Climate Control API Interface Specification

Introduction

Founten provides an open standards based real-time interface to all of the management features of their Wireless CommunicatingThermostats. The interface uses simple Http GET and POST commands to request thermostat settings or change active parameters. All responses are provided using standard XML notation. Transactions are only allowed over Secure Socket Layer (SSL) connections using password protection. These standard protocols are easily implemented by developers on all types of computing platforms using modern scripting and programming languages.

Provided Software

Because the API uses standard protocols, there is no restriction on what type of systems or software may access the Founten system. The client software simply establishes an SSL connection to their private web server address and initiates an http request. To further simplify access to the API, Founten has developed a Perl module which provides an easy to use wrapper which gives Perl developers an even more simplified interface to the API. This Perl module can also be used by developers as a reference on how to construct valid requests and how to process API responses.

Constructing an API Request

Http requests are constructed with a list of named values. The Founten API requires 6 elements in each request. These values are:

username – This is the Email ID of a valid Founten Site Manager user. This Email ID is created using the administrative interface of the Founten Web App.

password – This is the password for the Email ID specified above. Since the Founten system only allows SSL connections, the password is automatically encrypted for each request.

request – The value of this parameter should be either “set” or “get”. This identifies the type of request being made. “get” requests are intended to provide the current values of attributes being requested. “set” requests are intended to change the values of attributes.

object – This is the type of object to which the “get” or “set” is being applied. The currently supported object types are: “Thermostat”, or “ThermostatSchedule”. See the section titled “Object Attributes” for a list of attributes support by each object type.

selection – This is a set of attribute/value pairs which should be used as a query match for the “set” or “get” request. In a “get” request, items matching the selection will be returned in the XML reply. In a “set” request, items matching the selection will be modified in the Pelican system. Pairs are separated by semicolons (;) and attributes are separated from their values by colons (:) (See example below).

value – For “get” requests, this is a semicolon (;) separated list of attributes which are being requests. For a “set” request, this is a semicolon (;) separated list of attribute/value pairs to be modified. The attribute names are separated from their values by a colon (:). Semicolons and colons are invalid characters for either the attribute or the value.

When using HTTP GET, the 6 required elements must be formatted using standard http notation with the element name as shown above followed by an equal sign and then that elements value. The first element is preceded by a question mark (?) and the elements are separated by the ampersand (&) character. Standard http character escaping is supported. When using HTTP POST, standard encoding is supported.

The web address of the API interface is the full web address of the site followed by "/api.cgi". Therefore, a valid API request would be as follows:

https://demo.officeclimatecontrol.net/api.cgi?username=myname@gmail.com&password=mypassword&request=get&object=Thermostat&selection=name:TestThermostat;&value=heatSetting;coolSetting;temperature;

The API Result

The API will return a properly formatted XML version 1.0 reply. The first XML element will be named "response" and will contain a status element named "success" with a numeric result code. If an error is encountered, an element named "message" with a human readable text message will be returned with an explanation of the error. In addition, if the request type is "get" a set of elements will be returned for each item which matched the selection criteria.

Examples

The following XML reply would be returned from the HTTP request shown above:

```
<?xml version="1.0" encoding="UTF-8"?>
<result>
  <Thermostat>
    <name>TestThermostat</name>
    <heatSetting>68</heatSetting>
    <coolSetting>72</coolSetting>
    <temperature>70</temperature>
  </Thermostat>
  <success>1</success>
</result>
```

If "TestThermostat" wasn't found by the system, the following reply would be returned:

```
<?xml version="1.0" encoding="UTF-8"?>
<result>
  <success>0</success>
  <message>No thermostats found matching selection criteria.</message>
</result>
```

User Defined Attributes

There are two types of attributes supported for "Thermostat" objects. The first types are reserved and predefined. These attributes are defined below in the section titled "Object Attributes". In addition to the predefined attributes users can define, set and retrieve their own attributes.

Any attribute name which is not reserved will be treated like a User Defined Attribute. Defining a new attribute is accomplished by setting its value through a set request. Once set, these attributes can be accessed using get requests. There is no restriction on the number of user defined attributes or their values. User defined attributes can also be used as part of the selection criteria. For example, if the user wants to manage a group of thermostats through the API, they could define an attribute named “managed” and set its value to “yes” for the thermostats in the user’s group. Then they could include “managed:yes;” in their selection criteria to only affect their special group of thermostats.

Object Attributes

The currently supported object types are: “Thermostat”, or “ThermostatSchedule”. Attribute names are not case sensitive. Attribute values are case sensitive.

Thermostat Object

Name	Values	Set	Description
name	String	N	The configured name of the thermostat.
groupName	String	Y	The configured group name of the thermostat.
serialNo	String	N	The thermostat serial number.
system	Off, Auto, Heat, Cool	Y	The active system mode.
heatSetting	Integer	Y	The active Heat Setting.
coolSetting	Integer	Y	The active Cool Setting.
fan	Auto, On	Y	The active Fan Mode.
status	Occupied, Vacant	Y	Normally Occupied. Vacant when vacation schedule is active.
temperature	Decimal	N	The current measured temperature.
setBy	Station, Remote, Schedule	N	Indicates how the active thermostat settings were set.
frontKeypad	On, Off	Y	Thermostat keypad status.
cycleRate	Integer	Y	Configured target cycle rate.
anticipationDegrees	Decimal	Y	Configured anticipation degrees.
calibrationOffet	Decimal	Y	Configured calibration degrees.
heatStages	Integer	Y	Number of heat stages.
coolStages	Integer	Y	Number of cool stages.
fanStages	Integer	Y	Number of fan stages.
temperatureFormat	Fahrenheit, Celsius	Y	Active temperature format.
systemType	Conventional, Heat Pump	Y	Active system type.
minHeatSetting	Integer	Y	Lowest user settable heat setting.
maxHeatSetting	Integer	Y	Highest user settable heat setting.
minCoolSetting	Integer	Y	Lowest user settable cool setting.

maxCoolSetting	Integer	Y	Highest user settable cool setting.
fanCirculationTime	Integer	Y	Minutes/hour to circulate the fan.
heatingType	String	Y	Size of heating system.
coolingType	String	Y	Size of cooling system.
allowKeypadDisable	Yes, No	Y	Display keypad On/Off button.
weight	Integer	Y	Weighted value when used with Remote Sensors
schedule	On, Off	Y	Whether the schedule is active.
scheduleRepeat	Weekday, Weekly	Y	Schedule repeat.
scheduleSetTimes	2, 3, 4, Variable	Y	Number of set times per day in the schedule.
scheduleMultipleSystem	Yes, No	Y	System mode for each schedule set time.
scheduleMultipleFan	Yes, No	Y	Fan mode for each schedule set time.
scheduleSystem	Auto, Heat, Cool, Off	Y	System mode for schedule when scheduleMultipleSystem is No.
scheduleFan	Auto, On	Y	Fan mode for schedule when scheduleMultipleFan is Off.
scheduleFrontKeypad	On, Off	Y	Front keypad setting when scheduleMultipleSystem is Off.

ThermostatSchedule Object

Name	Values	Set	Description
name	String	N	The configured name of the thermostat.
dayOfWeek	Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Vacation	Y	The day of the week for this schedule set time.
setTime	Integer	Y	The indexed set time for this schedule entry.
startTime	String	Y	The time of day (24 hr format) for this schedule entry.
system	Auto, Heat, Cool, Off	Y	The system mode.
heatSetting	Integer	Y	The heat setting.
coolSetting	Integer	Y	The cool setting.
fan	Auto, On	Y	The fan setting.
keypad	On, Off	Y	The keypad setting.
delete	None	Y	Delete the specified schedule entry.
deleteAll	None	Y	Delete the schedule for the entire day.