# ****Designing a Cybersecurity Homelab for Threat Detection and Network Monitoring****

Edison Bujupi

Student, BA Cyber Security

AAB College

**Abstract**—In cybersecurity, applying and practicing security principles can be challenging without a controlled and secure environment for hands-on experience. That’s the mindset I had when starting this project. This home lab serves as a step-by-step guide for setting up, fine-tuning, and securing an IT infrastructure. While it operates on a smaller scale, the skills and concepts learned here are directly transferable to real-world, enterprise-level systems.

**Keywords**: Cybersecurity, Home Lab, Penetration Testing, Security Operations Center, pfSense, Splunk, Log Analysis

I. INTRODUCTION   
A homelab is a personal setup built to develop and refine skills in a specific domain typically from the comfort of your own home. This particular lab mirrors key components and tools found in enterprise environments, providing a secure space to explore, experiment, and understand how real-world systems function.

**Building the Host System**

For this lab, I used my **Lenovo Gaming Ideapad 3**, which provides more than enough performance to run a full Security Operations Center (SOC) homelab. The hardware specifications are as follows:

* **CPU:** AMD Ryzen 5 5600H 6-Core Processor
* **RAM:** 16 GB DDR4 Memory
* **STORAGE:** 512 GB NVMe SSD
* **GRAPHICS CARD:** NVIDIA GeForce GTX 1650 4 GB
* **HOST OPERATING SYSTEM:** Windows 11 Home
* **VIRTUALIZATION PLATFORM:** Oracle VirtualBox

This setup offers sufficient computing power to run multiple virtual machines, perform network monitoring, simulate attacks, and run SIEM tools like Splunk efficiently.

# Tools/OS Installed

* VirtualBox
* Kali Linux
* Windows 11
* Windows Server (Active Directory)
* Metasploit Framework
* pfSense (Firewall/Router)
* Splunk (SIEM)

Table of Figures

[Figure 1 pfSense Configuration Overview 3](#_Toc193910300)

[Figure 2 pfsense Configuration Overview 2 3](#_Toc193910301)

[Figure 3 Defined VLAN Interfaces in pfSense 4](#_Toc193910302)

[Figure 4 VLAN10 Static IP Configuration 5](#_Toc193910303)

[Figure 5 DHCP and DNS Configuration for VLAN10 6](#_Toc193910304)

[Figure 6 DHCP and DNS configuration for VLAN 20 7](#_Toc193910305)

[Figure 7 DHCP and DNS configuration for VLAN 30 8](#_Toc193910306)

[Figure 8 DHCP and DNS configuration for VLAN 40 9](#_Toc193910307)

[Figure 9 Assigned Interfaces to VLANs 10](#_Toc193910308)

[Figure 10 Firewall Rules 10](#_Toc193910309)

[Figure 11 Automatic Outbound NAT Configuration 11](#_Toc193910310)

[Figure 12 Ping Test to Kali VM 12](#_Toc193910311)

[Figure 13 Ping Test to Windows VM 12](#_Toc193910312)

[Figure 14 Remote Syslog Logging to Splunk 13](#_Toc193910313)

[Figure 15 Syslog Logs in Splunk from pfSense 13](#_Toc193910314)

[Figure 16 Nmap Scan from Kali to Windows 14](#_Toc193910315)

[Figure 17 Splunk Sourcetype Breakdown 14](#_Toc193910316)

[Figure 18 Splunk Forwarder Host Mapping 15](#_Toc193910317)

[Figure 19 Splunk Server Class Configuration 16](#_Toc193910318)

[Figure 20 Indexed Events in Splunk 16](#_Toc193910319)

[Figure 21 Server Manager - Local Server Summary 17](#_Toc193910320)

[Figure 22 Active Directory 17](#_Toc193910321)

[Figure 23 Server Manager 18](#_Toc193910322)

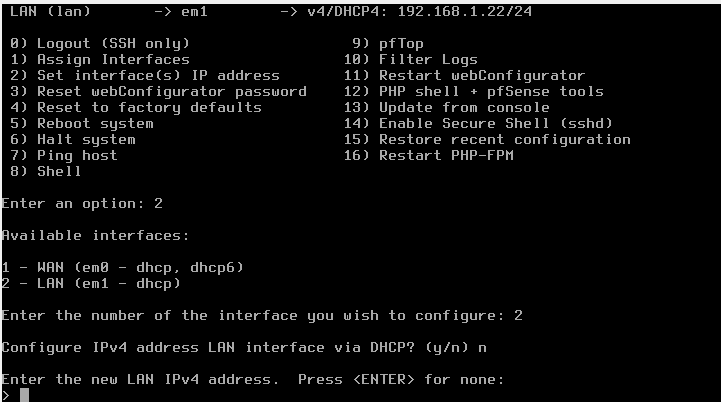


Figure pfSense Configuration Overview

In this lab, **pfSense** is configured as the central **firewall and router** to segment and protect the private homelab network.

# 

Figure pfsense Configuration Overview 2

It plays a key role in managing traffic between VLANs, enforcing security policies, and providing network services like DHCP and DNS.Access to the **pfSense web interface** is **restricted to the Kali Linux machine**, ensuring that only authorized devices can configure or monitor the firewall. This isolation simulates a real-world security practice where management interfaces are placed in restricted zones, accessible only by secure administrative systems.

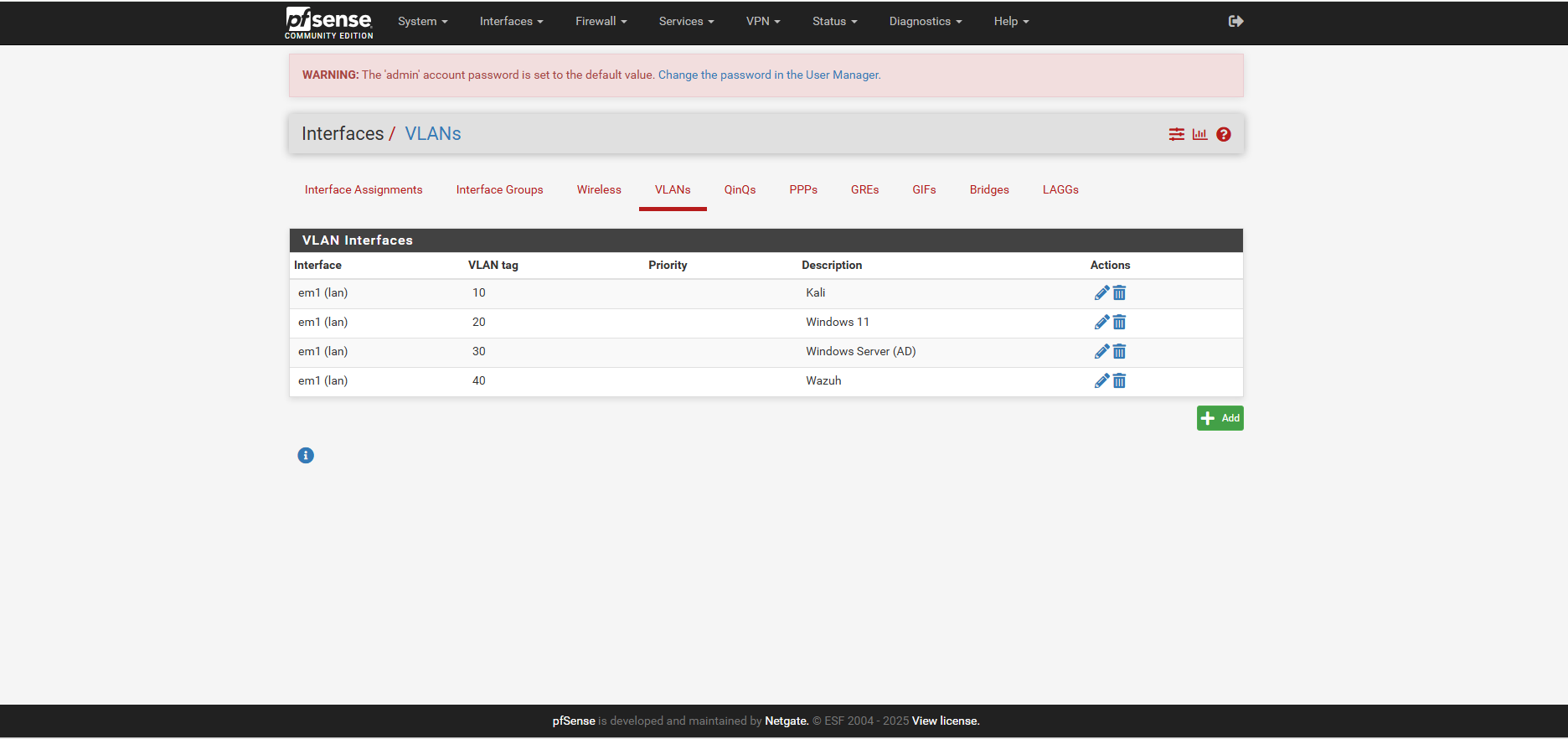


Figure Defined VLAN Interfaces in pfSense

In this homelab setup, multiple **VLAN interfaces** were created in pfSense to logically segment the network into isolated environments. Each VLAN represents a different zone or function within the infrastructure:

* **VLAN10 – Kali Linux (192.168.10.0/24):** Used for penetration testing and administrative access.
* **VLAN20 – Windows 11 (192.168.20.0/24):** Simulates a standard user workstation.
* **VLAN30 – Windows Server (AD) (192.168.30.0/24):** Hosts Active Directory and other server roles.
* **VLAN40 – Wazuh (192.168.40.0/24):** Dedicated to the SIEM agent for monitoring and log forwarding.

Each VLAN interface is assigned:

* A **static IP address** (gateway for that subnet),
* A **DHCP range** to dynamically assign IPs to hosts within the VLAN,
* A **DNS server**, often pointing to the VLAN’s gateway or AD DNS.

This segmentation enhances security, enables granular firewall rule enforcement, and mirrors enterprise-level network architecture.

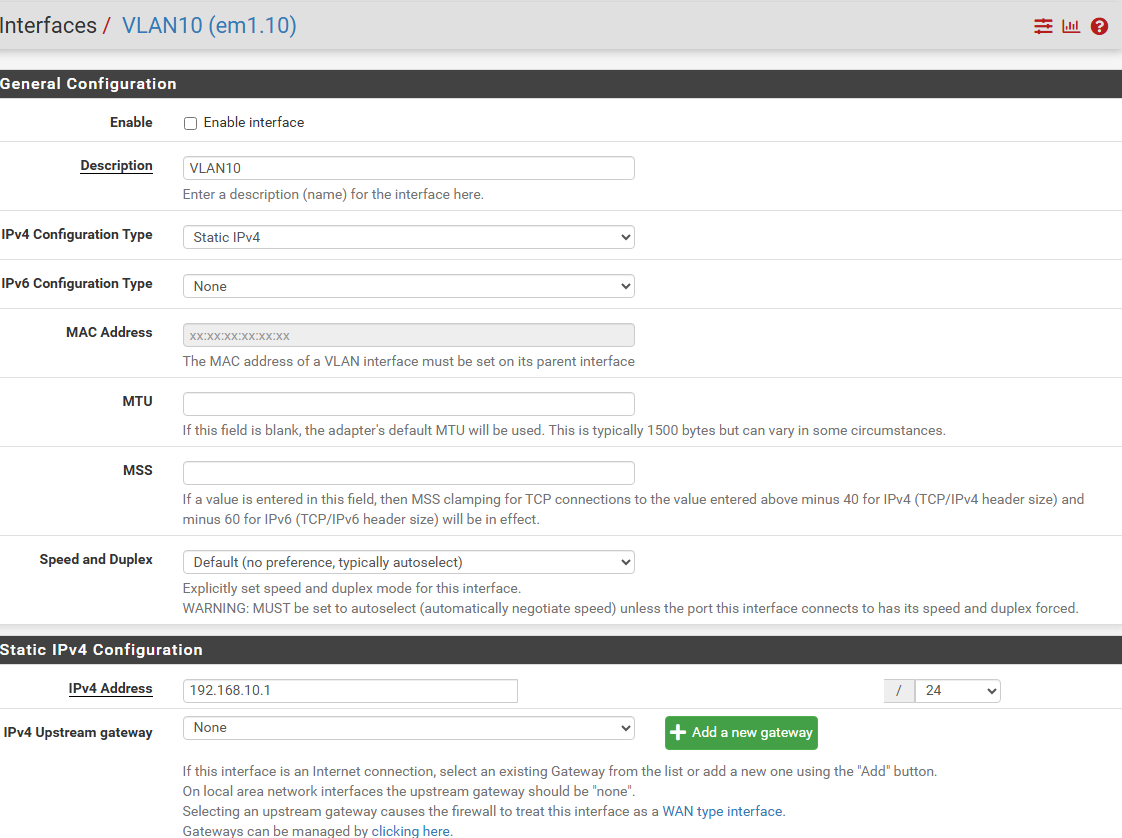


Figure VLAN10 Static IP Configuration

The interface is set to use **Static IPv4** with the address 192.168.10.1/24, acting as the gateway for devices in the VLAN10 subnet. No upstream gateway is defined since this interface is used for internal segmentation, not internet access.

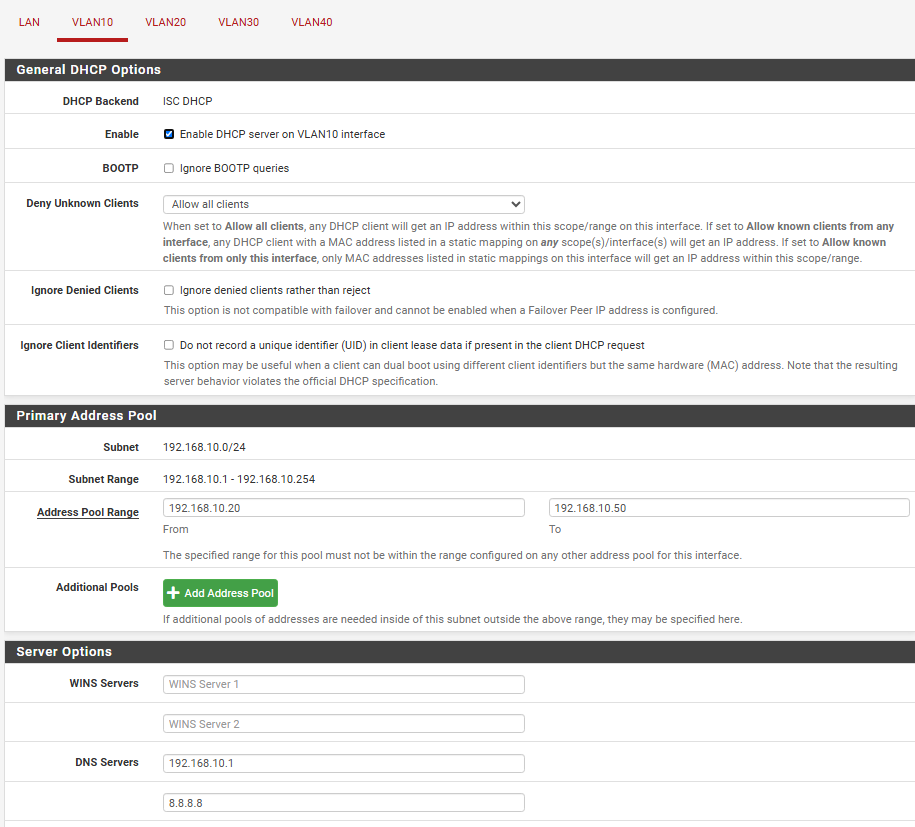


Figure DHCP and DNS Configuration for VLAN10

For **VLAN10**, DHCP and DNS were configured as follows:

* **DHCP Range**: 192.168.10.20 – 192.168.10.50  
  Automatically assigns IPs to devices in VLAN10 for seamless network access.
* **DNS Servers**:
  + 192.168.10.1 (local DNS via pfSense)
  + 8.8.8.8 (Google DNS for external resolution)

This hybrid setup ensures reliable local name resolution and external DNS fallback for internet connectivity.

# 

Figure DHCP and DNS configuration for VLAN 20

**VLAN20**

* **DHCP Range**: 192.168.20.20 – 192.168.20.50
* **DNS Servers**:
  + 192.168.20.1 (local DNS)
  + 8.8.8.8 (external fallback)

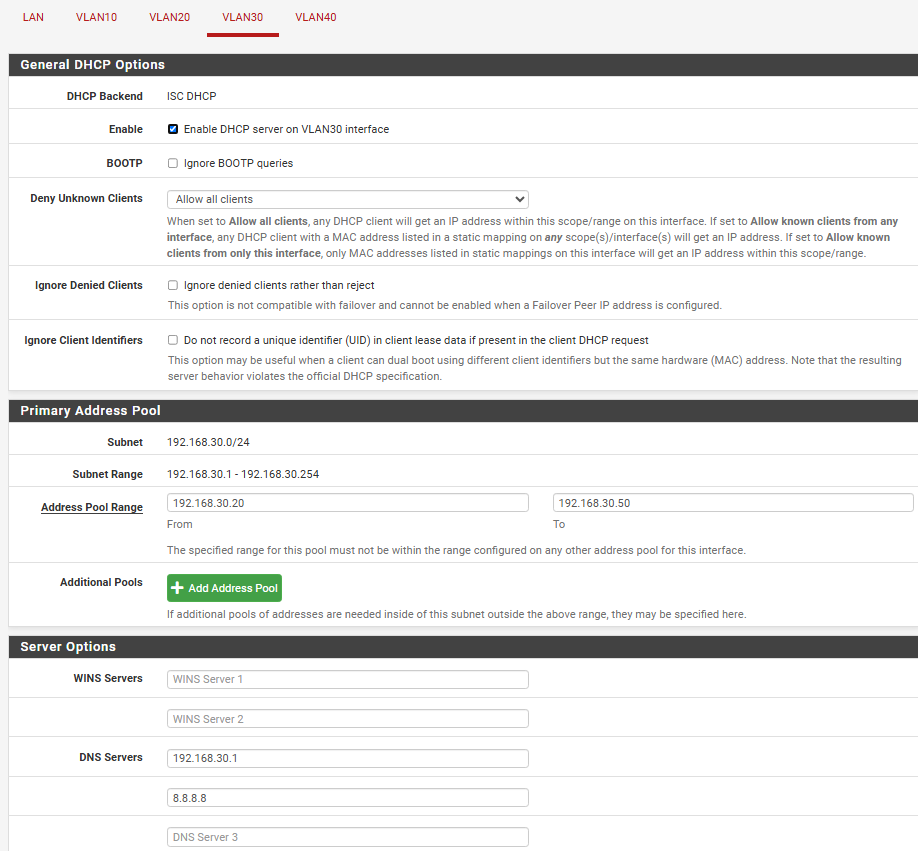


Figure DHCP and DNS configuration for VLAN 30

**VLAN30**

* **DHCP Range**: 192.168.30.20 – 192.168.30.50
* **DNS Servers**:
  + 192.168.30.1 (local DNS)
  + 8.8.8.8 (external fallback)

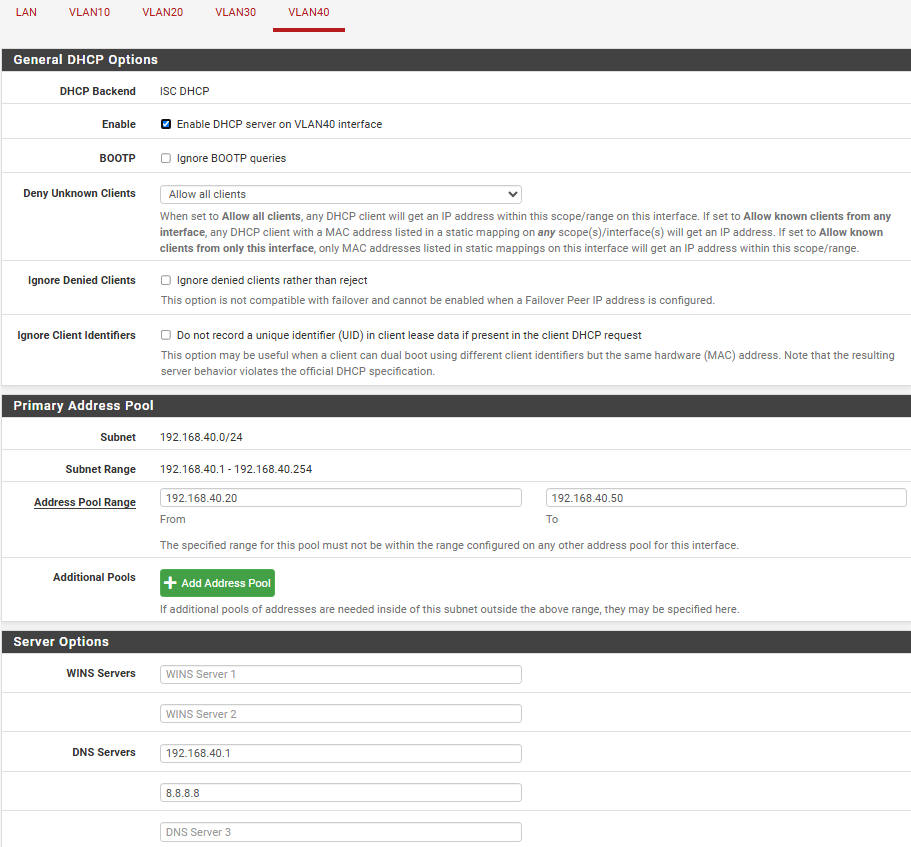


Figure DHCP and DNS configuration for VLAN 40

**VLAN40**

* **DHCP Range**: 192.168.40.20 – 192.168.40.50
* **DNS Servers**:
  + 192.168.40.1 (local DNS)
  + 8.8.8.8 (external fallback)

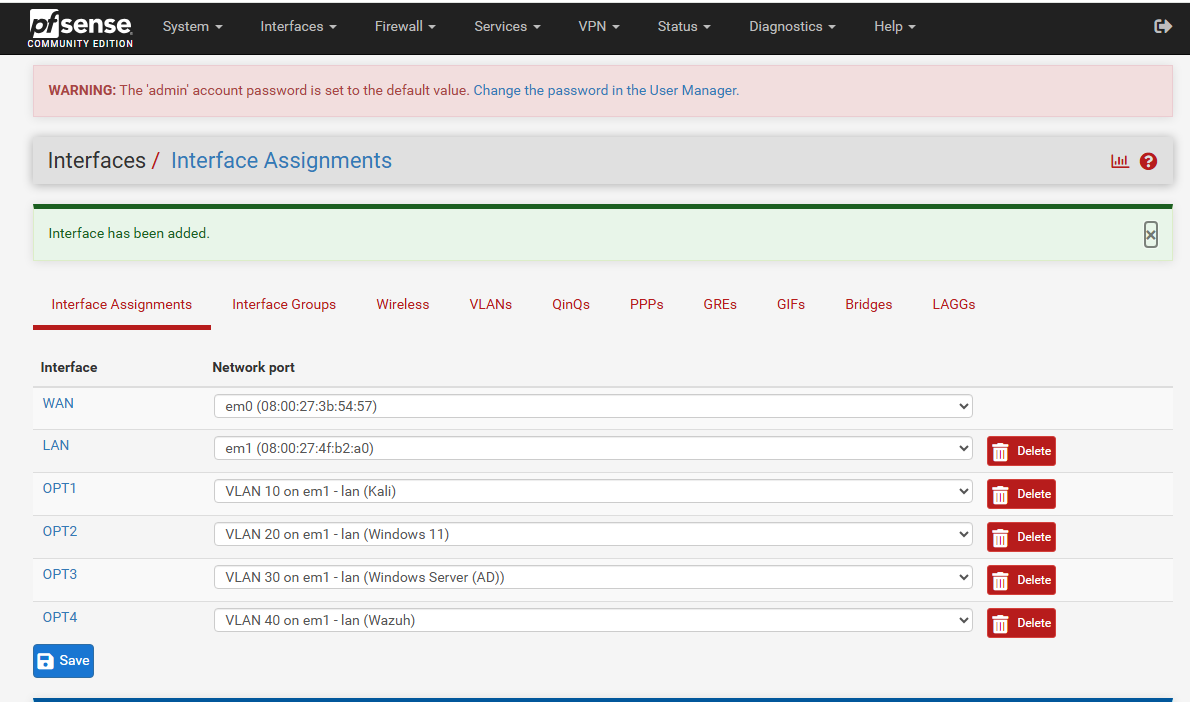


Figure Assigned Interfaces to VLANs

**Interface Assignments**  
Each VLAN is assigned to its own OPT interface, mapped to em1 (LAN) with specific tags for Kali, Windows 11, Windows Server (AD), and Wazuh. This setup enables logical network segmentation and traffic isolation.

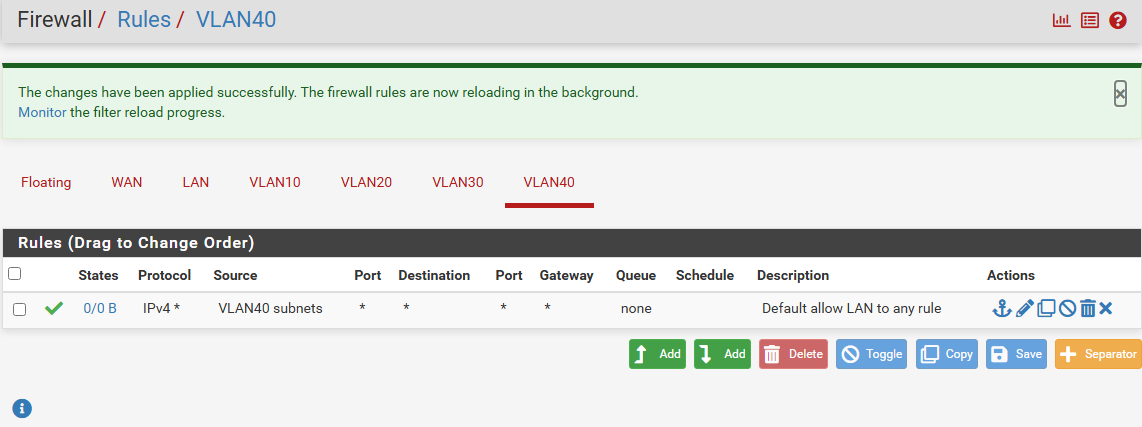


Figure Firewall Rules

**Firewall Rules**  
Firewall rules were added per VLAN to control traffic flow. Each rule allows full access from the respective VLAN subnet to any destination. This ensures basic connectivity and enables services like DHCP, DNS, and internet access while maintaining VLAN-level segmentation.

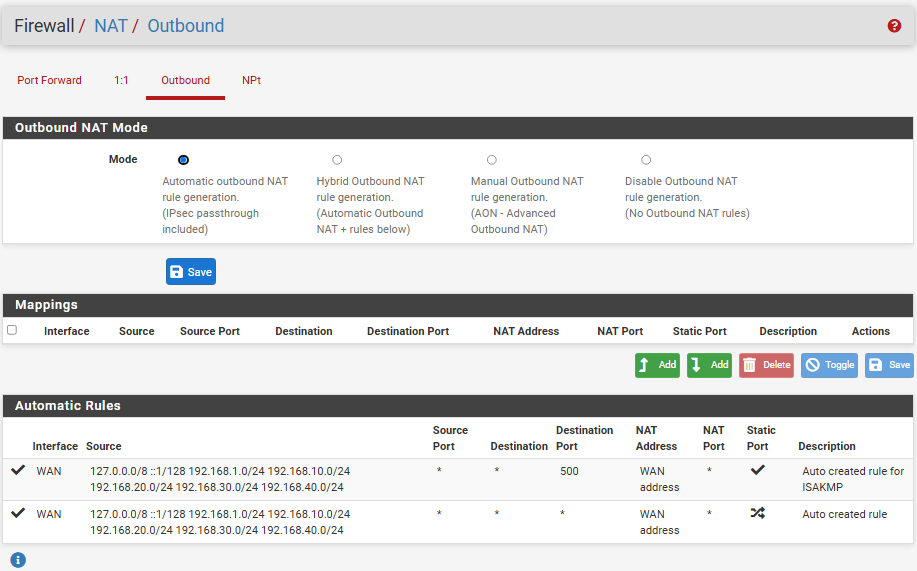


Figure Automatic Outbound NAT Configuration

**NAT Configuration**  
Automatic Outbound NAT mode is enabled to allow internal VLAN networks to reach external destinations (internet) via the WAN interface. pfSense dynamically creates rules for proper address translation, ensuring seamless internet access across VLANs without manual NAT rule setup.

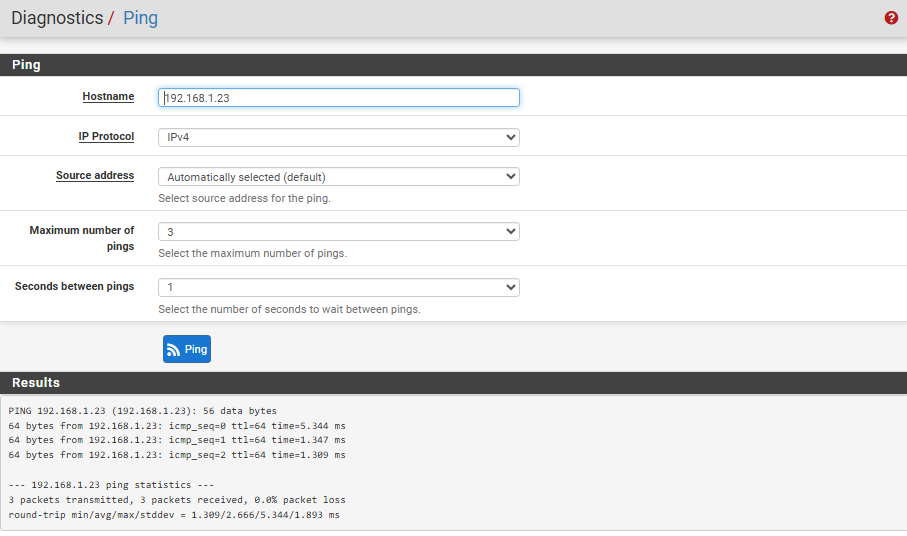


Figure Ping Test to Kali VM

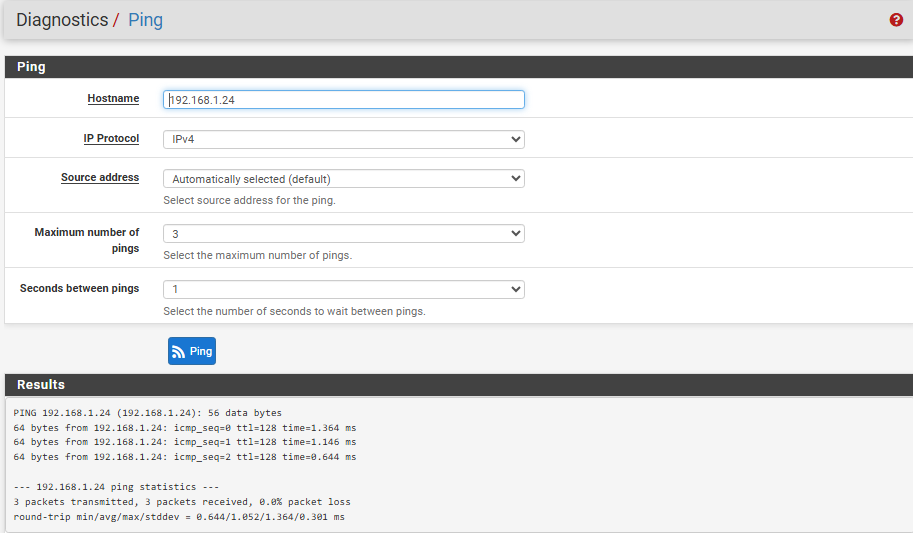


Figure Ping Test to Windows VM

**Connectivity Validation**  
Successful ping responses from pfSense to both Kali Linux and Windows machines confirm proper VLAN interface configuration, routing, and DHCP assignments. This ensures the firewall can communicate with internal network segments, validating overall network reachability.

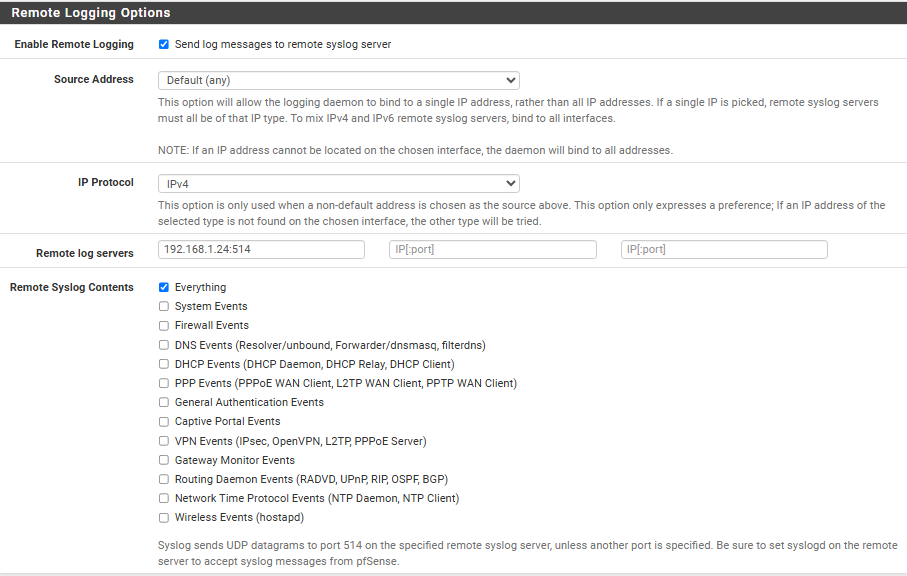


Figure Remote Syslog Logging to Splunk

**Remote Syslog Logging to Splunk**  
This configuration enables pfSense to forward all system logs (firewall, DHCP, authentication, etc.) to a centralized Splunk server (192.168.1.24:514). It ensures complete visibility into network activity, essential for security monitoring, threat detection, and incident response in the SOC lab.



Figure Syslog Logs in Splunk from pfSense

Syslog logs from pfSense are successfully ingested into Splunk for centralized monitoring and analysis.

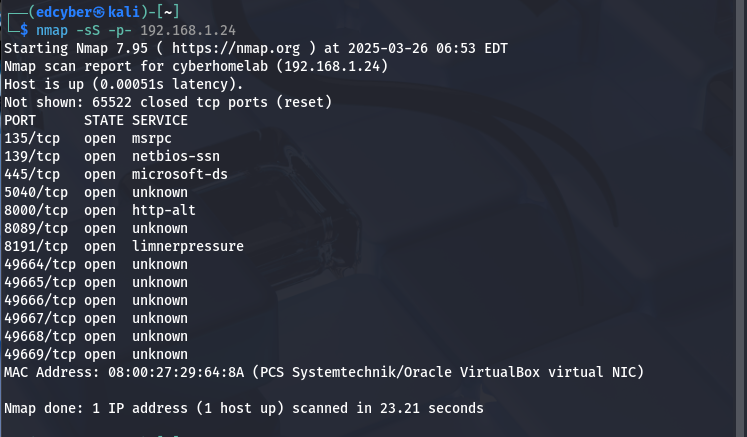


Figure Nmap Scan from Kali to Windows

This Nmap scan reveals numerous open ports on the Windows machine (192.168.1.24). This is due to Windows Security being intentionally disabled in the lab environment to simulate vulnerable systems and allow deeper penetration testing and realistic attack scenarios.

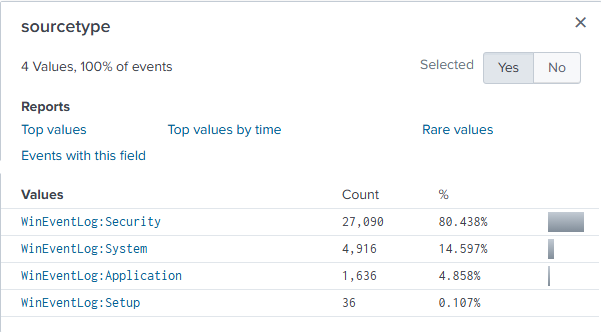


Figure Splunk Sourcetype Breakdown

This Splunk breakdown of sourcetypes shows that most collected events (80.4%) come from the Security log, followed by System, Application, and Setup logs. This highlights successful Windows Event Log monitoring and indicates visibility into authentication, system changes, and application behavior for threat detection.

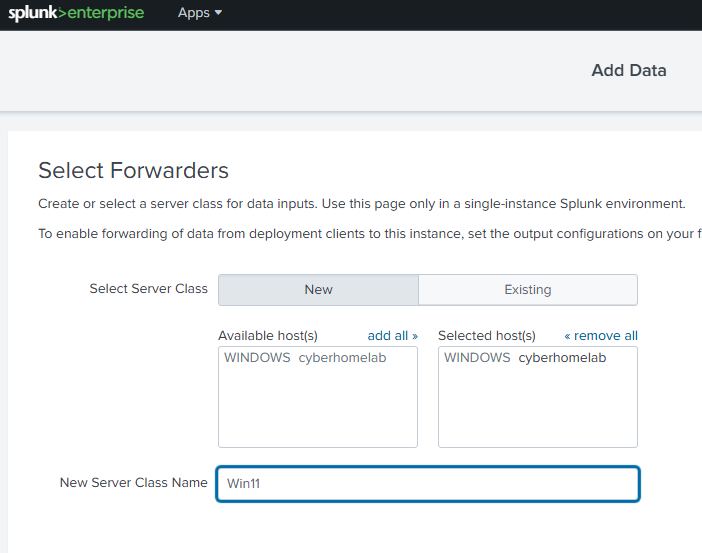


Figure Splunk Forwarder Host Mapping

Splunk forwarder successfully mapped to the host **cyberhomelab**, enabling real-time log forwarding from the Windows machine to Splunk for monitoring and analysis.

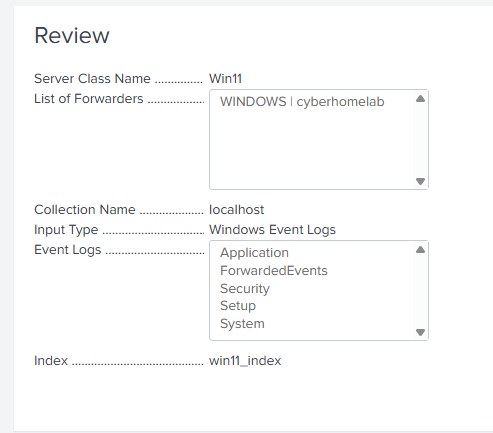


Figure Splunk Server Class Configuration

Splunk forwarder from the Windows 11 machine ("cyberhomelab") is properly connected. A custom index (win11\_index) was created, and all key event logs Application, Security, System, Setup, and ForwardedEvents are being collected, allowing centralized log analysis from the Windows endpoint.

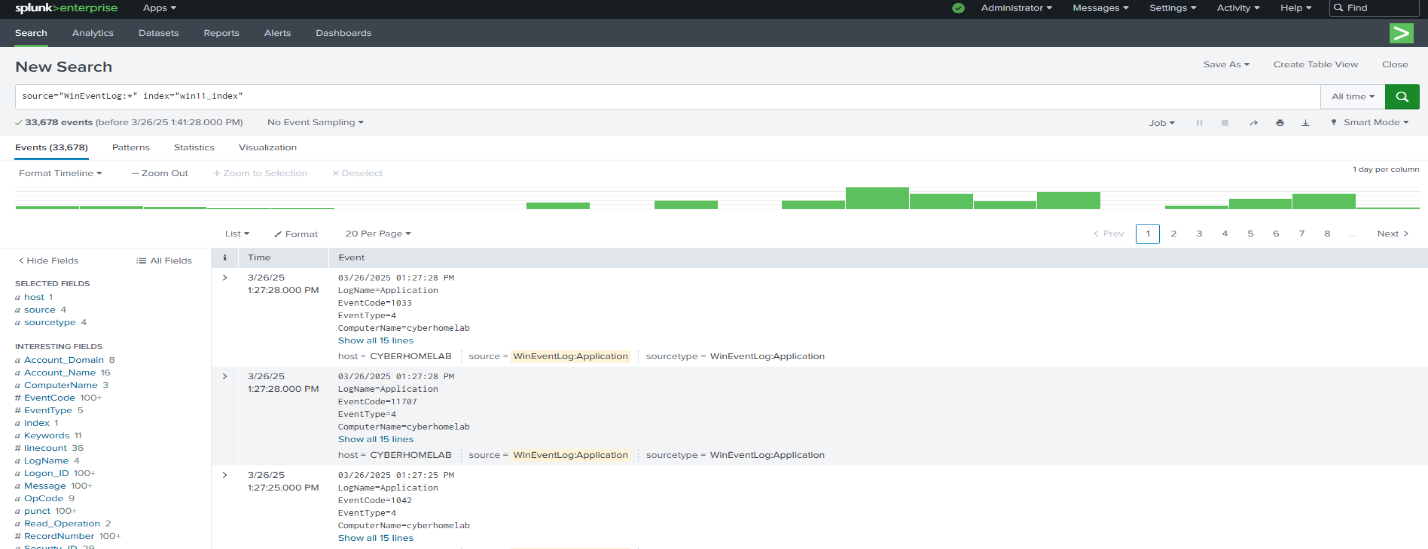


Figure Indexed Events in Splunk

Successful indexing in Splunk, showing parsed Windows event logs (Application, Security, etc.) stored in the custom win11\_index, ready for analysis and detection use cases.

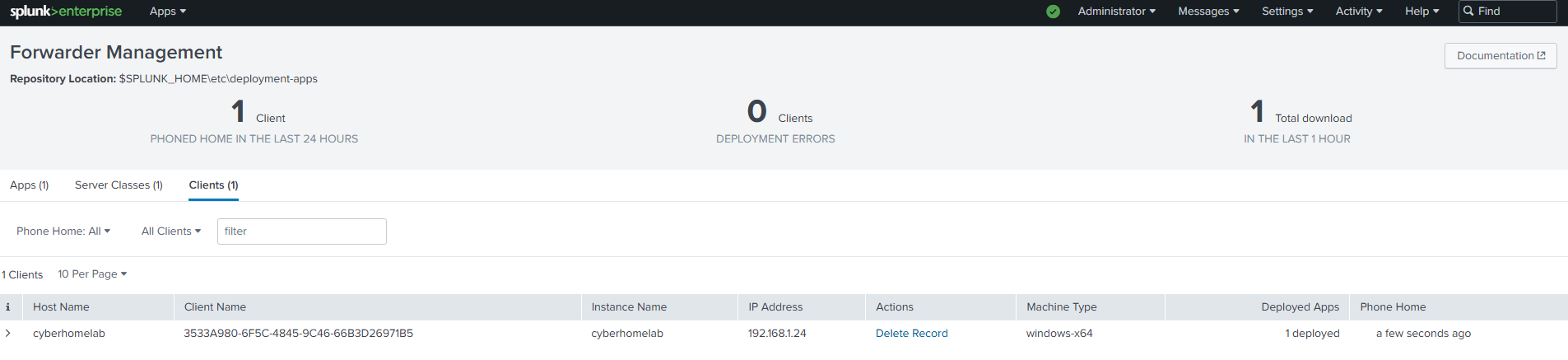


Figure Server Manager - Local Server Summary

Windows forwarder (cyberhomelab) is successfully connected to the Splunk deployment server, showing active communication, IP address, and deployed app status.

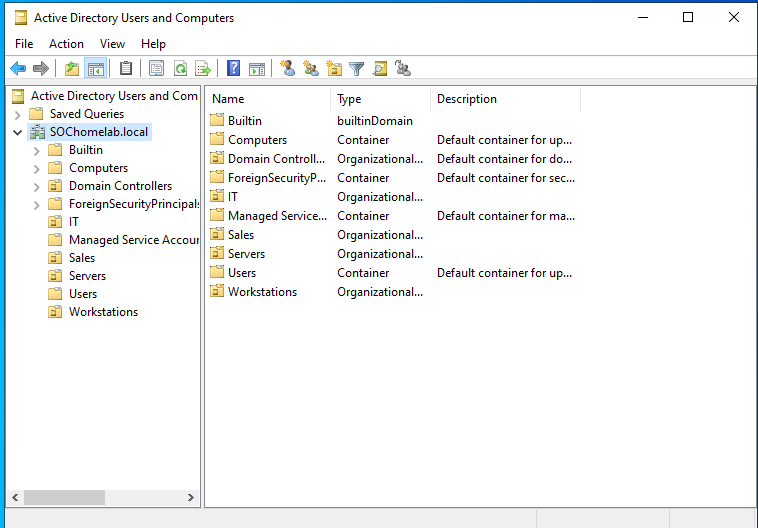


Figure Active Directory

**Active Directory structure** inside the configured domain SOChomelab.local. I created custom **Organizational Units (OUs)** including IT, Sales, Servers, Users, and Workstations to simulate a real enterprise structure. These OUs allow for **role-based access control**, **Group Policy Object (GPO)** targeting, and improved **administrative organization** within the networked environment.

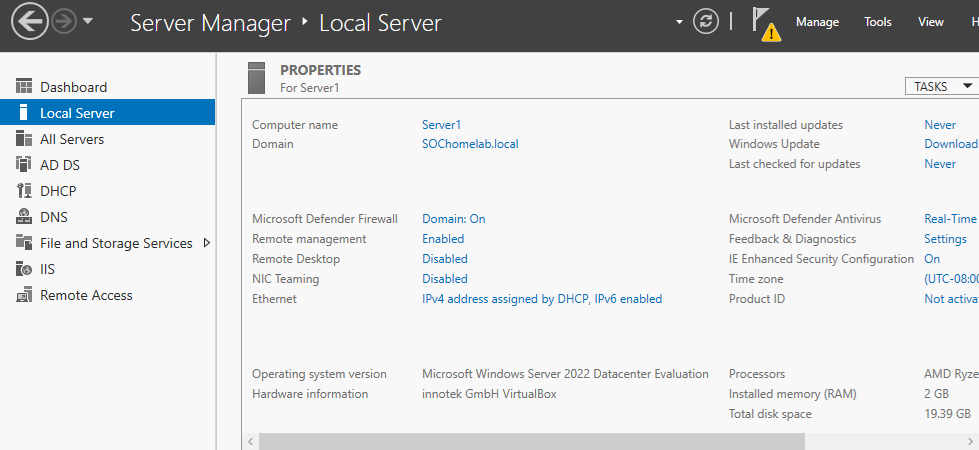
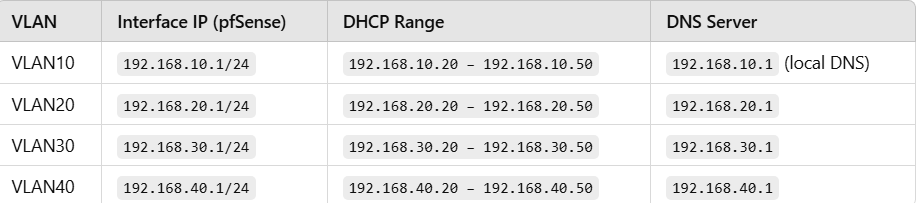


Figure Server Manager

**Windows Server 2022** machine, added to the domain SOChomelab.local. Set hostname to (Server1), operating system, and network configuration (DHCP assigned IP). The server is now functioning as a **Domain Controller**, forming the backbone of the lab's Active Directory infrastructure.

# Network Topology



# Tasks Completed

Installed and configured pfSense (LAN/WAN, DHCP, firewall rules)

Deployed Windows Server, installed Active Directory Domain Services (AD DS)

Created domain lab.local, added users and GPOs

Installed Kali Linux and Metasploit for offensive testing

Set up Splunk, forwarded logs from Windows and pfSense

Simulated attacks, analyzed logs and alerts in Splunk

In summary, the main thrust of this project was the building of a working penetration testing lab and the deployment of a Security Operations Center (SOC) homelab. The focus was the creation of a segmented network, the installation of pfSense to monitor and regulate traffic, and the integration of Splunk for centralized logging.

Looking ahead, future development will progress into more in-depth configuration and hardening of the Windows Server environment—hardening Active Directory structures, Group Policy administration, and emulating real-world enterprise-level environments to expand the defensive aspect of cybersecurity operations.