Data Concentrator API architecture for DC/DM/DR Participants

1 October 2024

Contents

Date	Version	Comments	Author
01/10/2024	1	Initial release	NESO go-live

1. Overview

Introduction

This document sets out the Data Concentrator API architecture the National Energy System Operator (NESO) has in place to facilitate the connection of third parties participating in the Dynamic Containment, Moderation and Regulation programmes.

Data Concentrator Service

This service is designed to provide maximum benefits to all parties, with the minimum disruption to their current business operations.

Market Participant's Responsibilities

- Establish internet connectivity to their IT infrastructure with a publicly accessible static IP address.
- Capture performance monitoring data at the required sample rate and accuracy.
- Transmit performance monitoring data to NESO utilising file uploads to the Data Concentrator service and in the required CSV file format.
- Test and validate capture and transmission of performance monitoring data prior to commissioning to the Data Concentrator service.
- Transmit availability/outage redeclarations to NESO utilising file uploads to the Data Concentrator service and in the required CSV file format.
- Test and validate creation and transmission of availability/outage redeclarations data prior to commissioning to the Data Concentrator service.
- In the event that any part of the market participant equipment (including the communications links to the Data Concentrator) fails, then the market participant will be required to repair such equipment within 5 working days of notification of the fault.
- Inform NESO of any planned outages to Data Concentrator API connectivity at least 48 hours in advance to **commercial.operation@nationalenergyso.com**.

NESO Responsibilities

- Establish connectivity between the Data Concentrator host and the market participant, issuing API credentials (URL, username, password)
- Configure the Data Concentrator to receive the market participant's data submissions.
- Test and structurally validate data submissions prior to commissioning.
- Provide first line support for service.

Note this document provides an overview of the information required to physically establish the communications between a Market Participant and NESO. It is assumed for the purposes of the document any legal and contractual activities as part of the overall connection process will have been completed prior to carrying out the technical process described within this document.

System Architectures

Data Concentrator API Overview

The market participant client will use HTTPS protocol to upload the data submissions to the Data Concentrator API over the public internet. The HTTP request message is secured using a TLS 1.2 tunnel with data and metadata supplied in the request body using MIME multipart.

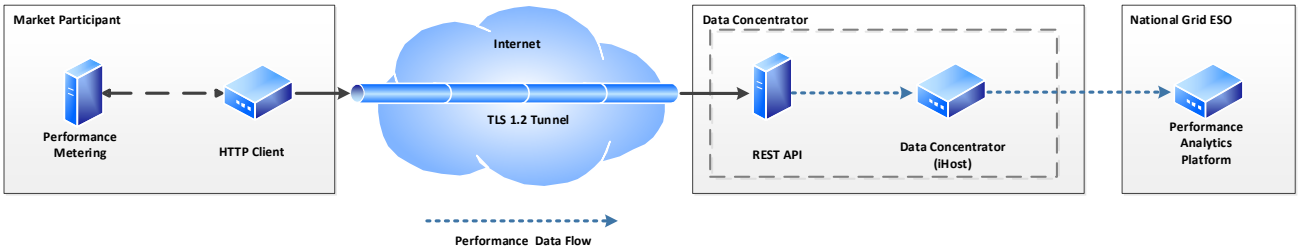
The Data Concentrator API is currently used for two different types of file submissions:

- Performance Monitoring

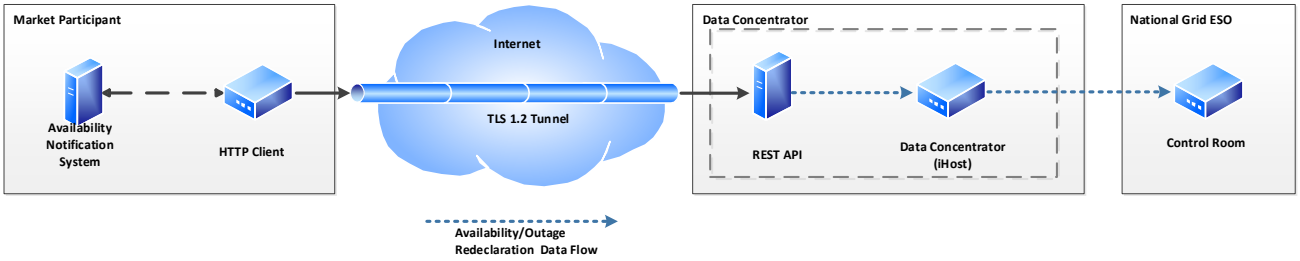
- Availability/Outage Redeclarations

The method of integration with the API is the same for both file types, with the market participant configuring their client to upload files to the same URL and use the same username/password credentials. The only difference between the file types is the file naming convention and file contents.

Performance Monitoring



Availability/Outage Redeclarations



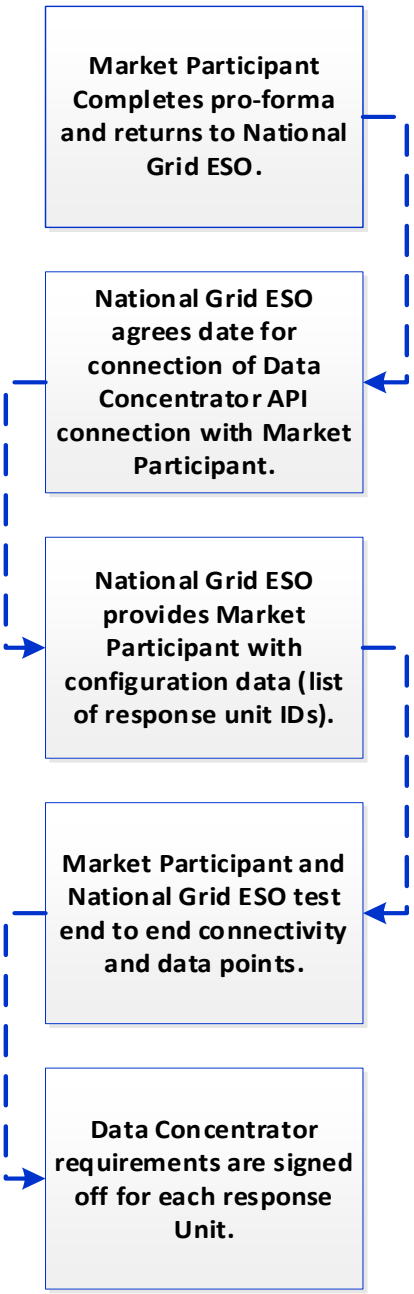
Data Concentrator API Commissioning

The successful provision of operational metering comprises of three phases.

- Data Exchange
- Set-up
- End to End Testing

All three phases need to be completed by the Market Participant to satisfy the requirements for participation in DC/DM/DR. The successful provision of performance monitoring and availability notifications is only one part of the process and completion of this stage does not mean the units are ready for participation.

NESO aims to complete a Data Concentrator API connection and to have it tested within 4 weeks of receipt of pro-forma in Section 5 (Metering Connection Pro Forma).



2. Data Concentrator API Integration

API Credentials

API credentials (username/password) will be issued to the market participant together with the API URL during connection set-up via secure encrypted email.

Each market participant client is assigned a unique API username (client ID) by NESO as part of the connection set-up process.

The password issued by NESO will be at least 56-characters and of high entropy. The market participant should ensure this password is kept secure on their client machine.

Implementation

HTTP Version

HTTP Version 1.1

Establishing Connection to the API

The market participant's HTTP client initiates a HTTPS connection to a FQDN registered in public DNS using TLS 1.2 to encrypt the communications. The market participant is advised of the FQDN and port to configure their client during connection set-up.

The server DNS record has a low Time to Live (TTL) and should not be cached by the client for longer than this TTL value. Ideally, the client should perform a fresh DNS lookup request each time it attempts to connect to the API server.

Encryption, Authentication and Access Control

The use of TLS 1.2 is enforced for all connections to the API server.

The market participant's client should be configured to use one of the following TLS cipher suites, which will be enforced by the server:

- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xC0, 0x2F)
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xC0, 0x30)
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 (0x00, 0x9E)
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (0x00, 0x9F)

The API server is configured with a X509 certificate signed by a public Certificate Authority. Client machines that keep their trusted root certificate store up to date (e.g., via OS security updates) will be able to automatically verify the server's certificate's chain of trust.

Clients are authenticated using HTTP Basic Authentication with username and password passed in the Authorization request header.

File Upload API

All CSV files are uploaded to the same API using a HTTP POST request to the **/ihost/deviceapi/files** endpoint URI path.

The Content-Type header must be set in the request to **multipart/form-data**. Further details on the structure of a multipart request are available in RFC2388.

The body of the request must consist of two media parts separated by boundary indicators.

The first media part contains the file metadata in JSON format. This part must include two headers:

- Content-Type header set to application/json; charset=UTF-8
- Content-Disposition header with value "form-data" and name "metadata"

The second media part contains the CSV file contents. This part must include two headers:

- Content-Type header with the media type for the file (or application/octetstream)
- Content-Disposition header with value "form-data" and name "data"

An example POST request (for Performance Monitoring CSV upload) is provided below:

```
POST /ihost/deviceapi/files HTTP/1.1
Host: HOSTNAME:PORT
Authorization: Basic REDACTED
Content-Type: multipart/form-data; boundary=8455223c3ebc41c19302cc882ef80dd8
Content-Length: CONTENT_LENGTH

--8455223c3ebc41c19302cc882ef80dd8
Content-Disposition: form-data; name="metadata"
Content-Type: application/json; charset=UTF-8

{"Name":"UNIT1_20200915142400000_20HZ_perfmonv1.csv","Process":true}
--8455223c3ebc41c19302cc882ef80dd8
Content-Disposition: form-data; name="data"
Content-Type: application/octet-stream

… CSV contents …
--8455223c3ebc41c19302cc882ef80dd8--
```

The API will respond to the upload request with one of the following response status codes:

- 201 Created: The file was successfully uploaded
- 400 Bad Request: The request was invalid
- 401 Unauthorized: Authentication failed check username and password are valid
- 404 Not Found: The URL supplied by the client was invalid

Any status code in range 500-599 should be treated as a transient error and the client should retry the upload.

File Metadata Structure

Field	JSON Data Type	Description	Example
Name	String	Name of the file	UNIT1_20200915174715_20HZ_perfmonv1.csv
Process	Boolean	Flag to indicate if the contents of the file should be processed. Should always be set to true.	true

Table 1 - File Metadata Fields

CSV File Contents

The "data" part of the request should contain the CSV file contents in the required file format.

For details of the Performance Monitoring CSV file format please refer to the published link below:

https://www.nationalgrideso.com/document/225781/download

For details of the Availability/Outage CSV file format please refer to the published link below:

https://www.nationalgrideso.com/document/230486/download

File Upload Frequency

The client should upload files to the API shortly after the file has been generated. The client should avoid uploading large number of files and prefer uploading files on a regular and predictable basis.

File Buffering and Retry Mechanism

If the upload fails for some reason, then the device should not delete the file from local storage and should re-attempt the upload later, waiting at least 1 minute between retries.

If a device loses communication with the data concentrator, then it possible that it has a backlog of files to be uploaded to the server. In this case, the client should upload the files in the order they were created, with the oldest file uploaded first. The client should also try not to flood the API server by waiting 30s between each subsequent successful file upload.

Submitting Test Files

The Data Concentrator service provides the option to submit test files to the API. The purpose of this is to allow the API integration and file validation to be tested with dummy test data in isolation. When submitting a test file to the API, the exact same structural validation is performed, however unlike real data submissions the file is not forwarded to the upstream NESO operational systems.

To make use of this option, append _test to the file name being uploaded to the API. For example:

UNIT1_20200915142400000_20HZ_perfmonv1_test.csv

UNIT1_20200915142400000_redecv1_test.csv

3. Metering Connection Pro forma

Please refer to the complementary "***Performance Monitoring Pro-forma for DC/DM/DR Participants***" document.

4. Glossary

Abbreviation	Meaning
API	Application Programming Interface
CSV	Comma Separated Values
FQDN	Fully Qualified Domain Name
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
HZ	Hertz
JSON	JavaScript Object Notation
NESO	National Energy System Operator
TLS	Transport Layer Security
URI	Uniform Resource Identifier
URL	Uniform Resource Locator

Table 2 – Glossary Table