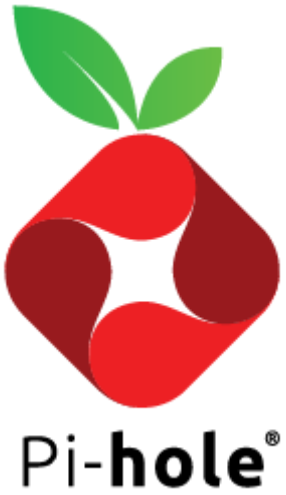
LABORATORIO DE REDES Y SISTEMAS OPERATIVOS



TEMÁTICA A TRABAJAR: Qué es, cómo instalar y puesta en marcha de Pi-Hole

INTEGRANTES:

CARDOZO, Marco

LIOTINE, Cristian

STANLEY, Kevin

ÍNDICE

1. Qué es Pi-Hole

Pi-Hole es una aplicación la cual cumple el rol de bloquear publicidades y rastreos de red a nivel de DNS, en cualquier dispositivo que se encuentre conectado en una red local. Al actuar como un servidor DNS privado, evita que se carguen contenidos no deseados en un dispositivo, el cual lo tenga configurado como DNS primario, ya que este servidor intercepta las solicitudes de dominios utilizados por anunciantes y rastreadores.

2. Requisitos para la instalación

Como Pi-Hole fue diseñado para ser de uso liviano las especificaciones requeridas son laxas.

A nivel hardware solo necesitamos un dispositivo con 512 MB de RAM y 2 GB de espacio en disco. Aunque lo recomendable sea 4 GB.

A nivel software nos basta con que el dispositivo tenga capacidad para ejecutar un sistema operativo basado en Linux como (Ubuntu, Debian, Raspberry Pi, etc.)

A nivel dependencias el instalador de Pi-Hole lo hace automáticamente y en el paso que sigue veremos qué aplicaciones son las que se instalarán.

3. Instalación de Pi-Hole

Pi-Hole nos provee con 4 opciones de instalación.

La primera es el uso de la aplicación curl para hacer la instalación usando un solo comando:

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

Nos puede pasar, que al querer ejecutar este comando, nos diga que no se encuentra la orden “curl”.

```
cristian@cristian-VirtualBox:~$ curl -ssL https://install.pi-hole.net | bash
No se ha encontrado la orden «curl», pero se puede instalar con:
sudo snap install curl  # version 8.1.2, or
sudo apt  install curl  # version 8.5.0-2ubuntu10.4
Consulte «snap info curl» para ver más versiones.
```

Para esto debemos ejecutar el siguiente comando:

sudo apt install curl
Y luego volver a ejecutar el comando de instalación mencionado anteriormente.

Usar el comando “**curl -ssL https://install.pi-hole.net | bash”** puede llegar a ser controversial, ya que estamos haciendo **“piping to bash”.** Esto es básicamente descargar un script y ejecutarlo mediante bash sin revisar los contenidos previamente.

La segunda y tercer opción tienen esto en cuenta y nos proveen con otras formas:

La segunda consiste en usar la aplicación git para clonar el repositorio de Pi-Hole. De esta manera podemos ver lo que se va a ejecutar en nuestro equipo

git clone --depth 1 https://github.com/pi-hole/pi-hole.git Pi-hole
cd "Pi-hole/automated install/"
sudo bash basic-install.sh

La tercer manera solo nos baja el instalador:

wget -O basic-install.sh https://install.pi-hole.net
sudo bash basic-install.sh

La cuarta manera es el uso de Docker para levantar Pi-Hole, opción la cual no desarrollaremos.

3.1 El instalador

Al ejecutar cualquiera de los tres comandos de arriba se nos presentará lo siguiente en la terminal:

Primero el script de instalación hará un chequeo de los permisos root ya que son necesarios para la correcta ejecución del script

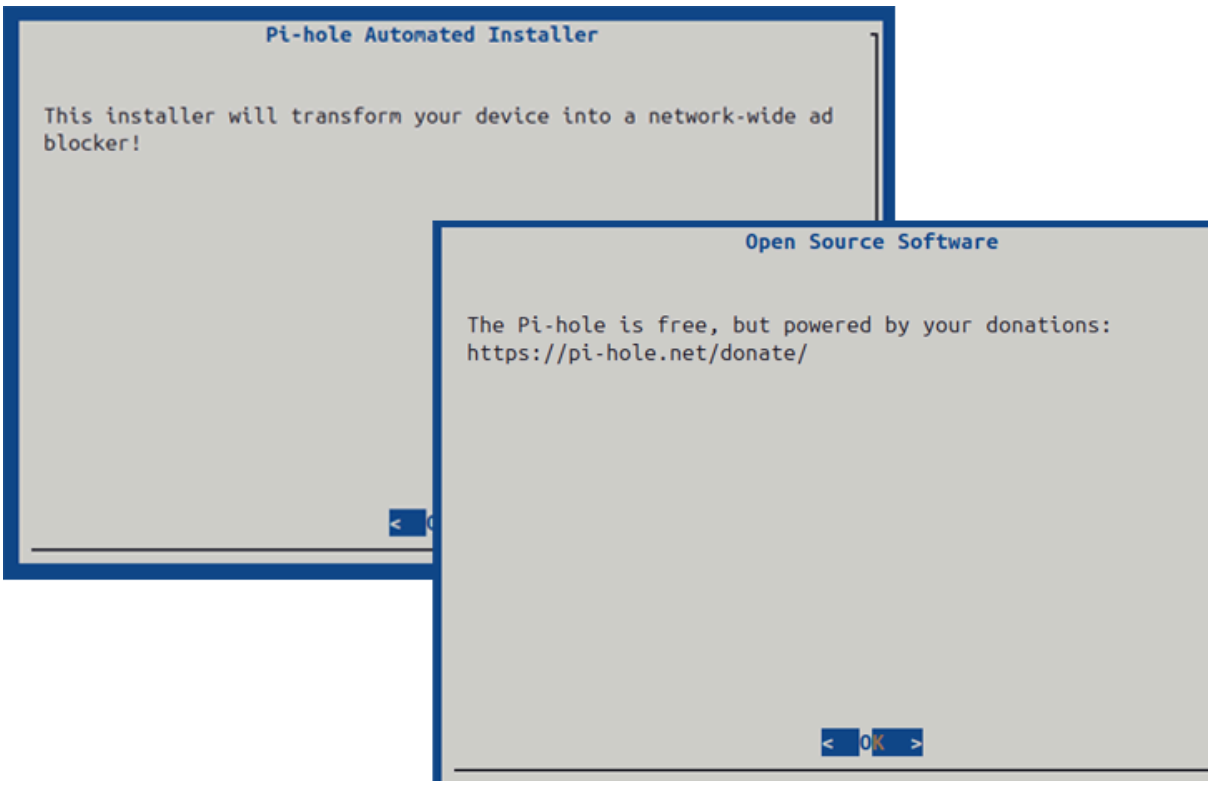
```
marco@marco-VirtualBox:~$ sudo curl -sSL https://install.pi-hole.net | bash

  [i] Root user check
  [i] Script called with non-root privileges
      The Pi-hole requires elevated privileges to install and run
      Please check the installer for any concerns regarding this requirement
      Make sure to download this script from a trusted source

  [✓] Sudo utility check

  [✓] Root user check
```

Luego el instalador descarga las dependencias necesarias para el chequeo de compatibilidad del sistema operativo (grep y dnsutils). Cuando detecte un sistema operativo compatible se instalan las dependencias para correr el script (git, iproute2, dialog, y ca-certificates)

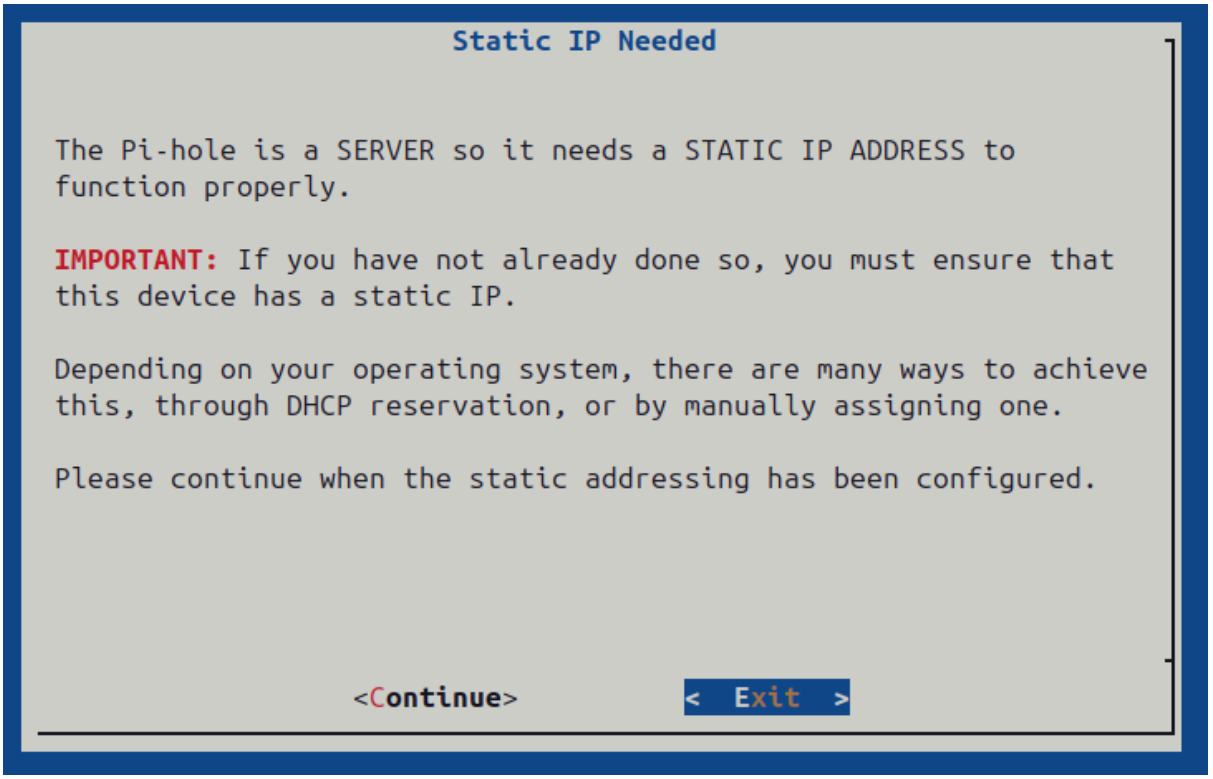
```
[i] SELinux not detected
[✓] Update local cache of available packages

[✓] Checking apt-get for upgraded packages... 86 updates available
[i] It is recommended to update your OS after installing the Pi-hole!

[i] Checking for / installing Required dependencies for OS Check...
[✓] Checking for grep
[✓] Checking for dnsutils

[✓] Supported OS detected
[i] Checking for / installing Required dependencies for this install script...
[✓] Checking for git
[✓] Checking for iproute2
[✓] Checking for dialog
[✓] Checking for ca-certificates
```

Cuando se complete la instalación de dependencias necesarias el instalador abrirá una interfaz gráfica gracias al programa “**dialog”** el cual secuencialmente nos guiará paso por paso en lo que irá haciendo.

Primero se nos dará la bienvenida al programa.

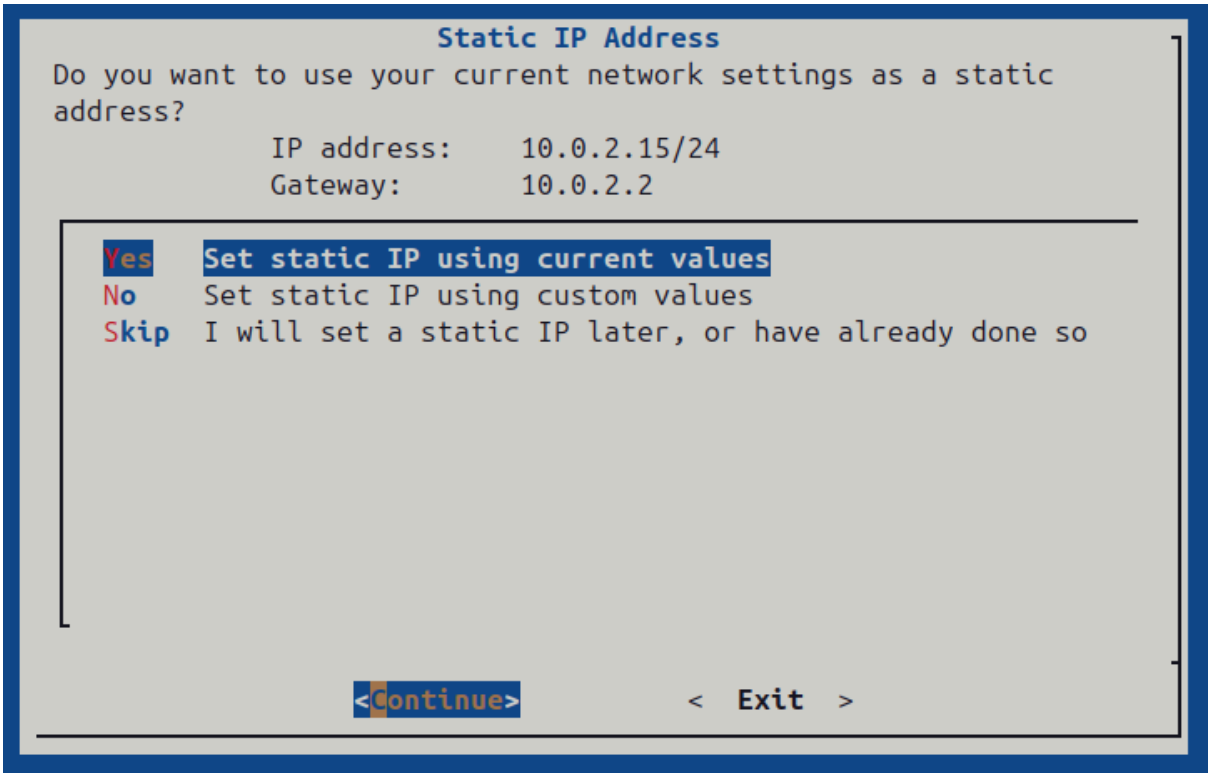


Luego se hará hincapié en un tema importante, la **ip estática**



Como Pi-Hole cumple la función de un servidor DNS, necesitará la configuración para que su ip sea la misma siempre. Nos da la tarea de cerciorarnos de que este dispositivo ya esté configurado con una ip estática antes de continuar. Esto puede hacerse manualmente o mediante configuración del Router con DHCP reservation.

En la siguiente ventana veremos como Pi-Hole nos da opciones para seguir con la instalación luego de haber entendido este inciso.



En este prompt, se nos dan 3 opciones con respecto a la configuración. El instalador el cual ejecuta un comando para obtener nuestra configuración de red nos ofrece:

1) **Setear la ip estática con los valores actuales en nuestra configuración de red.**
2) **Setear la ip estática con valores personalizados**
3) **Setear la ip estática en otro momento o si ya la hemos configurado anteriormente.**

Como estamos usando Ubuntu, cuando seleccionamos el seteo de ips el instalador modifica nuestra configuración de red para que la ip que detecte se convierta en estática.

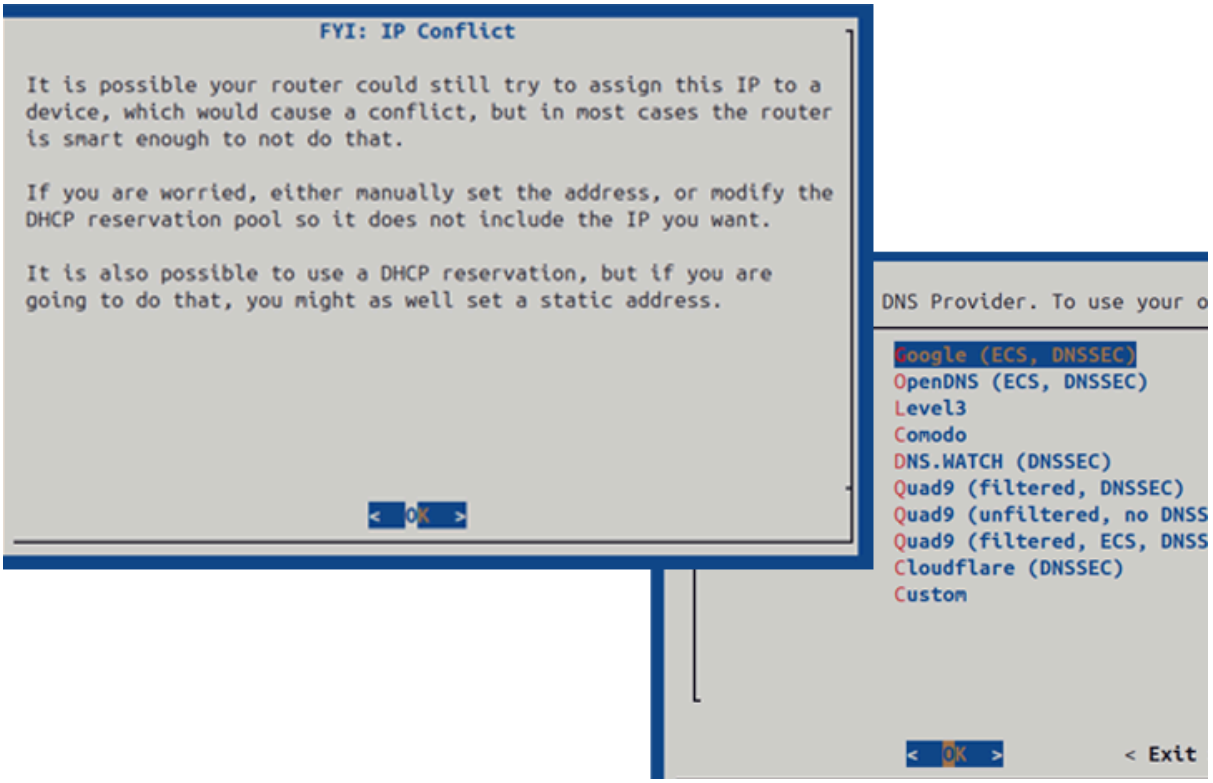
Esto lo hace editando un archivo de configuración del sistema ubicado en **/etc/** llamado **dhcpcd.conf**

```
# Generate SLAAC address using the Hardware Address of the interface
#slaac hwaddr
# OR generate Stable Private IPv6 Addresses based from the DUID
slaac private
interface enp0s3
        static ip_address=10.0.2.15/24
        static routers=10.0.2.2
        static domain_name_servers=
interface enp0s3
        static ip_address=192.168.1.46/24
        static routers=192.168.1.1
        static domain_name_servers=
```

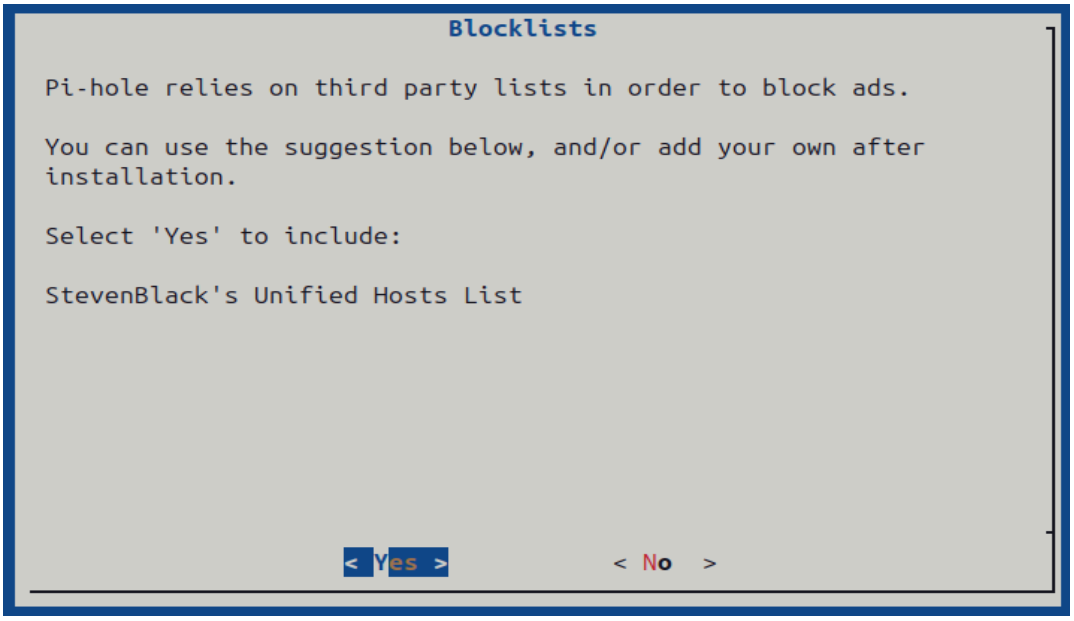
Donde se setea nuestra ip estática y gateway

En las siguiente ventanas se nos da a saber que es posible que nuestro Router todavía puede llegar a asignar esta ip a otro dispositivo lo cual causaría un conflicto, pero aclara que la mayoría de los routers son suficientemente inteligentes como para que no pase eso.
Si no estamos seguros de ir por el camino mencionado anteriormente, podríamos setear la ip manualmente o incluirla en la pool de reservas de DHCP.

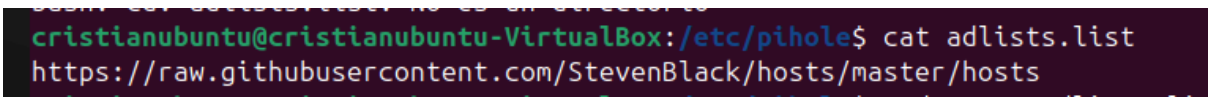
Luego se nos da a elegir un Upstream DNS provider para Pi-Hole que son los servidores por donde pasarán las consultas de dominio que haga Pi-Hole



Luego, en el próximo paso nos dice que pi-hole usa una lista de dominios para bloquear los anuncios que puedan llegar a aparecer.
Podemos usar la lista que nos brinda el mismo instalador “StevenBlack’s” o agregar una nosotros después de la instalación.

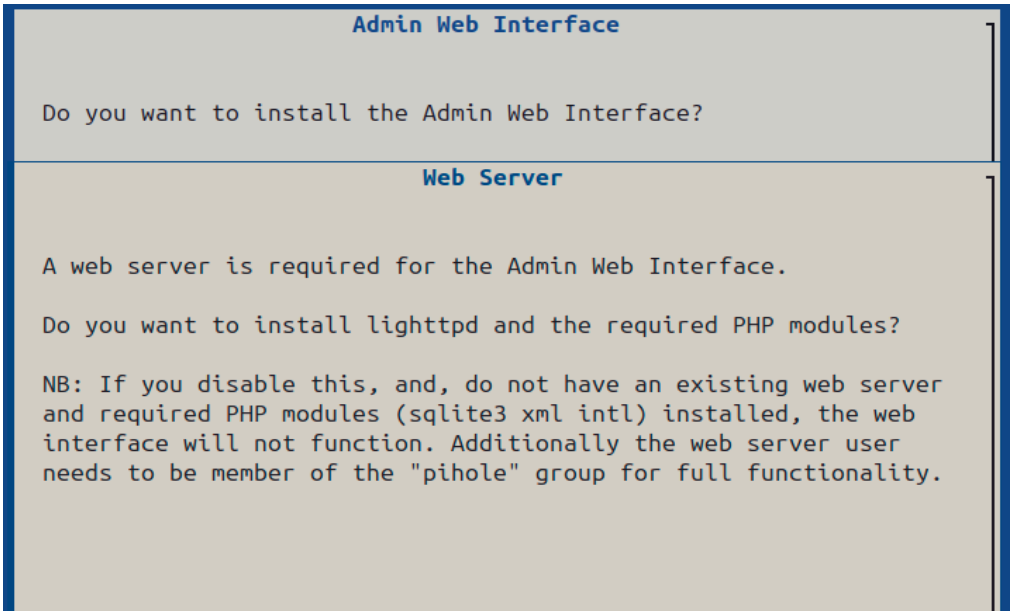


Para que el programa pueda consumir esta lista, la guarda en la ruta **“/etc/pihole/”** en el archivo **“adlists.list”,** en el que se encuentra una url (https://raw.githubusercontent.com/StevenBlack/hosts/master/hosts) que contiene todos estos dominios mencionados anteriormente.

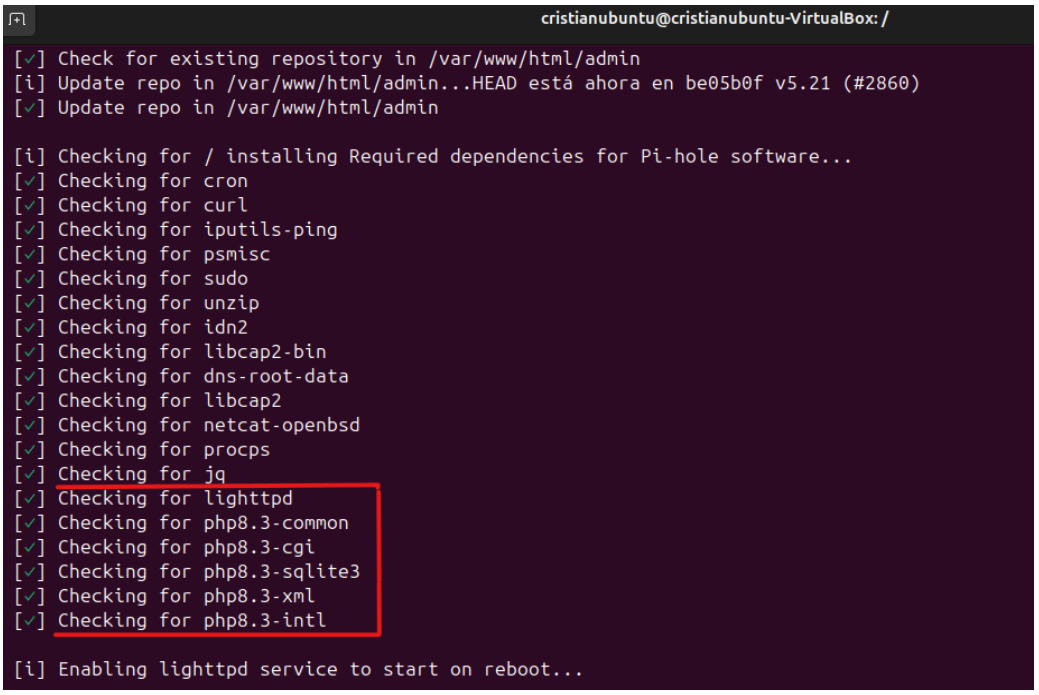


En el siguiente paso nos pregunta si queremos instalar una interfaz para que el usuario pueda utilizar el programa más fácilmente.

En caso que queramos instalarlo, es requerido instalar un servidor web, como **lighttpd** y los **módulos de PHP**. Si no instalamos lo mencionado anteriormente, la interfaz no funcionará.



En la siguiente imagen se muestra la parte en la que se instalan las dependencias mencionadas:

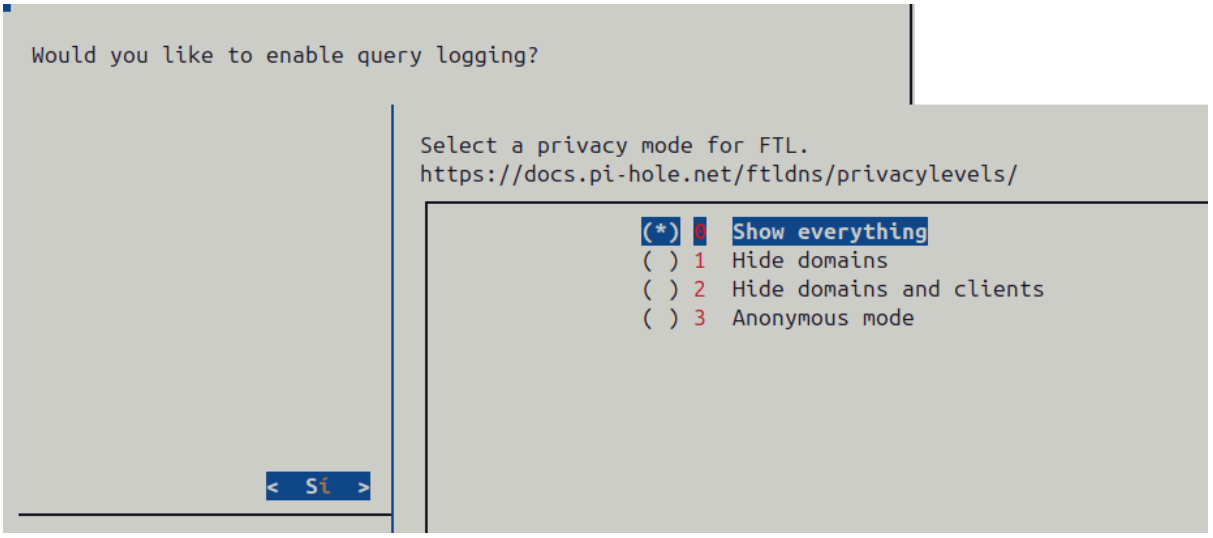


Llegando ya a la última parte de la instalación, nos pregunta si queremos habilitar los query logging, que nos van a servir para ver más detalladamente la información de los bloqueos de los anuncios.

Por otro lado también nos da opciones para seleccionar un modo de privacidad para FTL (Faster Than Light). Este es un componente de Pi-hole que maneja las consultas DNS y el análisis de tráfico en el sistema.

Y podemos seleccionar:

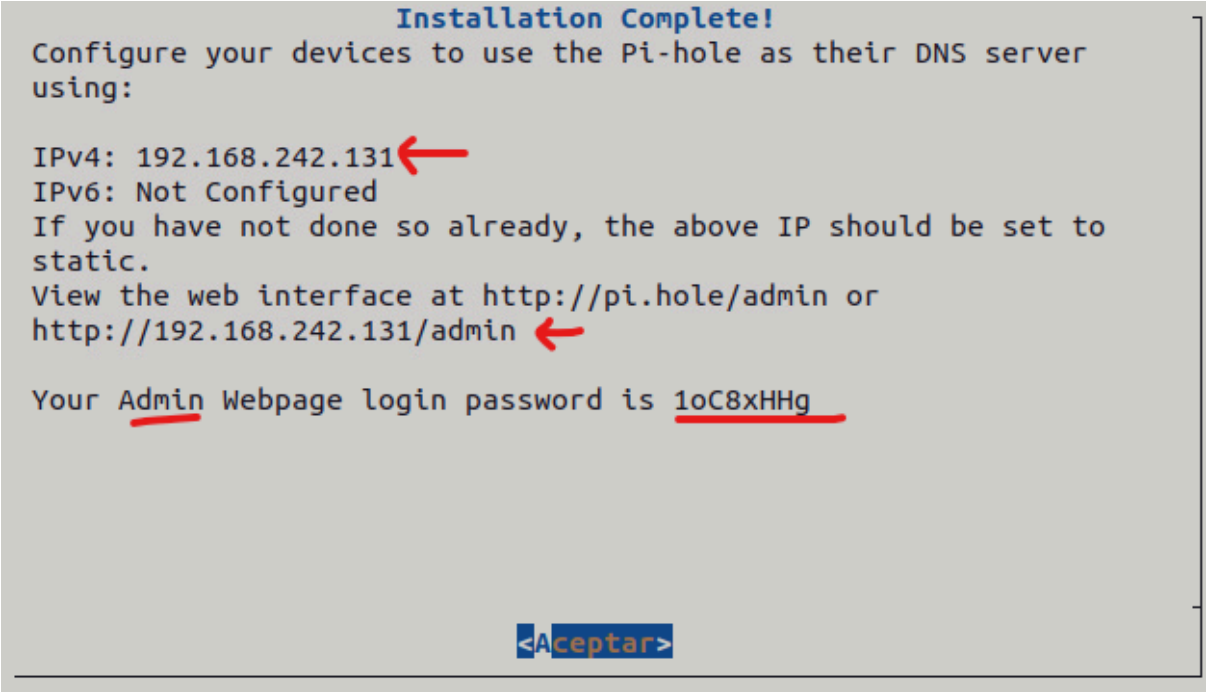
- Mostrar todos los dominios y clientes
- Ocultar los dominios
- Ocultar dominios y clientes
- Modo Anónimo



Aca podemos ver cómo se verían los query loggings en la aplicación:

Time	Type	Domain	Client	Status	Reply	Action
2024-12-03 01:00:00	PTR	220.220.67.208.in-addr.arpa	localhost	OK (answered by dns.sse.cisco.com#53)	DOMAIN (53.2ms)	Blacklist
2024-12-03 01:00:00	PTR	222.222.67.208.in-addr.arpa	localhost	OK (cache)	DOMAIN (0.0ms)	Blacklist
2024-12-03 00:43:34	PTR	220.220.67.208.in-addr.arpa	localhost	OK (answered by dns.sse.cisco.com#53)	DOMAIN (51.3ms)	Blacklist
2024-12-03 00:43:34	PTR	222.222.67.208.in-addr.arpa	localhost	OK (answered by dns.sse.cisco.com#53)	DOMAIN (90.9ms)	Blacklist

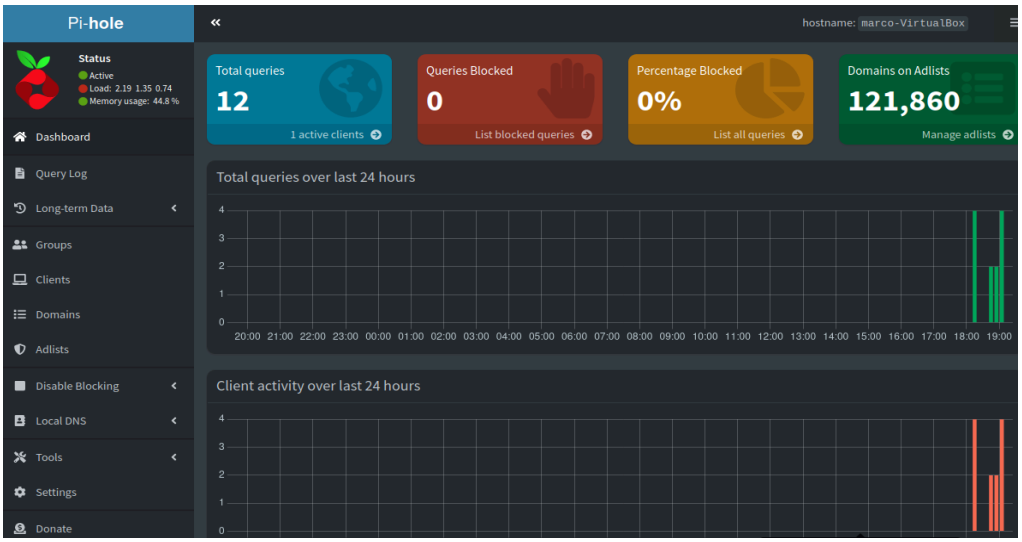
Para saber si la instalación se completó con éxito, deberíamos ver la siguiente pantalla con la IP estática que configuramos, donde también veremos la URL para acceder a la Interfaz Web, el usuario y contraseña del mismo para su ingreso.



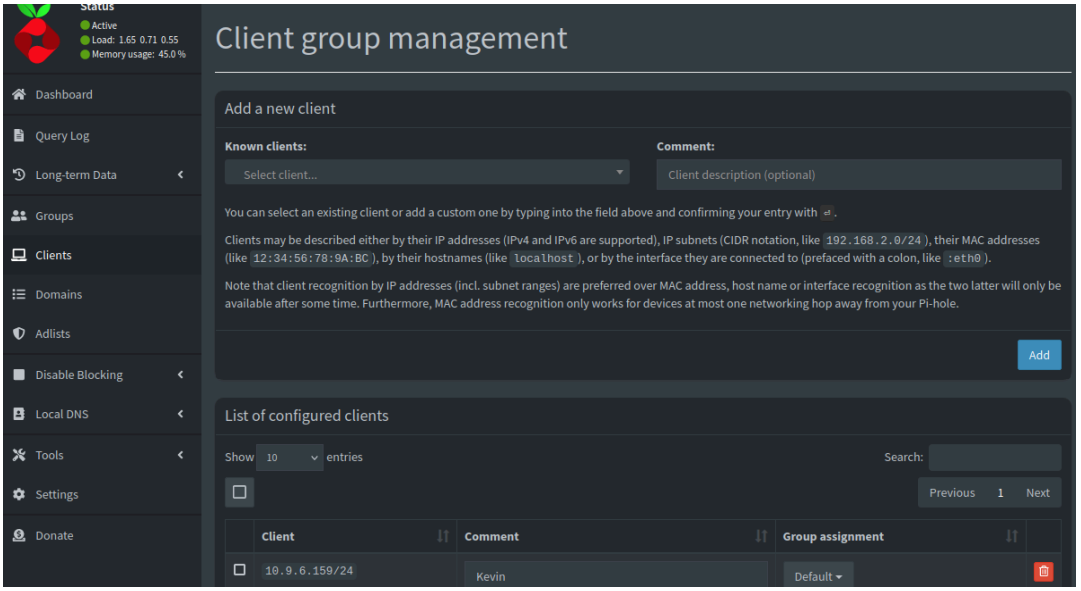
En caso de olvidar su contraseña, puede modificarla mediante el siguiente comando: **pihole -a -p**

3.2 Interfaz web

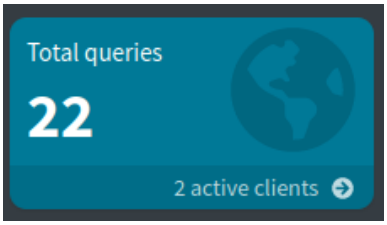
Al terminar la instalación se podrá acceder a la interfaz web de Pi-Hole mediante **localhost/admin** donde se nos pedirá una contraseña la cual nos fue brindada al finalizar la instalación.

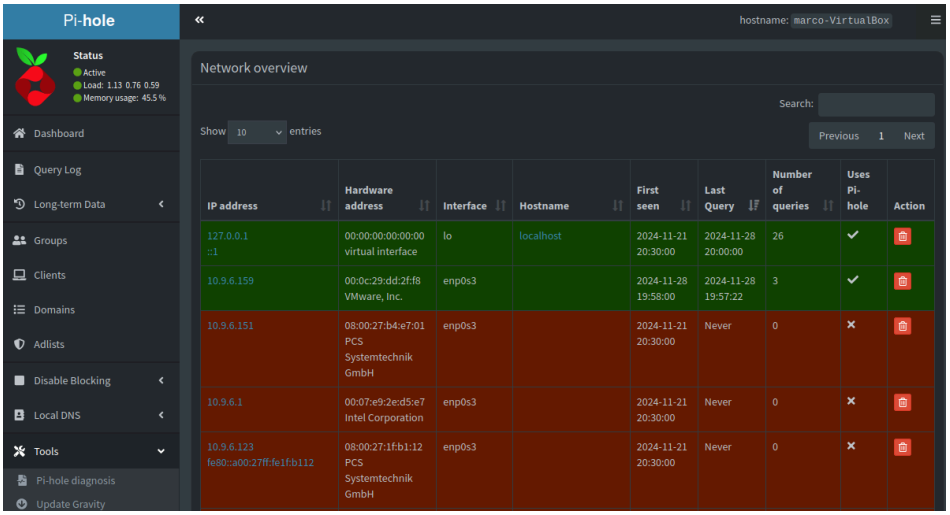


En el apartado "***Clients",*** se pueden añadir los clientes que van a usar el servicio mediante su ip/submascara o su dirección MAC. Esto nos sirve para identificarlos con nombres más amigables, o para ejercer diferentes bloqueos según el cliente. También, nos puede servir para poder identificar y monitorear el tráfico de diferentes dispositivos.

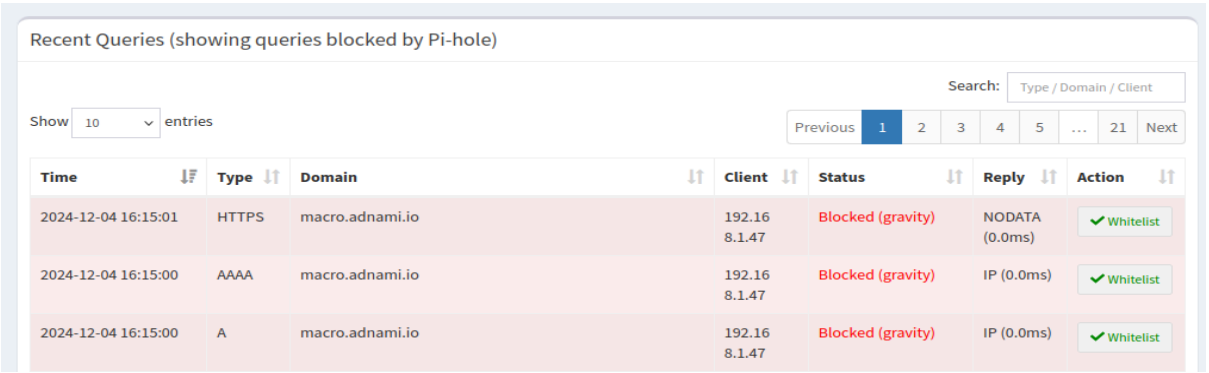


En el apartado “**Total queries**” nos indica la cantidad de clientes conectados en este momento y si ingresamos nos da informaciòn sobre el dispositivo, su número total de queries y si está usando Pi-Hole





En el apartado **“Blocked queries”** nos indica cuales y cuantas queries fueron bloqueadas por un dispositivo en cuestión.

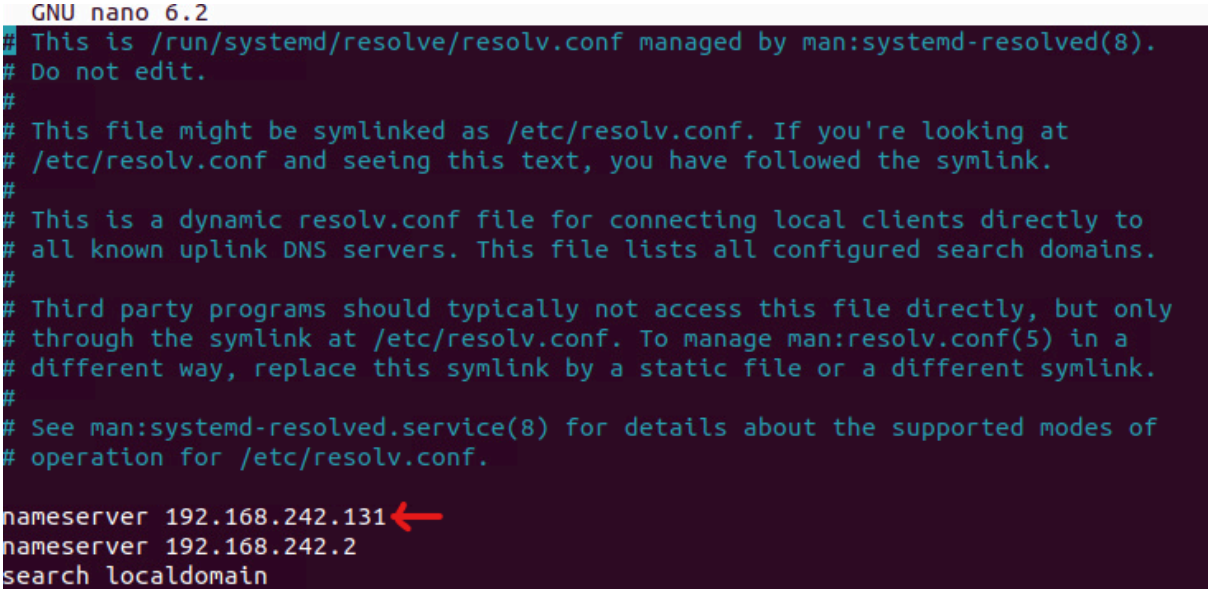


3.3 Usuario

El usuario que desee utilizar Pi-Hole deberá configurar su dispositivo para que use la dirección IP de la máquina, en donde este hosteado Pi-Hole, como su servidor DNS primario.

Para ello realizaremos los siguientes pasos:

Modificar el archivo **resolv.conf** mediante el comando **sudo nano /etc/resolv.conf.**
Ya dentro del archivo, agregar la línea “nameserver” seguido de la IP estática de la pc que tiene levantado Pi-Hole.
Por lo que nos quedaría de la siguiente manera:



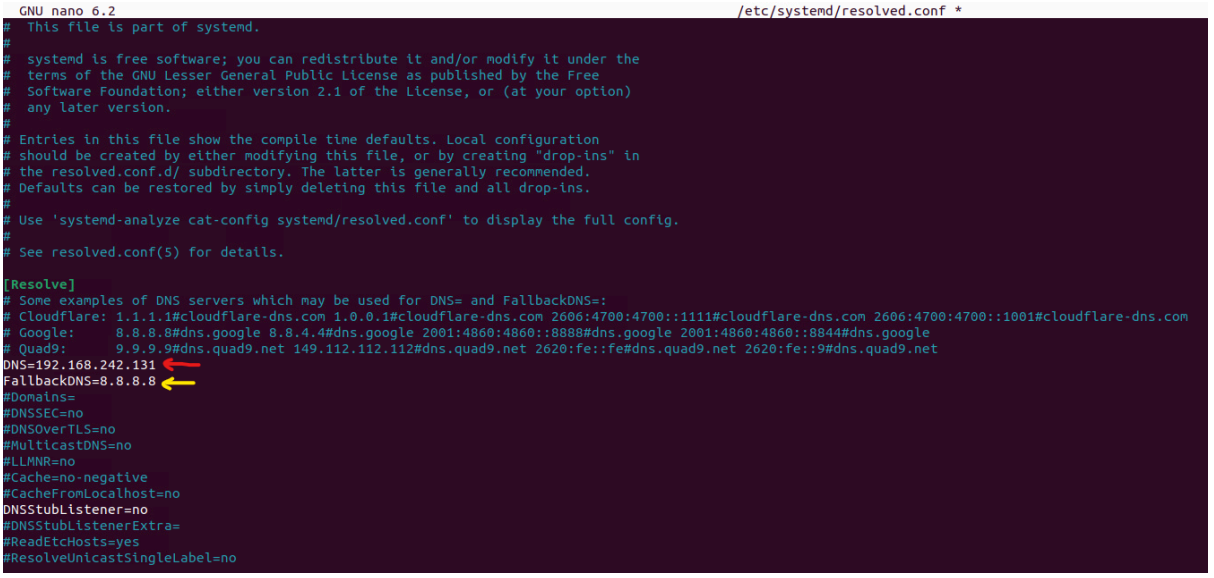
Una vez modificado, guardamos y cerramos el archivo.

Para que esta configuración no se borre debemos modificar el archivo **resolved.conf**. Este se encuentra alojado en **/etc/systemd** y
para ello debemos ejecutar el comando:

sudo nano /etc/systemd/resolved.conf

Habiendo ingresado al archivo debemos descomentar o agregar las líneas
DNS= (IP estática de Pihole)
FallbackDNS= (IP DNS Secundario)

Esta última línea indica la IP del DNS secundario en caso de fallar el primero.

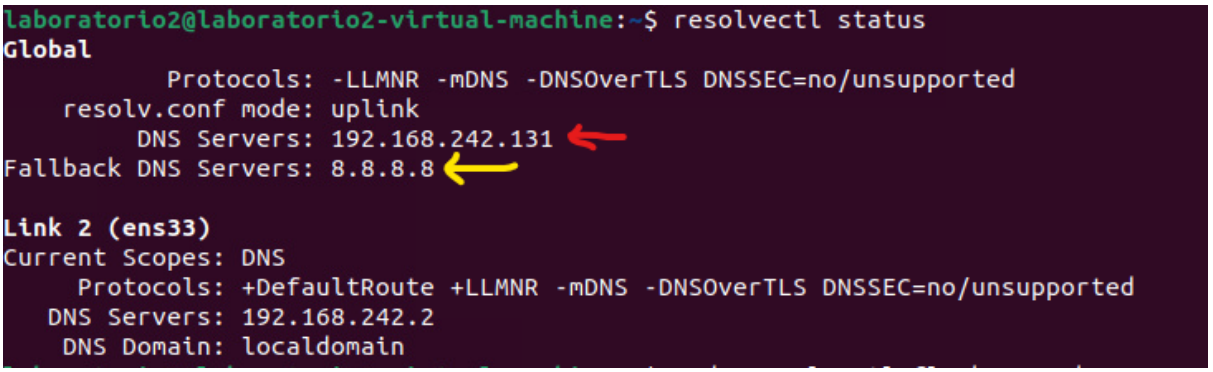


Ya configurado los archivos previos ejecutaremos una secuencia de comandos:

- Reinicia el servicio para aplicar los cambios:
  sudo systemctl restart systemd-resolved

- Verifica que se esté utilizando el Pi-hole como servidor DNS:
  resolvectl status

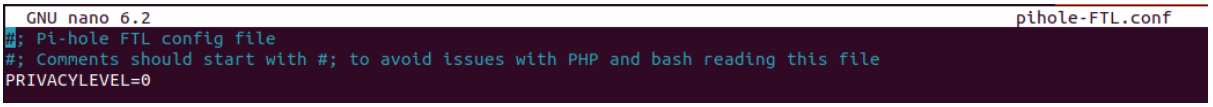
Si todo se configuró correctamente, al ejecutar este último comando veremos algo similar a esto:



3.4 Archivos importantes de Pi-Hole

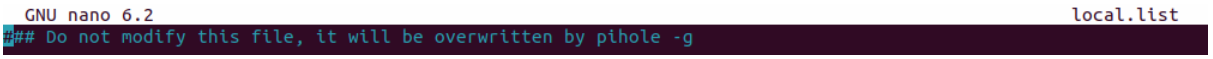
Pi-hole utiliza varios archivos de configuración para definir su comportamiento y personalizar su funcionamiento. A continuación, se presentan algunos de los archivos más importantes:

- **pihole-FTL.conf:** Este archivo se encuentra en **/etc/pihole/** y contiene configuración general para **FTL (Faster Than Light)**, el componente de Pi-hole que maneja las consultas DNS. Se puede editar este archivo para cambiar opciones como la configuración de la base de datos, la dirección del servidor DNS, la configuración de la caché y más.



- **dnsmasq.conf:** Este archivo se encuentra en **/etc/dnsmasq.d/** y contiene configuración para dnsmasq, el servidor DNS que se utiliza en Pi-hole. Se puede editar este archivo para cambiar opciones como la configuración de la zona DNS, la configuración de la caché y más.

- **local.list**: Este archivo se encuentra en **/etc/pihole/** y contiene una lista de dominios locales que Pi-hole debe resolver internamente, sin consultar a un servidor DNS externo. Se puede editar este archivo para agregar o eliminar dominios locales.



- **<u>gravity:</u>** Este archivo se encuentra en **/etc/pihole/** y contiene la lista de dominios bloqueados por Pi-hole. Se puede editar este archivo para agregar o eliminar dominios bloqueados.

Al ser formato DB nano no puede interpretar su contenido correctamente.

- **<u>blocklist:</u>** Este archivo se encuentra en **/etc/pihole/** y contiene la lista de bloques de IP y dominios que Pi-hole debe bloquear. Se puede editar este archivo para agregar o eliminar bloques de IP y dominios.

- **pihole-rc.local:** Este archivo se encuentra en **/etc/** y contiene comandos personalizados que se ejecutan cuando Pi-hole se inicia o se reinicia. Se puede editar este archivo para agregar comandos personalizados, como la configuración de la zona horaria o la activación de servicios adicionales.

Recuerda que, al editar estos archivos, debes hacerlo con cuidado y asegurarte de que la configuración sea válida y no afecte negativamente el funcionamiento de Pi-hole. Además, es recomendable hacer copias de seguridad de los archivos antes de editarlos.

4. Problemas que ocurrieron

4.1 Problema con WordPress

Problema con Pi-Hole al intentar abrir la interfaz Web, se abría WordPress, y para no borrar ninguna configuración se optó por deshabilitar el Servicio de Apache2.

También tuve que reinstalar el Pi-Hole luego de desactivar Apache2

4.2 Problema con Base de datos Pi-hole

Habiendo instalado por primera vez la aplicación, e intentando agregar un cliente nuevo, nos mostró un mensaje de error en el que decía que la base de datos no se había creado. Para arreglar esto, intentamos reinstalar la aplicación, y allí funcionó y no mostró más ese error.
Notamos que la primera vez que lo instalamos habíamos elegido la opción de: “**Setear la ip estática en otro momento**”, y la segunda vez, en la que nos funcionó, elegimos la opción de “**Setear la ip estática con los valores actuales en nuestra configuración de red.**”. Creemos que al elegir esa opción, la base de datos no fue creada debido a que no se había seteado ninguna ip estática.