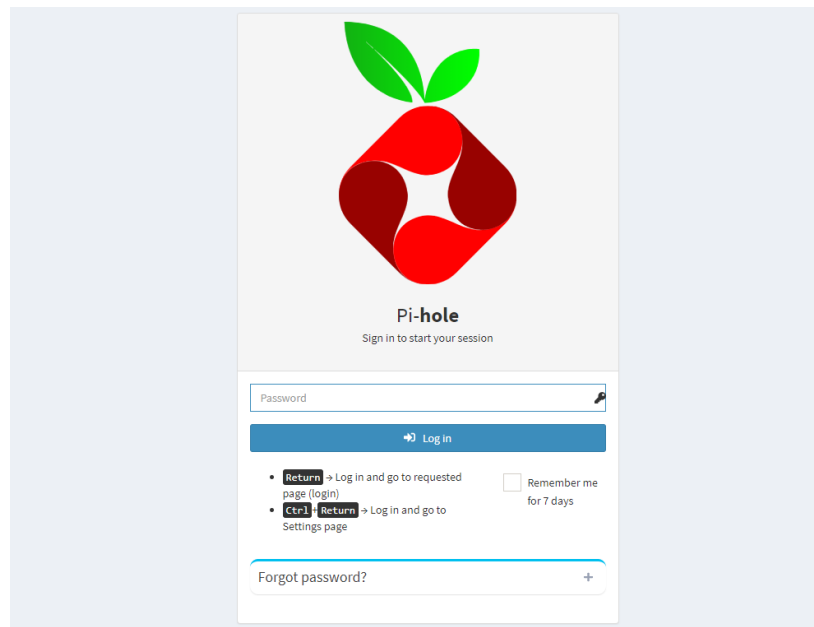# How to Setup a Pi-hole For IP Filtering and Recursive DNS

Project Summary:

The Pi-hole provides an easy way to setup advertising filtering for individual computers or your entire local area network. We can also configure this system to help increase our privacy by allowing the Pi to cache DNS information and act as the DNS server for our network, so that we do not have to pass this information to an outside source every time we visit a website. This project gives you the hand-on experience to set one of these up with the chance to try things or make mistakes on one of our lab Raspberry Pis, so no worries.
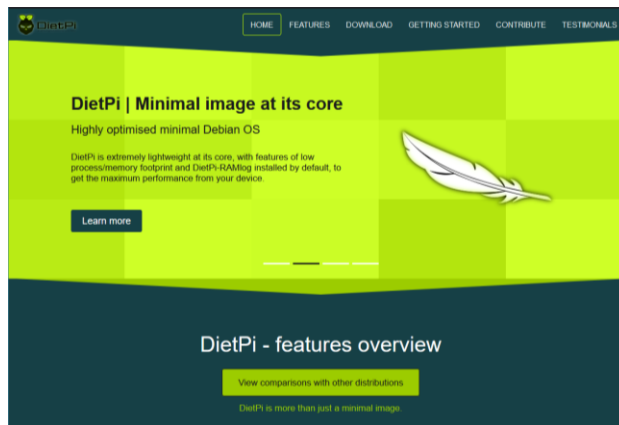
Parts Needed:
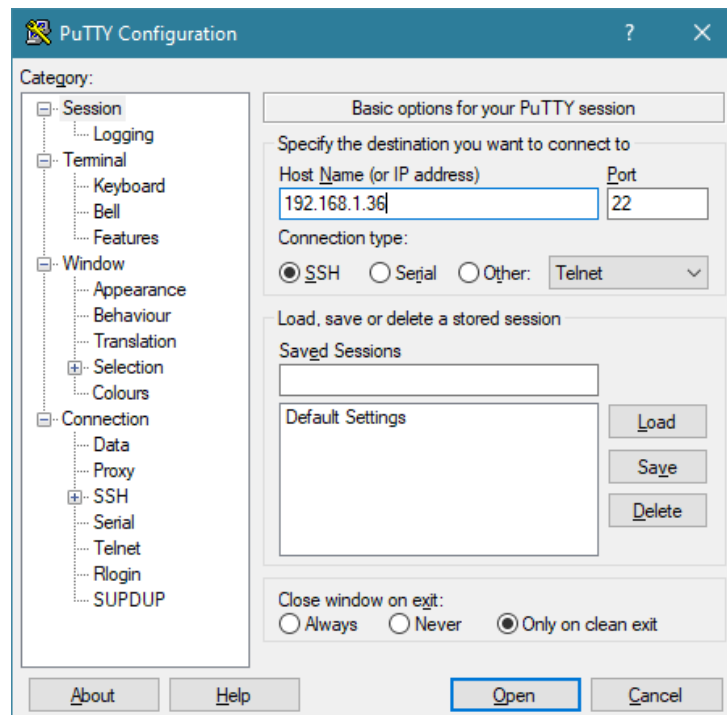
RaspberryPi

MicroSD Card Flashed with DietPi

Putty installed on a PC

Cat5 cable and Pi power cord

Step-by-Step Directions:

Download the DietPi OS https://dietpi.com/ and flash to card with Balena Etcher https://www.balena.io/etcher/ . We already have these in the lab, so just ask the lab tech to help get you up and running.
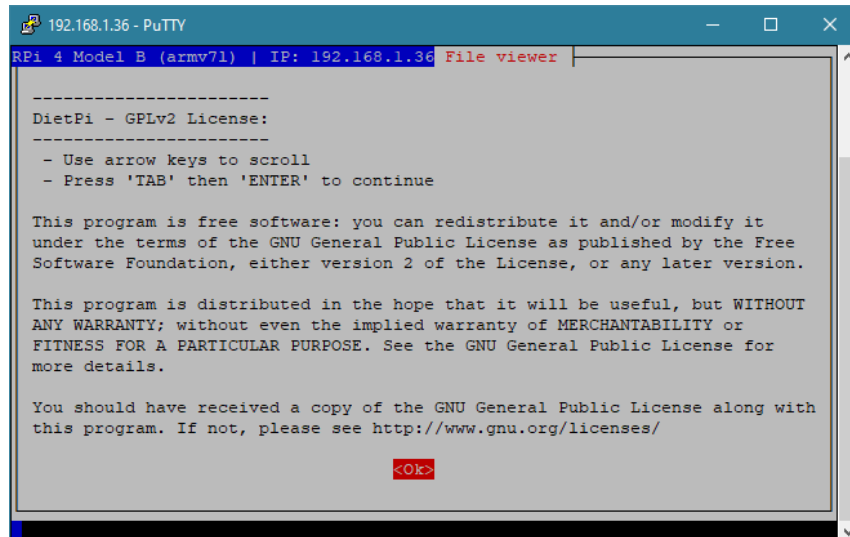

Find the device on your network and connect over SSH. Again, it is ok to ask for help if you are not sure of the correct IP address.
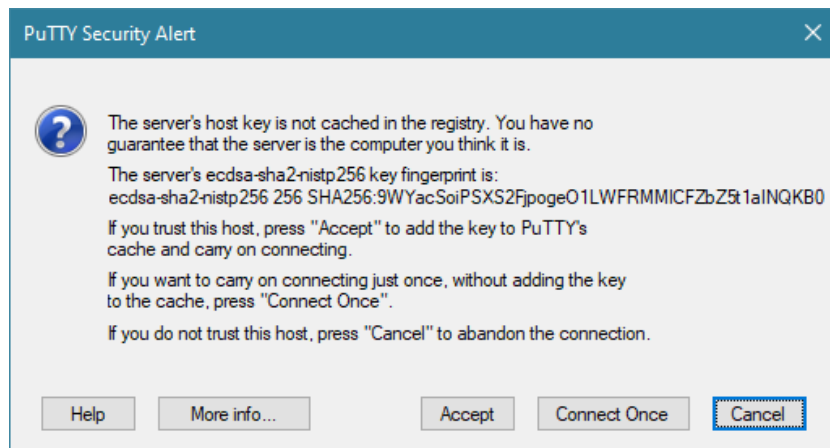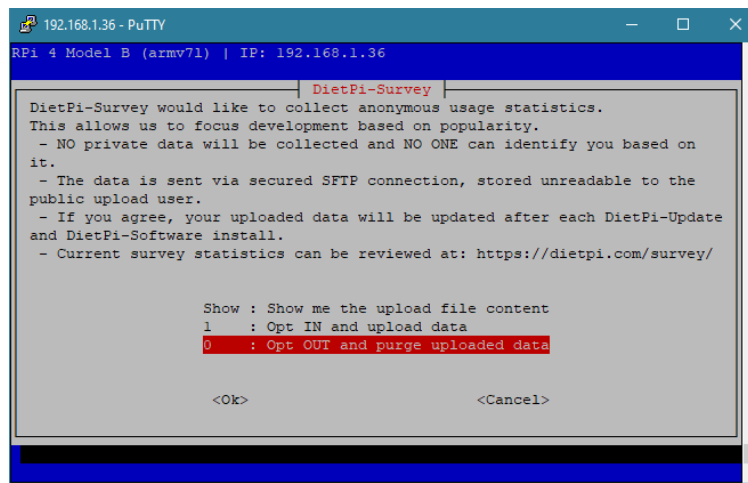
You might get a popup warning that this is a new connection. It is ok to click "Accept".
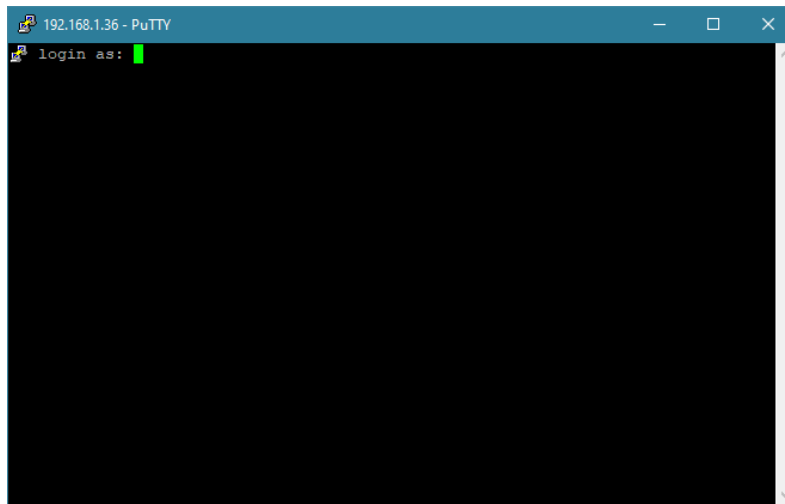

Login using the username **root** and the password **dietpi** .

Read and agree with the License to move forward by hitting **Enter**. The Pi will go through and update phase for a few of minutes with scrolling text on the screen, etc. This is expected and a good sign.

The DietPi Survey page will show up when the updates are competed. For the highest level of privacy, Choose **0: Opt Out** with the up/down arrow keys and hit **Enter**.
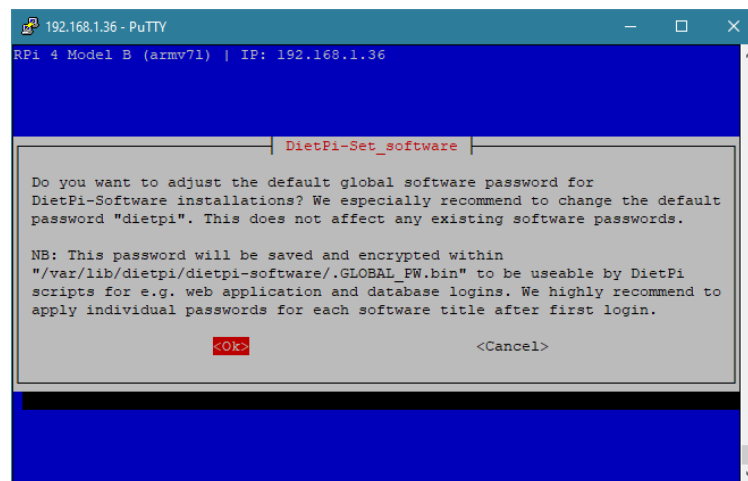

Next, DietPi will ask if you want to change the password to something other than the default one that everyone in the entire world could find with a simple Google search. Yes. Yes, we do. So, **Tab** over to <Ok> and hit enter.
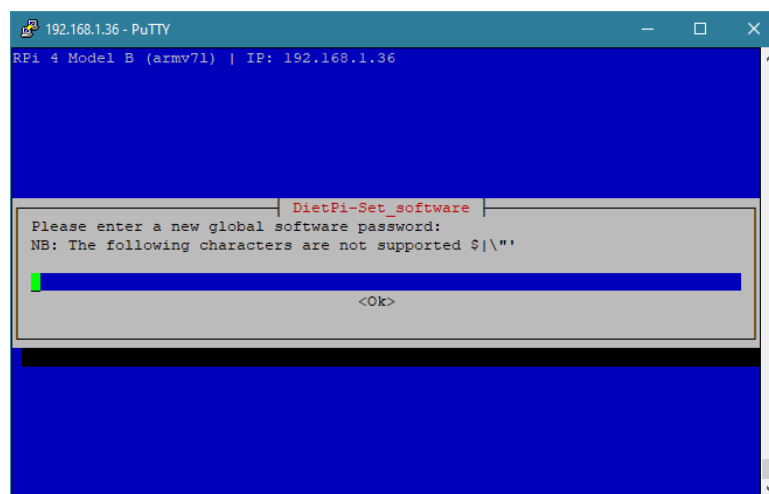

Choose a high entropy passphrase by picking at least three random words strung together. Also, write this down somewhere for the time being so that you do not forget it.
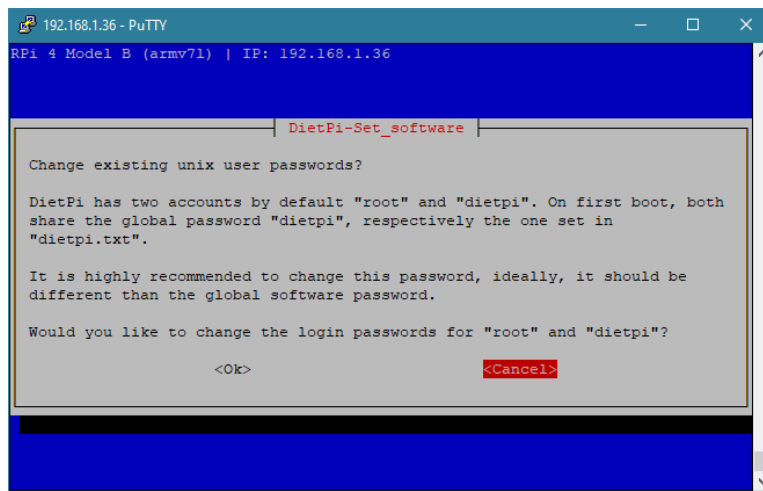
Same scenario for the other passwords. Pick a winner. Then, Choose <Cancel> on the next screen to not disable the Serial console.
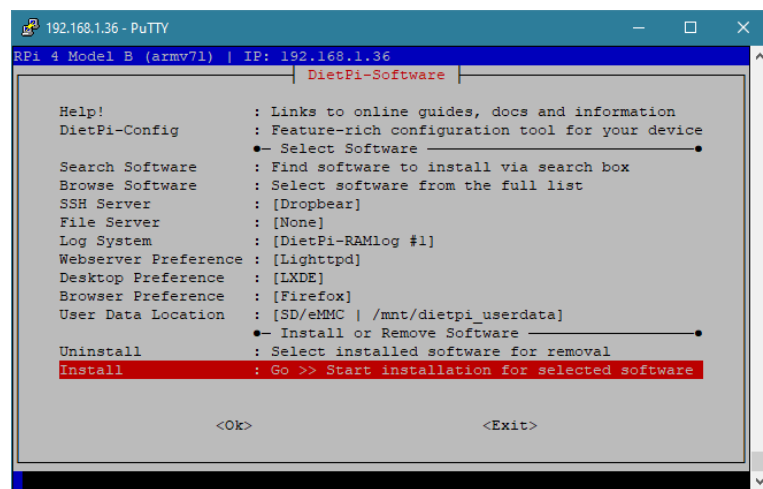
Now, we are finally at the DietPi OS Install menu! Good progress so far. We are trying to keep this OS install as lean as possible since we are working with a single board computer, so use the arrow keys to select **Install** and hit **Enter**.
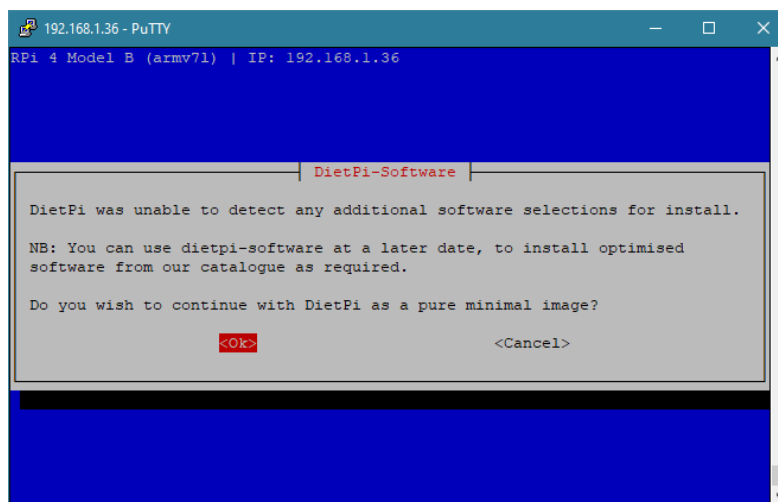
DietPi will double check, since we are doing a really lean installation. Select **<Ok>** and hit **Enter** to continue. This OS will go through and install all of our settings, etc.

Now that the base OS is setup, we can install PiHole on top of it. From the prompt, enter the following case-sensitive command: **curl -sSL https://install.pi-hole.net | bash** Then, relax for a few minutes while we go through another update and install sequence. Keep an eye out for relevant error or informational messages. If errors occur, rebooting the Pi with the reboot command, logging back in and trying the PiHole install command again will fix many issues.
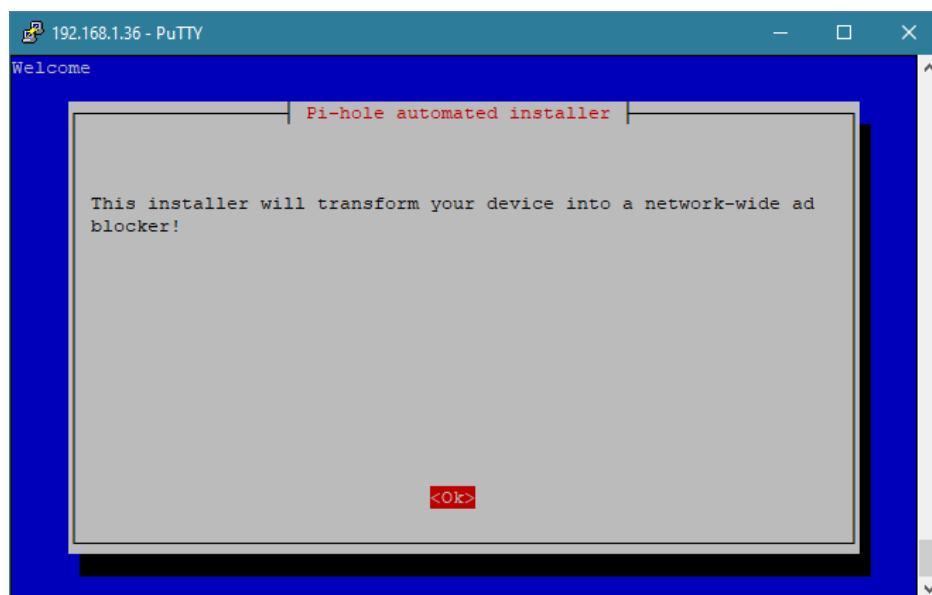

You will need to "Ok" your way through three more informational screens.
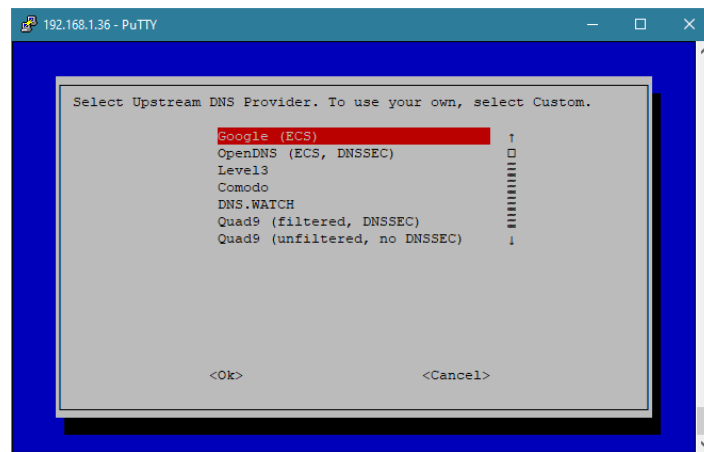
There is a short list of DNS providers to choose from, each with a few different tradeoffs. If you want reliability but not privacy, then Google is a good choice. However, services like OpenDNS and DNS.WATCH have vowed to keep track of much less information about your searches. Feel free to do a little internet research on sites such as: https://securitytrails.com/blog/dns-servers-privacy-security Pick your favorite and hit enter.
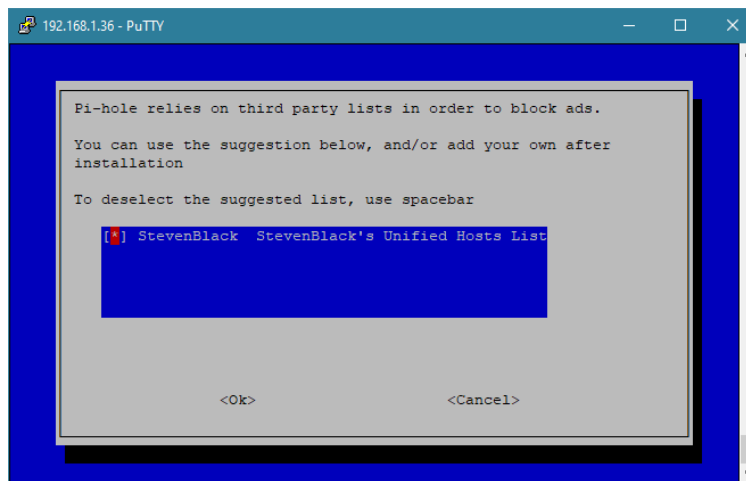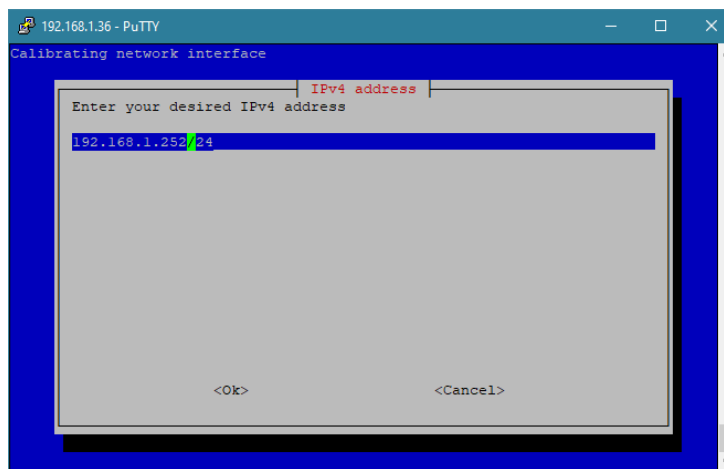

The suggested ad block list is a good one, so hit enter to continue.
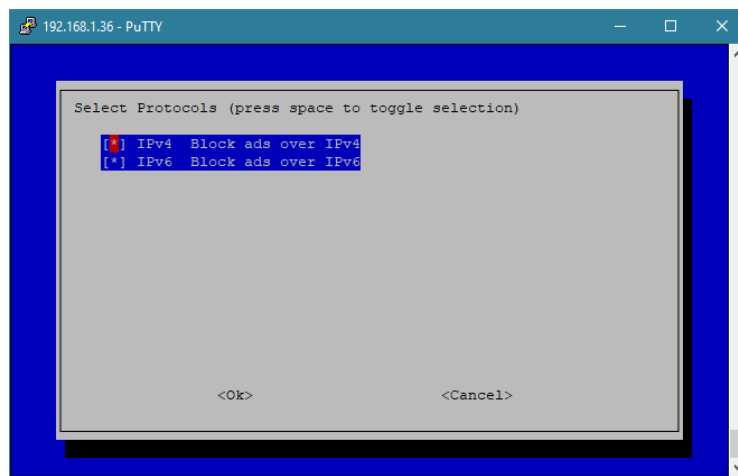
The default of running this for both IPv4 and IPv6 makes sense in most situations, so hit enter to move on.
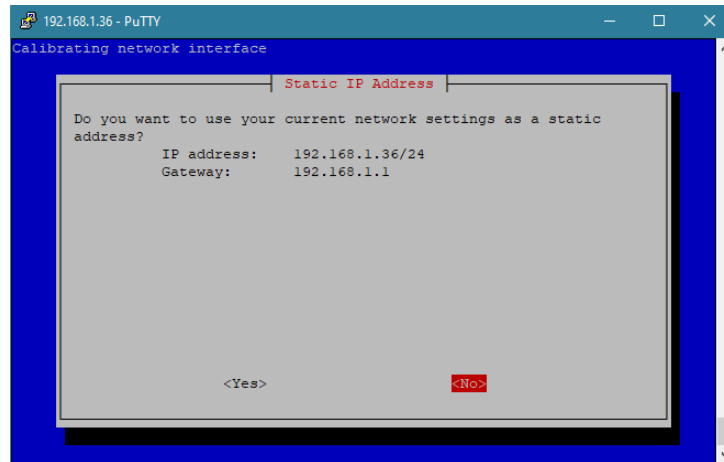
We do not want to use whatever random IP address got thrown our way by the DHCP server, so choose <No>.

Then, pick a higher range IP address, such as 192.168.1.252/24 that is not being used elsewhere. Also make sure the new address is outside of the range being assigned to your network by DHCP. Make a note of this IP Address as well before you hit enter.
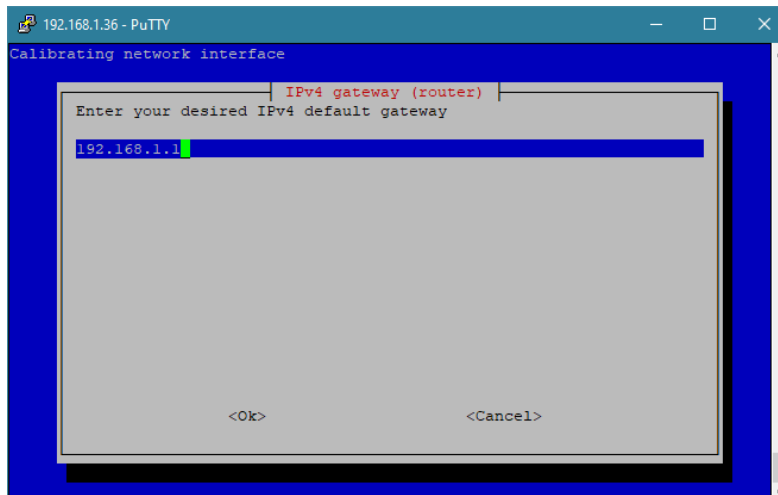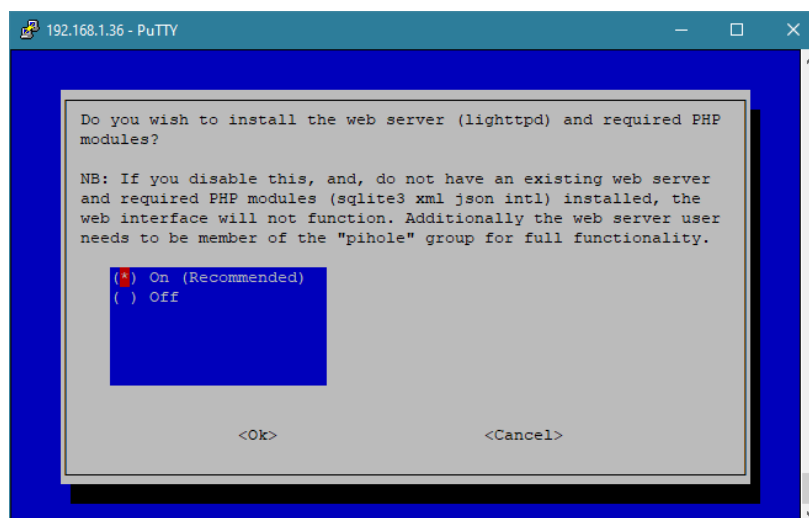

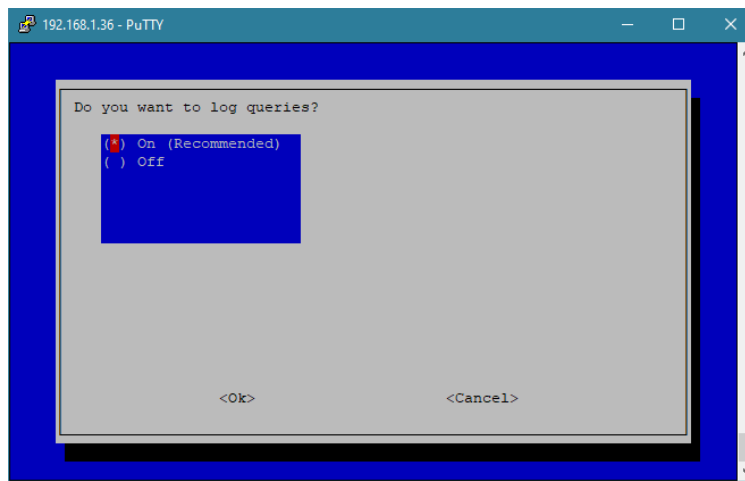The suggest default gateway is usually correct.
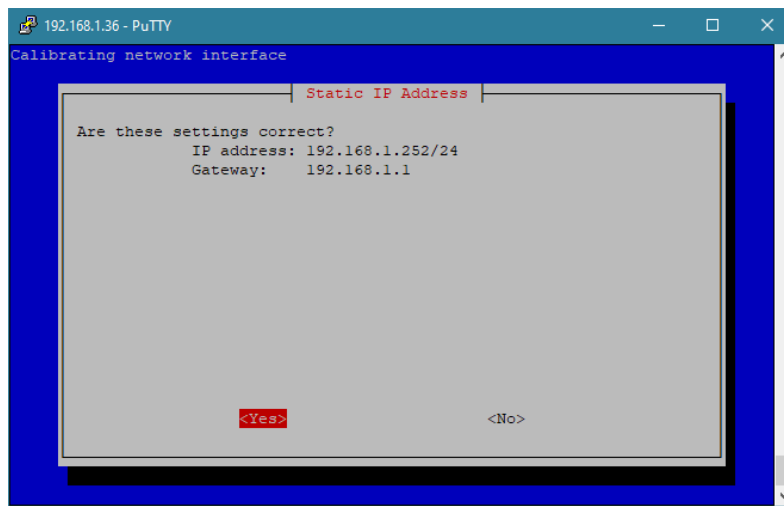
Pi-hole will give you a chance to doublecheck the settings before hitting enter to move on.
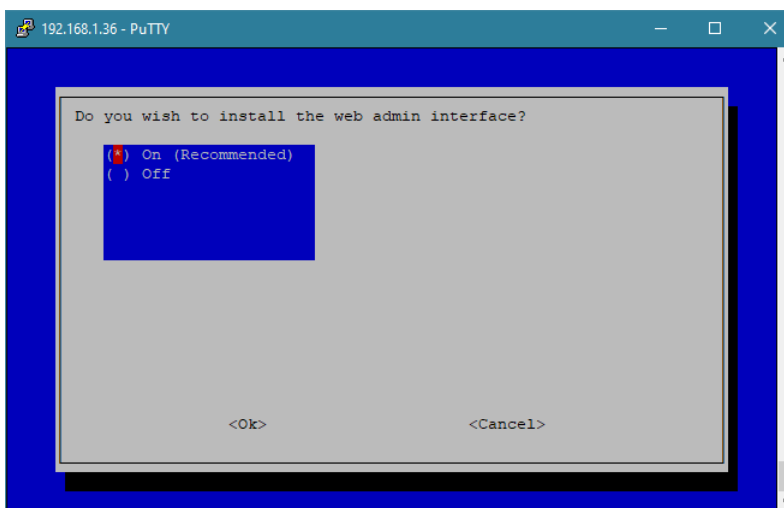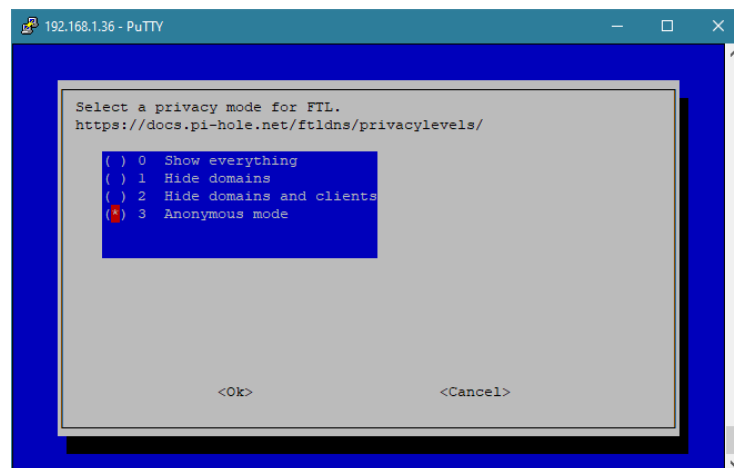

Yes, we want to install the web administration interface, so hit enter to continue.

Yes, we want to install these modules as well, so hit enter to continue.

Logging queries can be left on or turned off, depending on what you want to keep track of.

If you want to keep track of your employees or users searching, then the default show everything will accomplish this. Anonymous mode is the most private. As before, you are welcome to do a little research on the specifics of each settings to dial this into your needs. Hitting enter will finish setting up the pi-hole.
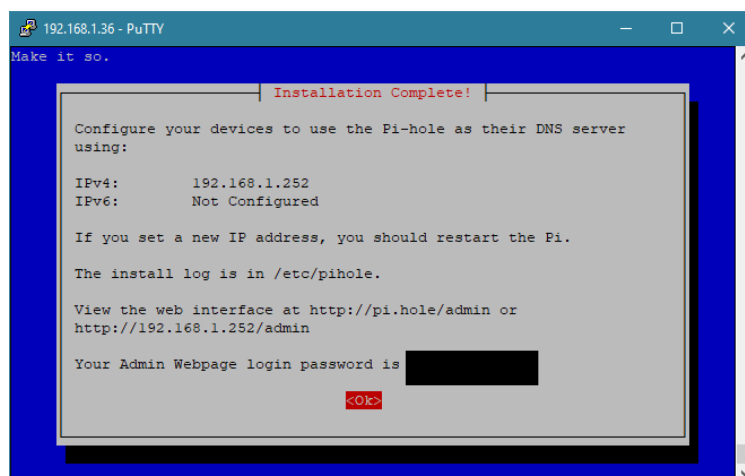
Make sure to write down the Admin Webpage login password before you hit enter to move on. We will need this password to log into the Pi-hole web interface a little later.

Boom! We're almost done with this part of the installation.

We do need to make sure that the DietPi OS knows about the IP address that we picked out earlier for the Pi-hole application. So, from the root@DietPi:~# prompt, type the **dietpi-config** command and hit enter. This will take you, shockingly, to the DietPi-Config menu. We need to arrow down to option number seven, Network Options: Adapters.

Now, arrow up to the Ethernet option and hit Enter to select it.
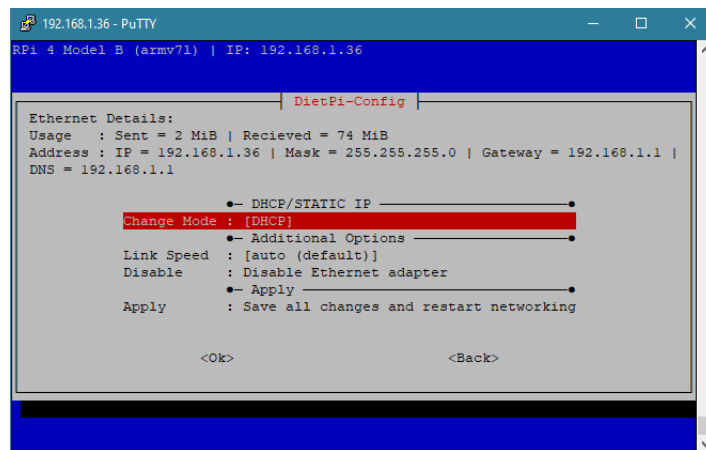
Here is the potential problem! The OS still thinks that it has to ask the DHCP server for an IP address every time that it boots up. To fix this, let's choose the Change Mode option and hit enter.
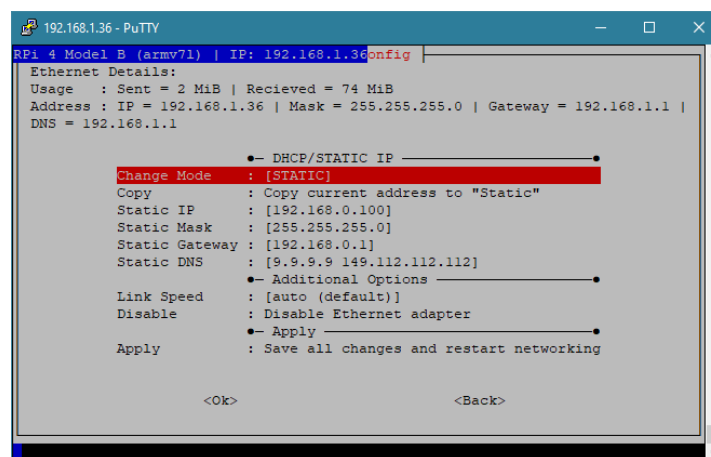
This will change the mode over to having the Pi's OS use a static IP address. We do need to work our way down the list of settings shown and make sure that they all make sense. Let's select Static IP and hit enter.

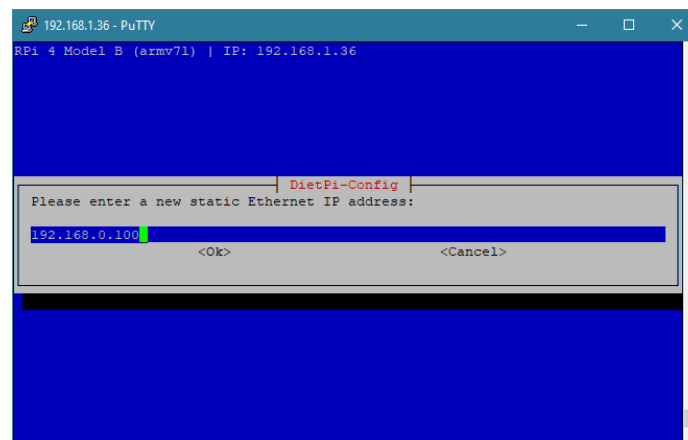Fixing this is just a matter of typing in the IP address that we selected earlier for the PiHole and hitting enter. Make sure to also select the **Static Gateway** option and change the IP address of your router, so the Raspberry Pi knows how to get out to the Internet. Usually, the rest of the IP configuration settings are fine by default.
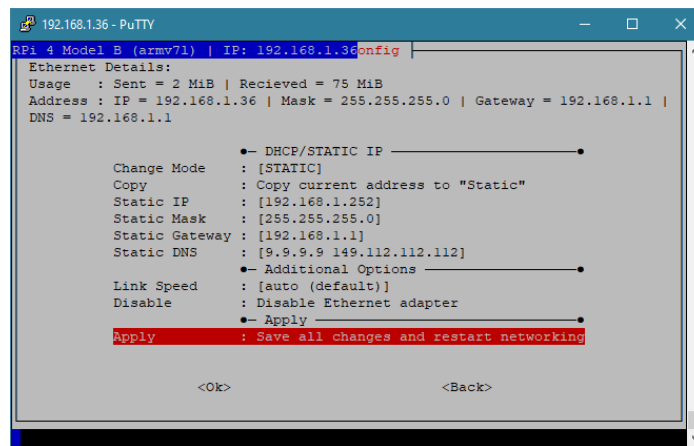
Once those two IP addresses have been changed, select **Apply : save all changes and restart networking** and hit enter.
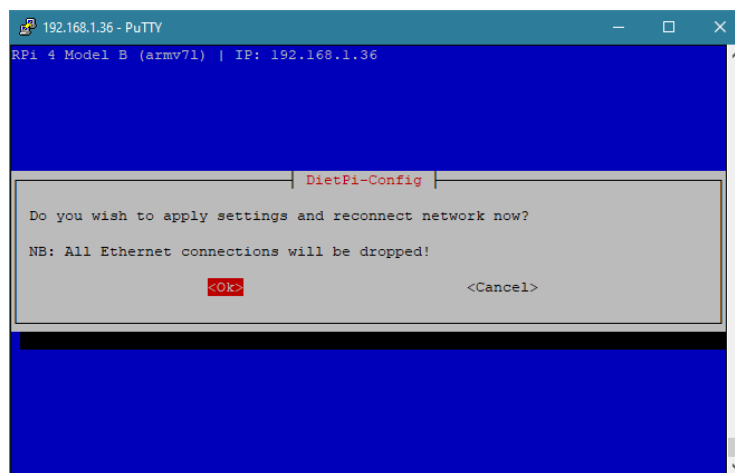
You should see a double check warning screen, but it is ok to select <Ok> and hit enter.
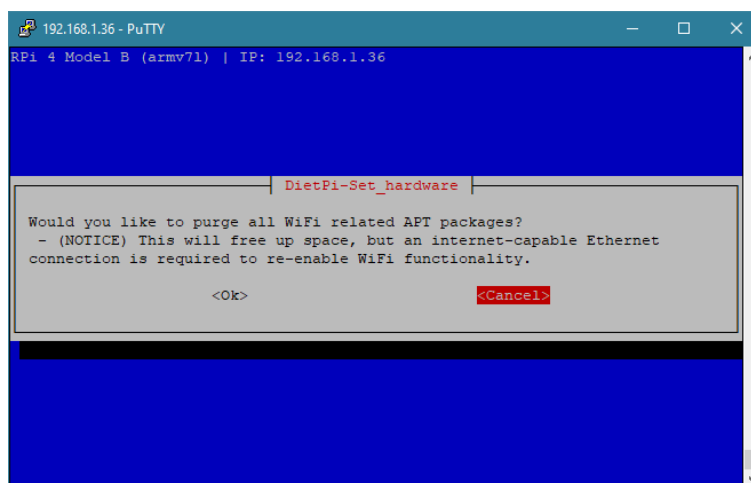

Here, we can select Cancel and hit enter. We will then need to wait two or three minutes for the Pi to reset and reload the OS and Pi-hole. It is safe to now close Putty, as we should not need it for a while, other than troubleshooting.

In your web browser, type the IP address that we just set up for Pi-Hole and DietPi. If everything is setup correctly, it should take you to the Pi-Hole logo page. Click the only link on this page to access the admin panel.
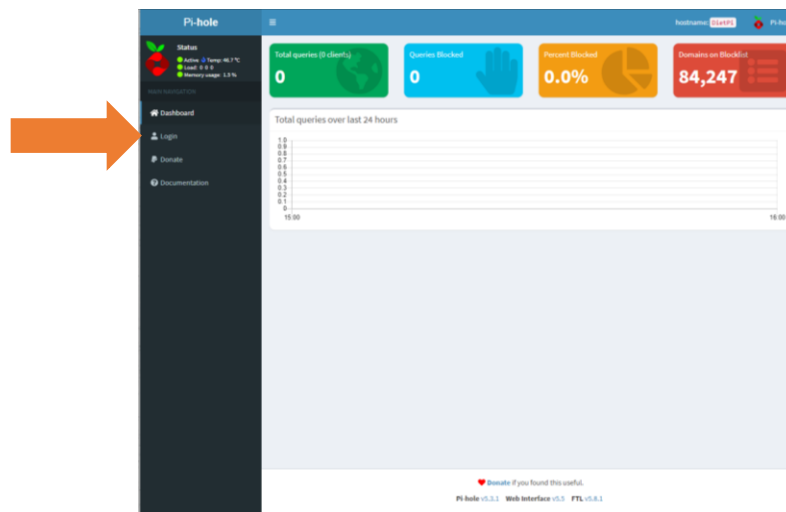

The Login link is on the left-hand side of the screen, so let's click it.
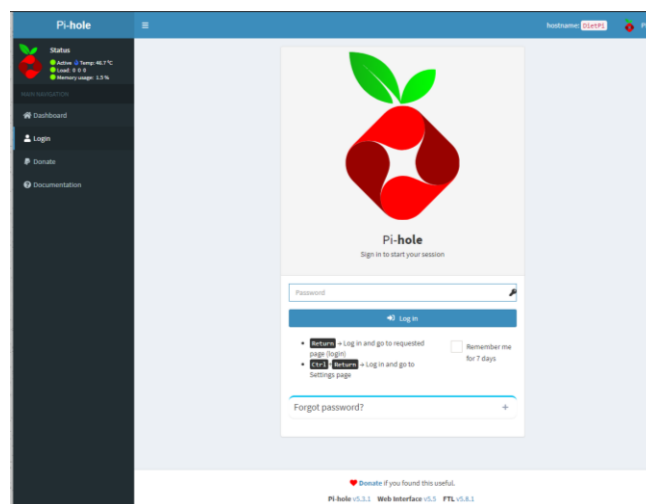
Pi-hole should now ask for the random password that it gave us during the later configuration steps.
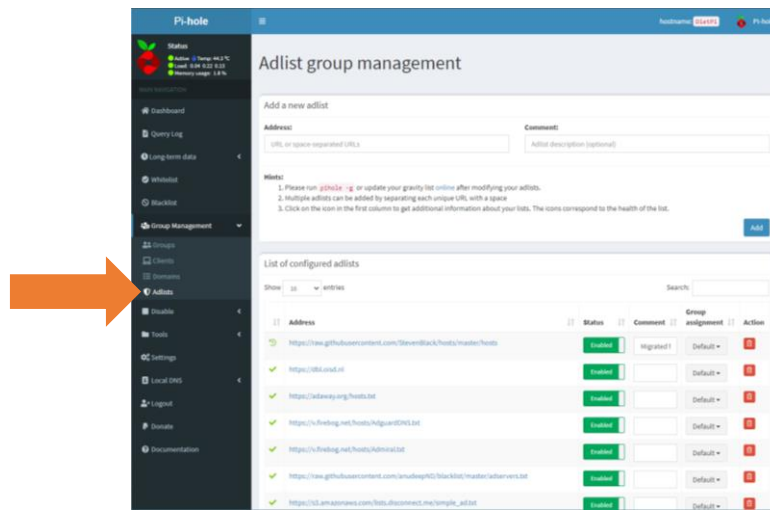
Clicking on **Group Management**, then **Adlists** will let you add to the lists of blocked IPs. Make sure to look at Hints: near the top, as hint number one has the "online" link that will let you actually update and apply the lists after they have been added.
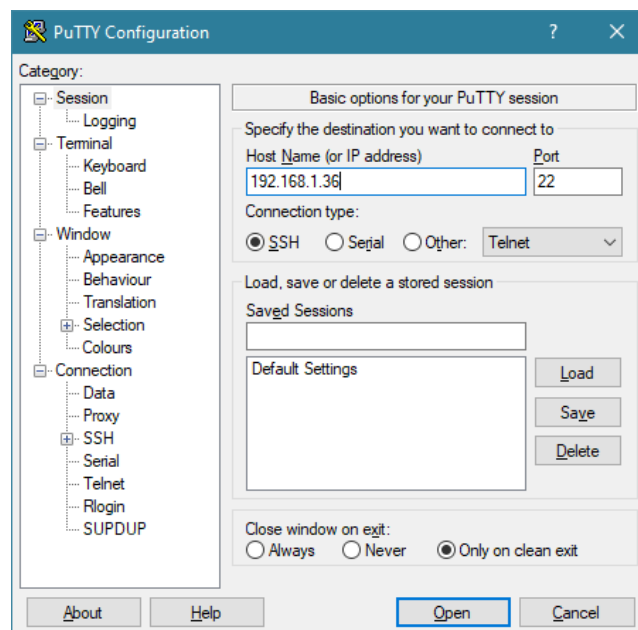
Now that all of the basics are working, we can go back into Putty and setup Pi-hole as a recursive DNS Server. This will allow the Pi to remember where websites are located and improve privacy by keeping our ISP or DNS provider from being able to keep a record of every website that you visit.
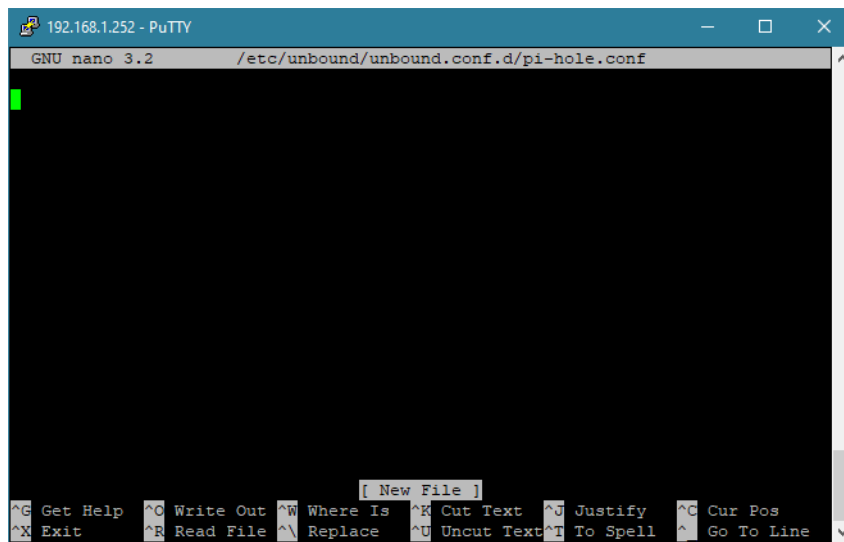
Login with the **root** username and password that you setup for the DietPi OS. (The early one, not the later one that works for the Pi-hole application.) Then, run the **sudo apt update** command. Then, run the **sudo apt install unbound** command.

The red "Failure" text is ok at this time.

```
192.168.1.252 - PuTTY                              —   □   ×
GNU nano 3.2          /etc/unbound/unbound.conf.d/pi-hole.conf       ^

[ New File ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify   ^C Cur Pos
^X Exit      ^R Read File   ^\ Replace   ^U Uncut Text ^T To Spell  ^  Go To Line  v
```

Next run the **sudo nano /etc/unbound/unbound.conf.d/pi-hole.conf** command to create the unbound config file that we need and open the empty file in the nano editor. Then copy the command block below and paste (Right Click) the commands below into the file:

```
server:
    # If no logfile is specified, syslog is used
    # logfile: "/var/log/unbound/unbound.log"
    verbosity: 0

    interface: 127.0.0.1
    port: 5335
    do-ip4: yes
    do-udp: yes
    do-tcp: yes

    # May be set to yes if you have IPv6 connectivity
    do-ip6: no

    # You want to leave this to no unless you have *native* IPv6. With 6to4
and
    # Terredo tunnels your web browser should favor IPv4 for the same reasons
    prefer-ip6: no

    # Use this only when you downloaded the list of primary root servers!
    # If you use the default dns-root-data package, unbound will find it
automatically
    #root-hints: "/var/lib/unbound/root.hints"

    # Trust glue only if it is within the server's authority
    harden-glue: yes

    # Require DNSSEC data for trust-anchored zones, if such data is absent,
the zone becomes BOGUS
    harden-dnssec-stripped: yes

    # Don't use Capitalization randomization as it known to cause DNSSEC
issues sometimes
    # see https://discourse.pi-hole.net/t/unbound-stubby-or-dnscrypt-
proxy/9378 for further details
    use-caps-for-id: no

    # Reduce EDNS reassembly buffer size.
```
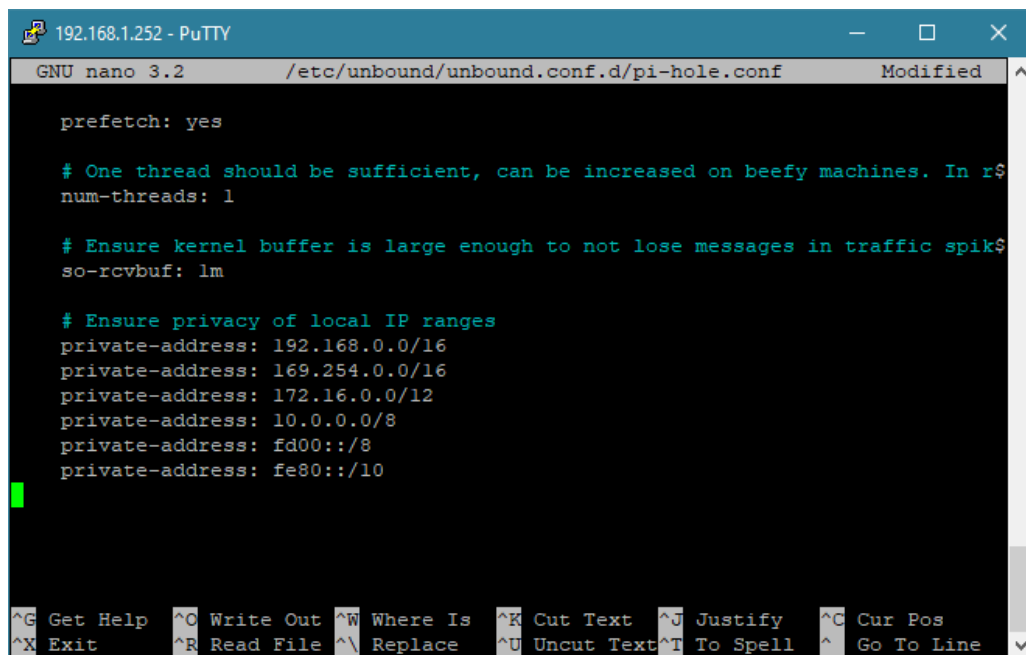
```
    # Suggested by the unbound man page to reduce fragmentation reassembly
problems
    edns-buffer-size: 1472

    # Perform prefetching of close to expired message cache entries
    # This only applies to domains that have been frequently queried
    prefetch: yes

    # One thread should be sufficient, can be increased on beefy machines. In
reality for most users running on small networks or on a single machine, it
should be unnecessary to seek performance enhancement by increasing num-
threads above 1.
    num-threads: 1

    # Ensure kernel buffer is large enough to not lose messages in traffic
spikes
    so-rcvbuf: 1m

    # Ensure privacy of local IP ranges
    private-address: 192.168.0.0/16
    private-address: 169.254.0.0/16
    private-address: 172.16.0.0/12
    private-address: 10.0.0.0/8
    private-address: fd00::/8
    private-address: fe80::/10
```


Once the commands are pasted into the nano editor, type **CTRL+X** to exit nano. It will give you the option to type **Y** for Yes to keep the changes and **Enter** to keep the file name the same.
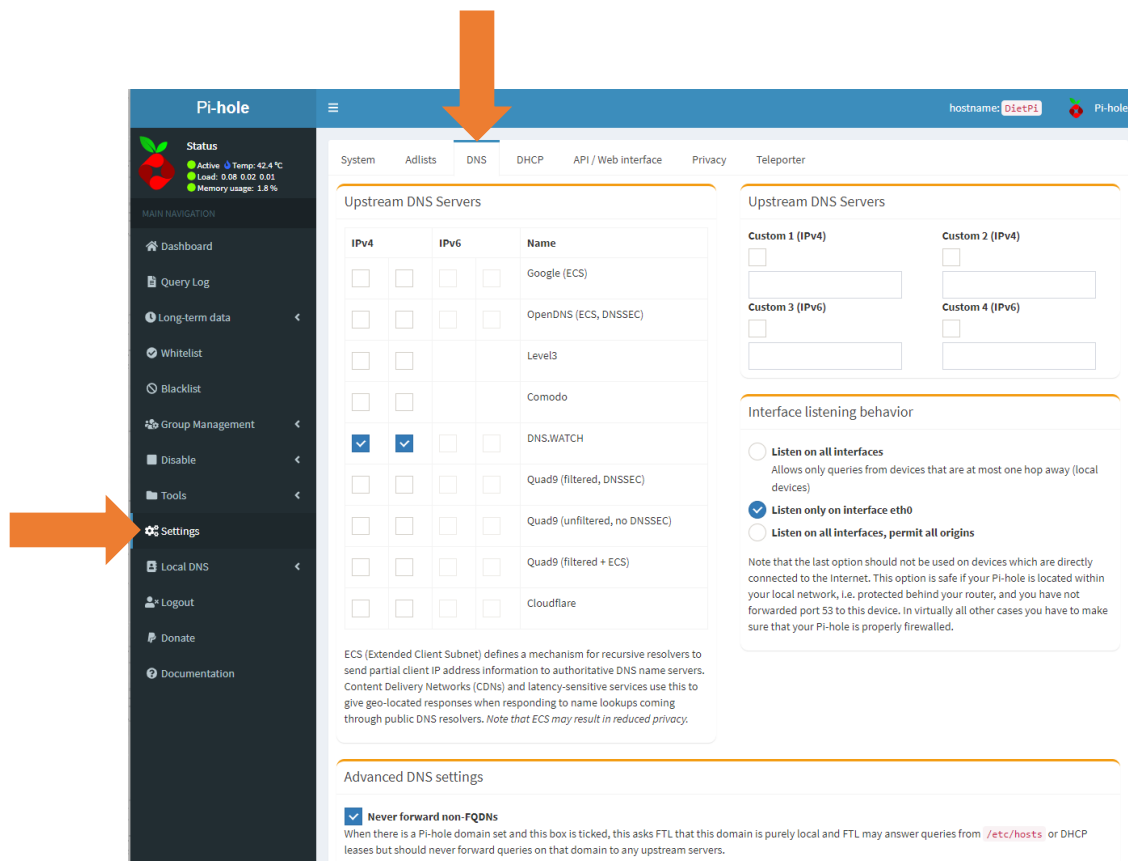
Next, log back into the Pi-hole's web console and go to **Settings** and then the **DNS** submenu. Uncheck whatever pair of boxes were selected earlier in the initial installation for Upstream DNS Servers.
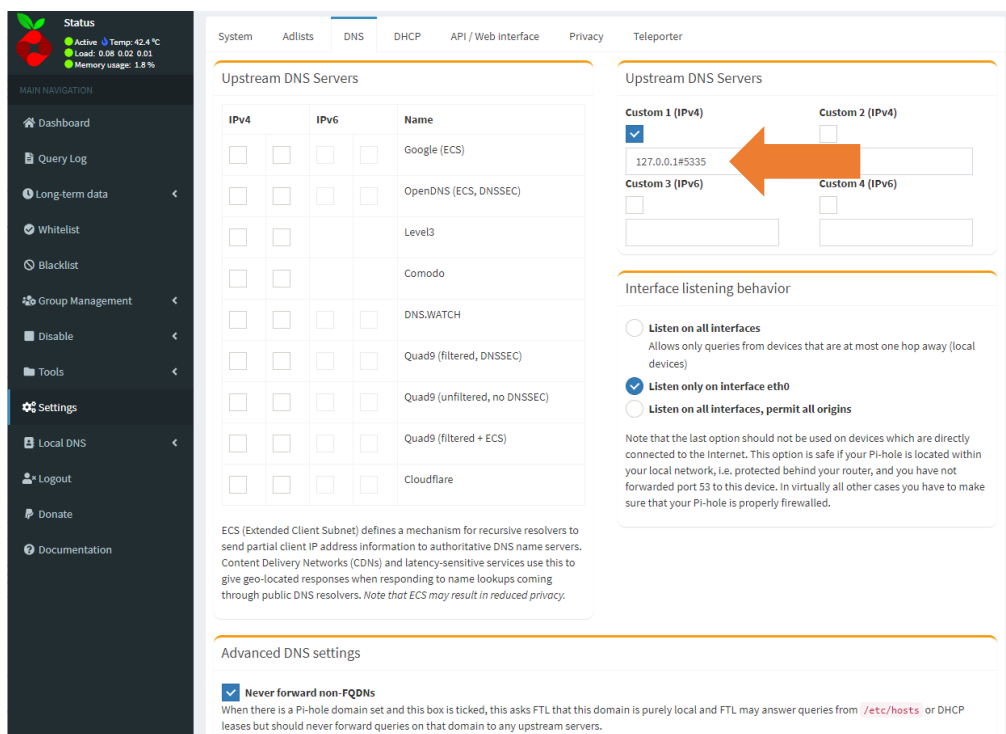
Then, check the box for Custom 1 (IPv4) and enter **127.0.0.1#5335** in the box for the IP Address and port number. Before you leave this screen, make sure to click **Save** at the bottom of the page.

Last, but not least, jump back into Putty and enter the **sudo service unbound restart** command. Wait a minute, then run **dig pi-hole.net @127.0.0.1 -p 5335** to check and see if everything is working. If Putty shows an answer similar to the one above, then you are up and running! Now, you can setup individual machines by pointing to the Pi-hole's IP address as the DNS server. Or if you have your router or DHCP server setup to hand out the Pi-hole's IP address as the DNS server for connecting computers, then you can have filtering available for your entire local network!

Relevant Links:

https://www.youtube.com/watch?v=4X6KYN1cQ1Y – A similar step by step install video. The producer makes a few different choices, but follows a similar process.

https://docs.pi-hole.net/guides/dns/unbound/ - The Pi-hole docs page with more information on unbound.

https://www.youtube.com/watch?v=FnFtWsZ8IP0 – A good walkthrough on how to setup unbound.

Websites with Ad lists to consider:

https://firebog.net/

https://dbl.oisd.nl