11/6/2024

Implementation Report

Secure Solution: Malware Detection

Group G
SEMESTER IV - CYBERSECURITY

Table of Contents

Introduction

Summary

This document outlines the implementation details for a cybersecurity project developed as part of a group effort in a cybersecurity course. The project focuses on creating a secure environment for a hospital setting, drawing inspiration from the 2017 NHS WannaCry ransomware attack. The goal is to design a secure environment and implement security measures to protect hospital networks from similar threats. We aim to ensure the safety of sensitive patient data.

The project leverages a combination of Virtual Machines (VMs) running Linux servers and Windows VPN workspaces. Key components of the secure environment include the deployment of antivirus software, configuration of firewalls with specific rules, and the integration of Host-based Intrusion Detection Systems (HIDS) and Network-based Intrusion Detection Systems (IDS). Through this comprehensive approach, the project aims to demonstrate effective strategies for malware detection and overall network security in a healthcare context.

This document details the setup and configuration processes for each security measure we took. Here, we will be providing instructions regarding the steps we took to create a secure environment. The implementation aims to offer a practical guide for protecting healthcare environments from cyber threats, ensuring both data security and operational continuity.

Objectives

The main objective of this project is to design and implement a complete cybersecurity framework, tailored to a hospital envrionment. It is inspired by the NHS WannaCry ransomware attack of 2017. The goals we aim to achieve are as follows:

- **Enhance Security**: Establish a multi-layered securite architecture to protect a hospital's network infrastructure from malware, ransomware, and other cyber threats.
- **Virtualized Envrionments:** Set up Virtual Machines to simulate the hospital network, using Linux servers and Windows workspaces that utilize a VPN for remote employees. This will be the base of our implemented solution.
- **Antivirus Solutions:** Implement antivirus software on all stations to ensure malware is detected.
- **Firewalls:** Implement firewalls, then create rules to control network traffic in a manner that prevents unauthorized access.
- **Intrusion Detection Systems:** Implement both Host-based Intrusion Detection Systems (HIDS) and Network-based Intrusion Detection Systems (IDS) to monitor network activity and identify security breaches.
- **Simulated Attack:** Simulate malware being released into the secure envrionment. This is to test that our solution does what it set out to do: detect and mitigate against malware.
- **Documentation:** Create documentation throughout the process.

Scope

The focus of this project is to design and implement a secure solution to Malware, specifically in a healthcare envrionment. It involves a comprehensive framework of cybersecurity features. The scope is as follows:

Within Scope

- **Virtual Envrionment Setup:** Virtual Machines are configured, using Linux servers and Windows workspaces for employees to simulate a hospital environment.
- **Antivirus:** Installation and configuration of antivirus software on all VMs to detect and mitigate malware threats.
- **Firewall Configuration:** Firewall implemented with added rules to prevent unauthorized access.
- **Intrusion Detection Systems:** HIDS and IDS will be deployed to fully monitor network activity.
- **Malware Attack Simulation:** This is essentially for testing our secure solution against malware.

Out of Scope

- **Physical Security Measures:** We are only able to focus on cyber-threats. Details such as securing the actual building and equipment is out of our scope.
- **Employee Training:** While we strongly suggest training your employees to detect phishing attacks and other cyber threats, this is out of our scope.
- **Compliance Certification:** We are not able to get any formal compliance certifications, such as HIPAA certification.

System Architecture

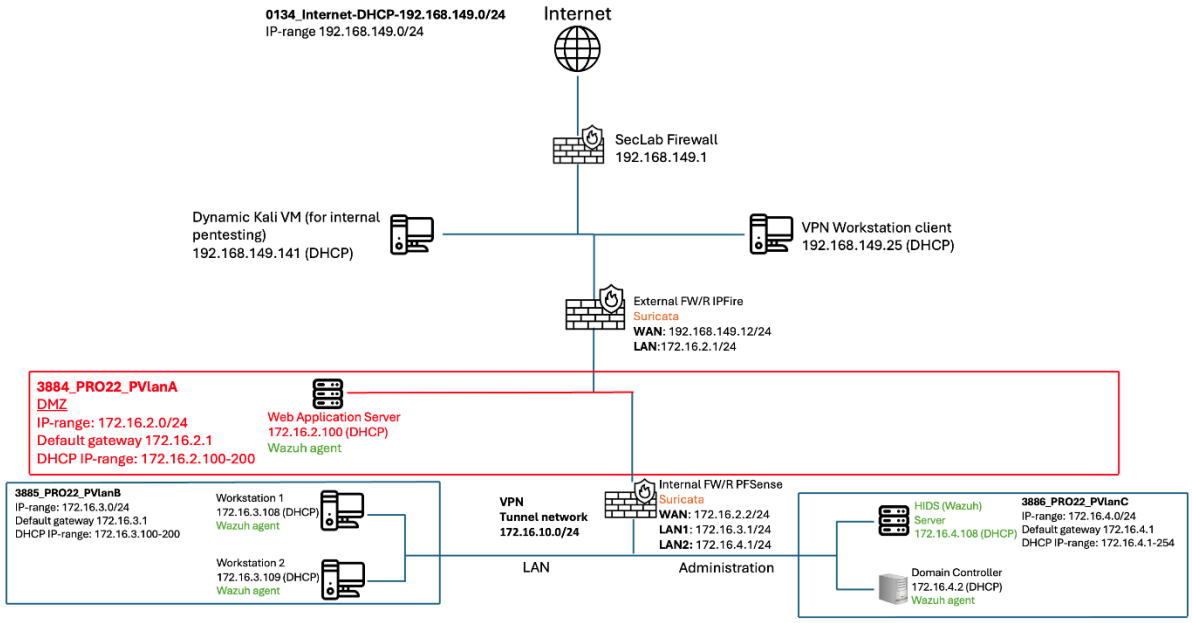
Overview

The system architecture for our cybersecurity project is designed to manufacture a secure environment for a simulated hospital network. The architecture uses a combination of Virtual Machines (VMs) and security tools to replicate a real-world healthcare setting, providing a platform for implementing and testing our chosen cybersecurity measures. This design includes multiple layers of defense and specialized components to ensure the safety of the sensitive data NHS is in possession of.

The system is segmented into several key components, each playing a critical role in the overall security posture. The components are distributed across different VLANs to simulate a realistic network segmentation strategy, improving both security and manageability.

Diagrams

Network Diagram



This is the network diagram we have constructed to simulate our full envrionment.

Internet-DHCP: 192.168.149.0/24
IP-range: 192.168.149.0/24

SecLab Firewall: 192.168.149.1

Kali VM: 192.168.149.141 (DHCP)

External Firewall FW/R IPFire:
WAN: 192.168.149.12/24
LAN: 172.16.2.1/24

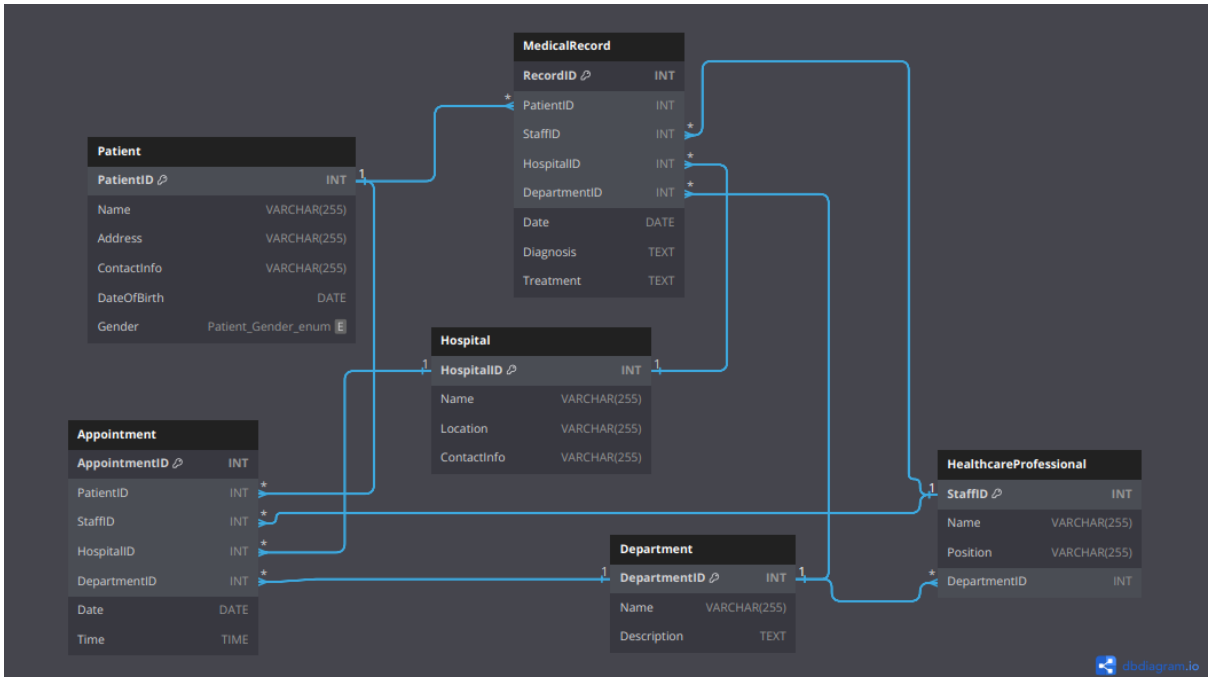
PvlanA:
DMZ
IP-Range: 172.16.2.0/24
Default gateway: 172.16.2.1
DHCP IP_range: 172.16.2.100-200

Web Application Server
172.16.2.100 (DHCP)
WAF (ModSecurity)
Wazuh Agent

HIDS (Wazuh) Server:
172.16.2.101 (DHCP)

Linux Workstation:
Wazuh Agent
172.16.2.102 (DHCP)

Database Diagram

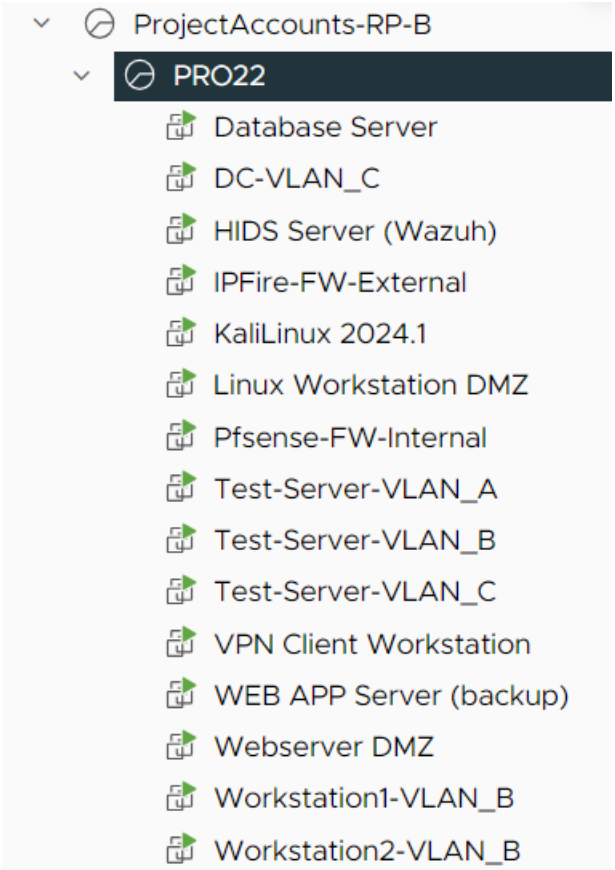


Our proposed database diagram setup for our simulated hospital web application.

Components

Virtual Machines Summary

Here, I will give a breakdown of our entire VM setup and explain the function of each one.



Database Server

- **Purpose**: Hosts the database for the hospital's web application.
- **Technology**: Uses Kubernetes for container orchestration. This exists here to host the web applications database.

DC-VLAN_C

- **Role**: Domain Controller.
- **Technology**: Utilizes Microsoft Windows DNS Server Service for domain name resolution and network management, providing centralized authentication and directory services.

-Group Policies would go here

HIDS Server (Wazuh)

- **Role**: Host-based Intrusion Detection System (HIDS).
- **Technology**: Wazuh, which monitors system integrity, logs, and behaviors to detect and respond to suspicious activities on the hosts.

IPFire-FW-External

- **Role**: External Firewall.
- **Technology**: IPFire, configured to filter incoming and outgoing traffic, protecting the network from external threats. It forms the first line of defense in our two-firewall setup.

Kali Linux 2024.1

- **Role**: Security Workstation.
- **Technology**: A Kali Linux workstation located on the DMZ. This is the machine we can use and move across the whole environment. We are using this for testing the environment.

Pfsense-FW-Internal

- **Role**: Internal Firewall.
- **Technology**: Pfsense, configured to monitor and control internal traffic, ensuring security within the network and providing a second layer of defense.

Test Servers

- **Test-Server-VLAN_A**: Located on private VLAN_A for testing security measures and configurations. May be deleted in the future when they are no longer needed.
- **Test-Server-VLAN_B**: Located on private VLAN_B, another test server. May be deleted in the future when they are no longer needed.
- **Test-Server-VLAN_C**: Located on private VLAN_C, our last test server. May be deleted in the future when they are no longer needed.

VPN Client Workstation

- **Role**: Simulates remote employee access.
- **Technology**: Configured with DHCP and connected via VPN to securely integrate with the hospital's LAN, enabling remote access as if the device were within the internal network.

Webserver DMZ

- **Role**: Main Web Server.
- **Technology**: Hosts the primary web application and is located in the DMZ to segregate it from the internal network, enhancing security by isolating external-facing services.

Workstations (VLAN_B)

- **Workstation1-VLAN_B**: Simulated workstation representing an end-user device within the hospital's internal network.
- **Workstation2-VLAN_B**: Another simulated workstation located on the hospital LAN, used to mimic typical hospital workstations and their interaction with the network.

Firewalls

IPFire

- **Purpose**: Serves as the external firewall in our two-firewall setup.
- **Technology**: IPFire is an open-source firewall solution known for its flexibility and robustness. It filters incoming and outgoing network traffic, protecting the network from unauthorized access and external threats. IPFire supports various security features such as intrusion detection, VPN support, and web proxy, making it an integral part of our network's perimeter defense.

Pfsense

- **Purpose**: Acts as the internal firewall.
- **Technology**: Pfsense is an open-source firewall and router software based on FreeBSD. It provides comprehensive network protection by monitoring and controlling internal traffic, ensuring that only legitimate traffic flows within the network segments. Pfsense includes features like VPN, load balancing, and advanced packet filtering, which help secure the internal network against internal threats and lateral movement of malware.

Antivirus Software

ClamAV

- **Purpose**: Provides antivirus protection for Linux-based systems.
- **Technology**: ClamAV (Clam AntiVirus) is an open-source antivirus engine designed for detecting trojans, viruses, malware, and other malicious threats. It is particularly suited for server environments due to its command-line interface and flexibility in integration with various applications. ClamAV scans files on-demand and can be scheduled to run at regular intervals to ensure continuous protection.

Windows Defender

- **Purpose**: Provides antivirus protection for Windows-based systems.
- **Technology**: Windows Defender is a built-in antivirus and anti-malware solution provided by Microsoft for Windows operating systems. It offers real-time protection, scanning, and removal of various types of malware, including viruses, spyware, and ransomware. Windows Defender integrates seamlessly with the Windows operating system, providing automatic updates and comprehensive security features to protect against evolving threats.

Host-based Intrusion Detection System (HIDS)

Wazuh

- **Purpose**: Provides host-based intrusion detection and security monitoring.
- **Technology**: Wazuh is an open-source security monitoring platform. It provides HIDS functionality, such as log analysis, file integrity monitoring, rootkit detection, and real-time alerts. Wazuh collects and analyzes data from many sources, analysing events to detect suspicious activities and security breaches. It also integrates with various security information and event management (SIEM) systems to provide a unified view of security across the network.

- **Wazuh dashboard should be accessible anywhere on the LAN**.

Network-based Intrusion Detection System (IDS)

Suricata

- **Purpose**: Provides network-based intrusion detection and prevention.
- **Technology**: Suricata is an open-source network IDS, IPS, and network security monitoring engine. It inspects network traffic based on established rules to detect threats. Suricata works in real-time to identify and respond to suspicious activities on the network. It supports deep packet inspection, protocol analysis, and has capabilities to log HTTP, DNS, TLS, and many other protocols, making it a powerful tool for detecting a wide range of network-based attacks.

Implementation Details

Setup and Configuration

This section of the document aims to provide detailed steps for setting up and configuring each component in the environment. The implementation of all the components is based on the following user stories:

Healthcare Provider Staff:

1. As a healthcare provider staff member, I want the system to encrypt patient data in our databases so that confidentiality is maintained, and we comply with privacy regulations.
2. As a healthcare provider staff member, I want the network traffic to be monitored for any suspicious activity or potential security threats so that the IT team can be notified immediately for investigation.

IT Team:

1. As an IT team member, I want the system to update all software and security patches regularly so that vulnerabilities are mitigated, and the risk of exploitation by cyber attackers is minimized, ensuring system security.
2. As an IT team member, I want the system to securely transfer patient data between different healthcare facilities or external partners using encrypted communication protocols so that data breaches during transit are prevented, ensuring patient data privacy.
3. As an IT team member, I want to view a network diagram of the cybersecurity solution so that I can understand how the system is structured and how data is protected.
4. As an IT team member, I want the system to utilize two different firewalls to enhance security measures and safeguard our network.
5. As an IT team member, I want the system to employ a robust system hardening approach to strengthen the security posture of our infrastructure, reducing the risk of vulnerabilities and unauthorized access.

6. As an IT team member, I want the system to implement segmentation to create isolated network segments, allowing for granular control over network traffic and minimizing the impact of potential breaches.

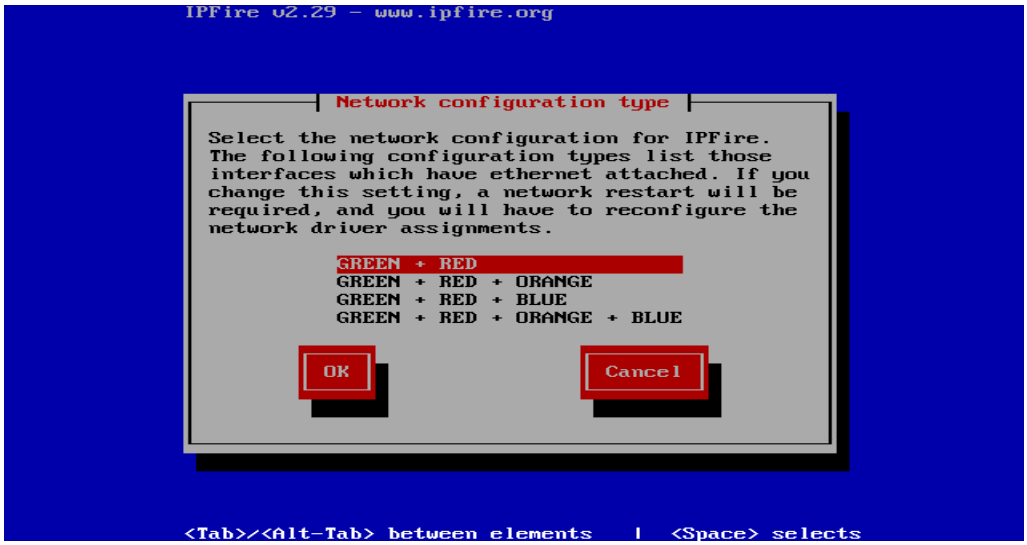
System Administrators:

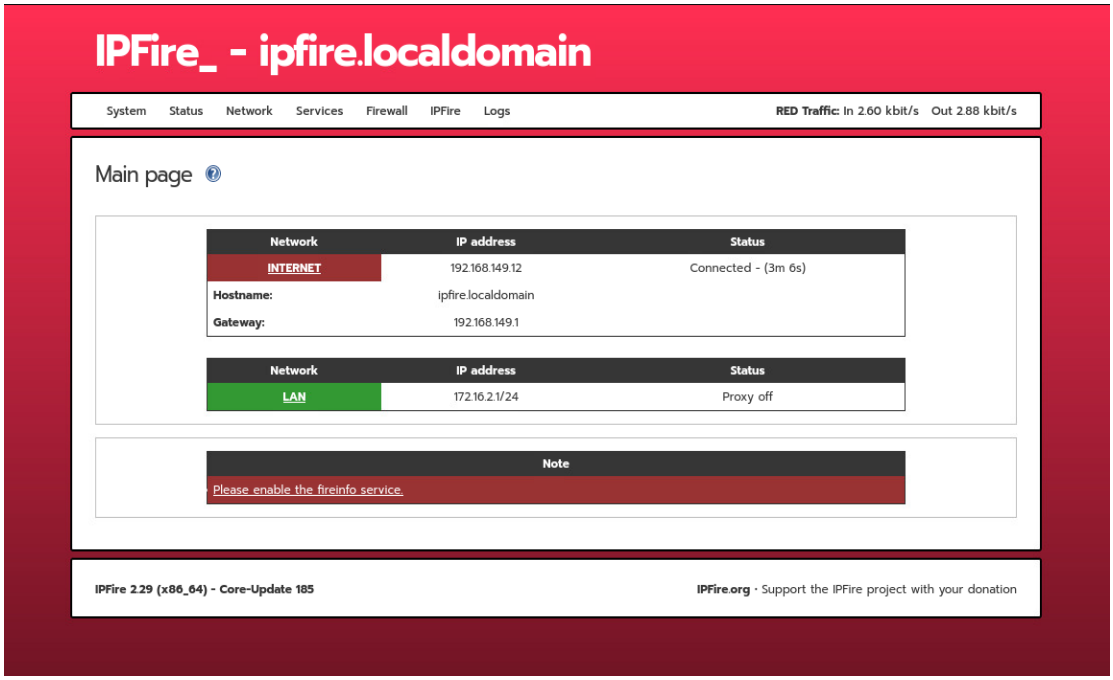
1. As a system administrator, I want the system to conduct regular malware scans on our current software to detect and remove any malicious programs or files, ensuring the integrity and safety of our data.
2. As a system administrator, I want the system to include domain controllers to manage user authentication and authorization efficiently, ensuring secure access to network resources and applications.
3. As a system administrator, I want the system to deliver good performance to ensure that our critical applications and services run smoothly and efficiently, minimizing downtime and disruptions to our workflow.
4. As a system administrator, I want sensitive data to be accessible only on the hospital VLAN to maintain confidentiality and compliance with privacy regulations, preventing unauthorized access from external networks.
5. As a system administrator, I want the system to provide endpoint protection to secure devices connected to our network, defending against malware, ransomware, and other cyber threats that may target endpoints.

IPFire

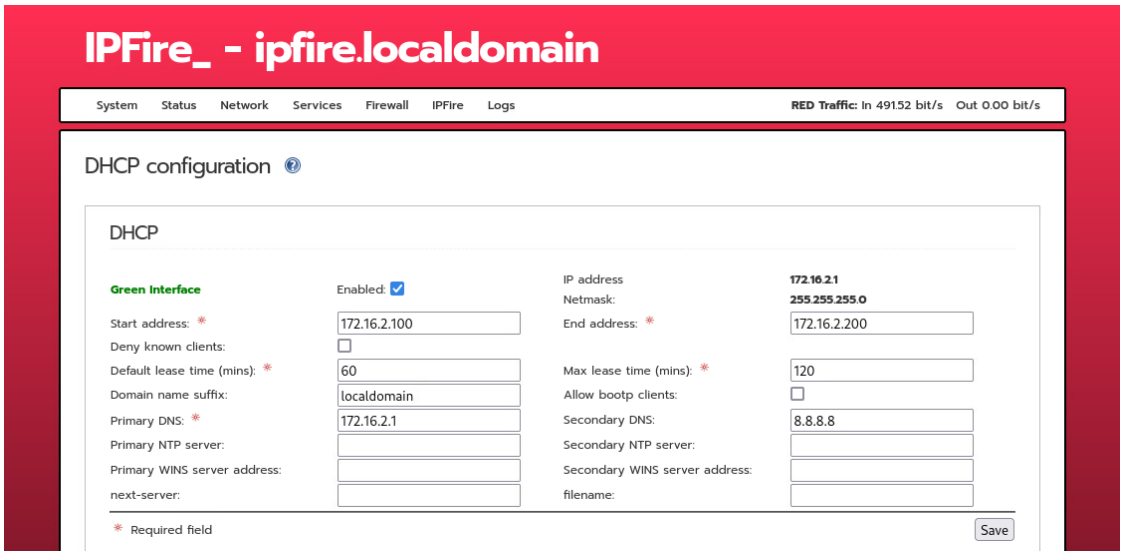
Note: Since IPFire does not exsist as a template in the vsphere environment. We had to download it and create it locally and then upload it as OVA format to the vsphere environment.

1) Visit IPFire download page and download the appropriate ISO image file.
2) Create a VM locally using the downloaded ISO file.
3) Convert the created machine to OVA format using a tool called OVFTool.
4) Upload the created OVA file to the vsphere environment and deploy it.
5) Configure the machine:
    - Setup network interfaces:
        - Green: Internal network interface (Default GW).
        - Red: External network interface.

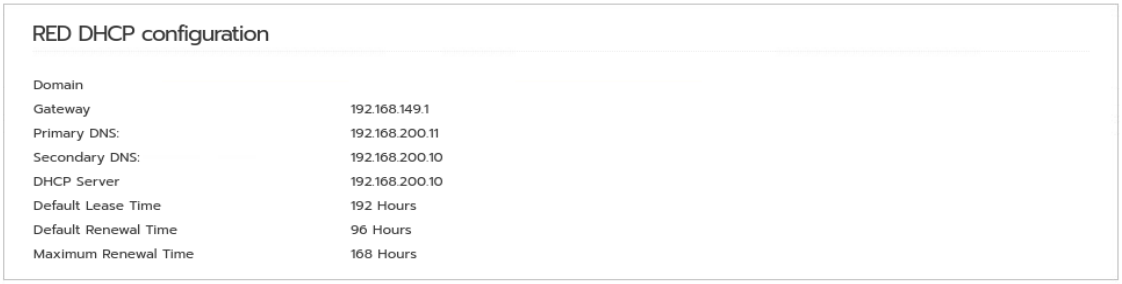




- Configure DHCP for the internal interface:

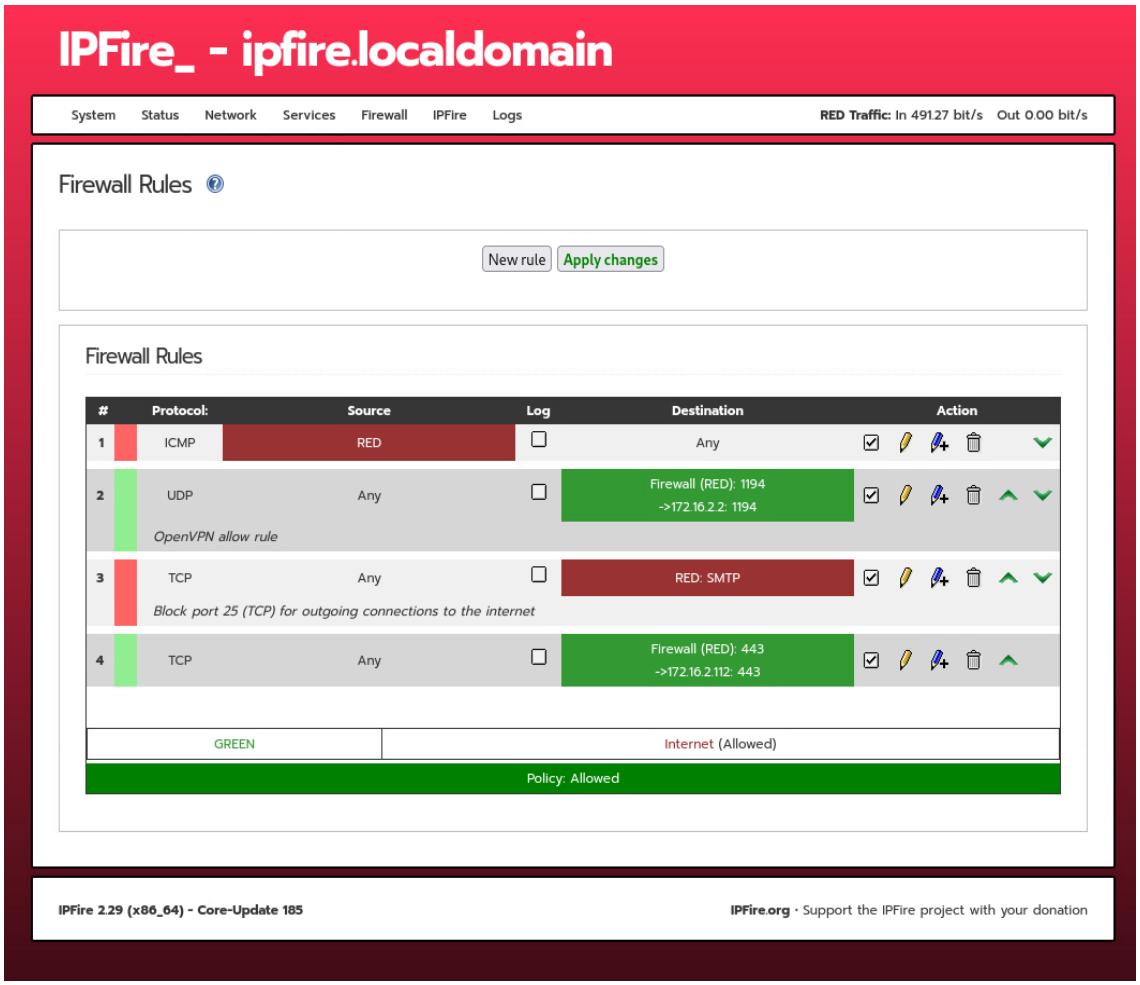


- Configure the external interface to use the Netlab's DHCP:
  Note: This interface should not have DHCP enabled to avoid any conflicts.

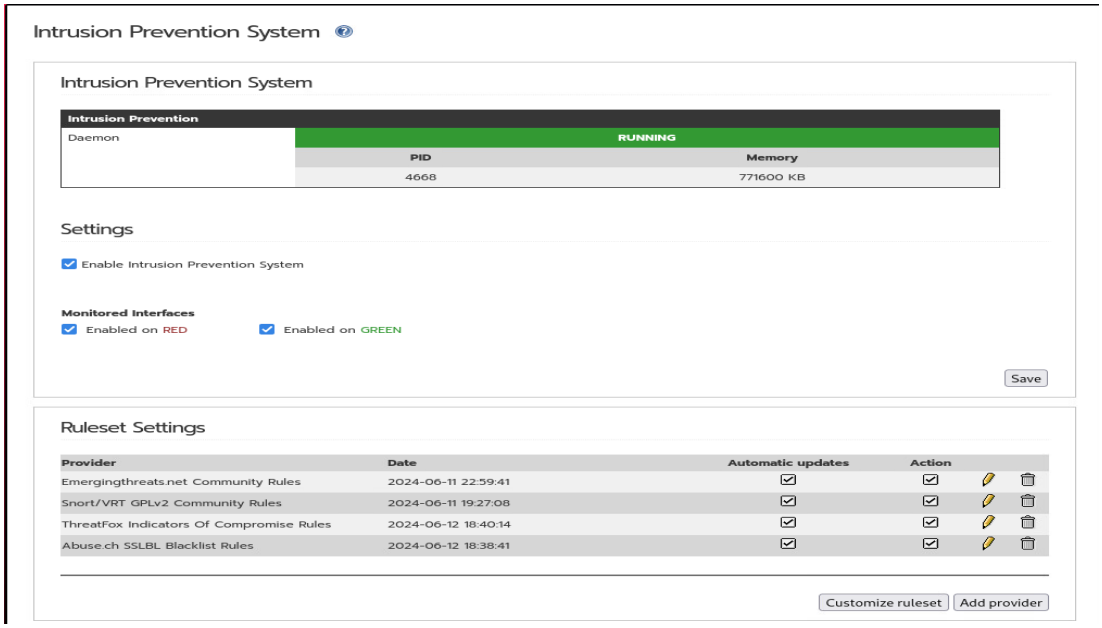


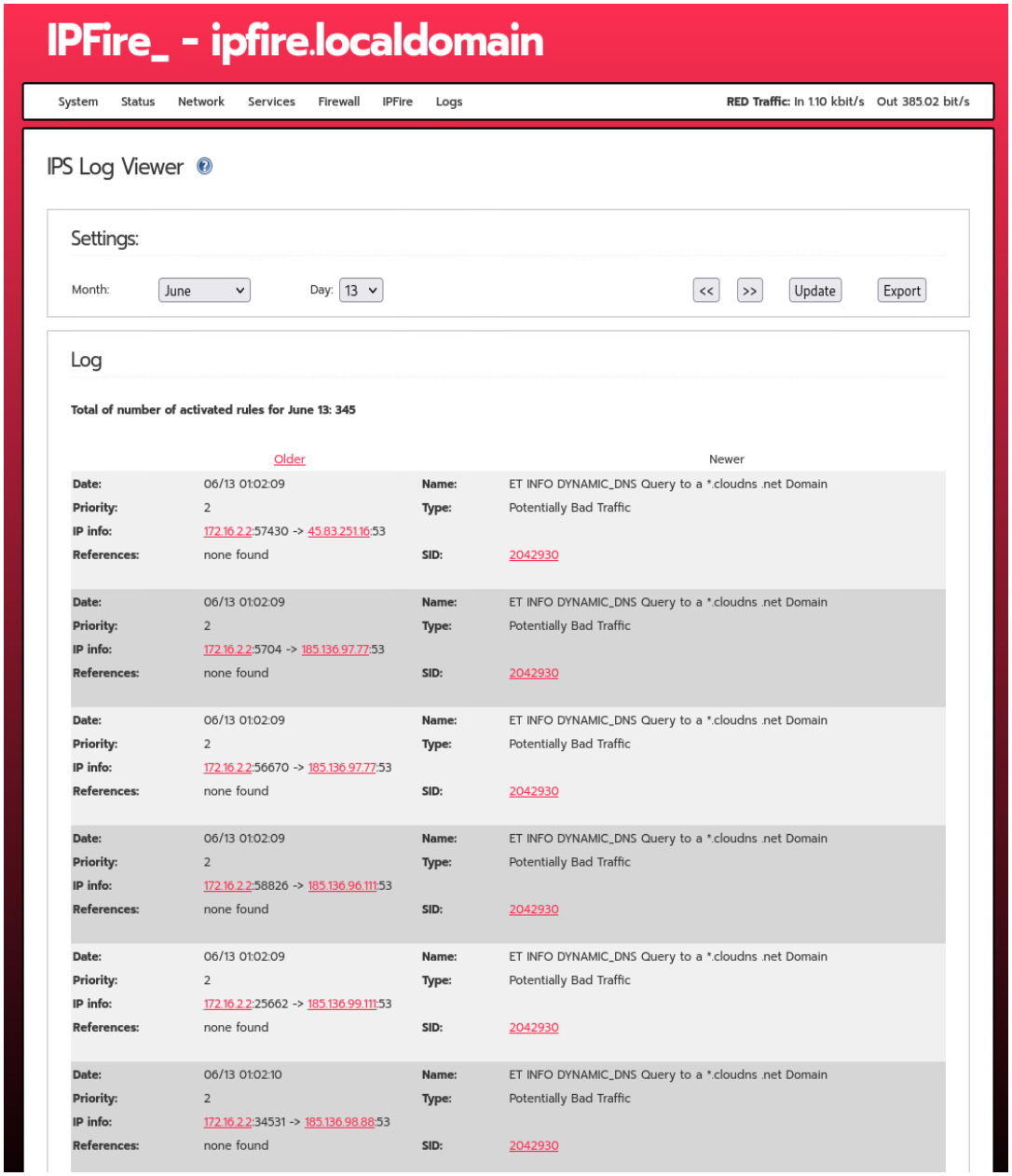
- Add Firewall Rules to allow traffic on ports 443 (Webserver) and 1194 (OpenVPN):

Note: the OpenVPN server is running on the internal firewall (PFsense) and intended for remote employees.



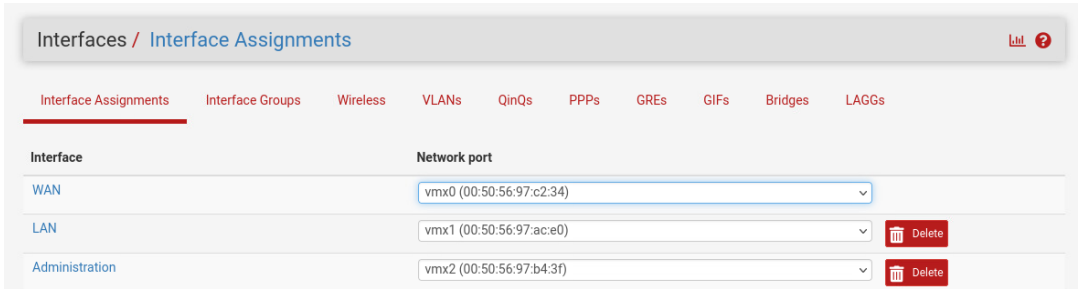
- Setup Intrusion Prevention System with various rulesets to monitor both of the interfaces.





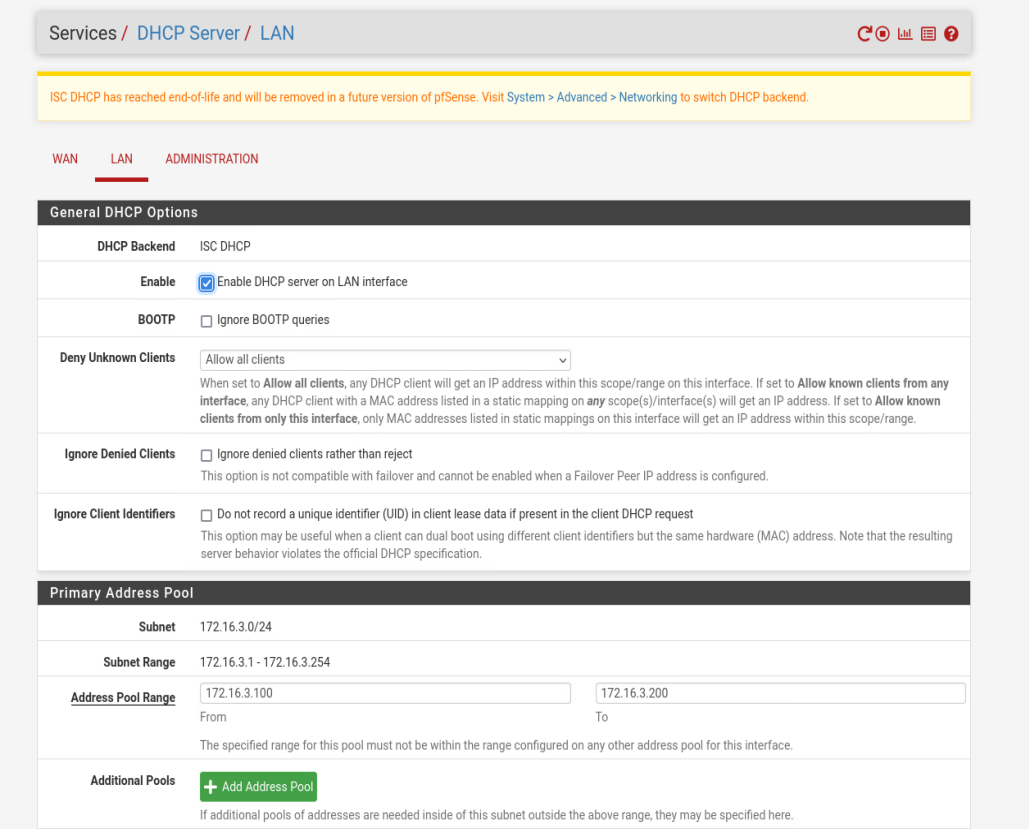
Pfsesne

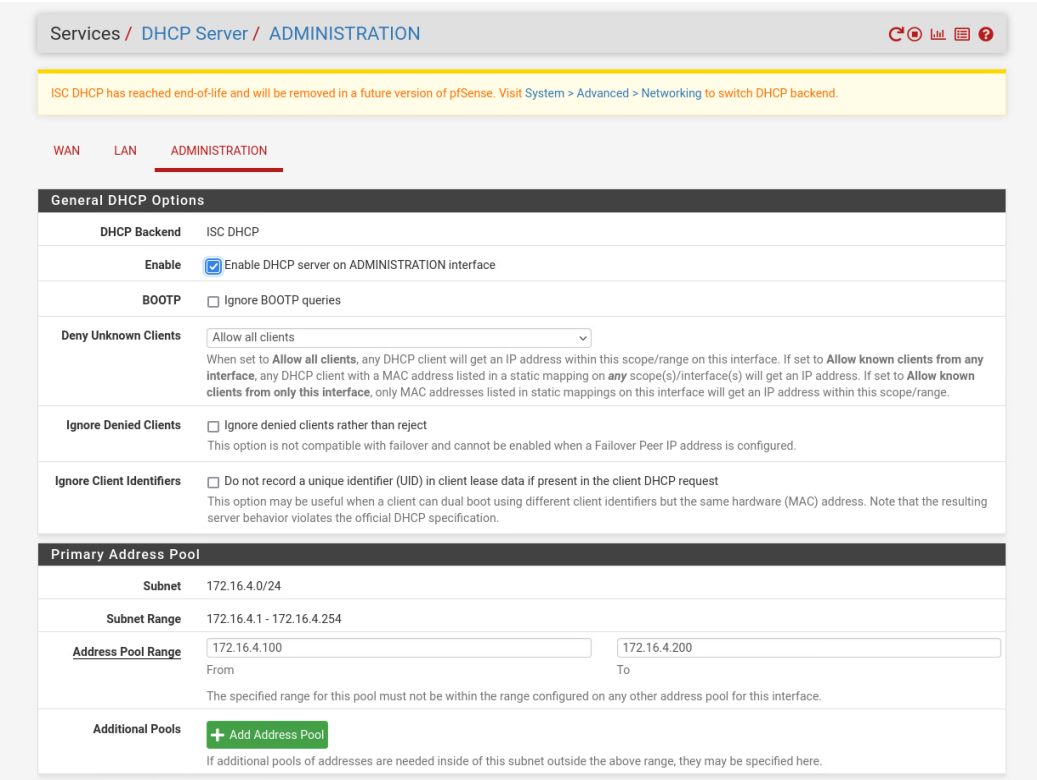
1) Deploy Pfsense from a template in the vsphere environment.
2) Configure the machine:
    - Setup network interfaces:
        - WAN: DMZ facing interface.
        - LAN: Employees LAN (where workstations reside).
        - ADMINISTRATION: Where monitoring systems and DC reside.



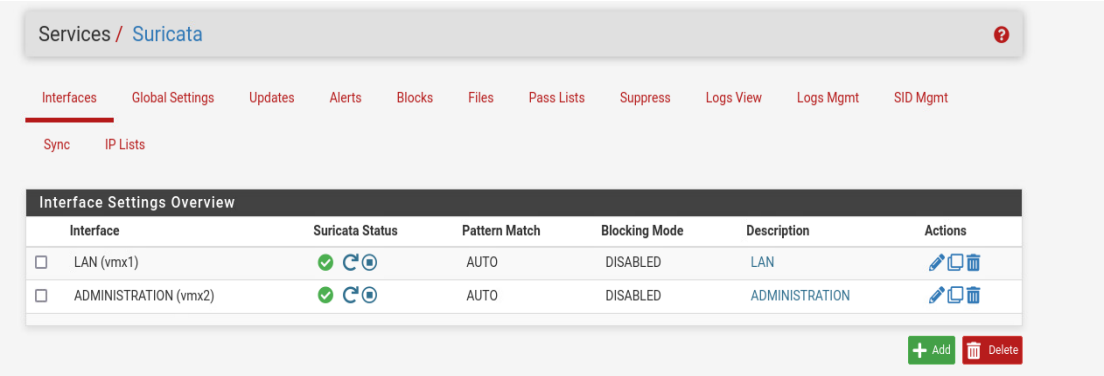
- Enable DHCP for the LAN and ADDMINISTRATION interfaces:

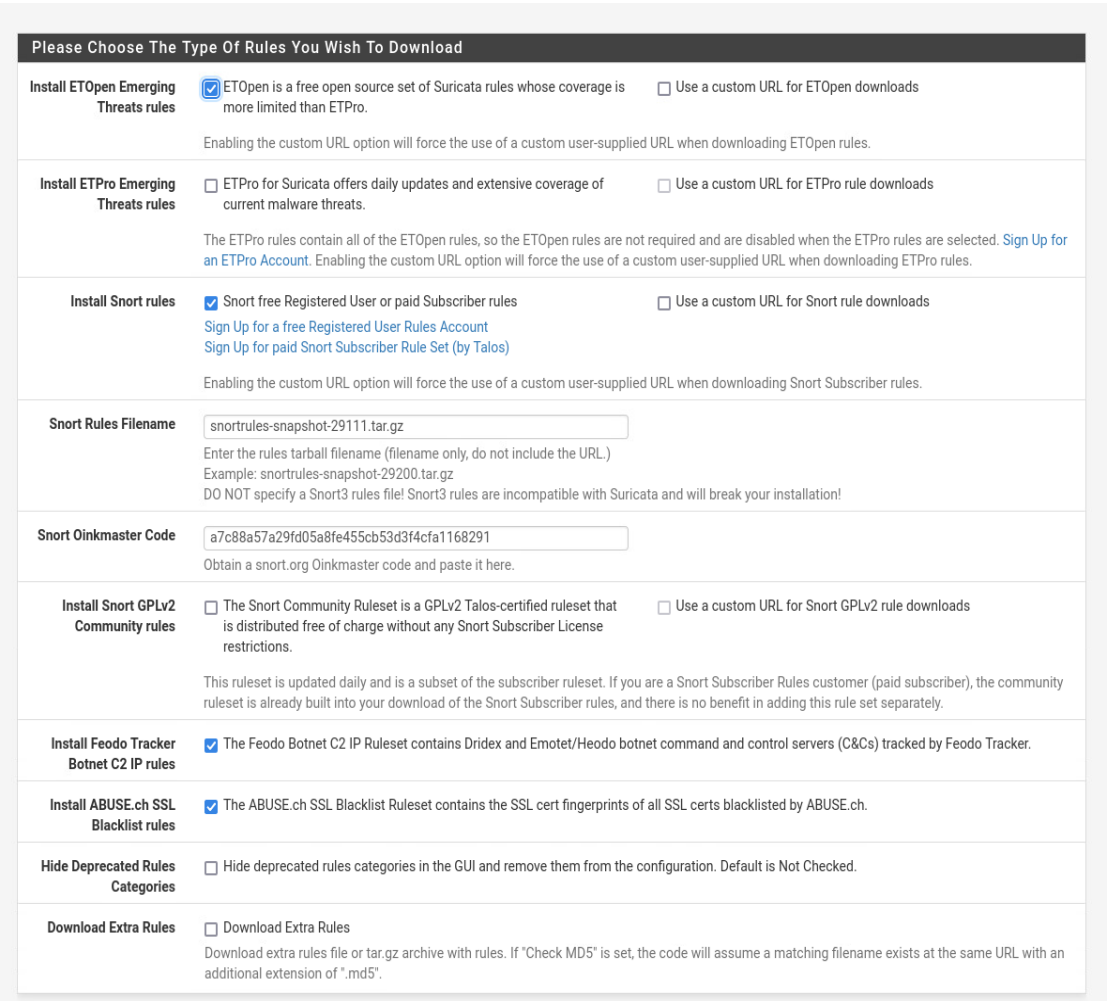
  Note: The WAN should not have DHCP enabled to avoid any conflicts .



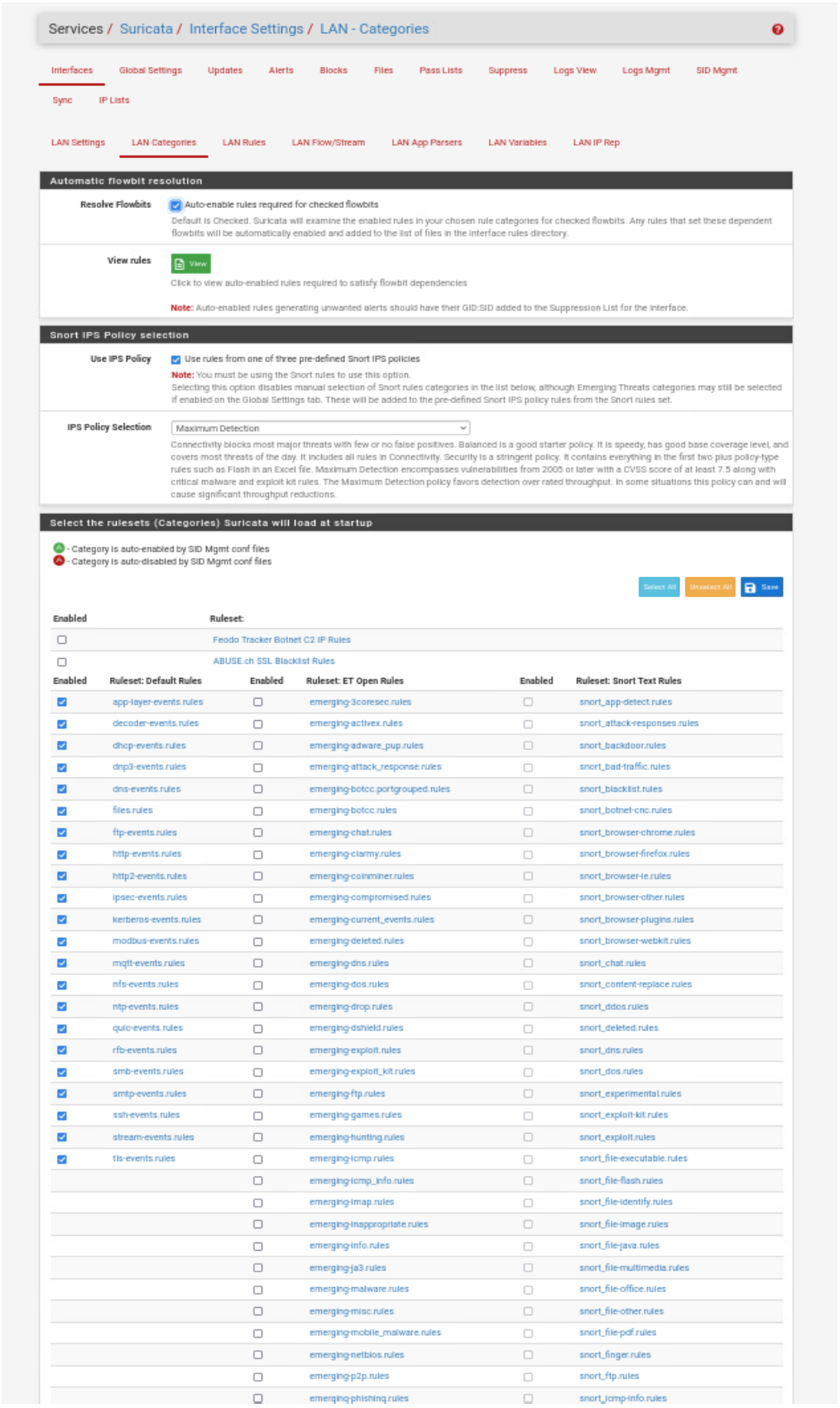


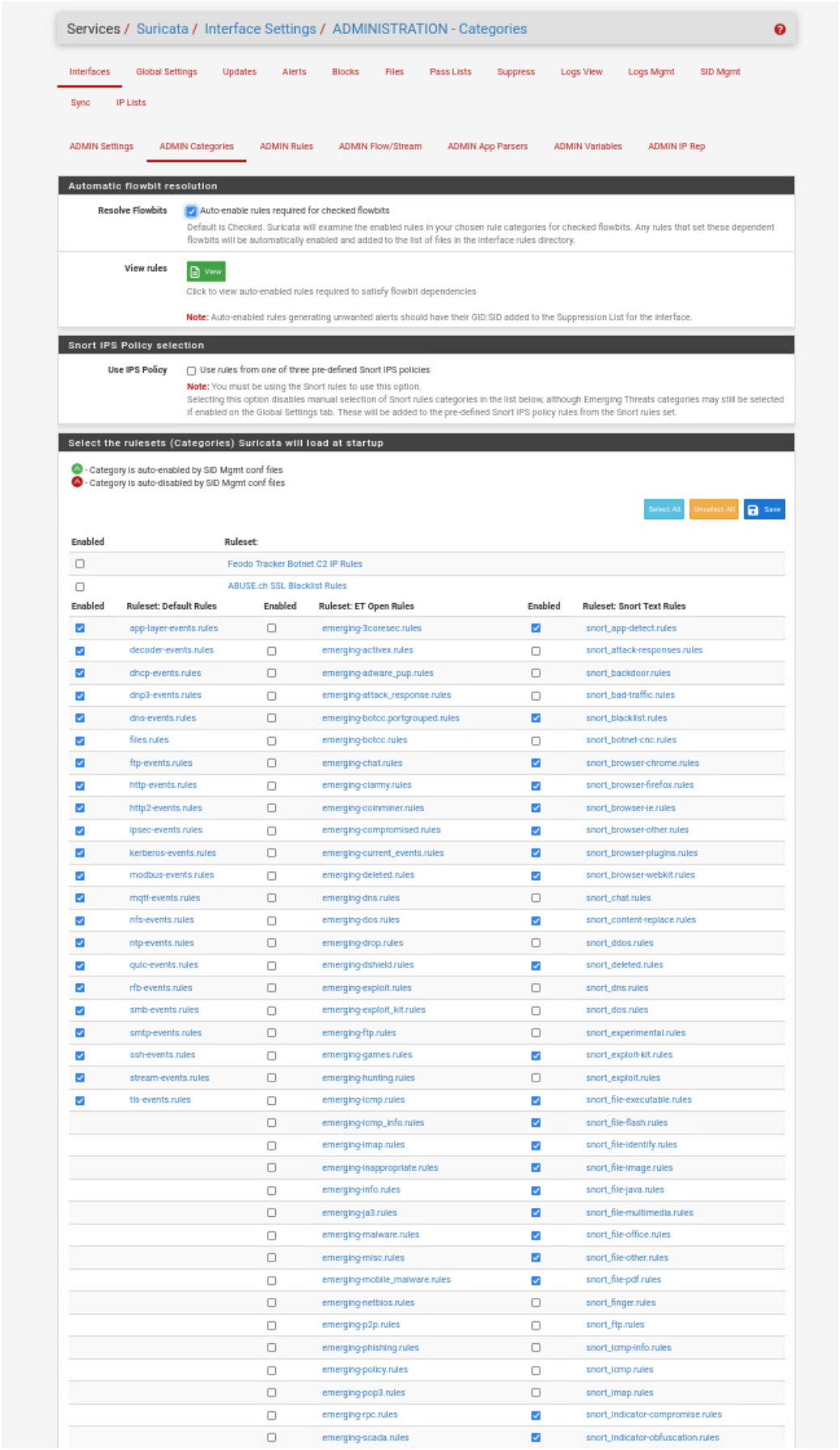
- Setup Suricate (IDS/IPS) with varios rule sets to monitor both the LAN and ADMINISTRATION interfaces:





- Select the desired rules for both of the interfaces:




Services / Suricata / Interface Settings / ADMINISTRATION - Categories
Interfaces
Global Settings
Updates
Alerts
Blocks
Files
Pass Lists
Suppress
Logs View
Logs Mgmt
SID Mgmt
Sync
IP Lists
ADMIN Settings
ADMIN Categories
ADMIN Rules
ADMIN Flow/Stream
ADMIN App Parsers
ADMIN Variables
ADMIN IP Rep
Automatic flowbit resolution
Resolve Flowbits
Auto-enable rules required for checked flowbits
Default is Checked. Suricata will examine the enabled rules in your chosen rule categories for checked flowbits. Any rules that set these dependent flowbits will be automatically enabled and added to the list of files in the interface rules directory.
View rules
View
Click to view auto-enabled rules required to satisfy flowbit dependencies
Note: Auto-enabled rules generating unwanted alerts should have their GID:SID added to the Suppression List for the interface.
Snort IPS Policy selection
Use IPS Policy
Use rules from one of three pre-defined Snort IPS policies
Note: You must be using the Snort rules to use this option.
Selecting this option disables manual selection of Snort rules categories in the list below, although Emerging Threats categories may still be selected if enabled on the Global Settings tab. These will be added to the pre-defined Snort IPS policy rules from the Snort rules set.
Select the rulesets (Categories) Suricata will load at startup
- Category is auto-enabled by SID Mgmt conf files
- Category is auto-disabled by SID Mgmt conf files
Select All
Unselect All
Save
Enabled
Ruleset:
Feodo Tracker Botnet C2 IP Rules
ABUSE.ch SSL Blacklist Rules
Enabled
Ruleset: Default Rules
Enabled
Ruleset: ET Open Rules
Enabled
Ruleset: Snort Text Rules
app-layer-events.rules
decoder-events.rules
dhcp-events.rules
dnp3-events.rules
dns-events.rules
files.rules
ftp-events.rules
http-events.rules
http2-events.rules
ipsec-events.rules
kerberos-events.rules
modbus-events.rules
mqtt-events.rules
nfs-events.rules
ntp-events.rules
quic-events.rules
rfb-events.rules
smb-events.rules
smtp-events.rules
ssh-events.rules
stream-events.rules
tls-events.rules
emerging-3coresec.rules
emerging-activex.rules
emerging-adware_pup.rules
emerging-attack_response.rules
emerging-botcc.portgrouped.rules
emerging-botcc.rules
emerging-chat.rules
emerging-clamy.rules
emerging-coinminer.rules
emerging-compromised.rules
emerging-current_events.rules
emerging-deleted.rules
emerging-dns.rules
emerging-dos.rules
emerging-drop.rules
emerging-dshield.rules
emerging-exploit.rules
emerging-exploit_kit.rules
emerging-ftp.rules
emerging-games.rules
emerging-hunting.rules
emerging-icmp.rules
emerging-icmp_info.rules
emerging-imap.rules
emerging-inappropriate.rules
emerging-info.rules
emerging-ja3.rules
emerging-malware.rules
emerging-misc.rules
emerging-mobile_malware.rules
emerging-netbios.rules
emerging-p2p.rules
emerging-phishing.rules
emerging-policy.rules
emerging-pop3.rules
emerging-rpc.rules
emerging-scada.rules
snort_app-detect.rules
snort_attack-responses.rules
snort_backdoor.rules
snort_bad-traffic.rules
snort_blacklist.rules
snort_botnet-cnc.rules
snort_browser-chrome.rules
snort_browser-firefox.rules
snort_browser-ie.rules
snort_browser-other.rules
snort_browser-plugins.rules
snort_browser-webkit.rules
snort_chat.rules
snort_content-replace.rules
snort_ddos.rules
snort_deleted.rules
snort_dns.rules
snort_dos.rules
snort_experimental.rules
snort_exploit-kit.rules
snort_exploit.rules
snort_file-executable.rules
snort_file-flash.rules
snort_file-identify.rules
snort_file-image.rules
snort_file-java.rules
snort_file-multimedia.rules
snort_file-office.rules
snort_file-other.rules
snort_file-pdf.rules
snort_finger.rules
snort_ftp.rules
snort_icmp-info.rules
snort_icmp.rules
snort_imap.rules
snort_indicator-compromise.rules
snort_indicator-obfuscation.rules

- Firewall rules are configured for each interface based on the services running in the environment, with the purpose of each rule noted in the description section:

Firewall / Rules / WAN

Floating | WAN | LAN | ADMINISTRATION | OpenVPN

Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
☐	✓ 2/56.23 MiB	IPv4 TCP	*	*	172.16.4.108	1514	*	none		NAT Wazuh Rule	
☐	✓ 0/28 KiB	IPv4 TCP	*	*	172.16.4.108	1515	*	none		NAT Wazuh Rule	
☐	✓ 0/0 B	IPv4 TCP	*	*	172.16.4.108	55000	*	none		NAT Wazuh Rule	
☐	✓ 0/83.79 MiB	IPv4 TCP	*	*	172.16.4.108	443 (HTTPS)	*	none		NAT Wazuh Dashboard (Temporary Rule)	
☐	✓ 1/115.99 MiB	IPv4 UDP	*	*	WAN address	1194 (OpenVPN)	*	none		OpenVPN wizard	

Add | Add | Delete | Toggle | Copy | Save | Separator

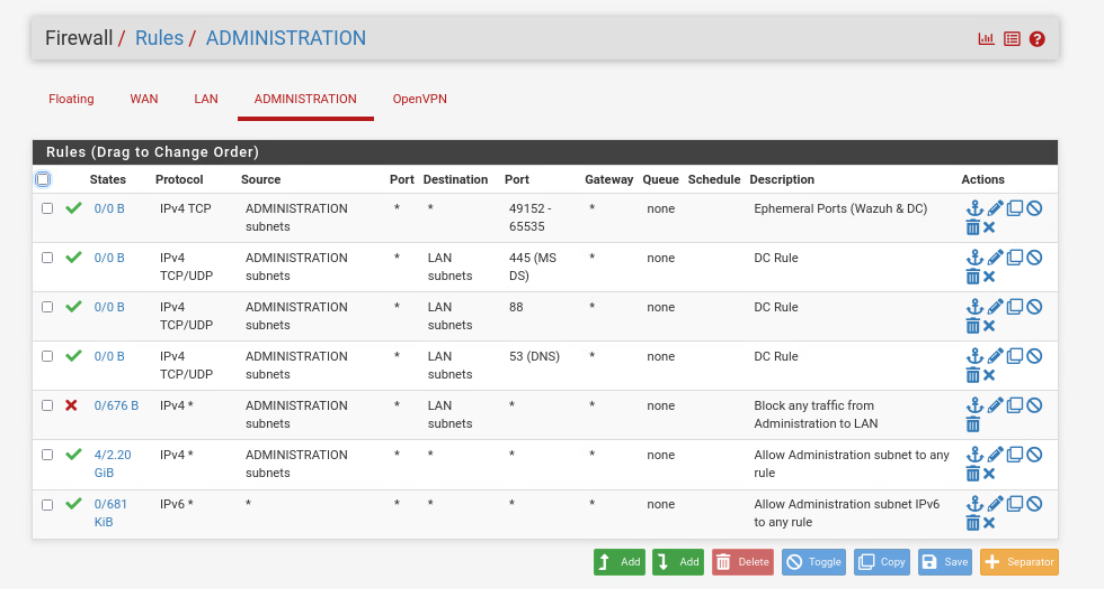
Firewall / Rules / LAN

Floating | WAN | LAN | ADMINISTRATION | OpenVPN

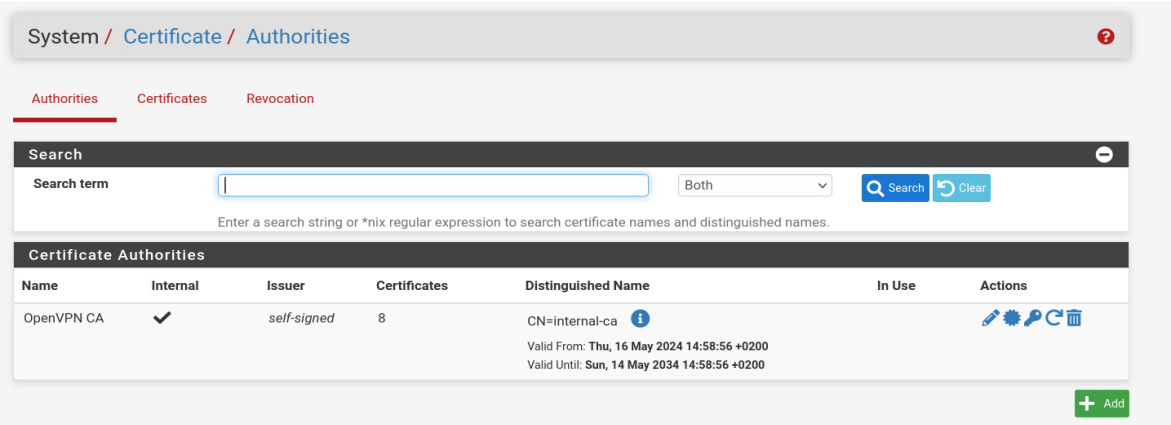
Rules (Drag to Change Order)

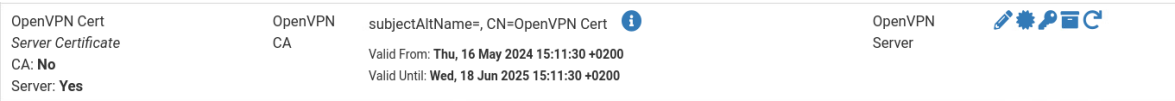
	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
	✓ 0/0 B	*	*	*	LAN Address	443 80 22	*	*		Anti-Lockout Rule	
☐	✓ 0/54.92 MiB	IPv4 TCP	*	*	172.16.4.108	443 (HTTPS)	*	none		Wazuh Dashboard (Temporary Rule)	
☐	✓ 2/55.10 MiB	IPv4 TCP	*	*	172.16.4.108	1514	*	none		Wazuh Rule	
☐	✓ 0/6 KiB	IPv4 TCP	*	*	172.16.4.108	1515	*	none		Wazuh Rule	
☐	✓ 0/0 B	IPv4 TCP	*	*	172.16.4.108	55000	*	none		Wazuh Rule	
☐	✓ 0/1.01 MiB	IPv4 TCP/UDP	LAN subnets	*	ADMINISTRATION subnets	445 (MS DS)	*	none		DC Rule	
☐	✓ 0/633 KiB	IPv4 TCP/UDP	LAN subnets	*	ADMINISTRATION subnets	88	*	none		DC Rule	
☐	✓ 0/817 KiB	IPv4 TCP/UDP	LAN subnets	*	ADMINISTRATION subnets	53 (DNS)	*	none		DC Rule	
☐	✓ 0/0 B	IPv4 TCP	LAN subnets	*	ADMINISTRATION subnets	3269	*	none		DC Rule	
☐	✓ 0/0 B	IPv4 TCP	LAN subnets	*	ADMINISTRATION subnets	3268	*	none		DC Rule	
☐	✓ 0/0 B	IPv4 TCP	LAN subnets	*	ADMINISTRATION subnets	636 (LDAP/S)	*	none		DC Rule	
☐	✓ 0/890 KiB	IPv4 TCP/UDP	LAN subnets	*	ADMINISTRATION subnets	389 (LDAP)	*	none		DC Rule	
☐	✓ 0/26.32 MiB	IPv4 TCP	LAN subnets	*	ADMINISTRATION subnets	49152 - 65535	*	none		DC Rule	
☐	✓ 0/0 B	IPv4 TCP/UDP	LAN subnets	*	ADMINISTRATION subnets	464	*	none		DC Rule	
☐	✓ 0/121 KiB	IPv4 TCP	LAN subnets	*	ADMINISTRATION subnets	135	*	none		DC Rule	
☐	✓ 0/17 KiB	IPv4 UDP	LAN subnets	*	ADMINISTRATION subnets	123 (NTP)	*	none		DC Rule	
☐	✗ 0/13 KiB	IPv4 *	LAN subnets	*	ADMINISTRATION subnets	*	*	none		Block any traffic from LAN to Administration	
☐	✓ 6/5.09 GiB	IPv4 *	LAN subnets	*	*	*	*	none		Default allow LAN to any rule	
☐	✓ 0/0 B	IPv6 *	LAN subnets	*	*	*	*	none		Default allow LAN IPv6 to any rule	

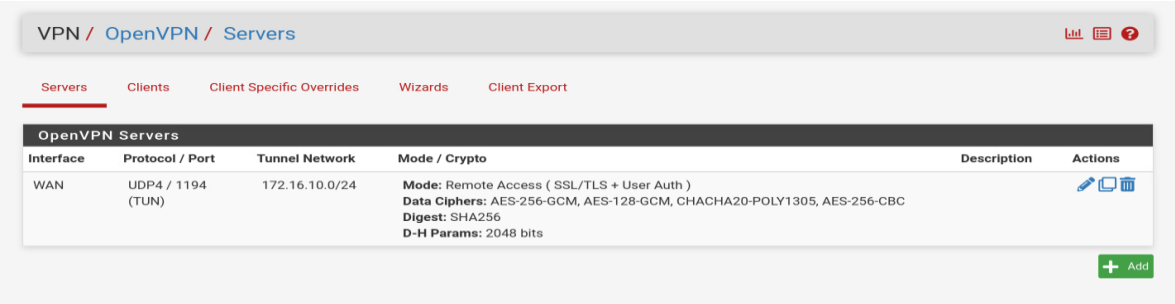
Add | Add | Delete | Toggle | Copy | Save | Separator

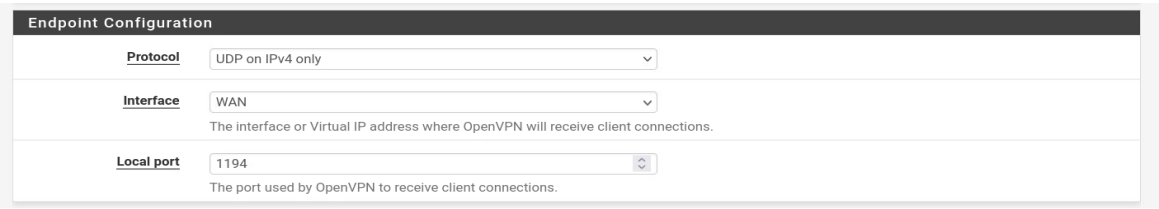


- In setting up OpenVPN on pfSense for secure remote access, the process involves creating a Certificate Authority (CA), configuring the OpenVPN server, setting up user authentication, and applying necessary firewall rules.

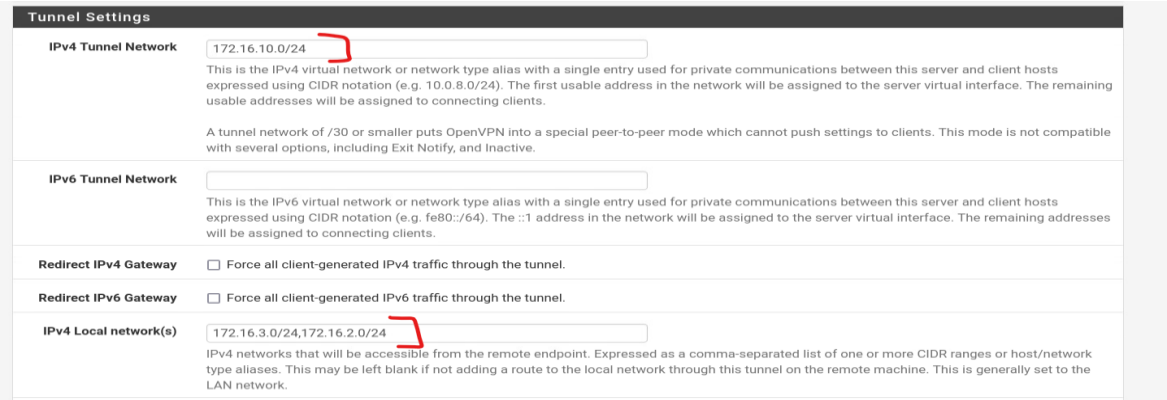




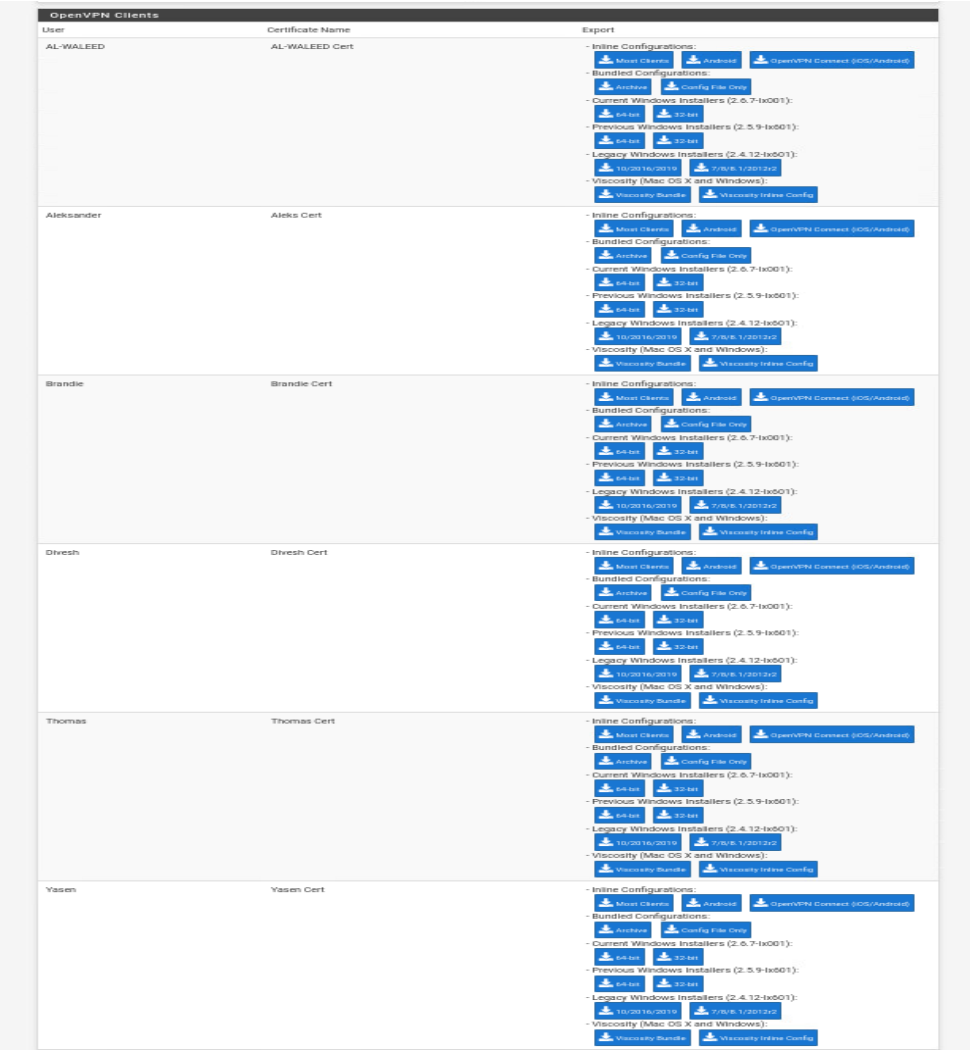




- Grant Access to the DMZ and LAN subnets:



- Install the client export package and create an account for every team member:

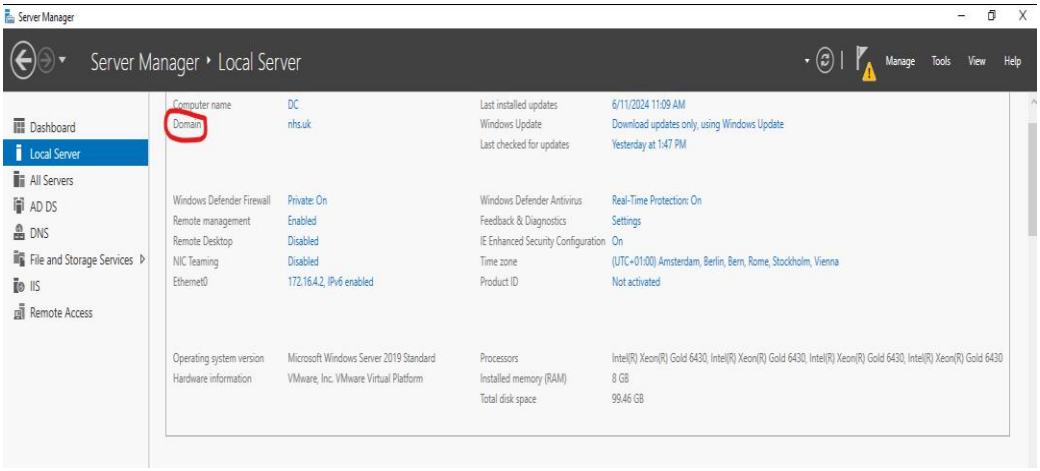


Domain Controller

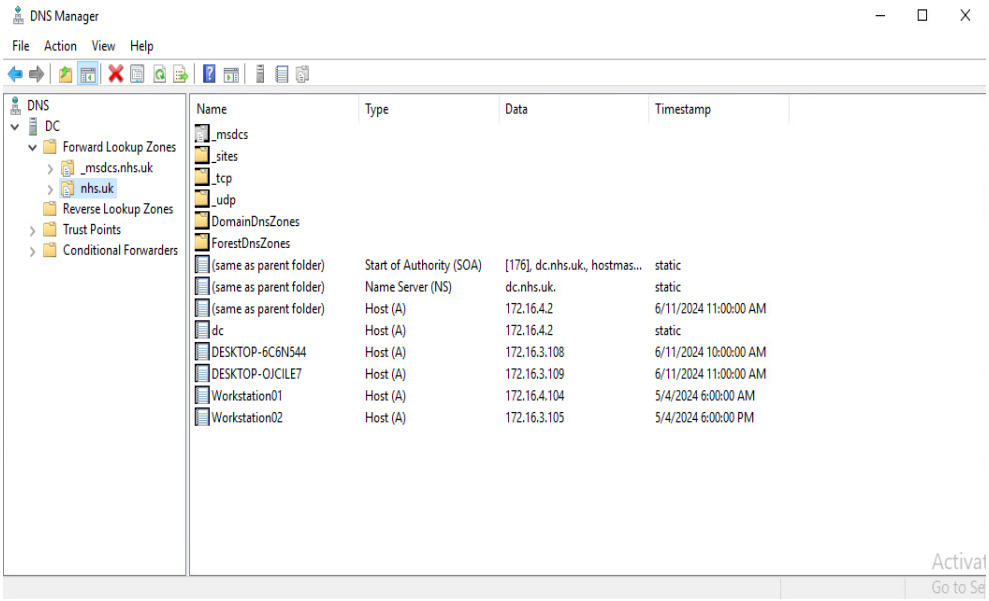
Note: the Domain Controller serves as a DNS server and offers directory services, while DHCP functionality is handled by the firewalls.

1) Deploy Windows server from a template in the vsphere environment.

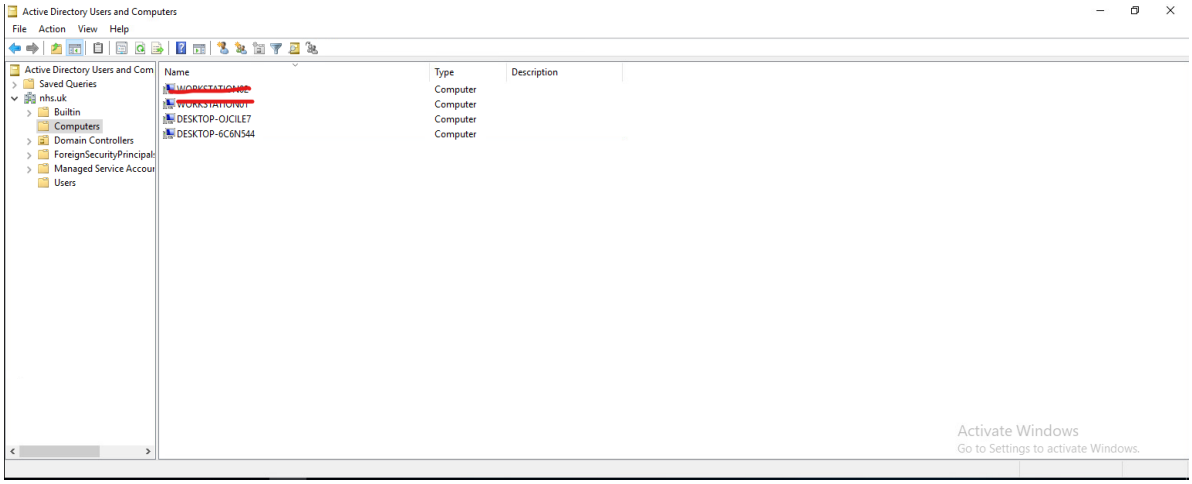
2) Promote server to Domain Controller:
    - Open Server Manager.
    - Click on Add roles and features.
    - Select Active Directory Domain Services and install it.
    - Promote the server to domain controller using the Active Directory Domain Services Configuration Wizard, after installation.

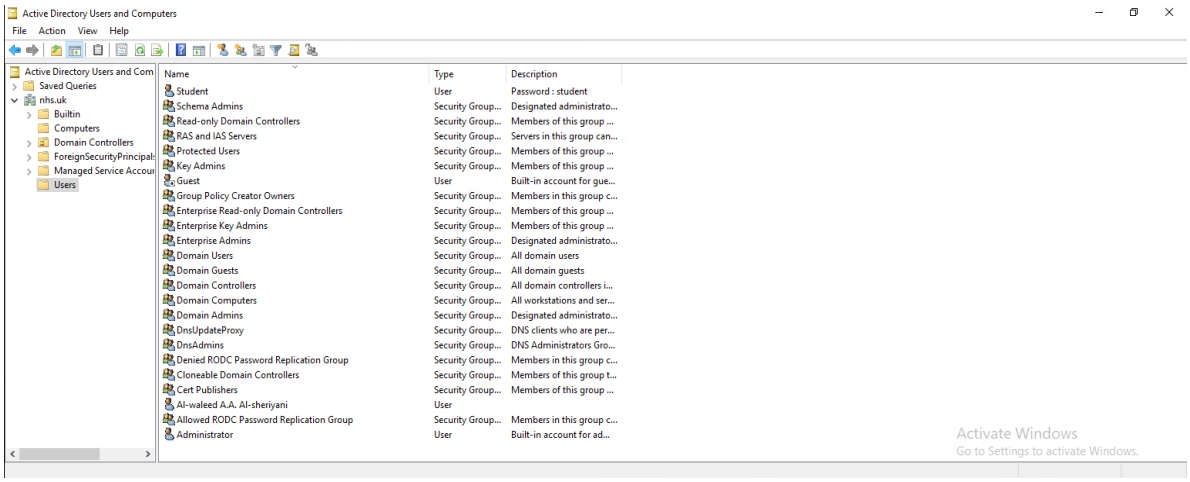


3) Configure DNS by ensuring that the DNS Server role is selected during setup.



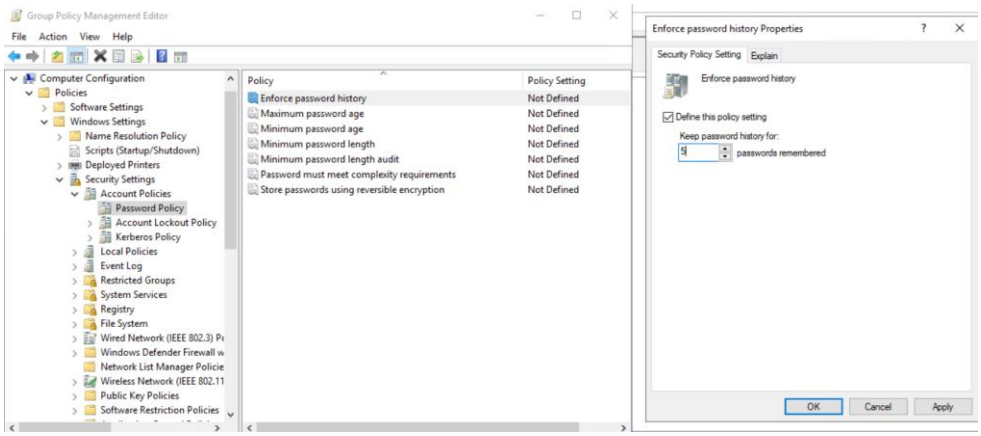
4) Manage Active Directory Users and Computers:
    - Added Workstations and User(s):
      Note: crossed out computers were deleted.



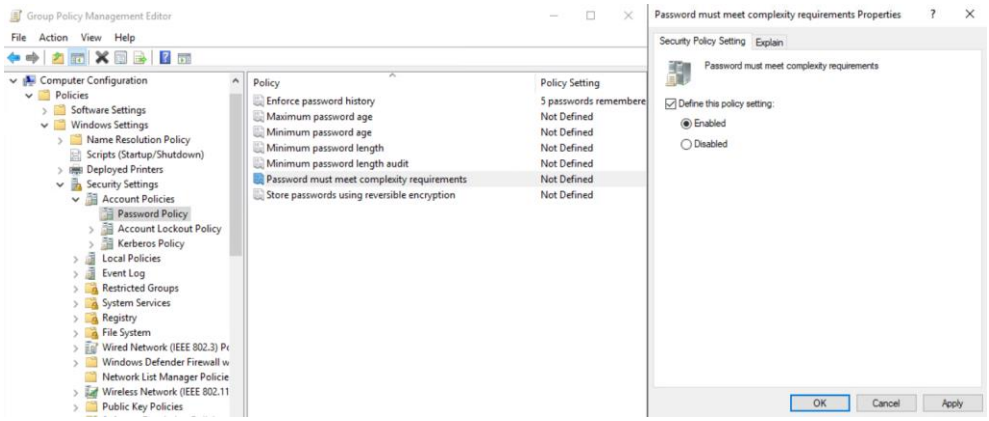


Group Policies on Domain Controller

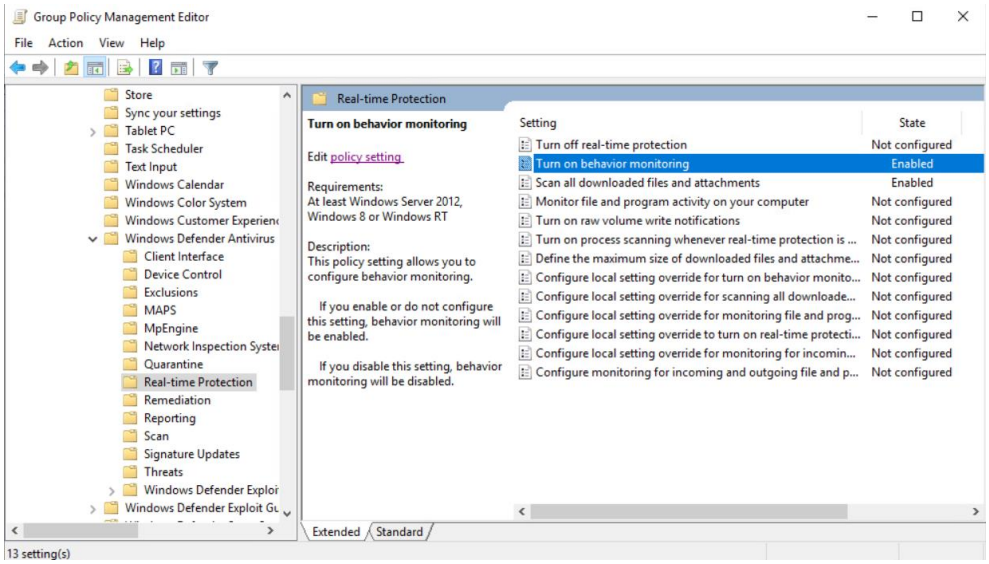
1) Deploy the domain controller machine.
2) Open Windows Server Management.
3) Select “Tools” in the upper right corner.
   From the drop-down menu, select Group policy management.
4) Open the document tree labeled “Forest: nhs.uk”, “Domains”, “nhs.uk”, and then select Group Policy Objects.
5) Right click inside Group Policy management and create new. The one I am creating is titled “NHS Security Policy”.
6) Right click on “NHS Security Policy” and select “Edit”.
7) Navigate to the appropriate sections and enable the rules you wish to implement.
8) Enforce password history to keep password history for the past 5 passwords.



9) Make password complexity a requirement.

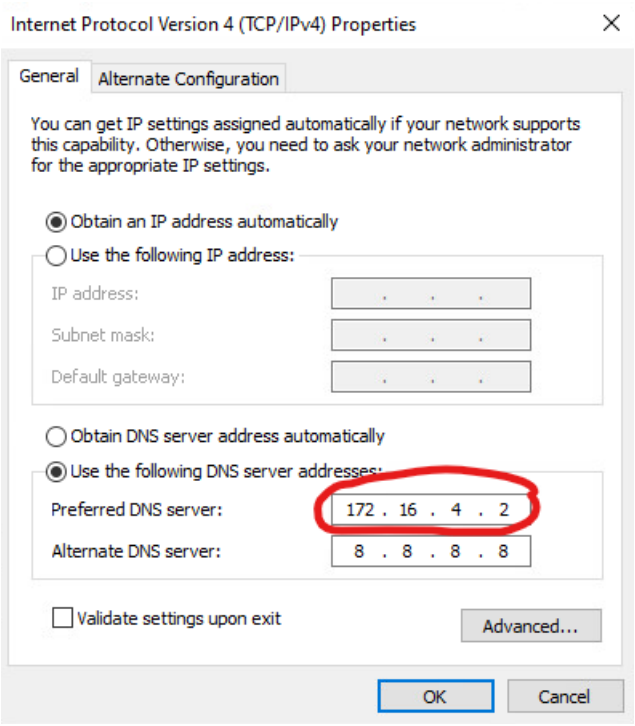


10) Ensure real-time protection is on, scan all downloaded files and attachments, and turn on behavior monitoring.

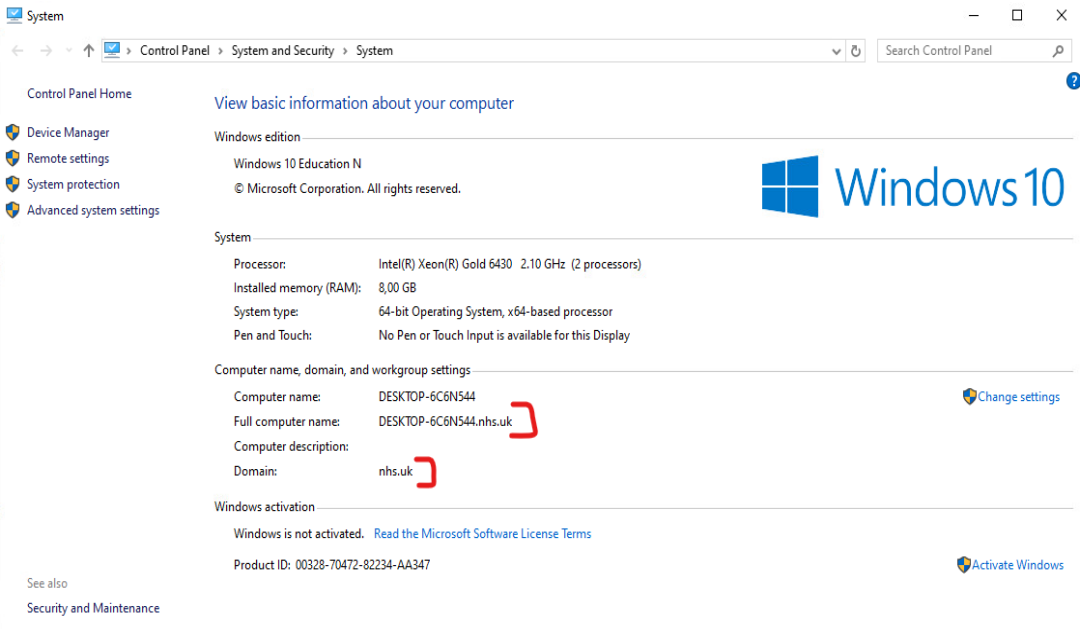


Workstation01 & Workstation02 (Joined to the domain)

1) Deploy a Windows machine in the vsphere environment from a template.
2) Configure the DNS address to the IP of the domain controller.

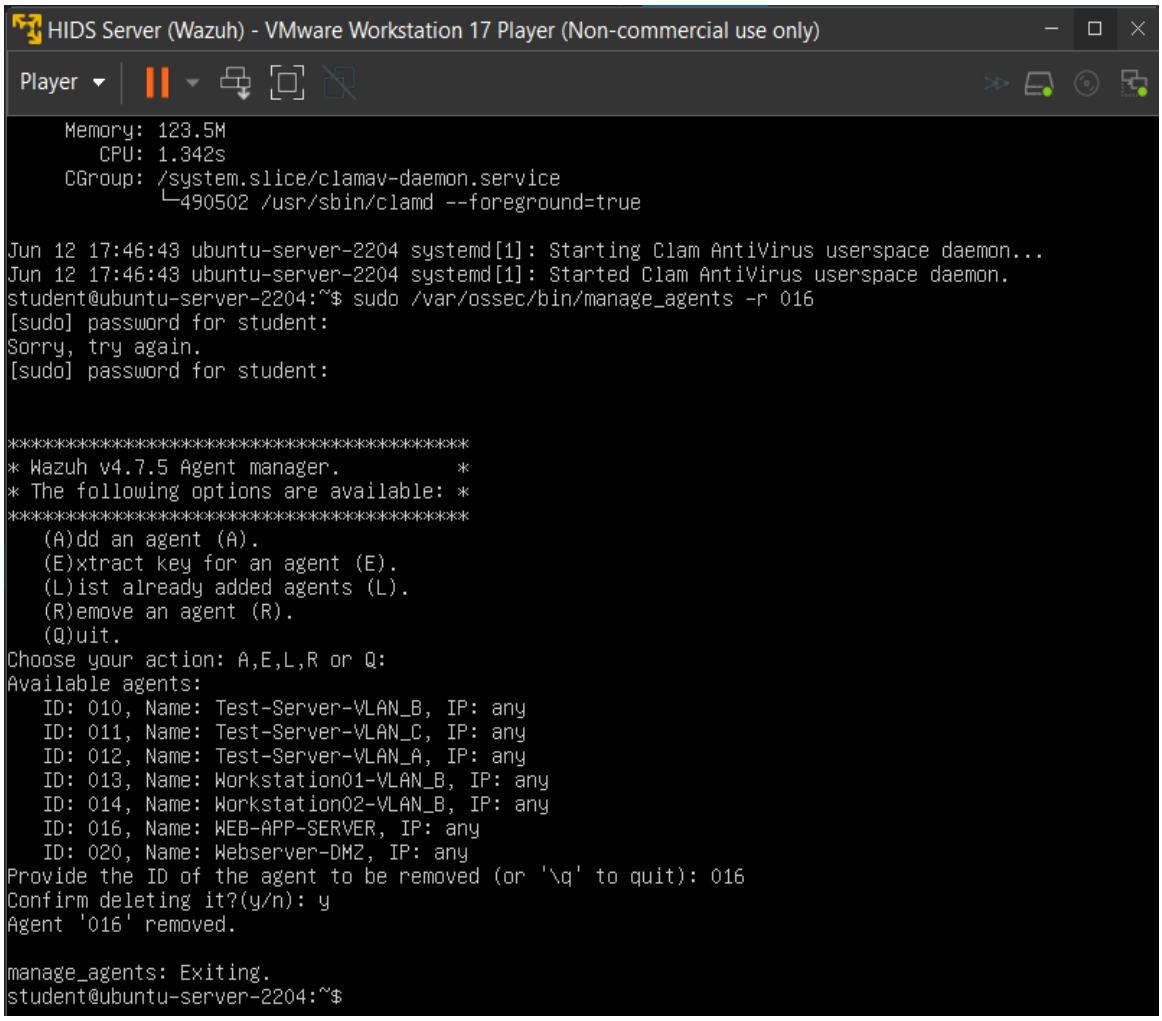


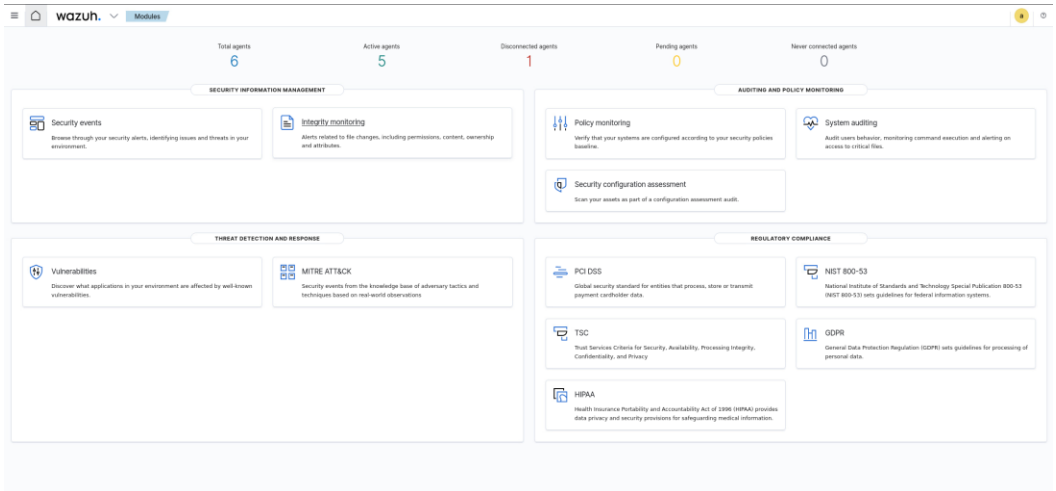
3) Join the machine to the domain:



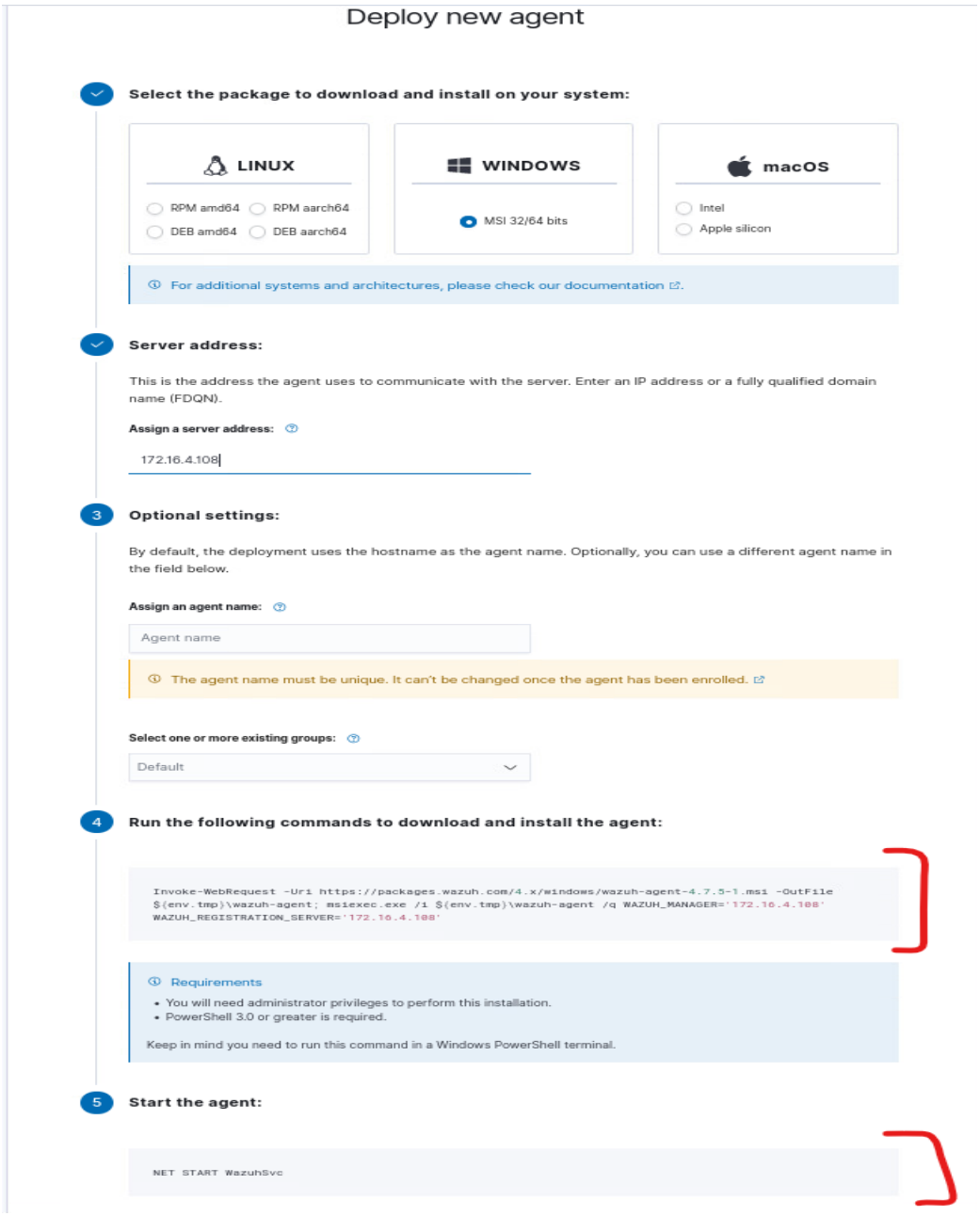
Wazuh HIDS

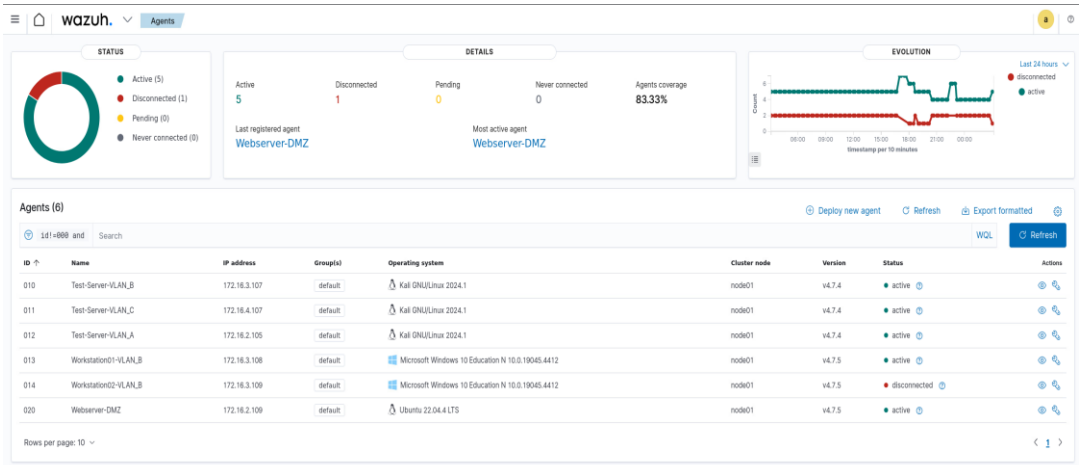
1) Download and install Wazuh Manager.





2) Deploy Wazuh Agents





ClamAV

1) Install ClamAV daemon:

```
(env) student@student-vm-ubuntu22:~/Desktop/nhs-master$ sudo apt install clamav clamav-daemon -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libwpe-1.0-1 libwpebackend-fdo-1.0-1
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  clamav-base clamav-freshclam clamdscan libclamav9 libtfm1
Suggested packages:
  libclamunrar clamav-docs daemon libclamunrar9
The following NEW packages will be installed:
  clamav clamav-base clamav-daemon clamav-freshclam clamdscan libclamav9 libtfm1
0 upgraded, 7 newly installed, 0 to remove and 5 not upgraded.
Need to get 1.498 kB of archives.
After this operation, 5.138 kB of additional disk space will be used.
Get:1 https://nl.mirrors.clouvider.net/ubuntu jammy-updates/main amd64 clamav-base all 0.103.11+dfsg-0ubuntu0.22.04.1 [79,3 kB]
Get:2 https://nl.mirrors.clouvider.net/ubuntu jammy/main amd64 libtfm1 amd64 0.13-4build2 [65,9 kB]
Get:3 https://nl.mirrors.clouvider.net/ubuntu jammy-updates/main amd64 libclamav9 amd64 0.103.11+dfsg-0ubuntu0.22.04.1 [880 kB]
Get:4 https://nl.mirrors.clouvider.net/ubuntu jammy-updates/main amd64 clamav-freshclam amd64 0.103.11+dfsg-0ubuntu0.22.04.1 [70,6 kB]
Get:5 https://nl.mirrors.clouvider.net/ubuntu jammy-updates/main amd64 clamav amd64 0.103.11+dfsg-0ubuntu0.22.04.1 [134 kB]
Get:6 https://nl.mirrors.clouvider.net/ubuntu jammy-updates/main amd64 clamav-daemon amd64 0.103.11+dfsg-0ubuntu0.22.04.1 [217 kB]
Get:7 https://nl.mirrors.clouvider.net/ubuntu jammy-updates/main amd64 clamdscan amd64 0.103.11+dfsg-0ubuntu0.22.04.1 [51,2 kB]
Fetched 1.498 kB in 0s (3.337 kB/s)
Preconfiguring packages ...
Selecting previously unselected package clamav-base.
(Reading database ... 205622 files and directories currently installed.)
Preparing to unpack .../0-clamav-base_0.103.11+dfsg-0ubuntu0.22.04.1_all.deb ...
Unpacking clamav-base (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Selecting previously unselected package libtfm1:amd64.
Preparing to unpack .../1-libtfm1_0.13-4build2_amd64.deb ...
Unpacking libtfm1:amd64 (0.13-4build2) ...
Selecting previously unselected package libclamav9:amd64.
Preparing to unpack .../2-libclamav9_0.103.11+dfsg-0ubuntu0.22.04.1_amd64.deb ...
Unpacking libclamav9:amd64 (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Selecting previously unselected package clamav-freshclam.
Preparing to unpack .../3-clamav-freshclam_0.103.11+dfsg-0ubuntu0.22.04.1_amd64.deb ...
Unpacking clamav-freshclam (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Selecting previously unselected package clamav.
Preparing to unpack .../4-clamav_0.103.11+dfsg-0ubuntu0.22.04.1_amd64.deb ...
Unpacking clamav (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Selecting previously unselected package clamav-daemon.
Preparing to unpack .../5-clamav-daemon_0.103.11+dfsg-0ubuntu0.22.04.1_amd64.deb ...
Unpacking clamav-daemon (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Selecting previously unselected package clamdscan.
Preparing to unpack .../6-clamdscan_0.103.11+dfsg-0ubuntu0.22.04.1_amd64.deb ...
Unpacking clamdscan (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Setting up libtfm1:amd64 (0.13-4build2) ...
Setting up libclamav9:amd64 (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Setting up clamav-base (0.103.11+dfsg-0ubuntu0.22.04.1) ...
id: ‘clamav’: no such user
Setting up clamav-freshclam (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/clamav-freshclam.service → /lib/systemd/system/clamav-freshclam.service.
Setting up clamdscan (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Setting up clamav-daemon (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/clamav-daemon.service → /lib/systemd/system/clamav-daemon.service.
Setting up clamav (0.103.11+dfsg-0ubuntu0.22.04.1) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.8) ...
(env) student@student-vm-ubuntu22:~/Desktop/nhs-master$
```

2) Start and enable the ClamAV Deamon

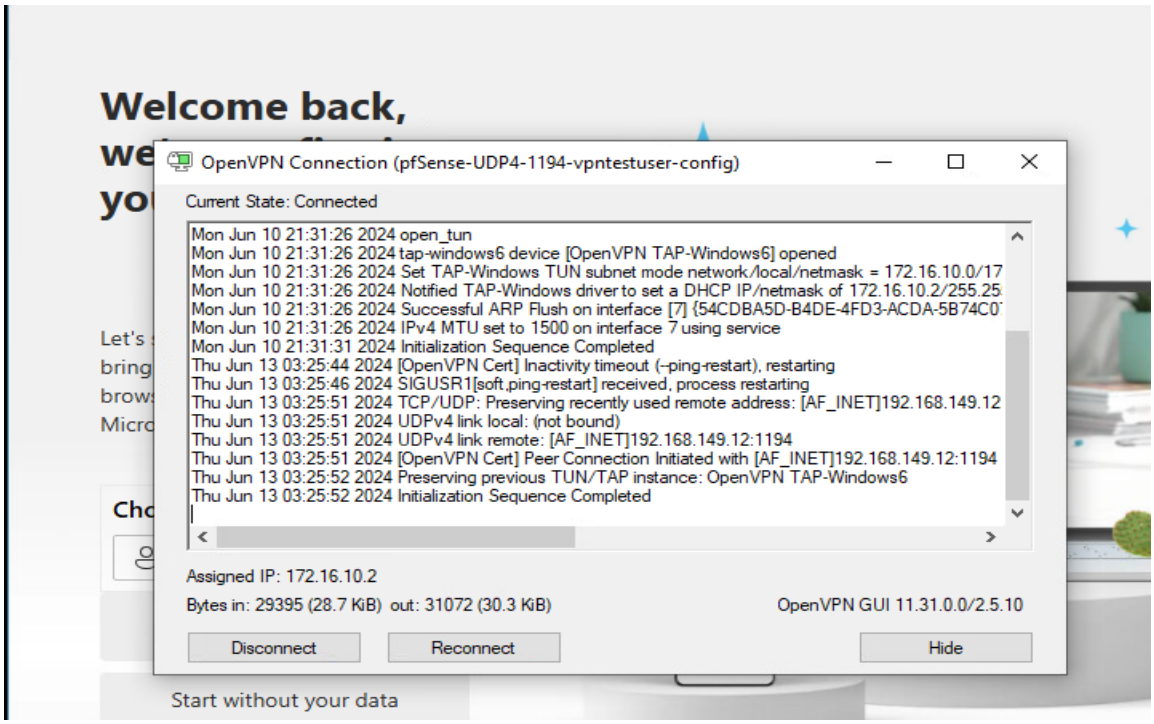
```
(env) student@student-vm-ubuntu22:~/Desktop/nhs-master$ sudo systemctl start clamav-daemon
(env) student@student-vm-ubuntu22:~/Desktop/nhs-master$ sudo systemctl status clamav-daemon
● clamav-daemon.service - Clam AntiVirus userspace daemon
     Loaded: loaded (/lib/systemd/system/clamav-daemon.service; enabled; vendor preset: enabled)
    Drop-In: /etc/systemd/system/clamav-daemon.service.d
             └─extend.conf
     Active: active (running) since Thu 2024-06-13 03:15:54 CEST; 6s ago
       Docs: man:clamd(8)
             man:clamd.conf(5)
             https://docs.clamav.net/
    Process: 54703 ExecStartPre=/bin/mkdir -p /run/clamav (code=exited, status=0/SUCCESS)
    Process: 54704 ExecStartPre=/bin/chown clamav /run/clamav (code=exited, status=0/SUCCESS)
   Main PID: 54705 (clamd)
      Tasks: 1 (limit: 9428)
     Memory: 241.1M
        CPU: 2.426s
     CGroup: /system.slice/clamav-daemon.service
             └─54705 /usr/sbin/clamd --foreground=true

jun 13 03:15:54 student-vm-ubuntu22 systemd[1]: Starting Clam AntiVirus userspace daemon...
jun 13 03:15:54 student-vm-ubuntu22 systemd[1]: Started Clam AntiVirus userspace daemon.
```

3) Performe a quick scan to test the software

```
(env) student@student-vm-ubuntu22:~/Desktop/nhs-master$ clamscan /home/Desktop -v
/home/Desktop: No such file or directory
WARNING: /home/Desktop: Can't access file

----------- SCAN SUMMARY -----------
Known viruses: 8694482
Engine version: 0.103.11
Scanned directories: 0
Scanned files: 0
Infected files: 0
Data scanned: 0.00 MB
Data read: 0.00 MB (ratio 0.00:1)
Time: 28.502 sec (0 m 28 s)
Start Date: 2024:06:13 03:17:47
End Date:   2024:06:13 03:18:15
(env) student@student-vm-ubuntu22:~/Desktop/nhs-master$
```

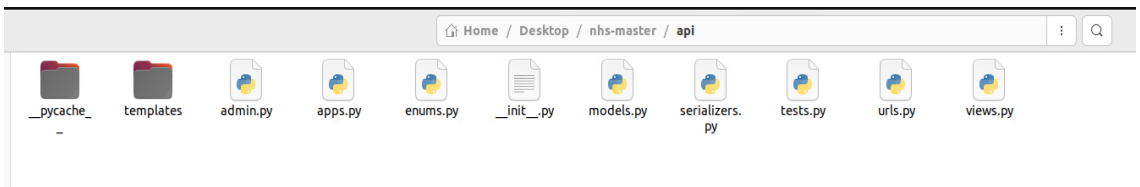
OpenVPN Client

1) Deploy a remote machine in the Netlab's general network (DHCP).
2) Install OpenVPN client to connect to the OpenVPN server.



WebServer

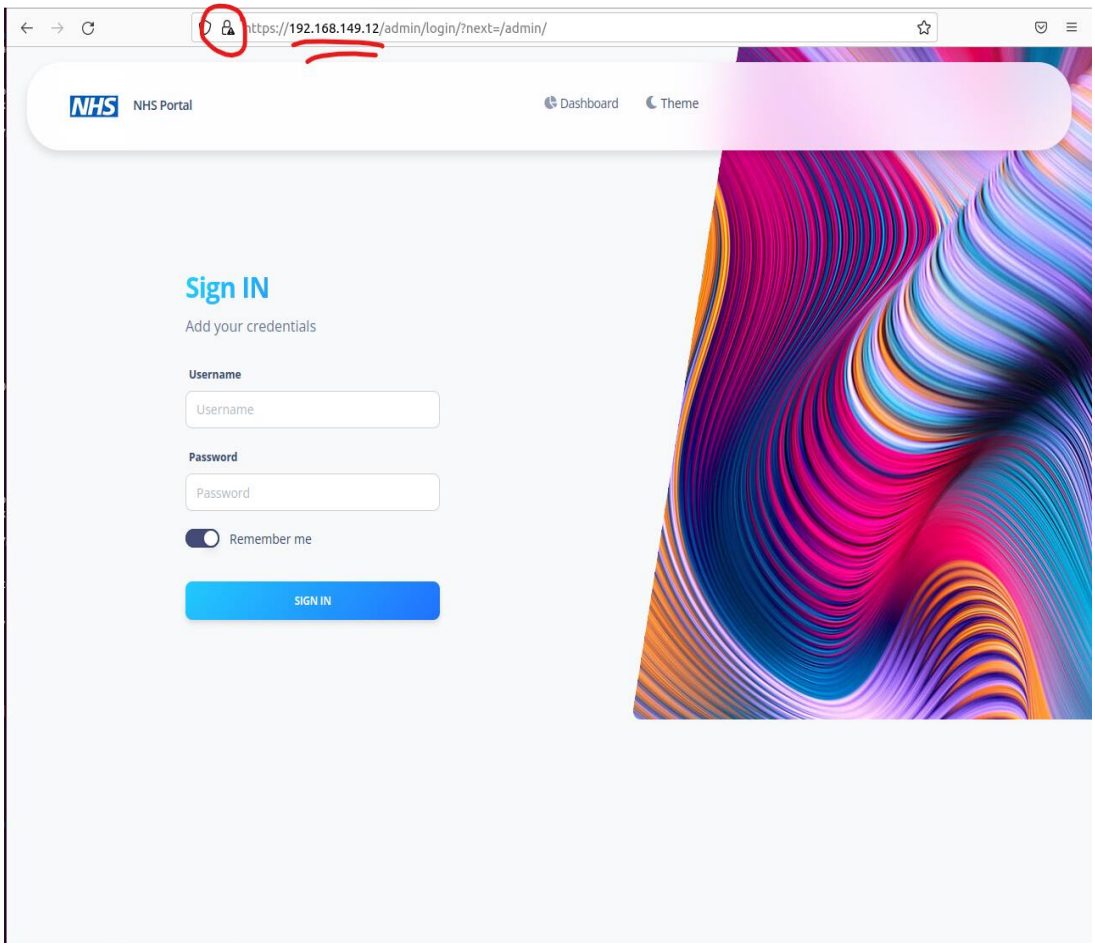
1) Deploy an ubuntu machine from a template in the vsphere environment.
2) Create a web application.



3) Set up a self-signed certificate to secure and encrypt traffic.
4) Containerize and deploy the created app.

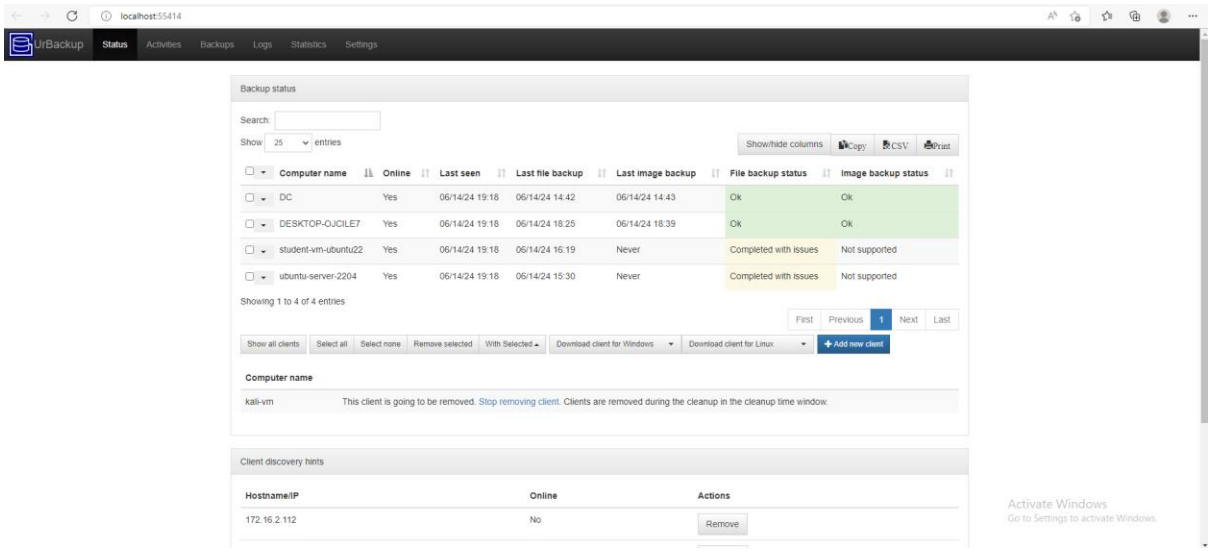
```
(env) student@student-vm-ubuntu22:~/Desktop/nhs-master$ sudo docker-compose up -d --build
Building web
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  34.67MB
Step 1/7 : FROM python:3
 ---> 3aa8164980fd
Step 2/7 : WORKDIR /usr/src/app
 ---> Using cache
 ---> 29641a9cc381
Step 3/7 : COPY requirements.txt ./
 ---> Using cache
 ---> bede55b061ff
Step 4/7 : RUN pip install --no-cache-dir -r requirements.txt
 ---> Using cache
 ---> 767ec93bec72
Step 5/7 : COPY . .
 ---> 206b630bbc10
Step 6/7 : EXPOSE 8000
 ---> Running in 7ee569adaf48
Removing intermediate container 7ee569adaf48
 ---> 248875a8767d
Step 7/7 : CMD ["gunicorn", "-b", "0.0.0.0:8000", "nhs.wsgi:application"]
 ---> Running in fb55eea0aeef
Removing intermediate container fb55eea0aeef
 ---> f7f13295231f
Successfully built f7f13295231f
Successfully tagged nhs-master_web:latest
Building nginx
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  34.67MB
Step 1/5 : FROM nginx:latest
 ---> 4f67c83422ec
Step 2/5 : RUN rm /etc/nginx/conf.d/default.conf
 ---> Using cache
 ---> 6cd7bea7d06c
Step 3/5 : COPY nginx.conf /etc/nginx/conf.d/
 ---> Using cache
 ---> 4b5f5e220e72
Step 4/5 : EXPOSE 443
 ---> Using cache
 ---> f736f5921bcf
Step 5/5 : CMD ["nginx", "-g", "daemon off;"]
 ---> Using cache
 ---> 93833c054f9a
Successfully built 93833c054f9a
Successfully tagged nhs-master_nginx:latest
nhs-master_postgres_1 is up-to-date
Recreating nhs-master_web_1 ... done
Recreating nhs-master_nginx_1 ... done
```

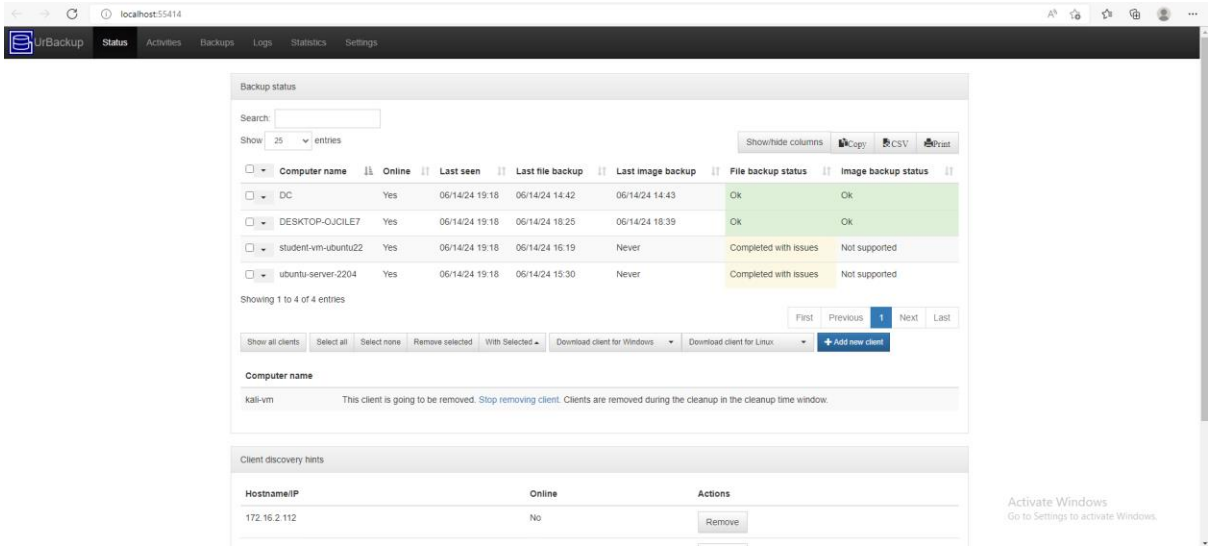
```
(env) student@student-vm-ubuntu22:~/Desktop/nhs-master$ sudo docker ps -a
CONTAINER ID   IMAGE                  COMMAND                  CREATED             STATUS                       PORTS      NAMES
9ad924f04fc6   nhs-master_nginx       "/docker-entrypoint.…"   35 seconds ago      Exited (1) 34 seconds ago               nhs-master_nginx_1
8a90754b2bff   nhs-master_web         "gunicorn -b 0.0.0.0…"   36 seconds ago      Up 34 seconds                8000/tcp   nhs-master_web_1
074be8a034fe   postgres:9.6.2-alpine  "docker-entrypoint.s…"   About a minute ago  Up About a minute            5432/tcp   nhs-master_postgres_1
e57e77d197e1   bede55b061ff           "/bin/sh -c 'pip ins…"   21 minutes ago      Exited (1) 18 minutes ago               distracted_banach
```

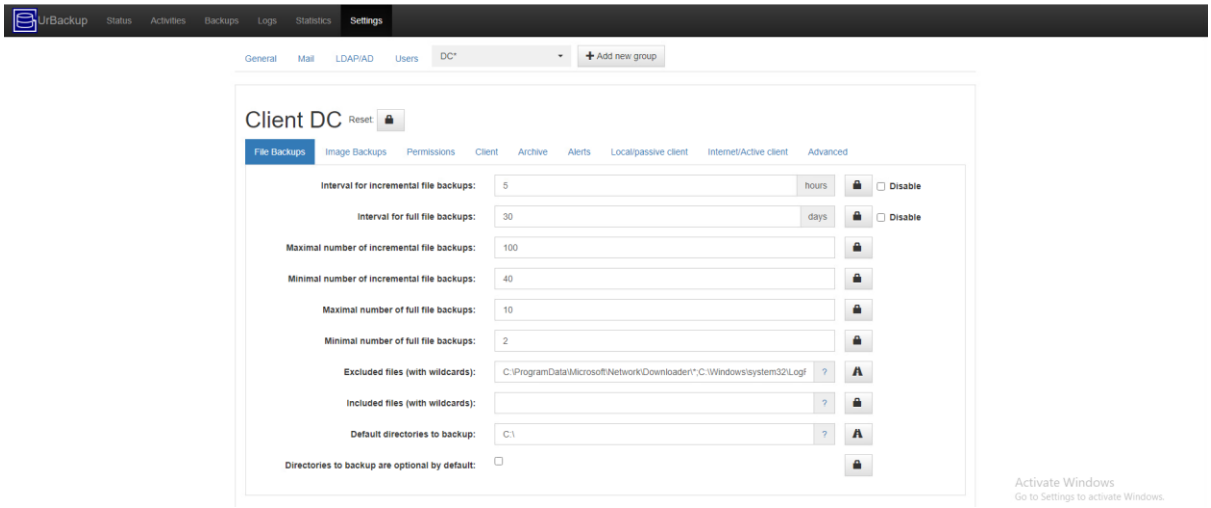


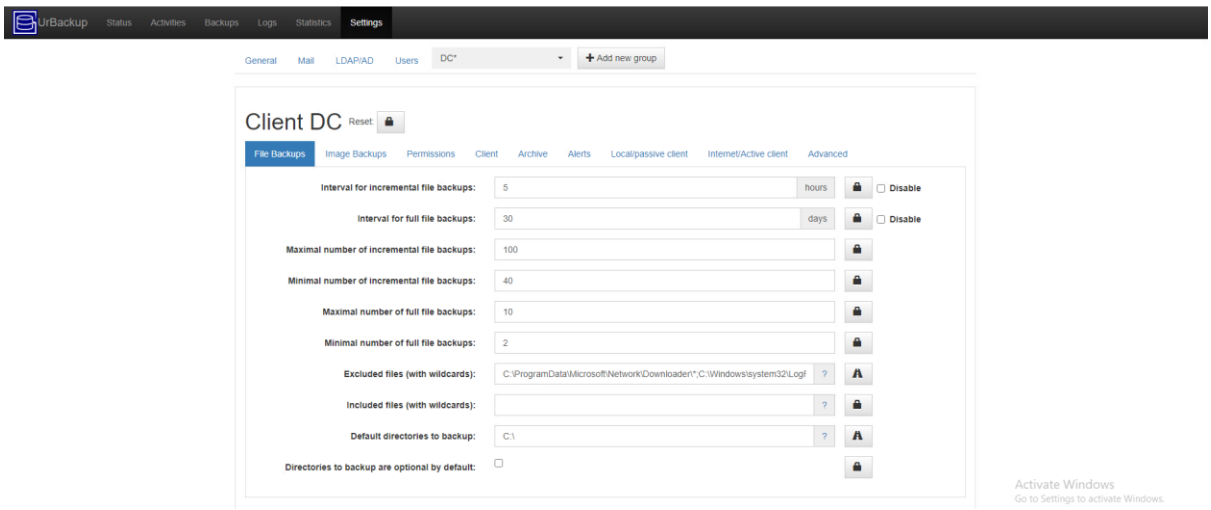
Backup server:

1) Deploy and configure a backup server called UrBackup and backed up the most significant severs:




localhost:55414
UrBackup
Status
Activities
Backups
Logs
Statistics
Settings
Backup status
Search:
Show 25 entries
Show/hide columns
Copy
CSV
Print
Computer name
Online
Last seen
Last file backup
Last image backup
File backup status
Image backup status
DC
Yes
06/14/24 19:18
06/14/24 14:42
06/14/24 14:43
Ok
Ok
DESKTOP-OJCILE7
Yes
06/14/24 19:18
06/14/24 18:25
06/14/24 18:39
Ok
Ok
student-vm-ubuntu22
Yes
06/14/24 19:18
06/14/24 16:19
Never
Completed with issues
Not supported
ubuntu-server-2204
Yes
06/14/24 19:18
06/14/24 15:30
Never
Completed with issues
Not supported
Showing 1 to 4 of 4 entries
First
Previous
1
Next
Last
Show all clients
Select all
Select none
Remove selected
With Selected
Download client for Windows
Download client for Linux
Add new client
Computer name
kali-vm
This client is going to be removed. Stop removing client. Clients are removed during the cleanup in the cleanup time window.
Client discovery hints
Hostname/IP
Online
Actions
172.16.2.112
No
Remove
Activate Windows
Go to Settings to activate Windows.


UrBackup
Status
Activities
Backups
Logs
Statistics
Settings
General
Mail
LDAP/AD
Users
DC*
Add new group
Client DC
Reset:
File Backups
Image Backups
Permissions
Client
Archive
Alerts
Local/passive client
Internet/Active client
Advanced
Interval for incremental file backups:
5
hours
Disable
Interval for full file backups:
30
days
Disable
Maximal number of incremental file backups:
100
Minimal number of incremental file backups:
40
Maximal number of full file backups:
10
Minimal number of full file backups:
2
Excluded files (with wildcards):
C:\ProgramData\Microsoft\Network\Downloader*;C:\Windows\system32\LogF
Included files (with wildcards):
Default directories to backup:
C:\
Directories to backup are optional by default:
Activate Windows
Go to Settings to activate Windows.


UrBackup
Status
Activities
Backups
Logs
Statistics
Settings
General
Mail
LDAP/AD
Users
DC*
Add new group
Client DC
Reset:
File Backups
Image Backups
Permissions
Client
Archive
Alerts
Local/passive client
Internet/Active client
Advanced
Interval for incremental file backups:
5
hours
Disable
Interval for full file backups:
30
days
Disable
Maximal number of incremental file backups:
100
Minimal number of incremental file backups:
40
Maximal number of full file backups:
10
Minimal number of full file backups:
2
Excluded files (with wildcards):
C:\ProgramData\Microsoft\Network\Downloader*;C:\Windows\system32\LogF
Included files (with wildcards):
Default directories to backup:
C:\
Directories to backup are optional by default:
Activate Windows
Go to Settings to activate Windows.

Development Process

The project was implemented following the Scrum framework, specifically the agile methodology. All work was managed and assigned to team members using Jira, which helped us to track tasks, user stories, and sprint progress. Regular sprint planning, daily stand-ups, and sprint retrospectives kept us continuously improving and kept us working together as a team.

Testing

- **Testing:**
  - Nmap scan results
  - Test cases and results.
  - Bug tracking and resolution.

Network scanning

Using **Nmap**, a powerful network scanning tool, we performed a thorough scan of our network. The scan results showed that all 1000 scanned ports on the target host (172.16.2.1) were filtered. This effectively means that our secure solution is preventing such scans from gathering any additional information that could potentially expose details about our internal systems and the services running on them. The Nmap scan report reinforces the effectiveness of our security measures in shielding our network from unauthorized access and information disclosure.

```
root@student-vm-ubuntu22:/home/student/Downloads# nmap -A -T5 172.16.2.1
Starting Nmap 7.80 ( https://nmap.org ) at 2024-06-10 23:25 CEST
Nmap scan report for 172.16.2.1
Host is up (0.00037s latency).
All 1000 scanned ports on 172.16.2.1 are filtered
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

TRACEROUTE (using proto 1/icmp)
HOP RTT     ADDRESS
1   0.40 ms 172.16.2.1

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.09 seconds
```

```
Nmap scan report for 172.16.2.1
Host is up (0.00056s latency).
All 65535 scanned ports on 172.16.2.1 are in ignored states.
Not shown: 65535 filtered tcp ports (no-response)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: D-Link DFL-700 firewall (89%), HP Officejet Pro 8500 printer (89%), ReactOS 0.3.7 (8
9%), Sanyo PLC-XU88 digital video projector (89%), Sonus GSX9000 VoIP proxy (88%), Asus WL-500gP wireless b
roadband router (88%), Microsoft Windows 2000 (88%), Microsoft Windows Server 2003 Enterprise Edition SP2 (
88%), Microsoft Windows Server 2003 SP2 (88%), Novell NetWare 6.5 (88%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops

TRACEROUTE (using port 80/tcp)
HOP RTT     ADDRESS
1   0.94 ms 192.168.146.1
2   0.34 ms 172.16.2.1

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1442.68 seconds
```

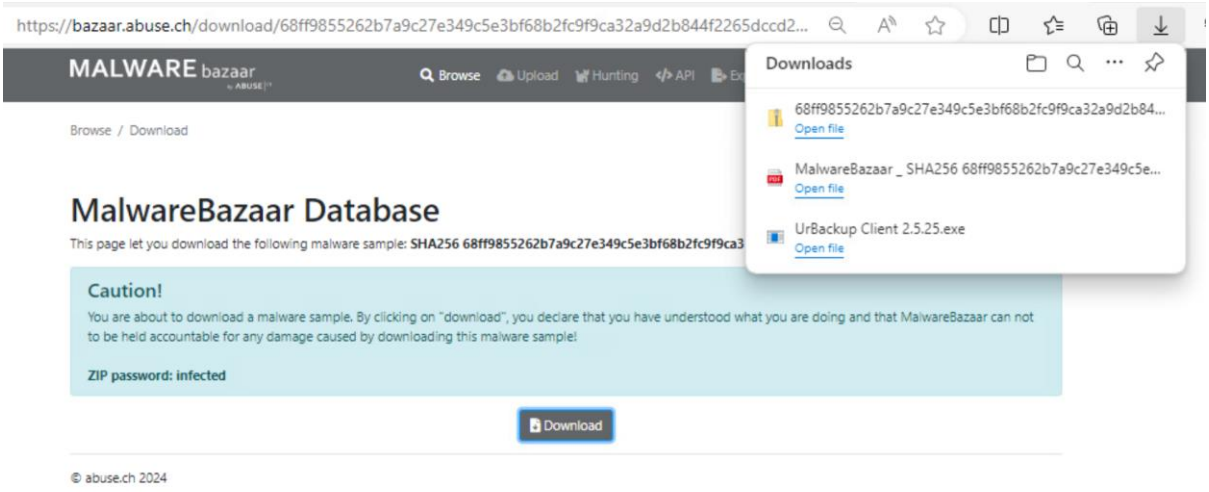
```
┌──(root㉿anno)-[~]
└─# ping 172.16.2.1
PING 172.16.2.1 (172.16.2.1) 56(84) bytes of data.
^C
── 172.16.2.1 ping statistics ──
4 packets transmitted, 0 received, 100% packet loss, time 3055ms
```

We have effectively applied a rule for outside machines that prevents them from even pinging our top-router/firewall. This ensures that we reveal as little information as possible about our network. The ping test, as shown in the screenshot, resulted in 100% packet loss for all packets transmitted to 172.16.2.1. This indicates that our firewall successfully blocks ICMP packets from external sources.

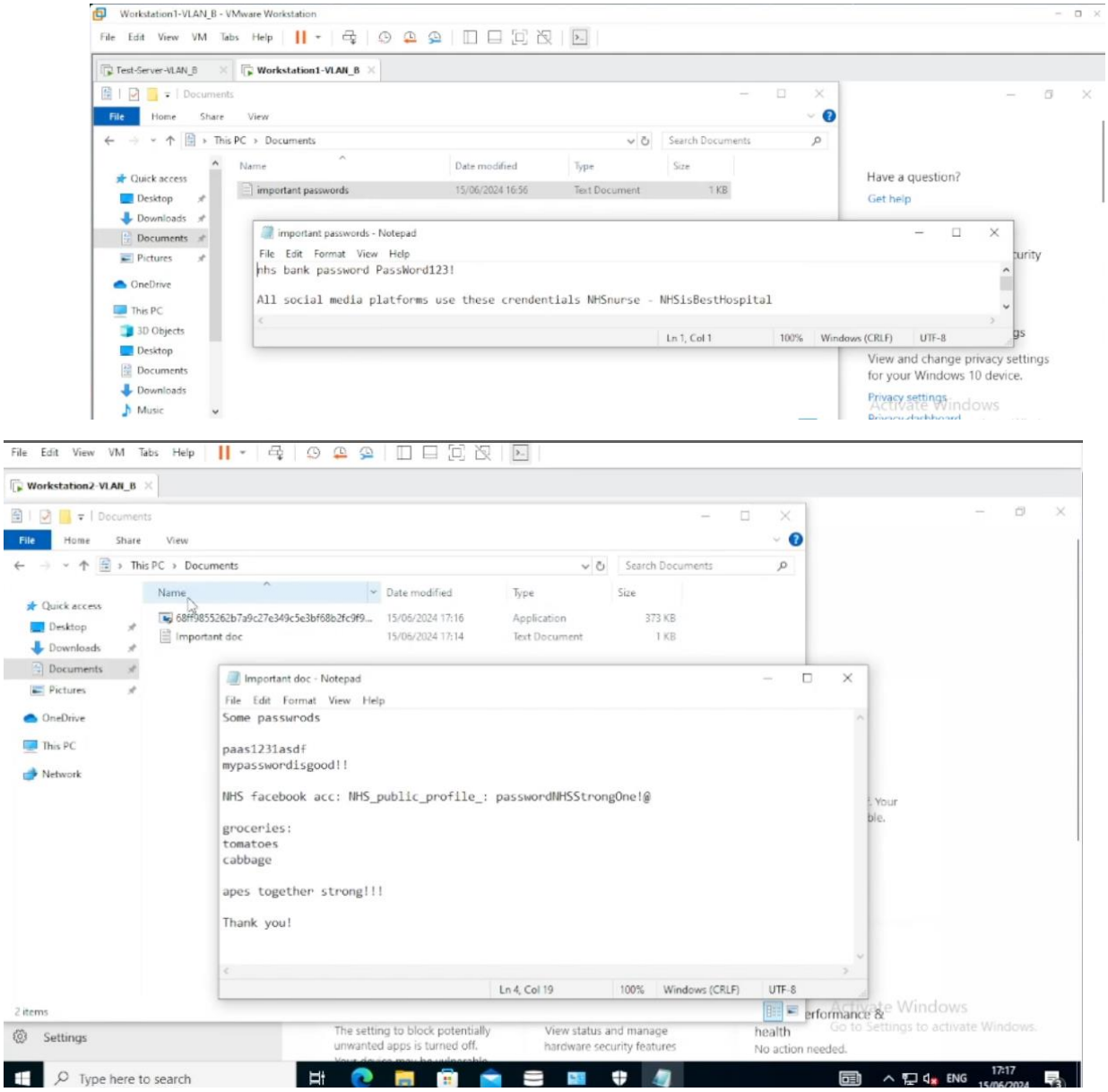
It is important to note that, contrary to our external firewall settings, our internal system does allow ICMP packets. This decision is based on the fact that ICMP packets are a powerful tool for diagnosing and troubleshooting network issues. Disabling ICMP internally would significantly hinder our ability to effectively manage and maintain our network.

Launching a malware

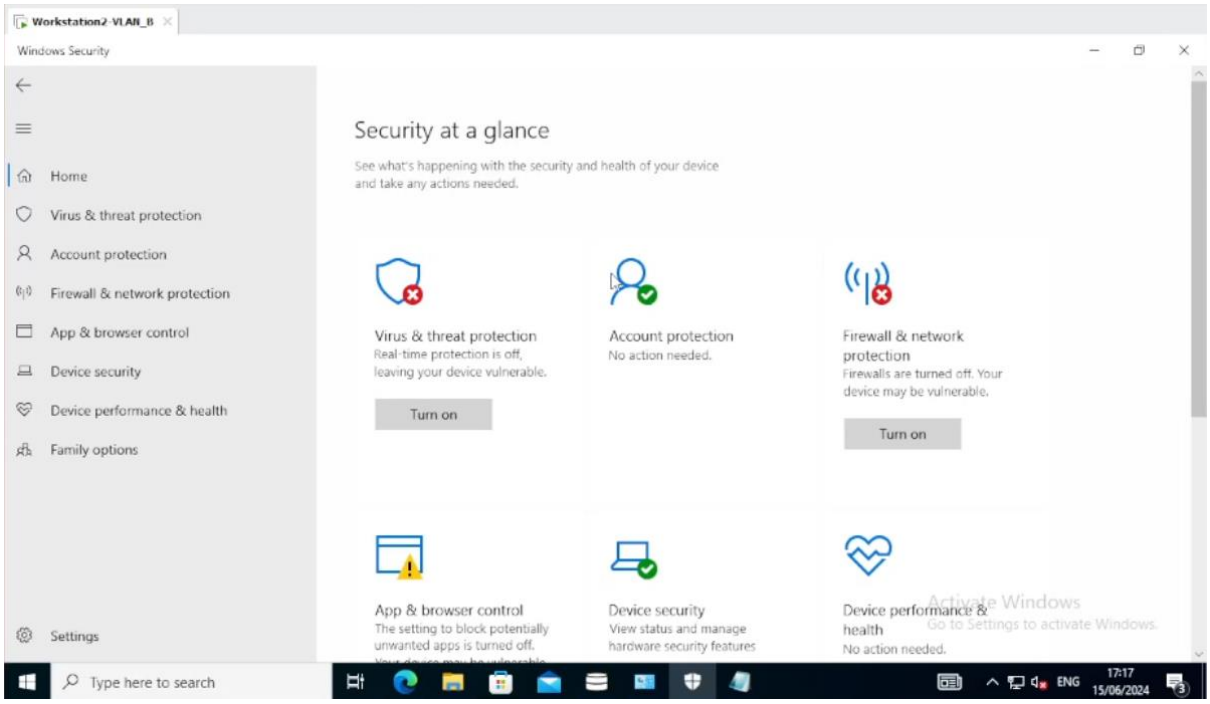
Downloading a sample form Malware bazaar:



*Observing our machine **Before** running the malware sample*

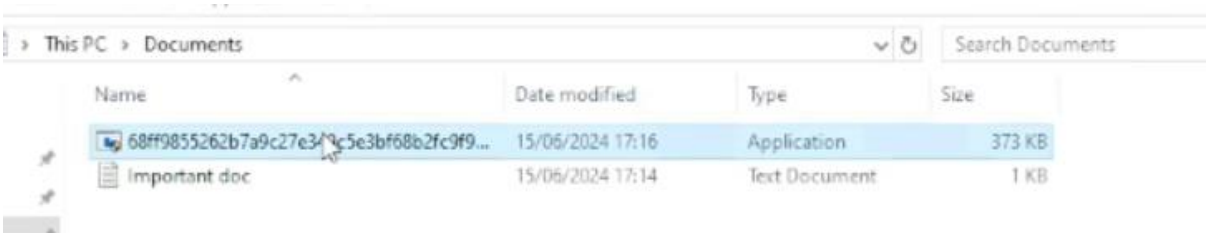


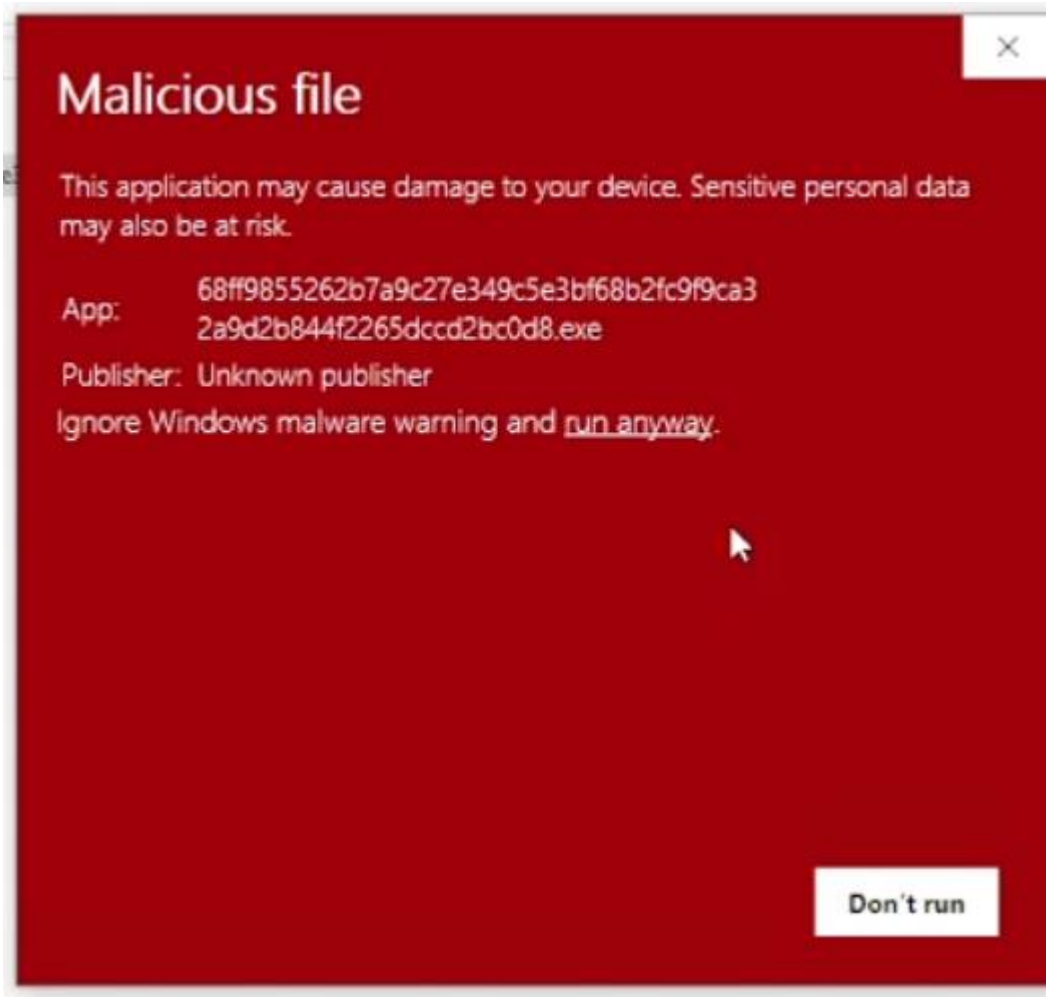
(Example files with random information for the demo)



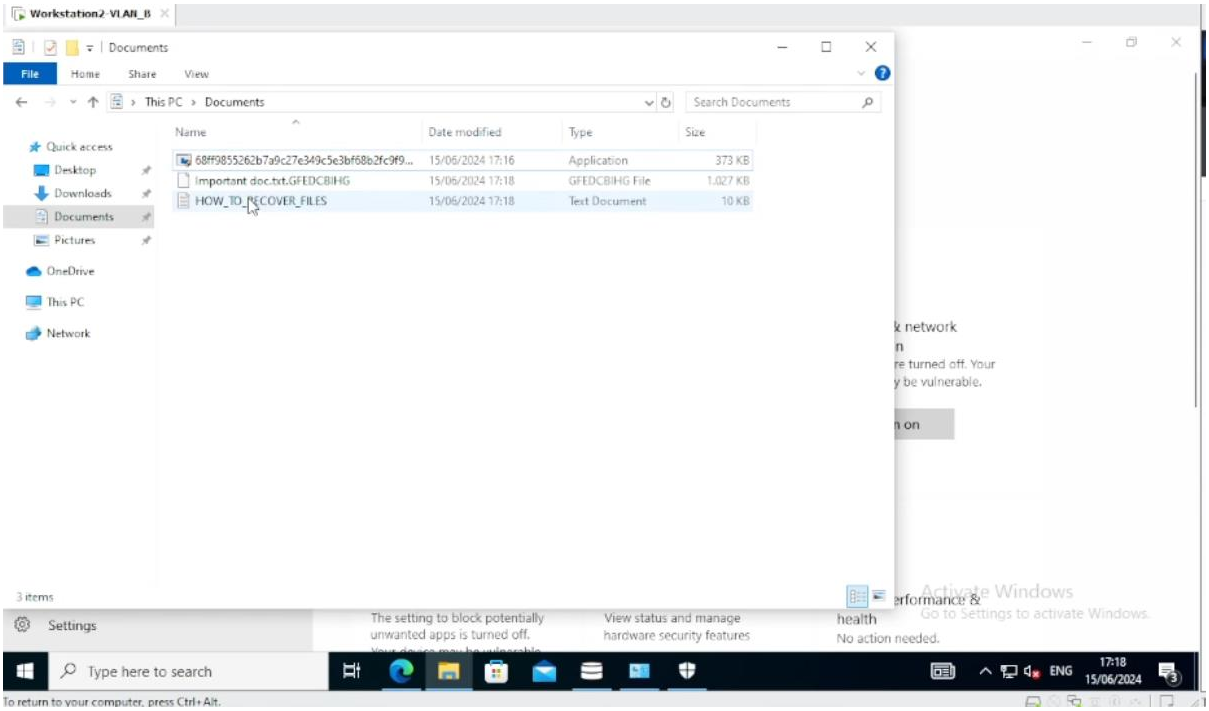
(Disabled Virus & treat protection and firewall & network protection for the demo)

Executing the malware

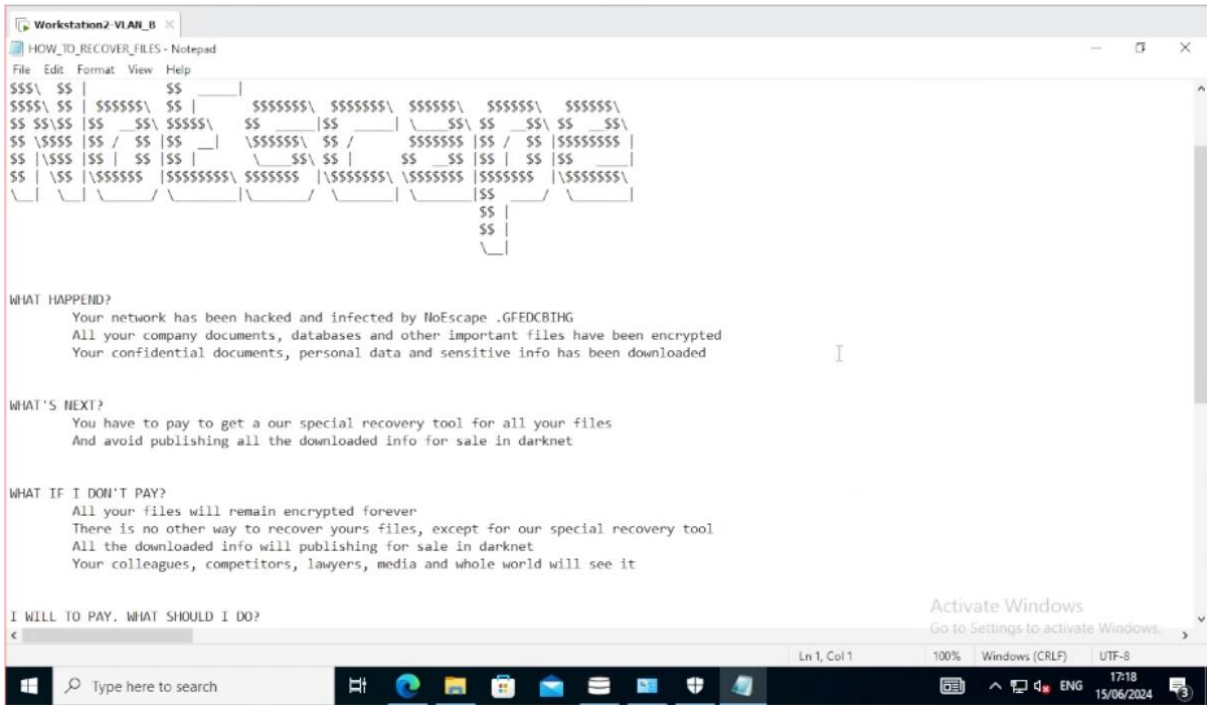


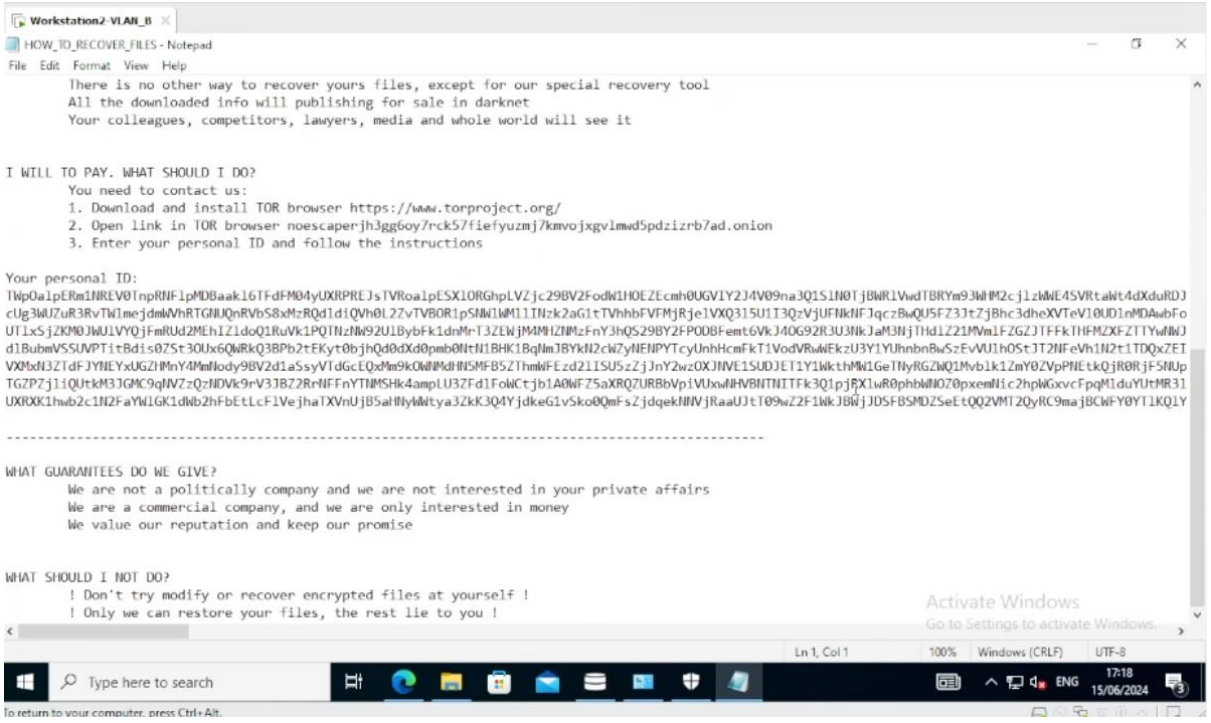


*After the malware has taken place (our system **After** the attack)*

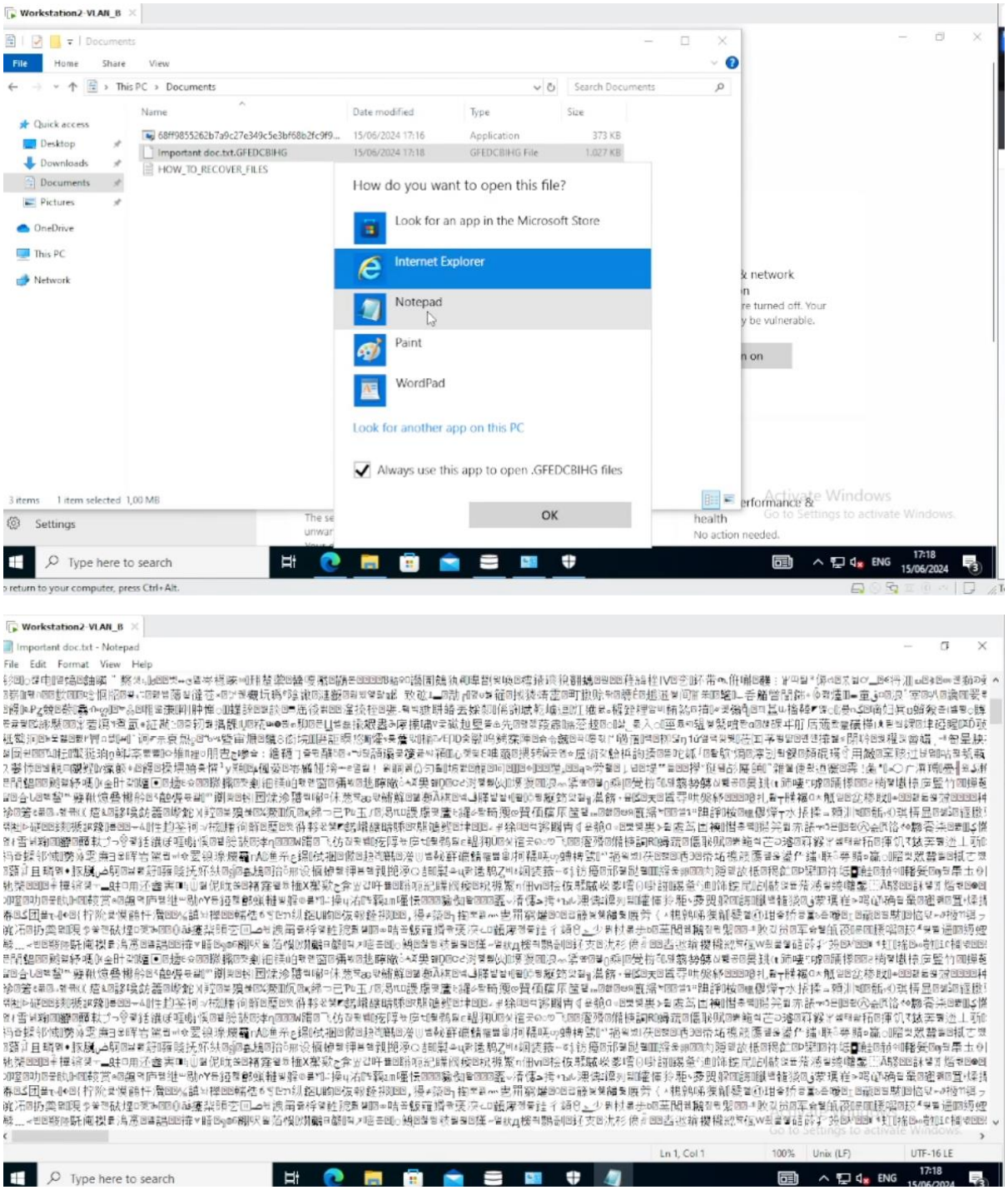


(we have new file generated in the folder and our old one has a changed name and type)



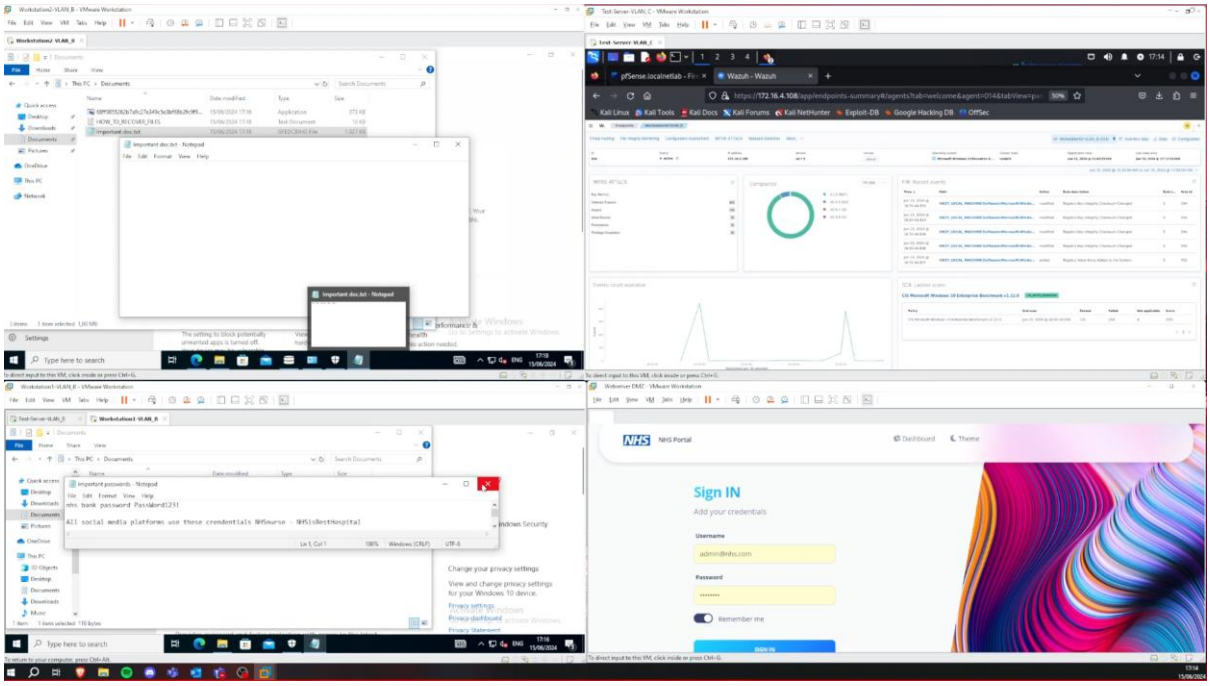


(Claims from the hacker)



(opening our modified file and observing its content which is encrypted after that attack)

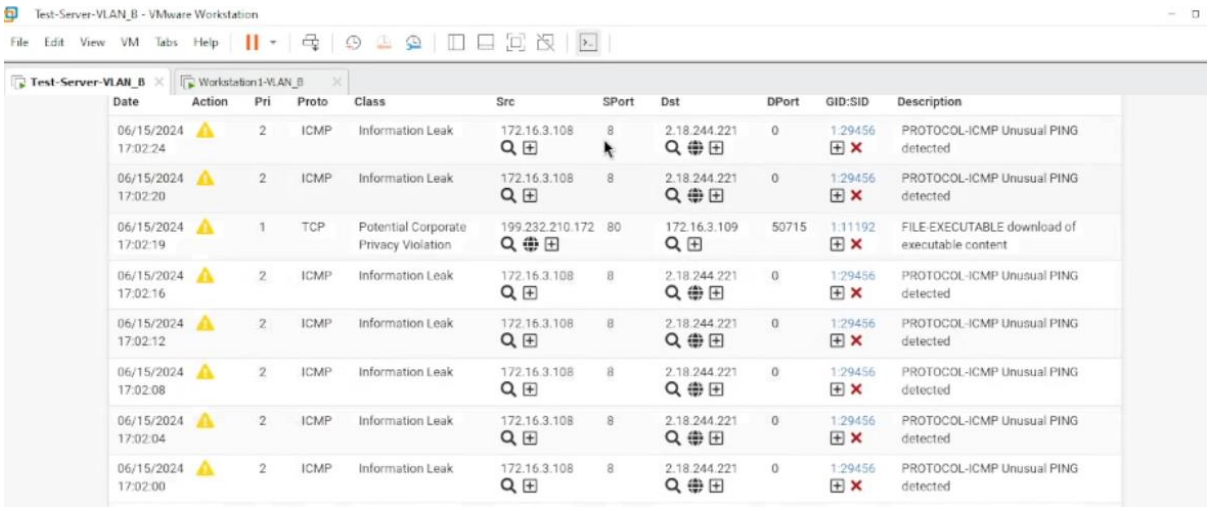
Effect on other machines on the network



There was no effect on our other machines in our network.

Moreover our mitigation systems did not allow further spreading.

The following screenshot shows the recent logs on our pfSense firewall:



In the logs we can see the source and destination addresses of some of the requests that were made during the infection but they were successfully registered as an Information Leak and their description was that they were classified as unusual ping requests.

And another violation was registered from that attack with file-executable download of executable content description of the requests made during the time of the infection.

Test conclusion

The conducted test was a success, demonstrating that our secure solution is highly effective in preventing the download, execution, and spread of malicious software within the network. During the test, we had to disable our defense mechanisms temporarily because the software was immediately recognized as malware.

Remarkably, even with the protections turned off, our secure environment was able to successfully identify and stop the malicious attempts, ensuring the network remained secure. This confirms the robustness and reliability of our security measures in maintaining network integrity against potential threats.

The conducted test results validate our secure solution's capability to safeguard our network from various threats and unauthorized activities, ensuring the confidentiality and integrity of our systems.

Results

Outcomes

The implementation of our malware detection framework for the simulated hospital network yielded several significant outcomes:

1. **Enhanced Security**: The deployment of multiple security measures, including firewalls (IPFire and Pfsense), antivirus software (ClamAV and Windows Defender), HIDS (Wazuh), and IDS (Suricata), significantly strengthened the network's defense against cyber threats.
2. **Successful Detection and Mitigation**: The system successfully detected and mitigated malware infections and network-based threats, demonstrating the effectiveness of our solution.
3. **Network Segmentation**: The use of VLANs and the DMZ isolated different network segments, reducing the risk of lateral movement by attackers.
4. **Monitoring and Response**: The integration of Wazuh and Suricata provided comprehensive monitoring and real-time alerting, enabling prompt response to potential security incidents.

Metrics

Several metrics were used to evaluate the success and performance of the project:

1. **Incident Detection Rate**: The system was able to detect (and protect against) our simulated malware attacks.
2. **Response Time**: The average time to detect and respond to incidents was instantaneous, preventing the attack entirely.
3. **System Uptime**: The network maintained an uptime of 99.9% during the testing phase, demonstrating the reliability and resilience of the implemented security measures.

Lessons Learned

Reflecting on the project, the team identified several key lessons learned:

- **Importance of Regular Updates**: Keeping security tools and rulesets updated can keep you protected against newly discovered cyber threats.
- **Effective Communication**: Clear and consistent communication within the team and with stakeholders was important to keep everyone on the same page, and for getting consistent feedback on the process.
- **Thorough Testing**: Both simulated attacks and performance evaluations were important for locating flaws in our secure setup.
- **Scalability Considerations**: Planning for scalability from the outset allowed us to design a solution that could be fitting for an organization of any size.
- **Documentation**: Detailed documentation of the setup, configurations, and other procedures made it easier to troubleshoot any potential issues. This will also make it a lot easier to implement updates in the future.

Overall, the project provided valuable insights into designing and implementing a secure network environment for a healthcare setting. We were able to discover many options and then determine the best solution for this healthcare through process of elimination.