Growbud REST API Documentation

This document provides information on how to use the Growbud REST API for querying sensor data.

Base URL

The base URL for the API will be provided to you separately. You will need to use this URL in your API requests.

Authentication

All API requests require authentication using a Bearer token. You will be provided with a unique API token for your account.

Include the following header in all requests:

```
Authorization: Bearer <your-api-token>
```

Replace `<your-api-token>` with the token provided to you.

API Credits

Each API request consumes credits from your account. You can purchase additional credits by contacting support.

Endpoints

1. Query Growbud Data

Retrieve sensor data from Growbud based on specified parameters.

- **URL**: `/api/query`
- **Method**: `GET`
- **Auth required**: Yes

Query Parameters

Parameter	Type	Description
duration	string	Duration of data to retrieve in hours (e.g., "24" for last 24 hours)
serialNumbers	string	List of sensor serial numbers in square brackets
format	string	(Optional) Response format: 'json' (default) or 'csv'

Success Response for JSON format (default)

- **Code**: 200 OK

- **Content-Type**: application/json
- **Content**: JSON object with metadata and data arrays for each sensor


```
{
  "E200201B": {
    "metadata": {
      "fields": ["_time", "air_temp", "humidity", "soil_vwc", "pore_ec",
"bulk_ec", "vpd", "battery_level", "rssi"],
      "units": {
        "air_temp": "°F",
        "humidity": "%",
        "soil_vwc": "%",
        "pore_ec": "mS/cm",
        "bulk_ec": "mS/cm",
        "vpd": "kPa",
        "battery_level": "V",
        "rssi": "dBm"
      }
    },
    "data": [
      {"_time": "2023-07-13T12:00:00Z", "air_temp": 79.33, "humidity":
71.39, "soil_vwc": 65.32, "pore_ec": 6.08, "bulk_ec": 3.63, "vpd": 0.98,
"battery_level": 7.43, "rssi": -71.56},
      {"_time": "2023-07-13T12:15:00Z", "air_temp": 79.45, "humidity":
71.22, "soil_vwc": 65.30, "pore_ec": 6.10, "bulk_ec": 3.64, "vpd": 0.99,
"battery_level": 7.43, "rssi": -71.58}
    ]
  },
  "A2002022": {
    "metadata": {
      "fields": ["_time", "air_temp", "humidity", "co2", "vpd",
"battery_level", "rssi"],
      "units": {
        "air_temp": "°F",
        "humidity": "%",
        "co2": "ppm",
        "vpd": "kPa",
        "battery_level": "V",
        "rssi": "dBm"
      }
    },
    "data": [
      {"_time": "2023-07-13T12:00:00Z", "air_temp": 79.19, "humidity":
74.18, "co2": 450, "vpd": 0.88, "battery_level": 7.40, "rssi": -71.03},
      {"_time": "2023-07-13T12:15:00Z", "air_temp": 79.24, "humidity":
74.05, "co2": 455, "vpd": 0.89, "battery_level": 7.40, "rssi": -71.05}
    ]
  }
}
```


Success Response for CSV format

- **Code**: 200 OK
- **Content-Type**: text/csv
- **Content**: CSV data with headers for each sensor

```
Sensor:E200201B
_time,air_temp,humidity,soil_vwc,pore_ec,bulk_ec,vpd,battery_level,rssi
2023-07-13T12:00:00Z,79.33,71.39,65.32,6.08,3.63,0.98,7.43,-71.56
2023-07-13T12:15:00Z,79.45,71.22,65.30,6.10,3.64,0.99,7.43,-71.58

Sensor:A2002022
_time,air_temp,humidity,co2,vpd,battery_level,rssi
2023-07-13T12:00:00Z,79.19,74.18,450,0.88,7.40,-71.03
2023-07-13T12:15:00Z,79.24,74.05,455,0.89,7.40,-71.05
```

Error Responses

- **Code**: 400 Bad Request
  - **Content**: `Missing required query parameters` or `Invalid duration parameter`
- **Code**: 401 Unauthorized
  - **Content**: `Unauthorized: No token provided` or `Unauthorized: Invalid token`
- **Code**: 403 Forbidden
  - **Content**: `No authorized sensors requested` or `API credits exhausted. Please recharge your account.`
- **Code**: 404 Not Found
  - **Content**: `User not found`
- **Code**: 500 Internal Server Error
  - **Content**: `Error querying data` or `Something broke!`

2. List Accessible Sensors

Retrieve a list of sensor serial numbers that the authenticated user has access to.

- **URL**: `/api/list-sensors`
- **Method**: `GET`
- **Auth required**: Yes

Success Response

- **Code**: 200 OK
- **Content-Type**: application/json
- **Content**: JSON object with an array of serial numbers

```
{
  "serialNumbers": ["E200201B", "E2002022", "A2002033"]
}
```

Error Responses

- Same as the Query Growbud Data endpoint

Usage Examples

Querying Growbud Data (JSON format)

Here's an example of how to use the query API with JavaScript fetch to get JSON data:

```
const apiUrl = 'YOUR_API_URL';
const apiToken = 'YOUR_API_TOKEN';

fetch(`${apiUrl}/api/query?duration=24&serialNumbers=[E200201B,E2002022]`,
{
  headers: {
    'Authorization': `Bearer ${apiToken}`
  }
})
.then(response => response.json())
.then(data => {
  console.log('Sensor E200201B data:', data.E200201B);
  console.log('Sensor E2002022 data:', data.E2002022);
})
.catch(error => console.error('Error:', error));
```

Querying Growbud Data (CSV format)

Here's an example of how to use the query API to get CSV data:

```
const apiUrl = 'YOUR_API_URL';
const apiToken = 'YOUR_API_TOKEN';

fetch(`${apiUrl}/api/query?duration=24&serialNumbers=
[E200201B,E2002022]&format=csv`, {
  headers: {
    'Authorization': `Bearer ${apiToken}`
  }
})
.then(response => response.text())
.then(csvData => {
  console.log('CSV Data:', csvData);
  // You can then parse this CSV data using a CSV parsing library
```

```
})
.catch(error => console.error('Error:', error));
```

Listing Accessible Sensors

Here's an example of how to use the list-sensors API:

```
const apiUrl = 'YOUR_API_URL';
const apiToken = 'YOUR_API_TOKEN';

fetch(`${apiUrl}/api/list-sensors`, {
  headers: {
    'Authorization': `Bearer ${apiToken}`
  }
})
.then(response => response.json())
.then(data => {
  console.log('Accessible Sensors:', data.serialNumbers);
})
.catch(error => console.error('Error:', error));
```

Notes

- JSON is the default format. Use `format=csv` in the query parameters to receive data in CSV format.
- The `duration` parameter specifies how many hours of data to retrieve, counting back from the current time.
- You can request data for multiple sensors by providing a list of serial numbers in square brackets, e.g., `[E200201B,E2002022]`.
- The API will only return data for sensors that your account has access to.
- If you request data for sensors you don't have access to, those sensors will be omitted from the response without causing an error.
- Large queries may take some time to process. Consider limiting the duration or the number of sensors if you're dealing with large datasets.
- For large datasets, consider using the CSV format for more efficient data transfer.
- Keep your API token secure. Do not share it or expose it in client-side code.
- If you need to use this API in a client-side application, consider setting up a proxy server to make the API calls and handle the token securely.
- Use the `/api/list-sensors` endpoint to get an up-to-date list of serial numbers you have access to before making queries.
- Data is aggregated in 15-minute intervals.
- All numeric values are rounded to two decimal places.
- Timestamps are rounded to the nearest minute.
- CO2 sensors (serial numbers starting with 'A') have a different set of fields compared to standard sensors.

Support

For any issues or questions regarding the API, if you need to regenerate your API token, or if you need to purchase additional API credits, please contact our support team at support@growbud.io