Hubitat Installation and Setup

Crestron Home Extension Drivers for the Hubitat automation hub provide a way to integrate a wide variety of Zigbee, Z-Wave, and Wi-Fi devices to the Crestron Home system. While the Hubitat hub is a full-featured home automation system primarily intended for do-it-yourself home owners, only a few basic features of the Hubitat system need to be learned and utilized to permit it to operate as a Zigbee/Z-Wave/Wi-Fi gateway for a Crestron Home system. These features are setup of the hub, adding devices, and selecting devices for Application Program Interfaces (API's).

The Hubitat hub operates locally and since it does not depend on an internet connection, it is immune from internet outages, as well as being very responsive. The Hubitat is compatible with nearly 500 Zigbee, Z-Wave, and Wi-Fi devices. The newest Hubitat hub version, the Elevation C-8, already has built-in hardware support for Matter/Thread, which will become operational with a future firmware release.

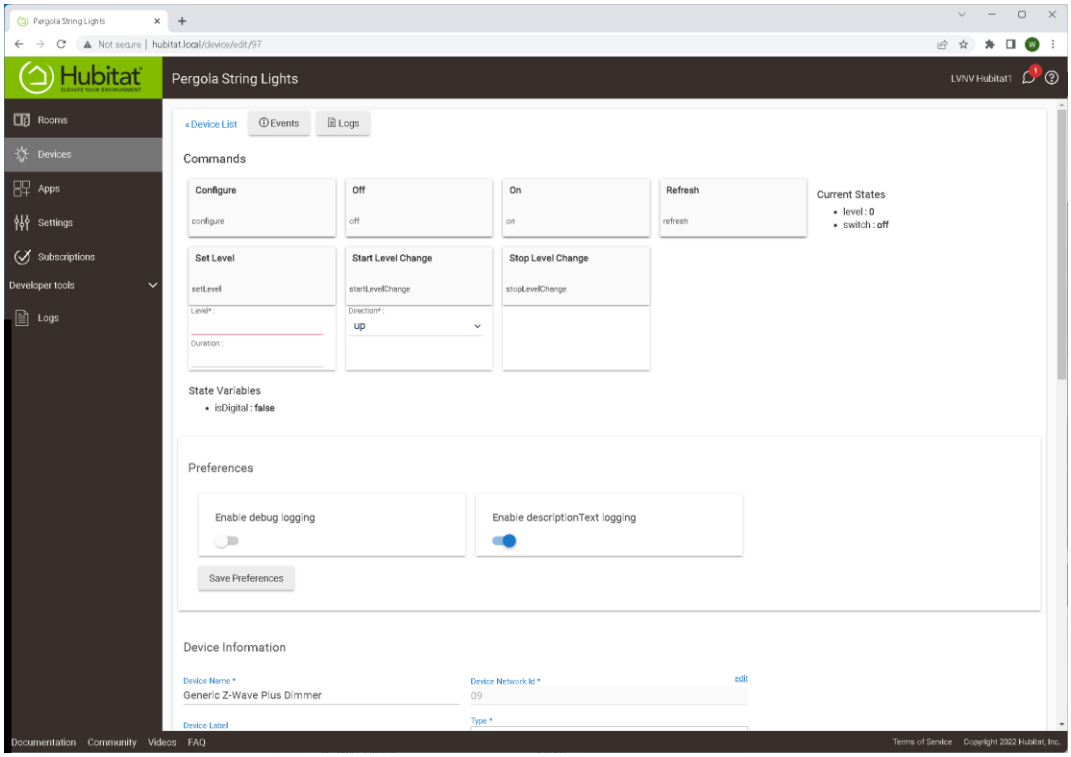
Hubitat Set Up

Three videos in the Tutorials section of the habitat.com website titled "How to Set Up and Register Your Hub" (3:40) , "How to Set Up Hubitat Elevation Model C-8 Hub: (5:14) and "How to Add Devices" (4:43) provide a helpful overview of these procedures. Be sure to not the IP address of the Hubitat. It will be needed for installation of the Extension Driver.

Adding Devices

An article in Residential Tech Today magazine written by Jay Basen gives an excellent overview of the Hubitat system and how it can be used to integrate a large variety of Z-Wave, Zigbee and IoT devices with Crestron systems. The article can be viewed at: https://restechtoday.com/integrating-a-wide-range-of-zigbee-z-wave-and-iot-devices-with-crestron/ .

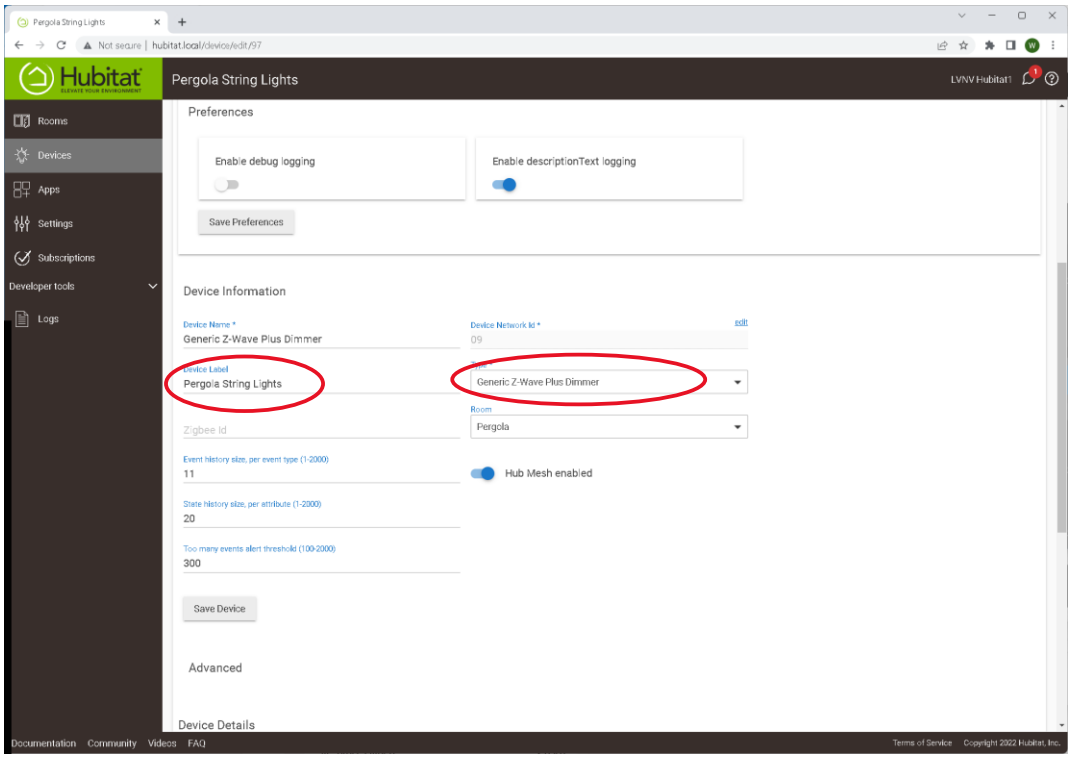
Once devices have been acquired, the Hubitat App Devices page can be used to directly control the device. In the Commands section of the Device control page shown below, buttons are provided to turn the device on, off and set it's dimming level. This can be accomplished only a few minutes after installing the Hubitat hub and acquiring the first device.



Preference settings shown should be reviewed and adjusted if desired. Enable description Text logging should be set to On to see device history in the Hubitat Logs. Be sure to click the “Save Preferences” button after making changes to the settings in the Preferences section. These settings can be changed at any time, even after the Extension Driver has been installed and is operating.

Further down on this page, in the Device Information section, the Device Label field contains a name that will be used as the label for the device controls and status in the End-User App. On the same page also verify that the Type field selection contains a description that is appropriate for the type of device, e.g. the word “Dimmer” or “Switch” for lighting devices. The user’s guide for each device type specific driver (Lights, Relays, Garage control, Sensors, Notifications, etc.) will specify the key words that should appear in the Type field.

It is important that the Hub Mesh enabled parameter be set to On for devices that are plugged in and not battery powered. Be sure to click the “Save Device” button after making changes to the fields in the Device Information section.



Adding Wi-Fi Devices

To add a Wi-Fi device, go to the Devices page of the Hubitat App and click the “Add Device” button. Typically, only Wi-Fi devices having a Protocol listed as LAN from the Hubitat List of Compatible Devices https://docs2.hubitat.com/en/devices/list-of-compatible-devices should be used to ensure that a supported Hubitat device driver will be used.

Start the installation by clicking on the “Find by brand” button, locate the device type under its brand name, and click on the device type button. Then locate the Device name in the list of devices and click on the “Add” button. After that, follow the instructions to install the device. Use the device’s app to add the device to the Wi-Fi network. After adding the device to the network, go to the Devices page for the new device, enter its IP address, and verify its operation by clicking on the various command buttons.

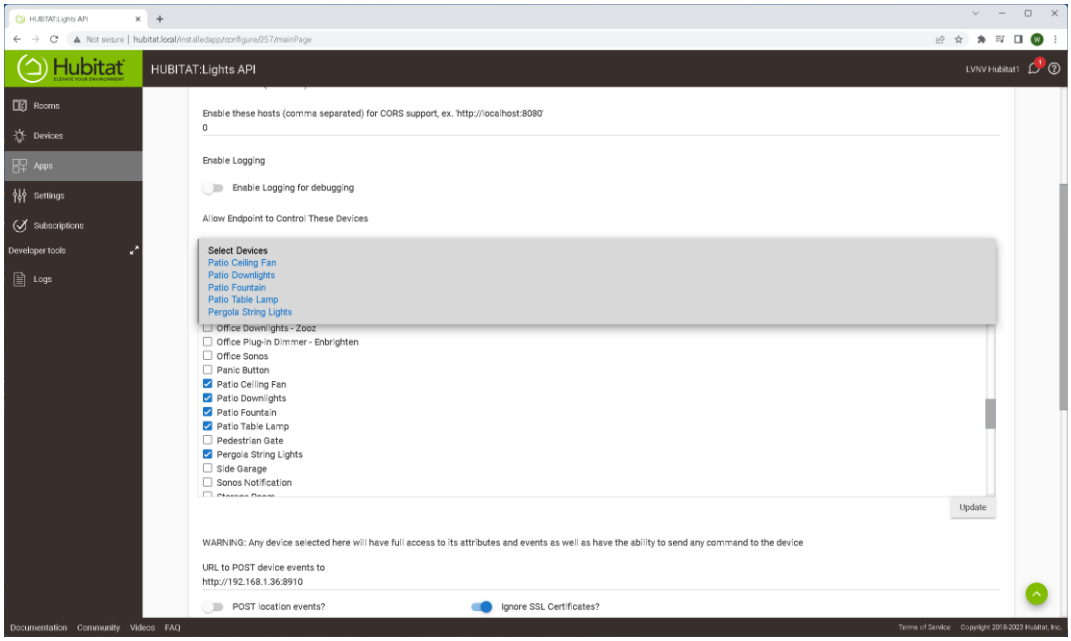
Some devices are parent devices that do not have device controls (e.g., On or Off) on the Device page. In those cases, go back to the Devices list, locate the parent device, note the list of child devices under it, and select a child device for control checkout on the Devices page.

Selecting Devices for API's

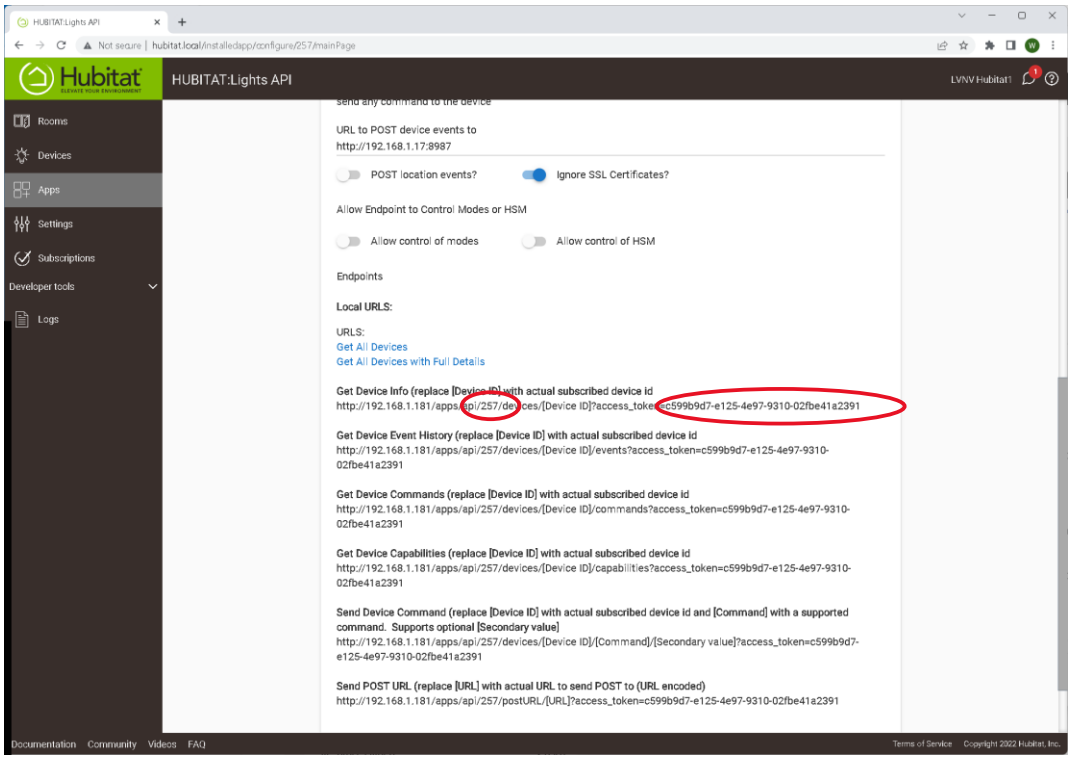
The primary method by which Crestron Home drivers communicate with a Hubitat hub and its attached devices is via the Hubitat Maker API (Application Program Interface) app. The Maker API app allows the user to "make" an interface for Crestron Home by simply checking boxes next to a list of Z-Wave, Zigbee and Wi-Fi devices.

The Maker API app allows the aggregation of multiple devices into a single uniform and consistent interface. While it is the responsibility of the installer to create and modify Maker API's, the process is neither difficult nor time consuming. A description of how to create a Maker API is available on the Hubitat website: http://docs2.hubitat.com/en/apps/maker-api.

Although Hubitat supports a large number of Built-In Apps, only the Maker API works with Extension Drivers for Hubitat and should be the only type of Built-In App utilized for Extension Drivers. The Maker API page shown below lists the devices that have been added to the Maker API, and also permits the addition and deletion of devices.



Further down on the same page, in the Local URLS section, the Maker API App ID and Access Token values are found. These values must be entered in the Installer Settings page of the Crestron Home Setup App when a new instance of the driver is installed. The small red oval above highlights where the Maker API App ID can be found and the elongated red oval highlights where the Access Token can be found.



Each instance of the Driver used requires its own uniquely named Maker API, with its own set of devices, Maker API ID, and Access Token.