Ontario EBT Protocol
Between Points

Draft
Version 2.00.3

Published by:

Ontario EBT Work Group

Contents

1. Executive Summary

This document defines the Internet Data Transport protocol and rules for EBT transactions for communication between Points and other Points as defined by the Ontario Energy Board's EBT Work Group for the deregulated Electric marketplace in Ontario, Canada.

2. Revision History

Author	Version	Date	Description
Grant Comissiong/ Jeff Feinrip	0.1	October 10, 2001	Initial version
Grant Comissiong	0.2	October 18, 2001	Incorporate feedback from first draft
Jeff Feinrip	0.3	November 8, 2001	Incorporate feedback from Oct 30, 2001 Working Group meeting
Jeff Feinrip	2.0	November 27, 2001	Incorporate feedback from Nov 26, 2001 Working Group meeting

3. Introduction

This document defines the Internet Data Transport protocol and rules for EBT transactions for communication between sending Points and ~~other~~ recipient Points as defined by the Ontario Energy Board's EBT Work Group for the deregulated Electric marketplace in Ontario, Canada.

This document assumes the reader is familiar with the Ontario EBT Specifications, the Ontario Data Transport Protocol, public key encryption, and HTTP.

3.1 Scope

The EBT Point to Point Protocol defines the technical and functional standard for EBT messaging between sending Points and ~~other~~ recipient Points in the Ontario deregulated electric market. This includes Point to Point messages and the responsibilities of Points.

This document does not cover the messaging between a Hub and its subscriber, Hubs and other Hubs, or Point to Hub ~~connections .~~connections. This document does not cover the contents or format of PIPE Documents carrying consumer information.

The EBT Point to Point Protocol consists of the following parts:
- Transport Level protocol and connections;
- The delivery of Functional Acknowledgements;
- Time synchronisation and time stamping of transactions;
- XML Parsers.

3.2 Definitions

This document uses the following definitions:

EBT Message	A transport-protocol object that includes HTTP information and encapsulates one PIPE Document.
EBT Document	One XML instance document consisting of a PIPE Document, which in turn is made up of one or more EBT transactions.
PIPE	Partner Interface Protocol for Energy.
Point	\<To be defined\>
Sending Point	Point that sends a message
Recipient Point	Point that receives a message

3.3 Guiding Principles for this Protocol

The following principles were used as guidelines to develop the Ontario EBT Point to Point Protocol:

- The rules for EBT communication between retailers and LDCs are defined in the (EBT) Standards Document and the Ontario EBT Data Transport Protocol. All solution offerings must conform to these standards.
- Each recipient Point must provide an inbound mailbox for other sending Points to upload EBTs to. Andll Points must upload their outbound EBTs to destination mailboxes.
- In Point to Point communication, the Market Participant Point is always the originator or end destination of the EBT; all other EBTs not specific to that Point are rejected.

3.4 References

For additional information, please see the following documents:

- "Ontario EBT Data Transport Protocol", Ontario EBT Working Group;

- "Ontario EBT Hub to Hub Protocol", Ontario EBT Working Group;

- "Ontario EBT Point to Hub Protocol", Ontario EBT Working Group;

- "Ontario Electronic Business Transactions (EBT) Standards Document" (Business Rules Document), Ontario EBT Working Group.

4. Transport Level Protocol

This section defines the transport level issues for Point to Point connections.

4.1 Basic Transport Level Protocol

The communication from a Point to another Point extends the protocol as is used in Hub to Subscriber, ~~ ~~Hub to Hub, and Hub to Point communications. This Transport Protocol encompasses the:

- Security protocol;
- Message protocol; and
- Delivery protocol.

A Point must have a unique identifier (an OEB License Number within the EBT Message standard) and a user id and password for each Point in the market that it needs to connect to.

Note that the basic Transport Level Protocol is ~~an~~ end to end and therefore the EBT message will be encrypted by a ~~sender (Point One)~~sending Point for its ~~destination (Point Two)~~ recipient Point. For example, a Point sending an EBT message to another Point will encrypt the EBT message with the destination Point's public key and sign the message with its own private key.

4.2 Point to Point Push Communication Model

Each send~~inger~~ Point connects to ~~another~~ a recipient Point in order to deliver an EBT to its destination. A recipient Point must maintain an inbound mailbox in order for other ~~sender~~ sending Points to have a place to push EBT messages to. A send~~inger~~ Point must also deliver outbound messages to other Point's mailboxes.

~~Points will meet the same processing guidelines to be defined by the OEB~~The sending Point will not push documents more frequently than every 15 minutes. A recipient Point must be able to handle at least one concurrent connection for each Hub and each trading partner who have implemented the Point to Point protocol.

4.3 Store and Forward Storage Requirements

Upon a successful EBT upload, the sending Point ~~(sender)~~ must maintain an archive of the sent EBT in accordance to the OEB mandated retention time period. Original encryption/signature keys must be retained and associated with the archived messages.

It is up to the discretion of the receiving Point what level of inbound message archiving and logging they wish to maintain to protect their interests.

4.4 Failures

The following is the minimum recommendation for handling Point to Point communication failures:

1. A protocol failure occurs any time a sender cannot connect to the recipient's server. For example, if connection to a recipient server fails, or posting a file fails, this is a protocol failure.
2. An exchange failure is when a sender has had continual protocol failures over a minimum two-hour period. Each entity is required to try at least 3 times over the two-hour period before flagging an exchange failure.
3. E-mail is the recommended mechanism to notify the intended recipient of protocol and exchange failures. This will assist in resolving and documenting problems.

5. Functional Acknowledgements

This section describes the requirements for Functional Acknowledgements between sending Points and recipient Points and other Points. Functional Acknowledgements are used to acknowledge the delivery of PIPE Documents. They identify invalid XML formatting of PIPE Documents and good and bad formatting of PIP transactions. A more complete description of Functional Acknowledgements can be found in the "Ontario Electronic Business Transactions (EBT) Standards Document" (Business Rules Document).

5.1 Operation of Functional Acknowledgements

5.1.1 XML Validation

The receiving Point will XML-validate each EBT Message that it receives. After checking for errors, the receiving Point will send a Functional Acknowledgement back to the sending Point as described in the "Ontario Electronic Business Transactions (EBT) Standards Document" (Business Rules Document).

5.1.2 Point Functional Acknowledgement Processing

The following lists the rules for a recipient Point to send Functional Acknowledgements to another a sending Point:

- If the Market Participant Recipient in the EBT document is the receiving Point then the FunctionalAcknowledgement should be of type Accept if the document passes XML validation and consists of all good PIPs.
- If the Market Participant Recipient in the EBT document is not the receiving Point then the FunctionalAcknowledgement should be of type DocReject.
- If the receiving Point is sending a FunctionalAcknowledgement of type PartialAccept, it will only process good transactions from the sending Point. The PartialAccept Functional Acknowledgement being returned to the sending Point will list which transactions were accepted and which transactions were rejected.

The receiving Point may send a DocReject Functional Acknowledgement if it detects other errors within the EBT document such as duplicate PIPE Document reference numbers and duplicate Transaction reference numbers from a sending Point.

- If the sending Point is unable to upload the EBT Message by loading it into the destination Mailbox of the receiving Point, then it must report the problem back to the receiving Point. Since all Points are validating based on public schemas at the OEB web site, this should only be due to network failures. This reporting is to be done via e-mail or some other mutually agreed upon method between the Points.
- When the sending Point receives a DocumentReject or PartialAccept Functional Acknowledgement, within four hours it will notify the receiving Point and trigger a process to resolve the issue. Best efforts should be made to resolve the issue.

5.1.3 Trading Partner Validation

The recipient will send a Document Reject Functional Acknowledgement if it determines that no trading partner agreement is in place.

5.2 Availability Requirements

For the purposes of dispute resolution, each Point must archive all EBT Messages it processes and their corresponding Functional Acknowledgements to the same criteria specified in the Hub to SubscriberData Transport protocol.

5.3 XML Parsers

This section describes a Point's requirements for XML Parsers.

Each party is free to select and use their own XML validating parser. If two parties disagree on the XML validity of a PIPE Document, final resolution will be determined by examination of the XML according to the schemas as published on the OEB web site. Validity will be determined using the current active specification of the XML Schema Definition Language as listed on the World Wide Web Consortium (W3C) web site.

5.4 Detection of Other Errors

Where the error reporting tools and mechanisms of the protocol (i.e., HTTPS responses and Functional Acknowledgements) do not provide a mechanism for communicating the type of error back to the sender, the issue will be escalated based on business arrangements between the market participants. At a minimum, the problem will be escalated via a telephone call to the operations staff of the originator of the transaction.

Such errors include:
- Decryption errors;
- Signature errors;
- Receipt of a Partial Functional Acknowledgement or Document Reject Functional Acknowledgement from a Point; and
- Lack of a Functional Acknowledgement from the recipient within the required four hours.

6. Time Synchronisation and Time Stamping

This section describes the Point requirements for time synchronisation and the time stamping of transactions. In a networked environment such as the EBT marketplace and since the arrival times for transactions are used for dispute resolution an accurate consistent time stamp is imperative.

6.1 Time Synchronisation

Each Point will maintain an accurate and consistent time by connecting through the NTP protocol to a service with an accuracy of at least that provided by a stratum-2 server[1]. There are various stratum-2 timeserver sources available, including free servers, subscription servers and potentially the IMO time-servers if they become accessible.

6.2 Time Stamping of Transactions

Each Point will date and time-stamp each EBT Document when it receives it.
The Point should use Eastern Standard Time with no Daylight Savings Time change as specified in the "Ontario Electronic Business Transactions (EBT) Standards Document".

---

[1] Stratum-1 servers connect to GPS or atomic clocks. Stratum-2 servers obtain their reference from stratum-1 servers. Startum-2 does not define an accuracy of the clock, but instead defines the number of hops to a source time-base. The resulting accuracy is dependent on the Internet and the number of routers between the various timeservers. By connecting to a stratum-2 timeserver, the point becomes stratum-3. Stratum-3 timeservers are expected to have a short-term drift of less than 3.7×10^{-7} in 24 hours.

7. Requirements of a Point

7.1 Routing of Messages

In Point to Point communication, the Market Participant Point is always the originator or end destination of the EBT. All other EBTs not specific to that the recipient Point will be rejected. All Points are expected to communicate directly.

7.17.2 Availability Requirements

Because the processing of EBT messages by Points is critical to the success of the marketplace, Points must commit to remain online connected to the Internet with no more than 15 minutes of downtime per day.

Where there are connectivity problems, it is suggested that a Point retry 3 times, waiting 15 minutes between retries before implementing the escalation process for handling failures. The initiator will log each transmission.

A suggested maintenance window exists between midnight Saturdays (EST) and noon Sundays (EST) when Point servers can be shut down for longer periods.

Different availability requirements for specific Points may apply if mutually agreed to by the two parties.

7.27.3 Market Participant Information

In order to send an EBT message to the correct Point, each Point must maintain Market Participant addressing information within their systems. The communications parameters to be exchanged between the Points include URL, User ID, Password, OEB license number, trading partner agreement contract information, and the e-mail address. In other words, the sending Point needs to know the URL, User ID, and Password to push EBTs to the receiving Point, and vice versa. The public key for PGP encryption/digital signature verification should be registered with the Massachusetts Institute of Technology worldwide PGP public key-server.

It is the responsibility of the initiating party to contact the second party to initiate communication set up.

7.37.4 Routing Information Messages

Only the Upload request type will be allowed for Point to Point communication:

```
Content-Disposition: form-data; name="request_type"

Upload
```

A Point will respond to the upload request based on the process and the response schema defined in the Ontario EBT Data Transport Protocol.

For the format of an HTTP Upload Request, see Appendix A - "HTTP Upload Request".
For the format of a HTTP Upload successful HTTP Response see Appendix B - "HTTP Upload Response".

Appendix A -- HTTP Upload Request

It is not the intention that the following sample be the basis for any programming effort. Compliance to IETF relevant RFC's and W3C standards should be consulted for syntactical references.

The following is an example of a Point to Point HTTP Upload Request:

```
POST /RecipientServer/mailbox HTTP/1.1
Date: Tue, 20 Dec 2000 08:12:31 GMT
Connection: Keep-Alive
Host: www.ontarioRecipientServer.com
Content-Language: en, fr
Content-Type: multipart/form-data; boundary=EBTpart;
Content-Length: 3222

--EBTpart
Content-Disposition: form-data; name="sender"

12345678

--EBTpart
Content-Disposition: form-data; name="user_id"

aUser

--EBTpart
Content-Disposition: form-data; name="user_password"

aPassword

--EBTpart
Content-Disposition: form-data; name="request_type"

Upload

--EBTpart
Content-Disposition: form-data; name="ebt_document";
filename="transaction.xml"
Content-Type: application/octet-stream

MIME-Version: 1.0
Content-Type: multipart/encrypted; boundary="=--";
protocol="application/pgp-encrypted"

--=--
Content-Type: pgp-encrypted

Version: 1

--=--
Content-Type: application/octet-stream
```

```
-----BEGIN PGP MESSAGE-----
Version: 2.6.2

hIwDY32hYGCE8MkBA/wOu7d45aUxF4Q0RKJprD3v5Z9K1YcRJ2fve87lMlDlx4OjeW
4GDdBfLbJE7VUpp13N19GL8e/AqbyyjHH4aS0YoTk10QQ9nnRvjY8nZL3MPXSZg9VG
QxFeGqzykzmykU6A26MSMexR4ApeeON6xzZWfo+0yOqAq6lb46wsvldZ96YA
AABH78hyX7YX4uT1tNCWEIIBoqqvCeIMpp7UQ2IzBrXg6GtukS8NxbukLeamqVW31y
t21DYOjuLzcMNe/JNsD9vDVCvOOG3OCi8=
=zzaA
-----END PGP MESSAGE-----
--=----
--EBTpart--
```

Appendix B -- HTTP Response

It is not the intention that the following sample be the basis for any programming effort. Compliance to IETF relevant RFC's and W3C standards should be consulted for syntactical references.

The following is an example of a Point to Point HTTP Upload Response:

```
HTTP/1.1 200 OK
Date: Tue, 20 Dec 2000 08:12:31 GMT
Host: www.ontarioRecipientServer.com
Content-Type: text/XML
Content-Length: 1234

<?xml version="1.0" encoding="UTF-8"?>
<RESPONSE xmlns:xsi=" http://www.w3.org/2001/XMLSchema"
xsi:noNamespaceSchemaLocation = "
http://www.oeb.gov.on.ca/Response.xsd">
  <HTTP_RESPONSE>
    <STATUS_CODE>200</STATUS_CODE>
    <REASON_PHRASE>OK</REASON_PHRASE>
    <REQUEST_TYPE>Upload</REQUEST_TYPE>
    <TIMESTAMP>Tue, 20 Dec 2000 08:12:31 GMT</TIMESTAMP>
  </HTTP_RESPONSE>
  <UPLOAD new_doc_id="12345678" doc_name="abcdefg.xml"/>
</RESPONSE>
```