

Testing Mac mini server

To use a Mac mini as a web server, you can leverage macOS's built-in web server functionality or install a third-party web server like Apache. macOS includes "Personal Web Sharing," which allows you to host a basic website directly from your Mac mini's Documents folder. Alternatively, you can install Apache using tools like Homebrew.

I want to use a LEMP tech stack because that is what the company i work for uses.

installing homebrew:

in my root directory i ran:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

output from terminal:

```
1 ==> Checking for `sudo` access (which may request your password)...
2 Password:
3 Sorry, try again.
4 Password:
5 ==> This script will install:
6 /opt/homebrew/bin/brew
7 /opt/homebrew/share/doc/homebrew
8 /opt/homebrew/share/man/man1/brew.1
9 /opt/homebrew/share/zsh/site-functions/_brew
10 /opt/homebrew/etc/bash_completion.d/brew
11 /opt/homebrew
12
13 Press RETURN/ENTER to continue or any other key to abort:
14 ==> /usr/bin/sudo /usr/sbin/chown -R avadennis:admin /opt/homebrew
15 ==> /usr/bin/sudo /bin/mkdir -p /Users/avadennis/Library/Caches/Homebrew
16 ==> /usr/bin/sudo /bin/chmod g+rxw /Users/avadennis/Library/Caches/Homebrew
17 ==> /usr/bin/sudo /usr/sbin/chown -R avadennis /Users/avadennis/Library/Caches/Homebrew
18 ==> Downloading and installing Homebrew...
19 remote: Enumerating objects: 7815, done.
20 remote: Counting objects: 100% (3494/3494), done.
21 remote: Compressing objects: 100% (178/178), done.
22 remote: Total 7815 (delta 3442), reused 3316 (delta 3316), pack-reused 4321 (from 5)
23 ==> Updating Homebrew...
24 ==> Downloading https://ghcr.io/v2/homebrew/portable-ruby/portable-
ruby/blobs/sha256:40e7f5d7514a7e9757facdd39006f7a351d3d7986d3a228be13c8b1c3216727b
25 ##### 100.0%
26 ==> Pouring portable-ruby-3.4.3.arm64_big_sur.bottle.tar.gz
27 ==> Installation successful!
28
29 ==> Homebrew has enabled anonymous aggregate formulae and cask analytics.
30 Read the analytics documentation (and how to opt-out) here:
31   https://docs.brew.sh/Analytics
32 No analytics data has been sent yet (nor will any be during this install run).
33
34 ==> Homebrew is run entirely by unpaid volunteers. Please consider donating:
35   https://github.com/Homebrew/brew#donations
36
37 ==> Next steps:
```

```
38 - Run brew help to get started
39 - Further documentation:
40   https://docs.brew.sh
```

Then i needed to install nginx:

```
i ran brew install nginx
```

there was a lot of output but here is some possible important parts I grabbed from it

```
1 Docroot is: /opt/homebrew/var/www
2
3 The default port has been set in /opt/homebrew/etc/nginx/nginx.conf to 8080 so that
4 nginx can run without sudo.
5
6 nginx will load all files in /opt/homebrew/etc/nginx/servers/.
7
8 To start nginx now and restart at login:
9   brew services start nginx
10 Or, if you don't want/need a background service you can just run:
11   /opt/homebrew/opt/nginx/bin/nginx -g daemon\ off\;
12 ==> Summary
13 🍺 /opt/homebrew/Cellar/nginx/1.27.5: 27 files, 2.5MB
14 ==> Running `brew cleanup nginx`...
15 Disable this behaviour by setting HOMEBREW_NO_INSTALL_CLEANUP.
16 Hide these hints with HOMEBREW_NO_ENV_HINTS (see `man brew`).
17 ==> Upgrading 15 dependents of upgraded formulae:
18 Disable this behaviour by setting HOMEBREW_NO_INSTALLED_DEPENDENTS_CHECK.
19 Hide these hints with HOMEBREW_NO_ENV_HINTS (see `man brew`).
20 git 2.48.0 -> 2.49.0, openssl@1.1 1.1.1u -> 1.1.1w, openssl@3 3.4.1 -> 3.5.0, folly 2025.02.17.00 ->
2025.04.14.00, fizz 2025.02.17.00 -> 2025.04.14.00, libfido2 1.15.0 -> 1.15.0_1, mysql 9.2.0 -> 9.3.0, node
23.7.0 -> 23.11.0, python@3.13 3.13.2 -> 3.13.3, ruby 3.4.2 -> 3.4.3, wangle 2025.02.17.00 -> 2025.04.14.00,
edencommon 2025.02.17.00 -> 2025.04.14.00, fbthrift 2025.02.17.00 -> 2025.04.14.00, fb303 2025.02.17.00 ->
2025.04.14.00, watchman 2025.02.17.00 -> 2025.04.14.00
21
22 ...
23
24 Upgrading from MySQL <8.4 to MySQL >9.0 requires running MySQL 8.4 first:
25 - brew services stop mysql
26 - brew install mysql@8.4
27 - brew services start mysql@8.4
28 - brew services stop mysql@8.4
29 - brew services start mysql
30
31 We've installed your MySQL database without a root password. To secure it run:
32   mysql_secure_installation
33
34 MySQL is configured to only allow connections from localhost by default
35
36 To connect run:
37   mysql -u root
38
39 To start mysql now and restart at login:
40   brew services start mysql
41 Or, if you don't want/need a background service you can just run:
42   /opt/homebrew/opt/mysql/bin/mysqld_safe --datadir=/opt/homebrew/var/mysql
43 ==> Summary
44 🍺 /opt/homebrew/Cellar/mysql/9.3.0: 333 files, 291.9MB
```

```

45 ==> Running `brew cleanup mysql`...
46 Removing: /opt/homebrew/Cellar/mysql/9.2.0... (329 files, 286.3MB)
47
48 ...
49
50 The Tcl/Tk GUIs (e.g. gitk, git-gui) are now in the `git-gui` formula.
51 Subversion interoperability (git-svn) is now in the `git-svn` formula.
52
53 zsh completions and functions have been installed to:
54   /opt/homebrew/share/zsh/site-functions
55
56 ...
57
58 Docroot is: /opt/homebrew/var/www
59
60 The default port has been set in /opt/homebrew/etc/nginx/nginx.conf to 8080 so that
61 nginx can run without sudo.
62
63 nginx will load all files in /opt/homebrew/etc/nginx/servers/.
64
65 To start nginx now and restart at login:
66   brew services start nginx
67 Or, if you don't want/need a background service you can just run:
68   /opt/homebrew/opt/nginx/bin/nginx -g daemon\ off\;
69 ==> mysql
70 Upgrading from MySQL <8.4 to MySQL >9.0 requires running MySQL 8.4 first:
71   - brew services stop mysql
72   - brew install mysql@8.4
73   - brew services start mysql@8.4
74   - brew services stop mysql@8.4
75   - brew services start mysql
76
77 We've installed your MySQL database without a root password. To secure it run:
78   mysql_secure_installation
79
80 MySQL is configured to only allow connections from localhost by default
81
82 To connect run:
83   mysql -u root
84
85 To start mysql now and restart at login:
86   brew services start mysql
87 Or, if you don't want/need a background service you can just run:
88   /opt/homebrew/opt/mysql/bin/mysqld_safe --datadir=/opt/homebrew/var/mysql
89 ==> git
90 The Tcl/Tk GUIs (e.g. gitk, git-gui) are now in the `git-gui` formula.
91 Subversion interoperability (git-svn) is now in the `git-svn` formula.
92
93 zsh completions and functions have been installed to:
94   /opt/homebrew/share/zsh/site-functions

```

To start Nginx:

I used `brew services start nginx`

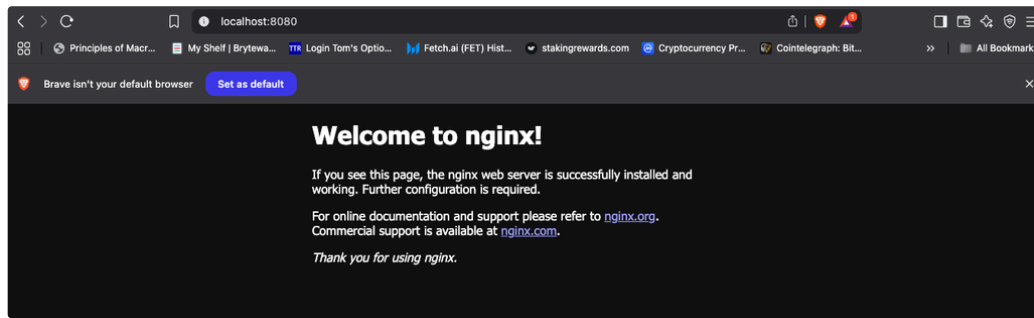
output:

```

1 ==> Successfully started `nginx` (label: homebrew.mxcl.nginx)

```

so now you can access it in a browser: using this as the url: <http://localhost:8080>



next I installed PHP:

```
brew install php
```

the output was a lot of downloads and it also gives you directions on how to enable php in apache and i think that is because apache is native to a mac system

checking what php version i am running using `php -v`

and it shows I am running 8.4.6:

```
1 PHP 8.4.6 (cli) (built: Apr  8 2025 19:55:31) (NTS)
2 Copyright (c) The PHP Group
3 Built by Homebrew
4 Zend Engine v4.4.6, Copyright (c) Zend Technologies
5     with Zend OPcache v8.4.6, Copyright (c), by Zend Technologies
```

Homebrew's PHP will run on port **9000** for PHP-FPM by default.

now i need to configure nginx to run php. This means setting up **Nginx to pass .php files to PHP-FPM** so that your server can actually **run PHP scripts**, not just serve plain `.html` files.

🧠 Why This Matters 🔗

Nginx is a **web server**, but it **doesn't process PHP code itself**. Instead, it sends `.php` files to **PHP-FPM (FastCGI Process Manager)**, which interprets the PHP and returns the resulting HTML back to Nginx, which then sends it to your browser.

Without this configuration, Nginx will just download PHP files or give errors instead of executing them.

so now i am going to do this by changing stuff in the conf file

i ran: `nano /opt/homebrew/etc/nginx/nginx.conf`

and that opened the conf file in an editable mode using nano

and currently my default server looks like this:

```
1  server {
2      listen      8080;
3      server_name localhost;
4
5      #charset koi8-r;
6
7      #access_log logs/host.access.log main;
8
9      location / {
10         root    html;
11         index  index.html index.htm;
```

```
12     }
```

chatGPT says I should update it to look like this

```
1 server {
2     listen 8080;
3     server_name localhost;
4
5     root /opt/homebrew/var/www;
6     index index.php index.html;
7
8     location / {
9         try_files $uri $uri/ =404;
10    }
11
12    location ~ /\.php$ {
13        include fastcgi_params;
14        fastcgi_pass 127.0.0.1:9000;
15        fastcgi_index index.php;
16        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
17    }
18 }
19
```

then i did

- Press `Control + O` to **save the file**.

(In the nano text editor, "**write out**" is used to mean "**save my changes to disk**." To save your file in nano, you select Ctrl+O)

- Press `Enter` to confirm.
- Press `Control + X` to **exit nano**.

then i restarted nginx: `brew services restart nginx`

I made an error that i found out about using `nginx -t`

here was the error

```
1
2 nginx: [emerg] "server" directive is not allowed here in /opt/homebrew/etc/nginx/nginx.conf:47
3 nginx: configuration file /opt/homebrew/etc/nginx/nginx.conf test failed
```

i didn't delete the previous server code so i went to the file and deleted it.

I made another mistake:

```
1 nginx: [emerg] "location" directive is not allowed here in /opt/homebrew/etc/nginx/nginx.conf:59
2 nginx: configuration file /opt/homebrew/etc/nginx/nginx.conf test failed
```

i looked at the code and saw i had an extra bracket

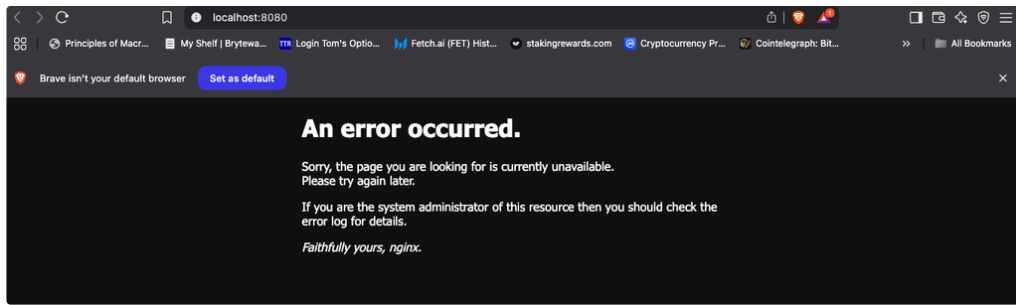
Now nginx is working as you can see with running again: `nginx -t`

```
1 nginx: the configuration file /opt/homebrew/etc/nginx/nginx.conf syntax is ok
2 nginx: configuration file /opt/homebrew/etc/nginx/nginx.conf test is successful
```

but php is not working still with using

```
echo "<?php phpinfo(); ?>" > /opt/homebrew/var/www/index.php
```

cause i get this error:



chatgpt suggested i check if Check if PHP-FPM is running and this was my output after running: `brew services list`

```
1 Name Status User File
2 mysql none
3 nginx started avadennis ~/Library/LaunchAgents/homebrew.mxcl.nginx.plist
4 php none
```

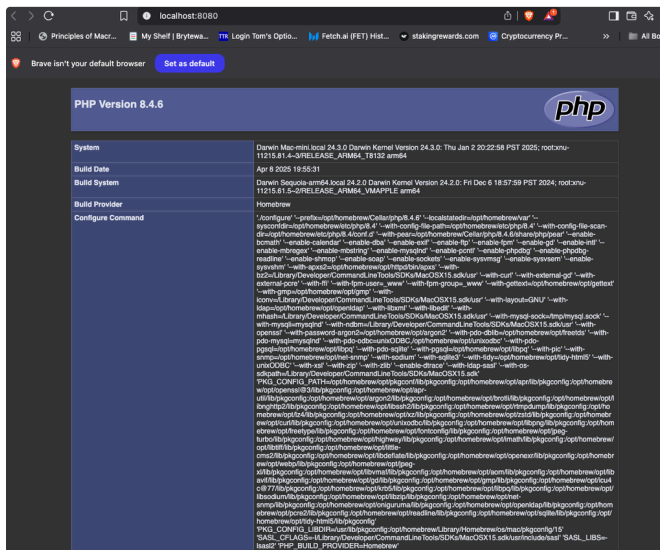
so That means **PHP-FPM is not running**, so Nginx has no way to process PHP files — which is exactly why you're seeing the "unavailable" error.

so i tried try starting the PHP service using: `brew services start php`

and output was:

```
1 ==> Successfully started `php` (label: homebrew.mxcl.php)
```

now i am able to see the info page!



That means my Nginx server is correctly routing PHP through PHP-FPM — my **LEMP stack is working!**

so here is currently whats working: [🔗](#)

- Nginx is serving content on <http://localhost:8080>
- PHP-FPM is processing `.php` files via a Unix socket
- Your server is running locally on your Mac Mini

now for the database:

I ran `brew install mysql` and here was the output:

```
1 ==> Downloading https://formulae.brew.sh/api/formula.jws.json
2 ==> Downloading https://formulae.brew.sh/api/cask.jws.json
```

```
3 Warning: mysql 9.3.0 is already installed and up-to-date.
4 To reinstall 9.3.0, run:
5   brew reinstall mysql
```

now I started the mysql services by running: `brew services start mysql`

here was the output:

```
1 ==> Successfully started `mysql` (label: homebrew.mxcl.mysql)
```

i remembered this `brew services list` command checked what was running so i ran it again.

This was the output:

```
1 Name  Status  User      File
2 mysql stopped avadennis ~/Library/LaunchAgents/homebrew.mxcl.mysql.plist
3 nginx started avadennis ~/Library/LaunchAgents/homebrew.mxcl.nginx.plist
4 php   started avadennis ~/Library/LaunchAgents/homebrew.mxcl.php.plist
```

Now for hosting online: [🔗](#)

Run this in Terminal:

```
ipconfig getifaddr en0
```

ERROR [🔗](#)

and i got nothing in return.

What I found out is the `en0` has a `status: inactive` line which means that this particular network interface (likely `en0`) is **not connected**. So your Mac is using a **different interface** for the internet connection.

So that above command did not work so I ran another thing it suggested instead which was this:

```
ifconfig | grep "inet " | grep -v 127.0.0.1
```

My output was this:

```
1      inet 10.0.0.168 netmask 0xffffffff00 broadcast 10.0.0.255
```

That means your Mac mini's **local IP address is:** `10.0.0.168` — it's connected to the network

now I need to Set Up Port Forwarding on my Router

I tried running going to these by opening a browser and go to: `http://10.0.0.1` (or sometimes `http://192.168.0.1`)

ERROR [🔗](#)

I was not able to access them. That suggests your **router might use a different IP** for its admin panel, or you're behind a **hotspot or managed network** that blocks router access

so i ran this in the terminal:

```
netstat -nr | grep default
```

and my output was this:

```
1 default          10.0.0.1          UGScg            en1
```

so this confirms your **router's IP address is** `10.0.0.1`.

which means it is either being blocked or a browser problem.

ERROR [🔗](#)

I know i have the correct IP address and so what I will test is using a different browser

I have been working on safari but i tried putting the url into chrome.

The chrome browser worked! Safari can be tricky sometimes so I don't recommend it for testing.

ERROR [🔗](#)

I could not login because I did not have the proper credentials

I looked it up and it turns out that it is default to `admin` and `password` so i tried that and it worked and then i changed the password. Now I am in!

ERROR [🔗](#)

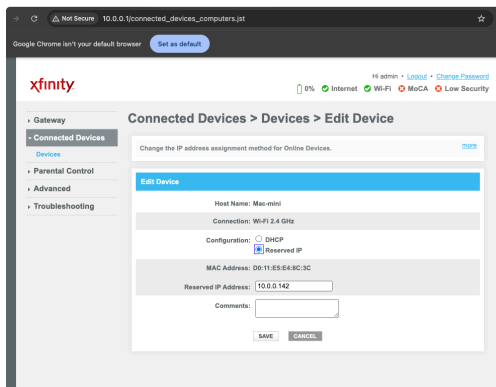
There is no port forwarding option and I dont know what to do. Instructions say something about a gateway so I set up the Gateway which doesnt make sense because it is already set up.

However that actually game me more options to choose from in the app and it game me an option to do port forwarding. In the nav bar `wifi` and `security` were not options. Now they exists...odd

ERROR [🔗](#)

i found out IP address is not static but dyanmic

I went to xfinty and changed it to reserve IP



Then i went to my phone an added the mac mini to port forwarding

ERROR [🔗](#)

the IP address dont match: `ifconfig | grep inet` this shows 10.0.0.168 and the app shows 10.0.0.168 but xfinity admin side shows 10.0.0.142

Turns out I added the Mac not the mac mini so i fixed that to add the mac mini on xfinity

ERROR [🔗](#)

this became an hour long process of confusion because on the xfinity site there is a mac-mini that says it was offline and then a mac. And the Mac was using the IP address I needed so I was like, what is going on?!?! Turns out my mac-mini is actually `Mac` on the xfinity site so it was correct the whole time

34:25:be:68:a3:db	DHCP	-67 dBm	GHz	EDIT
Mac	Reserved IP	undefined dBm	Wi-Fi 5 GHz	EDIT
LivingRoom	DHCP	undefined dBm	Wi-Fi 5 GHz	EDIT
ADD DEVICE WITH RESERVED IP				
Offline Devices				
Host Name	DHCP/Reserved IP	Connection		
Mac-mini	Reserved IP	Wi-Fi 5 GHz	EDIT	

now i just need to put int the url for the site and get it up and running

ERROR [🔗](#)

The url for using the IP 4 address is not working as suggested : 73.169.222.111

but it did say you might have to use the ip4 address that xfinity give you which is this:

10.0.0.168 which worked.

OMG I DID IT!!!!!!

try it out!

<http://10.0.0.168/>

ERROR [🔗](#)

lol jk that is for when you are connected to the wifi here.

Try this though, this, it is my **public IPv4 address**, which means anyone on the internet can **potentially** access your Mac Mini's server.

<http://73.169.222.111>