Overview

Quick Summary about my Homelab and projects Im working on

- My Homelab Setup
- Homelab Projects

My Homelab Setup

Network Setup

The network infrastructure is built using Ubiquiti equipment, leveraging its firewall features to ensure high-speed internet access and robust security. Security configurations, including firewall settings, are managed through Ubiquiti's system and router capabilities.

Devices are organized into designated VLANs—such as trusted wired devices, trusted wireless devices, IoT devices, and the homelab—each governed by strict firewall rules for specific communication protocols. For instance, while trusted devices are allowed to communicate with IoT devices, the IoT devices are restricted from initiating communication with trusted devices. The IoT network is fully isolated, preventing it from accessing any other devices in the house.

Additionally, I utilize Ubiquiti's IDS (Intrusion Detection System) and IPS (Intrusion Prevention System) at the highest level, along with an OSI Layer 7 Next-Generation Firewall. Firewall rules are implemented using Zone-Based Firewall features to enhance security and manage network traffic effectively. Homelab servers are also isolated from the rest of the network, with one wired Linux PC granted full access and another wired PC limited to proxy-only access.

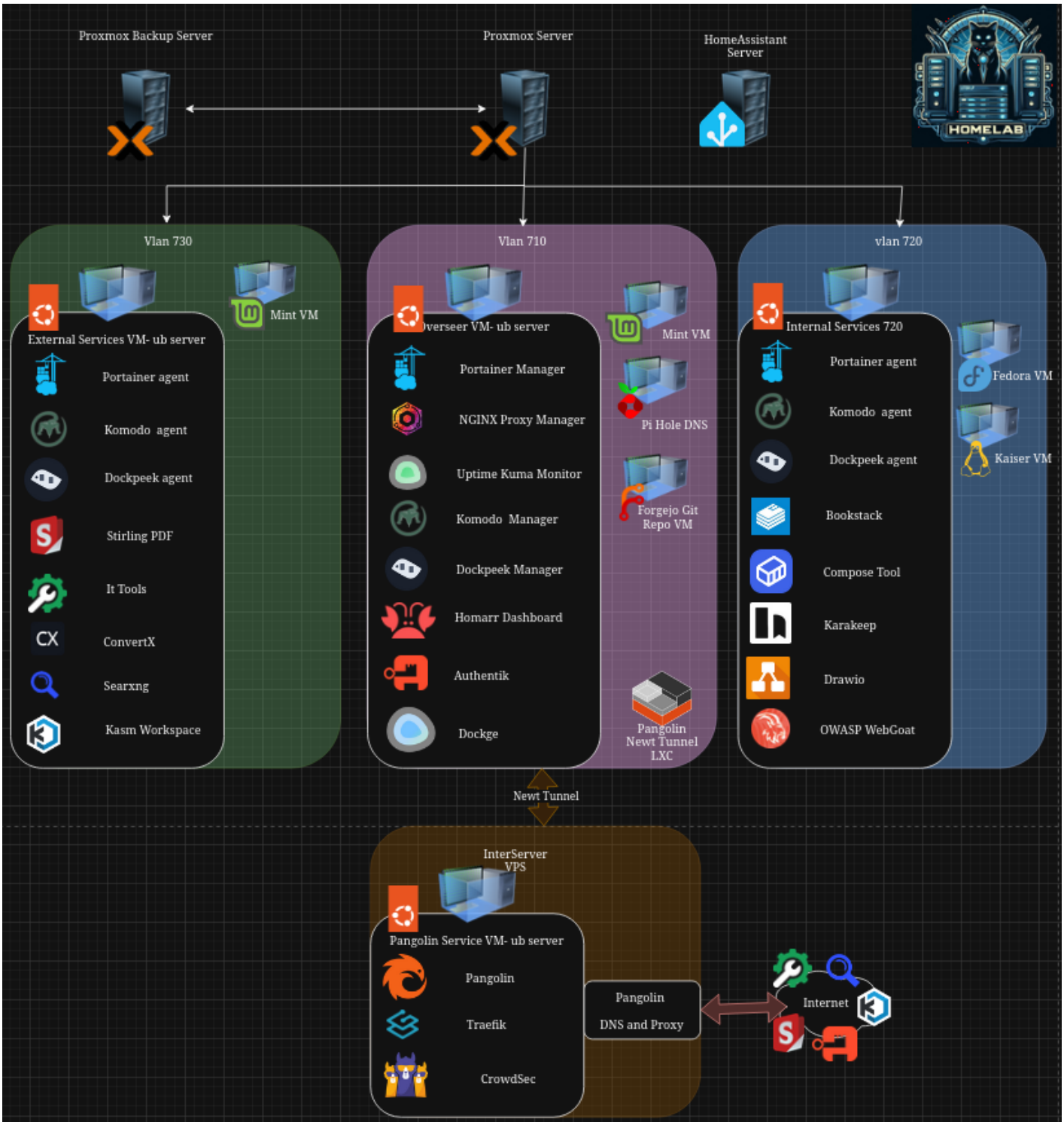
Lastly, no VLANs are configured with port forwarding. Instead, the homelab and Synology VLANs use Cloudflare Tunnel for traffic tunneling as needed. For internal homelab and Synology management, I rely on NGINX Proxy Manager with Let's Encrypt and Cloudflare DNS Challenge, ensuring all traffic is securely encrypted with TLS.

network

Server Setup

My homelab is set on three mini PCs: one running Home Assistant for homelab and smart device monitoring, one running a Proxmox virtualization environment and the another dedicated to Proxmox Backup Server. Both systems have their management interfaces isolated within a dedicated VLAN, ensuring secure administrative access. The broader homelab is segmented into three functional VLANs—Overseer, External, and Internal—each hosting a mix of Ubuntu and Fedora Server VMs for core services, alongside matching desktop VMs for testing and experimentation. Security and access is achieved with zone-based firewall trough Unifi network.

While all local devices can communicate with homelab servers via https, only one jumpbox machine running Fedora image can access non proxied traffic. Also, any non-critical outgoing traffic is being blocked or redirected through https.



Overseer VLAN (Control & Monitoring)

Inspired by Fallout, the Overseer VLAN serves as the central command layer. It hosts key infrastructure services including:

- **Komodo** for Docker container orchestration

- **UpTime Kuma** for uptime and service health monitoring
- **Dockeep** for Docker monitoring
- **Authentik** as the identity provider with role-based access control
- **Pi-Hole** for DNS filtering
- **NGINX Proxy Manager** for reverse proxying
- **Forgejo** Git server for version control and infrastructure-as-code

These services have tightly controlled access to both External and Internal VLANs, with return traffic restricted to specific ports. Overseer also maintains ICMP-only access to the management VLAN, allowing Kuma to monitor Proxmox nodes without exposing sensitive interfaces. Komodo and Dockeep are connected to other VMs via dedicated agents.

External VLAN (Public-Facing & DMZ)

The External VLAN is designed for secure, isolated access to public-facing services. It includes:

- **SearxNG,** a privacy-focused meta search engine
- **Kasm Workspace,** providing ephemeral browser containers for secure link handling
- **IT-Tools,** a utility suite for diagnostics and encoding
- **Stirling PDF,** a utility suite for editing pdfs

Remote access is proxied trough VPS-connected Pangolin gateway using Traefik and CrowdSec for dynamic routing and behavioral firewalling for full self-hosted zero-trust perimeter

Kasm and SearxNG resides in a DMZ zone with no port forwarding, and all traffic to SearxNG and Kasm is routed through Pangolin, ensuring strict ingress control and enhanced privacy.

In this VLAN, there is a standalone VM running Rust Desk for remote support. In the future, with the addition of NetBird or Tailscale, this VM will replace one currently hosted on a Linode VPS.

Internal VLAN (Personal Services & Testing)

The Internal VLAN is tailored for personal device support and local services. It includes:

- **Bookstack** a documentation service
- **Karakeep, a lightweight bookmark and note taking app**
- Two desktop VMs—one Fedora, one Kaiser—for testing and sandboxing

This VLAN remains isolated from external exposure, with future plans to expand its role in home automation and private service delivery.

Security & Access Control

Security in the homelab is enforced through a layered approach combining VLAN isolation, strict firewall policies, and centralized identity management. Most services are protected behind Authentik, which acts as the identity provider (IDP) using OIDC (OpenID Connect) and Proxy authentication methods. This ensures consistent, role-based access across applications, with support for multi-factor authentication (MFA), conditional access rules, and audit logging. Whether accessing Forgejo, Komodo, Homarr, or internal utilities like BookStack and Karakeep, users are authenticated through Authentik, minimizing credential sprawl and enhancing traceability.

Network segmentation further reinforces security, with each VLAN operating under tightly scoped firewall rules. Overseer services can only communicate with External and Internal VLANs through explicitly defined ports, and return traffic is strictly regulated. The External VLAN’s DMZ zone, hosting Kasm and SearxNG, is hardened with no port forwarding and ingress routed through a VPS-based Pangolin gateway using Traefik and CrowdSec. This setup replaces third-party tunnels with a self-hosted zero-trust perimeter, enabling dynamic routing and real-time threat mitigation.

Homelab Projects

This page goes over few major applications running on my Homelab

RustDesk

RustDesk is an open-source remote desktop solution that I host locally on Akamai and Linode cloud platforms. This setup ensures secure, private, and encrypted communication channels, making it ideal for assisting family and friends with technical issues. By hosting RustDesk locally, I retain full control over the server, minimizing reliance on third-party services while offering a highly reliable and secure alternative to commercial remote desktop solutions.

Kasm Workspace

Kasm provides "Browser as a Service," allowing me to spin up isolated containers for securely running a browser of my choice. This is especially useful for safely opening suspicious or malicious links. Kasm functions as a hypervisor, capable of creating containers with preconfigured images for various use cases. These include browsers, lightweight Linux desktop environments, or specific standalone applications. It's perfect for one-time, quick sessions with software I don't need installed permanently on my PC. The isolation provided by Kasm ensures that any risky or experimental activity is contained, safeguarding the rest of my system.

Searxng

Searx is a self-hosted, open-source search engine designed for privacy and customizability. It runs on my server, giving me complete control over my search data and ensuring a safe browsing experience. Searx uses unique, non-identifiable profiles to query multiple search engines simultaneously, consolidating results into a single page. This setup provides diverse results while removing trackers and ensuring that all links returned are free of invasive tracking mechanisms. Searx empowers me with private and efficient searches without compromising on security or functionality.

Home Assistant

Home Assistant is my go-to open-source home automation platform that acts as the central hub for managing and monitoring all my smart devices within my homelab. It plays a vital role in my setup by integrating with various services and systems, providing me with real-time monitoring and alerts for my infrastructure. I have it collecting data from my Proxmox virtualization environment, network devices, and other core services, which keeps me informed about the status of my machines, including temperature monitoring and uptime notifications. As I look to expand my homelab, I plan to enhance Home Assistant to support IoT devices, like Zigbee-enabled smart lights and sensors. This will not only streamline the management of my homelab but also create a more interconnected and automated living space, making it easier for me to monitor and control my environment efficiently.

Future Ideas

Zero Trust Service- With Pangolin already providing tunneling service for my application, exposing them securely without port forwarding, I plan to replace the traditional VPN with a zero trust service like Netbird or Tailscale with Headscale is an exciting step towards enhancing your network security and flexibility. By implementing a zero trust architecture, you can ensure that every device and user is authenticated and authorized before accessing your resources, significantly reducing the risk of unauthorized access.

Expanding server pool- Potentially adding additional mini pc to my server pool as another Proxmox environment creating cluster for redundancy and high availability.