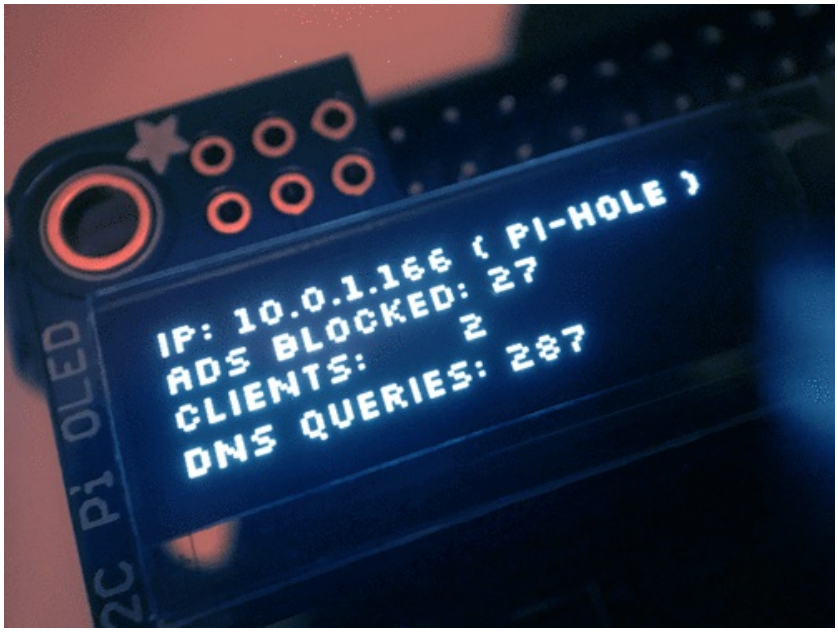
adafruit learning system

Pi Hole Ad Blocker with Pi Zero W

Created by lady ada



Last updated on 2017-08-16 04:03:33 AM UTC

Guide Contents

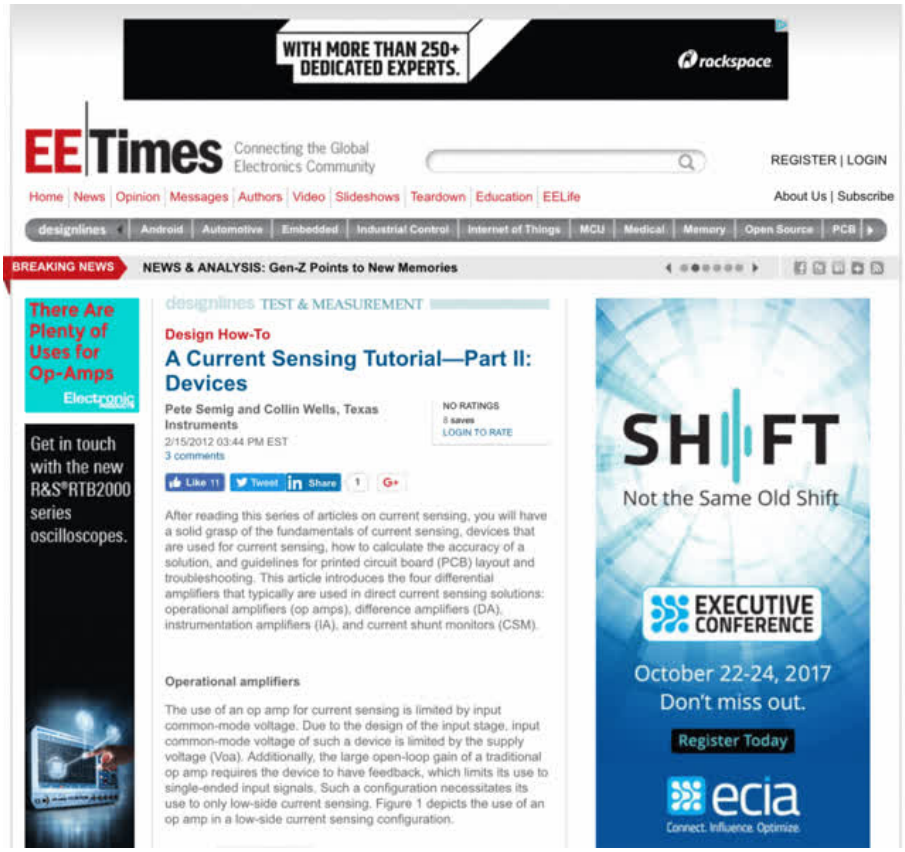
Overview

A long time ago we made a Pi into a WiFi gateway that also blocked ads but the Pi Hole project does a*way* better job!

This project will make your Pi Zero W act as a DNS (**D**omain **N**ame **S**erver) The kind of device that tells you that**adafruit.com** is known as IP address 104.20.38.240.

Except Pi Hole DNS will do a special trick, when it is asked for the IP address of ads.adserver.com (for example) it will return nothing! So you will never even connect to the ad server and get the ad. Your connection will be faster, less data, and no intrusive ads. It works great on computers, tablets, phones, etc. Even if you cannot run an ad-blocker plugin on your phone or tablet, this will work and ad-blocker-detectors can't tell you're running it.

Unlike our WiFi gateway demo, you do not have to set up the Pi as your access point, you will only use it as a DNS ad blocker so it will not act as a bottleneck



We upgraded our Pi Zero Pi Hole with a little display, that makes setting up clients easy and also gives you some nifty stats!

Follow along with this guide to DIY your own

Project Parts

This project can be done with any Raspberry Pi, but for the most adorably compact version we're using a Pi Zero W - this has enough power to do what we want, and has built in WiFi too!

Pi Zero W base parts

Its easiest if you pick up a Pi Zero W budget pack as it contains most everything you need

Raspberry Pi Zero W Budget Pack - Includes Pi Zero W

PRODUCT ID: 3410
Note: Due to popular demand, there might be some delay in shipping products containing Pi Zero W! Remember those cereal commercials that would always say, "part of a complete...
http://adafru.it/yuB
$34.50
IN STOCK

But you can also just DIY with the minimum requirements:

1 x Pi Zero W
the type of low cost game-changing product Raspberry Pi's known for - the super light, super lean microcomputer we've come to know and low, but now with built-in WiFi.
Add To Cart
1 x 4G or larger SD Card
You will be burning this card with Raspbian Jessie Lite so its ok if its blank or pre-burned
Out of Stock
Notify Me
1 x Adafruit Pi Zero Enclosure
Adafruit's classic, sturdy plastic enclosure. Keeps your Pi Zero safe and sleek.
Add To Cart
1 x 5V 1A USB wall adapter
This one is plenty good and you can use any Micro USB cable with it
Add To Cart
1 x 5V 2.4A USB wall adapter
Super powerful for any uses, and comes with a built in MicroUSB cable
Out of Stock
Notify Me

Pi OLED Display Addition

If you want to add an OLED display (which is suggested!) you'll also need:

Adafruit PiOLED - 128x32 Monochrome OLED Add-on for Raspberry Pi

PRODUCT ID: 3527
If you're looking for the most compact li'l display for a Raspberry Pi (most likely a Pi Zero) project, this might be just the thing you need! The Adafruit 128x32 PiOLED...
http://adafru.it/wVd
$14.95
IN STOCK

If you are using a Pi Zero W you'll need to add 2x20 headers too

1 x 2x20 Male Header
Solder this in to plug in Pi HATs, GPIO cables, etc as you would into a normal Pi. Requires soldering
Add To Cart

or

1 x 2x20 No-Solder Hammer Headers
If your soldering isn't quite up to scratch, or you just don't own a soldering iron yet, then these nifty hammer headers from Pimoroni might be just what you need.
Add To Cart

Other things you may need...

You also need a way to burn that SD card!

USB MicroSD Card Reader/Writer - microSD / microSDHC / microSDXC

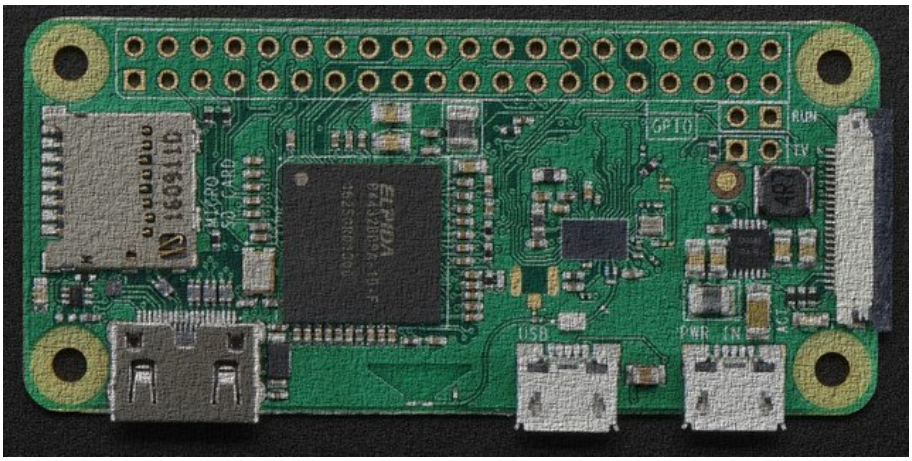
PRODUCT ID: 939
This is the cutest little microSD card reader/writer - but don't be fooled by its adorableness! It's wicked fast and supports up to 64 GB SDXC cards! Simply slide the card into the edge...
http://adafru.it/ree
$5.95
IN STOCK

Using a Pi 3 Instead of Pi Zero W

Instead of a Pi Zero W you can directly substitute in a Pi 3 which also has built in WiFi, you won't need the 2x20 header in that case

1 x Raspberry Pi 3
Raspberry Pi 3 with WiFi built in!
Add To Cart

Prepare the Pi



The Pi Zero W is a very minimal computer, so it requires a little work to get it up and running.

We have a guide on how to set up your Pi Zero W 'headless' which is how we recommend you get started. Check out the guide for how to do that!

Set up your Pi Zero W
http://adafru.it/yuC

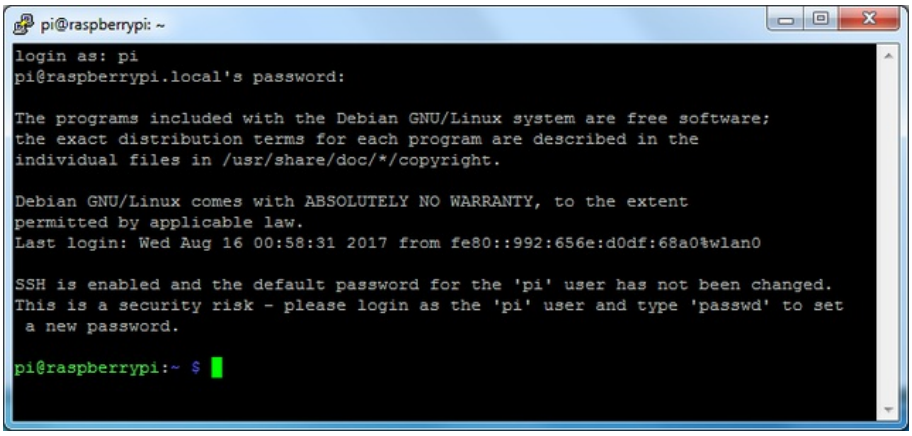
Here's the quick-start for people with some experience:

1. Download the latest **'Lite'** Raspbian (http://adafru.it/fQi) to your computer
2. Burn the Lite Raspbian to your micro SD card (http://adafru.it/dDL) using your computer
3. Re-plug the SD card into your computer (don't use your Pi yet!) and set up your wifi connection by editing supplicant.conf (http://adafru.it/yuD)
4. Activate SSH support (http://adafru.it/yuD)
5. Plug the SD card into the Pi Zero W
6. If you have an HDMI monitor we recommend connecting it up via the mini HDMI adapter we provide in the budget pack - so you can see that it's booting OK
7. Plug in power to the Pi Zero W - you will see the green LED flicker a little. The Pi Zero will reboot while it sets up so wait a good 10 minutes
8. If you are running Windows on your computer, install Bonjour support so you can use **.local** names, you'll need to reboot Windows after installation (http://adafru.it/IPE)
9. You can then **ssh** into **raspberrypi.local** (http://adafru.it/jvB)

Install Pi Hole

Pre-Check

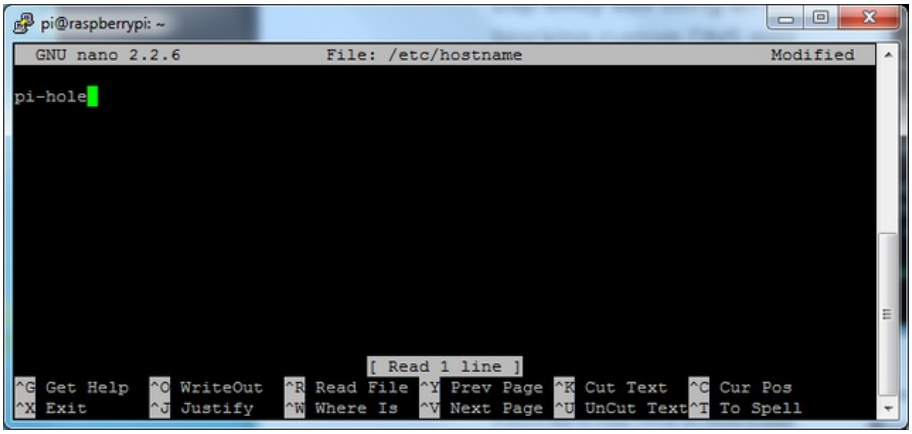
OK once you have set your Pi up and the WiFi is connecting to your home or office network, and your can ssh into it, continue with these easy steps! If you cannot connect via ssh yet, go back and read some of our guides until you are able to log into your Pi.



Change Hostname

I like to do this first so I dont get confused between all the different Pi's in the house

Edit the hostname with **sudo nano /etc/hostname** and put something else on that first line, like **pi-hole**



Reboot and when you ssh in again, use **pi-hole.local**

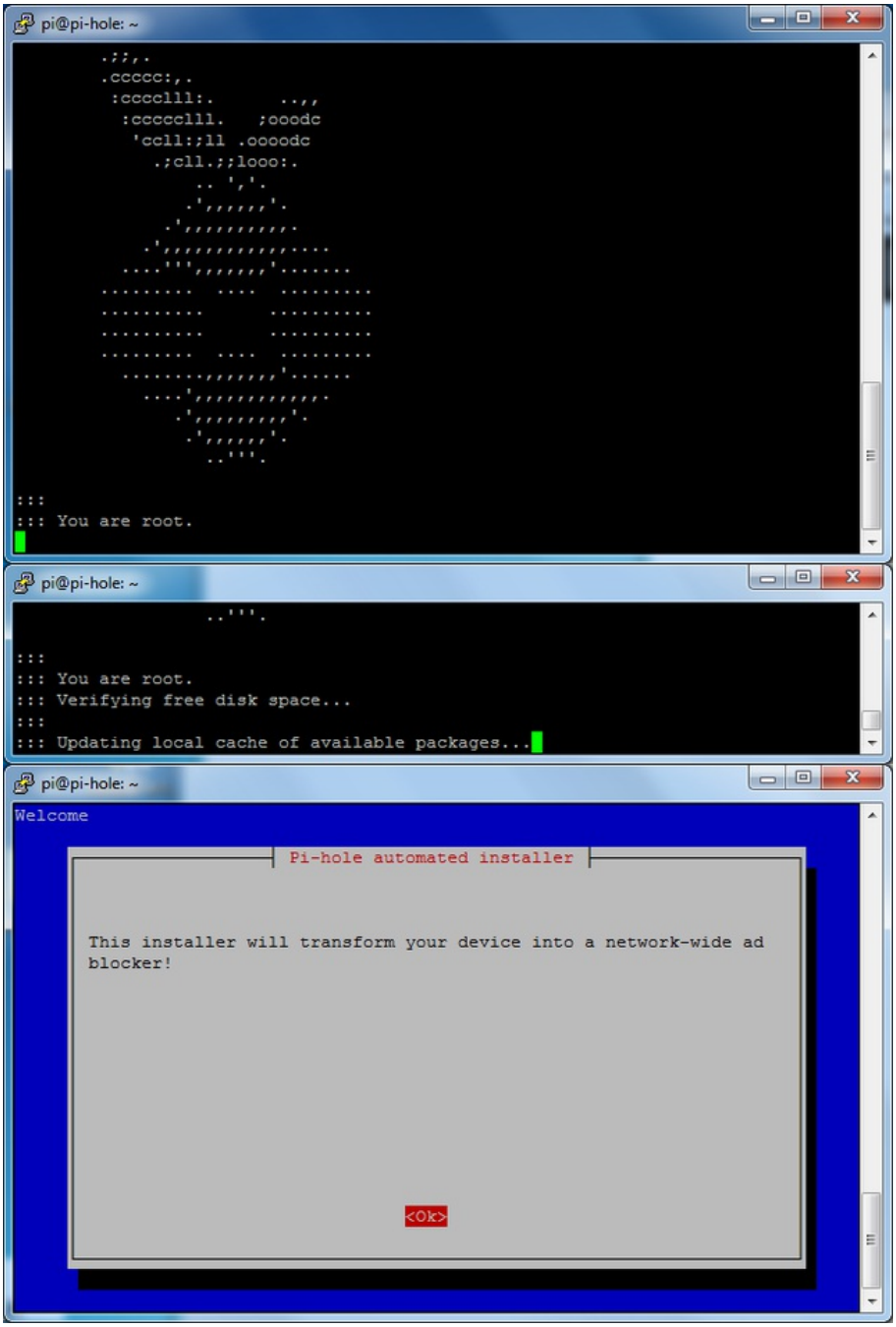
Now's also a good time to change the Pi's password with **passwd**

Run Pi Hole Installer

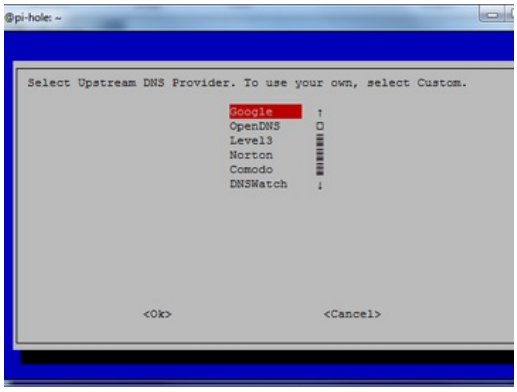
There's more information on how installation at https://pi-hole.net/ (http://adafru.it/yuE) - as of the writing of this guide, its easier to just run

```
curl -sSL https://install.pi-hole.net | bash
```

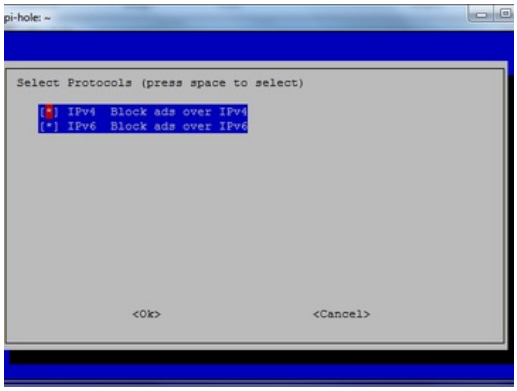
It will take quite a while to install, and may seem to 'hang' at points. Just let it do its thing for about 20 minutes!



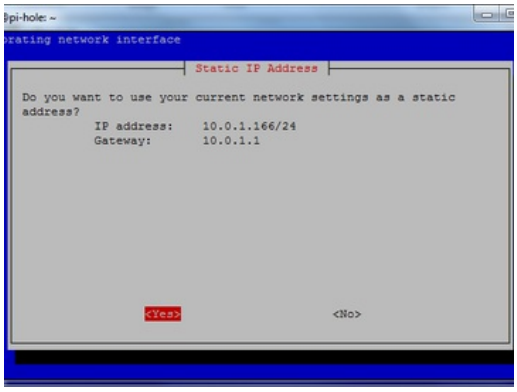
Configuration

- 

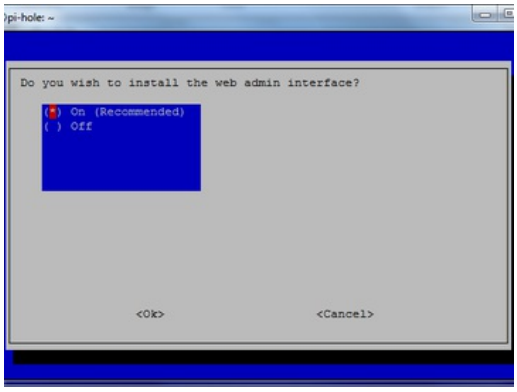
Pick who will be the upstream DNS (for non-ad blocked sites) - Google is fine and will probably be up all the time

- 

99% of people will use IPv4 - if you need IPv6 you'd know!

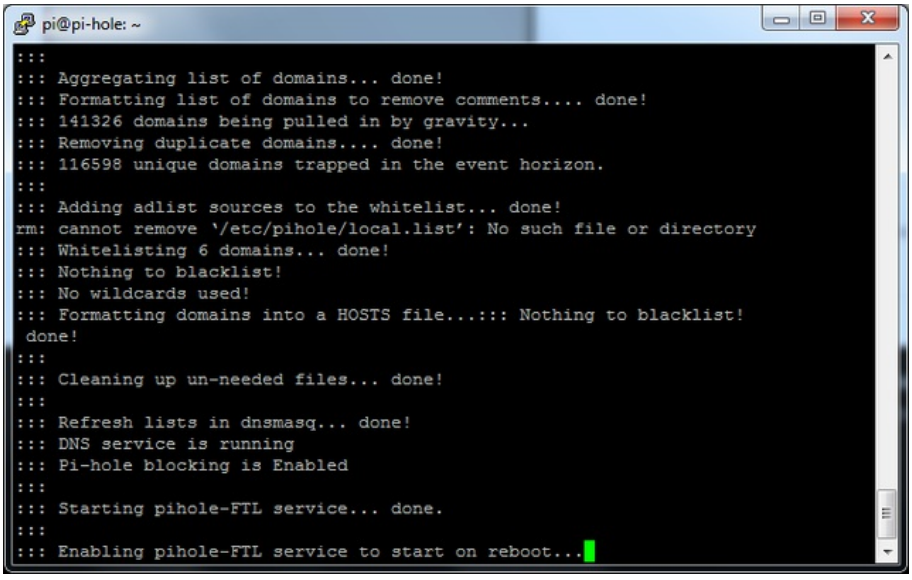
- 

The installer will automatically try to set the*dynamic* IP address it got from your router to be fixed. This works well enough, if you have an advanced network set up, you can configure a custom IP address

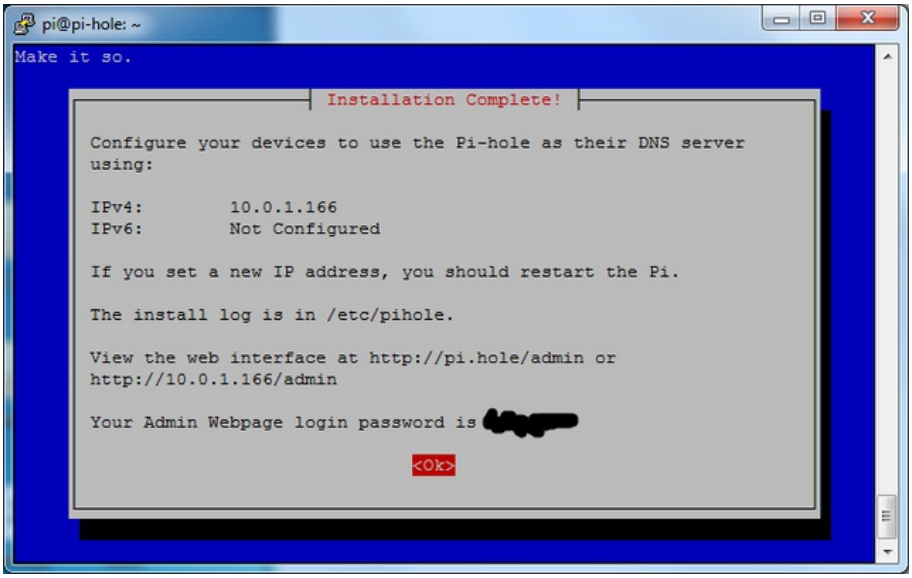
- 

The Web Interface is kinda cool, and is password protected. We'll be showing most of the stats on the little OLED but we still need the API to be running so keep this on

It will keep installing! Just hold tight...

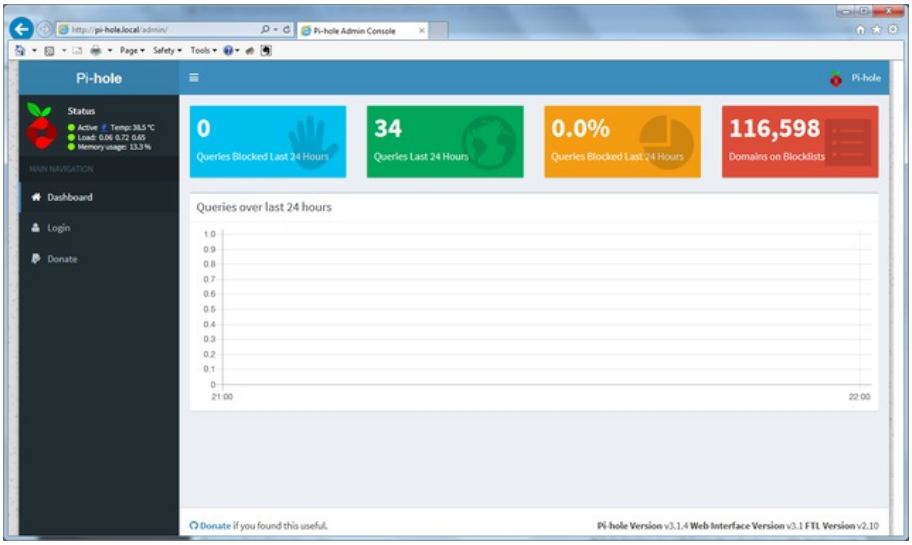


When its done you'll get this final config screen! Copy & paste the password into another window for now



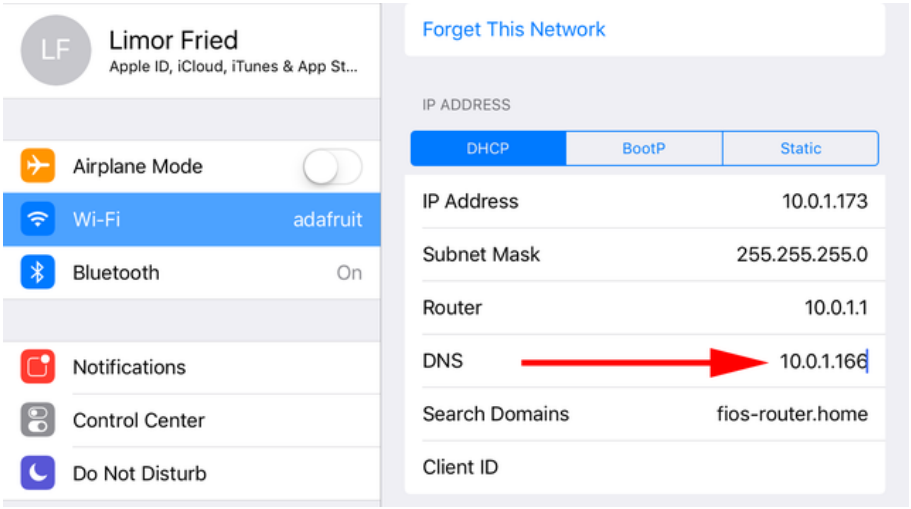
Test Admin Page

On your desktop computer or tablet, visit http://pi-hole.local/admin/ (http://adafru.it/yuF) And you should see an administration panel!

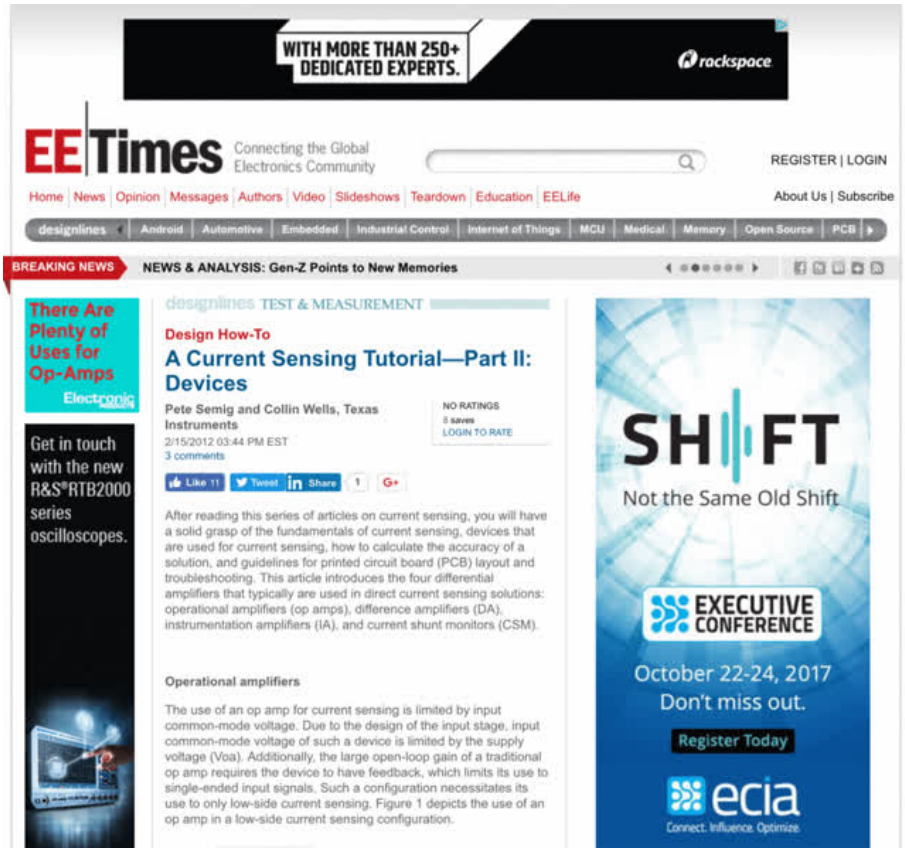


Test Blocking

On your tablet, phone, computer, etc - Set up your**DNS** server in the network settings to be the IP address of the Pi



You *may* need to restart your network or browser to have it kick in, also there may be some cached ads so don't worry if not everything is blocked. Visit your favorite site with ads (not adafruit.com cuz we don't have any! :) and see the difference!



Now that you've got that done, lets continue and install the PiOLED for a small display!

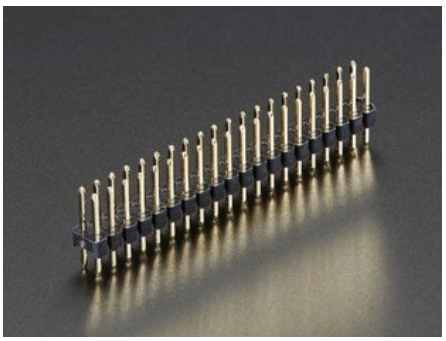
Install PiOLED

Our little PiOLED add on makes a very cute and easy way to display the Pi Hole stats. We were inspired to add this when we saw this tweet!

What a perfect use! Here's how to add it on for some nice stats. It also displays the hostname and IP address so if you forget it you can just look at the display. It will also tick up when its in use so you can tell its working.

Install 2x20 Header

If you are using a Pi Zero you'll need to solder in or somehow attach a 2x20 header so you can plug in the Pi OLED. Use either a plain 2x20 header and solder it in using an iron + some solder... (http://adafru.it/drI)



Break-away 0.1" 2x20-pin Strip Dual Male Header

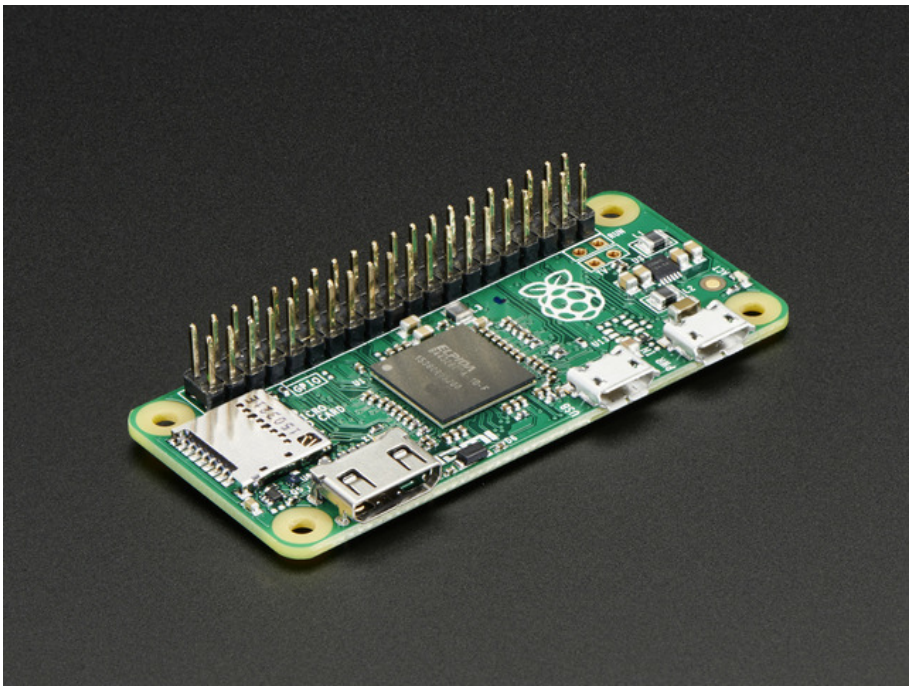
PRODUCT ID: 2822
If we could eat headers, we'd have them for breakfast, lunch, and dinner. But we can't :(So we're making the best of it and selling them! This 2x20-pin...
http://adafru.it/vlb
$0.95
IN STOCK

Or you can use Hammer headers which do not need soldering!

GPIO Hammer Headers - Solderless Raspberry Pi Connectors

PRODUCT ID: 3413
If your soldering isn't quite up to scratch, or you just don't own a soldering iron yet, then these nifty hammer headers from Pimoroni might be just what you need. They come...
http://adafru.it/yvd
$6.50
IN STOCK

Either way, you'll want to end up with something like this:

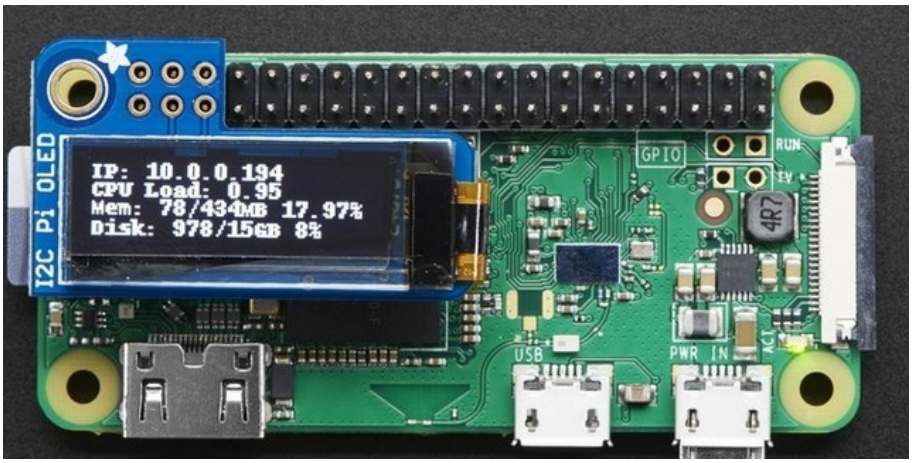


Add Pi OLED support

While still logged into the Pi Hole Pi Zero W, and in the terminal shell, you can install support for the PiOLED. Follow our step by step guide to add I2C support and install the Python library

Install PiOLED Software
http://adafru.it/yva

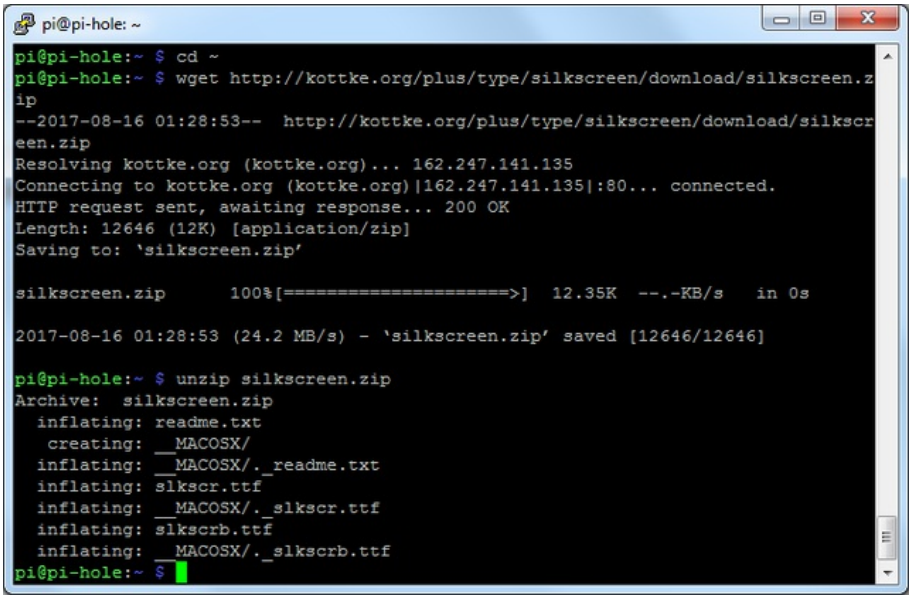
After you get to the part where you run **sudo python stats.py** and get this screen, come back here!



First thing I did is update the font so its a little clearer. I used Kottke's free Silkscreen font which looks great on small screens. (http://adafru.it/yvb)

It's easy to install on your Pi, run

```
cd ~
wget http://kottke.org/plus/type/silkscreen/download/silkscreen.zip
unzip silkscreen.zip
```



Here's the new stats.py code.

Create a new file with **nano ~pi/stats.py** and paste this in! Then save it

```
# Copyright (c) 2017 Adafruit Industries
# Author: Ladyada, Tony DiCola & James DeVito
#
# Permission is hereby granted, free of charge, to any person obtaining a copy
# of this software and associated documentation files (the "Software"), to deal
# in the Software without restriction, including without limitation the rights
# to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the Software is
# furnished to do so, subject to the following conditions:
#
```

```
# The above copyright notice and this permission notice shall be included in
# all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
# AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
# LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
# OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
# THE SOFTWARE.
import time

import Adafruit_GPIO.SPI as SPI
import Adafruit_SSD1306

from PIL import Image
from PIL import ImageDraw
from PIL import ImageFont

import subprocess
import json
import requests

api_url = 'http://localhost/admin/api.php'

# Raspberry Pi pin configuration:
RST = None     # on the PiOLED this pin isnt used

# 128x32 display with hardware I2C:
disp = Adafruit_SSD1306.SSD1306_128_32(rst=RST)

# Initialize library.
disp.begin()

# Clear display.
disp.clear()
disp.display()

# Create blank image for drawing.
# Make sure to create image with mode '1' for 1-bit color.
width = disp.width
height = disp.height
image = Image.new('1', (width, height))

# Get drawing object to draw on image.
draw = ImageDraw.Draw(image)

# Draw a black filled box to clear the image.
draw.rectangle((0,0,width,height), outline=0, fill=0)

# Draw some shapes.
# First define some constants to allow easy resizing of shapes.
padding = -2
top = padding
bottom = height-padding
# Move left to right keeping track of the current x position for drawing shapes.
x = 0

# Load nice silkscreen font
font = ImageFont.truetype("/home/pi/slkscr.ttf", 8)

while True:
    # Draw a black filled box to clear the image.
    draw.rectangle((0,0,width,height), outline=0, fill=0)

    # Shell scripts for system monitoring from here : https://unix.stackexchange.com/questions/119126/command-to-display-memory-usage-disk-usage-and-cpu-load
    cmd = "hostname -I | cut -d\' \' -f1"
    IP = subprocess.check_output(cmd, shell = True )
    cmd = "hostname"
    HOST = subprocess.check_output(cmd, shell = True )
    cmd = "top -bn1 | grep load | awk '{printf \"CPU Load: %.2f\", $(NF-2)}'"
    CPU = subprocess.check_output(cmd, shell = True )
    cmd = "free -m | awk 'NR==2{printf \"Mem: %s/%sMB %.2f%%\", $3,$2,$3*100/$2 }'"
    MemUsage = subprocess.check_output(cmd, shell = True )
    cmd = "df -h | awk '$NF==\"/\"{printf \"Disk: %d/%dGB %s\", $3,$2,$5}'"
```

```
    Disk = subprocess.check_output(cmd, shell = True )

    # Pi Hole data!
    try:
      r = requests.get(api_url)
      data = json.loads(r.text)
      DNSQUERIES = data['dns_queries_today']
      ADSBLOCKED = data['ads_blocked_today']
      CLIENTS = data['unique_clients']
    except:
      time.sleep(1)
      continue

    draw.text((x, top),       "IP: " + str(IP) + "( " + HOST + ")",  font=font, fill=255)
    draw.text((x, top+8),     "Ads Blocked: " + str(ADSBLOCKED), font=font, fill=255)
    draw.text((x, top+16),    "Clients:     " + str(CLIENTS),  font=font, fill=255)
    draw.text((x, top+24),    "DNS Queries: " + str(DNSQUERIES),  font=font, fill=255)

    # skip over original stats
    #draw.text((x, top+8),     str(CPU), font=font, fill=255)
    #draw.text((x, top+16),    str(MemUsage),  font=font, fill=255)
    #draw.text((x, top+25),    str(Disk),  font=font, fill=255)

    # Display image.
    disp.image(image)
    disp.display()
    time.sleep(.1)
```

Download stats.py file
http://adafru.it/yvc

You'll notice its very similar to the original **stats.py** but we've added PiHole API support. Here's how we did that!

First up, Pi Hole stats are available through the web server, in json format, so we need to add web requests and json parsing to python. Then set the URL for the API access, which is localhost (the same computer) and through the admin page:

```
import subprocess
import json
import requests

api_url = 'http://localhost/admin/api.php'
```

We load up the nice Silkscreen font here, in 8 point type. Note that we have to have the full path of the file.

```
# Load nice silkscreen font
font = ImageFont.truetype("/home/pi/slkscr.ttf", 8)
```

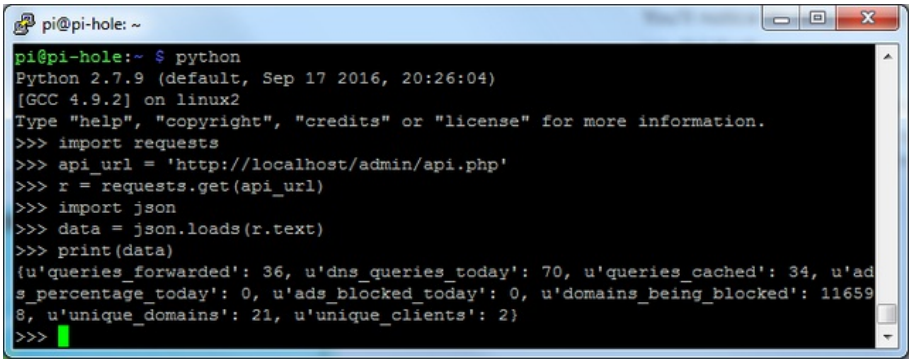
This is where we grab the API data. I put it in a **try** block, so it would retry in case the API access failed for some reason

```
    # Pi Hole data!
    try:
      r = requests.get(api_url)
      data = json.loads(r.text)
      DNSQUERIES = data['dns_queries_today']
      ADSBLOCKED = data['ads_blocked_today']
      CLIENTS = data['unique_clients']
    except:
      time.sleep(1)
      continue
```

If you want to print out different info, run this small script in python to see what is available:

```
import json
import requests

api_url = 'http://localhost/admin/api.php'
r = requests.get(api_url)
data = json.loads(r.text)
print(data)
```



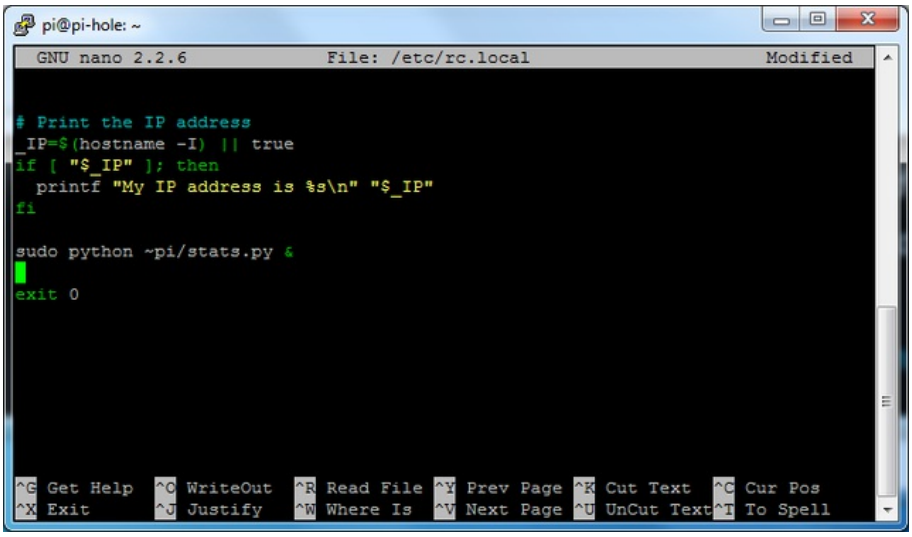
You can also customize the display printout, but i liked having the IP first, then the pi hole stats below:

```
draw.text((x, top),       "IP: " + str(IP) + "( " + HOST + ")",  font=font, fill=255)
draw.text((x, top+8),     "Ads Blocked: " + str(ADSBLOCKED), font=font, fill=255)
draw.text((x, top+16),    "Clients:     " + str(CLIENTS),  font=font, fill=255)
draw.text((x, top+24),    "DNS Queries: " + str(DNSQUERIES),  font=font, fill=255)
```

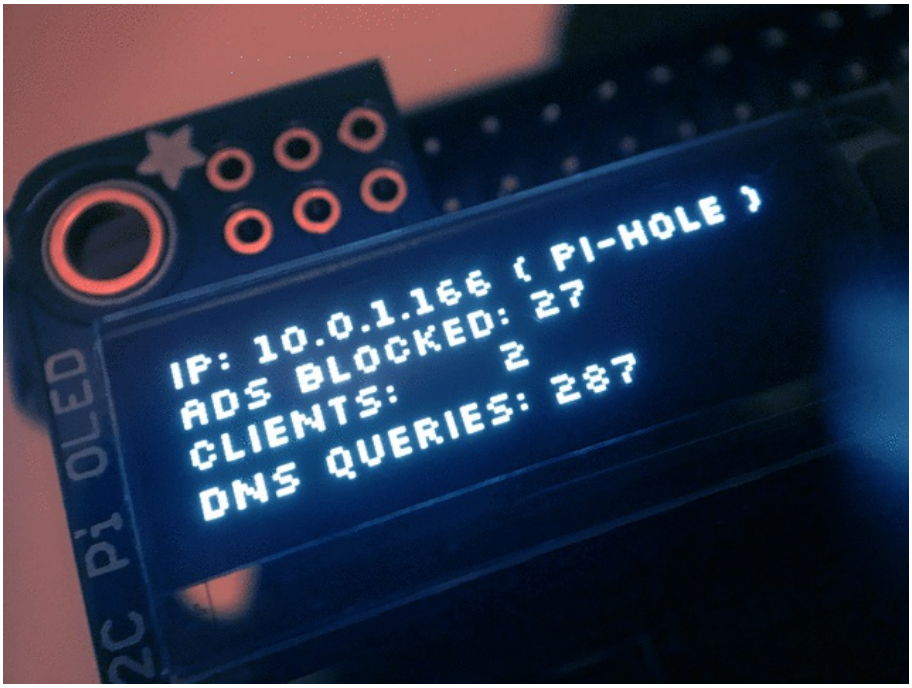
Test & Stats at Startup

Once you have the script saved, you can run it with **sudo python ~pi/stats.py** and look on the OLED to make sure you see your IP address and such!

Lastly we just want to make this run at boot. We'll do that the easy way by editing /etc/rc.local with **sudo nano /etc/rc.local** and adding **sudo python ~pi/stats.py &** before **exit 0**



Then save and you can reboot to test it out


IP: 10.0.1.166 (PI-HOLE)
ADS BLOCKED: 27
CLIENTS: 2
DNS QUERIES: 287
2C Pi OLED