020","messageCode":64,"status":{"lowBattery":0}}]}
```

JSON Format Differences When Using Encrypted Messaging

If Encrypted Messaging has been enabled for the devices on an account, a different formatter is used to create the JSON data. With Encrypted Messaging enabled, *null* values are not transmitted, and any properties with *null* values will not appear in the object. When Encrypted Messaging is disabled, properties with *null* values will appear.

Response

The web service should respond to HTTP-POST sessions with a response code. A response code of 200 is considered a success. Anything else is considered a failure.

Status Code	Description
200	OK – The session was successful
400	Bad Request – The session was unsuccessful

Device to Device Messaging with IPC

An important aspect of IPC is to allow for messages to be sent between two inReach devices without requiring explore.delorme.com user accounts. A message is routed to another user's device when the recipient of a message is of the form `joe@inreach.yourdomain.com`

When the DeLorme gateway receives a message whose destination email is at the subdomain `inReach.yourdomain.com`, the message is forwarded via IPC Outbound to enterprise web service without sending an email. The enterprise customer's web service is then responsible for mapping the email address to an IMEI and using IPC Inbound to send the message to the recipient. If the email address cannot be mapped to an IMEI the web service should send a response to the sender informing them.

To facilitate this, the customer must enter the subdomain they wish to use into the inReach Portal Connect web page of explorer.delorme.com. It is recommended that format of the email address and the subdomain name be kept as short as reasonable.

The gateway will filter out messages addressed to a registered enterprise subdomain, and not send emails.