

## Using the HTTP Server Driver for Siri Control

### Overview

The HTTP server driver provides a mechanism for triggering events based on receiving an HTTP request from a third-party device. This driver can be used to create triggers on the processor that may be used to create shortcuts on a supporting iOS device. Once these shortcuts are in place, end-users may enjoy Siri's voice control as well.

### Configuring the Driver

Start by adding the driver to the global area of your completed system file. Once the driver has imported, click on the driver in the system tree and select driver properties in the driver configuration partition.

<b>Driver Configuration</b>	<b>System Settings</b>	
Driver Utilities...	Processor	Default (Master)
<b>Driver Properties</b>	<b>Server Settings</b>	
Driver Events	Port	8080
Driver Info	Heading	Heading
	Text	Some text
	Render Buttons	<input type="checkbox"/>
	Number of events	1
	<b>Event 1</b>	
	Name	Event 1
	Action	/event1
	<b>Licensing</b>	
	Unlock code	

### System Settings

Processor – Select the processor the driver runs on (Master Processor is the default.)

### Server Settings

- Port – select a TCP port for the driver to listen on. Note: Do not use a port that is being used and does not use Port 80.
- Heading – It is possible to create a web page for preliminary testing before entering the Siri shortcuts on the iOS device. Enter a name for the overall system, such as "Smith Residence." This heading name will appear on top of the testing page.
- Text – Like the heading, this is used to describe the shortcuts you will test on the webpage. Set this to "Siri Control" to describe the testing.
- Render Buttons – Leaving this checked will build a testing web page, where the heading and text appear in addition to the buttons tied to events. Be sure to uncheck this for security purposes after testing is completed.
- Number of Events – The number of events you wish to use. The driver supports 100 events. Another driver instance can be added for additional events, but you must select another port number in the second driver.

### Events

- Name – Enter in a name for the event that is close to the actual Siri command. For example, Watch Cable, or Foyer Light On."
- Action - A unique identifier for your events HTTP request. This action can be a forward slash, followed by an identifier. /watchcable and /foyerlighton are examples of identifiers using the event name as an example.

Go through and configure each name and action for all your intended shortcuts and voice control items.

Number of events	6
<input type="checkbox"/> Event 1	
Name	Office Lamp On
Action	/officelampon
<input type="checkbox"/> Event 2	
Name	Garage Door Close
Action	/garagedoorclose
<input type="checkbox"/> Event 3	
Name	Arm Home
Action	/armhome
<input type="checkbox"/> Event 4	
Name	Foyer Chandelier On
Action	/foyerchandelieron
<input type="checkbox"/> Event 5	
Name	Office Watch TV
Action	/officewatchtv
<input type="checkbox"/> Event 6	
Name	Watch Family Room Roku
Action	/watchfamilyroomroku

## Configuring Events for Each Action

Now that you have identified each action and named them accordingly, it is now time to program events for each action. Click on "Driver Events" in the driver configuration partition. Select and program each event in the list.

Driver Configuration

Driver Utilities...

Driver Properties

Driver Events

Driver Info

Enable	Category	Event	Has Macro
<input checked="" type="checkbox"/>	HTTP Events	Office Lamp On	Yes
<input type="checkbox"/>	HTTP Events	Garage Door Close	No
<input type="checkbox"/>	HTTP Events	Arm Home	No
<input type="checkbox"/>	HTTP Events	Foyer Chandelier On	No
<input type="checkbox"/>	HTTP Events	Office Watch TV	No
<input type="checkbox"/>	HTTP Events	Watch Family Room Roku	No

Button Properties

command

Lutron RadioRA2

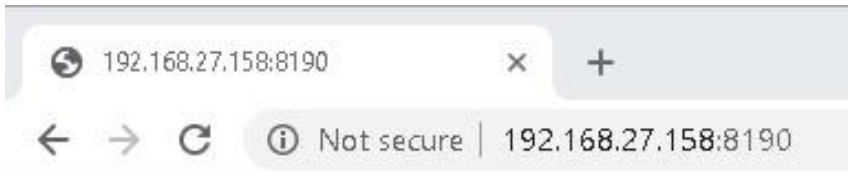
Switches \ Switch Commands

Office Lamp (ID 2)

On

### Testing Events Using a Web Browser

If you checked the "Render Buttons" argument in the driver, you would be able to test each event you created from a web browser. Enter in the following address: [HTTP://\[PROCESSOR IP\]:\[PORT\]](http://[PROCESSOR IP]:[PORT]) The port is the port that was specified in the driver.



Test each button and make sure it triggers the event properly. When you are certain your events are working correctly, uncheck the "render buttons" in the driver properties. This function should only be used for testing, and if left, could pose a security risk.

## Smith Residence

### Siri Commands

Office Lamp On

Garage Door Close

Arm Home

Foyer Chandelier On

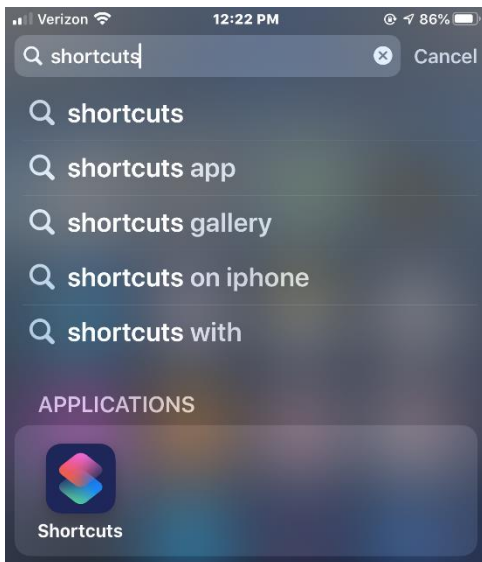
Office Watch TV

Watch Family Room Roku

If you continue having issues, ensure that the driver was configured correctly. You may also try using another port. If you continue to experience problems, please contact RTI for assistance. Do not proceed until you have tested the buttons properly.

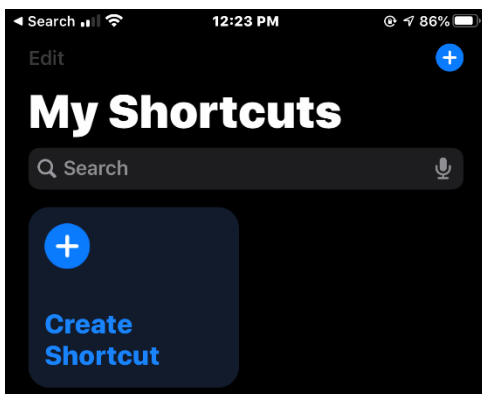
### Create Shortcuts on IOS Device.

You may now create the shortcuts on the iOS device. A list of supported devices is below. NOTE: Creating these on one device should make them appear on other devices using the same account. For example, if you create these on your iPhone, it should appear on your iPad as well.

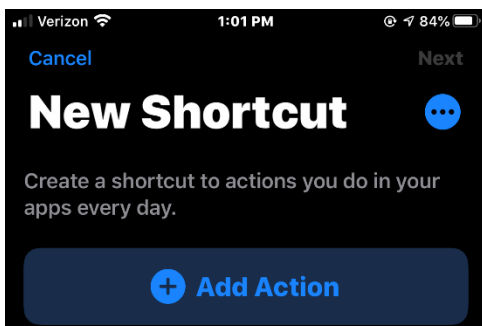


First, look for the Siri shortcuts by searching for Shortcuts. If you don't have them and can't download them from the app store, that may mean your device does not support Siri's shortcuts and won't be able to use this function.

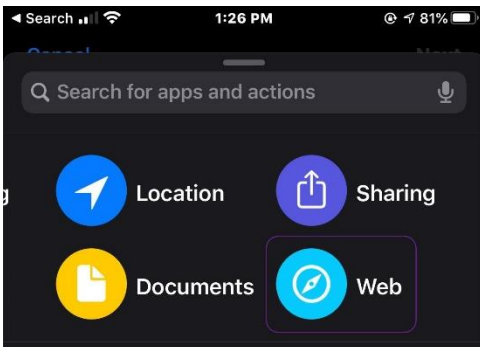
Select "Shortcuts" and open the app.



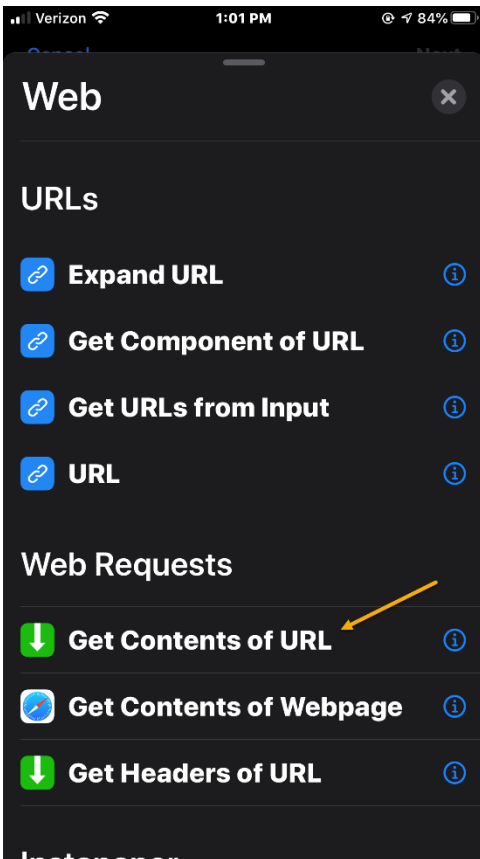
Next, select "Create Shortcut."



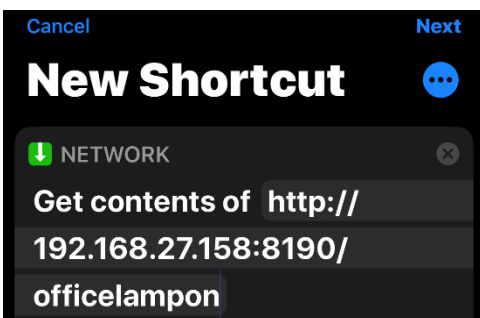
Select "Add Action."



Scroll until and select the "Web" option and select it.

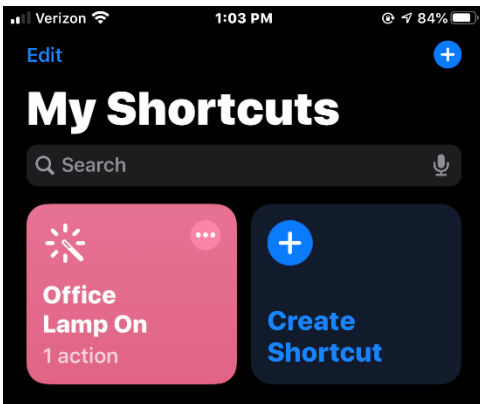


Scroll down to the category called "Web Requests" and select the option "Get Contents of URL."



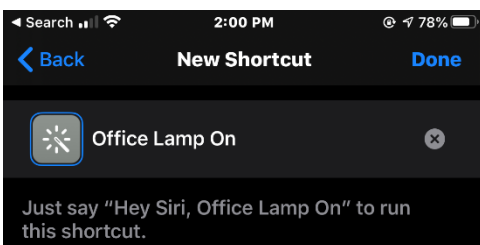
Select the URL and enter the processor IP address, TCP Port, and action name that was configured in the driver. Use the following format:  
HTTP://[Processor IP]:[Port]/function

Click "Next" when you are done.

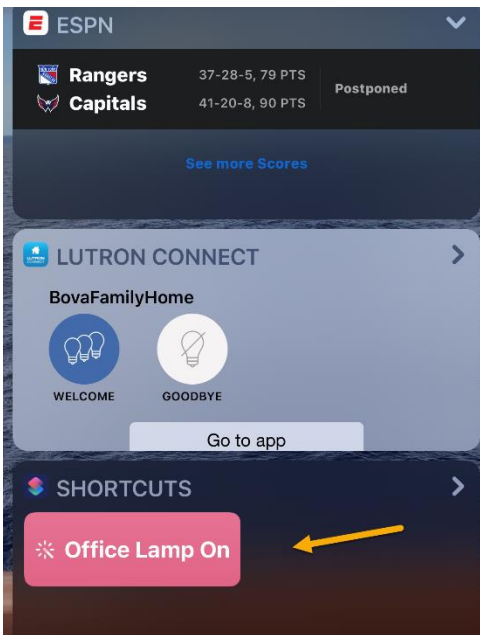


A shortcut is now created. The first time you press the shortcut, it may ask you for permission to access the URL. Select "Yes" so that subsequent presses do not require permission.

Test the button and make sure it works.



Give your action a name. It will be the name of the shortcut as well as the voice command for Siri. Click "Done" and continue to enter in additional shortcuts for all the actions created in the driver.



All shortcuts created will appear in the widget menu on your iOS device. On an iPhone, widgets can be accessed by swiping to the left on the home screen.

Other devices, such as iPads, provide access to widgets on their home pages.

Scroll down until you see your shortcuts and press them for additional testing. Continue to test by saying, "Siri, Office Lamp On."

All that's left to do now is create shortcuts for your other events/actions. If you press and hold the ellipsis in the top right corner of your newly created shortcut, you will find there's a duplicate option in the context menu that appears. You can use this duplicate option to make a copy of your first shortcut quickly and then you just need to modify the end of your GET request and uniquely name the duplicated shortcut.

If you create these shortcuts your iPhone and also have an iPad that is signed in to the same Apple ID, the shortcuts will also automatically appear on the iPad providing it also has Siri Shortcuts installed from the App store.

Now all that's left to do is start asking Siri to trigger your actions. To do this, activate Siri on your iPhone, iPad, Apple Watch, or Homepod and say the name of the shortcut you want to trigger.

### Driver Notes

- This driver was tested with an iPhone 7 and newer generation iPad. It should be compatible with all iOS devices that support shortcuts.
- This driver was tested on an Apple Watch Generation 5. It should work on previous watch generations that support shortcuts (Generation 1 and 2 do not.)
- Using this driver outside of the home could pose a security risk, and RTI does not recommend setting up the driver for remote commands.
- Once the shortcuts have been tested on the web page, RTI recommends removing the render button's checkmark in the driver for security purposes.
- This driver is not an official Apple integration and is not supported by Apple.