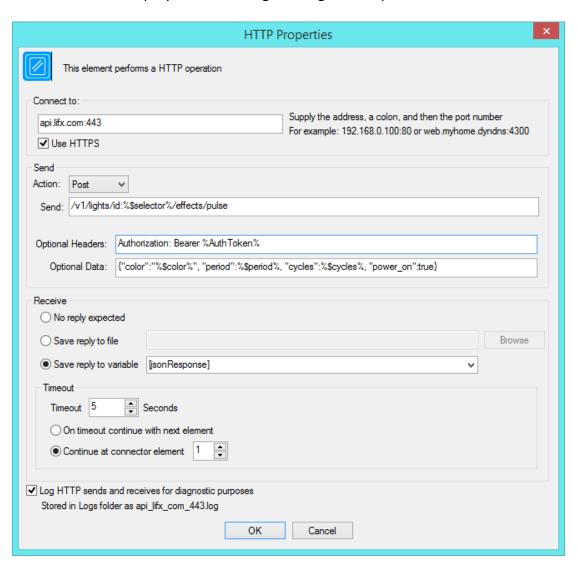## The HTTP Element

The HTTP element gives you access to any net-based service or connection that "talks" http. The element has several features that need some explanation.

Note: You need to know what you are doing as the HTTP element just exposes a way to perform HTTP operations and doesn't hide any details.

Here is the element properties and I'll go through all the parts.

Connect To

The "Connect to" is the web location suffixed by the port to use. In this case it is connecting to port 443 (which is the standard HTTPS port number). In the checkbox below determines if the operation should use standard HTTP or secure HTTPS. Regardless of that checkbox, you still must put the correct port number in the connect. The connect string could contain embedded HCA expressions as can the Send, Headers, and Data as described below.

Send

The items in the Send group are where the action happens as they are the parts of the HTTP operation. These are:

- The "Action" dropdown selects the HTTP operation to perform, either a Put, Post, or Get operation. Whatever you are communicating with should explain what kind of action to use. Typically, a Get is to acquire some data and PUT and POST are for sending data.

- The string in the "Send" edit is the main part of the HTTP operation. It is where the message is going. Generally, this is some sort of a page reference.

- "Optional Headers" are the headers for the HTTP operation.

- "Optional Data" is the data sent typically as part of a POST or PUT.

There are several key points about all of this:
- The format and contents of the data in all three strings depends upon what you are communicating with. You will need to know that from documentation from the web site or service.

- Each of these strings can have embedded HCA expression in them and those are evaluated, and the final string constructed before the HTTP operation is carried out. The example element shown above uses several embedded expressions here. An embedded expression is surrounded by % characters.

- The strings for Header and Data often need to have CR and LF characters in them. You can put in the text \r and \n. These two-character sequences are replaced by the carriage and newline characters when found. Typically, headers separate the parts with CR-LF characters.

Receive

Depending upon the operation you are doing there may be a result or not. If there isn't one or you don't care about the result you can choose the "No reply expected".

If you are expecting a reply it can be saved in a variable or in a file. A good use of a file is for when receiving an image - as when communicating with a camera for a snapshot or when receiving a web page that you want to display.

Many web-based devices use JSON to encode their inputs and outputs. HCA has JSON decode facilities that you can use. There is a technical note on that.
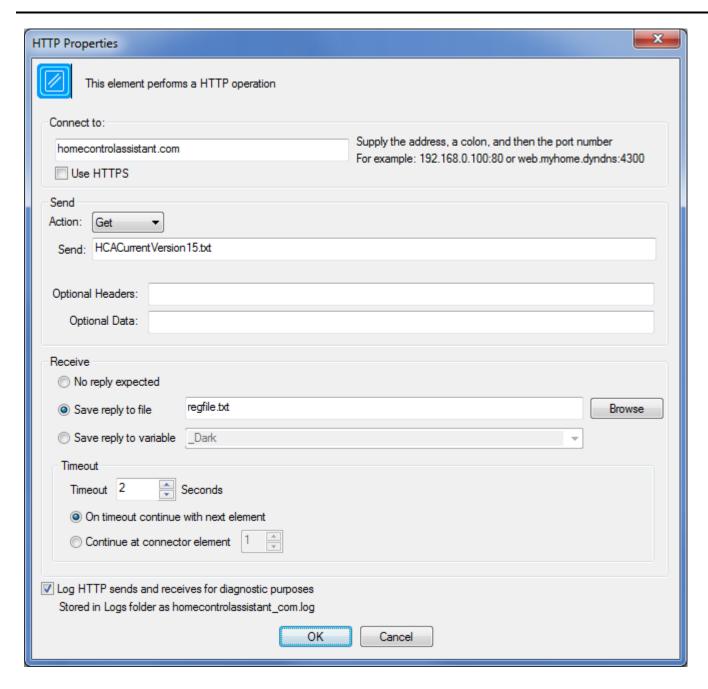
Log
The last element of the dialog is a checkbox to enable logging. It can be complex to see exactly what you are sending and receiving and if you enable logging then a file will show the complete transaction. This is very helpful for debugging.

An example you can try

Here is a HTTP element to read the file that HCA uses when it starts to see if your copy of HCA is out of date. The file HCACurrentVersion14.txt is in the root of the HCATech.com web site. This element will "GET" that file and save it to your disk.

HTTP Properties

This element performs a HTTP operation

Connect to:

homecontrolassistant.com

Supply the address, a colon, and then the port number
For example: 192.168.0.100:80 or web.myhome.dyndns:4300

☐ Use HTTPS

Send

Action: Get

Send: HCACurrentVersion15.txt

Optional Headers:

Optional Data:

Receive

◯ No reply expected

◉ Save reply to file   regfile.txt   Browse

◯ Save reply to variable   _Dark

Timeout

Timeout: 2 Seconds

◉ On timeout continue with next element

◯ Continue at connector element   1

☑ Log HTTP sends and receives for diagnostic purposes
Stored in Logs folder as homecontrolassistant_com.log

OK   Cancel

##end##