**Secure Network Design and Implementation project**

**CSCE 5585**

**Tools Used**

*Network Emulation*: Gns3

*Virtualizatio*: Vmware Workstation

*Pentesting:* Kali Linux-nmap,wireshark,hydra,ping

*IDS*: Fail2ban

*Firewall*: Watchguard xtmV

*Server*:Ubuntu Server

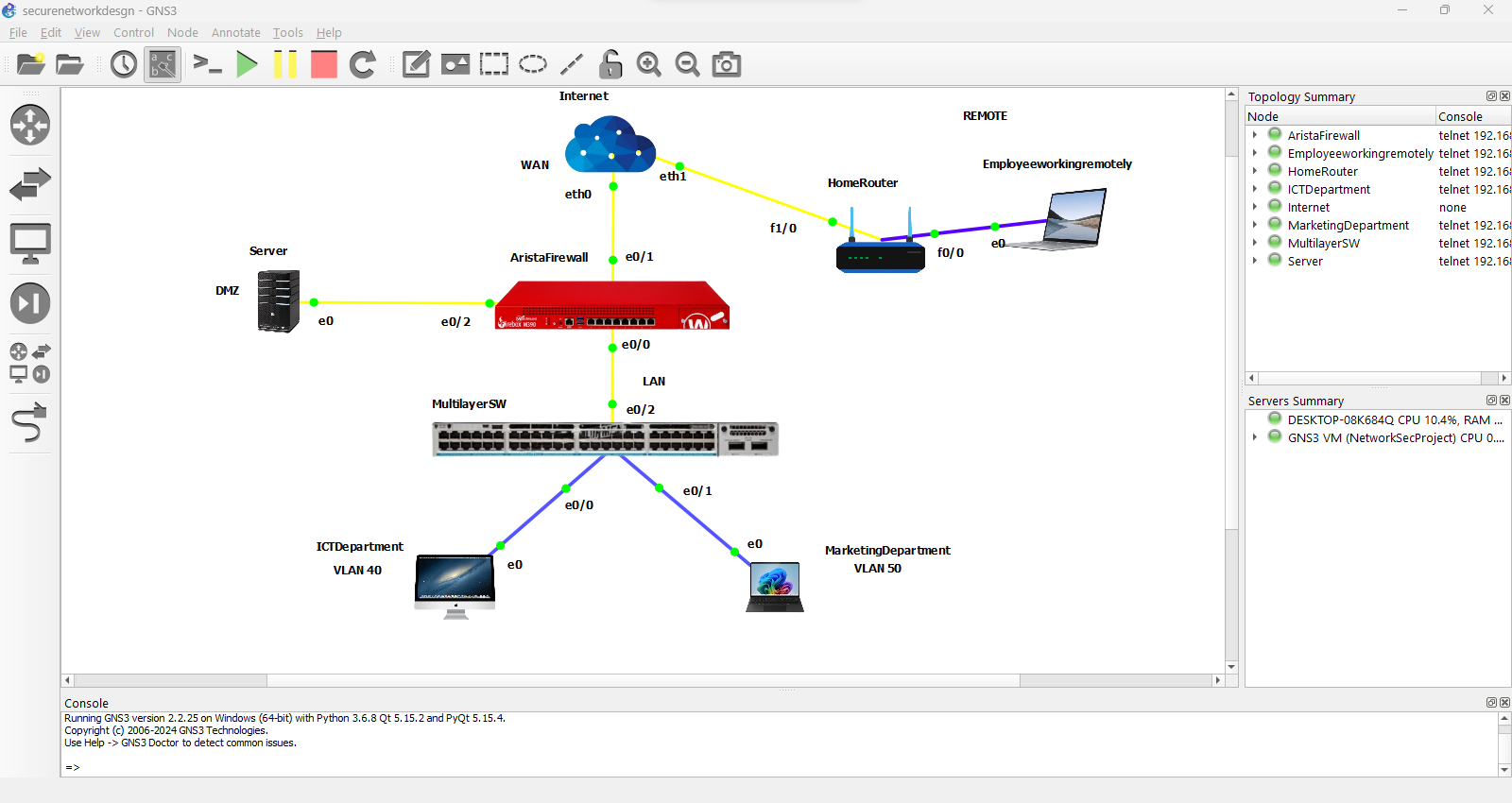
**Network Topology and Configuration**

***Network setup***

To do the emulation the network GNS3 emulation application was used . Some were devices was imported from the host machine and some from VMware Workstation. The WatchGuard Firewall, DMZ server (Ubuntu server), and penetration testing machine (Kali Linux) were executed on VMware Workstation; GNS3 server and remote machine were directly with the host system. The screenshot below depicts the network topology already set up in GNS3.

**Fig 1.1**

*Network Topology*

**

***Layer 3 Switch Configuration***

The Layer 3 switch was configured to support two VLANs:

*ICT Department (VLAN 40)*

*Marketing Floor Department (VLAN 50)*

The VLAN interfaces were enabled for VLAN communication between the VLANs. To manage the IP addresses dynamically a DHCP pool was created for each VLAN. The VLAN traffic was able to pass through the interfaced connecting the switch to the firewall. Below are the configurations executed on the Layer 3 switch:

***VLAN and interfaces configuration***

*MultilayerSW(config)# vlan 40*

*MultilayerSW(config-vlan)# name ICTDepartment*

*MultilayerSW(config-vlan)# exit*

*MultilayerSW(config)# vlan 50*

*MultilayerSW(config-vlan)# name MarketingDepartment*

*MultilayerSW(config-vlan)# exit*

*MultilayerSW(config)# interface e0/0*

*MultilayerSW(config-if)# switchport mode access*

*MultilayerSW(config-if)# switchport access vlan 40*

*MultilayerSW(config-if)# no shutdown*

*MultilayerSW(config-if)# exit*

*MultilayerSW(config)# interface e0/1*

*MultilayerSW(config-if)# switchport mode access*

*MultilayerSW(config-if)# switchport access vlan 50*

*MultilayerSW(config-if)# no shutdown*

*MultilayerSW(config-if)# exit*

*MultilayerSW(config)# interface e0/2*

*MultilayerSW(config-if)# switchport trunk encapsulation dot1q*

*MultilayerSW(config-if)# switchport mode trunk*

*MultilayerSW(config-if)# switchport trunk allowed vlan 40,50*

*MultilayerSW(config-if)# no shutdown*

*MultilayerSW(config-if)# exit*

*MultilayerSW(config)# interface vlan 40*

*MultilayerSW(config-if)# ip address 192.168.40.1 255.255.255.0*

*MultilayerSW(config-if)# no shutdown*

*MultilayerSW(config-if)# exit*

*MultilayerSW(config)# interface vlan 50*

*MultilayerSW(config-if)# ip address 192.168.50.1 255.255.255.0*

*MultilayerSW(config-if)# no shutdown*

*MultilayerSW(config-if)# exit*

*MultilayerSW(config)# ip routing*

*MultilayerSW(config)# ip dhcp pool vlan40*

*MultilayerSW(dhcp-config)# network 192.168.40.0 255.255.255.0*

*MultilayerSW(dhcp-config)# default-router 192.168.40.1*

*MultilayerSW(config)# ip dhcp excluded-address 192.168.40.1 192.168.40.40*

*MultilayerSW(config)# ip dhcp pool vlan50*

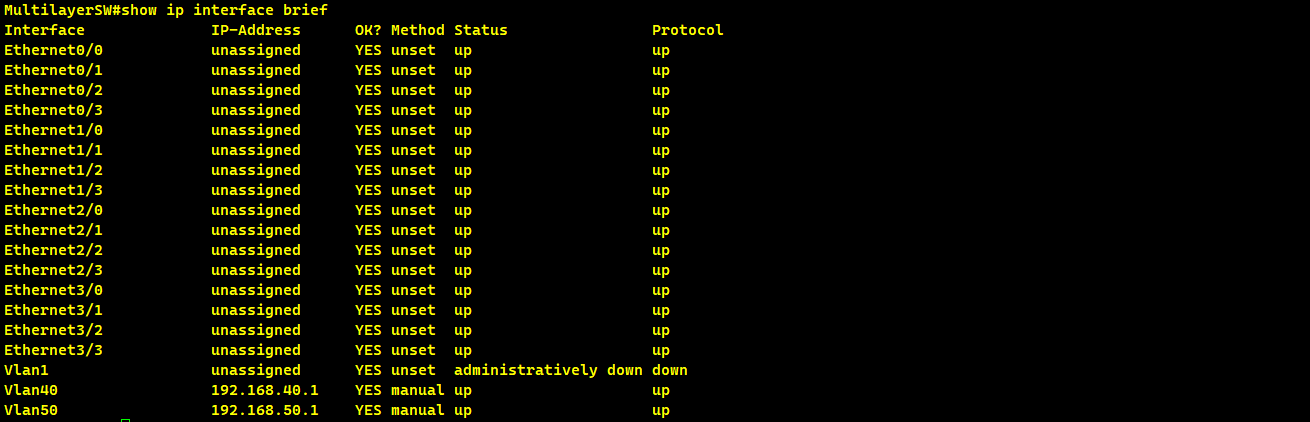
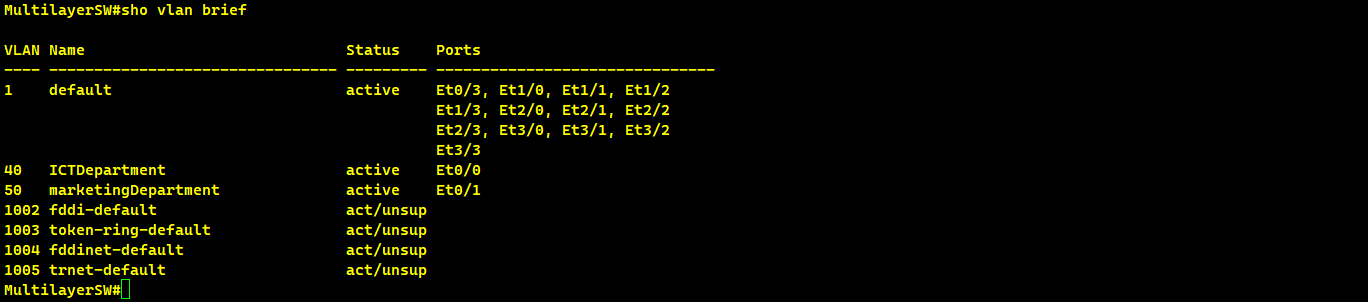
*MultilayerSW(dhcp-config)# network 192.168.50.0 255.255.255.0*

*MultilayerSW(dhcp-config)# default-router 192.168.50.1*

*MultilayerSW(config)# ip dhcp excluded-address 192.168.50.1 192.168.50.50*

**Fig 1.2**

*Layer 3 configurations*



**Firewall Setup and Configuration**

Three interfaces were configured on the WatchGuard Firewall:

* WAN: Configured to use DHCP.
* LAN: Specifically designed to operate as a bridge interface.
* DMZ: Configured with a static IP.

**Firewall Policies and VPN Configuration**

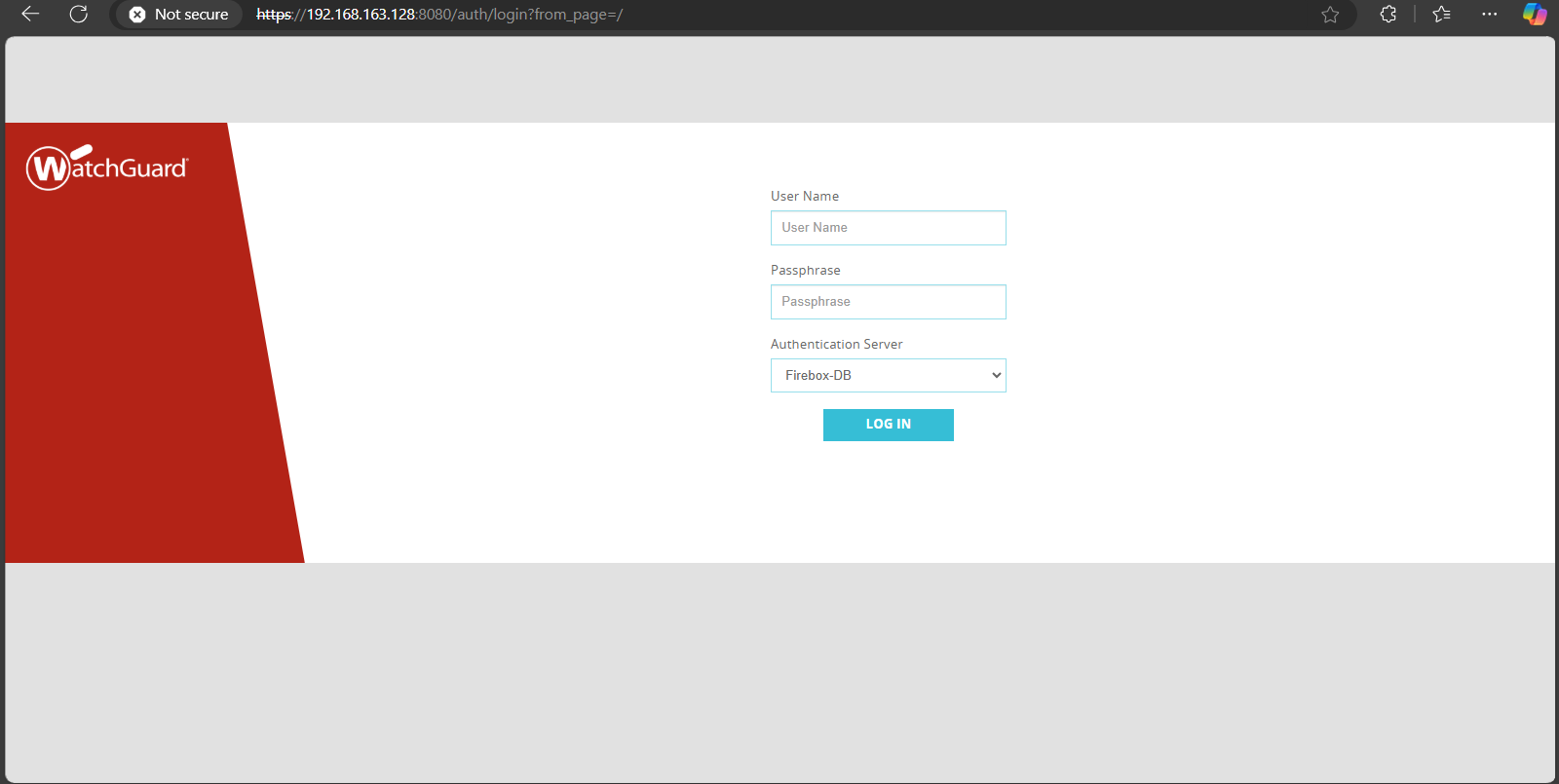
There were also policies enabled to permit only certain traffic between VLANs, HTTP, HTTPS, SMTP, POP3, ICMP for DMZ and traffic to the SSL VPN for LAN.

AES and SHA256 was used for encryption and authentication for SSL VPN for the remote use. This made it possible to manage data access using the identified user groups while the users were authenticated from the firewall level.

The client for the SSL VPN was obtained from the portal of the firewall and then run on the remote machine. Secure access certificate was also installed. The confidentiality of traffic was confirmed by using wireshark protocol.

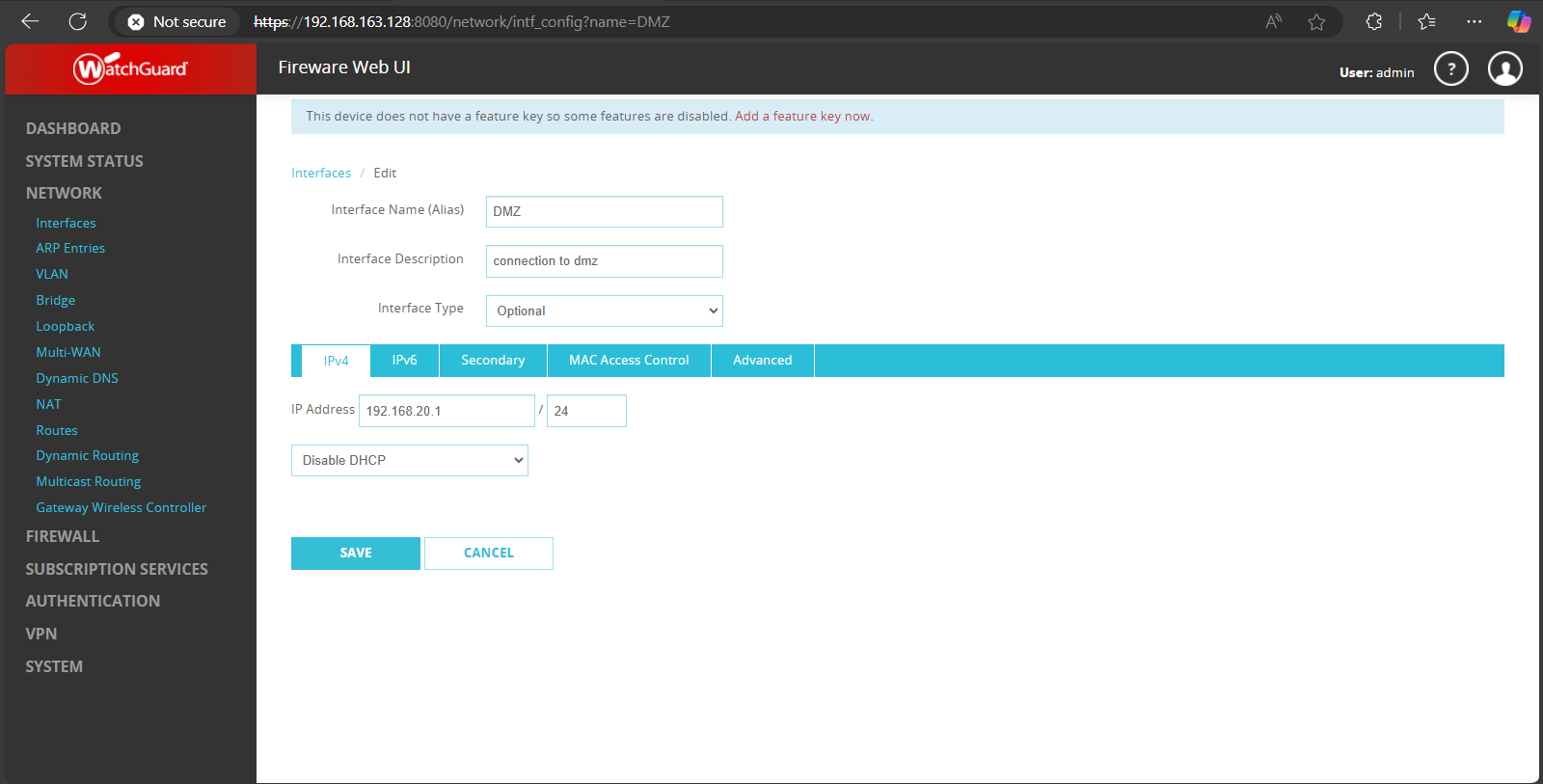
**Fig 1.3**

*Watchguard Web UI*



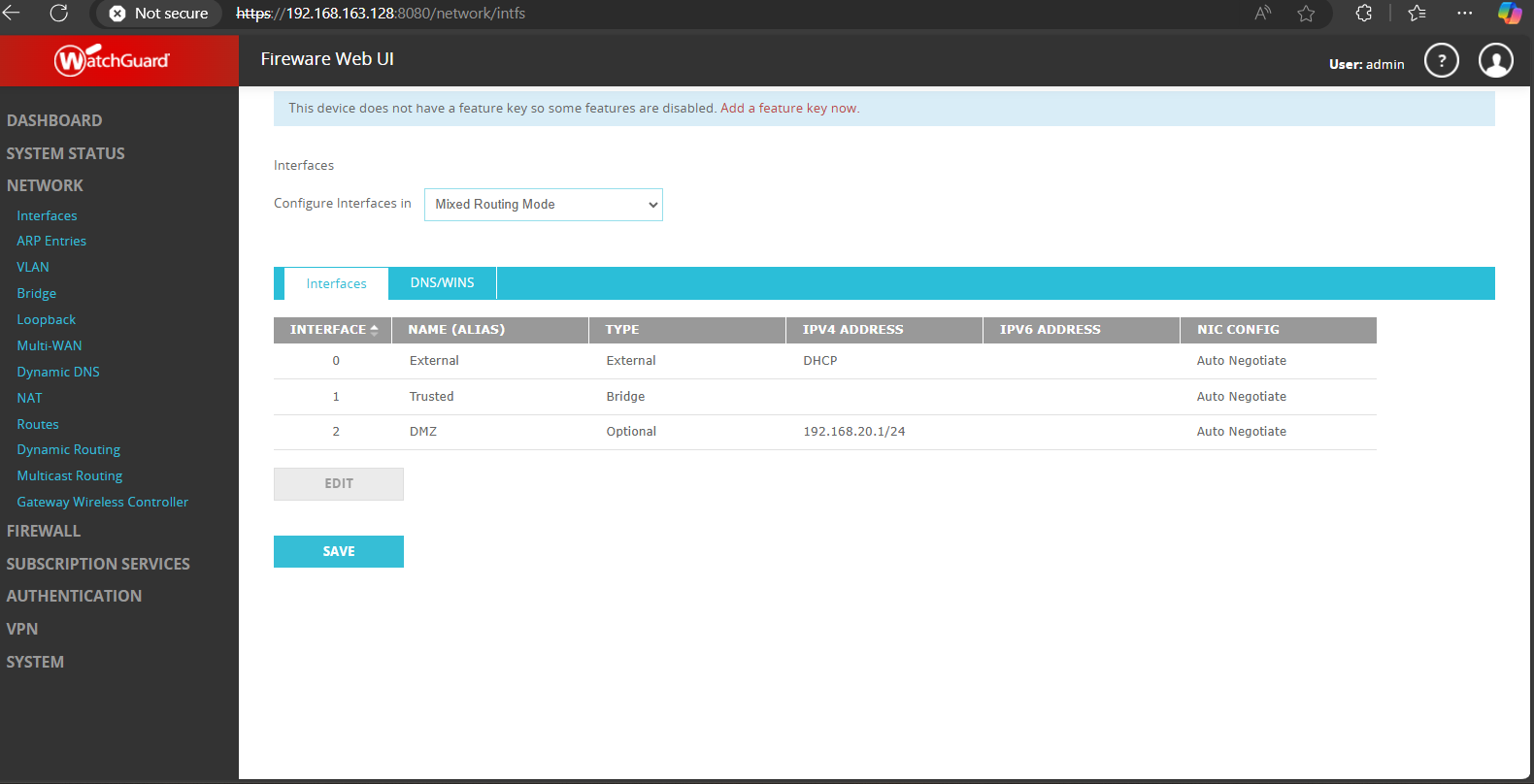
**Fig 1.4**

*DMZ interface configuration*



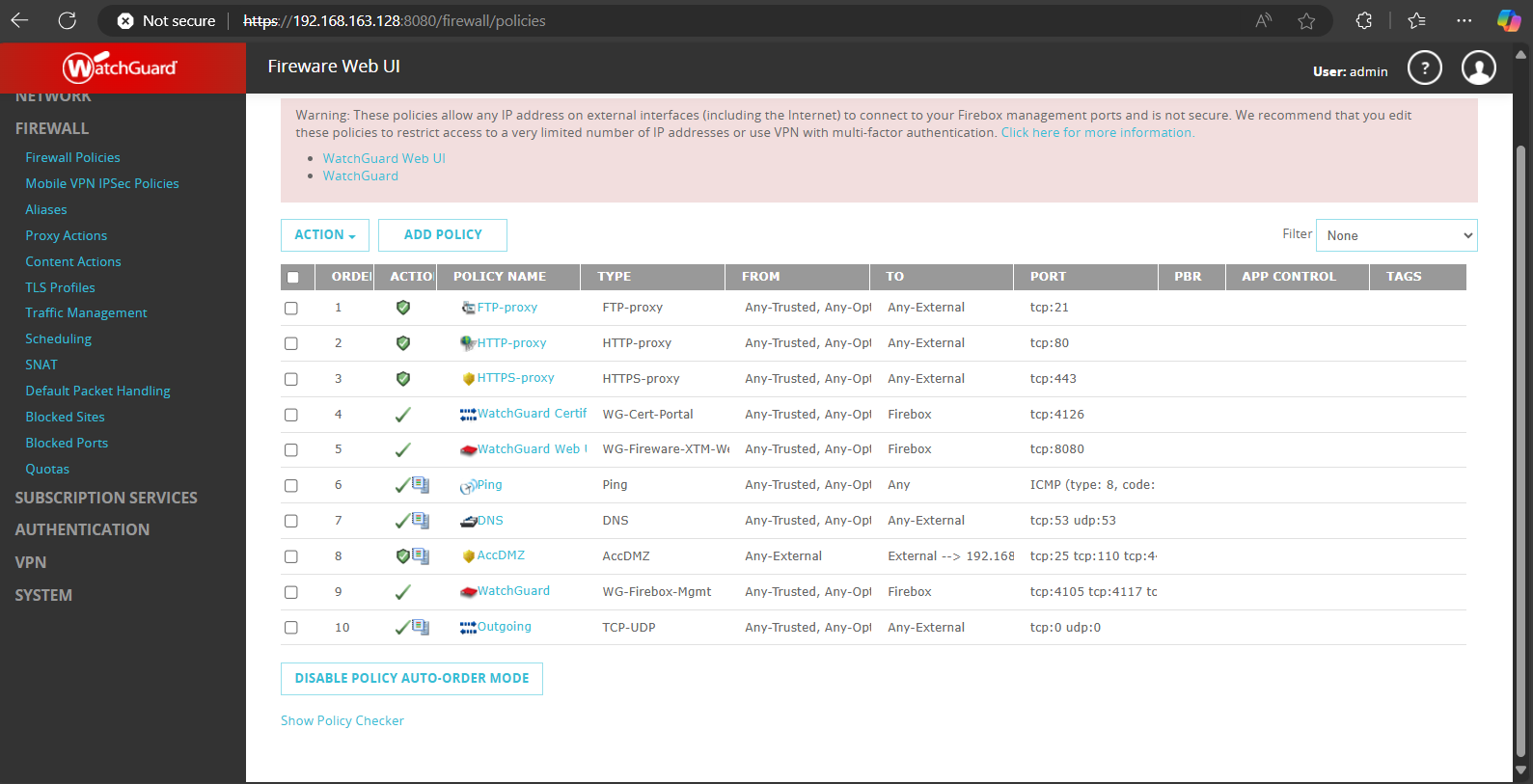
**Fig 1.5**

*LAN,WAN and DMZ interface configured*



**Fig 1.6**

*Configured firewall rules/policies*

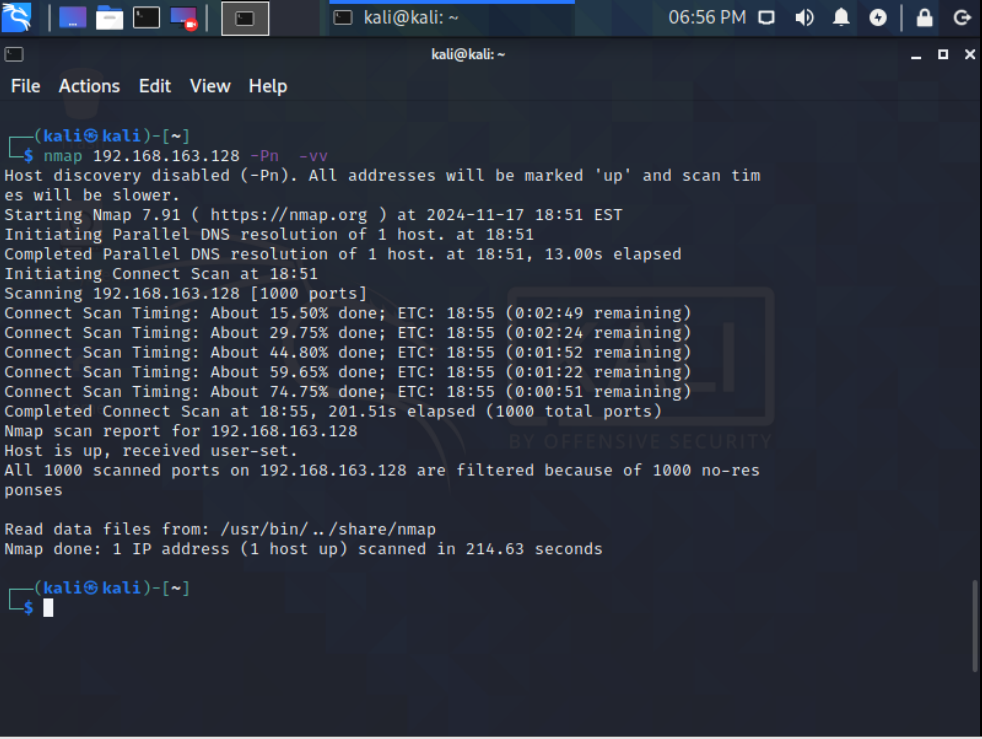


**Firewall testing**

To test the configured policies and the exposed ports to the internet we used Nmap to scan the internet facing interface / gateway.The policies looked good with only the necessary exposed port and in filtered state.The firewall was also able to drop/deny any traffic violating the policies.

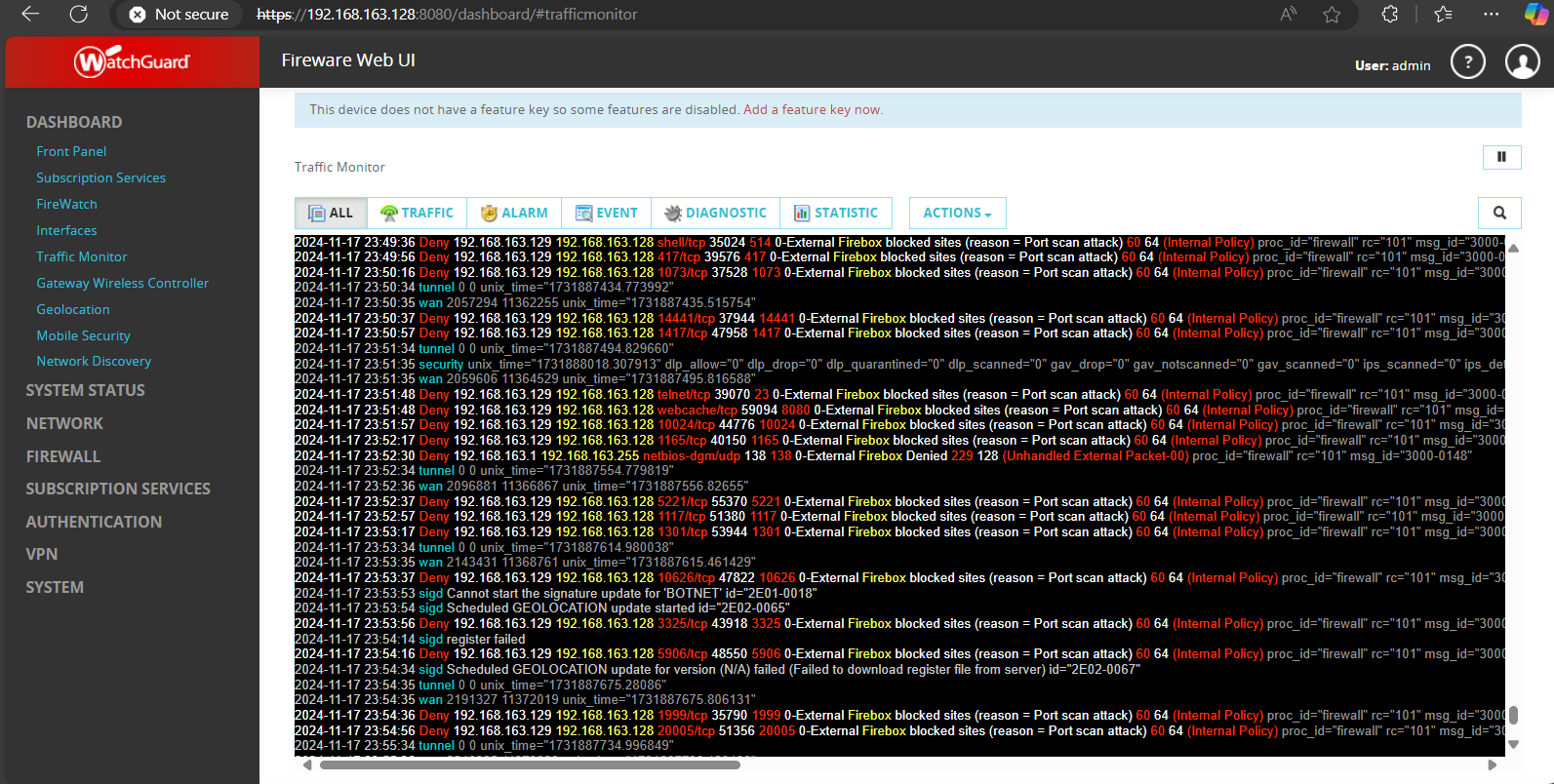
**Fig 1.7**

*Firewall policies testing with Nmap*



**Fig 1.8**

*Firewall policies blocking nmap scans*



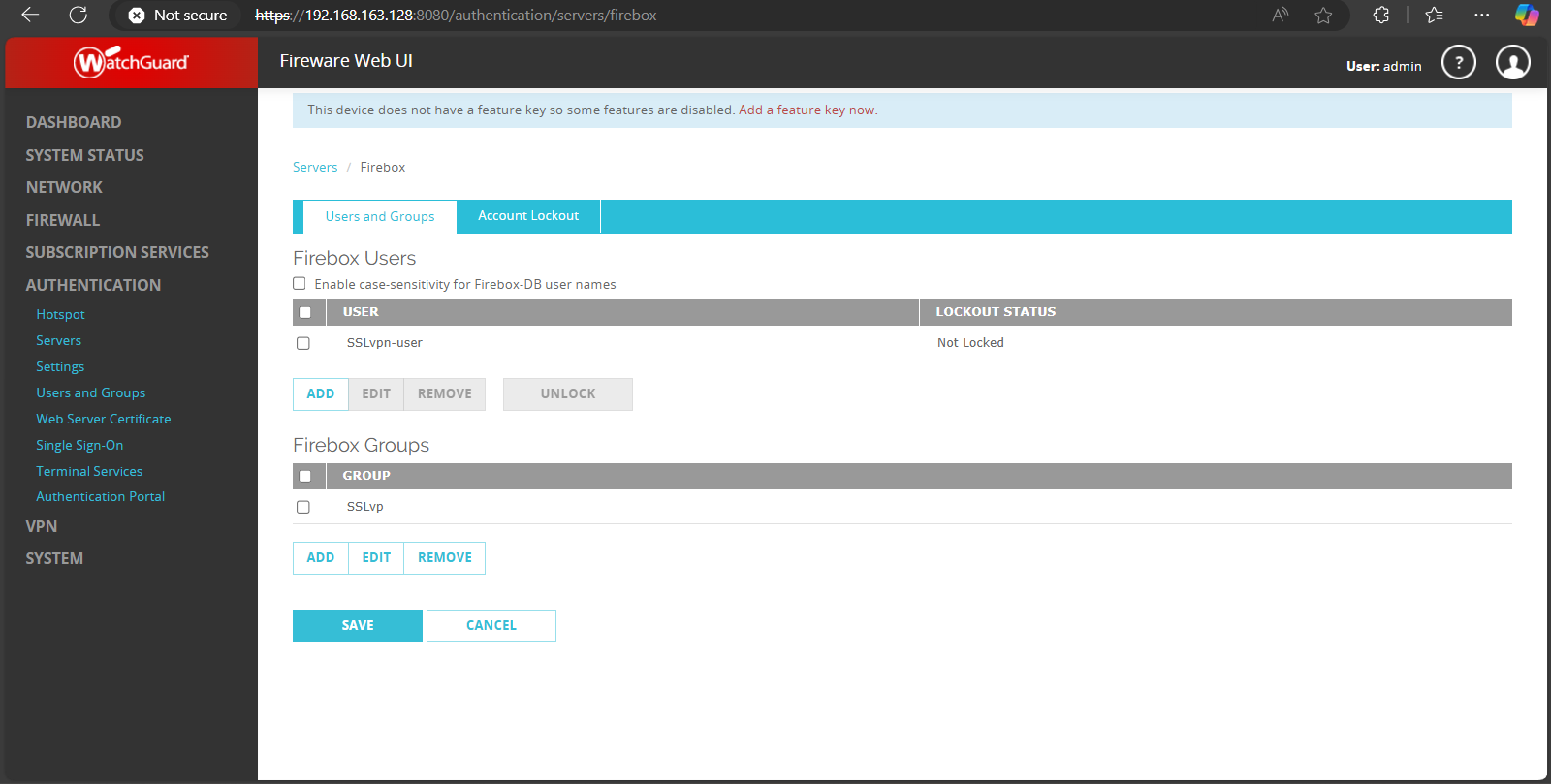
**VPN Configuration**

We configures SSL vpn to allowa internal resource/service access by the remote machine.AES and SHA256 was used for encryption and authentication for SSL VPN for the remote use. This made it possible to manage data access using the identified user groups while the users were authenticated from the firewall level.

The client for the SSL VPN was obtained from the portal of the firewall and then run on the remote machine. Secure access certificate was also installed. The confidentiality of traffic was confirmed by using wireshark protocol.

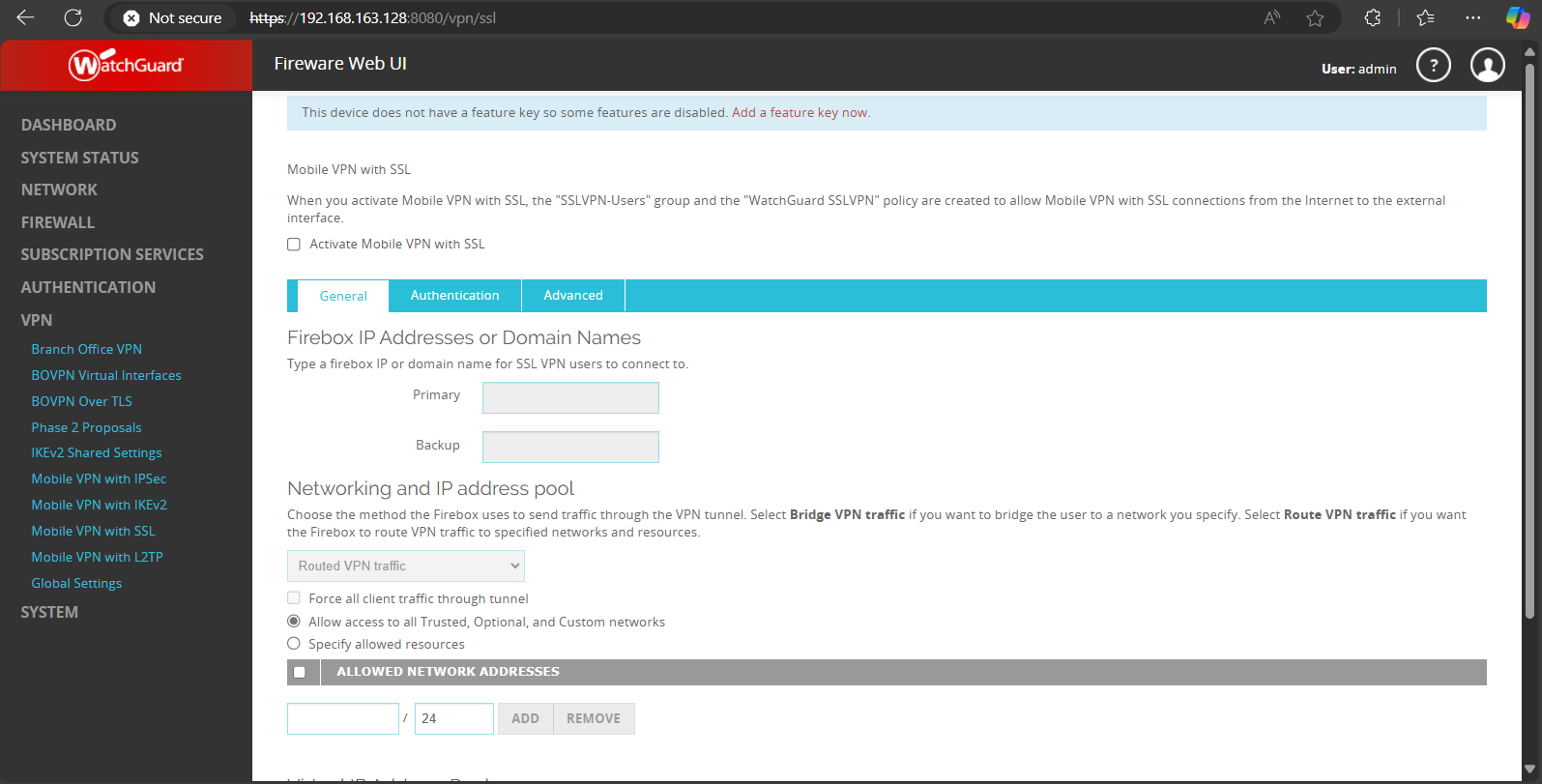
**Fig 1.9**

*SSL vpn user and group configuration*



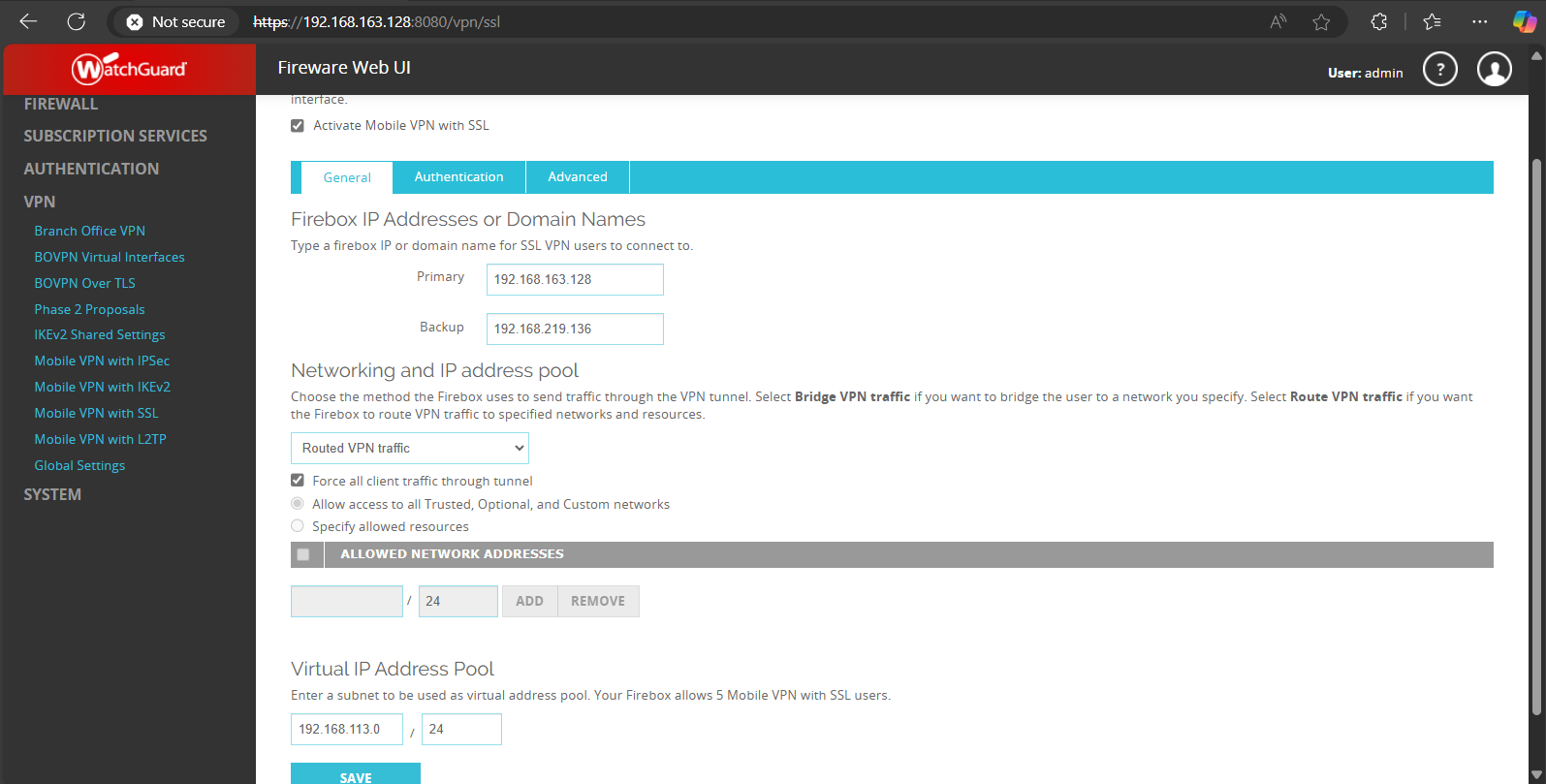
**Fig 2.0**

*SSL vpn configuration*



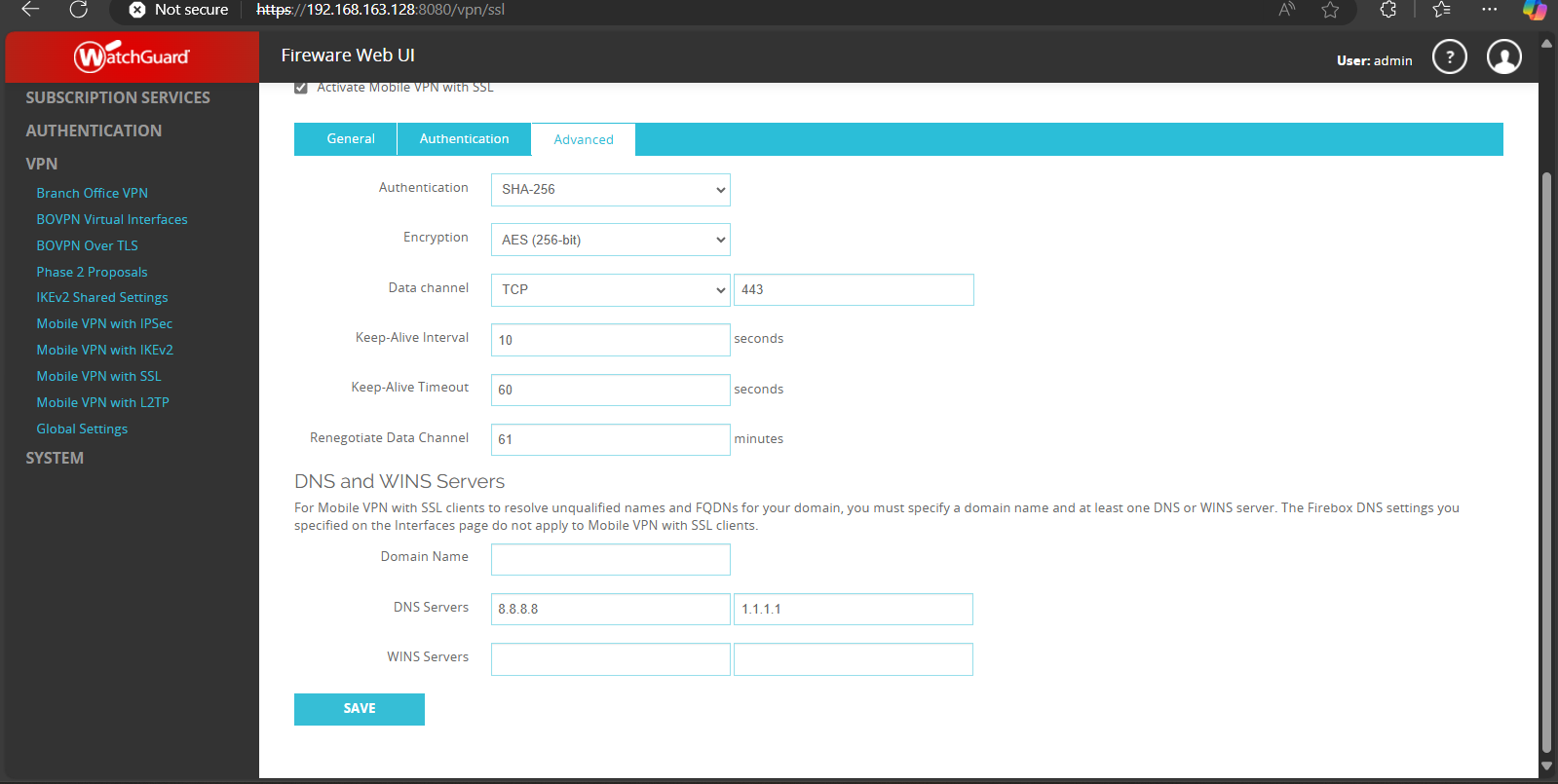
**Fig 2.1**

*Configured firewall rules/policies*



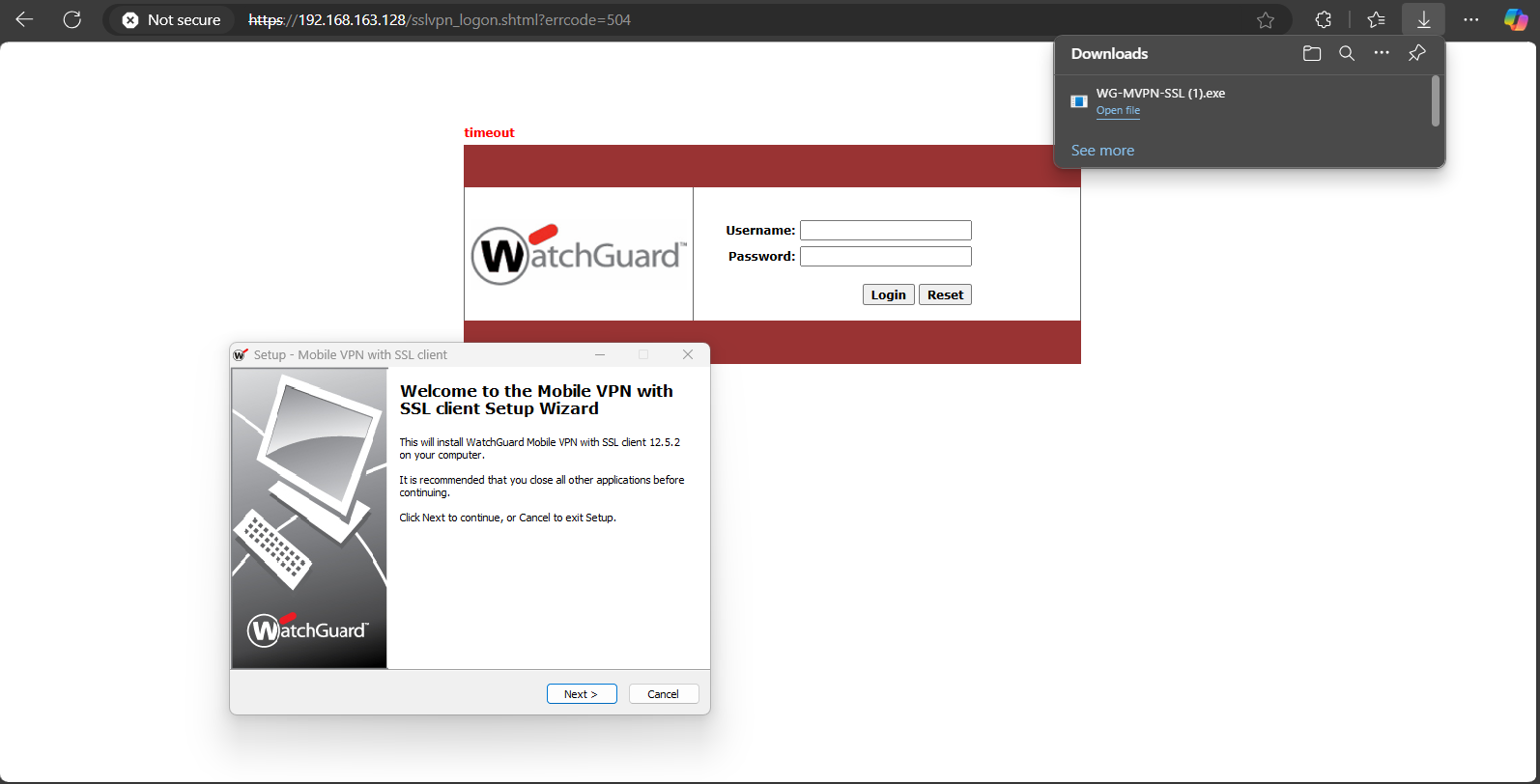
**Fig 2.2**

*SSL vpn authentication and encryption configuration*



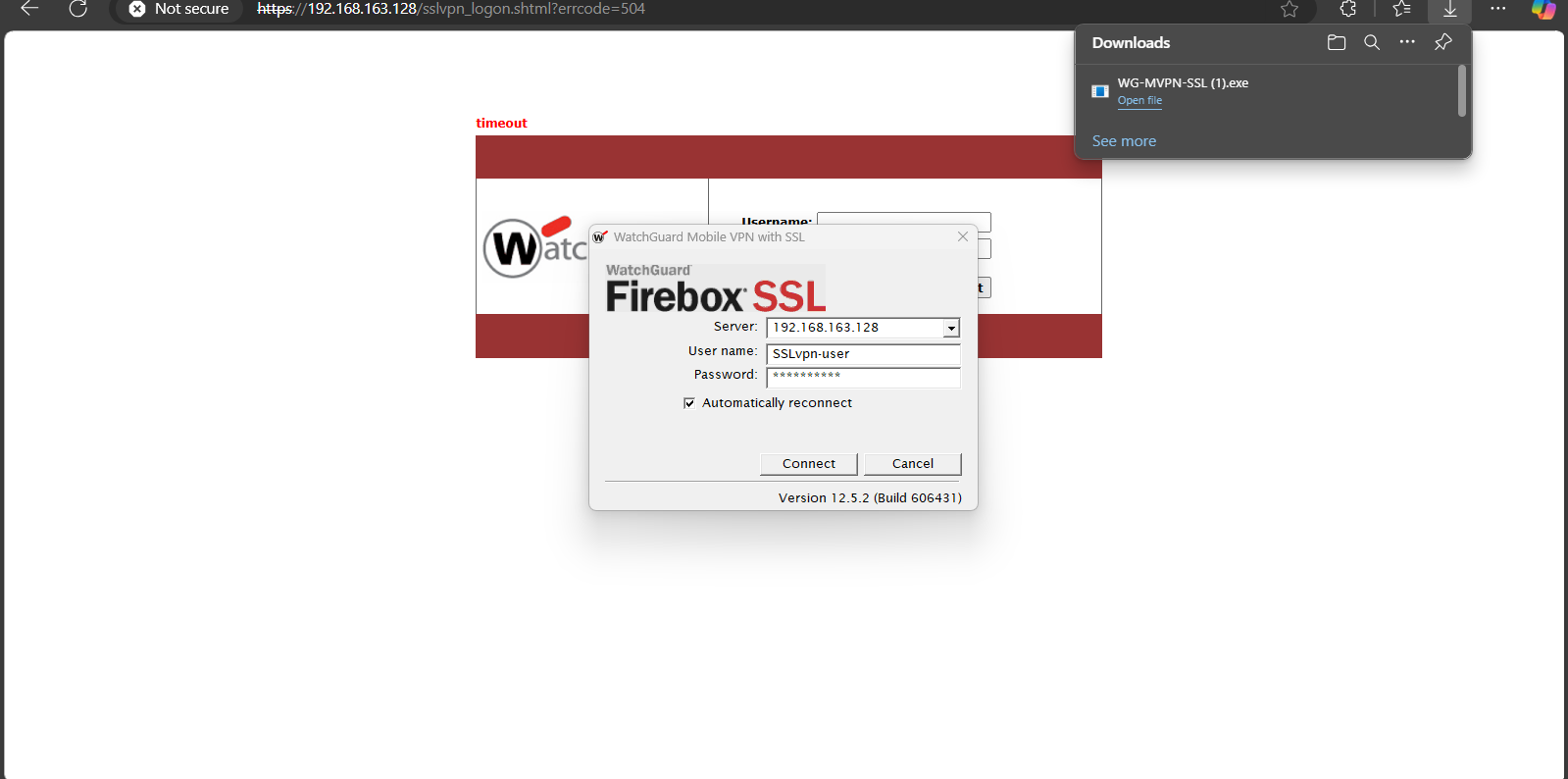
**Fig 2.3**

*SSL vpn user portal*



**Fig 2.4**

*SSL vpn connection establishment*

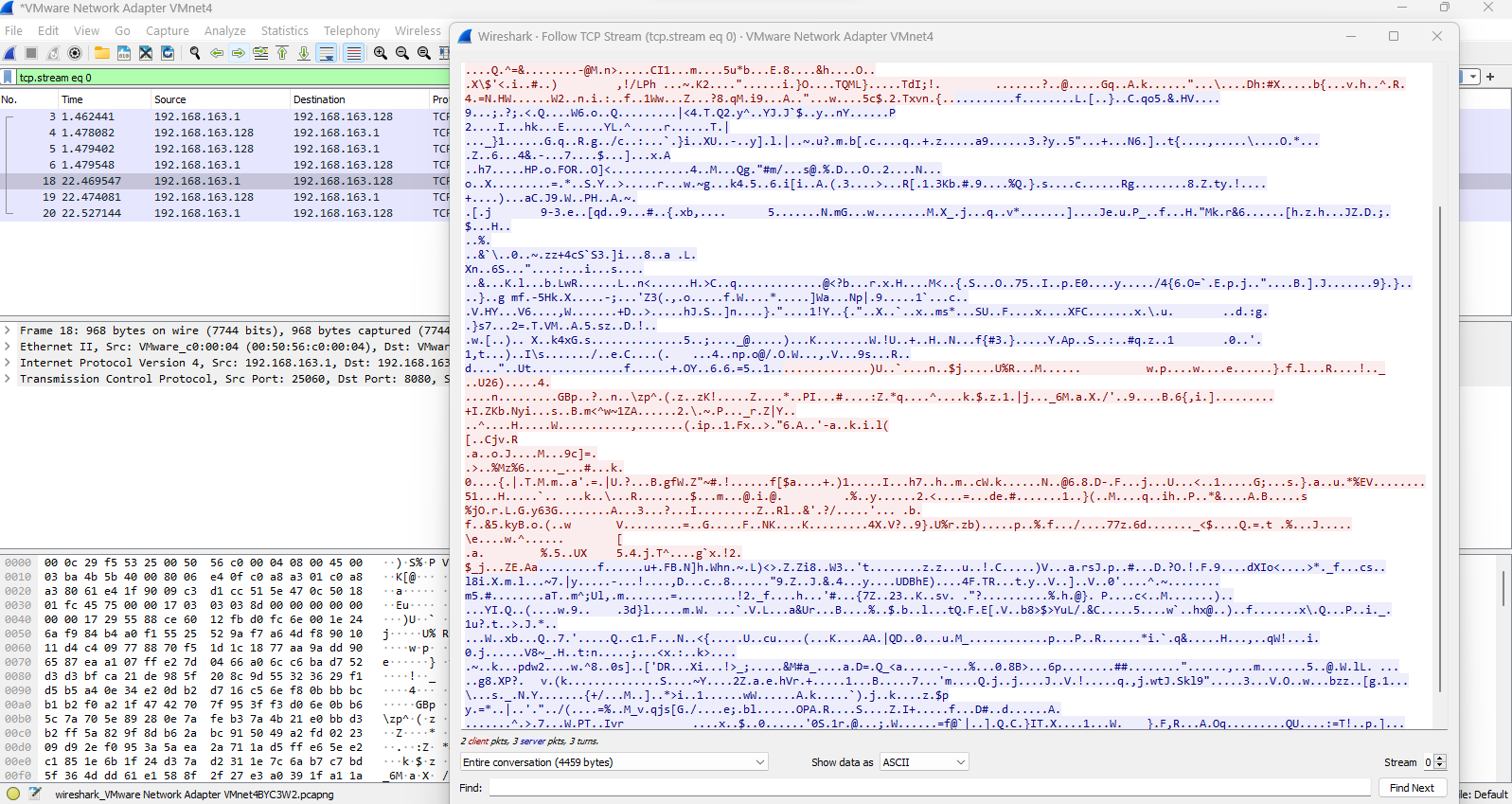


**Testing SSL vpn encryption**

To test the remote access vpn we monitored the specific vpn traffic with wireshark to verify the encryption which showed that the traffic was encrypted .

**Fig 2.5**

*SSLvpn encryption testing*



**IDS Implementation**

A Linux Ubuntu based DMZ server was deployed on virtual machines with content IDS system called Fail2ban IDS. A custom SSH intrusion rule was configured as follows:

*[sshd]*

*enabled = true*

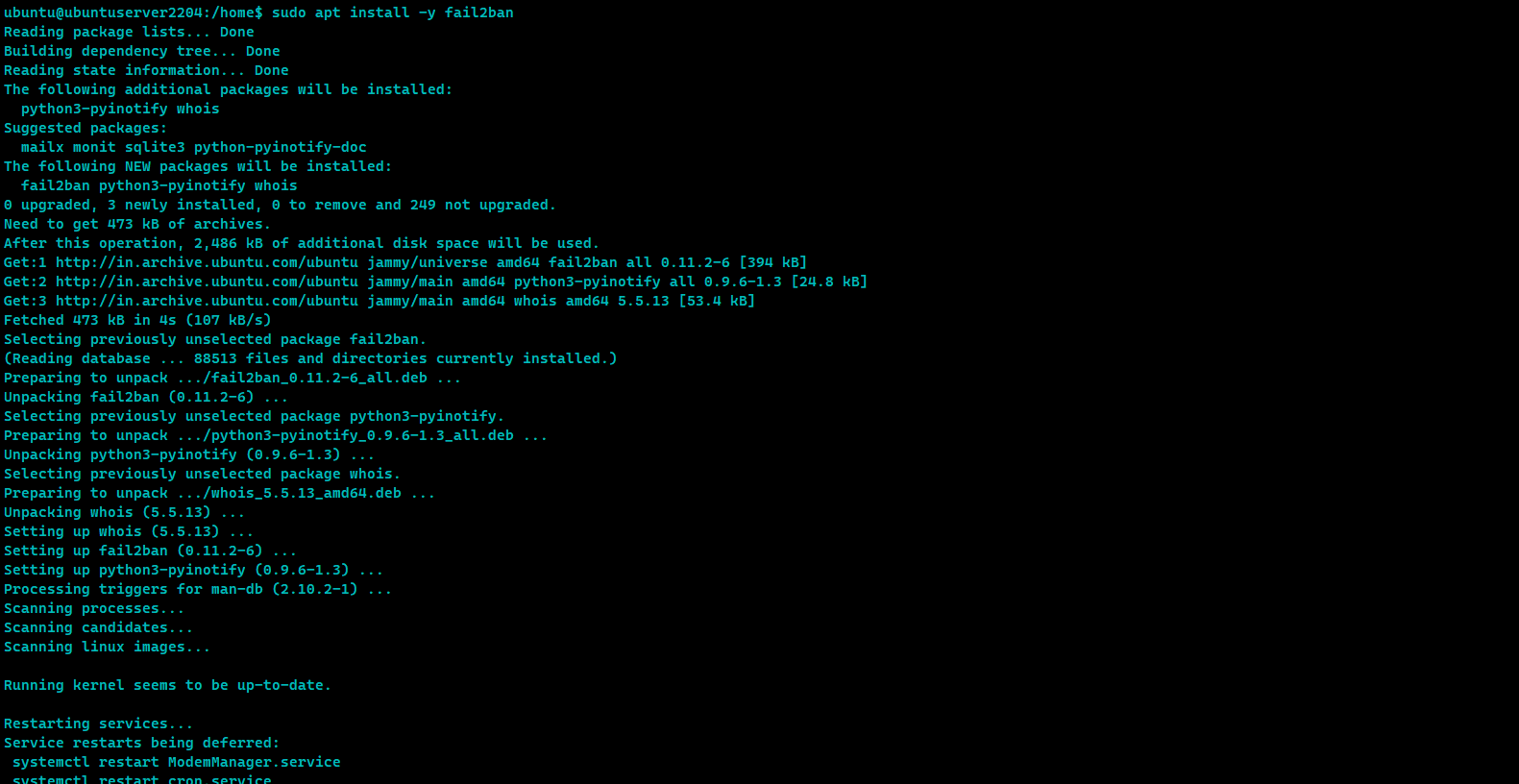
*port = ssh*

*logpath = /var/log/auth.log*

*bantime = 3600*

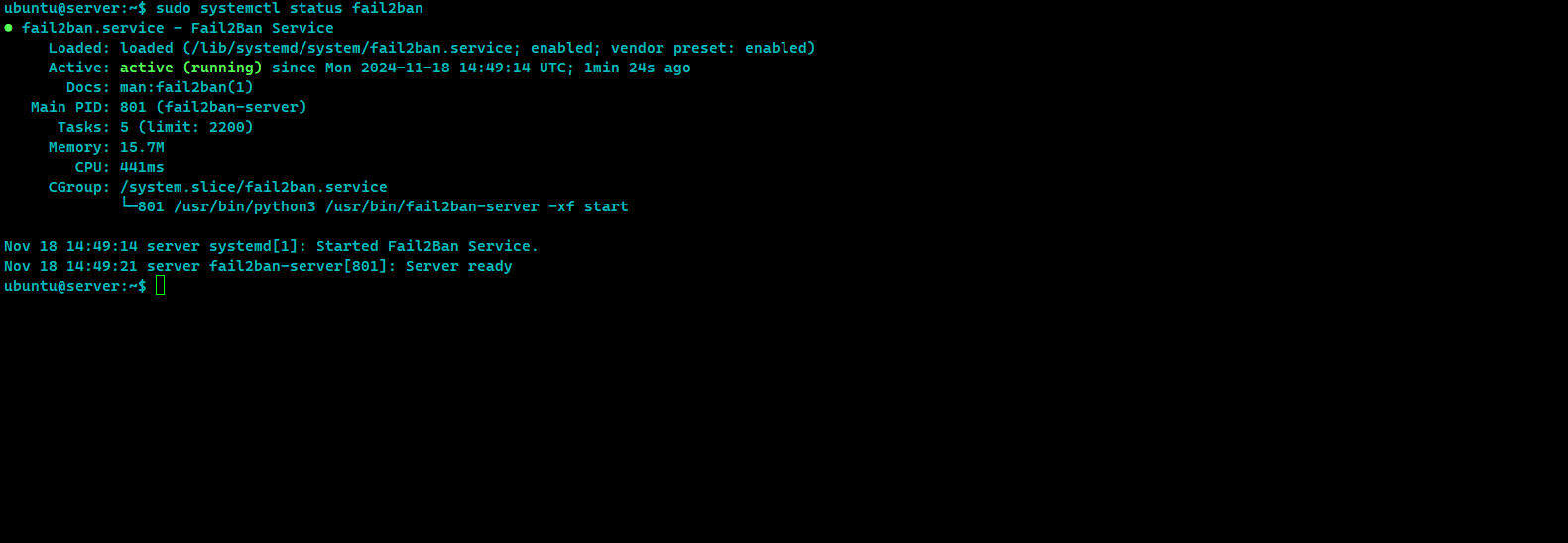
**Fig 2.6**

*Installing Fail2ban IDS*

**

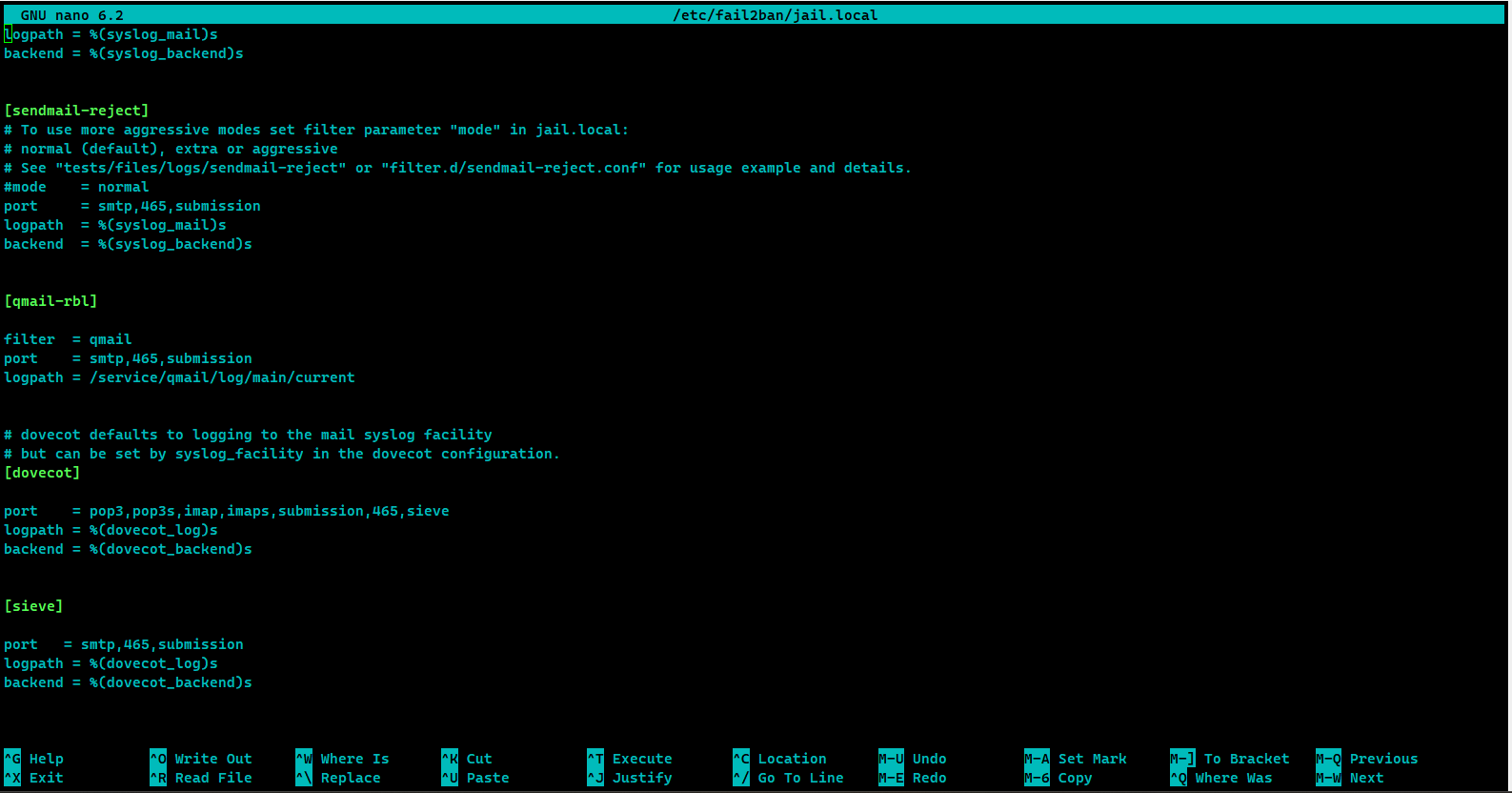
**Fig 2.7**

*IDS status*

**

**Fig 2.8**

*IDS configuration and adding custom rule*

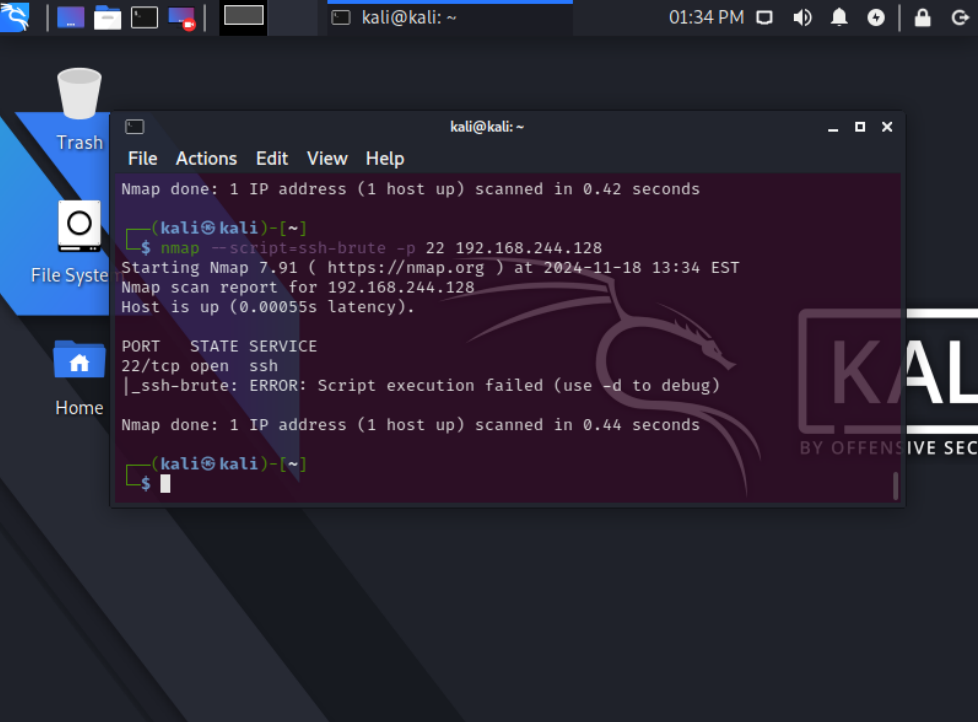
**

**IDS testing.**

To verify that the IDS can detect anomalous traffic we simulated ssh bruteforce attack to ward the server and Fail2ban successfully detected the traffic.

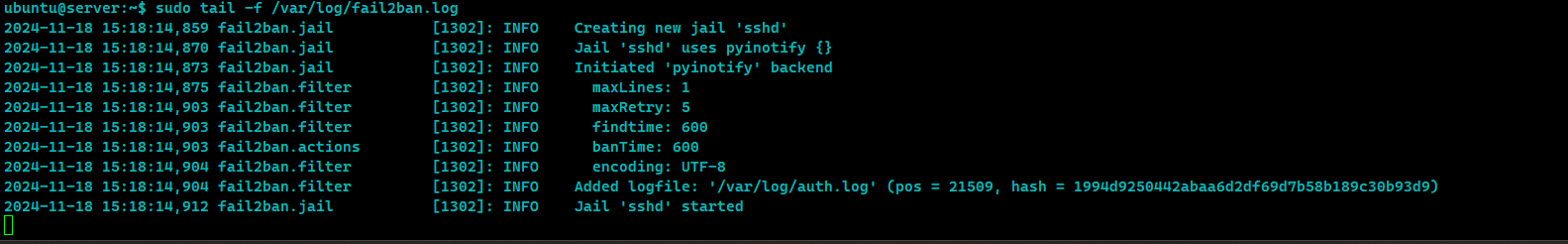
**Fig 2.9**

*Testing IDS with simulated attack*

**

**Fig 3.0**

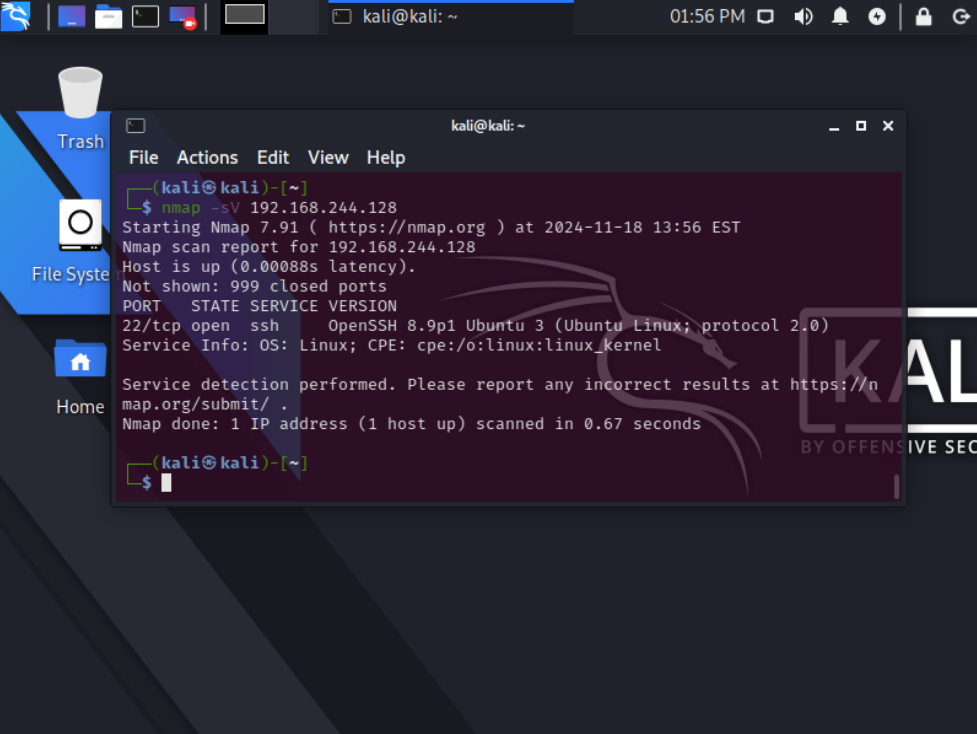
*IDS testing: log shows ssh bruteforce blocked*

**

**Testing and Security Assessment**

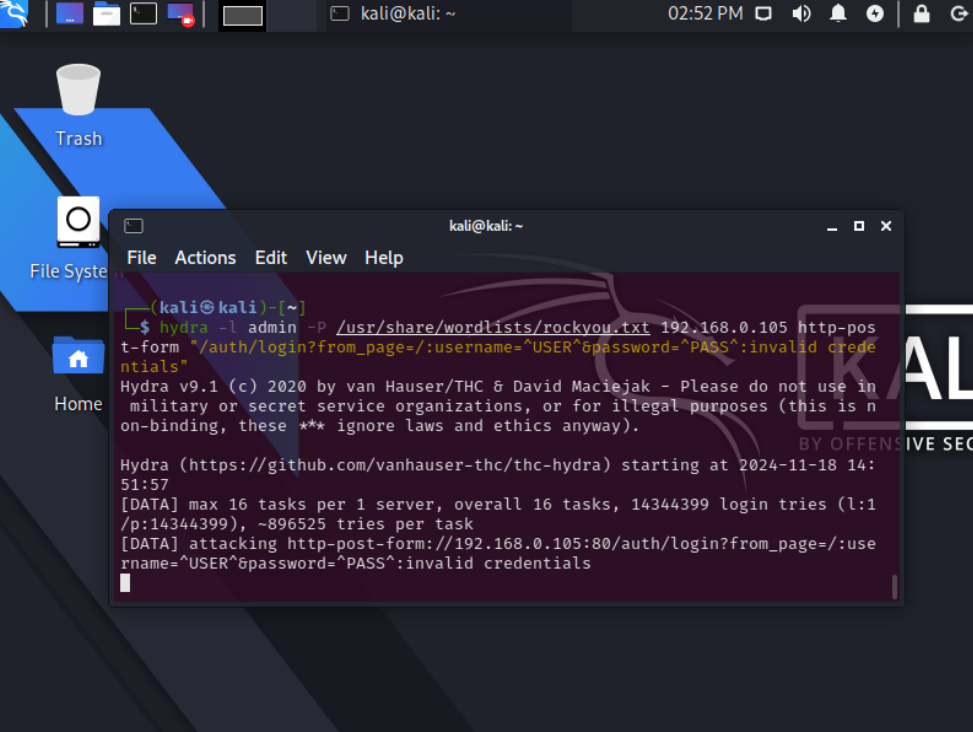
To test and verify the implemented security we used nmap to footprint the network and tried bruteforce attack to the firewall web UI which actually had positive results meaning it even went to an extent of the set policies recognizing the malicious traffic and dropping the attacking machine traffic.We used Nmap and hydra in this two cases.One of the weakness identified was that port 22 was exposed to the internet which can be a risk of being exploited by a zero day.

**Fig 3.1**

*IDS testing: log shows ssh bruteforce blocked*

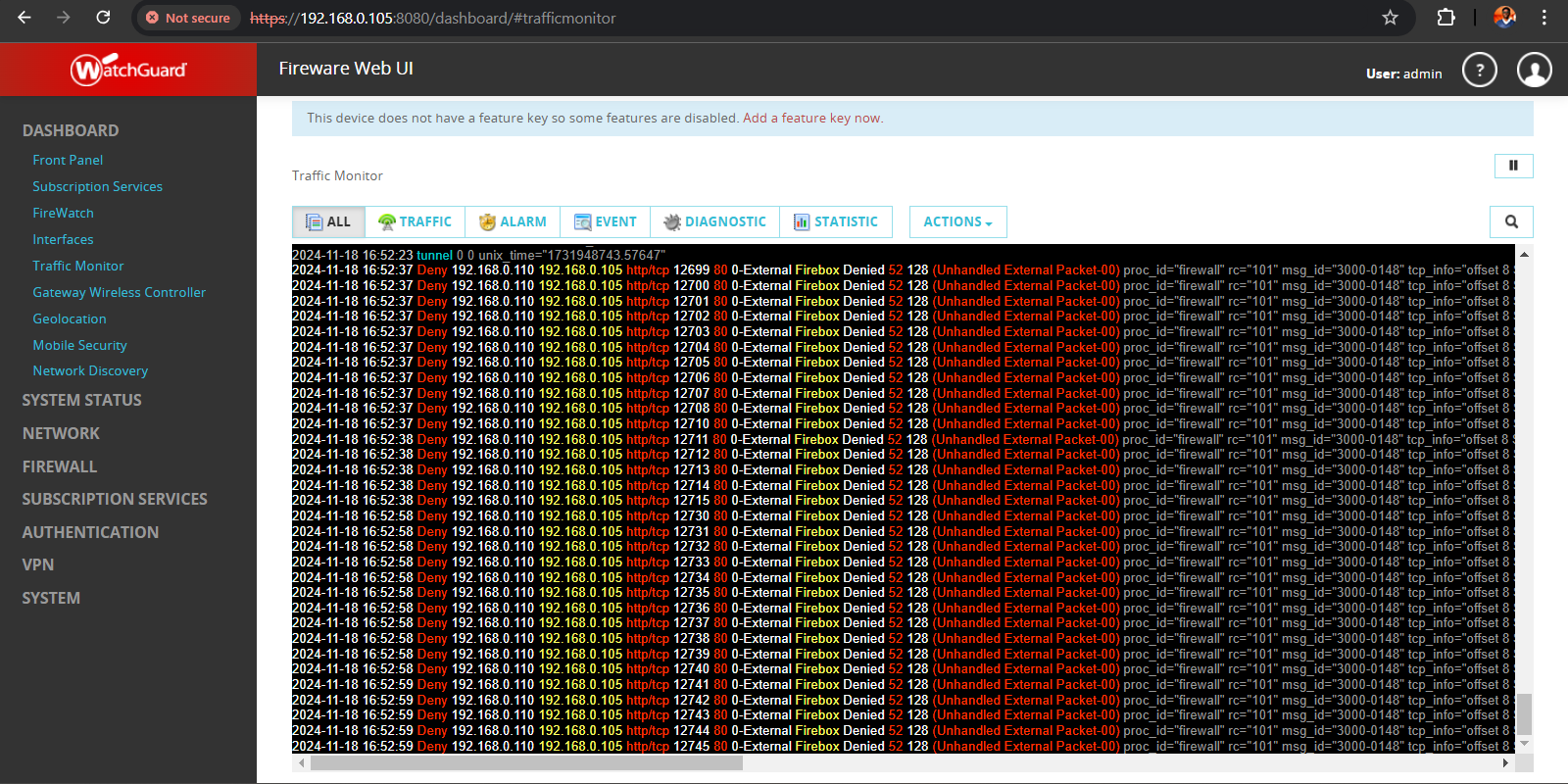
**Fig 3.2**

*Attack vector II: firewall web UI login bruteforce.*



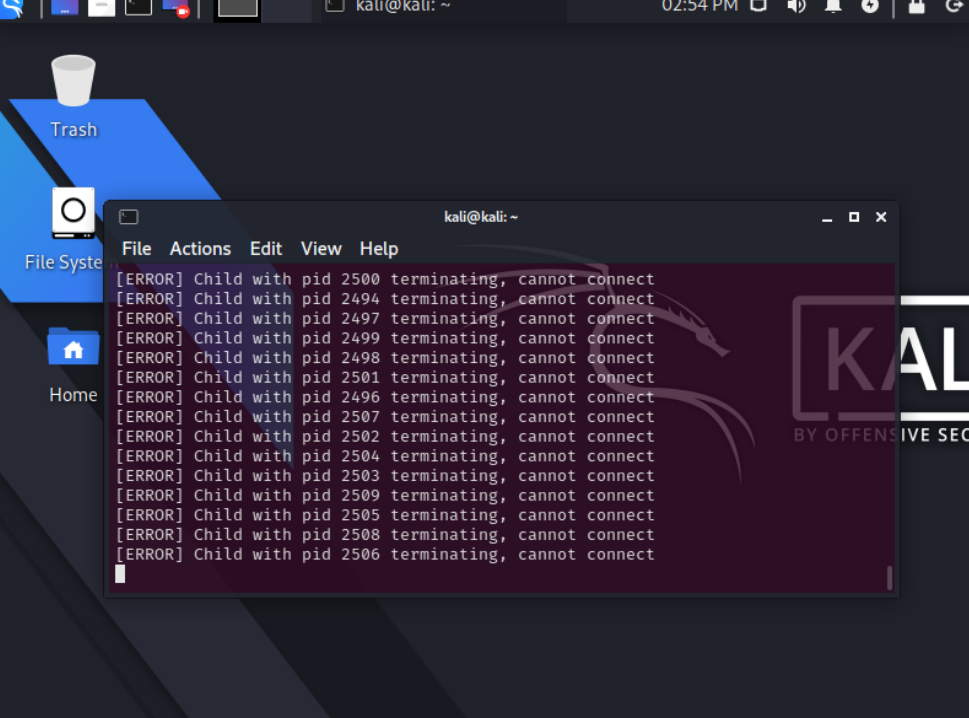
**Fig 3.3**

*Firewall blocked hydra bruteforce probes*



**Fig 3.4**

*Firewallweb ui login bruteforce attack failed.*



**Risk Assessment**

The risks identified during the vulnerability assessment and exploitation in the network include expose of unsued ports like ssh.

**Mitigation**

Exposing only required/used ports and with different ports not well known ports