

Project Report on

**Secure Network Deployment with DMZ, iptables, Snort IDS, and Centralized Monitoring using Nagios**

**Submitted by**

**Astha Bhende (240344223042)**

Under the guidance of

**Mr. Gajanan Taur**

**In partial fulfillment of the award of** **Post Graduate Diploma in**

**IT Infrastructure, Systems and Security**

**(PG-DITISS)**



**Sunbeam Institute of Information Technology,**

**Pune (Maharashtra)**

**PG-DITISS -2025**

# DECLARATION

We declare that this written submission represents our ideas in our own words and where others ideas or words have been included; we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed**.**

Place: Pune

Date:

**Astha Bhende**

(240344223042)

# CERTIFICATE

This is to certify that the project report entitled **“Secure Network Deployment with DMZ, iptables, Snort IDS, and Centralized Monitoring using Nagios”**, submitted by **Astha Bhende** is the bonafide work completed under our supervision and guidance in partial fulfillment for the award of Post Graduate Diploma in IT Infrastructure, Systems and Security (PG-DITISS) of Sunbeam Institute of Information Technology, Pune (M.S.).

Place: Pune

Date:

**Mr. Gajanan Taur Mr. Vishal Salunkhe**

Guide Course Coordinator

**Mr. Nitin Kudale**

CEO

Sunbeam Institute of Information Technology

Pune (M.S.) – 411057

# APPROVAL CERTIFICATE

This Project II report entitled **“Secure Network Deployment with DMZ, iptables, Snort IDS, and Centralized Monitoring using Nagios”** by **Astha Bhende(2403442230042)** is approved for Post Graduate Diploma in IT Infrastructure, Systems and Security (PG-DITISS) of Sunbeam Institute of Information Technology, Pune (M.S.).

Place: Pune

Date:

Examiner:

**(Signature)**

**(Name)**

# CONTENTS

|  |  |
| --- | --- |
| **TITLE** | **PAGE NO** |
| **Declaration** |  |
| **Certificate** |  |
| **Approval Certificate** |  |
| **Abstract** | 1 |
| **1.INTRODUCION** | 2 |
| 1.1 Applications | 3 |
| 1.2 Organization and Project Plan | 4 |
| **2. LITERATURE SURVEY** | 5 |
| Paper 1 | 4 |
| Paper 2 | 4 |
| Paper 3 | 5 |
| **3. SYSTEM DEVELOPMENT AND DESIGN** | 7 |
| 3.1 Proposed System | 7 |
| 3.2 Flow Chart | 8 |
| 3.3 Technology used | 9 |
| 3.3.1 Web Application | 9 |
| 3.3.2 Database | 10 |
| 3.3.3 Roundcube | 10 |
| 3.3.4 Nagios | 12 |
| 3.3.5 Security | 13 |
| **4. PROJECT OUTPUT** | 14 |
| **5. CONCLUSION** | 18 |
| 5.1 Conclusion | 18 |
| 5.2 Future Scope | 18 |
| **REFERENCES** | 19 |

# ABSTRACT

With the increasing frequency of cyber attacks, enterprises require a **multi-layered security architecture** to protect sensitive information and ensure uninterrupted services. This project focuses on designing and implementing a secure network using **open-source technologies** such as **pfSense (firewall), Snort (Intrusion Detection System), iptables (internal firewall), and Nagios (monitoring)**.

The architecture is divided into three zones: **Internet, DMZ (Demilitarized Zone), and Private Network**. The **DMZ** hosts public-facing servers (Web Server and Mail Server) protected by Snort IDS, while the **Private Network** contains the Database Server secured by iptables firewall. **pfSense** acts as the perimeter firewall between the Internet and DMZ. **Nagios** provides centralized monitoring and alerting for all critical services.

Additionally, the project integrates **AWS EC2 for deployment, Git for version control, Docker for containerization, and Jenkins for automation**. The outcome is a robust **defense-in-depth security model** that minimizes attack surface, detects intrusions, and ensures system reliability.

This work demonstrates how **open-source tools and DevOps practices** can be combined to build a scalable, secure, and cost-effective enterprise network infrastructure.

# INTRODUCTION

In today’s digital era, organizations rely heavily on interconnected systems to deliver services such as web applications, email, and databases. However, this connectivity also increases exposure to cyber threats including unauthorized access, malware, denial-of-service (DoS) attacks, and data breaches. Traditional single-layer defenses, such as a basic firewall, are no longer sufficient to safeguard critical infrastructures. Instead, **defense-in-depth** — the practice of implementing multiple layers of security controls — has become a necessity for modern enterprise networks.

This project focuses on building a **multi-layered security architecture** that simulates a real-world enterprise environment. The design leverages both **preventive** and **detective** security mechanisms, combining firewalls, intrusion detection, monitoring, and automation to provide end-to-end protection and visibility.

The network is divided into three major zones:

**Perimeter Zone (Internet-facing)** — Managed by **pfSense**, which functions as the first line of defense, controlling inbound and outbound traffic through strict firewall and NAT rules.

**Demilitarized Zone (DMZ)** — Hosts public-facing services such as a **Web server** and a **Mail server**. To detect malicious attempts, **Snort IDS** is deployed here, monitoring inbound and outbound packets for suspicious patterns.

**Private/Internal Zone** — Contains sensitive resources such as a **Database server**, protected by an internal **iptables firewall gateway**, ensuring only authorized traffic from DMZ applications can access it.

## Applications

· **Enterprise Web / Mail Hosting** — host public web and mail services in DMZ so attackers can't reach internal DB directly.

· **Education & Training** — lab environment for students to learn network segmentation, IDS rules, and monitoring.

· **Small Business Security** — affordable baseline security stack for SMBs using open-source tools.

· **Cloud Hybrid Deployments** — design adaptable to cloud (AWS EC2) with security groups and NAT/GW changes.

· **Security Monitoring & Forensics** — Snort and Nagios logs provide evidence and early detection.

## 1.2 Project Plan

**Table: Activities Details**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.**  **No.** | **ACTIVITY** | **WEEK** | | | |
| **1** | **2** | **3** | **4** |
| 1 | Project group formation |  |  |  |  |
| 2 | Project work to be started in respective labs |  |  |  |  |
| 3 | First review with PPT presentation |  |  |  |  |
| 4 | Design Use-Case view as per project |  |  |  |  |
| 5 | Design Block diagram as per project |  |  |  |  |
| 6 | Second review with PPT presentation |  |  |  |  |
| 7 | Selection |  |  |  |  |
| 8 | Final review with PPT presentation |  |  |  |  |
| 9 | Implementation coding as per project |  |  |  |  |
| 10 | Testing, Troubleshooting with different techniques |  |  |  |  |
| 11 | Created Soft copy of project and then final hard copy |  |  |  |  |

# 2. LITERATURE SURVEY

## **Paper 1**: - A Qualitative Study of DevOps Usage in Practice

**Author:** Floris Erich, C. Amrit & M. Daneva

**Description:** Organizations are introducing agile and lean software development techniques in operations to increase the pace of their software development process and to improve the quality of their software. They use the term DevOps, a portmanteau of development and operations, as an umbrella term to describe their efforts. In this paper we describe the ways in which organizations implement DevOps and the outcomes they experience. We first summarize the results of a Systematic Literature Review that we performed to discover what researchers have written about DevOps. We then describe the results of an exploratory interview-based study involving six organizations of various sizes that are active in various industries. As part of our findings, we observed that all organizations were positive about their experiences and only minor problems were encountered while adopting DevOps.

## **Paper 2:** **- Devops, A New Approach To Cloud Development & Testing**

**Author:** Dhaya Sindhu Battina

**Description:** The main purpose of this paper is to explore DevOps and its applications in Cloud development and testing. There's no denying it: DevOps and cloud go hand in hand. This trend will only continue since the bulk of cloud development projects now use DevOps. The advantages of utilizing DevOps with cloud applications are increasingly becoming evident. Competing well in the market necessitates a company's ability to supply services and applications at a rapid rate. To be effective, management procedures and tools need a model that is both swift and dependable. Because of this, we must automate the DevOps processes utilizing cloud and noncloud DevOps automation technologies while designing cloud-native apps. The purpose of this article is to discuss how to migrate DevOps to the cloud and improve software development and operational agility. Likewise, this project will examine ways to expand such DevOps processes and automation to public and/or private clouds. If one is interested in learning more about how the emerging field of DevOps is changing the IT industry, read this paper. Understanding how DevOps and the Cloud work together to aid organizations in transforming themselves is the ultimate objective.

## **Paper 3: -** Review paper on Snort and reviewing its applications in different fields

## Author: Harpreet Sandhu, Manpreet Kaur.

**Description:** In today's era everyone wants security in data transformation but it is very difficult to protect your system and data from attackers. There are some software's and methods which gives you the surety of security like snort. Snort is a network-centric item. As an intrusion identification system, it could investigate movement inline or offline. Snort basically depends ahead a "known bad" alternately "suspected bad" approach, watching movement for examples that relate with pernicious or suspicious action. At snort detects such activity, it called (passive mode) or square (active mode). The primary may be an IDS; the second an IPS. This is a review paper which includes the information about snort, its working, installation process, components of snort, modes of snort, rules of snort and its uses.

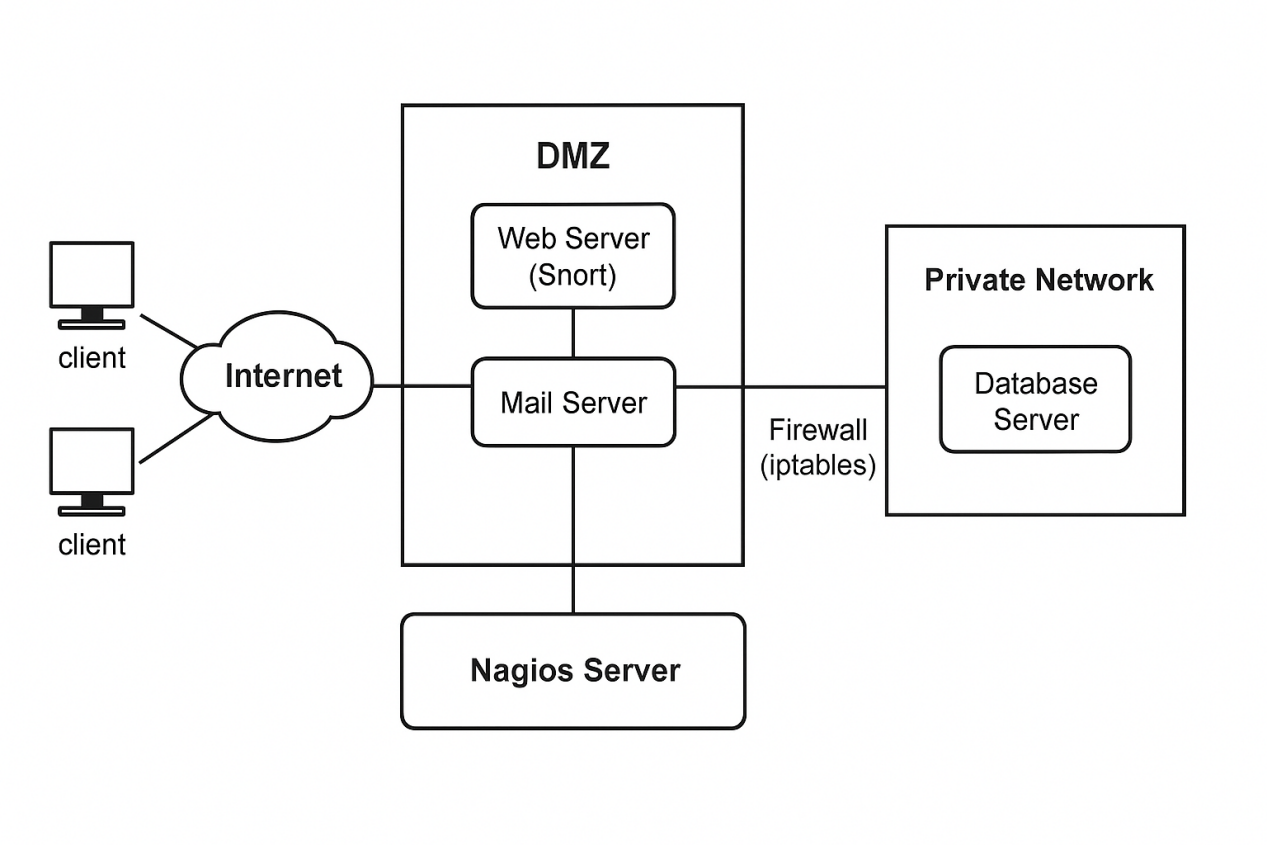
# SYSTEM DEVELOPMENT AND DESIGN

## 3.1 Proposed System

We propose a system where we are setting up two Amazon EC2 instances to host your web application. Storing our application code in a version-controlled repository (Git-Hub).

Set up a CI/CD pipeline using Jenkins. On code changes, trigger an automated build and deployment process using Jenkins. Setting up another EC2 instance to host the Nagios monitoring server. Install Nagios plugins to monitor various aspects of your infrastructure, such as server health, resource usage, and application responsiveness. Configuring AWS Security Groups to control inbound and outbound traffic to our EC2 instances. And also configuring Snort rules to detect various types of network traffic anomalies and security threats.

## 3.2 Flow chart



**Figure: Flowchart**

## 3.3 Technology used

### 3.3.1 Web Application

**Tools:** Flask, Apache2, SSL/TLS

**3.3.1.1 Flask**

Flask is a lightweight Python-based web framework used to build and host the custom web application. It is highly flexible, easy to integrate with databases, and supports modular extensions. In this project, Flask is used to create the front-end user interface that interacts with the backend database.

**3.3.1.2 Apache2**

Apache2 is a powerful and widely used open-source web server. It handles HTTP/HTTPS requests and delivers the Flask application to users securely and efficiently. With its modular architecture, Apache supports features like SSL/TLS encryption, load balancing, and reverse proxying. In this project, Apache2 is used as the hosting server for the web application, ensuring availability and reliability.

### 3.3.1.3 SSL/TLS

SSL (Secure Sockets Layer) and TLS (Transport Layer Security) are cryptographic protocols that secure data communication between client and server. By enabling HTTPS, these protocols encrypt sensitive information such as login credentials and database queries. In this project, SSL/TLS ensures secure data transmission and prevents eavesdropping or man-in-the-middle attacks.

### 3.3.3 Database

### **3.3.3.1 MySQL**

MySQL is a widely used relational database management system (RDBMS) known for its speed, reliability, and ease of use. It supports ACID compliance, replication, and clustering, making it suitable for enterprise-grade applications. In this project, MySQL is used to store structured data such as user information, transactions, and system logs.

### 3.3.3.2 PostgreSQL

PostgreSQL is an advanced open-source RDBMS that provides robust features for handling complex queries, JSON storage, and high-volume data. It is standards-compliant, supports role-based access, and ensures strong data integrity. In this project, PostgreSQL is considered as an alternative to MySQL for organizations requiring advanced query capabilities and scalability.

**3.3.4 Roundcube**

Roundcube is a modern webmail client that offers a user-friendly graphical interface for accessing email via browsers. It includes features such as an address book, drag-and-drop support, and plugin extensions. In this project, Roundcube integrates with Postfix and Dovecot to provide a complete enterprise-grade mail service.

### ****Key Features of Roundcube****

**Web-based Email Access**

Provides a modern, responsive web interface accessible from any browser.

Eliminates dependency on desktop email clients.

**IMAP Support**

Connects directly with IMAP servers (e.g., Dovecot).

Allows folder management, message search, and real-time synchronization.

**User-Friendly Interface**

Intuitive design with drag-and-drop message management.

Supports threaded messages and customizable layouts.

**Address Book Integration**

Built-in contact management.

Supports importing/exporting contacts in vCard format.

**Security Features**

Supports SSL/TLS encryption for secure communication.

Provides authentication integration with mail servers.

**Plugin Support**

Highly extensible with community and custom plugins.

Plugins available for calendars, task management, 2FA, spam filters, etc.

**Multi-language Support**

Available in over 70 languages, making it accessible for global users.

**HTML Email Composition**

Rich text editor with WYSIWYG (What You See Is What You Get) email composing.

Supports attachments and inline images.

**Cross-Platform Compatibility**

Works on all major browsers and operating systems.

### 3.3.5 Nagios XI

### Nagios is an open-source monitoring system that provides comprehensive monitoring and alerting capabilities for IT infrastructure components. It helps organizations monitor the health and performance of their networks, servers, applications, and services, enabling proactive identification and resolution of issues before they impact business operations.

**Key features of Nagios include**:

### Monitoring Hosts and Services: Nagios can monitor various types of hosts (servers, devices) and services (applications, network services) by regularly checking their availability and responsiveness.

### Alerting: Nagios generates alerts when it detects that a monitored host or service has a problem. Alerts can be sent via email, SMS, or other notification methods to ensure timely response and issue resolution.

### Threshold Monitoring: Nagios enables you to define thresholds for various metrics (CPU usage, memory usage, response time) and generate alerts when those thresholds are exceeded.

### Flexible Notification: Nagios supports flexible notification configurations, allowing you to define who should be notified based on the time of day, the severity of the issue, and other criteria.

### Plugins: Nagios uses plugins to perform monitoring checks. There are a wide variety of pre-built plugins available, and you can also create custom plugins to monitor specific aspects of your environment.

**3.3.6 Security**

**3.3.6.1 Snort IDS**

Snort is an open-source Intrusion Detection System (IDS) that analyzes network traffic in real-time. It uses signature-based and anomaly-based detection techniques to identify suspicious activity such as brute-force attacks, port scans, and malware signatures. In this project, Snort is deployed in the DMZ to detect malicious traffic targeting web and mail servers.

**3.3.6.2 fail2ban**

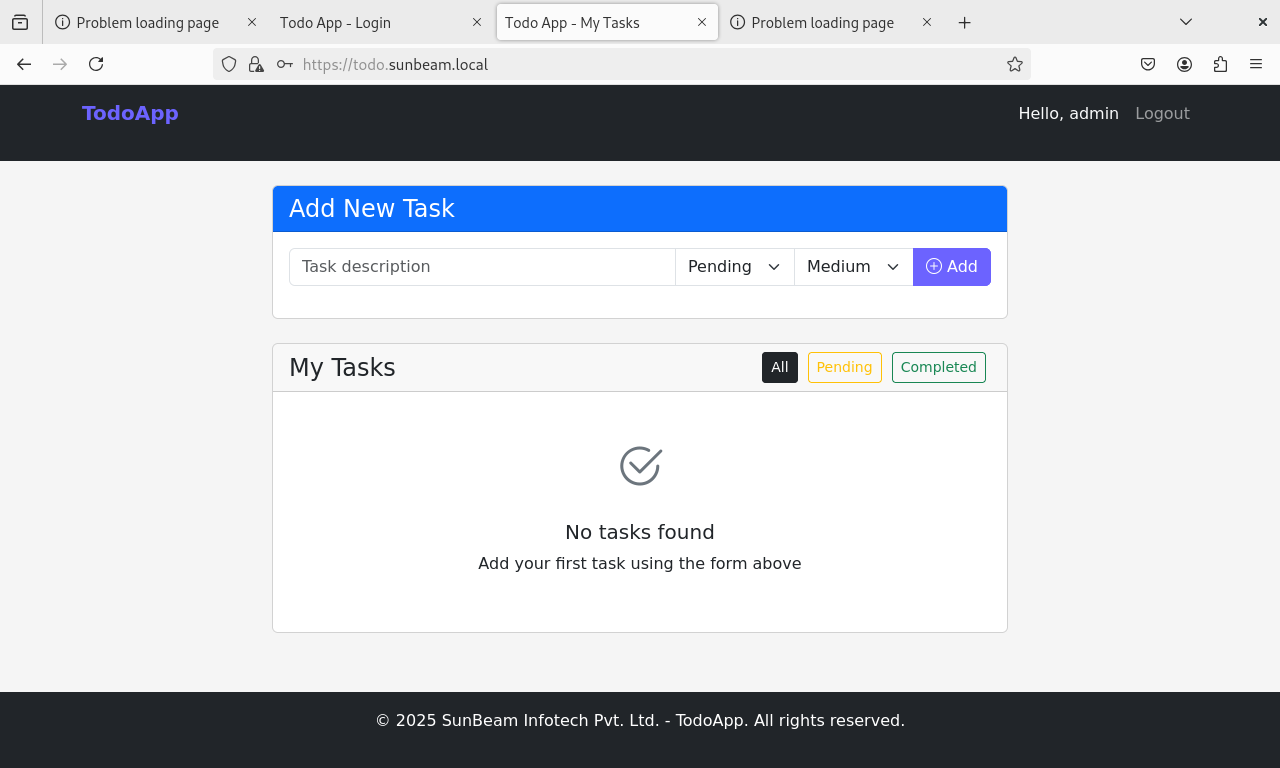
fail2ban is a security tool that protects servers from brute-force attacks by monitoring log files for repeated failed login attempts. Once an attack is detected, fail2ban automatically blocks the offending IP address by updating firewall rules. In this project, fail2ban is used to harden services such as SSH and Apache against repeated intrusion attempts.

**3.3.6.3 iptables**

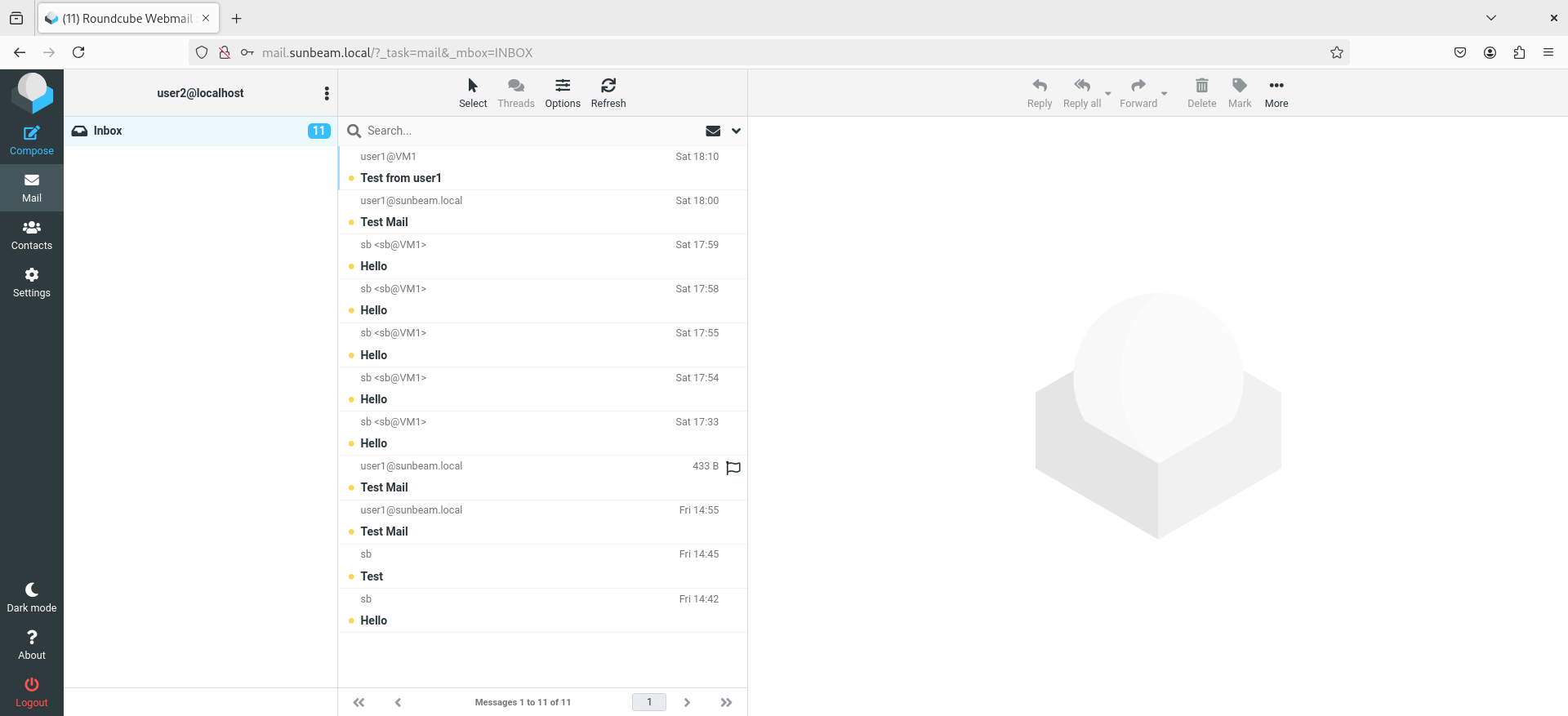
iptables is a Linux-based firewall utility that provides packet filtering, NAT (Network Address Translation), and port forwarding. It is highly flexible and enables administrators to define granular security policies. In this project, iptables is configured as the internal firewall to restrict unauthorized access to the database and internal network.

# 4. Project Output

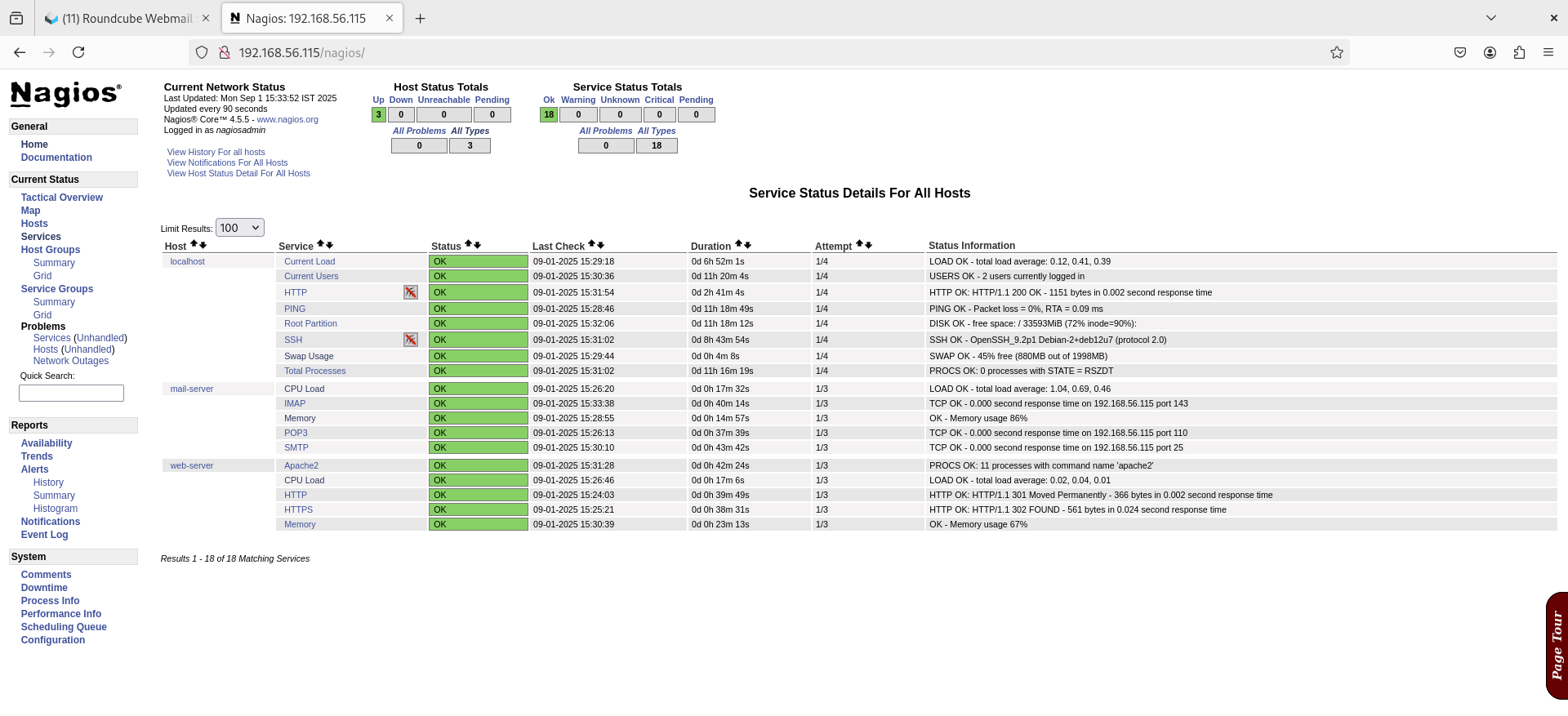
## Todo App



## Roundcube

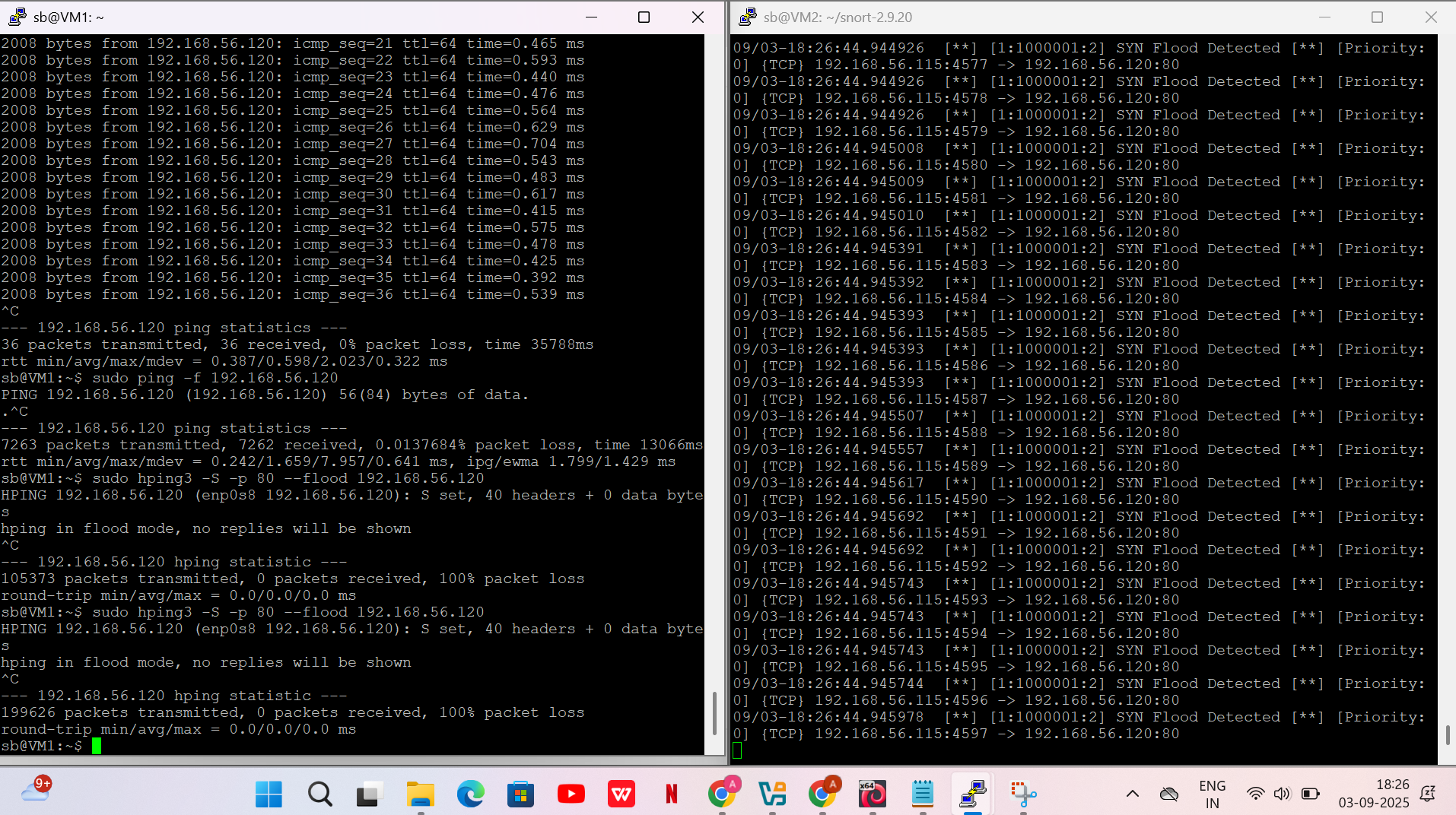


## Nagios

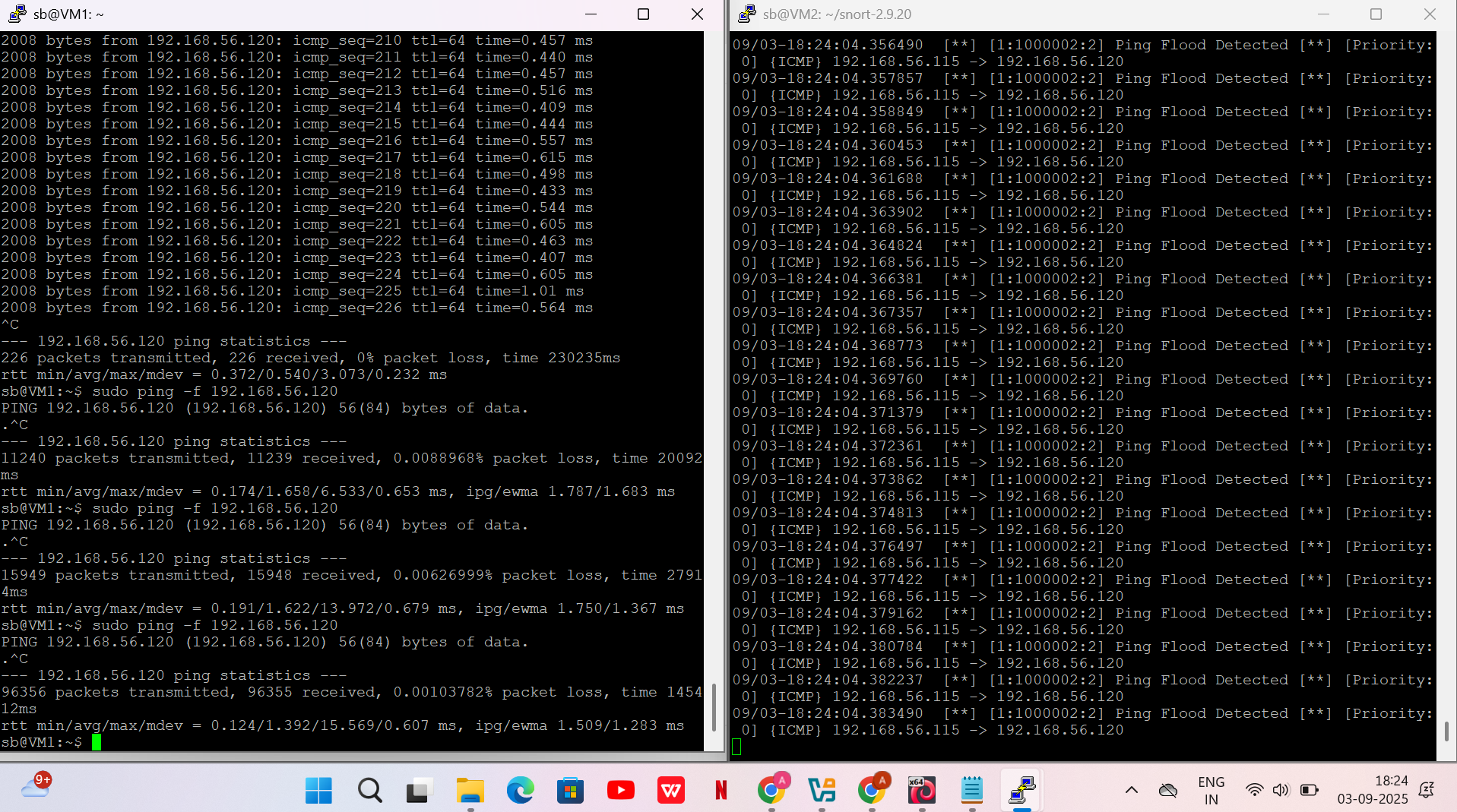


## Snort

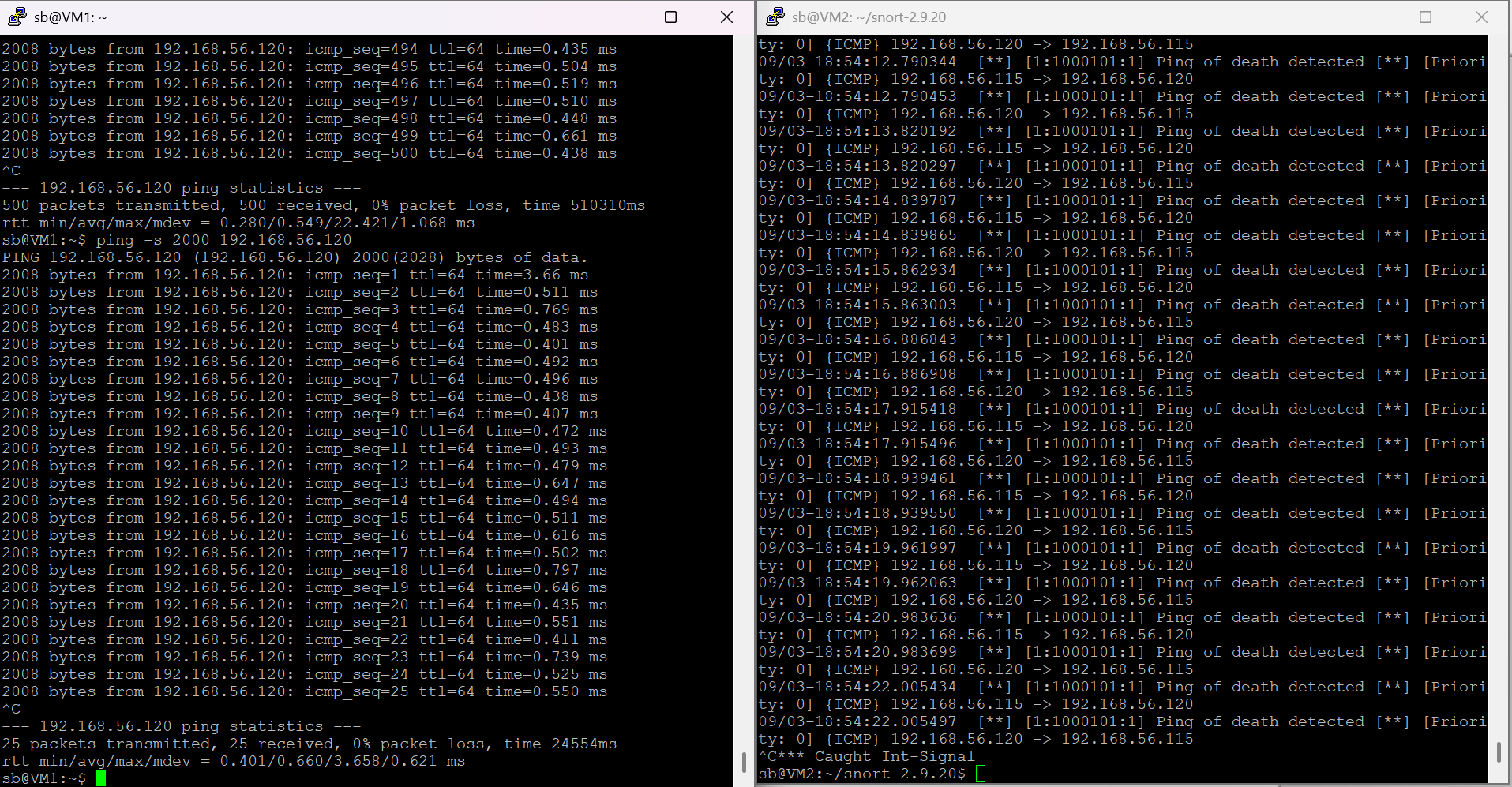
* + 1. **SYN Flood**

****

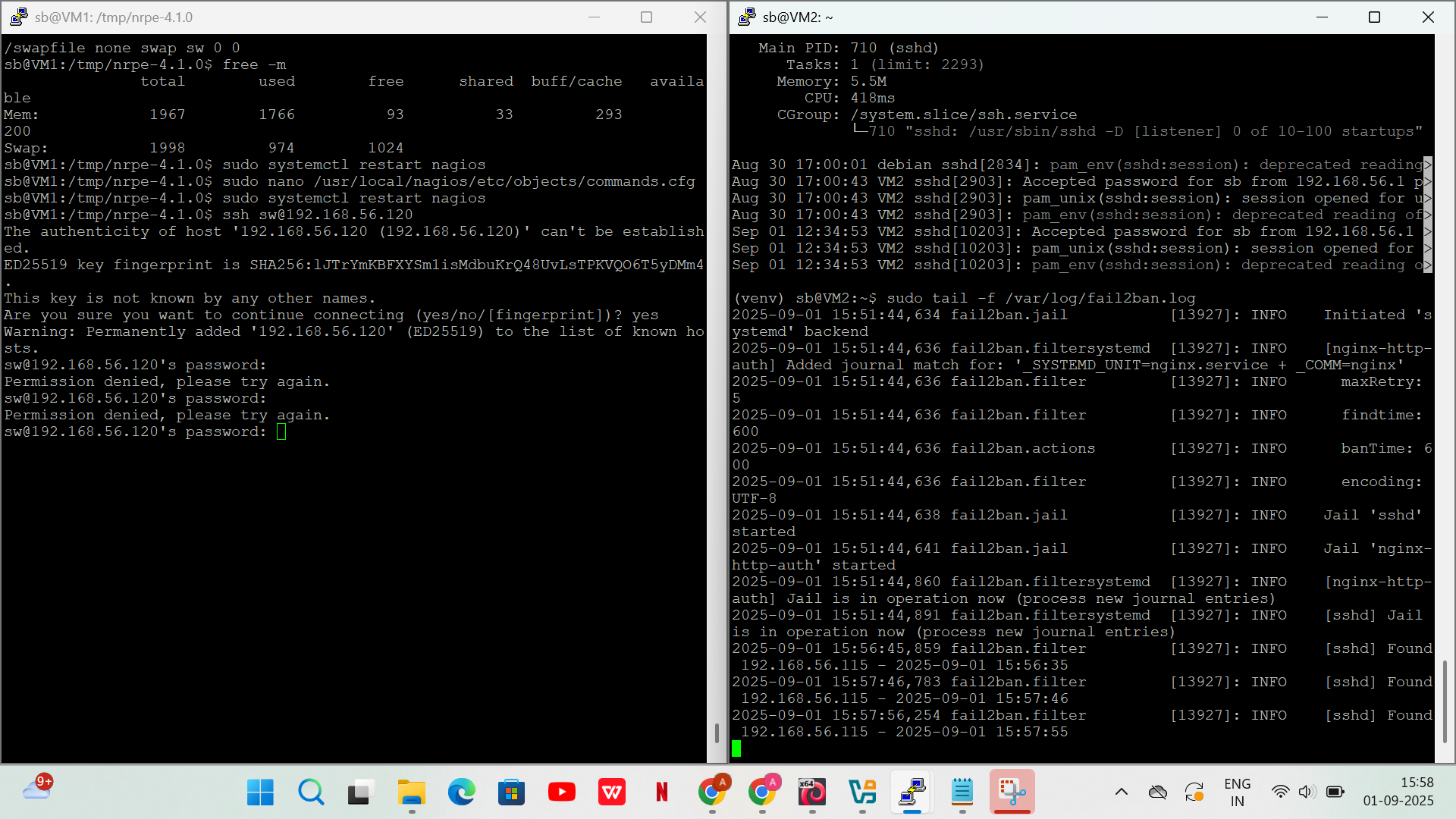
**4.4.2 Ping Flood**

****

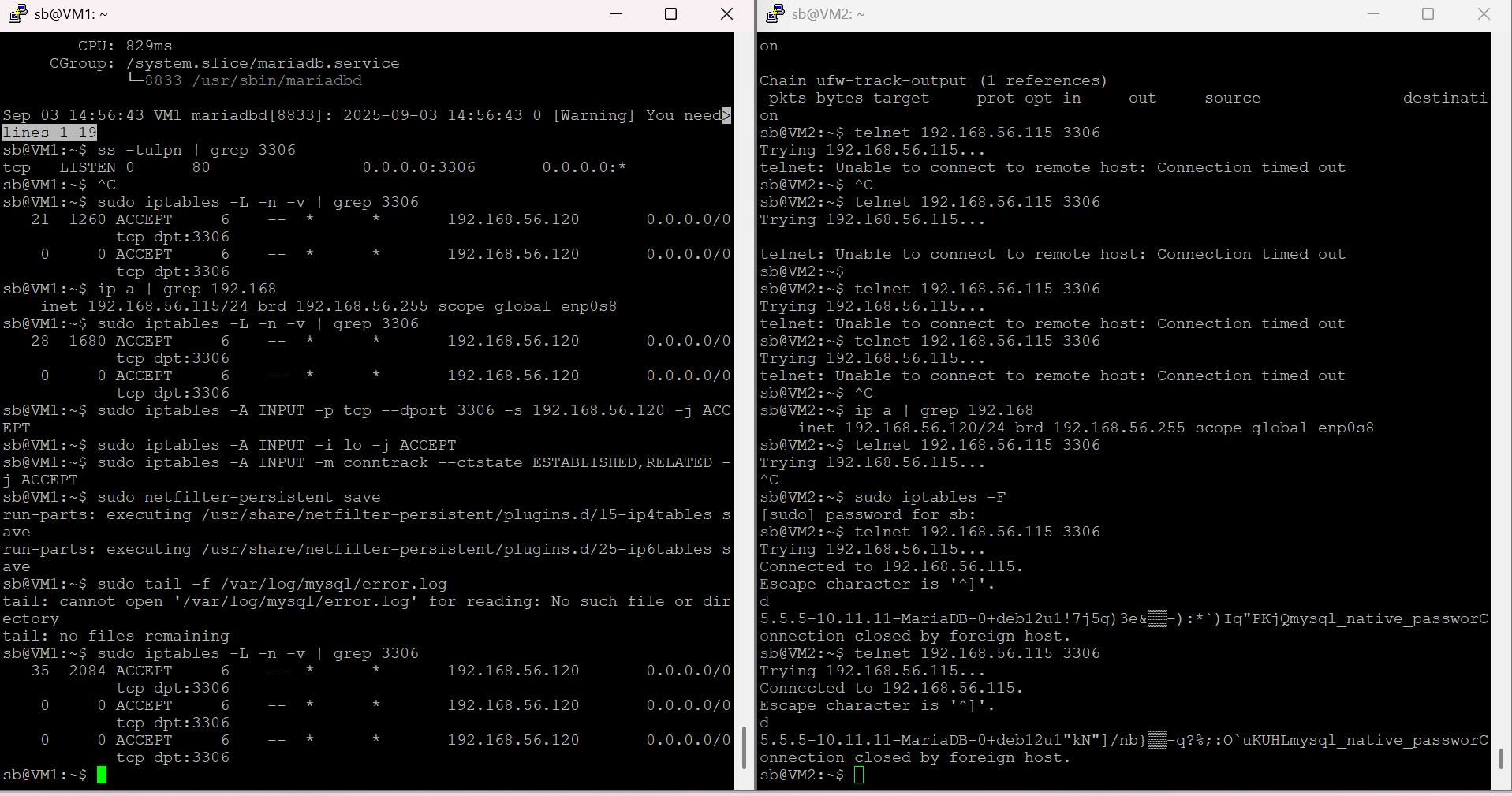
* + 1. **Ping of Death**

****

* 1. **fail2ban**



* 1. **Iptables Rules**

****

# 5. CONCLUSION

## 5.1 Conclusion

This project shows how to combine perimeter filtering (pfSense), DMZ service isolation, host-based IDS (Snort), internal segmentation (iptables), and centralized monitoring (Nagios) into a single coherent security solution. The layered approach reduces attack surface, improves detection capabilities, and provides administrators with visibility for proactive response.

## 5.2 Future Scope

**SIEM Integration:** forward Snort & system logs to ELK/Graylog/SecurityOnion for correlation and long-term analytics.

**Upgrade IDS:** evaluate Suricata (multithreaded) or deploy IDS inline as IPS for blocking.

**Automated Response:** integrate with SOAR platforms to automate containment (e.g., block IP in pfSense on high-confidence Snort alert).

**High availability:** pfSense HA, DB clustering, replicated Nagios (or use Prometheus + alertmanager).

**Policy as Code:** Use Ansible/Puppet/Terraform for reproducible deployments & compliance checks.

# REFERENCES

**Paper 1:** - A qualitative study of DevOps usage in practice. Journal of Software: Evolution and Process, 29(6), e1885. https://doi.org/10.1002/smr.1885

Author: Erich, F., Amrit, C., & Daneva, M. (2017).

**Paper 2:** - Devops: A new approach to cloud development and testing. International Journal of Advanced Research in Computer Science, 7(2), 15–18.

Author: Battina, D. S. (2016).

**Paper 3:** - Review paper on Snort and reviewing its applications in different fields. International Journal of Computer Science and Mobile Computing, 3(6), 119–123.

Author: Sandhu, H., & Kaur, M. (2014)