

# **Project Based Evaluation**

**Project Report**

**Semester-IV (Batch-2023)**

**ZenWall**



**Supervised By:**

Mrs. Baljit Kaur

**Submitted By:**

Maridul Walia, 2310991897, G21

Mehak Walia, 2310991901, G21

Michael Dhiman, 2310991902, G21

Ramya Padala, 2310992587, G21

**Department of Computer Science and Engineering  
Chitkara University Institute of Engineering & Technology,  
Chitkara University, Punjab**

## **Abstract**

ZenWall is a Bash-scripted GUI firewall and network configuration utility built with Zenity for Linux operating systems. ZenWall is aimed at making advanced firewall operations easy to perform using a graphical interface so that system security becomes simple for non-tech people.

ZenWall fills the gap between robust yet complicated firewall utilities and the requirements of end-users who would like or need a more straightforward interface. It is particularly ideal for students, teachers, and novice system administrators who want to be able to control network security with little risk of misconfiguration. In general, ZenWall improves system safety and facilitates learn-by-doing hands-on learning of firewall principles in a controlled context.

## Table of Contents

Sr. No	Section	Page No.
1.	Introduction	1
2.	Problem Definition and Requirements	3
3.	Proposed Design	4
4.	Results	7
5.	Testing and Validation	29
6.	UML Diagrams	31
7.	References	34

# 1. Introduction

## 1.1 Background

As cybersecurity attacks and internet-accessible systems increased, firewalls became an important mechanism to prevent Linux-based systems from being accessed without authorization. Historically, command-line interfaces such as iptables and ufw have been the preferred utilities for controlling firewall rules. Unfortunately, for many users — and particularly new users of Linux — these utilities are hard to use. Command-line errors may lead to misconfigured systems, leaving them vulnerable to attacks or even rendering users locked out. Although most Linux users are highly technical, more and more users from a wide range of backgrounds are embracing Linux due to its flexibility and open-source aspect. These users need easier tools to handle important features such as firewalls and network setup.

Zenity is a minimalist and effective means to incorporate graphical dialogs into shell scripts. This qualifies it as an ideal tool to build light-weight GUI applications in Linux without recourse to other programming frameworks. Relying on Zenity, ZenWall brings a simplified and step-by-step interface for vital network and firewall administration.

Manual rule management is prone to errors, though. ZenWall mitigates these obstacles by compartmentalizing firewall commands into rational GUI-based actions, prompting proper usage and consistency throughout sessions. It also promotes user awareness of real-time bandwidth consumption, incoming connections, and system vulnerabilities — enabling improved decision-making and security stance.

## 1.2 Objectives

- To provide an easy-to-use firewall configuration tool with a graphical interface.
- To abstract complex firewall and networking commands into simple, menu-driven options.
- To allow users to perform tasks such as enabling/disabling UFW, configuring ports, and blocking IPs without command-line involvement.

- To incorporate real-time monitoring, suspicious IP detection, and log maintenance within a single interface.
- To design a modular, script-based tool that is portable, extensible, and efficient.
- To promote awareness of system security among new users and enable safe experimentation and learning.
- To offer a framework that encourages novice Linux users to explore secure practices through intuitive interaction.
- To demonstrate the potential of Zenity and Bash integration for automating critical administrative tasks.

### **1.3 Significance**

ZenWall bridges the gap between advanced terminal-based tools and new users as well as everyday Linux users. Removing the terror factor from using terminal commands, ZenWall permits additional users to enact useful firewall policies and define their network configurations safely. The strength of ZenWall is its ability to expose security tools to more people — particularly in the context of schools' networks, home Linux configurations, or small business networks where there is not room to pay experienced IT professionals.

Besides, ZenWall is a learning tool. It shows how easy shell scripting and Linux commands can be built into a complete application. This makes it an excellent case study for both students and teachers in Linux administration, cybersecurity, and scripting.

It also facilitates enhanced adherence to best practice with logging of user activity and session history. Such auditability makes ZenWall feasible for use in applications requiring surveillance or debugging. The project acts as a template for other CLI-to-GUI conversion projects, encouraging developers to develop tools that are accessible without sacrificing functionality.

## **2. Problem Definition and Requirements**

### **2.1 Problem Statement**

The majority of Linux firewalls assume good command-line and networking skills. This holds back new users, students, or administrators who do not have lots of Linux experience from properly hardening their systems. There is a need for a simple GUI tool that encapsulates the main firewall capabilities and network diagnosis without losing transparency and adjustability.

### **2.2 Software Requirements**

- Operating System: Linux (Ubuntu, Kali, Debian-based distributions)
- Shell Environment: Bash
- GUI Toolkit: Zenity (for displaying dialogs)
- Network Tools: ufw, ss, ip, nmap, gnome-terminal, and optionally iftop for bandwidth monitoring

### **2.3 Hardware Requirements**

- RAM: Minimum 2GB (recommended 4GB for monitoring tools)
- CPU: 1 GHz processor or better
- Storage: 500MB free disk space for logs, snapshots, and tool installation
- Display: GUI-enabled desktop environment (GNOME)

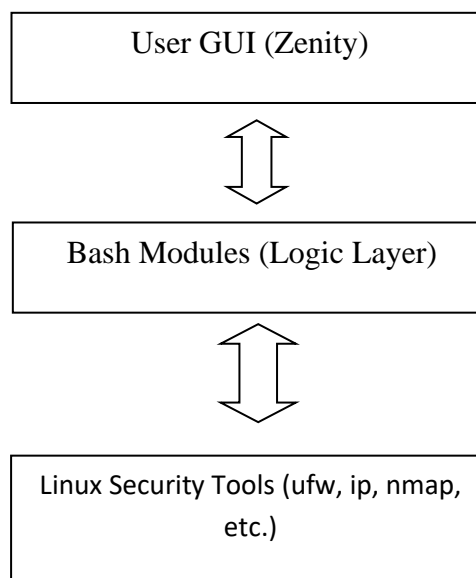
## 3. Proposed Design

### 3.1 Overview

ZenWall is implemented as a modular shell script, utilizing Zenity for graphical interfaces. Each function, such as enabling the firewall or scanning for suspicious IPs, is encapsulated within a dedicated script block. The tool is invoked through a main menu, which displays options using a Zenity list dialog. Based on the user's choice, the corresponding action is executed.

The modularity of ZenWall allows for easy addition or removal of features. Each component — user interface, backend logic, logging, and snapshot generation — is handled in its own scope, ensuring clear separation of concerns.

### 3.2 Architecture Diagram



- User GUI: Created using Zenity dialogs, captures user input.
- Bash Modules: Handle logic, validation, command invocation.
- Linux Network Tools: Execute system-level actions and return result

### 3.3 File Structure

/home/user/

- |— firewall\_config.sh           # Main execution script
- |— firewall\_network.log       # Log file for user actions and events
- |— firewall\_network\_error.log   # Log file for errors
- |— firewall\_snapshot\_DATE.txt # Timestamped firewall status backups

/home/user/Desktop/

- |— firewall\_config.desktop       # Desktop shortcut (optional)

### 3.4 Core Functional Components

- Main Menu: A Zenity listbox offering options like enable/disable firewall, allow/deny ports, block IP, etc.
- Firewall Toggle: Enables or disables UFW using shell commands.
- Port Management: Prompts user for port input and applies UFW rules.
- IP Management: Blocks or unblocks specific IP addresses.
- Network Info: Displays system's network stats (ip, ss, hostname, etc.).
- Monitoring Tools: Launches iftop in a terminal window for live bandwidth viewing.
- Suspicious Scan: Uses nmap to detect potentially vulnerable open ports.
- Snapshot & Logs: Captures firewall state into a timestamped file and maintains a log of all actions.
- Strict Lockdown: The Strict Lockdown feature blocks all incoming network connections except from one trusted IP address, effectively isolating the system for maximum security. It also creates a backup of existing firewall rules before applying lockdown, allowing for easy restoration later.



### **3.5 User Flow and Interaction Design**

The user interaction is intuitive and follows a linear flow:

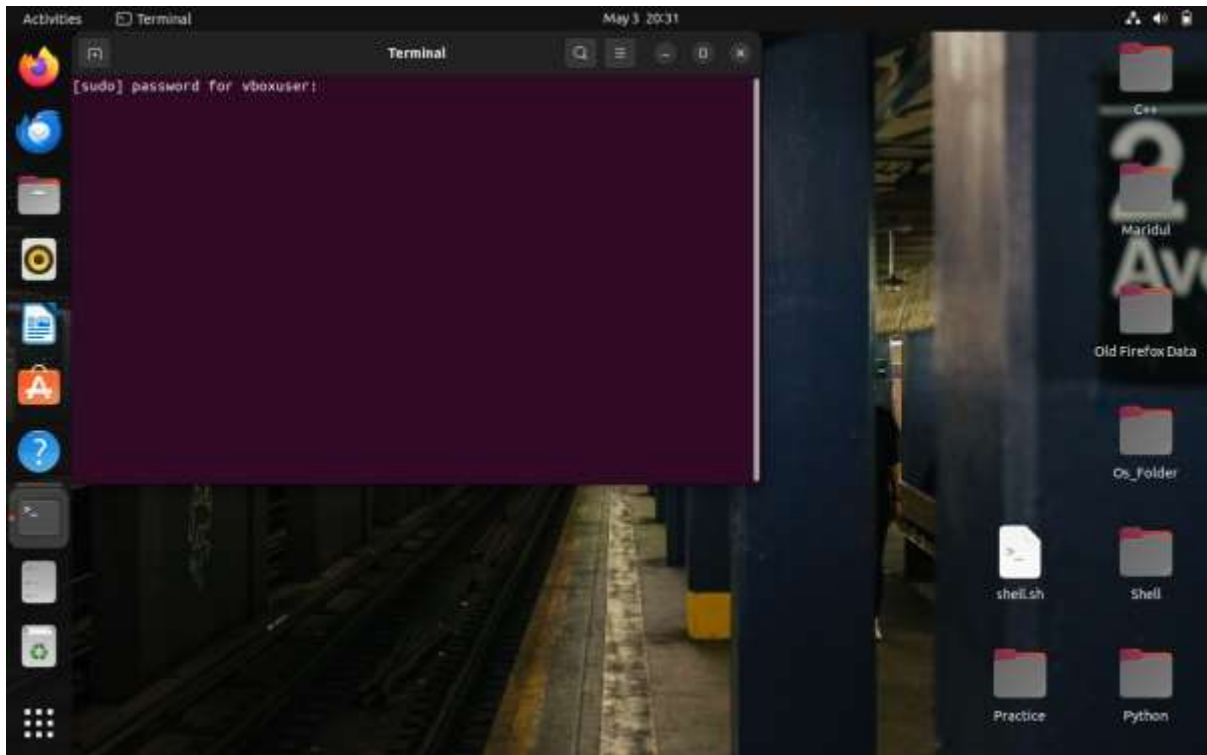
- Launch ZenWall via terminal or desktop shortcut
- Choose an option from the main menu (e.g., toggle firewall, block IP)
- Provide required input in Zenity prompts
- View success messages or output in dialogs
- Optionally, view the generated log or snapshot file
- Exit or continue using other features

### **3.6 Design Principles**

- Modularity: Each function resides in a separate logic block
- User-Centered Design: Simplified dialog-based approach
- Safety: All commands include validation and error-checking
- Portability: No external dependencies beyond Zenity and standard tools
- Extensibility: New features can be added without breaking structure

## 4. Results

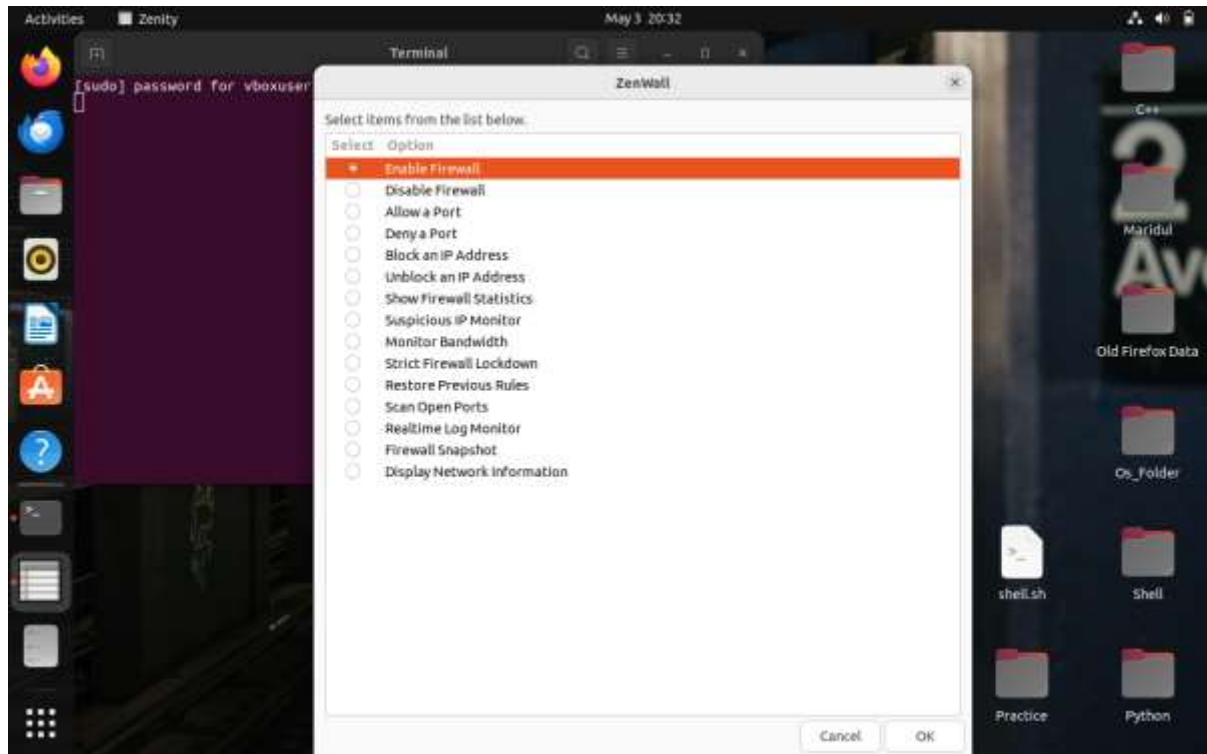
### 4.1 Root Privilege Prompt



#### 4.1.1 Root User prompt

Certain operations within ZenWall require elevated privileges to ensure they execute correctly and securely. For instance, tasks such as enabling or disabling UFW, blocking or unblocking ports, or scanning the network for suspicious activity often require sudo access. To handle this elegantly, ZenWall is designed to detect whether the user has the necessary permissions and, if not, launches a terminal window that prompts for the root password.

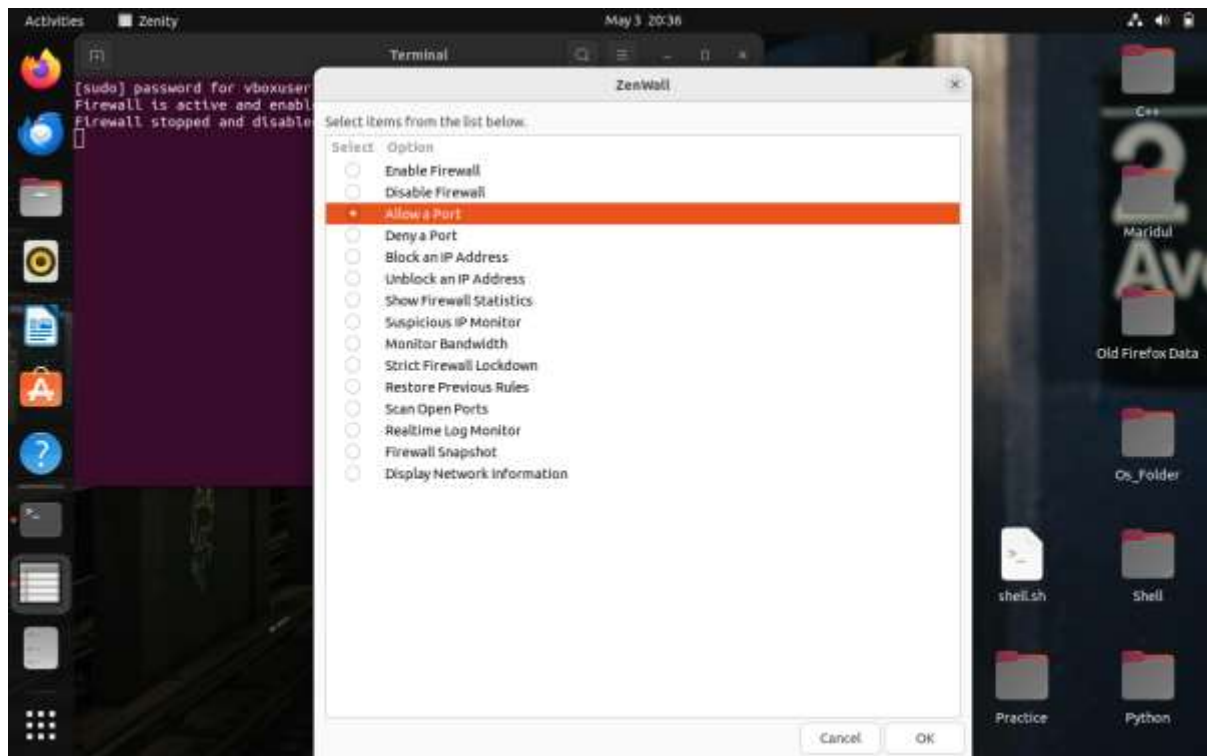
## 4.2 Main Menu



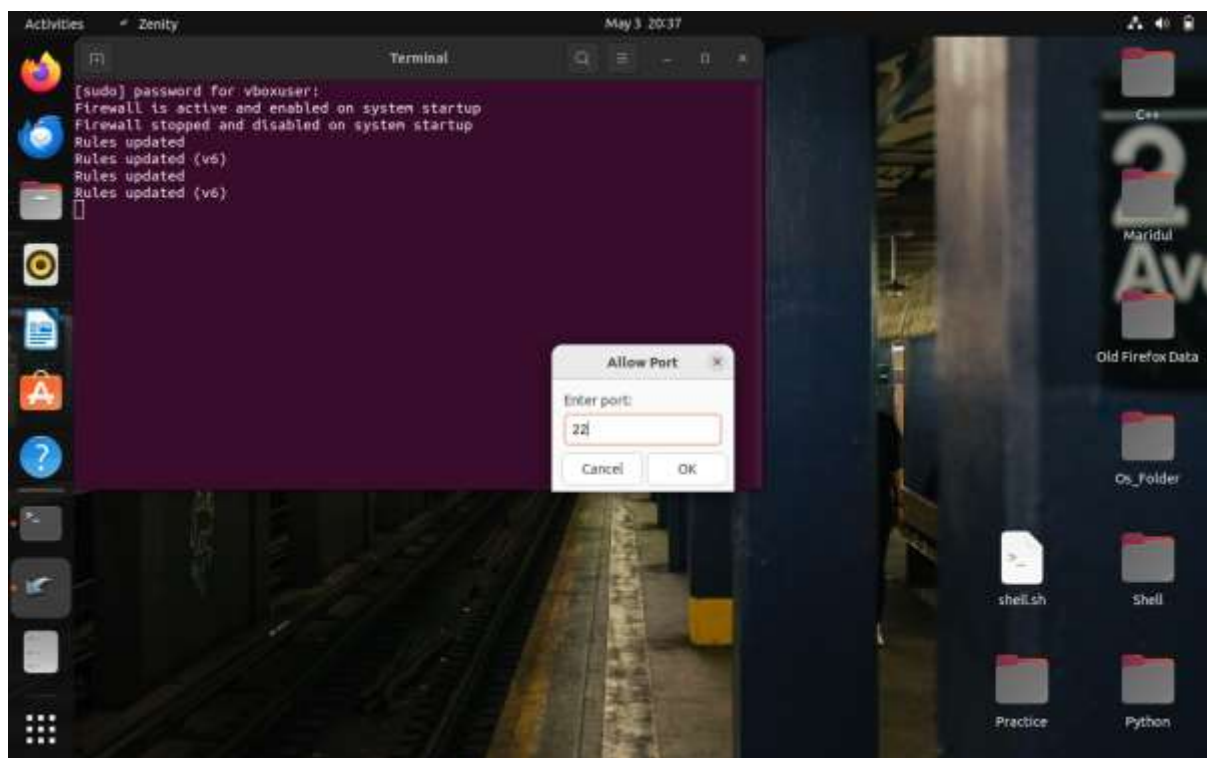
### 4.2.2 Main Menu

The tool launches a categorized menu listing available firewall and network tasks. Users select an option through a Zenity list box, which then triggers the appropriate action.

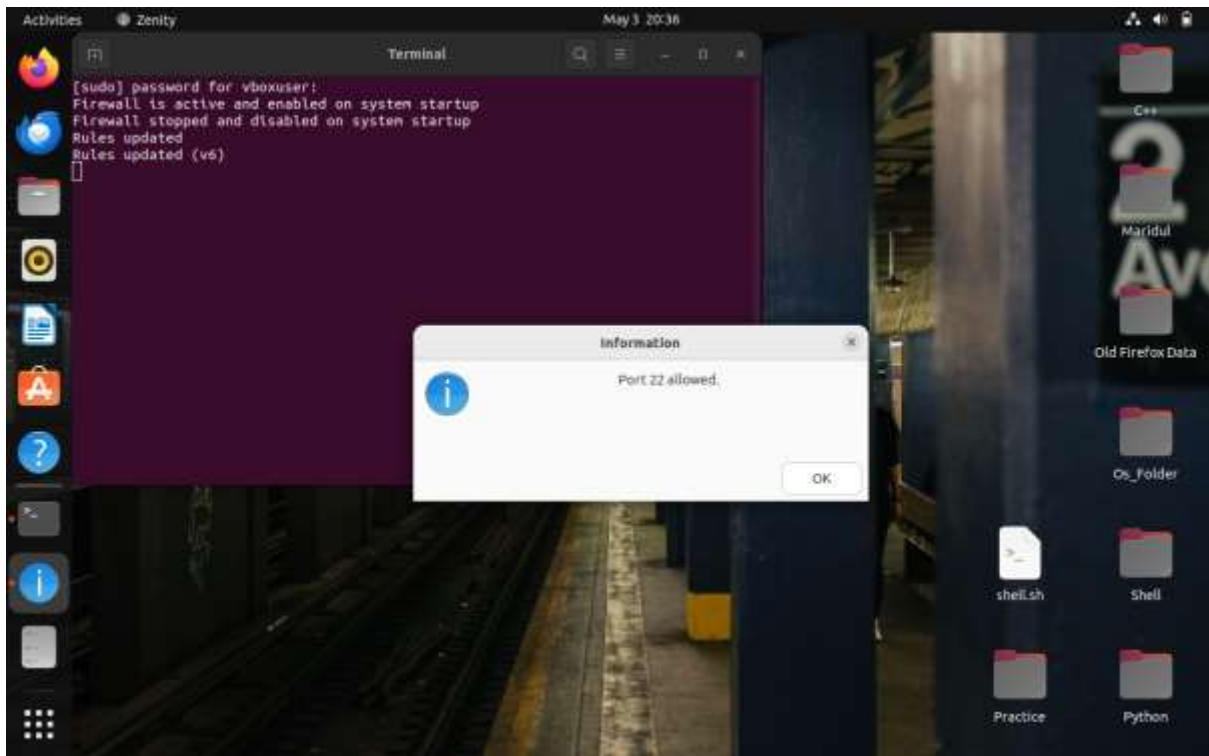
## 4.3 Port Management



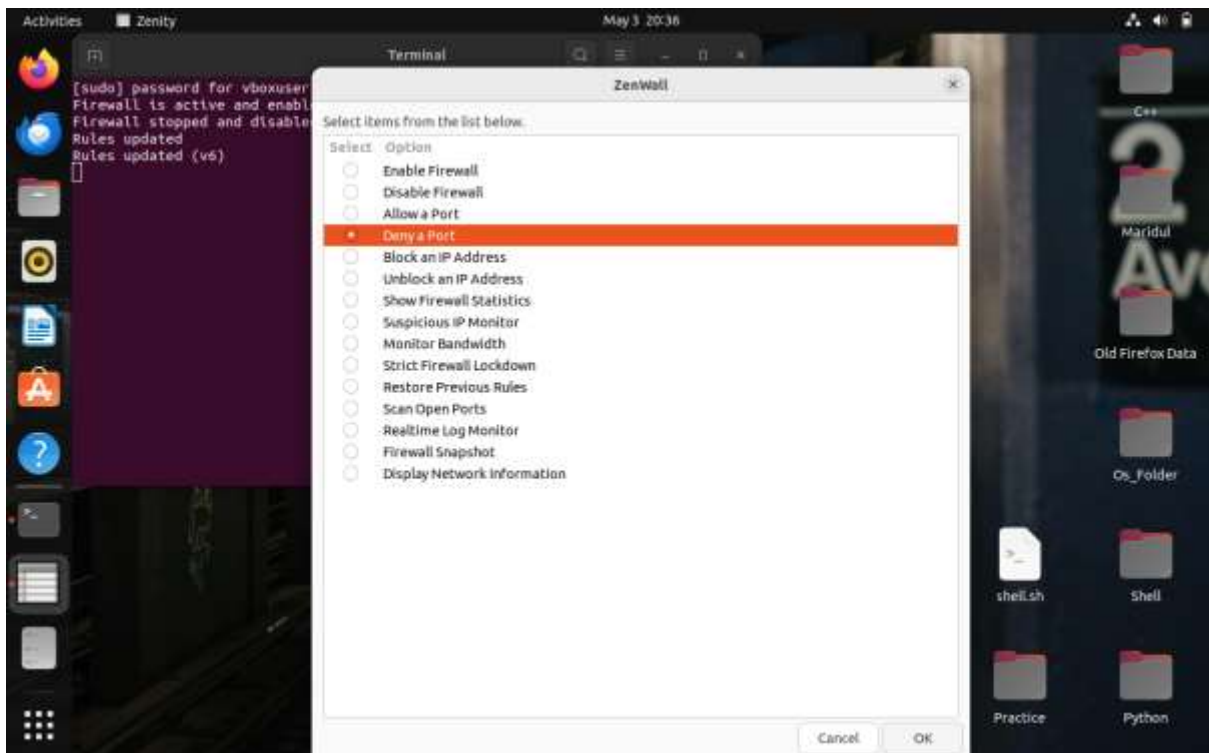
### 4.3.1.1 Allow Port



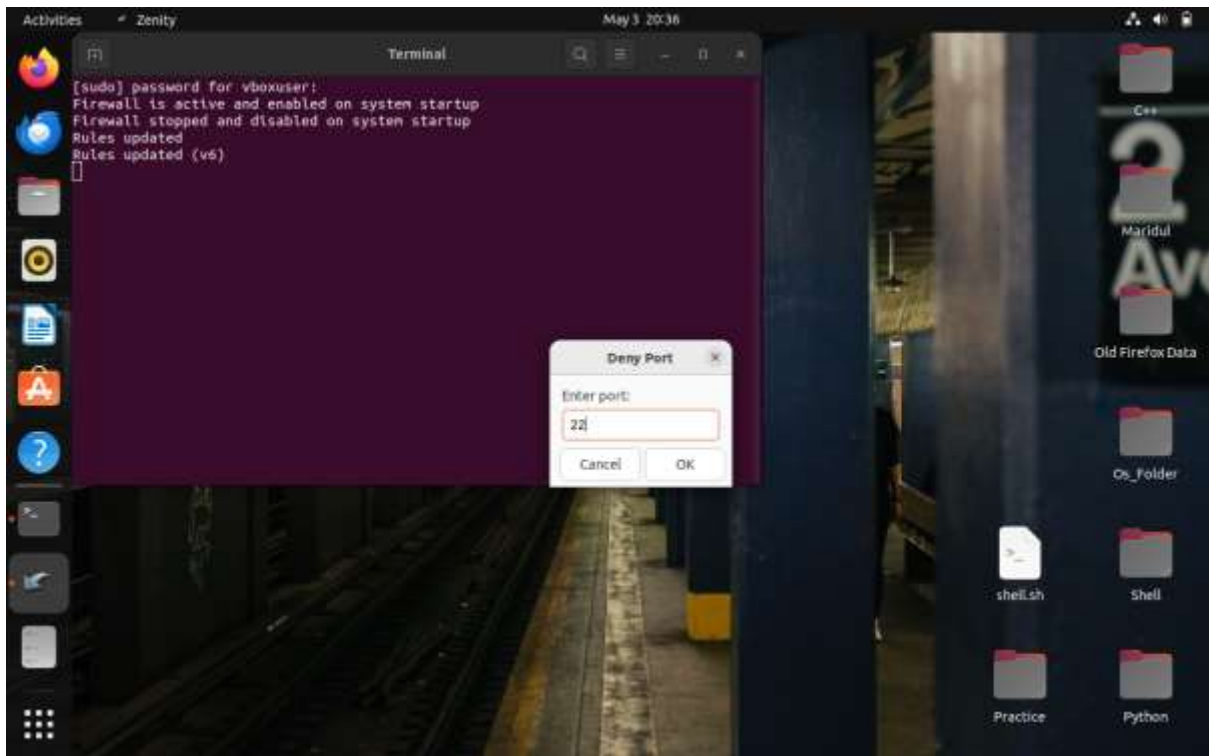
### 4.3.1.2 Enter Port



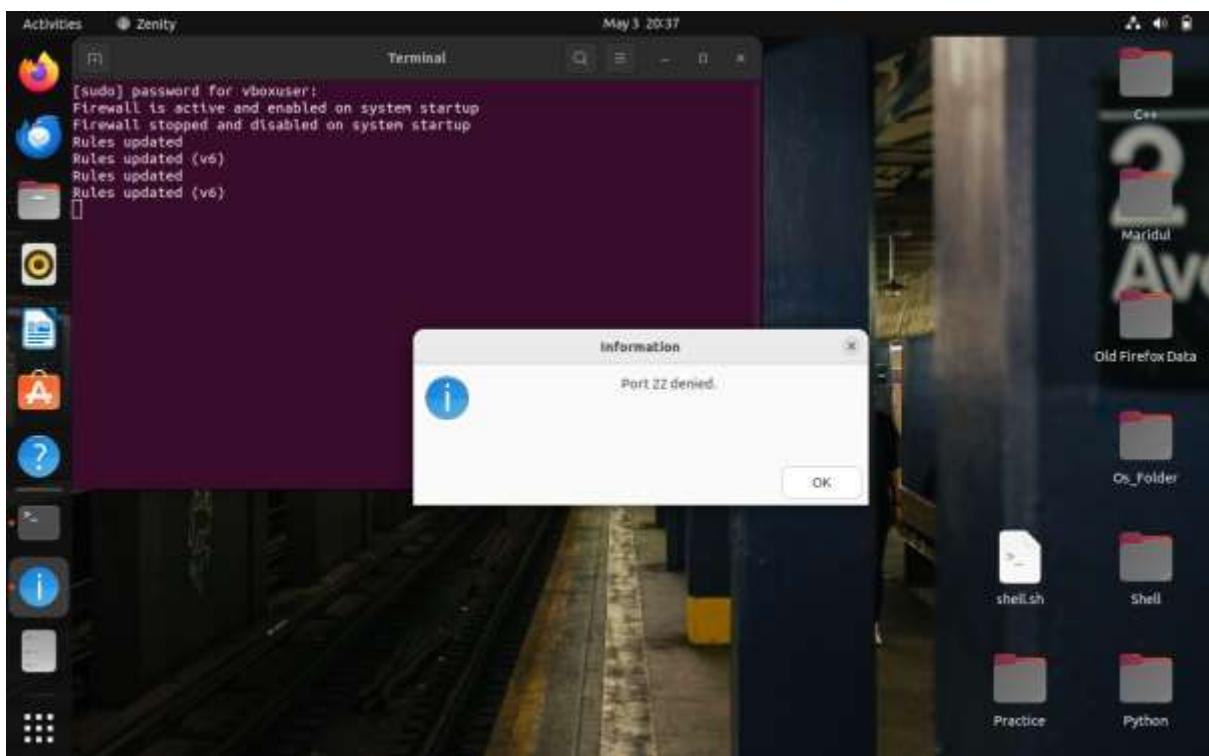
#### 4.3.1.3 Port Allowed



#### 4.3.2.1 Deny Port



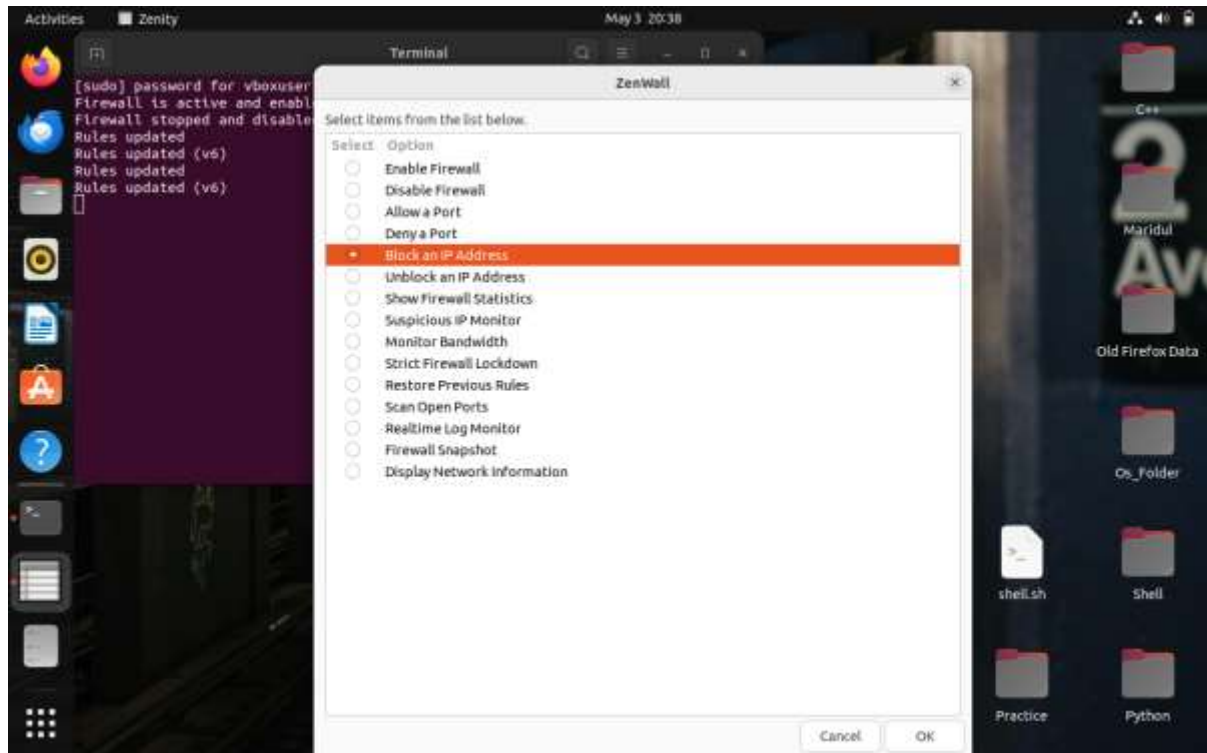
#### 4.3.2.2 Enter Port



#### 4.3.2.3 Port Denied

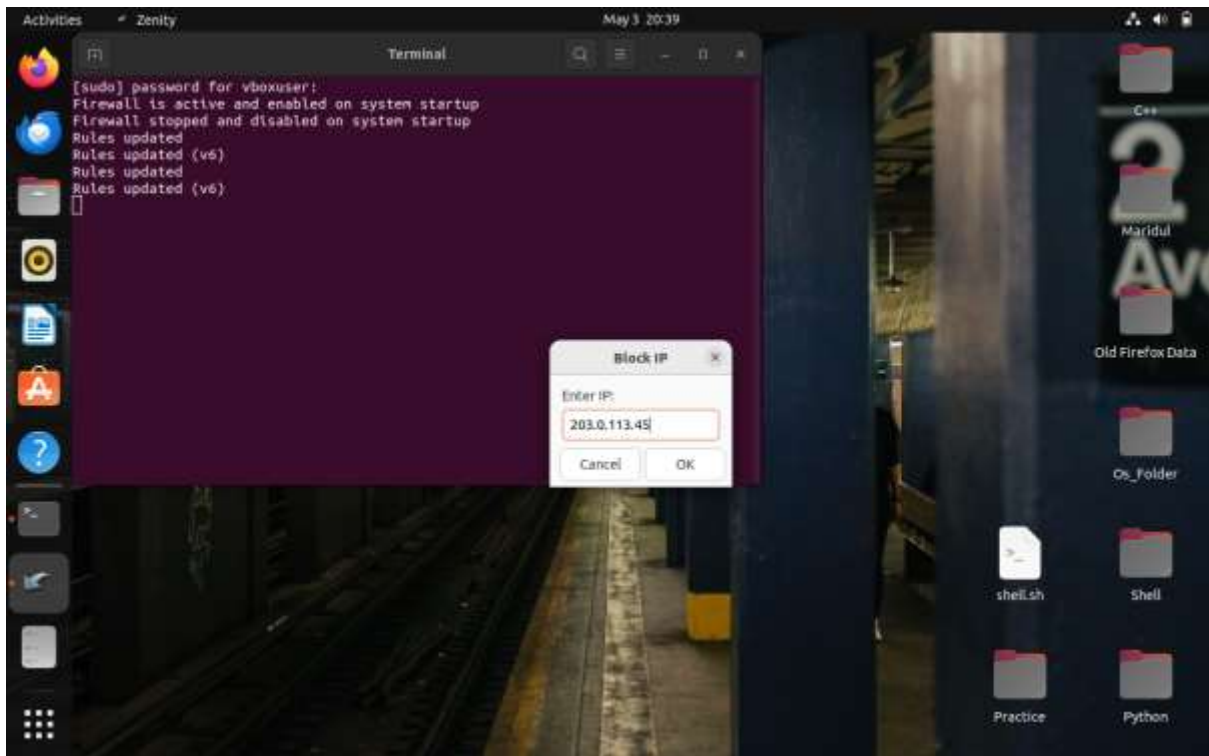
When selected, the user is prompted to enter a port number. The script validates input and then applies UFW rules to allow or deny access. Actions are logged with timestamps for auditing.

## 4.4 IP Blocking/Unblocking

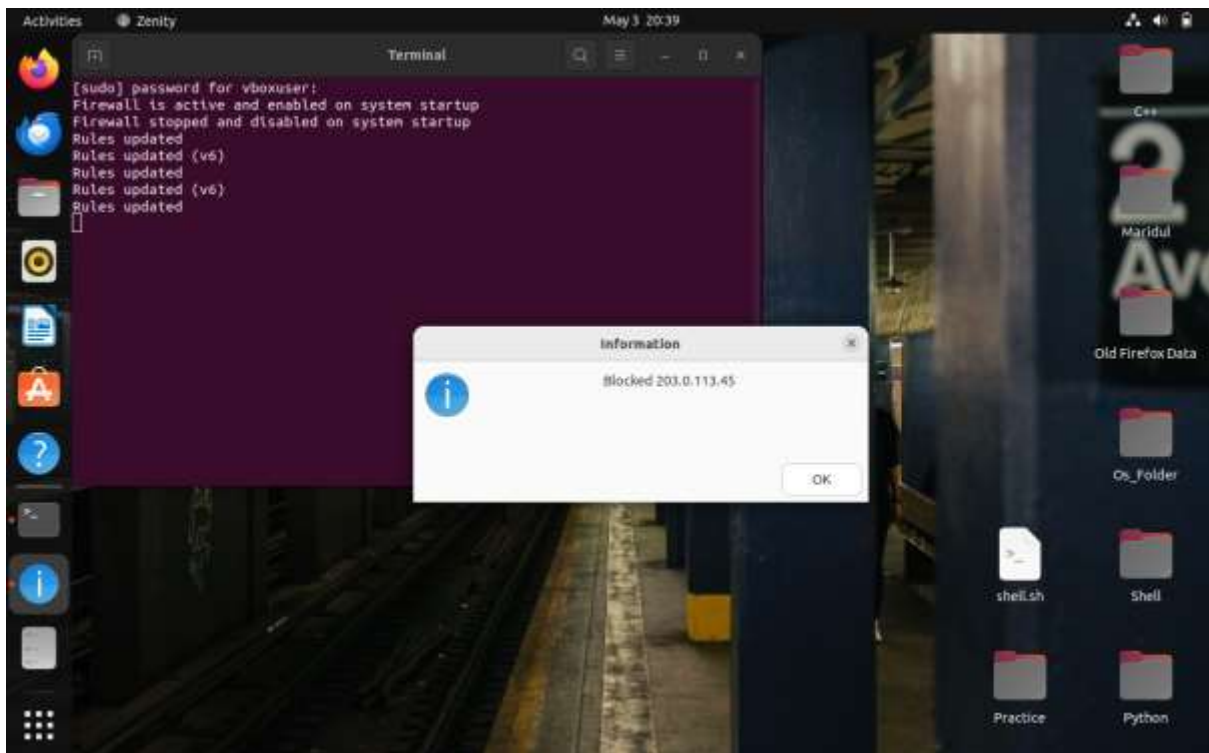


### 4.4.1.1 Block an IP Address



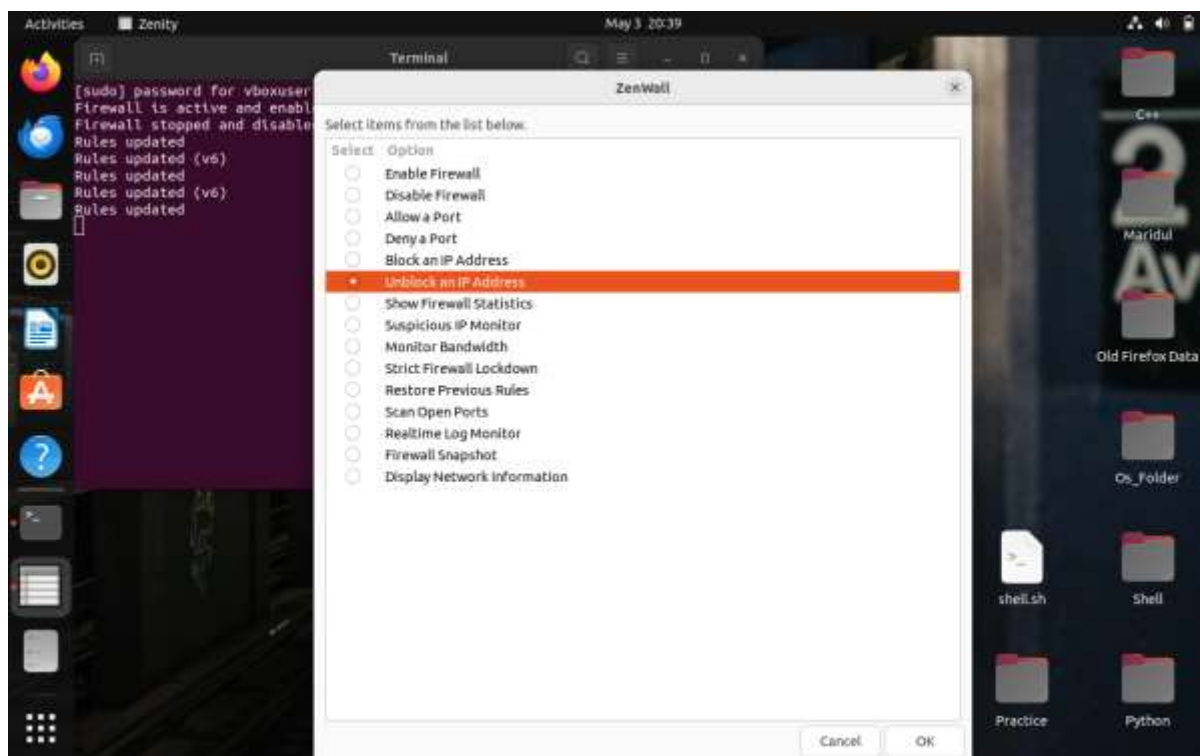


#### 4.4.1.2 Enter an IP Address

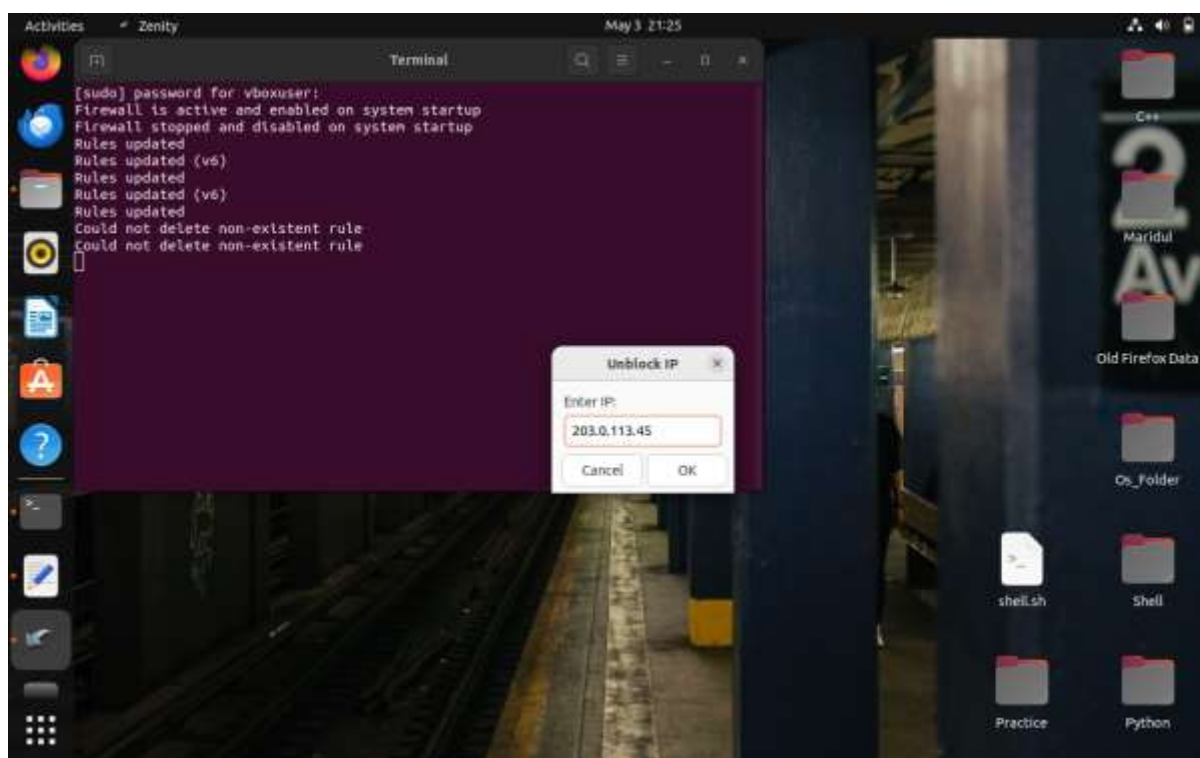


#### 4.4.1.3 Blocked IP Address

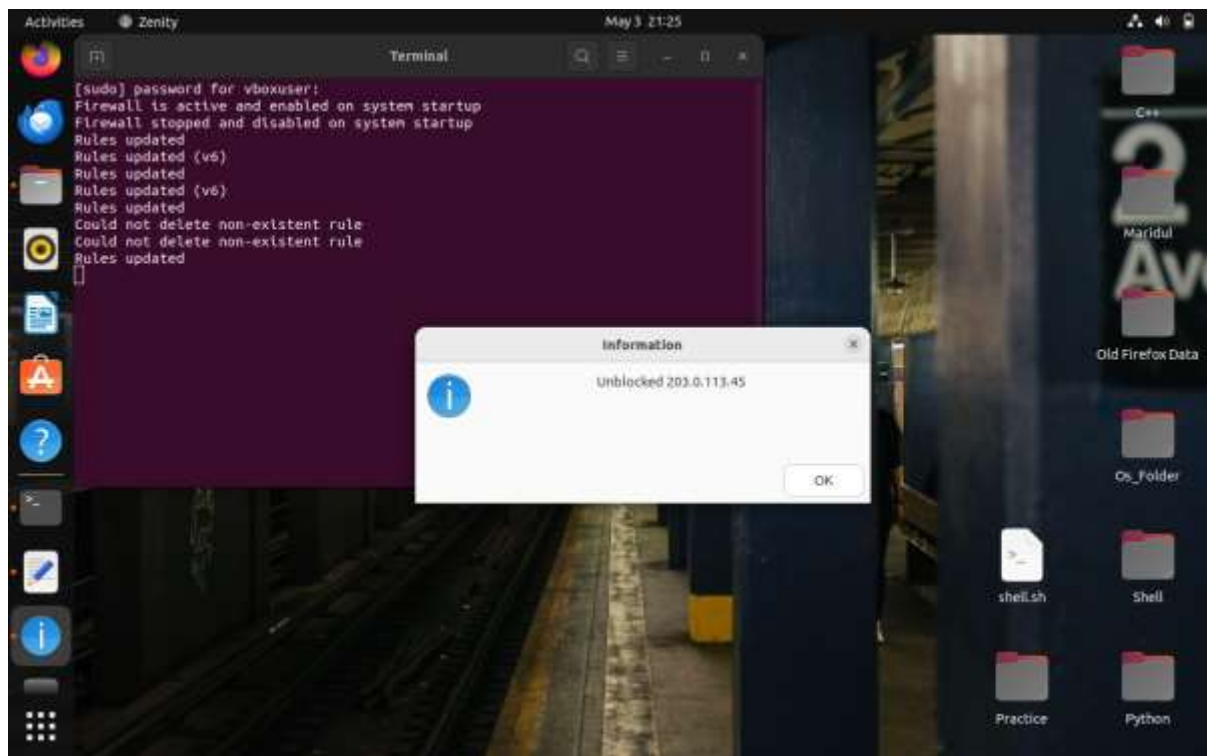




#### 4.4.2.1 Unblock IP Address



#### 4.4.2.2 Enter an IP Address



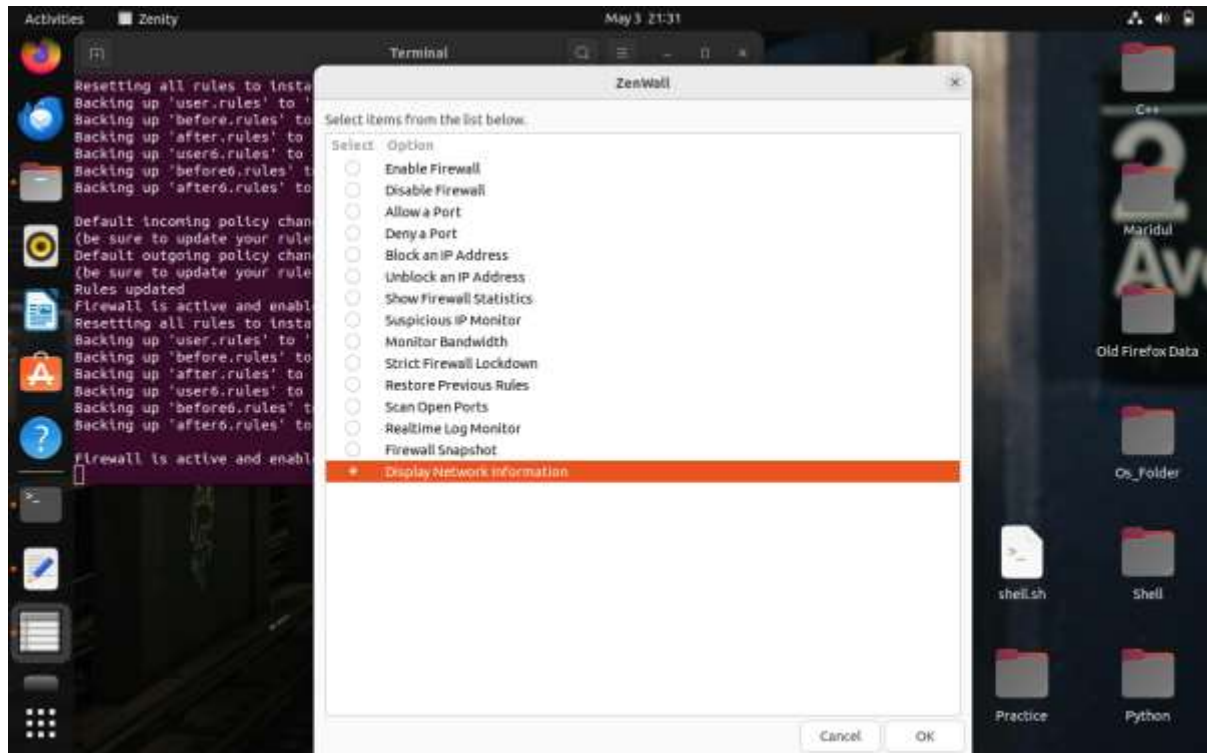
#### 4.4.2.3 Unblocked IP Address

Zenity captures the IP address input, checks its validity, and modifies the UFW rules accordingly. Confirmation dialogs inform the user of success or failure.

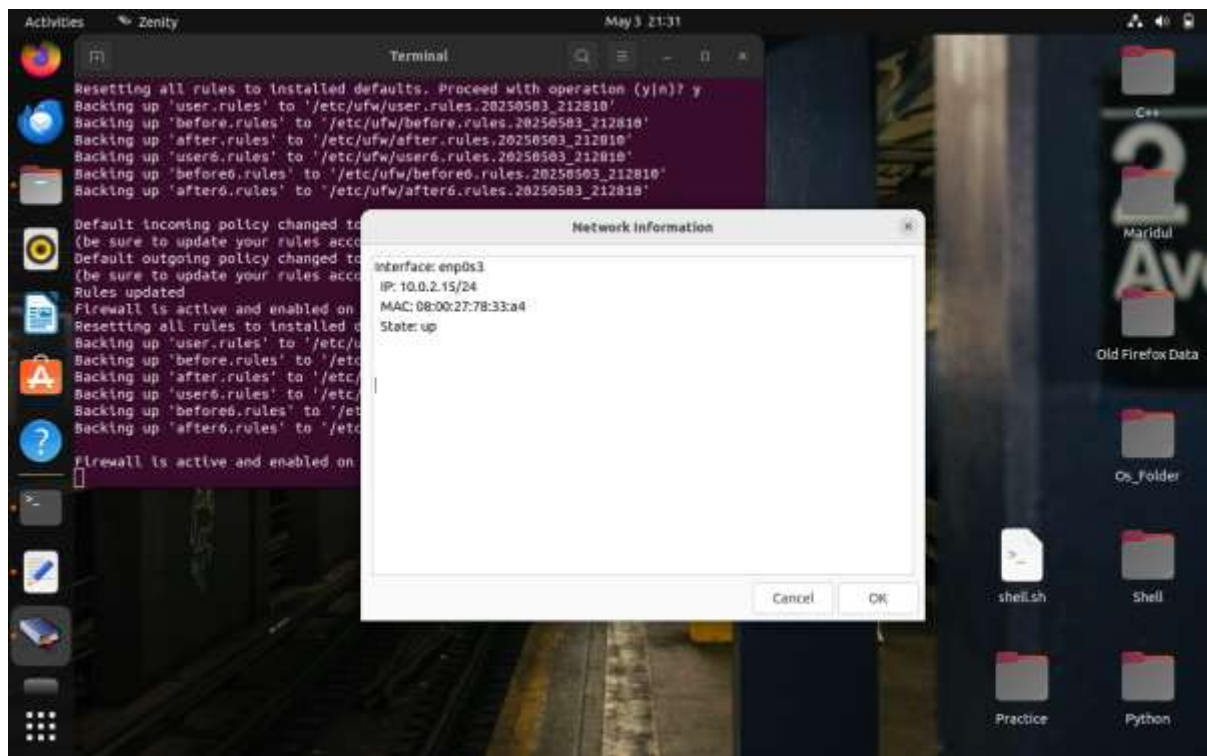


This module opens a gnome-terminal and runs iftop, a utility that visually displays bandwidth usage by IP and port in real-time.

## 4.6 Network Information Display



### 4.6.1 Display Network Information

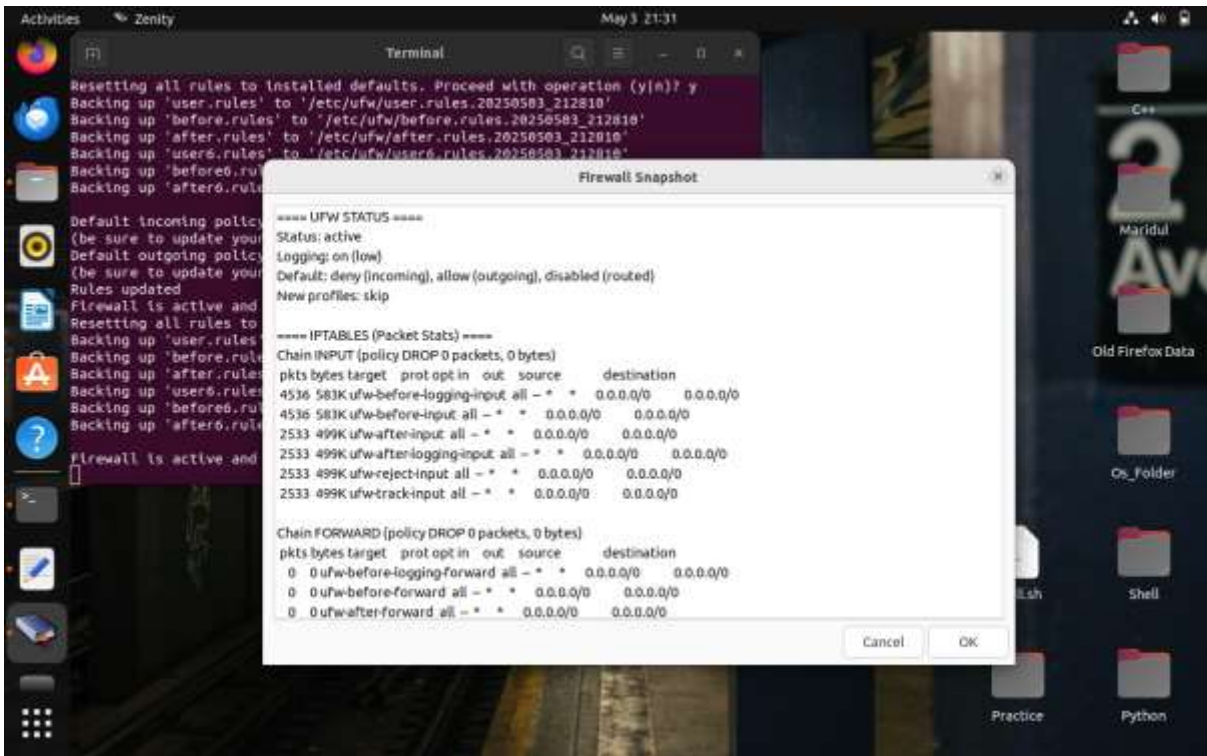


#### 4.6.2 Network Information Window

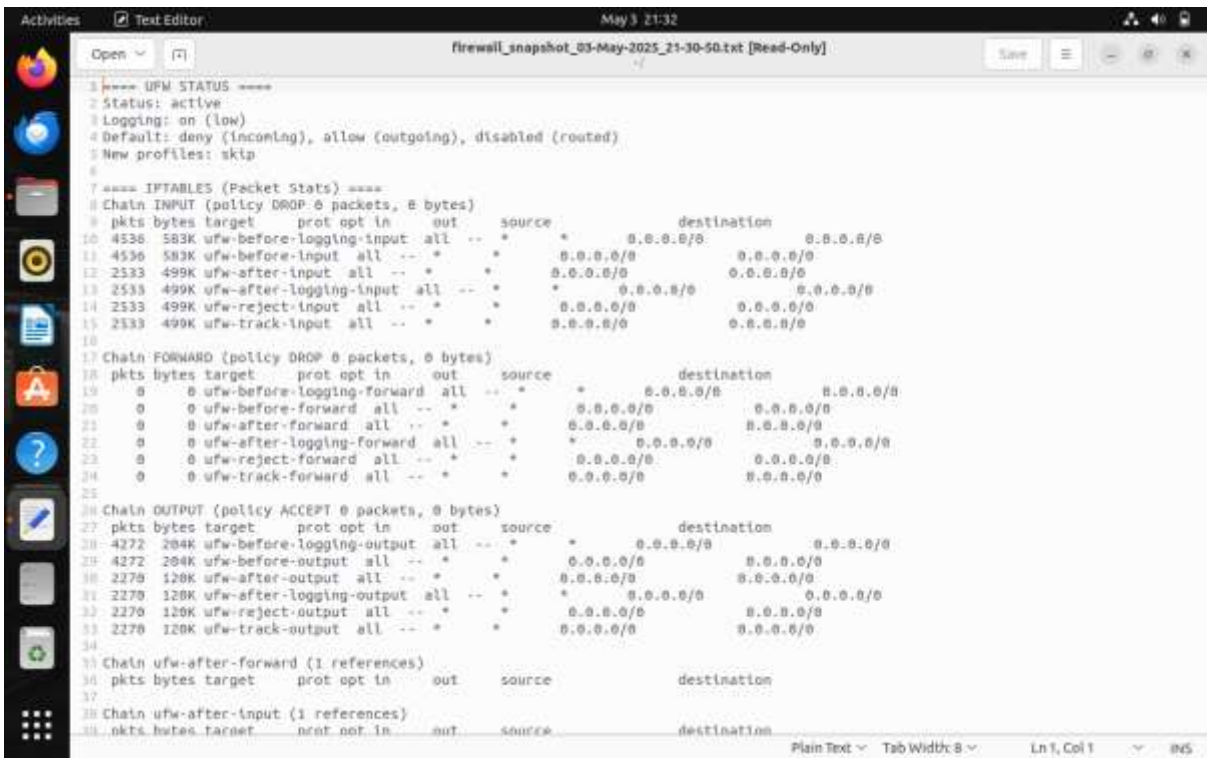
Displays system IP address, MAC address, default gateway, and active connections using `ip`, `ss`, and related commands. The result is shown in a scrollable Zenity text-info dialog.



## 4.7 Firewall Snapshot



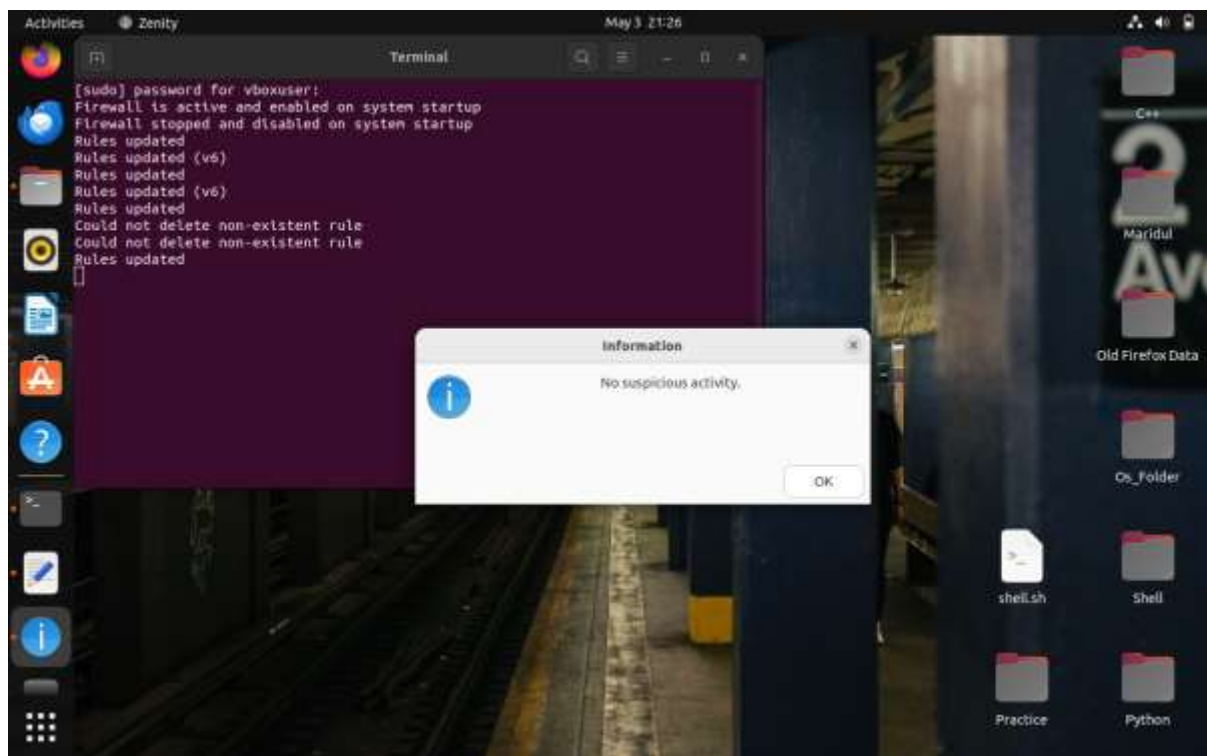
### 4.7.1 Firewall Snapshot Data



### 4.7.2 Firewall Snapshot File

When triggered, this module captures the current UFW status and rules, writing them to a snapshot file with a timestamp. These snapshots serve as historical references for restoring settings.

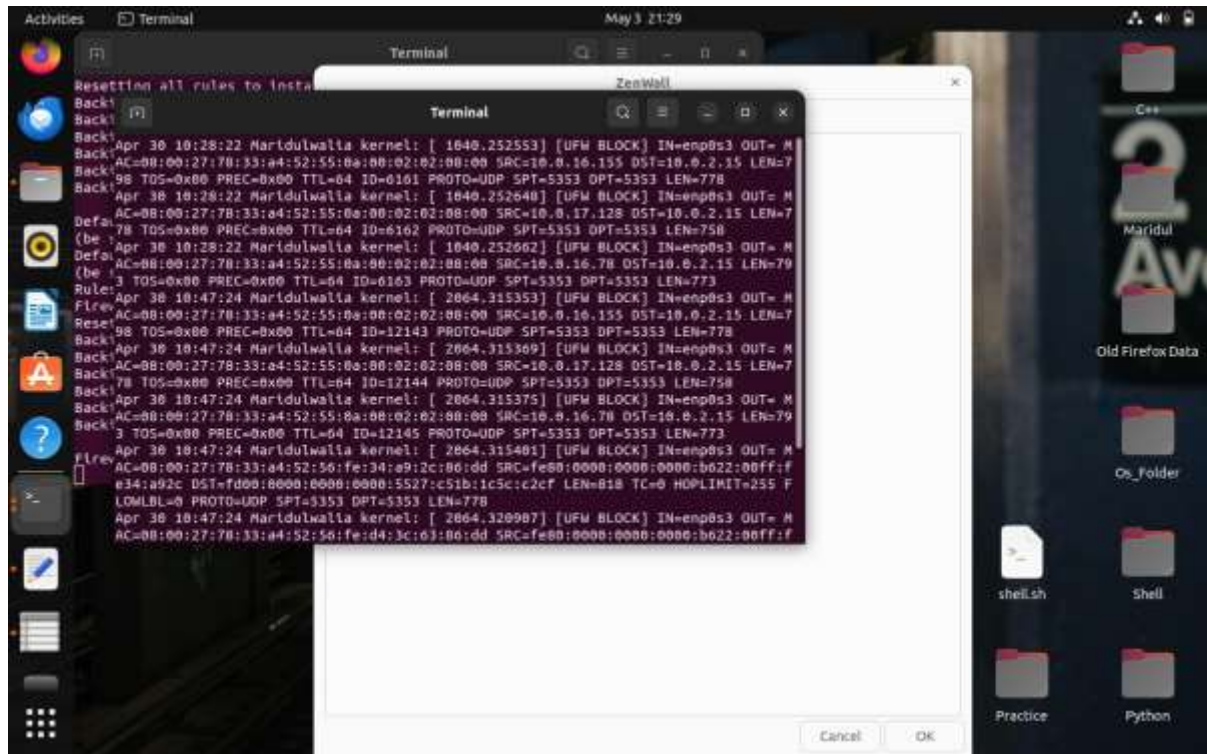
## 4.8 Suspicious IP Monitor



### 4.8 Suspicious Activity

Scans the network for open ports known to be vulnerable (e.g., 23, 3389, 5900) using nmap. Outputs a summary of potentially risky hosts for further analysis.

## 4.9 Real-time Log Monitoring

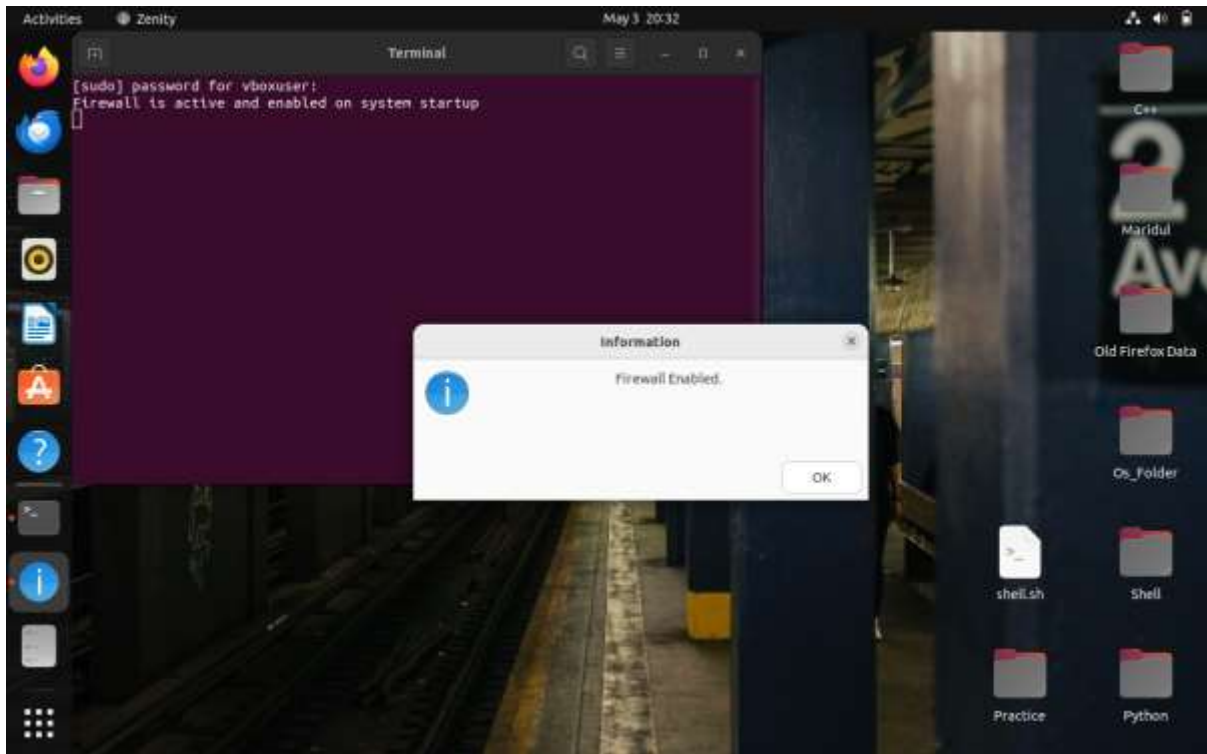


### 4.9 Real-time Log Monitor

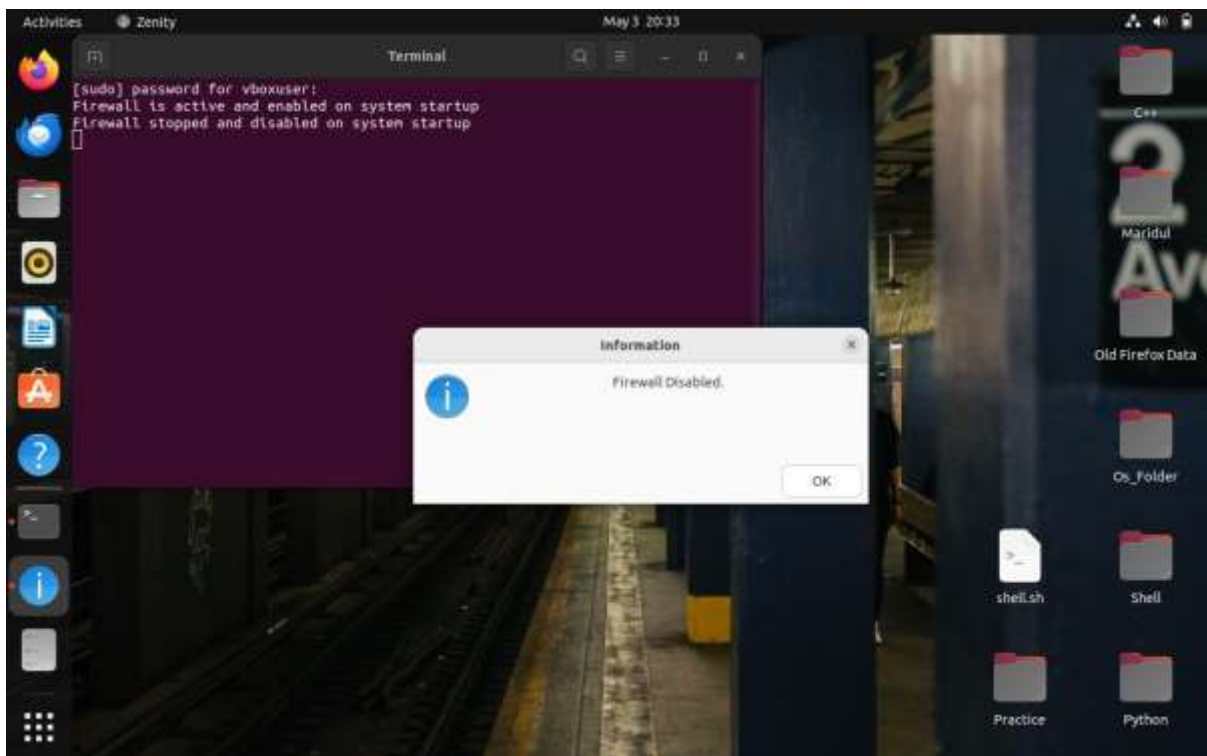
This feature will allow the user to view updates in real time using either Zenity progress info or a tailing terminal output.



## 4.10 Firewall Enable/Disable



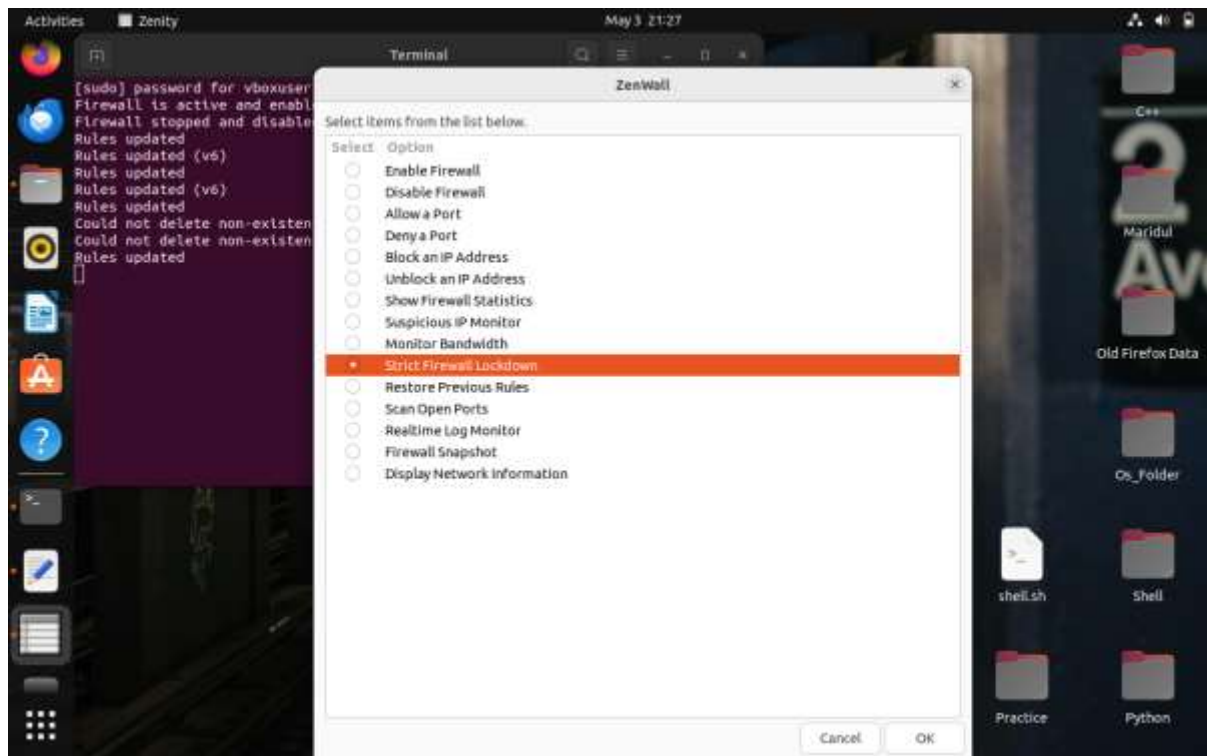
### 4.10.1 Firewall Enabled



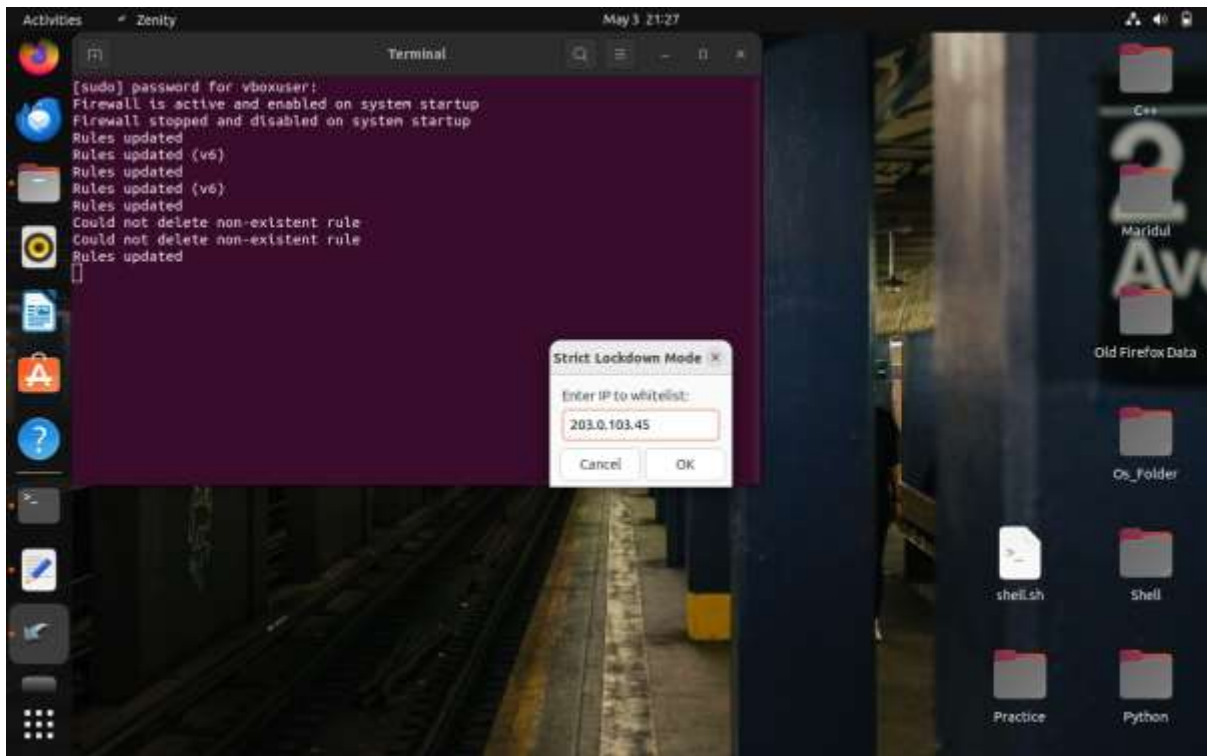
### 4.10.1 Firewall Disabled

When selected, the script prompts the user with a confirmation dialog. Based on the choice, it enables or disables the UFW firewall using appropriate commands. A Zenity information box confirms the action, and all changes are logged with timestamps.

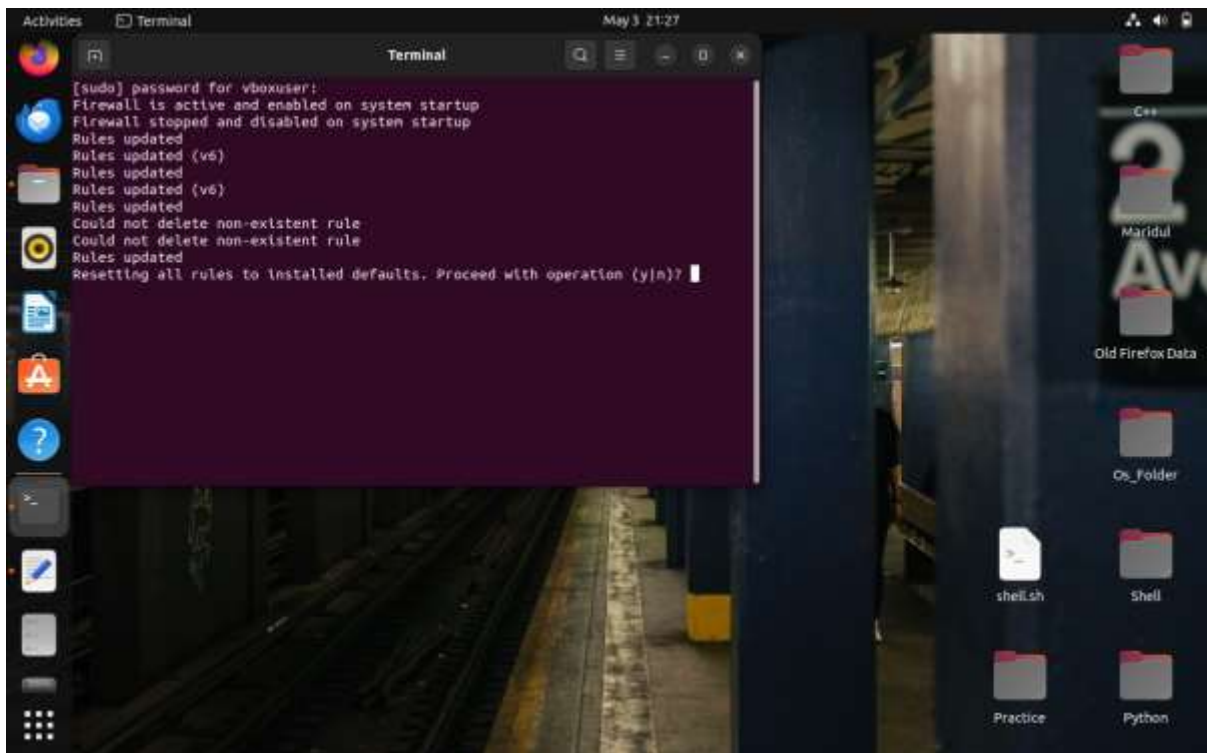
## 4.11 Strict Firewall Lockdown



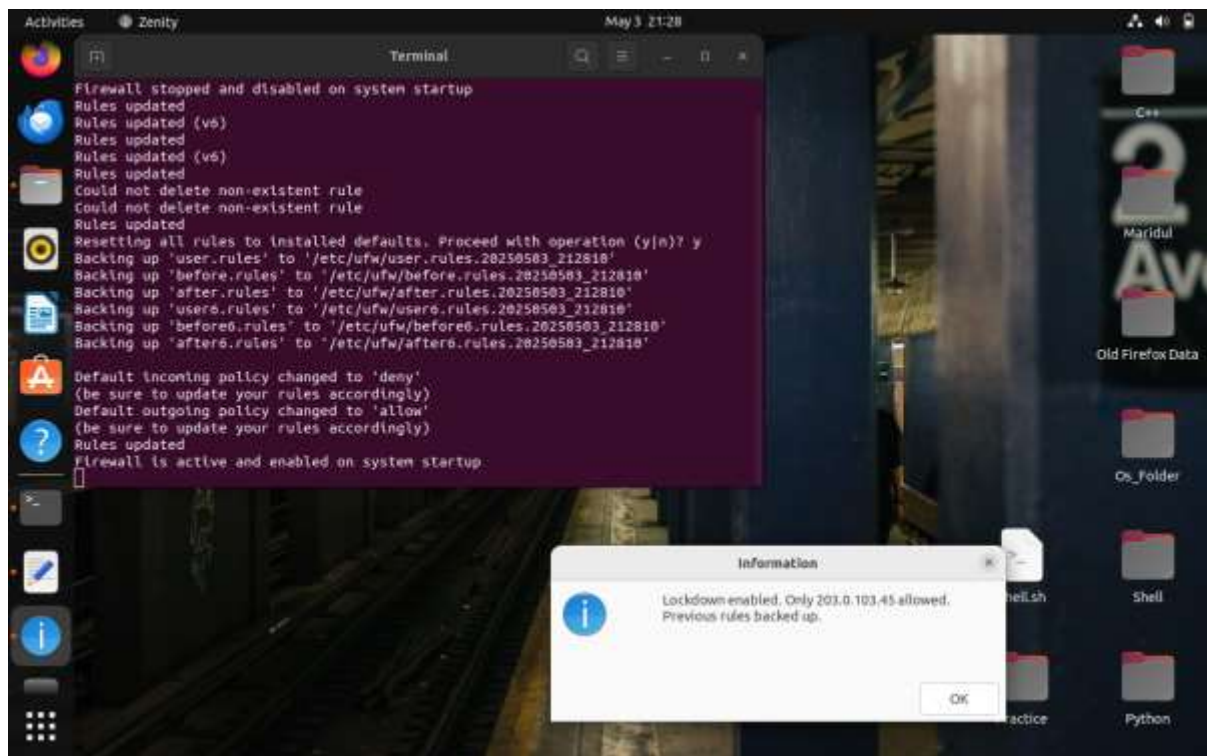
### 4.11.1 Strict Lockdown



4.11.1 Enter an IP to whitelist



4.11.3 Confirmation Window

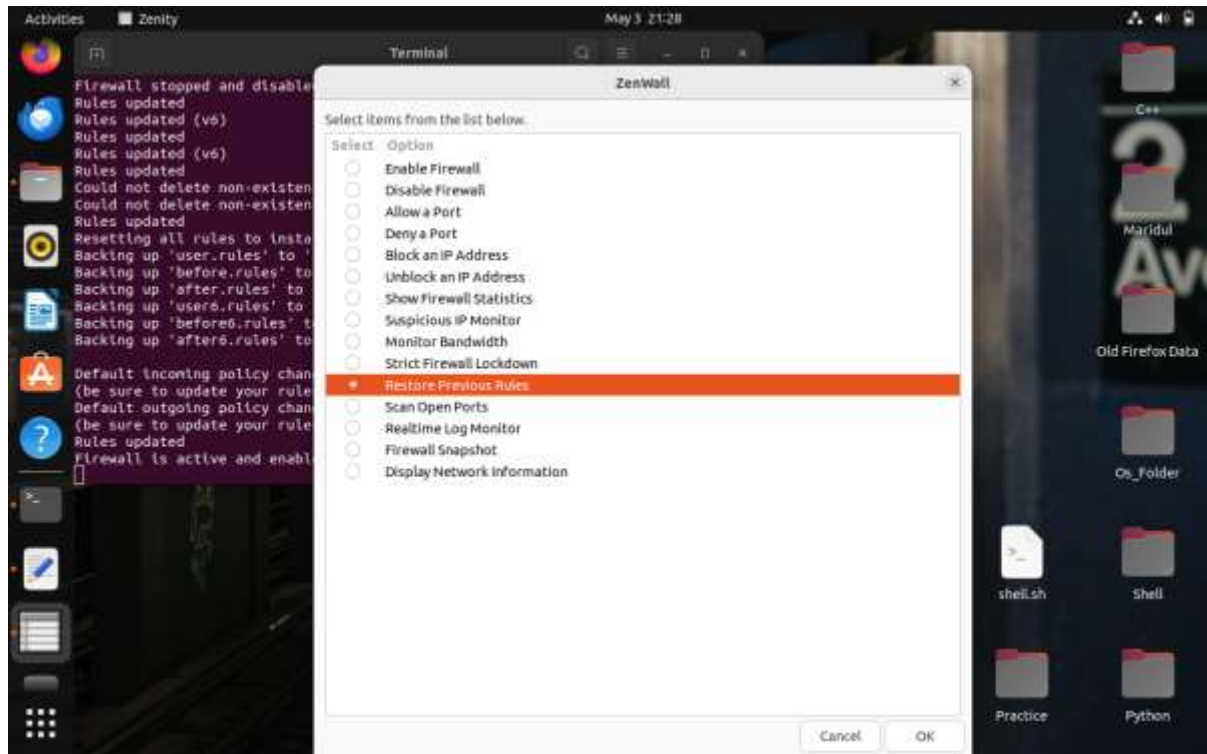


#### 4.11.4 Strict Lockdown Enabled

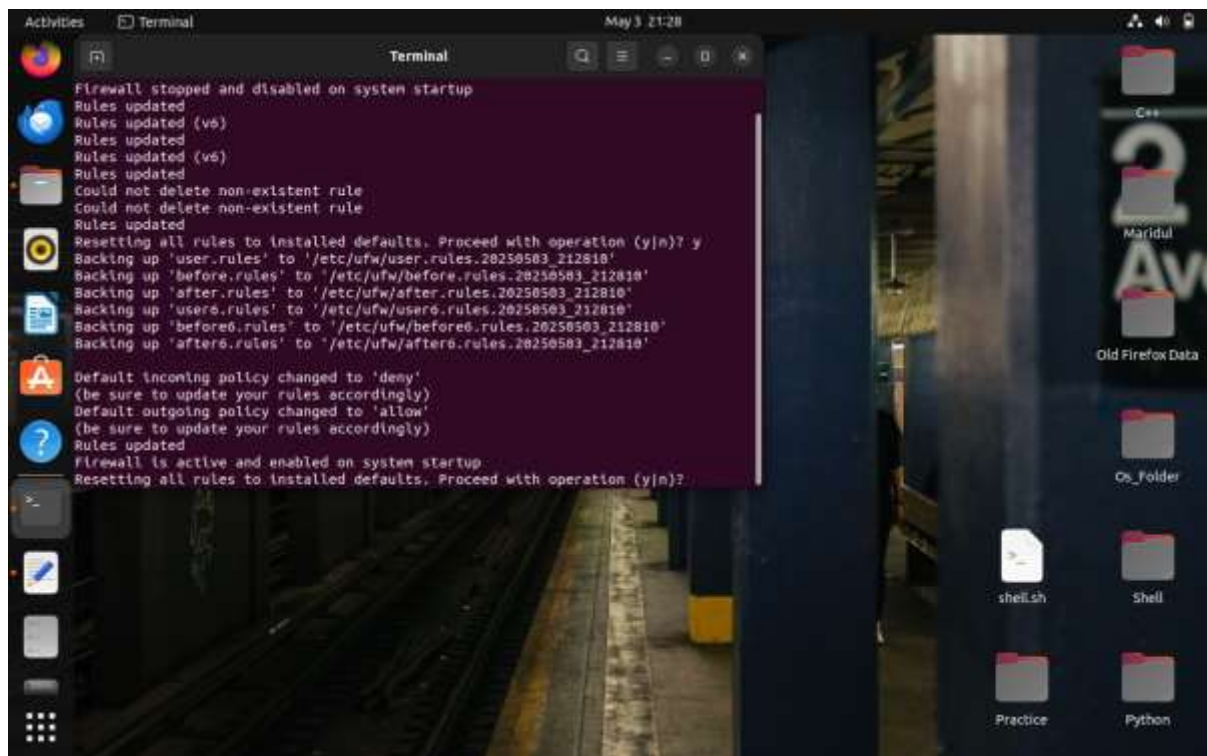
This feature allows users to initiate an emergency lockdown by blocking all incoming traffic except from one trusted IP address. The user is prompted to input the trusted IP, after which the current UFW rules are backed up. The lockdown is applied, and a confirmation dialog notifies the user. Backup files are created and stored for later restoration.



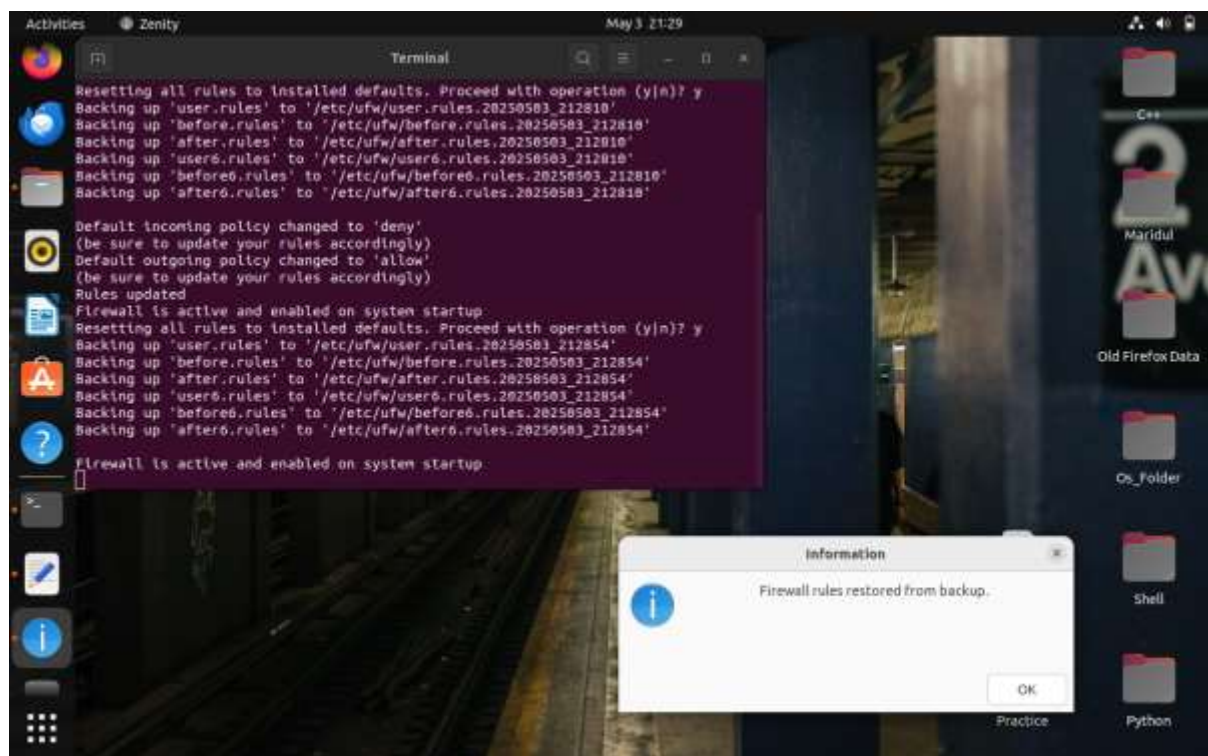
## 4.12 Restore Previous Rules



### 4.12.1 Restore Previous Rules Before Lockdown



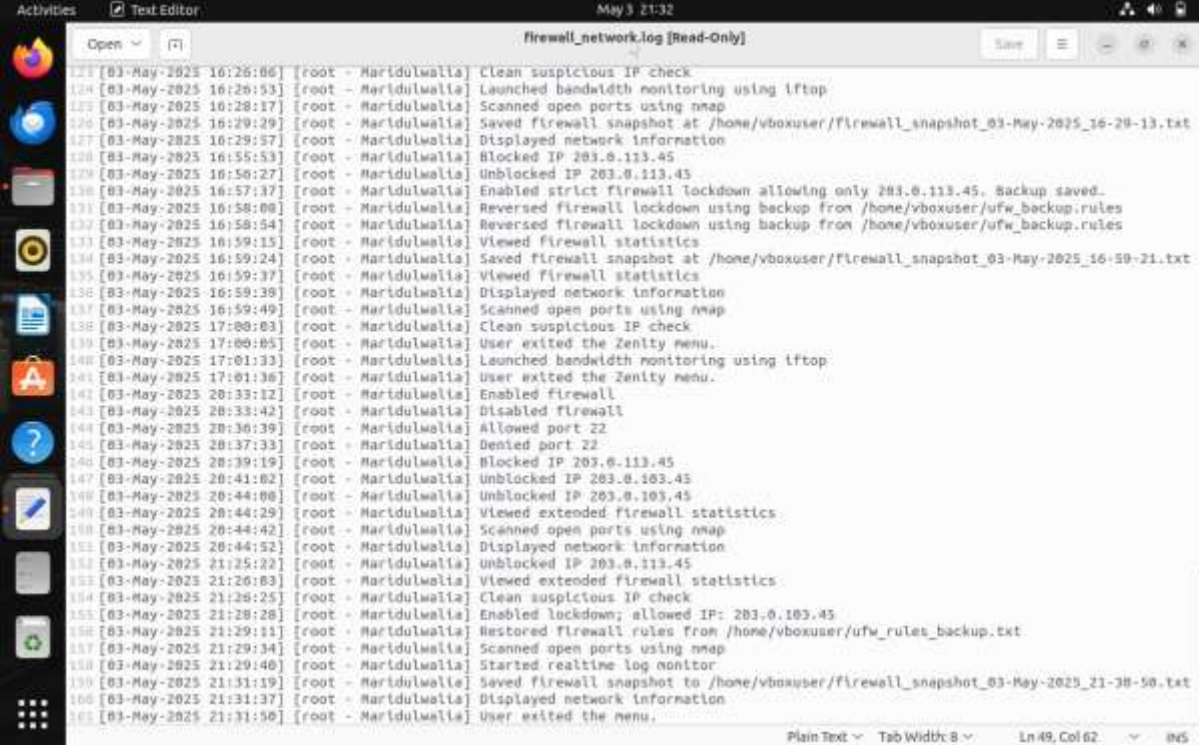
### 4.12.2 Confirmation window



#### 4.12.1 Previous Rules Before Lockdown Restored

Available only when a backup exists, this option lets users restore the firewall to its previous state before lockdown. Upon selection, the script re-applies the saved rules from the backup file. A dialog box confirms successful restoration, and the process is recorded in the log.

## 4.13 Logging



```
121 [03-May-2025 16:26:06] root - Maridulwalla Clean suspicious IP check
124 [03-May-2025 16:28:53] root - Maridulwalla Launched bandwidth monitoring using iftop
125 [03-May-2025 16:28:17] root - Maridulwalla Scanned open ports using nmap
126 [03-May-2025 16:29:29] root - Maridulwalla Saved firewall snapshot at /home/vboxuser/firewall_snapshot_03-May-2025_16-29-13.txt
127 [03-May-2025 16:29:57] root - Maridulwalla Displayed network information
128 [03-May-2025 16:55:53] root - Maridulwalla Blocked IP 203.0.113.45
129 [03-May-2025 16:56:27] root - Maridulwalla Unblocked IP 203.0.113.45
130 [03-May-2025 16:57:37] root - Maridulwalla Enabled strict firewall lockdown allowing only 203.0.113.45. Backup saved.
131 [03-May-2025 16:58:00] root - Maridulwalla Reversed firewall lockdown using backup from /home/vboxuser/ufw_backup.rules
132 [03-May-2025 16:58:54] root - Maridulwalla Reversed firewall lockdown using backup from /home/vboxuser/ufw_backup.rules
133 [03-May-2025 16:59:15] root - Maridulwalla Viewed Firewall statistics
134 [03-May-2025 16:59:24] root - Maridulwalla Saved firewall snapshot at /home/vboxuser/firewall_snapshot_03-May-2025_16-59-21.txt
135 [03-May-2025 16:59:37] root - Maridulwalla Viewed Firewall statistics
136 [03-May-2025 16:59:39] root - Maridulwalla Displayed network information
137 [03-May-2025 16:59:49] root - Maridulwalla Scanned open ports using nmap
138 [03-May-2025 17:00:03] root - Maridulwalla Clean suspicious IP check
139 [03-May-2025 17:00:05] root - Maridulwalla User exited the Zenity menu.
140 [03-May-2025 17:01:33] root - Maridulwalla Launched bandwidth monitoring using iftop
141 [03-May-2025 17:01:36] root - Maridulwalla User exited the Zenity menu.
142 [03-May-2025 20:33:12] root - Maridulwalla Enabled firewall
143 [03-May-2025 20:33:42] root - Maridulwalla Disabled firewall
144 [03-May-2025 20:36:39] root - Maridulwalla Allowed port 22
145 [03-May-2025 20:37:33] root - Maridulwalla Denied port 22
146 [03-May-2025 20:39:19] root - Maridulwalla Blocked IP 203.0.113.45
147 [03-May-2025 20:41:02] root - Maridulwalla Unblocked IP 203.0.103.45
148 [03-May-2025 20:44:00] root - Maridulwalla Unblocked IP 203.0.103.45
149 [03-May-2025 20:44:29] root - Maridulwalla Viewed extended firewall statistics
150 [03-May-2025 20:44:42] root - Maridulwalla Scanned open ports using nmap
151 [03-May-2025 20:44:52] root - Maridulwalla Displayed network information
152 [03-May-2025 21:25:22] root - Maridulwalla Unblocked IP 203.0.113.45
153 [03-May-2025 21:26:03] root - Maridulwalla Viewed extended firewall statistics
154 [03-May-2025 21:28:25] root - Maridulwalla Clean suspicious IP check
155 [03-May-2025 21:28:28] root - Maridulwalla Enabled lockdown; allowed IP: 203.0.103.45
156 [03-May-2025 21:29:11] root - Maridulwalla Restored firewall rules from /home/vboxuser/ufw_rules_backup.txt
157 [03-May-2025 21:29:34] root - Maridulwalla Scanned open ports using nmap
158 [03-May-2025 21:29:40] root - Maridulwalla Started realtime log monitor
159 [03-May-2025 21:31:19] root - Maridulwalla Saved firewall snapshot to /home/vboxuser/firewall_snapshot_03-May-2025_21-30-50.txt
160 [03-May-2025 21:31:37] root - Maridulwalla Displayed network information
161 [03-May-2025 21:31:50] root - Maridulwalla User exited the menu.
```

### 4.13.1 Firewall Log File

All critical user actions—such as enabling/disabling the firewall, applying lockdown, restoring rules, or modifying ports and IPs—are recorded in a centralized log file with timestamps and user information. The log file helps in auditing past actions and identifying changes in firewall behavior.

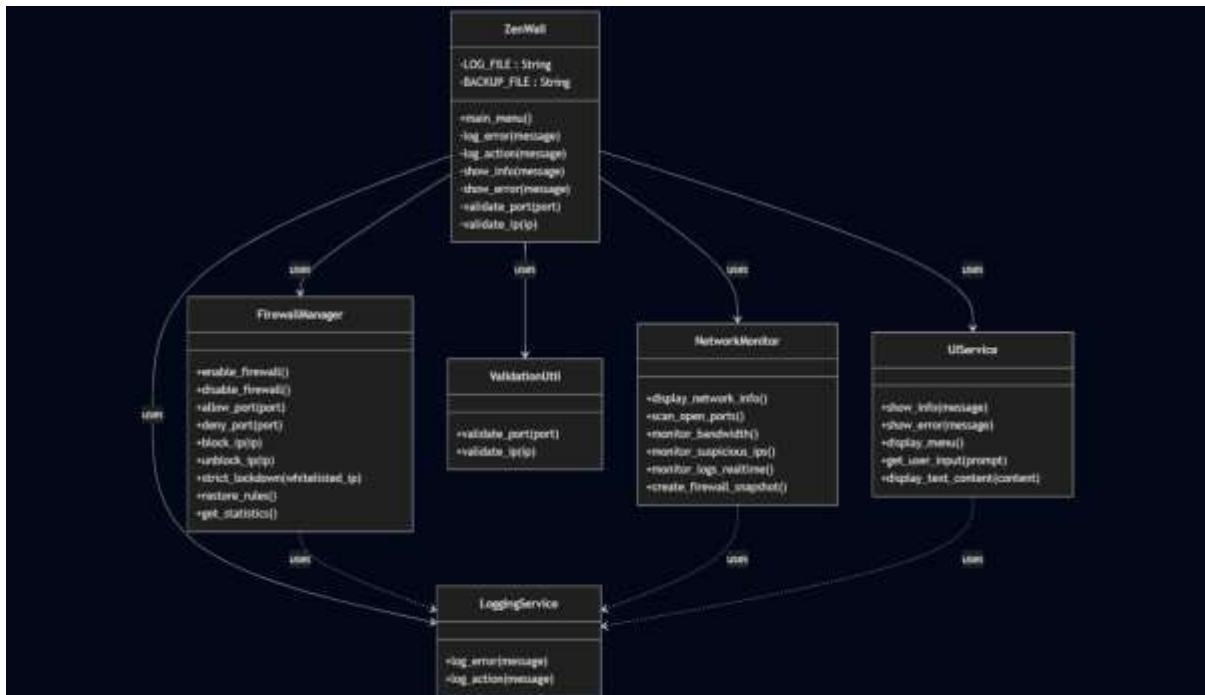
## 5. Testing and Validation

Test Case	Input	Expected Output	Actual Output
Allow Port	22	Port 22 allowed.	Port 22 allowed
Deny Port	23	Port 23 denied.	Port 23 denied.
Unblock IP Address	10.0.0.5	Unblocked 10.0.0.5.	Unblocked 10.0.0.5
Block IP Address	10.0.0.5	Blocked 10.0.0.5	Blocked 10.0.0.5
Show Firewall Statistics	N/A	Displays UFW and iptables stats	Displays UFW and iptables stats
Enable Firewall	N/A	Firewall Enabled.	Firewall Enabled.
Disable Firewall	N/A	Firewall Disabled.	Firewall Disabled.
Suspicious IP Monitor - Active	N/A	Suspicious activity found.	Suspicious activity found.
Suspicious IP Monitor - Inactive	N/A	No suspicious activity.	No suspicious activity.
Strict Firewall Lockdown	192.168.1.100	Lockdown enabled. Only 192.168.1.100 allowed.	Lockdown enabled. Only 192.168.1.100 allowed.
Restore Previous Rules	N/A	Firewall rules restored from backup.	Firewall rules restored from backup.
Scan Open Ports	N/A	Displays open ports on localhost	Displays open ports on localhost
Realtime Log Monitor	N/A	Opens UFW log in terminal	Opens UFW log in terminal

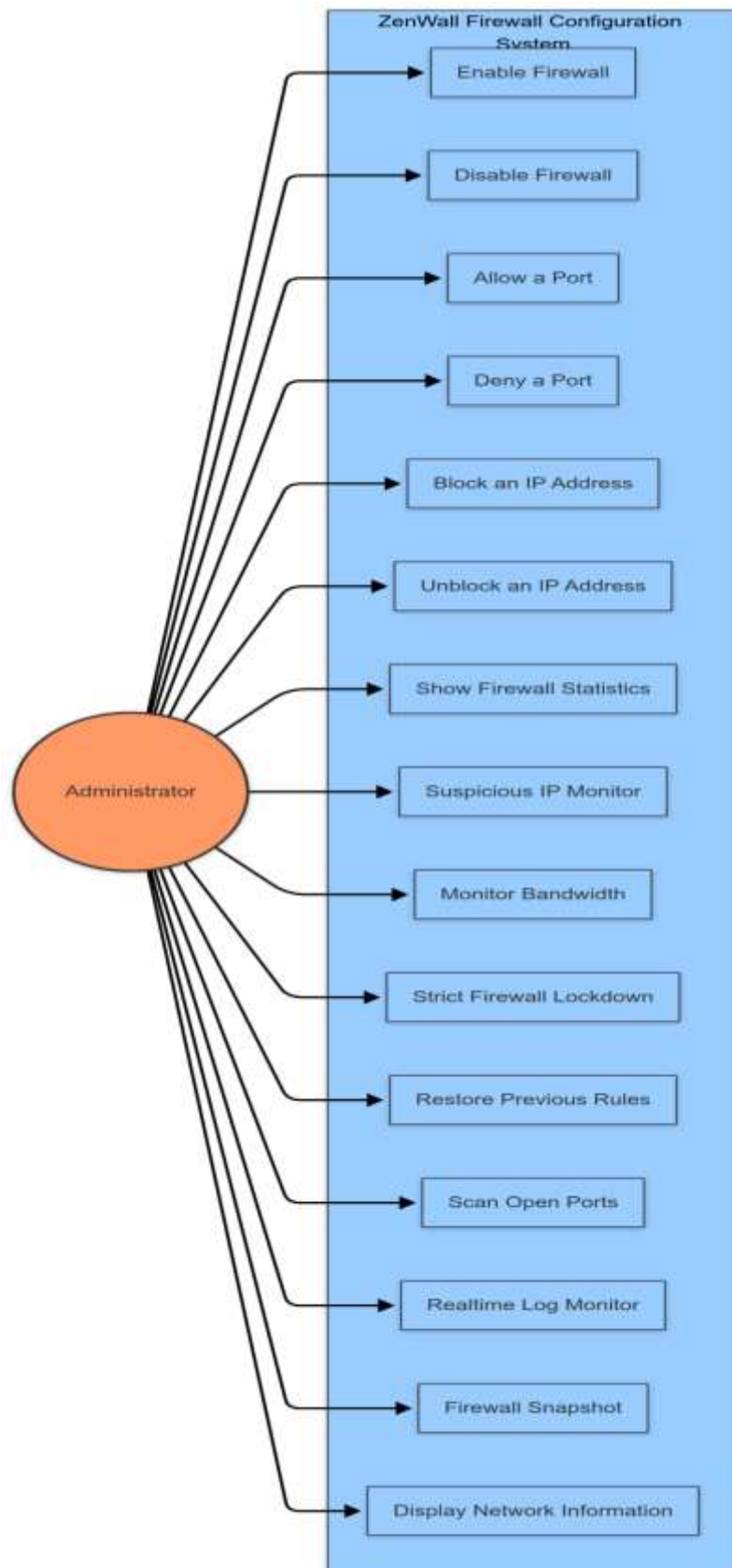


Firewall Snapshot	N/A	Saves and displays firewall snapshot	Saves and displays firewall snapshot
Display Network Information	N/A	Shows network interfaces and details	Shows network interfaces and details
Script Run Without Root Privileges	N/A	Error: Please run this script as root.	Error: Please run this script as root.
Missing Required Command (e.g., zenity)	N/A	Error: zenity is not installed. Please install it	Error: zenity is not installed. Please install it

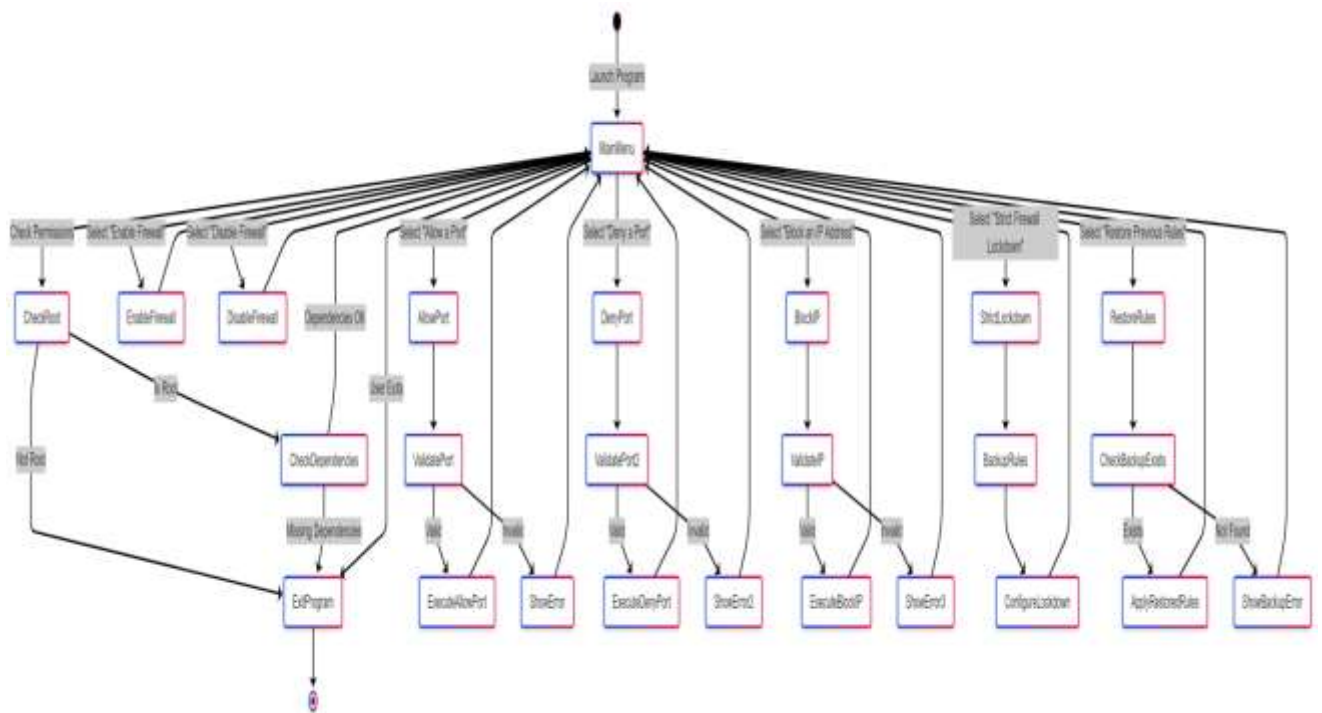
## 6. UML Diagrams



6.1 Class Diagram



6.2 Use Case Diagram



6.3 Activity Diagram

## 7. References

- Ubuntu Community Documentation – <https://help.ubuntu.com/community/UFW>
- man pages for Linux commands: ufw, iptables, zenity, iftop, nmap, ss, netstat, tail
- Stack Overflow Discussions – <https://stackoverflow.com>
- Ask Ubuntu Forums – <https://askubuntu.com>
- Linuxize Tutorials – <https://linuxize.com>