

Sunvec Connect

User API **v2.0**

**Version
Information**

Version	Editor	Time	Remark
V2.0	Edgar Mena	2024-11-26	Creating

Contents

1. Introduction	1
2. Application Scope	1
3. Operating Procedures.....	1
4. HTTP Request Paradigm	1
4.1. Interface Format and Protocol.....	1
5. Common Parameter	2
5.1. TokenId	2
5.2. Common Return Object.....	2
6. API Permission	3
7. Active Query of Device Data.....	3
7.1. Query Real-time Data.....	3
8. Appendix.....	6
8.1. Device Status Mapping	6
8.2. Error Code	7

1. Introduction

The **Sunvec Connect** API provides remote capability calls for third-party. The third party sends HTTPS/HTTP GET or POST requests to the API server to query, modify and remotely control **Sunvec Connect** platform data.

The third party needs to add corresponding parameters in the request message according to the API instructions to call the API. **Sunvec Connect** platform will return the processing results based on the processing of the request.

2. Application Scope

This document is applicable to the Sunvec Connect Cloud platform and the inverters of Sunvec Smart Solar Inverter.

3. Operating Procedures

1. Log in <https://portal.sunvec-x.com/#/login> to register an account.
2. Obtain the unique tokenId through the menu [service-API].
3. Call the API according to the [Sunvec-API Interface] document.

4. HTTP Request Paradigm

4.1. Interface Format and Protocol

The API operates the platform data through Https protocol and returns the results in JSONformat.

Interface address:

https://<URL>/api/<API_VERSION>/<RESOURCE_NAME>/<RESOURCE_ACTION>

Notice:

(1) URL: access domain, the specific API address can be obtained according to the menu[service-API].

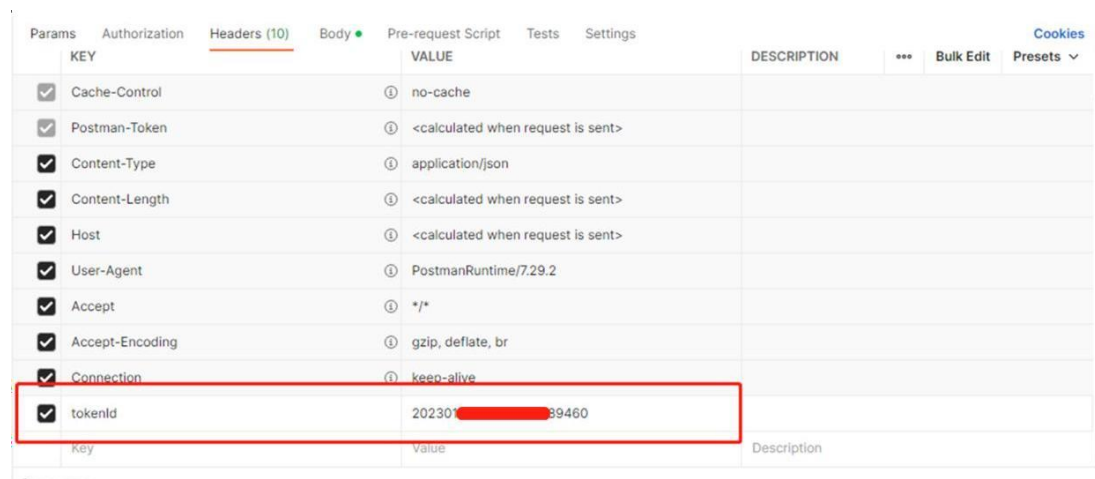
- (2) API_VERSION: version number, the current version number is v2.
- (3) RESOURCE_NAME: indicates the resource to be operated, such as userMng,distributorMng, siteMng, etc.
- (4) RESOURCE_ACTION: indicates the action on the resource, such as list, add, etc.

5. Common Parameter

5.1. tokenId

Parameter	Notes
tokenId	Get from [service-API]. Please keep the token safe and do not leak it.

Notice: When calling the interface, tokenId needs to be placed in Headers.



The screenshot shows the Postman Headers tab with a list of headers. The 'tokenId' header is highlighted with a red box. The value for 'tokenId' is '20230' followed by a redacted area and '89460'.

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> Cache-Control	no-cache	
<input checked="" type="checkbox"/> Postman-Token	<calculated when request is sent>	
<input checked="" type="checkbox"/> Content-Type	application/json	
<input checked="" type="checkbox"/> Content-Length	<calculated when request is sent>	
<input checked="" type="checkbox"/> Host	<calculated when request is sent>	
<input checked="" type="checkbox"/> User-Agent	PostmanRuntime/7.29.2	
<input checked="" type="checkbox"/> Accept	/*/*	
<input checked="" type="checkbox"/> Accept-Encoding	gzip, deflate, br	
<input checked="" type="checkbox"/> Connection	keep-alive	
<input checked="" type="checkbox"/> tokenId	20230' [REDACTED] 89460	

When commissioning the interface, the interface testing tool [Postman] can be used to commission and place tokens in Headers.

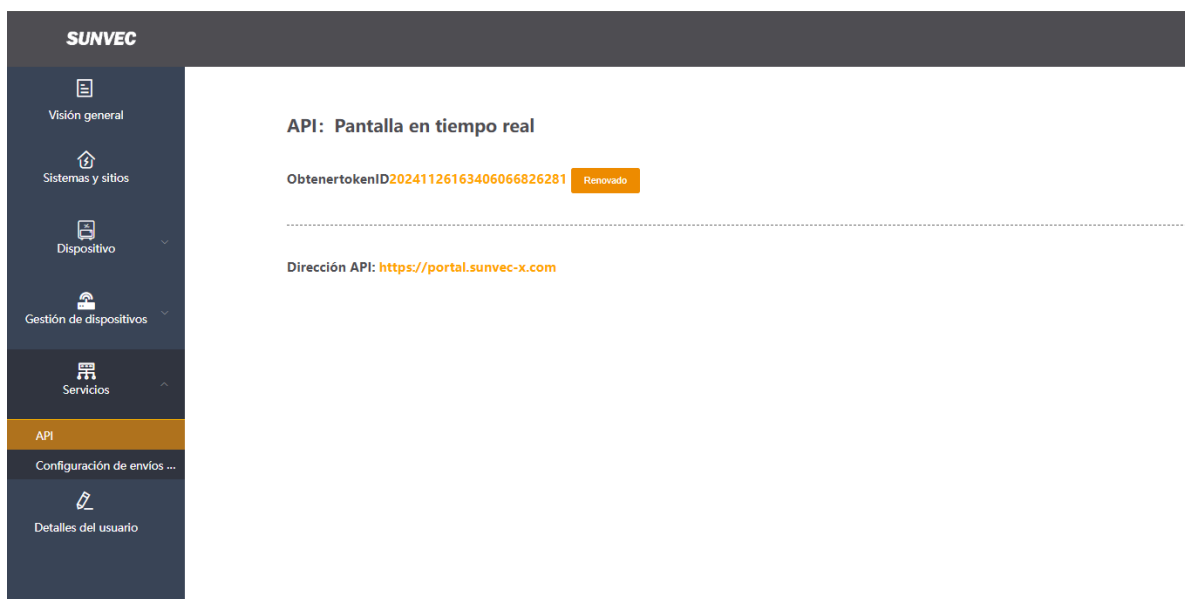
5.2. Common Return Object

Name	Type	Description
success	Boolean	true=successful, false=unsuccessful
exception	String	Error message
result	Object	Concrete data returned
code	Integer	Concrete status code returned

6. API Permission

When users request an interface, they must use tokenId to submit an API request to Sunvec **Connect Cloud**. The tokenId request path is [service-API].

Log in to **Sunvec Connect Cloud**, click [service], enter the API page, and obtain tokenId. After the request is successful, tokenId will be bound to the distributor/installer account.



7. Active Query of Device Data

7.1. Query Real-time Data

The format of requested data

Type	Definition	Must	Description
Partial URL	/api/v2/dataAccess/realtimel nfo /get		
Content Type	application/json		
HTTP Method	POST		
Request Content	wifiSn	Yes	Registration Number

Example

Request address

<https://<URL>/api/v2/dataAccess/realtimeInfo/get>

Request message

```
{
  "wifiSn": "SUT****VB1"
}
```

Returned message

```
{
  "success": true,
  "exception": "Query success!",
  "result": {
    "inverterSN": "H34*****6008",
    "sn": "SR*****RM",
    "acpower": 152,
    "yieldtoday": 0.2,
    "yieldtotal": 2420.5,
    "feedinpower": 0,
    "feedinenergy": 0,
    "consumeenergy": 0,
    "feedinpowerM2": null,
    "soc": null,
    "peps1": null,
    "peps2": null,
    "peps3": null,
    "inverterType": "4",
    "inverterStatus": "102",
    "uploadTime": "2024-08-20 09:21:43",
    "batPower": null,
    "powerdc1": 162,
    "powerdc2": 0,
    "powerdc3": null,
    "powerdc4": null,
    "batStatus": null
  },
  "code": 0
}
```

```
}

```

Explanation of the return

Name	Type	Description
success	Boolean	ttrue=successful, false=unsuccessful
exception	String	Error message
result	Object	Concrete data returned
inverterSN	String	Inverter SN
sn	String	Wi-Fi module SN
acpower	Float	AC output power
yieldtoday	Float	Today's yield
yieldtotal	Float	Total yield
feedinpower	Float	Power supplied to the grid
feedinenergy	Float	Total grid export power
consumeenergy	Float	Total grid import power
feedinpowerM2	Float	Meter 2 power
soc	Float	Battery capacity
peps1	Float	EPS A phase active power
peps2	Float	EPS B phase active power
peps3	Float	EPS C phase active power
inverterType	String	Inverter type
inverterStatus	String	Inverter operating conditions, see Appendix 8.1. "Device status mapping" for values
uploadTime	String	Upload time format 2016-10-26 17:33:01
batPower	Float	Battery terminal power
powerdc1	Float	PV1 input power
powerdc2	Float	PV2 input power
powerdc3	Float	PV3 input power
powerdc4	Float	PV4 input power

batStatus	String	Battery status: 0 normal, 1 fault, 2 disconnected
code	Integer	0: Success, see Appendix 8.2. "Error Code"for other codes

8. Appendix

8.1. Device Status Mapping

Inverter

Code	Status
100	Waiting for operation
101	Self-test
102	Normal
103	Recoverable fault
104	Permanent fault
105	Firmware upgrade
106	EPS detection
107	Off-grid
108	Self-test mode (Italian safety regulations)
109	Sleep mode
110	Standby mode
111	Photovoltaic wake-up battery mode
112	Generator detection mode
113	Generator mode
114	Fast shutdown standby mode
130	VPP mode
131	TOU-Self use
132	TOU-Charging

133	TOU-Discharging
-----	-----------------

8.2. Error Code

Error code	Returned Message
1001	Interface Unauthorized
1002	Parameter validation failed
1003	Data Unauthorized
1004	Duplicate data
2001	Operation failed
2002	Data not found