

NERDA BETA

NEAR REAL-TIME DATA ACCESS

API GUIDE

nerda.opengrid.com

CONTENTS

Introduction to NeRDA.....	3
NeRDA API Guide	4
Creating an Account	4
Terms and Conditions.....	5
Configuring your API Key	5
Accessing APIs	7
NeRDA Contacts	12

Introduction to NeRDA

The Near Real-time Data Access (NeRDA) Portal is making near real-time data available to our stakeholders and interested parties.

We're helping the transition to a smart, flexible system that connects large-scale energy generation right down to the solar panels and electric vehicles installed in homes, businesses and communities right across the country.

In line with our Open Networks approach, our Near Real-time Data Access (NeRDA) portal is live and making available power flow information from our EHV, HV, and LV networks, taking in data from a number of sources, including SCADA PowerOn, our installed low voltage monitoring equipment, load model forecasting tool, connectivity model, and our Long-Term Development Statement (LTDS).

Making near real-time data accessible from DNOs is facilitating an economic and efficient development and operation in the transition to a low carbon economy. NeRDA is a key enabler for the delivery of Net Zero - by opening network data, it is creating opportunities for the flexible markets, helping to identify the best locations to invest flexible resources, and connect faster.

You can access this information via our informative near real-time Dashboard and download portions of data or connect to our API and receive an ongoing stream of near real-time data.

NeRDA API Guide

This guide provides instruction on accessing and utilising the NeRDA APIs. The NeRDA data is made available through the NeRDA Portal API Console. To access this, you must first log in to your NeRDA Portal Account and configure your API Key.

Creating an Account

Before being able to use the NeRDA APIs, you must create an account.

From any of the pages in the Portal, use the three-bar menu icon in the top right to access the “Sign In / Create Account” button.

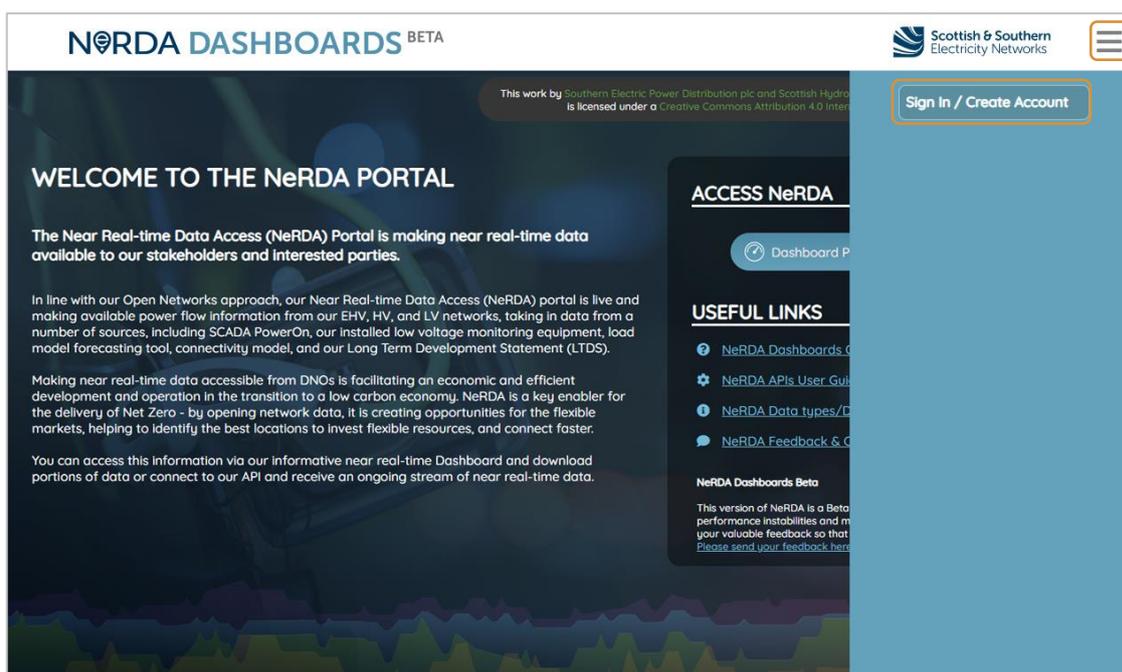


Figure 1: NeRDA Portal Homepage - API access login

After selecting this, you will be redirected to the SSEN login / registration portal where you are able to log in with an existing SSEN account or create a new account. Upon completion of the login / registration, you will be directed back to the NeRDA Portal.

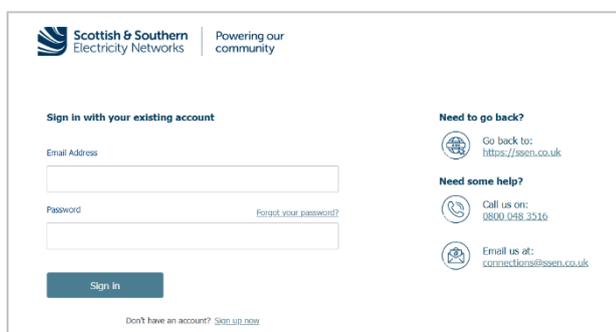


Figure 2: SSEN Login Portal

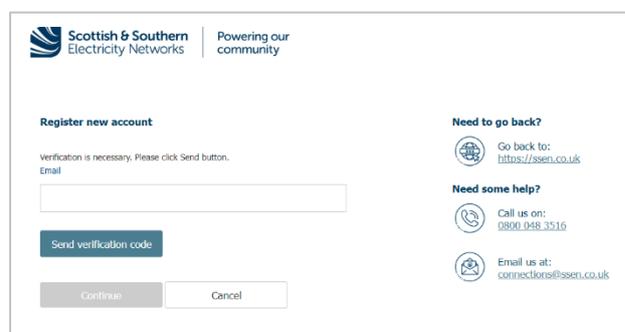


Figure 3: Registration portal

Terms and Conditions

The Portal is configured to present current terms and conditions to users on first login or whenever the Terms and Conditions have been updated. To access any further features, you must accept the terms. If you do not accept the terms, you will be unable to proceed.

The terms are presented as shown in the screenshot below.

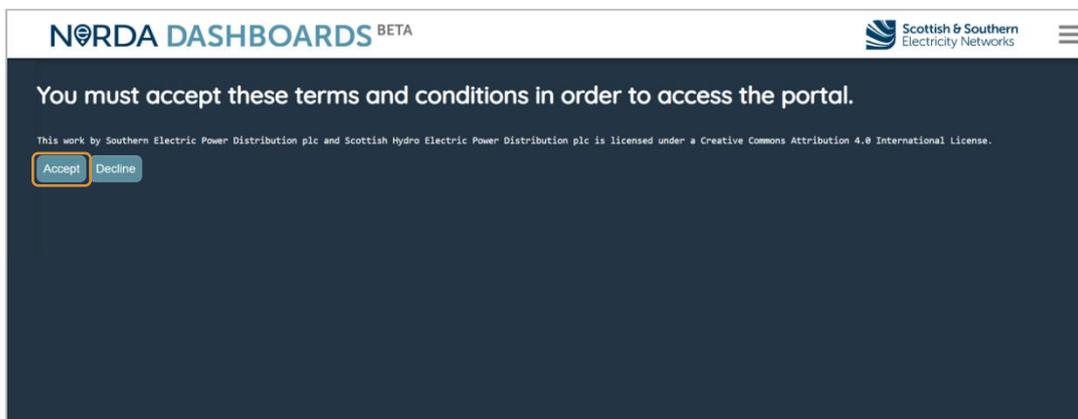


Figure 2: NeRDA API Portal Terms and Conditions

Selecting “Accept” will register your acceptance of the terms and conditions and you will be directed to the homepage.

Configuring your API Key

After accepting the Terms and Conditions for accessing NeRDA data, the API Console can be accessed by clicking on the User icon in the Portal and selecting “View APIs and Manage Keys” as shown in Figure 3 (you will need to register as outlined in the previous section).

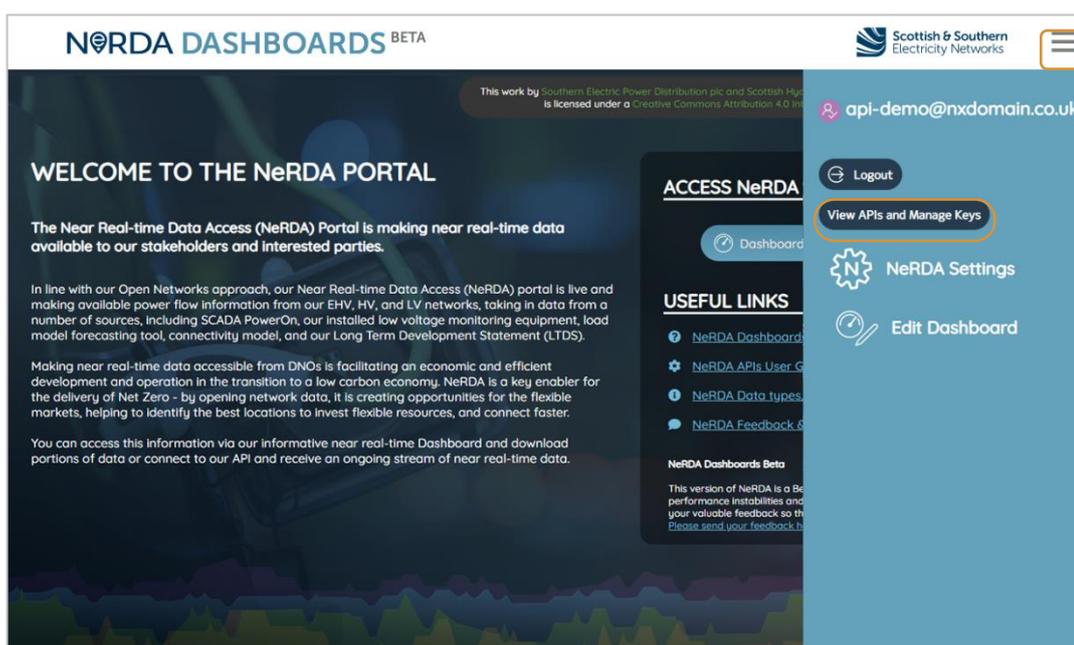


Figure 3: Locating the API Console

After selecting this item, you will be directed to the User API Console. This will display a page as shown in Figure 4.

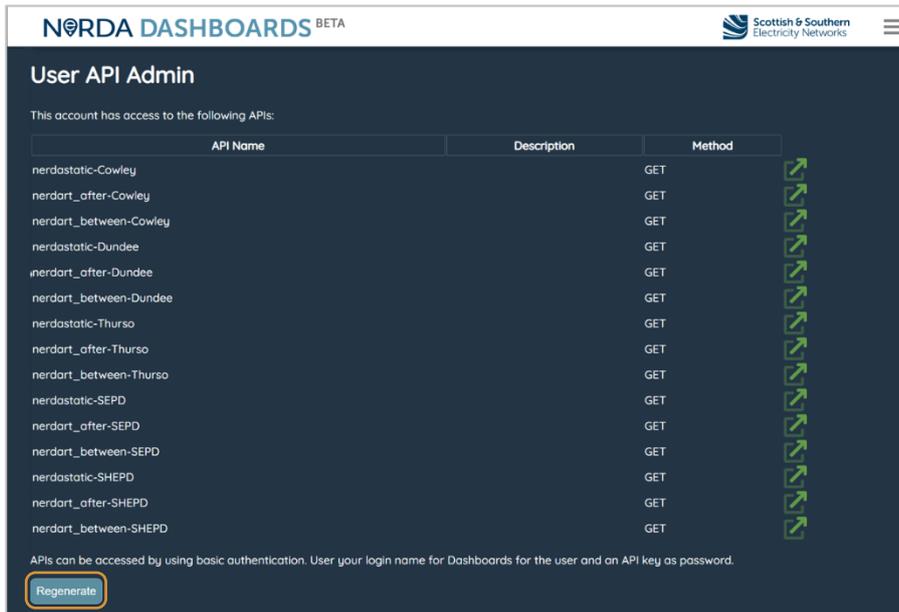


Figure 4: API Console Initial View

In this state, no access has been configured for the specified user. To start using the APIs, you must generate an API key by clicking the “**Regenerate**” button.

This will present an additional section as shown in Figure 5.

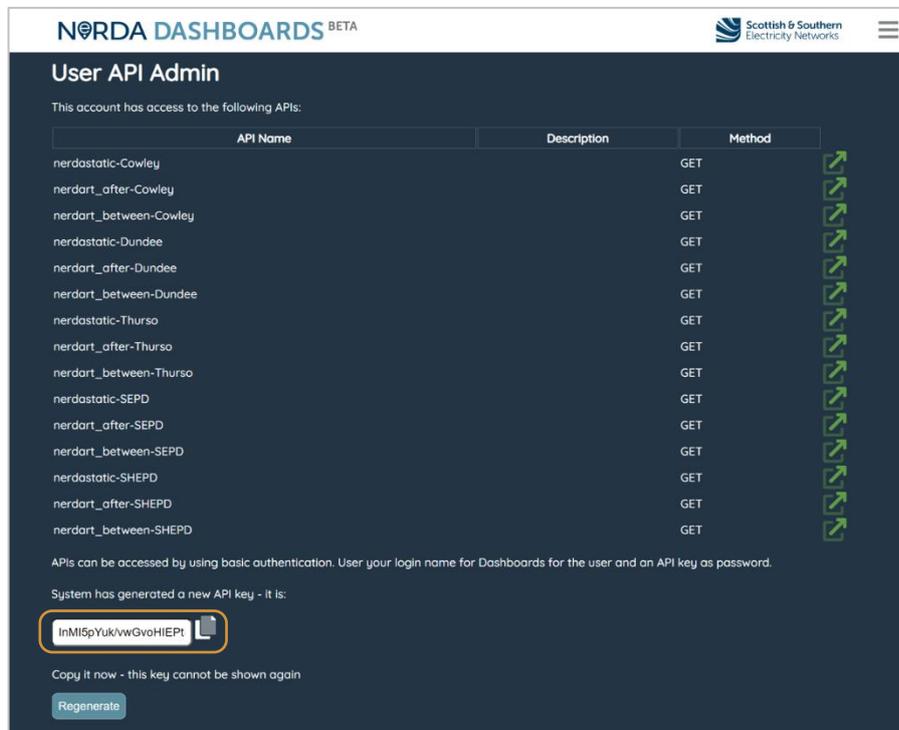


Figure 5: API Console showing API Key

The key must be copied from this screen – the “Copy” icon can be used to copy this to the clipboard, or the key in the text box can be manually copied.

In this example, the key presented is:

InMI5pYuk/vwGvoHIEPtSWYoIyBwP06aLBOB1t1fTps=

Any following access to the page will display as shown in Figure 6 this allows confirmation of the key that has been set (through the first 6 letters) but the full key cannot be retrieved again. A new key can be generated by using the “Regenerate” button again. This clears the previous key.

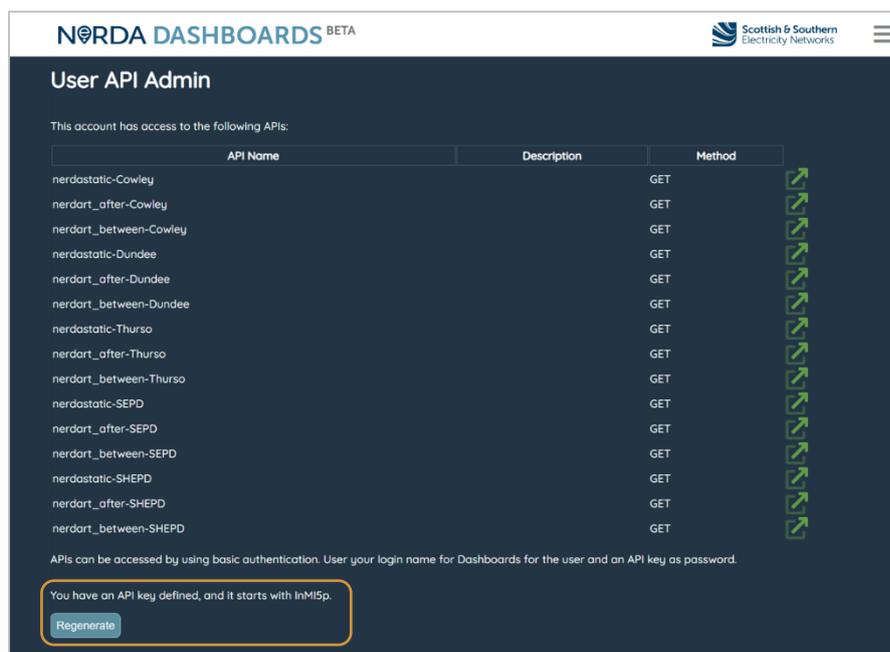


Figure 6: API Console after key generation

APIs will also function in the browser whilst a NeRDA Portal session is open.

Accessing APIs

NeRDA makes data available within the NeRDA portal, but also provides a number of APIs that can be used to query the system for substations, find the measurement points and retrieve data.

The NeRDA system holds around 800m data points and ingests live data, so the APIs allow the following actions to be undertaken:

- Find substations – can be BSPs, Primaries or Secondaries, and the lines or measurements points for the substation
- Find the measurements available for each line / point – these are limited to a maximum of x entries per request but can be queried using a start date and / or end date so you can make multiple requests to pull historical data required

Substations Endpoint

The substations endpoint can be found under the “nerdastatic” APIs in your API console. These are currently split by network areas according to the rollout of the system. By default, each will return all items for the specified area.

You can also append an NRN (Network Reference Number) found on the various substation pages – to return a specific substation, e.g. Figure 7 shows the page for the Hayes BSP and shows the NRN as 2521 at the top and the “2521 Static API” also links directly to the “nerdastatic” API for this site.

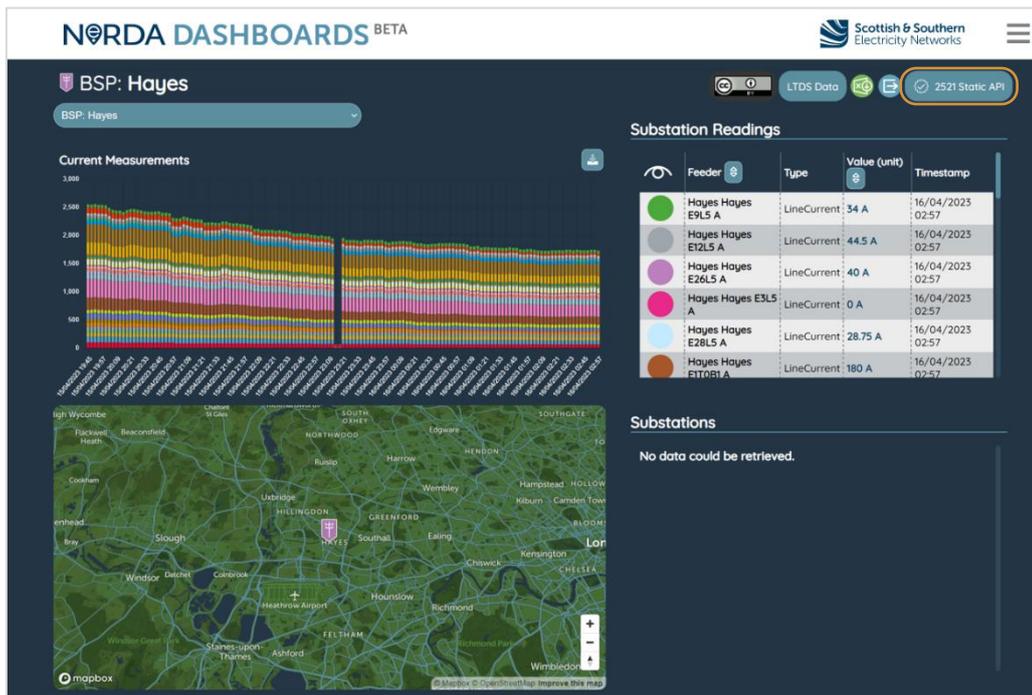


Figure 7: Substation Page with API Links

The URL, whether manually appending or using the in-page link, is:

<https://nerda.opengrid.com/api/nerdastatic-SEPD/2521>

Accessing this URL returns a record containing details on the lines at the station, the location of the station and the measurements available on those lines. The “nerda_measurement_id” on the individual measurements can then be used to retrieve time-series data.

```

{
  "transformers": [],
  "sds_site_id": "Hayes",
  "latitude": "51.504300359492944",
  "switches": [
    {
      "switch_name": "Hayes E24L5",
      "nerda_switch_uid": "_2446781b-f978-4992-8089-770a3a643233",
      "measurements": [
        {
          "nerda_measurement_id": "_6a8dcaff-ea6b-4ea4-b78e-0a7f86c7bf61",
          "unitMultiplier": "none",
          "measurementType": "LineCurrent",
          "unitsSymbol": "A"
        },
        {
          "nerda_measurement_id": "_d4cdb6d4-62a3-439c-abf8-1351ae0dc85e",
          "unitMultiplier": "none",
          "measurementType": "SwitchPosition",
          "unitsSymbol": "none",
          "switch_id": "_4abe847c-8e05-4afa-9a8a-d9b3576c8b69"
        }
      ]
    },
    {
      "switch_name": "Hayes E2L5",
      "nerda_switch_uid": "_8d7663f4-7870-47d9-b490-7cffe290f2d",
      "measurements": [
        {
          "nerda_measurement_id": "_522f0e6f-ef52-4a34-805b-9ba402343c4a",
          "unitMultiplier": "none",
          "measurementType": "SwitchPosition",
          "unitsSymbol": "none",
          "switch_id": "_de74c443-2f9d-480e-9841-3e21875d749d"
        },
        {
          "nerda_measurement_id": "_78c864b3-da2e-4c52-bf63-865603b1b7a4",
          "unitMultiplier": "none",
          "measurementType": "LineCurrent",
          "unitsSymbol": "A"
        }
      ]
    },
    {
      "switch_name": "Hayes E30L5",
      "nerda_switch_uid": "_f12d8f19-4bc2-4076-a9ba-c6d379f14788",
      "measurements": [
        {
          "nerda_measurement_id": "_593928e3-c759-48ac-9779-6f30c81114d5",
          "unitMultiplier": "none",
          "measurementType": "SwitchPosition",
          "unitsSymbol": "none",
          "switch_id": "_f42acef7-1c61-4419-9d1a-191e5f268861"
        }
      ]
    }
  ]
}

```

Figure 8: Nerda Static API Payload

Retrieving Time Series Data

Once a measurement of interest is identified, time series data is retrieved using the `nerdart_after` or `nerdart_between` endpoints.

The following parameters can be used on these endpoints:

Parameter	Description	Required / Optional
measurement	nerda_measurement_id to request data for, from the static endpoint	Required
after	Start date to obtain data from in yyyy-MM-dd'T'hh:mm:ss.SSS'Z' format	Required on both endpoints
before	End date to obtain data to in yyyy-MM-dd'T'hh:mm:ss.SSS'Z' format	Required on nerdart_between

In the example payload shown in Figure 8, the “LineCurrent” measurement for line `Hayes E24L5` has a `nerda_measurement_id` of “`_6a8dcaff-ea6b-4ea4-b78e-0a7f86c7bf61`”, so the “`nerdart_after`” endpoint URL can be constructed to be:

https://nerda.opengrid.com/api/nerdart_after-SEPD?measurement=_6a8dcaff-ea6b-4ea4-b78e-0a7f86c7bf61&after=2023-04-13T08:00:00.000Z

When this URL is accessed, the payload shown in Figure 9 is received.

In this case, data covering available data on the measurement alongside relevant readings is returned. Note that for some figures, a minimum, maximum and RMS (root mean squared) analogue value exists within the measurement reading representing the appropriate values for the measurement.

The API user should select the appropriate measurement, identified by the name of the analogue value, and then use the “value_history” array to retrieve the individual values for each timestamp.

The “nerdart_between” endpoint functions in the same manner, but allows end-time of the query to be specified in addition to the start-time.

Measurements are presented identically.

```

{
  "simulated": null,
  "measurementSource": null,
  "measurementID": null,
  "aliasPI": null,
  "aliasSCADA": null,
  "maxValue": null,
  "minValue": null,
  "normalValue": null,
  "positiveFlowIn": null,
  "AnalogValues": [
    {
      "__cid": "_bb24355b-b23c-48f6-a234-2f95dc2b9543",
      "__resource": "2732f18b-d906-4691-bcef-8588ad211fa7",
      "UUID": "_bb24355b-b23c-48f6-a234-2f95dc2b9543",
      "aliasName": "HAYE~E24L5~AMPS~AI",
      "description": null,
      "mRID": null,
      "name": "Hayes Hayes E24L5 A",
      "energyIdentCodeEic": null,
      "shortName": null,
      "nrn": null,
      "proposed": null,
      "UID": null,
      "sensorAccuracy": null,
      "timeStamp": "2023-04-16T03:03:03Z",
      "value": 43.5,
      "value_history": [
        {
          "__resource": "2732f18b-d906-4691-bcef-8588ad211fa7",
          "__cid": "_bb24355b-b23c-48f6-a234-2f95dc2b9543",
          "attribute": "value",
          "__ts": "2023-04-16T03:03:03Z",
          "value": 43.5
        },
        {
          "__resource": "2732f18b-d906-4691-bcef-8588ad211fa7",
          "__cid": "_bb24355b-b23c-48f6-a234-2f95dc2b9543",
          "attribute": "value",
          "__ts": "2023-04-16T03:00:06Z",
          "value": 43.5
        },
        {
          "__resource": "2732f18b-d906-4691-bcef-8588ad211fa7",
          "__cid": "_bb24355b-b23c-48f6-a234-2f95dc2b9543",
          "attribute": "value",
          "__ts": "2023-04-16T02:57:03Z",
          "value": 43.5
        },
        {
          "__resource": "2732f18b-d906-4691-bcef-8588ad211fa7",
          "__cid": "_bb24355b-b23c-48f6-a234-2f95dc2b9543",
          "attribute": "value",
          "__ts": "2023-04-16T02:54:02Z",
          "value": 43.5
        }
      ]
    }
  ]
}

```

Figure 9: Sample "After" Payload

NeRDA Contacts



NeRDA Portal:

nerda.opengrid.com



For general enquiries and API access please contact:

SSENSmart@sse.com



NeRDA page on SSEN website:

ssen.co.uk/our-services/tools-and-maps/nerda-portal