

Homelabbing

As part of commitment to my development of IT skills this personal homelab built simulates a small enterprise environment. This project as my first own homelab gave me valuable insight of network configuration, Docker containers, server management and self-hosted applications. This project ultimately strengthened my practical and theoretical troubleshooting skills, networking and problem solving skills.

I built my own personal homelab using a refurbished office computer NUC7i3DNH. By utilizing Ubuntu LTS will actively allow me to gain practical experience and improve my quality basic tasks with my own self hosted infrastructure not reliant on 3rd party software. The server is configured with Docker to host multiple applications such as Jellyfin for media access, Immich for cloud photo management and home network services.

HARDWARE COMPONENTS NUC7i3DNH

RAM - DDR4 8GB

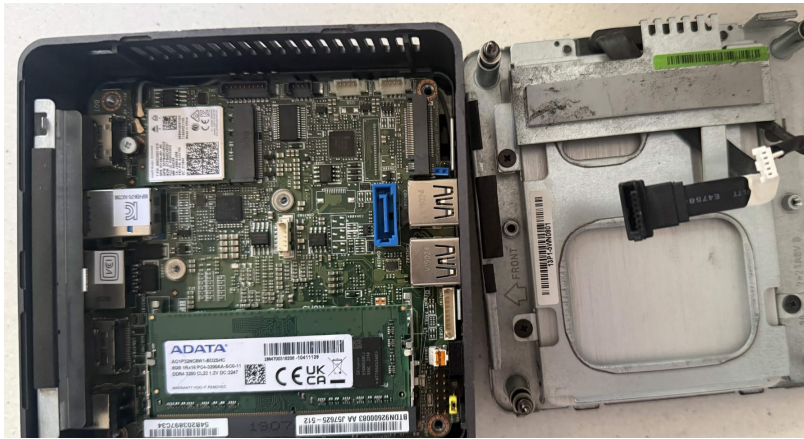
CPU - Intel Core i3-7100U dual-core

HDD - Seagate 4TB

M.2 SSD - Western Digital 512 GB

Upgrading NUC7i3DNH

The standard version of the NUC7i3DNH only comes with a 128gb SSD, this is a significant issue as I plan to store large amounts of media and files whilst configuring my network. I have upgraded the storage to be in total 4.5 TB in total



Removing plate to access HDD bracket



Installing m.2 512GB SSD



Once put back together it is fully assembled to be configured and flashed with Ubuntu Server LTS

Network Layout

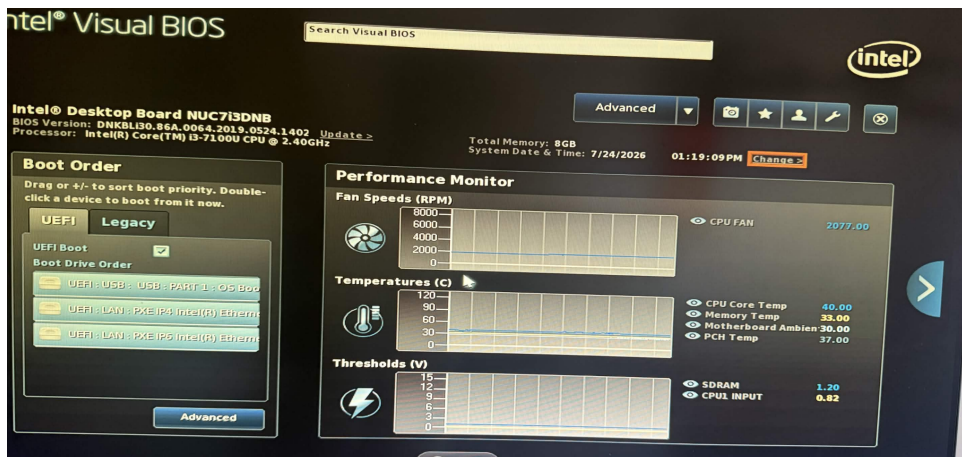
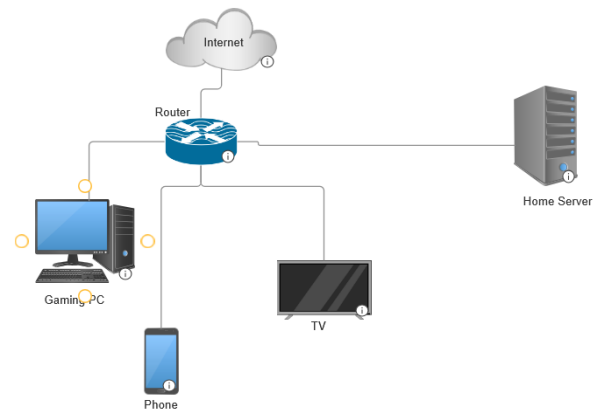
For the server to work it will need to be configured with a static IP so all devices can easily connect to the home server to allow access to software.

Setting up Ubuntu on home server

First i downloaded the software from Ubunutus official site, then flashed it on a USB drive using balenaEtcher

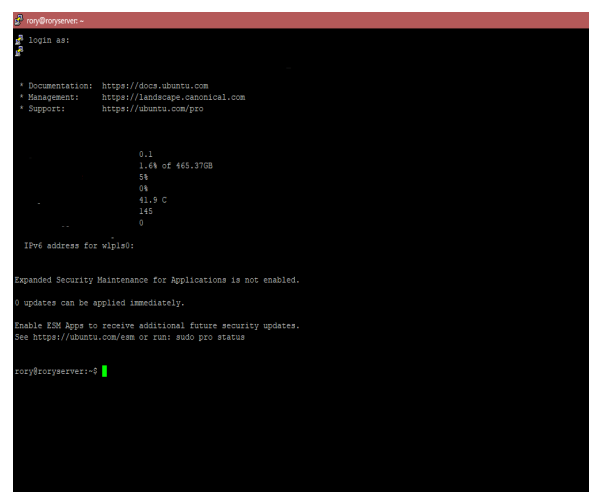
Then entering BIOS i selected the boot

drive



After I set up a SSH client connection using puTTY which will allow me to remotely access the server without having to physically connect to the server using a monitor.

After that I configured a static IP by disabling DHCP using nano text editor inside the server via remote SSH connection through puTTY inside Linux's netmap configuration file.



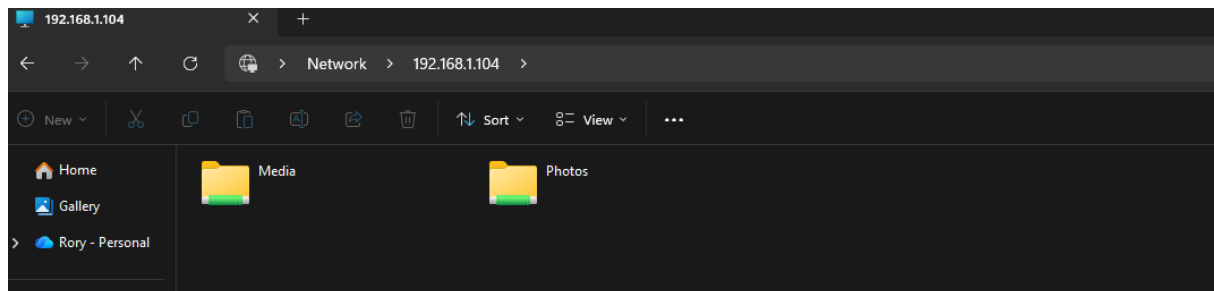
Docker & Portainer

Docker is a super useful tool that will allow me to run multiple applications such as jellyfin, immich and other future tools that I can use in their own isolated containers.

This will significantly improve efficiency and ease of access to further updating or changing tools used as I continue to maintain the home server.

Samba and File Sharing

After basic configuration of directories and installing Samba onto the server it will allow me to drag and drop photos and media that I own into my server remotely from my main PC.



Connecting it all together

By simply installing Jellyfin and Immich onto Portainer using their repositories, easily inputting the required yaml file code and editing the volumes section to include my TCP ports required and specific directory allowed me to successfully remotely connect to the applications from multiple devices.

Cybersecurity Defences

Utilizing the uncomplicated firewall(ufw) tool will block all ports that my ubuntu server is not using. Threat actors using active reconnaissance tools will block an unapproved port limiting my attack surface.

Including ufw tools i will also include fail2ban which will ban any outside IP address after failing authentication or suspicious behaviours.

Future improvements and Maintenance

My main goals for future improvements is installing a VPN tunnel / mesh to allow me to remotely connect to the linux server without risking public exposure through port forwarding. I would also include a monitoring app to provide conditions of components and containers to ensure that failures in the server are immediately notified and fixed.

Maintaining my server is extremely important as it is how I will rely on accessing all my media. For this case setting up constant updates for applications and containers is critical to ensuring my security. Furthermore my next project will be a NAS raspberry pi to hold large amounts of off site data just in case of failures in the server.