Project 4 - PI-HOLE with Splunkforwarder

Guide by: Cesar Mendivil

This is a guide on how to create your very own Raspberry Pi add blocker using PI-Hole. There are a few methods to configure PI-HOLE, for instance via Docker running Debian distro. But for this example and tutorial, we will be using a Raspberry Pi 5, which is more beefier that it’s predecessor.

PART 1

Install an Operating System on Your Raspberry Pi

To use your Raspberry Pi, you need to install an operating system (OS). By default, Raspberry Pis look for an OS on any SD card inserted in the SD card slot. Depending on your Raspberry Pi model, you can also boot an OS from other storage devices, such as USB drives, storage connected via a HAT, and network storage.

Requirements for Installing an OS

To install an operating system on a storage device for your Raspberry Pi, you'll need:

- A computer to image the storage device into a boot device
- A way to connect your storage device to that computer

Most Raspberry Pi users choose microSD cards as their boot device.

Recommended Tool: Raspberry Pi Imager

We recommend using Raspberry Pi Imager to install an OS. Raspberry Pi Imager is a tool that helps you download and write images on macOS, Windows, and Linux. It includes many popular operating system images for Raspberry Pi and supports loading images directly from Raspberry Pi or third-party vendors like Ubuntu.

With Raspberry Pi Imager, you can also preconfigure credentials and remote access settings for your Raspberry Pi. It supports images in the .img format as well as container formats like .zip.

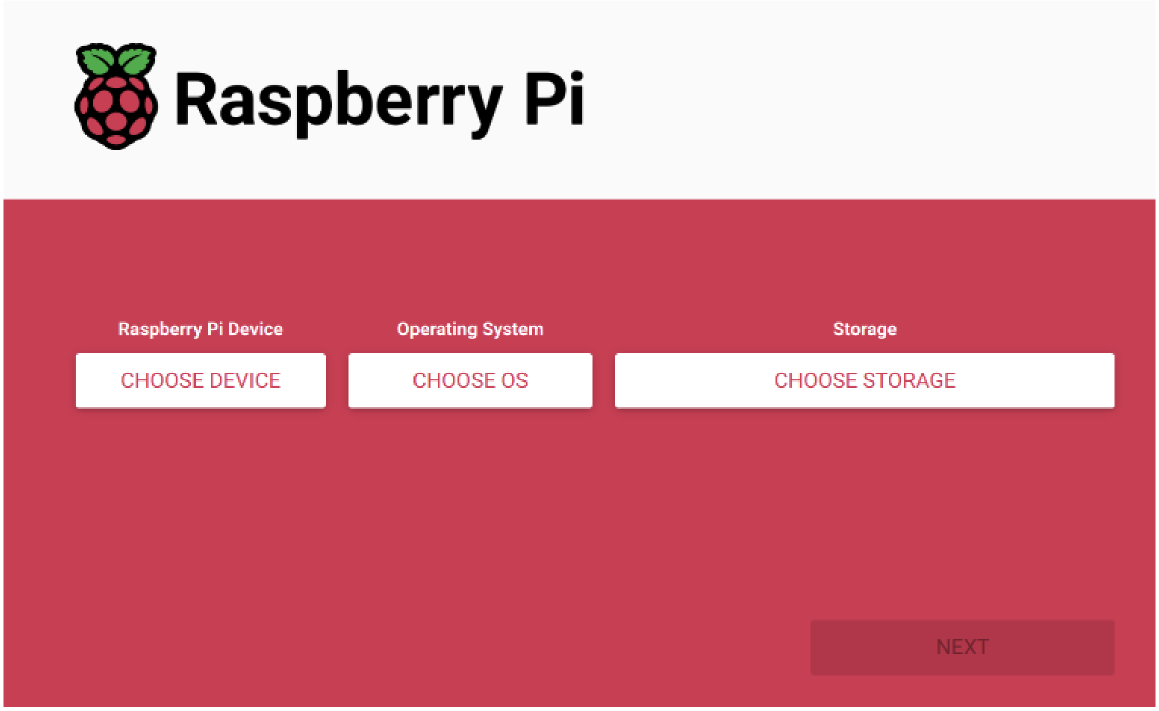
Alternative Installation Method

If you don't have another computer to write an image to a boot device, you may be able to install an operating system directly on your Raspberry Pi from the internet.

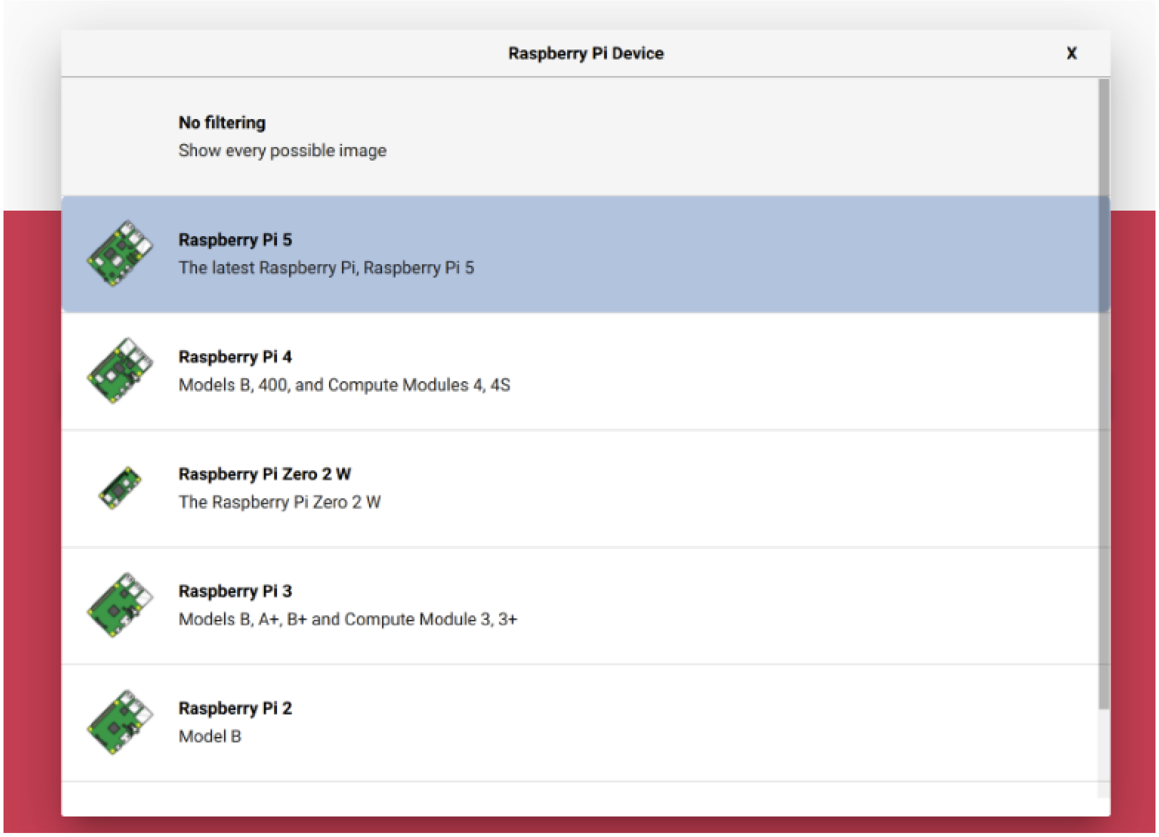
Install using imager

Since we are installing Pihole on a Raspberry Pi, we are going to use an SD card to download the imager we need

1. Download the latest version of the imager software from the following link https://www.raspberrypi.com/software/
2. Plug the SD Card in your PC (we are using Windows 11 for this instance)
3. Run the installer and you should see the following page



4. Choose the device you are using (in this case, we have the Raspberry Pi 5)



No filtering
Show every possible image

Raspberry Pi 5
The latest Raspberry Pi, Raspberry Pi 5

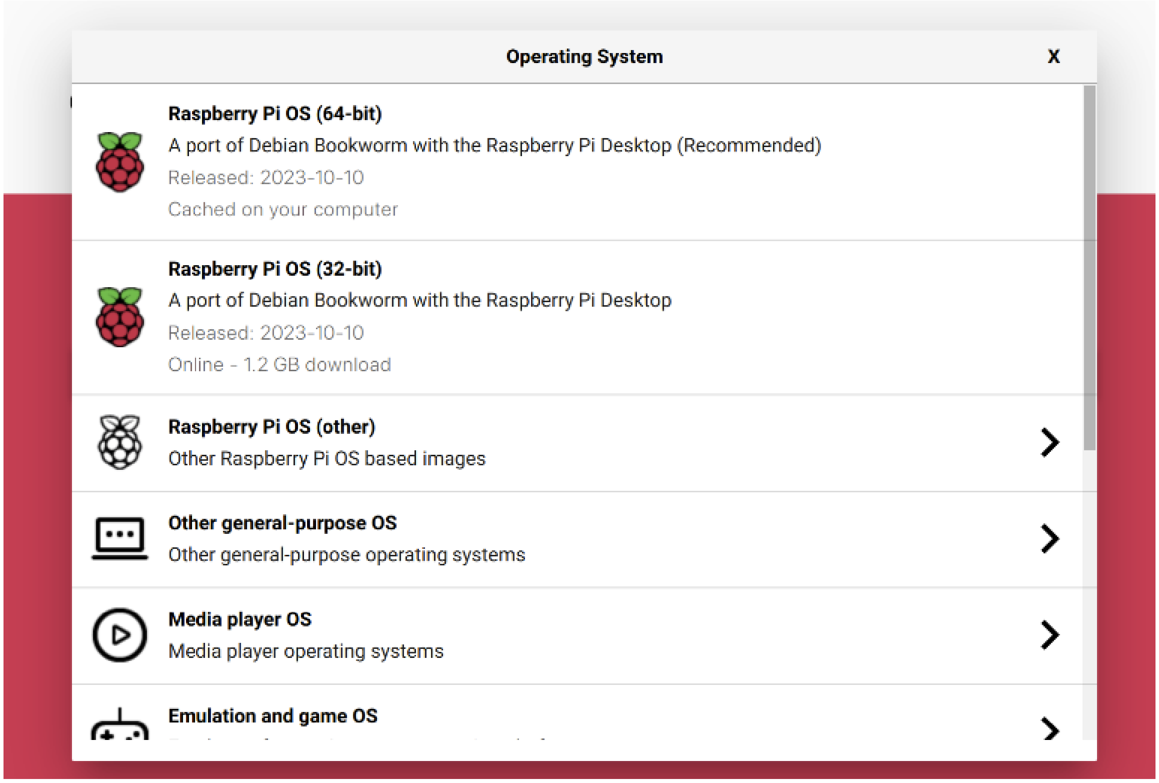
Raspberry Pi 4
Models B, 400, and Compute Modules 4, 4S

Raspberry Pi Zero 2 W
The Raspberry Pi Zero 2 W

Raspberry Pi 3
Models B, A+, B+ and Compute Module 3, 3+

Raspberry Pi 2
Model B

5. Next we will choose the OS. We decided do use the headless lite version, (we will SSH into it later using Termius) Raspberry Pi OS (32-bit)



Operating System X

Raspberry Pi OS (64-bit)
A port of Debian Bookworm with the Raspberry Pi Desktop (Recommended)
Released: 2023-10-10
Cached on your computer

Raspberry Pi OS (32-bit)
A port of Debian Bookworm with the Raspberry Pi Desktop
Released: 2023-10-10
Online - 1.2 GB download

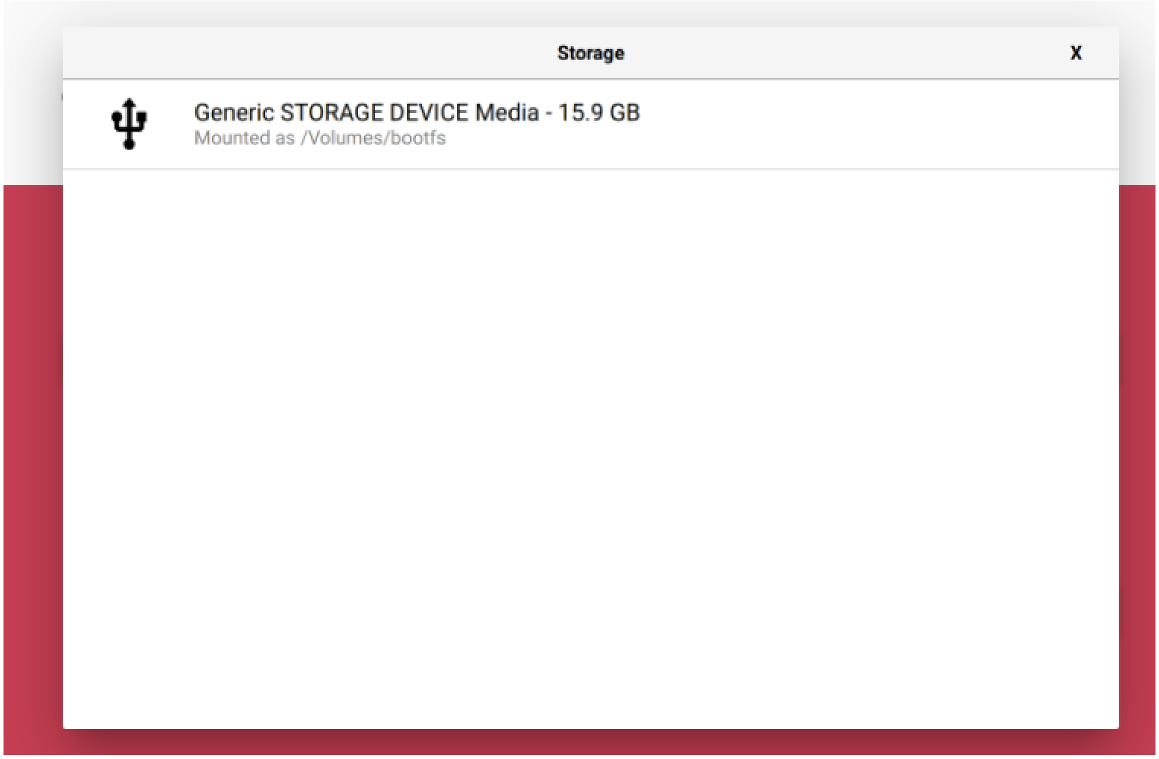
Raspberry Pi OS (other) >
Other Raspberry Pi OS based images

Other general-purpose OS >
Other general-purpose operating systems

Media player OS >
Media player operating systems

Emulation and game OS >

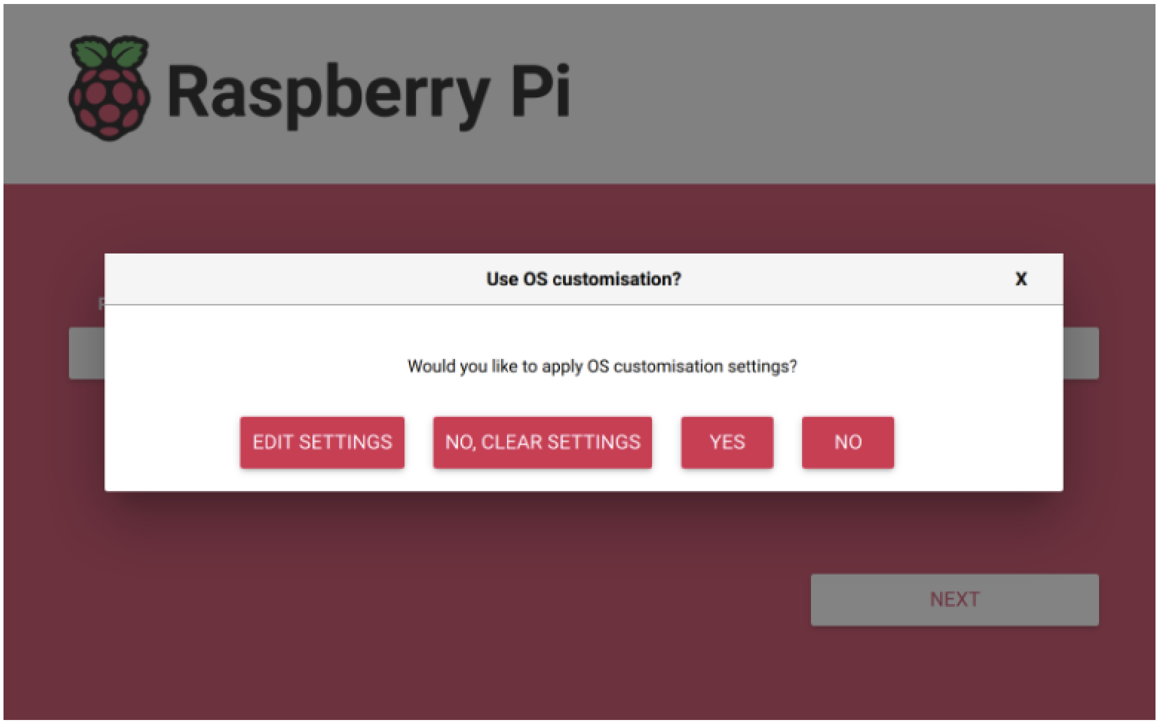
6. Choose the correct storage, (be careful not to write this on another storage you have plugged in)



Storage

Generic STORAGE DEVICE Media - 15.9 GB
Mounted as /Volumes/bootfs

7. Click Next, and you'll arrive at the following screen



Use OS customisation?

Would you like to apply OS customisation settings?

8. Click on “Edit Settings” and configure the following (this is highly recommended so you do not have to configure it through the Linux interface manually, later):

OS Customization

The OS customization menu allows you to set up your Raspberry Pi before its first boot. You can preconfigure the following settings:

- Username and password
- Wi-Fi credentials
- Device hostname
- Time zone
- Keyboard layout
- Remote connectivity

Initial Setup

When you first open the OS customization menu, you might see a prompt asking for permission to load Wi-Fi credentials from your host computer. If you respond "yes," Imager will prefill Wi-Fi credentials from your current network. If you respond "no," you can enter Wi-Fi credentials manually.

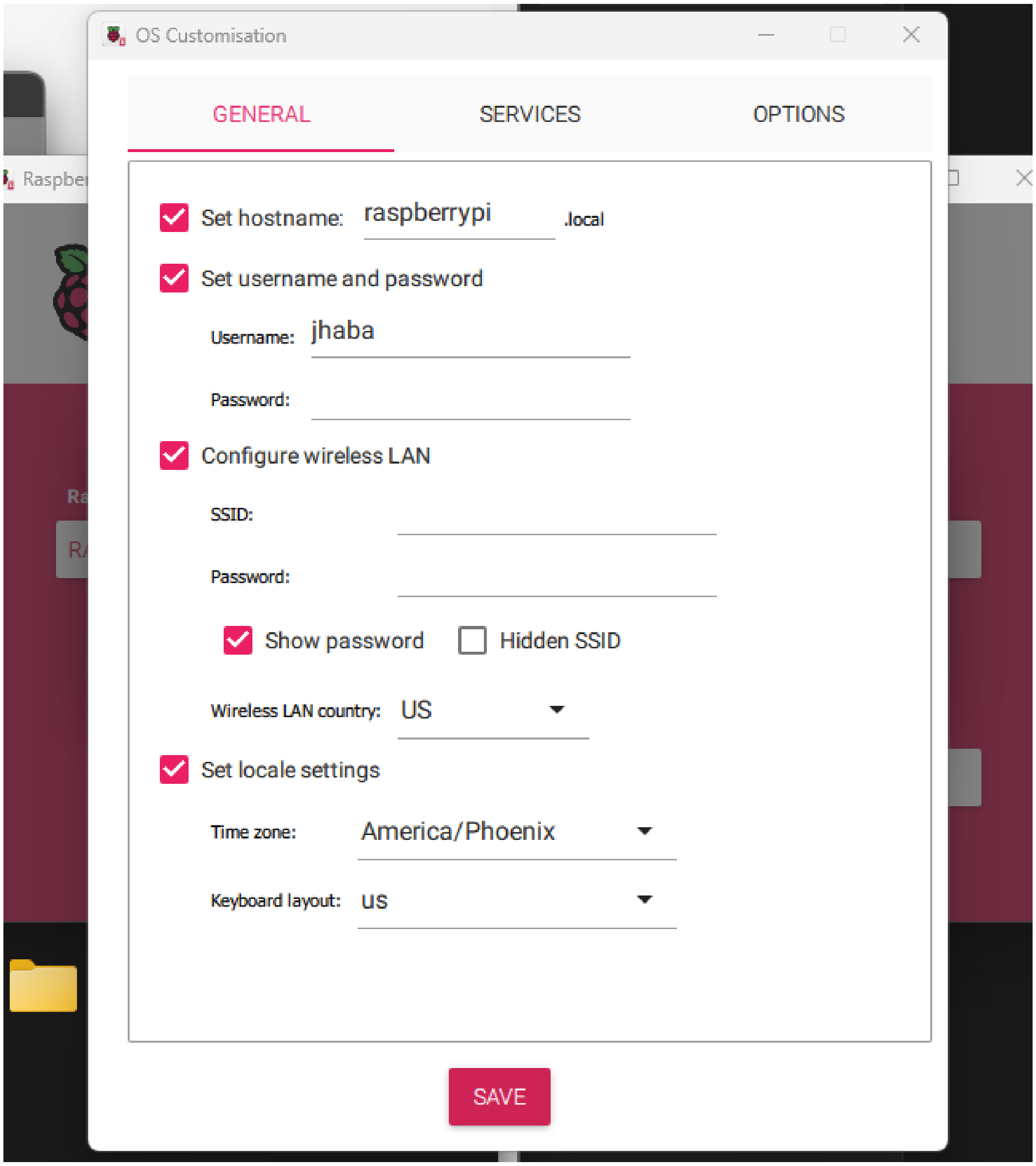
Configuration Options

- **Hostname:** Defines the hostname your Raspberry Pi broadcasts to the network using mDNS. Other devices on the network can communicate with your Pi using `<hostname>.local` or `<hostname>.lan`.
- **Username and Password:** Sets the admin user account credentials for your Raspberry Pi.
- **Wireless LAN:** Allows you to enter the SSID and password for your wireless network. If your network does not broadcast its SSID, enable the "Hidden SSID" setting. Imager uses your current country as the "Wireless LAN country" by default, which controls the Wi-Fi broadcast frequencies. Enter wireless LAN credentials if you plan to run a headless Raspberry Pi.

- **Locale Settings:** Defines the time zone and default keyboard layout for your Pi.

General Settings and Remote Connectivity

- **Services Tab:** Includes settings to help you connect to your Raspberry Pi remotely.
    - **Enable SSH:** If you plan to use your Raspberry Pi remotely, check this box. This is especially important for a headless setup.
    - **Password Authentication:** Allows SSH access using the username and password provided in the general tab of OS customization.
    - **Public-Key Authentication:** Preconfigures your Raspberry Pi for passwordless SSH authentication using a private key from your current computer. If you have an RSA key in your SSH configuration, Imager uses that public key. If not, you can click "Run SSH-keygen" to generate a new key pair, and Imager will use the new public key.



☑ Set hostname: raspberrypi .local

☑ Set username and password

Username: jhaba

Password:

☑ Configure wireless LAN

SSID:

Password:

☑ Show password ☐ Hidden SSID

Wireless LAN country: US

☑ Set locale settings

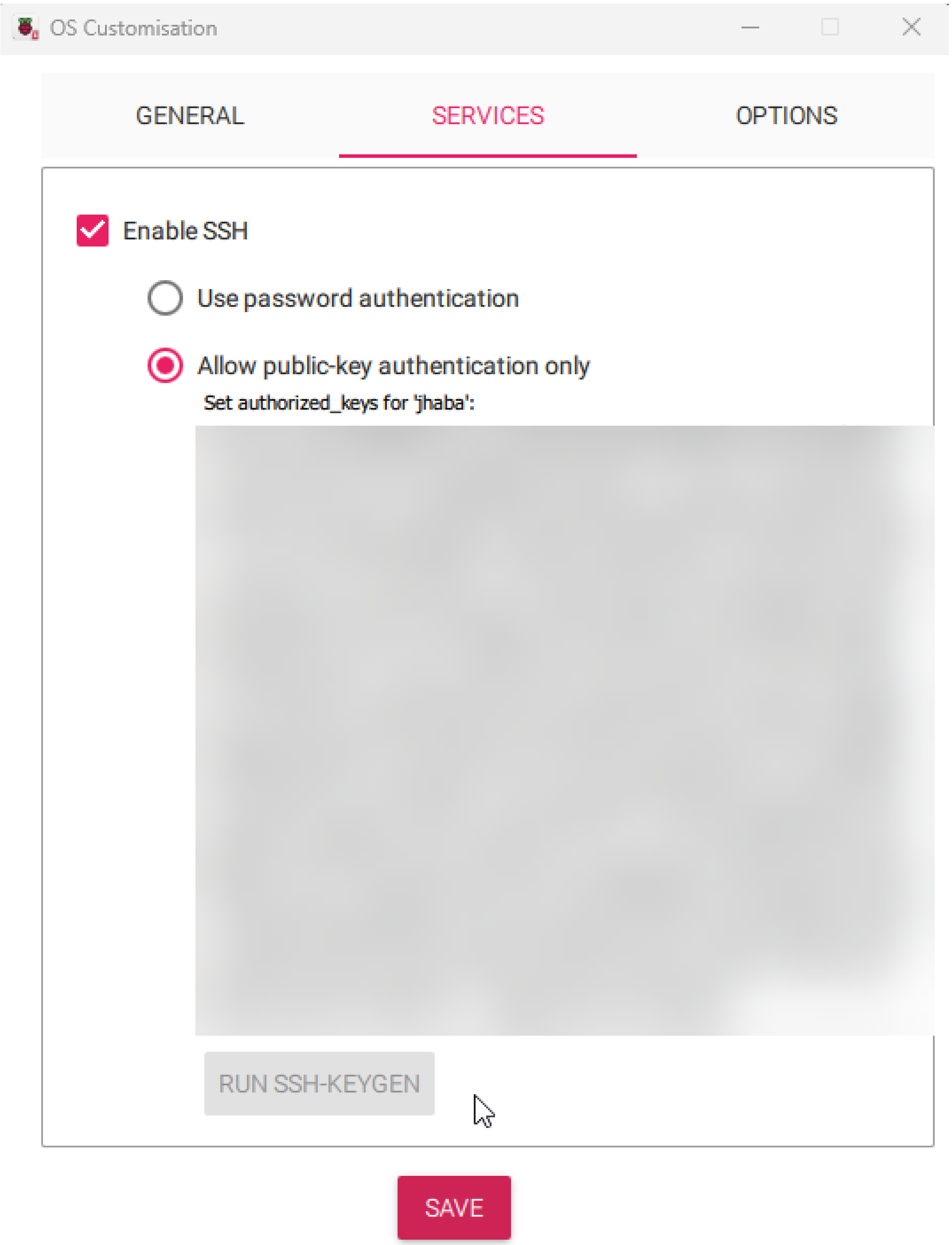
Time zone: America/Phoenix

Keyboard layout: us

9. The **Services** tab includes options to help you connect to your Raspberry Pi remotely.

- **Enable SSH:** If you plan to access your Raspberry Pi over the network, check this box. This is essential for headless setups.
- **Password Authentication:** Select this option to SSH into your Raspberry Pi using the username and password provided in the general tab of OS customization.
- **Public-Key Authentication:** Select this option to set up passwordless SSH authentication using a private key from your current computer. If you

already have an RSA key in your SSH configuration, Imager will use that public key. If not, you can click "Run SSH-keygen" to generate a new key pair, and Imager will use the newly generated public key.



☑ Enable SSH

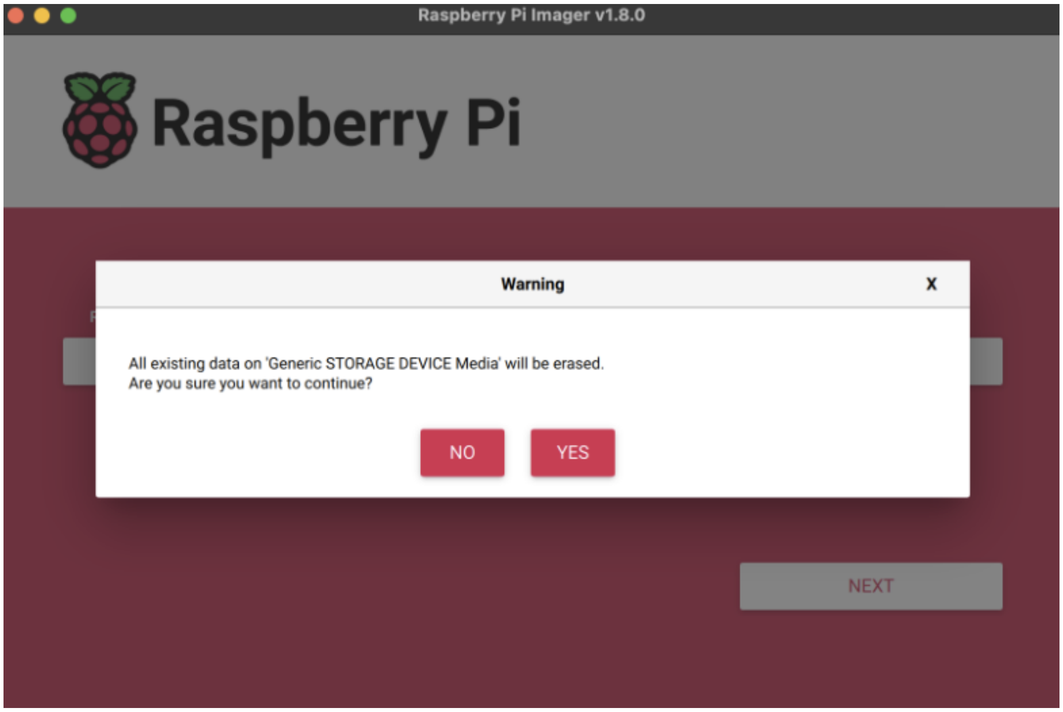
○ Use password authentication

● Allow public-key authentication only

Set authorized_keys for 'jhaba':

[redacted]

10. **Save Settings:** Once you have finished entering your OS customization settings, click **Save**.
11. **Apply Settings:** Click **Yes** to apply the customization settings when writing the image to the storage device.
12. **Confirm and Write:** Respond **Yes** to the "Are you sure you want to continue?" prompt to begin writing data to the storage device.



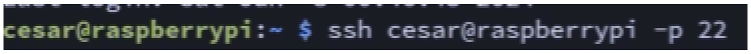
All existing data on 'Generic STORAGE DEVICE Media' will be erased.
Are you sure you want to continue?

Part 2

Connect via SSH

1. **Open a Terminal:**
   - **Windows Users:** You may need to install a terminal application. We suggest using https://termius.com/.

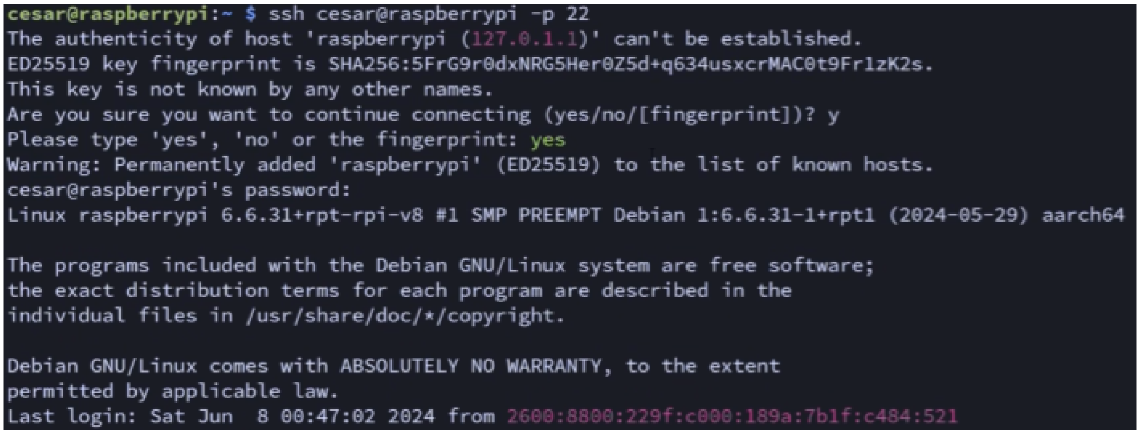
- **Command:** Enter the following command to connect to your Raspberry Pi, replacing `<username>` with your chosen username from Imager:



```
cesar@raspberrypi:~ $ ssh cesar@raspberrypi -p 22
```

2.

- **Connection Confirmation:**
  - If prompted with "Are you sure you want to continue connecting?", type `yes`.
  - Enter the password you set during the advanced configuration when prompted.
- **Successful Connection:**
  - You will know you've connected successfully when you see the following prompt with your configured username and hostname:

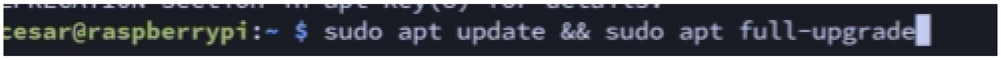


```
cesar@raspberrypi:~ $ ssh cesar@raspberrypi -p 22
The authenticity of host 'raspberrypi (127.0.1.1)' can't be established.
ED25519 key fingerprint is SHA256:5FrG9r0dxNRG5Her0Z5d+q634usxcrMAC0t9Fr1zK2s.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added 'raspberrypi' (ED25519) to the list of known hosts.
cesar@raspberrypi's password:
Linux raspberrypi 6.6.31+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.31-1+rpt1 (2024-05-29) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Jun  8 00:47:02 2024 from 2600:8800:229f:c000:189a:7b1f:c484:521
```

3.

Update Packages:

- Run the following commands to ensure all your packages are up to date:



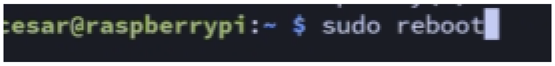
```
cesar@raspberrypi:~ $ sudo apt update && sudo apt full-upgrade
```

4.

Reboot Raspberry Pi:

- After the updates, reboot your Raspberry Pi to apply all changes:

Note: Running this command will disconnect your SSH session. Wait a few seconds for your Raspberry Pi to reboot, then reconnect using the SSH command:



```
cesar@raspberrypi:~ $ sudo reboot
```

Part 3

Install Pi-hole

1. **Run Setup Script:**

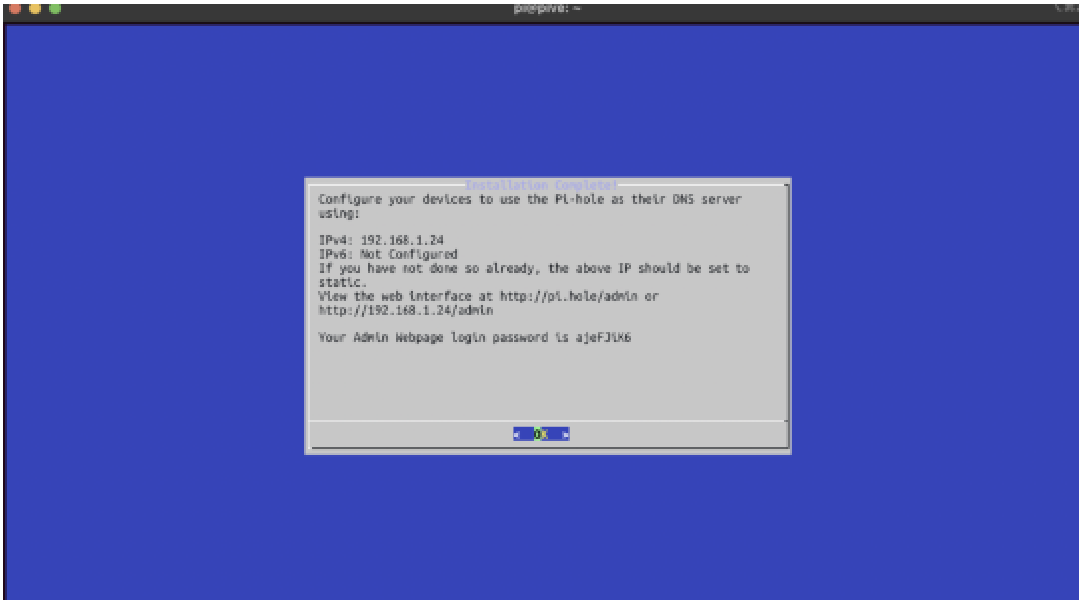
   - Execute the following command to run the Pi-hole setup script:

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

2. **Setup Tips:**

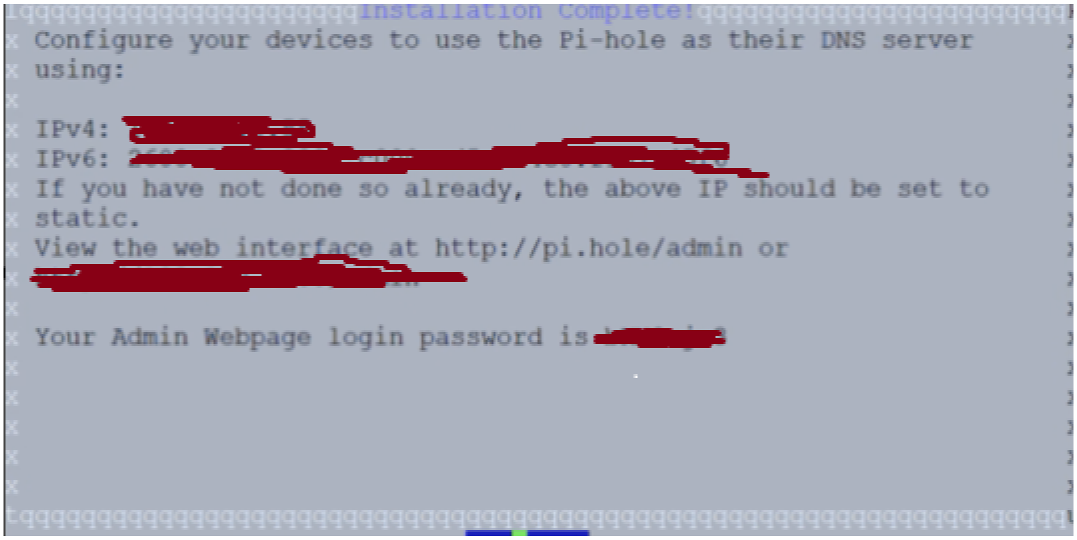
   - **Static IP Address Warning:** Click Continue when warned about needing a static IP address; this will be addressed later.
   - **Interface Selection:** Choose `wlan0` to use your Raspberry Pi’s Wi-Fi connection.
   - **Upstream DNS Provider:** Select OpenDNS.
   - **Additional Features:**
     - Include StevenBlack’s Unified Hosts List.
     - Install the Admin Web Interface.
     - Install `lighttpd` and the necessary PHP modules to run the Admin Web Interface.

- **Logging and Privacy:**
    - Enable query logging.
    - Select Anonymous mode for privacy level.



```
Installation Complete!
Configure your devices to use the Pi-hole as their DNS server
using:

IPv4: 192.168.1.24
IPv6: Not Configured
If you have not done so already, the above IP should be set to
static.
View the web interface at http://pi.hole/admin or
http://192.168.1.24/admin

Your Admin Webpage login password is ajeFJiK6
```

3. **Completion:**
    - When you see "Installation complete!", the setup is finished. This screen will display the IP address of your Pi-hole, a link to the admin interface, and your administrator password.
    - **Save Credentials:**
        - Record the administrator password in a secure place, such as a password manager.
        - Save the IP address for configuring a static IP address shortly.



```
Installation Complete!
Configure your devices to use the Pi-hole as their DNS server
using:

IPv4: [redacted]
IPv6: [redacted]
If you have not done so already, the above IP should be set to
static.
View the web interface at http://pi.hole/admin or
[redacted]

Your Admin Webpage login password is [redacted]
```

4. **Access Admin Interface:**
    - Note that Pi-hole provides a single administrator account without a username.
    - Open the admin interface by pressing the Control key (Command on macOS) and clicking the link using the IP address (e.g., `http://192.168.1.24/admin`). Avoid using the pi.hole domain link until Pi-hole is configured as your DNS provider.
    - Alternatively, copy and paste the link into your browser if Control + click doesn't work in your terminal.
    - Authenticate using the admin password from the setup script output.
    - Bookmark the Pi-hole admin console for future access and maintenance.

Part 4

Configure Your Network to Use Your Pi-hole

WARNING: The following tasks require changes to your wireless network's global settings, which might temporarily disrupt your internet connection. Proceed with caution.

Access Your Router's Admin Interface

1. **Find the Router's IP Address:**
   - Run this command on your Raspberry Pi to obtain the IP address of your router:

     ```
     $ nmcli -f IP4.GATEWAY device show wlan0
     ```

   - Alternatively, check for a sticker on your router labeled "admin URL" or similar.
2. **Log In:**
   - Enter the IP address (sometimes with the suffix /admin) into your browser.
   - Use the username and password found on your router’s sticker or provided by your ISP.

Options for Using Pi-hole to Block Ads

1. **Configure Pi-hole as the DNS Server for Your Network:**
   - Assign your Raspberry Pi a static IP address from your router’s interface.
   - Point your router’s DNS server settings to Pi-hole’s static IP address.
2. **Configure Pi-hole as the DHCP Provider for Your Network:**
   - If your router doesn't support static IP addresses or DNS servers, change the DHCP server in your router settings to Pi-hole.
3. **Manually Point Devices to Pi-hole for DNS:**
   - Individually configure each device to use Pi-hole as its DNS server in their network settings.

Assign Your Raspberry Pi a Static IP Address

1. **Get the Current IP Address:**
   - Run:

```
$ hostname -I
```

- Note the current IP address (e.g., `192.168.1.24`).

2. **Get the MAC Address:**
    - Run:

    ```
    $ nmcli -f GENERAL.HWADDR device show wlan0
    ```

    - Note the MAC address (e.g., `A8:42:EA:58:E0:1C`).

3. **Configure a Static IP:**
    - In your router’s admin interface, set a static IP for your Raspberry Pi using its MAC address.
    - This setting might be under "Advanced" or "DHCP Reservations".

Set Pi-hole as Your Network’s Default DNS Server

1. **Access DNS Settings:**
    - In your router’s admin interface, look for DNS settings under sections like "Internet," "DHCP," or "Internet Connection."

2. **Enter Pi-hole’s IP Address:**
    - Input Pi-hole’s static IP address in the DNS fields.
    - Avoid adding any secondary DNS servers after Pi-hole’s IP to prevent bypassing ad-blocking.

Configure Pi-hole as the DHCP Provider

1. **Enable DHCP in Pi-hole:**
    - Go to Pi-hole admin console (http://<Pi-hole-IP>/admin).
    - Navigate to "Settings" > "DHCP" and check "DHCP server enabled."
    - Save the settings.

2. **Update Router Settings:**

- Set your router to delegate DHCP tasks to Pi-hole.

Manually Point Devices to Pi-hole for DNS

1. **Adjust DNS Settings on Devices:**
    - In device Wi-Fi preferences, under "Advanced" settings, set Pi-hole's IP as the DNS server.
2. **Ensure Static IP:**
    - Follow instructions to assign your Raspberry Pi a static IP to avoid connectivity issues.

Verify Pi-hole Functionality

1. **Check Ad-blocking:**
    - Visit Adblock Tester. Your score should improve significantly with Pi-hole.
2. **Access Pi-hole Admin Console:**
    - Go to http://pi.hole/admin/login.php.
3. **Inspect Sites:**
    - Visit usual ad-heavy sites to confirm ads are blocked.
4. **Check Queries Blocked:**
    - Look at the "queries blocked" statistic on the Pi-hole dashboard.
5. **Check Device DNS:**
    - Verify that other devices use Pi-hole's IP in their DNS settings.

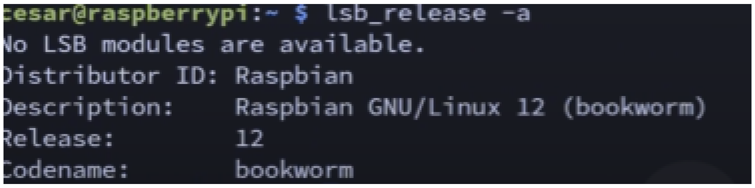
Troubleshooting: If issues persist, restart your router to renew DHCP leases and apply new settings.

Congratulations! Your network is now protected from ads. Enjoy your enhanced privacy and security.

OPTIONAL SETUP FOR SPLUNK

My Environment

- **Hardware:** Raspberry Pi 5
- **Operating System:**



```
cesar@raspberrypi:~ $ lsb_release -a
No LSB modules are available.
Distributor ID: Raspbian
Description:    Raspbian GNU/Linux 12 (bookworm)
Release:        12
Codename:       bookworm
```

- **Software:**
  - Pi-hole v5.2.4
  - Web Interface v5.4
  - FTL v5.7
- **Additional Setup:** Splunk running in a Docker container on a Synology NAS or in this case we have a server hosting splunk through our website.

Preparation

You will need:

- A free Splunk account to get the latest version of the Splunk Universal Forwarder.
- Pi-hole up and running.
- Root access to your Pi-hole server.

Installation Steps

1. **Sign Up for Splunk:**
   - Create an account at Splunk.com.
2. **Download Splunk Universal Forwarder:**

- Navigate to the download page and get the `wget` command line for the ARM package.
- Note: The latest ARM package available was 8.2.1, which didn't work for 32-bit Raspbian. Retrieve the 8.2.1 ARM package from the previous package repository.
- Verify if your Splunk UF is 32-bit or 64-bit:

  ```
  shCopy code
  file /opt/bin/splunkforwarder/splunk
  ```

3. **Extract and Prepare:**
   - Extract the package to `/opt`.
   - Create a new user `splunk` to run Splunk UF.
   - Change ownership and set environment variable:

     ```
     shCopy code
     export SPLUNK_HOME=/opt/splunk
     chown -RP splunk:splunk /opt/splunkforwarder
     ```

4. **Configure Inputs:**
   - Create a new file `/opt/splunkforwarder/etc/system/local/inputs.conf`:

     ```
     plaintextCopy code
     # inputs.conf
     [monitor:///var/log/pihole.log]
     disabled = 0
     sourcetype = pihole
     index = pihole

     [monitor:///var/log/pihole-FTL.log]
     disabled = 0
     ```

```
sourcetype = pihole:ftl
index = pihole
```

5. **Configure Outputs:**
   - Create a new file `/opt/splunkforwarder/etc/system/local/outputs.conf`:

     ```
     plaintextCopy code
     [tcpout]
     defaultGroup=indexer

     [tcpout:indexer]
     server=192.168.0.55:9997
     ```

   - Note: Replace `192.168.0.55` with your Splunk indexer or Splunk Heavy Forwarder IP address.
6. **Enable Boot Start:**
   - Set Splunk Universal Forwarder to start at boot time:

     ```
     shCopy code
     [sudo] $SPLUNK_HOME/bin/splunkforwarder enable boot-start
     ```

7. **Reboot:**
   - Reboot the Splunk Universal Forwarder or the entire Raspberry Pi.
8. **Start Splunk:**
   - Switch to the `splunk` user and start Splunk:

     ```
     shCopy code
     /opt/splunk/bin/splunkforwarder start --accept-license
     ```

```
-user splunk
```

Pi-hole Configuration

1. **Log Queries:**
    - Create a new file `/etc/dnsmasq.d/02-pihole-splunk.conf` with:

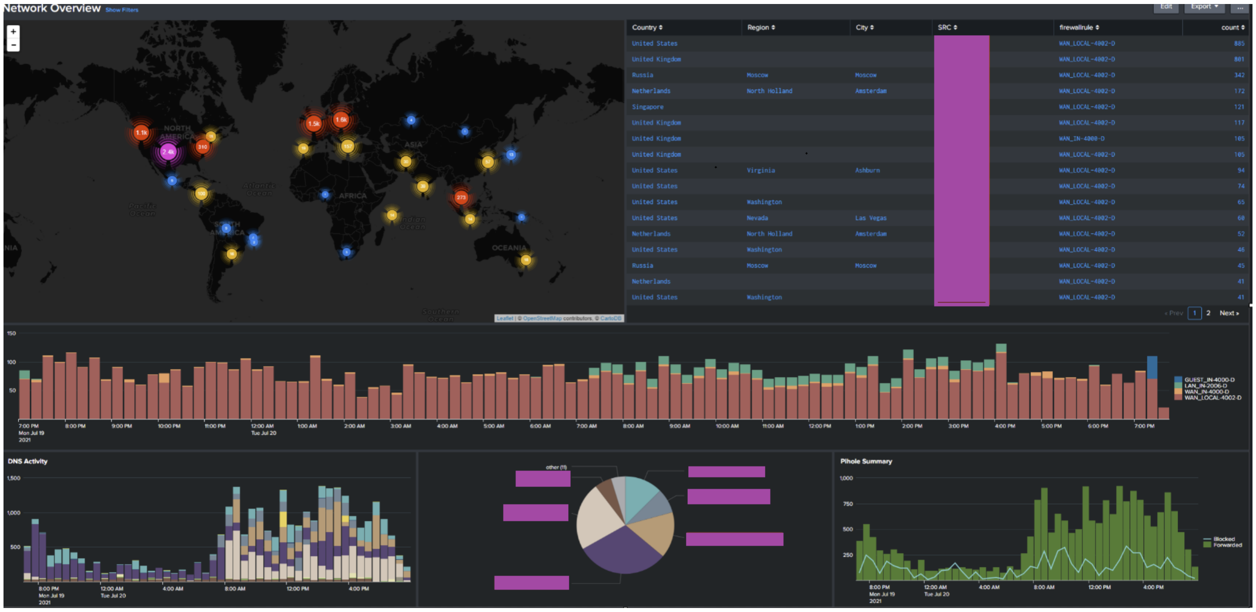
    ```
    plaintextCopy code
    log-queries=extra
    ```

2. **Restart:**
    - Restart Pi-hole or reboot the Raspberry Pi.

Splunk Configuration

1. **Download Pi-hole TA:**
    - Install the Pi-hole TA for field extraction of Pi-hole logs through this link https://splunkbase.splunk.com/app/4121/
2. The other option is to pull the source through Splunk and setup the Index and customize your own dashboard. This step we will not be providing.

End result:



Country ⇕	Region ⇕	City ⇕	SRC ⇕	firewallrule ⇕	count ⇕
United States			[redacted]	WAN_LOCAL-4002-D	885
United Kingdom			[redacted]	WAN_LOCAL-4002-D	801
Russia	Moscow	Moscow	[redacted]	WAN_LOCAL-4002-D	342
Netherlands	North Holland	Amsterdam	[redacted]	WAN_LOCAL-4002-D	172
Singapore			[redacted]	WAN_LOCAL-4002-D	121
United Kingdom			[redacted]	WAN_LOCAL-4002-D	117
United Kingdom			[redacted]	WAN_IN-4000-D	105
United Kingdom			[redacted]	WAN_LOCAL-4002-D	105
United States	Virginia	Ashburn	[redacted]	WAN_LOCAL-4002-D	94
United States			[redacted]	WAN_LOCAL-4002-D	74
United States	Washington		[redacted]	WAN_LOCAL-4002-D	65
United States	Nevada	Las Vegas	[redacted]	WAN_LOCAL-4002-D	60
Netherlands	North Holland	Amsterdam	[redacted]	WAN_LOCAL-4002-D	52
United States	Washington		[redacted]	WAN_LOCAL-4002-D	46
Russia	Moscow	Moscow	[redacted]	WAN_LOCAL-4002-D	45
Netherlands			[redacted]	WAN_LOCAL-4002-D	41
United States	Washington		[redacted]	WAN_LOCAL-4002-D	41

« Prev 1 2 Next »