Titre: Title:	A Secure Next-Generation Firewall Architecture Against DDoS Attacks
Auteur: Author:	Sadaf Pourmand
Date:	2025
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Pourmand, S. (2025). A Secure Next-Generation Firewall Architecture Against DDoS Attacks [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/64977/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/64977/
Directeurs de recherche: Advisors:	Samuel Pierre
Programme: Program:	Génie informatique

POLYTECHNIQUE MONTRÉAL

Affiliée à l'Université de Montréal

A Secure Next-Generation Firewall Architecture Against DDoS Attacks

SADAF POURMAND

Département de génie informatique et génie logiciel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise es sciences appliquées*

Génie informatique et génie logiciel

Avril 2025

POLYTECHNIQUE MONTRÉAL

Affiliée à l'Université de Montréal

Ce mémoire intitulé:

A Secure Next-Generation Firewall Architecture Against DDoS Attacks

Présenté par **Sadaf POURMAND**

en vue de l'obtention du diplôme de *Maîtrise es science appliquées*

a été dûment accepté par le jury d'examen constitué de:

Alejandro QUINTERO, président

Samuel PIERRE, membre et directeur de recherche

Martine BELLAÏCHE, membre

DEDICATION

This dissertation is dedicated to my beloved parents, my mother Manijeh Salimi and my father Bahram Pourmand, whose unwavering support, guidance, and encouragement have been the foundation of my academic journey. Their sacrifices and love have been my greatest source of motivation, and for that, I am forever grateful.

I also dedicate this work to my husband, Siavash Mirzaei, whose strength, patience, and constant encouragement have been invaluable throughout this challenging process. Your belief in me has made all the difference.

To you all, I extend my deepest appreciation and eternal gratitude. This accomplishment is as much yours as it is mine.

ACKNOWLEDGEMENTS

I would like to extend my sincerest gratitude to my research director, Professor Samuel Pierre, professor at Polytechnique Montreal and director of LARIM, whose exceptional mentorship, thoughtful guidance, and steadfast support have been invaluable throughout my research journey. His insightful expertise, combined with his unshakable commitment to my academic progress, has made this path both rewarding and intellectually stimulating. I consider myself fortunate to have been under his supervision during this pivotal phase of my education.

I am equally indebted to the members of my dissertation committee, for their time, effort, and dedication in reviewing my work. Their constructive feedback and suggestions have significantly enriched this dissertation, and I am truly grateful for their thoughtful contributions.

I would like to offer special thanks to my co-supervisor, Dr. Franjieh El Khoury, LARIM's research associate and coordinator, whose exceptional guidance and patient support throughout this process have been beyond measure. Her expertise and encouragement provided me with both clarity and confidence during the most challenging stages of this research. Her role in shaping this dissertation, through insightful advice and rigorous evaluation, cannot be overstated.

Additionally, I am deeply thankful to my colleagues and collaborators at LARIM Laboratory, whose contributions and collective effort greatly supported my progress. Their camaraderie and willingness to assist have been instrumental in creating a collaborative environment that facilitated the successful completion of this research.

On a more personal note, I would like to express my heartfelt appreciation to my family, whose persistent belief in me has been the bedrock of my achievements. To my parents, who provided boundless love and support, I owe a deep debt of gratitude. Their encouragement has been a constant source of strength, and without their trust in my abilities, none of this would have been possible.

Lastly, I am deeply grateful to my friend, Fatemeh, whose encouragement, patience, and companionship have been indispensable throughout this journey. Her unyielding support helped me remain focused and motivated, and I will always cherish her role in making this achievement a reality. To all those who have guided, supported, and encouraged me, I extend my most heartfelt thanks.

RÉSUMÉ

Dans un monde de systèmes et de technologies qui évolue rapidement, la sécurité des réseaux n'est pas seulement une priorité, c'est une nécessité absolue. Les infrastructures de réseau et les systèmes d'information doivent continuer à se concentrer sur la manière de les sécuriser contre les cyberattaques. Le nombre de cybercriminels a considérablement augmenté au cours des dernières années, et il est de la plus haute importance de renforcer les mesures de cybersécurité et d'accroître la sensibilisation afin de protéger les actifs sensibles contre les menaces potentielles.

La demande indéniable de solutions de sécurité efficaces ne cesse de croître, car les cyberattaques se sont généralisées, affectant les systèmes et les réseaux de divers secteurs d'activité. Cette tendance souligne l'urgence de mettre en œuvre des méthodes et des stratégies avancées pour contrer le nombre croissant et la sophistication des cyberattaques, qui constituent une menace sérieuse pour les individus et les organisations à l'échelle mondiale.

L'objectif principal des mesures de sécurité actuelles est d'anticiper et de gérer les cybermenaces courantes, en particulier les attaques par déni de service “Denial of Service” (DoS) et par déni de service distribué “Distributed Denial of Service” (DDoS). Il s'agit de tirer parti de scénarios prédictifs et d'outils de sécurité avancés pour identifier les vulnérabilités et prévoir les schémas d'attaque. Lorsqu'une menace est détectée, des mécanismes de défense automatisés et en temps réel sont immédiatement mis en œuvre pour atténuer l'impact de l'attaque.

Cette recherche vise à développer une stratégie globale de cybersécurité contre les attaques DDoS. Elle intègre l'utilisation d'un pare-feu de nouvelle generation “Next-Generation Firewall” (NGFW), du trafic “Voice over Internet Protocol” (VoIP), de la gestion des informations et des événements de sécurité “Security Information and Event Management” (SIEM), d'un serveur Web, de Kali Linux, et de commutateurs et routeurs de couche 2 sur trois couches distinctes du réseau : l'accès, la zone démilitarisée “Demilitarized Zone” (DMZ) et le cœur. En combinant ces technologies, le mécanisme assure une protection solide et une atténuation efficace des menaces à chaque couche du réseau.

Dans un premier temps, nous configurons une topologie de réseau sur la plateforme EVE-NG et sur VMware Workstation. La configuration et la politique du pare-feu de nouvelle génération sont essentielles pour la détection en temps réel et les stratégies d'atténuation des attaques DDoS, en

établissant une architecture de réseau sécurisée. Il empêche les accès non autorisés, applique des contrôles d'accès stricts et utilise des techniques avancées de prévention des menaces, qui protègent l'intégrité et garantissent la disponibilité des ressources critiques du réseau.

Le trafic VoIP, un service essentiel est très sensible et fait souvent l'objet d'attaques DDoS. En raison de sa dépendance à l'égard des communications en temps réel et des exigences strictes en matière de latence, les interruptions des services VoIP peuvent avoir un impact considérable sur la disponibilité opérationnelle.

La gestion SIEM dote de la sécurité des réseaux d'informations exploitables et d'une intelligence axée sur l'analyse, permettant une surveillance de la sécurité en temps réel, une identification avancée des menaces, une gestion des incidents et une détection des anomalies basée sur l'analyse comportementale.

Le serveur web sert d'interface web dans la couche DMZ et constitue une autre cible de choix pour les attaquants DDoS. Le système Kali Linux joue le rôle de l'attaquant DDoS dans l'environnement simulé et génère un trafic d'inondation pour submerger et perturber les services.

Les commutateurs relient les dispositifs du réseau à différentes couches, principalement à la couche 2 (couche de liaison de données) pour transmettre les trames sur la base des adresses “Media Access Control” (MAC). Les routeurs acheminent les paquets de la source à la destination par l’intermédiaire des chemins optimaux grâce à des protocoles de routage, afin d'assurer une transmission efficace des données sur les réseaux interconnectés.

Après une évaluation approfondie de toutes les exigences en matière de sécurité, l’infrastructure réseau devance les attaquants DDoS potentiels, ayant construit un modèle de défense adaptatif conçu pour protéger ses services. L'intégration du NGFW et du SIEM fonctionne en tandem pour créer un cadre de sécurité dynamique et résilient qui non seulement détecte et atténue les menaces, mais évolue également en permanence pour faire face aux risques émergents dans le paysage cybernétique en constante évolution.

L'évaluation des deux scénarios, l'un sans mesures de sécurité et l'autre avec des mécanismes de défense en place, a révélé des différences significatives dans les performances des ressources de Firepower. La charge du processeur a considérablement diminué, passant d'un pic de 88 % à une utilisation moyenne de 23 % sur tous les cœurs de traitement. L'utilisation de la mémoire du système a légèrement baissé à 59 %, tandis que la consommation de bande passante réseau a fortement chuté, avec un débit sortant passant de 352,86 kbit/s à 10,77 kbit/s. Les journaux de prévention des intrusions ont confirmé que Firepower a détecté et bloqué les modèles d'attaque dès les premières phases, ce qui a réduit le temps nécessaire pour identifier les menaces et éliminer le trafic non autorisé.

Une comparaison directe de ces scénarios a démontré les risques d’un réseau non protégé, où le trafic d’attaque non filtré surchargeait les ressources et exposait des services critiques. En revanche, le déploiement des politiques de sécurité multi-niveaux de Firepower, associé à l’analyse des points de terminaison de Wazuh SIEM, a constitué une architecture de protection efficace, garantissant la stabilité du réseau, optimisant l'utilisation des ressources et maintenant la disponibilité des services. Ces résultats ont validé la robustesse du cadre de sécurité proposé pour prévenir les attaques DDoS à grande échelle et protéger les infrastructures réseau.

ABSTRACT

In the rapidly advancing world of systems and technologies, evolving network security is not just a priority, it is a critical necessity. The focus of network infrastructures and information systems should remain on how to make them secure against cyberattacks. The rate of cyber criminals has significantly risen over the past few years, and it is of the utmost importance to strengthen cybersecurity measures and raise awareness to protect sensitive assets from potential threats.

The undeniable demand for effective security solutions continues to grow, as cyber breaches have become widespread, affecting systems and networks across various industries. This trend emphasizes the urgency to implement advanced methods and strategies to counter the rising number and sophistication of cyberattacks, which pose serious threats to both individuals and organizations globally.

The primary goal of current security metrics is to anticipate and manage common cyber threats, particularly Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. This involves leveraging predictive scenarios and advanced security tools to identify vulnerabilities and forecast attack patterns. When a threat is detected, automated, real-time defense mechanisms are immediately employed to mitigate the attack's impact.

This research aims to develop a secure Next-Generation Firewall architecture against DDoS attacks. It incorporates the use of Next-Generation Firewall (NGFW), Voice over Internet Protocol (VoIP) Traffic, Security Information and Event Management (SIEM), Web Server, Kali Linux, and Layer 2 Switches and Routers across three distinct network layers of Access, Demilitarized Zone (DMZ), and Core. By combining these technologies, the mechanism ensures robust protection and efficient threat mitigation at each network layer.

Initially, we will configure a network topology on the EVE-NG platform and VMware Workstation. The Next-Generation Firewall configuration and policy are pivotal in real-time detection and mitigation strategies against DDoS attacks, establishing a secure network architecture. It prevents unauthorized access, enforces strict access controls, and uses advanced threat prevention techniques, which protect the integrity and ensure the availability of critical network resources.

VoIP traffic, an essential service within organizations, is highly sensitive and often subjected to DDoS attacks. Due to its reliance on real-time communication and strict latency requirements,

disruptions in VoIP services can severely impact operational availability. SIEM management equips the network and data security ecosystem with actionable insights and analytics-driven intelligence, enabling real-time security monitoring, advanced threat identification, incident management, and anomaly detection based on behavioral analytics.

The Web Server serves as the web interface in the DMZ layer and counts as another prime target for DDoS attackers. Kali Linux system plays the role of the DDoS attacker in the simulated environment and generates flooding traffic to overwhelm and disrupt services. Switches connect network devices at different layers, primarily at Layer 2 (Data Link Layer) to forward frames based on “Media Access Control” (MAC) addresses. Routers deliver packets from source to destination by optimal paths through routing protocols to provide efficient data transmission across interconnected networks.

After thoroughly assessing all security requirements, the network infrastructure advances ahead of potential DDoS attackers, having built an adaptive defense architecture designed to protect its services. The integration of the NGFW and SIEM work in tandem to create a dynamic, resilient security architecture that not only detects and mitigates threats but also continuously evolves to address emerging risks in the ever-changing cyber landscape.

The evaluation of both scenarios, one without security controls and the other with defense mechanisms in place, revealed substantial differences in Firepower’s resource metrics. Processor load dropped significantly, from a peak of 88% to an average utilization of 23% across all processing cores. System memory usage declined slightly to 59%, while network bandwidth demand decreased sharply, with outbound data flow falling from 352.86 kbit/s to 10.77 kbit/s. Intrusion prevention logs confirmed that Firepower detected and blocked attack patterns at an early stage, which minimized the time required to identify threats and remove unauthorized traffic.

A direct comparison of these scenarios demonstrated the risks of an unprotected network, where unrestricted attack traffic overwhelmed resources and exposed critical services. In contrast, the deployment of Firepower’s multi-layered security policies, alongside Wazuh SIEM’s endpoint analysis, provided a strong mitigation strategy that preserved network stability, optimized resource usage, and ensured uninterrupted service. These results validated the strength of the proposed security architecture to prevent large-scale DDoS flood attacks and protect network infrastructures.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

Ack	Acknowledgment
AI	Artificial Intelligence
APT	Advanced Persistent Threats
AMP	Advanced Malware Protection
ACL	Access Control List
API	Application Programming Interface
AIOHTTP	Asynchronous HTTP
CAGR	Compound Annual Growth Rate
CIAC	Computer Incident Advisory Capability
CIDoS	Cloud-internal Denial of Service
CPU	Central Processing Unit
CVE	Common Vulnerabilities and Exposures
DDoS	Distributed Denial of Service
DEC	Digital Equipment Corporation
DHCP	Dynamic Host Configuration Protocol
DoS	Denial of Service
DPI	Deep Packet Inspection

DMZ	Demilitarized Zone
DNS	Domain Name System
DUT	Device Under Test
EDR	Endpoint Detection and Response
EFT	Electronic Funds Transfer
EDOS	Economic Denial of Sustainability
ETH	Ethernet
EPS	Events Per Second
EU ITSRM	European Union Information Technology Security Risk Management
EVE-NG	Emulated Virtual Environment-Next Generation
FIN	Finish
FISMA	Federal Information Security Management Act
FMC	Firepower Management Center
FTD	Firepower Threat Defense
GDPR	General Data Protection Regulation
GB	Gigabyte
GHz	Gigahertz
GUI	Graphical User Interface
GNU	Gnu's Not Unix

HIPAA	Health Insurance Portability and Accountability Act
HIDS	Host Intrusion Detection System
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IAM	Identity and Access Management
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IEC	International Electrotechnical Commission
I/O	Input/Output
IoC	Indicators of Compromise
IoT	Internet of Thing
IP	Internet Protocol
IPS	Intrusion Prevention System
ISO	International Organization for Standardization
ISE	Identity Service Engine
ISP	Internet Service Provider
IDPS	Intrusion Detection and Prevention System
Kbit	Kilobit
LDAP	Lightweight Directory Access Protocol

LOG	Logging
LFA	Link Flooding Attack
MGCP	Media Gateway Control Protocol
ML	Machine Learning
NAT	Network Address Translation
NIDPS	Network Intrusion Detection and Prevention System
NGFW	Next-Generation Firewall
NGIPS	Next-Generation Intrusion Prevention System
NIST	National Institute of Standard and Technology
Nmap	Network Mapper
NIC	Network Interface Card
OCTAVE	Operationally Critical Threat, Asset, and Vulnerability Evaluation
OSCAR	Obtain Information, Strategies, Collect Evidence, Analyze, and Report
OSI	Open Systems Interconnection
OSSEC	Open Source Host-based Intrusion Detection System
OS	Operating System
PCI DSS	Payment Card Industry Data Security Standard
PSTN	Public Switched Telephone Network
QoS	Quality of Service

RDDoS	Ransom Distributed Denial of Service
RMF	Risk Management Framework
RTP	Real-Time Transport Protocol
RAM	Random-Access Memory
Rx	Receive
SIEM	Security Incident and Event Management
SIP	Session Initiation Protocol
SCP	Secure Copy Protocol
SME	Small and Medium-sized Enterprise
SMS	Short Message Service
SOAR	Security Orchestration, Automation, and Response
SQL	Structured Query Language
SR-DRDoS	Spoof Reflection Distributed Reflection Denial of Service
SSH	Secure Shell
SSL	Secure Sockets Layer
SYN	Synchronize
SYSLOG	System Logging
SNMP	Simple Network Management Protocol
TCP	Transmission Control Protocol

TIP	Threat Intelligence Platform
TLS	Transport Layer Security
Tx	Transmit
TTL	Time To Live
UDP	User Datagram Protocol
UEBA	User and Entity Behavior Analytic
URL	Uniform Resource Locator
UTM	Unified Threat Management
VoIP	Voice over IP
VPN	Virtual Private Network
VPC	Virtual Private Client
VM	Virtual Machine
WAF	Web Application Firewall
XDR	Extended Detection and Response
XSS	Cross-Site Scripting

CHAPTER 1 INTRODUCTION

In recent years, large-scale service infrastructures have emerged as attractive targets for cybercriminals, drawn by the opportunity to access sensitive data and extract significant value. As cyber-attacks grow both in frequency and complexity, these institutions face mounting challenges in identifying and mitigating potential threats. With the continuous expansion of the digital ecosystem, critical digital infrastructures must contend with a web of interdependent vulnerabilities [1]. To protect their systems and operations, they must employ highly strategic and robust risk management approaches [1].

Citing this fact, we can rely on *Next-Generation Firewall (NGFW)* [2] mechanisms, which is an appliance that inspects traffic in terms of contents and regulates security policies in the application layer [2]. This level of protection is vital to detect and mitigate sophisticated cyber attacks that can compromise critical assets and disrupt essential services [2]. Another security metric would remain for the *Security Incident and Event Management (SIEM)* [3] system, which is an integral part of every security structure. SIEM collects and correlates data from various sources, enabling the detection of anomalies and potential threats before they escalate [3]. By offering a centralized view of the organization's security posture, SIEM enhances incident response and ensures compliance with industry regulations, helping institutions protect sensitive data and maintain operational integrity [3].

In this dissertation, we will propose a secure architecture against *Distributed Denial of Service (DDoS)* attacks with a focus on *Next-Generation Firewall (NGFW)* strategies [2] and *Security Incident and Event Management (SIEM)* capabilities [3]. In this chapter, we will discuss the concept of the *Next-Generation Firewall* and *SIEM* systems alongside *DDoS* attacks and *Voice over IP (VoIP)* applications. Besides, we will propose the main problem statement followed by the specific problems with the hope of detecting and mitigating malicious attempts, then the main and specific objectives of this research, to the end with the plan of the dissertation.

1.1 Basic definitions and concepts

This section delves into the foundational definitions and key principles of *Next-Generation Firewalls (NGFW)* and *Security Information and Event Management (SIEM)* systems, and highlights their critical role in the defense of network institutions against *Distributed Denial of Service (DDoS)* attacks. Detection is identifying malicious traffic patterns or anomalies, such as high request rates, malformed packets, or protocol violations, through real-time inspection, signature match, and behavior analysis by NGFW and SIEM tools [4]. Mitigation is the act of responding immediately to identified threats by enforcement of firewall policies, rate limits, or packet drops to prevent disruption and to ensure service continuity and network integrity [4].

A firewall is a network security device that applies access control policies to allow or block traffic based on IP address, port number, and protocol at the network and transport layers (Layer 3 and Layer 4) [3]. A Next-Generation Firewall extends the function of a traditional firewall by use of deep packet inspection, intrusion prevention, and threat intelligence to analyze traffic at the application layer (Layer 7) [3]. A traditional firewall applies rules based on static criteria, while an NGFW adds application-layer analysis, user identification, and real-time threat prevention to block advanced network threats.

Additionally, we will examine the weaknesses present in *Voice over IP (VoIP)* systems, which are frequent targets of such threats. Subsequently, we will outline the problem statement, followed by a discussion of the research objectives. The section will conclude with an overview of the dissertation’s structure.

1.1.1 Next-Generation Firewall System

In response to the mounting cyber threats, particularly those targeting critical service infrastructures, the field of cybersecurity has undergone substantial innovation. As these organizations continue to expand their digital operations, the urgency for more advanced security frameworks has become paramount. Protecting sensitive data, maintaining seamless service delivery, and mitigating the ever-present risks of cyber-attacks, including *Distributed Denial of Service (DDoS)*, have become central concerns in safeguarding the stability and reputation of institutions.

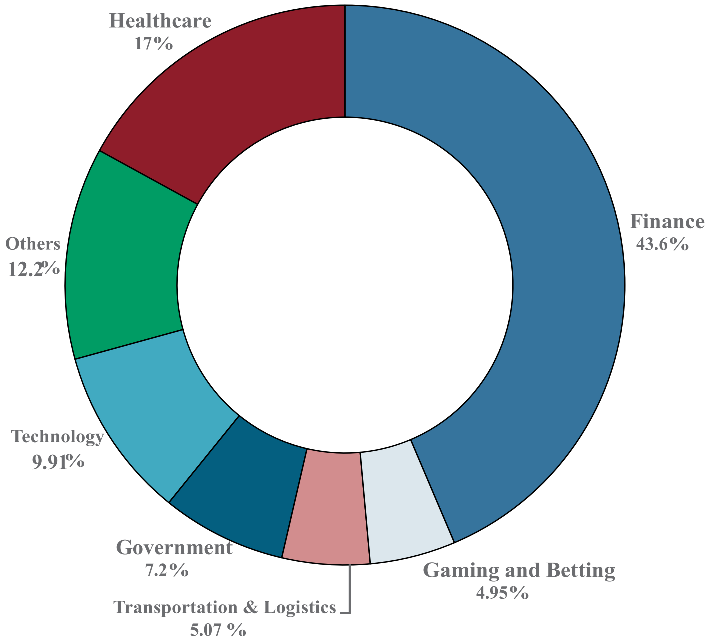


Figure 1.1: Most Attacked Industries in 2024 H1 (source: Radware [1])

Considering recent reports, we have observed a dramatic increase in the rate of DDoS attacks targeting high-value digital infrastructures over the past few years, and this trend is expected to continue in the years ahead.

In the first quarter of 2024, certain industries faced a disproportionate share of network layer *DDoS* attacks as shown in Figure 1.1 [1]. Notably, organizations within finance experienced almost 44% of the global attack activity [1]. Furthermore, attacks have grown so much in fact that, on average, businesses can be expected to deal with a *DDoS* attack around eleven times a year, almost once a month [1].

The *Next-Generation Firewall (NGFW)* technology is the key solution in response to the *DDoS* attack threat, as it can identify and control traffic at the application layers by looking deep into the application layer, i.e., the packet's payload. Traditionally, firewalls focus on the network protocols and the headers of the packets. Application intelligence is an essential part of *NGFW*. It means that *NGFW* is capable of identifying applications regardless of port and protocol, identifying a user and linking user identity with the packet, and identifying the true intention of the payload. The concept of *NGFW* is to achieve everything a traditional firewall does with expanded capabilities that combine novel identification technologies, high-performance, and extra innovative features [2].

1.1.2 SIEM Systems

As we discussed earlier, one of the fundamental applications in security domains is the *Security Information and Event Management (SIEM)* systems. *SIEM* is a security solution that helps organizations recognize potential security threats and vulnerabilities, detects user behavior anomalies, and provides an incident response. *SIEM* aggregates and analyses activities from different sources across the IT infrastructure. *SIEM* collects data from network devices, servers, domain controllers, and more. It will then store, normalizee, and aggregate the data before applying analytics to the data to discover trends and detect threats. It will then alert the organization of any threats and anomalies. It is a highly efficient data orchestration system for managing ever-evolving threats as well as regulatory compliance and reporting [3].

Security Information and Event Management (SIEM) systems have been developed in response to help administrators design security policies and manage events from different sources. Generally, a simple *SIEM* is composed of separate blocks (e.g., source device, log collection, parsing normalization, rule engine, log storage, and event monitoring) that can work independently from each other, but without them all working together, the *SIEM* will not function properly. Figure 1.2 depicts the basic components of a regular *SIEM* solution [4].

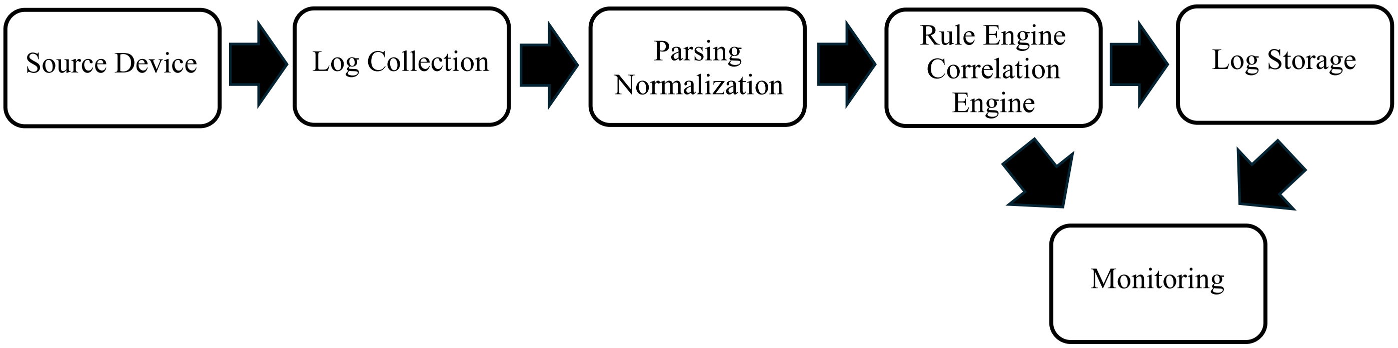


Figure 1.2: SIEM Basic Components

A *SIEM* solution performs key functions including log collection, which gathers information from various source systems and stores it centrally for historical access. It also involves log normalization, ensuring data from different sources is unified into a common model for easier structuring. Log aggregation combines data based on shared attributes, removing duplicates, while log correlation links events across systems with varying formats and times, allowing for single actionable events. Lastly, reporting provides historical reports and real-time monitoring for comprehensive data presentation [4].

1.1.3 Overview of DDoS Attack

Distributed Denial of Service (DDoS) is a cyber attack method where a large number of attackers from different locations simultaneously launch attacks on a specific target, thereby paralyzing the functionality of legitimate users. The first attack report was submitted by the *Computer Incident Advisory Capability (CIAC)* [5]. Since then, *DDoS* attacks have begun to proliferate on the Internet. Malicious actors have launched attacks on targets for various purposes, such as economic crimes, revenge, show-off, and cyber-warfare [5].

Attackers started sequestering machines connected to the Internet, making it possible to circumvent the weakness of the single launch point of the attack. Therefore, *DDoS* attacks combine several connected devices to attack a target. In the classical model, *DDoS* attacks exhaust the computing resources of the victim's infrastructure by creating multiple connections from different sources. *Link Flooding Attack (LFA)* is another way to degrade or disrupt the victim's service, congesting critical links to isolate the victim's network from the Internet [6]. Another way to cause damage to the victim is by pursuing the *Economic Denial of Sustainability (EDoS)* [6]. In *EDoS*, the goal is to consume the victim's resources, forcing the victim to allocate more computing resources. This action increases the cost necessary to keep the service running. *Cloud-internal Denial of Service (CIDoS)* consumes server resources utilizing multiple virtual machines hosted on the victim's physical host [6]. The attacker increases the workload of virtual machines to consume more resources from the host and consequently stops the service [6]. *Ransom DDoS (RDDoS)* is another way to damage utilizing variations of the *DDoS* attack. In this case, the attackers request the payment of ransom to suspend or not launch *DDoS* attacks against the victim [6].

The standard parties of a DDoS attack are the attackers, the infected devices, and the victim. A zombie, web robot, or bot is a malware-infected device connected to the Internet that executes programmed tasks [6]. Bots perform actions such as sending bulk electronic correspondence (spam), sniffing traffic, capturing sensitive information, phishing, click fraud, keylogging, disseminating software for cryptocurrency mining, and launching *DDoS* attacks. A "robot network" or botnet is a group of several bots remotely controlled by attackers or botmasters [6]. A victim is a server or computer network that holds some resources for the correct operation of a service [6]. A botmaster sends commands to the botnet to initiate connections to the victim to conduct a DDoS attack. The duration of DDoS attacks varies from minutes to days, reaching terabits per second or surpassing millions of requests-per-second [6].

In 2023 there was a more recent example of the power of *DDoS* attacks involving GitHub, which is known as one of the world's software development platforms. During this incident, an effective *DDoS* attack specifically targeted GitHub’s infrastructure [7]. The attackers skillfully used a combination of volumetric and application layer attack techniques causing disruptions, to the platform’s services and impacting its user community. GitHub, which plays a role in software development encountered difficulties, in maintaining its services amidst this onslaught. This incident served as a reminder of how *DDoS* attackers adapt and find ways to bypass mitigation strategies [7].

Figure 1.3 [6] illustrates a *Distributed Denial of Service (DDoS)* attack, where a botmaster remotely controls a network of compromised devices, or bots, to launch coordinated attacks. By exploiting vulnerabilities in internet-connected devices such as television, laptop, IP camera, smart phone, the botmaster infects them with malicious code and later commands them to flood target systems with traffic, overwhelming resources and disrupting legitimate access.

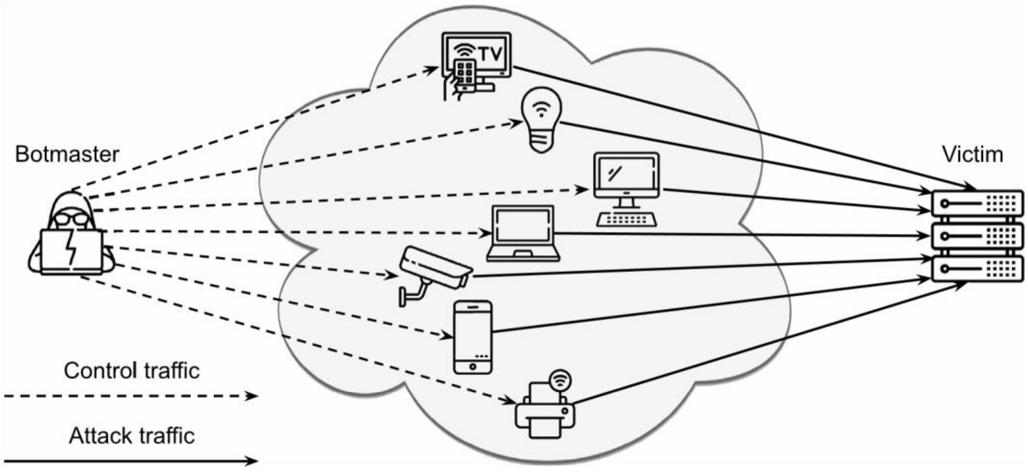


Figure 1.3: Basic Concept of DDoS Attack [6]

1.1.4 Voice over IP System

Voice over Internet Protocol (VoIP) is a technology that enables voice communication at any time over the Internet. *VoIP* offers a high degree of accessibility relative to circuit-switched networks with interesting benefits such as phone functionality, teleconferencing and convergence of services with low cost [8]. The use of *VoIP* is increased due to the millions of users that have used *VoIP* technology in recent years. This lead to many difficulties and security issues because attackers attack user networks to steal important information [8]. With the increment of *VoIP* users, malicious attackers also increased. Many attacks, such as call tampering, *Denial of Service (DoS)* and *Distributed Denial of Service (DDoS),* call hijacking, man-in-the-middle, ID spoofing, spamming, and eavesdropping are *VoIP* security attacks that disrupt *VoIP* traffic [8].

VoIP systems use data transfer protocols such as the *Real-Time Transport Protocol (RTP)* to transfer voice and multimedia data over packet-switched IP networks [9]. In addition, they use signaling protocols such as *H.323* or the *Session Initiation Protocol (SIP)* to manage communication sessions [9].

Therefore, *VoIP* systems are vulnerable to attacks that are generated from two sources, network protocols such as the *User Datagram Protocol (UDP)* and *VoIP* specific protocols such as *SIP* [9]. *VoIP* specific attacks usually are not detected by network security systems, extra security mechanisms are therefore necessary for *VoIP* systems to detect and prevent these attacks [9].

VoIP networks have recently been vulnerable to many security threats. In addition, the intensity of attacks seems to have been growing, this might be a result of the rapid increase in the capabilities of tools used by attackers [9]. Two of the most harmful and specific types of *VoIP* attacks are *denial of service (DoS)* and *Distributed Denial of Service (DDoS)*. The main objective of these attacks is to prevent legitimate users from using *VoIP* services. These attacks may affect *VoIP* service availability by targeting one or many *VoIP* servers. Such attacks can thus affect business productivity and lead to revenue loss [9].

The challenges discussed above open up several key research directions. Firstly, as new forms of *VoIP*-specific attacks, including advanced *DDoS* methods, continue to emerge, there is a pressing need for adaptive detection techniques that can dynamically respond to these evolving threats. Furthermore, there is a demand for highly efficient detection mechanisms, as existing solutions often prioritize accuracy at the cost of complexity and resource consumption, reducing their effectiveness in real-world applications. Notably, real-time detection has become an important area of focus, particularly in complex and high-demand environments like *VoIP* systems, where lightweight, efficient methods are essential to balance performance with security.

1.1.5 Concepts of Proposed Security Solutions

We will implement security measures in our architecture through a layered defense strategy that includes *Next-Generation Firewalls (NGFW)* for in-depth traffic analysis, *Security Information and Event Management (SIEM)* for continuous threat monitoring, and controlled traffic simulation to evaluate network robustness. This integrated approach facilitates early threat identification and response, strengthens oversight, and reinforces the infrastructure against *Distributed Denial of Service (DDoS)* attacks, providing reliable protection for essential operations.

The primary security challenge addressed in this dissertation is *Distributed Denial of Service (DDoS)* attacks, a significant threat to network institutions. *DDoS* attacks aim to overwhelm targeted servers or network resources by flooding them with excessive traffic, ultimately disrupting legitimate user access and impairing essential services. This issue becomes even more critical in *VoIP* systems, where attacks such as call tampering and resource exhaustion can severely impact communication reliability. Given the increasing sophistication of *DDoS* methods and the role of interconnected devices as potential attack vectors, addressing this vulnerability is essential.

To counter these threats, we will develop a multi-layered strategy that deploys *Next-Generation Firewalls (NGFW)* and *Security Information and Event Management (SIEM)* systems across segmented network zones. This topology establishes a defense-in-depth approach, compartmentalizing critical assets to reduce the attack surface and enhance traffic governance. *NGFWs* offer granular inspection capabilities at the application layer, enabling precise traffic control and adaptive filtering to counter sophisticated attack vectors. Simultaneously, the *SIEM* system functions as an intelligence hub, aggregating, normalizing, and correlating event data from diverse network segments to facilitate anomaly detection and expedite threat response. This integrated approach fortifies the network against high-impact threats like *DDoS* attacks, ensuring resilience, service availability, and regulatory compliance in network infrastructure. We will discuss the proposed architecture and more details about these technologies and concepts in Chapter 3. The next section will explain our problem statement in this domain.

1.2 Problem Statement

A *DDoS* attack can occur in a critical network through spoofing techniques, where attackers manipulate the IP address of the traffic to make it seem like it's coming from a legitimate source [10]. This can make identifying and blocking fake traffic challenging for the organization's security systems [10]. In the business sector, *DDoS* attacks can be highly detrimental as they can prevent customers from accessing their accounts, conducting transactions, or utilizing online services [10]. Consequently, this can lead to substantial operational disruptions, reputation damage, and erosion of stakeholder trust. To mitigate these attacks, organizations should implement strong security measures such as *Next-Generation Firewalls (NGFW)*, *Security Information and Event Management (SIEM)* systems, intrusion detection systems, content filtering, and collaboration with third-party security providers [10].

The "DDoS Attack Mitigation Framework" was initiated by the *National Institute of Standards and Technology (NIST)* [11]. This architecture lists concerns for detecting, locating, and halting *DDoS* attacks. Following the risk assessment, policies and procedures should be developed to mitigate these threats [11]. The information security framework must include a robust network infrastructure. The infrastructure of a network must be designed to handle high traffic volumes without becoming overburdened. This can be achieved by integrating powerful vendors of *Next-Generation Firewall (NGFW)* and *Security Information and Event Management (SIEM)* [11].

Moreover, DDoS attacks often serve as distractions, diverting attention from other critical security incidents like data breaches, insider threats, or advanced persistent threats, thereby amplifying the impact on the targeted institution [12].

Therefore, continuous monitoring and testing are required in the system of network institutions, which can be achieved by *SIEM* systems that identify unusual traffic patterns or system anomalies [12]. The information security framework should incorporate plans for business continuity and calamity recovery [12]. These plans should ensure that essential business operations can continue even in the event of a DDoS attack [12].

The information security framework should include dependable network infrastructure, advanced security technologies (i.e., SIEM, EDR and XDR devices), employee training, incident response procedures, continuous monitoring and testing [12].

Considering these factors, our main research question is as follows: What strategies can Next-Generation Firewalls adopt to secure enterprise networks against DDoS attacks? From this primary question, the following two secondary questions are derived:

1. Which method to identify the most problematic types and scenarios of DDoS (Distributed Denial-of-Service) attacks?

2. Which Next-Generation Firewall strategy should be proposed to detect and mitigate real-time DDoS attacks?

1.3 Research objectives

The core objective of this dissertation is to propose a secure Next-Generation Firewall architecture against DDoS attacks. More specifically, this research aims to:

1. Analysis of the most problematic types and scenarios of DDoS attacks;
2. Propose a Next-Generation Firewall architecture to detect and mitigate real-time DDoS attacks;
3. Implement the proposed architecture;
4. Evaluate the performance of the proposed architecture in terms of security against DDoS attacks;

1.4 Dissertation plan

The remainder of this dissertation is organized as follows. Chapter 2 presents a comprehensive review of related work on cybersecurity for network infrastructures, with a particular focus on *Distributed Denial of Service (DDoS)* attacks and defense mechanisms. Chapter 3 presents the proposed secure Next-Generation Firewall architecture against DDoS attacks. The results and the evaluation of the performance of the proposed architecture are discussed in Chapter 4. Finally, Chapter 5 summarize the research work, outlines its limitations, and proposes avenues for future exploration.

CHAPTER 2 LITERATURE REVIEW

In this chapter, we will demonstrate a comprehensive literature review of Next-Generation Firewall (NGFW) and Security Information and Event Management (SIEM) systems, DDoS attack detection, and mitigation approaches to the presence of VoIP traffic. Besides, we will analyze the most recent defensive measures against DDoS attacks. By analyzing current strategies and methodologies within the domain of securing network infrastructures against DDoS attacks, we identify existing gaps and explore advanced tools, which enable us to develop targeted and effective solutions aligned with the objectives of this dissertation.

2.1 Firewall System

There is a substantial body of related work in the field of cybersecurity on Next-Generation Firewall (NGFW) and Security Information and Event Management (SIEM) strategies against DDoS attacks. This section will examine related works across two critical areas: advanced methodologies utilizing NGFW architectures and SIEM integration for DDoS anomaly detection and robust DDoS mitigation frameworks designed to enhance network resilience and maintain operational continuity. After this section, we will summarize our findings within these two areas to highlight gaps and inform our research direction.

Rapid advancements in technology and digitization continually push the world toward new milestones and increasingly complex challenges. Breakthrough innovations like the Internet of Things (IoT), cloud computing, and blockchain enhance, automate, and elevate everyday life. However, these technologies also introduce a host of new vulnerabilities, compounding existing online threats. Individuals and businesses alike face escalating risks from cyber threats and attacks, leaving them increasingly exposed to malicious actors. To fulfill stringent network security requirements, the infrastructure must incorporate enterprise-grade security appliances from leading manufacturers, providing advanced security capabilities and extensive configuration features. Nowadays, companies take advantage of Next-Generation Firewall devices to provide available services for customers and networks and assist security experts. A traditional firewall is like the first security agency, it blocks or allows data (passengers) based on where it is going, whether or not it is part of a legitimate network connection, and where it comes from. An NGFW is more like the second security agency, it inspects data on a deeper level to identify and block threats that may be hidden in normal-seeming traffic.

2.1.1 Traditional Firewall

Firewalls began with packet filtering in 1988, developed by Digital Equipment Corporation (DEC) to block specific IP addresses without targeting viruses or threats [13]. In 1990, American Telephone and Telegraph Company (AT&T) Bell Labs enhanced this with stateful firewalls for better packet handling [13]. The growth of the internet led DEC to introduce the application gateway firewall in 1991 to detect attacks, followed by the web application firewall in 1997 for web server security [13]. Unified Threat Management (UTM), introduced in 2004, combined firewall types into one system, later evolving into Next-Generation Firewall (NGFW) technology in 2009 [13]. NGFW, now widely used in industries, leverages machine learning for improved threat detection [13]. Subsequent sections will detail each firewall type, focusing solely on network firewalls.

2.1.2 Packet Filtering Firewall

As the earliest form of firewall, the packet-filtering firewall manages traffic by analyzing incoming and outgoing packets. It uses rules based on IP addresses, protocols, and port numbers to determine which packets to allow or block. Its basic design makes it straightforward to set up and effective for simple filtering tasks [14]. However, this firewall does not examine the payload at the application layer or authenticate users, leaving it vulnerable to IP spoofing and inadequate for defending against sophisticated attacks [14].

2.1.3 Stateful Firewall

The stateful firewall is an enhanced version of the packet-filtering firewall, operating based on the Transmission Control Protocol (TCP) three-way handshake process. It tracks active connections by monitoring the flow of packets in both directions [14]. To enable this, the firewall has a cache that records each traffic flow. When a new connection, indicated by a Transmission Control Protocol Synchronization (TCP SYN) packet, passes through, the firewall creates a profile for it using header details like IP addresses and port numbers [14]. Future connections are checked against this cache, allowing packets to pass if a matching entry is found. If no match exists, the firewall uses basic packet filtering rules to decide if the connection should be added. Entries are removed from the cache when the connection ends (via a Transmission Control Protocol Finish (TCP FIN) packet) or becomes idle for too long, at which point the firewall blocks the flow since it's no longer recorded in the cache [14].

While more secure and effective than packet-filtering firewalls, the stateful firewall requires extra storage for caching. However, it has drawbacks, such as excessive traffic, as seen in DoS attacks, which can lead to cache table overflow. Furthermore, it shares some limitations with packet-filtering firewalls, including a lack of payload inspection and user authentication capabilities.

2.1.4 Application-Gateway Firewall

In contrast to earlier firewall types, the application gateway firewall offers control over network traffic at the application level. Rather than acting as a traditional "wall" between the source and destination, it functions more like a "gateway" that inspects packets at the application layer and uses proxies to establish sessions with remote users. The gateway creates a secure connection when a client attempts to access the server's resources. All resources from the server first pass through the gateway and are only forwarded to the client if requested.

This extra layer of security shields the server's network resources from direct client access. Should any network resources be compromised, the gateway is impacted initially, allowing for quick resolution without affecting the client [15]

Despite this benefit, the added layer slows down traffic flow between the client and server, as the secure gateway introduces latency in data transmission.

2.1.5 Circuit-Level Gateway Firewall

The circuit-level gateway firewall serves as a "gateway" linking clients to servers, similar to an application gateway firewall, but it functions at the session layer (layer 5) of the Open Systems Interconnection (OSI) model instead of the application layer (layer 7). This operational level shifts the focus from the server's network resources to the actual connections established between the client and server. The circuit-level gateway firewall provides a proxy that creates a virtual circuit for transparent end-to-end communication, facilitating TCP connections between the client and server. This process ensures that connections are valid and that server responses align with client requests. Discrepancies between requests and responses could suggest an attempt to exfiltrate data from the network [16].

Operating exclusively at the OSI model's session layer, circuit-level gateway firewalls do not offer packet-level security. They simply validate that connections are authentic without analyzing the data transmitted. This results in higher speed compared to application-gateway firewalls, though with reduced security. As a result, they are commonly used in conjunction with additional firewall types.

2.1.6 Web-Application Firewall

The Web-Application Firewall (WAF) is a type of application firewall tailored for web application security. It serves as a security component within an application proxy, shielding the back-end web server from various threats. WAF examines Hypertext Transfer Protocol (HTTP) and Hypertext Transfer Protocol Secure (HTTPS) requests and analyzes network traffic patterns to detect potential risks. When harmful packets or suspicious patterns are detected, the WAF blocks the HTTP requests or terminates the session between the client and server to prevent attacks [17].

WAF is particularly effective at blocking known web-based threats, such as Cross-Site Scripting (XSS), Structured Query Language (SQL) injections, and DDoS attacks [18]. However, they are less effective against zero-day vulnerabilities, as their threat detection relies heavily on pattern recognition. When an unfamiliar attack lacks a known pattern, WAFs struggle to detect it, making it challenging to counter zero-day threats. Yet, because zero-day exploits are rare and many websites remain vulnerable to known attacks, the demand for WAFs continues to rise. The Web-Application Firewall Market size is estimated at USD 6.80 billion in 2024 and is expected to reach USD 14.60 billion by 2029, growing at a Compound Annual Growth Rate (CAGR) of less than 19.90% during the forecast period (2024-2029) [19]. Figure 2.1 [19] illustrates the WAF mechanism, where the WAF inspects traffic to block malicious requests from attackers and permits legitimate user access to the web server. This filtering process protects the server against typical web threats, including SQL injection and XSS [19].

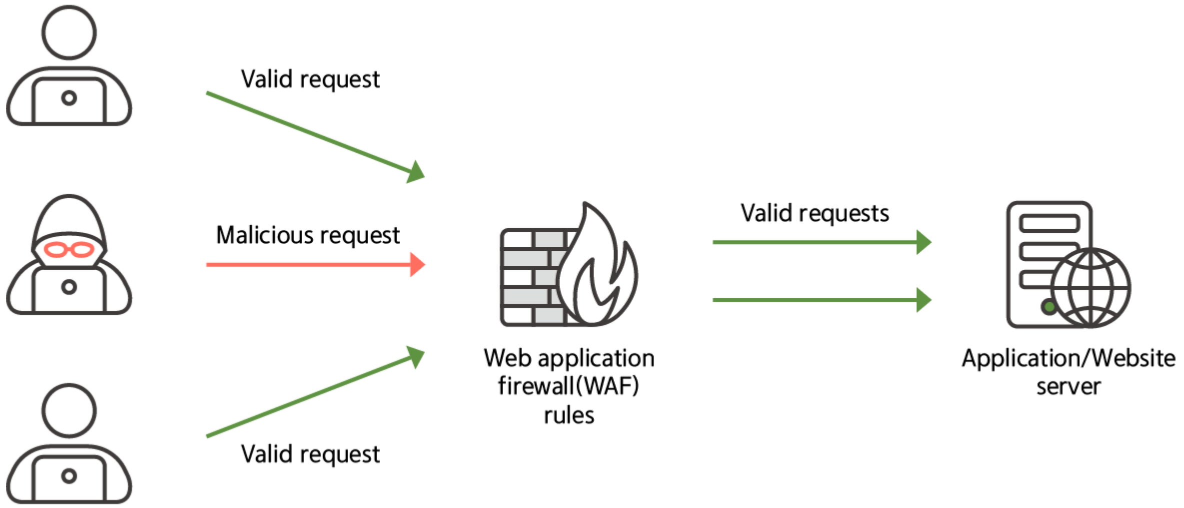

Valid request
Malicious request
Valid request
Web application firewall(WAF) rules
Valid requests
Application/Website server

Figure 2.1: Web-Application Firewall Concept [19]

2.1.7 Unified Threat Management

Unified Threat Management (UTM) is a strategy where a single device, either software or hardware, integrates various security functions to address multiple types of cyber threats, earning the title "unified threat management." UTM encompasses a range of defensive techniques from traditional firewalls, including rule-based packet filtering, stateful inspection, deep packet inspection, intrusion detection and prevention systems, application and circuit-level gateways, among others, as shown in Figure 2.2 [20].

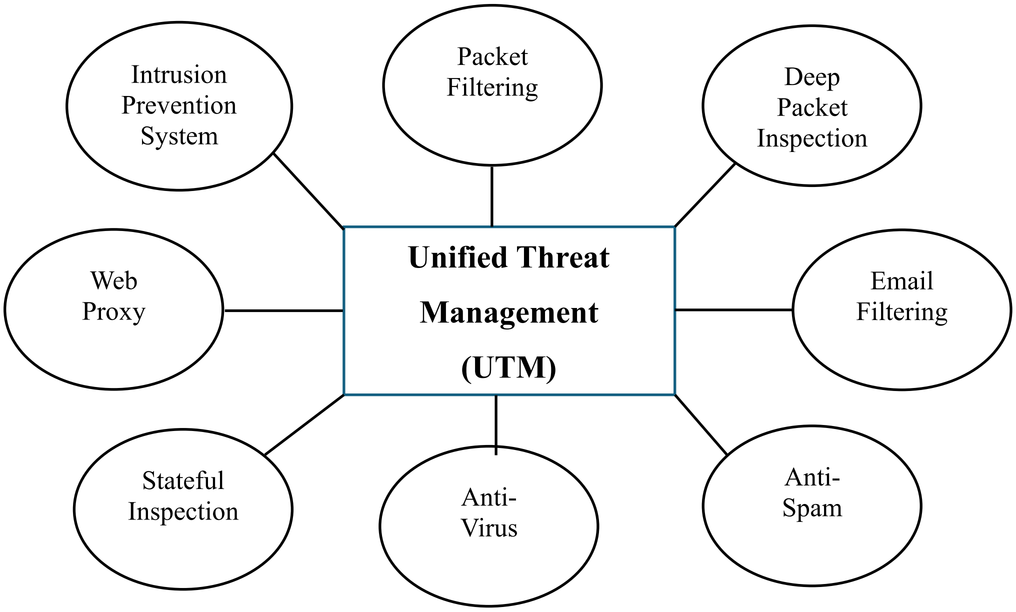


Figure 2.2: Unified Threat Management Features [20]

Beyond standard firewall features, UTM includes anti-spam and anti-phishing measures through comprehensive content analysis. By streamlining security management, UTM reduces the need for separate firewalls tailored to specific threats, making it easy to install and maintain. This versatility positions UTM as a top choice for industrial network defense [21].

Despite offering layered protection in a single system, UTM systems have a vulnerability to single points of failure, if one-layer malfunctions, it can jeopardize the entire network's security. Furthermore, these devices tend to reduce network efficiency as packets undergo multiple stages of scrutiny, which can cause substantial delays in traffic flow [22].

2.1.8 Firewall Summary

In earlier sections, six different firewall types were covered, each with unique characteristics, benefits, and limitations. Table 2.1 outlines the OSI layer at which each firewall functions, along with its respective advantages and disadvantages.

Table 2.1: Firewall Comparison

Types of Firewalls	**Firewall Features**		
	OSI Model Layer	**Advantages**	**Disadvantages**
Packet Filtering Firewall	Network Layer (Layer 3)	Efficient, Easy to setup	Do not inspect the payload, No user authentication
Stateful Firewall	Network Layer (Layer 3)	More efficient	Takes more space, Vulnerable to DDoS attack
Circuit Gateway Firewall	Session Layer (Layer 5)	Defense in layer 5, Detects if data is being exfiltrated	Additional time to process traffic
Application -Gateway Firewall	Application Layer (Layer 7)	Defense in layer 7, Detect malicious payload	Additional time to process traffic
Web-Application Firewall (WAF)	Application Layer (Layer 7)	Defense against known web attacks	Unable to detect zero-day attacks
Unified Threat Management (UTM)	All Layers	All layer protection, Additional protection, such as anti-spam	Single point of failure, Could decrease performance drastically

Concerning the prior firewall weaknesses, we can easily assert that none of these traditional solutions adequately addresses the sophisticated and multi-layered nature of modern DDoS attacks. Therefore, this limitation underscores the fundamental necessity of Next-Generation Firewall (NGFW) as a cornerstone in the establishment of a resilient and dynamically adaptive security architecture that protects against sophisticated and advanced DDoS attacks.

2.2 Next-Generation Firewall

The Next-Generation Firewall (NGFW) represents a highly evolved security device that blends conventional firewall capabilities with modern, advanced functionalities. As part of the third-generation firewall technology, it conducts in-depth packet inspection, applies application-layer filtering, and uses Intrusion Prevention System (IPS) to block intrusions. NGFW addresses the limitations of traditional firewalls through the protection of internal networks from DDoS attacks and the provision of secure data transfer via Secure Sockets Layer (SSL). This firewall incorporates a range of methods, including IPS, Uniform Resource Locator (URL) filtering, antivirus, and deep inspection, along with legacy approaches to enhance overall security [23].

Today, we encounter a wide range of cyber threats daily. These threats vary in form and can directly target the operating system through malware, worms, viruses, trojans, and ransomware. They may also exploit communication channels between devices, as seen with man-in-the-middle and DDoS attacks, or belong to newer, more sophisticated threats like zero-day exploits and Advanced Persistent Threat (APT) [2]. The complexity of cybercrime and the capabilities of the organizations behind these attacks make it impossible to prevent every cyber-attack [2]. Therefore, the focus shifts to damage control through preventive measures, recovery protocols, specialized training, and advanced technologies, where NGFW proves invaluable [2]. Traditional firewalls primarily relied on packet filtering, using source and destination addresses, port numbers, and protocol fields [24]. In contrast, NGFW offers advanced functionalities, including application-aware traffic filtering, user identity verification, and access to real-time threat intelligence [24].

Unlike traditional firewalls, which operate at the network and transport layers (OSI layers 3 and 4), NGFW is engineered to understand the complex nature of modern traffic patterns. Standard firewalls primarily filter traffic using basic attributes such as IP addresses, ports, and protocols, providing a perimeter-based security approach suitable for early network models. Perwaiz *et al.* [25] identify that while traditional firewalls were once effective, they lack the depth required to defend against contemporary, evasive cyber threats, necessitating more dynamic, context-aware solutions like NGFW.

NGFW enhances traditional packet filtering by extending its inspection capabilities to the application layer [25]. Where a traditional firewall might only recognize that traffic is using port 80, an NGFW can interpret the traffic type, distinguishing legitimate web browser activity from potentially harmful software masked within the flow. With this deeper insight, NGFW is equipped to detect and block threats that traditional firewalls may overlook [25].

2.2.1 Key Features of Next-Generation Firewall

NGFW includes specific attributes that differentiate it from conventional firewalls, providing superior protection features as follows:

Application Awareness: Freet and Agrawal [26] highlight that Next-Generation Firewall enables application-level traffic detection, making it possible to accurately identify the type of application active on the network, such as social media, cloud-based platforms, or business software. This feature enables security administrators to create more granular policies, allowing access to specific applications for certain users while denying it to others. It also helps in the detection and control of application-layer attacks, which now constitute a larger share of contemporary threats.

Integrated Intrusion Prevention Systems (IPS): A recent enhancement to NGFW includes the integration of IPS technology, adding an extra defense layer to guard against network intrusions [27]. Unlike traditional firewalls that allow or block traffic through predefined rules, IPS actively scans and intercepts known threats embedded within traffic [27]. This feature is particularly effective against zero-day exploits and APT, which exploit unknown vulnerabilities that cannot be mitigated by standard patches [27].

Deep Packet Inspection (DPI): NGFW performs a thorough analysis of data packets as they travel across the network. This function enables the firewall to look past header information and assess the packet payload, improving its ability to recognize threats [27]. Through Deep Packet Inspection (DPI), NGFW detects malware, encrypted attacks, and other complex threats that conventional firewalls cannot address [27].

User Identity Management: The system interfaces with directory services such as Active Directory or Lightweight Directory Access Protocol (LDAP), enforcing security policies tied to individual user Identification (ID) rather than IP addresses alone [27]. This setup increases adaptability, permitting administrators to define rules that align with users' roles in the organization. This capability is especially advantageous in large global companies, where different teams may need specific access privileges for software and data [27].

Real-Time Threat Intelligence: NGFW incorporates real-time threat intelligence feeds as a standard capability. In contrast to traditional firewalls, NGFW updates its threat databases frequently, adding insights on emerging risks and offering enhanced protection for enterprises against new vulnerabilities.

After the examination of the critical capabilities of NGFW, we now present a consolidation of findings and a comparison of the attributes of conventional firewalls with the advanced features offered by NGFW.

2.2.2 Next-Generation Firewall Summary

Dehghantanha *et al.* [28] emphasize that NGFW serves as an essential component for major enterprises in modern cybersecurity architecture. With capabilities like application-layer visibility and sophisticated Deep Packet Inspection (DPI), NGFW provides elevated traffic observability, empowering organizations to detect and neutralize complex threats with greater precision. These functionalities enable enterprises to bolster their security framework and lower risk exposure by restricting uncontrolled and undesirable application traffic that seeks to access internal network systems.

Gill *et al.* [29] identify real-time Electronic Funds Transfer (EFT) systems as a domain where instant threat detection and response remain essential to protect critical information. Credit unions benefit from NGFW by ensuring that only authorized traffic passes through their networks. Real-time threat intelligence and Intrusion Prevention System (IPS) provide organizations with early warnings of potential risks, a proactive layer of defense. Additionally, implementing NGFW reduces the number of security devices needed, which improves management efficiency. Furthermore, NGFW enables identity-based policy enforcement across user interfaces and departmental access controls, which result in data security and risk mitigation strategies. Table 2.2

presents the comparative analysis of conventional and next-generation firewall based on various parameters.

Table 2.2: Comparison between Conventional and Next-Generation Firewalls

Parameters	**Conventional Firewall**	**NGFW**
Port Filtering Support	Yes	Yes
IP Address Filtering Support	Yes	Yes
Protocol-Based Filtering Support	Yes	Yes
Intrusion Prevention System (IPS) Support	No	Yes
Network Address Translation (NAT) Support	Yes	Yes
Virtual Private Network (VPN) Support	Yes	Yes
Awareness at the Application Layer	No	Yes
OSI Layers Covered	Layers 2 to 4	Layers 2 to 7
Throughput	Low	High
Visibility and Control of Application	Supports Partially	Supports Completely

Following a comprehensive evaluation of Firewall Generations, the NGFW emerges as the most advanced and effective solution to combat sophisticated DDoS attacks. We also aim to analyze the capabilities of Security Information and Event Management (SIEM) systems and their role in proactive threat detection and seamless integration with NGFW for enhanced protection. Detailed insights into this synergy appear in the subsequent section.

2.3 Security Information and Event Management (SIEM)

Our prior analysis of the critical functions of various firewall architectures established that *Next-Generation Firewall (NGFW)* exhibits significant superiority over traditional firewalls, particularly in their advanced threat detection capabilities and comprehensive protection methodologies. NGFW delivers advanced perimeter defense, actively inspecting and controlling network traffic at the edge.

In today's world, every company and business organization depends on a high-performance network for effective operation. A robust network monitoring solution is essential to monitor network activities efficiently. This software can identify irregularities in network operations, offering advantages such as more time for core activities, enhanced security, greater control, and potential cost savings [42]. Consequently, security teams must assess a constant flow of security alerts that require attention. This surge has rendered the volume of alerts difficult to manage, which calls for a prioritization process. Since no alert, whether recent or from the past six months, can be disregarded in security evaluations, addressing this workload stands as an essential, yet complex, responsibility.

Over the past two decades, countless forms of cyberattacks have surfaced. Common examples include DoS, DDoS, SQL injection (SQLi), XSS, cyber vandalism, espionage, and web session hijacking, among others [43]. A variety of security systems, including firewalls, Intrusion Detection System (IDS), and Intrusion Prevention System (IPS), have been deployed globally to counter such threats. However, these attacks often evade detection by a single security device, creating the need for a layered security approach. Security Information and Event Management (SIEM) systems provide an effective solution, capable of quickly analyzing large volumes of data and delivering insights in minimal time. SIEM enables security teams to maintain an up-to-date awareness of the organization's overall security posture [44].

To provide a unified platform for security monitoring and management, SIEM tools can integrate with various security systems, such as firewalls, IDS, and antivirus software [45]. These systems incorporate source devices, logging, data normalization, correlation engines, log repositories, and monitoring workflows to oversee and evaluate network infrastructure. Additionally, SIEM issues alerts to administrators via email or Short Message Service (SMS) whenever incidents arise [45].

The domain of Security Information and Event Management (SIEM) systems encompasses a sophisticated spectrum of commercial and open-source solutions, each designed to address distinct operational imperatives. The subsequent section reviews cutting-edge literature, assesses architectural advancements and intrinsic limitations, and identifies the most efficient, scalable, and context-aware solutions tailored for the complexities of real-world operational ecosystems.

2.3.1 Literature Review on SIEM System

Increasingly complex and frequent cyber threats have highlighted the need for robust solutions that quickly identify and address potential attacks. Significant research has examined the obstacles security teams encounter and explored methods to strengthen attack detection and response procedures.

Elradi *et al.* [46] explore the complex obstacles faced by cybersecurity professionals and introduce an integrated platform as a solution. This proposed platform combines a Security Information and Event Management (SIEM) system, a Threat Intelligence Platform (TIP), and a Security Orchestration, Automation, and Response (SOAR) system. By merging these tools, the platform enhances the speed and efficiency of threat detection, response, and mitigation.

Research by Gibadullin and Nikonorov [47] reveals that open-source solutions for automating incident management can shorten response times, reduce reliance on specialized experts, and increase automation by implementing a fault-resistant model with a distributed architecture. This report includes a literature review on available tools, examining how integration can enhance threat detection and incident response.

Tariq *et al.* [48] conducted an in-depth evaluation of prominent open-source SIEM options suited for Small and Medium-sized Enterprise (SME), mapping their security requirements to the functionalities provided by various systems. This analysis delivers valuable guidance for SME in selecting an effective SIEM solution. Essential features identified include log management, detection engines, visualization, asset identification, alerting, and reporting capabilities, with Wazuh emerging as the recommended tool for log monitoring and alerting.

Wazuh, a Host Intrusion Detection System (HIDS), serves to detect and prevent attacks targeting web servers. Stanković *et al.* [49] describe Wazuh's functions, including security analysis, intrusion detection, log review, and vulnerability assessment. Wazuh can recognize certain common web server attacks, like Secure Shell (SSH) brute force, while tracking security events, integrity changes, compliance with policies, and threat activities.

Muhammad *et al.* [50] stress the need for a cohesive network security system tailored to local environments, aiming to strengthen cybersecurity for SME. Their model combines Wazuh, Dionaea honeypot, and Cuckoo Sandbox to address the detection and control of a wide range of attacks, providing a robust approach for improved threat resilience in SME.

Suryantoro *et al.* [51] reveal the vulnerabilities of port 80 and introduce the OSCAR (Obtain Information, Strategies, Collect Evidence, Analyze, and Report) methodology, emphasizing SIEM solutions like Wazuh for cyber threat detection.

2.3.2 SIEM System Summary

Current studies underline the necessity of effective cybersecurity incident detection and response. While previous solutions contribute positively, there remains a need for integrated security tools and techniques to establish a more robust defense framework. An all-encompassing approach, incorporating SIEM, SOAR, threat intelligence, and automation in incident management, is essential for enhancing cybersecurity responses. Past research indicates that solutions like Wazuh provide key functionalities in log monitoring, incident control, and threat assessment. Hence, we intend to create a new system integrating Wazuh SIEM and Next-Generation Firewall to identify and combat cyber threats, followed by testing its efficiency in strengthening cybersecurity for network organizations. Table 2.3 summarizes our literature review on the integration and effectiveness of SIEM system types and capabilities.

Table 2.3: Literature Summary of SIEM System

No	Reference	Research Topic
1	Elradi *et al.* [46]	Introduces an integrated platform combining SIEM, TIP, and SOAR to enhance threat detection, response, and mitigation.
2	Gibadullin and Nikonorov [47]	Reveals that open-source solutions for automating incident management can reduce response times and reliance on specialized experts through a fault-resistant model.
3	Tariq *et al.* [48]	Conducted an in-depth evaluation of open-source SIEM options for SMEs, identifying Wazuh as the recommended tool for log monitoring and alerting.
4	Stanković *et al.* [49]	Describes Wazuh's capabilities in security analysis, intrusion detection, log review, and vulnerability assessment for web servers.
5	Muhammad *et al.* [50]	Presents a cohesive network security model using Wazuh, Dionaea honeypot, and Cuckoo Sandbox to strengthen SME cybersecurity.
6	Suryantoro *et al.* [51]	Highlights vulnerabilities of port 80 and introduces the OSCAR methodology, focusing on Wazuh.

After a comprehensive examination of cybersecurity tools and their feature analysis, the focus shifts to one of the most severe threats. The next section seeks to critically analyze various types of DDoS attacks and their methodologies, along with strategies for detection, mitigation, and prevention to ensure the resilience of network infrastructures.

2.4 Distributed Denial of Service (DDoS) Attack

Avizienis *et al.* [52] define confidentiality, integrity, and availability as the core tenets of computer security, with attack and defense strategies designed to undermine or preserve these principles. Gruschka *et al.* [53] characterize DDoS attacks as malicious events that hinder or prevent the normal operation of networks or services, depriving legitimate users of access. These attacks leverage weaknesses in network systems to disrupt or paralyze services. Comer *et al.* [54] note that DDoS attacks typically overwhelm systems by sending massive packets to the target infrastructure. However, such incidents may also result from software glitches, hardware failures, resource scarcity, environmental conditions, or a combination of these factors.

2.4.1 Different Phases of DDoS Attack

This section provides an overview of the three distinct phases of *Distributed Denial of Service (DDoS)* attacks and explains various attack types. The identified phases of a *Distributed Denial of Service (DDoS)* attack are as follows:

Target Selection: According to Gaurav *et al.* [55], DDoS attacks start with target selection, where attackers identify which systems or organizations to disrupt. Motives include personal grievances, attempts to gain fame by attacking high-profile servers, or involvement in cyber warfare to destabilize critical national infrastructure. Some attackers seek intellectual challenges to test their skills, while others aim for financial gain, with corporate competitors launching DDoS attacks to harm rivals' reputations and exhaust their resources.

Agent Recruitment: Gaurav *et al.* [55] outline two key strategies attackers use to expand their network of zombie machines for launching attacks. In the active approach, attackers scan networks with tools like Network Mapper (Nmap) to locate systems with known vulnerabilities, utilizing data from sources like Common Vulnerabilities and Exposures (CVE). After exploiting these weaknesses, attackers gain access to the systems and install malicious software, such as Trojan horses or buffer overflow scripts, to turn the systems into zombie machines. In contrast, the passive method uses web browser vulnerabilities by sharing harmful files or building fake websites to compromise the victim's system.

Actual attack: In this phase, the attacker utilizes compromised systems to execute an assault on the targeted system. The approach taken varies based on the victim's characteristics, as shown in Table 2.4.

Table 2.4: Different Types of DDoS Attack

Target	**Possible Attack**	**Impact**
Server (Web, Mail, Proxy)	Malformed packet attack, Flooding attack	Unavailability of system resources, System running out of memory
Server (Storage)	Flooding attack	Server Crash
Communication Bandwidth	Flooding attack	Bandwidth Depletion

2.4.2 Different Types of DDoS Attacks

Singh and Gupta [56] assert that DDoS attacks target system availability and prevent authorized users from accessing essential services. To execute such attacks, adversaries either overwhelm the victim's network bandwidth with excessive traffic or deplete system resources needed for normal functionality. These tactics categorize DDoS attacks into two core types: bandwidth saturation and

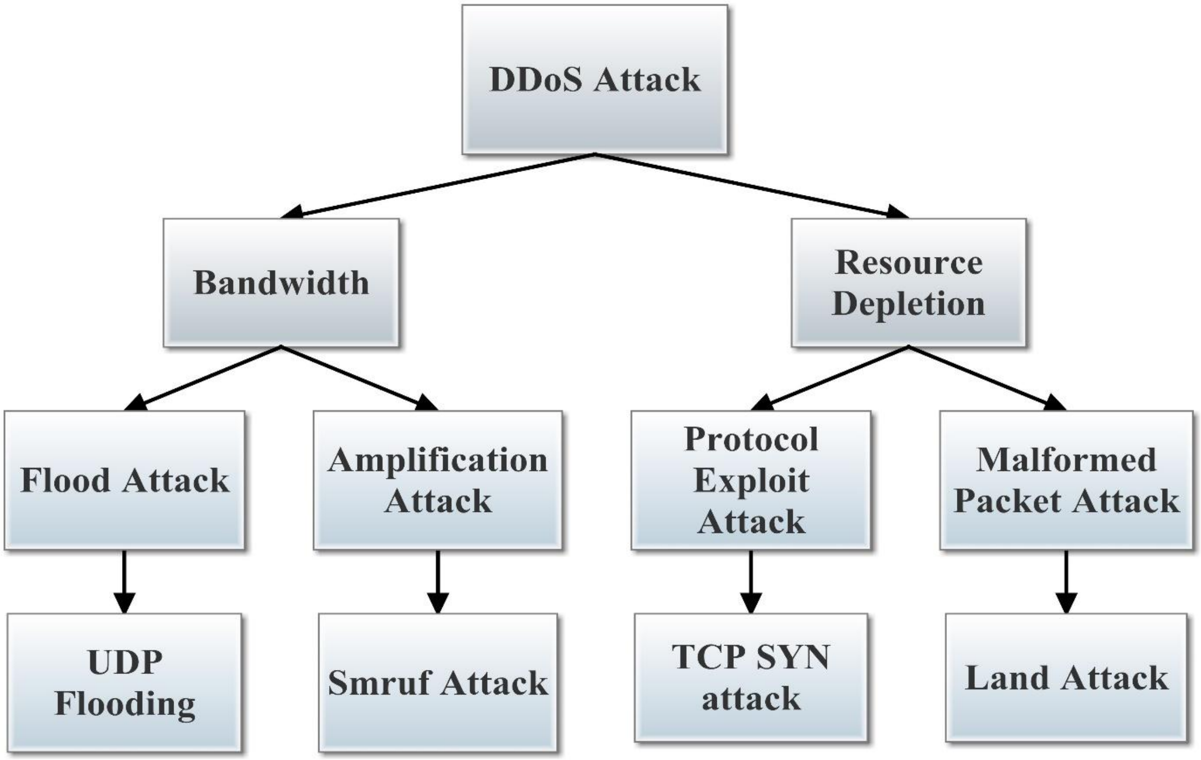


Figure 2.3: DDoS Attack Classification [56]

resource exhaustion, as depicted in Figure 2.3 [56], which elaborates on their operational differences.

2.4.2.1 Bandwidth Attack

Singh and Gupta [56] note that this attack method depends on hijacked systems, referred to as bots or zombies, to inundate the target with malicious traffic. The attack either depletes the victim's bandwidth capacity or drains its processing resources, impairing its ability to serve genuine users. These attacks are broadly divided into two categories: flood-based attacks and amplification techniques.

Flood Attack: Flooding attacks are classified into three main types: UDP, HTTP, and Internet Control Message Protocol (ICMP) floods. In a UDP flood, the attacker sends an excessive volume of UDP packets to random ports. The victim is required to process these packets and respond with ICMP "destination unreachable" messages when no application exists on the specified ports, which depletes system resources. HTTP floods utilize legitimate TCP connections to overwhelm servers with a large number of POST or GET requests, which consumes computational capacity and disrupts legitimate traffic. ICMP floods exploit the protocol by sending a high volume of ICMP ECHO REPLY packets with spoofed source addresses, forcing the victim to respond to false sources, which exhausts bandwidth and processing power.

Amplification Attack: The attack uses the ICMP protocol to exhaust the bandwidth of the victim. The attacker forges the source IP address in ICMP-ECHO packets to match the victim's and sends them across the network. Devices in the network respond to these requests, creating an excessive volume of traffic directed at the victim, which disrupts and disables the system.

2.4.2.2 Resource Depletion Attack

Singh and Gupta [56] discovered the impact of resource depletion attacks, which misuse network protocols or deliver malformed packets to exhaust a system's resources, ultimately denying access to legitimate users. Examples of such attacks include Protocol Exploitation and Malformed Packet techniques.

Protocol Exploitation Attack: Protocol exploitation attacks, such as TCP-SYN and PUSH+ACK floods, exploit vulnerabilities in the TCP handshake and buffer operations. In a TCP-SYN attack, the attacker overwhelms the handshake process by sending numerous SYN packets without completing the sequence, which causes the destination's queue to overload and denies legitimate connections. PUSH+ACK attacks target the buffer management mechanism by sending excessive Acknowledgment (ACK) packets with the PUSH flag enabled, which forces the victim to process and transmit packets prematurely. This exhausts system resources, depletes processing capacity, and impairs performance.

Malformed Packet Attack: The Ping of Death is a classic example of a Malformed Packet Attack, where attackers transmit ICMP packets exceeding the standard size of 64 bytes. These oversized packets exploit vulnerabilities in the target system and deplete its resources, resulting in service disruption. Similarly, the Teardrop Attack leverages TCP/IP fragmentation deficiencies by transmitting overlapping packet fragments. The victim system, unable to properly reassemble these fragments, exhausts its processing capabilities and becomes incapacitated, leaving no resources for legitimate traffic. The Land Attack, another sophisticated variant of a Malformed Packet Attack, generates packets with identical source and destination IP addresses. This malformed configuration overwhelms the target system's resource allocation mechanisms, culminating in critical resource exhaustion and systemic failure.

The previous sections explored the phases and taxonomy of DDoS attacks and highlighted their destructive goals and significant impact on network infrastructures. Building on this foundation, it becomes essential to analyze their implications on one of the most vulnerable and critical network architectures of Voice over Internet Protocol (VoIP).

2.5 DDoS Attack in Voice over IP (VoIP) Network

As the deadlines for the "copper switch-off" approach, with the complete transition from the traditional *Public Switched Telephone Network (PSTN)* to *Voice over Internet Protocol (VoIP)*, a new set of challenges arises, unique to this Internet-based communication system [57].

VoIP technologies, such as *Session Initiation Protocol (SIP)*, *Media Gateway Control Protocol (MGCP)*, and *H.323*, rely on an open operational model, unlike the closed architecture of traditional circuit-switched PSTN systems [58]. This openness enables attackers to execute malicious actions from remote locations, whereas similar exploits previously required physical access to the communication lines. Moreover, IP-based voice services have become primary targets for *Distributed Denial of Service (DDoS)* attacks, which were historically limited to data-centric services [58].

Despite their simplicity and low cost, *Distributed Denial of Service (DDoS)* attacks pose a significant risk to global businesses, especially those utilizing VoIP networks. Many enterprises now depend on VoIP platforms, such as Skype and Microsoft Teams [58], for daily operations, and the prevalence of remote work further amplifies the impact of such disruptions. Thus, maintaining effective and updated defenses against *Distributed Denial of Service (DDoS)* attacks is essential to counter the increasingly sophisticated methods employed by attackers.

The vulnerabilities inherent in Voice over IP (VoIP) networks expose them to an increased volume of Distributed Denial of Service (DDoS) attacks. Among the various VoIP protocols, Session Initiation Protocol (SIP) emerges as a critical target due to its central role in the establishment and management of communication sessions. By analyzing the intricacies of SIP call setup and the methods attackers employ for exploitation, we arrange to provide a comprehensive discussion of these threats and potential countermeasures.

2.5.1 DDoS Attack on SIP

The SIP call setup process, as illustrated in Figure 2.4 [59], involves a structured sequence of actions. It begins when the caller initiates the session and transmits a SIP INVITE message to the recipient to request participation in the session or call [59]. The recipient may issue various responses before granting acceptance, such as an indication that the call is queued or that an alert has been sent to the recipient [59]. Once the recipient answers, an OK message is sent to the caller, who then responds with an ACK message to complete the handshake. At this point, both parties can exchange media, such as voice or video data [59].

To conclude the session, one party sends a BYE message to the other, who confirms its receipt by sending an OK message, thereby finalizing the termination of the call [59].

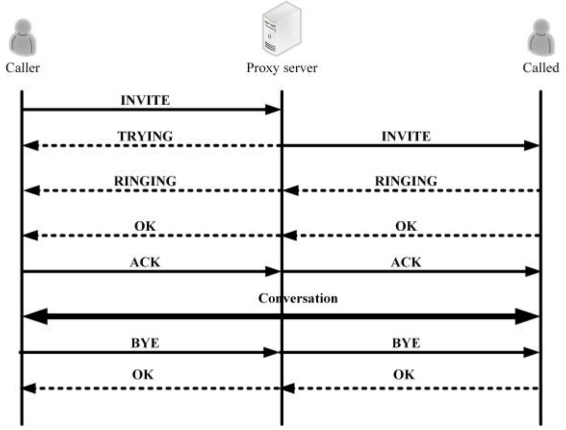


Figure 2.4: Session Initiation Protocol (SIP) call establishment [59]

A *Distributed Denial of Service (DDoS)* attack on a SIP service arises when an adversary disrupts VoIP resource availability for authorized users. Attackers frequently target the SIP server to deny subscriber access to services or impair service quality. Two prevalent methods employed in such attacks are flooding and malformed message exploits.

Flooding attacks [60] saturate SIP servers or clients with an excessive volume of SIP messages, such as INVITE, REGISTER, or BYE requests shown in Figure 2.5 [60], which exhaust critical resources like memory and processing capacity. This causes the SIP component to become inoperative for legitimate users [60]. Malformed message attacks [60], on the other hand, manipulate legitimate SIP messages to introduce errors, leading to partial system failures or forced reboots when the SIP component attempts to process the malformed input [60].

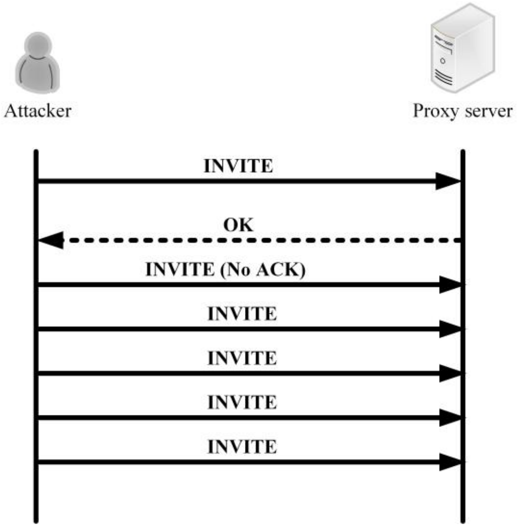


Figure 2.5: An Invite Flooding Attack [60]

2.5.2 Literature Review on DDoS Attacks in VoIP Systems

The VoIP industry faces significant threats from Distributed Denial of Service (DDoS) attacks, primarily due to vulnerabilities in the Session Initiation Protocol (SIP), as noted by Rafique *et al.* [61]. They describe flooding as the "easiest" method to execute a DDoS attack, where a SIP server becomes overwhelmed with excessive requests, which exhaust resources such as memory, Central Processing Unit (CPU), and bandwidth. This depletion results in insufficient resources to provide services to legitimate users. Mitigation strategies involve the use of stateful SIP servers and the application of message authentication techniques. Although their analysis of flooding as a DDoS attack method is comprehensive, they fail to address several other attack vectors commonly exploited in such incidents.

The significant vulnerability of VoIP systems to DDoS attacks largely stems from their reliance on UDP. While UDP minimizes packet delay and loss by not requiring a handshake, this absence of inherent authentication makes it inherently more susceptible to DDoS attacks. Kai *et al.* [62] notes that many firewalls address this issue by closing UDP ports. Similarly, Ormazabal *et al.* [63] argue that SIP's dependence on UDP exposes it to attacks like spoofing, hijacking, and tampering. While these studies do not address TCP vulnerabilities, such as TCP SYN flood attacks, the role of UDP in amplifying VoIP's exposure to DDoS attacks remains a significant concern.

The work of Sisalem *et al.* [64] delves deeply into the mechanics of DDoS attacks, analyzing how attackers exploit different server resources based on the request type. For example, an attack on the CPU forces the server to perform operations that require significant processing power, such as identity validation, application execution, or interaction with unreachable servers. Similarly, an excessive number of session initiation requests drains server memory as it tries to manage each instance, often referred to as a "brute force" attack. However, this term is inaccurate, as traditional brute-force attacks depend on systematic trial-and-error methods.

Ormazabal *et al.* [65] provide a more precise classification and define such methods as flood attacks, where attackers overload a server with excessive packet volumes beyond its capacity to process, thus disrupting legitimate traffic. Flood attacks consist of signal floods, which rely on SIP INVITE or REGISTER requests, and media floods, where open ports encounter disordered or meaningless packets.

2.5.3 Summary of DDoS Attack Reviews in VoIP System

Existing surveys on DDoS attacks reveal substantial deficiencies in scope and analytical depth. Numerous studies confine their analysis to narrow categories of DDoS attacks, which restrict the development of a more comprehensive classification framework. Although these studies overlook TCP's susceptibility to specific DDoS attacks, such as TCP SYN floods, they undeniably emphasize the significant role UDP plays in VoIP's vulnerability to such threats.

Our research aims to bridge these gaps by offering a comprehensive and technical investigation into DDoS attack types. Table 2.5 provides key insights from scholarly research, outlines the advantages and limitations of each article, and illuminates the inherent vulnerabilities within VoIP systems.

Table 2.5: Literature Summary of DDoS Attack in VoIP System

No	Reference	Advantages	Disadvantages
1	Rafique *et al.* [61]	Offers stateful SIP servers and message authentication to mitigate SIP flooding DDoS attacks.	Does not address other attack vectors beyond SIP flooding.
2	Kai *et al.* [62]	Highlights UDP vulnerability and proposes closing UDP ports to mitigate DDoS risks in VoIP.	Limited focus on TCP vulnerabilities such as TCP SYN flood attacks.
3	Ormazabal *et al.* [63]	Identifies risks from UDP in SIP systems, such as spoofing, hijacking, and tampering.	Excludes broader DDoS strategies and mitigation for other protocol vulnerabilities.
4	Sisalem *et al.* [64]	Provides detailed analysis of how different server resources (CPU, memory) are exploited based on the type of attack, such as identity validation or brute force memory exhaustion.	Inaccurate classification of "brute force" attacks as they lack traditional trial-and-error methods.
5	Ormazabal *et al.* [65]	Defines flooding attacks with precision, categorizing them into signaling floods (e.g., SIP INVITE, REGISTER) and media floods (e.g., disordered packets).	Focuses only on SIP-related floods without addressing broader attack categories or mitigation gaps.

In the next section, we present an in-depth analysis of detection mechanisms, focusing on innovative techniques and their comparative performance in mitigating DDoS attacks in VoIP structures.

2.5.4 Literature Review on DDoS Countermeasures in VoIP Systems

Various countermeasures for DDoS attacks in VoIP scenarios have been explored and implemented in recent studies. The subsequent section analyzes and compares these approaches to determine the most effective solutions.

In their study, Cadet and Fokum [66] propose a mitigation system that includes an Intrusion Protection System (IPS) that uses Snort. Their method offers simplicity in deployment and provides efficiency in the mitigation of common UDP floods, with an average detection and response time of 500 milliseconds. The system also reduces SIP server memory usage by 20% compared to traditional methods of handling attacks. However, it fails to counter more covert DDoS attacks. Another drawback lies in its reliance on static filters, which limits scalability and flexibility. The approach suggested in this paper resolves these issues by removing the need for attack detection that focuses instead on a responsive mechanism that neutralizes stealth. Additionally, the use of dynamic protocol switching creates a flexible and broadly applicable solution.

A hybrid rules-based method that combines a Handler and Bloom filter is proposed by Ganesan and Msk [67] to mitigate both high- and low-rate attacks. This approach uses a dynamic blacklist in the Handler layer, which undergoes continuous updates via the Bloom filter. It evaluates packet flow attributes to ensure accurate packet-user mapping and addresses spoofing tactics often employed by attackers. While this two-tier model demonstrates faster detection and fewer inaccuracies compared to standalone implementations of the Handler or Bloom filter, it still reports significant false positive and negative rates (1.6% and 4.13%, respectively) during tests that involve fake signaling attacks.

In their recent work, Ivy and Priya [68] introduce an improved trust-based detection and prevention framework, which expands on Cadet and Fokum [66] methodology. Rather than outright block IPs that exceed predefined thresholds, this method assigns clients a trust level that adjusts dynamically based on their request patterns. Initially set at a baseline, the trust level increases with consistent, benign activity and decreases when irregular, high-request volumes are detected. This dynamic approach minimizes risks, such as mistakenly blacklisting trusted users experiencing technical issues. However, trust-based models like those by Ganesan and Msk [67] and Ivy and Priya [68] remain vulnerable to DDoS attacks that use compromised, high-trust hosts, as such threats take longer for servers to identify and mitigate.

Tas *et al.* [69] utilizes a dynamic approach that evaluates traffic patterns across the entire network over different time intervals. Historical traffic trends are compared to current network data to determine whether activity levels fall within expected ranges. To account for variations in traffic flow, such as those caused by weekdays versus weekends, data is analyzed hourly, daily, weekly, or monthly. This method reduces the need to track individual client behavior, conserves server memory, and eliminates the misclassification of legitimate clients as attackers. In tests against a simulated SIP-based DDoS attack, this method reduced CPU load from 71% to 18% [69].

Tsiatsikes *et al.* [70] adopt a statistical method that uses entropy and log file data to address DDoS attacks on SIP servers. While this approach has a longer detection time than Cadet and Fokum [66] (1.8 to 16.8 seconds), it performs a detailed offline analysis of audit trails to identify attack patterns. Furthermore, the method preserves user privacy by applying anonymization and obfuscation to sensitive sections of the log files.

Ahmad and Singh [71] suggest the integration of an SIP proxy server with enhanced security features and an advanced application-layer stateless firewall to counter DDoS attacks on VoIP systems. Their approach relies on continuous communication between the SIP server and the firewall to maintain a trusted list of user IP addresses, which enables automatic updates to firewall rules. Additionally, although the design temporarily uses a stateless firewall, the implementation of a stateful firewall for standard operations could address limitations such as insufficient traffic and packet inspection that arise from reliance on stateless firewalls alone.

Given the effectiveness of the application of the stateful firewall to counter DDoS attacks, a comprehensive and adaptive strategy for detection emerges as a practical choice. In this context, the application of Next-Generation Firewalls offers a resilient solution against DDoS attacks in VoIP traffic scenarios. Therefore, we consider Next-Generation Firewalls in our proposed approach.

2.5.5 Summary of DDoS Countermeasures in VoIP System

This section provides a summary of the insights from the literature review on DDoS countermeasure approaches in VoIP network topologies and categorizes them in Table 2.6 based on shared characteristics and identified limitations.

Table 2.6: Literature Summary of DDoS Countermeasures in VoIP System

No	Reference	Advantages	Disadvantages
1	Cadet and Fokum [66]	Effectively stopped UDP flooding attacks in 500 ms. Saved 20% of SIP server memory.	Weak against DDoS attacks. Weak against stealthier DoS attacks, such as when the attacker uses IP masking techniques. Static filter thresholds.
2	Ganesan and Msk [67]	Uses a dynamic blacklist of malicious hosts. Aims to mitigate both high-rate and low-rate DDoS attacks. Analyses packets at the flow level for greater detail.	A significant volume of false positives and negatives. Despite the dynamic blacklist, the solution is still vulnerable to DDoS attacks if enough IP addresses are used.
3	Ivy and Priya [68]	Uses a trust level for each host, improving the responsiveness of the attack and reducing false positives/negatives caused by day-to-day latency drops. The algorithm dynamically updates based on real-time traffic analysis.	While the use of host trust levels is effective, this method is especially vulnerable to an attack using compromised trusted hosts, as this will take much longer to detect due to the already established trust levels.
4	Tas *et al.* [69]	Statistical approach, with the option of using data gathered on an hourly, daily, weekly, or monthly basis. Shown to reduce CPU load significantly	As the method is only tested on the novel SR-DRDoS attack, its effectiveness against common forms of DDoS attack is unknown.
5	Tsiatsikes *et al.* [70]	Deep offline analysis of log files. Anonymisation of users. Easy to deploy and compatible with existing SIP installations.	Longer detection time compared to other prevention mechanisms High computation requirements
6	Ahmad and Singh [71]	Targeted towards and built for VoIP networks. The trusted host list is updated dynamically. Effective against a variety of DDoS attacks.	Not backed up by experimentation. Raises risks associated with using a stateless firewall full-time due to the lack of detailed traffic and packet inspection.

Evaluating the high-impact severity of DDoS attacks establishes a critical basis for progress into comprehensive risk assessment frameworks. The subsequent section examines secure architecture for precise risk quantification and justifies the importance of proactive measures. By applying industry standards, organizations adopt structured processes to address identified vulnerabilities, optimize resource allocation, and elevate the resilience of their security infrastructures against sophisticated DDoS attacks.

2.6 Risk Assessment and Protection Measures

In the context of information security, “likelihood” refers to the probability that a security threat might exploit existing vulnerabilities, affecting an organization’s assets. This concept is crucial in risk assessment, as it provides an estimate of how often a security incident could occur within a certain period [30]. Likelihood guides risk management, helping develop targeted strategies to reduce possible threats. Recognizing likelihood allows organizations to rank risks by their probability, ensuring that resources are effectively allocated to mitigate the most significant security breaches [30].

Various standards and frameworks such as *International Organization for Standardization/ International Electrotechnical Commission (ISO/IEC 27005:2018)*, the *National Institute of Standard and Technology (NIST 800 series)*, *Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE)*, and the *European Union Information Technology Security Risk Management (EU ITSRM) method*, offer structured approaches for managing information security risks and focus on the likelihood assessment [30]. These frameworks deliver robust methodologies that empower organizations to perform systematic risk evaluation and remediation, reinforcing regulatory adherence and enhancing the integrity of their security architecture [30].

The *NIST* Special Publication *800-30* Guide for Conducting Risk Assessments states that the likelihood of occurrence is:

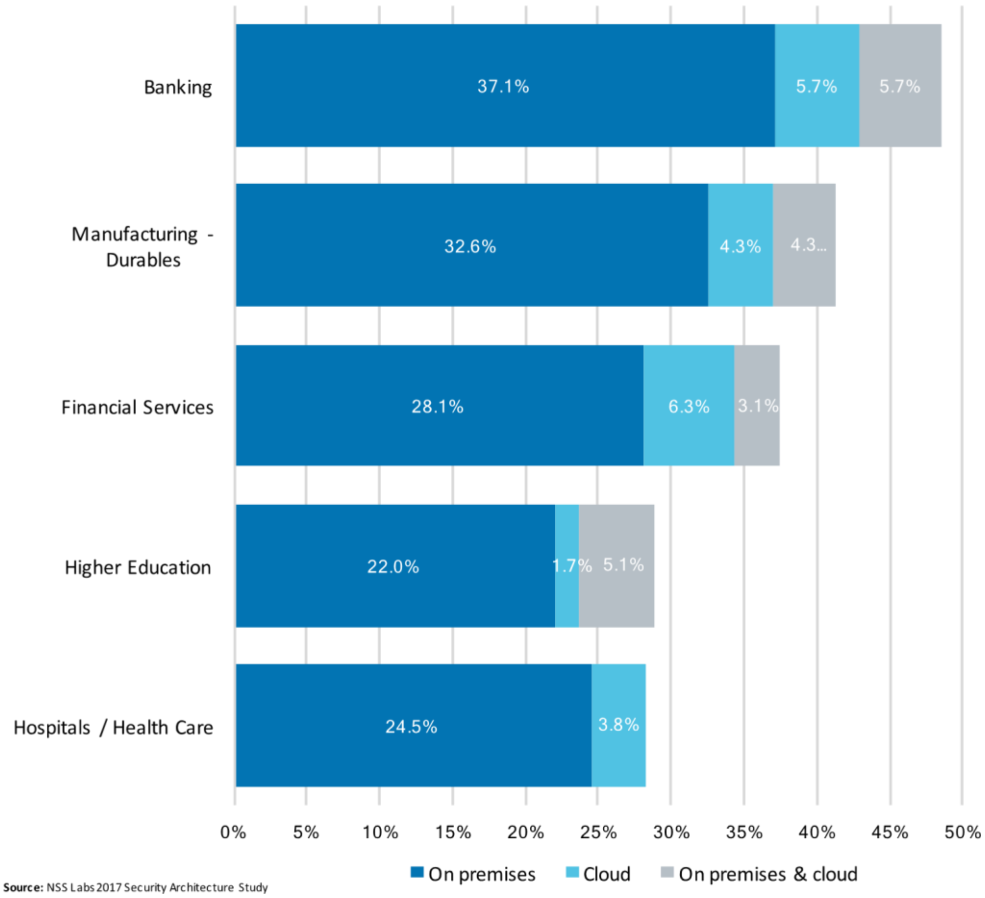
“The probability that a given threat is capable of exploiting a given vulnerability or a set of vulnerabilities.”

The *NIST 800* series on risk management, developed by the National Institute of Science and Technology, provides the U.S. government with comprehensive guidelines on security policies and procedures. Adherence to *NIST 800* standards is compulsory within the United States. The inaugural document of this series, known as the "Risk Management Framework" (RMF), was published as NIST SP 800-37 Revision 1 in 2010. This framework is structured to meet the demands of Federal Information Systems and complies with key legislation, including the Federal Information Security Management Act (FISMA) (2014), the Privacy Act of 1974, and Federal Information Processing Standards. It applies to all organizational types. [31].

The *NIST 800-37* is asset-based, and it consists of a seven-step process: (1) Prepare, (2) Categorize, (3) Select, (4) Implement, (5) Assess, (6) Authorize and (7) Monitor.

The preparation phase defines the context for risk management, while the categorization stage evaluates assets and personnel involved (risk identification). The subsequent steps—select, implement, and assess—comprise the risk treatment process. The authorization phase involves a decision from senior management on risk acceptance, and the monitoring stage focuses on ongoing risk assessment and thorough documentation [31].

Figure 2.6 [32] depicts the allocation of security infrastructures across different breach detection systems, with a significant proportion of enterprises for on-premises deployments, followed by hybrid (integrated on-premises and cloud) and exclusively cloud-based configurations [32]. This distribution underscores the inclination of larger enterprises toward on-premises solutions, driven by the need for heightened control, regulatory compliance, and data sovereignty [32]



Source: NSS Labs 2017 Security Architecture Study

Figure 2.6: Breach Detection System Deployment [32]

Given the comprehensive risk assessment and analysis of security standards, it is imperative to transition to advanced security methodologies capable of real-time detection and mitigation of DDoS attacks. As outlined in the current section, on-premises solutions, specifically Next-Generation Firewall (NGFW), represent the most practical and adaptive strategy to address the diverse needs of various network topologies and services. More importantly, Security Information and Event Management (SIEM) systems provide extensive, real-time monitoring and detailed analysis of security incidents across the entire network infrastructure. This dissertation integrates NGFW and SIEM systems to develop a robust security framework and offers a detailed technical discussion of each solution's unique protective features and functionalities in the next few sections.

2.6.1 Protection Strategies of Next-Generation Firewall

Chen *et al.* [33] justify NGFW and IPS as fundamental components of modern enterprise security architecture, serving primarily for vulnerability management. Their effectiveness in large enterprise environments depends on defensive requirements and their adaptive capacity to recognize and counter threats. The reliability of these systems incorporates multiple integrated elements and advanced technologies, enhancing their capability to address enterprise security needs.

Threat Intelligence Integration: Dörterler *et al.* [34] describe real-time threat feeds as essential components of both NGFW and IPS products, which can be implemented as either physical or virtual devices. These systems utilize global databases and advanced analytics to detect and block specific threats. Threat intelligence further empowers NGFW and IPS to uncover and neutralize threats, including zero-day vulnerabilities, by continuously updating threat databases with new Indicators of Compromise (IoC).

Nyati *et al.* [35] articulate the essential role of real-time threat intelligence within telematics systems in fleet management, emphasizing its capacity to enhance both security posture and operational continuity. This integration equips telematics systems with proactive detection and countermeasures for evolving cyber threats, thereby securing core fleet operations. NGFW and IPS leverage a rich spectrum of data ranging from external threat intelligence sources to proprietary security logs, to make split-second, risk-based decisions, determining whether to block, alert, or quarantine suspicious actions. For large-scale enterprises with complex, interconnected devices and user bases, this functionality is vital for sustaining network integrity and defending against sophisticated, multi-vector threats.

AI & Machine Learning for Threat Detection: Huang *et al.* [36] discuss the significant impact of Artificial Intelligence (AI) and Machine Learning (ML) on NGFW and IPS, as these technologies enable detailed analysis of complex network behaviors. Systems equipped with AI and ML can detect subtle changes in traffic patterns, such as spikes in unauthorized logins, unexpected data flows, or atypical access requests that may signal security threats. AI-based anomaly detection uncovers threats like insider breaches and APT that static security protocols might miss. With ML continually refining behavior models, NGFW and IPS become more sensitive to actual threats, minimizing false alarms. For organizations with large data volumes, AI and ML provide scalable, independent monitoring for strong, consistent security. Figure 2.7 [37] presents a cybersecurity machine learning workflow, starting with defining a specific security issue, such as novel threats [37]. Next, data collection and labeling generate a high-quality dataset to avoid biases. System design and learning construct a model that accurately detects relevant patterns while minimizing errors. Performance evaluation uses appropriate metrics to assess model effectiveness before deployment [37]. Finally, the model operates in real-world environments, adapting to evolving threats as a robust, learning-based Intrusion Detection System (IDS) [37].

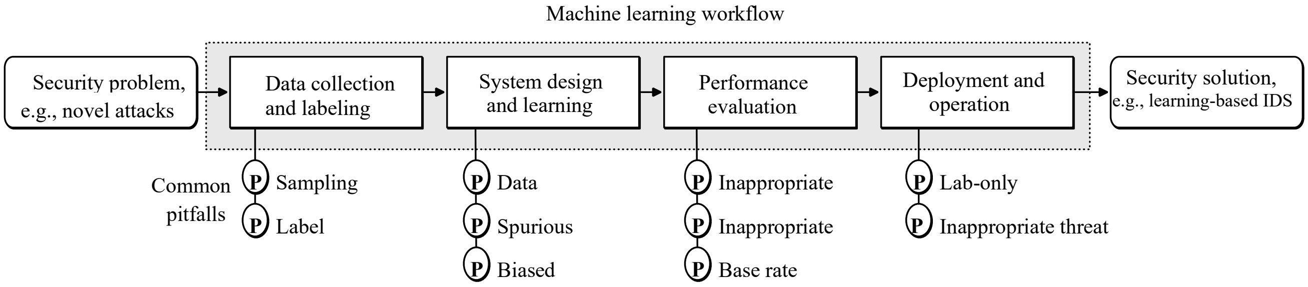


Figure 2.7: Cybersecurity Machine Learning Workflow [37]

Sandboxing and Behavioral Analysis: Iseed *et al.* [38] identify a sandbox as a critical element within NGFW and IPS, where potentially harmful code runs in a secure environment before it reaches the network. This method allows NGFW and IPS to conduct a close examination of threat behaviors by executing unknown files in isolation. Paired with behavioral analysis, the sandbox offers robust protection by observing file actions and assessing behavioral patterns across users, applications, and devices. This two-layered approach equips NGFW and IPS to detect complex threats like ransomware and covert data exfiltration, which may evade standard file analysis.

Large organizations depend on this advanced functionality to uncover threats that standard identification methods may miss. By analyzing the actions of files and various entities, an organization can lower the risk of security breaches.

Korhonen *et al.* [39] point out that the rise in encrypted traffic, especially with SSL and Transport Layer Security (TLS) protocols, poses a significant challenge for NGFW and IPS. Standard security tools cannot thoroughly inspect encrypted packets, leading to blind spots in network defense. Given that a large portion of today's traffic is encrypted, cybercriminals can take advantage of these limitations, embedding malicious actions within encrypted data streams and evading detection.

Phillips *et al.* [40] explain how advanced NGFW and IPS tackle threats through SSL/TLS inspection, where traffic is decrypted, examined for hidden threats, and re-encrypted before proceeding. This method prevents deeply embedded malicious content in encrypted broadcast streams from evading detection. However, Phillips cautions that performing SSL/TLS inspection quickly, without impacting network performance, remains a significant challenge.

Policy Management and Compliance Features: Laine *et al.* [41] note the necessity for NGFW and IPS systems in enterprise networks to meet regulatory demands, such as those set by the General Data Protection Regulation (GDPR), Health Insurance Portability and Accountability Act (HIPAA), and Payment Card Industry Data Security Standard (PCI DSS). These systems allow organizations to enforce security policies that align with legal requirements. Through policy management, NGFW and IPS control traffic, authorize specific applications, and restrict access to sensitive data. Additionally, these tools support automated compliance by ensuring that only authorized users can reach secure assets. Their built-in audit and logging functions document network activity, aiding in compliance reporting for auditors. In large enterprises, this capacity is essential to prevent penalties for non-compliance with established security regulations.

For large enterprises, the use of threat intelligence, AI-driven machine learning, sandboxing, behavioral assessment, SSL/TLS inspection, and policy controls within NGFW and IPS systems is critical for effective vulnerability management. These capabilities allow organizations to proactively identify and counter emerging threats, embedding security measures across multiple levels of the organization and development lifecycle. In this research, *Cisco Firepower Threat Defense (FTD)* functions as the Next-Generation Firewall and provides a unified security approach across the attack lifecycle, from prevention to detection to response, integrating all essential capabilities discussed.

After all the protective factors in NGFW, their lack of centralized event correlation limits the detection of multi-stage DDoS attacks. SIEM systems close this gap by the unification of data from diverse sources, which allows comprehensive event correlation, anomaly identification, and actionable insights. Together, NGFW, IPS, and SIEM systems create a cohesive architecture to address sophisticated threats and ensure enterprise-wide security visibility. The next section explores the detection strategies of SIEM systems which form the backbone of modern enterprise security.

2.6.2 Detection Strategies of SIEM System

According to Mokalled *et al.* [72], SIEM systems universally support the fundamental tasks of event collection, storage, and correlation within managed environments. Despite these commonalities, SIEM platforms vary significantly based on their market orientation. These variations are classified into three main categories: mandatory features, which comprise the system’s core functions; basic features, which are optional but enhance performance and utility; and advanced features, which integrate innovative functionalities to address emerging needs. This classification reflects the continuous evolution of SIEM technologies.

Mandatory Features:

Log Management Capability: SIEM systems store logs collected from data sources after analysis and threat detection. These stored logs act as a valuable resource for forensic investigations. The duration for retention of these logs varies from a few weeks to several years.

Log Normalization Capability: Not all logs obtained from various data sources are essential for threat detection in SIEM systems. Once collected, the logs are parsed to derive structured data and arranged in a standardized format for further analysis by the SIEM.

Correlation Engine Detection Strategy: An SIEM system aims to analyze and link events from multiple devices to detect possible security threats. If events "x" and "y" or "x," "z," and "y" coincide, it must alert the network administrator. Anomaly and rule-based detection serve as the core strategies for event correlation in SIEM systems.

Visualization Dashboards: SIEM systems offer charts, graphs, and custom visual tools to support security analysts in understanding the network's security status. These visualization platforms allow analysts to observe events, assess patterns, and recognize anomalies within the system.

Asset Discovery: The initial step to secure a network involves understanding its environment. The asset discovery feature enables the automatic identification of all devices within the network, such as endpoints, servers, firewalls, routers, and IDS/IPS systems.

Alerting and Reporting Capability: Alerts and reports in real-time enable network administrators to address attacks without delay. Timely event reports are essential for delivering actionable insights into the network's operational environment.

Number of Data Sources: Data sources represent the network nodes monitored by an SIEM system, including components like Windows and Linux operating systems, routers, switches, firewalls, servers, and databases. The scope of this feature varies based on the network scale, whether it is a small setup or a large enterprise infrastructure.

Events Per Second (EPS): The primary purpose of an event log or management system is to deal with events generated at a specific time within the system. Events Per Second (EPS) measures the capability of an SIEM system and shows how many events it processes per second.

Basic Features:

Application Monitoring: An SIEM system supervises standard data sources such as Windows and Linux operating systems, routers, switches, firewalls, databases, and servers. They detect irregularities in these assets and verify that intended actions, such as user or data additions and deletions, execute successfully without complications.

Vulnerability Scanning: A vulnerability scanner inspects the network and identifies points where security exploits may exist within the system.

Anomaly-based Correlation: Rule-based correlation forms the foundation of most open-source SIEM solutions. Although anomaly-based correlation remains optional, its integration offers significant additional benefits to the SIEM system.

Advanced Features:

User and Entity Behavior Analytics (UEBA): UEBA acts as a feature that permits SIEM systems to monitor typical actions of users and devices on the network. It detects irregularities or unusual events that arise within the system. This includes an examination of the behavior of internal employees, external contractors, and organizational partners. To support this, the SIEM system combines the management of user and application profiles with machine learning methods to identify inappropriate actions.

Security Orchestration, Automation and Response (SOAR): SOAR addresses three primary areas of security: (i) security orchestration, which manages workflows and unifies security operation components; (ii) security automation, which automates repetitive tasks and processes in security operations; and (iii) security incident response, which identifies and manages security threats and incidents. SOAR solutions serve as a complement to current SIEM systems and, when combined with tools like Threat Intelligence Platform (TIP), Endpoint Detection and Response (EDR), or Next-Generation Firewall (NGFW), create a proactive platform for early detection, prevention, and response to cybersecurity threats.

Endpoint Detection and Response (EDR): EDR tools capture data from endpoint and network events and store the information in a centralized database for examination, detection, and alert generation. A software agent deployed on the host system acts as the main component for data collection and notification delivery. Analytic tools identify actions to enhance organizational security, address internal and external threats, and ensure constant oversight of endpoints. EDR tools improve network visibility for security analysts and provide real-time responses to harmful activities.

A comparative analysis of various SIEM systems focuses on the features and capabilities detailed in section (cf. §2.3.1 & §2.3.2) [46-51]. This analysis, combined with the extensive performance examination, highlights Wazuh SIEM as a powerful solution because of its advanced features and comprehensive analytical capabilities.

Distributed Denial of Service (DDoS) attacks present a multidimensional threat landscape that demands a unified and strategic approach instead of reliance on singular solutions. Advanced security measures such as *Next-Generation Firewall (NGFW)*, Intrusion Prevention and Detection Systems (IPS/IDS), *Security Information and Event Management (SIEM)* platforms, innovations, and robust security policies, must be employed to reduce DDoS attacks. These technologies integrate with end-to-end encryption, Identity and Access Management (IAM), and real-time anomaly detection to fortify the security framework against this evolving threat. This requires defining a security architecture to meet alignment with regulatory compliance, risk management objectives, and resilience against DDoS attacks.

2.6.3 Summary of DDoS Attack Detection Solution

In Section 2.6, we reviewed different detection and mitigation approaches that are proposed against DDoS attacks. We grouped current solutions based on their primary method into two categories: *Next-Generation Firewall Protection Measures*, and *SIEM System Detection Measures*. Table 2.7 considers these DDoS attack methods depending on the detection solutions and protection metrics.

In conclusion, numerous detection and mitigation frameworks exist to address DDoS attacks. However, many of these solutions are ineffective against DDoS detection and mitigation in real-time, leading to elevated rates of DDoS attacks. This necessitates the adoption of advanced methodologies, such as the integration of Next-Generation Firewall (NGFW) with Security Information and Event Management (SIEM) systems, to meet the stringent security demands of such environments [33-41] [72].

Table 2.7: Summary of DDoS Detection and Mitigation Solutions

Detection Solution	Protection Metrics
Next-Generation Firewall (NGFW) [33-41]	Threat intelligence integration, AI & machine learning for threat detection, Sandboxing and behavioral analysis, Policy management and compliance features
Security Information and Event Management (SIEM) [72]	Log management capability, Log normalization capability, Correlation engine detection strategy, Visualization dashboards, Asset discovery, Alerting and reporting capability, Number of data sources, EPS, Application monitoring, Vulnerability scanning, Anomaly-based correlation, UEBA, SOAR, EDR

2.6.4 Advanced Defense Architecture with NGFW and SIEM Integration

A review of prior studies shows that existing research primarily relies on traditional firewalls, Intrusion Prevention System (IPS), and Intrusion Detection System (IDS) to counter DDoS attacks. However, no studies address the prevention of advanced DDoS attacks with multi-layer defense mechanisms of Next-Generation Firewall, IPS/IDS, and SIEM. This research fills this gap by proposing a secure architecture to prevent and mitigate DDoS attacks.

Putra and Surantha [73] researched to design a solution for vulnerabilities in HTTP ports, SQL Injection, and DDoS attacks on servers in internal networks. Their study used Damn Vulnerable

Web App and Vega Vulnerability Scanner tools and integrated Security Information and Event Management (SIEM), Cisco Identity Service Engine (ISE), and Intrusion Prevention System (IPS) on Cisco Firepower devices. The experiments included SQL Injection and DDoS attacks on a Windows server. The results showed that combining Cisco ISE and Security Information and Event Management (SIEM) with Cisco Firepower devices as IPS successfully blocked these attacks, which a standalone SIEM application could not achieve.

Bul'ajoul *et al.* [74] explored the deployment of a Network Intrusion Detection and Prevention System (NIDPS) that uses the Snort tool within a Novel NIDPS architecture framework. The study revealed that Snort, as a NIDPS, fails to fully detect or block attacks during high-traffic conditions or rapid data transmission. The research uses Wincap, Flooder Packet, and TCP replay tools to simulate TCP and UDP attack traffic. Results show that the integration of the Novel NIDPS architecture with a Cisco Layer 3 switch provides more effective attack detection and prevention capabilities, even under heavy or high-speed traffic that targets internal networks.

Duppa and Surantha [75] analyzed the performance of traditional Intrusion Prevention Systems (IPS) versus Next-Generation IPS (NGIPS). Their research focused on how NGIPS addresses the limitations of traditional IPS, specifically attacks that target HTTP ports or layer 7 vulnerabilities. The tests, performed with Kali Linux as the attack tool, included SQL injection and malicious site exploitation in an internal network setup. Cisco Firepower acted as the NGIPS solution. The findings showed that NGIPS with Cisco Firepower provided superior protection against SQL injection and malicious site exploits compared to traditional IPS.

Soewito and Andhika [76] analyzed the effectiveness of the Next-Generation Firewall and Security Information and Event Management (SIEM) implemented to secure smart houses and corporate networks. The research used a comparative method to test DDoS attacks, phishing, and SQL injection on both smart house and corporate networks. The experimental results concluded that the integration of Next-Generation Firewall and Security Information and Event Management (SIEM) provided superior performance in protection for smart house and corporate networks and improved the security of data communication networks against Internet-based threats.

Research by Kishan *et al.* [77] evaluates the inability of traditional firewalls to handle advanced threats such as targeted and data-centric attacks. The study provides a detailed survey of current and next-generation firewalls, with a focus on their advanced capabilities. It discusses the technologies integrated into Next-Generation Firewall (NGFW) for enhanced network security and compares their benefits to those of traditional firewalls. The paper also explores primary network security objectives, newly emerging threats, and potential approaches to protect networks.

With reference to this related work, our study adopts Cisco Firepower as the chosen Next-Generation Firewall (NGFW) and Wazuh as the Security Information and Event Management (SIEM) system for the proposed DDoS detection and mitigation framework. The following chapter provides further details about Cisco Firepower and Wazuh SIEM.

CHAPTER 3 PROPOSED ARCHITECTURE

In this chapter, we discuss the proposed secure Next-Generation Firewall architecture against DDoS attacks. Firstly, we analyze the prevalent DDoS threat vectors and their operational dynamics, which exposes their effect on network stability and highlights the weaknesses of traditional defense mechanisms. The outcomes of this analysis guide the selection of tools and techniques in the proposed architecture. This latter defines a security solution based on Cisco Firepower NGFW and Wazuh SIEM to detect, block, and respond to DDoS attacks in real time. This proposed architecture ensures alignment between the problem scope and the architectural solution that addresses the core objective of this research.

Our proposed architecture is composed of four components, as shown in Figure 3.1. The first component, *Access Network*, includes the primary ingress points to the infrastructure, with end-user devices, the Ostinato traffic generator, and the Virtual Private Client (VPC), all interconnected through a high-performance access switch. The second component, *Demilitarized Zone (DMZ) Network,* regulates controlled access to critical services exposed to internal and external traffic. It connects the web server and Wazuh Security Information and Event Management (SIEM) via a DMZ switch. The third component, *Core Network,* functions as the architectural backbone, facilitates high-speed data exchange, enforces advanced security controls, and integrates Cisco Firepower with the core router. The fourth component, *Attacker Network,* represents the adversarial environment used to simulate real-world cyber threats. It hosts the Kali Linux attacker, connected through the Internet Service Provider (ISP) router, which executes Transmission Control Protocol Synchronize (TCP SYN) flood, Internet Control Message Protocol (ICMP) flood, and Session Initiation Protocol User Datagram Protocol (SIP UDP) flood to test the infrastructure's resilience against volumetric and protocol-based attacks.

These four components operate in harmony to establish a comprehensive security framework. The Access, DMZ, and Core networks adopt a systematic approach to securing an entire system from threats by addressing the limitations of individual components and unifying them into a comprehensive defensive framework.

In the remaining, we will detail each considered component of the proposed architecture, their responsibilities, and how they are involved meeting security needs. Furthermore, we will discuss the implementation of these four components and how they can perform together as a network infrastructure within an organization.

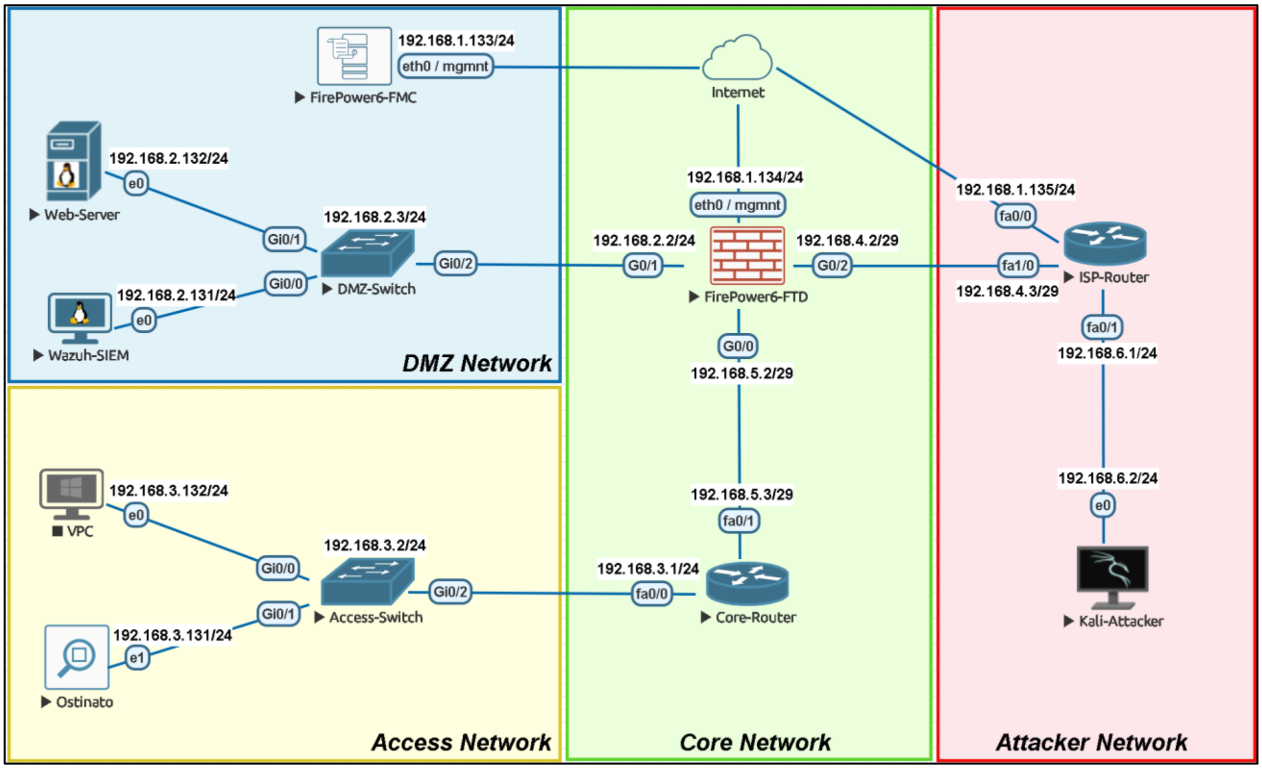


Figure 3.1: Proposed Architecture

3.1 Access Network

The architecture organizes the network into manageable divisions to enforce security policies uniformly [78]. The access network nodes act as the initial layer where endpoints, such as Ostinato and Virtual Private Client (VPC) [79], connect to the network infrastructure. This component focuses on controlled traffic generation, endpoint-level tests, and access management. It enforces strict authentication and authorization mechanisms to validate user and device credentials before it grants access to the broader network. The access network ensures segmentation from core and DMZ network components, provides a secure environment for network activities, and controls inbound and outbound traffic flows to prevent unauthorized access or malicious actions.

In the remaining, we will present the network subnets, the simulation tool, simulation system requirement, simulation installation, Ostinato traffic generator and its deployment, VPC deployment, and access switch deployment.

3.1.1 Network Subnets

We use different network subnets in our topology to increase the level of security and subdivide our networks into efficient subnetworks. Besides, we implement Hypertext Transfer Protocol (HTTP), to simulate real-world communication networks and analyze traffic behavior. Moreover, we configure static routes in the core and ISP router since it is more efficient than dynamic routing for medium networks with few paths to manage. Static routing doesn't advertise over the network, which provides better security and maintenance.

The Internet Protocol (IP) forms a key component of the Internet protocol suite, which also includes the Transmission Control Protocol, collectively referred to as TCP/IP. This suite establishes the foundational rules for packetizing, addressing, transmitting, routing, and receiving data over networks. IP addressing defines a logical framework to assign unique identifiers to devices within a network. Each device on the internet requires a distinct IP address to enable uninterrupted communication and efficient data transfer. Table 3.1 presents the IP address planning across various network components and outlines their roles within the architecture.

The previous section defined network subnets, static routes, and IP addressing. Each node received unique IP assignments to justify logical separation and communication between Access, DMZ, Core, and Attacker networks. Moving forward, the simulation phase employs Emulated Virtual Environment Next Generation (EVE-NG) to emulate virtual devices and test our proposed architecture's efficiency for the detection and mitigation of Distributed Denial of Service (DDoS) attacks in real-time.

Table 3.1: IP Address Planning

No	Host	Role	IP Address
1	FirePower-FTD	Firewall Threat Defense for real time DDoS detection and mitigation	192.168.1.134/24
2	FirePower-FMC	Firewall Management Center for FTD management and configuration	192.168.1.133/24
3	Core Router	Cisco router for connection between core network to other networks	192.168.3.1/24
4	ISP Router	Cisco router acts as as gateway between internal and external networks	192.168.1.135/24
5	DMZ Switch	Cisco switch for connection between Web server and Wazuh SIEM	192.168.2.3/24
6	Access Switch	Cisco switch for connection between Virtual PC and Ostinato	192.168.3.2/24
7	Wazuh SIEM	SIEM for event correlation, intrusion detection, log analysis and file integrity	192.168.2.131/24
8	Web Server	Linux Tiny Core serves as a web server within DMZ network	192.168.2.132/24
9	Virtual PC (VPC)	Windows host acts as a endpoint node	192.168.3.132/24
10	Ostinato	Traffic Generator for SIP traffic generation	192.168.3.131/24
11	Kali Attacker	Kali Linux emulates DDoS traffic	192.168.6.2/24

3.1.2 Simulation Tool

The host machine of our methodology runs the simulation environment on a Windows 11 Pro operating system with 32 Gigabytes (GB) of Random-Access Memory (RAM) and a 2.6 Gigahertz (GHz) Intel core i7 processor. For simulation applications, EVE-NG version 6.2.0-3-Community has been utilized to emulate diverse network architectures and complex scenarios.

EVE-NG serves as a powerful network emulation platform that allows the design and simulation of virtual networks through the integration of a wide range of Virtual Machine (VM) and network devices [84]. It addresses the needs of network professionals, IT learners, and educators who aim to replicate complex network scenarios for purposes, such as testing, skill enhancement, and certification preparation [84].

EVE-NG provides a user-friendly graphical interface that simplifies the creation of virtual network topologies and the deployment of virtualized devices [84]. It contains built-in modules that allow users to integrate network services, such as Dynamic Host Configuration Protocol (DHCP), DNS, and web servers, into simulated networks [84]. The free Community Edition offers fundamental features, supports basic virtual network environments, works with various virtual devices, and grants access to the community forum for troubleshooting and advice [84]. On the other hand, the PRO edition involves enterprise users and network professionals which meets advanced and complex simulation requirements [84].

The decision to use EVE-NG as a simulation platform derives from its superior capability to emulate various network devices including routers, switches, servers, workstations, and security appliances. This versatility makes EVE-NG the ideal tool to model realistic network topologies in multi-platform scenarios that require dynamic and complex path selection. In addition, EVE-NG can support devices from multiple vendors and platforms, a key factor in this research.

The initial installation and configuration phase of EVE-NG involves detailed hardware and software adjustments to ensure compatibility and optimal performance. This phase is critical, as proper configuration at the outset determines the success of subsequent network simulations. Once the core infrastructure is in place, the initial software setup follows to ensure that the emulator remains fully operational and ready for various network test scenarios.

3.1.3 Simulation System Requirement

EVE-NG operates under the requirement of an Intel CPU that supports VT-x/EPT features. The software functions on Ubuntu Xenial Xerus 16.04.X LTS 64bit (recommended with any processor), VMware ESXi 6.0 or later, VMware Workstation 12.5 or later, VMware Fusion 8 or later, VMware Player 12.5 or later, or a Google Cloud platform VM [84]. We deploy VMware Workstation 17 Pro as the virtualization platform to host our EVE-NG environment. This platform provides advanced hypervisor capabilities that ensure optimal performance, resource allocation, and seamless integration of virtualized network components.

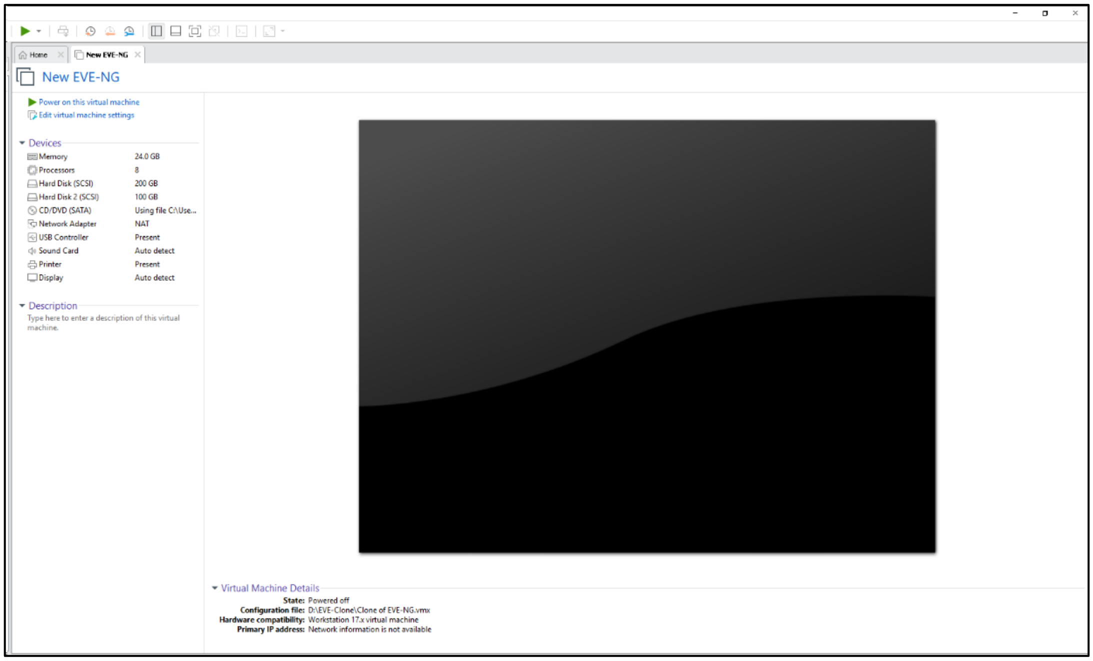
3.1.4 Simulation Installation

Within the EVE-NG Web interface, we can easily create and modify network topologies. There exist different approaches for the EVE-NG installation process, and we implement them with reference to its official cookbook [84] as follows:

1- We Obtain the EVE-NG Community Edition ISO file from the official EVE-NG website.

2- The VM is created and configured to meet EVE-NG's operational requirements. We launch VMware Workstation on a personal laptop with 32 GB of RAM and a 2.6 GHz Intel core i7 processor and select the Custom configuration option for precise resource allocation.

3- We choose the operating system as Ubuntu 64-bit, as EVE-NG runs on an Ubuntu server base. Also, we assign sufficient CPU resources of 8 vCPUs with VT-x hardware virtualization enabled, as this ensures support for nested virtualization. We allocate 24 GB of RAM for handling resource-intensive network topologies. We configure the storage through the creation of a virtual disk with 300 GB capacity to ensure the disk type is set to thin provisioning to save space or thick provisioning for performance. For networking, we use the appropriate adapter: NAT mode for our scenario. Finally, we confirm the configuration and save the VM setup, as shown in Figure 3.2.



Devices

Memory	24.0 GB
Processors	8
Hard Disk (SCSI)	200 GB
Hard Disk 2 (SCSI)	100 GB
CD/DVD (SATA)	Using file C:\Use...
Network Adapter	NAT
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

Description

Type here to enter a description of this virtual machine.

Virtual Machine Details

State:	Powered off
Configuration file:	D:\EVE-Clone\Clone of EVE-NG.vmx
Hardware compatibility:	Workstation 17.x virtual machine
Primary IP address:	Network information is not available

Figure 3.2: EVE-NG Resource Allocation

4- The EVE-NG Community Edition ISO file is mounted to the virtual machine. We begin by selecting the newly created VM and attaching the downloaded EVE-NG ISO as the bootable media. We power on the virtual machine and boot it into the ISO. Upon startup, the EVE-NG installer menu appears, where we select the Install EVE-NG Community option to initiate the installation process. Upon successful completion of the installation process, the system reboots and finalizes the operating system and EVE-NG software setup.

5- After EVE-NG software setup, we verify connectivity to the EVE-NG management interface through the Chrome browser and enter the IP Address: http://<192.168.1.50>. The system prompts for credentials, with “admin” as the default username and “eve” as the password. This completes the installation process and prepares the EVE-NG environment for virtual image uploads and network topology simulations. Figure 3.3 shows the EVE-NG login page after successful installation.

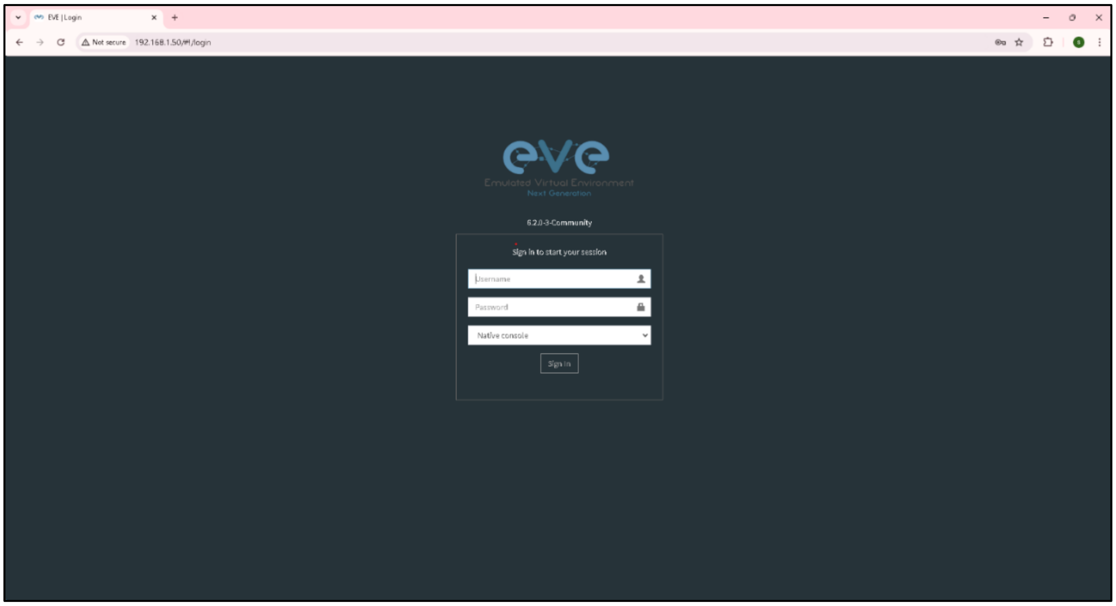


Figure 3.3: EVE-NG Login Page

6- Upon initialization of a project in EVE-NG, we access a comprehensive menu interface that provides essential tools for the management and configuration of our network simulation components. The Add Object option allows us to add new virtualized network devices and networks into our architecture. The Nodes option configures device parameters, such as hostname, IP addresses, and interface properties, to match our scenario requirements. The Networks option defines virtual networks, such as subnet masks to establish connectivity across the Access, DMZ, and Core networks. Through Startup Configuration, we save, retrieve, and modify the initial configurations of devices for consistent deployment. The Images feature customizes device icons, while Configured Objects controls device properties. Figure 3.4 depicts our proposed architecture as a project after configurations in the EVE-NG environment.

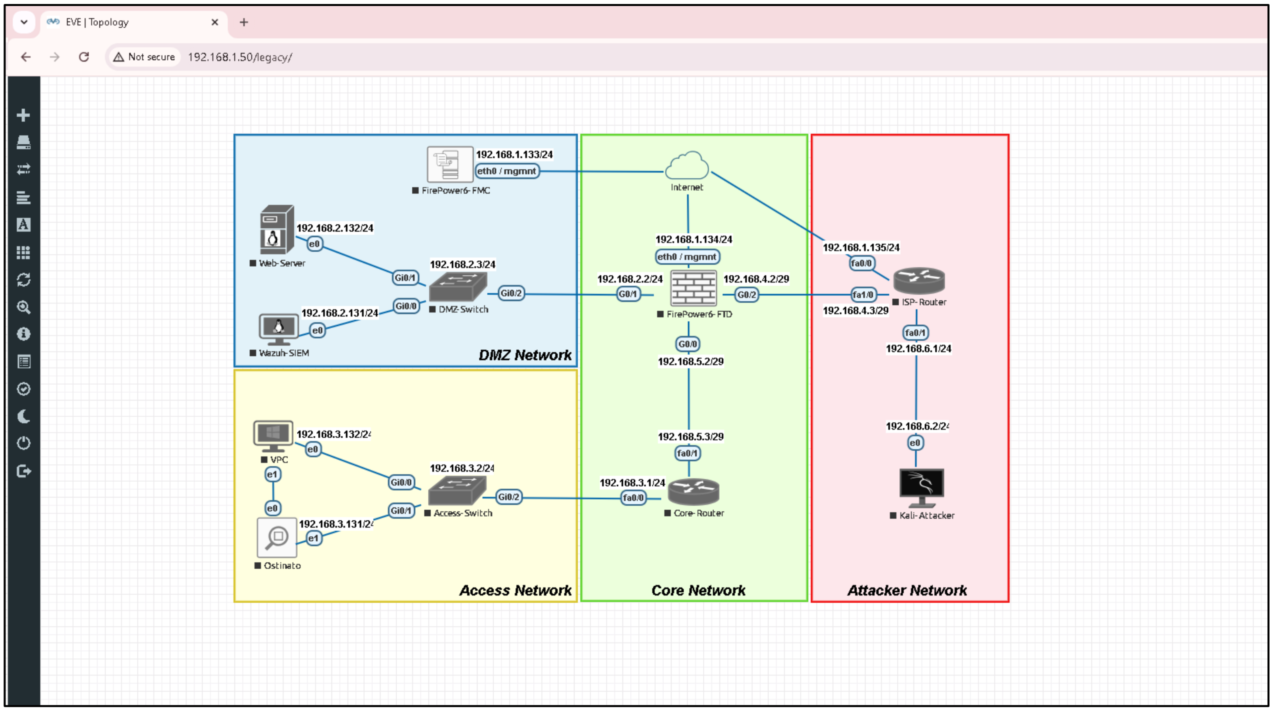


Figure 3.4: Project Architecture in EVE-NG

With the simulation environment configured in EVE-NG, we now advance to the setup of essential network components, with the Ostinato traffic generator as the initial point.

3.1.5 Ostinato Traffic Generator

Ostinato is an open-source tool designed for network traffic generation such as HTTP and SIP, packet modification, and analysis. The platform also includes a Python API for task automation [83]. Users can configure and send packet streams with customized protocol fields through Ethernet or wireless networks [83]. Ostinato, deployed within the Access Network of our architecture, serves as a sophisticated traffic generator capable of the generation of SIP traffic. This allows the comprehensive analysis of how well Cisco Firepower and Wazuh SIEM accurately uncover and neutralize DDoS attempts directed at VoIP services.

Figure 3.5 [83] illustrates the architecture of Ostinato, structured around two key binaries: the controller and the drone. The controller serves as the GUI interface that manages the drone or executes tasks through a Python script. The drone connects directly to the Device Under Test (DUT) and handles traffic creation and data capture [83].

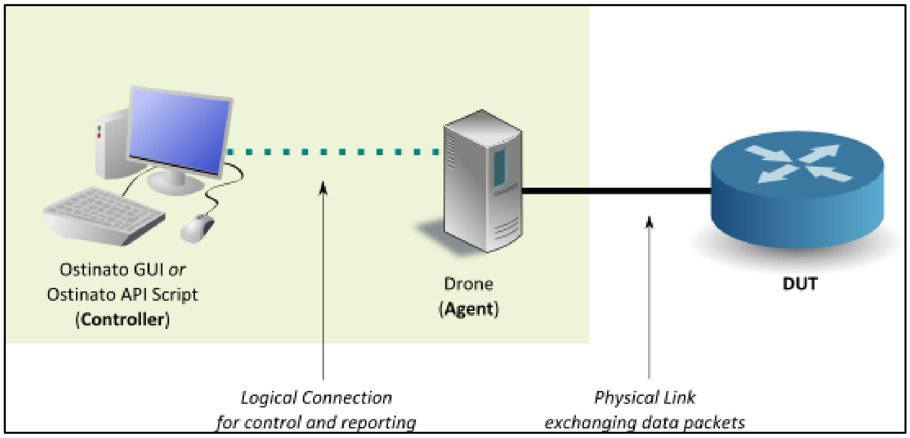


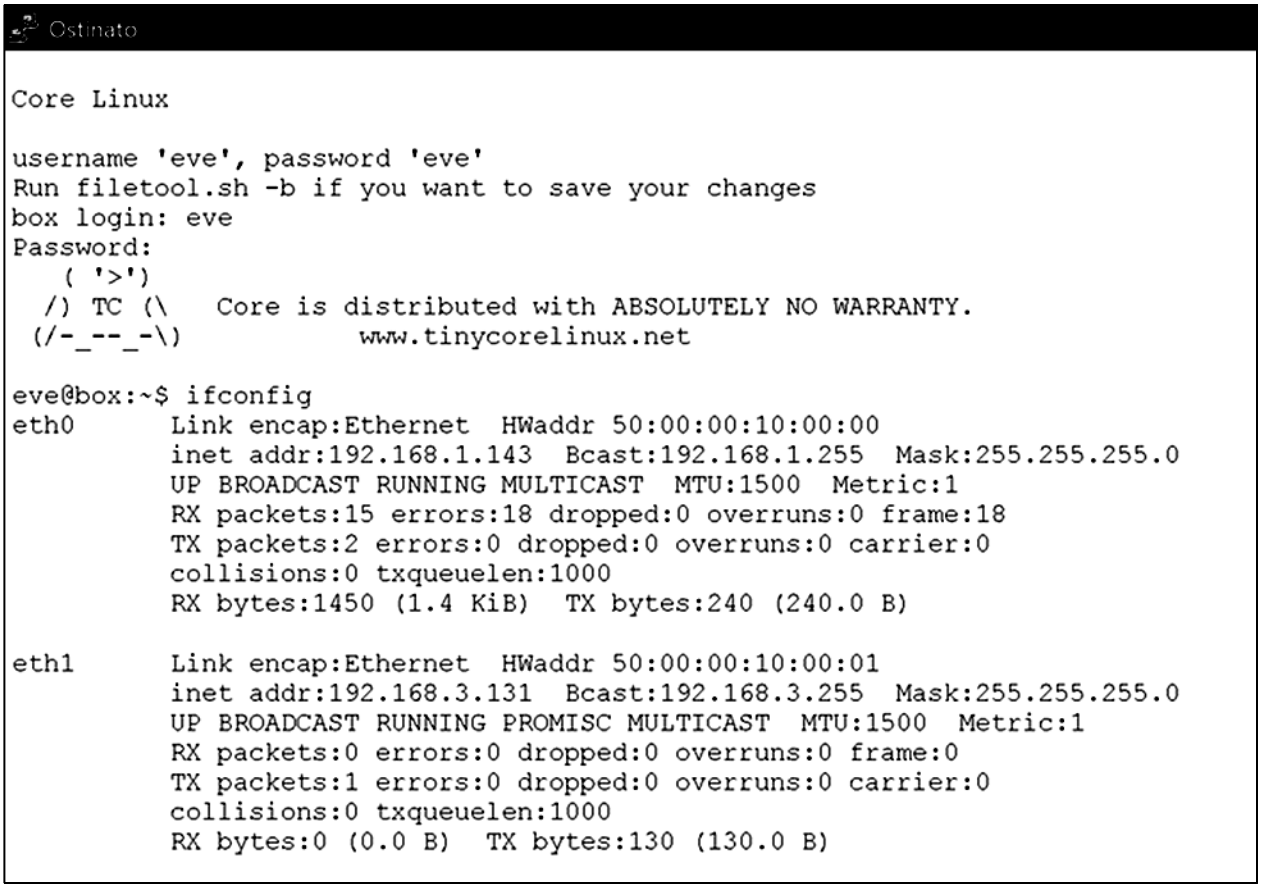
Figure 3.5: Ostinato Architecture [83]

3.1.6 Ostinato Traffic Generator Deployment

Ostinato is an open-source, cross-platform network packet crafter/traffic generator and analyzer with a friendly GUI [83]. Craft and send packets of several streams with different protocols at different rates. Ostinato aims to be "Wireshark in Reverse" [83] and become complementary to Wireshark. Ostinato comprises two main components the Ostinato GUI console and the Drone [83], which functions as a probe-like process. The Ostinato GUI allows users to define detailed packet specifications, while the Drone operates with elevated privileges, directly interfacing with the machine's Network Interface Card (NIC) to transmit packets upon user initiation [83].

1- We download the Ostinato installer for the Windows 10 Pro version and complete the installation process on the host machine within the Access Network. After launching Ostinato, we verify that the application detects the correct network interfaces connected to the Access Switch in our architecture.

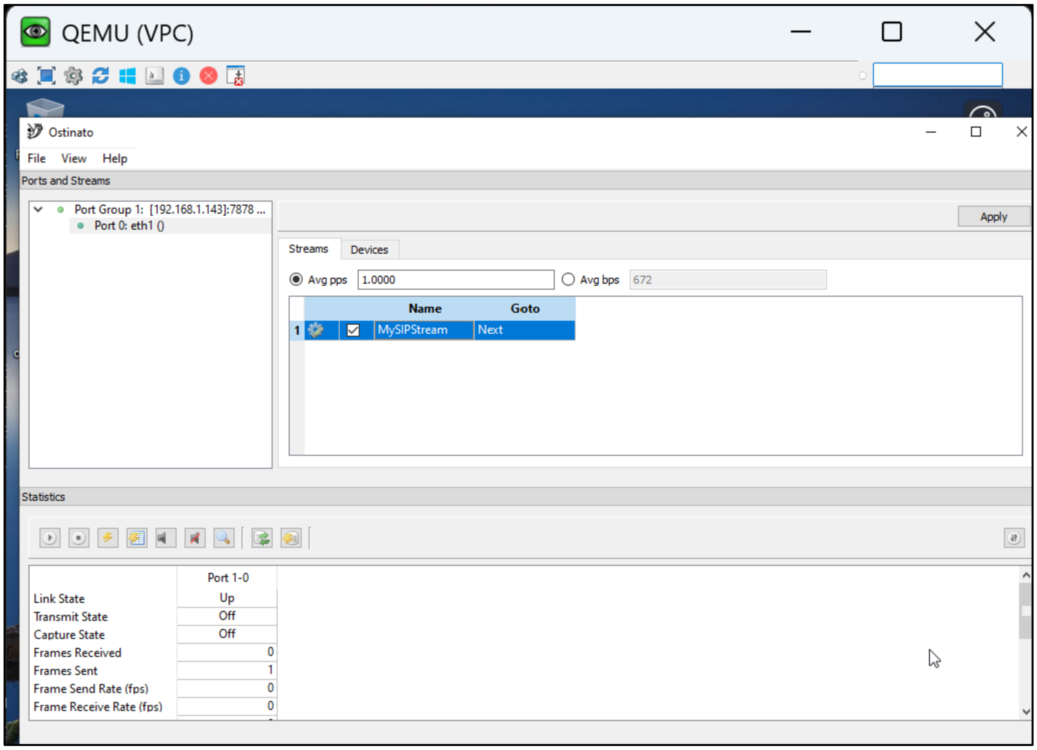
2- We configure the Ostinato traffic generator with two network interfaces, as shown in Figure 3.6. The eth0 interface has the IP address 192.168.1.143 with a subnet mask of 255.255.255.0 and serves the Ostinato GUI. The eth1 interface has the IP address 192.168.3.131 and a subnet mask of 255.255.255.0 and connects it to the access switch for traffic injection and tests as the Drone performs.



```
Core Linux

username 'eve', password 'eve'
Run filetool.sh -b if you want to save your changes
box login: eve
Password:
   ( '>')
  /) TC (\   Core is distributed with ABSOLUTELY NO WARRANTY.
 (/-_--_-\)           www.tinycorelinux.net

eve@box:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 50:00:00:10:00:00
          inet addr:192.168.1.143  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:15 errors:18 dropped:0 overruns:0 frame:18
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1450 (1.4 KiB)  TX bytes:240 (240.0 B)

eth1      Link encap:Ethernet  HWaddr 50:00:00:10:00:01
          inet addr:192.168.3.131  Bcast:192.168.3.255  Mask:255.255.255.0
          UP BROADCAST RUNNING PROMISC MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:130 (130.0 B)
```

Figure 3.6: Ostinato IP Address Configurations

3- We create a new traffic stream by right-clicking on the selected interface and choosing the Add Stream option in the Ostinato GUI. We assign a name to the stream, such as "MySIPStream," to ensure clarity in identification during tests. After we create the stream, we access its configuration menu to define the packet structure, headers, and payloads required for SIP traffic, as shown in Figure 3.7.



Port Group 1: [192.168.1.143]:7878 ...
Port 0: eth1 ()

● Avg pps 1.0000 ○ Avg bps 672

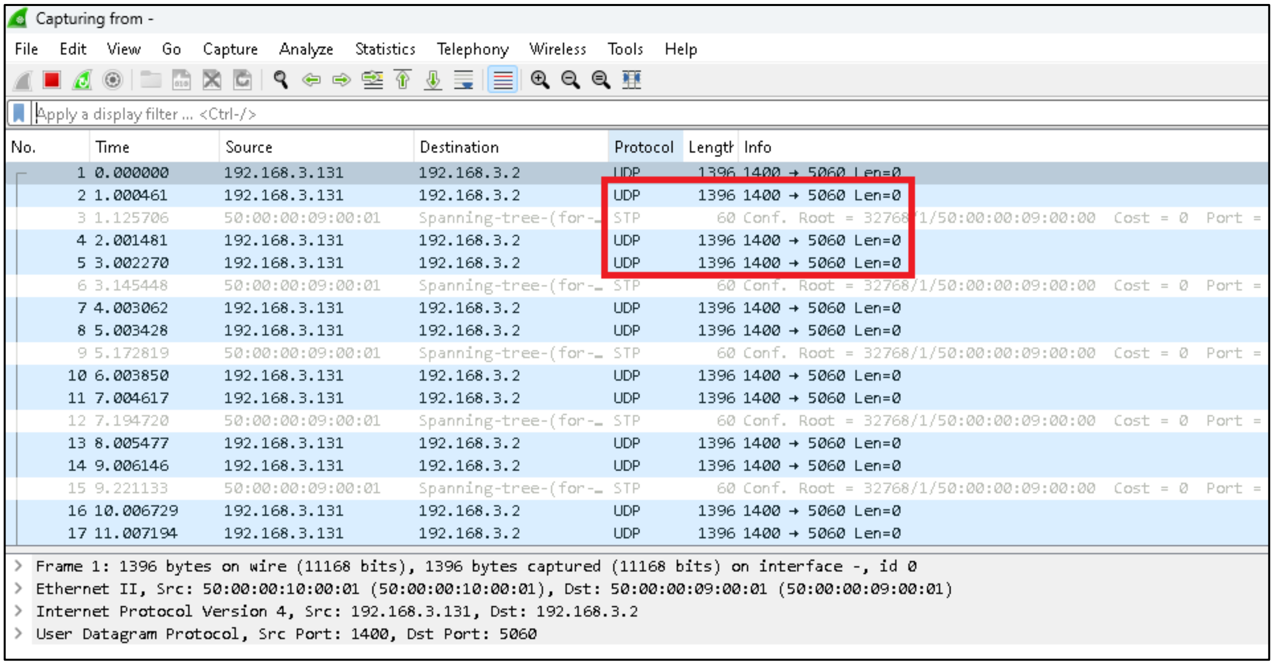
		Name	Goto
1	☑	MySIPStream	Next

	Port 1-0
Link State	Up
Transmit State	Off
Capture State	Off
Frames Received	0
Frames Sent	1
Frame Send Rate (fps)	0
Frame Receive Rate (fps)	0

Figure 3.7: Ostinato GUI Configurations

4- We select the Packet Builder option within the stream configuration menu. We define the protocol layers by adding Ethernet, IP, UDP, and SIP headers sequentially. We specify the source and destination IP addresses to reflect the endpoints in our VoIP scenario. For the SIP header, we configure fields such as From, To, Call-ID, and Request-URI to simulate an actual SIP session. Once all fields are set, we save the packet structure to finalize the stream setup.

5- We define the traffic parameters, such as the transmission rate, packet size, and destination MAC address. We select the desired transmission mode, whether continuous or limited to a specific packet count. After setting the parameters, we click "Start" to initiate the traffic flow. We use network tools like Wireshark to validate the accuracy of the generated SIP packets, as shown in Figure 3.8.

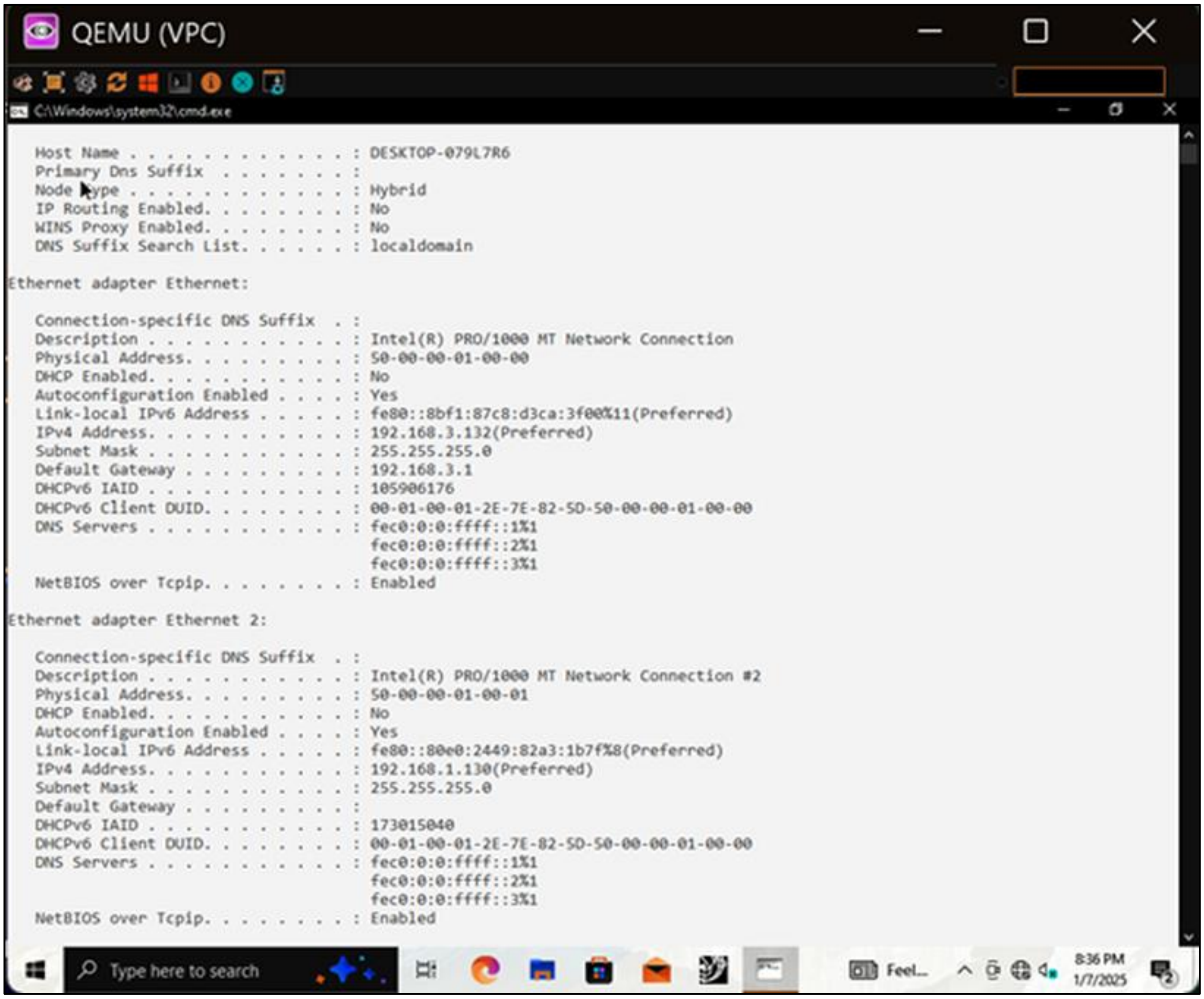

Capturing from -
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.   Time        Source              Destination          Protocol Length Info
    1 0.000000    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
    2 1.000461    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
    3 1.125706    50:00:00:09:00:01   Spanning-tree-(for-… STP        60 Conf. Root = 32768/1/50:00:00:09:00:00  Cost = 0  Port =
    4 2.001481    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
    5 3.002270    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
    6 3.145448    50:00:00:09:00:01   Spanning-tree-(for-… STP        60 Conf. Root = 32768/1/50:00:00:09:00:00  Cost = 0  Port =
    7 4.003062    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
    8 5.003428    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
    9 5.172819    50:00:00:09:00:01   Spanning-tree-(for-… STP        60 Conf. Root = 32768/1/50:00:00:09:00:00  Cost = 0  Port =
   10 6.003850    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
   11 7.004617    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
   12 7.194720    50:00:00:09:00:01   Spanning-tree-(for-… STP        60 Conf. Root = 32768/1/50:00:00:09:00:00  Cost = 0  Port =
   13 8.005477    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
   14 9.006146    192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
   15 9.221133    50:00:00:09:00:01   Spanning-tree-(for-… STP        60 Conf. Root = 32768/1/50:00:00:09:00:00  Cost = 0  Port =
   16 10.006729   192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0
   17 11.007194   192.168.3.131       192.168.3.2          UDP      1396 1400 → 5060 Len=0

> Frame 1: 1396 bytes on wire (11168 bits), 1396 bytes captured (11168 bits) on interface -, id 0
> Ethernet II, Src: 50:00:00:10:00:01 (50:00:00:10:00:01), Dst: 50:00:00:09:00:01 (50:00:00:09:00:01)
> Internet Protocol Version 4, Src: 192.168.3.131, Dst: 192.168.3.2
> User Datagram Protocol, Src Port: 1400, Dst Port: 5060
```

Figure 3.8: SIP Traffic Generation by Ostinato

3.1.7 Virtual PC Deployment

The VPC has two Ethernet adapters configured with unique IPv4 addresses 192.168.3.132 for Ethernet 1 and 192.168.1.130 for Ethernet 2, each with a subnet mask of 255.255.255.0. The default gateway is set as 192.168.3.1, which provides connection to other networks as illustrated in Figure 3.9. Ethernet 1 is connected to the Ostinato traffic generator to enable communication between the Ostinato controller and Drone agent.

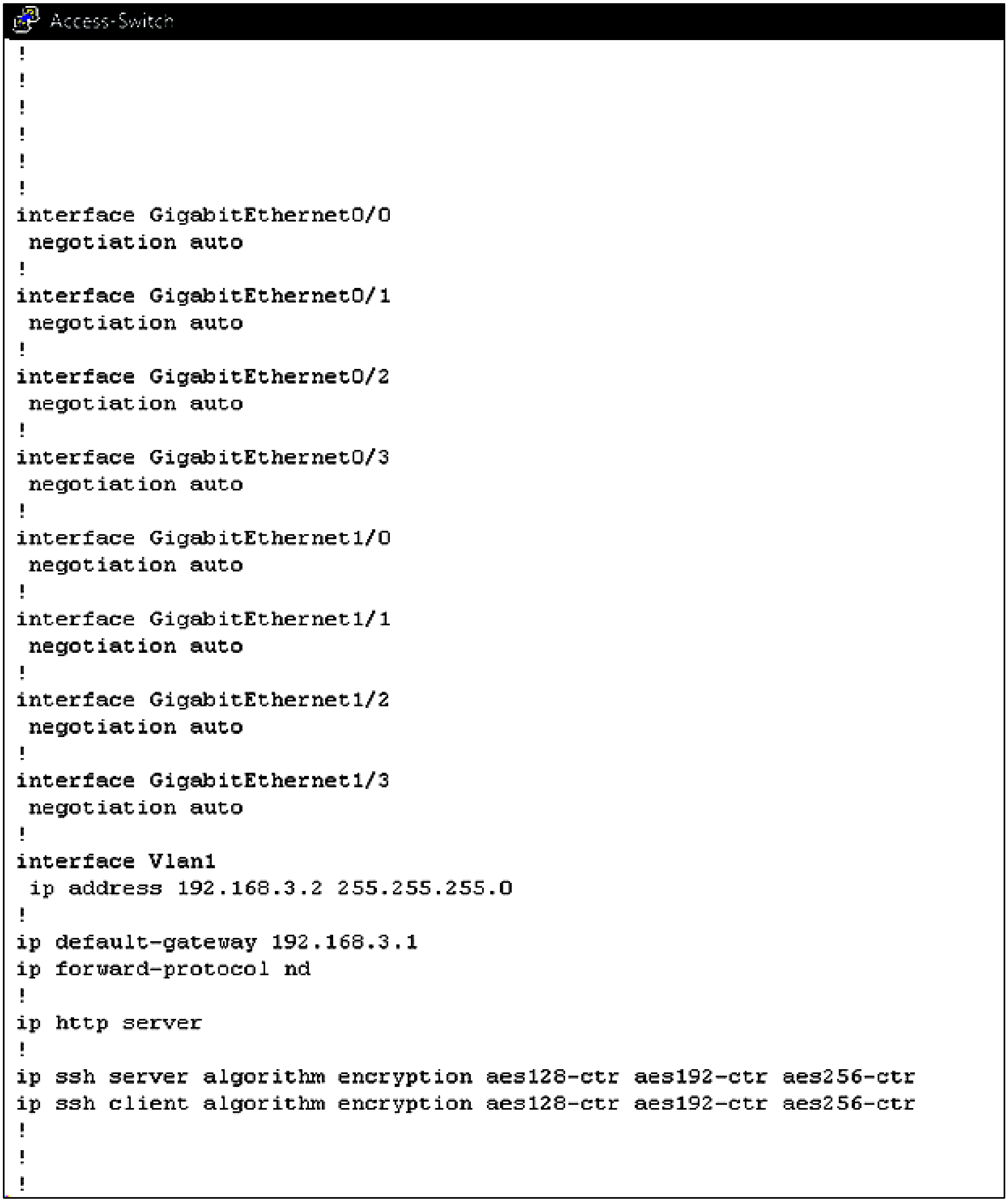


```
   Host Name . . . . . . . . . . . . : DESKTOP-079L7R6
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : localdomain

Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) PRO/1000 MT Network Connection
   Physical Address. . . . . . . . . : 50-00-00-01-00-00
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::8bf1:87c8:d3ca:3f00%11(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.3.132(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.3.1
   DHCPv6 IAID . . . . . . . . . . . : 105906176
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2E-7E-82-5D-50-00-00-01-00-00
   DNS Servers . . . . . . . . . . . : fec0:0:0:ffff::1%1
                                       fec0:0:0:ffff::2%1
                                       fec0:0:0:ffff::3%1
   NetBIOS over Tcpip. . . . . . . . : Enabled

Ethernet adapter Ethernet 2:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) PRO/1000 MT Network Connection #2
   Physical Address. . . . . . . . . : 50-00-00-01-00-01
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::80e0:2449:82a3:1b7f%8(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.1.130(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :
   DHCPv6 IAID . . . . . . . . . . . : 173015040
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2E-7E-82-5D-50-00-00-01-00-00
   DNS Servers . . . . . . . . . . . : fec0:0:0:ffff::1%1
                                       fec0:0:0:ffff::2%1
                                       fec0:0:0:ffff::3%1
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

Figure 3.9: Virtual PC Interface Configurations

3.1.8 Access Switch Deployment

We configure the Access Switch to facilitate communication between devices in the Access Network. Each Gigabit Ethernet interface uses auto-negotiation to optimize link establishment with connected endpoints. VLAN 1 is assigned the IP address 192.168.3.2/24, with a default gateway set to 192.168.3.1. These settings ensure reliable and secure connectivity for network components, such as VPC and traffic generators, within the Access Network. All the configuration parameters are shown in Figure 3.10.



```
!
!
!
!
!
!
interface GigabitEthernet0/0
 negotiation auto
!
interface GigabitEthernet0/1
 negotiation auto
!
interface GigabitEthernet0/2
 negotiation auto
!
interface GigabitEthernet0/3
 negotiation auto
!
interface GigabitEthernet1/0
 negotiation auto
!
interface GigabitEthernet1/1
 negotiation auto
!
interface GigabitEthernet1/2
 negotiation auto
!
interface GigabitEthernet1/3
 negotiation auto
!
interface Vlan1
 ip address 192.168.3.2 255.255.255.0
!
ip default-gateway 192.168.3.1
ip forward-protocol nd
!
ip http server
!
ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
!
!
!
```

Figure 3.10: Access Switch Configuration

After completing the Ostinato traffic generator, VPC, and switch configurations to establish efficient traffic pathways and interconnectivity between network components, we transition to setting up the DMZ network.

3.2 Demilitarized Zone (DMZ) Network

The DMZ, also referred to as the isolation network, performs as a buffer between secure internal networks and less secure external networks. It resolves the issue of external access to internal servers after firewall deployment, which often complicates the provision of public-facing services like Web servers [79]. Positioned between the internal and external networks, the DMZ contains publicly accessible server resources, such as corporate web servers. By applying a DMZ, the internal network gains an additional security layer to add a barrier that protects against potential threats beyond what Next-Generation firewalls provide [79]. This architecture forces attackers to bypass multiple layers of security, such as external routers, and internal shield routers before they access the internal network [79]. In this section, we present Wazuh SIEM and its deployment, Wazuh indexer, Wazuh server, Wazuh dashboard, web server and DMZ switch deployment.

3.2.1 Wazuh SIEM

Wazuh is an open-source security solution which facilitates security event correlation, intrusion detection, log analysis, and file integrity validation in our scenario. Based on Open-Source Host-based Intrusion Detection System (OSSEC), a Host-Based Intrusion Detection System (HIDS), Wazuh extends OSSEC's capabilities by integrating centralized management, an enhanced web interface, and support for multiple integrations [82]. The platform comprises the following components:

Wazuh-Server: The Wazuh-Server handles the collection and analysis of security data from various sources, such as agents, syslog, and logs from different platforms [82].

Wazuh Agents: Configured on endpoints such as laptops, desktops, servers, or virtual machines, it delivers capabilities for prevention, detection, and response [82].

Wazuh-Indexer: The Wazuh Indexer accepts input from Wazuh agents and other integrated systems, organizes it into manageable formats, and indexes the data for efficient analysis [82]. This functionality ensures the data remains accessible, secure, and ready for advanced threat detection and enterprise-wide monitoring [82].

Fluent-Bit: Wazuh works alongside Fluent Bit, a minimal yet powerful utility for log collection and transfer. It retrieves log data from multiple sources on monitored endpoints and directs it to the Wazuh-Manager or Graylog [82].

Graylog: Graylog provides centralized log management that facilitates data collection, processing, and storage. When integrated with Wazuh, it replaces the Wazuh-Manager and Indexer to provide greater flexibility and scalability for managing large log data volumes [82].

The Wazuh architecture relies on agents deployed on monitored endpoints to transmit security data to a central server [82]. The central server processes and analyzes the received data before sending it to the Wazuh indexer for storage and retrieval [82]. The Wazuh indexer cluster is a collection of one or more nodes that communicate with each other to perform read and write operations on indices [82]. Figure 3.11 [82] represents a Wazuh deployment architecture. It shows the solution components and how the Wazuh server and the Wazuh indexer nodes can be configured as clusters, providing load balancing and high availability [82].

The Wazuh agent transmits events to the Wazuh server to enable analysis and threat identification. For this process, the agent connects to the server service, which by default listens on port 1514, though this can be configured [82]. The Wazuh server decodes the incoming events and evaluates them against predefined rules using its analysis engine. Events matching a rule are enriched with additional information, such as the rule ID and name, to generate alerts [82].

The Wazuh server utilizes Filebeat to secure the transmission of alerts and event data to the Wazuh indexer, employing TLS encryption. Filebeat tracks the output data from the Wazuh server and forwards it to the indexer, which listens on port 9200/TCP by default [82]. After indexing, the data becomes available for analysis and visualization through the Wazuh dashboard [82].

The Vulnerability Detection module updates the inventory of vulnerabilities and generates alerts to provide insights into potential system weaknesses [82].

The Wazuh dashboard interacts with the Wazuh RESTful Application Programming Interface (API) (listening on port 55000/TCP by default) to retrieve configuration and status information about the Wazuh server and agents [82]. Moreover, it also enables the modification of agent or server configurations via API calls, with all communications secured by TLS encryption and authenticated through username and password credentials [82].

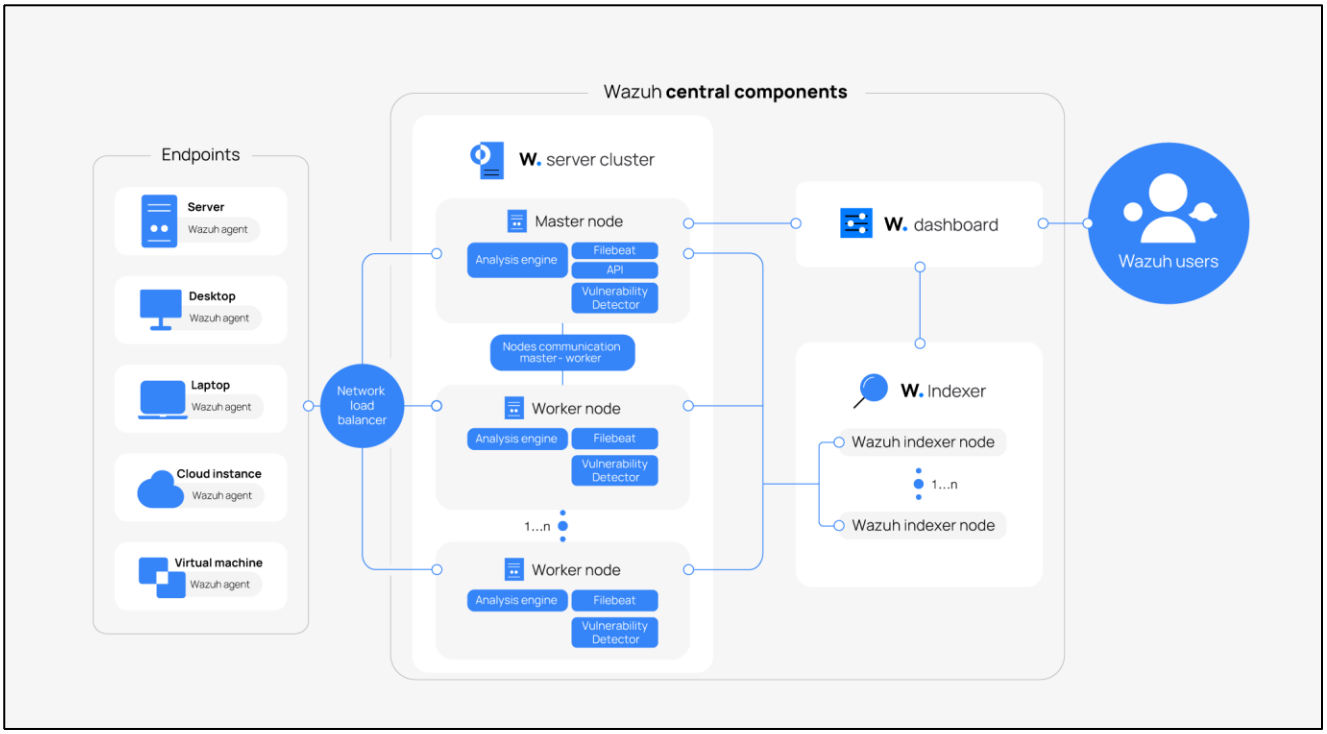


Figure 3.11: Wazuh Deployment Architecture [82]

3.2.2 Wazuh SIEM Deployment

Wazuh provides a comprehensive security framework designed to protect endpoints and cloud workloads through its advanced XDR and SIEM capabilities [82]. The system operates with a single, versatile agent and relies on three essential elements: the Wazuh server for management, the Wazuh indexer for data storage and retrieval, and the Wazuh dashboard for monitoring and analysis [82]. Wazuh is free and open source. Its components abide by the GNU General Public License, version 2, and the Apache License, Version 2.0 (ALv2) [82].

The Wazuh indexer and Wazuh server can be installed on a single host or be distributed in cluster configurations. For more deployment flexibility and customization, we install the Wazuh central components by starting with the Wazuh indexer deployment.

3.2.3 Wazuh Indexer

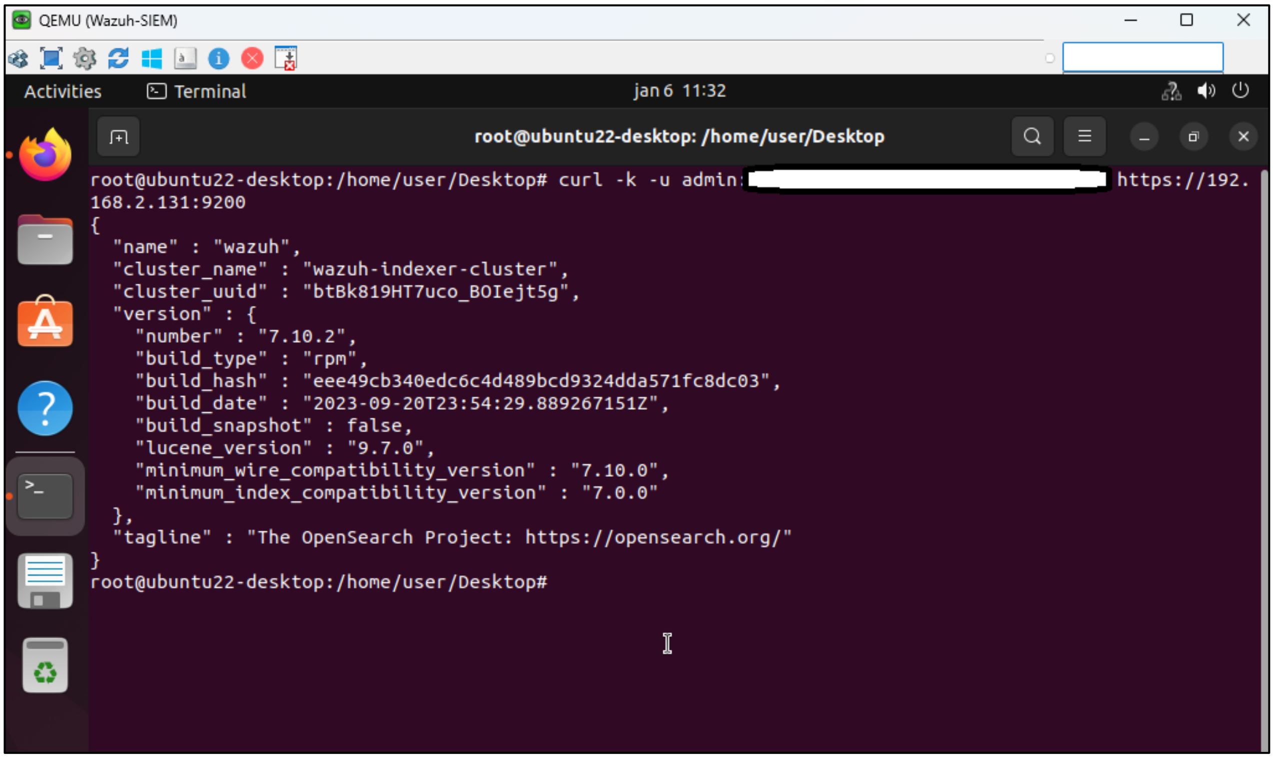
The Wazuh indexer is a highly scalable, full-text search and analytics engine [82]. This Wazuh central component indexes and stores alerts generated by the Wazuh server and provides near real-time data search and analytics capabilities [82]. We aim to install Wazuh version 4.8 on Ubuntu 22.04.5 LTS operating system. The allocation of hardware resources for our Wazuh server is 16 GB of RAM and 8 cores of CPU.

1- We begin by downloading the installation assistant script provided by Wazuh. This script simplifies the setup process by automating key configurations. After the download, we execute the script with the --generate-config-files option. This step generates essential configuration files, such as SSL certificates, cluster keys, and other necessary components for deployment. These files are packaged into a single archive named wazuh-install-files.tar. Once created, we securely copy this archive to each server that participates in the deployment, including the Wazuh server, Wazuh Indexer nodes, and Wazuh Dashboard nodes, to ensure consistency across all systems.

2- On Wazuh Indexer node, we ensure the presence of the wazuh-install-files.tar archive in the working directory. This archive includes the configuration files required for installation. We then download the latest version of the installation script onto each Indexer node. After verifying the script is ready, we execute it with the --wazuh-indexer option, specifying the name of the node (e.g., node-1). This command installs the Wazuh Indexer software and applies the settings defined in the configuration files.

3- To finalize the deployment, we perform cluster initialization. On the Wazuh Indexer nodes, we run the indexer-security-init.sh script. This script loads the new certificate details and establishes trust relations among all nodes in the cluster. Once the security settings are applied, we verify the cluster status by running diagnostic commands. This process confirms that all nodes operate and function as an integrated cluster. At this stage, the Wazuh Indexer cluster reaches full configuration and handles index tasks, completing the foundational setup required for the security platform's operation.

4- We run the following command to confirm that the installation is successful. We replace <ADMIN_PASSWORD> with the password gotten from the output of the previous command. Also, we replace <WAZUH_INDEXER_IP> with the configured Wazuh indexer IP address, as shown in Figure 3.12.



```
root@ubuntu22-desktop:/home/user/Desktop# curl -k -u admin:[redacted] https://192.
168.2.131:9200
{
  "name" : "wazuh",
  "cluster_name" : "wazuh-indexer-cluster",
  "cluster_uuid" : "btBk819HT7uco_BOIejt5g",
  "version" : {
    "number" : "7.10.2",
    "build_type" : "rpm",
    "build_hash" : "eee49cb340edc6c4d489bcd9324dda571fc8dc03",
    "build_date" : "2023-09-20T23:54:29.889267151Z",
    "build_snapshot" : false,
    "lucene_version" : "9.7.0",
    "minimum_wire_compatibility_version" : "7.10.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "The OpenSearch Project: https://opensearch.org/"
}
root@ubuntu22-desktop:/home/user/Desktop#
```

Figure 3.12: Output of Successful Wazuh Indexer Installation

The Wazuh indexer is now successfully installed, and we can proceed with installing the Wazuh server.

3.2.4 Wazuh Server

The Wazuh server analyzes the data received from the Wazuh agents, triggering alerts when threats or anomalies are detected [82]. It is also used to remotely manage the agents' configuration and monitor their status [82]. This central component includes the Wazuh manager and Filebeat.

1- We download the Wazuh installation assistant script by executing the following command:

curl -sO https://packages.wazuh.com/4.9/wazuh-install.sh

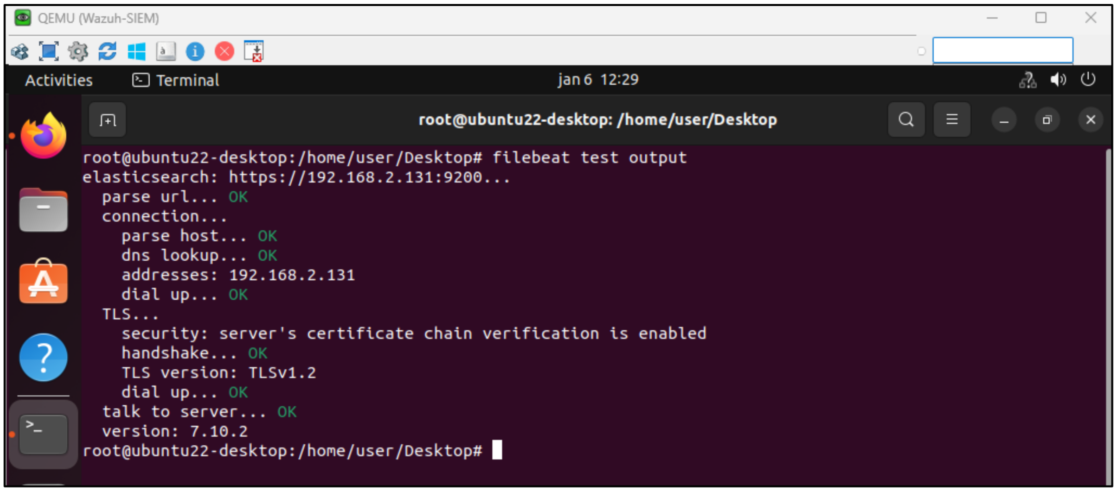
This script provides automation for the installation and configuration of the Wazuh server.

2- We run the installation script with the --wazuh-server option, specifying the node name defined in our config.yml file.

bash wazuh-install.sh --wazuh-server wazuh-1

This command installs the Wazuh server, with the Wazuh manager and Filebeat components. The Wazuh manager collects and analyzes data from deployed agents and triggers alerts for detected threats or anomalies. Filebeat forwards alerts and archived events to the Wazuh indexer.

3- We run the filebeat test output command to verify the installation of the Wazuh manager and Filebeat services, as shown in Figure 3.13. We ensure these services are active and function as expected.



```
root@ubuntu22-desktop:/home/user/Desktop# filebeat test output
elasticsearch: https://192.168.2.131:9200...
  parse url... OK
  connection...
    parse host... OK
    dns lookup... OK
    addresses: 192.168.2.131
    dial up... OK
  TLS...
    security: server's certificate chain verification is enabled
    handshake... OK
    TLS version: TLSv1.2
    dial up... OK
  talk to server... OK
  version: 7.10.2
root@ubuntu22-desktop:/home/user/Desktop# 
```

Figure 3.13: Output of Successful Filebeat Installation

The Wazuh server installation is now complete, and we can proceed with installing the Wazuh dashboard.

3.2.5 Wazuh Dashboard

This central component is a flexible and intuitive web interface for mining, analyzing, and visualizing security data [82]. It provides out-of-the-box dashboards, allowing users to seamlessly navigate through the interface. With the Wazuh dashboard, users can visualize security events, detect vulnerabilities, file integrity monitoring data, configuration assessment results, cloud infrastructure monitoring events, and regulatory compliance standards.

1- We download the Wazuh installation assistant script by executing the following command:

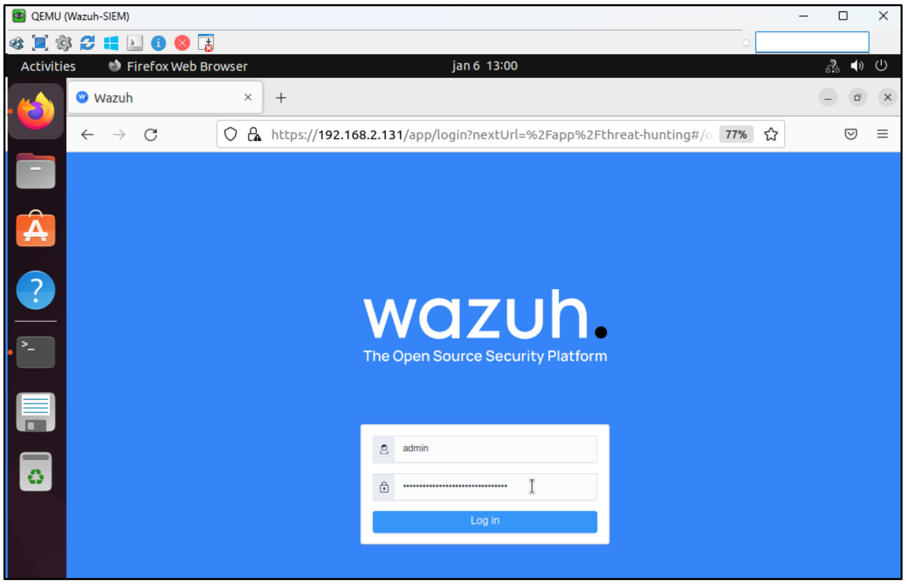
curl -sO https://packages.wazuh.com/4.9/wazuh-install.sh

This script facilitates the automated installation and configuration of the Wazuh dashboard.

2- We run the installation script with the --wazuh-dashboard option, specifying the node name as defined in our config.yml file.

bash wazuh-install.sh --wazuh-dashboard dashboard

3- Once we complete the Wazuh installation, the output displays the access credentials and a confirmation message that indicates the installation's success. With Wazuh successfully installed and configured, we open a web browser and navigate to the Wazuh IP address: https://<192.168.2.131>, as illustrated in Figure 3.14.



https://192.168.2.131/app/login?nextUrl=%2Fapp%2Fthreat-hunting#/c

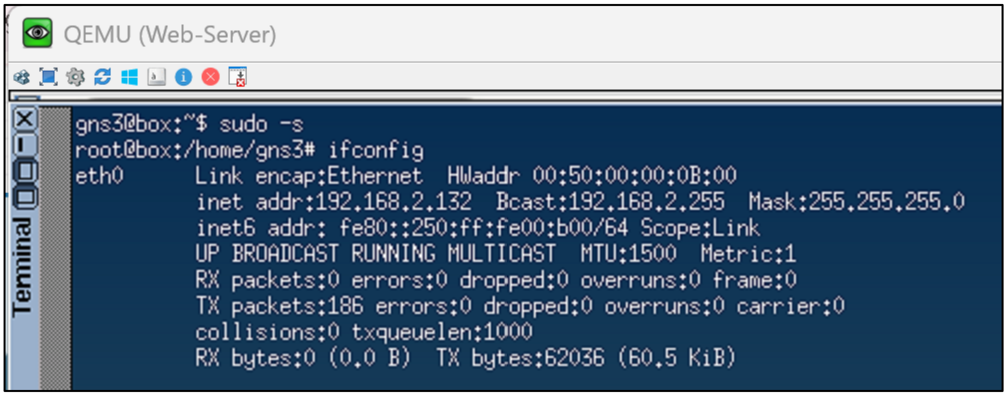
admin

Figure 3.14: Wazuh Login Page

3.2.6 Web Server Deployment

1- We prepare Tiny Core Linux on the virtual machine. We download the Tiny Core ISO from the official website and boot the virtual machine with this ISO. We assign sufficient resources of 512 MB of RAM and 2 Cores of CPU to fulfill the requirements for web server functionality. After boot, we log in using the default credentials (tc as the username and password).

2- We configure the network interface to ensure connectivity within the architecture. Using the ifconfig command, we assign a static IP address to the interface. We configure the eth0 interface with an IPv4 address of 192.168.2.132 and a subnet mask of 255.255.255.0, and assign it to the DMZ network segment, as shown in Figure 3.15.



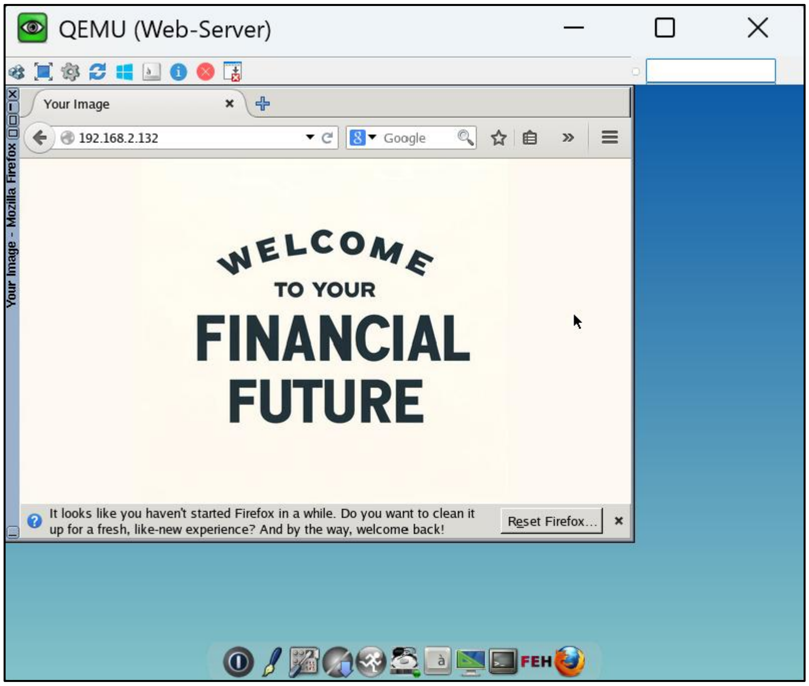
```
gns3@box:~$ sudo -s
root@box:/home/gns3# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:50:00:00:0B:00
          inet addr:192.168.2.132  Bcast:192.168.2.255  Mask:255.255.255.0
          inet6 addr: fe80::250:ff:fe00:b00/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:186 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:62036 (60.5 KiB)
```

Figure 3.15: Linux Tiny Core Interface Configuration

3- We install a lightweight web server, such as BusyBox HTTPd, which is compatible with Tiny Core Linux. With the tce-load command, we add the required packages. We execute tce-load -wi busybox-httpd to download and install the HTTPd package.

4- We create the directory structure for the web server with the mkdir -p /var/www/html command. We place the web content into this directory. We then configure the HTTPd service by editing the /usr/local/etc/httpd.conf file to define the document root as /var/www/html. We also set the server to use port 80 with the appropriate directive in the configuration file.

5- We start the HTTPd service and use the httpd -f /usr/local/etc/httpd.conf command to load the configuration file and initialize the service. We verify that the web server is operational by using a browser to open the server with the assigned IP address: http://<192.168.2.132>, as illustrated in Figure 3.16. We test the functionality and confirm the web content appears correctly, which completes the setup of the Tiny Core Linux as our web server.

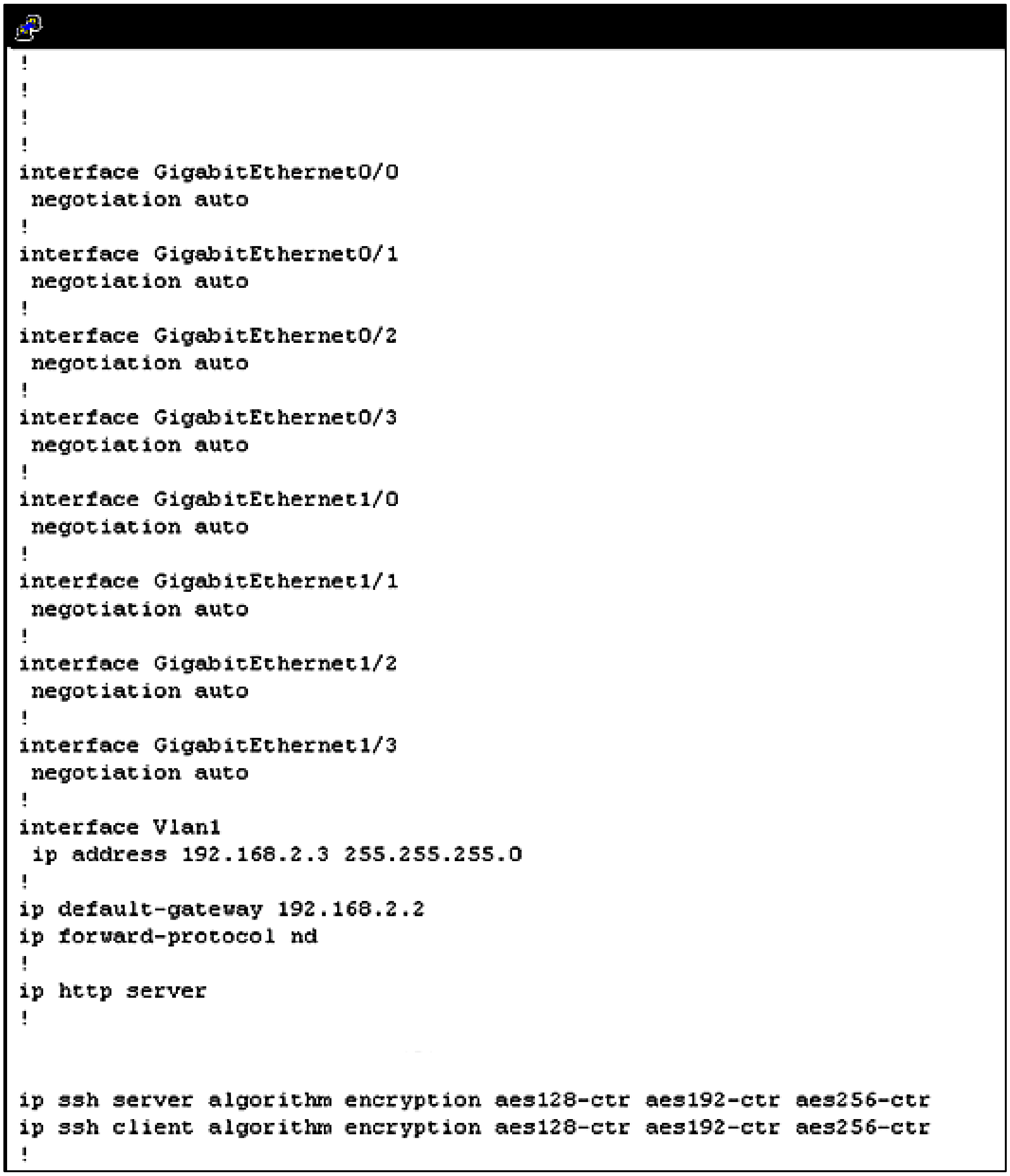


It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Figure 3.16: Web Server Home Page

3.2.7 DMZ Switch Deployment

We configure the DMZ Switch to manage network traffic within the DMZ segment of the architecture. Each Gigabit Ethernet interface auto-negotiates to establish compatibility with connected devices. VLAN 1 has the IP address 192.168.2.3/24, with a default gateway at 192.168.2.2 to facilitate traffic flow. The configuration uses auto-negotiation for all Gigabit Ethernet interfaces to optimize link parameters. All the configurations are illustrated in Figure 3.17.



```
!
!
!
!
interface GigabitEthernet0/0
 negotiation auto
!
interface GigabitEthernet0/1
 negotiation auto
!
interface GigabitEthernet0/2
 negotiation auto
!
interface GigabitEthernet0/3
 negotiation auto
!
interface GigabitEthernet1/0
 negotiation auto
!
interface GigabitEthernet1/1
 negotiation auto
!
interface GigabitEthernet1/2
 negotiation auto
!
interface GigabitEthernet1/3
 negotiation auto
!
interface Vlan1
 ip address 192.168.2.3 255.255.255.0
!
ip default-gateway 192.168.2.2
ip forward-protocol nd
!
ip http server
!


ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
!
```

Figure 3.17: DMZ Switch Configuration

After finalizing the Wazuh SIEM, web server, and switch setups to optimize data flow and establish secure communication across network components, we advance to the Core network configuration.

3.3 Core Network

The core network is the critical backbone, facilitating seamless integration between Access, DMZ, and external networks. Advanced routing protocols define optimal data transmission paths with high network resilience and minimal latency in data transfer. This network establishes a secure and efficient data-traversal framework across critical network segments, upholds enterprise-grade reliability, scalability, and fault isolation, and minimizes link or node disruption risks. In this section, we details Cisco Firepower, core router and their deployments.

3.3.1 Cisco Firepower

Cisco Firepower, a cutting-edge Intrusion Detection and Prevention System (IDPS) developed by Cisco Systems, addresses advanced security challenges through its ability to detect and counteract sophisticated DDoS attacks aligned with the goals of our research methodology [80].

Cisco Firepower integrates seamlessly into Cisco's expansive security infrastructure that functions either independently or in combination with solutions such as Firepower Threat Defense (FTD) systems to fortify organizational defenses [80].

The primary capabilities of this IDPS include these:

Advanced Threat Detection: By employing a hybrid approach of signature-based and behavior-based detection, Cisco Firepower identifies both familiar and novel threat patterns [80]. Through its connection to Cisco Talos, the company's advanced threat intelligence division, it maintains real-time updates on emerging vulnerabilities and achieves comprehensive protection against dynamic security challenges [80].

Real-time Prevention: It can take immediate action to neutralize or eliminate threats, which stops them from harming the network and connected systems [80]. This includes termination of network connections, removal of harmful packets, or isolation of compromised devices [80].

Context-aware Analysis: It evaluates network traffic and system activities within the broader operational environment to deliver precise and actionable insights [80]. This approach examines the connections among users, devices, applications, and other network components to enhance threat identification accuracy and minimize false positives [80].

File Trajectory and File Reputation: Cisco Firepower analyzes file movements within the network and assesses their credibility based on criteria such as source, destination, and behavioral patterns [80]. This capability enables organizations to uncover risks, such as malicious software or efforts to exfiltrate sensitive information [80].

SSL/TLS Decryption: It decrypts SSL/TLS encrypted traffic to inspect and analyze encrypted communications for possible threats. This capability allows organizations to identify and stop cyber attacks that exploit encryption to bypass detection [80].

Application Visibility and Control: It detects and manages a vast array of applications that allow organizations precise oversight and control of their network traffic [80]. This capability eliminates unauthorized applications, which reduces risk exposure and enhances defenses against application-level attacks [80].

URL Filtering: Cisco Firepower enforces website access control through category-based, user-specific, or customized rules [80]. This enables organizations to block harmful or compromised websites, which reduces exposure to phishing attempts and malware infections [80].

Security Intelligence: Cisco Firepower utilizes external threat intelligence feeds, such as Cisco Talos, to detect and block malicious IP addresses, domains, and URLs [80]. This ensures that organizations remain safeguarded against known cyber threats and maintain proactive defenses against emerging attacks [80].

The Cisco Secure Firewall Packet Processing framework [81] details the comprehensive lifecycle of a packet as it traverses through various layers of inspection within the firewall, as shown in Figure 3.18 [81]. The process begins with the ingress interface, where the system determines whether the packet belongs to an existing connection or requires deeper analysis. If the connection is validated, the packet is processed swiftly through the Fastpath [81]. For new connections, prefiltering policies and Layer 3/4 Access Control List (ACL) enforce high-level security rules to either allow or drop traffic at an early stage. Key functions such as defragmentation, VPN decryption, and application-layer checks are handled by the Lina engine, which also applies Quality of Service (QoS) and prepares traffic for further inspection.

For more comprehensive inspection, the Snort engine performs advanced threat analysis. This includes SSL decryption, Domain Name System (DNS) and URL blacklisting, identity-based policy enforcement, and application-specific filtering. It also leverages Snort rules to detect sophisticated threats and applies file policies to block malicious payloads. The Snort engine delivers a final verdict on whether to drop or forward the packet, while the Lina engine ensures routing, NAT, and QoS updates for outbound traffic. This integrated architecture combines the speed and efficiency of the Lina engine with the deep inspection capabilities of the Snort engine that builds a security solution in terms of detection and mitigation of DDoS attacks.

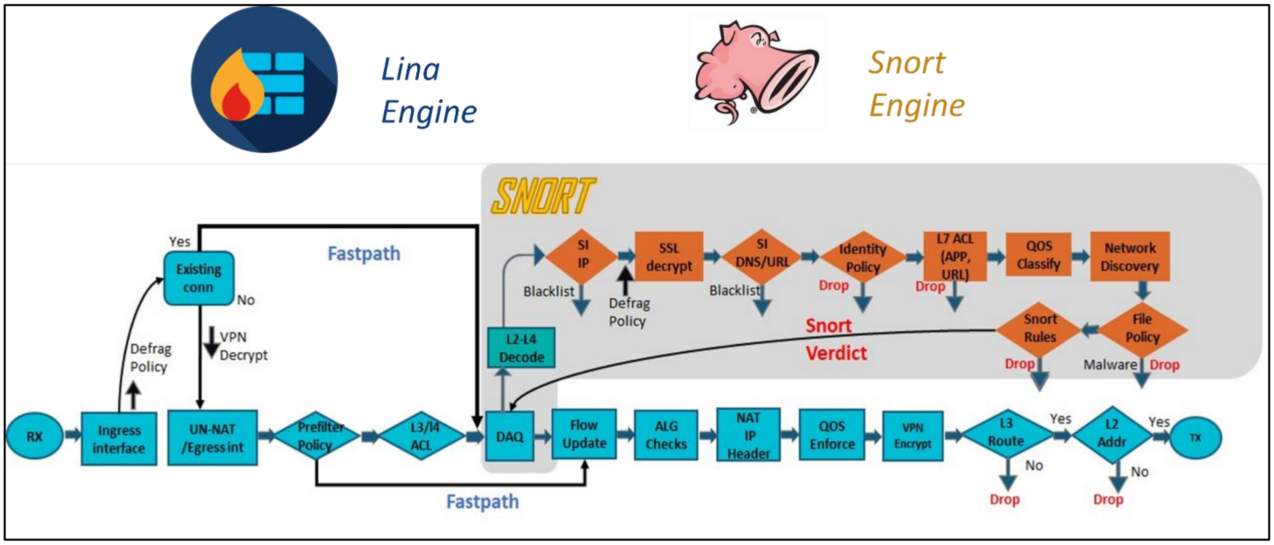


Figure 3.18: Cisco FirePower Secure Packet Processing [81]

3.3.2 Cisco Firepower Deployment

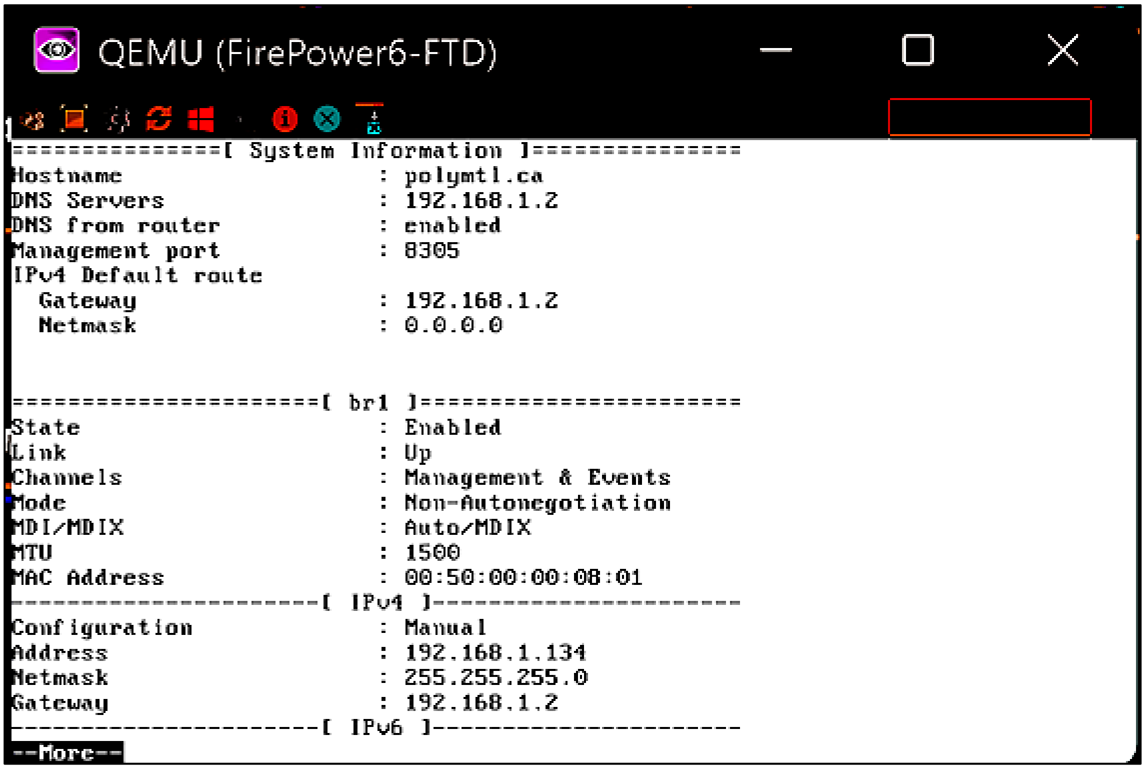
The effectiveness of Cisco's security solutions lies in the seamless collaboration between FTD and FMC [85]. FTD serves as the primary line of defense, integrated directly into the network to counter threats at their origin, while FMC provides centralized intelligence and overall control to orchestrate security operations [85]. This combination delivers a unified and fortified approach to manage and neutralize threats [85].

The integration of FTD with FMC ensures real-time updates of security policies and threat intelligence [85]. Once FTD identifies a new threat, it relays the threat details to FMC, which promptly updates rule sets across all network segments [85]. This coordinated response halts the threat's spread, ensuring containment and minimizing potential damage [85]. The Cisco Firepower deployment in our architecture includes several steps as herby:

1- We transfer the Cisco FTD virtual appliance image to the EVE-NG environment. Through secure protocols like Secure Copy Protocol (SCP) or WinSCP, we place the image in the correct directory on the EVE-NG server (/opt/unetlab/addons/qemu/). After the transfer, we verify the image integrity to confirm no corruption occurs during the process.

2- We access the EVE-NG GUI interface and create a new node for the FTD virtual machine. We select the FTD template from the available options. For optimal performance, we allocate 4 vCPUs, 8 GB of RAM, and a virtual disk with 40 GB storage. These resources meet Cisco's hardware requirements for proper operation.

3- After we save the configuration, we power on the FTD node. Once the system completes boot, the CLI prompt displays the initial setup wizard. We proceed to configure the management IP address, subnet mask, and gateway, as shown in Figure 3.19. This step establishes connectivity with the FMC later in the process.

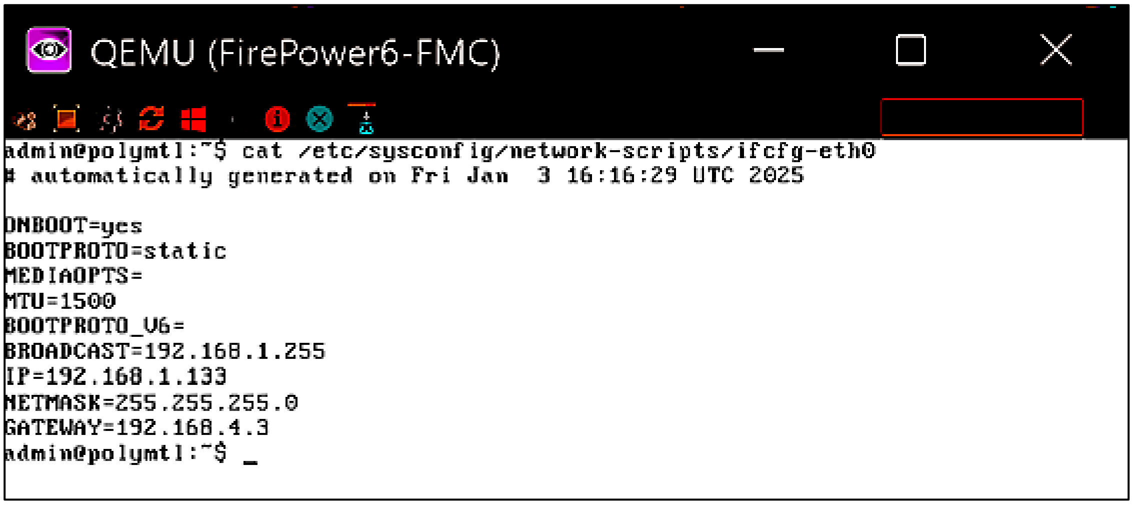


```
===============[ System Information ]===============
Hostname                  : polymtl.ca
DNS Servers               : 192.168.1.2
DNS from router           : enabled
Management port           : 8305
IPv4 Default route
  Gateway                 : 192.168.1.2
  Netmask                 : 0.0.0.0


======================[ br1 ]=======================
State                     : Enabled
Link                      : Up
Channels                  : Management & Events
Mode                      : Non-Autonegotiation
MDI/MDIX                  : Auto/MDIX
MTU                       : 1500
MAC Address               : 00:50:00:00:08:01
----------------------[ IPv4 ]----------------------
Configuration             : Manual
Address                   : 192.168.1.134
Netmask                   : 255.255.255.0
Gateway                   : 192.168.1.2
----------------------[ IPv6 ]----------------------
--More--
```

Figure 3.19: FTD Management IP Address Configuration

4- After completing the image configuration for Cisco FTD, we begin a similar process for Cisco FMC to set up its system. We upload the Cisco FMC image to the same EVE-NG server directory (/opt/unetlab/addons/qemu/) as the FTD image. After we verify the file integrity, we create a new FMC node in EVE-NG. We allocate 4 vCPUs, 12 GB RAM, and 60 GB virtual disk storage to ensure smooth operation.

5- We power on the FMC node and allow it to complete the boot process. The CLI displays a prompt where we configure the management IP address, subnet mask, and gateway. Figure 3.20 presents the IP address configuration for the management interface of Cisco FMC, which includes the assigned IP, subnet mask, and default gateway.



```
admin@polymtl:~$ cat /etc/sysconfig/network-scripts/ifcfg-eth0
# automatically generated on Fri Jan  3 16:16:29 UTC 2025

ONBOOT=yes
BOOTPROTO=static
MEDIAOPTS=
MTU=1500
BOOTPROTO_V6=
BROADCAST=192.168.1.255
IP=192.168.1.133
NETMASK=255.255.255.0
GATEWAY=192.168.4.3
admin@polymtl:~$ _
```

Figure 3.20: FMC Management IP Address Configuration

6- We use a browser to connect to the FMC management IP address: https://<192.168.1.133>. We input the default login credentials with username “admin” and password “Admin123” to access the FMC dashboard. We perform basic configurations, such as defining DNS servers, and NTP servers for time synchronization, and setting the system hostname. To confirm proper installation, we verify the FMC system status through the GUI. This step ensures the FMC functions correctly and manages the FTD nodes in subsequent steps. Figure 3.21 shows FMC's login page after successful installation. After initial configurations, the FMC’s overview dashboard is illustrated in Figure 3.22.

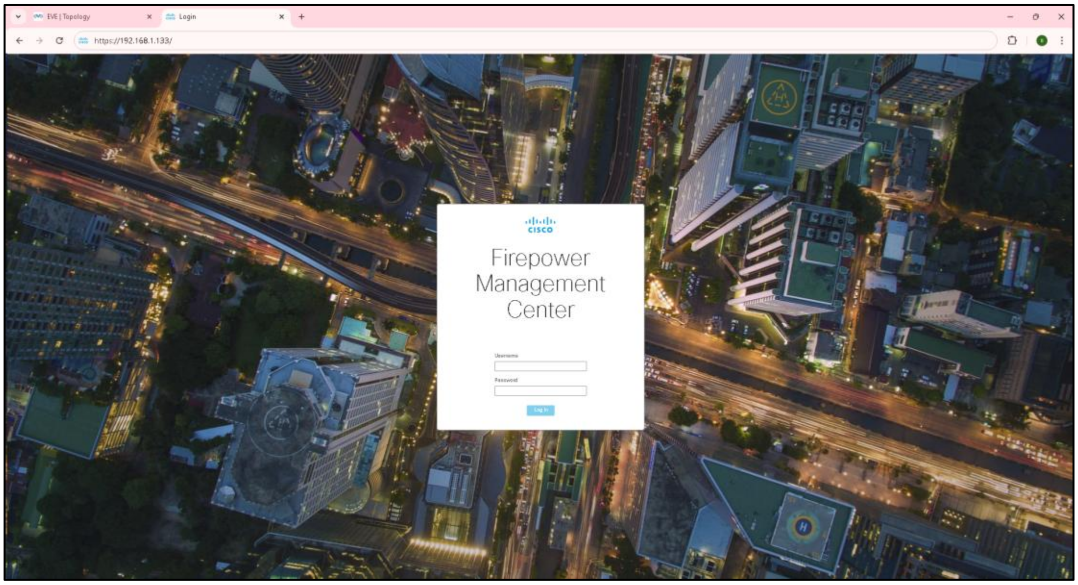
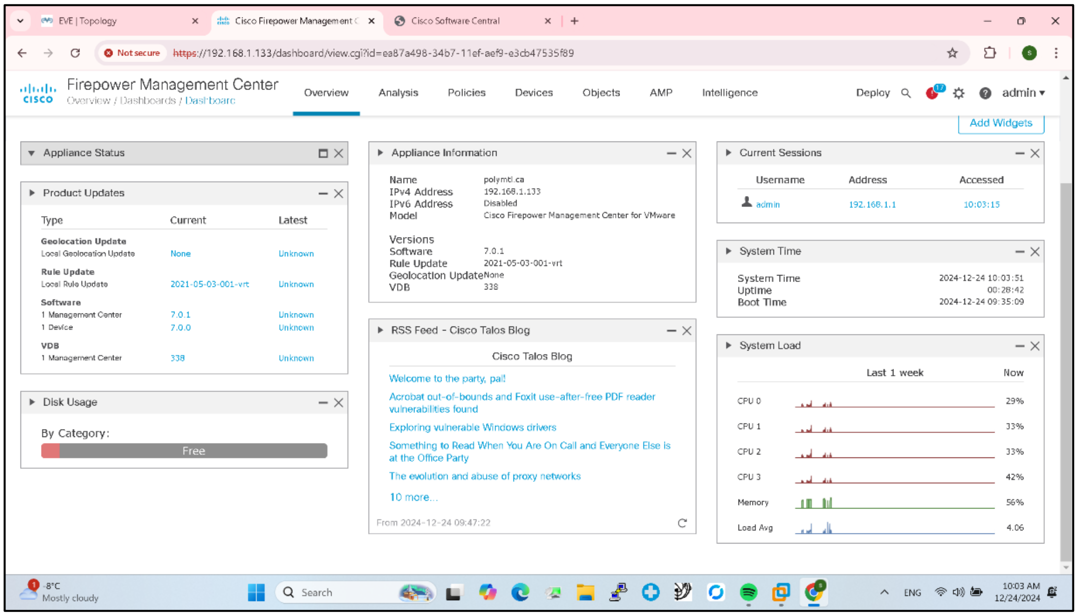


Figure 3.21: Cisco FMC Login Page


EVE | Topology
Cisco Firepower Management C
Cisco Software Central
Not secure
https://192.168.1.133/dashboard/view.cgi?id=ea87a498-34b7-11ef-aef9-e3cb47535f89
CISCO
Firepower Management Center
Overview / Dashboards / Dashboard
Overview
Analysis
Policies
Devices
Objects
AMP
Intelligence
Deploy
17
admin
Add Widgets
Search
ENG
10:03 AM
12/24/2024

Appliance Status

Product Updates

Type	Current	Latest
Geolocation Update		
Local Geolocation Update	None	Unknown
Rule Update		
Local Rule Update	2021-05-03-001-vrt	Unknown
Software		
1 Management Center	7.0.1	Unknown
1 Device	7.0.0	Unknown
VDB		
1 Management Center	338	Unknown

Disk Usage

By Category:

Free

Appliance Information

```
Name               polymtl.ca
IPv4 Address       192.168.1.133
IPv6 Address       Disabled
Model              Cisco Firepower Management Center for VMware

Versions
Software           7.0.1
Rule Update        2021-05-03-001-vrt
Geolocation UpdateNone
VDB                338
```

RSS Feed - Cisco Talos Blog

Cisco Talos Blog

Welcome to the party, pal!

Acrobat out-of-bounds and Foxit use-after-free PDF reader vulnerabilities found

Exploring vulnerable Windows drivers

Something to Read When You Are On Call and Everyone Else is at the Office Party

The evolution and abuse of proxy networks

10 more...

From 2024-12-24 09:47:22

Current Sessions

Username	Address	Accessed
admin	192.168.1.1	10:03:15

System Time

System Time	2024-12-24 10:03:51
Uptime	00:28:42
Boot Time	2024-12-24 09:35:09

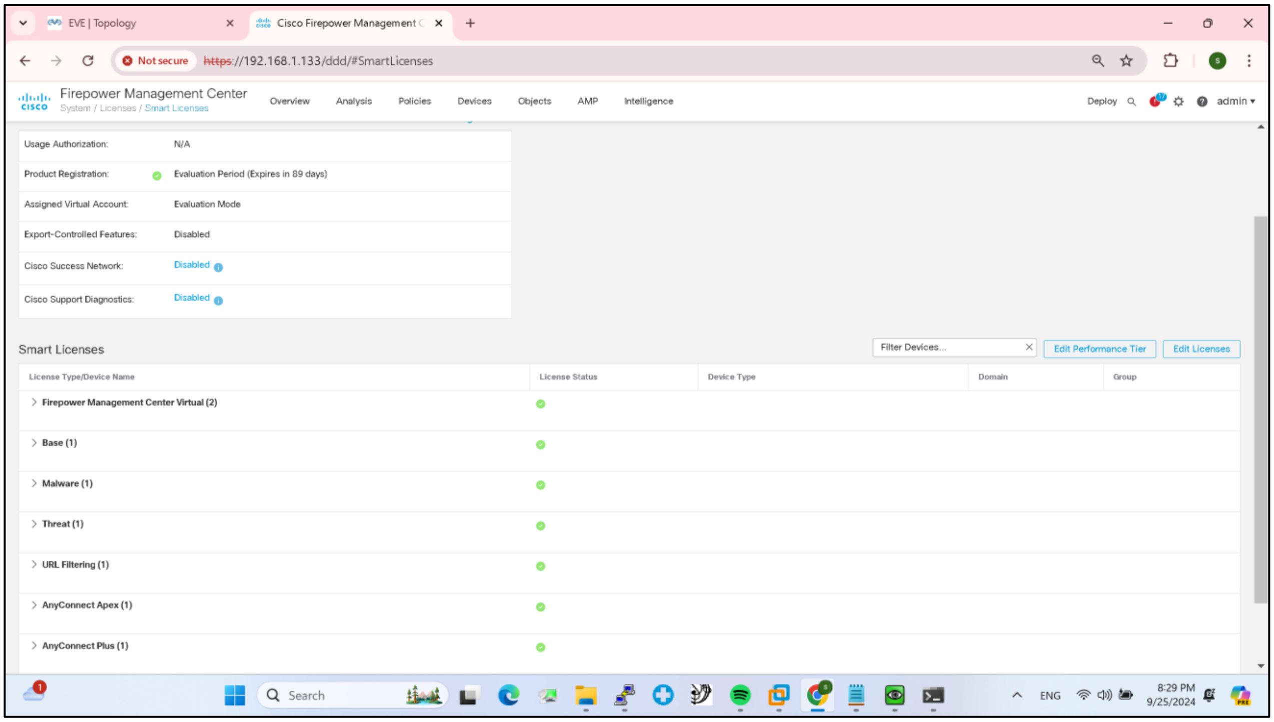
System Load

	Last 1 week	Now
CPU 0		29%
CPU 1		33%
CPU 2		33%
CPU 3		42%
Memory		56%
Load Avg		4.06

-8°C
Mostly cloudy

Figure 3.22: Cisco FMC Overview Dashboard

7- Afterwards, we must activate the appropriate license on the FMC virtual host. For this purpose, we enter the smart account details in the FMC interface to activate the 90-day evaluation period license. A proper activation license ensures that the FTD device benefits from these enhanced security features like Threat, URL Filtering, and Malware Protection. Figure 3.23 provides a comprehensive view of the activated license and available features.



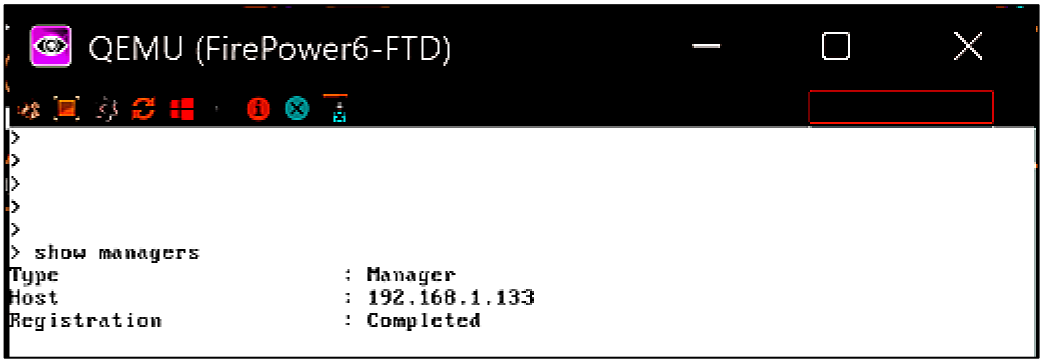
Usage Authorization:	N/A
Product Registration:	● Evaluation Period (Expires in 89 days)
Assigned Virtual Account:	Evaluation Mode
Export-Controlled Features:	Disabled
Cisco Success Network:	Disabled
Cisco Support Diagnostics:	Disabled

Smart Licenses

License Type/Device Name	License Status	Device Type	Domain	Group
Firepower Management Center Virtual (2)	●			
Base (1)	●			
Malware (1)	●			
Threat (1)	●			
URL Filtering (1)	●			
AnyConnect Apex (1)	●			
AnyConnect Plus (1)	●			

Figure 3.23: Cisco FMC Evaluation License and Features

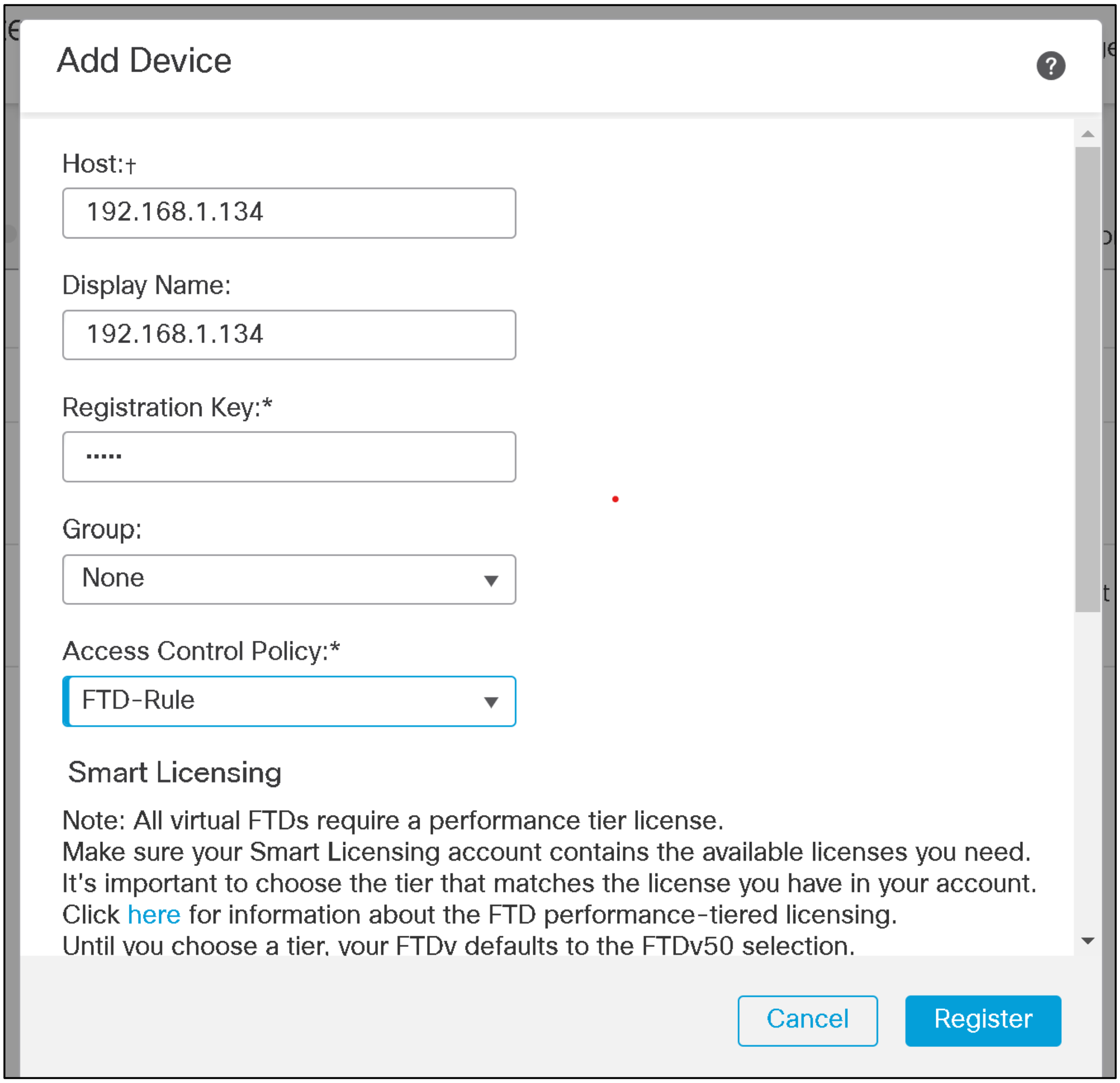
8- To establish communication between the FTD and FMC, we access the FTD CLI through a terminal. Then, we execute the configure manager add <FMC_IP> <PRE_SHARED_KEY> command that links the FTD device to the FMC securely. We execute the show manager command to verify the connection. This command displays the status of the FMC-FTD pairing and confirms successful registration, as shown in Figure 3.24.



```
>
>
>
>
>
> show managers
Type                    : Manager
Host                    : 192.168.1.133
Registration            : Completed
```

Figure 3.24: FMC-FTD Successful Registration in FTD Cli

9- After successful registration of the FTD device to the FMC, we return to the FMC interface to deploy a test policy on the newly added FTD device. We navigate to the Devices section and Device Management tab in the FMC interface. This section handles the integration of new devices into the FMC framework and confirms the communication link functions correctly. We click Add Device and enter the FTD device's management IP address. We also specify a pre-shared key, which establishes a secure link between the FMC and FTD. Figure 3.25 outlines the specific parameters necessary to define a secure connection between the FMC and FTD.



Add Device

Host:†
192.168.1.134

Display Name:
192.168.1.134

Registration Key:*
•••••

Group:
None

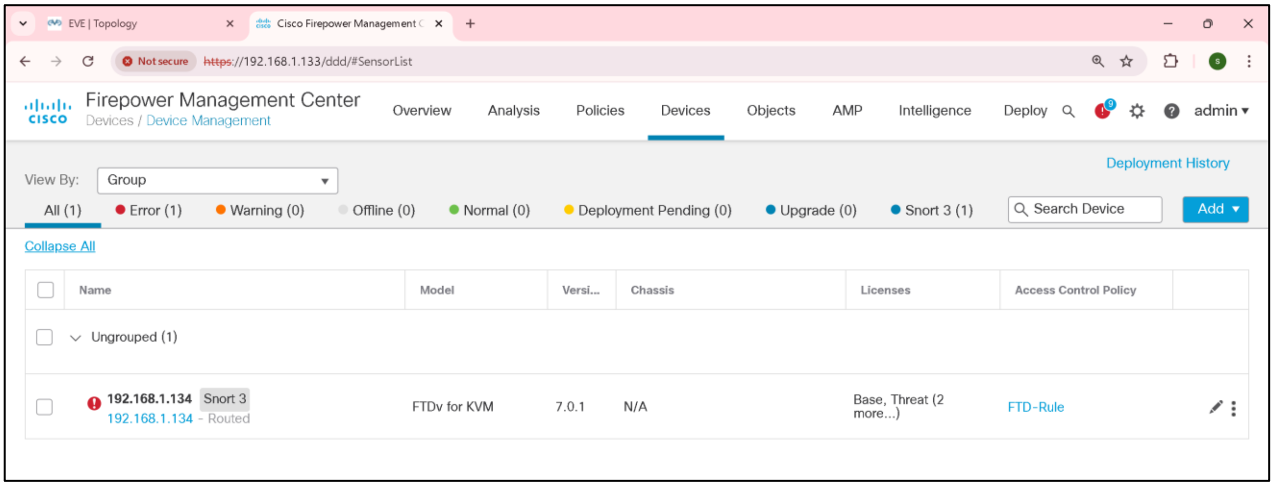
Access Control Policy:*
FTD-Rule

Smart Licensing

Note: All virtual FTDs require a performance tier license.
Make sure your Smart Licensing account contains the available licenses you need.
It's important to choose the tier that matches the license you have in your account.
Click here for information about the FTD performance-tiered licensing.
Until you choose a tier, your FTDv defaults to the FTDv50 selection.

Figure 3.25: Specific Parameters for Secure Connection between FMC and FTD

10- We review the details entered and confirm the addition of the FTD device to the FMC inventory. Figure 3.26 provides evidence of the FMC connection and its ability to manage the device for proper configurations and enforcement of real-time security policies.



View By: Group

□	Name	Model	Versi...	Chassis	Licenses	Access Control Policy
□	Ungrouped (1)					
□	192.168.1.134 Snort 3 192.168.1.134 - Routed	FTDv for KVM	7.0.1	N/A	Base, Threat (2 more...)	FTD-Rule

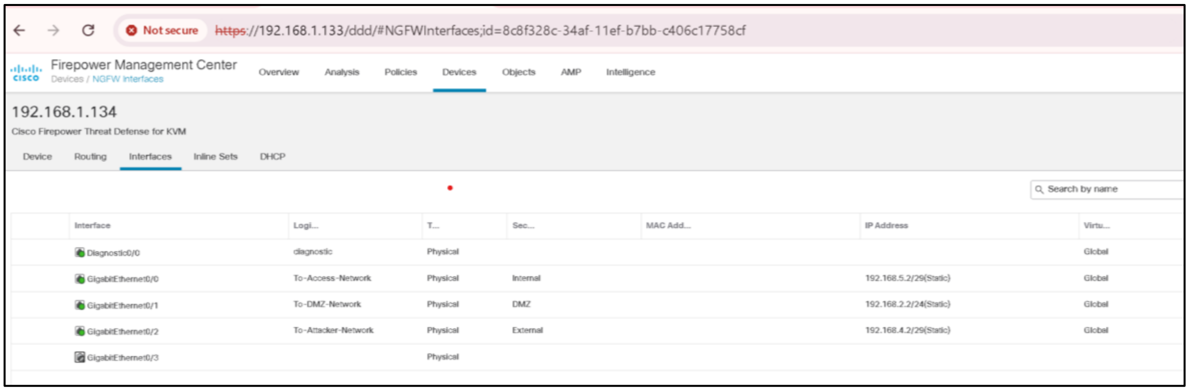
Figure 3.26: FMC-FTD Successful Registration in FMC Interface

11- To configure the FTD device, we proceed with detailed network interface configuration. The first step assigns roles and ensures proper alignment of each interface to its designated network.

Interface GigabitEthernet0/0 is configured as the bridge between FTD and core router, represented with the IP address 192.168.5.2/29. This interface creates communication with the Access Network devices.

The second interface, GigabitEthernet0/1, is allocated for connection from FTD to DMZ switch with the IP address 192.168.2.2/24. This interface provides secured access to DMZ-hosted services like web servers and SIEM solutions.

For external traffic, interface GigabitEthernet0/2 is set to connect the FTD to the external or attacker network, defined with the IP address 192.168.4.2/29. This setup routes external threats, such as simulated DDoS traffic, through the FTD to enable thorough analysis and counteraction. Figure 3.27 shows all the IP addresses and network configurations.


Not secure
https://192.168.1.133/ddd/#NGFWInterfaces;id=8c8f328c-34af-11ef-b7bb-c406c17758cf
Firepower Management Center
Devices / NGFW Interfaces
Overview
Analysis
Policies
Devices
Objects
AMP
Intelligence
Device
Routing
Interfaces
Inline Sets
DHCP
Search by name

192.168.1.134
Cisco Firepower Threat Defense for KVM

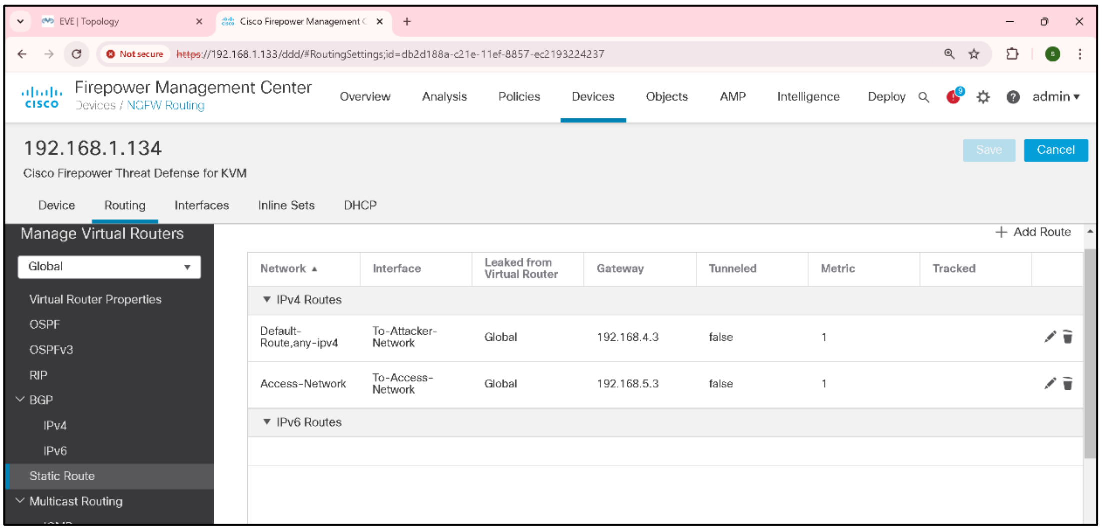
Interface	Logi...	T...	Sec...	MAC Add...	IP Address	Virtu...
Diagnostic0/0	diagnostic	Physical				Global
GigabitEthernet0/0	To-Access-Network	Physical	Internal		192.168.5.2/29(Static)	Global
GigabitEthernet0/1	To-DMZ-Network	Physical	DMZ		192.168.2.2/24(Static)	Global
GigabitEthernet0/2	To-Attacker-Network	Physical	External		192.168.4.2/29(Static)	Global
GigabitEthernet0/3		Physical				

Figure 3.27: Cisco FTD IP Address Interfaces

12- After network address configurations, we define static routes to establish precise traffic paths across the network segments. In the Route section under the Devices tab, we select Static Route to define paths for specific network traffic. The configuration starts with the creation of a route to the Access Network. We specify the network address of 192.168.3.0/24 and assign it to the To-Access-Network interface. The gateway address is set to 192.168.5.3, which serves as the next-hop router.

Next, we add a default route to handle traffic that does not match any specific route in the routing table. We set the destination as 0.0.0.0/0, which defines all other traffic types. The next-hop gateway for this route is 192.168.4.3, which represents the Attacker Network. All the static route configurations are illustrated in Figure 3.28.

After verifying the entries, we save the static route configurations. Following this, we deploy the changes to the FTD device to ensure the routes take effect.



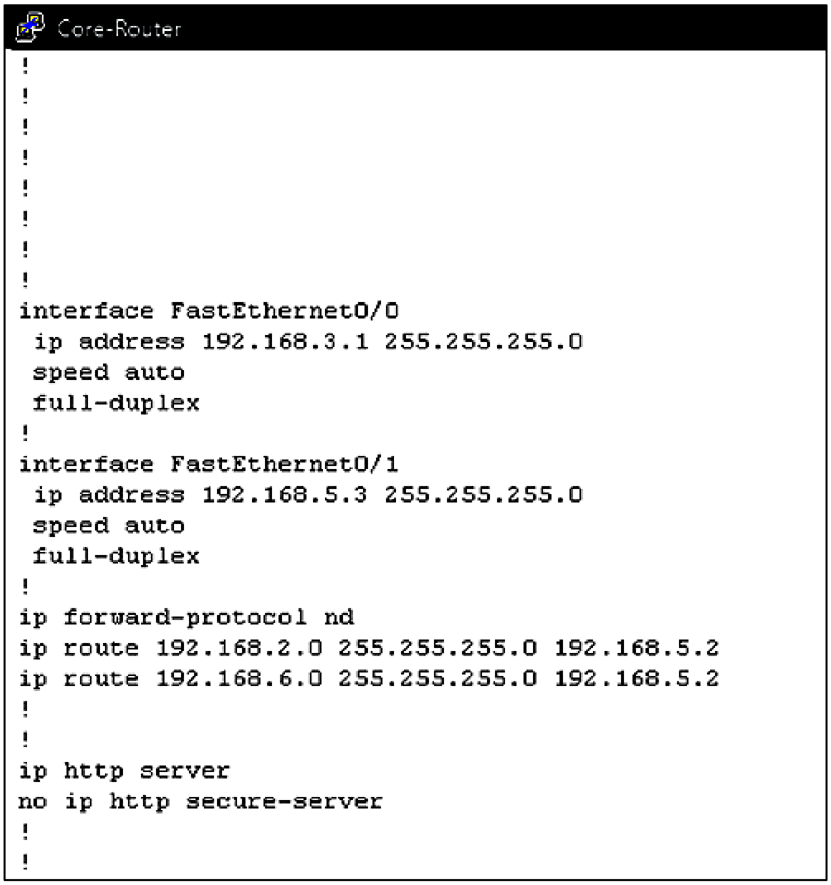
192.168.1.134
Cisco Firepower Threat Defense for KVM

Network ▲	Interface	Leaked from Virtual Router	Gateway	Tunneled	Metric	Tracked
▼ IPv4 Routes						
Default-Route,any-ipv4	To-Attacker-Network	Global	192.168.4.3	false	1	
Access-Network	To-Access-Network	Global	192.168.5.3	false	1	
▼ IPv6 Routes						

Figure 3.28: Cisco FTD Static Routes

3.3.3 Core Router Deployment

We configure the Core Router to act as the central route point in our architecture. Two interfaces, FastEthernet0/0 and FastEthernet0/1, use IP addresses 192.168.3.1 and 192.168.5.3, each with a subnet mask of 255.255.255.0. We define static routes to direct traffic for the DMZ (192.168.2.0) and the Attacker Network (192.168.6.0) through the next-hop gateway 192.168.5.2. We disable the HTTP secure server to improve security and limit unnecessary exposure, ensuring efficient and secure route functionality. All the configurations are presented in Figure 3.29.



```
!
!
!
!
!
!
!
!
interface FastEthernet0/0
 ip address 192.168.3.1 255.255.255.0
 speed auto
 full-duplex
!
interface FastEthernet0/1
 ip address 192.168.5.3 255.255.255.0
 speed auto
 full-duplex
!
ip forward-protocol nd
ip route 192.168.2.0 255.255.255.0 192.168.5.2
ip route 192.168.6.0 255.255.255.0 192.168.5.2
!
!
ip http server
no ip http secure-server
!
!
```

Figure 3.29: Core Router Configuration

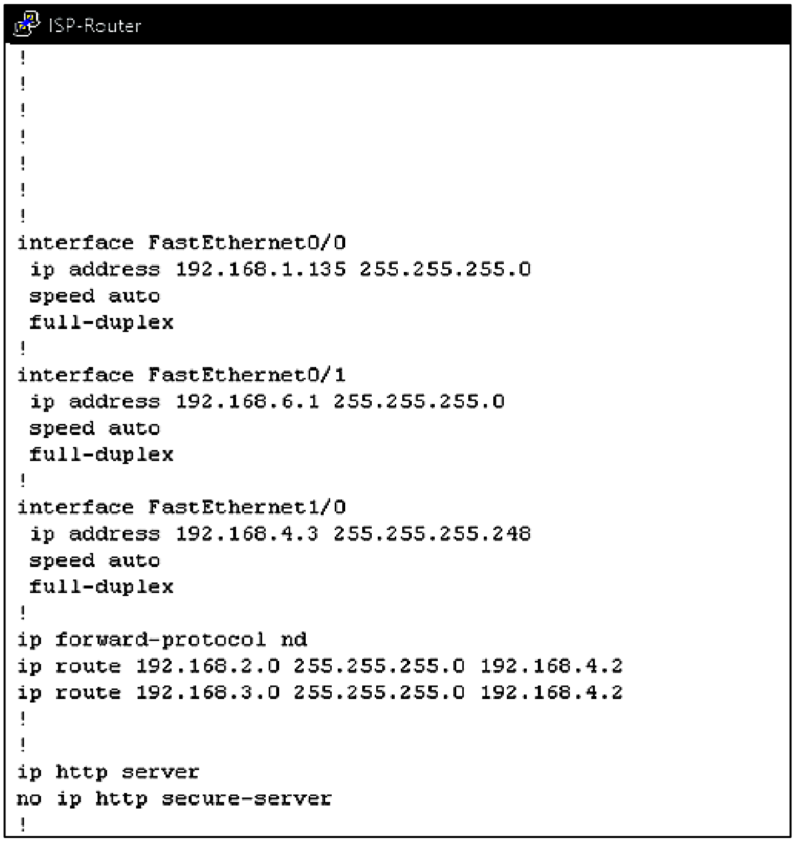
After finalizing the deployment of Cisco Firepower and Core router configurations to streamline data exchange and fortify secure interconnectivity across network entities, we progress to the structured implementation of the attacker network.

3.4 Attacker Network

The Attacker Network operates as an untrusted external domain, structured to simulate complex threat scenarios and evaluate the resilience of the security framework. It introduces advanced cyber threats, such as DDoS attacks and sophisticated intrusion vectors that subject the system's defenses to rigorous assessment. In this section, we will present the ISP router, Kali Linux, and their deployments.

3.4.1 ISP Router Deployment

We configure the ISP Router to act as the gateway between the internal and external networks in the architecture. It includes three interfaces: FastEthernet0/0 with IP 192.168.1.135, FastEthernet0/1 with IP 192.168.6.1, and FastEthernet1/0 with IP 192.168.4.3, each assigned a subnet mask appropriate to their network role. Static routes direct traffic for the DMZ (192.168.2.0) and Access Network (192.168.3.0) through the next-hop gateway 192.168.4.2. We disable the HTTP secure server to reduce unnecessary services and improve security. All the configurations are depicted in Figure 3.30.



```
!
!
!
!
!
!
!
interface FastEthernet0/0
 ip address 192.168.1.135 255.255.255.0
 speed auto
 full-duplex
!
interface FastEthernet0/1
 ip address 192.168.6.1 255.255.255.0
 speed auto
 full-duplex
!
interface FastEthernet1/0
 ip address 192.168.4.3 255.255.255.248
 speed auto
 full-duplex
!
ip forward-protocol nd
ip route 192.168.2.0 255.255.255.0 192.168.4.2
ip route 192.168.3.0 255.255.255.0 192.168.4.2
!
!
ip http server
no ip http secure-server
!
```

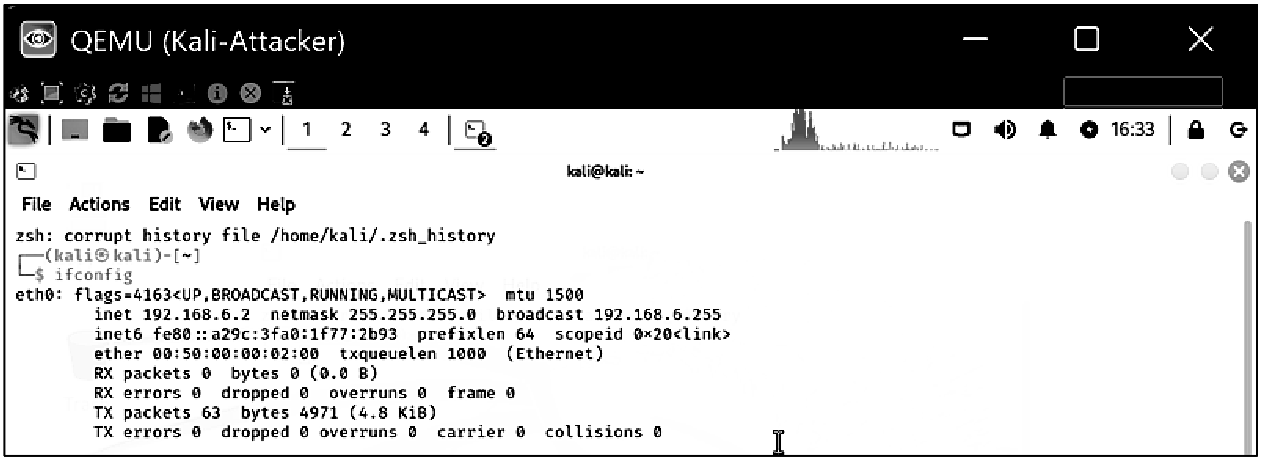
Figure 3.30: ISP Router Configuration

3.4.2 Kali Linux

Kali Linux operates as the central component within the Attacker Network that emulates DDoS traffic to evaluate the effectiveness of our security architecture. Located in an isolated network segment, it enables the creation of advanced DDoS attacks, TCP-SYN floods, and various exploit methods aimed at weaknesses in the DMZ, Access, and Core networks. This setup offers a controlled yet realistic environment to evaluate how defensive systems, Cisco FTD and Wazuh SIEM detect, counter, and mitigate potential DDoS attacks efficiently.

3.4.3 Kali Attacker Deployment

The configuration of Kali Linux in this architecture shows the interface eth0 with an assigned IP address of 192.168.6.2 and a subnet mask of 255.255.255.0, as shown in Figure 3.31. This setup ensures the system's connectivity within the Attacker Network. This configuration enables the Kali Linux system to generate and transmit malicious DDoS traffic to evaluate the security measures of our scenario.



```
zsh: corrupt history file /home/kali/.zsh_history
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.6.2  netmask 255.255.255.0  broadcast 192.168.6.255
        inet6 fe80::a29c:3fa0:1f77:2b93  prefixlen 64  scopeid 0×20<link>
        ether 00:50:00:00:02:00  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 63  bytes 4971 (4.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Figure 3.31: Kali Linux Interface Configurations

3.5 Summary

This chapter presented our proposed architecture for the *DDoS attack detection and mitigation* purpose that provides security via the *Next Generation Firewall (NGFW) and Security Information and Event Management (SIEM)* system capabilities. We propose our architecture in three different segments of *Access, Core, and Demilitarized Zone (DMZ) networks*. Also, An Attacker Network generates large-scale DDoS flood traffic, emulates hostile activities, and allows a thorough assessment of implemented security defenses.

The network components deployed in the architecture include Cisco FTD, FMC, Wazuh SIEM, Kali Linux, Ostinato Traffic Generator, Linux Tiny Core Web Server, and VPC. Each component is described in detail and highlighted its role and configuration within the network. Cisco Firepower protects the network with the detection and mitigation of advanced threats, while Wazuh SIEM provides centralized event correlation and analysis. Kali Linux emulates malicious traffic, and Ostinato generates SIP packets for testing. The Linux Tiny Core Web Server, located in the DMZ, ensures the secure hosting of public-facing services. VPC node simulates user-side activities and validates the network's data flow integrity.

Switches and routers within the DMZ, Access, and Core networks implement VLAN segmentation, static routes, and advanced routing protocols to ensure seamless communication and strict isolation between network segments. The simulation environment uses EVE-NG, a virtual network emulation platform, which allows realistic tests of the architecture's effectiveness.

The final sections cover the integration of all network components and their collective operation within the proposed architecture. Each node has a responsibility to construct a comprehensive, layered security framework. The architecture shows its ability to detect, analyze, and mitigate DDoS attacks in real-time while preserving operational efficiency and security. This chapter lays the foundation for the experimental validation of the secure architecture in the next chapter.

CHAPTER 4 IMPLEMENTATION AND RESULTS

In Chapter 3, we presented our proposed architecture to provide security of infrastructure and services in three different network segments. Furthermore, we discussed the deployment of network nodes in our scenario and their role in enforcing strict security measures and advanced threat detection methods. In this chapter, we examine our security solution's performance in terms of real-time DDoS detection and mitigation. Additionally, we provide a detailed analysis of our results from the evaluation of the integrated architecture in threat identification and response accuracy under various DDoS attack conditions.

4.1 Simulation Scenario

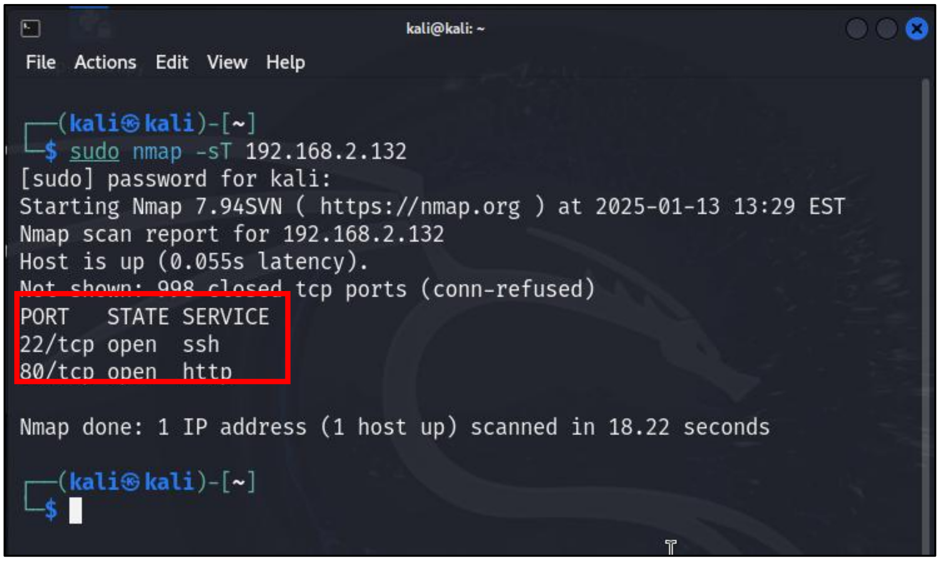
We conduct an in-depth analysis under two distinct scenarios: one without security policies and another with robust security measures in place. Initially, we simulate multiple types of DDoS attacks through the Kali attacker while Cisco Firepower operates in transparent mode without predefined security policies. In the second scenario, we implement comprehensive security policies within Cisco Firepower to fortify the network. We then reintroduce the same DDoS attack vectors to test the enhanced architecture rigorously. By comparison of the outcomes of these scenarios, we assess the effectiveness of our DDoS detection and mitigation strategy and emphasize improvements in response accuracy, threat containment, and network resilience.

4.2 DDoS Flood Scenario Under Absence of Security Policy

Networks face threats from attackers who employ different techniques to access sensitive data [86]. A compromised web portal or device could lead to significant network damage and service interruptions [86]. Numerous companies, such as Google, banking institutions, and Microsoft, promote Ethical Hacking to detect vulnerabilities in their networks and reward ethical hackers with substantial prizes [86]. Furthermore, organizations hire network consultants to evaluate weaknesses and provide tailored strategies to enhance asset and network security [86].

Ethical hacking addresses flaws and vulnerabilities in devices and networks. We apply non-destructive methods to evaluate our network's security. This practice functions as a safety audit and professional evaluation of our system protections. Various methods exist for data acquisition, vulnerability analysis, system exploitation, and test result evaluation [86]. Ethical hacking depends heavily on automation tools, as manual techniques lack efficiency and consume significant time. NMAP [86] is a prominent tool for port analysis and service validation in secure operations [86].

We employ NMAP to discover the vulnerability aspects of the web server in our system. The command sudo nmap -sT 192.168.2.132 initiates a TCP connect scan on the target IP address 192.168.2.132, which belongs to the web server hosted in the DMZ network, as shown in Figure 4.1. The scan detects two open ports port 22, which runs the SSH service, and port 80, which hosts the HTTP service. These results show that the web server is active, and two open ports are analyzed to determine their accessibility from the attacker's machine. The other 998 TCP ports are reported as closed, which confirms that access is restricted on those ports. Consequently, ports 80 and 22 become potential targets for the Kali attacker to exploit or send queries.



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sT 192.168.2.132
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-13 13:29 EST
Nmap scan report for 192.168.2.132
Host is up (0.055s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 18.22 seconds

┌──(kali㉿kali)-[~]
└─$ 
```

Figure 4.1: NMAP Port Scan Results on Web Server

The Kali attacker then attempts to access the web server through the web browser on her local machine, as shown in Figure 4.2. By inputting the web server’s IP address into the URL bar of Mozilla Firefox on Kali Linux, the attacker connects to the web server located within the organization’s DMZ network. This shows that the web server responds to requests from the attacker's machine.

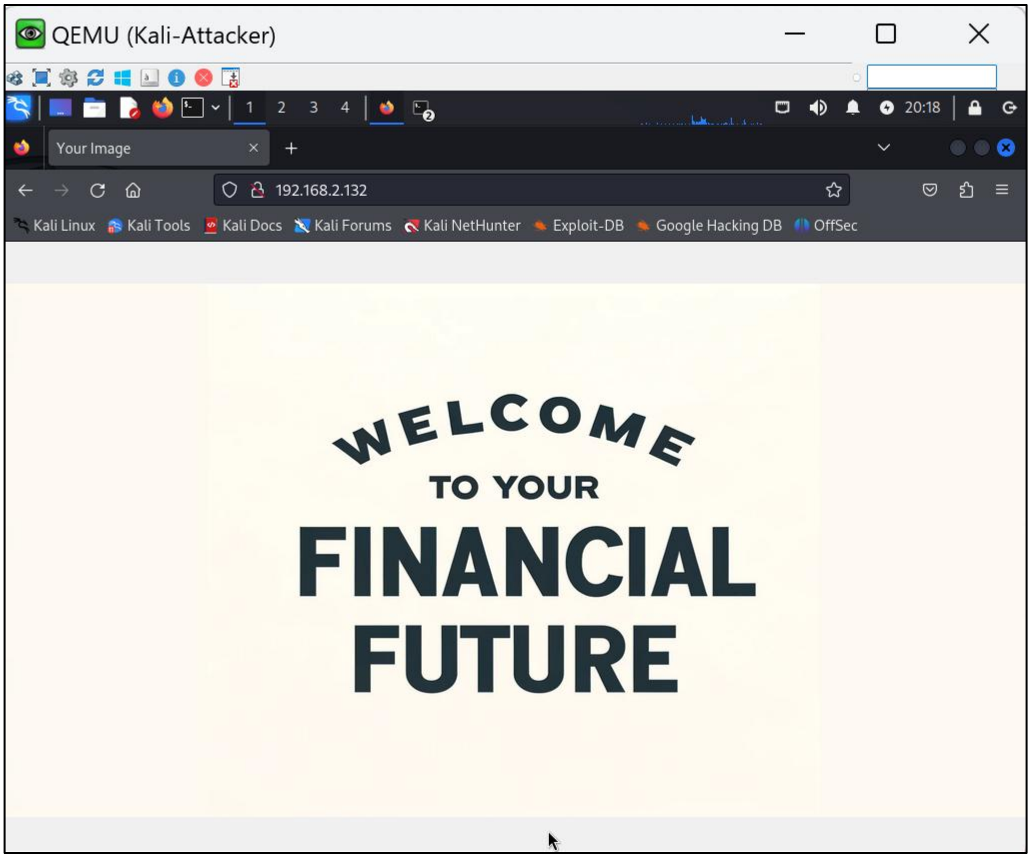


Figure 4.2: Web Server Access from Kali Attacker Machine

With the confirmation of uninterrupted access to the web server, the Kali attacker decides it is now time to launch different types of DDoS attacks. From the Kali Linux terminal, the attacker begins to execute specific commands to initiate DDoS flood attacks and target the web server for disruption.

4.3 TCP SYN Attack Under Absence of Security Policy

A wide range of network DDoS attacks target systems and computers to exploit sensitive data or overwhelm network resources [87]. These attacks often aim to deny legitimate users access to services. One example is the TCP SYN [87] flood attack, where an attacker sends many SYN packets every second. In a SYN flood attack, the attacker sends a large number of SYN packets per second to the victim without waiting for the SYN-ACK response [87]. As a result, when a legitimate user tries to establish a TCP connection, the server does not respond due to resource exhaustion, as illustrated in Figure 4.3 [87].

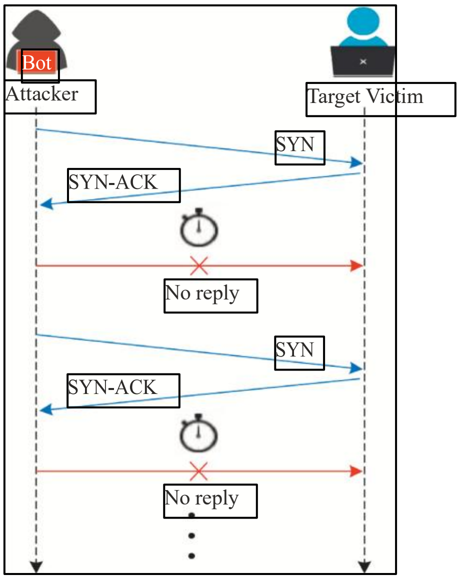
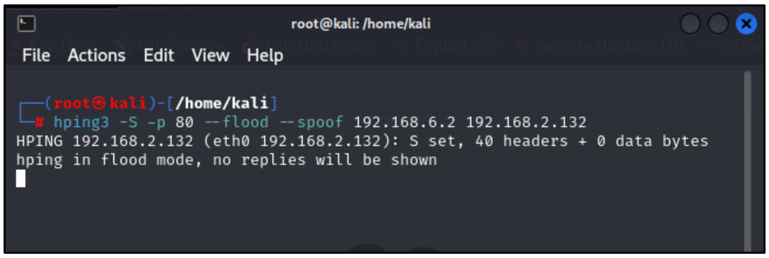


Figure 4.3: TCP SYN Flood Attack [87]

We execute the TCP SYN flood attack from the Kali attacker to the web server in our architecture. We use the command hping3 -S -p 80 --flood --spoof 192.168.6.2 192.168.2.132 to initiate a flood of SYN packets that target the web server at IP address 192.168.2.132 on port 80, as shown in Figure 4.4. The tool hping3 is a versatile packet craft utility that generates and sends customized TCP/IP packets, which is effective for test and attack simulation. In this command, the -S flag sets the SYN flag in the TCP header, -p 80 specifies the target port, --flood enables high-speed packet generation, and --spoof 192.168.6.2 sets a fake source IP to disguise the attacker's identity. As we use ethical hacking, there is no need to cover the attacker’s identity. This flood overwhelms the server's resources by sending a continuous stream of SYN packets without any wait for SYN-ACK responses, which results in a potential DDoS attack.



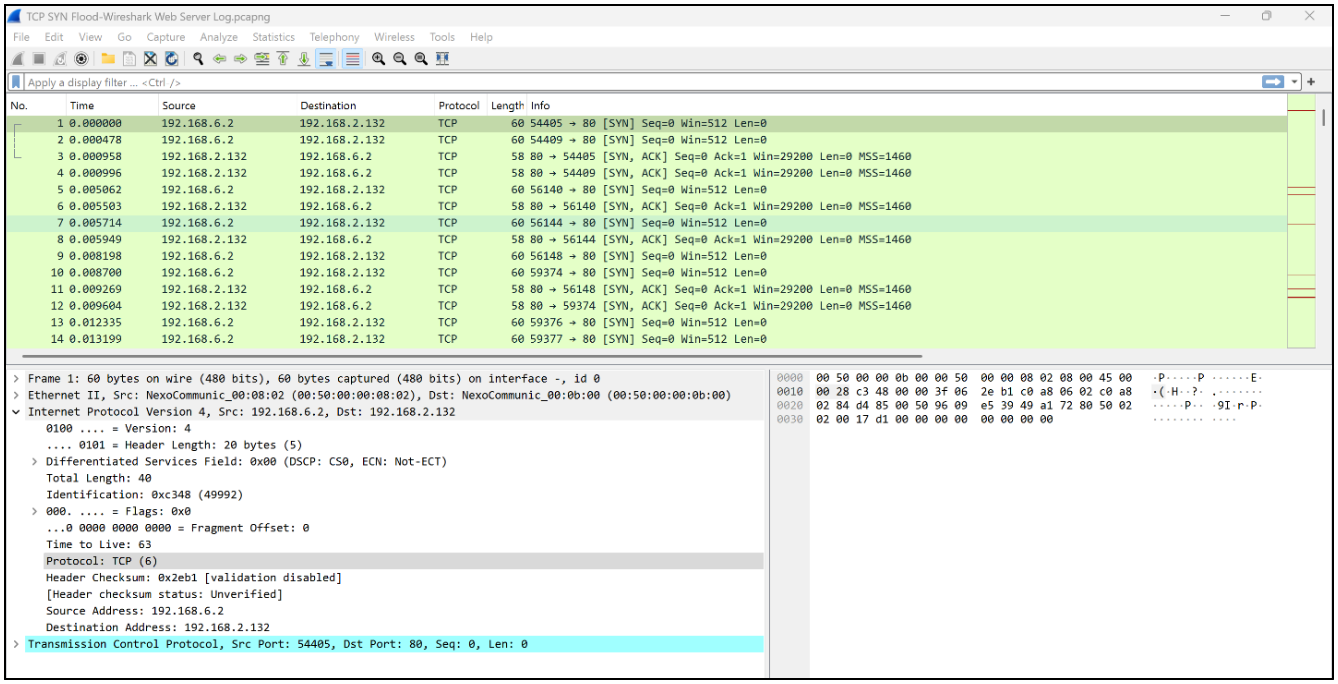
```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -S -p 80 --flood --spoof 192.168.6.2 192.168.2.132
HPING 192.168.2.132 (eth0 192.168.2.132): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
█
```

Figure 4.4: Execution of TCP SYN Flood Attack Command

4.3.1 Wireshark Capture Analysis of TCP SYN Attack

We run Wireshark packet captures to receive and analyze packets. While we gather raw data, the tool converts it into a readable format for detailed analysis. We extract statistical reports for incidents during the investigation process. Wireshark, as a packet and protocol analyzer, captures live data flows over the network. We use Wireshark to capture the SYN flood attack on both the Kali attacker and web server sides. We also utilize its GUI to generate accurate visual representations of network activity.

Wireshark capture of the TCP SYN flood attack is presented in Figure 4.5, where the Kali attacker with source IP 192.168.6.2 sends repeated SYN packets to the web server at destination IP 192.168.2.132 on port 80. Each packet uses unique source ports and identical sequence numbers of zero, which disrupts the server's ability to process legitimate connection requests. The server allocates resources for half-open TCP connections, which exhausts its capacity and causes denial of service for legitimate users.


TCP SYN Flood-Wireshark Web Server Log.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.  Time      Source         Destination    Protocol Length Info
   1 0.000000  192.168.6.2    192.168.2.132  TCP      60 54405 → 80 [SYN] Seq=0 Win=512 Len=0
   2 0.000478  192.168.6.2    192.168.2.132  TCP      60 54409 → 80 [SYN] Seq=0 Win=512 Len=0
   3 0.000958  192.168.2.132  192.168.6.2    TCP      58 80 → 54405 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
   4 0.000996  192.168.2.132  192.168.6.2    TCP      58 80 → 54409 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
   5 0.005062  192.168.6.2    192.168.2.132  TCP      60 56140 → 80 [SYN] Seq=0 Win=512 Len=0
   6 0.005503  192.168.2.132  192.168.6.2    TCP      58 80 → 56140 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
   7 0.005714  192.168.6.2    192.168.2.132  TCP      60 56144 → 80 [SYN] Seq=0 Win=512 Len=0
   8 0.005949  192.168.2.132  192.168.6.2    TCP      58 80 → 56144 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
   9 0.008198  192.168.6.2    192.168.2.132  TCP      60 56148 → 80 [SYN] Seq=0 Win=512 Len=0
  10 0.008700  192.168.6.2    192.168.2.132  TCP      60 59374 → 80 [SYN] Seq=0 Win=512 Len=0
  11 0.009269  192.168.2.132  192.168.6.2    TCP      58 80 → 56148 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
  12 0.009604  192.168.2.132  192.168.6.2    TCP      58 80 → 59374 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
  13 0.012335  192.168.6.2    192.168.2.132  TCP      60 59376 → 80 [SYN] Seq=0 Win=512 Len=0
  14 0.013199  192.168.6.2    192.168.2.132  TCP      60 59377 → 80 [SYN] Seq=0 Win=512 Len=0
```

```
> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface -, id 0
> Ethernet II, Src: NexoCommunic_00:08:02 (00:50:00:00:08:02), Dst: NexoCommunic_00:0b:00 (00:50:00:00:0b:00)
v Internet Protocol Version 4, Src: 192.168.6.2, Dst: 192.168.2.132
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 40
    Identification: 0xc348 (49992)
  > 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 63
    Protocol: TCP (6)
    Header Checksum: 0x2eb1 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.6.2
    Destination Address: 192.168.2.132
> Transmission Control Protocol, Src Port: 54405, Dst Port: 80, Seq: 0, Len: 0
```

```
0000  00 50 00 00 0b 00 00 50  00 00 08 02 08 00 45 00   ·P·····P ······E·
0010  00 28 c3 48 00 00 3f 06  2e b1 c0 a8 06 02 c0 a8   ·(·H··?· .·······
0020  02 84 d4 85 00 50 96 09  e5 39 49 a1 72 80 50 02   ·····P·· ·9I·r·P·
0030  02 00 17 d1 00 00 00 00  00 00 00 00               ········ ····
```

Figure 4.5: Wireshark Capture of TCP SYN Flood Attack from Kali Attacker to Web Server

A notable capability of the Wireshark packet analyzer is its use of distinct colors to represent packets from various protocols, which makes it easier for us as security engineers to identify and examine packets of different types simultaneously.

Wireshark’s I/O Graph window counts or calculates summary statistics over intervals. If a packet or field does not occur in a given interval, the calculation might yield zero. This is particularly likely for very small intervals. Figure 4.6 shows the I/O graph derived from Wireshark capture during the TCP SYN flood attack initiated by the Kali Linux attacker toward the web server.

The x-axis shows time in seconds, and the y-axis represents the packet rate in packets per second. The packet rate remains high, above 1000 packets per second, for most of the graph. A sharp drop in the packet rate occurs near the 400-second mark, followed by low packet rates, which specify server exhaustion. This pattern shows the effect of the attack on the server's capacity to process legitimate connections.

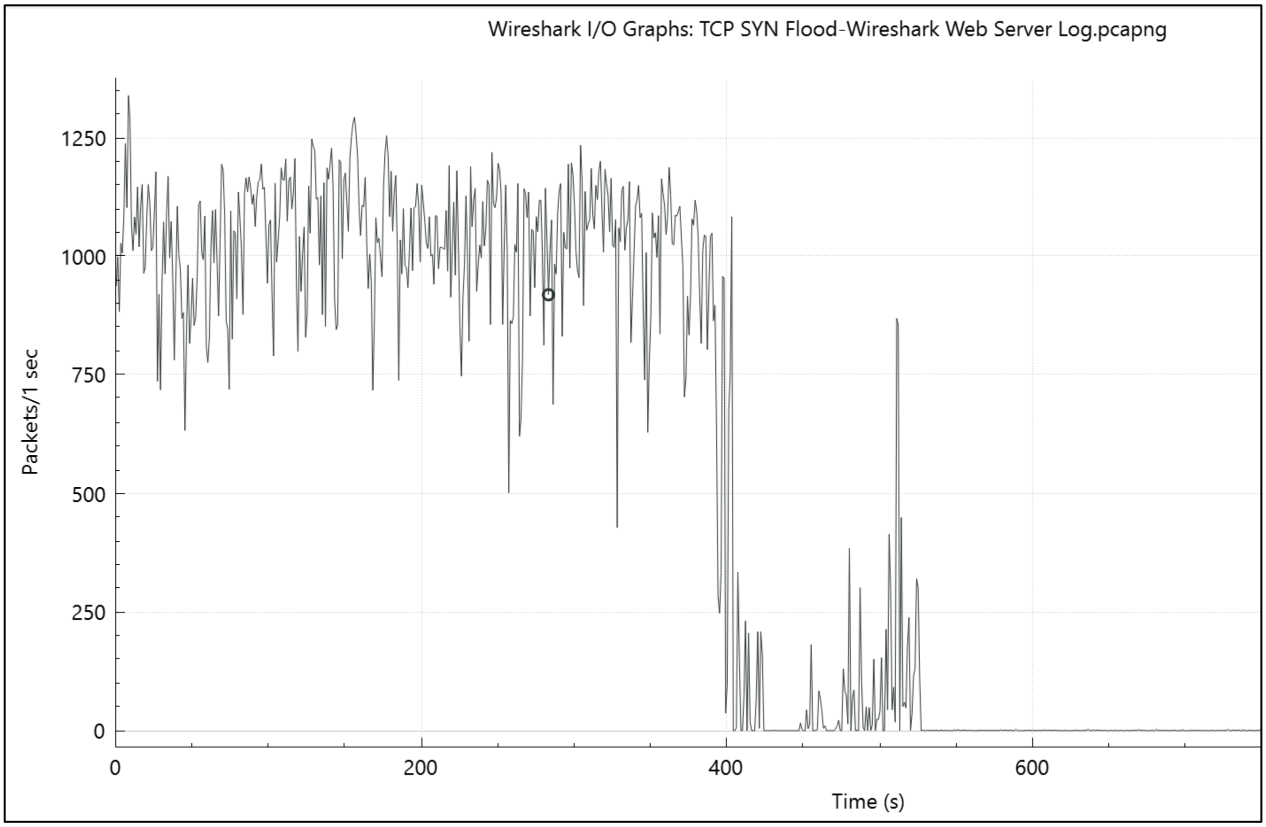


Figure 4.6: Wireshark I/O Graph of TCP SYN Flood Attack

4.4 ICMP Flood Attack Under Absence of Security Policy

An ICMP flood attack floods a network or device with a massive number of ICMP messages to exhaust its resources [88]. ICMP is a protocol designed for network management and diagnostics that allows error reporting and congestion control. In an ICMP flood attack, this protocol is misused to disrupt the target system [88]. The attacker sends a high volume of ICMP echo request packets, often called "ping" packets, to the target system [88].

Figure 4.7 represents an ICMP flood attack where an attacker uses a bot to send excessive ICMP Echo Request packets (commonly known as "ping" packets) to a target system [89]. The target replies with ICMP Echo Reply packets, which exhausts its resources and network bandwidth. This high volume of requests and replies exhausts the target’s capacity and prevents normal operations.

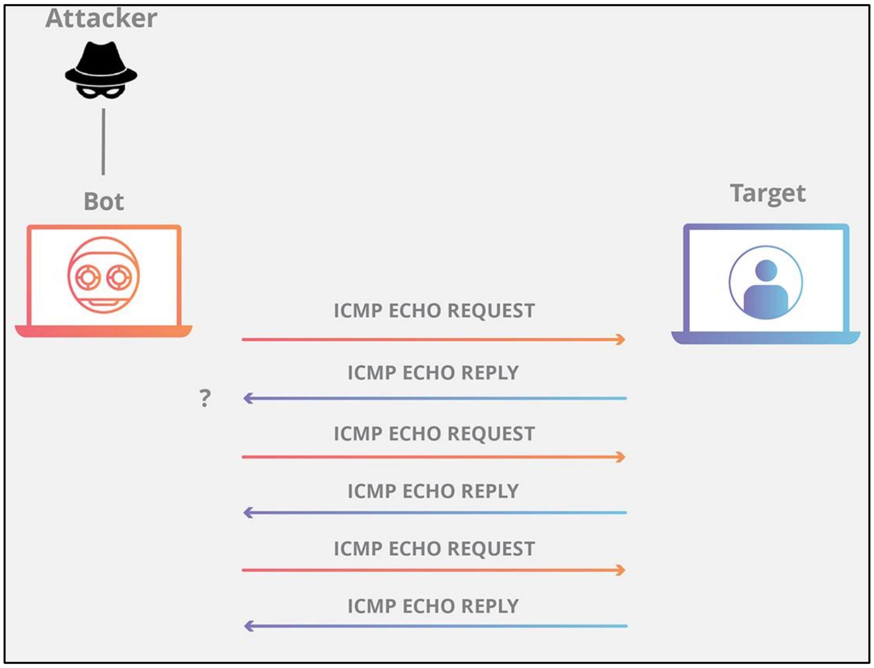
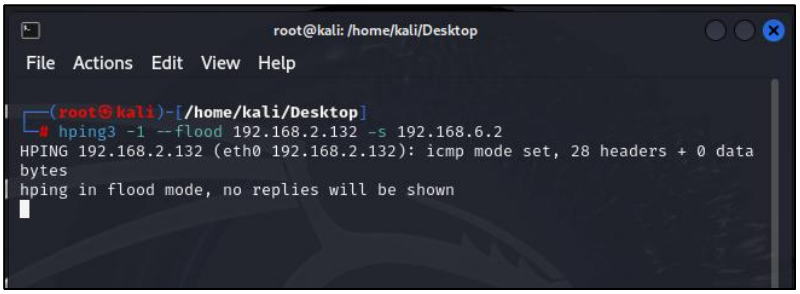


Figure 4.7: ICMP Flood Attack [89]

We perform the ICMP flood attack from the Kali attacker to the web server in our setup. We run the command hping3 -1 --flood -s 192.168.6.2 192.168.2.132 to bombard the web server at IP address 192.168.2.132 with a heavy stream of ICMP packets, as shown in Figure 4.8. In this command, the -1 option activates ICMP mode, --flood produces a rapid flow of packets, and -s

192.168.6.2 defines the attacker source IP. This attack floods the server's resources with ICMP packets to demonstrate a potential DDoS scenario.

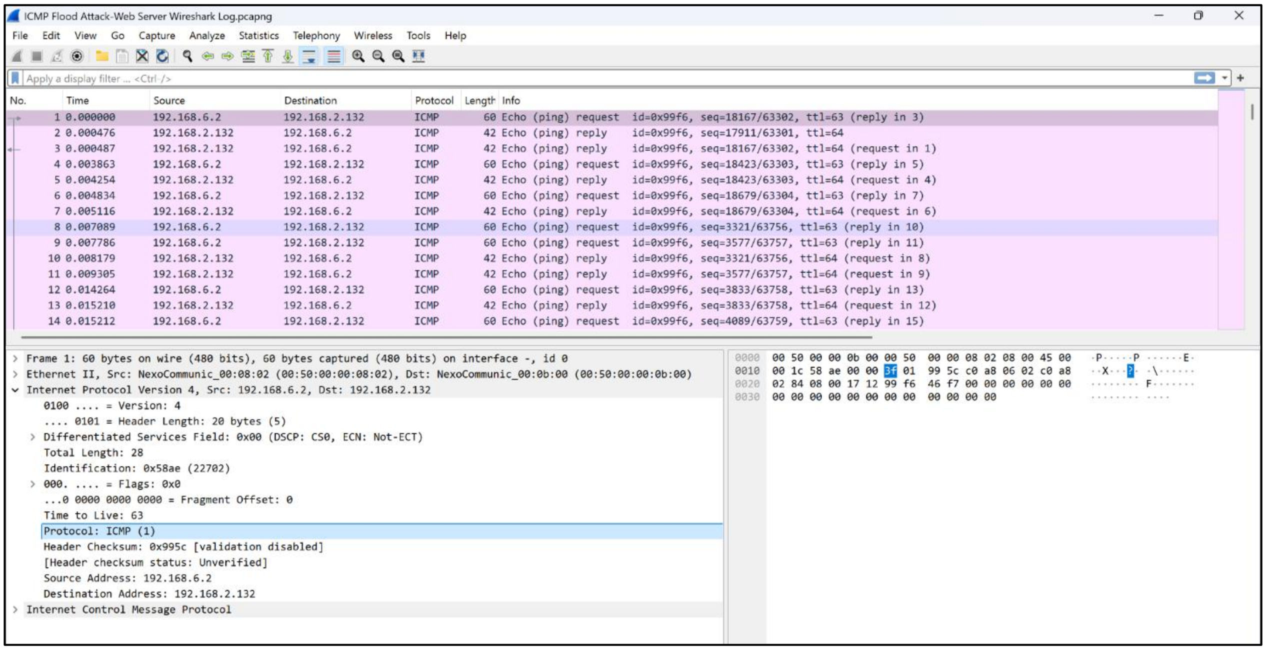


```
┌──(root㉿kali)-[/home/kali/Desktop]
└─# hping3 -1 --flood 192.168.2.132 -s 192.168.6.2
HPING 192.168.2.132 (eth0 192.168.2.132): icmp mode set, 28 headers + 0 data
bytes
hping in flood mode, no replies will be shown
```

Figure 4.8: Execution of ICMP Flood Attack Command

4.4.1 Wireshark Capture Analysis of ICMP Flood Attack

Wireshark capture of the ICMP flood attack is displayed in Figure 4.9, where the Kali attacker with source IP 192.168.6.2 generates multiple ICMP echo requests to the web server at destination IP 192.168.2.132. The packets show identical Time to Live (TTL) values with sequential identifiers and consistent payload sizes. The excessive number of packets sent in rapid succession exhausts the server’s processing capacity, which results in disruption to normal operations and potential denial of service for valid users.


ICMP Flood Attack-Web Server Wireshark Log.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.    Time        Source          Destination     Protocol  Length Info
     1 0.000000    192.168.6.2     192.168.2.132   ICMP      60 Echo (ping) request  id=0x99f6, seq=18167/63302, ttl=63 (reply in 3)
     2 0.000476    192.168.2.132   192.168.6.2     ICMP      42 Echo (ping) reply    id=0x99f6, seq=17911/63301, ttl=64
     3 0.000487    192.168.2.132   192.168.6.2     ICMP      42 Echo (ping) reply    id=0x99f6, seq=18167/63302, ttl=64 (request in 1)
     4 0.003863    192.168.6.2     192.168.2.132   ICMP      60 Echo (ping) request  id=0x99f6, seq=18423/63303, ttl=63 (reply in 5)
     5 0.004254    192.168.2.132   192.168.6.2     ICMP      42 Echo (ping) reply    id=0x99f6, seq=18423/63303, ttl=64 (request in 4)
     6 0.004834    192.168.6.2     192.168.2.132   ICMP      60 Echo (ping) request  id=0x99f6, seq=18679/63304, ttl=63 (reply in 7)
     7 0.005116    192.168.2.132   192.168.6.2     ICMP      42 Echo (ping) reply    id=0x99f6, seq=18679/63304, ttl=64 (request in 6)
     8 0.007089    192.168.6.2     192.168.2.132   ICMP      60 Echo (ping) request  id=0x99f6, seq=3321/63756, ttl=63 (reply in 10)
     9 0.007786    192.168.6.2     192.168.2.132   ICMP      60 Echo (ping) request  id=0x99f6, seq=3577/63757, ttl=63 (reply in 11)
    10 0.008179    192.168.2.132   192.168.6.2     ICMP      42 Echo (ping) reply    id=0x99f6, seq=3321/63756, ttl=64 (request in 8)
    11 0.009305    192.168.2.132   192.168.6.2     ICMP      42 Echo (ping) reply    id=0x99f6, seq=3577/63757, ttl=64 (request in 9)
    12 0.014264    192.168.6.2     192.168.2.132   ICMP      60 Echo (ping) request  id=0x99f6, seq=3833/63758, ttl=63 (reply in 13)
    13 0.015210    192.168.2.132   192.168.6.2     ICMP      42 Echo (ping) reply    id=0x99f6, seq=3833/63758, ttl=64 (request in 12)
    14 0.015212    192.168.6.2     192.168.2.132   ICMP      60 Echo (ping) request  id=0x99f6, seq=4089/63759, ttl=63 (reply in 15)
```

```
> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface -, id 0
> Ethernet II, Src: NexoCommunic_00:08:02 (00:50:00:00:08:02), Dst: NexoCommunic_00:0b:00 (00:50:00:00:0b:00)
v Internet Protocol Version 4, Src: 192.168.6.2, Dst: 192.168.2.132
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 28
    Identification: 0x58ae (22702)
  > 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 63
    Protocol: ICMP (1)
    Header Checksum: 0x995c [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.6.2
    Destination Address: 192.168.2.132
> Internet Control Message Protocol
```

```
0000  00 50 00 00 0b 00 00 50  00 00 08 02 08 00 45 00   ·P····P ······E·
0010  00 1c 58 ae 00 00 3f 01  99 5c c0 a8 06 02 c0 a8   ··X···?· ·\······
0020  02 84 08 00 17 12 99 f6  46 f7 00 00 00 00 00 00   ········ F·······
0030  00 00 00 00 00 00 00 00  00 00 00 00               ········ ····
```

Figure 4.9: Wireshark Capture of ICMP Flood Attack from Kali Attacker to Web Server

Figure 4.10 represents the Wireshark I/O graph of an ICMP flood launched from the Kali machine, which targets the web server. The x-axis details the elapsed time in seconds, while the y-axis reflects the number of packets transmitted per second. For the majority of the period, the packet count remains at elevated levels, peaking near 1000 packets per second. A sudden decline is visible toward the end of the graph, signaling resource depletion on the server. This behavior demonstrates significant strain on the server's capacity that disrupts its normal operations.

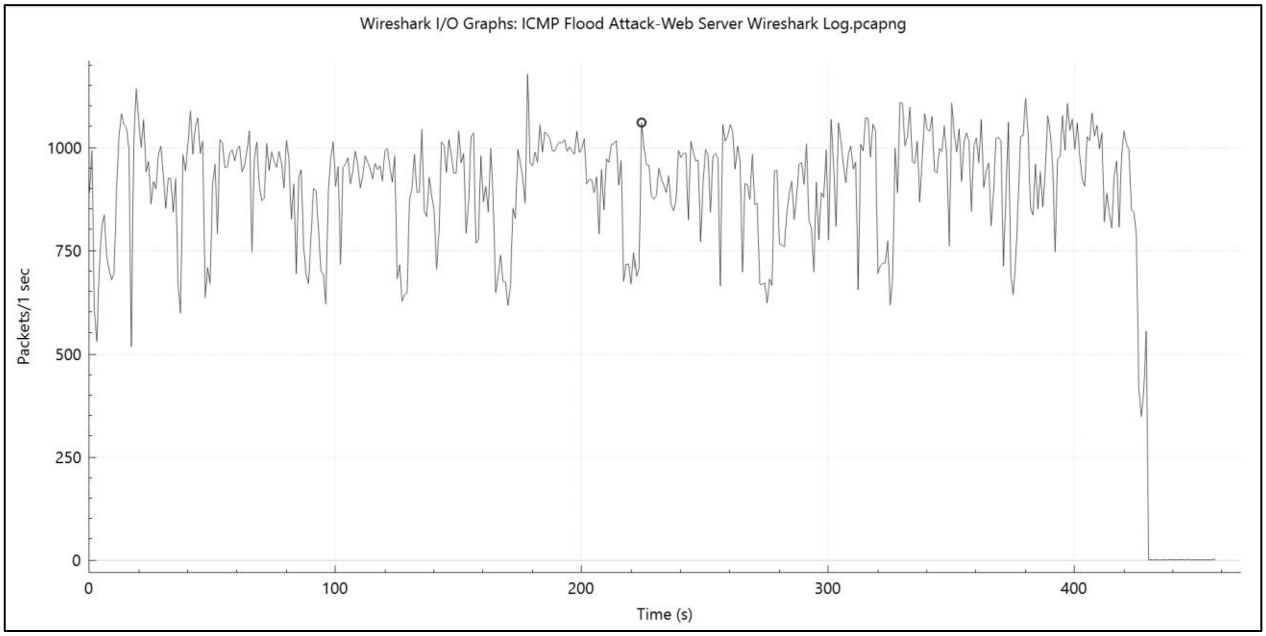


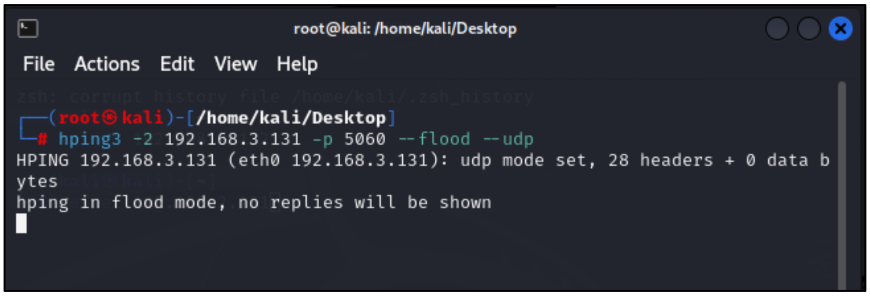
Figure 4.10: Wireshark I/O Graph of ICMP Flood Attack

4.5 SIP UDP Flood Attack Under Absence of Security Policy

A SIP UDP flood is an application layer DDoS attack that aims to exhaust the resources of a targeted SIP user agent (server or client) and bring a denial-of-service state to the SIP service [90]. This attack sends repeated OPTIONS requests to a SIP user agent over UDP to render it unable to process new connections [90].

The attack sends a large volume of OPTIONS requests to an SIP user agent on UDP port 5060 or another active SIP port [90]. These requests come from multiple malicious machines and force the SIP user agent to reply with SIP traffic. The resource exhaustion occurs directly on the victim SIP user agent and disrupts its normal operation within the network [90].

We launch the SIP UDP flood command from the Kali machine against the SIP endpoint, which is the responsibility of the Ostinato traffic generator in our setup. The command hping3 -2 192.168.3.131 -p 5060 --flood --udp initiates a high-volume stream of UDP packets directed at the SIP endpoint with IP address 192.168.3.131 on port 5060, as shown in Figure 4.11. Here, the -2 option sets UDP as the protocol, -p 5060 points to the SIP communication port, and --flood enables rapid and continuous transmission of packets. This activity forces the endpoint to handle excessive traffic, leading to resource depletion and the interruption of normal SIP service.

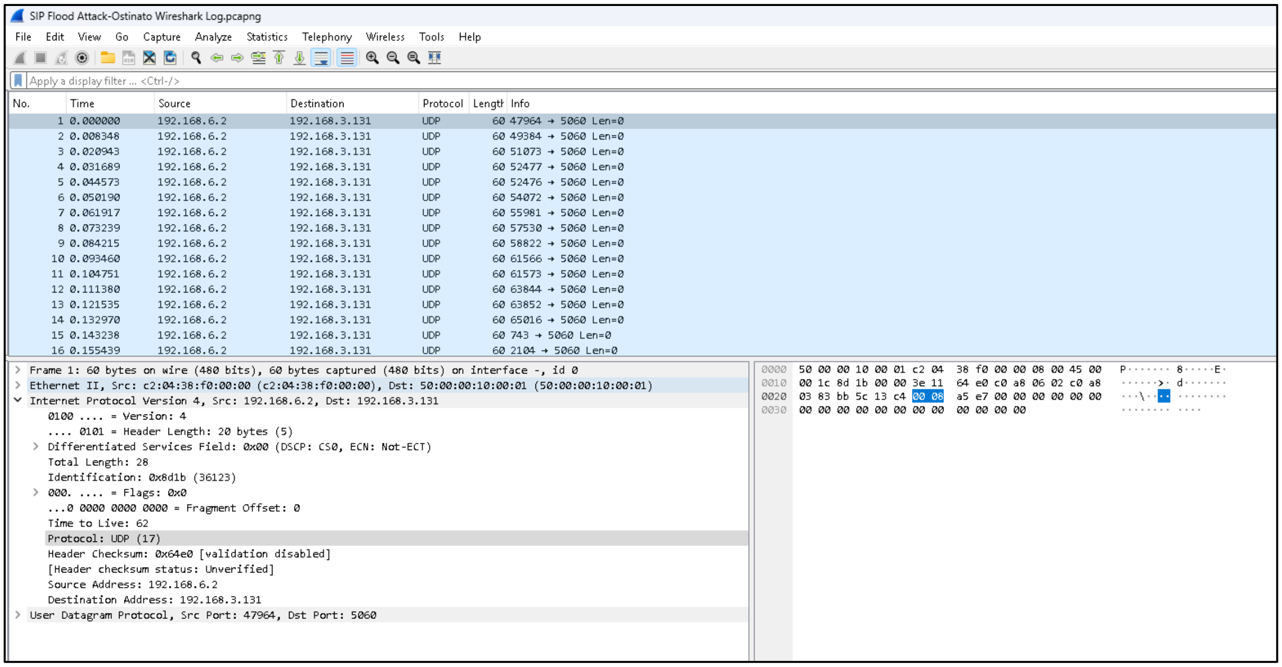


```
┌──(root㉿kali)-[/home/kali/Desktop]
└─# hping3 -2 192.168.3.131 -p 5060 --flood --udp
HPING 192.168.3.131 (eth0 192.168.3.131): udp mode set, 28 headers + 0 data b
ytes
hping in flood mode, no replies will be shown
█
```

Figure 4.11: Execution of SIP UDP Flood Attack Command

4.5.1 Wireshark Capture Analysis of SIP UDP Flood Attack

Wireshark capture of the SIP UDP flood attack is shown in Figure 4.12, where the Kali attacker with source IP 192.168.6.2 sends a continuous stream of UDP packets to the Ostinato traffic generator at destination IP 192.168.3.131 on port 5060. Each packet has identical payloads and varying source ports, which causes the target to consume resources unnecessarily. The SIP endpoint struggles to handle the large volume of packets, leading to resource exhaustion and service disruption. This attack demonstrates the impact of high-volume UDP traffic on the target system's capacity.



SIP Flood Attack-Ostinato Wireshark Log.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.   Time       Source        Destination    Protocol Length Info
    1 0.000000   192.168.6.2   192.168.3.131  UDP      60 47964 → 5060 Len=0
    2 0.008348   192.168.6.2   192.168.3.131  UDP      60 49384 → 5060 Len=0
    3 0.020943   192.168.6.2   192.168.3.131  UDP      60 51073 → 5060 Len=0
    4 0.031689   192.168.6.2   192.168.3.131  UDP      60 52477 → 5060 Len=0
    5 0.044573   192.168.6.2   192.168.3.131  UDP      60 52476 → 5060 Len=0
    6 0.050190   192.168.6.2   192.168.3.131  UDP      60 54072 → 5060 Len=0
    7 0.061917   192.168.6.2   192.168.3.131  UDP      60 55981 → 5060 Len=0
    8 0.073239   192.168.6.2   192.168.3.131  UDP      60 57530 → 5060 Len=0
    9 0.084215   192.168.6.2   192.168.3.131  UDP      60 58822 → 5060 Len=0
   10 0.093460   192.168.6.2   192.168.3.131  UDP      60 61566 → 5060 Len=0
   11 0.104751   192.168.6.2   192.168.3.131  UDP      60 61573 → 5060 Len=0
   12 0.111380   192.168.6.2   192.168.3.131  UDP      60 63844 → 5060 Len=0
   13 0.121535   192.168.6.2   192.168.3.131  UDP      60 63852 → 5060 Len=0
   14 0.132970   192.168.6.2   192.168.3.131  UDP      60 65016 → 5060 Len=0
   15 0.143238   192.168.6.2   192.168.3.131  UDP      60 743 → 5060 Len=0
   16 0.155439   192.168.6.2   192.168.3.131  UDP      60 2104 → 5060 Len=0
```

```
> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface -, id 0
> Ethernet II, Src: c2:04:38:f0:00:00 (c2:04:38:f0:00:00), Dst: 50:00:00:10:00:01 (50:00:00:10:00:01)
v Internet Protocol Version 4, Src: 192.168.6.2, Dst: 192.168.3.131
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 28
    Identification: 0x8d1b (36123)
  > 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 62
    Protocol: UDP (17)
    Header Checksum: 0x64e0 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.6.2
    Destination Address: 192.168.3.131
> User Datagram Protocol, Src Port: 47964, Dst Port: 5060
```

```
0000  50 00 00 10 00 01 c2 04  38 f0 00 00 08 00 45 00   P······· 8·····E·
0010  00 1c 8d 1b 00 00 3e 11  64 e0 c0 a8 06 02 c0 a8   ······>· d·······
0020  03 83 bb 5c 13 c4 00 08  a5 e7 00 00 00 00 00 00   ···\···· ········
0030  00 00 00 00 00 00 00 00  00 00 00 00               ········ ····
```

Figure 4.12: Wireshark Capture of SIP UDP Flood Attack from Kali Attacker to SIP Endpoint

Figure 4.13 shows the I/O graph of the SIP UDP flood attack from the Kali attacker to the Ostinato traffic generator, which acts as a SIP endpoint. The x-axis represents time in seconds, and the y-axis shows the number of packets sent per second. The graph shows irregular packet rates, with peaks at 100 packets per second and several drops to zero. These variations indicate burst traffic designed to overwhelm the target. The sharp drops reflect moments of reduced packet flow from resource depletion. This attack shows the effect of UDP flood traffic on the target's ability to manage incoming requests.

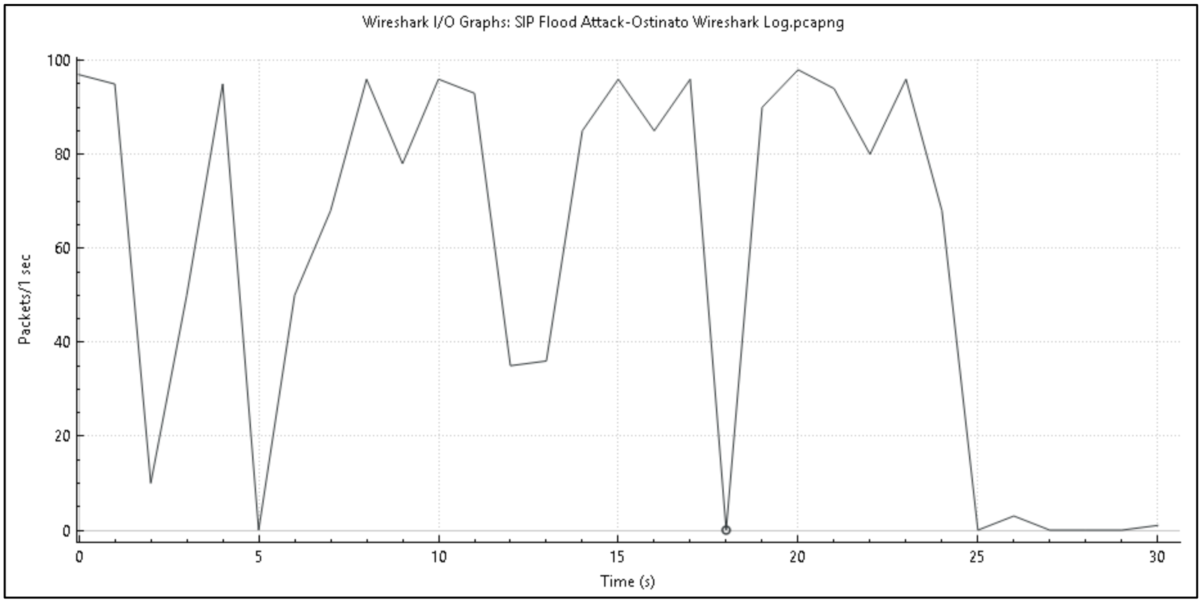


Figure 4.13: Wireshark I/O Graph of SIP UDP Flood Attack

The analysis of various DDoS attacks, such as TCP SYN floods, ICMP floods, and SIP UDP floods, shows how attackers use protocol flaws to exhaust target resources and cause service disruption. Each attack has specific traffic patterns and packet attributes visible in Wireshark captures. Before implementing security policies, Cisco Firepower logs in transparent mode provide essential insights into the success rate of attack flows, which enable a baseline analysis of traffic behavior without interference from active security measures.

4.6 Cisco Firepower Log Analysis in Transparent Mode

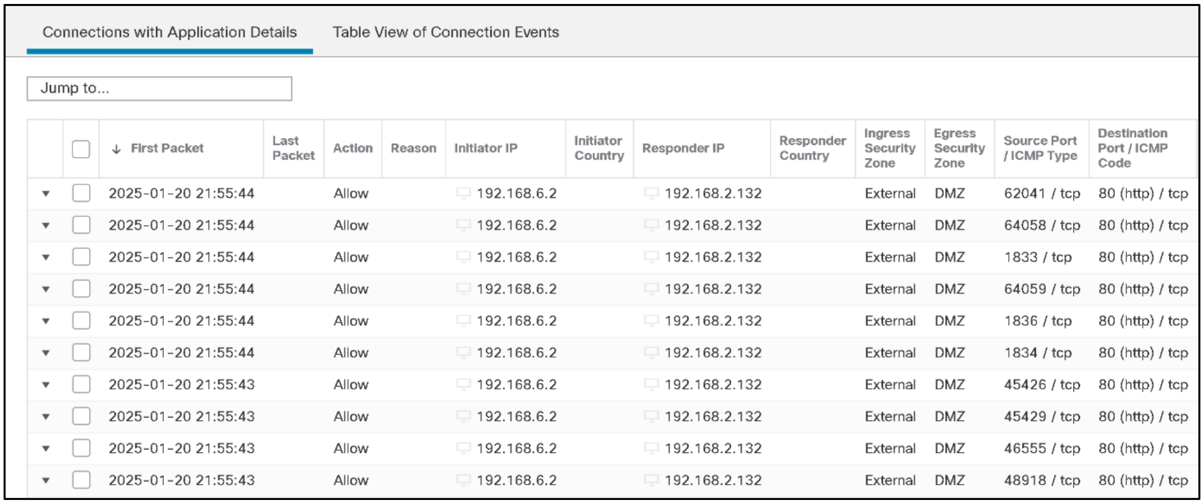
Cisco FMC is the centralized management platform for Cisco FTD security products, which are part of Cisco's NGFW solutions [91]. FMC provides visibility, control, and automation for security policies across Cisco's network security appliances, such as firewalls, IPS, and Advanced Malware Protection (AMP) [91].

The FTD system logs provide the information to monitor and troubleshoot the FTD appliance [91]. The logs are useful both in routine troubleshooting and in incident handling [91]. The FTD appliance supports both local and external logging. Local logging can help us troubleshoot the live issues [91]. External logging is a method of collecting logs from the FTD appliance to an external System Logging (Syslog) server [91].

In this stage, we configure the Cisco FTD device in transparent mode with no security policies applied. This setup allows all attack traffic to pass through the device without interference and provides a complete view of the traffic patterns. This approach captures and analyzes the details of the attack traffic. This configuration helps with the investigation of various log parameters in the Cisco FTD system, such as connection events, packet metadata, and flow statistics. These details provide a baseline to understand attack behavior and evaluate the device's log capabilities before any security policies are applied.

4.6.1 Analysis of Firepower Logs During the TCP SYN Flood Attack

We investigate the connection events captured in the Cisco FMC during the TCP SYN flood attack while the device is in transparent mode, as illustrated in Figure 4.14. The log entries indicate that the attack traffic from the initiator IP 192.168.6.2 to the responder IP 192.168.2.132 is allowed to pass through the device. The traffic flows originate from various source ports, such as 62041 and 64058, and target the destination port 80 with the protocol set to TCP. The ingress security zone is marked as External, while the egress security zone is labeled as DMZ, which provides the path of the traffic through the network. Each entry reflects the source and destination IPs, ports, and protocols without any block action to confirm that no security policy interferes with the flow of the attack traffic.



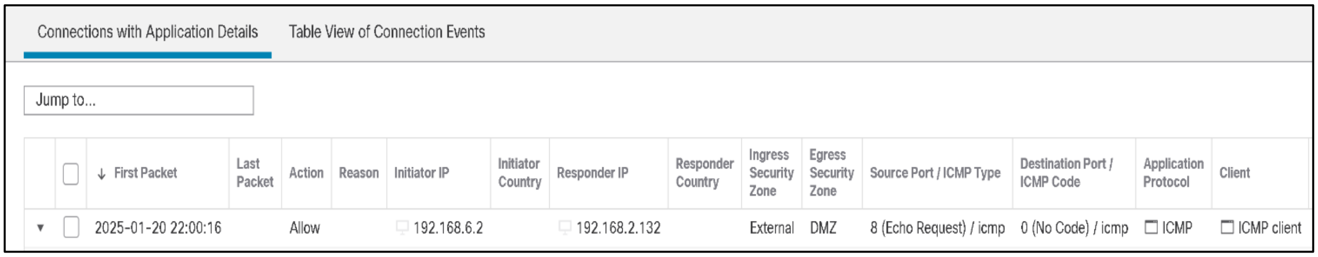
Connections with Application Details
Table View of Connection Events
Jump to...

	□	↓ First Packet	Last Packet	Action	Reason	Initiator IP	Initiator Country	Responder IP	Responder Country	Ingress Security Zone	Egress Security Zone	Source Port / ICMP Type	Destination Port / ICMP Code
▾	□	2025-01-20 21:55:44		Allow		192.168.6.2		192.168.2.132		External	DMZ	62041 / tcp	80 (http) / tcp
▾	□	2025-01-20 21:55:44		Allow		192.168.6.2		192.168.2.132		External	DMZ	64058 / tcp	80 (http) / tcp
▾	□	2025-01-20 21:55:44		Allow		192.168.6.2		192.168.2.132		External	DMZ	1833 / tcp	80 (http) / tcp
▾	□	2025-01-20 21:55:44		Allow		192.168.6.2		192.168.2.132		External	DMZ	64059 / tcp	80 (http) / tcp
▾	□	2025-01-20 21:55:44		Allow		192.168.6.2		192.168.2.132		External	DMZ	1836 / tcp	80 (http) / tcp
▾	□	2025-01-20 21:55:44		Allow		192.168.6.2		192.168.2.132		External	DMZ	1834 / tcp	80 (http) / tcp
▾	□	2025-01-20 21:55:43		Allow		192.168.6.2		192.168.2.132		External	DMZ	45426 / tcp	80 (http) / tcp
▾	□	2025-01-20 21:55:43		Allow		192.168.6.2		192.168.2.132		External	DMZ	45429 / tcp	80 (http) / tcp
▾	□	2025-01-20 21:55:43		Allow		192.168.6.2		192.168.2.132		External	DMZ	46555 / tcp	80 (http) / tcp
▾	□	2025-01-20 21:55:43		Allow		192.168.6.2		192.168.2.132		External	DMZ	48918 / tcp	80 (http) / tcp

Figure 4.14: Firepower FTD Log Capture During the TCP SYN Flood Attack in Transparent Mode

4.6.2 Analysis of Firepower Logs During the ICMP Flood Attack

We analyze the connection events logged by Cisco FMC during the ICMP flood attack in transparent mode, as shown in Figure 4.15. The attack originates from the source IP 192.168.6.2 as the Kali attacker and targets the destination IP 192.168.2.132, which is the web server in our architecture. The logs indicate that ICMP echo requests, identified as type Echo Request, are traversing the system without restriction. The traffic flows from the External security zone, where the attacker segments to the DMZ security zone, the web server location, with all actions marked as Allow to verify no filtering or policy enforcement in the Firepower NGFW.



Connections with Application Details | Table View of Connection Events

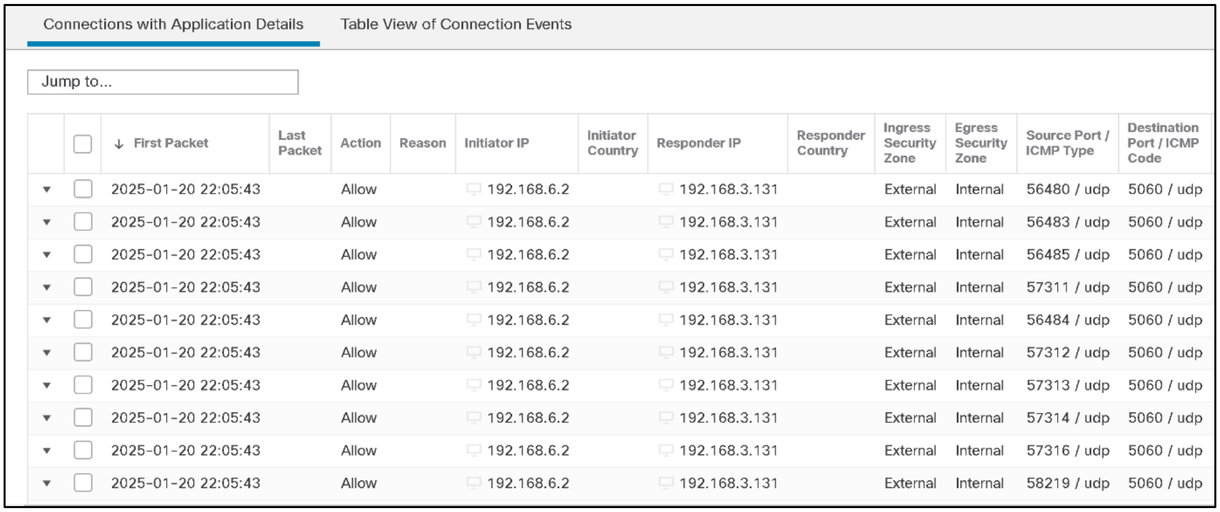
Jump to...

	□	↓ First Packet	Last Packet	Action	Reason	Initiator IP	Initiator Country	Responder IP	Responder Country	Ingress Security Zone	Egress Security Zone	Source Port / ICMP Type	Destination Port / ICMP Code	Application Protocol	Client
▾	□	2025-01-20 22:00:16		Allow		192.168.6.2		192.168.2.132		External	DMZ	8 (Echo Request) / icmp	0 (No Code) / icmp	ICMP	ICMP client

Figure 4.15: Firepower FTD Log Capture During the ICMP Flood Attack in Transparent Mode

4.6.3 Analysis of Firepower Logs During the SIP UDP Flood Attack

We examine connection logs from Cisco FMC during the SIP UDP flood incident, with the device set to transparent mode, as shown in Figure 4.16. The flood originates at 192.168.6.2, which is the Kali attacker IP address toward the SIP endpoint at 192.168.3.131 over UDP port 5060. Each record in the table indicates Allow for the action to signify no active policy blocks the traffic. The data flows from the External zone into the Internal zone, with varying source ports such as 56480 and 57311, while the target port consistently remains 5060. This log focuses on details of the unimpeded traffic with insights into the behavior of UDP-based SIP flood patterns. The configuration allows unrestricted observation of the attack to aid in baseline traffic analysis and preparatory stages for policy design.



Connections with Application Details | Table View of Connection Events

Jump to...

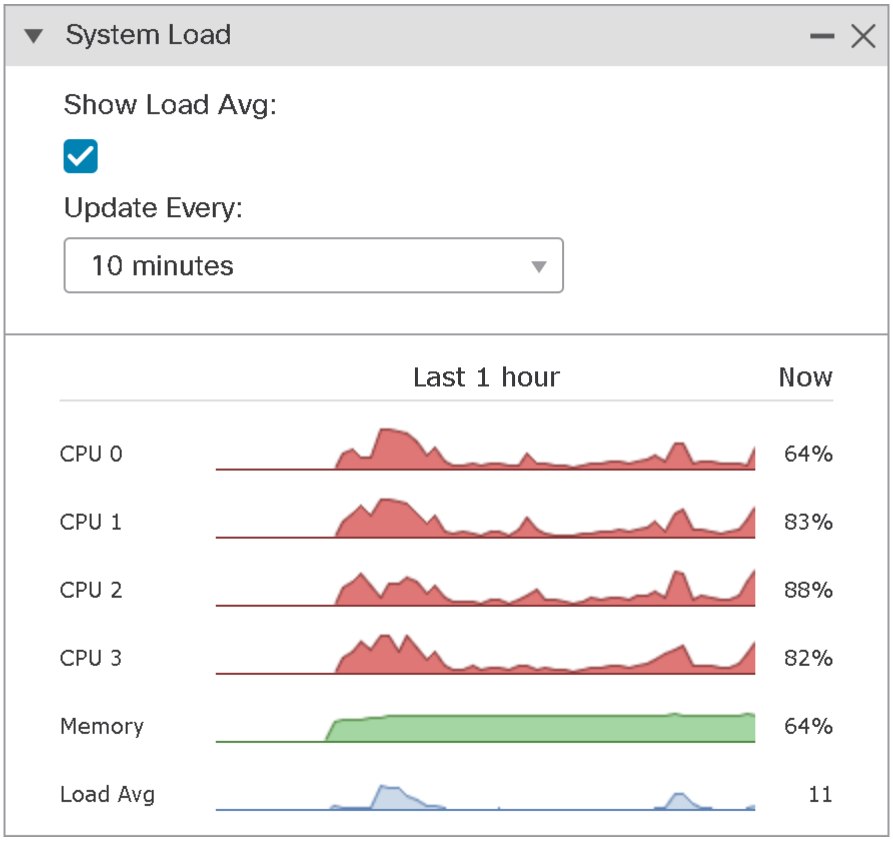
	□	↓ First Packet	Last Packet	Action	Reason	Initiator IP	Initiator Country	Responder IP	Responder Country	Ingress Security Zone	Egress Security Zone	Source Port / ICMP Type	Destination Port / ICMP Code
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	56480 / udp	5060 / udp
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	56483 / udp	5060 / udp
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	56485 / udp	5060 / udp
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	57311 / udp	5060 / udp
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	56484 / udp	5060 / udp
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	57312 / udp	5060 / udp
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	57313 / udp	5060 / udp
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	57314 / udp	5060 / udp
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	57316 / udp	5060 / udp
▾	□	2025-01-20 22:05:43		Allow		192.168.6.2		192.168.3.131		External	Internal	58219 / udp	5060 / udp

Figure 4.16: Firepower FTD Log Capture During the SIP UDP Flood Attack in Transparent Mode

4.6.4 Analysis of Firepower Resource Usage During Flood Attacks

One of the most significant indicators of an attack is its impact on the resource usage of security devices. During an attack, aside from the traffic logs, the attack traffic that passes through security devices causes a noticeable increase in CPU and memory usage compared to non-attack periods. This resource usage directly affects the performance and efficiency of the security infrastructure. Hence, we consider this point as a critical aspect of our analysis. In our thesis methodology, we focus on the traffic characteristics and log parameters while also analyzing the system resource usage of the Cisco FTD device. By comparing the baseline resource usage during TCP SYN, ICMP flood, and SIP UDP flood attacks both with and without security policy implementation, we evaluate the capacity and limitations of the Cisco Firepower under attack conditions. This detailed approach ensures a better understanding of the impact of DDoS attacks on security devices and forms the foundation for designing more effective mitigation strategies.

We consider the resource utilization of the Cisco Firepower device in transparent mode, with attack traffic allowed to pass through without restriction. CPU usage is shown in Figure 4.17 for four cores: CPU 0 at 64%, CPU 1 at 83%, CPU 2 at 88%, and CPU 3 at 82%. These values show uneven load distribution, with some cores nearing maximum capacity. Memory usage is steady at 64%, which reflects significant resource allocation to handle the traffic. The system load average is 11 over the past hour, which highlights a high workload level relative to the device's processing capacity. This data illustrates the impact of the attack traffic on system resources and indicates strain on the CPU and memory under high volumes of attack traffic.



System Load

Show Load Avg:

☑

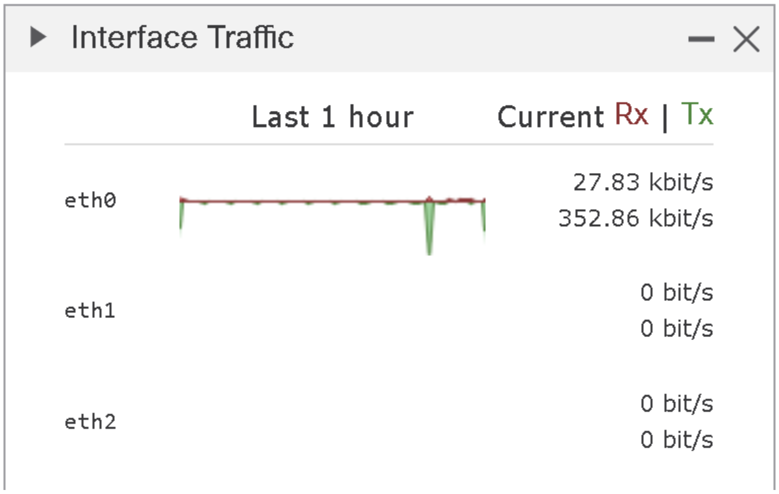
Update Every:

10 minutes

	Last 1 hour	Now
CPU 0		64%
CPU 1		83%
CPU 2		88%
CPU 3		82%
Memory		64%
Load Avg		11

Figure 4.17: Firepower Resource Utilization During the Flood Attack in Transparent Mode

We notice the interface traffic statistics of the Cisco Firepower device operating in transparent mode, with attack traffic permitted to flow through. The data reveals activity on the interface eth0, while eth1 and eth2 remain idle with no recorded traffic, as shown in Figure 4.18. The eth0 interface handles incoming traffic with a current Receive (Rx) rate of 27.83 Kilobit per second (Kbit/s) and a significantly higher Transmit (Tx) rate of 352.86 kbit/s. This disparity shows that eth0 processes inbound attack traffic and forwards a larger volume of packets after passing through the device. The high transmit rates on eth0 reflect the substantial load caused by the attack traffic passing through the device in transparent mode.



eth0 27.83 kbit/s 352.86 kbit/s
eth1 0 bit/s 0 bit/s
eth2 0 bit/s 0 bit/s

Figure 4.18: Firepower Interface Traffic During the Flood Attack in Transparent Mode

Building on the analysis of Cisco Firepower logs in transparent mode under flood attack scenarios, the next step involves employing Wazuh SIEM for the advanced detection of malicious activities. Cisco Firepower provides granular data on network behavior and resource impact, but its integration with Wazuh SIEM elevates the security strategy to a higher level. This integration combines Firepower's network-level visibility with Wazuh's ability to apply sophisticated detection rules and perform contextual analysis.

4.7 Wazuh SIEM Report Analysis for Attack Detection

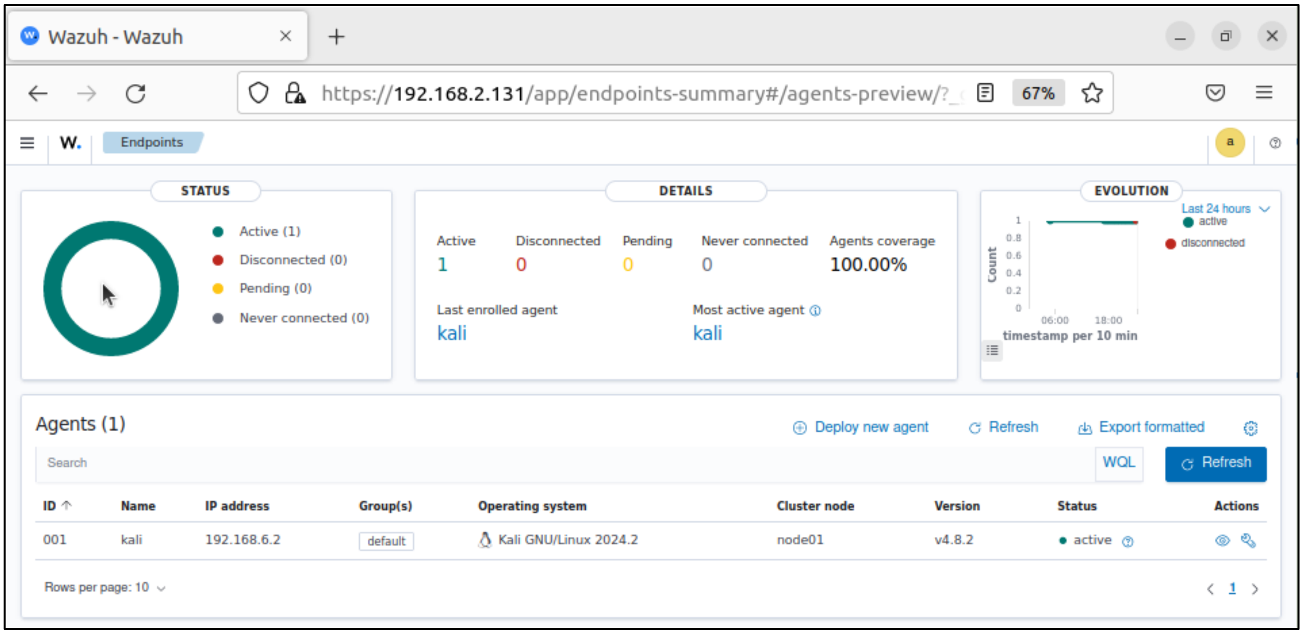
We deploy Wazuh SIEM to establish an early-stage detection mechanism for DDoS attacks and to conduct detailed log analysis during flood scenarios. By integrating Wazuh into our architecture, we enable real-time monitoring and comprehensive analysis of malicious actions associated with DDoS flood attempts. To align with the objectives of our thesis methodology, we focus on leveraging Wazuh's key features to analyze and detect threats effectively.

We use the Threat Hunt module to identify anomalous behaviors and unusual traffic patterns indicative of DDoS attack attempts. The Malware Detect feature allows us to uncover any malicious binaries or trojan files that attackers use to execute the flood attack. Through Configuration Assess, we detect misconfigurations that expose the system to exploitation. Ultimately, the Vulnerability Detect feature provides insights into known CVEs and weaknesses in the Kali system with a clear view of exploitable entry points that attackers could use.

4.7.1 Wazuh Endpoint Deployment

We deploy the Wazuh agent on the Kali Linux attacker machine to detect and analyze malicious activities across multiple metrics. This setup enables the agent to collect detailed security data for various attack scenarios, such as TCP SYN floods, ICMP floods, and SIP UDP flood attacks, as part of our thesis methodology. The Wazuh agent monitors critical system events, file changes, process behavior, and network traffic from the attacker. The collected data is sent to the Wazuh manager for centralized correlation and report generation.

The Wazuh module represents the total and active agents in the dashboard of Figure 4.19. The dashboard lists one active agent named Kali, whose IP address is 192.168.6.2, and runs on Kali GNU/Linux 2024.2. The Details section reports 100% agent coverage and identifies Kali as the last enrolled and most active agent. This dashboard validates the proper functioning of the Wazuh agent to confirm its connectivity and readiness for security event collection in the architecture.



Active	Disconnected	Pending	Never connected	Agents coverage
1	0	0	0	100.00%

Last enrolled agent
kali

Most active agent
kali

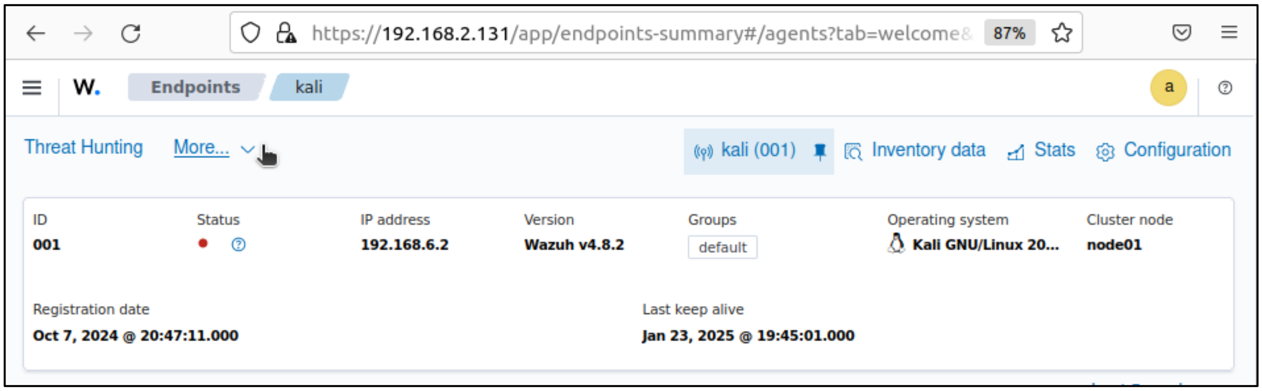
Agents (1)

ID	Name	IP address	Group(s)	Operating system	Cluster node	Version	Status	Actions
001	kali	192.168.6.2	default	Kali GNU/Linux 2024.2	node01	v4.8.2	● active	

Figure 4.19: Active Agent Status Report on the Wazuh Dashboard

Figure 4.20 represents a detailed view of the Kali attacker node in the architecture, with critical information about its status and activity. The agent, identified as 001, runs on IP address 192.168.6.2 with Wazuh agent version v4.8.2. The agent was registered on October 7, 2024, at

20:47:11, and its most recent communication with the Wazuh manager was on January 23, 2025, at 19:45:01, showing its active state.



ID	Status	IP address	Version	Groups	Operating system	Cluster node
001	●	192.168.6.2	Wazuh v4.8.2	default	Kali GNU/Linux 20...	node01

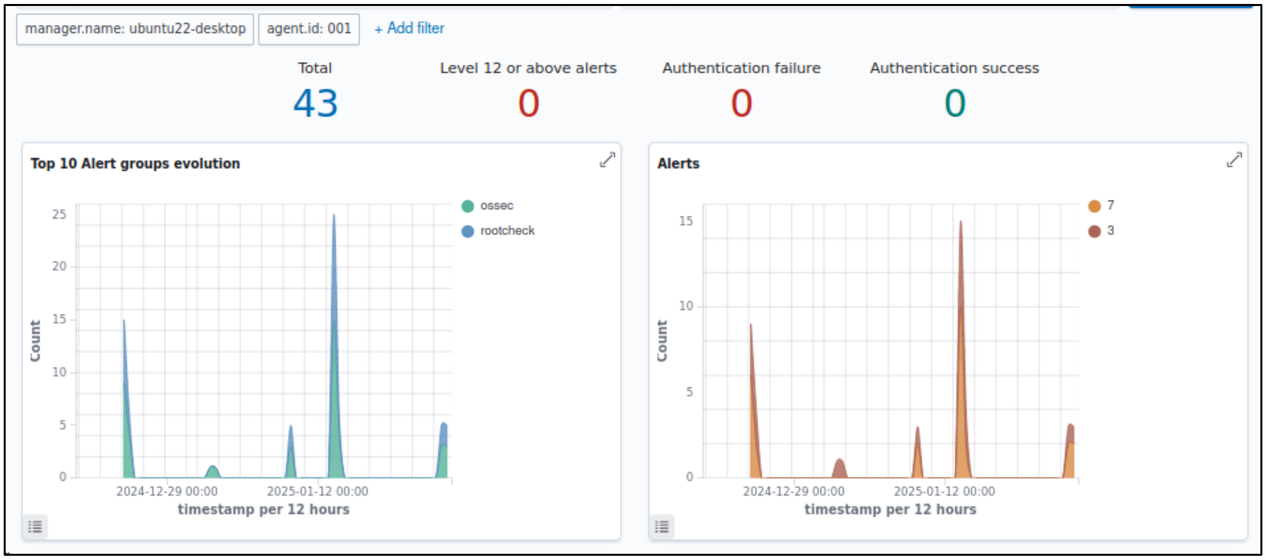
Registration date	Last keep alive
Oct 7, 2024 @ 20:47:11.000	Jan 23, 2025 @ 19:45:01.000

Figure 4.20: Detailed View of Kali Attacker on the Wazuh Dashboard

4.7.2 Wazuh Threat Hunting Report

The Wazuh dashboard provides a detailed snapshot of Threat Hunting from the Kali attacker in Figure 4.21. Over the observed period, the system recorded 43 alerts, which shows activity that requires attention but remains manageable. The Top 10 Alert Groups Evolution graph shows two primary sources of events: ossec and rootcheck. The ossec group shows a significant spike, with 25 alerts concentrated in a single 12-hour window, a clear sign of suspicious activity detected by Wazuh's integrity checks and log analysis in Kali attacker. The rootcheck group adds smaller bursts of events that are associated with checks for rootkits and unusual configurations.

The Alerts graph breaks down the timeline of events by severity, with color coding to separate levels 7 and 3. A peak in severity 7 alerts aligns with the OSSEC spike, a sign of potential impact.



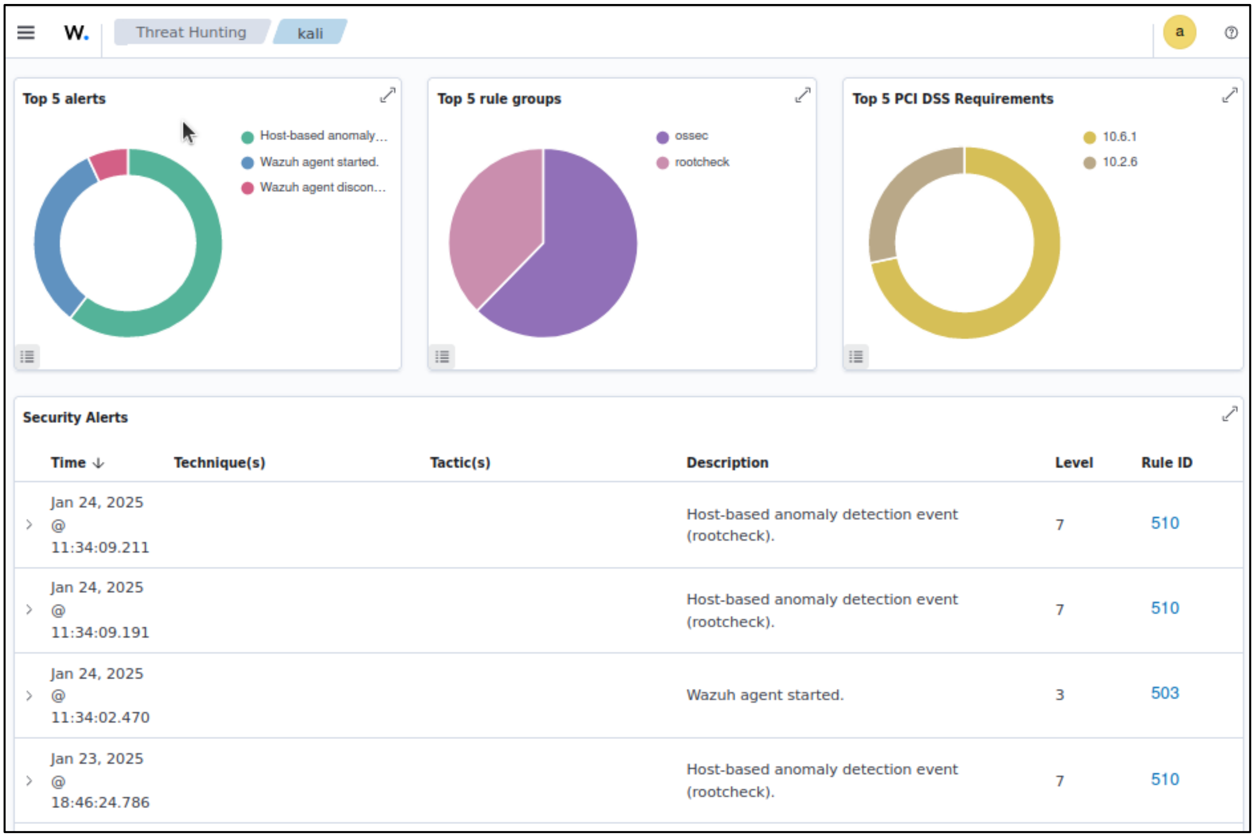
manager.name: ubuntu22-desktop | agent.id: 001 | + Add filter

Total 43 | Level 12 or above alerts 0 | Authentication failure 0 | Authentication success 0

Figure 4.21: Threat Hunting View of Kali Attacker on the Wazuh Dashboard

Moreover, the Threat Hunting report on the Wazuh dashboard underscores Host-based anomaly detection as the primary alert category for the Kali attacker, as presented in Figure 4.22. This metric comes from the rootcheck module, which scans the system for potential security anomalies, such as unauthorized changes, suspicious files, or Indicators of Compromise (IoC). The consistent presence of Host-based anomaly detection events with a severity level of 7 shows that the Kali attacker performs actions flagged as abnormal by the Wazuh agent's integrity and rootkit detection rules.

The frequent occurrence of this metric shows the importance of host-based monitoring to detect localized threats that network-level tools cannot catch. The rule ID 510 linked to these events refers to a specific set of Wazuh rules that detect anomalies in real-time. These alerts confirm that the Kali attacker behaves in ways that deviate from expected norms due to its role in simulating DDoS flood attacks within the monitored environment.



Security Alerts

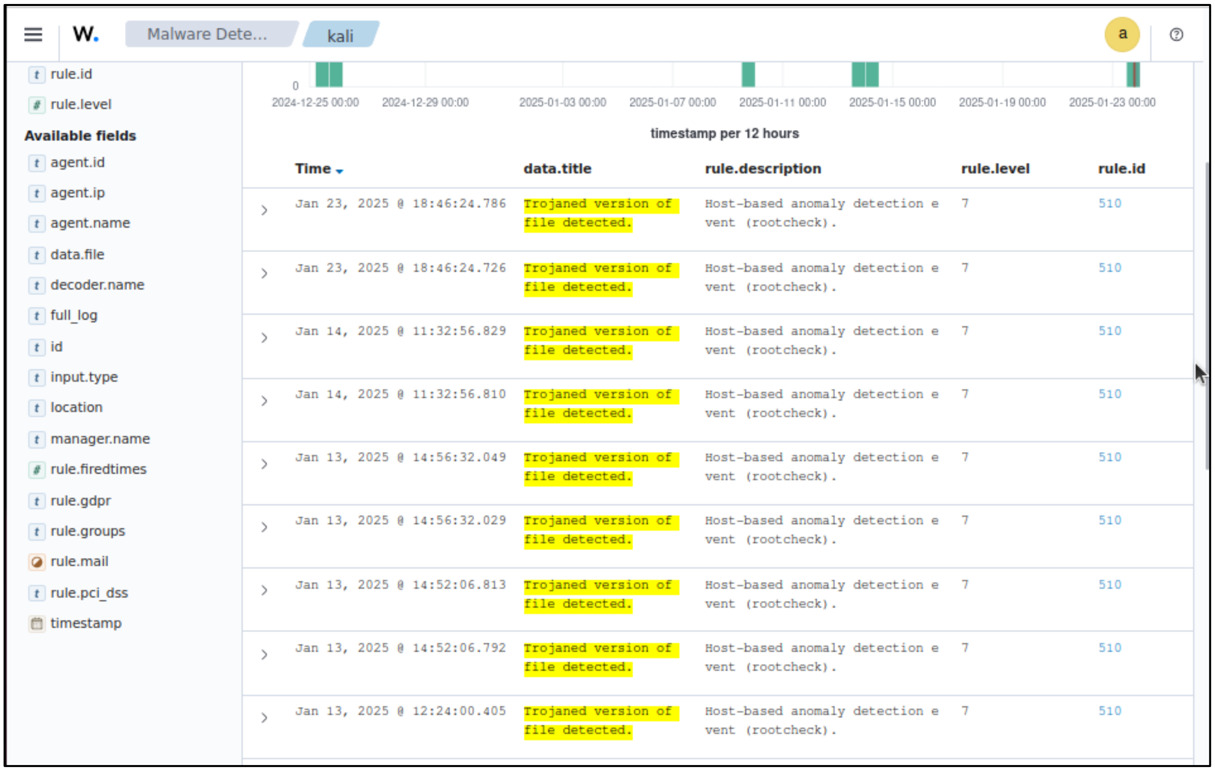
Time ↓	Technique(s)	Tactic(s)	Description	Level	Rule ID
Jan 24, 2025 @ 11:34:09.211			Host-based anomaly detection event (rootcheck).	7	510
Jan 24, 2025 @ 11:34:09.191			Host-based anomaly detection event (rootcheck).	7	510
Jan 24, 2025 @ 11:34:02.470			Wazuh agent started.	3	503
Jan 23, 2025 @ 18:46:24.786			Host-based anomaly detection event (rootcheck).	7	510

Figure 4.22: Host-Based Anomaly Detection for Kali Attacker on the Wazuh Dashboard

4.7.3 Wazuh Malware Detection Report

Wazuh provides anomaly and malware detection capabilities to monitor suspicious binaries and malicious activities on endpoints. We explore the Malware Detection section of the Kali attacker on the Wazuh dashboard in Figure 4.23, where we observe repeated alerts from the rootcheck module. The data.title column lists multiple detections of a Trojaned version of a file detected, categorized as a Host-based anomaly detection event (rootcheck).

In the context of the Kali attacker, Wazuh has identified repeated detections of Trojaned versions of system files. Such attempts typically involve the deployment of altered binaries that execute high-volume attack traffic to overwhelm target systems. The alerts report that the Kali attacker exhibits deliberate tampering with system files to simulate real-world DDoS attacks.



Time	data.title	rule.description	rule.level	rule.id
Jan 23, 2025 @ 18:46:24.786	Trojaned version of file detected.	Host-based anomaly detection event (rootcheck).	7	510
Jan 23, 2025 @ 18:46:24.726	Trojaned version of file detected.	Host-based anomaly detection event (rootcheck).	7	510
Jan 14, 2025 @ 11:32:56.829	Trojaned version of file detected.	Host-based anomaly detection event (rootcheck).	7	510
Jan 14, 2025 @ 11:32:56.810	Trojaned version of file detected.	Host-based anomaly detection event (rootcheck).	7	510
Jan 13, 2025 @ 14:56:32.049	Trojaned version of file detected.	Host-based anomaly detection event (rootcheck).	7	510
Jan 13, 2025 @ 14:56:32.029	Trojaned version of file detected.	Host-based anomaly detection event (rootcheck).	7	510
Jan 13, 2025 @ 14:52:06.813	Trojaned version of file detected.	Host-based anomaly detection event (rootcheck).	7	510
Jan 13, 2025 @ 14:52:06.792	Trojaned version of file detected.	Host-based anomaly detection event (rootcheck).	7	510
Jan 13, 2025 @ 12:24:00.405	Trojaned version of file detected.	Host-based anomaly detection event (rootcheck).	7	510

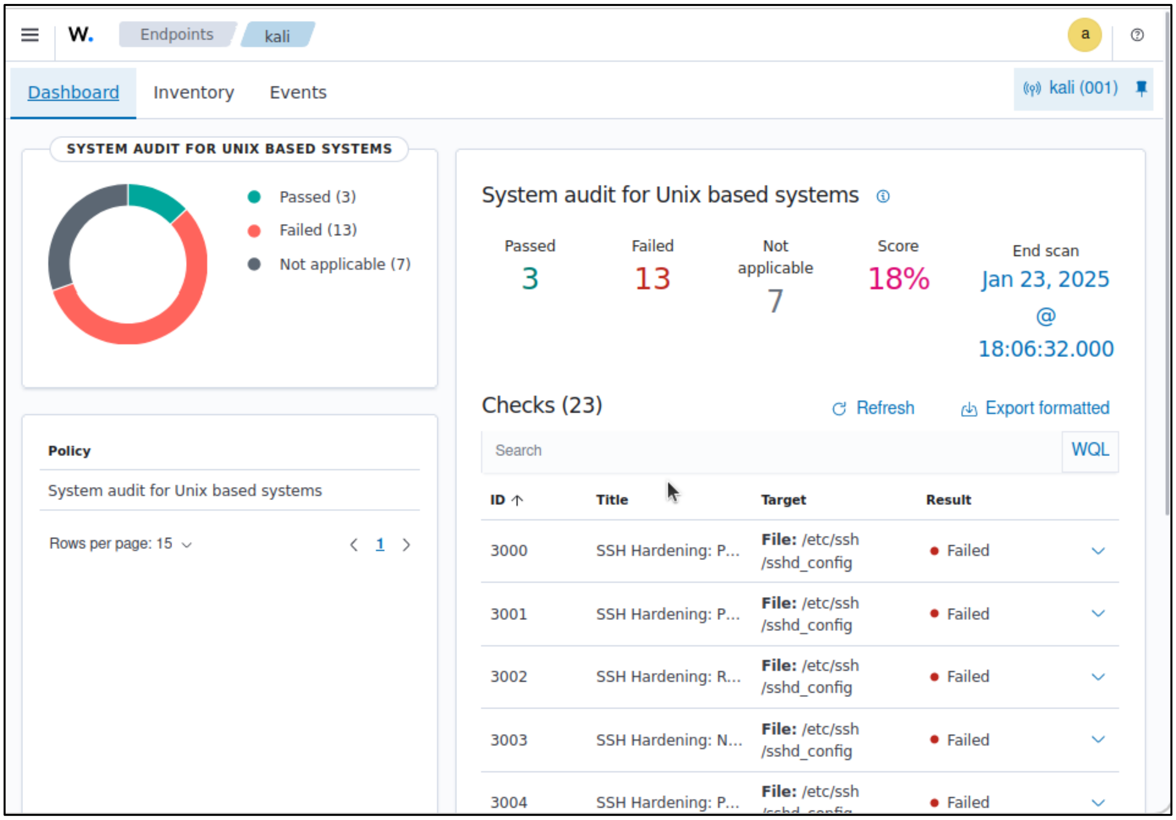
Figure 4.23: Malware Detection for Kali Attacker on the Wazuh Dashboard

4.7.4 Wazuh Configuration Assessment Report

Configuration assessment is the process of verifying that endpoints comply with a defined set of rules related to configuration settings and approved application usage. This process evaluates the current system setup against recognized industry standards and organizational policies to detect potential vulnerabilities and improper configurations.

The results of a baseline security audit conducted on the Kali attacker node are displayed in Figure 4.24. The audit focuses on Unix-based systems and evaluates the compliance of the Kali node against predefined security hardening policies. A total of 23 checks were performed, with only 3 checks passed successfully, 13 failed, and 7 marked as not applicable. The overall compliance score is 18%, which signals critical vulnerabilities in the system's configuration.

The findings acknowledge significant vulnerabilities, which show that the Kali attacker is not adequately hardened to withstand potential threats.



SYSTEM AUDIT FOR UNIX BASED SYSTEMS

- ● Passed (3)
- ● Failed (13)
- ● Not applicable (7)

Policy

System audit for Unix based systems

Rows per page: 15 ‹ 1 ›

System audit for Unix based systems

Passed	Failed	Not applicable	Score	End scan
3	13	7	18%	Jan 23, 2025 @ 18:06:32.000

Checks (23) Refresh Export formatted

Search WQL

ID ↑	Title	Target	Result
3000	SSH Hardening: P...	**File:** /etc/ssh /sshd_config	● Failed
3001	SSH Hardening: P...	**File:** /etc/ssh /sshd_config	● Failed
3002	SSH Hardening: R...	**File:** /etc/ssh /sshd_config	● Failed
3003	SSH Hardening: N...	**File:** /etc/ssh /sshd_config	● Failed
3004	SSH Hardening: P...	**File:** /etc/ssh	● Failed

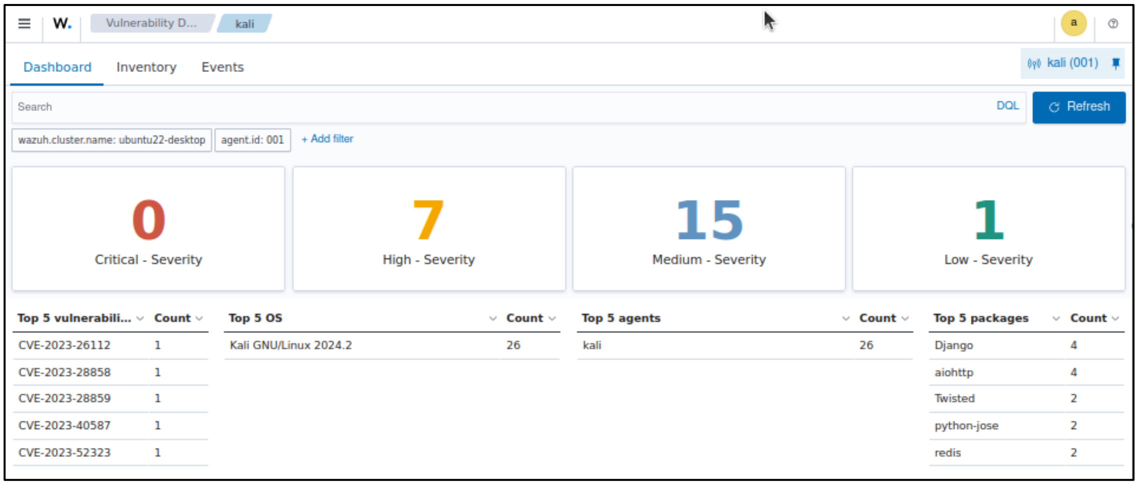
Figure 4.24: Configuration Assessment of Kali Attacker on the Wazuh Dashboard

4.7.5 Wazuh Vulnerability Detection Report

Vulnerabilities are security flaws in computer systems that threat actors can exploit to gain unauthorized access to these systems. After exploitation, malware and threat actors may be able to perform remote code execution, exfiltrate data, and carry out other malicious activities. The Wazuh vulnerability detection module helps us discover vulnerabilities in the operating system and applications installed on the monitored endpoint.

The vulnerability detection report for the Kali attacker node is presented in Figure 4.25, with a breakdown of 7 high-severity, 15 medium-severity, and 1 low-severity vulnerability. No critical vulnerabilities are reported. Key software packages, including Django and Asynchronous HTTP (aiohttp), show security flaws in essential application components.

Most vulnerabilities fall within base scores of 5.0 to 7.5, which places them in the moderate to high-risk categories.



0 Critical - Severity

7 High - Severity

15 Medium - Severity

1 Low - Severity

Top 5 vulnerabili...	Count
CVE-2023-26112	1
CVE-2023-28858	1
CVE-2023-28859	1
CVE-2023-40587	1
CVE-2023-52323	1

Top 5 OS	Count
Kali GNU/Linux 2024.2	26

Top 5 agents	Count
kali	26

Top 5 packages	Count
Django	4
aiohttp	4
Twisted	2
python-jose	2
redis	2

Figure 4.25: Vulnerability Detection of Kali Attacker on the Wazuh Dashboard

The chronological arrangement of events provides a comprehensive view of the Kali system's behavior, which constructs a clear pathway for analyzing unusual actions. This data underscores Wazuh’s role in detecting anomalies and equipping us as security engineers with actionable insights to address threats effectively within the architecture.

After a detailed analysis of Cisco Firepower and Wazuh SIEM reports during flood attacks in transparent mode, valuable insights are obtained about traffic patterns, resource usage, and the device's behavior under high-volume attack scenarios. With this comprehensive baseline established, it is now the right time to move to the next phase, which is policy deployment with the aim of the DDoS mitigation approach.

4.8 Cisco Firepower Policy Deployment

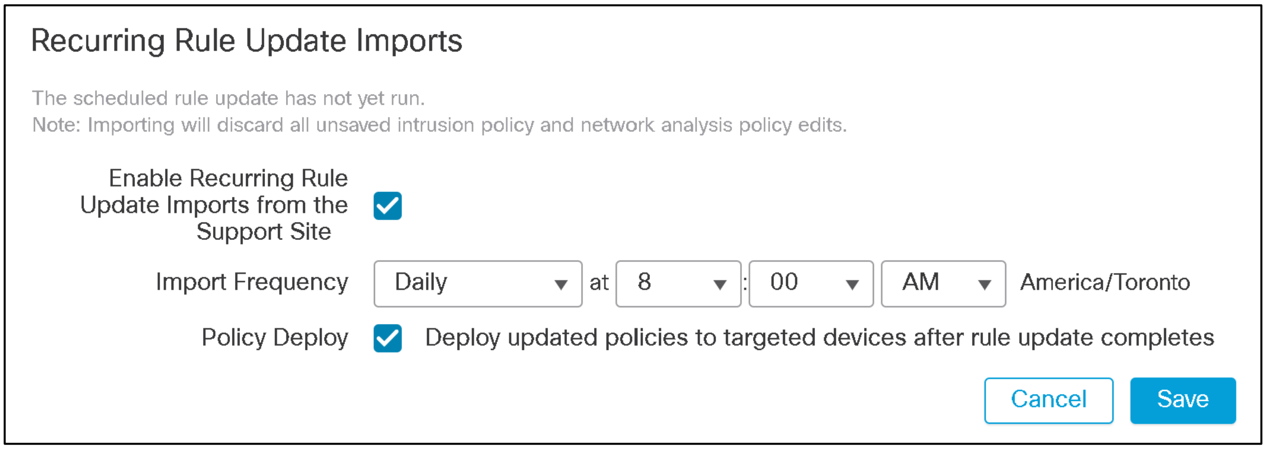
The stage after log analysis and the discovery of DDoS attacks requires immediate and decisive action to mitigate and block flood traffic to prevent harmful impacts and service disruption on the internal servers. This action is essential for the availability, integrity, and stability of the system under attack. According to our thesis methodology, this process uses the advanced security capabilities of Cisco Firepower to move from detection to active prevention.

To block the attack traffic, we configure critical security factors on Cisco Firepower. These include access control policies to filter malicious packets, intrusion prevention rules to detect and terminate attack flows, rate-limit policies to control excessive traffic from suspicious sources, and deep packet inspection rules to identify hidden threats in the traffic.

4.8.1 Rule and Geolocation Updates

As new vulnerabilities become known, the Cisco Talos Intelligence Group releases intrusion rule updates. These updates affect intrusion rules, preprocessor rules, and the policies that use the rules. Intrusion rule updates are cumulative, and Cisco recommends that always import the latest update. Therefore, we implement the Rule Update management interface of Cisco FMC to maintain up-to-date intrusion policies and network analysis rules, as shown in Figure 4.26.

In the Recurring Rule Update Imports section, we enable recurring rule updates. This option enables the system to schedule updates to retrieve the latest rules from the official site at a specified frequency and time. We define the daily routine at 8 AM and select Policy Deploy to ensure the rules are applied to targeted devices once updated and click Save to confirm the schedule.



Recurring Rule Update Imports

The scheduled rule update has not yet run.
Note: Importing will discard all unsaved intrusion policy and network analysis policy edits.

Enable Recurring Rule Update Imports from the Support Site ☑

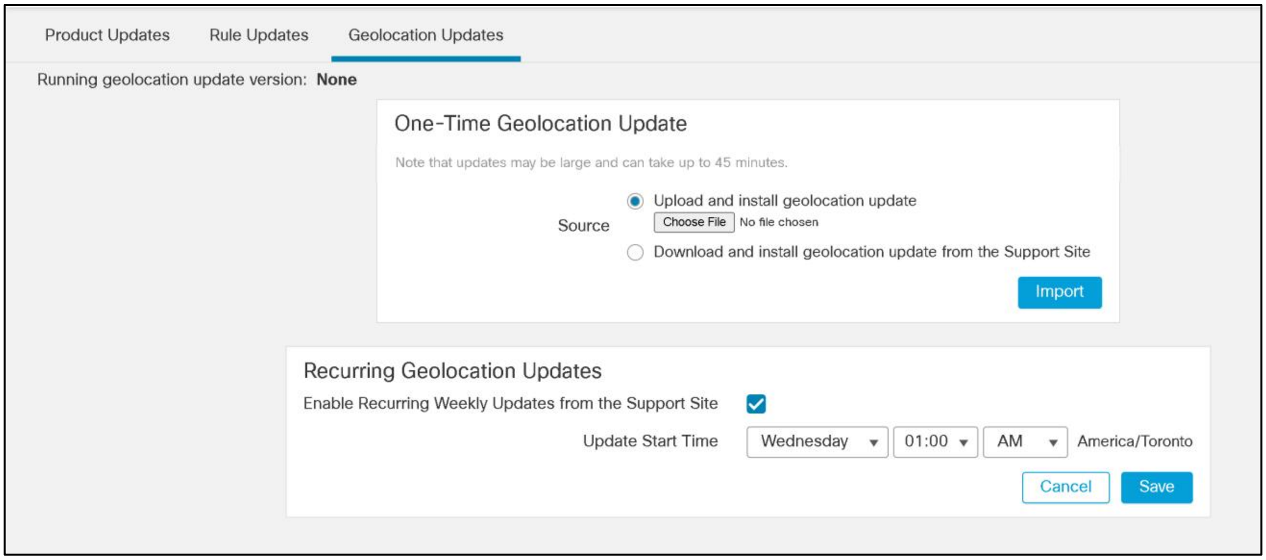
Import Frequency: Daily at 8 : 00 AM America/Toronto

Policy Deploy ☑ Deploy updated policies to targeted devices after rule update completes

Figure 4.26: Rule Update Management Interface in Cisco FMC

Additionally, we implement the Geolocation Update feature in Cisco FMC, which enables updating geolocation databases to enhance network traffic analysis and security, as shown in Figure 4.27.

In the Recurring Geolocation Updates section, we enable weekly updates by selecting the checkbox. We define the update schedule by setting the day, time, and time zone for automated updates. Then, we confirm the recurring schedule to allow the system to download and apply the updates automatically from the official Cisco site.



Running geolocation update version: **None**

One-Time Geolocation Update

Note that updates may be large and can take up to 45 minutes.

Source
● Upload and install geolocation update
No file chosen
○ Download and install geolocation update from the Support Site

Recurring Geolocation Updates

Enable Recurring Weekly Updates from the Support Site ☑

Update Start Time: Wednesday ▾ 01:00 ▾ AM ▾ America/Toronto

Figure 4.27: Geolocation Update in Cisco FMC

4.8.2 Access Control Policy Configuration

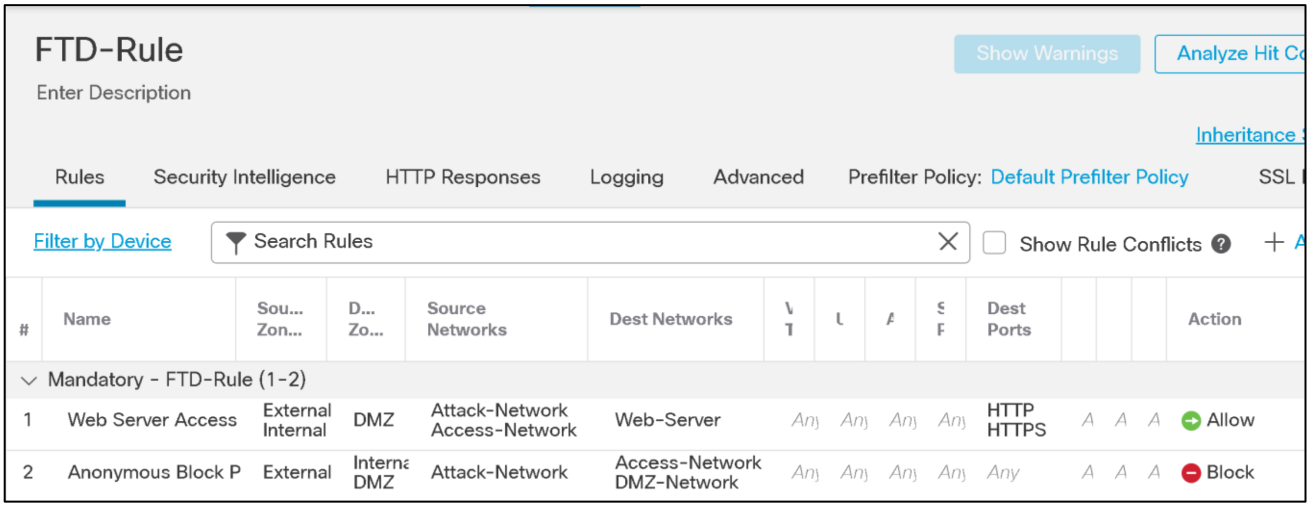
FTD access control policy is the first policy that we implement since it is the most normal policy that anyone configure in any firewall. With the FTD ACL policy, we can not only filter the traffic in layer 3 and layer 4, but many applications and micro-applications can be controlled through the ACL policy.

We configure FTD Rules in Cisco FMC to limit access control between different network zones in the architecture. Two rules are visible in Figure 4.28 in the Mandatory - FTD-Rule section, set to control traffic flow and enforce security policies.

The first rule, labeled Web Server Access Policy, allows traffic from the external and internal zones to the DMZ. The source networks are Attack and Access zones, and the destination network is the Web server. The action for this rule is Allow, specifically for HTTP and HTTPS protocols to provide legitimate web traffic to reach the web server.

The second rule, labeled Anonymous Block Policy, blocks traffic from the external zone to the internal and DMZ. The source network is the Attack Network, and the destination network includes both the Access and DMZ networks. The action for this rule is Block to prevent unauthorized or suspicious access from the attack network to sensitive internal resources.

These rules establish a balance between allowing necessary traffic for web services and blocking unauthorized or malicious traffic targeting internal resources.



FTD-Rule

Enter Description

#	Name	Sou... Zon...	D... Zo...	Source Networks	Dest Networks	V T	U	A	S P	Dest Ports				Action
Mandatory - FTD-Rule (1-2)														
1	Web Server Access	External Internal	DMZ	Attack-Network Access-Network	Web-Server	Any	Any	Any	Any	HTTP HTTPS	A	A	A	● Allow
2	Anonymous Block P	External	Interna DMZ	Attack-Network	Access-Network DMZ-Network	Any	Any	Any	Any	Any	A	A	A	● Block

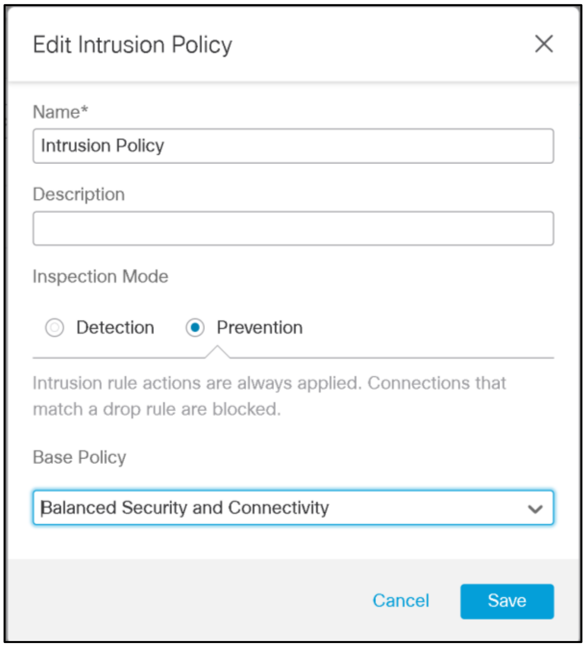
Figure 4.28: Access Control Policy in Cisco FMC

4.8.3 Intrusion Policy Configuration

An intrusion policy uses intrusion and preprocessor rules, which are collectively known as intrusion rules, to examine the decoded packets for attacks based on patterns. The rules can either prevent (drop) the threatening traffic and generate an event or simply detect (alert) it and generate an event only. As the system analyzes traffic, the network analysis decoding and preprocessing phase occurs before and separately from the intrusion prevention phase. Together, network analysis and intrusion policies provide broad and deep packet inspection. They can help us detect, alert, and protect against network traffic that could threaten the availability, integrity, and confidentiality of hosts and their data.

Figure 4.29 shows the configuration interface for an intrusion policy in the Cisco FMC, which is part of our architecture used to protect the network from malicious DDoS attacks. We set the dialog field for name entry as an intrusion policy.

The inspection mode is set to Prevention, which blocks any traffic that matches predefined intrusion rules and ensures real-time defense against identified threats. The base policy selected is Balanced Security and Connectivity, which integrates robust security measures with optimized network performance to prevent disruptions while maintaining system protection. This policy provides a strategic balance between threat defense and system availability.



Edit Intrusion Policy

Name*

Intrusion Policy

Description

Inspection Mode

○ Detection ● Prevention

Intrusion rule actions are always applied. Connections that match a drop rule are blocked.

Base Policy

Balanced Security and Connectivity

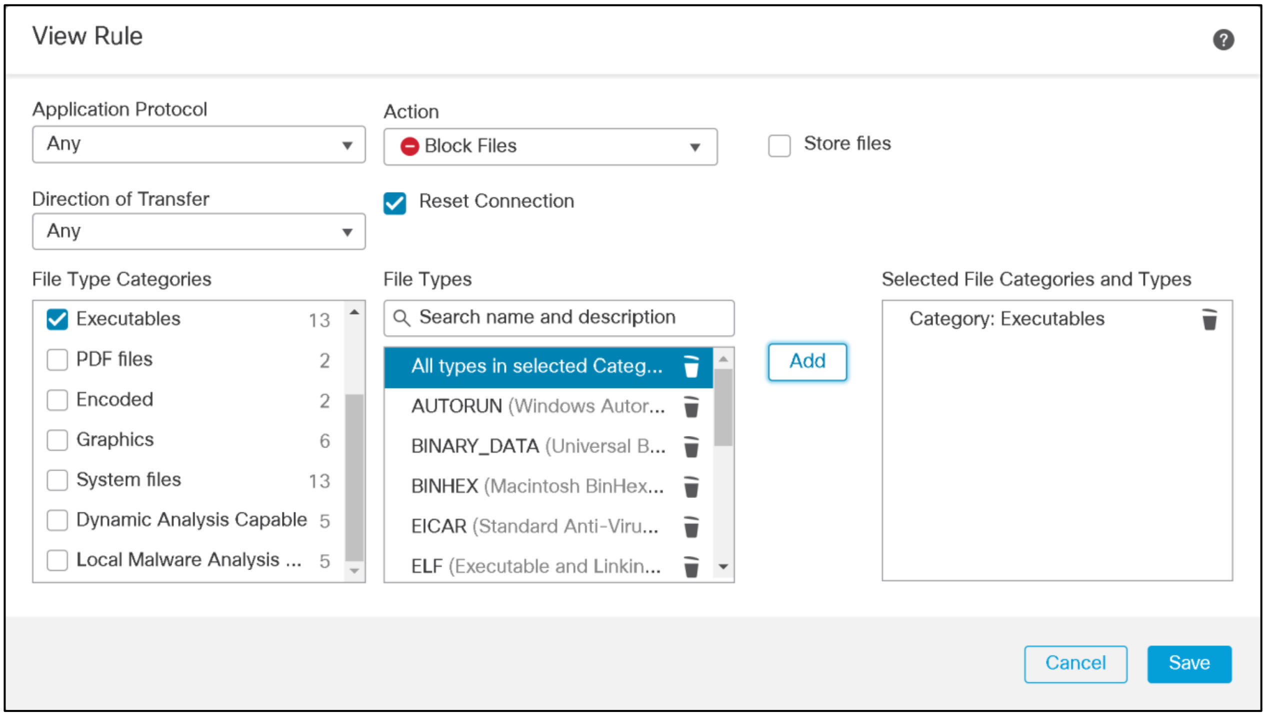
Figure 4.29: Intrusion Policy Configuration in Cisco FMC

4.8.4 Malware and File Policy Configuration

A Malware and File policy is a set of configurations that the secure firewall uses to control file transmission and protect a network from malware. It is part of the overall access control configuration. Associating a file policy to an access control rule ensures that before the system passes a file in traffic that matches an access control rule's condition, it first inspects the file.

We configure file and malware policy within Cisco FMC, as shown in Figure 4.30, designed to block and manage potentially malicious files in our network. The policy specifies the application protocol as Any, meaning the policy applies to all protocols without restriction. The action is set to Block Files to verify that specified file types are denied access. The Reset Connection option is enabled, immediately terminating connections associated with blocked files to prevent further risk.

The File Type Categories section highlights a selected category, Executables, which encompasses 13 file types. These are common executable files often targeted by attackers for malware distribution. The right panel confirms that all types within the Executables category are included in the policy. This configuration establishes the active policy to prevent the transfer of executable files that match the defined criteria to minimize the risk of malware intrusion within the architecture.



View Rule

Application Protocol: Any

Action: Block Files

□ Store files

Direction of Transfer: Any

☑ Reset Connection

File Type Categories

☑	Executables	13
□	PDF files	2
□	Encoded	2
□	Graphics	6
□	System files	13
□	Dynamic Analysis Capable	5
□	Local Malware Analysis ...	5

File Types

Search name and description

- All types in selected Categ...
- AUTORUN (Windows Autor...
- BINARY_DATA (Universal B...
- BINHEX (Macintosh BinHex...
- EICAR (Standard Anti-Viru...
- ELF (Executable and Linkin...

Selected File Categories and Types

Category: Executables

Figure 4.30: Malware and File Policy Configuration in Cisco FMC

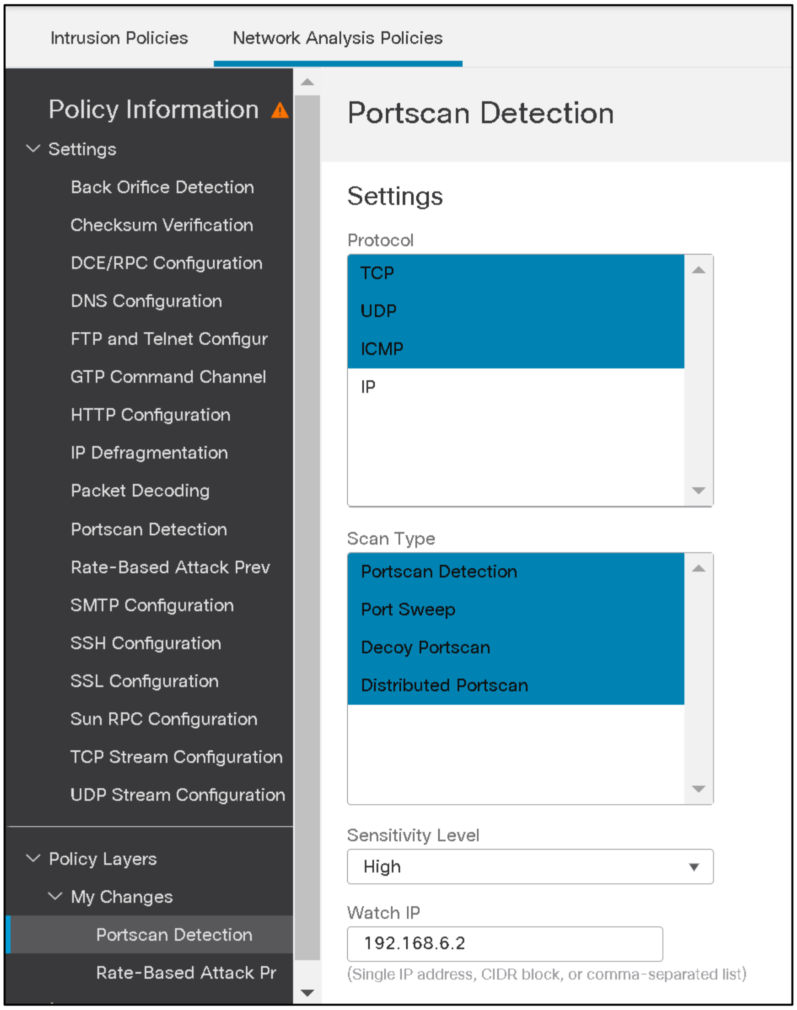
4.8.5 Network Analysis Configuration

Network analysis governs many traffic preprocessing options and is invoked by advanced settings in our access control policy. Network analysis-related preprocessing occurs after security intelligence matching and SSL decryption but before intrusion or file inspection begins. By default, the system uses the balanced security and connectivity network analysis policy to preprocess all traffic handled by an access control policy.

The configuration details for portscan detection under the Network Analysis Policy section of Cisco FMC are illustrated in Figure 4.31, designed to improve the security of our network architecture. The configuration identifies reconnaissance attempts, which act as precursors to more focused attacks.

The protocol options include TCP, UDP, and ICMP, which cover key communication methods often exploited by attackers. These protocols allow us to detect potential scans across various layers of network communication. The scan type section includes four modes: Portscan Detection identifies targeted scans on a specific host, Port Sweep detects wide-range scans across multiple IP addresses, Decoy Portscan uncovers attempts to mask the scan origin, and Distributed Portscan detects coordinated scans from multiple sources.

The sensitivity level is set to High to provide detailed detection of suspicious activity without missing subtle reconnaissance attempts. The Watch IP field shows the specific focus on the Kali attacker IP address 192.168.6.2, which isolates scans directed at or originating from this address.



Portscan Detection

Settings

Protocol
TCP
UDP
ICMP
IP

Scan Type
Portscan Detection
Port Sweep
Decoy Portscan
Distributed Portscan

Sensitivity Level
High

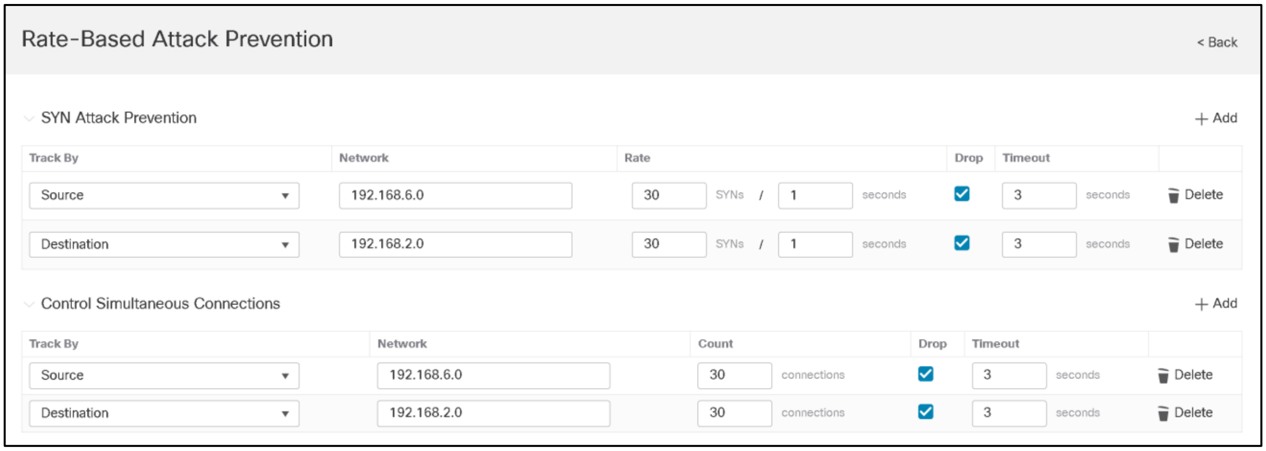
Watch IP
192.168.6.2
(Single IP address, CIDR block, or comma-separated list)

Figure 4.31: PortScan Detection Policy Configuration in Cisco FMC

Furthermore, we configure Rate-Based Attack Prevention under the Network Analysis Policy in Cisco FMC, as depicted in Figure 4.32. This setup aims to control and mitigate high-volume traffic anomalies, such as SYN floods and excessive simultaneous connections, such as TCP SYN, ICMP, and SIP UDP Flood attacks within our network.

In the SYN Attack Prevention section, we define rules to track traffic by destination and source. For the destination network 192.168.2.0 as DMZ and source network 192.168.6.0 as the attacker, the system limits the rate to 30 SYN packets per second. When traffic exceeds this threshold, the system activates the Drop action to prevent further packets and sets a timeout of 3 seconds to reset connections. This ensures immediate response to anomalous traffic rates that can overwhelm network resources.

The Control Simultaneous Connections section applies similar constraints to connection counts. For both the destination network 192.168.2.0 and the source network 192.168.6.0, the system restricts simultaneous active connections to a count of 30. If this count exceeds the defined threshold, the system drops the connection and enforces a timeout of 3 seconds. This configuration prevents resource exhaustion caused by an excessive number of concurrent connections.



Rate-Based Attack Prevention

< Back

SYN Attack Prevention

+ Add

Track By	Network	Rate	Drop	Timeout	
Source	192.168.6.0	30 SYNs / 1 seconds	☑	3 seconds	Delete
Destination	192.168.2.0	30 SYNs / 1 seconds	☑	3 seconds	Delete

Control Simultaneous Connections

+ Add

Track By	Network	Count	Drop	Timeout	
Source	192.168.6.0	30 connections	☑	3 seconds	Delete
Destination	192.168.2.0	30 connections	☑	3 seconds	Delete

Figure 4.32: Rate-Based Attack Prevention Configuration in Cisco FMC

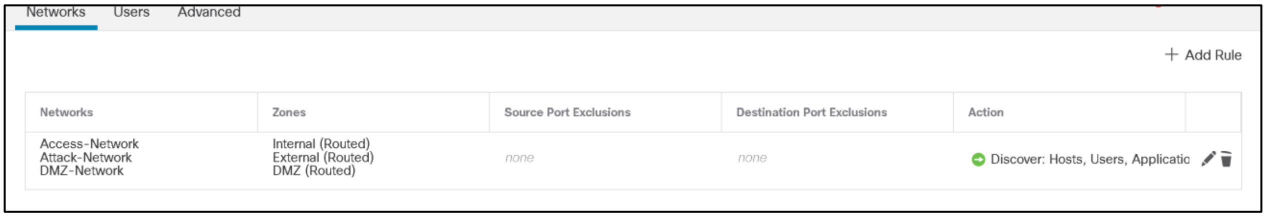
4.8.6 Network Discovery Configuration

A discovery policy specifies the networks and ports that a secure firewall system passively monitors to generate discovery data based on the network traffic passing through the Cisco FMC.

A discovery rule defines the hosts, applications, and non-authoritative users to monitor. Similarly, a discovery rule can exclude networks and zones from discovery.

The Network Discovery rule configuration within the policy section of Cisco Firepower is presented in Figure 4.33. This rule defines how Cisco Firepower identifies and catalogs hosts, users, and applications across different network zones. The table lists three distinct networks, Access Network, Attack Network, and DMZ Network, each mapped to specific zones.

The Access Network is associated with the Internal zone, the Attack Network is tied to the External zone, and the DMZ Network is linked to the DMZ. No source or destination port exclusions are specified to provide comprehensive analysis across all ports for the listed networks. The action for these networks is set to Discover to allow the system to collect detailed information about active devices, user activity, and running services in these zones. We include these networks to ensure complete visibility and monitoring across critical parts of the infrastructure.



Networks	Zones	Source Port Exclusions	Destination Port Exclusions	Action	
Access-Network Attack-Network DMZ-Network	Internal (Routed) External (Routed) DMZ (Routed)	none	none	Discover: Hosts, Users, Applicatio	

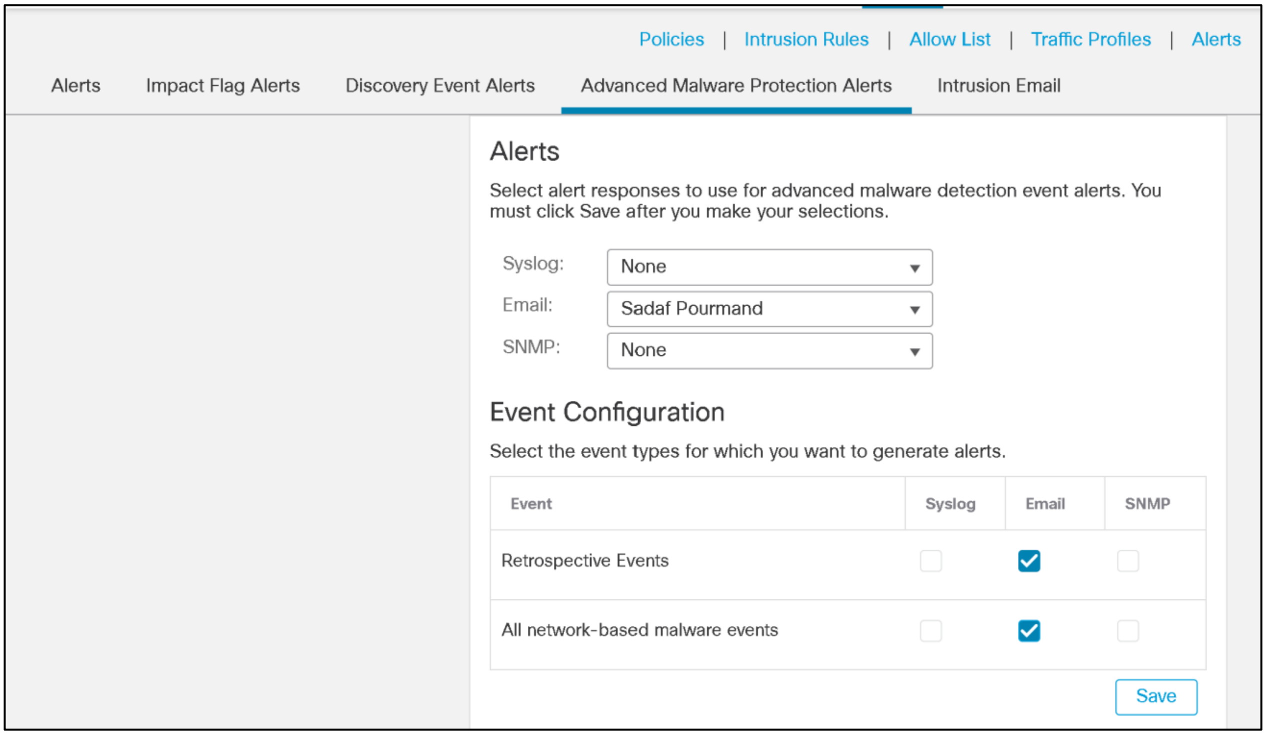
Figure 4.33: Network Discovery Configuration in Cisco FMC

4.8.7 Alert Configuration

External event notification via Simple Network Management Protocol (SNMP), Syslog, or email can help with critical-system monitoring. The Firepower Management Center uses configurable alert responses to interact with external servers. An alert response is a configuration that represents a connection to an email, SNMP, or syslog server. They are called responses because we can use them to send alerts in response to events detected by Firepower.

We implement alert configuration for different activities such as the Advanced Malware Protection and Intrusion in Cisco Firepower, as shown in Figure 4.34. The configuration panel allows the selection of response mechanisms, including Syslog, Email, and SNMP, to notify administrators of the system.

In the displayed configuration, the Email option is directed to Sadaf Pourmand to ensure immediate notification of critical events. Under the Event Configuration section, we enable alerts for Retrospective Events and all network-based malware events to facilitate comprehensive monitoring and reporting of potential threats. This configuration guarantees that any malware-related activities or intrusions within the network trigger prompt notifications to the designated email to enhance the architecture's threat response capabilities.


Policies | Intrusion Rules | Allow List | Traffic Profiles | Alerts
Alerts
Impact Flag Alerts
Discovery Event Alerts
Advanced Malware Protection Alerts
Intrusion Email
Save

Alerts

Select alert responses to use for advanced malware detection event alerts. You must click Save after you make your selections.

Syslog: None

Email: Sadaf Pourmand

SNMP: None

Event Configuration

Select the event types for which you want to generate alerts.

Event	Syslog	Email	SNMP
Retrospective Events	☐	☑	☐
All network-based malware events	☐	☑	☐

Figure 4.34: Alert Configuration in Cisco FMC

At this stage of the thesis methodology, we implemented all essential security configurations in Cisco Firepower to effectively detect and mitigate DDoS attacks in real time. This step includes defining advanced policies, such as rule updates, access control policy, intrusion prevention, malware and file detection, rate-based attack prevention, and network analysis and discovery, specifically tailored to address TCP SYN, ICMP, and SIP UDP flood attacks. Once the policies are deployed, we initiate these flood attack scenarios to evaluate the system's capability to handle high-traffic events while maintaining service availability.

4.9 Cisco Firepower Log Analysis After Policy Deployment

After deploying the security policies, we initiate a second round of DDoS attack simulations, such as TCP SYN, ICMP, and SIP UDP floods, to evaluate the effectiveness and robustness of the configured measures.

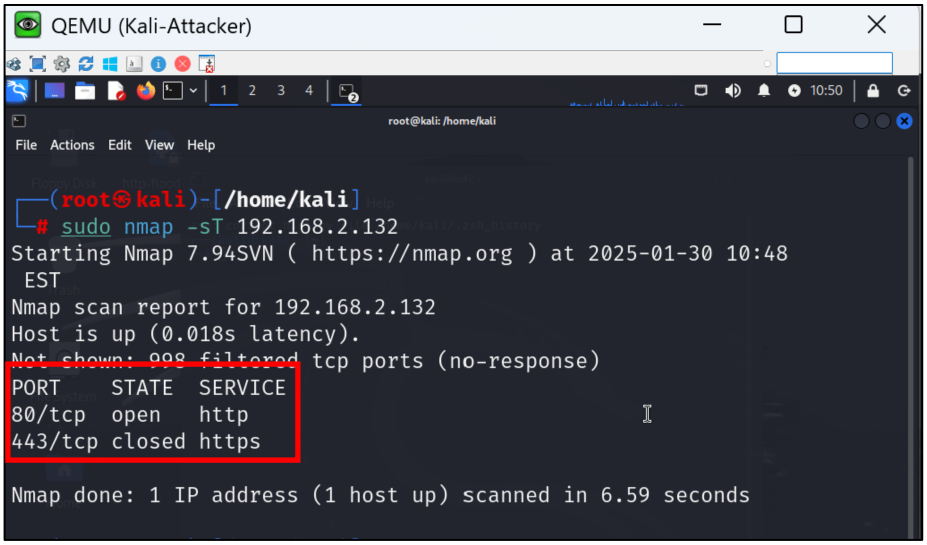
Cisco Firepower captures and logs the incoming attack traffic within its log and analysis module. This log occurs after the activation of intrusion detection and prevention configurations, which ensures the policies actively intercept threats and respond to the simulated attacks.

To examine the effectiveness of these configurations, we compare the collected logs and system behavior under the post-policy state with those from the pre-policy configuration state. This comparison provides the improvements introduced by the implemented policies. It also evaluates their impact on the block and mitigation of flood traffic, reduction of resource consumption, and prevention of service interruptions. By systematic analysis of these results, we validate the ability of Cisco Firepower to respond dynamically and protect the network from various high-risk DDoS attack scenarios, as outlined in the thesis roadmap.

4.9.1 Port Scan Result After Firepower Policy Deployment

We execute the NMAP port scan through Kali attacker after the deployment of security policies on Cisco Firepower. The target IP address 192.168.2.132 was scanned using the -sT scan type, and the report shows only port 80 (HTTP) as open, while port 443 (HTTPS) is marked, as shown in Figure 4.35. The scan also notes that 998 TCP ports are filtered, with no response detected from these ports. The host remains active with a latency of 0.018 seconds, and the scan is completed in 6.59 seconds.

When compared to the previous scan conducted before policy deployment, significant differences are observed. In the earlier scan, two ports were open (22 for SSH and 80 for HTTP), and 998 ports were closed with connection refusal. In contrast, after deploying Firepower policies, port 22 no longer appears as open, and most ports are filtered rather than closed. This shift demonstrates the efficaciousness of the Firepower policies in restricting access and filtering traffic to limit exposure to potential vulnerabilities.

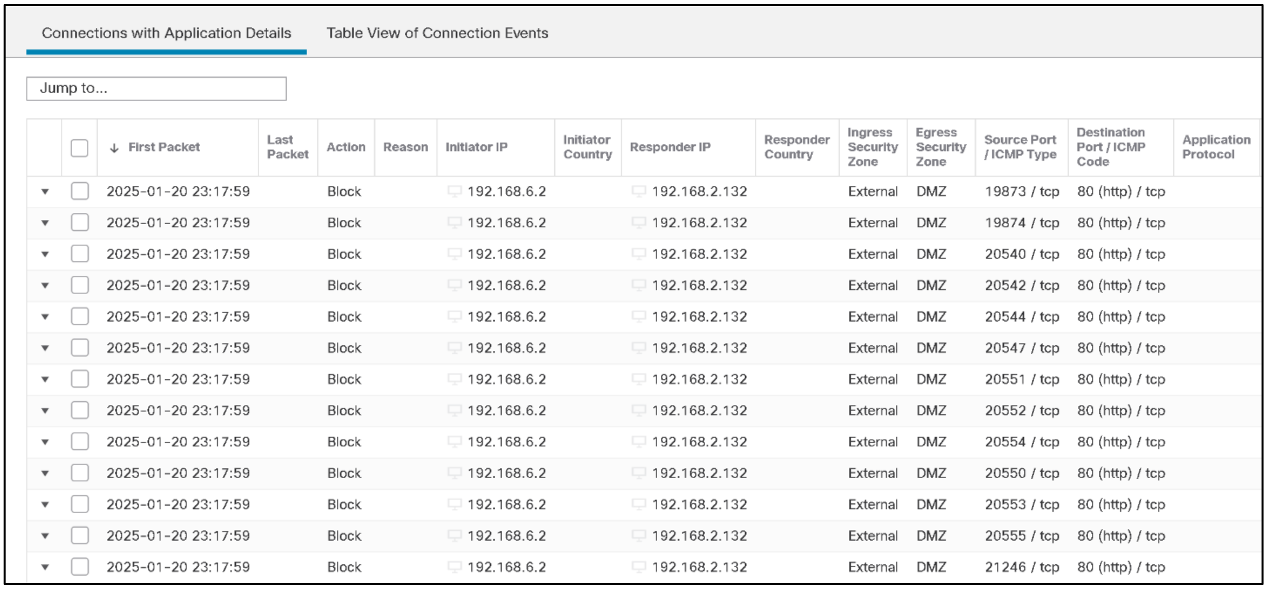


```
┌──(root㉿kali)-[/home/kali]
└─# sudo nmap -sT 192.168.2.132
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-30 10:48
 EST
Nmap scan report for 192.168.2.132
Host is up (0.018s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT    STATE  SERVICE
80/tcp  open   http
443/tcp closed https

Nmap done: 1 IP address (1 host up) scanned in 6.59 seconds
```

Figure 4.35: Port Scan Result After Firepower Policy Deployment

4.9.2 Firepower Log Analysis After TCP SYN Flood Policy Deployment

We inspect the connection events logged in the Cisco FMC after the deployment of security policies to mitigate the TCP SYN Flood attack, as depicted in Figure 4.36. The logs indicate that the Firepower system effectively blocks the attack traffic, as evidenced by the Block action recorded for all entries. The attack originates from the initiator Kali attacker IP address 192.168.6.2 towards the web server at the responder IP address 192.168.2.132 over TCP port 80 (HTTP). The logs also provide details on the source ports used by the attacker, which vary across sessions, with the randomization typical of flood attack patterns. This output indicates the functionality of the security policies, which mitigates the attack and preserves the web server's availability within the secured architecture.



Connections with Application Details | Table View of Connection Events

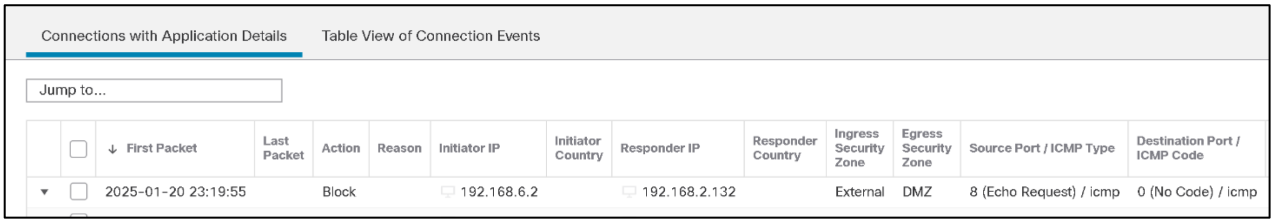
Jump to...

	□	↓ First Packet	Last Packet	Action	Reason	Initiator IP	Initiator Country	Responder IP	Responder Country	Ingress Security Zone	Egress Security Zone	Source Port / ICMP Type	Destination Port / ICMP Code	Application Protocol
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	19873 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	19874 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20540 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20542 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20544 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20547 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20551 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20552 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20554 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20550 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20553 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	20555 / tcp	80 (http) / tcp	
▾	□	2025-01-20 23:17:59		Block		192.168.6.2		192.168.2.132		External	DMZ	21246 / tcp	80 (http) / tcp	

Figure 4.36: Firepower FTD Log Capture During the TCP SYN Attack After Policy Deployment

4.9.3 Firepower Log Analysis After ICMP Flood Policy Deployment

We capture the logged connection events on Cisco Firepower after deploying security policies designed to counter ICMP Flood attacks, as shown in Figure 4.37. The system denies all harmful traffic, as shown by the Block action across the entries. The attack originates from 192.168.6.2 as the Kali attacker and targets 192.168.2.132 as the web server within the DMZ. The details reveal attempts to exploit ICMP Echo Requests, shown by source and destination port values of 8 and 0, along with TCP-based flows targeting port 80 for HTTP services. This configuration demonstrates the effectiveness of the Firepower policies in preventing the ICMP Flood attacks and safeguarding the web server from potential harm.



Connections with Application Details | Table View of Connection Events

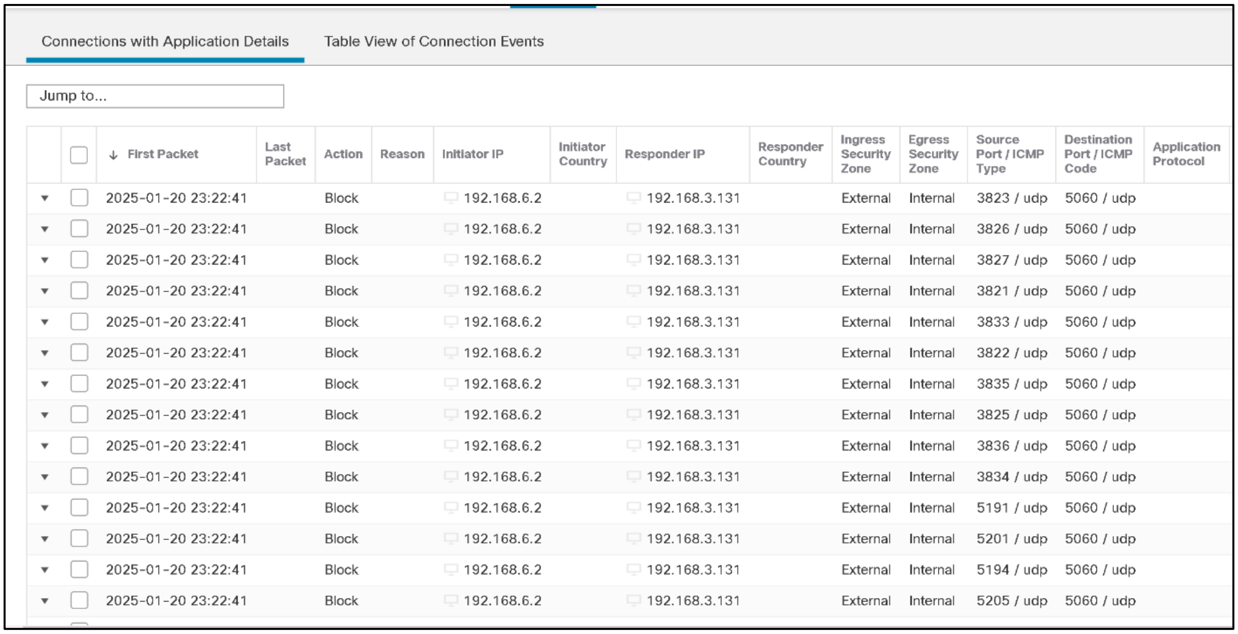
Jump to...

	□	↓ First Packet	Last Packet	Action	Reason	Initiator IP	Initiator Country	Responder IP	Responder Country	Ingress Security Zone	Egress Security Zone	Source Port / ICMP Type	Destination Port / ICMP Code
▾	□	2025-01-20 23:19:55		Block		192.168.6.2		192.168.2.132		External	DMZ	8 (Echo Request) / icmp	0 (No Code) / icmp

Figure 4.37: Firepower FTD Log Capture During the ICMP Attack After Policy Deployment

4.9.4 Firepower Log Analysis After SIP UDP Flood Policy Deployment

We monitor the logged connection events on Cisco Firepower after the implementation of policies to combat the SIP UDP Flood attack in the architecture, as shown in Figure 4.38. Malicious traffic that originates from the Kali attacker IP 192.168.6.2 and targets the Ostinato traffic generator as responder IP 192.168.3.131 is effectively blocked, as shown in the Action column. The attack uses UDP with source ports that vary dynamically to target the SIP service on port 5060. The configured policies block the flood attempts, protect the stability of the SIP service, and defend the internal network from disruption. The event logs prove the Firepower system's capability to detect and stop this type of high-risk attack.



Connections with Application Details
Table View of Connection Events
Jump to...

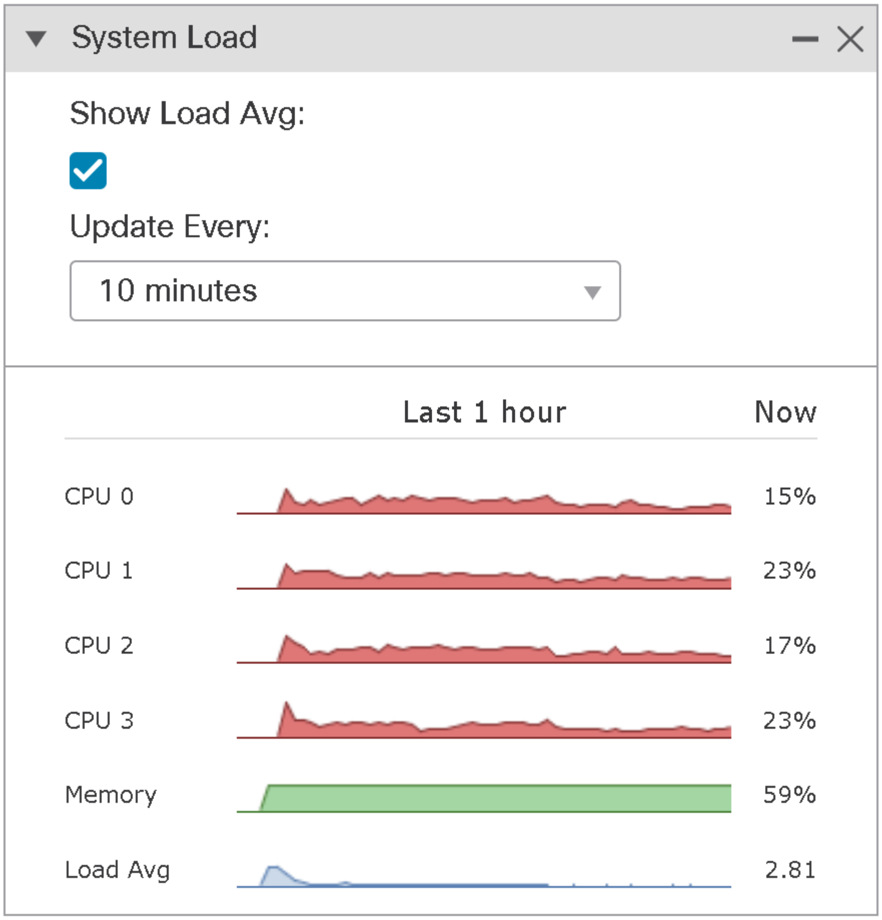
	□	↓ First Packet	Last Packet	Action	Reason	Initiator IP	Initiator Country	Responder IP	Responder Country	Ingress Security Zone	Egress Security Zone	Source Port / ICMP Type	Destination Port / ICMP Code	Application Protocol
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3823 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3826 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3827 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3821 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3833 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3822 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3835 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3825 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3836 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	3834 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	5191 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	5201 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	5194 / udp	5060 / udp	
▼	□	2025-01-20 23:22:41		Block		192.168.6.2		192.168.3.131		External	Internal	5205 / udp	5060 / udp	

Figure 4.38: Firepower FTD Log Capture During the SIP UDP Attack After Policy Deployment

4.9.5 Firepower Resource Usage Post-Policy Flood Attacks

We capture Firepower's resource consumption during DDoS flood attacks after policy deployment, as shown in Figure 4.39. CPU utilization across all cores (CPU 0 at 15%, CPU 1 at 23%, CPU 2 at 17%, and CPU 3 at 23%) shows a drastic reduction compared to the pre-policy state, where CPU usage reached levels of 64%, 83%, 88%, and 82% respectively. Memory usage decreases slightly, moving from 64% in the pre-policy state to 59% after the policies are deployed. The load average drops significantly from 11 in the pre-policy scenario to 2.81 to signify a major improvement in system performance after policy conditions.

This comparison emphasizes the efficiency of the security measures implemented in Firepower. Policies such as access control policies and intrusion prevention effectively block malicious traffic and prevent the system from reaching resource consumption thresholds. The drastic drop in CPU utilization and load average indicates the successful mitigation of resource exhaustion when the attack is blocked at an early stage.


System Load

Show Load Avg: ☑

Update Every: 10 minutes

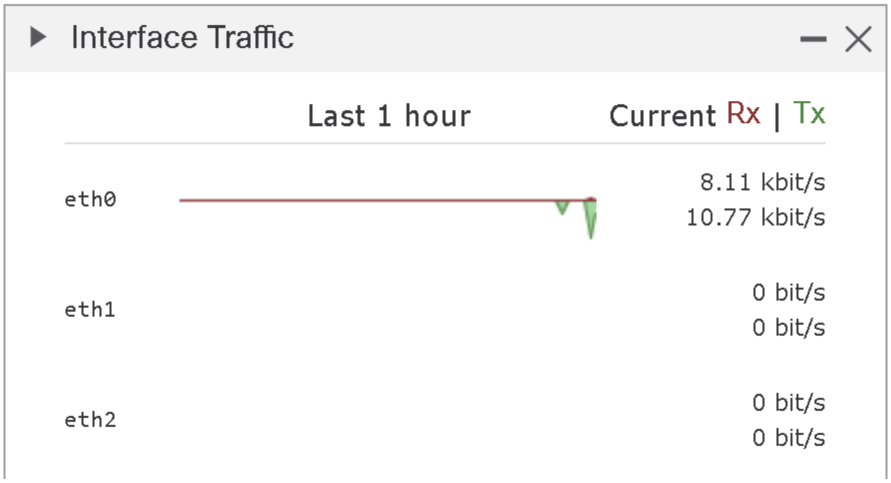
	Last 1 hour	Now
CPU 0		15%
CPU 1		23%
CPU 2		17%
CPU 3		23%
Memory		59%
Load Avg		2.81

Figure 4.39: Firepower Resource Utilization During the Flood Attack After Policy Deployment

Furthermore, we monitor the bandwidth consumption of Cisco Firepower during DDoS flood attacks after the deployment of security policies. The eth0 interface shows significantly reduced traffic rates in Figure 4.40, compared to the pre-policy condition, with incoming traffic at 8.11 kbit/s and outgoing traffic at 10.77 kbit/s.

The current traffic rates reflect the successful implementation of mitigation strategies. In comparison with bandwidth consumption before policy deployment, the earlier state showed unrestricted traffic on eth0, with outgoing traffic peaking at 352.86 kbit/s due to unfiltered DDoS flood packets. After deploying security policies, malicious traffic is now identified and blocked, drastically reducing bandwidth utilization.

In our scenario, the resource analysis exhibits that Cisco Firepower countered various DDoS flood attacks of TCP SYN, ICMP, and SIP UDP floods by blocking malicious traffic at the perimeter level. The deployed security policies prevented attack traffic from accessing components like the web server and the Ostinato traffic generator. The significant reduction in CPU utilization, memory usage, and system load reflects the optimized performance and resilience of the system. Firepower successfully mitigated the resource strain caused by high-volume attack traffic, which kept availability and operational stability intact while defending the secured assets. This result proves the strength of the deployed measures against the complex nature of DDoS flood scenarios.



Interface Traffic

	Last 1 hour	Current Rx | Tx
eth0		8.11 kbit/s 10.77 kbit/s
eth1		0 bit/s 0 bit/s
eth2		0 bit/s 0 bit/s

Figure 4.40: Firepower Interface Traffic During the Flood Attack After Policy Deployment

4.10 Summary

This chapter outlined the evaluation and validation of the proposed architecture through a comprehensive analysis and experiment. Initially, we executed NMAP port scans from the Kali attacker to the web server to identify system vulnerabilities. Subsequently, we launched DDoS flood attacks, such as TCP SYN, ICMP, and SIP UDP floods, towards the DMZ and internal zones to exhaust system resources and disrupt services. In the first scenario, we configured Cisco Firepower in transparent mode without active security policies to allow unrestricted attack traffic to traverse the device and reach the web server and Ostinato SIP agent. This phase focused on the analysis of attack traffic through Firepower's monitoring features, such as the connection summary.

In the second scenario, we applied advanced security measures within Cisco Firepower, which included rule and geolocation updates, access control policies, malware and file prevention mechanisms, port scan and rate-based attack prevention, and alert configurations to enable real-time attack detection and mitigation. After policy deployment, we re-initiated the DDoS flood attacks, noting that Cisco Firepower effectively identified and blocked all malicious traffic. Additionally, resource and bandwidth utilization metrics showed a significant reduction in system strain after applying these policies. The ability of Cisco Firepower to block attack traffic, protect resources, and maintain operational integrity under high-traffic conditions serves as concrete proof of its effectiveness in defense against DDoS attacks to fulfill the objectives of our thesis methodology and roadmap. We create table 5.1 to compare resource and bandwidth use before and after Firepower policy deployment.

Table 5.1: Comparison of Firepower Resource and Bandwidth Consumption

Metrics	**Before Policy Deployment**	**After Policy Deployment**
CPU Usage (Average)	64% - 88%	15% - 23%
Memory Usage	64%	59%
Bandwidth Usage (eth0 RX)	27.83 kbit/s	8.11 kbit/s
Bandwidth Usage (eth0 TX)	352.86 kbit/s	10.77 kbit/s

CHAPTER 5 CONCLUSION

This dissertation presented our integrated security architecture for the protection of network infrastructure with a focus on VoIP service through the NGFW and SIEM systems. Furthermore, we evaluated the proposed architecture's performance against sophisticated DDoS attacks. This chapter concludes the dissertation. The first section discusses a summary of the work. The second details the limitations of our work. Finally, the third section provides future works and directions.

5.1 Summary of Works

Cybersecurity has gained critical importance in defending information, data, and network resources from cyber threats that attempt to steal sensitive data and obstruct and deny access to network services. The rapid growth of digital technology and the broad adoption of the Internet increase these risks. Inadequate security controls and the lack of legal frameworks further challenge security systems, leaving organizations more vulnerable to data breaches and network disruptions.

DDoS attacks endanger the stability of networks and websites which requires strong defensive measures and proper prevention strategies to shield data and system resources from disruption. DDoS attacks fall into three primary categories based on their purpose, with distinctions in execution and consequences. These include application-layer DDoS attacks that exploit weaknesses in websites, protocol-level DDoS attacks like TCP SYN Flood, and volume-based DDoS attacks that use spoofed packets to launch ICMP floods, UDP floods, and related threats.

By conducting a comprehensive literature review on the DDoS attack detection and mitigation measures, specifically in VoIP networks as vulnerable systems, we identify several weaknesses and shortcomings in the current systems. The weaknesses include limited efficiency in distinguishing legitimate traffic from attack-generated packets, reliance on static threshold-based detection that fails to adapt to evolving attack patterns, lack of deep inspection mechanisms tailored for SIP and RTP-based attacks, absence of integrated defense strategies that combine anomaly detection with real-time mitigation, and insufficient scalability of existing solutions in high-traffic environments. Therefore, several questions come to mind. What strategies can Next-Generation Firewalls adopt to secure VoIP networks against DDoS attacks? Which method to identify the most problematic types and scenarios of DDoS attacks? Which Next-Generation Firewall strategy should be proposed to detect and mitigate real-time DDoS attacks?

Addressing the previously identified limitations, this dissertation presents a structured approach for the mitigation of DDoS attacks in real-time by integration of an advanced Next-Generation Firewall system with centralized threat intelligence from a SIEM platform. This framework improves protection of network infrastructures by identifying the most critical DDoS attack vectors and high-risk scenarios. The design focuses on a firewall-driven security architecture capable of interception of DDoS attacks as they occur to contribute adaptive defense mechanisms that respond to emerging attack patterns with precision.

To achieve our objectives, we devised a secure architecture that encompasses three distinct network layers: Access, DMZ, and Core zones. Additionally, an attacker network was introduced to emulate real-world DDoS attack scenarios where adversaries attempt to exploit potential vulnerabilities. Within this scenario, various network elements and devices were strategically deployed, accompanied by the integration of two key security solutions: Cisco Firepower for advanced threat prevention and Wazuh SIEM for centralized monitoring and threat intelligence. In the following paragraphs, we provide a detailed discussion of each component, its specific role within the architecture, and how it contributes to fulfilling the objectives of real-time attack detection and mitigation.

The Access Network Layer handles user traffic and enforces initial security policies to allow only legitimate requests to enter the network. Two network nodes of the Ostinato traffic generator, which acts as an SIP endpoint and generates VoIP traffic and VPC, are in the access zone.

The DMZ Network Layer serves as an intermediary zone between the external and internal networks. It contains critical services such as the web server and SIEM to monitor and analyze network activity.

The Core Network Layer functions as the backbone, provides internal communication, and processes security policies to protect sensitive assets. Cisco Firepower plays a pivotal role in the core zone for threat detection and mitigation by enforcing real-time traffic filtering and intrusion prevention mechanisms.

ISP and Core Routers are configured with static routes to direct traffic efficiently, while DMZ and Access Switches manage inter-device communication. The proposed security architecture employs VLAN segmentation and static routing to isolate traffic flows between zones, reducing the attack surface and preventing lateral movement of threats.

The attacker network, represented by the Kali Linux system, generates large-scale DDoS flood traffic to test the network's resilience against various attack types. The network simulation environment operates through EVE-NG, a virtualized testbed that provides a controlled setting for security framework evaluation under real-world attack scenarios.

A fundamental aspect of this architecture is its multi-layered defense approach, which combines intelligent traffic filtering, adaptive anomaly detection, and automated threat response to counteract DDoS attacks in real-time. The integration of Cisco Firepower and Wazuh SIEM enables correlated security event analysis, where Firepower enforces inline traffic inspection and rule-based attack mitigation, while Wazuh SIEM provides comprehensive log analysis, endpoint telemetry, and forensic insights. This synergy enhances attack detection accuracy, reduces false positives, and ensures proactive security orchestration.

Up to this point, we have structured the network with well-defined segments and strategically deployed device nodes within their respective zones, which establishes a cohesive and interoperable framework within the architecture. As a next step, we shifted our focus on the Kali attacker to discover different vulnerabilities on the web server where it would be the point of exploitation. To achieve this purpose, the attacker scans the web server using NMAP, which reveals open ports and active services, specifically SSH (port 22) and HTTP (port 80). With this information, the attacker attempts to access the web server through a direct connection to confirm its availability.

Exploiting the identified vulnerabilities, the web server faced TCP SYN and ICMP flood attacks, while the Ostinato traffic generator was subjected to a SIP UDP flood, all aimed at exhausting system resources and causing service failure. Wireshark packet capture provided a detailed analysis of anomalous network behavior and traffic anomalies. The captured data confirms that the flood traffic triggered an abnormal surge of unauthorized packets, which resulted in excessive half-open connections, network congestion, and increased response delays.

We had two different scenarios when the attacker launched DDoS flood attacks. The first scenario placed Cisco Firepower in transparent mode without active security policies, which allowed all attack traffic to pass through without restriction. This configuration provided a baseline to observe the impact of attack traffic on system resources, network behavior, and security logs. The logs from Cisco Firepower indicated that all attack flows successfully reached their targets, which confirmed the network's vulnerability due to the absence of security enforcement. The resource utilization report showed that CPU usage peaked at 88%, memory usage remained at 64%, and bandwidth consumption on interface eth0 recorded an outgoing traffic rate of 352.86 kbit/s. These values demonstrated the strain placed on system resources when exposed to high-volume attack traffic.

In addition to Firepower logs, we deployed Wazuh SIEM to detect early signs of attack activity and analyze security anomalies within the Kali attacker's environment. The Threat Hunting module in Wazuh reported an increase in suspicious network activity, host-based anomalies, and excessive process execution on the Kali attacker. The Malware Detection module flagged Trojaned versions of system binaries, which justified modifications in system files used for attack automation. The Configuration Assessment and Vulnerability Detection reports identified security misconfigurations and unpatched CVEs in the attacker's system, which reinforced the hypothesis that a compromised attacker system could execute a high-volume flood attack. Wazuh's ability to correlate endpoint activity with network-based intrusion logs provided an in-depth view of the attack strategy.

In the second scenario, we deployed advanced security policies in Cisco Firepower to detect and mitigate the DDoS flood attacks in real-time. These policies included rule updates, geolocation-based filtering, access control policies, intrusion prevention rules, rate-based attack prevention, port scan detection, and malware and file inspection. The rule updates ensured Firepower had the latest threat intelligence to recognize evolving attack patterns, while geolocation filtering restricted traffic from untrusted regions. The access control policies allowed only legitimate traffic to pass and blocked unauthorized and malicious connections. The IPS actively detected and dropped packets that matched known attack signatures, particularly SYN floods, ICMP flood attempts, and UDP-based SIP floods. Additionally, rate-based attack prevention policies restricted excessive connection attempts by capping TCP SYN packets at 30 per second and limiting simultaneous connections to prevent resource exhaustion. The malware and file policies blocked suspicious executable files that attackers could use to deliver payloads and automate attack mechanisms.

After the deployment of these security policies, we launched the same DDoS flood attacks again to evaluate Firepower's effectiveness. The logs revealed a dramatic shift, which showed that all malicious packets were now blocked at the perimeter before reaching target resources. TCP SYN, ICMP, and SIP UDP flood packets were all marked as denied in Firepower's connection events, which prevented further impact on system operations.

The effect of these security policies was also reflected in Firepower's resource monitor, which reported a significant drop in CPU utilization from 88% to an average of 23% across all cores. Memory usage decreased slightly to 59%, while bandwidth consumption was reduced, with outgoing traffic dropping from 352.86 kbit/s to just 10.77 kbit/s. The intrusion prevention logs indicated that Firepower successfully identified and mitigated attack flows at an early stage, which reduced the time needed to detect threats and block malicious traffic.

A comparison of both scenarios showed that the first scenario demonstrated how an unprotected network was fully vulnerable to DDoS attacks, which led to high resource consumption and complete exposure of services. In contrast, the second scenario convinced that Firepower's multi-layered security policies, combined with Wazuh SIEM's endpoint analytics, effectively mitigated flood attacks, preserved system stability, and maintained service availability. These results validated the efficiency of our proposed security architecture to handle high-traffic DDoS flood attacks and protect network infrastructure.

5.2 Limitations

Despite the effectiveness of our proposed security framework for DDoS detection and mitigation, several limitations exist at different stages of the architecture. The first constraint lies in the scope of attack vectors tested. While we analyzed three major DDoS attack types of TCP SYN flood, ICMP flood, and SIP UDP flood attacks, other sophisticated multi-vector DDoS attacks, such as DNS amplification and HTTP-based attacks, were not part of our evaluation.

Another key limitation relates to policy refinement and automated threat response. The study primarily relied on manually configured policies within Cisco Firepower to mitigate flood attacks. Although these policies successfully block malicious traffic and optimize resource consumption, the lack of a dynamic policy adaptation mechanism may hinder real-time adjustments to new attack strategies.

Furthermore, the granularity of Wazuh SIEM analysis presents another limitation. While Wazuh effectively detected attack indicators, event correlation across multiple security layers was not fully explored. A more comprehensive cross-layer correlation mechanism between Cisco Firepower and Wazuh SIEM could enhance contextual threat analysis by aggregating logs from diverse sources, such as system events, authentication records, and application-layer interactions.

The network topology constraints in our testbed also introduce limitations. The study took place within a controlled lab environment, with predefined network zones and static routing configurations. In real-world enterprise networks, dynamic traffic flows, encrypted communication channels, and distributed attack surfaces present additional challenges that were not fully addressed in this study.

Another significant limitation is the resource constraints of the test environment. The Firepower device in our architecture operated within a specific hardware capacity, meaning the findings may not directly scale to high-performance enterprise firewalls with multi-gigabit throughput requirements.

5.2 Future Work

For future improvements and extensions of our proposed security framework, several enhancements can strengthen its effectiveness against evolving DDoS attacks. The first major enhancement involves the expansion of attack scenarios beyond the current TCP SYN flood, ICMP flood, and SIP UDP flood attacks. Future work should incorporate multi-vector and application-layer attacks such as DNS amplification, HTTP GET/POST floods, and encrypted traffic-based DDoS attacks.

Another key improvement is the integration of machine learning and artificial intelligence-driven threat detection. Currently, the system relies on static rule-based policies for the identification and mitigation of attacks. The introduction of anomaly detection models and behavior-based analysis significantly enhances early attack identification, reduces response time, and improves adaptability. AI-based models dynamically adjust security policies based on real-time traffic patterns and make the system more autonomous and resilient.

Future research should also focus on the automation of policy adjustments within Cisco Firepower. The current approach relies on manually configured policies, which may not efficiently respond to zero-day or rapidly evolving attack techniques. Implementation of an adaptive policy mechanism that automatically updates firewall rules and intrusion prevention policies based on threat intelligence feeds strengthens the architecture's defense capabilities.

In terms of scalability, future work should evaluate the system in large-scale enterprise environments. The current testbed operates within a controlled lab setting with limited traffic loads. The expansion of the testing environment to include high-bandwidth, real-world enterprise networks provides a more accurate performance assessment under production conditions.

Finally, the framework should extend its defense mechanisms beyond DDoS attacks. Future work can explore its applicability in the mitigation of other cyber threats, such as insider attacks, data exfiltration attempts, and ransomware activities. By broadening the security scope, the proposed architecture serves as a comprehensive cybersecurity solution for enterprise networks.

REFERENCES

[1] Radware, “2024 Global Threat Analysis Report Thank You! | Radware,” Radware.com, 2024. https://www.radware.com/h1-2024-global-threat-analysis-report-lpc-39853846/ (accessed Oct. 23, 2024).

[2] J. Liang and Y. Kim, “Evolution of Firewalls: Toward Securer Network Using Next Generation Firewall,” IEEE Xplore, Jan. 01, 2022. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9720435 (accessed Apr. 08, 2022).

[3] D. Sim, H. Guo, and L. Zhou, “A SIEM and Multiple Analysis Software Integrated Malware Detection Approach,” Dec. 2023, doi: https://doi.org/10.1109/soli60636.2023.10425463.

[4] G. G. Granadillo, S. G. Zarzosa, and R. Diaz, “Security Information and Event Management (SIEM): Analysis, Trends, and Usage in Critical Infrastructures,” Sensors, vol. 21, no. 14, p. 4759, Jul. 2021, doi: https://doi.org/10.3390/s21144759.

[5] Q. Li *et al.*, “A comprehensive survey on DDoS defense systems: New trends and challenges,” *Computer Networks*, vol. 233, p. 109895, Sep. 2023, doi: https://doi.org/10.1016/j.comnet.2023.109895.

[6] A. B. de Neira, B. Kantarci, and M. Nogueira, “Distributed denial of service attack prediction: Challenges, open issues and opportunities,” Computer Networks, vol. 222, p. 109553, Feb. 2023, doi: https://doi.org/10.1016/j.comnet.2022.109553.

[7] P. S A, S. P. S, and H. B, “DDoS and Botnet Attacks: A Survey of Detection and Prevention Techniques.” 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), pp. 1-7, 2024, doi: 10.1109/adics58448.2024.10533615.

[8] Hafiz, Usman Inayat, Muhammad Fahad Zia, F. Ali, T. Jabeen, and Syed Moshin Ali, “Voice Over Internet Protocol: Vulnerabilities and Assessments,” 2021 International Conference on Innovative Computing (ICIC), Nov. 2021, doi: https://doi.org/10.1109/icic53490.2021.9692955.

[9] H. Ahmadi, G. Arji, L. Shahmoradi, R. Safdari, M. Nilashi, and M. Alizadeh, "The application of internet of things in healthcare: a systematic literature review and classification," *Universal Access in the Information Society,* vol. 18, no. 4, pp. 837-869, 2019/11/01 2019.

[10] H. Razavi, Mohammad Reza Jamali, Morvaridsadat Emsaki, A. Ahmadi, and Mostafa Hajiaghei-Keshteli, “Quantifying the Financial Impact of Cyber Security Attacks on Banks: A Big Data Analytics Approach,” Sep. 2023, doi: https://doi.org/10.1109/ccece58730.2023.10288963.

[11] C. Blekos, “Intrusion Detection System in Financial Institu tions ,” SCHOOL OF SCIENCE & TECHNOLOGY-THESSALONIKI – GREECE, 2022.

[12] P. S A, S. P. S, and H. B, “DDoS and Botnet Attacks: A Survey of Detection and Prevention Techniques.” 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), pp. 1-7, 2024, doi: 10.1109/adics58448.2024.10533615.

[13] J. Liang and Y. Kim, “Evolution of Firewalls: Toward Securer Network Using Next Generation Firewall,” IEEE Xplore, Jan. 01, 2022. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9720435.

[14] Wool, Avishai.“Packet filtering and stateful firewalls” Handbook of Information Security 3 (2006): 526-536.

[15] “Application Gateway” Network Encyclopedia, May 3, 2020. https://networkencyclopedia.com/application-gateway/. accessed on Nov. 04, 2024.

[16] Schultz, E. Eugene.“83-10-41 Types of Firewalls” Internet: http://www.ittoday.info/AIMS/DSM/83-10-41. accessed on Nov. 04, 2024.

[17] Gupta, Namit, Abakash Saikia, and D. Sanghi.“Web application firewall” Indian Institute of Technology, Kanpur 61 (2007): 62.

[18] “WAF vs. NGFW: Which Technology Do You Need,” F5 Networks. https://www.f5.com/c/landing/waf-vs-ngfw-which-technology-do-you-need.

[19] Mordor Intelligence, “Web Application Firewall (WAF) Market - Share, Size & Trends,” Mordorintelligence.com, 2024. https://www.mordorintelligence.com/industry-reports/web-application-firewall-market (accessed Nov. 04, 2024).

[20] “Integrated Threat Prevention,” PALO ALTO NETWORKS: Integrated Threat Prevention Datasheet.

[21] Agham, Vinit.“Unified threat management” International Research Journal of Engineering and Technology 3, no. 4 (2016): 32-36.

[22] Todd McGuiness.“Defense in Dept”. sans.org. Archived from the original on 22 Dec 2017. Retrieved 22 December 2017.

[23] A. Makhdoomi, N. Jan, Palak, and N. Goel, “Conventional and next-generation firewalls in network security and its applications,” *2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, Nov. 2022, doi: https://doi.org/10.1109/icccis56430.2022.10037674.

[24] Doriguzzi-Corin, R. (2020). Methods and Techniques for Dynamic Deployability of Software Defined Security Services. arXiv preprint arXiv:2004.02876.

[25] Perwaiz, S. S. (2021). Critical Infrastructure Protection: Modeling Utility Network Security.

[26] D. Freet and R. Agrawal, "Network security and next-generation firewalls," in Proc. Int. Conf. Technol. Manag. (ICTM), 2016, p. 23.

[27] W. Khan, "Optimizing vulnerability management for large-scale enterprises: Strategies for identifying and mitigating risks through advanced vulnerability assessments," *Int. J. Adv.*

Res. Eng. Technol. (IJARET), vol. 13, no. 10, Oct. 2022. [Online]. Available: https://iaeme.com/Home/issue/IJARET?Volume=13&Issue=10.

[28] H. Haughey, G. Epiphaniou, H. Al-Khateeb, and A. Dehghantanha, "Adaptive traffic fingerprinting for darknet threat intelligence," in Cyber Threat Intelligence, pp. 193-217, 2018.

[29] K. Kallepalli and U. B. Chaudhry, "Intelligent security: Applying artificial intelligence to detect advanced cyber attacks," in Challenges in the IoT and Smart Environments: A Practitioners' Guide to Security, Ethics and Criminal Threats, Cham: Springer Int. Publishing, 2021, pp. 287-320.

[30] C. Rae, Apr. 18, 2024. https://www.isms.online/glossary/likelihood/ (accessed Nov. 07, 2024).

[31] M. Fahrurozi, S. A. Tarigan, M. Alam Tanjung, and K. Mutijarsa, "The Use of ISO/IEC 27005: 2018 for Strengthening Information Security Management (A Case Study at Data and Information Center of Ministry of Defence)," IEEE Xplore, Oct. 01, 2020. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9271748 (accessed Apr. 25, 2022).

[32] "Enterprise Intelligence Brief: Breach Detection Systems – NSS Labs," *NSS Labs*, Oct. 30, 2019. https://nsslabs.com/reports/security-controls-in-the-us-enterprise-breach-security-breach-detection-systems/ (accessed Nov. 07, 2024).

[33] K. Neupane, R. Haddad, and L. Chen, "Next-generation firewall for network security: A survey," in Proc. SoutheastCon 2018, Apr. 2018, pp. 1-6.

[34] G. Uçtu, M. Alkan, İ. A. Doğru, and M. Dörterler, "A suggested testbed to evaluate multicast network and threat prevention performance of Next Generation Firewalls," *Future Gener. Comput. Syst.*, vol. 124, pp. 56-67, 2021.

[35] Nyati, S. (2018). Transforming Telematics in Fleet Management: Innovations in Asset Tracking, Efficiency, and Communication. International Journal of Science and Research (IJSR), 7(10), 1804-1810.

[36] A. Alshamrani, S. Myneni, A. Chowdhary, and D. Huang, "A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities," *IEEE Commun. Surv. Tuts.*, vol. 21, no. 2, pp. 1851-1877, 2019.

[37] D. Arp, E. Quiring, and F. Pendlebury, "Dos and Don'ts of Machine Learning in Computer Security."

[38] I. Abusamrah, A. Madhoun, and S. Iseed, "Next-generation firewall, deep learning endpoint protection, and intelligent SIEM integration," 2021.

[39] Korhonen, J. (2019). Outbound SSL/TLS decryption: Security impact of SSL/TLS interception.

[40] Phillips, M. (2014). TLS Filter: An Application-Level Firewall for Transport Layer Security. Technical Report. URL: http://www. doc. ic. ac. uk/teaching/distinguished projects/2014/m.phillips. pdf.

[41] Laine, P. (2013). Functional Testing of BYOD Features with Next-Generation Firewall: PAN-OS Version 5.0. x.

[42] G. Gonzalez-Granadillo, S. Gonzalez-Zarzosa, and R. Diaz, “Security information and event management (SIEM): analysis, trends, and usage in critical infrastructures,” Sensors, vol. 21, no. 14, article no. 4759, 2021. https://doi.org/10.3390/s21144759.

[43] S. Saleem, M. Sheeraz, M. Hanif, and U. Farooq, “Web server attack detection using machine learning,” in Proceedings of 2020 International Conference on Cyber Warfare and Security (ICCWS), Islamabad, Pakistan, 2020, pp. 1-7.

[44] H. Mokalled, R. Catelli, V. Casola, D. Debertol, E. Meda, and R. Zunino, “The applicability of a SIEM solution: requirements and evaluation,” in Proceedings of 2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Napoli, Italy, 2019, pp.132-137.

[45] G. Granadillo, M. El-Barbori, and H. Debar, "New types of Alert Correlation for Security Information and Event Management Systems," in 8th International Conference on New Technologies, Mobility and Security, NTMS, Larnaca, Cyprus.

[46] M. D. Elradi, K. A. Abdelmajeed, and M. O. Abdulhaleem, "Cyber security professionals’ challenges: A proposed integrated platform solution," Electr. Sci. Eng., vol. 3, no. 2, 2021. doi: 10.30564/ese.v3i2.3376.

[47] R. Gibadullin and V. Nikonorov, "Development of the system for automated incident management based on open-source software," in Proc. 2021 Int. Russian Autom. Conf. (RusAutoCon), Kazan, Russia, 2021, pp. 521-525. doi: 10.1109/RusAutoCon52004.2021.9537385.

[48] A. Tariq, J. Manzoor, M. A. Aziz, Z. U. A. Tariq, and A. Masood, "Open source SIEM solutions for an enterprise," Emerald Publishing Ltd., vol. 31, no. 1, pp. 88–107, 2022. doi: 10.1108/ICS-09-2021-0146.

[49] S. Stanković, S. Gajin, and R. Petrović, "A review of Wazuh tool capabilities for detecting attacks based on log analysis," in Proc. IX Int. Conf. IcETRAN, Novi Pazar, Serbia, 2022, pp. 6-9.Available: https://www.etran.rs/2022/zbornik/ICETRAN22_radovi/068RTI2.6.pdf.

[50] R. M. Muhammad, I. D. Irawati, and M. Iqbal, "Integrated security system implementation for network intrusion," J. Hunan Univ. (Natural Sci.), vol. 48, no. 6, 2021. Available: http://jonuns.com/index.php/journal/article/view/619.

[51] T. Suryantoro, B. P. D. P., and W. Andriyani, "The analysis of attacks against port 80 webserver with SIEM Wazuh using detection and OSCAR methods," in Proc. 2022 5th Int. Seminar Res. Inf. Technol. Intell. Syst. (ISRITI), Yogyakarta, Indonesia, 2022. doi: 10.1109/ISRITI56927.2022.10052950.

[52] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Trans. Dependable Secure Comput.*, vol. 1, no. 1, pp. 11–33, 2004.

[53] N. Gruschka and N. Luttenberger, "Protecting web services from DoS attacks by SOAP message validation," in IFIP, S. Fischer-Hübner, K. Rannenberg, L. Yngström, and S. Lindskog, Eds. USA: Springer, 2006, pp. 171-182.

[54] D. E, Computer networks and internets. Harlow: Pearson Education, 2015.

[55] A. Gaurav, B. B. Gupta, W. Alhalabi, A. Visvizi, and Y. Asiri, "A comprehensive survey on DDoS attacks on various intelligent systems and it's defense techniques," International Journal of Intelligent Systems, vol. 37, no. 12, pp. 11407–11431, Sep. 2022, doi: https://doi.org/10.1002/int.23048.

[56] A. Singh and B. B. Gupta, "Distributed denial-of-service (ddos) attacks and defense mechanisms in various web-enabled computing platforms:Issues, challenges, and future research directions, International Journal on Semantic Web and Information Systems (IJSWIS), vol. 18, no. 1, pp.1–43, 2022.

[57] Godlovitch, I.; Kroon, P. Copper Switch-off: European Experience and Practical Considerations (No. WIK-Consult White Paper); WIK-Consult GmbH: Bad Honnef, Germany, 2020.

[58] J. Kafke and T. Viana, "Call Me Maybe: Using Dynamic Protocol Switching to Mitigate Denial-of-Service Attacks on VoIP Systems," Network, vol. 2, no. 4, pp. 545–567, Oct. 2022, doi: https://doi.org/10.3390/network2040032.

[59] W. Nazih, W. S. Elkilani, H. Dhahri, and T. Abdelkader, "Survey of Countering DoS/DDoS Attacks on SIP Based VoIP Networks," *Electronics*, vol. 9, no. 11, p. 1827, Nov. 2020, doi: https://doi.org/10.3390/electronics9111827.

[60] D. Golait and N. Hubballi, "Detecting Anomalous Behavior in VoIP Systems: A Discrete Event System Modeling," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 3, pp. 730–745, Mar. 2017, doi: https://doi.org/10.1109/tifs.2016.2632071.

[61] M. Zubair Rafique, Muhammad Ali Akbar, and M. Farooq, "Evaluating DoS Attacks against Sip-Based VoIP Systems," *Lirias (KU Leuven)*, Nov. 2009, doi: https://doi.org/10.1109/glocom.2009.5426247.

[62] S. Kai, G. Zhe, and Meng Chunzhi, "VoIP transmission mechanism based on TCP," *The Journal of China Universities of Posts and Telecommunications*, vol. 23, no. 6, pp. 90–96, Dec. 2016, doi: https://doi.org/10.1016/s1005-8885(16)60075-4.

[63] G. Ormazabal, S. Nagpal, Eilon Yardeni, and Henning Schulzrinne, "Secure SIP: A Scalable Prevention Mechanism for DoS Attacks on SIP Based VoIP Systems," pp. 107–132, Oct. 2008, doi: https://doi.org/10.1007/978-3-540-89054-6_6.

[64] D. Sisalem, J. Kuthan, and S. Ehlert, "Denial of service attacks targeting a SIP VoIP infrastructure: attack scenarios and prevention mechanisms," *IEEE Network*, vol. 20, no. 5, pp. 26–31, Sep. 2006, doi: https://doi.org/10.1109/mnet.2006.1705880.

[65] G. Ormazabal, S. Nagpal, Eilon Yardeni, and Henning Schulzrinne, "Secure SIP: A Scalable Prevention Mechanism for DoS Attacks on SIP Based VoIP Systems," pp. 107–132, Oct. 2008, doi: https://doi.org/10.1007/978-3-540-89054-6_6.

[66] F. Cadet and D. T. Fokum, “Coping with denial-of-service attacks on the IP telephony system,” *SoutheastCon*, Mar. 2016, doi: https://doi.org/10.1109/secon.2016.7506691.

[67] V. Ganesan and M. msk, “A scalable detection and prevention scheme for voice over internet protocol (VoIP) signaling attacks using handler with Bloom filter,” *International Journal of Network Management*, vol. 28, no. 2, p. e1995, Aug. 2017, doi: https://doi.org/10.1002/nem.1995.

[68] Ivy, B.P.U.; Priya, M.A. Detection and Prevention of Distributed Denial of Service Attacks in VoIP. Taga J. Graphic Technol. 2018, 14, 1985–2000.

[69] I. M. Tas, B. G. Unsalver, and S. Baktir, “A Novel SIP Based Distributed Reflection Denial-of-Service Attack and an Effective Defense Mechanism,” *IEEE Access*, vol. 8, pp. 112574–112584, 2020, doi: https://doi.org/10.1109/access.2020.3001688.

[70] Zisis Tsiatsikas, Dimitris Geneiatakis, Georgios Kambourakis, and A. D. Keromytis, “An efficient and easily deployable method for dealing with DoS in SIP services,” vol. 57, pp. 50–63, Feb. 2015, doi: https://doi.org/10.1016/j.comcom.2014.11.002.

[71] W. Ahmad and D. Singh, "VoIP security: A model proposed to mitigate DDoS attacks on SIP-based VoIP network," in A Multi-Disciplinary Research Book, vol. 1, Research for Resurgence, Nagpur, India, 2018, pp. 37–48.

[72] H. Mokalled, R. Catelli, V. Casola, D. Debertol, E. Meda, and R. Zunino, “The Guidelines to Adopt an Applicable SIEM Solution,” *Journal of Information Security*, vol. 11, no. 01, pp. 46–70, 2020, doi: https://doi.org/10.4236/jis.2020.111003.

[73] A. S. Putra and N. Surantha, "Internal threat defense using network access control and intrusion prevention system," Int. J. Adv. Comput. Sci. Appl., vol. 10, no. 9, 2019.

[74] W. Bul’ajoul, A. James, and S. Shaikh, "A new architecture for network intrusion detection and prevention," IEEE Access, vol. 7, pp. 18558–18573, 2019.

[75] G. Duppa and N. Surantha, "Evaluation of network security based on next-generation intrusion prevention system," *Telkomnika*, vol. 17, no. 1, pp. 39–48, 2019.

[76] B. Soewito and C. E. Andhika, "Next-generation firewall for improving security in company and IoT network," in Proc. 2019 Int. Seminar Intell. Technol. Its Appl. (ISITIA), Aug. 2019, pp. 205-209.

[77] K. Neupane, H. Rami, and C. Lei, "Next-generation firewall for network security: A survey," IEEE Access, 2018.

[78] W. Zegeye and M. Odejobi, "Telemetry networks cyber security architecture," in Proc. Int. Telemetering Conf., vol. 57, 2022.

[79] P. Wang, “Research on firewall technology and its application in computer network security strategy,” Frontiers in Computing and Intelligent Systems, vol. 2, no. 2, pp. 42–46, Dec. 2022, doi: https://doi.org/10.54097/fcis.v2i2.3931.

[80] A. Trisolino, “Analysis of Security Configuration for IDS/IPS.”

[81] G. Hijazi, *Think Like a TAC Engineer*. Cisco.

[82] Wazuh, “Components - Getting started with Wazuh · Wazuh documentation,” *documentation.wazuh.com*. https://documentation.wazuh.com/current/getting-started/components/index.html

[83] B. R. Patil, M. Moharir, P. K. Mohanty, G. Shobha, and S. Sajeev, “Ostinato - A Powerful Traffic Generator,” *IEEE Xplore*, Dec. 01, 2017. https://ieeexplore.ieee.org/document/8447596 (accessed Nov. 25, 2020).

[84] EVE-NG. EVE-NG documentation. https://www. eve-ng.net/index.php/documentation, Available on June 2023.

[85] M. Schule, “Understanding Cisco FTD and FMC: A Comprehensive Guide,” *Orhanergun.net*, Jun. 28, 2024. https://orhanergun.net/understanding-cisco-ftd-and-fmc-a-comprehensive-guide (accessed Dec. 18, 2024).

[86] A. Arote and U. Mandawkar, “Android Hacking in Kali Linux Using Metasploit Framework,” *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, pp. 497–504, Jun. 2021, doi: https://doi.org/10.32628/cseit2173111.

[87] N. A. L. Mabsali, H. Jassim, and J. Mani, “Effectiveness of Wireshark Tool for Detecting Attacks and Vulnerabilities in Network Traffic,” *www.atlantis-press.com*, Jan. 30, 2023. https://www.atlantis-press.com/proceedings/iciitb-22/125984173 (accessed Mar. 12, 2023).

[88] J. Asokan, A. Kaleel Rahuman, B. Suganthi, Shaik Fairooz, M. Sundar, and V. Elamaran, “A Case Study Using Companies to Examine the Nmap Tool’s Applicability for Network Security Assessment,” Aug. 2023, doi: https://doi.org/10.1109/icoac59537.2023.10249544.

[89] Cloudflare, “What is ICMP? | Internet Control Message Protocol | Cloudflare,” *Cloudflare*, 2023. Available: https://www.cloudflare.com/learning/ddos/glossary/internet-control-message-protocol-icmp/

[90] J. Tang, Y. Cheng, and Y. Hao, “Detection and Prevention of SIP Flooding Attacks in Voice over IP Networks,” *Proceedings IEEE INFOCOM*, 2012.

[91] SOCFortress, “SOCFortress Integrations — Cisco Firepower Management Center (FMC),” *Medium*, Oct. 15, 2024. https://socfortress.medium.com/socfortress-integrations-cisco-firepower-management-center-fmc-9b934a1863b7 (accessed Jan. 22, 2025).