

## Topic 2.7: Enabling External Access to your Web Server – MacOS


**Note 1:** If other computers are unable to access the server on your computer after following this procedure while using the school network, it maybe because of the reason stated in *Step 6: Router/AP Isolation*. You might wish to try it on your home network if you can't get it working on the school network.

**Note 2:** Remember that you will be allowing other computers to access your web server by turning off default security settings. If you no longer wish to allow other computers to have access to your web server, you should re-enable the security.

Unlike Windows (which lets you open a specific port number), macOS's built-in firewall is application-based. It asks: "Should this specific app (e.g., `python3`, `node`, `httpd`) accept incoming connections?"

So, you do not "open port 8000." Instead, you allow your web server application to receive incoming traffic. Once the app is allowed, its designated port (8000, 80, or any other) becomes accessible.

### Step 1: Check if the Firewall is Even On

1. Go to **Apple Menu**  > **System Settings**.
2. Click **Network** in the left sidebar.
3. Click **Firewall**.
4. If it says "**Firewall: Off**", you can skip to Step 3. If it is "**On**", proceed to Step 2.

### Step 2: Allow Your Web Server Application

When you run your web server for the first time while the firewall is on, macOS usually pops up a dialog saying: "Do you want the application 'python' to accept incoming network connections?" You should click "**Allow**".

If you missed that prompt, or you want to add an application manually:

1. Go to **System Settings** > **Network** > **Firewall**.
2. Click the **Options...** button.
3. Click the + (plus) button under the list of applications.
4. Navigate to your web server application. For example:
  - **Python** – `/usr/bin/python3`` or `/usr/local/bin/python3``
5. Select the app and click **Add**.
6. Ensure the setting for that app is "**Allow incoming connections**".
7. Click **OK**.

### Step 3: Disable "Stealth Mode" (Crucial!)

While you are in the Firewall Options, look for a checkbox labeled:

> "**Enable stealth mode**"

If this is checked, it means your computer will not respond to any unsolicited probes from other machines. This effectively makes your web server invisible to local network devices.

**Solution:** Uncheck "**Enable stealth mode**" and click OK. This allows other computers to actively see and connect to your server.

## Step 4: The Special Case of Port 80

Here is a critical technical detail:

- Port 8000 (or any port > 1024): You can run your web server as a normal user (without `sudo`). The firewall will prompt you to allow the app.
- Port 80 (or any port < 1024): These are "privileged ports" on Unix/macOS. Only the root user can bind to them.
- You **must** start your server with `sudo`, e.g., `sudo python3 -m http.server 80`.
- Because you are running it as `root`, the firewall may block the connection by default for security reasons.
- **Solution:** You have two choices:
  1. Use port 8000 instead (much easier).
  2. If you must use port 80, you may need to temporarily disable the firewall entirely (`sudo /usr/libexec/ApplicationFirewall/socketfilterfw --setglobalstate off`) – **not recommended** for security. The better way is to use **port forwarding** (e.g., forward port 80 to port 8000 using `pfctl`), but that is advanced.

## Step 5: Find Your Mac's Local IP Address

Other computers need to know which computer to connect to. They cannot use `localhost`.

1. Open **Terminal**.
2. Type: `ifconfig | grep inet` or `ipconfig getifaddr en0` (for Wi-Fi) or `en1` (for Ethernet).
3. Look for an IP address like **192.168.1.x** or **10.0.0.x**. This is your Mac's local IP.
4. On another computer, open a browser and type: `http://192.168.1.x:8000` (replace with your actual IP and port).

## Step 6: Troubleshooting – Still Can't Connect?

If you have followed all the steps and it still does not work, check these common issues:

1. Server is binding to `127.0.0.1` (localhost) only.
  - Many servers default to `127.0.0.1`, which means "listen only to my own computer."
  - Fix: Start your server with `0.0.0.0` (listen on all network interfaces).
  - For Python: change the config.HOST to from `localhost` to `0.0.0.0`
2. **Router/AP Isolation** (rare on home networks).
  - Some public or guest Wi-Fi networks isolate connected devices. Check if your Mac and the other device are on the same subnet.
3. Third-party firewall software.
  - If you have Little Snitch, Lulu, or other firewall tools, they may override macOS settings. Check their rules
4. Check if the port is actually listening.
  - In Terminal, run: `sudo lsof -i :8000`
  - This should show your server process in the list with a state of `LISTEN`. If it shows `127.0.0.1`, the server is not binding to `0.0.0.0`.