TED PRO Third Party Posting API

Based on ECC Firmware Revision R400

Table of Contents

Introduction

Starting with ECC version 400 and later, it's possible to post cumulative usage data from the TEDPRO to a 3rd party server. The MTUPRO can post cumulative usage data starting with version 443 and later. This feature allows developers to create remote services that can receive TEDPRO data, without requiring end users to place their TEDPRO on the internet.

Third-party posting consists of two phases:

- Activation – A user registers their TEDPRO with a remote server.
- Post – An "activated" TEDPRO posts cumulative usage data for one or more MTU's to the server

In this document you will notice that many of the XML elements start with <ted5000>. This is to help ensure backwards compatibility w/ older model TED devices.

Activation

To configure the TEDPRO to post to a 3rd party service, the following steps will need to be executed by the user:

1. User selects "Activate Energy Posting" from the main menu.
2. User is prompted with a screen to enter 3rd party activation information:

- URL of the authentication service.
- Unique Identifier (provided by the 3rd party) to uniquely identify this ted on their system. This can be as simple at the GW ID, or a password (20 character max).
- A disclaimer about data security will is presented via this popup; activation means they acknowledge the privacy risks.

3. User hits submit. TEDPRO makes a HTTP POST request to the 3rd party authentication service sending the unique identifier supplied in section 2. The mime type of this post is application/xml.

Sample Activation XML

```
<ted5000Activation>
     <Gateway>210000</Gateway>
     <Unique>123456</Unique>
</ted5000Activation>
```

- **Gateway** is the hardware identifier (Serial Number) of the ECC unit making the post
- **Ver** is the version of the firmware running on the ECC (or the version of the firmware running on the MTUPRO if its posting directly)
- **Unique** is the unique identifier that was supplied in step 2. This is optional and is meant as an extra security token to be provided to the 3rd party service if they wish to use it.

4. The third-party server will need to provide an activation response to this request in the form of the following XML. Activation can be performed by the user more than once.

Sample Activation Response XML

```
<ted5000ActivationResponse>
     <PostServer>www.theenergydetective.com</PostServer>
     <UseSSL>T</UseSSL>
     <PostPort>443</PostPort>
     <PostURL>/postdata.cgi</PostURL>
     <AuthToken>MySecurityToten</AuthToken>
     <PostRate>10</PostRate>
     <HighPrec>T</HighPrec>
</ted5000ActivationResponse>
```

- **PostServer**- The ip address or dns resolvable name of the server (80 character max)
- **PostPort** - the port that the server is running on
- **PostURL** - the path of the service receiving the post (80 character max)
- **UseSSL** - whether or not to post data via SSL

- **AuthToken**- Unique token provided by the 3rd party used to as an authentication mechanism when data is submitted by the TEDPRO (optional) (20 character max)
- **PostRate** - The number of minutes between data posts. (1 - 15)
- **HighPrec** – Enables/disables “High Precision” mode. A value of “T” enables High Precision mode which will include fractional watt-hours. Any other value will floor each value to the nearest watt hour.

Posting Data

Once activated, the TEDPRO will begin to post the following data to the service defined above. This will be posted as an HTTP POST and will use the mime type application/xml: The data will be in the following format:

Sample Post XML

```
<ted5000 GWID="200000" auth="MySecurityToken">
  <COST mrd="15" fixed="10.00" min="5.00"/>
  <DEMAND kVA="0">
    <demandCost timestamp="1338319620" peak="12.184" cost="22.36"/>
  </DEMAND>
  <MTU ID=100000 type="0">
    <cumulative timestamp="123456000"  watts="10000" rate="0.12345" pf=”99.9”/>
    <cumulative timestamp="123456060"  watts="10005" rate="0.12345" pf=”99.9”/>
    <cumulative timestamp="123456120"  watts="10010" rate="0.12345" pf=”99.9”/>
  </MTU>
  <MTU ID=100001 type="0">
     <cumulative timestamp="123456000"  watts="10000" rate="0.12345" pf=”99.9”/>
     <cumulative timestamp="123456060"  watts="10005" rate="0.12345" pf=”99.9”/>
     <cumulative timestamp="123456120"  watts="10010" rate="0.12345" pf=”99.9”/>
  </MTU>
</ted5000>
```

The “**COST**” element is used to provide information for calculating monthly cost. Data posted from an MTUPRO will show these fields zeroed out. The following attributes are present:

- **mrd** (Meter Read Date): This is the first day of the next billing cycle where all monthly values reset to zero.
- **fixed** (Fixed Amount): If the utility charges a fixed amount, this value is displayed here (0 if not specified).
- **min** (Minimum Amount): If the utility charges a minimum amount per month, the value is displayed here (0 if not specified)

The “**DEMAND**” element is used to provide information regarding the demand charge rate and value for a given minute. If Demand charges are not enabled for the gateway, this value is not displayed. In addition, if no demand charges have been calculated since the Gateway’s last reboot (e.g. still collecting 15 minutes worth of data) this element will not be present either. This data is not available on the MTUPRO. The DEMAND element has one attribute:

- **kVA** :if a value of 1 is specified, Demand Charge is being calculated based on kVA. Otherwise, Demand Charge is being calculated based on kWh.

The “**demandRate”** element will be repeated for each “minute” reading since the last post. It contains the Demand Charge rate in effect for that particular minute.

- **timestamp:** the "Unix Epoch" based timestamp of the reading.
- **peak:** The Demand being used to calculate the rate. If kVA is 1, this value is units of kVA. Otherwise, it’s in units of kWh.
- **cost:** The “demand charge” when the reading was made (units of dollars)

The "**MTU**" element will be repeated for each MTU in the system. If the gateway is only configured for 1 MTU, this block will only be repeated once. This element has two attributes:

- ID: the unique hardware id of the MTU
- VER: The version of the MTU firmware
- TYPE: the configuration type of the MTU

    - 0 = Net :Positive values are considered consumption, negative values are considered generation
    - 1 = Load: All values are considered to be generation. Developers are encouraged to take the absolute positive value of the number posted to compensate for end users having their CT’s installed backwards.
    - 2 = Generation: All values are considered to be generation. Developers are encouraged to take the absolute negative value of the number posted to compensate for end users having their CT’s installed backwards.
    - 3 = Stand-Alone: All values are considered to be generation. Developers are encouraged to take the absolute positive value of the number posted to compensate for end users having their CT’s installed backwards.
    - 255 = MTUPRO: All values are posted directly from an MTUPRO so Net, Load, Generation need to be configured on the 3rd party server.

The "**cumulative**" element will be repeated for each queued "minute" reading since the last post. The device can store up to 15 "minute" readings. If more than 15 minutes elapse between successful posts, only last 15 minutes will be resent. Each cumulative element has the following attributes:

- **timestamp:** the "Unix Epoch" based timestamp of the reading.
- **watts:** the cumulative reading in watt hours by the MTU (since its creation) at the specified timestamp. Fractional Watt-Hours are represented as a decimal value up to 4 digits of precision.
- **rate:** The "rate in effect" when the reading was made (based on GW Utility configuration).
- **Pf:** The average "Power Factor" for the corresponding minute.
- **Voltage:** the average voltage for the corresponding minute.

The 3rd Party Server should respond w/ a HTTP **200** code if successful. Any code >= 300 will be considered to be an error code.

Debugging

Posting statistics as well as error codes can be viewed on the stats.htm page (e.g. http://tedPRO/stats.htm)

The error codes are:

Activation Error Codes

- 0 – success
- 100 - Parsing error of activation xml file
- 101 - Timeout accessing activation file (also file not found)

Posting Error Codes

Typically this code will be the result code of whatever was returned by the server after the post (e.g. 200 = Success, 400= Bad Request). In addition to these codes, the TEDPRO may also return one of the following codes:

- 1000 - Socket error connecting to the server
- 2000 - Timeout posting data to the server
- 3100 - SSL Handshake timeout
- 4000 - Posting Data (Not an error code, just an indication that data is currently posting)

Quick Start

To start your 3rd party posting development, it may be best to try a static xml file for activation, and a simple server-side PHP script to receive posts from your TEDPRO.

Sample Activation Response XML

```
<ted5000ActivationResponse>
     <PostServer>demo.theenergydetective.com</PostServer>
     <UseSSL>F</UseSSL>
     <PostPort>80</PostPort>
     <PostURL>/postData.php</PostURL>
     <AuthToken>myToken</AuthToken>
     <PostRate>1</PostRate>
     <HighPrec>T</HighPrec>
</ted5000ActivationResponse>
```

Copy this file to your web server. Please make sure that your webserver is configured allow this file to be accessed via HTTP POST. Be sure to test it in IE or Firefox to make sure it's visible via your web connection: i.e. http://demo.theenergydetective.com/act.xml . This URL is what will be used for the "Activation URL" when you activate 3rd party posting via Footprints.

On the server, you can use the following PHP script to receive the XML post from the TEDPRO, and write the XML file to a directory.

Sample PHP Post Data Script

```
<?php
#change the /home/demo/xml path to the directory you wish to have xml posts
#written to.

$outp = fopen("/home/demo/xml/xmlfile." . date("YmdHis") . ".xml", "w");
fwrite($outp, $HTTP_RAW_POST_DATA);
fclose($outp);

?>
```

The URL path to this file is what is used for the PostServer and PostURL fields of the static activation xml file. Once activated you should see XML files being written within a few minutes.