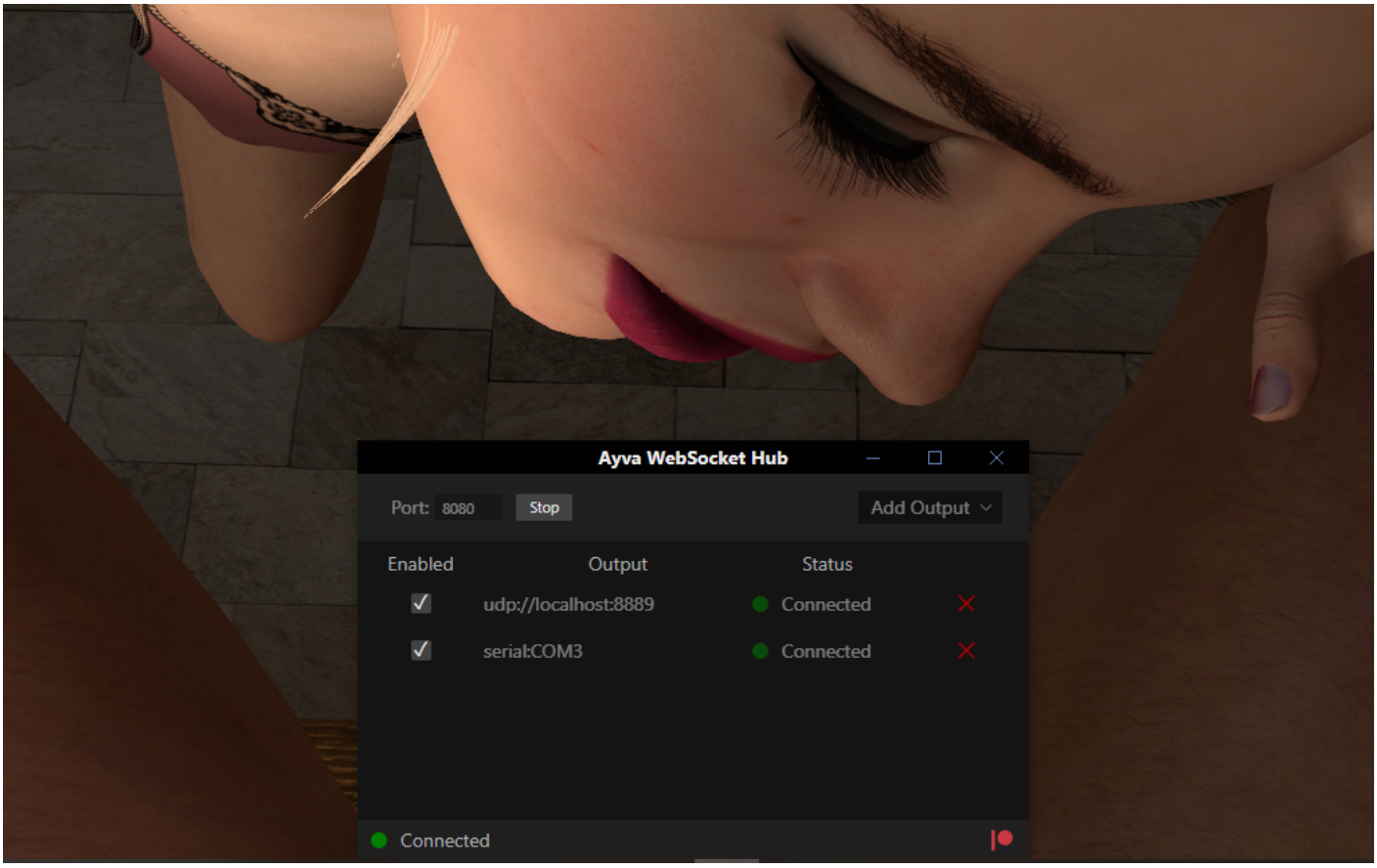
Ayva WebSocket Hub

A desktop application that allows streaming TCode from the web to multiple outputs, including Virt-A-Mate.

- Overview
- Installation
  - Installing on Windows
  - Installing on macOS
- Guide
  - Interface Overview
  - Output Types
  - Appendix
- Troubleshooting

Overview



Ayva WebSocket Hub is an open source *desktop* application that can stream TCode from *web* applications to any number of outputs (such as serial and network outputs) via a WebSocket. This is to bypass the limitations of web browser technologies that prevent making unsecure connections to devices over WiFi and/or streaming data into desktop applications.

The most common use cases are:

- Streaming TCode into a Virt-a-mate scene (requires the BusDriver plugin).
- Streaming TCode to a WiFi-enabled device.
- Streaming TCode to multiple devices and/or outputs at once.

It works with Ayva Software applications such as Ayva Stroker Lite and Ayva Remote. However, it can stream data from any application that can connect to a WebSocket.

Demos:

https://drive.google.com/file/d/1MM_YowIbsuETLv051WsqQs38sVMGvJZS/view

https://drive.google.com/file/d/1FOQllbYSnlJoBqvN3TjJSsV6DI5dRlyg/view

Latest Release (**Windows** and **macOS**):

https://github.com/ayvasoftware/ayva-websocket-hub/releases

How to Install:

https://osr.wiki/books/ayva-websocket-hub/chapter/installation

Github Repository:

https://github.com/ayvasoftware/ayva-websocket-hub

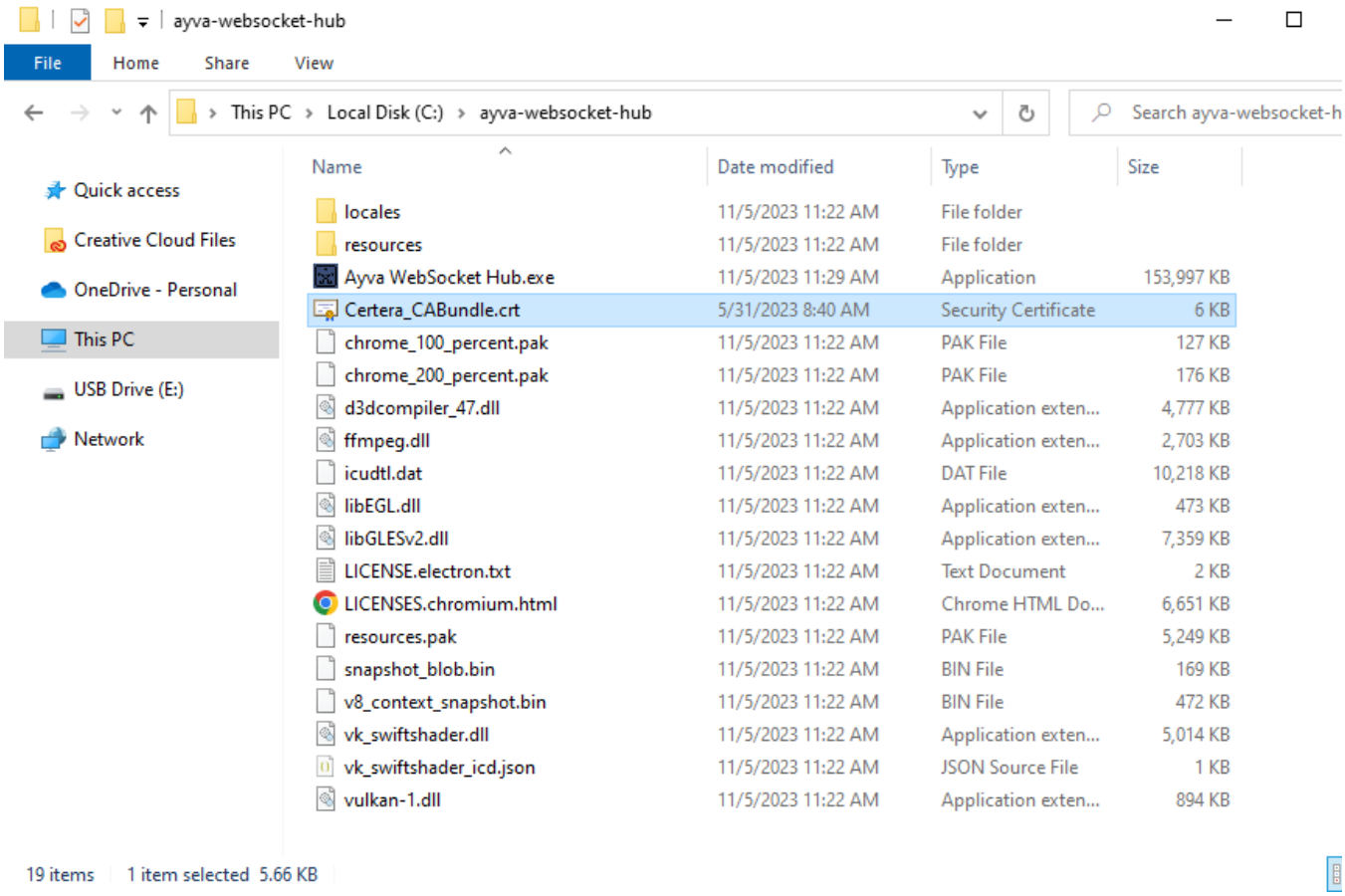
Installation

Installation steps for Windows and macOS.

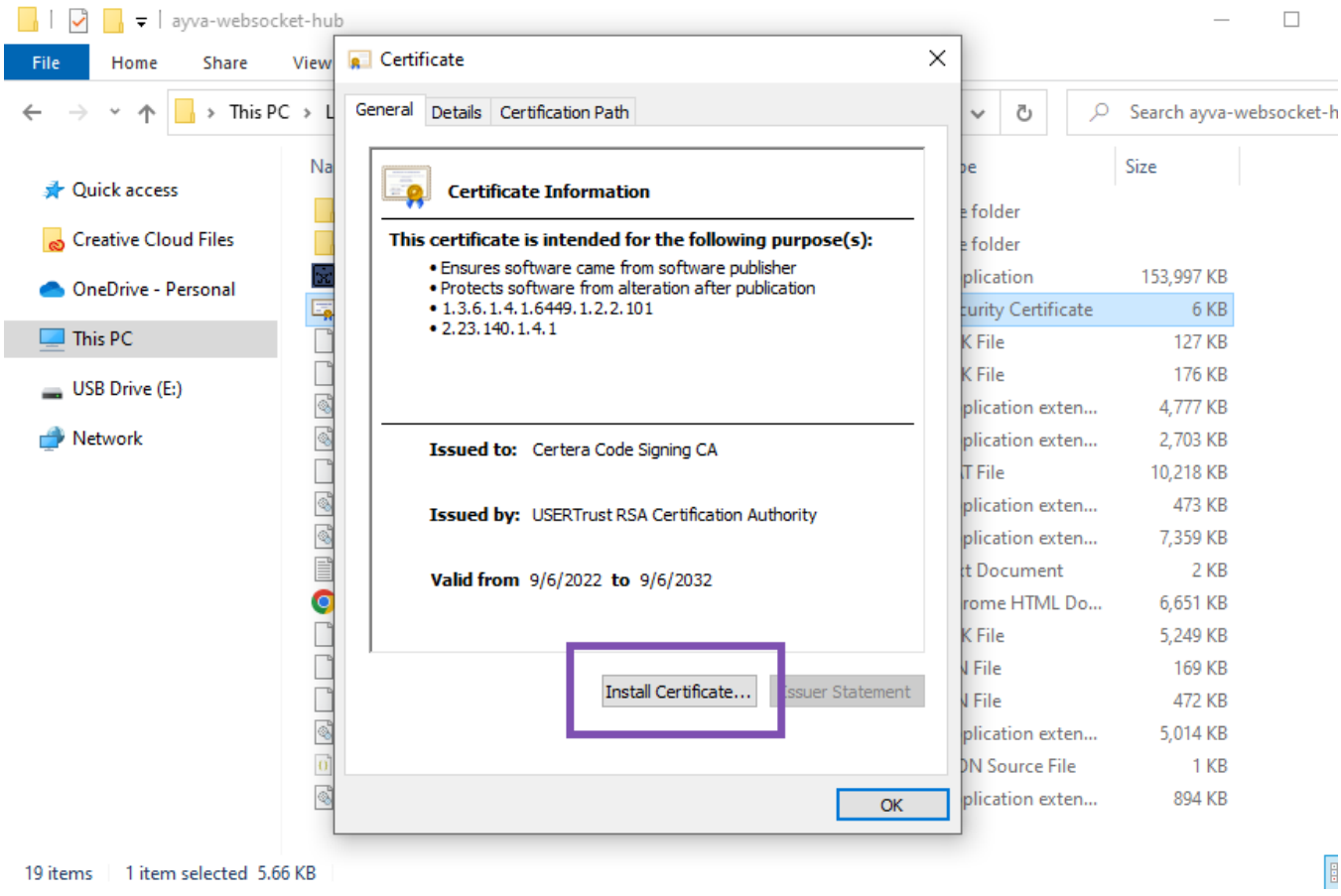
Installing on Windows

1. **Download** and unzip **Ayva WebSocket Hub (Windows).zip** into a directory of your choosing.

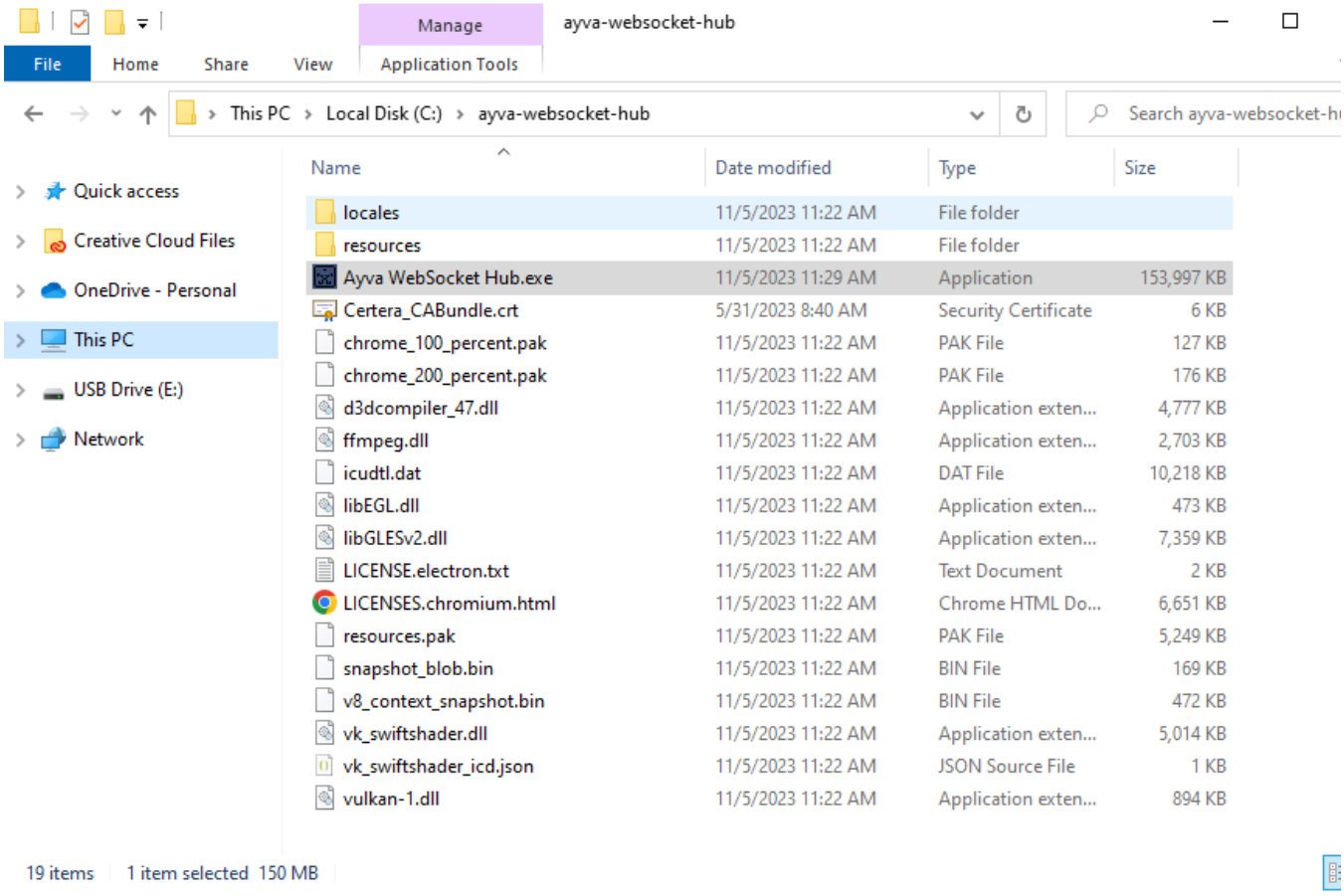
2. (Optional) **Certificate Installation.** Install the certificate for Certera by double-clicking on the **Certera_CABundle.crt** in the unzipped folder. *Note: This step is not required, but may prevent "Unknown Publisher" messages when running the application. Certera is the Certificate Authority (CA) that was used to digitally sign the application.*



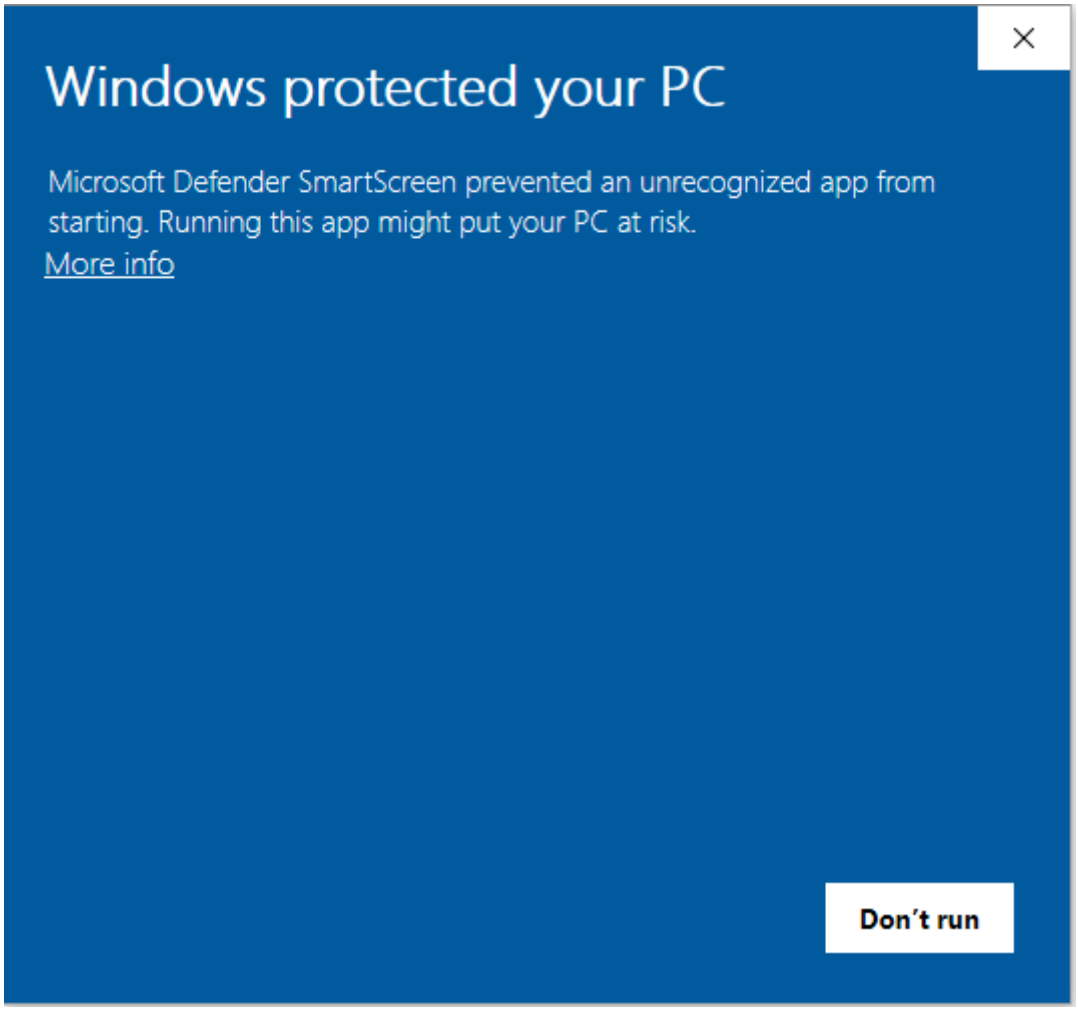
Click **Install Certificate** and keep the default settings in the installation wizard.



3. Double-click **Ayva WebSocket Hub.exe** to run the application.

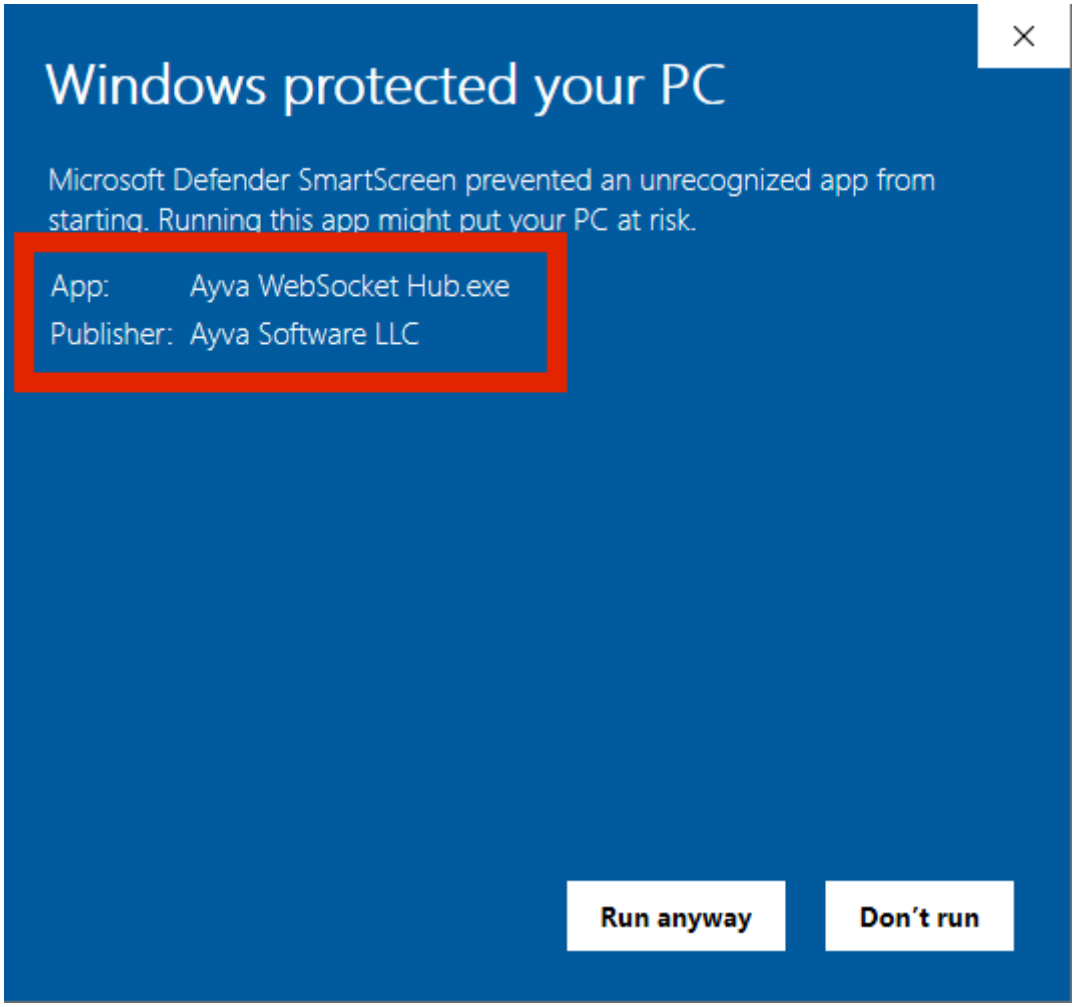


You may see the following warning message:

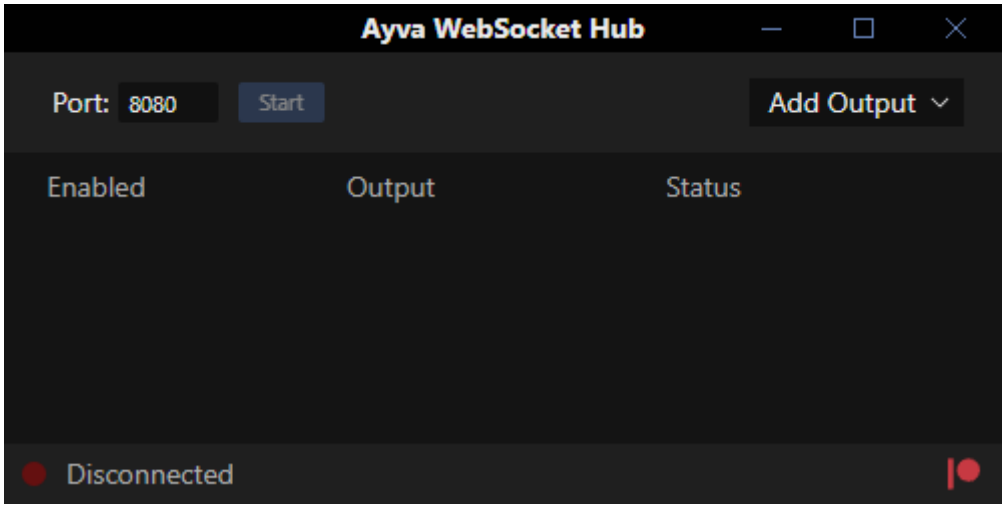


This is because the kind of certificate Ayva WebSocket Hub was signed with builds reputation over time.
Eventually this will stop appearing after enough users have installed the application (or enough time has passed).

If you click "More info" you can verify that the Publisher is Ayva Software LLC.



Click "Run anyway" and the application should open:

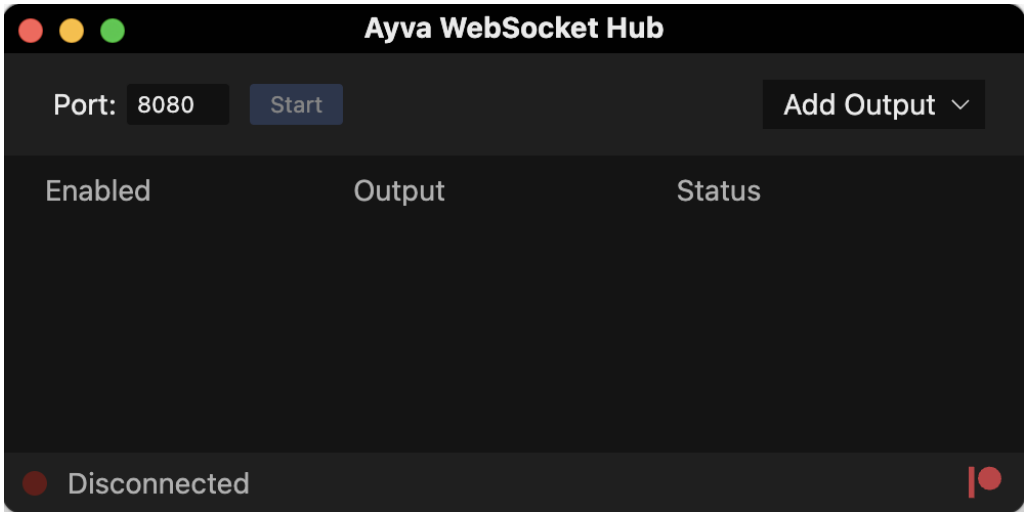


4. Consult the Guide for information about how to use it.

Installing on macOS

1. **Download** and unzip **Ayva WebSocket Hub (macOS).zip** into a directory of your choosing (your **Application**s folder is recommended).

2. Double-click **Ayva WebSocket Hub.app** and the application should open.

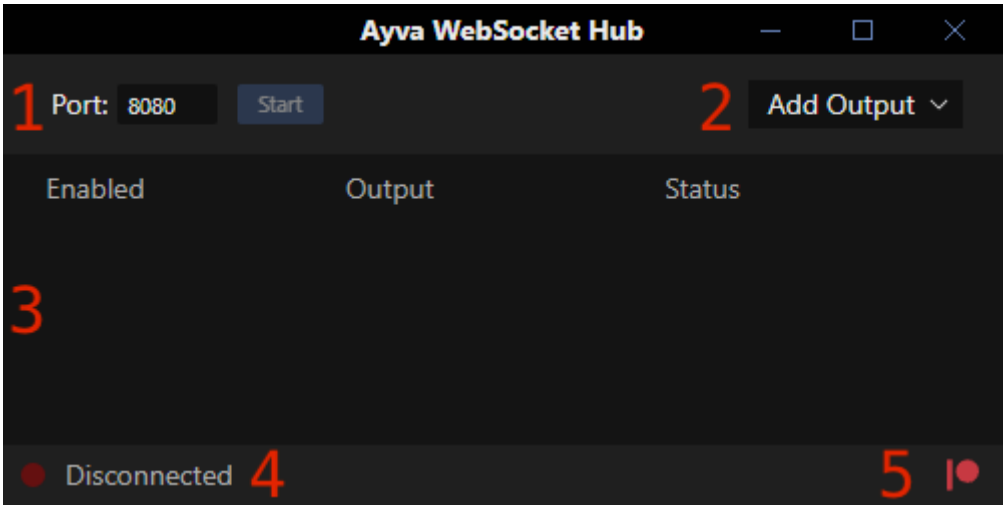


3. Consult the Guide for information about how to use it.

Guide

Guide

Interface Overview



1. **Listening Port** - Specify the port for Ayva WebSocket Hub to monitor incoming connections. Any TCode received on this port will be routed to the connected outputs. Click the *Start* button to begin monitoring for incoming connections. Data forwarding will commence immediately upon establishing a connection.

2. **Add Output** - Use this dropdown menu to select and add output streams.

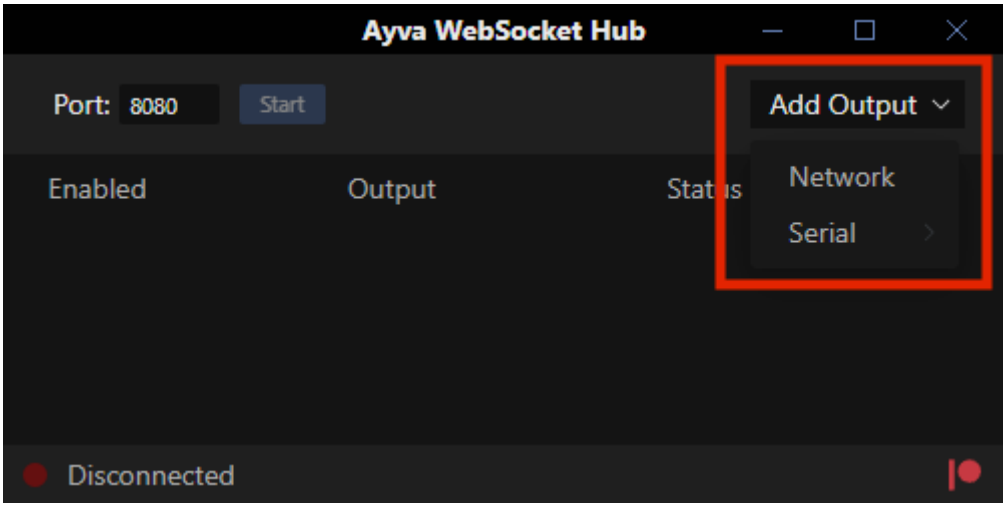
3. **Outputs** - This table lists all configured outputs—which are saved across sessions—along with their connection status. You can enable or disable streaming to any listed output by using the checkbox in the *Enabled* column.

4. **Incoming Connection State** - This indicates the status of the incoming connection, which can be *Disconnected*, *Listening* or *Connected.*

5. **Patreon Link** - Ayva Software is supported by the community. This is a link to the Patreon.

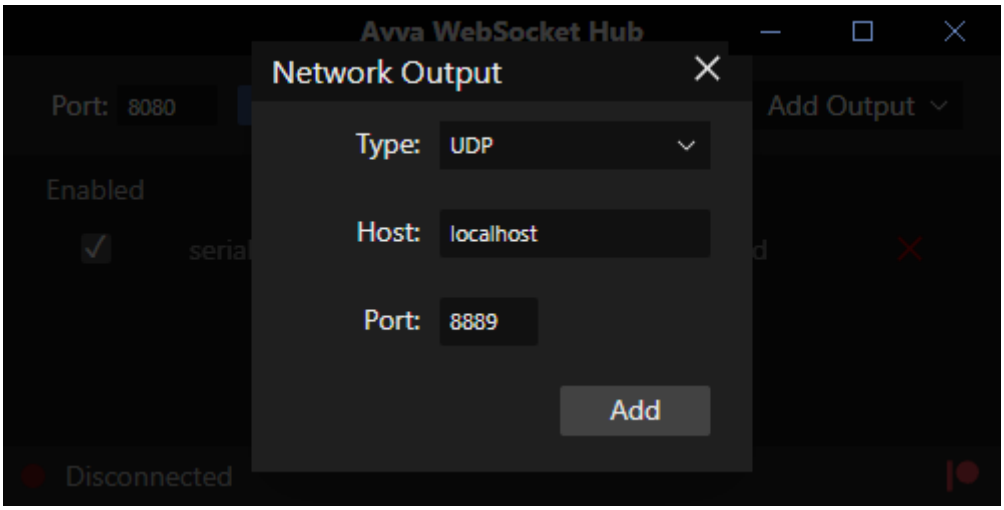
Guide

Output Types



There are two main types of outputs: **Network** and **Serial**.

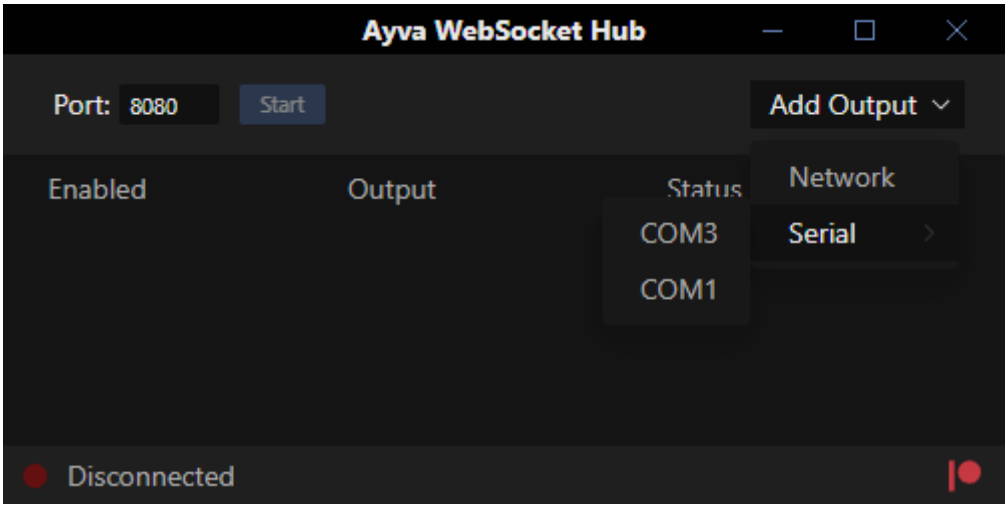
Network

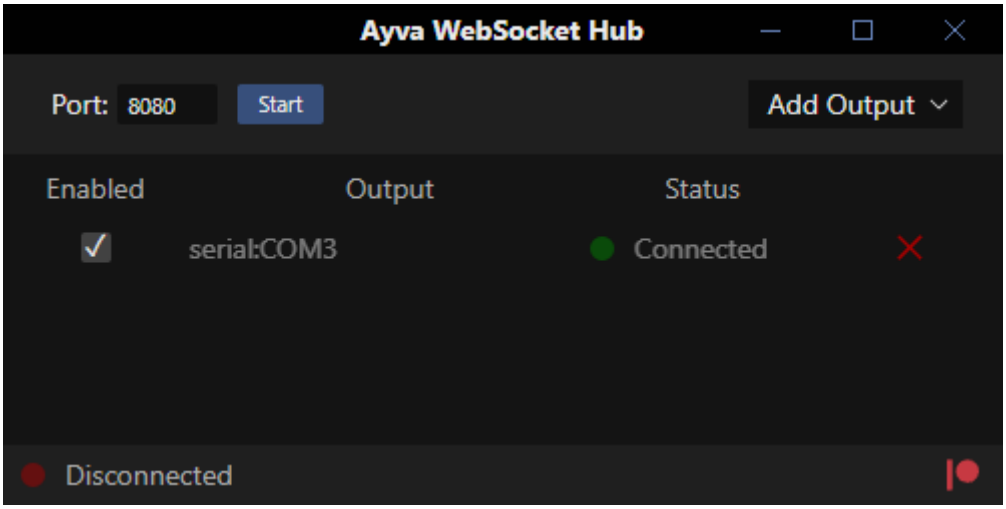


Network outputs enable connections to devices or software via TCP or UDP protocols. In TCP mode, Ayva WebSocket Hub attempts to establish a connection to a WebSocket at the `/ws` endpoint on the specified host and port. In UDP mode, packets are sent directly to the designated host and port. Note that because UDP is *connectionless*, a UDP output will always be displayed as *Connected*, even if no application or device is actively listening at the specified host and port.

TCP is typically used to connect to WiFi-enabled devices, while UDP is typically used to stream to a Virt-a-mate plugin such as BusDriver (though it can also be used for WiFi-enabled devices that support UDP).

Serial

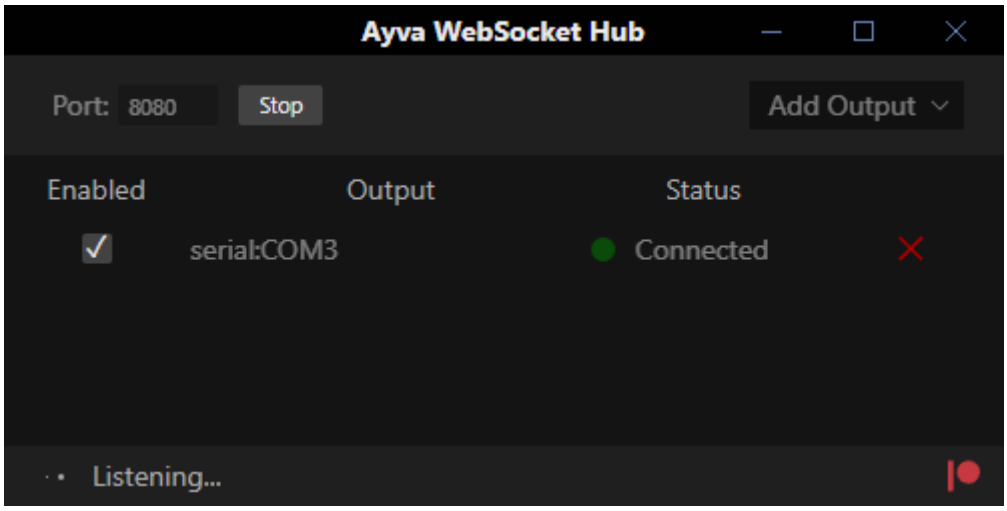


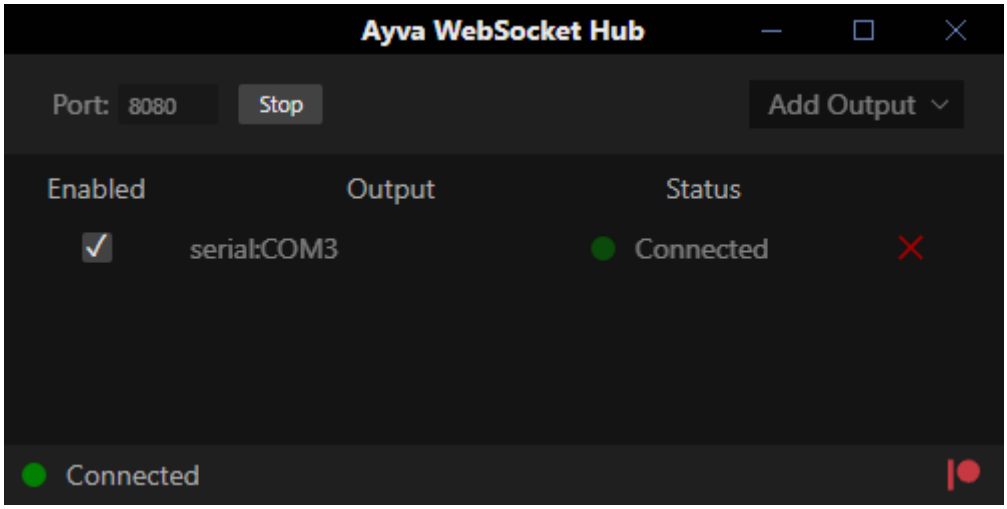


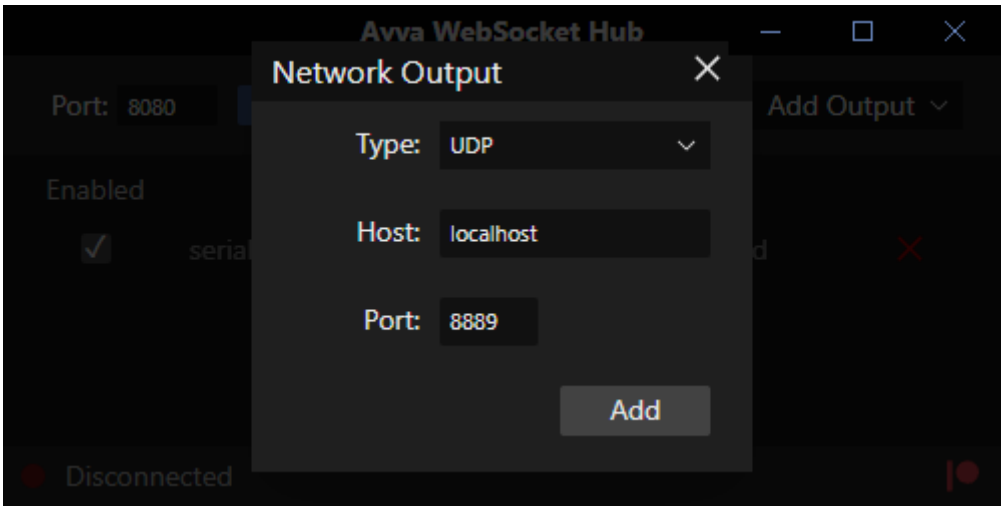
Serial outputs facilitate device connections via the Serial protocol, typically through USB. However, they can also operate over virtual serial ports that emulate physical connections for other protocols, including Bluetooth Classic.

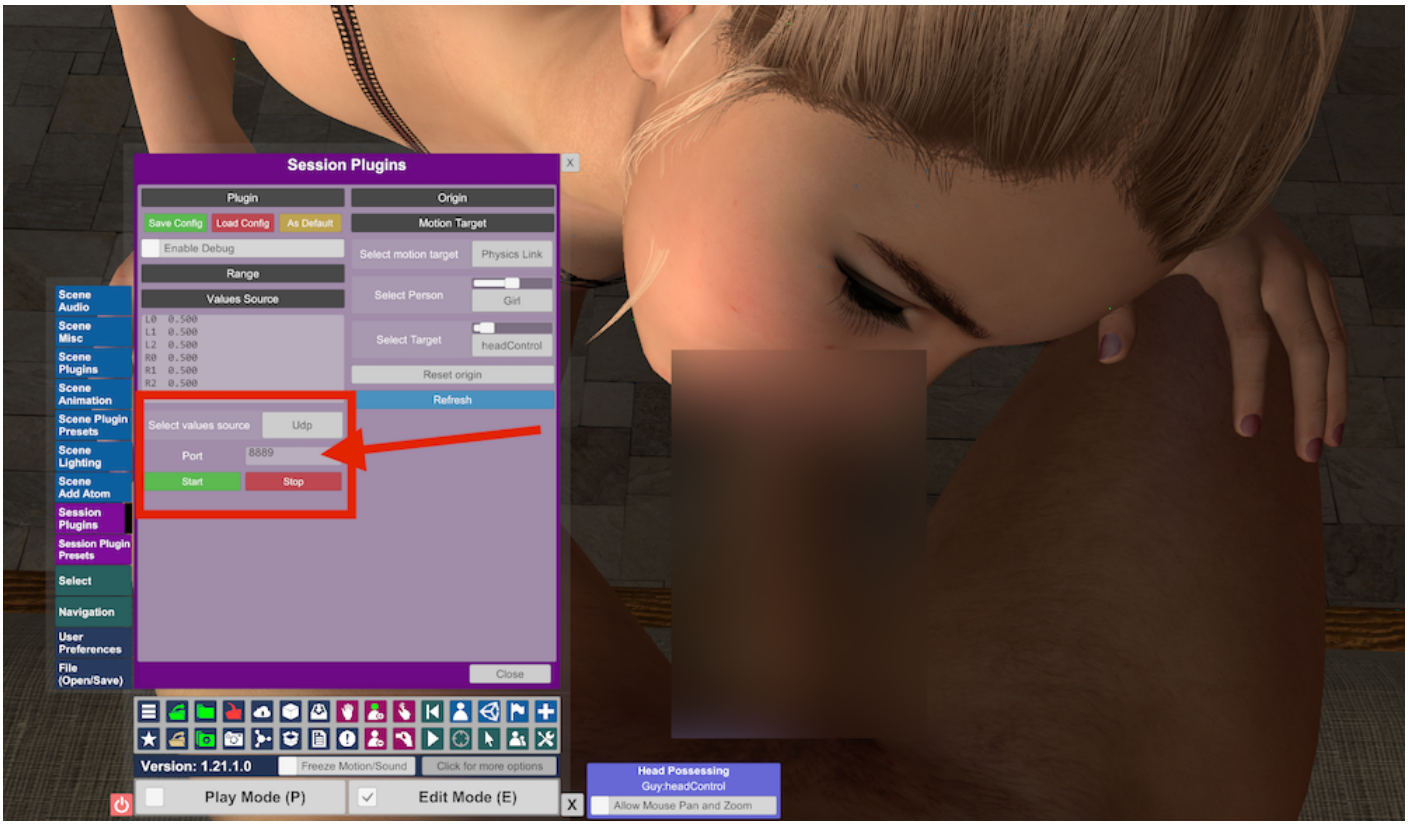
Appendix

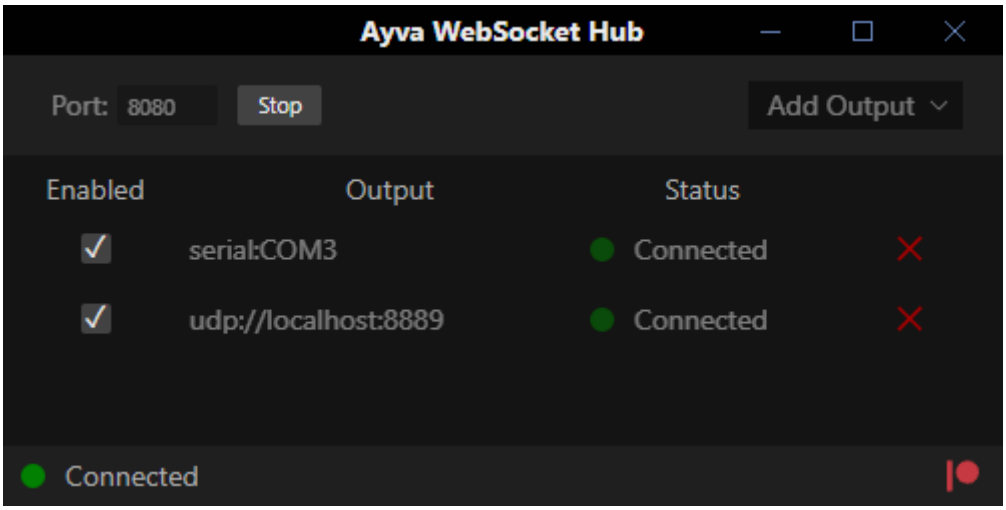
To be incorporated into the Guide



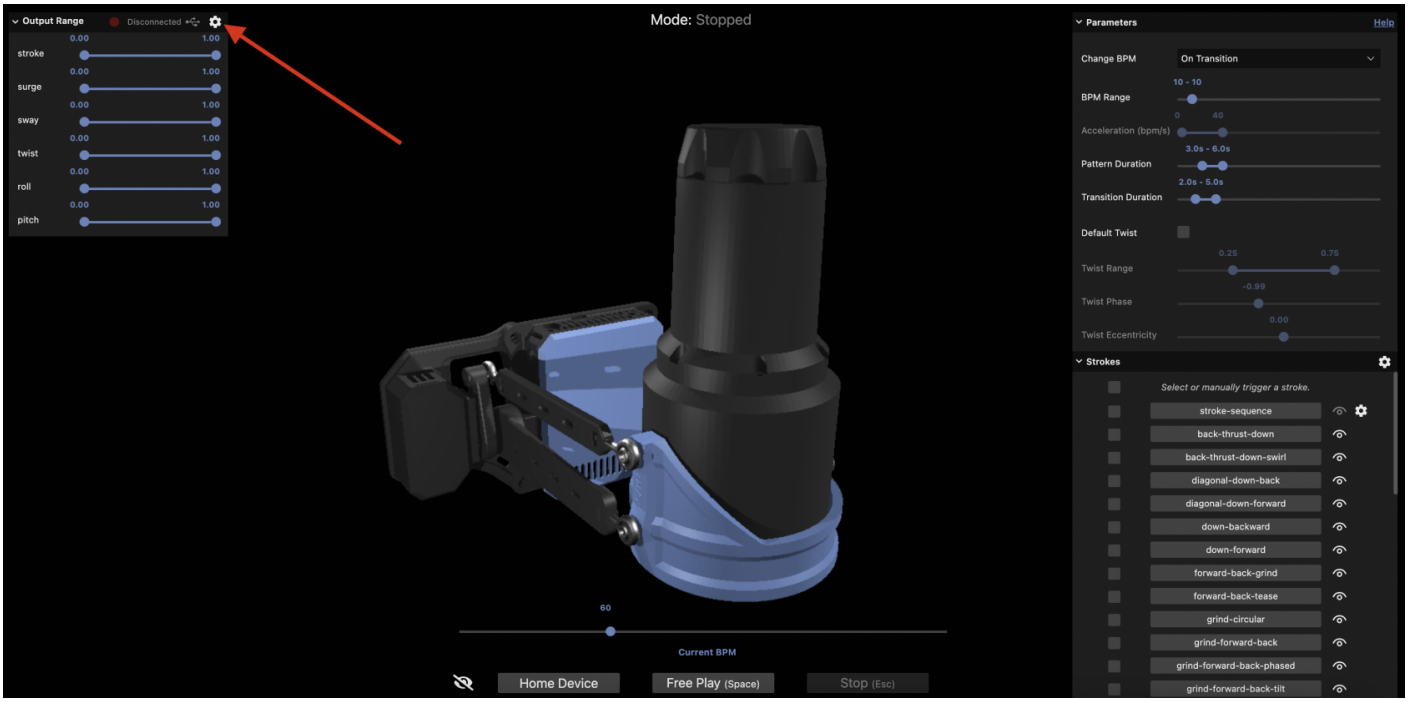


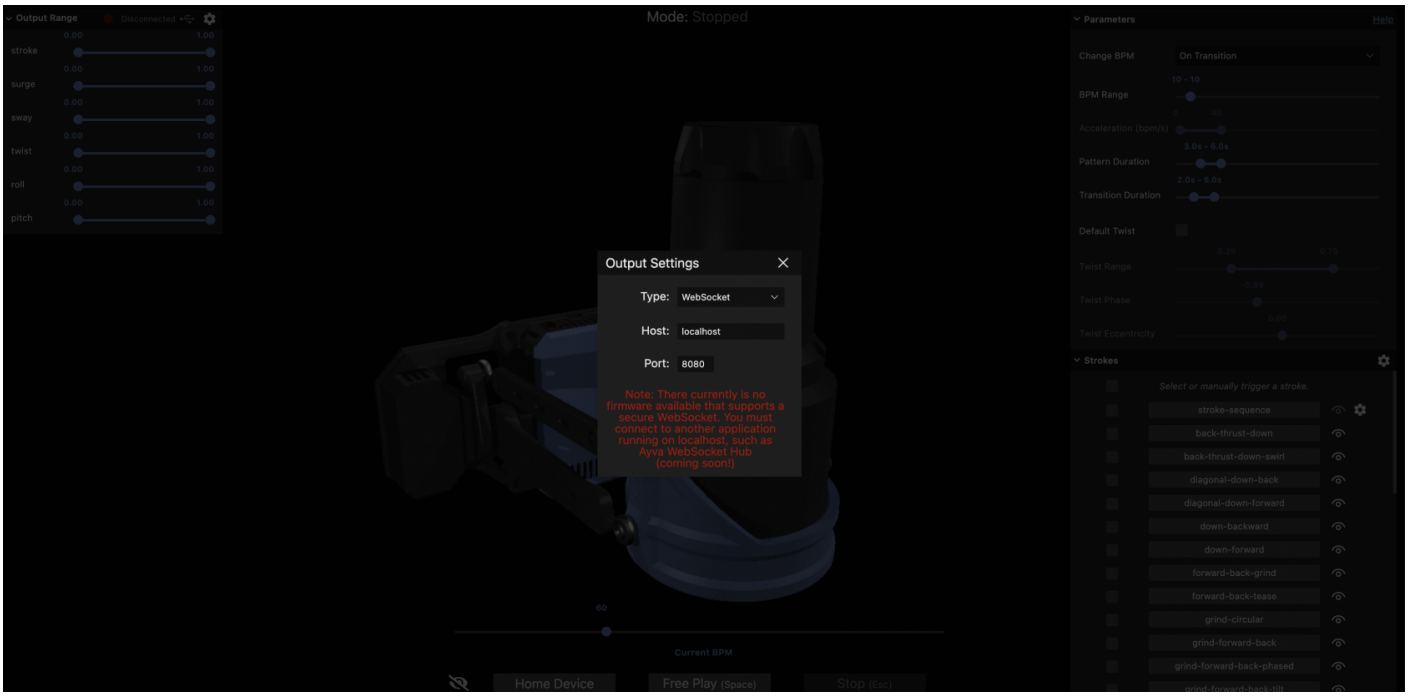

Avva WebSocket Hub
Port: 8080
Add Output
Enabled
Disconnected
Network Output
Type: UDP
Host: localhost
Port: 8889
Add


Session Plugins
Plugin
Origin
Save Config
Load Config
As Default
Motion Target
Enable Debug
Select motion target
Physics Link
Range
Values Source
Select Person
Girl
Select Target
headControl
Reset origin
Refresh
Select values source
Udp
Port
8889
Start
Stop
Close
Scene Audio
Scene Misc
Scene Plugins
Scene Animation
Scene Plugin Presets
Scene Lighting
Scene Add Atom
Session Plugins
Session Plugin Presets
Select
Navigation
User Preferences
File (Open/Save)
Version: 1.21.1.0
Freeze Motion/Sound
Click for more options
Play Mode (P)
Edit Mode (E)
Head Possessing
Guy:headControl
Allow Mouse Pan and Zoom



Example: Streaming from Ayva Stroker Lite




Output Range
Disconnected
stroke
surge
sway
twist
roll
pitch
Mode: Stopped
Output Settings
Type:
WebSocket
Host:
localhost
Port:
8080
Note: There currently is no firmware available that supports a secure WebSocket. You must connect to another application running on localhost, such as Ayva WebSocket Hub (coming soon!)
Current BPM
Home Device
Free Play (Space)
Parameters
Help
Change BPM
On Transition
BPM Range
Acceleration (bpm/s)
Pattern Duration
Transition Duration
Default Twist
Twist Range
Twist Phase
Twist Eccentricity
Strokes
Select or manually trigger a stroke.
stroke-sequence
back-thrust-down
back-thrust-down-swirl
diagonal-down-back
diagonal-down-forward
down-backward
down-forward
forward-back-grind
forward-back-tease
grind-circular
grind-forward-back
grind-forward-back-phased

Troubleshooting

I see this error in BusDriver. What's wrong?

`Exception caught: System.FormatException: Input string was not in the correct format ...`

This is from a bug that was fixed in BusDriver. Download and install the latest version and it will go away.