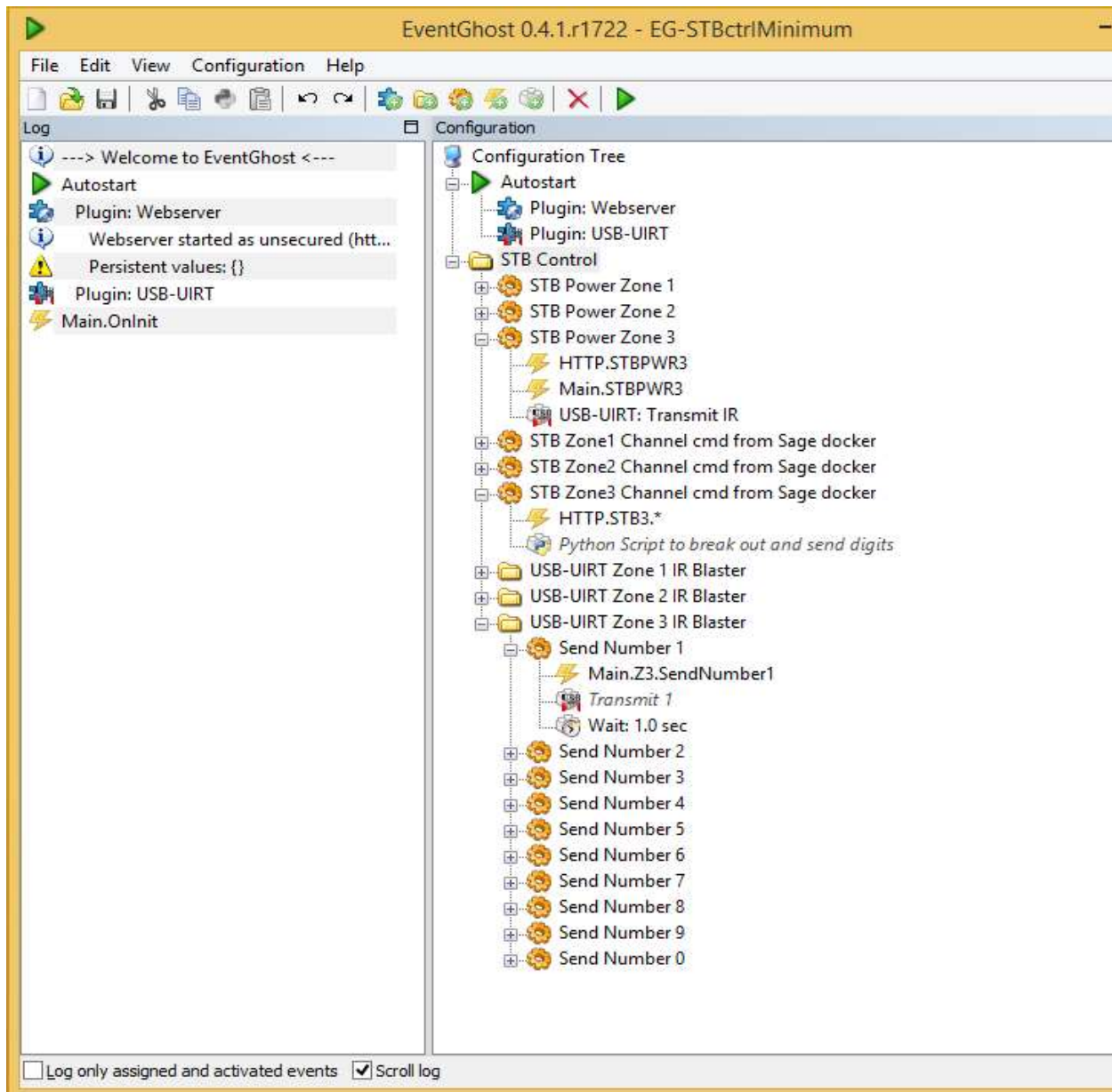


## Using EventGhost running in Windows (VM) for USB-UIRT Multi-zone support with Linux

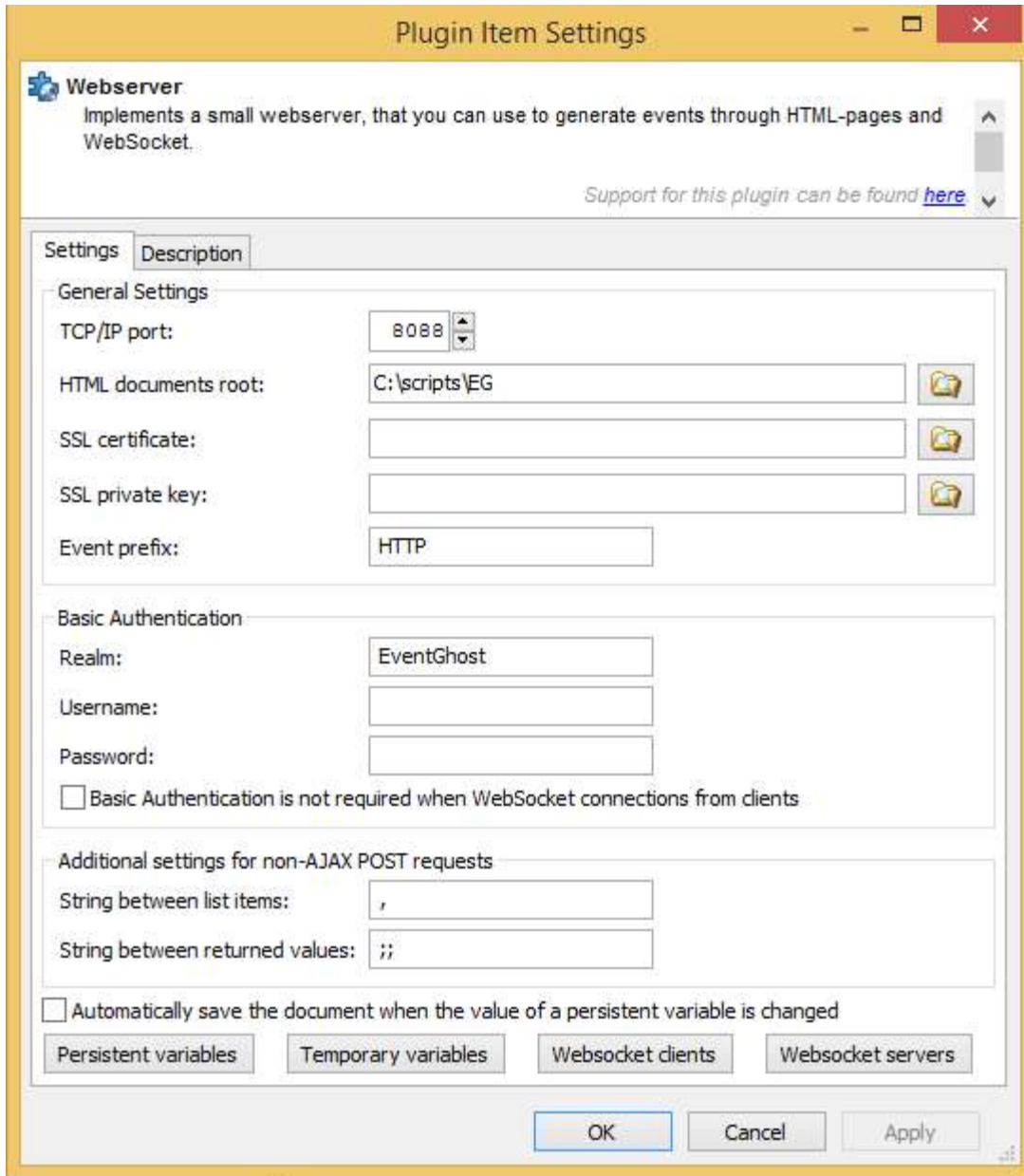
I recently switched my server to an UnRaid docker and still needed my USB-UIRT to support multiple zones to control multiple set top boxes, which the Linux driver doesn't support. So, I used my favorite utility program EventGhost on a Windows VM to handle it for me. Here is how to do it along with the files needed.

Note that UnRaid is not needed. This will run on any Windows system that can control the USB-UIRT.

1. On your Windows system (in my case a Win 8.1 VM on my UnRaid server that also serves as a SageTV network encoder with an HDPVR 2), **download and install EventGhost** from <http://www.eventghost.net/>. I use the 0.4 R1722 release but this should work fine on the 0.5 pre-release. **Don't run it yet though.**
2. **Install the USB-UIRT driver** found at <http://www.usbuirt.com/support.htm> and plug in the USB-UIRT. Plug in the IR emitter into the USB-UIRT that you should already have if you want to use multiple zones.
3. **Unzip the attached files** into an easy to find folder on the Windows system (C:\scripts\EG) in my case.
4. **Start EventGhost** (EG from here on), and if it minimizes then find the icon in the system tray (lower right) and double-click it to expand it.
5. Open the **EG-STBctrlMinimum.xml** file in your easy to find folder, which is an EG configuration file. Ignore any errors you may get for now. You should see something like this:

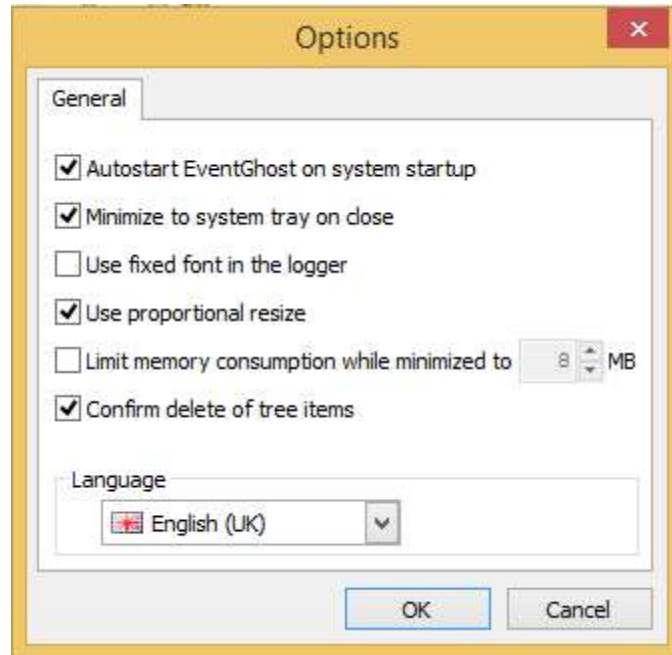


6. In the Autostart section look for **Plugin: Webserver** and double-click it. A configuration box will open. Set the HTML documents root folder to your easy to find folder, and set the TCP/IP port to your desired port or leave it at 8088.



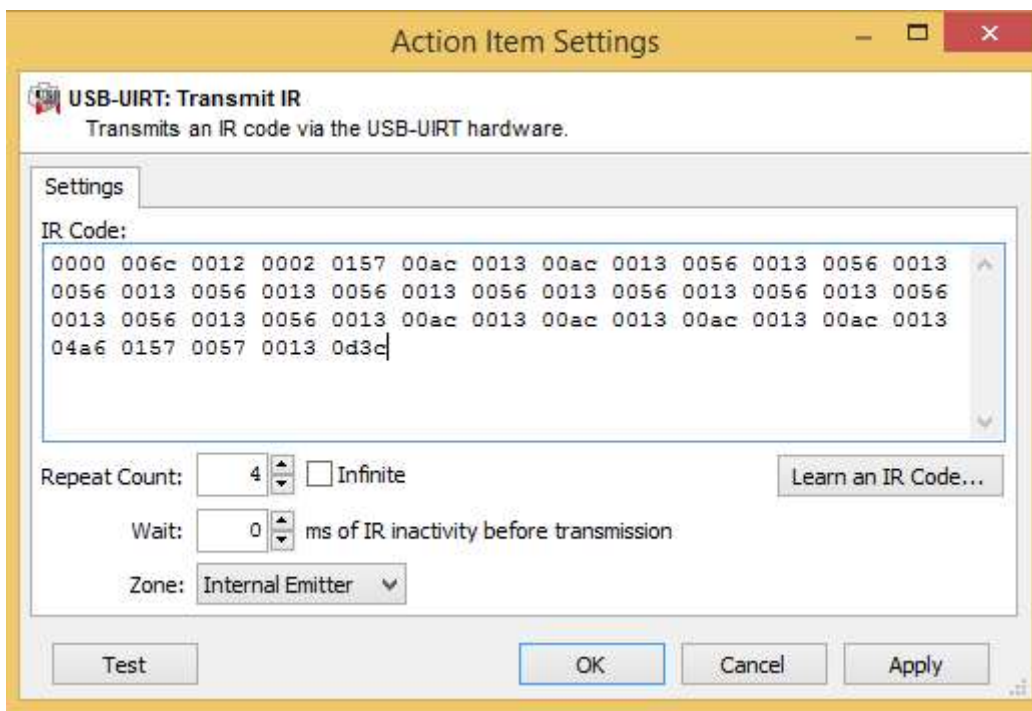
NOTE: You can do a lot of things with the webserver, and examples can be found on the EG forum. For now though all we have is a simple "hello" web page if anyone actually looks at it.

7. In the **EG Options**, be sure to set "Autostart EG on system startup" and you might want to check "Minimize to system tray on close" as well.



8. **Here is the only tricky part.** My config file knows IR commands for an Arris/Motorola STB. I had a Samsung STB for a while that also used this command set. If things don't work, then you'll need to change the IR commands for each number and optionally the Power command if you want to turn the power off/on remotely.

To change the digits, look under the "USB-UIRT Zone X IR Blaster" folders and then under the "Send Number X" macros you will see a command such as "Transmit 1". Double-click that to get to the setup options.



Now use the "Learn an IR code" button to teach EG the code for that digit using your remote control and the USB-UIRT. After the new code is learned you can use the Test button to try it out. Once you're done, Apply and OK to finish. Now do the other nine numbers in that folder, and then if you need to teach the codes for the other STB/Zones as well.

9. **File->SAVE** the EG configuration file or you'll lose all those changes you just made.

10. **Copy the "gentuner" file** from the zip file to your SageTV server folder on the Linux machine or docker and open it with a file editor.

```
IP="192.168.1.5:8088"
CMD=$1
REMOTE=$2
KEY=$3
CHANNEL=$3

if [ "$CMD" = "REMOTES" ]; then
    echo "STB1"
    echo "STB2"
    echo "STB3"
elif [ "$CMD" = "KEYS" ]; then
    echo "not required"
elif [ "$CMD" = "SEND" ]; then
    echo "not required"
    exit 1
elif [ "$CMD" = "TUNE" ]; then
    wget "http://\$IP/index.htm?\$REMOTE.\$CHANNEL" > /dev/null 2>&1
```

```
elif [ "$CMD" = "CAN_TUNE" ]; then
    echo "OK"
else
    exit 1
fi
```

**Change the IP variable** to your EG/Windows system address and the port to whatever you set the EG webserver port to. Save your changes. **Make sure the file is executable** (chmod 777 gentuner).

If you look in this file, you will see that the command that gets sent to EG looks like

[http://\\$IP/index.htm?\\$REMOTE.\\$CHANNEL](http://$IP/index.htm?$REMOTE.$CHANNEL)

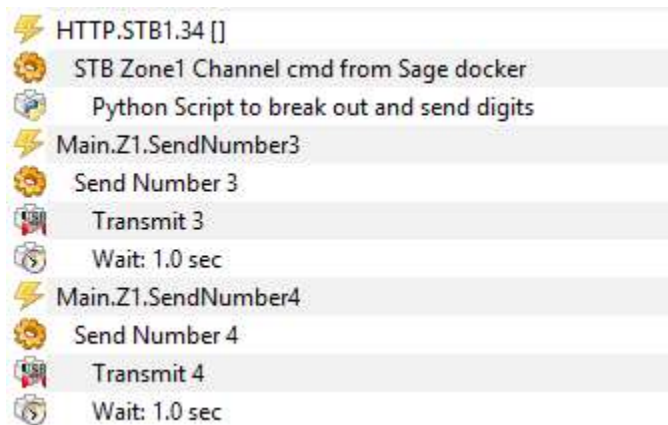
where \$IP is the IP address and port, \$REMOTE is STB1, STB2 or STB3 appropriately, and \$CHANNEL is the channel.

wget is used to send the command from a script. If for some reason your Linux system doesn't have wget installed then you'll need to install it.

11. At this point, you should be able to test it out by running a test command in Linux such as

**./gentuner TUNE STB1 34**

If it works then you'll see messages in the log window of EG showing a command received such as "HTTP.STB1.34[]" followed by a Python script that breaks out each digit and ultimately transmits the code for each digit.



If you poke around the Configuration Tree some you should be able to see what is going on.

Let me know if you have any questions or problems.

BTW EventGhost is an extremely powerful program and worth a look if you want to automate functions or need a way to get program A to interface to program B, interface with hardware, or even send messages from one system to another. This example doesn't even scratch the surface of what it can do.