Datav8 API Manual

Revision 1, June 2023

Inquiries for wholesale, technical support and API mailing list: admin@datav8.com

Introduction

This manual details how to access Datav8 data and parameters via a REST API. This allows bespoke applications to use the data captured by the Datav8 hardware.
Examples of such usage would include live monitoring dashboards/data walls, integration into building management systems and custom alarm systems.

It is expected that the initial setup of the Datav8 loggers is performed in the normal web interface at https://www.datav8.com. On this platform a user account will be set up and all required loggers added to this user account. Loggers must then be set up according to their specific module configuration (unless using a starter kit, in which case this is pre-configured). The user account details created in this way will be used to access the API and only the loggers accessible to this user account can be accessed through the API.

Loggers can be added to multiple user accounts so it is possible to have one (or multiple) dedicated accounts for API access and a different user account to manually log in via the normal web interface.

There are situations in which a higher degree of automation might be required, such as the ability to create a user account, add a logger to a user account or change logger parameters, however these are rare and thus are not included in this manual for the sake of brevity. Should these be required for your application please contact us for further details.

Logging In

Before any access can be made to the API a login procedure must be followed. This login procedure will result in a status code as well as a “token” used with further API calls, and lists of logger IMEI numbers and names.

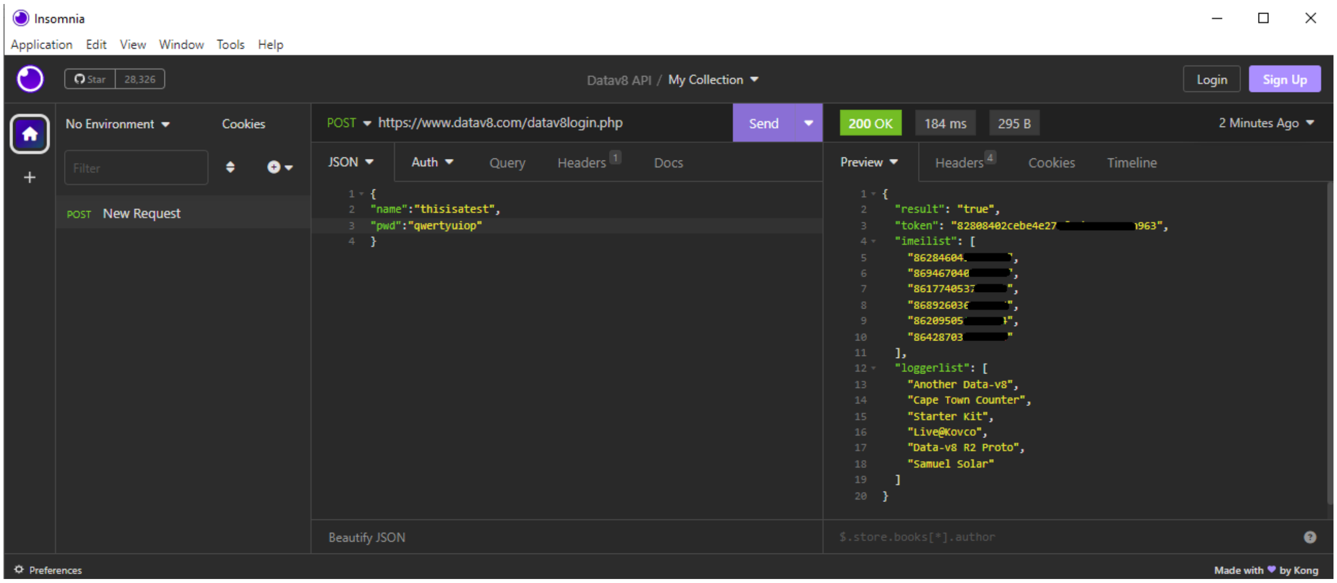
The login request is a POST made to the following URL:
https://www.datav8.com/datav8login.php
With the login request the following data is needed, in JSON format:

```
{
“name”: “Your_user_name”,
“pwd”: “Your_password_to_log_in”
}
```

The name can be either the chosen username or the nominated email address or cellphone number associated with the account.

If you wish to test this without writing code then the free “Insomnia” tool is recommended, it may be downloaded from https://insomnia.rest/download

Here is an Insomnia screenshot of a successful login:



```
{
"name":"thisisatest",
"pwd":"qwertyuiop"
}
```

```
{
  "result": "true",
  "token": "82808402cebe4e27[redacted]963",
  "imeilist": [
    "86284604[redacted]",
    "869467040[redacted]",
    "8617740537[redacted]",
    "868926036[redacted]",
    "86209505[redacted]",
    "86428703[redacted]"
  ],
  "loggerlist": [
    "Another Data-v8",
    "Cape Town Counter",
    "Starter Kit",
    "Live@Kovco",
    "Data-v8 R2 Proto",
    "Samuel Solar"
  ]
}
```

As shown above the login was attempted with the username “thisisatest” and password of “qwertyuiop”.
The login succeeded and thus the API returned four items.

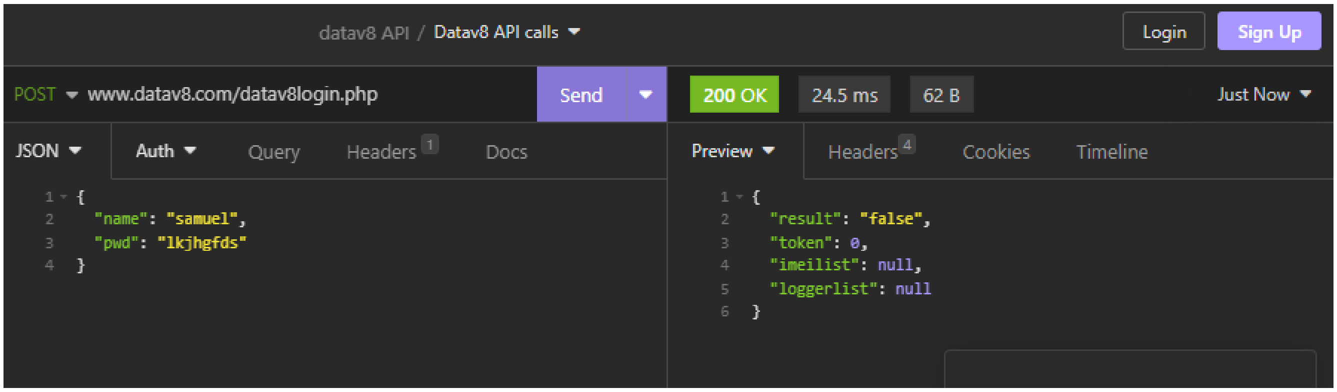
The first item, “result” takes on these values:
“true”: The login succeeded.
“emailnotvalidated”: The email address of the user has not been validated and thus the account is not active.
“false”: The login failed because the username or password are incorrect.

The second item is a token which is a randomized string. Further calls to the API will require this token to be used. Tokens are typically valid for a week after a successful log in and do not change if another log in occurs within the token’s validity period. It is recommended that all actions on the API are preceded by a login order to extend token validity and prevent errors due to invalid tokens in later API calls.

The third item is a list of the IMEI numbers that have been linked to this account. These are the only loggers that may be accessed for this account. IMEI numbers are unique to each data logger and this is ensured in manufacturing.

The fourth item is a list of names given to the data loggers associated with the account. Names are user allocated for ease of reference and thus may be non-unique. The names and IMEI numbers are given in the same order.

Here is a screenshot of a failed login attempt:



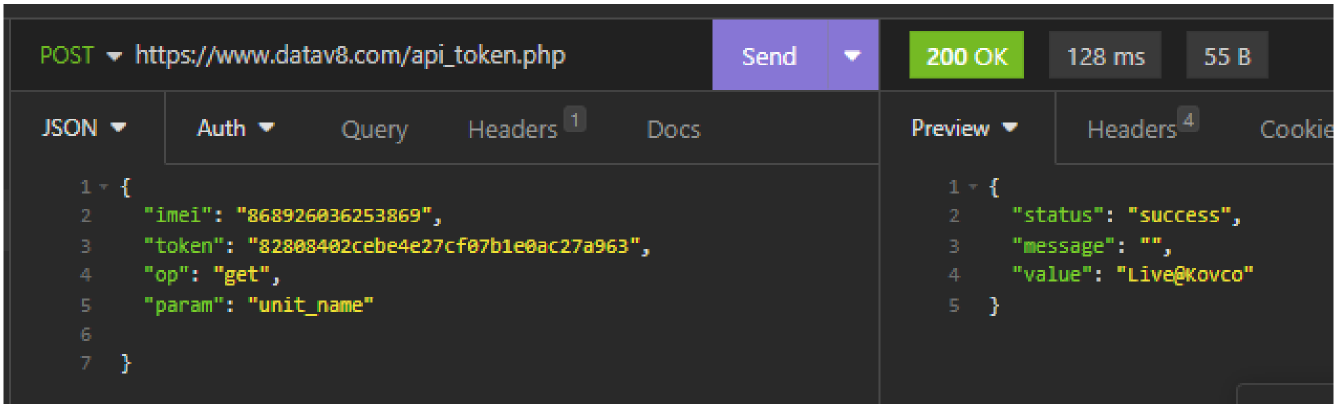
```
{
  "name": "samuel",
  "pwd": "lkjhgfds"
}
```

```
{
  "result": "false",
  "token": 0,
  "imeilist": null,
  "loggerlist": null
}
```

Obtaining a Logger’s Parameters

The parameters of the overall logger can be obtained by means of the next set of API calls. This is a POST made to https://www.datav8.com/api_token.php with a JSON data structure that specifies which logger is being referred to, the token returned by the login process, an operation name and the name of the parameter in question.

In this example we query a logger for its name:



```
{
  "imei": "868926036253869",
  "token": "82808402cebe4e27cf07b1e0ac27a963",
  "op": "get",
  "param": "unit_name"

}
```

```
{
  "status": "success",
  "message": "",
  "value": "Live@Kovco"
}
```

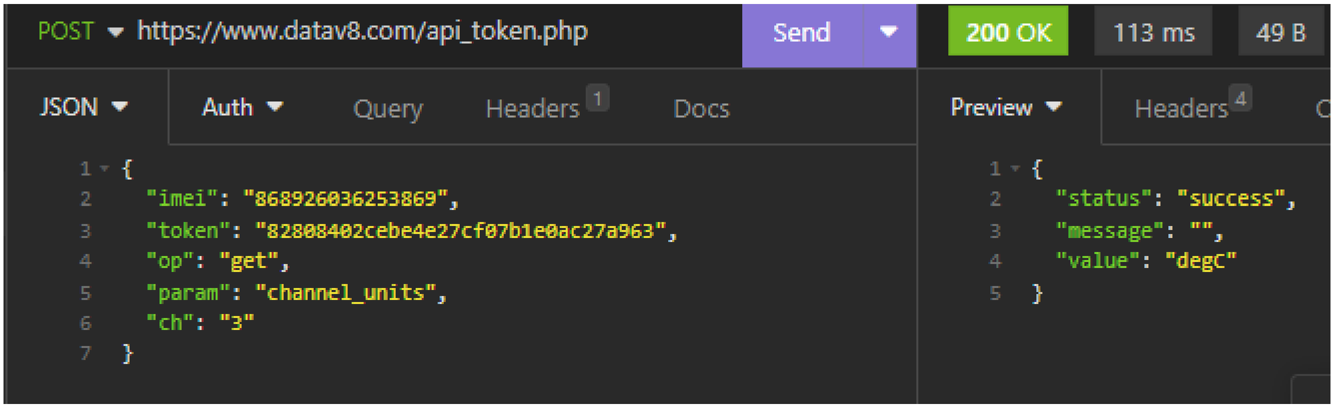
We supplied a valid IMEI number and token which were returned by the login API call. The “op” variable is set to “get” which is used for obtaining a logger’s parameters and the “param” was set to “unit_name” to indicate that we are retrieving the user-allocated logger name.

The following are valid values for param:

Parameter	Returned information	Channel number needed
unit_name	User allocated name for the logger	No
unit_location	User allocated name for the logger’s location	No
monitor_heartbeat	Whether the server will send an alert email if the logger stops reporting in.	No

unit_owner	The user allocated name of the logger’s owner	No
expiry	The expiry date for the logging contract	No
channel_name	The user allocated name of the channel	Yes
thresh_upper	The upper alarm threshold for the channel	Yes
thresh_lower	The lower alarm threshold for the channel	Yes
alarm_delay	The number of minutes for which the channel value must exceed a threshold before an alarm is triggered	Yes
channel_active	Whether the channel is in use	Yes
monitor_active	Whether the channel is set to trigger an alarm if the threshold is exceeded.	Yes
channel_units	The units of measurement applied to the channel	Yes
sensor_type	The type of sensor connected to the channel	Yes
criticalalarm	If this is true then an alarm is sent every hour for as long as a threshold is exceeded.	Yes

Here is an example of a call which queries a data logger for the units of measurement on channel 3:



POST https://www.datav8.com/api_token.php

```
{
  "imei": "868926036253869",
  "token": "82808402cebe4e27cf07b1e0ac27a963",
  "op": "get",
  "param": "channel_units",
  "ch": "3"
}
```

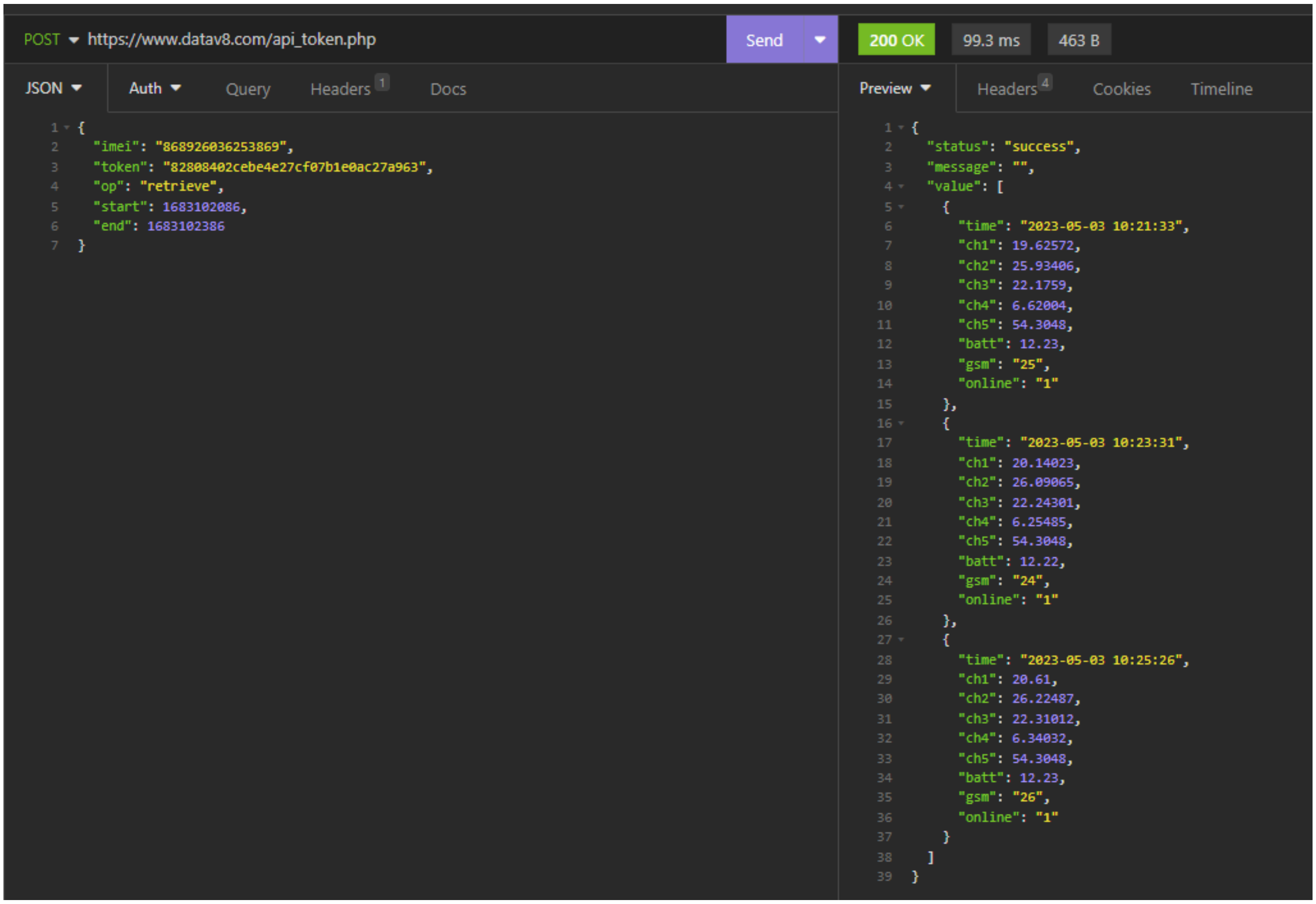
```
{
  "status": "success",
  "message": "",
  "value": "degC"
}
```

In this example the param was set to “channel_units” from the table above and the channel was set to 3 by means of setting “ch” to “3”.
Valid channels are 1-5 as the loggers each have five physical channels.

The status field in the return will indicate success or failure. In the case of success the message field will be empty. In the case of failure the message field will indicate the reason. Possible failure messages are as detailed later in this document.

Retrieving Logged Data

When retrieving data the operation is “retrieve”. In addition the IMEI and token are needed. The start and end times of the data are needed in the standard Unix epoch time format. Here is an example of a data retrieval call:



```
https://www.datav8.com/api_token.php
```

```
{
  "imei": "868926036253869",
  "token": "82808402cebe4e27cf07b1e0ac27a963",
  "op": "retrieve",
  "start": 1683102086,
  "end": 1683102386
}
```

```
{
  "status": "success",
  "message": "",
  "value": [
    {
      "time": "2023-05-03 10:21:33",
      "ch1": 19.62572,
      "ch2": 25.93406,
      "ch3": 22.1759,
      "ch4": 6.62004,
      "ch5": 54.3048,
      "batt": 12.23,
      "gsm": "25",
      "online": "1"
    },
    {
      "time": "2023-05-03 10:23:31",
      "ch1": 20.14023,
      "ch2": 26.09065,
      "ch3": 22.24301,
      "ch4": 6.25485,
      "ch5": 54.3048,
      "batt": 12.22,
      "gsm": "24",
      "online": "1"
    },
    {
      "time": "2023-05-03 10:25:26",
      "ch1": 20.61,
      "ch2": 26.22487,
      "ch3": 22.31012,
      "ch4": 6.34032,
      "ch5": 54.3048,
      "batt": 12.23,
      "gsm": "26",
      "online": "1"
    }
  ]
}
```

The value objects each have the same structure as shown in the table below.

Name	**Meaning**
time	The timestamp of the reading. The format is YYYY-MM-DD HH:mm:ss times are in South African standard time.
ch1	The reading taken on channel 1. The units of measurement may be retrieved with a “get” operation as shown above.
ch2	The reading taken on channel 2. The units of measurement may be retrieved with a “get” operation as shown above.
ch3	The reading taken on channel 3. The units of measurement may be retrieved with a “get” operation as shown above.

ch4	The reading taken on channel 4. The units of measurement may be retrieved with a “get” operation as shown above.
ch5	The reading taken on channel 5. The units of measurement may be retrieved with a “get” operation as shown above.
batt	The supply voltage to the logger in Volts
gsm	The GSM CSQ level on a scale of 0-31 where 0 is no signal and 31 is the maximum possible signal
online	Loggers have three hours of onboard memory to retain logs in the case of network failure or lack of airtime. Data points which are captured into this memory are uploaded once the network becomes available again. These are considered “offline” logs because they are not uploaded in real time. Data points which offline will have an online value of “0”. Data points which are uploaded in real time will have an online value of “1”.

Failure Messages

The following failure messages may be produced by the API:

Message	**Cause**
invalid imei token pair	The token is not valid for the IMEI number specified. This can occur if using an expired token to access data or if the IMEI number is not associated with the token in the query. Loggers must be linked to a user account before accessing them. Normally this is done on the Datav8 web interface but may be automated if needed through extended API calls not covered in this document.
invalid op	The operation requested is not valid.
invalid start	When retrieving data the start time may not be before Jan 1 2015 and may not be later than the present time.
invalid end	When retrieving data the end time may not be before Jan 1 2015 and may not be later than the present time.

start>end	When retrieving data the start time must be before the end time.
invalid param	The parameter requested is not available or is invalid
invalid channel	Channels must be between 1 and 5
invalid value	Some parameters can only take on a limited range of values. As an example “channel_active” must be either “0” (disabled) or “1” (enabled). Other values are invalid.