



# Crestron Module Documentation for



# THERMOSTATS



## GENERAL INFORMATION

SIMPLWINDOWS NAME:	Crestron Nvent Nuheat API Demo (v1.2).smw								
CATEGORY:	Web API								
VERSION:	v1								
SUMMARY:	This demo includes the CORE module and the TSTAT module (one per thermostat) allowing communication with Nvent Nuheat web server to retrieve information and allow basic controls of the Nvent Nuheat thermostats. <a href="https://api.mynuheat.com/">https://api.mynuheat.com/</a>								
GENERAL NOTES:	<ul style="list-style-type: none"> <li>- Each Nvent Nuheat thermostats must be registered under the customers web account. <a href="https://identity.mynuheat.com/account/login?">https://identity.mynuheat.com/account/login?</a></li> <li>- The API access must be requested (at the bottom of the page) to obtain a client ID and secret key. The redirect URI must be specified as <a href="http://localhost">http://localhost</a>. <a href="https://go.nvent.com/connected-home.html">https://go.nvent.com/connected-home.html</a></li> <li>- The Crestron processor will generate the URL to reach the content access form the users must agree on (can be done on a TSW panel or via a computer browser).</li> <li>- Once agreed, copy the return URL and sent it to the Crestron processor using the setup page on the TSW or toolbox.</li> <li>- The tstat list summary is used to retrieve the number and locations of tstats, as well as to know if data on a tstat has changed.</li> <li>- Burst and Rate limits</li> </ul> <p>Rate limits: the overall "reasonable" allowed rate of calls to the API over a longer period. Usually defined in days and months. Eg. 250.000 in a month.</p> <p>Burst limits: the limit of allowed request in a short timeframe, usually defined in seconds or minutes.</p> <p><b>i</b> Current limits: these limits are defined as a per user &amp; per client basis.</p> <div style="display: flex; align-items: flex-start;"> <div style="flex: 1;"> </div> <table border="1" style="margin-left: 20px;"> <tbody> <tr> <td>/second</td> <td>50</td> </tr> <tr> <td>/30 min</td> <td>750</td> </tr> <tr> <td>/12 hrs</td> <td>20,000</td> </tr> <tr> <td>/week</td> <td>250,000</td> </tr> </tbody> </table> </div>	/second	50	/30 min	750	/12 hrs	20,000	/week	250,000
/second	50								
/30 min	750								
/12 hrs	20,000								
/week	250,000								
CRESTRON HARDWARE REQUIRED:	3-series/4-series processor connected to the Internet								



## CORE MODULE

### Parameters:

CLIENT_ID	S	As obtained when registered for API access.
CLIENT_SECRET	S	As obtained when registered for API access.
FILE_PATH	S	The location storage on the processor for the tokens.
POLLING_SPEED	S	Specified time interval for data refresh.

### Input Signals:

INIT	D	Pulse to initialize the module.
AUTH_CODE_URL\$	S	Input the URL returned from Nvent Nuheat after the content access form was agreed.
GET_TOKENS	D	Pulse to retrieve initial set of tokens once access has been granted.
REFRESH_TOKENS	D	Pulse to force a manual refresh of the tokens.
STATUS_CODE_CLEAR	D	Pulse to close status message popup (warnings or errors from Nvent Nuheat).
SETPOINT_SENT_AS_C	D	Set to 1 if the setpoints will be sent as Celsius, set to 0 if sent as Fahrenheit.
GET_TSTAT_SUMMARY	D	Pulse to poll periodically the list of tstats/changes.
SUSPEND_POLLING	D	Set high when setpoints are adjusted on the UI.

### Output Signals:

BUSY	D	High if module is busy processing data.
GET_AUTH_CODE_URL\$	S	The URL to use to grant access to the processor (can be pushed to the TSW or copy to a computer browser).
TOKEN_EXPIRY	A	Current token validity countdown.
STATUS_CODE_NOTIF	D	High when a warning/error status code has been received from Nvent Nuheat server.
STATUS_CODE	A	Warning/error status code received.
STATUS_MESSAGE\$	S	Warning/error status message received.
NUMBER_OF_TSTATS	A	Number of tstats on the account.
POLL_TSTAT_INSTANCE	A	Used by the Core module to refresh tstats info.



## TSTAT MODULE

### Parameters:

INDEX	A	Unique instance number for this tstat.
-------	---	--

### Input Signals:

REFRESH_TSTAT	A	Instance ID received from the Core module if the tstat info needs to be refreshed.
GET_INFO	D	Pulse to refresh the tstat info.
SEND_UPDATE	D	Pulse to send updates settings (setpoint, mode, etc).
SET_HEAT	A	4-digits value to be sent for heat (ie. 2200 = 22 deg. C).
SCHEDULE_MODE	D	Pulse the corresponding schedule mode to be sent (see below).

### Output Signals:

OFFLINE	D	Will report if tstat is offline.
BUSY	D	High if module is busy processing data.
NAME\$	S	Name/location set on the tstat.
IDENTIFIER\$	S	Serial# of the tstat.
GROUP_ID	A	Reports the group ID if tstat is grouped with other tstats.
OPERATING_MODE	D	Indicates the operating mode (see below).
IS_HEATING	D	Current heating status.
CURRENT_TEMP	A	Current temperature in Celsius and Fahrenheit.
SETPOINT	A	Current heat setpoint feedback.
SCHEDULE_MODE	D	Current schedule mode feedback.
ERROR\$	\$	Will report errors with this tstat.



## NOTES

### Nvent Nuheat operating mode:

The tstat will report its operating mode as set on the device itself.

In Auto mode, the thermostat uses its schedule to adjust the temperature during the day. When the thermostat is set to Operating Mode Manual, the thermostats Schedule and ScheduleMode functionality is disabled, and only the SetPointTemp can be changed.

### Nvent Nuheat schedule mode:

When using Auto, the tstat will use the schedule to define the temperature it should be set to. When using Hold, you temporarily set the temperature until the next schedule event. When using Permanent, the temperature is held indefinitely.

### Programming recommendation:

The tstats list order may change as devices are added or removed, potentially shifting tstats in the list, you can use the name or identifier fields to route controls/FB (via crosspoints or buffers) to the proper room on your UI.

### Testing:

OPS USED FOR TESTING:	DIN-AP3 - v1.601.3935.28796
SIMPL WINDOWS:	4.14.20
DEVICE DB:	200.01.001.00
CRESTRON DB:	200.00.004.00
SIMPL# LIBRARY:	NventNuheatLib_v1
DEVICE API:	Nvent Nuheat API v1

### Revision History:

Date	Comments
February.2020	v1 Initial Release
August.2020	v1 Recompile with 200 DB



## INTEGRATION GUIDE

**Step 1: Request API access on nVent developer portal at the bottom of the page, to obtain your Client ID and Secret (specify to get it from Wally Lo ([Wally.Lo@nVent.com](mailto:Wally.Lo@nVent.com)) and <http://localhost> as the redirect URI in the “Intended Use for API” box):**

<https://go.nvent.com/connected-home.html>

ARE YOU A SOFTWARE DEVELOPER OR PROGRAMMER INTERESTED IN GETTING STARTED WITH THE SIGNATURE API?

The Signature API is a REST API for interacting with the nVent NUHEAT Signature thermostat. We hope this will help you integrate electric floor heating systems the way you want to. We use familiar HTTP conventions (GET, PUT, POST, DELETE), meaningful URLs, authentication and status codes for easy consumption of the API with off-the-shelf clients. All responses are returned as JSON.

The full API documentation and API access is only available to registered users. Please complete the request form and become a member of our exciting developer community for the nVent NUHEAT Signature thermostat.

\* First Name:  \* Last Name:   
\* Address:  Address 2:   
\* Job Title:  \* Company Name:   
\* Postal Code:  \* Country:   
\* Email Address:  \* Phone Number:   
\* Intended Use for API:   
\* Desired Client Name:  \* Redirect URL:   
 I agree to the [Terms & Conditions](#) as they pertain to nVent NUHEAT's Signature API documentation.  
nVent is committed to protecting and respecting your privacy, and we'll only use your personal information to administer your account and to provide the products and services you requested from us. You may unsubscribe from these communications at any time. For more information on how to unsubscribe, our privacy practices, and how we are committed to protecting and respecting your privacy, please review our [Privacy Policy](#).  
By clicking submit below, you consent to allow Nuheat.com to store and process the personal information submitted above to provide you the content requested.

**Step 2: Input the Client ID and Secret Key provided by nVent on the Crestron module:**

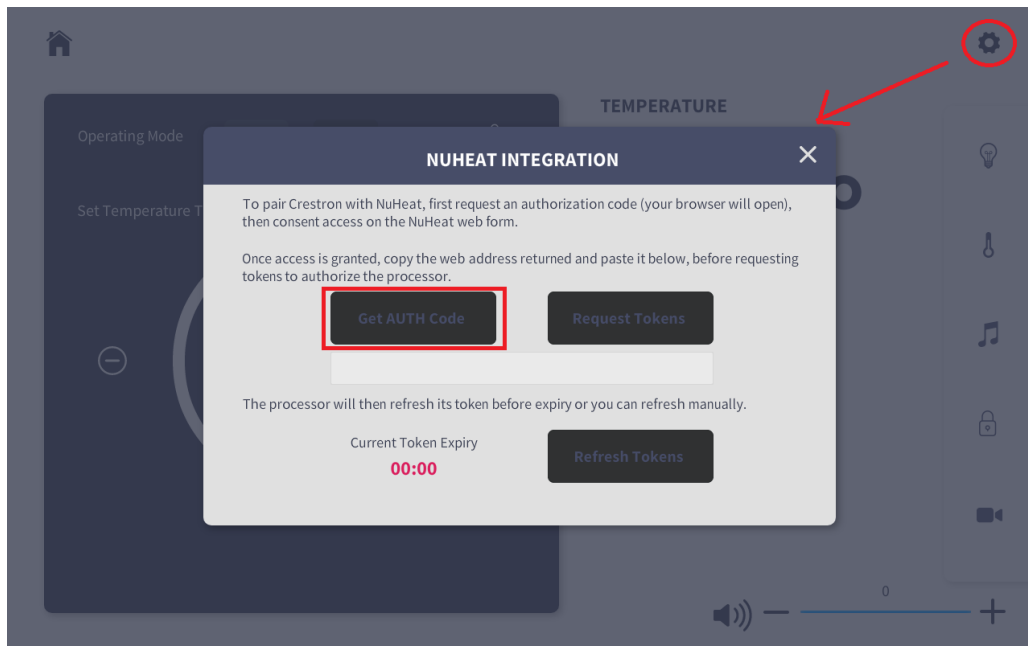
S-1.2 : Nvent NuheatAPI Core (v1.1)

Module Input	Module Output
NUHEAT_INIT	[INIT] [BUSY] NUHEAT_BUSY
NUHEAT_AUTH_CODES	[GET_AUTH_CODE_URLS] NUHEAT_AUTH_URLS
NUHEAT_GET_TOKENS	[AUTH_CODE_URLS]
NUHEAT_REFRESH_TOKENS	[GET_TOKENS] [TOKEN_EXPIRY] NUHEAT_TOKEN_EXPIRY
NUHEAT_STATUS_CODE_CLEAR	[REFRESH_TOKENS] NUHEAT_STATUS_CODE_NOTIF
NUHEAT_UNITS_CELSIUS	[STATUS_CODE_CLEAR] [STATUS_CODE_NOTIF] NUHEAT_STATUS_CODE
NUHEAT_GET_TSTAT_LIST	[STATUS_CODE] NUHEAT_STATUS_MESSAGES
NUHEAT_SUSPEND_POLLING	[SETPOINT_SENT_AS_C] [STATUS_MESSAGES] NUHEAT_STATUS_MESSAGES
	[GET_TSTAT_SUMMARY] [NUMBER_OF_TSTATS] NUHEAT_NUMBER_OF_TSTATS
	[SUSPEND_POLLING] [POLL_TSTAT_INSTANCE] NUHEAT_REFRESH_TSTAT_INSTANCE

CLIENT\_ID as provided by Nuheat  
SECRET\_KEY as provided by Nuheat  
FILE\_PATH \\NVRAM\NUHEAT\Tokens.json  
POLLING\_SPEED 30s



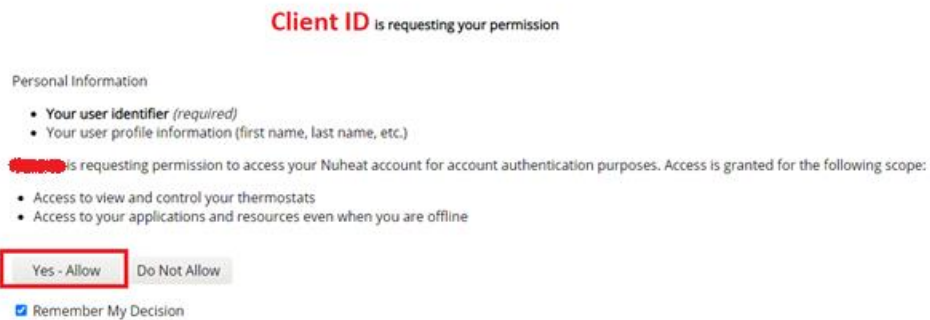
**Step 3: Compile the demo program and request authorization from the demo XPanel/toolbox debugger or a TSW panel (web browser will open):**



**Step 4: Authenticate with mynuheat by accessing the web form, logging in with the customers credentials and grant access:**

- On the demo XPanel/debugger
  - o Copy the authorization URL in a web browser, authenticate (customer credentials) and allow permission

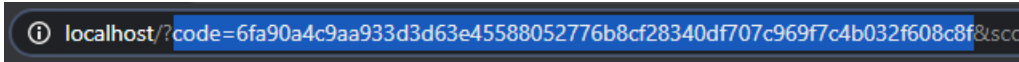
o



- On the TSW, the same process appears in the panel's web browser



**Step 5: Copy the entire returned URL (or at least from “code=” to last character before “&”) from the browser and paste it in toolbox debugger or the demo xpanel, before requesting Tokens:**



1. In Toolbox debugger, copy it into AUTH\_CODE\$ and then request tokens (pulse GET\_TOKENS)

```
[-]  NUHEAT_AUTH_CODE$ = http://localhost/?code=f9d90c8501acb67163a0c4b5beb46  
[-]  NUHEAT_AUTH_URL$ = https://identity.mynuheat.com/connect/authorize?client_id=
```

2. On demo XPanel, copy it into the text field and then request tokens

**NUHEAT INTEGRATION**

To pair Crestron with NuHeat, first request an authorization code (your browser will open), then consent access on the NuHeat web form.

Once access is granted, copy the web address returned and paste it below, before requesting tokens to authorize the processor.

**1**

The processor will then refresh its token before expiry or you can refresh manually.

Current Token Expiry  
**59:59**

**2** **Request Tokens**

**Refresh Tokens**

When successfully authorized, the module will receive the current token expiry time (typically 3600s) and it will start counting down. The module will auto-refresh the token before its expiry, but a manual refresh button is also provided.