“Secure Home-Lab”

“Ensuring Privacy and Cybersecurity in Home-Lab Environments"

whitepaper

Author: Jayabalaji Sathiyamoorthi

System Software Architect, Intel Corporation

https://jb-balaji.github.io/

17 March 2025

Table of Contents

Table of Figures

White Paper: Homelab - Ensuring Privacy and Cybersecurity in Home-Lab Environments

Executive Summary

In the current digital environment, privacy and cybersecurity are critical considerations for both individuals and organizations. As technology advances, the demand for secure, controlled environments for learning, testing, and personal use has increased substantially. This white paper examines the concept of a Homelab, a robust solution that utilizes containerized [1] applications to improve privacy and cybersecurity while offering a versatile infrastructure for a range of use cases.

1. What is a Homelab?

A **Homelab** refers to a personal, self managed computing environment often small-scale, network of computers or Raspberry Pi-clusters as seen in Figure 1 for experimentation, learning about various technologies. It can include hardware such as lite weight PCs, Raspberry Pi, routers, switches, and storage devices, along with software and virtualized environments to host various applications. They provide a practical way for hobbyists, developers, and IT professionals to explore and develop their hosted services in a controlled, isolated environment.

Figure 1 Raspberry Pi-Cluster in a Homelab setup comprising of three Raspberry Pi devices

2. Why Self-Hosted Homelab?

A secure self-hosted Homelab is crucial because it allows individuals to maintain full control over their data, minimizing reliance on third-party services that may pose privacy or security risks. A reference architecture of a Homelab is illustrated in Figure 3. By managing essential services in a controlled environment, users can implement custom security measures, protect sensitive information, and reduce exposure to cyber threats. Furthermore, Homelab can host media servers providing an opportunity to experiment with media management, storage, and streaming technologies. Here are some compelling reasons in favor of self-hosted Homelab services.

2.1 Privacy and Control

A secure Homelab fosters a deeper understanding of cybersecurity practices, enabling users to develop and test their own solutions in a safe, isolated environment. This not only strengthens personal privacy but also enhances overall digital resilience. For instance sensitive password data can be stored on-site, leveraging full control over the data without relying on third-party vendors which had instance of security breach [2]. Eg. Selfhsoted Password managers

2.2 Cost-Effective Testing and Development

By using existing hardware or inexpensive devices like Raspberry Pi or old laptops, a Homelab offers a cost-effective solution for developers to test software, experiment with configurations, or run multiple services. It avoids the need to incur the costs of cloud computing services or renting servers.

2.3 Containerized Services and Orchestration

Docker makes it easy to deploy and manage a variety of self-hosted services in a Homelab. Docker containers offer an efficient and flexible way to deploy, scale, and manage applications in a home-lab environment, with containers, users can easily set up, scale, and update various applications—such as media servers, databases, and web apps—while maintaining isolation and efficiency. Kubernetes enhances this by automating container orchestration, ensuring optimal performance and resource management. The extensive community [3] support and pre-configured images streamline the setup and maintenance of these services. Here's how Docker enhances the home-lab experience:

2.3.1 Consistent Environments

Docker ensures consistency across different systems, eliminating the "it works on my machine" problem. This consistency is crucial for replicating real-world scenarios and ensuring reproducibility of experiments

2.3.2 Resource Efficiency

Containers are lightweight and share the host system's kernel, making them more resource-efficient than traditional virtual machines. This allows users to run more services on the same hardware

2.3.3 Rapid Deployment and Scaling

Docker enables quick deployment of applications and easy scaling. Users can quickly spin up new instances of services or entire environments for testing and learning purposes

2.3.4 Isolation and Security

Docker's security model, which includes features like kernel namespaces, control groups, and restricted capabilities, provides a good balance between isolation and efficiency

2.4 Learning and Skill Development

Homelabs are an excellent platform for experimenting with various technologies, such as virtualization, networking, cloud computing, or containerization. They provide an environment for practical learning, helping users understand the concepts and tools that are used in real-world enterprise environments.

3 Suitable Hardware for Homelab Solutions

When setting up a Homelab, selecting the right hardware is crucial to ensure efficient performance, scalability, and cost-effectiveness. The hardware must meet the specific demands of the services being run while staying within the user's budget. For a Homelab that utilizes Docker containers for various services, it is important to consider factors like processing power, memory, storage capacity, and power consumption. Two primary categories of devices that are widely used for Homelab solutions are **Raspberry Pi (RPi)** and **Intel Atom-based devices**.

3.1 Raspberry Pi (RPI): A Suitable Device for Homelabs

The **Raspberry Pi (RPI)**, particularly its latest models (such as the RPi 5), has gained immense popularity in the Homelab space due to its affordability, energy efficiency, and flexibility. Let's explore why the RPi is a suitable choice for running Homelab services.

3.1.1 Low Cost and Accessibility

One of the most compelling reasons for choosing the Raspberry Pi for a Homelab setup is its low cost. At the time of this paper, a basic Raspberry Pi 5 [4] board without accessories can cost as little as \$50–\$125, depending on the model and configuration (e.g., 2GB, 4GB, 8GB or 16GB of RAM). This affordability makes it accessible to tech enthusiasts, students, and individuals on a tight budget who wish to build an efficient and scalable Homelab without significant upfront investment.

3.1.2 *Power Efficiency and Compact Size*

The Raspberry Pi consumes as little as 2.5–7 watts [4] [5] of power (depending on the model). This energy efficiency is particularly beneficial for users who want to run 24x7 services with minimum power costs and reduce their environmental impact. The compact size of the Raspberry Pi (approximately the size of a credit card) means it requires very little physical space, users can stack several units or place them in various configurations to create a clustered environment for scaling workloads without occupying much space.

Figure 2 A Raspberry Pi Zero cluster consisting of three Pi Zero W devices

3.1.3 *Flexibility and Expandability*

The Raspberry Pi is highly flexible, with the ability to run docker containers, it can be used for a wide variety of applications, such as web servers, media servers, or even as a network-attached storage (NAS) device. The Raspberry Pi5 comes with a PCIe 2.0 [4] interface, capable of delivering higher transfer speeds and endless possibilities to add additional peripherals.

3.1.4 *Community and Documentation*

The Raspberry Pi has an extensive community [6] [3] of users and a wealth of resources, guides [6], and tutorials [3] available. This makes it easier for beginners to set up and troubleshoot their Homelab services. The open-source nature of the RPi ecosystem encourages experimentation and contributes to the growing repository of solutions for common Homelab use cases.

3.1.5 *Limitations of Raspberry Pi*

Although the Raspberry Pi is highly suitable for many Homelab applications, it does have some limitations:

- **Processing Power**: While the Raspberry Pi 5 is more powerful than its predecessors, it is still not as powerful as an Intel mini pcs. This can limit performance when running resource-intensive applications or handling large-scale workloads.
- **Networking**: The built-in networking on the Raspberry Pi 5 (Gigabit Ethernet) [4] is sufficient for most small Homelab use cases, but it may become a bottleneck when handling high-throughput applications like large-scale databases or video streaming.
- **No Built-In Storage**: The Raspberry Pi relies on microSD cards for its primary storage, which is not as durable or fast as traditional SSDs or HDDs. External storage options are required for better performance. This issue can be mitigated by using the latest Pi5 device that supports PCIe, allowing for the direct connection of an SSD.

3.1.6 Ideal Use Cases for Raspberry Pi in Homelabs

- Running lightweight services such as **Pi-hole**, **Nextcloud**, or **Grafana** [7] for monitoring small-scale infrastructure
- Creating a **media server** (Jellyfin or Plex) for streaming content on local devices
- Hosting **GitLab** or **Nextcloud** for personal cloud storage
- Building **home automation systems** or testing **IoT (Internet of Things)** devices

3.2 Intel-Based Devices: A More Powerful Option for Homelabs

While the Raspberry Pi is suitable for smaller and less resource-demanding Homelab environments, **Intel-based devices** (e.g., Intel NUCs, mini PCs, or traditional desktop systems) provide more processing power and versatility. These devices are often chosen by users who need more intensive computational capabilities or are running more demanding applications.

3.2.1 Intel NUC: Compact and Powerful

The **Intel NUC (Next Unit of Computing)** is a small-form-factor PC that comes with a variety of processors, including Intel Atom Processors. These mini PCs are excellent for users who need more computing power than a Raspberry Pi can provide but still want a compact and energy-efficient device.

- **Processing Power**: The Intel N100 [8] processor offers significantly more computing power compared to the Raspberry Pi, making it ideal for running more demanding applications like Prometheus, Jellyfin, or complex data analytics tools.
- **RAM and Storage**: The Intel NUC typically supports higher RAM configurations (up to 64GB depending on the model) and faster storage options like SSDs, which improves overall system responsiveness and service performance. This is especially useful when dealing with large datasets or running multiple containers simultaneously.
- **Networking**: Intel NUC devices generally have better network interfaces than the Raspberry Pi, including gigabit Ethernet and sometimes even Wi-Fi 6. This makes them suitable for high-performance network applications like VPN servers or data-intensive services.
- **Expandability**: The NUC allows users to easily expand storage and RAM, which makes it a more flexible choice for growing Homelab setups. It also supports multiple displays and peripherals, which is useful for more advanced use cases.

3.2.2 Traditional Intel Desktops

In some cases, users may opt to repurpose older desktop systems with Intel processors for Homelab use. These systems can provide much more processing power, storage, and memory compared to single-board computers like the Raspberry Pi. For example, a system with an Intel i7 processor, 16GB of RAM, and a 1TB SSD would be capable of running multiple resource-intensive Docker containers with ease.

- **More Storage and RAM**: Traditional desktops typically offer more storage and memory, allowing for the hosting of large databases or heavy media servers.
- **Full Operating System Support**: Intel-based devices generally support a broader range of operating systems (e.g., Linux, Windows, or specialized server OS like Ubuntu Server). This provides greater flexibility in terms of which services and configurations can be deployed.
- **Reliability**: Desktop devices typically provide more stability and redundancy in hardware (e.g., RAID configurations for storage) than single-board computers.

3.2.3 Ideal Use Cases for Intel Devices in Homelabs

- Hosting resource-demanding applications like **Prometheus**, **Grafana**, or **Jellyfin** with multiple users
- Building a **private cloud** with services like **Nextcloud** and **Docker Swarm**
- Running a **VPN server**, **home security systems**, or multiple **virtual machines** for testing complex setups

- Managing large-scale data storage with more robust hard drive options (e.g., HaT [9] with several SSDs)

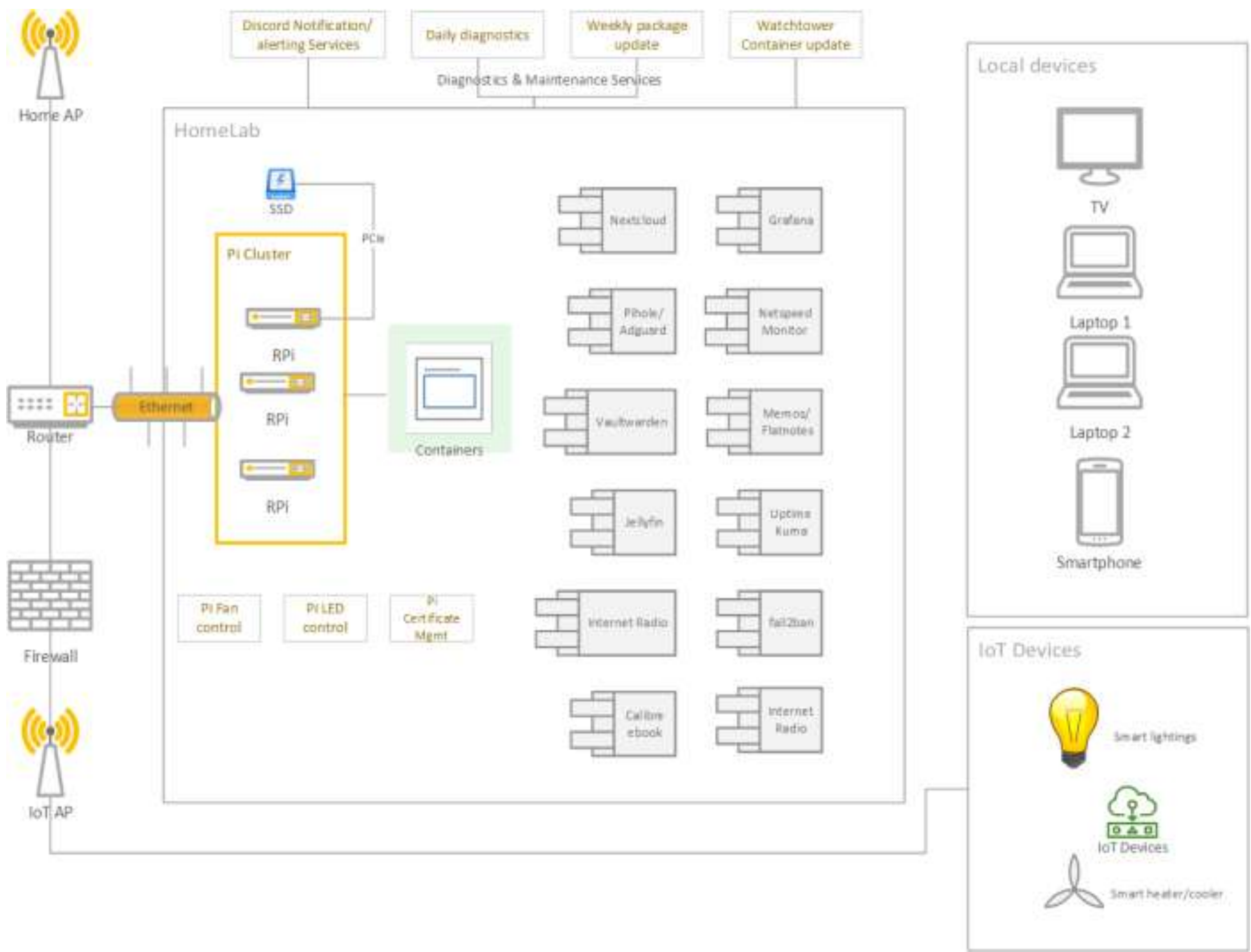

Figure 3 Illustration of setting up a Homelab with various services and decentralizing non-secure devices, along with managed services for updates & diagnostics

4 Advantages of Homelab

4.1 Hands-on Experience

A Homelab enables users to gain practical experience that theoretical knowledge alone cannot provide. Building, configuring, and troubleshooting real-world setups enhances understanding and prepares users for industry-level task.

4.2 Full Control Over Resources

With a Homelab, users have complete control over hardware, software, network settings, and configurations. This level of control allows for experimentation with different setups and a deeper understanding of the underlying systems.

4.3 Flexibility and Customization

Homelabs can be tailored to meet the specific needs of the user, whether for developing a specific application, creating a private cloud, hosting services, or simulating network setups. Docker containers, virtualization, and open-source tools enable the creation of highly customizable environments.

4.4 Enhanced Security and Privacy

Running services and applications locally in a Homelab ensures that sensitive data does not reside on external servers or third-party cloud platforms. For those concerned with privacy, this can be a major advantage, as users can implement their own security policies.

4.5 Lower Costs

Rather than paying for cloud services or renting data center space, setting up a Homelab can be done with minimal cost by repurposing old hardware or using affordable solutions like Raspberry Pi or Virtual Machines (VMs). Additionally, open-source tools help avoid licensing costs.

5 Disadvantages of Homelab

5.1 Limited Resources

The most significant drawback of a Homelab is that it is constrained by the resources of the user's hardware. Homelabs usually cannot scale as seamlessly as cloud-based environments and are subject to limitations in processing power, memory, and storage capacity.

5.2 Complexity in Setup and Maintenance

Setting up a Homelab can be technically challenging, especially for beginners. It may require familiarity with system administration, networking, and troubleshooting. Additionally, maintaining a Homelab with frequent updates and ensuring all services remain operational can become time-consuming.

5.3 Power and Space Consumption

A Homelab often involves running multiple machines or servers, which can be power-hungry and require substantial physical space. This can lead to increased electricity bills and demand for dedicated space in a home.

5.4 Reliability Issues

Homelabs are typically not designed with redundancy or failover mechanisms in place. In the event of hardware failure, users may experience downtime, resulting in loss of services or data unless proper backup systems are in place.

5.5 Security Vulnerabilities

While a Homelab provides privacy, it also places the responsibility for securing the infrastructure entirely on the user. Without proper security measures like firewalls, encryption, or intrusion detection, a poorly secured Homelab could expose users to potential cyber threats.

6 Services at a Glance

All services in a Homelab are typically run as Docker containers. Docker provides a lightweight, efficient method to run applications in isolated environments, making it easy to deploy, manage, and scale services.

Figure 4 Grafana Dashboard of a selfhosted Pi-Cluster [10]

The following is a list of commonly deployed services [3] in a Homelab:

Category	Description of the Self hosted service
File Management and backup	**Nextcloud**: A self-hosted cloud platform that provides file synchronization, sharing, and collaboration. It supports various apps like calendars, contacts, and email integration. **Syncthing**: A peer-to-peer file synchronization tool, Syncthing securely syncs files across devices without using centralized servers.
Password & Key Management	**Bitwarden** is a password manager that helps users securely store and manage passwords, generate strong passwords, and sync them across different devices. Running Bitwarden in a Homelab ensures that sensitive credentials are kept secure preventing adversaries [2] who tend to target third-party cloud services.
Network and Security	**Pi-hole/ Adguard**: Network-wide ad blocker that acts as a DNS sinkhole, blocking ads and tracking domains for improved privacy and performance. **WireGuard**: A modern, fast VPN protocol that ensures secure communication over the internet by creating private, encrypted tunnels. **Traefik**: A dynamic reverse proxy designed to handle modern microservices and Docker environments, simplifying routing and load balancing for applications.
Databases	**MySQL**: A widely used open-source relational database management system, MySQL powers many websites and applications, known for its reliability and ease of use. **PostgreSQL**: An advanced open-source relational database that supports complex queries, scalability, and a wide range of data types, suitable for demanding applications.
Home Automation	**Home Assistant**: An open-source home automation platform that integrates with a wide range of smart devices to automate tasks, monitor systems, and create custom routines.
Monitoring and Analysis	**Grafana**: A powerful open-source data visualization [10] and monitoring tool, commonly used for visualizing metrics and monitoring applications in real-time. A reference analysis can be seen in Figure 4.

	Prometheus: An open-source monitoring and alerting toolkit designed for reliability and scalability, particularly suited for dynamic cloud-native environments. **cAdvisor** (Container Advisor) is a tool for monitoring and analyzing container performance.
Dashboard	**Homer** is a simple, customizable dashboard for accessing various self-hosted services from a central page as in Figure 5. **Heimdall** is a highly customizable easy to use dashboard for accessing self-hosted applications.
Media Servers	**Plex**: A popular media server that allows users to stream movies, TV shows, and music to various devices and support for multiple formats. **Jellyfin**: An open-source alternative to Plex, Jellyfin allows users to organize and stream their media content across devices with similar functionality.
Digital Library	**Calibre** is an eBook management tool that helps users organize, convert, and read eBooks. In a Homelab environment, it can be used to create a personal digital library accessible from any device.
Security	**Fail2ban** is a security tool that protects servers from brute-force attacks. It works by monitoring logs for suspicious activity and blocking IP addresses that repeatedly fail login attempts. It is commonly used in Homelabs to enhance security.
Alerting	**Uptime Kuma** is a self-hosted monitoring tool that helps users track the status of websites and services. It provides alerts and detailed logs when services go down, helping ensure that critical applications are running as expected.
Update Management	**Watchtower** is an automatic update service for Docker containers. It helps ensure that all your containers are running the latest versions by regularly checking for updates and automatically restarting containers with the new images.

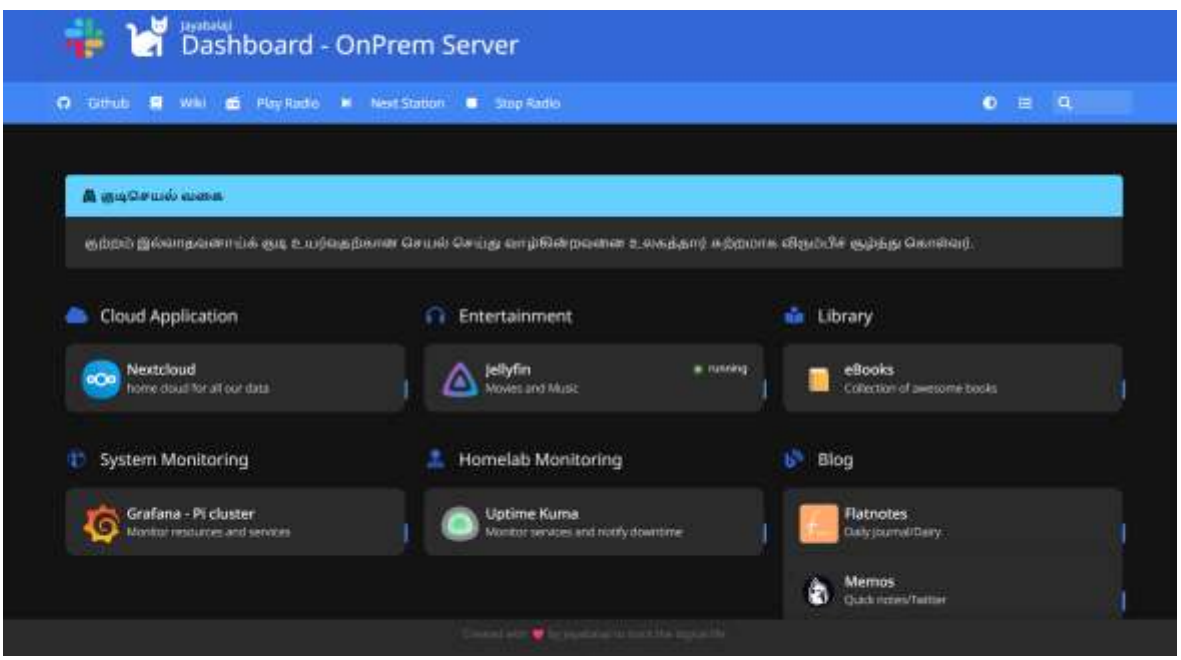

Figure 5 Homer Dashboard listing all the various services that can be accessed from the main page

7 Choice of hardware

In summary both **Raspberry Pi** and **Intel-based devices** have their own merits when it comes to setting up a Homelab. The Raspberry Pi5 is an excellent choice for lightweight, cost-effective solutions with low power consumption, ideal for experimenting with simpler services and learning about networking and containerization. However, for more power-intensive applications and scalability, Intel-based devices [11] (such as Intel NUCs or Intel N100 mini pc) provide more robust computing power, memory, storage, and flexibility. The ideal choice between these two options depends on the specific needs of the user, such as the scale of the Homelab, the services being run, and the desired level of performance [5]. By carefully evaluating these factors, users can create an optimized Homelab environment that fits their personal or professional objectives.

8 Conclusion

In conclusion, setting up a Homelab offers invaluable opportunities for individuals to improve security & privacy, experiment, and optimize their understanding of modern technologies. The ready to deploy docker containers [3] and an active tech community enables easy hosting of personal services like cloud storage or media streaming, a Homelab provides a tailored environment to achieve these goals. The choice between cost-effective Raspberry Pi devices or more powerful Intel-based systems depends on the user's specific needs and desired scalability [5]. As digital infrastructure demands increase, Homelabs are poised to become increasingly accessible and integral for tech enthusiasts, developers, and small businesses seeking innovation, technical expertise enhancement, and complete data control.

9 Call to Action

Embark on the exciting Homelab journey, whether you aim for a simple media server powered by jellyfin or plex into advanced system monitoring with Prometheus and Grafana, the possibilities are endless. Begin your journey by selecting a hardware platform (Raspberry Pi or Intel device) that aligns with your specific needs and project scope, gathering essential components like storage, network equipment, and power supplies. It can empower any individual to control their data, enhance security, and mitigate security threats while keeping software up to date.

Acknowledgement

The author would like to extend sincere gratitude to the reviewers, particularly Dr. Thorsten Zenner, professor at University of Reutlingen, Germany, for his insightful and constructive feedback. His meticulous review significantly enhanced the clarity and which is instrumental in shaping the final version of this white paper.

Bibliography

[1] F. H. Chowdhury, The Docker Handbook – Learn Docker for Beginners, FreeCodeCamp, 2021.

[2] "Cybersecurity Dive : LastPass breach timeline: How a monthslong cyberattack unraveled," [Online]. Available: https://www.cybersecuritydive.com/news/lastpass-cyberattack-timeline/643958/. [Accessed Nov 2024].

[3] "Github Awesome-selfhosted," Dec 2024. [Online]. Available: https://github.com/awesome-selfhosted/awesome-selfhosted.

[4] "Official Raspberry Pi documentation," Jan 2025. [Online]. Available: https;//www.raspberrypi.org/documentation/.

[5] P. S. C. Benchmark, "CPU Benchmark of different CPU classes: Mobile/Embedded," Nov 2024.

[6] "reddit selfhosted forum," Aug 2024. [Online]. Available: https://www.reddit.com/r/selfhosted/.

[7] "Official Grafana Documentation," Nov 2024. [Online]. Available: https://grafana.com/docs/.

[8] I. Corporation, "Intel Processor N100," Intel Corporation, Jan 2025. [Online]. Available: https://www.intel.com/content/www/us/en/products/sku/231803/intel-processor-n100-6m-cache-up-to-3-40-ghz/specifications.html.

[9] J. Geerling, "PoE HATs for Pi 5 add NVMe," [Online]. Available: https://www.jeffgeerling.com/blog/2024/3rd-party-poe-hats-pi-5-add-nvme-fit-inside-case. [Accessed Aug 2024].

[10] "Raspberry Pi Docker Monitoring using Grafana Dashboard," March 2024. [Online]. Available: https://grafana.com/grafana/dashboards/15120-raspberry-pi-docker-monitoring/.

[11] "Anand Tech," Dec 2024. [Online]. Available: https://www.anandtech.com/tag/nuc.