

CCI Configuration Suite Lite Overview

Control Concepts, Inc.

Date: 4/15/2022

Revision: 2

Crestron 3-Series Controller Webserver Authentication

By default the webserver on Crestron controller has no authentication and SSL is disabled. Basic Authentication can be enabled along with SSL (HTTPS). For this project, we are assuming that SSL will be enabled along with Authentication.

Control Concepts Configuration Suite Overview

Description

The Crestron controller allows for custom webpages to be created using the onboard IIS webserver. Crestron calls the technology CWS (Crestron Web Scripting).

Control Concepts (CCI) has created a module suite that automatically generates configuration webpage(s) on the Crestron controller and uses the created webpage(s) to transfer data to the controller and running programs. This document describes how the CCI solution operates.

Crestron CWS Basics

The HTTP protocol uses URL paths to access the webserver and send requests. The Crestron CWS root URL path is [http\[s\]://<processor ip address>/cws](http[s]://<processor ip address>/cws) for 3-series and 4-series processors and [http\[s\]://<vc4 ip address>/VirtualControl/Rooms/<room ID>/cws](http[s]://<vc4 ip address>/VirtualControl/Rooms/<room ID>/cws) for VC-4. This is the root path to access the CWS functionality.

The control system needs to add “routes” on top of this URL path in order to use and manage the Crestron CWS feature. Many routes can be added, but the CCI solution uses a two level route, [<root path shown above>/<custom root>/webconfig](http[s]://<root path shown above>/<custom root>/webconfig). The “custom root” URL segment portion can be virtually anything, and there can be multiple in any system. The included CCI Config Processor module allows you to determine the custom root segment to use for a particular webpage. For the purposes of this document, we will use “cci” as the custom root segment going forward.

CCI Configuration Suite Details

The CCI Configuration Suite contains several modules that are all used in conjunction to create and communicate with a configuration webpage that lives on the Crestron controller.

The primary module in the package, CCI Config Processor, is the “traffic cop” module. Its primary function is defining the URL route to a configuration webpage, listening for changes on that webpage and passing the updated webpage data down to the individual component modules to send updates back to the control system. The lite version of the configuration suite library allows one CCI Config Processor module to be registered with a maximum of ten categories.

The individual component modules are used to define what items will be included on the webpage. The component modules register to a specific command processor. The command processor, in turn, will take the information from all the registered component modules and use that to 1) create a JSON file on the processor (for persistent configuration storage) and 2) generate and dynamically build the webpage. If the JSON file already exists when the program starts, it will use the saved values to build the webpage. The lite version of the configuration suite library allows a maximum of 30 component modules to be registered.

The root path that has been set on the command processor will determine the URL you will need to use to access the webpage. If, for example, your controller IP address is 192.168.1.100 and you set a root path on the processor of “ConferenceRoomA”, the URL would be:

[http\[s\]://192.168.1.100/cws/ConferenceRoomA/webconfig](http[s]://192.168.1.100/cws/ConferenceRoomA/webconfig)

For the above 3-series/4-series example, the JSON file that is generated would live on the processor in the global “User” folder and would be named “ConferenceRoomA.json”.

If the same information was used in a program loaded to a VC-4 room with the ID of “TESTROOM1”, the URL would be:

[http\[s\]://192.168.1.100/VirtualControl/Rooms/TESTROOM1/cws/ConferenceRoomA/webconfig](http[s]://192.168.1.100/VirtualControl/Rooms/TESTROOM1/cws/ConferenceRoomA/webconfig)

On a VC-4 system, each running program will have its own local filesystem. So for the above VC-4 example, the JSON file that is generated would live on the VC-4 server at this location:

</opt/crestron/virtualcontrol/RunningPrograms/TESTROOM1/User/ConferenceRoomA.json>

Note that if you have authentication enabled on your controller, you would use https instead of http and you would be required to enter your credentials before accessing the webpage.

Example Webpage

Below is an example of the webpage that will be created upon program start. Note that the categories and options shown are examples only and what you actually see in your browser will depend entirely on which categories and component modules you have added to your program. The example below is a slightly modified version of the page that will be generated by the included demo program.

WebConfig

temp

CONTROL CONCEPTS

Category 1

Category 2

Boolean Item

☐ True

☒ False

Integer Item

0

Signed Integer Item

0

String Item

Enter text

Max Characters: 100

This field cannot be blank

IP Address Item

IP Address

0.0.0.0

This field cannot be blank or 0.0.0.0

IP Port

23

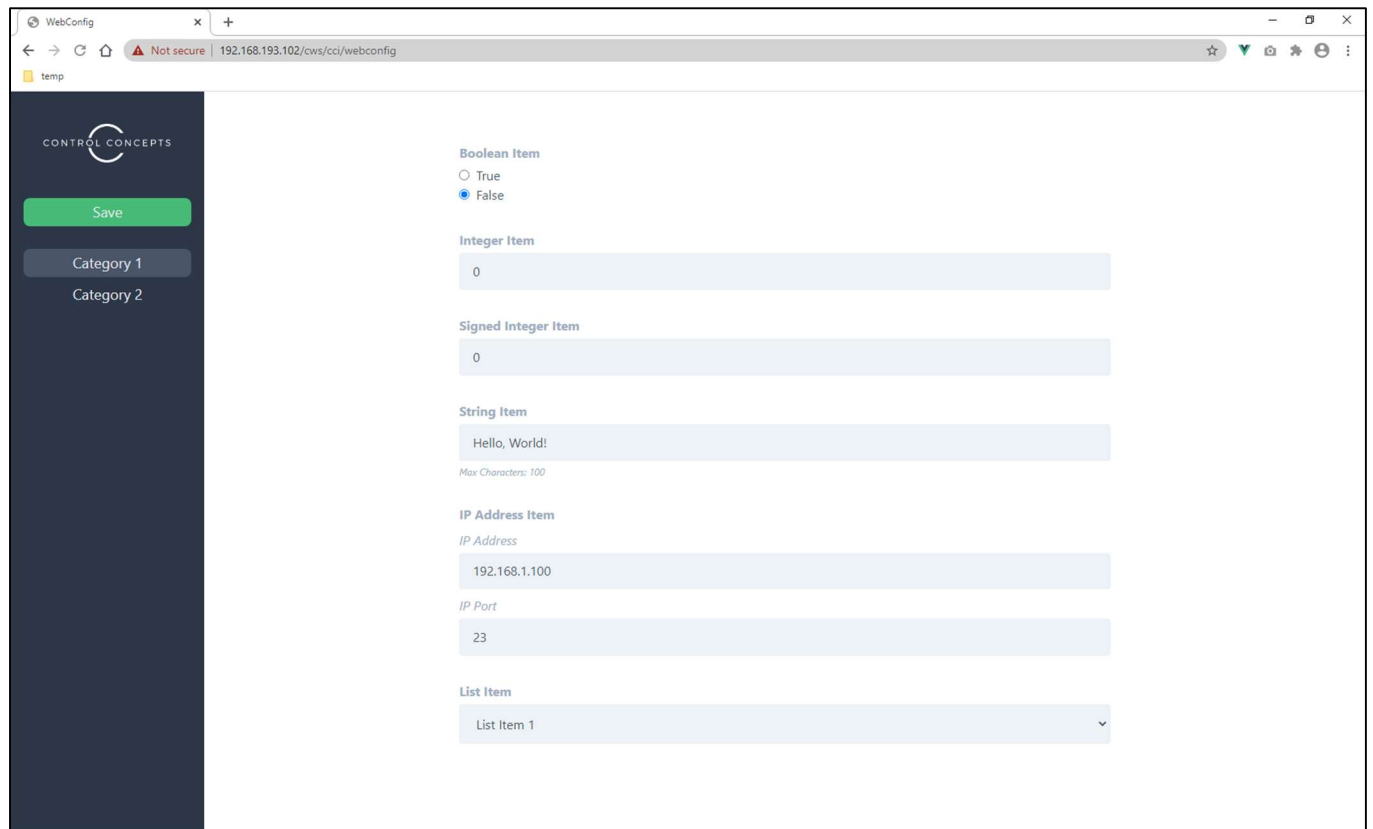
List Item

List Item 1

A few things of note from the screenshot.

- Notice there is an error indicator next to Category 1. This is because there are items in that category that have errors. Once the errors for all items in a category have been cleared, the error indicator will disappear and a Save button will appear, allowing you to send the updated data back to the controller.
- The String item is highlighted because we have set the `IsOptional` property on the corresponding component module to `False`, meaning this field is not allowed to be empty. Once we fill in some data in this field, the error will disappear.
- The IP address item is highlighted because the IP address entered is not valid. Once a valid IP address is entered, the error will disappear.

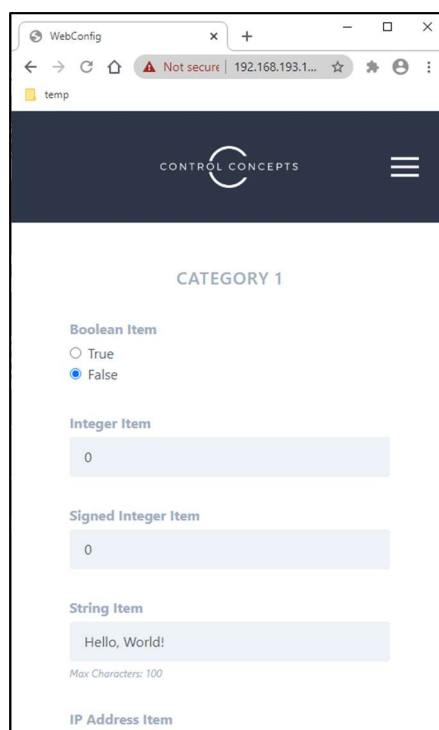
Below is an example showing the same page with all errors cleared. Note that now the Save button is visible. Pressing this button will send all changed values back to the control system.



The screenshot shows a web browser window with the address bar displaying "WebConfig" and the URL "192.168.193.102/cws/cc/webconfig". The page has a dark blue sidebar on the left with the "CONTROL CONCEPTS" logo at the top. Below the logo is a green "Save" button, followed by two tabs: "Category 1" (selected) and "Category 2". The main content area is white and contains the following form elements:

- Boolean Item:** Two radio buttons, "True" and "False". The "False" button is selected.
- Integer Item:** A text input field containing the value "0".
- Signed Integer Item:** A text input field containing the value "0".
- String Item:** A text input field containing the value "Hello, World!". Below the field is the text "Max Characters: 100".
- IP Address Item:** Two text input fields. The first is labeled "IP Address" and contains "192.168.1.100". The second is labeled "IP Port" and contains "23".
- List Item:** A dropdown menu showing "List Item 1" with a downward arrow.

One last thing to note is that the webpage is fully responsive, meaning you will be able to use it from smaller devices such as tablets or phones.



The screenshot shows the same web application on a mobile device. The browser window is narrower, and the sidebar has collapsed into a hamburger menu icon (three horizontal lines) in the top right corner. The "CONTROL CONCEPTS" logo is centered at the top of the dark blue header. Below the header, the text "CATEGORY 1" is centered. The form elements are the same as in the desktop view, but they are stacked vertically and scaled to fit the mobile screen width.