Life Is On Schneider Electric

# Hotel Room Controller (HRC)

## Application Programming Interface (API) Description

**Version 1**

## Document History

| Version | Description | Author | Date |
|---|---|---|---|
| 1.0 | Initial version | Hotel Team | 08/03/2016 |
| 1.1 | Revised for HRC App V1.7.7 | Hotel Team | 07/21/2021 |
| 1.2 | Page layout and proofreading | Tech Pubs | 07/29/2021 |
| 1.3 | Replaced “Ecomode” with “Eco mode” | Tech Pubs | 08/06/2021 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

# Contents



## List of Figures

# 1. Overview


Figure 1. Hotel Room Controller

The HRC (Hotel Room Controller) is a controller for hotel guest rooms that allows the integration of multiple devices in the guest rooms including lighting fixtures, switches, dimmers, courtesy panels, air conditioning, curtains, occupancy sensors, door locks, and more. It allows the creation of various scenes and support for energy efficiency programs using key cards switch or occupancy-based algorithms.

This document contains descriptions of the web services interface that will allow you to monitor and control everything in the guest room.

# 2. Cybersecurity

Guest Room Management System (GRMS) security is a relatively new concern for many organizations. Once proprietary, stand-alone systems, guest room management systems are now routinely networked to other systems including Property Management Systems (PMS), Housekeeping applications, and door lock servers to name a few. Many proprietary technologies have been phased out in favor of industry-standard solutions. This has resulted in substantial growth in the use of open protocols.

While these innovations offer real benefits to GRMS consumers, their use requires careful assessment of their impact on security. To mitigate the security risks, organizations should apply best practices for securing their systems.

It is important to consider establishing a boundary around the GRMS and provide ways to control and monitor access. The decisions made during design determine many of the security options available in later lifecycle phases. Therefore, it is essential to solicit input from the people who will be responsible for the installation and operation of the system. Physical security, network infrastructure, and device selection are important elements of the design process.

No security plan is complete unless it addresses the need for physical security. Physical security prevents unauthorized access to the iBMS and GRMS devices, networks, and information. Without it, intruders have the means to circumvent all other methods of protection.

Consider the following when making design decisions:

- Combine multiple barriers to access, such as building, room, and cabinet access control.
- Locate mission-critical devices in access-controlled areas or locked cabinets. Preventing unauthorized physical access to network devices such as routers, firewalls, and switches is a must.
- Protect communication cable runs with conduit or ruggedized cable chases.
- Isolating the system from the outside world until all components of a security plan are in place is best practice.
- Do not control access points to the room (doors etc.) from HRC outputs.

Effective, intelligent GRMS security requires more than a well-thought-out design. It requires continued vigilance throughout all phases of a system's lifecycle. The first step for any organization is the creation of a comprehensive security plan to manage the risks associated with each lifecycle phase. A common understanding of the best practices for securing a GRMS forms the foundation of the planning process. Armed with this information, network administrators, facilities personnel, systems integrators, and other contributors can evaluate alternatives and determine the best approach for their application.

**Please be aware that this API document is confidential and should not be redistributed freely. A formal Non-Disclosure Agreement should be in place with all recipients of this document.**

# 3. Authentication

The HRC supports basic authentication over HTTP. Encryption is done using Base64.

The default values are:
- Username: **administrator**
- Password: **password**

Note that all calls are done through standard GET requests.

Example using JavaScript/JQuery – First you need to create your encryption key using the Base64.encode function:

```
function createAuthKey(user, password) {

var token = user + ':' + password;
var hash = Base64.encode(token);
return "Basic " + hash;
}

//Next you can do your ajax request:

var auth = createAuthKey('myusername', 'mypassword');
var url = 'http://x.x.x.x';

//jquery ajax request

$.ajax({
url : url,
method : 'GET',
beforeSend : function(req) {
req.setRequestHeader('Authorization', auth);
}});
```

Figure 2. Example using JavaScript/JQuery

# 4. How to Get Values

To get values from the guest room, you need to call various files that will give you the information in bulk. Returned data is in XML format.

Here are the various files you can call:

| File Name | Data Provided |
|---|---|
| All.cgx | This interface is the "master" interface which returns absolutely everything in the room. That means the entire content of all the previously described interfaces. |
| AV.cgx | Provides the status of your AV equipment such as the sound volume. |
| Courtesy.cgx | Courtesy contains all the attributes related to the courtesy panel such as DND (Do Not Disturb), MUR (Make Up Room), etc. |
| Curtain.cgx | Allows you to get the status of the curtains and sheers in the room. |
| HVAC.cgx | This interface contains all the values related to the air conditioning.<br>Ex.: setpoint, temperature, humidity, fan speed, etc. |
| Lighting.cgx | Provides the status of all the lights in the room.<br>A light can be on/off and if dimmable will have a dimming level.<br>Lights are under the “Lights” attribute while dimmers are under the “Dimmers” attribute. |
| Room.cgx | This interface provides data on generic things about the room such as the room number, whether it is rented or unrented, what is the favorite language of the guest, etc. |

| File Name | Data Provided |
|---|---|
| Scene.cgx | This interface provides the ID of all the scenes that can be triggered. Note that a scene does not have any status. It can only be launched. |
| WebIF.cgx | Provides in a single place all the equipment that be controlled by the guest such as Lights, Dimmers, Curtains, DND and MUR requests, Scenes and HVAC. |

# 5. Lighting

The lighting interface provides the values of all your lighting fixtures. They are two kinds of lighting: lights and dimmers.

## 5.1 Lights Attribute

Lights are under the "Lights" attribute. The index goes from 1 to 48. Value can be 0-1000.

- For On/Off light, 0: Off; >0 On (1000 as the advised value)
- For dimmer, the value will the light level to be controlled. 0: 0%, 1000: 100%


Figure 3. Lights attribute

# 6. HVAC

This interface contains all the values related to the air conditioning in the guest room. It can be found in the "HVAC" attribute.

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <HVAC>
      <text>
        <index>CoolSetpoint</index>
        <id>t17060</id>
        <value>83.7</value>
      </text>
      <text>
        <index>HeatSetpoint</index>
        <id>t16902</id>
        <value>80.7</value>
      </text>
      <text>
        <index>SetPoint</index>
        <id>t16392</id>
        <value>82.2</value>
      </text>
      <text>
        <index>FanMode</index>
        <id>t17032</id>
        <value>6</value>
      </text>
      <text>
        <index>SystemMode</index>
        <id>t17034</id>
        <value>1</value>
      </text>
      <text>
        <index>SequenceOfOperation</index>
```


Figure 4. HVAC

Many of the values that can be read depend on the thermostat itself. Currently, four thermostats are supported: SE8000, TC500, TC300 and TC900.

The following sections will describe each of these values.

## 6.1 Occupied Setpoint

For SE8000, note that if you set different setpoint values for heating and cooling, the logic will be as followed:

- In cool mode, the cooling setpoint is used.
- In heat mode, the heating setpoint is used.
- In auto mode, the setpoint = (cool setpoint + head setpoint) / 2.

## 6.2 Cool Setpoint and Heat Setpoint (SE8000 only)

The SE8000 provides the flexibility to have a cooling setpoint as well as a heating setpoint. In real life, for hotel applications, the guest should only provide one setpoint and the thermostat will be smart enough to do an auto-changeover to cool or heat appropriately. In other words, in most cases, these values should always be the same.

The setpoints are decimal values under the IDs "t17060" and "t16902".

## 6.3 Fan Mode

The fan mode value will depend on the settings defined in the fan menu sequence property.

The possible values are:
- 1: Low
- 2: Medium
- 3: High
- 4: Auto
- 5: On (supported by SE8000 only)
- 6: Off

Again, some of those values might never appear depending on the fan menu sequence.

## 6.4 System Mode

The System Mode allows the guest to decide whether to heat, cool, turn on or off the air conditioning. Note that this value is dependent on the "Sequence Of Operation" value which defines what is allowed to be used by the guest. This can vary depending on the brand standard of the hotel or the capability of the Fan Coil Unit (FCU).

The possible values are:
- 1: Off
- 2: Auto
- 3: Cool
- 4: Heat

Note that more and more, many hotel brands will dictate that only Off and Auto are available. Cooling and heating should be set to an "auto changeover" mode to automatically cool or heat depending on the environmental values.

## 6.5 Sequence of Operation

This provides you with the possible system mode values provided to the guest.
- • 1: Cooling Only
- • 2: Heating Only
- • 3: Cooling with Electric Reheat
- • 4: Heating with Electric Reheat
- • 5: Electric Reheat only

## 6.6 Fan Menu

This provides the Fan Menu Sequence. You can use this value to decide on what to show to the guest on your interface depending on what was made available at the thermostat and what is supported by the air conditioning.

The possible values are:
- 1: Low-Med-High
- 2: Low-High
- 3: Low-Med-High-Auto
- 4: Low-High-Auto
- 5: On-Auto
- Temperature Data Unit

This represents the data unit of measure (Celsius or Fahrenheit) used for data points as setpoints/temperature/dead band etc.

## 6.7 Room Temperature & Room Humidity

Those values provide the current temperature and humidity in the room. Note that humidity is only provided when a humidity sensor has been installed (typically using an SE8350).

- Temperature (t16906) is a decimal value.
- Humidity (t16908) is an integer.

## 6.8 Temperature Scale

This is the display unit for the end-user HMI (Guest), i.e. SE8000 Thermostat Touch Panel:

- 1: Celsius
- 2: Fahrenheit

You use this value to display °F or °C in your interface.

## 6.9 Min Cool and Max Heat

These values are the minimum and maximum values allowed for the setpoint. The guest cannot go below or over these values during the on-screen selection of the setpoint.

## 6.10 Dead Band

The dead band is the zone around the setpoint where the temperature can vary. The temperature should never be outside the setpoint ± dead band.

The dead band can be a value between 2°F (1°C) and 5°F (2.5°C).

This is normally dictated by the hotel brand standards and should not be changed.

## 6.11 Eco Mode

Supported by TC900 only. The TC900 provides energy-saving functions that can be enabled or disabled through the Eco mode function.

Possible values are:

- 0: Disable
- 1: Enable

# 7. Courtesy Panel

The courtesy panel interface provides the values and identifiers for the following attributes:

| Description | Attribute | Identified | Possible Values |
|---|---|---|---|
| DND (Do Not Disturb) | DND | t17012 | 0: Off; 1: On |
| MUR (Make Up Room) | MUR | t17014 | 0: Off; 1: On |
| Please Wait | Not defined | Not defined | Not defined |
| Service Call | SERVICE | t09151 | 0: Off; 1: On |

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header/>
  <soap:Body>
    <Courtesy>
      <text>
        <index>DND</index>
        <id>t17012</id>
        <value>0</value>
      </text>
      <text>
        <index>MUR</index>
        <id>t17014</id>
        <value>0</value>
      </text>
      <text>
        <index>Service</index>
        <id>t09151</id>
        <value>0</value>
      </text>
    </Courtesy>
  </soap:Body>
</soap:Envelope>
```

Figure 5. Courtesy panel

# 8. Curtains

The curtain interface currently only supports the Open and Close values for reading.

## 8.1 Curtain Status

You can have up to 8 curtains/sheers controlled from which you can read the status.

Possible status values are:
- 0: Closed
- 1: Opened

| Index | Identifier | Value |
|---|---|---|
| 1 | t17018 | 0: Closed; 1: Opened |
| 2 | t17020 | 0: Closed; 1: Opened |
| 3 | t17022 | 0: Closed; 1: Opened |
| 4 | t16429 | 0: Closed; 1: Opened |
| 5 | t16939 | 0: Closed; 1: Opened |
| 6 | t16958 | 0: Closed; 1: Opened |
| 7 | t17024 | 0: Closed; 1: Opened |
| 8 | t17026 | 0: Closed; 1: Opened |

## 8.2 Curtain Control

However, to control the curtain, you have the following possible curtain command states:
- 1: Standby
- 2: Open
- 3: Close
- 4: Stop

Note that the ID of set and get functions are different!

The Get request will allow you to see the various indexes available to you (It is the "control" id).

The Identifier that can be triggered along with the associated values are as follows:

| Index | Identifier | Value |
|---|---|---|
| 1 | t16427 | 1: standby; 2: open; 3: close; 4: stop |
| 2 | t17044 | 1: standby; 2: open; 3: close; 4: stop |
| 3 | t17046 | 1: standby; 2: open; 3: close; 4: stop |
| 4 | t17048 | 1: standby; 2: open; 3: close; 4: stop |
| 5 | t17050 | 1: standby; 2: open; 3: close; 4: stop |
| 6 | t17052 | 1: standby; 2: open; 3: close; 4: stop |
| 7 | t17054 | 1: standby; 2: open; 3: close; 4: stop |
| 8 | t17056 | 1: standby; 2: open; 3: close; 4: stop |


Figure 6. Curtain control

# 9. Room Status

The room status provides several useful information on various settings or items in the room such as:

- Occupancy:
  - 1: Rented/Occupied
  - 3: Unrented
  - 4: Rented/Unoccupied
- Rented/Unrented:
  - 1: Rented
  - 3: Unrented
- Room Number
- Door Status
- Door Trigger
- Windows Status
- Language

# 10. Languages

This also follows the language settings of the SE8000. The value is defined by the Property Management System (PMS) during the check-in process.

The possible values are:

| | |
|---|---|
| 1 | English |
| 2 | French |
| 3 | Spanish |
| 4 | Chinese |
| 5 | Russian |
| 6 | Arabic |
| 7 | Bulgarian |
| 8 | Czech |
| 9 | Danish |
| 10 | Dutch |
| 11 | Finnish |
| 12 | German |
| 13 | Hungarian |
| 14 | Indonesian |
| 15 | Italian |
| 16 | Norwegian |
| 17 | Polish |
| 18 | Portuguese |
| 19 | Slovak |
| 20 | Swedish |
| 21 | Turkish |

Figure 7. Languages

The default value is English.

# 11. Scenes

Scenes can only be triggered. When querying this interface, you will only get the identifiers of all the scenes available. Values will always be 0.

To trigger a scene, please refer to section “4 How to Get Values” on page 2.

# 12. Audio/Video

Currently not available in this version of the interface.

# 13. How to set values

Setting values is done by a simple HTTP GET request with standard parameters. The file to call is evo.xml.

Standard request will look as followed: http://x.x.x.x/evo.xml?\[ID\]=\[Value]
- x.x.x.x is the IP of the HRC.
- ID is the unique ID of the parameter that you are trying to change
- Value is the new value that you want to assign to the ID

Here are a few examples:
Turn "On" the DND and "Off" the MUR: http://x.x.x.x/evo.xml?t17012=1&t17014=0

Launch a scene:
The value is the index of the scene you want to trigger (starting at 2; Index 1 is the default value of the scene attribute).

For example, Master On is 2, Master Off is 3, Welcome Day is 4, etc.
To trigger the Welcome Day scene, the request would be: http://x.x.x.x/evo.xml?t17058=4

Once the scene has been launched the value of the scene goes back to 1. (if you were to do a "read" of the attribute, the value should always be 1.

# 14. Other Information Worth Reading

- During testing of the interface using your web browser, do make sure that you are not hitting the cache of your browser. If the values do not change in your XML, most likely, this will be the issue. Remember to clear the browser cache after any upgrade of application firmware in your HRC.
- Make sure that you do not go over the max length of your URL when trying to do too many set values.

# 15. Modbus TCP

The HRC also supports communication through Modbus TCP. Cybersecurity is a strong consideration when using either BACnet or Modbus open protocols. As such the guest room door or any lockable access point should never be directly controlled via an HRC output. For cybersecurity purposes, ensure that your firewall and managed switches are properly configured.

For Modbus, the standard TCP port (Port 502) is used.

| Modbus Communication | |
|---|---|
| Port | 502 |
| Modbus ID | 255 |

The register list is following the same interfaces as the Web Services defined in the previous section.

| Name | Registered Number | Read | Write | Type | Value Description |
|---|---|---|---|---|---|
| Light 1 | 16980 | 3 | 6 | float | 0 - 1000 |
| Light 2 | 16982 | 3 | 6 | float | 0 - 1000 |
| Light 3 | 16984 | 3 | 6 | float | 0 - 1000 |
| Light 4 | 16986 | 3 | 6 | float | 0 - 1000 |
| Light 5 | 16988 | 3 | 6 | float | 0 - 1000 |
| Light 6 | 16990 | 3 | 6 | float | 0 - 1000 |
| Light 7 | 16992 | 3 | 6 | float | 0 - 1000 |
| Light 8 | 16994 | 3 | 6 | float | 0 - 1000 |
| Light 9 | 16996 | 3 | 6 | float | 0 - 1000 |

| Name | Registered Number | Read | Write | Type | Value Description |
|---|---|---|---|---|---|
| Light 10 | 16998 | 3 | 6 | float | 0 - 1000 |
| Light 11 | 17000 | 3 | 6 | float | 0 - 1000 |
| Light 12 | 17002 | 3 | 6 | float | 0 - 1000 |
| Light 13 | 17004 | 3 | 6 | float | 0 - 1000 |
| Light 14 | 16469 | 3 | 6 | float | 0 - 1000 |
| Light 15 | 17008 | 3 | 6 | float | 0 - 1000 |
| Light 16 | 17010 | 3 | 6 | float | 0 - 1000 |
| Light 17 | 16904 | 3 | 6 | float | 0 - 1000 |
| Light 18 | 16960 | 3 | 6 | float | 0 - 1000 |
| Light 19 | 16402 | 3 | 6 | float | 0 - 1000 |
| Light 20 | 16964 | 3 | 6 | float | 0 - 1000 |
| Light 21 | 16966 | 3 | 6 | float | 0 - 1000 |
| Light 22 | 16968 | 3 | 6 | float | 0 - 1000 |
| Light 23 | 16972 | 3 | 6 | float | 0 - 1000 |
| Light 24 | 16412 | 3 | 6 | float | 0 - 1000 |
| Light 25 | 16560 | 3 | 6 | float | 0 - 1000 |
| Light 26 | 16494 | 3 | 6 | float | 0 - 1000 |
| Light 27 | 16496 | 3 | 6 | float | 0 - 1000 |
| Light 28 | 16498 | 3 | 6 | float | 0 - 1000 |
| Light 29 | 16500 | 3 | 6 | float | 0 - 1000 |
| Light 30 | 16502 | 3 | 6 | float | 0 - 1000 |
| Light 31 | 16504 | 3 | 6 | float | 0 - 1000 |
| Light 32 | 16506 | 3 | 6 | float | 0 - 1000 |
| Light 33 | 16508 | 3 | 6 | float | 0 - 1000 |
| Light 34 | 16510 | 3 | 6 | float | 0 - 1000 |
| Light 35 | 16512 | 3 | 6 | float | 0 - 1000 |
| Light 36 | 16514 | 3 | 6 | float | 0 - 1000 |
| Light 37 | 16516 | 3 | 6 | float | 0 - 1000 |
| Light 38 | 16518 | 3 | 6 | float | 0 - 1000 |
| Light 39 | 16520 | 3 | 6 | float | 0 - 1000 |
| Light 40 | 16522 | 3 | 6 | float | 0 - 1000 |
| Light 41 | 16524 | 3 | 6 | float | 0 - 1000 |
| Light 42 | 16526 | 3 | 6 | float | 0 - 1000 |
| Light 43 | 16528 | 3 | 6 | float | 0 - 1000 |
| Light 44 | 16530 | 3 | 6 | float | 0 - 1000 |
| Light 45 | 16532 | 3 | 6 | float | 0 - 1000 |
| Light 46 | 16534 | 3 | 6 | float | 0 - 1000 |
| Light 47 | 16536 | 3 | 6 | float | 0 - 1000 |
| Light 48 | 16538 | 3 | 6 | float | 0 - 1000 |
| Curtain Status 1 | 17018 | 3 | 6 | holding register | 0: closed; 1: opened |
| Curtain Status 2 | 17020 | 3 | 6 | holding register | 0: closed; 1: opened |
| Curtain Status 3 | 17022 | 3 | 6 | holding register | 0: closed; 1: opened |
| Curtain Status 4 | 16429 | 3 | 6 | holding register | 0: closed; 1: opened |

| Name | Registered Number | Read | Write | Type | Value Description |
|---|---|---|---|---|---|
| Curtain Status 5 | 16939 | 3 | 6 | holding register | 0: closed; 1: opened |
| Curtain Status 6 | 16958 | 3 | 6 | holding register | 0: closed; 1: opened |
| Curtain Status 7 | 17024 | 3 | 6 | holding register | 0: closed; 1: opened |
| Curtain Status 8 | 17026 | 3 | 6 | holding register | 0: closed; 1: opened |
| Curtain Control 1 | 16427 | 3 | 6 | holding register | 1: Standby; 2: Fully Open; 3: Fully Close; 4: Stop; |
| Curtain Control 2 | 17044 | 3 | 6 | holding register | 1: Standby; 2: Fully Open; 3: Fully Close; 4: Stop; |
| Curtain Control 3 | 17046 | 3 | 6 | holding register | 1: Standby; 2: Fully Open; 3: Fully Close; 4: Stop; |
| Curtain Control 4 | 17048 | 3 | 6 | holding register | 1: Standby; 2: Fully Open; 3: Fully Close; 4: Stop; |
| Curtain Control 5 | 17050 | 3 | 6 | holding register | 1: Standby; 2: Fully Open; 3: Fully Close; 4: Stop; |
| Curtain Control 6 | 17052 | 3 | 6 | holding register | 1: Standby; 2: Fully Open; 3: Fully Close;4: Stop; |
| Curtain Control 7 | 17054 | 3 | 6 | holding register | 1: Standby; 2: Fully Open; 3: Fully Close;4: Stop; |
| Curtain Control 8 | 17056 | 3 | 6 | holding register | 1: Standby; 2: Fully Open; 3: Fully Close;4: Stop; |
| DND | 17012 | 3 | 6 | holding register | 0: on; 1: off (Need to interlock with MUR) |
| MUR | 17014 | 3 | 6 | holding register | 0: on; 1: off (Need to interlock with DND) |
| Service | 17151 | 3 | 6 | holding register | 0: on; 1: off (Need to interlock with DND) |
| Scene | 17058 | 3 | 6 | float | **1: None**<br>**2: Master On**<br>**3: Master Off**<br>**4: Welcome Day**<br>**5: Welcome Night**<br>**6: Scene 5**<br>**...**<br>17: Scene 16 |
| Sequence of Operation | 17038 | 3 | 6 | holding register | **1: Cool Only**<br>**2: Heat Only**<br>**3: Cool-Rlt**<br>**4: Heat-Rlt**<br>**5: Heat/Cool**<br>**6: Cl/ht-rht**<br>**(Shared by all thermostats)** |
| Fan Menu | 17036 | 3 | 6 | holding register | **1: L-M-H; 2: L-H; 3: L-M-H-A;**<br>**4: On-Auto**<br>**(Shared by all thermostats)** |
| Min Cool | 9024 | 3 | 6 | holding register | **0-100**<br>**(Shared by all thermostats)** |
| Max Heat | 9025 | 3 | 6 | holding register | **0-100**<br>**(Shared by all thermostats)** |
| Dead Band | 16432 | 3 | 6 | holding register | **2-5**<br>**(Shared by all thermostats)** |
| Room Temperature | 16906 | 3 | | float | **0-100 Thermostat 1** |
| Room Humidity | 16908 | 3 | | holding register | **0-100 (unit: %) Thermostat 1** |

| Name | Registered Number | Read | Write | Type | Value Description |
|---|---|---|---|---|---|
| Fan Mode | 17032 | 3 | 6 | holding register | **1: Low; 2: Mid; 3: High; 4: Auto Thermostat 1** |
| System Mode | 17034 | 3 | 6 | holding register | **1: Off; 2: Cool; 3: Heat; 4: Auto Thermostat 1** |
| Occupied Setpoint | 16392 | 3 | 6 | float | **0-100 Thermostat 1** |
| Eco mode | 9444 | 3 | 6 | holding register | **0: disabled; 1: enabled Thermostat 1** |
| Room Temperature | 16480 | 3 | 6 | float | **0-100 Thermostat 2** |
| Room Humidity | 16482 | 3 | 6 | holding register | **0-100 (unit: %) Thermostat 2** |
| Fan Mode | 16486 | 3 | 6 | holding register | **1: Low; 2: Mid; 3: High; 4: Auto Thermostat 2** |
| System Mode | 16488 | 3 | 6 | holding register | **1: Off; 2: Cool; 3: Heat; 4: Auto Thermostat 2** |
| Occupied Setpoint | 16484 | 3 | 6 | float | **0-100 Thermostat 2** |
| Eco mode | 9445 | 3 | 6 | holding register | **0: disabled; 1: enabled Thermostat 2** |
| Room Temperature | 16490 | 3 | 6 | float | **0-100 Thermostat 3** |
| Room Humidity | 16492 | 3 | 6 | holding register | **0-100 (unit: %) Thermostat 3** |
| Fan Mode | 16542 | 3 | 6 | holding register | **1: Low; 2: Mid; 3: High; 4: Auto Thermostat 3** |
| System Mode | 16544 | 3 | 6 | holding register | **1: Off;2: Cool;3: Heat;4: Auto Thermostat 3** |
| Occupied Setpoint | 16540 | 3 | 6 | float | **0-100 Thermostat 3** |
| Eco mode | 9446 | 3 | 6 | holding register | **0: disabled; 1: enabled Thermostat 3** |
| Room Temperature | 16548 | 3 | 6 | float | **0-100 Thermostat 4** |
| Room Humidity | 16550 | 3 | 6 | holding register | **0-100 (unit: %) Thermostat 4** |
| Fan Mode | 16554 | 3 | 6 | holding register | **1: Low; 2: Mid; 3: High; 4: Auto Thermostat 4** |
| System Mode | 16556 | 3 | 6 | holding register | **1: Off; 2: Cool; 3: Heat; 4: Auto Thermostat 4** |
| Occupied Setpoint | 16552 | 3 | 6 | float | **0-100 Thermostat 4** |
| Eco mode | 9447 | 3 | 6 | holding register | **0: disabled; 1: enabled Thermostat 3** |
| Temperature Data Unit | 16583 | 3 | 6 | holding register | **1: °C; 2: °F** |