

Initial Installation and Set Up

- [Running the Stream Chamber For the First Time](#)
- [Installing DLC Files](#)
- [Utilizing the Stream Deck Functionality](#)

Running the Stream Chamber For the First Time

Currently, you download a ZIP file containing the latest version from your user page on the main website.

To get set up afterwards, follow these steps:

1. Unzip the "StreamChamber.zip" file where you want to keep it
2. Navigate to "[Your Folder]/V1.XX/Windows" and run the Shortcut, this starts up the Stream Chamber *and* the webserver powering the web app and websockets
3. You will now be presented with a menu asking for your username and your license key
4. Next will be a pop up asking where to put your CSVs, which contain all the information for your teams, players, squads, and sponsors
 1. As well, if you have custom image sequences or videos as part of DLC, they would go here in their own folder underneath the folder you just selected. For example "[Your CSV Folder]/CustomContent/" and then in there would be any image sequences or videos that need it.
 2. Regular image and video scenes are fine with just the file path in the conductor.
5. To launch the web app on the same PC, go [here](#) or to <http://localhost:30000> in your address bar.

You should now be all set up and ready to move on to the section about what each panel does!

Installing DLC Files

Currently, to install any DLCs you get, you just have to copy them to the same directory as the main content.

For example, on my PC, if I am given a file called "pakchunk1-Windows.pak", that would be my DLC file, and I would put it in this folder:

"[Path to StreamChamber]\V1.15\Windows\StreamChamber\Content\Paks", where you would find any other DLCs you copied in, as well as the base "pak", called **"pakchunk0-Windows.pak"**.

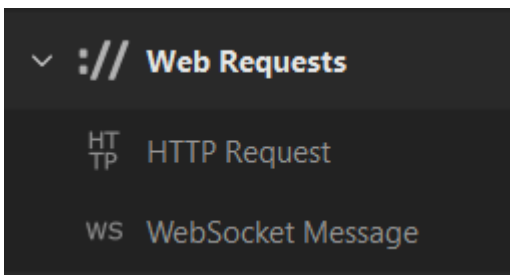
Utilizing the Stream Deck Functionality

The Recommended Way (WebSockets)

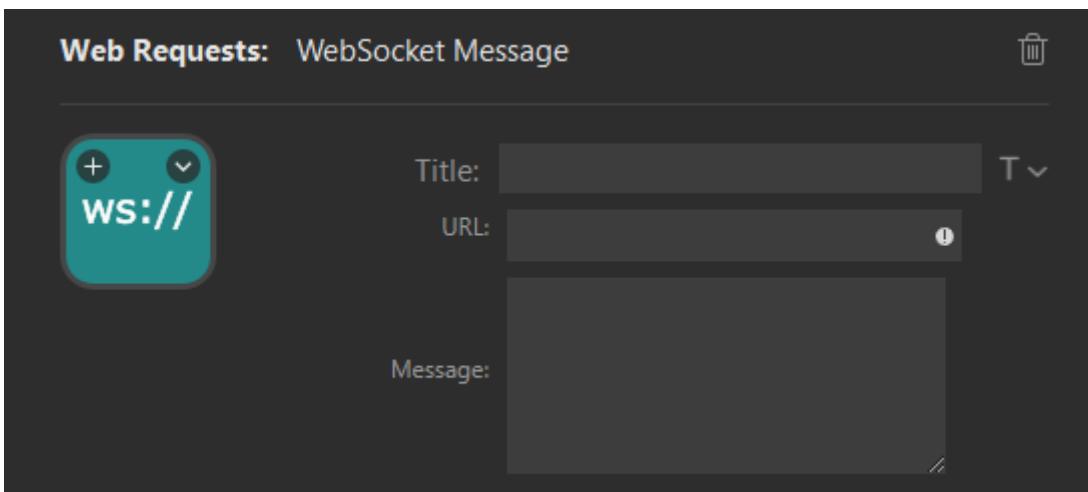
In order to use your Stream Deck with the Stream Chamber, you need to download the free plugin at this link and install it to your Stream Deck software.

<https://marketplace.elgato.com/product/web-requests-d7d46868-f9c8-4fa5-b775-ab3b9a7c8add>

Once you have done that, you should be able to see a category called Web Requests on the right side menu.



From there, you drag in the *WebSocket Message* option and should see this:



For this demo, we will use the idle screen as the scene we are creating the button for.

1. In the *Title* box, type "Idle"

2. In the *URL* box, you type "ws://127.0.0.1:30020" to connect to the web socket the Stream Chamber creates for you once it finishes launching. you *must* include the ws:// at the beginning otherwise this will not work
3. In the *Message* box, you need to copy and paste this code block below into it, as this is where the actual "magic" happens.


```
{
  "MessageName": "http",
  "Parameters": {
    "Url": "/remote/preset/rcmain/function/change scene",
    "Verb": "put",
    "Body": {
      "Parameters": {
        "Scene": "idle"
      }
    },
    "GenerateTransaction": true
  }
}
```

When you are creating the other buttons, you want to change line 8 to have the correct scene parameter. So for example, winner would be:

```
{
  "MessageName": "http",
  "Parameters": {
    "Url": "/remote/preset/rcmain/function/change scene",
    "Verb": "put",
    "Body": {
      "Parameters": {
        "Scene": "winner"
      }
    },
    "GenerateTransaction": true
  }
}
```

And the only difference is the word on line 8. Make sure to keep everything else, quotation marks and so on, as is, because the Stream Chamber expects the format in this specific way.

Your final button information should look like this:

Web Requests: WebSocket Message 

+ -
WS://
Idle

Title: Idle T v

URL: ws://127.0.0.1:30020 ✓

Message:

```
{
  "MessageName":
  "http",
  "Parameters": {
    "Url":
    "/remote/preset/rcmain/function/change scene",
    "Verb": "put",
    "Body": {
      "Parameters":
      {
        "Scene": "idle"
      },
      "GenerateTransaction": true
    }
  }
}
```

You can, of course, change the look of the button (custom image coming soon), and this won't change any functionality.