G06-DST-UG25T2 – Training Material

<Bridging The Digital Divide>

Unit: COIT13236

Student 1: Shahriar Haque - S0259035

Student 3: Nicholas Hirning - 12158631

Project Mentor: Ahmad Saeed (a.saeed@cqu.edu.au)

Date: 04-08-2025

CQUniversity Australia

**GitHub Repository:**

<https://github.com/cquict2025/nis-y25t2-project-g06-dst-ug25t2/tree/main>

# Executive Summary

The implementation of the wireless mesh network is achieved through the installation of OpenWrt on a Raspberry Pi 5. Using antennas to extend the network range and a renewable solar energy source in conjunction with a battery allows for the physical, rural deployment of the network. With sufficient geographical deployment, it can provide the mesh network with greater, unobstructed signal coverage and the longevity of the hardware can be maintained by protecting it from weather conditions. Maintaining the solar panel’s exposure to sunlight to provide power is key for its continued operation, however the network can be maintained for an extended period without sunlight due to the implementation of a battery to store excess generated power to be used during nights and poor weather conditions.

The network configuration consists of increased security meshes including WPA-2-PSK for the requirement of password authentication and encryption. Firewalls have been configured to restrict the access of the SSH port and further unknown traffic to reduce risks. SSH connections have additionally been restricted using digital certificates to authenticate users providing a higher level of security over password authentication.

# Terms & Definitions

**ESSID:** Extended Service Set Identifier is used for managing and distributing traffic of a wireless network with multiple access points not visible to the user (This vs. That n.d.).

**LAN:** Local Area Network defines a limited area of interconnected devices such as in a singular house.

**LuCI:** Lua Control Interface, “a web interface used to administer OpenWrt” (OpenWrt 2023).

**MicroSD:** An SD (Secure Digital) is a memory card used for file storage, in this case used by the Raspberry Pi 5 to store, read, and write its OS and relevant files.

**OpenWrt:** A free open-source Linux-based operating system. OpenWrt provides firmware for home routers.

**Operating System (OS):** “A System software that manages all the resources of the computing device” (GeeksforGeeks 2025).

**Raspberry Pi 5:** A high-performance compact computer with both wired and wireless networking capabilities through its Gigabit Ethernet and its dual-band 802.11ac Wi-Fi, providing short range wireless communications (Raspberry Pi n.d.).

**Root:** Root in Linux terminology refers to the user with the highest level of permission. A root user has access to all commands and files.

**SSH:** Secure Shell (SSH) allows for the secure transmission of commands across an unsecure network. This allows for the remote connection and execution of the command-line interface (CLI) (Cloudflare n.d.).

**SSID:** Service Set Identifier is used to identify and select a specific wireless local area network (WLAN) for users to connect (This vs. That n.d.).

**WAN:** Wide Area Network defines an extended range of communication through relays connecting various LAN networks together.

**Wireless Access Point:** Refers to the SSID of a network for users to identify and connect a device to the specified network.

**WPA-2-PSK:** “Wi-Fi Protected Access 2—Pre-Shared-Key or WPA2-Personal, which is used to protect network access and data transmission by using an AES (Advanced Encryption Standard) or TKIP (Temporal Key Integrity Protocol) encryption method” (Beal, V 2022).

Table of Contents

[Executive Summary 0](#_Toc209452554)

[Terms & Definitions 0](#_Toc209452555)

[1 Introduction 1](#_Toc209452556)

[1.1 What is Covered 1](#_Toc209452557)

[1.2 What is Not Covered 1](#_Toc209452558)

[1.3 Assumptions & Expectations 1](#_Toc209452559)

[2 Setup & Installation 2](#_Toc209452560)

[2.1 Requirements 2](#_Toc209452561)

[2.2 Hardware Setup 2](#_Toc209452562)

[2.3 OpenWrt Setup 3](#_Toc209452563)

[2.4 Network Setup 4](#_Toc209452564)

[2.5 Network Configuration 5](#_Toc209452565)

[2.6 BATMAN Setup 6](#_Toc209452566)

[2.7 BATMAN Configuration - Server 6](#_Toc209452567)

[2.8 Wireless Configuration - Server 7](#_Toc209452568)

[2.9 Client Configuration 7](#_Toc209452569)

[3 Maintenance & Health 7](#_Toc209452570)

[3.1 Updating 8](#_Toc209452571)

[4 Troubleshooting 9](#_Toc209452572)

[4.1 Potential Issues & Solutions 9](#_Toc209452573)

[5 Appendix 10](#_Toc209452574)

[6 References 18](#_Toc209452575)

# Introduction

This training material aims to provide a knowledge base for deploying, operating and maintaining the network mesh and its counterparts. This is done by providing a step-by-step tutorial on the installation process of OpenWrt, an OS (Operating System) onto a Raspberry Pi 5. Utilising antennas and a power source, the Raspberry Pi 5 has the necessary hardware components to provide a portable and low-cost solution for a rural mesh network deployment.

## What is Covered

* Installation and configuration of the OpenWrt operating system.
* Maintaining and updating OpenWrt.
* Best security practices.
* Alternative security options and useful configurations necessary for security.
* Troubleshooting. Common issues and how to diagnose problems and find solutions.

## What is Not Covered

* Configuration options outside of the project requirements such as alternative DNS solutions and VPN setup and configuration. Steps on how to implement VPN and DNS are outlined in the [Technical Mesh Network Setup & Installation document](https://github.com/cquict2025/nis-y25t2-project-g06-dst-ug25t2/blob/main/documents/network/12161580%20-%20Technical%20Mesh%20Network%20Setup%20%26%20Installation.docx).

## Assumptions & Expectations

It is assumed the user has little to no experience with:

* Linux-based operating systems.
* Router configurations and settings.
* Network security and protocols.
* Best security practices such as password policies.

# Setup & Installation

This section relates to the setup and installation of OpenWrt onto a Raspberry Pi 5.

## Requirements

* MicroSD or SD reader – To install or re-install the OpenWrt OS.
* Rufus - A tool to convert OS images into bootable media.
* Raspberry Pi 5 – The hardware used to provide networking features and host the OpenWRT OS.
* Antenna – Extends the range capabilities of the Raspberry Pi 5.
* Computer – An Ethernet cable connected from the computer to the Raspberry Pi 5 to access the initial web interface configuration (LuCI) of OpenWrt. Or a screen / monitor connected with a micro-HDMI to the Raspberry Pi 5 to access its command-line interface.
* Solar panel – Used to generate the power to run the Raspberry Pi 5.
* Battery – Stores the generated power from the solar panel to be used by the Raspberry Pi 5.

## Hardware Setup

This section encompasses instructions and optimal setup of the mesh network hardware including its antenna, solar panel, battery, and the Raspberry Pi 5.

Optimal location setups should adhere to these key points:

* Solar panel must be exposed to sunlight to generate power. (It will require at least 5-6 hours of direct sunlight a day unless the power system is upgraded).
* The Raspberry Pi 5 itself is not to be exposed to the weather including direct sunlight, water or dust as there is a risk of damaging or shortening the device’s lifespan. (This can be alleviated by creating a waterproof housing).
* Use elevated terrain to avoid flood water and provide greater signal coverage.

Insert a suitable USB-C power supply to the Pi (Minimum required is 5W at 5A, more power can be used as desired if running Pi with more demanding hardware, with the maximum power draw the Pi5 can support being 27w at 5A). The Pi should automatically power up assuming a supported power supply was connected. Power button should indicate a solid green if it is ready and booted.

Power delivery system will require a minimum:

* 5w power output
* 20w of Solar Pannels
* 20000mah Battery pack.

This will be enough to run the Pi5 and charge the battery pack for 24h usage within the sunlight period. This is under the assumption that no external accessories are run from the Pi5, if more are added then the power delivery system will need to be upgraded appropriately as required, e.g. large antennae.

To reduce the ongoing power consumption the Pi5 will be setup using SSH and when maintained in person by a technician, they are required to bring their own battery powered mobile display to plug in so there is no display required on site.

Instructions:

* Pi5 should be securely mounted away from the elements
* Solar panels need to be positioned north facing in direction of sunlight
* Batter pack should be positioned out of direct elements to prolong lifespan
* Solar panel must be connected to the UPS system and UPS system to battery pack
* Pi5 will be connected to the battery pack via a type C to USB power connector
* Antennae will be plugged into the USB port on the Pi5 (If the Pi5 has a top hat there are PCIE variants of antennae and alternatively a wireless access point can be connected to the WAN port or an ethernet to USB adapter to extend the range instead)

Refer to the following figures for annotations outlining the setup of cables between devices:

***Figure 1: Shahriar Haque 2025, Raspberry Pi 5 Annotated 1***

***Figure 2: Shahriar Haque 2025, Raspberry Pi 5 Annotated 2***

***Figure 3: Shahriar Haque 2025, Raspberry Pi 5 Annotated 3***

***Figure 4: Shahriar Haque 2025, Raspberry Pi 5 Annotated 4***

***Figure 5: Shahriar Haque 2025, Raspberry Pi 5 Annotated 5***

## OpenWrt Setup

* Downloading OpenWrt image from sight and an imaging tool such as Rufus.
* Get correct up-to-date version of OpenWrt from website.
* Select specific type with web interface package.
* How to load image as a bootable media.

Refer to ***Figure 6: James Ira Arellano 2025, OpenWrt Firmware Selector***outlining the packages to be installed with OpenWrt sourced from <https://firmware-selector.openwrt.org/>.

Before downloading the factory (SQUASHFS) version of OpenWrt we require two packages containing the drivers to make the Comsol CMGB25 2.5G Ethernet Adapter recognisable to our Raspberry Pi 5. SQUASHFS is used over EXT4 is that it is read-only which allows it to be easily restored to its factory state. These two packages are as follows and are to be added to the ‘customise installed packages’ list alongside luci-ssl to support HTTPS connections of HTTP webpages:

* kmod-usb-net-asix-ax88179
* kmod-usb-net-rtl8152
* luci-ssl

The image file is not usable as-is for installing OpenWrt onto the Raspberry Pi 5’s MicroSD. To do so, we require an additional tool to create a bootable drive. One example of such a tool is Rufus. Alternatives for Linux devices include Balena Etcher or DD via CLI. This will be used to convert our OpenWRT image file into a bootable drive. After both files have been downloaded. Launch Rufus and select the MicroSD in the device dropdown panel. Secondly, clicking select will open the file explorer, navigate to where you downloaded the OpenWrt image and select it. Before proceeding, ensure any relevant files are saved from this device as all existing data will be erased. Once these two selections have been made and the device is ready to be erased, you may click start to proceed formatting the device into a bootable OpenWrt drive.

Refer to ***Figure 7: Nicholas Hirning 2025, Rufus Homepage***, the webpage to download the Rufus tool from, available at: <https://rufus.ie/en/?pubDate=20250731>

Refer to ***Figure 8: Nicholas Hirning 2025, Rufus 4.9.2256*** for configurations used.

Refer to ***Figure 9: Nicholas Hirning 2025, USB MicroSD reader*** for an example of the MicroSD reader used.

## Network Setup

The easiest way to first gain access to the LuCI web interface is by a LAN connection which is established by connecting a computer via an Ethernet cable to the Raspberry Pi 5. You should then be able to access this interface explained under the network configuration section (Personally I required to restart my pc for the LAN connection to be established). Otherwise, there is the upstream ethernet (WAN) method if you connect the USB to an ethernet adapter to a USB 3.0 port on the Raspberry Pi 5. Proceed to connect this to a LAN port on your home router, or any switch that is uplinked to the main router.

If a LAN connection is not possible, an alternative solution is to use a micro-HDMI cable connected to a powered display to access the OpenWrt command-line interface. From there, a wireless access point can be enabled. For the built-in Wi-Fi of the Raspberry Pi 5, these commands should be run with radio0, however other devices may require radio1 or otherwise. After these steps the OpenWrt wireless access point should be visible to other devices in range.

* uci show wireless
  + Displays the list of wireless hardware and their configuration. In this case we are enabling the Raspberry Pi 5’s built in dual-band 802.11ac Wi-Fi. We can configure the Comsol CMGB25 2.5G Ethernet Adapter from the web interface later.
* uci set wireless.radio0.country=’AU’
  + Sets the country code to Australia.
* uci set wireless.radio0.disabled=’0’
  + Enables the wireless access point upon commit.
* uci set wireless.default\_radio0.encryption=’psk2’
  + Sets the wireless access point encryption method.
* uci set wireless.default\_radio0.key=’yourpasswordhere’
  + Sets the password for the wireless access point.
* uci commit wireless
  + Commits and applies changes
* wifi reload
  + Reloads the Wi-Fi interfaces with the newly configured settings.

Additionally, vi /etc/config/wireless shows the config file for the wireless interface. Press Insert or I to type. Press escape and type :w to write to file. It is important to set the encryption and key before enabling the wireless access point as this would make the connection accessible to all putting the network at a significant security risk. Configuration of the network can be down within the command line such as adding firewall rules for SSH ports and more. You can access more examples in the OpenWrt uci documentation (OpenWrt 2024).

Once a secondary device has connected you may access the LuCI web interface by entering <http://192.168.1.1> into your search browser. You may need to remove s from https as the initial setup is not a secure one. The default username is root and there is no password. Leaving the password field empty, you can then login.

The first thing that should be done is adding a root password to the web interface. This will prevent the configuration of the web interface from being accessed by unknown users. There will be a large banner warning of there being no password. By selecting go to password configuration, you can create a new password. Ensure that this password is unique to administrative roles and adheres to the password policies and is not used as the password to connect to the wireless network.

Refer to ***Figure 10: Nicholas Hirning 2025, LuCI Router Password Page***.

## Network Configuration

* OpenWrt settings and configs for establishing a mesh network.
* Antenna configuration.
* Wireless Access Point Setup (Excluding configuration which is covered in Wireless Configuration).
* Setup SSH digital certificate.
* Firewall configuration.

Once a root password has been configured the next step

The next step is to create our WAN interface if it wasn’t automatically done so (it likely won’t be). At the top ribbon, hover on Network > Interfaces to edit our interfaces. Select “add new interface”, on the new menu, designate a name for this interface, WAN is what I used as it is our upstream internet link. Select a DHCP client for the protocol and the device should be “eth1”. As for the LAN connection, ensure that the device is using a static address of 192.168.1.1 as configured by first boot and flash, and that it is using eth0 as the onboard ethernet is crucial for LAN-side connection and management.

***Figure 11: James Ira Arellano 2025, OpenWrt LuCI New Interface***

Once this interface is added, ensure you click “save & apply” in the blue button below the interfaces to apply the changes. Patiently wait for the interface to pick up an IP address from your home network and when it does, it will allow us to gain internet access.

***Figure 12: James Ira Arellano 2025, OpenWrt LuCI Interfaces***

After setting up our WAN interface, we can then SSH to the Pi with our onboard LAN device.

Ensure that you are using SSH with the user “root”. Leaving out the user or entering the wrong user will result in SSH connection to not establish.

***Figure 13: James Ira Arellano 2025, Linux Konsole ssh connection***

Enter your root password if one was created. The terminal will then prompt you if you want to accept the SSH fingerprint. Accepting it will connect us to the Pi. Once connected, packages can be updated for security concerns by typing ‘opkg update’.

## BATMAN Setup

“B.A.T.M.A.N. is derived from “Better Approach To Mobile Adhoc Networking” (OpenWrt 2024).

For the implementation of BATMAN four packages are needed:

* kmod-mac80211 – A driver for the Wi-Fi 802.11 mesh.
* iw – Advanced wireless management (CLI).
* wireless-tools – Optional, but Helpful Wi-Fi tools.
* kmod-batman-adv – A mesh protocol for layer 2 networking.

These drivers can be installed by running ‘opkg update’ followed by ‘opkg install kmod-mac80211 iw wireless-tools’. We will then need the kernel module for BATMAN to allow the function of the layer 2 mesh network. This module is installed by executing ‘opkg install kmod-batman-adv’.

Further packages recommended by Arellano, J (2025) for the network configuration include:

* luci-proto-batman-adv - Allows for LuCI configuration of batman-adv
* luci-app-opkg - To install software from web interface if chosen
* tcpdump *or* iftop - For debugging and monitoring of traffic
* nano – A text editor from an SSH terminal

## BATMAN Configuration - Server

The first step to configuring BATMAN is through creating a new interface in the interfaces section of the web interface. This new interface is to use the Batman Device protocol and can be named bat0.

***Figure 14: James Ira Arellano 2025, bat0 interface (LuCI)***

The bat0 interface is to be further configured within the mesh routing section. Changes to be made include:

Routing Algorithm - BATMAN\_V:

* Providing more optimised routing.
* Uses layer 3 routing to improve reachability.
* Suitable for larger networks.

Avoid Bridge Looks – Enabled:

* Ensures bridged LAN connections against broadcast loops which will bring the LAN connection down.

Gateway Mode – Server:

* Identify the primary Raspberry Pi 5 that acts as the gateway to the internet as the server.
* Additional devices connected to the mesh network are to be labelled as ‘Client’.

The next step is to create an alias interface to associate a radio with this new BATMAN mesh. Create a new interface whilst leaving the device as unspecified and setting the protocol as “Batman Interface”. A new menu will appear where we will specify the ‘Batman Device’ as bat0. Leave all other options as default.

***Figure 15: James Ira Arellano 2025, batmesh interface (LuCI)***

Next is the configuration of the LAN bridge. Implementing on both the server and client will allow them to communicate with other nodes and allow client nodes to access the internet. In the Interfaces section of the web interface (LuCI) select the devices tab and select bat0 and the ethernet adapter eth0 (eth0 being our LAN).

***Figure 16: James Ira Arellano 2025, LAN Bridging (LuCI)***

## Wireless Configuration - Server

Once the BATMAN mesh configuration is complete, we will then need to configure and enable the Wireless Access Point (WAP). Create or configure the WAP, ensure the operating frequency, channel, and width in the device configuration section are consistent between nodes and apply the following settings in the interface configuration section:

* Mode: 802.11s.
* Mesh ID: A unique name separate to the SSID. (OpenWRT-BATMAN-Mesh in this example).
* Network: batmesh.

In the wireless security section of the WAP, select WPA3-SAE and set a password that meets the password policy guidelines. Some older hardware may not support newer security and encryption methods. If any issues occur select WPA2 then try WPA. Once done, in the advanced settings tab, disable forward mesh peer traffic. This option will change our routing protocol from OLSR by default to our BATMAN routing protocol.

## Client Configuration

Section to be completed.

# Maintenance & Health

This section relates to update procedures outlined by the OpenWrt (2025) documentation and standard practices for maintaining security and stability of the mesh network devices.

Physical cleaning of dust and debris from the hardware such as fans and reapplying thermal paste if used by the hardware can help reduce temperatures for greater longevity.

Sometimes updating an OS can be unsuccessful which can be caused by selecting the incorrect firmware from the OpenWrt firmware selector website or other reasons such as lack of backwards compatibility. As such it is important to backup settings and maintain a copy of the previous OS. Furthermore, it may be required to install from Factory (SQUASHFS) image rather than sysupgrade as hardware may fail to boot due to corrupt partitions or files.

## Updating

* To update packages run ‘opkg update’.
* Update the OpenWrt Operating System for vulnerability patches, performance and stability fixes
* Update OpenWrt configurations when amendments are made for updated and improved security settings

“Sysupgrade via LuCI or CLI works by optionally saving specified configuration files, wiping the entire file system, installing the new version of OpenWrt and then restoring back the saved configuration files. This means that any parts of the file system that are not specifically saved will be lost” (OpenWrt 2025). Configuration files such as those in /etc/config will be restored which encompasses network settings, Wi-Fi settings, and more.

Three steps outlined by the OpenWrt (2025) sysupgrade documentation involve:

1. Setup for data migration by keeping settings and additional sysupgrade.conf entries.
2. Export / save installed package list / manifest.
3. Obtain / verify new installation sysupgrade image in addition to a known stable fallback in-case of an upgrade failure.

A customised backup can be made through LuCI under System, Backup / Flash Firmware / Configuration. Once complete a backup can be created by selecting backup. Restoring backups is done by selecting restore and loading the backup file.

When upgrading OpenWrt it is important to use the sysupgrade image (SQUASHFS) rather than the factory image from the firmware selector website. You can customise the installed packages with any additions made by the user.

An upgrade can be done through the LuCI web interface under System, backup / flash firmware by selecting the Flash new firmware image option. Otherwise, it can be similarly achieved through the command-line with the sysupgrade command. Once an upgrade has been completed, additional packages installed by the user will need to be reinstalled and further configuration may be required for new or reverted settings.

# Troubleshooting

## Potential Issues & Solutions

* Lists potential issues and solutions the device in terms of hardware and software, both in network configuration and more.
* Updating OpenWrt whilst maintaining security settings can lead to conflicts and errors in configuration files after significant changes have been made to the OS. To solve this, it is recommended by OpenWrt (2025) to not keep settings once every 12-16 months and reconfigure manually.
* Older hardware may not be compatible with certain configuration options such as WPA3 and as such must rely on widely supported alternatives such as WPA2 or WPA.

***Figure 17: Nicholas Hirning 2025, SSH Key Verification Warning***

When reinstalling OpenWrt from scratch connecting through ssh may not work. This is because a new key fingerprint is generated which does not match the key which is saved by the client device. To fix this we can remove the key from the list of known hosts file found in:

* C:\Users\USERNAME HERE\.ssh

Opening this file using any text editor, simply remove the entry relating to 192.168.1.1.

# Appendix

A close up of a device

AI-generated content may be incorrect.

Figure : Shahriar Haque 2025, Raspberry Pi 5 Annotated 1

A close up of a device

AI-generated content may be incorrect.

Figure : Shahriar Haque 2025, Raspberry Pi 5 Annotated 2

A close up of a device

AI-generated content may be incorrect.

Figure : Shahriar Haque 2025, Raspberry Pi 5 Annotated 3

A solar panel with blue lights

AI-generated content may be incorrect.

Figure : Shahriar Haque 2025, Raspberry Pi 5 Annotated 4

A computer on a table

AI-generated content may be incorrect.

Figure : Shahriar Haque 2025, Raspberry Pi 5 Annotated 5

A screenshot of a computer

AI-generated content may be incorrect.

Figure : James Ira Arellano 2025, OpenWrt Firmware Selector

A screenshot of a computer

AI-generated content may be incorrect.

Figure : Nicholas Hirning 2025, Rufus Homepage

A screenshot of a computer

AI-generated content may be incorrect.

Figure : Nicholas Hirning 2025, Rufus 4.9.2256

A close up of a usb flash drive

AI-generated content may be incorrect.

Figure : Nicholas Hirning 2025, USB MicroSD reader

A screenshot of a computer

AI-generated content may be incorrect.

Figure : Nicholas Hirning 2025, LuCI Router Password Page

A screenshot of a computer

AI-generated content may be incorrect.

Figure : James Ira Arellano 2025, OpenWrt LuCI New Interface

A screenshot of a computer program

AI-generated content may be incorrect.

Figure : James Ira Arellano 2025, OpenWrt LuCI Interfaces

A screenshot of a computer

AI-generated content may be incorrect.

Figure : James Ira Arellano 2025, Linux Konsole ssh connection

A screenshot of a computer

AI-generated content may be incorrect.

Figure : James Ira Arellano 2025, bat0 interface (LuCI)

A screenshot of a computer

AI-generated content may be incorrect.

Figure : James Ira Arellano 2025, batmesh interface (LuCI)

A screenshot of a computer

AI-generated content may be incorrect.

Figure : James Ira Arellano 2025, LAN Bridging (LuCI)

A screenshot of a computer

AI-generated content may be incorrect.

Figure : Nicholas Hirning 2025, SSH Key Verification Warning

# References

Beal, V 2022, *WPA2-PSK*, viewed 31 August 2025, <https://www.webopedia.com/definitions/wpa2-psk/>

Cloudflare n.d., *What is SSH? | Secure Shell (SSH) protocol*, viewed 31 August 2025, <https://www.cloudflare.com/learning/access-management/what-is-ssh/>

GeeksforGeeks 2025, *What is an Operating System?*, viewed 31 August 2025, <https://www.geeksforgeeks.org/operating-systems/what-is-an-operating-system/>

OpenWrt 2023, *Web interface overview*, viewed 31 August 2025, <https://openwrt.org/docs/guide-user/luci/webinterface.overview>

OpenWrt 2024, *B.A.T.M.A.N. / batman-adv*, viewed 10 September 2025[https://openwrt.org/docs/guide-user/network/wifi/mesh/batman?s[]=open&s[]=one](https://openwrt.org/docs/guide-user/network/wifi/mesh/batman?s%5b%5d=open&s%5b%5d=one)

OpenWrt 2024, *The UCI system*, viewed 31 August 2025, <https://openwrt.org/docs/guide-user/base-system/uci>

OpenWrt 2025, *Upgrading OpenWrt firmware using LuCI and CLI*, viewed 31 August 2025, <https://openwrt.org/docs/guide-user/installation/generic.sysupgrade>

Raspberry Pi n.d., *Raspberry Pi 5*, viewed 30 August 2025, <https://www.raspberrypi.com/products/raspberry-pi-5/>

This vs. That n.d., *ESSID vs. SSID*, viewed 31 August 2025, <https://thisvsthat.io/essid-vs-ssid>