

# Partner API Usage Document

This document describes the remote MPOWER API for partners, which is part of enerXchange.

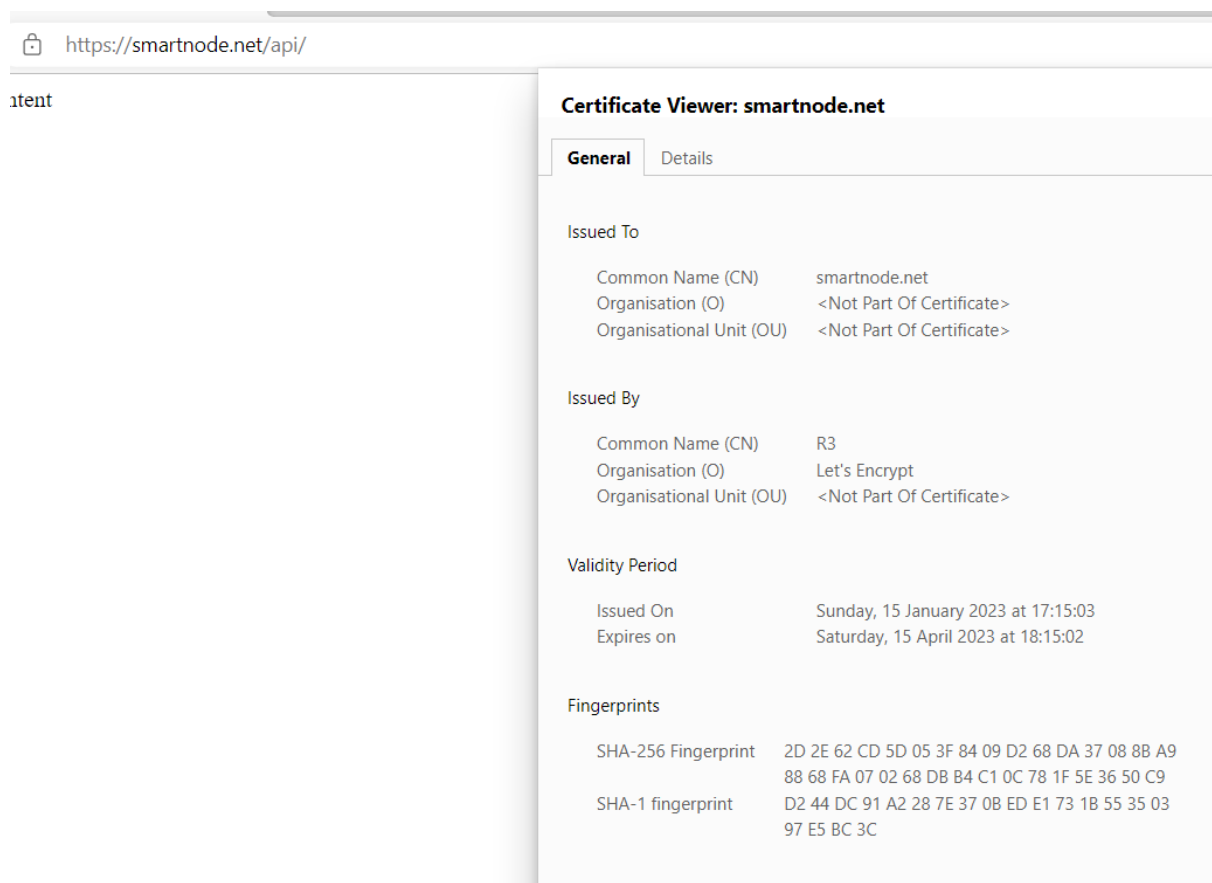
Postgres serves as the data source for the API to partners.

API is accessed through HTTPS requests by client. All requests can use GET method

## Server certificate and security

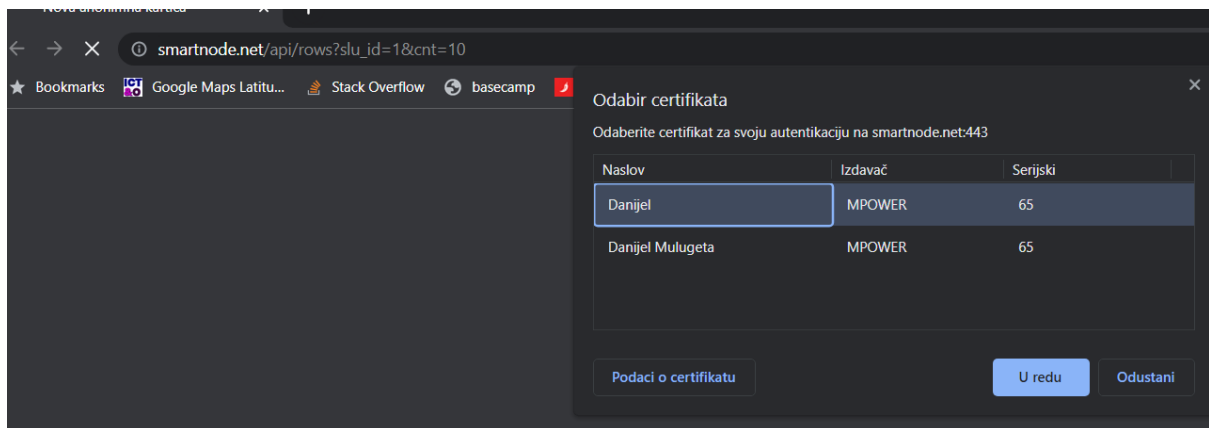
All data for Cityxchange is available at <https://smartnode.net/cityxchange/>

Server smartnode has a valid certificate so our partners can trust this server.  
Certificate can be verified in any of the browsers



## Partner API Usage – Client certificate

Partner API is accessible only with valid client certificate issued by MPOWER.  
As an example – if partner is accessing API through Chrome browser



The browser will force user to choose a client certificate.

Thus, any type of client ( Java, Python, C++ ... ) will be required to embed client certificate into its HTTP request.

Partner /api currently offers the following

- Query latest data from each SLU
- Query last N recorded data for chosen SLU
- Query data for chosen SLU in a time period

There is no delay and data is visible to partner the moment it is received from SLU.

## Get Latest Data

To get the latest data point from every SLU, send a GET request to:

<https://smartnode.net/cityxchange/latest>

Request has no parameters.

You will receive data as JSONArray in this JSON format:

```
[{"p1": "0.944000006", "p2": "0.916000009", "p3": "0.930000007", "k1": "56468.7031", "k2": "33536.7578", "k3": "40628.2188", "a1": "16.6000004", "a2": "8.5", "a3": "8.10000038", "t": "2023-03-01 08:38:08.489041", "slu_id": 5, "v1": 232, "v2": 229, "v3": 232}, {"p1": "0.99599992", "p2": "0.999000013", "p3": "0.975000024", "k1": "21433.2344", "k2": "27465.4434", "k3": "43973.6289", "a1": "17.7000008", "a2": "14.8000002", "a3": "5", "t": "2023-03-01 08:38:12.474987", "slu_id": 2, "v1": 230, "v2": 227, "v3": 232}, {"p1": "0.949000001", "p2": "0.963", "p3": "0.927999973", "k1": "477586.875", "k2": "463515.156", "k3": "430837.5", "a1": "47.9000015", "a2": "41.9000015", "a3": "37.5", "t": "2023-03-01 08:38:20.595864", "slu_id": 4, "v1": 233, "v2": 233, "v3": 234}, {"p1": "0.985000014", "p2": "0.981000006", "p3": "0.947000027", "k1": "71403.4609", "k2": "630
```

```

75.7031","k3": "42590.6602","a1": "28","a2": "25.8999996","a3": "9.10000038
","t": "2023-03-01
08:38:12.239053","slu_id": 1,"v1": 227,"v2": 229,"v3": 230},{ "p1": "0.99900
0013","p2": "0.950999975","p3": "0.992999971","k1": "33114.4648","k2": "242
00.4629","k3": "19345.0625","a1": "30.7000008","a2": "10.6000004","a3": "32
.9000015","t": "2023-03-01
08:38:06.918012","slu_id": 3,"v1": 227,"v2": 231,"v3": 231}]

```

## Get Number of Rows for a SLU

To get SLU data for a certain number of data points ( newest to oldest ) send a GET as

[https://smartnode.net/cityxchange/rows?slu\\_id=1&cnt=10](https://smartnode.net/cityxchange/rows?slu_id=1&cnt=10)

Request has parameters:

**slu\_id** : unit number for inquired SLU

**cnt** : number of rows to be returned

You will receive data in this JSON format:

```

{
  "data": [
    {
      "p1": "0",
      "p2": "0",
      "p3": "0",
      "k1": "0.074767001",
      "k2": "0.0906300023",
      "k3": "0",
      "a1": "0",
      "a2": "0",
      "a3": "0",
      "t": "2023-01-27 17:00:06.461941",
      "v1": 0,
      "v2": 225,
      "v3": 0
    },
    {
      "p1": "0",
      "p2": "0",
      "p3": "0",
      "k1": "0.074767001",
      "k2": "0.0906300023",
      "k3": "0",

```

```
"a1": "0",
"a2": "0",
"a3": "0",
"t": "2023-01-27 16:59:51.356999",
"v1": 0,
"v2": 225,
"v3": 0
},
{
  "p1": "2.25500011",
  "p2": "0",
  "p3": "0",
  "k1": "0",
  "k2": "0",
  "k3": "0",
  "a1": "0.00906299986",
  "a2": "0",
  "a3": "0",
  "t": "2023-01-27 16:59:36.373468",
  "v1": 0,
  "v2": 0,
  "v3": 0
},
{
  "p1": "0",
  "p2": "0",
  "p3": "0",
  "k1": "0.074767001",
  "k2": "0.0906300023",
  "k3": "0",
  "a1": "0",
  "a2": "0",
  "a3": "0",
  "t": "2023-01-27 16:59:06.290986",
  "v1": 0,
  "v2": 225,
  "v3": 0
},
{
  "p1": "0",
  "p2": "0",
  "p3": "0",
  "k1": "0.074767001",
  "k2": "0.0906300023",
  "k3": "0",
  "a1": "0",
  "a2": "0",
  "a3": "0",
```

```
"t": "2023-01-27 16:58:51.277361",
"v1": 0,
"v2": 229,
"v3": 0
},
{
  "p1": "0",
  "p2": "0",
  "p3": "0",
  "k1": "0.074767001",
  "k2": "0.0906300023",
  "k3": "0",
  "a1": "0",
  "a2": "0",
  "a3": "0",
  "t": "2023-01-27 16:58:37.254111",
  "v1": 0,
  "v2": 228,
  "v3": 0
},
{
  "p1": "0",
  "p2": "0",
  "p3": "0",
  "k1": "0.074767001",
  "k2": "0.0906300023",
  "k3": "0",
  "a1": "0",
  "a2": "0",
  "a3": "0",
  "t": "2023-01-27 16:58:22.237784",
  "v1": 0,
  "v2": 229,
  "v3": 0
},
{
  "p1": "0",
  "p2": "0",
  "p3": "0",
  "k1": "0.074767001",
  "k2": "0.0906300023",
  "k3": "0",
  "a1": "0",
  "a2": "0",
  "a3": "0",
  "t": "2023-01-27 16:58:07.253821",
  "v1": 0,
  "v2": 229,
```

```

    "v3": 0
  },
  {
    "p1": "0",
    "p2": "0",
    "p3": "0",
    "k1": "0.074767001",
    "k2": "0.0906300023",
    "k3": "0",
    "a1": "0",
    "a2": "0",
    "a3": "0",
    "t": "2023-01-27 16:57:52.214828",
    "v1": 0,
    "v2": 229,
    "v3": 0
  },
  {
    "p1": "0",
    "p2": "0",
    "p3": "0",
    "k1": "0.074767001",
    "k2": "0.0906300023",
    "k3": "0",
    "a1": "0",
    "a2": "0",
    "a3": "0",
    "t": "2023-01-27 16:57:37.190325",
    "v1": 0,
    "v2": 228,
    "v3": 0
  }
],
"slu_id": 1
}

```

## Get All Data for SLU in Given Time Period

To get SLU data from start datetime up till some end datetime, send a GET request as:

[https://smartnode.net/cityxchange/time?slu\\_id=1&begin=1674000000&end=1674830999](https://smartnode.net/cityxchange/time?slu_id=1&begin=1674000000&end=1674830999)

**slu\_id** : unit number for inquired SLU

**begin** : is a Unix epochtime of the beginning of the period for which to receive data

**end** : is a Unix epochtime of the end of period for which to receive data

Unix epochtime values should be in seconds ( not milliseconds )

You will receive data in this JSON format ( *Note* : Response will be limited to 10000 datapoints )

```
{ "data": [ { "p1": "0", "p2": "0", "p3": "0", "k1": "0.074767001", "k2": "0.0906300023",  
  "k3": "0", "a1": "0", "a2": "0", "a3": "0", "t": "2023-01-27  
17:00:06.461941", "v1": 0, "v2": 225, "v3": 0 }, { "p1": "0", "p2": "0", "p3": "0", "k1": "  
0.074767001", "k2": "0.0906300023", "k3": "0", "a1": "0", "a2": "0", "a3": "0", "t": "2  
023-01-27  
16:59:51.356999", "v1": 0, "v2": 225, "v3": 0 }, { "p1": "2.25500011", "p2": "0", "p3": "  
0", "k1": "0", "k2": "0", "k3": "0", "a1": "0.00906299986", "a2": "0", "a3": "0", "t": "2  
023-01-27  
16:59:36.373468", "v1": 0, "v2": 0, "v3": 0 }, { "p1": "0", "p2": "0", "p3": "0", "k1": "0.  
074767001", "k2": "0.0906300023", "k3": "0", "a1": "0", "a2": "0", "a3": "0", "t": "202  
3-01-27  
16:59:06.290986", "v1": 0, "v2": 225, "v3": 0 }, { "p1": "0", "p2": "0", "p3": "0", "k1": "  
0.074767001", "k2": "0.0906300023", "k3": "0", "a1": "0", "a2": "0", "a3": "0", "t": "2  
023-01-27  
16:58:51.277361", "v1": 0, "v2": 229, "v3": 0 }, { "p1": "0", "p2": "0", "p3": "0", "k1": "  
0.074767001", "k2": "0.0906300023", "k3": "0", "a1": "0", "a2": "0", "a3": "0", "t": "2  
023-01-27  
16:58:37.254111", "v1": 0, "v2": 228, "v3": 0 }, { "p1": "0", "p2": "0", "p3": "0", "k1": "  
0.074767001", "k2": "0.0906300023", "k3": "0", "a1": "0", "a2": "0", "a3": "0", "t": "2  
023-01-27  
16:58:22.237784", "v1": 0, "v2": 229, "v3": 0 }, { "p1": "0", "p2": "0", "p3": "0", "k1": "  
0.074767001", "k2": "0.0906300023", "k3": "0", "a1": "0", "a2": "0", "a3": "0", "t": "2  
023-01-27  
16:58:07.253821", "v1": 0, "v2": 229, "v3": 0 }, { "p1": "0", "p2": "0", "p3": "0", "k1": "  
0.074767001", "k2": "0.0906300023", "k3": "0", "a1": "0", "a2": "0", "a3": "0", "t": "2  
023-01-27 16:57:52.214828", "v1": 0, "v2": 229, "v3": 0 }, "slud_id": 1 }
```