Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Факультет комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)
Кафедра кібербезпеки
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр
(освітній рівень)

на тему: " Огляд та практичне використання брандмауерів в операційній системі FreeBSD"

Виконав: студент (ка)
Спеціальності:
125 «Кібербезпека»
(шифр і назва напряму підготовки, спеціальності)
Пахода Владислав Юрійович
підпис (прізвище та ініціали)

Керівник Лечаченко Т. А.
підпис (прізвище та ініціали)

Нормоконтроль Лобур Т.Б.
підпис (прізвище та ініціали)

Завідувач кафедри Загородна Н.В.
підпис (прізвище та ініціали)

Рецензент
підпис (прізвище та ініціали)

м. Тернопіль – 2023

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра кібербезпеки
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Загородна Н.В.
(підпис) (прізвище та ініціали)
«__» _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 125 Кібербезпека
(шифр і назва спеціальності)

Студенту Паході Владиславу Юрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Огляд та практичне використання брандмауерів в операційній системі FreeBSD

Керівник роботи Лечаченко Тарас Анатолійович, PhD доктор філософії., асистент кафедри КБ
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 03 » 04 2023 року № 4/7-349

2. Термін подання студентом завершеної роботи 12.06.2023

3. Вихідні дані до роботи Вимоги до операційної системи FreeBSD

4. Зміст роботи (перелік питань, які потрібно розробити)
Проаналізувати типи брандмауерів та принципи їх роботи.
Проаналізувати брандмауери PF, IPFW і IPFilter для FreeBSD
Визначити переваги та недоліки PF, IPFW і IPFilter брандмауерів
Провести налаштування та тестування PF в FreeBSD
Повести налаштування PF для захисту від brute force та пом'якшення DDoS атак
Безпека життєдіяльності, основи охорони праці
Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
Тема, мета, задачі. Типи брандмауерів. Принципи роботи брандмауерів. Політики безпеки брандмауера. Призначення брандмауера. Огляд доступних брандмауерів для FreeBSD. Огляд брандмауера PF. Огляд брандмауера IPFW. Огляд брандмауера IPFilter. Переваги та недоліки PF брандмауера. Переваги та недоліки IPFW брандмауера. Переваги та недоліки IPFilter брандмауера. Налаштування FreeBSD як шлюзу з NAT та брандмауером. Параметри налаштування FreeBSD в конфігураційному файлі для виконання функцій

шлюзу/etc/rc.conf. Налаштування PF. Здійснення brute force атаки на SSH-сервер. Результати проведення brute force атаки на SSH-сервер без налаштувань захисту PF Фрагмент конфігурації PF для захисту від brute force. Результати проведення brute force атаки на SSH-сервер після налаштувань PF . Результати проведення brute force атаки на SSH-сервер після налаштувань PF . Висновки

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи хорони праці	Пилипець М.І., проф. кафедри МТ		

7. Дата видачі завдання 19.01.2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	20.01 – 24.01	Виконано
2.	Підбір джерел про основи роботи брандмауерів та їх типи в тому числі і для FreeBSD	25.01 – 30.01	Виконано
3.	Опрацювання джерел в галузі дослідження	01.02 – 20.02	Виконано
4.	Провести налаштування та тестування PF в FreeBSD	25.02 – 15.03	Виконано
5.	Повести налаштування PF для захисту від brute force та пом’якшення DDoS атак	16.03-05.04	Виконано
6.	Оформлення розділу «Теоретичні основи брандмауерів»	21.02 – 10.03	Виконано
7.	Оформлення розділу «Брандмауери в операційній системі FreeBSD»	11.03 – 25.03	Виконано
8.	Оформлення розділу «Налаштування та тестування PF в FreeBSD»	10.04 – 05.05	Виконано
9.	Виконання завдання до підрозділу «Безпека життєдіяльності, основи охорони праці»	10.05 – 21.05	Виконано
10.	Оформлення кваліфікаційної роботи	23.05 – 06.06	Виконано
11.	Нормоконтроль	06.06 – 10.06	Виконано
12.	Перевірка на плагіат	11.06 – 12.06	Виконано
13.	Попередній захист кваліфікаційної роботи	14.06 – 15.06	Виконано
14.	Захист кваліфікаційної роботи	23.06.2023	

Студент _____ Пахода В.Ю.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Лечаченко Т. А.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Огляд та практичне використання брандмауерів в операційній системі FreeBSD // Кваліфікаційна робота ОР «Бакалавр» // Пахода Владислав Юрійович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп’ютерно-інформаційних систем і програмної інженерії, кафедра кібербезпеки, група СБс-41 // Тернопіль, 2023 // С. 60 , рис. – 15, табл. – - , кресл. – 23 , додат. – .

Ключові слова: FREEBSD, PF, IPFW, IPFILTER, DDOS, BRUTE FORCE, NAT.

Дана кваліфікаційна робота присвячена дослідженню важливості та ролі брандмауерів у забезпеченні безпеки мережі та захисту комп'ютерних систем з використанням операційної системи FreeBSD.

В роботі ретельно досліджено ключові принципи функціонування брандмауерів, зокрема фільтрацію пакетів, контроль доступу та виявлення вторгнень. Запропоновано порівняльний аналіз різних брандмауерів, таких як IPFW, PF та IPFilter, що дозволило з'ясувати їх переваги та недоліки. Окрему увагу приділено операційній системі FreeBSD як платформі для встановлення та налаштування брандмауерів.

Робота розглядає різні сценарії використання брандмауерів у реальних умовах, зокрема налаштування фільтрації пакетів, налаштування правил контролю доступу та виявлення вторгнень. Проаналізовано використання брандмауерів для боротьби з атаками грубої сили (brute force) та пом'якшенням DDoS-атак, а також використання динамічних правил для реагування на зміни трафіку в реальному часі.

Особлива увага приділена брандмауеру PF, який є потужним інструментом мережевої безпеки у системі FreeBSD.

Результати дослідження свідчать про успішну реалізацію поставлених завдань та досягнення поставленої мети. Матеріали, представлені у роботі, можуть бути використані адміністраторами мереж, системними інженерами та студентами для поліпшення методів захисту.

ANNOTATION

Overview and practical use of firewalls in the FreeBSD operating system // Thesis of educational level "Bachelor"// Vladyslav Pakhoda // Ternopil Ivan Puluj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Cybersecurity, group СБс-41 // Ternopil, 2023 // P. 60 fig. - 15, tab. - __, chair. 21 , added. – -.

Keywords: FREEBSD, PF, IPFW, IPFILTER, DDOS, BRUTE FORCE, NAT.

The thesis is devoted to the study of the importance and role of firewalls in network security and protection of computer systems using the FreeBSD operating system. In today's world where network security is important, firewalls are a necessary piece of infrastructure to protect network resources from malicious attacks and unauthorized access.

The paper carefully examines the key principles of firewall operation, including packet filtering, access control, and intrusion detection. A comparative analysis of various firewalls such as IPFW, PF and IPFilter is proposed, which made it possible to find out their advantages and disadvantages. Particular attention is paid to the FreeBSD operating system as a platform for installing and configuring firewalls.

The work studies considers various scenarios for using firewalls in real conditions, including packet filtering settings, access control rules and intrusion detection settings. We analyzed the use of firewalls to combat brute force attacks and mitigate DDoS attacks, as well as the use of dynamic rules to respond to real-time traffic changes.

Particular attention is paid to the PF firewall, which is a powerful network security tool in a FreeBSD system.

The results of the study testify to the successful implementation of the tasks set and the achievement of the set goals. The materials presented in the work can be used by network administrators, system engineers and students to improve protection methods.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

NGFW	—	Next-Generation Firewalls
VPN	—	Virtual Private Network
IPFW	—	Internet Protocol Firewall
PF	—	Packet Filter
RDR	—	Redirect
UDP	—	User Datagram Protocol
DoS	—	Denial of Service
DDoS	—	Distributed Denial of Service
LAN	—	Local Area Network
WAN	—	Wide Area Network
SSH	—	Secure Shell
TCP	—	Transmission Control Protocol
HTTP	—	Hypertext Transfer Protocol Secure
PAM	—	Pluggable Authentication Modules

ВСТУП

Забезпечення безпеки мережі та захисту комп'ютерних систем стали надзвичайно актуальними завданнями в сучасному цифровому світі. З ростом загроз інформаційній безпеці стає все більш необхідною ефективна оборона, яка забезпечує надійний захист від потенційних загроз. Одним з найважливіших елементів безпеки мережі є використання брандмауерів.

Брандмауери є важливими компонентами мережевої безпеки, які дозволяють контролювати та фільтрувати трафік мережі, регулювати доступ до ресурсів та захищати систему від небажаного вторгнення. Одною з операційних систем, які підтримують широкий спектр брандмауерів, є FreeBSD.

Огляд та практичне використання брандмауерів у операційній системі FreeBSD є актуальною темою, оскільки FreeBSD визнана як надійна, гнучка та безпечна операційна система зі значними можливостями мережевої конфігурації.

Ця кваліфікаційна робота має на меті провести огляд та дослідження брандмауерів в операційній системі FreeBSD. Буде проведено детальний аналіз функцій, можливостей та особливостей різних брандмауерів, що доступні у FreeBSD, зокрема IPFW, PF та IPFilter. Дослідження також охопить практичне використання брандмауерів у реальних умовах, налаштування та конфігурацію, а також оцінку їх ефективності у захисті мережі.

У наступних розділах буде розглянуто безпеку мережі та роль брандмауерів у захисті комп'ютерних систем. Досліджено загальні принципи функціонування брандмауерів, такі як фільтрація пакетів, контроль доступу та виявлення вторгнень. Розглянуто процес встановлення та налаштування брандмауера. Також проведено дослідження різних сценаріїв використання брандмауерів у реальних умовах, таких як налаштування фільтрації пакетів, налаштування правил контролю доступу та виявлення спроб атак.

Результати цього дослідження можуть бути корисними для адміністраторів мереж, системних інженерів та в навчальних цілях.

Подальше дослідження у цій області може сприяти розвитку мережевої безпеки, поліпшенню методів захисту та забезпеченню безпеки мережевих інфраструктур. Завдяки цьому дослідженню буде досягнуто кращого розуміння брандмауерів у контексті FreeBSD та їх впливу на загальну безпеку мереж.

1 ТЕОРЕТИЧНІ ОСНОВИ БРАНДМАУЕРІВ

Безпека мережі є надзвичайно важливою аспектом сучасних комп'ютерних систем. У світі, де мережеві загрози та кібератаки постійно зростають, необхідно мати ефективні заходи безпеки для захисту важливих даних і забезпечення надійності мережевих інфраструктур. Одним з ключових інструментів безпеки мережі є брандмауери.

Брандмауер є програмним або апаратним засобом, який контролює трафік мережі, фільтрує пакети та приймає рішення щодо пересилання або блокування цього трафіку відповідно до заданих правил. Основна функція брандмауера полягає в захисті мережі від несанкціонованого доступу, зовнішніх загроз та зловмисних атак.

1.1 Типи брандмауерів

Існує кілька типів брандмауерів.

Packet-filtering Firewalls (Фільтруючі брандмауери)*:* Це найпростіший тип брандмауера, який використовує задані правила для перевірки кожного пакета, що проходить через нього. Ці правила базуються на IP-адресах відправника та одержувача, номерах портів, типах протоколів, тощо. Він працює на мережевому (3-му) рівні моделі OSI, де він аналізує пакети даних без врахування контексту, у якому вони використовуються. Цей тип брандмауера не може відстежувати стан підключення і відповідно він менш ефективний проти деяких типів атак [1].

Робота фільтруючих брандмауерів базується на правилах, які визначають, які пакети даних мають бути пропущені або заблоковані. Ці правила можуть використовувати різні параметри, такі як IP-адреси, порти, протоколи, типи пакетів і т.д.

Фільтруючі брандмауери аналізують заголовки пакетів даних, щоб визначити, чи відповідають вони правилам фільтрації. Якщо пакет відповідає правилам, його можна пропустити через брандмауер. У протилежному випадку,

якщо пакет не відповідає правилам або відповідає правилам блокування, він може бути відхилений або заблокований.

Основні переваги фільтруючих брандмауерів включають:

- Захист мережі. Вони дозволяють блокувати небажаний або шкідливий трафік, такий як атаки, віруси, спам або несанкціонований доступ.
- Керування доступом. Фільтруючі брандмауери можуть встановлювати правила для обмеження доступу до ресурсів мережі, контролювати певні види трафіку або встановлювати політику безпеки.
- Простота конфігурації. Вони зазвичай мають простий інтерфейс конфігурації, що дозволяє легко встановлювати та змінювати правила фільтрації.
- Висока продуктивність. Фільтруючі брандмауери здатні швидко обробляти пакети даних і забезпечувати високу продуктивність мережі.

Stateful Inspection Firewalls (Брандмауери з перевіркою стану)*:* Цей тип брандмауера є більш розвинутим і може відслідковувати стан підключення. Вони аналізують пакети не лише на основі IP-адреси, порту і типу протоколу, але і в контексті сесії або підключення. Це означає, що вони можуть розуміти, чи є конкретний пакет частиною вже існуючого з'єднання або нового з'єднання, що дозволяє більш точно контролювати трафік і запобігати атакам [1].

Також це означає, що брандмауери з перевіркою стану ведуть журнал проходження пакетів даних через мережу і відстежують стан кожного з'єднання. Вони зберігають інформацію про вже встановлені з'єднання та їх стани, такі як відкриті порти, статус пакетів SYN, ACK і т.д.

При отриманні нового пакета даних брандмауер перевіряє, чи відповідає він стану вже існуючого з'єднання. Якщо пакет відповідає наявному стану з'єднання, то він пропускається. В протилежному випадку, якщо пакет не відповідає стану або не пройшов інші правила фільтрації, брандмауер блокує або відкидає цей пакет.

Переваги брандмауерів з перевіркою стану включають:

- Вищий рівень безпеки. Вони забезпечують більш глибокий аналіз трафіку, враховуючи стан з'єднання, що допомагає виявити та блокувати складніші атаки, такі як атаки з підробленими пакетами.
- Забезпечення цілісності з'єднань. Брандмауери з перевіркою стану відстежують стан кожного з'єднання і переконуються, що воно залишається цілим і активним протягом сеансу зв'язку.
- Ефективне управління ресурсами. Через використання інформації про стан з'єднань, брандмауери можуть оптимізувати роботу та керувати ресурсами мережі, наприклад, швидше обробляти дозволені пакети та зменшувати непотрібний трафік.

Proxy Firewalls (Проксі-брандмауери): Це тип брандмауерів, які виконують функції проксі-серверів для контролю трафіку між внутрішньою мережею і зовнішнім середовищем, таким як Інтернет. Вони працюють на рівні застосунку (так званий аплікаційний проксі) і здатні аналізувати, фільтрувати та переадресовувати мережевий трафік.

Проксі-брандмауери використовуються для забезпечення безпеки та контролю доступу до Інтернету. Вони діють як посередники між внутрішньою мережею і зовнішнім середовищем, приховуючи справжні IP-адреси та ідентифікатори комп'ютерів внутрішньої мережі. Це дозволяє забезпечити приватність та анонімність внутрішніх систем, а також ускладнює нападам зовнішніх загроз на мережу.

Основні переваги проксі-брандмауерів включають наступне:

- Фільтрація трафіку. Проксі-брандмауери можуть використовувати правила фільтрації, щоб блокувати небажаний трафік, такий як шкідливі або небезпечні веб-сайти, віруси або спам.
- Контроль доступу: Вони дозволяють управляти доступом до ресурсів Інтернету, обмежувати певні види контенту або встановлювати правила для користувачів внутрішньої мережі.
- Кешування: Проксі-брандмауери можуть кешувати часто використовувані веб-сторінки або файли, що дозволяє прискорити доступ до них і зменшити використання пропускної здатності Інтернету.

- Захист від зовнішніх загроз: Вони допомагають захистити внутрішню мережу від зловмисників, що намагаються отримати несанкціонований доступ або провести атаки ззовні.

Next-Generation Firewalls (андмауери нового покоління): Це найбільш сучасний тип брандмауерів, який включає в себе додаткові функції, такі як захист від вторгнень (IPS), ідентифікацію користувачів, захист від вірусів і т.д. NGFW комбінують функції традиційних брандмауерів з більш сучасними функціями для надання більш глибокого рівня захисту [1].

NGFW включають в себе основні функції фільтрації пакетів для блокування небажаного трафіку, але також надають додаткові можливості, такі як:

- Глибока інспекція пакетів (Deep Packet Inspection)/ NGFW можуть аналізувати вміст пакетів на рівні застосунку, що дозволяє виявляти та блокувати складніші загрози, включаючи шкідливе програмне забезпечення, вразливості та атаки на рівні застосунку.
- Управління доступом на основі користувачів (User-Based Access Control). NGFW можуть ідентифікувати та контролювати доступ користувачів до ресурсів мережі на основі їхніх ідентифікаторів, наприклад, логінів або ролях.
- Інтеграція з системами захисту від загроз (Threat Intelligence Integration). NGFW можуть інтегруватись з системами розвідки загроз та інтелектуального аналізу, що дозволяє отримувати оновлену інформацію про потенційні загрози та використовувати її для виявлення та блокування атак.
- VPN та безпека мережевих з'єднань. NGFW можуть підтримувати встановлення зашифрованих віртуальних приватних мереж, що забезпечує безпеку комунікації між віддаленими розташуваннями або користувачами, що працюють з-за меж мережі.
- Інтеграція з системами керування загрозами (Threat Management Integration). NGFW можуть поєднуватись з системами керування загрозами (Threat Management Systems) для автоматизованої інтеграції, обміну інформацією та координації заходів забезпечення безпеки.

1.2 Принципи роботи та політики безпеки брандмауера

Принципи роботи брандмауерів базуються на контролі мережевого трафіку і застосуванні правил, які визначають, який трафік дозволено або заборонено пропускати через брандмауер. Основні принципи роботи брандмауерів включають такі елементи [2]:

- Фільтрація пакетів. Брандмауери фільтрують мережевий трафік на основі правил, встановлених адміністратором. Вони аналізують заголовки пакетів, включаючи інформацію про джерело, призначення, порти, протоколи та інші параметри, і вирішують, чи дозволити чи заборонити прохід такого пакета відповідно до цих правил.
- Зонування. Брандмауери можуть розділяти мережу на зону довіри (наприклад, внутрішню мережу) і небезпечну зону (наприклад, зовнішній Інтернет). Також можливе створення інших тип зон. Правила політики безпеки встановлюються залежно від цієї роздільної лінії, і мета полягає в захисті зони довіри від потенційних загроз з інших зон.
- Правила політики безпеки. Брандмауери працюють на основі набору правил, які визначають дозволений або заборонений трафік. Ці правила встановлюються адміністратором і включають в себе інформацію про джерело, призначення, порти, протоколи, а також можуть враховувати контекстну інформацію про стан з'єднання.
- Перевірка стану з'єднання (Stateful Inspection). Деякі брандмауери, відомі як брандмауери з перевіркою стану, відстежують стан активних з'єднань і використовують цю інформацію для розпізнавання легітимного трафіку. Вони спостерігають за сесіями, що протікають через брандмауер, і дозволяють лише трафік, який входить в межі легітимного з'єднання.
- Network Address Translation. Багато брандмауерів підтримують функцію трансляції мережевих адрес (NAT), яка дозволяє змінювати IP-адреси і порти пакетів при проходженні через брандмауер. NAT може

використовуватися для захисту внутрішньої мережі, приховування справжніх IP-адрес і підтримки більшої кількості пристроїв в мережі.

- VPN і шифрування. Багато брандмауерів підтримують віртуальні приватні мережі (VPN) і можуть шифрувати трафік між віддаленими мережами або користувачами. Вони забезпечують безпечний тунель для передачі даних через незахищені мережі, що гарантує конфіденційність та цілісність інформації.
- Журналювання і моніторинг. Брандмауери можуть вести журнали подій, які містять інформацію про заборонені спроби доступу, атаки, недостовірний трафік тощо. Ці журнали можуть бути використані для аналізу безпеки, виявлення аномальних активностей та вдосконалення політик безпеки.

Політика безпеки брандмауера визначає, як брандмауер повинен реагувати на різний трафік. Зазвичай існують дві основні політики: "заборонити всі, дозволити деякі" та "дозволити всі, заборонити деякі".

Основна різниця між ними полягає у підході до керування трафіком.

Політика "заборонити всі, дозволити деякі" (deny-all, allow-some). Ця політика передбачає блокування всього трафіку за замовчуванням і дозволяє пропускати тільки той трафік, який ви явно визначили як дозволений. Це найбезпечніший підхід, оскільки весь трафік буде блокуватись, якщо ви не встановите правила для його дозволу. Ця політика вимагає детального визначення правил для дозволу необхідного трафіку, але забезпечує більший контроль і обмежує можливі загрози.

Політика "дозволити всі, заборонити деякі" (allow-all, deny-some). Ця політика, навпаки, дозволяє весь трафік пропускати через брандмауер за замовчуванням, але блокує певні види трафіку, визначені як небезпечні або небажані. Зазвичай ціллю цієї політики є забезпечення зручності в роботі мережі, але при цьому можуть бути деякі ризики, оскільки недозволений трафік може пройти через брандмауер, якщо не встановлено відповідні правила блокування.

1.3 Застосування брандмауера

Брандмауери використовуються в різних контекстах.

Застосування брандмауерів можуть включати такі сценарії:

- Захист внутрішніх мереж. Брандмауери встановлюються між внутрішньою мережею і зовнішніми мережами, такими як Інтернет. Вони контролюють вхідний і вихідний трафік, блокують небажаний трафік і захищають внутрішню мережу від зловмисників, шкідливих програм і атак.
- Захист серверів. Брандмауери можуть бути встановлені на серверах, що містять цінну інформацію або надають важливі послуги. Вони фільтрують трафік і блокують небажані або потенційно шкідливі запити, забезпечуючи захист серверів від атак і зловмисного використання.
- Захист VPN. Брандмауери можуть забезпечувати безпеку віртуальних приватних мереж, які використовуються для забезпечення захищеної комунікації між віддаленими розташованими мережами або працівниками, які працюють з віддалених місць. Вони контролюють доступ до VPN-тунелів і забезпечують безпеку передачі даних.
- Контроль доступу в домашніх мережах. Брандмауери можуть бути використані в домашніх мережах для контролю доступу до Інтернету. Вони дозволяють налаштовувати правила для блокування небажаного контенту, фільтрування веб-сайтів, контролю діяльності дітей в Інтернеті і т.д.

Всі ці сценарії вимагають встановлення відповідних політик безпеки в брандмауері, включаючи налаштування правил фільтрації трафіку, контролю доступу, мережевої адресації та інших параметрів.

2 БРАНДМАУЕРИ В ОПЕРАЦІЙНІЙ СИСТЕМІ FREEBSD

2.1 Огляд операційної системи FreeBSD

FreeBSD є вільно поширюваною та відкритою операційною системою, яка базується на системі UNIX. Вона має довгу історію розробки та активну спільноту розробників, що сприяє її постійному вдосконаленню та забезпечує надійність та стабільність системи [3].

Однією з головних переваг FreeBSD є її надійність. Операційна система володіє високою стабільністю та мінімальною кількістю відмов. Це забезпечує безперебійну роботу системи навіть при тривалому навантаженні та інтенсивному використанні ресурсів.

Другою важливою характеристикою FreeBSD є її гнучкість та масштабованість. Операційна система може працювати на різних апаратних платформах, від серверів до вбудованих систем. FreeBSD також підтримує велику кількість апаратного забезпечення, що робить його варіантом для різноманітних завдань.

Мережеві можливості FreeBSD також заслуговують на увагу. Операційна система надає повну підтримку TCP/IP протоколів, що дозволяє побудувати потужні мережеві інфраструктури.

Однією з особливостей FreeBSD є його система керування пакетами, відома як "pkg". Ця система дозволяє легко встановлювати, оновлювати та керувати пакетами програмного забезпечення. FreeBSD також має велику колекцію пакетів, які можна легко встановити з офіційних репозиторіїв.

Операційна система FreeBSD також славиться своєю безпекою. Розробники FreeBSD приділяють велику увагу захисту системи, постійно виправляючи виявлені уразливості та надаючи оновлення безпеки. FreeBSD також має вбудовані інструменти безпеки, такі як брандмауери, криптографічні сервіси та механізми контролю доступу.

Загалом, FreeBSD є потужною операційною системою з багатим набором функцій та високим рівнем надійності. Його гнучкість, масштабованість та мережеві можливості роблять його привабливим вибором для різноманітних

завдань, від серверних систем до вбудованих пристроїв. FreeBSD є популярним вибором серед системних адміністраторів та розробників, які цінують його надійність, безпеку та гнучкість.

У контексті операційної системи FreeBSD, брандмауери виконують важливу роль у забезпеченні безпеки мережі. FreeBSD надає кілька варіантів брандмауерів, включаючи IPFW, PF та IPFilter, які надають широкі можливості для налаштування правил фільтрації трафіку та забезпечення безпеки мережі.

2.2 Огляд доступних брандмауерів для FreeBSD

Операційна система FreeBSD має ряд вбудованих рішень для брандмауера, включаючи IPFW, PF, та IPFilter.

IPFW - це система брандмауера та трафік-шейпера, включена у базову систему FreeBSD [4]. IPFW розроблено для обробки пакетів на рівні ядра ОС, що дозволяє надзвичайно ефективну роботу.

IPFW пропонує набір правил, за якими пакети мережі відправляються, приймаються або відкидаються. Ці правила можуть бути дуже простими (наприклад, заборонити весь трафік) або дуже складними, з урахуванням різних параметрів, таких як IP-адреси, порти, протоколи та навіть вміст пакетів.

Також IPFW має модуль dummynet, що може використовуватися для обмеження пропускної здатності мережі. Це корисно при тестуванні додатків на їх здатність працювати в умовах поганого з'єднання або для обмежень по швидкості певних IP адрес чи мереж.

PF – це також брандмауер. Він походить від OpenBSD і є стандартним брандмауером в цій системі [4]. PF є потужним та гнучким інструментом, який включає в себе повноцінний брандмауер, трафік-шейпер, балансувальник навантаження та систему NAT.

PF дозволяє створювати складні набори правил для керування мережевим трафіком, включаючи перевірку стану (stateful inspection), що відстежує стан мережевих з'єднань і дозволяє більш тонку настройку політики безпеки.

IPFilter, або просто IPF, це крос-платформений брандмауер і NAT інструмент, що доступний в різних Unix-подібних системах, включаючи FreeBSD [4]. IPFilter підтримує обидва режими роботи - stateful і stateless, і дозволяє створювати доволі складні набори правил.

2.2.1 Огляд можливостей брандмауера PF

PF є потужним та гнучким брандмауером. Він використовує простий та зрозумілий конфігураційний формат, що дозволяє швидко налаштовувати правила та контролювати трафік мережі [4].

Нижче наведено кроки для налаштування брандмауера PF.

Для ввімкнення PF та PFlog в FreeBSD, потрібно додати наступні рядки до /etc/rc.conf:

```
pf_enable="YES"
pflog_enable="YES"
pf_rules="/etc/pf.conf"
pflog_logfile="/var/log/pflog"
```

Сервіс PF та PFlog можна запустити за допомогою команд:

```
#service pf start
#service pflog start
```

Pflog зберігає дані журналу в /var/log/pflog, який можна аналізувати. Pflog журналує лише пакети, для яких встановлено опцію "log" в правилах PF.

На рисунку 2.1 показано вивід команди `service pf status` та `service pflog status` для підтвердження коректного старту PF та PFlog.

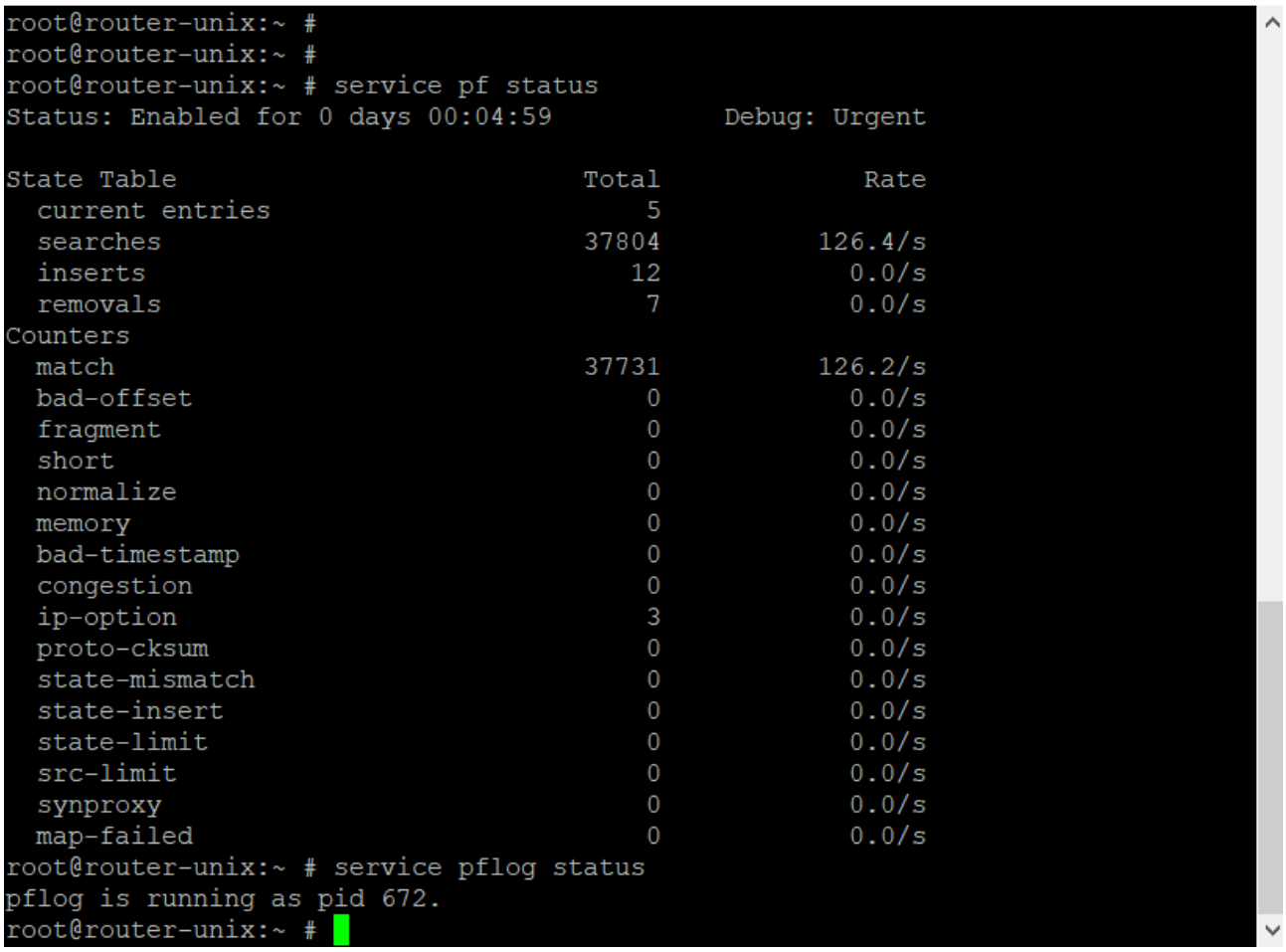

Рисунок 2.1 – Вивід команди `service pf status` та `service pflog status`

Правила PF визначаються в файлі /etc/pf.conf [5]. Файл може бути доволі складним, з великою кількістю можливих параметрів та опцій. Правила PF відрізняються своєю гнучкістю.

Синтаксис для правил фільтрації пакетів.

`[action] [direction] [log] [quick] [on interface] [family] [proto protocol] [from src_addr [port src_port]] [to dst_addr [port dst_port]] [flags tcp_flags] [state] [options]`

У цьому синтаксисі:

– `action` - визначає дію, що буде виконана для пакету. Зазвичай це pass (пропустити пакет) або block (заблокувати пакет);

– `direction` - визначає напрямок трафіку: in для вхідного трафіку, out для вихідного;

– `log` - якщо ця опція включена, всі пакети, які відповідають цьому правилу, будуть зареєстровані;

– `quick` - якщо ця опція включена, обробка правил припиняється, як тільки пакет відповідає цьому правилу;

– `interface` - ім'я мережевого інтерфейсу, до якого застосовується правило;

– `family` - сімейство адрес, на яке поширюється правило: inet для IPv4, inet6 для IPv6;

– `protocol` - протокол, який використовує пакет: tcp, udp, icmp та ін.;

– `from src_addr [port src_port]` - IP-адреса та (опціонально) порт відправника;

– `to dst_addr [port dst_port]` - IP-адреса та (опціонально) порт отримувача;

– `flags tcp_flags` - прапори TCP, що використовуються під час встановлення з'єднання;

– `state` - стан пакета в контексті перевірки стану;

– `options` - додаткові опції, які можуть змінювати поведінку правила, такі як keep state, modulate state, synproxy state, no-sync, sloppy, та ін.

Приклад простого правила яке дозволяє всі вхідні з'єднання на порт 22 (SSH) з будь-якої мережі.

```
pass in on em0 proto tcp from any to any port 22
```

Це не складний синтаксис, але він надає велику гнучкість при написанні правил фільтрації пакетів.

Операція NAT в PF використовується для зміни IP-адреси в пакетах, що проходять через брандмауер.

Основний синтаксис правил NAT в PF.

```
nat [on interface] [proto protocol] [from src_addr [port src_port]] [to dst_addr [port dst_port]] -> translation_addr [port trans_port] [static-port]
```

У цьому синтаксисі:

– `on interface` - інтерфейс, через який проходять пакети. Якщо він не вказаний, PF застосовує правило до всіх інтерфейсів;

– `proto protocol` - протокол пакетів (наприклад, tcp, udp, icmp);

– `from src_addr [port src_port]` - IP-адреса та/або порт вихідного пакету;

– `to dst_addr [port dst_port]` - IP-адреса та/або порт цільового пакету;

– `-> translation_addr [port trans_port]` - нова IP-адреса та/або порт, на які повинна бути змінена адреса джерела;

– `static-port` - якщо це вказано, порт вихідного пакету не змінюється. Це корисно для протоколів, які вимагають збереження портів, таких як UDP-стрімінг або протоколи голосового чату;

Приклад простого правила NAT в PF:

```
nat on em0 from 192.168.1.0/24 to any -> (em0)
```

Це правило змінює адресу джерела всіх пакетів, що проходять через інтерфейс *em0* і приходять з мережі *192.168.1.0/24*, на IP-адресу інтерфейсу *em0*.

Redirect є ще одним видом перетворення адрес, що використовується в PF, але, на відміну від NAT, RDR використовується для зміни адреси та/або порту призначення пакету. Це часто використовується для перенаправлення вхідних підключень на альтернативні сервери або порти в середині мережі.

Основний синтаксис для правил RDR в PF.

```
rdr [on interface] [proto protocol] [from src_addr [port src_port]] [to dst_addr [port dst_port]] -> translation_addr [port trans_port] [tag tagname] [tagged tagname] [random]
```

У цьому синтаксисі:

– `on interface` - інтерфейс, через який проходять пакети. Якщо він не вказаний, PF застосовує правило до всіх інтерфейсів;

– `proto protocol` - протокол пакетів (наприклад, tcp, udp, icmp);

– `from src_addr [port src_port]` - IP-адреса та/або порт вихідного пакету;

– `to dst_addr [port dst_port]` - IP-адреса та/або порт цільового пакету;

– `-> translation_addr [port trans_port]` - нова IP-адреса та/або порт, на які повинен бути змінений адреса призначення;

– `tag tagname` - ця опція додає мітку до пакетів, що відповідають правилу. Мітки можна використовувати в інших правилах для вибору пакетів;

– `tagged tagname` - ця опція вибирає пакети, що мають вказану мітку;

– `random` - ця опція вибирає випадковий порт для перенаправлення, якщо вказано декілька портів в полі translation_port;

Приклад простого правила RDR:

```
rdr on em0 proto tcp from any to any port 80 -> 192.168.1.2 port 8080
```

Це правило перенаправляє весь вхідний трафік TCP на порт 80 до машини з IP-адресою 192.168.1.2 на порт 8080.

2.2.2 Огляд можливостей брандмауера IPFW

IPFW є вбудованим брандмауером в операційній системі FreeBSD, який надає можливості для фільтрації пакетів, обмеження пропускної здатності та перенаправлення портів[6].

Для ввімкнення IPFW в FreeBSD, потрібно додати наступний рядок до /etc/rc.conf.

```
firewall_enable="YES"
```

Після цього IPFW можна запустити за допомогою команди:

```
#service ipfw start
```

Правила IPFW визначаються в файлі /etc/ipfw.rules. Це можуть бути доволі складні набори правил.

Основний синтаксис для правил IPFW виглядає так:

```
ipfw add [number] [action] [protocol] from [src_ip] [src_port] to [dst_ip] [dst_port] [options]
```

У цьому синтаксисі:

– `number` - це номер, який можна призначити цьому правилу. Правила виконуються в порядку зростання цих номерів;

– `action` - це дія, яку ви хочете виконати для пакетів, які відповідають цьому правилу. Дії можуть бути такими: allow, deny, count, skipto, divert, tee, fwd, тощо;

– `protocol` - це протокол пакету, на який ви хочете застосувати це правило. Може бути ip, tcp, udp, icmp, тощо;

– `src_ip [src_port]` - це IP-адреса та порт джерела, до якого ви хочете застосувати це правило. Можна використовувати маски підмережі;

– `dst_ip [dst_port]` - це IP-адреса та порт призначення, до якого ви хочете застосувати це правило. Можна використовувати маски підмережі;

– `options` - це додаткові опції, які ви хочете застосувати до правила. Опції включають в себе наступні значення: in, out, xmit, recv, via, setup, frag, options, тощо.

Приклад правила IPFW.

```
ipfw add 100 allow tcp from any to any 80 in via em0 setup keep-state
```

Це правило дозволяє вхідний TCP-трафік на порт 80 через інтерфейс em0 та використовує опцію keep-state для відслідковування стану з'єднання.

Використовуючи IPFW можена виконувати NAT. IPFW підтримує NAT через модуль natd. Для використання NAT в IPFW потрібно включити natd, вказавши наступне в /etc/rc.conf.

```
natd_enable="YES"
natd_interface="<interface>"
natd_flags="-dynamic -m"
```

Сервіс natd можна запустити за допомогою команд:

```
#service natd start
```

Щоб використовувати NAT потрібно вказати natd в якості адреси перенаправлення в команді divert.

Правило може виглядати так:

```
ipfw add divert natd all from any to any via em0
```

Це правило перенаправляє всі пакети, які проходять через інтерфейс em0, до natd для обробки NAT. В цьому прикладі em0 – це зовнішній інтерфейс, через який ви хочете виконувати NAT.

На рисунку 2.2 показано вивід команди `service ipwf status` та `service nad status` для підтвердження коректного старту IPWF та natd.

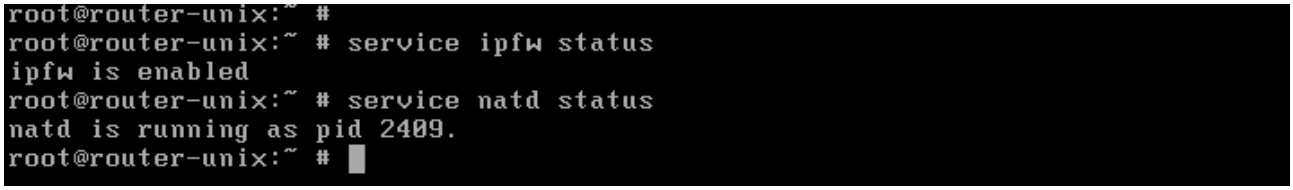

Рисунок 2.2 – Вивід команди `service ipwf status` та `service nad status`

В IPFW, є можливість перенаправити трафік за допомогою команди fwd. Вона використовується для перенаправлення пакетів на інший хост або порт.

Основний синтаксис команди є наступним:

```
ipfw add fwd [target_IP],[target_port] [protocol] from [src_ip] to [dst_ip] [src_port] [dst_port]
```

У цьому синтаксисі:

– `target_IP` - це цільова IP-адреса, на яку буде пересланий трафік;

– `target_port` - це цільовий порт, на який буде пересланий трафік;

– `protocol` - це протокол, який використовується для трафіку (наприклад, tcp або udp);

– `src_ip`- це початкова IP-адреса, з якої надходить трафік;

– `dst_ip` - це кінцева IP-адреса, до якої надходить трафік;

– `src_port` - це початковий порт, з якого надходить трафік;

– `dst_port` - це кінцевий порт, до якого надходить трафік;

Приклад правила з перенаправленням:

```
ipfw add fwd 192.168.1.2,8080 tcp from any to any 80
```

Це правило перенаправить весь вхідний трафік TCP, що приходить на порт 80, на хост 192.168.1.2 на порт 8080.

Перенаправлення трафіку може бути корисним для багатьох сценаріїв, включаючи балансування навантаження, перенаправлення трафіку на проксі-сервер або створення мережевого тунелю.

Для правильної роботи перенаправлення може знадобитися включити підтримку перенаправлення в ядрі FreeBSD. Для цього можна використати наступну команду:

```
#sysctl net.inet.ip.fw.one_pass=0
```

Ця команда змінить системний параметр, що контролює, як IPFW обробляє перенаправлення. Значення 0 дозволяє IPFW обробити пакет двічі: один раз перед перенаправленням, і один раз після нього.

Також в FreeBSD є вбудована підтримка NAT в IPFW, яка дозволяє виконувати NAT без потреби використання зовнішніх служб, таких як natd. Це покращує продуктивність.

Нижче наведено кроки для налаштування будованого NAT в IPFW.

Увімкніть IPFW. Додайте наступні рядки до файлу /etc/rc.conf:

```
firewall_nat_enable="YES"
firewall_nat_interface="<interface>"
```

В цьому прикладі `<interface>` - це інтерфейс, через який потрібно виконувати NAT.

Приклад конфігурації NAT:

```
ipfw -q nat 1 config if em0
ipfw add 100 nat 1 all from any to any via em0
```

В цьому прикладі створено екземпляр NAT з номером 1 і перенаправлено всі пакети через цей екземпляр NAT.

Це базова конфігурація вбудованого NAT в IPFW. Можна також використовувати додаткові параметри в команді `ipfw -q nat config` для налаштування додаткових параметрів NAT, таких як перенаправлення портів (redirect_port) або перенаправлення адрес (redirect_addr).

Функція redirect_address в IPFW використовується для реалізації статичного NAT. Це дозволяє вказати одну або декілька IP-адрес, які будуть перенаправлені на іншу IP-адресу.

Ось основний синтаксис для redirect_address:

```
ipfw nat [instance] config redirect_addr [external_IP] [internal_IP]
```

У цьому синтаксисі:

– `instance` - є номером екземпляра NAT, який ви використовуєте.

– `external_IP` – є IP-адресою, пакети на яку ви хочете перенаправити;

– `internal_IP` - є IP-адресою, на яку ви хочете перенаправити звернення до `external_IP`.

Наприклад:

```
ipfw nat 1 config redirect_addr 192.0.2.1 192.168.1.1
```

Це правило перенаправить всі пакети, адресовані 192.0.2.1, на 192.168.1.1.

Функція redirect_addr використовується для статичного NAT, тобто він не динамічно відображає порти, як це робить динамічний NAT. Він просто перенаправляє всі пакети з однієї IP-адреси на іншу.

Функція redirect_port в IPFW використовується для реалізації перенаправлення портів (Port Forwarding) або Destination NAT. Це дозволяє перенаправляти пакети, які приходять на конкретний зовнішній IP-адрес і порт, до іншого внутрішнього IP-адреси і порту.

Ось основний синтаксис для redirect_port:

```
ipfw nat [instance] config redirect_port [protocol] [external_IP]:[external_port] [internal_IP]:[internal_port]
```

У цьому синтаксисі:

– `instance` - є номером екземпляра NAT, який ви використовуєте;

– `protocol` - є протоколом, який ви хочете перенаправити (наприклад, tcp або udp);

– `external_IP:external_port` - є IP-адресою і портом, які ви хочете перенаправити;

– `internal_IP:internal_port` - є IP-адресою і портом, на які ви хочете перенаправити `external_IP:external_port`;

Наприклад:

```
ipfw nat 1 config redirect_port tcp 203.0.113.1:80 192.168.1.2:8080
```

Це правило перенаправить всі TCP-пакети, які приходять на 203.0.113.1 на порт 80, на 192.168.1.2 на порт 8080.

2.2.3 Огляд можливостей брандмауера IPFilter

IPFilter, відомий також як IPF, є надійним брандмауером та системою NAT, яка включена в базову систему FreeBSD [4]. Він підтримує широкий спектр функціональності, включаючи фільтрацію пакетів, перенаправлення трафіку та багато іншого.

Для включення IPF в FreeBSD, потрібно додати наступні рядки до файлу /etc/rc.conf.

```
ipfilter_enable="YES"
ipfilter_rules="/etc/ipf.rules"
```

В цьому прикладі ipfilter_rules вказує на файл, який має містити набори правил IPF.

Після цього IPF можна запустити за допомогою команди:

```
#service ipfilter start
```

На рисунку 2.3 показано вивід команди `service ipfilter status` для підтвердження коректного старту IPF [7].

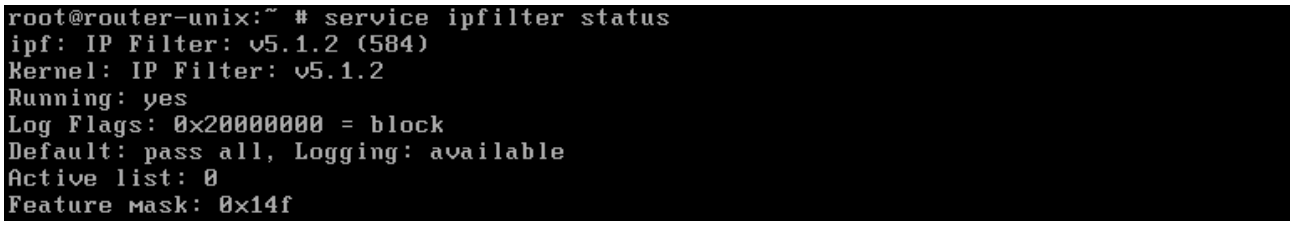

Рисунок 2.3 – Вивід команди `service ipfilter status`

Основний синтаксис правила IPFilter виглядає наступним чином.

```
[action] [direction] [log] [quick] on [in-interface | out-interface] proto [protocol] from [src-addr] to [dst-addr] port= [source_port/destination_port] [options]
```

У цьому синтаксисі:

– `action` - може бути `pass` (дозволяти) або `block` (блокувати);

– `direction` - може бути `in` (вхідний) або `out` (вихідний);

– `log` - вказує, що пакети, відповідні правилу, мають бути зареєстровані в системному журналі (логування);

– `quick` - не обов'язкова директива, яка призупиняє обробку подальших правил, якщо це правило застосовується;

– `on` - вказує на інтерфейс, до якого застосовується правило;

– `protocol` - вказує на протокол, наприклад, tcp, udp, icmp і т.д;

– `src-addr` - вказують на джерельний IP-адрес або діапазон адрес;

– `dst-addr` - вказують на призначений IP-адрес або діапазон адрес;

– `options` - використовується для вказівки додаткових опцій, такі як мітки TCP, тощо;

– source_port/destination_port вказує на порти джерела і призначення для пакетів TCP або UDP.

Приклад правила:

```
block in quick on em0 proto tcp from any to 192.0.2.0/24 port = 22
```

Це правило негайно блокує всі вхідні TCP пакети на інтерфейсі em0, які намагаються досягти будь-якої машини в мережі 192.0.2.0/24 на порт 22.

IPF має вбудований модуль NAT. Для використання NAT в IPF, потрібно додати наступні рядки до файлу /etc/rc.conf.

```
ipnat_enable="YES"
ipnat_rules="/etc/ipf_ipnat.rules"
```

Після цього ipnat можна запустити за допомогою команди:

```
#service ipnat start
```

Файл ipf_ipnat.rules має містити набори правил NAT, які визначають, як пакети перенаправляються і транслюються [8].

IPF підтримує перенаправлення трафіку через правила NAT. Можна використовувати команду rdr для перенаправлення вхідного трафіку на іншу IP-адресу або порт.

В IPFilter NAT може бути конфігурован за допомогою правил map та rdr в файлі правил NAT.

Синтаксис для правил nat та rdr.

```
map {interface} {source-addr}/mask -> {dest-addr}/mask
rdr {interface} {source-addr}/mask port {port} -> {dest-addr}/mask port {port}
```

У цьому синтаксисі:

– map - використовується для трансляції вихідних пакетів;

– rdr - використовується для трансляції вхідних пакетів (тобто перенаправлення портів);

– interface - це мережевий інтерфейс, на якому застосовуються правила.

– source-addr - це вихідний IP-адрес або діапазон адрес;

– dest-addr - це IP-адрес призначення або діапазон адрес;

– mask - це маска підмережі для вихідних IP-адрес та IP-адрес призначення;

– `port` - це порт або діапазон портів для перенаправлення (для rdr).

Приклади правила map:

– `map em0 192.168.1.0/24-> 203.0.113.0/32`

Це правило змінює адресу джерела всіх пакетів, що проходять через інтерфейс em0 і приходять з мережі 192.168.1.0/24, на IP-адресу 203.0.113.0.

Приклади правила перенаправлення портів rdr:

```
rdr em0 0.0.0.0/0 port 8080 -> 192.168.1.100 port 80
```

Це правило перенаправляє всі пакети, що приходять на порт 8080 з будь-якої адреси, до адреси 192.0.2.10 на порт 80.

2.3 Переваги та недоліки PF, IPFW і IPFilter. Вибір брандмауера

Як вже було сказано PF, IPFW і IPFilter є три різні брандмауера, доступні в операційній системі FreeBSD.

Ось порівняльний аналіз цих брандмауерів.

PF.

PF є популярним брандмауером і має декілька переваг та недоліків у FreeBSD.

Переваги PF:

- Гнучкість. PF надає широкі можливості для настройки правил фільтрації трафіку. Він підтримує різні умови фільтрації, такі як IP-адреси, порти, протоколи, стан з'єднання та багато інших параметрів, що дозволяє створювати гнучкі та детальні політики безпеки.
- Перевірка стану. PF підтримує перевірку стану (stateful inspection), що дозволяє відстежувати стан з'єднання та налаштовувати правила на основі цього стану. Це спрощує налаштування правил для дозволу або блокування пакетів, враховуючи їхній контекст.
- Потужність. PF має багато додаткових функцій, таких як NAT, боротьба з DoS-атаками та інші. Це робить PF більш потужним та універсальним інструментом для забезпечення безпеки мережі.

- Простота управління. PF має структурований синтаксис та зрозумілі правила, що спрощує управління правилами фільтрації. Конфігураційні файли PF мають зрозумілу структуру, що полегшує їх редагування та зрозуміння.

Недоліки PF:

- Навчання синтаксису. Навчання синтаксису та основ PF може вимагати певного часу та зусиль, особливо для новачків. Розуміння всіх можливостей та параметрів PF може бути викликом для користувачів без попереднього досвіду;
- Потреба в ресурсах. PF може вимагати певних ресурсів системи для обробки великого обсягу трафіку та складних правил фільтрації. Для масштабних мереж або систем з обмеженими ресурсами це може бути фактором, який треба враховувати;
- Відсутність графічного інтерфейсу. PF використовує текстові конфігураційні файли для налаштування правил. Це може бути не зручним для користувачів, які більш звикли до графічного інтерфейсу.

В цілому, PF в FreeBSD є потужним інструментом для забезпечення безпеки мережі, проте він може вимагати певного часу та зусиль для навчання та налаштування.

IPFW.

IPFW є іншим брандмауером, що використовується в FreeBSD.

Переваги IPFW:

- Вбудована підтримка. IPFW входить до стандартного комплекту поставки FreeBSD, що означає, що він доступний "з коробки" і не потребує додаткових налаштувань або встановлення.
- Простота використання. IPFW має простий синтаксис та легко зрозумілі правила. Це робить його відмінним вибором для користувачів з обмеженим досвідом управління брандмауерами.
- Підтримка старших версій FreeBSD. IPFW є одним з найдавніших брандмауерів у FreeBSD та підтримується у старших версіях операційної

системи. Це робить його варіантом для тих, хто використовує старіші версії FreeBSD.

Недоліки IPFW:

- Обмежені можливості. IPFW має обмежений набір функцій порівняно з іншими брандмауерами, такими як PF. Він може бути менш потужним для складних сценаріїв фільтрації трафіку або для використання додаткових функцій, таких як NAT.
- Відсутність активного розвитку. IPFW не отримав активного розвитку протягом останніх років і не має такої активної спільноти користувачів, як у PF. Це може означати, що IPFW може бути менше підтримуваним або отримувати менше оновлень та виправлень помилок порівняно з іншими брандмауерами.
- Відсутність графічного інтерфейсу. IPFW також використовує текстові конфігураційні файли для налаштування правил, що може бути менш зручним для користувачів, які звикли до графічного інтерфейсу.

IPFilter.

IPFilter є ще одним брандмауером, який використовується в FreeBSD. Переваги IPFilter:

- Простота використання. IPFilter має простий та зрозумілий синтаксис правил, що дозволяє легко створювати та налаштовувати правила фільтрації трафіку.
- Висока продуктивність. IPFilter відомий своєю швидкодією та ефективністю. Він працює на низькому рівні мережевого стеку, що дозволяє обробляти великий обсяг трафіку з мінімальним впливом на продуктивність системи.
- Гнучкість та розширюваність. IPFilter підтримує широкий спектр функцій, включаючи фільтрацію на основі IP-адрес, портів, протоколів та інших атрибутів пакетів.

Недоліки IPFilter:

- Відсутність активного розвитку. IPFilter не отримує активного розвитку та оновлень протягом останніх років. Це може призвести до відсутності нових функцій та виправлень помилок, а також може вплинути на сумісність з новими версіями FreeBSD.
- Обмежений набір функцій. У порівнянні з деякими іншими брандмауерами, IPFilter може мати обмежений набір функцій, таких як глибокий інспектор пакетів.
- Відсутність графічного інтерфейсу. IPFilter також використовує конфігураційні файли для налаштування правил, що може бути менш зручним для користувачів, які звикли до графічного інтерфейсу.

Вибір між IPFilter, PF і IPFW залежить конкретних потреб і вимог. Ось деякі фактори, які можуть вплинути на вибір:

- Функціональність. PF вважається потужнішим і більш гнучким брандмауером, оскільки він підтримує широкий спектр функцій, керування пропускною здатністю, модульність та інше.
- Синтаксис та простота використання. IPFW має простий синтаксис, що може зробити його більш привабливим для новачків. PF також має зрозумілий синтаксис та підтримку багатьох зручних функцій. IPFilter також має простий синтаксис, але він може бути менш інтуїтивно зрозумілим для деяких користувачів.
- Активний розвиток та підтримка. PF є проектом що активно розвивається, який отримує оновлення та підтримку від FreeBSD-команди розробників. IPFW також отримує певний рівень підтримки, оскільки це одна з основних брандмауерних систем FreeBSD. IPFilter не має активного розвитку в останні роки, що може вплинути на його сумісність з новими версіями FreeBSD та наявність нових функцій.
- Потреби і вимоги проекту. Вибір брандмауера повинен відповідати конкретним потребам і вимогам проекту.

В кінцевому рахунку, немає однозначної відповіді на питання, який брандмауер є найкращим. Вибір залежить від потреб, знань та вимог проекту.

Але оскільки ми плануємо використовувати наш брандмауер не лише як фільтр пакетів та NAT а і для боротьби з атаками грубої сили (brute force) та пом’якшенням DDoS атак то вибір однозначна падає на PF.

3 НАЛАШТУВАННЯ ТА ТЕСТУВАННЯ PF В FREBSD

3.1 Налаштування FreeBSD як шлюзу з NAT та брандмауером

FreeBSD може ефективно використовуватись як шлюз з функцією NAT та брандмауером для забезпечення безпеки мережі. У цьому режимі FreeBSD діє як центральна точка входу та виходу для мережі, обробляючи трафік між внутрішньою мережею і зовнішньою мережею.

На рисунку 3.1 наведено схему мережі де операційна системи FreeBSD використовується як шлюз з NAT та брандмауер.

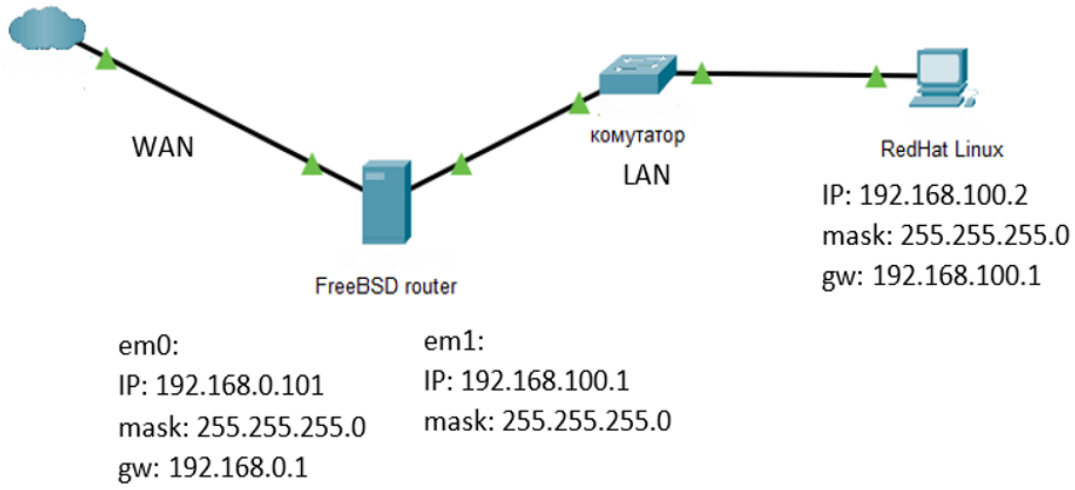

Рисунок 3.1 – Схему включення FreeBSD як шлюзу з NAT та брандмауером

Ця схема показує загальний принцип роботи FreeBSD як шлюзу з NAT.

Локальна мережа (LAN) - це внутрішня мережа, де знаходяться клієнтські комп'ютери та інші пристрої. Вона налаштована з використанням приватних IP-адрес з мережі 192.168.100.0/24.

Зовнішня мережа (WAN) - це мережа, до якої підключений FreeBSD шлюз. Зовнішній інтерфейс FreeBSD може мати публічну IP-адресу, надану провайдером Інтернету. У нашому випадку це також приватні IP-адреси з мережі 192.168.0.0/24.

У цій схемі інтерфейс em1 відноситься до локальної мережі, а інтерфейс em0 - до зовнішньої мережі.

FreeBSD налаштовується з використанням PF для забезпечення NAT. Це дозволяє пристроям з локальної мережі виходити в Інтернет, використовуючи публічну IP-адресу шлюзу.

Також FreeBSD виконує роль шлюзу, маршрутизуючи трафік між локальною мережею та зовнішньою мережею. Він також забезпечує функції безпеки, фільтруючи певні типи трафіку та застосовуючи правила брандмауера [3].

На рисунку 3.2 показано параметри налаштування FreeBSD в конфігураційному файлі /etc/rc.conf.

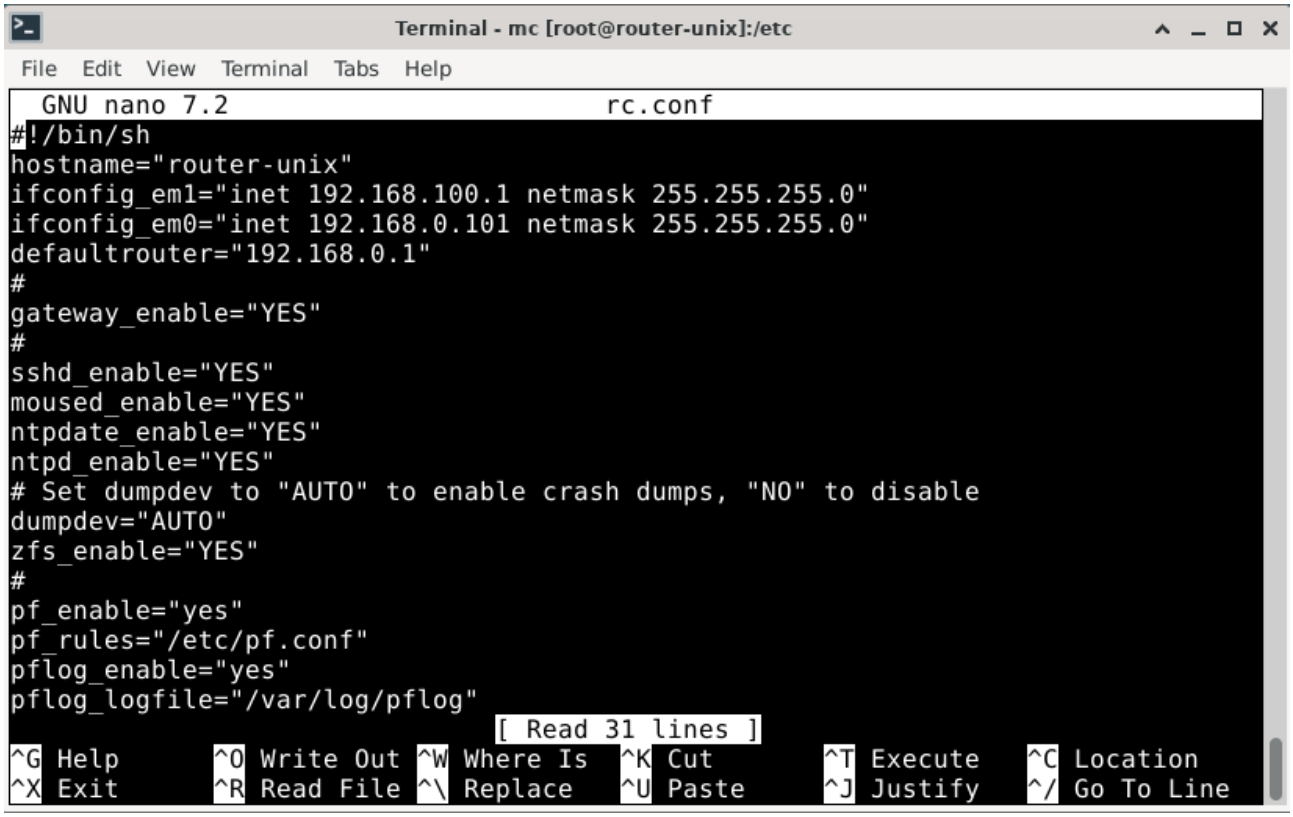

Рисунок 3.2 – Вміст конфігураційного файлу /etc/rc.conf

Конфігураційний параметр `gateway_enable="YES"` в файлі /etc/rc.conf в операційній системі FreeBSD вказує, що дана система виконує функцію шлюзу та повинна виконувати маршрутизацію між мережами. Застосування даного параметра є одним із важливих кроків для налаштування FreeBSD як шлюзу з NAT та маршрутизацією між мережами.

`ifconfig_em1="inet 192.168.100.1 netmask 255.255.255.0"` – цей рядок налаштовує інтерфейс `em1` з IP-адресою 192.168.100.1 та маскою підмережі

255.255.255.0. Це відповідає конфігурації внутрішнього інтерфейсу, який з'єднаний з локальною мережею.

`ifconfig_em0="inet 192.168.0.101 netmask 255.255.255.0"` - цей рядок налаштовує інтерфейс `em0` з IP-адресою 192.168.0.101 та маскою підмережі 255.255.255.0. Це відповідає конфігурації зовнішнього інтерфейсу, який з'єднаний з зовнішньою мережею або Інтернет.

`defaultrouter="192.168.0.1"` - цей рядок встановлює IP-адресу шлюзу за замовчуванням (default gateway) як 192.168.0.1. Це вказує, що всі мережеві пакети, які не належать до локальної мережі, повинні бути надіслані через цей шлюз.

Здійснимо налаштування PF. Створимо файл /etc/pf.conf з наступним вмістом [4].

```
ext_if = " em0"
int_if = " em1"
 localnet = $int_if:network
 set limit { states 40000, frags 40000, src-nodes 4000 }
 set optimization aggressive
 nat on $ext_if from $localnet to any -> ($ext_if)
#
pass quick from self to self keep state
pass quick from self to any keep state
#
block in inet proto tcp to $ext_if port ssh
block in inet proto tcp to $ext_if port 80
block in inet proto tcp to $ext_if port 443
#
pass from { lo0, $localnet } to any keep state
```

Наведений приклад конфігураційного файла PF використовується для налаштування операційної системи FreeBSD як шлюзу з NAT з функцією брандмауера та має обмеження для станів.

Ось детальний опис кожного параметра конфігурації:

1) Визначення макросів для зовнішнього та внутрішнього інтерфейсів.

```
ext_if = "em0"  # макрос для зовнішнього інтерфейсу
```

```
int_if = "em1"  # макрос для внутрішнього інтерфейсу
```

У цьому випадку `em0` використовується як зовнішній інтерфейс, який з'єднується з Інтернетом, а `em1` - як внутрішній інтерфейс, який з'єднується з локальною мережею.

2) Визначення макросу для локальної мережі, використовуючи внутрішній інтерфейс.

```
localnet = $int_if:network
```

Цей макрос використовується для визначення IP-адреси та маски підмережі внутрішнього інтерфейсу `em1`. Це дозволяє зручно використовувати цю мережу в правилах конфігурації.

3) Встановлення обмежень для станів, фрагментів та вузлів.

```
set limit { states 40000, frags 40000, src-nodes 4000 }
```

Цей рядок встановлює обмеження для різних аспектів функціонування PF. У вказаному прикладі встановлені такі обмеження:

– `states` - максимальна кількість активних мережевих станів, які PF може відслідковувати одночасно;

– `frags` - максимальна кількість мережевих фрагментів, які PF може обробляти одночасно;

– `src-nodes` - максимальна кількість унікальних джерел (IP-адрес), які PF може відслідковувати одночасно.

4) Встановлення агресивного рівня оптимізації.

```
set optimization aggressive
```

Цей рядок вказує PF використовувати агресивні алгоритми оптимізації для покращення продуктивності та обробки трафіку. Це може підвищити навантаження на процесор, але при цьому забезпечується краща пропускна здатність та швидкодія PF.

5) Налаштування правила для NAT.

```
nat on $ext_if from $localnet to any -> ($ext_if)
```

Це правило здійснює NAT з IP-адрес внутрішньої мережі `$localnet` на IP-адрес зовнішнього інтерфейсу `($ext_if)`. Всі пакети, що проходять з внутрішнього інтерфейсу до будь-якої зовнішньої адреси, будуть автоматично транслюватись на зовнішню IP-адресу інтерфейсу.

6) Встановлення правил пропуску та блокування трафіку.

```
pass quick from self to self keep state
pass quick from self to any keep state
block in inet proto tcp to $ext_if port ssh
block in inet proto tcp to $ext_if port 80
block in inet proto tcp to $ext_if port 443
```

Ці правила дозволяють пропускати (`pass`) або блокувати (`block`) певні типи трафіку. У цьому випадку, правила дозволяють пропускати трафік між самим собою (`self`) та між шлюзом і будь-якою адресою (`any`), зберігаючи при цьому стан з'єднання (`keep state`). Також встановлені правила блокування вхідного трафіку TCP на портах SSH (порт 22), HTTP (порт 80) та HTTPS (порт 443).

7) Пропуск трафіку від локальної мережі до будь-якої мережі.

```
pass from { lo0, $localnet } to any keep state
```

Це правило дозволяє пропускати трафік від локальної мережі (включаючи lo0, який представляє локальний інтерфейс) до будь-якої мережі, зберігаючи стан з'єднання.

На рисунку 3.3 та 3.4 можна побачити що робоча станція з операційною Red Hat Linux має стабільний доступ до мережі Інтернет.

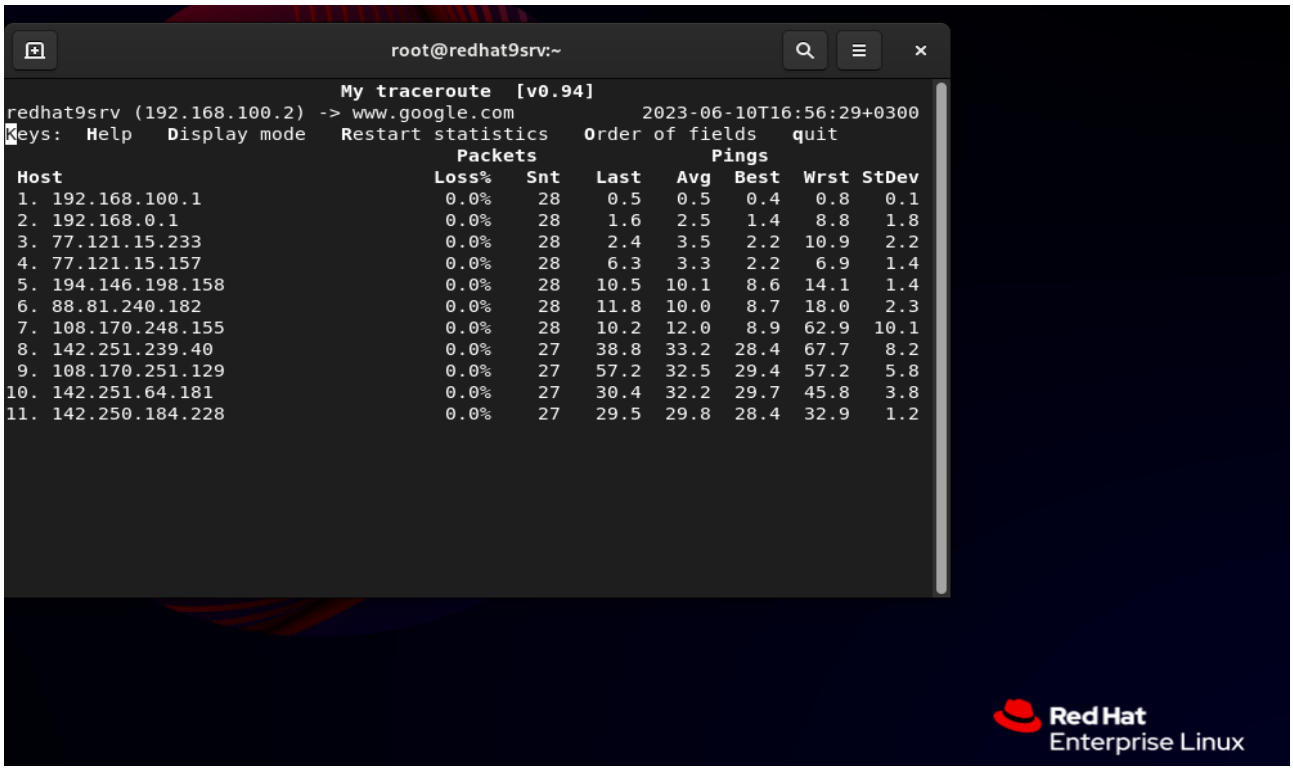

Рисунок 3.3 – Вивід команди mtr www.google.com на робочій станції з операційною Red Hat linux

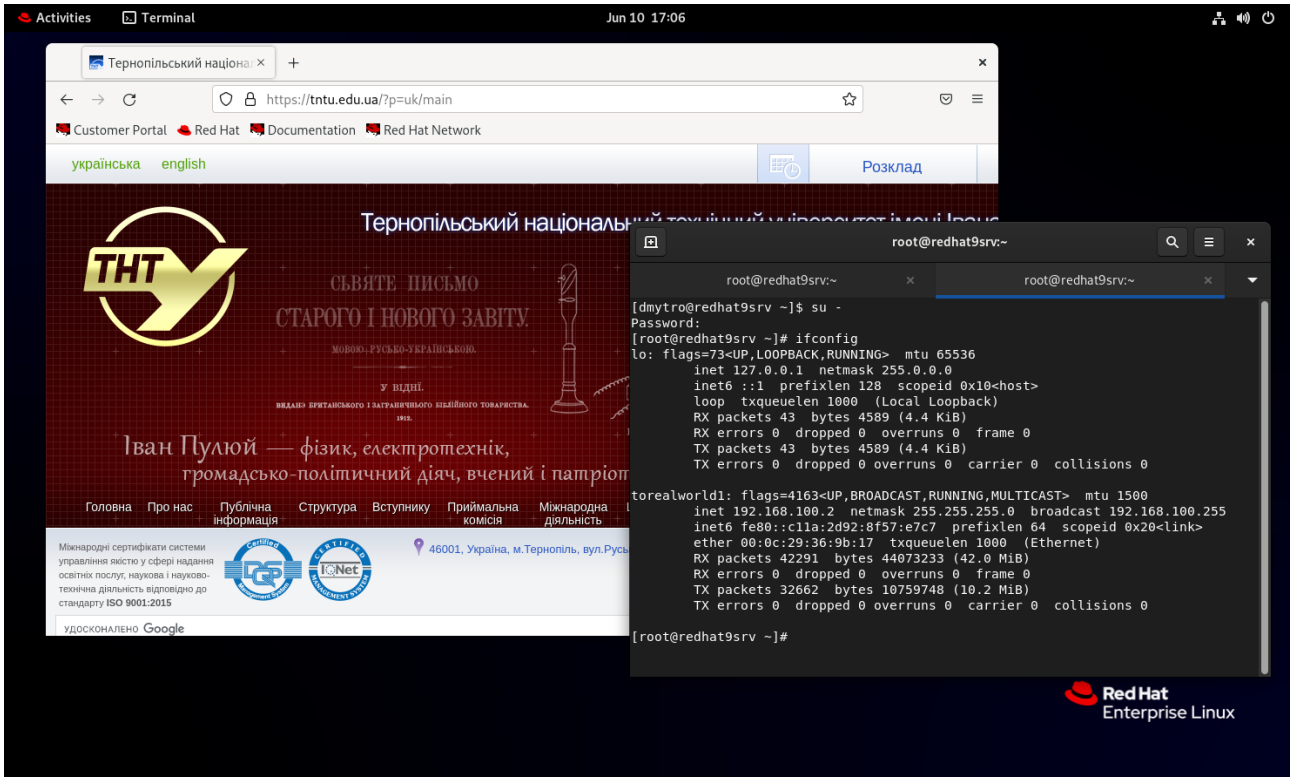

Рисунок 3.4 – Вивід команди ifconfig та приклад завантаженої сторінки на робочій станції з операційною Red Hat

Цей приклад конфігураційного файлу PF демонструє базове налаштування операційної системи FreeBSD як шлюзу з NAT з використанням правил та обмежень для керування трафіком.

3.2 Здійснення brute force атаки на SSH-сервер

Brute force атака на SSH-сервер полягає у спробах послідовно перебрати всі можливі комбінації логінів та паролів з метою незаконного доступу до системи. Це дуже небезпечна атака, яка може призвести до компрометації безпеки сервера та небажаного доступу до конфіденційної інформації.

На рисунку 3.5 наведено схему мережі, яка буде використовуватись для проведення brute force атаки на SSH-сервер.

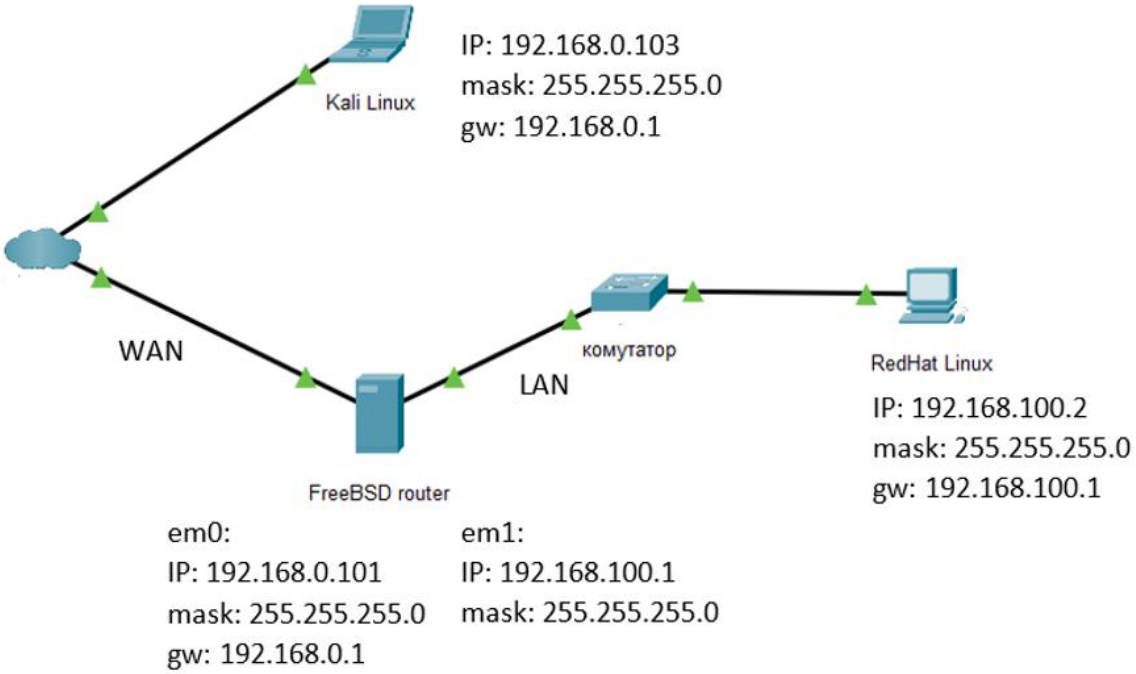

Рисунок 3.5 – Схема мережі для проведення brute force атаки

Для показу методики проведення brute force атаки добавимо користувача vlad з паролем 123456 в операційну систему FreeBSD (Рис.3.5)

```
root@router-unix:/etc# adduser
Username: vlad
Full name: Vlad Pahoda
Uid (Leave empty for default):
Login group [vlad]:
Login group is vlad. Invite vlad into other groups? []: wheel
Login class [default]:
Shell (sh csh tcsh bash rbash nologin) [sh]:
Home directory [/home/vlad]:
Home directory permissions (Leave empty for default):
Use password-based authentication? [yes]:
Use an empty password? (yes/no) [no]:
Use a random password? (yes/no) [no]:
Enter password:
Enter password again:
Lock out the account after creation? [no]:
Username   : vlad
Password   : *****
Full Name  : Vlad Pahoda
Uid        : 1011
Class      :
Groups     : vlad wheel
Home       : /home/vlad
Home Mode  :
Shell      : /bin/sh
Locked     : no
OK? (yes/no): y
adduser: INFO: Successfully added (vlad) to the user database.
```

Рисунок 3.5 – Занесення користувача в операційну FreeBSD

Атаку здійснимо з Kali Linux [9].

Kali Linux є дистрибутивом, спеціально розробленим для проведення тестування на проникнення та здійснення різних видів кібератак. Один з

популярних інструментів, доступних у Kali Linux, для проведення атак грубої сили на SSH-сервери і не тільки - це Hydra [10].

Hydra є багатопотоковим інструментом для атак грубої сили, який може використовуватись для відновлення паролів шляхом спроби всіх можливих комбінацій паролів. Нижче наведений загальний опис процесу використання Kali Linux та Hydra для атаки грубої сили на SSH-сервер.

1) Підготовка для атаки грубої сил.

Перед виконанням атаки грубої сили на SSH-сервер, потрібно знати інформацію про цільовий сервер, таку як його IP-адресу, порт SSH і мати список можливих паролів та користувачів.

Також в нашому випадку потрібно видалити правило з PF, яке блокує доступ по SHH `block in inet proto tcp to $ext_if port ssh.`

Та додати наступне правило.

```
pass in log inet proto tcp to $ext_if port ssh
```

де:

– `pass` - це ключове слово, яке дозволяє вказаному трафіку проходити через брандмауер;

– `Log` - це ключове слово, яке вказує брандмауеру реєструвати трафік, що відповідає цьому правилу.

2) Використаємо консольну команду для запуску атаки грубої сили на SSH-сервер. Нижче наведений загальний синтаксис команди.

```
hydra -l <username> -P <password_list> ssh://<target_IP> -t <threads_count>
```

де:

– `username` - ім'я користувача SSH, пароль якого потрібно підібрати;

– `password_list` - шлях до файлу, який містить список можливих паролів для спроби;

– `target_IP` - IP-адреса цільового SSH-сервера;

– `threads_count` - кількість потоків, які будуть використовуватись Hydra для атаки (не обов’язковий).

3) Після запуску Hydra почне перебирати всі можливі комбінації паролів зі списку. Якщо вона успішно знайде відповідну пару то виведе успішний результат, який буде містити ім'я користувача та пароль.

Як можна побачити з рисунку 3.6 підбір паролю користувача пройшов успішно. Оскільки ми навмисно встіновили пароль **123456** та явно вказали ім’я користав **vlad** в параметрах запуску Hydra процес підбору завершився позитивно за дуже короткий проміжок часу.

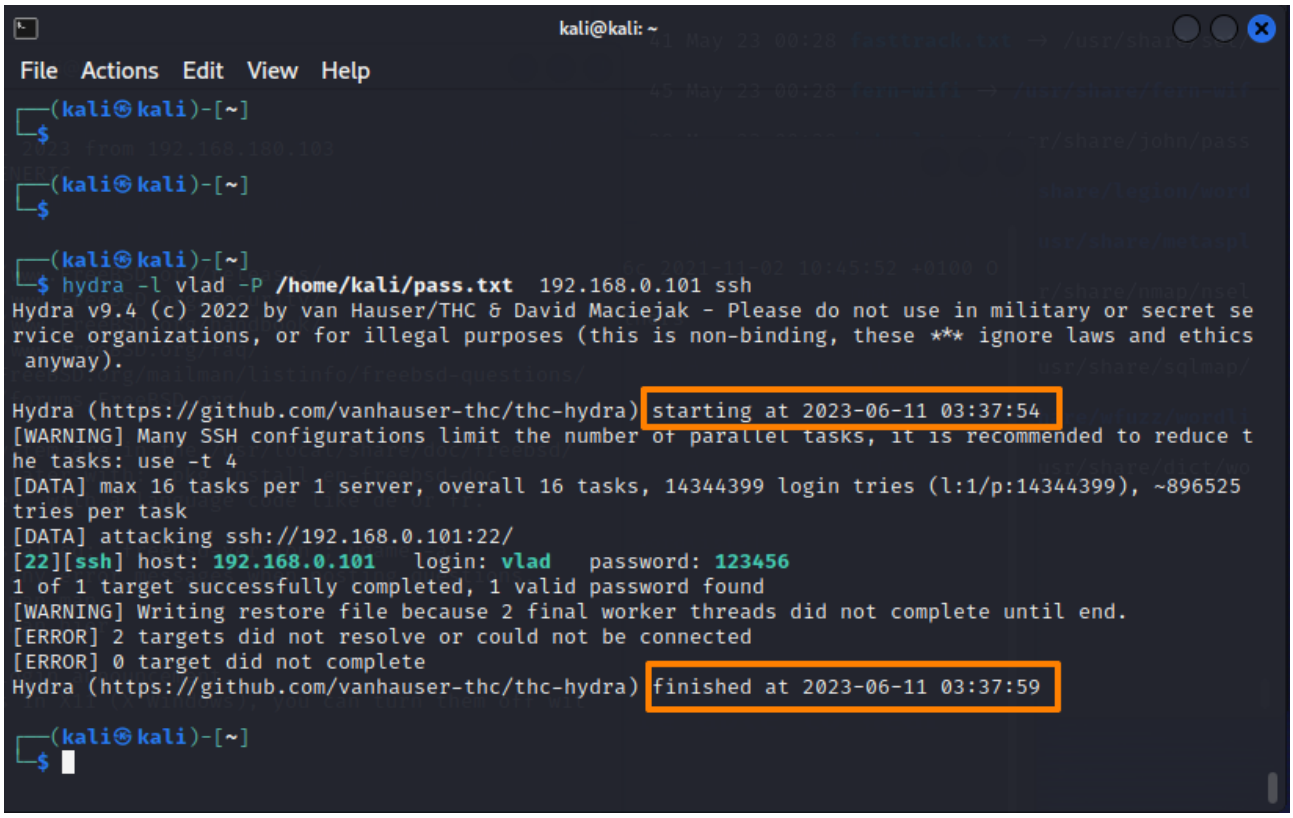

Рисунок 3.6 – Результати проведення brute force атаки на SSH-сервер

У FreeBSD лог-файл, що містить події з SSH авторизацією, зазвичай знаходиться за шляхом /var/log/auth.log. Для перегляду цього лог-файлу можете скористатись текстовим редактором або командою cat. На рисунку 3.7 показано вивід команди `cat /var/log/auth.log`

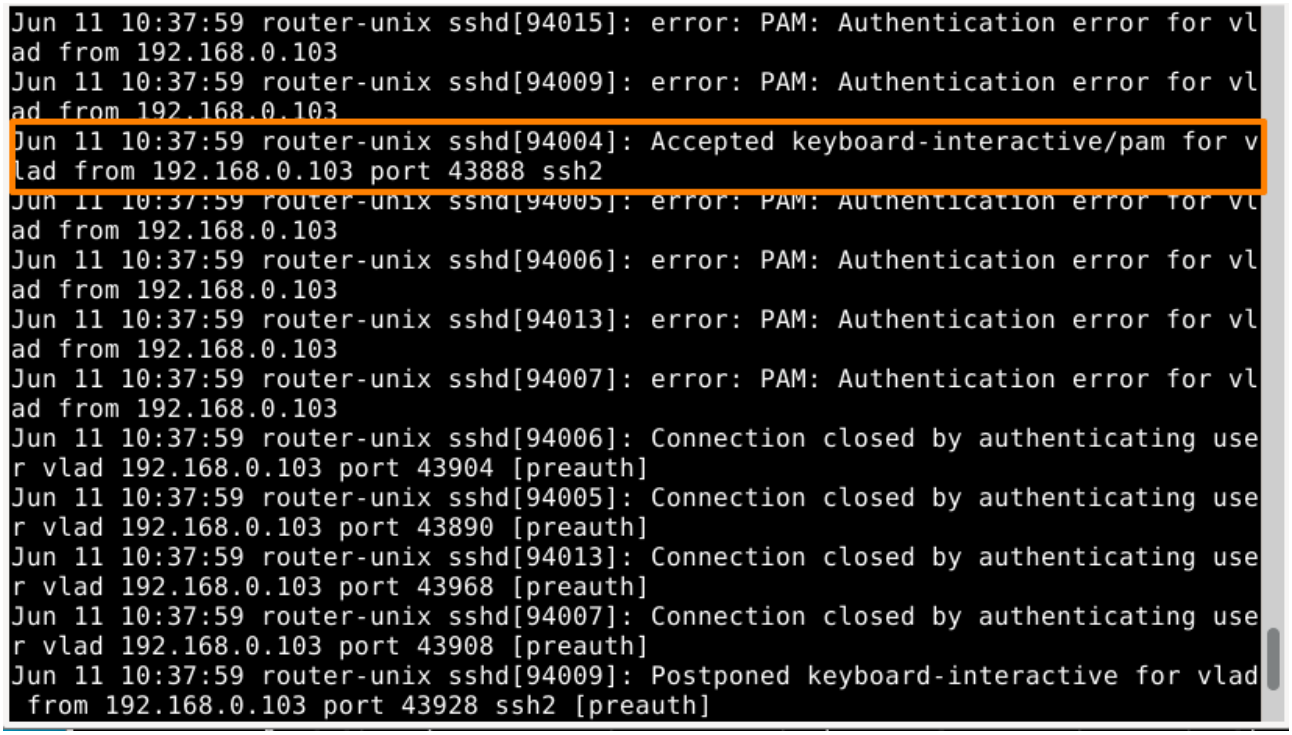

Рисунок 3.7 – Вивід команди cat /var/log/auth.log

Виділений запис означає, що о 10:37:59 на маршрутизаторі з іменем `"router-unix"` була здійснена SSH-авторизація. Ідентифікатор процесу SSH-сервера (sshd[94004]) вказує на процес, що обробляв з'єднання. `"Accepted keyboard-interactive/pam"` вказує на успішну авторизацію з використанням клавіатурного інтерактивного методу аутентифікації і PAM. Користувач з іменем "vlad" був авторизований з IP-адреси 192.168.0.103.

Це типове повідомлення у логах SSH, яке фіксує подію успішної авторизації до системи через SSH. Це важлива інформація, яка дозволяє відстежувати вхід до системи та проводити аудит дії користувачів. Також дане повідомлення підтверджує підбір паролю користувача.

Також ми можемо побачити події зв’язані з підбором пароля в /var/log/pflog файлі.

Команда `tcpdump -ner /var/log/pflog` використовується для аналізу вмісту лог-файлу pflog за допомогою утиліти tcpdump.

Основні прапорці команди tcpdump використовуються для наступного:

– `-n` - вказує tcpdump не виконувати розпізнавання імен хостів або портів, тобто виводить IP-адреси та порти в числовому форматі;

– `-e` - включає вивід даних кадру Ethernet;

– `-r /var/log/pflog` - вказує tcpdump читати лог-файл /var/log/pflog.

Дана команда відкриває лог-файл pflog для аналізу пакетів, які були спіймані брандмауером PF. Вивід цієї команди буде містити інформацію про заголовки кожного пакету, включаючи джерело та призначення, тип пакету, інформацію про протоколи та інші відомості.

На рисунку 3.8 показано вивід команди `tcpdump -ner /var/log/pflog`

```
10:37:57.308874 rule 5/0(match): pass in on em0: 192.168.0.103.43994 > 192.168.0
.101.22: Flags [S], seq 3230420907, win 64240, options [mss 1460,sackOK,TS[|tcp]
>
10:37:57.308973 rule 5/0(match): pass in on em0: 192.168.0.103.44000 > 192.168.0
.101.22: Flags [S], seq 3010046272, win 64240, options [mss 1460,sackOK,TS[|tcp]
>
10:37:57.308997 rule 5/0(match): pass in on em0: 192.168.0.103.44002 > 192.168.0
.101.22: Flags [S], seq 2348292043, win 64240, options [mss 1460,sackOK,TS[|tcp]
>
10:37:57.309010 rule 5/0(match): pass in on em0: 192.168.0.103.44010 > 192.168.0
.101.22: Flags [S], seq 1671721371, win 64240, options [mss 1460,sackOK,TS[|tcp]
>
10:37:57.309086 rule 5/0(match): pass in on em0: 192.168.0.103.44016 > 192.168.0
.101.22: Flags [S], seq 3282638416, win 64240, options [mss 1460,sackOK,TS[|tcp]
>
root@router-unix:/etc # tcpdump -ner /var/log/pflog
```

Рисунок 3.8 – Вивід команди tcpdump -ner /var/log/pflog

Розшифровуючи виділений запис отримуємо наступну інформацію про пакет:

- Час: 10:37:57.308973.
- Правило: rule 5/0(match).
- Дія: pass (проходить через брандмауер).
- Інтерфейс: in on em0 (прийнято з вхідного інтерфейсу em0).
- Джерело: 192.168.0.103.44000 (IP-адреса та порт джерела).
- Призначення: 192.168.0.101.22 (IP-адреса та порт призначення).
- Прапопець: Flags [S] (встановлений флаг SYN, початок з'єднання).
- Послідовність: seq 3010046272 (номер послідовності пакета).
- Вікно: win 64240 (розмір вікна).
- Опції: options [mss 1460, sackOK, TS[|tcp] (додаткові опції пакета).

Цей запис описує пакет, який був прийнятий на вхідному інтерфейсі em0 з IP-адреси 192.168.0.103 та порту 44000 і був спрямований до IP-адреси

192.168.0.101 на порт 22 (SSH). Пакет має встановлений флаг SYN, що свідчить про початок з'єднання. Додаткові опції пакета включають розмір MSS (Maximum Segment Size), підтримку Selective Acknowledgment (SACK) та відмітки часу (TCP Timestamps).

Це лише один запис з лог-файлу pflog, який містить інформацію про пакети, що пройшли через брандмауер. За допомогою подібних записів можна аналізувати трафік, виявляти аномалії та розбирати проблеми безпеки мережі.

3.3 Налаштування PF для захисту від brute force та пом’якшення DDoS атак

Використання PF для захисту від brute force атак включає написання правил, які дозволяють встановлювати обмеженя на кількість спроб підключення від однієї IP-адреси за певний період часу. Якщо IP-адреса перевищує встановлене обмеження, вона автоматично додється до "чорного списку" та блокується PF.

Для захисту від DDoS-атак, PF має бути налаштований для обмеження кількості одночасних з'єднань з одного джерела та обмеження загальної пропускної здатності. Завдяки функції SYNPROXY PF може захистити систему від SYN flood атак, за допомогою яких атакуючий може спробувати виснажити ресурси системи, викликаючи велику кількість напіввідкритих з'єднань. SYNPROXY використовується для перехоплення TCP SYN пакетів, перевірки, чи можуть вони встановити з'єднання, і лише після цього пакети передаються до пункту призначення [4].

Далі наведено конфігурацію PF для захисту від brute force на певні порти та налаштування пом’якшення DDoS атак.

```
#!/bin/sh
ext_if = "em0"
int_if = "em1"
localnet = $int_if:network
#
set skip on lo0
set limit { states 40000, frags 40000, src-nodes 4000 }
set optimization aggressive
#
```

```
scrub in all
#
nat on $ext_if from $localnet to any -> ($ext_if)
#
pass in log inet proto tcp to $ext_if port ssh
#
pass from { lo0, $localnet } to any keep state
#
table <ddos> persist
block drop in quick from <ddos>
#
pass in on $ext_if proto tcp from any to $ext_if port {http, https} flags S/SA synproxy state (source-track rule, max-src-nodes 1000, max-src-conn 100, max-src-states 1000, max-src-conn-rate 100/60, overload <ddos> flush global)
#
table <bruteforcessh> persist
block drop in quick from <bruteforcessh>
#
#block bruteforce SSH
pass in quick on $ext_if proto tcp from any to $ext_if port ssh flags S/SA keep state (max-src-conn 15, max-src-conn-rate 3/120, overload <bruteforcessh> flush global)
#
table <bruteforcemail> persist
block drop in quick from <bruteforcemail>
#
#block  bruteforce MAIL
pass in quick on $ext_if proto tcp from any to $ext_if port {993, imap, 587, 465 } flags S/SA keep state (max-src-conn 15, max-src-conn-rate 20/120, overload <bruteforcemail> flush global)
#
table <badip> persist  file "/etc/badip"
block drop in quick from <badip>
```

Ось детальний опис кожного параметра конфігурації:

– `ext_if = "em0"` - змінна `ext_if` використовується для визначення зовнішнього інтерфейсу, тобто інтерфейсу;

– `int_if = "em1"` - Змінна `int_if` використовується для визначення внутрішнього інтерфейсу, тобто інтерфейсу, який з'єднує сервер з внутрішньою мережею;

– `localnet = $int_if:network` - змінна `localnet` використовується для визначення внутрішньої мережі, з якою пов'язаний внутрішній інтерфейс. За допомогою `$int_if:network` вона отримує значення мережі, до якої належить внутрішній інтерфейс;

– `set skip on lo0` - цей параметр вказує PF пропускати всі пакети, що надходять через інтерфейс lo0 (локальний інтерфейс). Це потрібно для забезпечення локального зв'язку на самому сервері, і такі пакети не підлягають обробці PF;

– `set limit { states 40000, frags 40000, src-nodes 4000 }` - цей параметр встановлює обмеження на різні аспекти обробки пакетів. В даному випадку встановлені наступні обмеження:

– `states` - максимальна кількість активних мережевих станів, які PF може відслідковувати одночасно;

– `frags` - максимальна кількість мережевих фрагментів, які PF може обробляти одночасно;

– `src-nodes` - максимальна кількість унікальних джерел (IP-адрес), які PF може відслідковувати одночасно.

– `set optimization aggressive` - цей рядок вказує PF використовувати агресивні алгоритми оптимізації для покращення продуктивності та обробки трафіку. Це може підвищити навантаження на процесор, але при цьому забезпечується краща пропускна здатність та швидкодія PF;

– `scrub in all` - цей параметр включає очищення (scrubbing) всіх вхідних пакетів. Очищення включає перевірку та виправлення неправильно сформованих пакетів, а також вилучення небезпечних частин пакетів, що можуть бути використані для атак;

— table <ddos> persist - ця команда створює таблицю з назвою <ddos> і позначає її як постійну (persist). Це означає, що таблиця буде зберігати свої дані між перезавантаженнями системи;

— block drop in quick from <ddos> - це правило блокує (drop) всі пакети, що надходять з IP-адрес, які містяться в таблиці <ddos>. Ключове слово quick вказує на прохід правила без подальшої обробки інших правил;

— pass in on $ext_if proto tcp from any to $ext_if port {http, https} - це вказує PF пропустити (pass) вхідні TCP-пакети, які приходять з будь-якої адреси (from any) і призначені для зовнішнього інтерфейсу (to $ext_if) на порти HTTP та HTTPS (port {http, https});

— flags S/SA synproxy state - цей параметр встановлює прапорці (flags) для пакетів, що відповідають цьому правилу. S/SA вказує, що пакет повинен мати встановлені прапорці SYN та ACK. Крім того, використовується synproxy, що дозволяє використовувати проксі для обробки SYN-флуду і зменшення навантаження на сервер;

— (source-track rule, max-src-nodes 1000, max-src-conn 1000, max-src-states 10000, max-src-conn-rate 100/60, overload <ddos> flush global) - ця частина правила включає розширені параметри для контролю ресурсів і захисту від DDoS-атак;

— source-track rule - параметр, який включає відстеження джерела пакетів, що відповідають цьому правилу;

— max-src-nodes 1000 – встановлює максимальну кількість джерел пакетів на 1000;

— max-src-conn 100 - встановлює максимальну кількість одночасних з'єднань від одного джерела на 100;

— max-src-states 1000 - встановлює максимальну кількість станів з'єднання для одного джерела на 1000;

— max-src-conn-rate 100/60 - встановлює максимальну швидкість з'єднань від одного джерела на 100 з'єднань за 60 секунд;

— overload <ddos> flush global - цей параметр використовує таблицю <ddos> для перенаправлення заблокованих пакетів відповідно до налаштувань

таблиці. Параметр `flush global` вказує на видалення заблокованих пакетів з глобальної черги;

– `table <badip> persist` - ця команда створює таблицю з назвою `<badip>` і позначає її як постійну (`persist`). Це означає, що таблиця буде зберігати свої дані між перезавантаженнями системи;

– `file "/etc/badip"` - цей параметр вказує на шлях до файлу `/etc/badip`, в якому знаходяться IP-адреси, які завантажаться в таблицю `<badip>` після перезавантаження операційної системи або сервісу PF. Цей файл містить список IP-адрес, розділених новим рядком;

– `block drop in quick from <badip>` - це правило блокує (`drop`) всі вхідні пакети, які мають джерело з IP-адресами, що містяться в таблиці `<badip>`. Ключове слово `quick` вказує на швидкий прохід правила без подальшої обробки інших правил. Ці правила дозволяють блокувати вхідні пакети з певних IP-адрес, які знаходяться в таблиці <*badip*>. Файл /etc/badip можна регулярно оновлювати, автоматично додаючи нові IP-адреси з таблиць `<bruteforcemail>` та `<bruteforcessh>` або в ручному режимі, щоб забезпечити ефективний захист від потенційно шкідливих джерел. Також можна видаляти застарілі записи.

Здійснимо brute force атаку згідно пункту 3.2 . Хоча в пункті 3.2 Hydra підібрала пароль за лічені секунди. Як можна побачити з рисунку 3.9 підбір паролю користувача пройшов невдало навіть попри те що ми навмисно встановили пароль 123456 та явно вказали ім’я користав vlad в параметрах запуску Hydra.

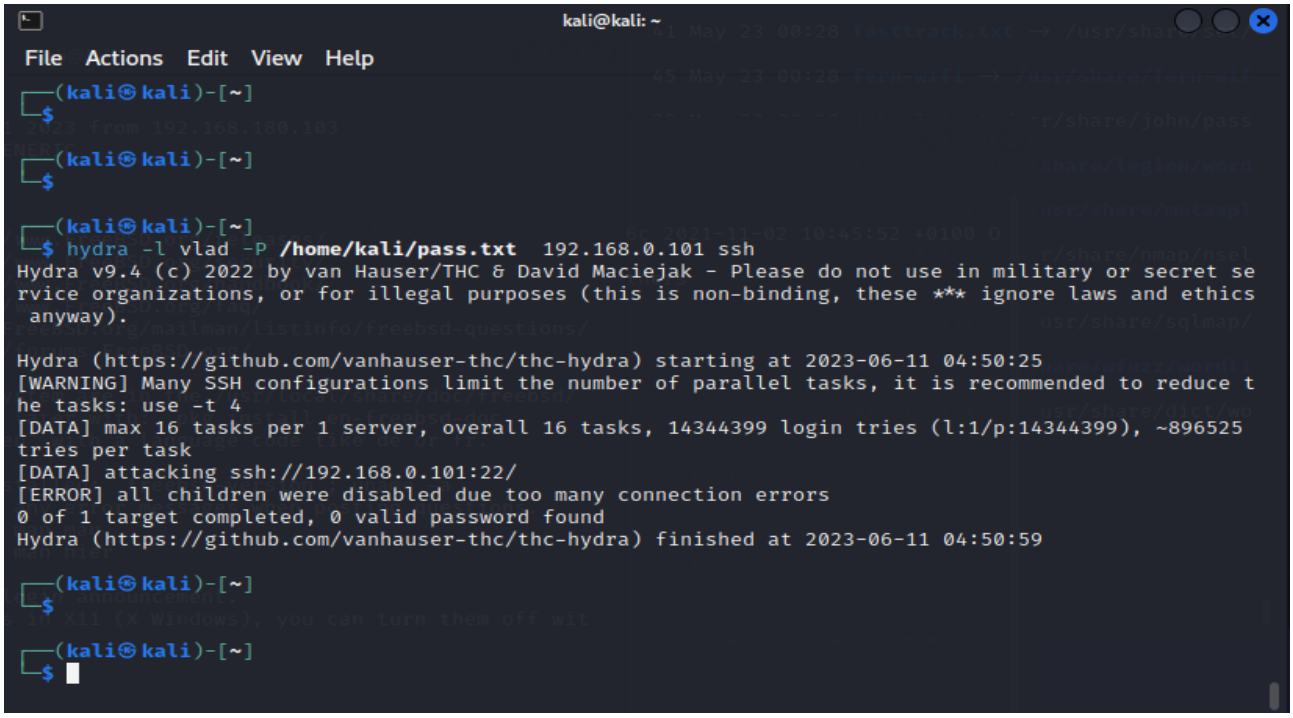

Рисунок 3.9 – Результати невдалого проведення brute force атаки на SSH-сервер

Це означає що захист від brute force атак за допомогою PF працює надійно та швидко.

Також підтвердженням того що захист спрацював надійно є вивід команди `pfctl -t bruteforcessh -T show` (Рис.3.10).

```
root@router-unix:/etc# pfctl -t bruteforcessh -T show
   192.168.0.103
root@router-unix:/etc#
```

Рисунок 3.10 – Вміст `bruteforcessh` таблиці

Цей вивід вказує на те що IP адреса Kali Linux додана в таблицю `bruteforcessh` і є заблокована.

На рисунку 3.11 показано вивід команди `cat /var/log/auth.log` при невдалій спробі brute force атаки.

```
Jun 11 11:50:26 router-unix sshd[94234]: Received disconnect from 192.168.0.103 port 53262:11: Bye Bye [preauth]
Jun 11 11:50:26 router-unix sshd[94234]: Disconnected from authenticating user vlad 192.168.0.103 port 53262 [preauth]
Jun 11 11:51:46 router-unix sshd[94237]: Fssh_ssh_dispatch_run_fatal: Connection from 192.168.0.103 port 53288: Operation timed out [preauth]
Jun 11 11:51:46 router-unix sshd[94236]: Fssh_ssh_dispatch_run_fatal: Connection from 192.168.0.103 port 53272: Operation timed out [preauth]
Jun 11 11:51:46 router-unix sshd[94238]: Fssh_ssh_dispatch_run_fatal: Connection from 192.168.0.103 port 53294: Operation timed out [preauth]
root@router-unix:/etc #
```

Рисунок 3.11 – Вивід команди cat /var/log/auth.log при невдалій спробі brute force атаки

Виділений запис в фалі auth.log означає наступне:

- `Jun 11 11:51:46` – час запису події;
- `router-unix` – ім'я хоста або пристрою;
- `sshd[94237]` – ідентифікатор процесу sshd, який обробляв з'єднання SSH;
- `Fssh_ssh_dispatch_run_fatal` - повідомлення про помилку;
- `Connection from 192.168.0.103 port 53288` - з'єднання з IP-адреси 192.168.0.103 і порту 53288;
- `Operation timed out` - відповідь на запит зайняла занадто багато часу;
- `[preauth]` – подія сталася до аутентифікації користувача (pre-authentication).

Запис свідчить про з'єднання SSH з IP-адреси 192.168.0.103, проте з'єднання було припинено через тайм-аут. Це сталось по причині того, що IP адреса Kali Linux була додана в таблицю `bruteforcessh` і заблокована.

Брандмауер PF надає гнучкі та досить прості механізми для налаштування правил фільтрації пакетів і контролю доступу. Це дозволяє точно визначати, які типи трафіку допускаються і які блокуються.

PF дозволяє налаштовувати правила, спеціально призначені для виявлення і обмеження brute force атак. Можна встановити обмеження на кількість спроб аутентифікації з одного джерела, швидкість спроб або заблокувати IP-адреси, що здійснюють надмірну кількість спроб.

За допомогою PF можна виявляти та пом'якшувати DDoS атаки. Можна встановити правила, які відслідковують велику кількість одночасних з'єднань або велику швидкість нових з'єднань з одного джерела і блокують такий трафік. Крім того, можна використовувати SYNPROXY для захисту від DDoS атак, які використовують SYN flood.

PF підтримує динамічні правила, які дозволяють реагувати на зміни трафіку в реальному часі. Це дозволяє ефективно виявляти і реагувати на атаки, забезпечуючи надійний захист мережі.

Отже, використання брандмауера PF дозволяє успішно захистити мережу від brute force атак і пом'якшити DDoS атаки. Його гнучкість, простота налаштування та підтримка динамічних правил роблять його ефективним інструментом мережевої безпеки.

4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Долікарська допомога при масивній зовнішній кровотечі

Масивна зовнішня кровотеча є надзвичайно небезпечним станом, який може призвести до серйозної загрози життю постраждалої особи. Невідкладна надання домедичної допомоги може врятувати життя і запобігти подальшій крововтраті.

Домедична допомога постраждалим при кровотечі є важливою процедурою, яку можуть виконувати особи без медичної освіти.

Ознаками масивної зовнішньої кровотечі є будь-що з нижченаведеного:

- швидке, інтенсивне витікання крові з рани;
- пульсуючий характер кровотечі (кров б’є фонтаном);
- пляма крові біля постраждалого, яка швидко збільшується;
- значне просякнення одягу постраждалого кров’ю;
- порушення або втрата свідомості у постраждалого без ознак черепно-мозкової травми, при наявності зовнішньої кровотечі;
- бліда шкіра, холодні кінцівки тощо, при наявності зовнішньої кровотечі.

Наказ Міністерства охорони здоров'я України від 09.03.2022 р. № 441 " Про затвердження порядків надання домедичної допомоги особам при невідкладних станах" встановлює порядки надання домедичної допомоги постраждалим при при масивній зовнішній кровотечі. У цьому порядку термін “тепловий удар” вживаються у такому значенні - невідкладний стан, викликаний дією високої температури навколишнього середовища, що спричиняє системні розлади у постраждалого [11].

Надання домедичної допомоги постраждалим при масивній зовнішній кровотечі передбачає такі кроки:

1) Переконайтесь що небезпеки для вас немає.

2) Закличте оточуючих на допомогу. Якщо є кілька свідків, зверніться до конкретної особи, щоб вона надала допомогу.

3) Перед початком надання допомоги, за можливості, захистіть себе за допомогою індивідуальних засобів захисту, таких як рукавички, маска і захист для очей.

4) Якщо є кровотеча з рани на кінцівці, і вона видно:

а) Здійсніть максимальний тиск на рану руками;

б) Накладіть пов'язку, що чинитиме тиск на рану, і оцініть її ефективність;

в) Якщо кровотеча зупинилась, заспокойте постраждалого, викличте екстрену медичну допомогу та слідуйте вказівкам диспетчера;

г) Якщо кровотеча не зупинилась, накладіть кровоспинний джгут на відстані 5-7 см вище рани. Уникайте накладання джгута безпосередньо на суглоби ліктя або коліна.

ґ) Перевірте ефективність накладеного кровоспинного джгута. Якщо кровотеча зупинилась, зафіксуйте час накладання на джгуті або запишіть його на видимому місці. Якщо неможливо записати час, повідомте медичному персоналу. Якщо вмієте, перевірте пульс нижче джгута. Якщо пульс є, здійсніть додатковий тиск кровоспинним джгутом або накладіть ще один джгут вище. Якщо кровотеча не зупинилась, продовжуйте надавати прямий тиск на рану до прибуття медичної бригади або тампонуйте рану.

5) При кровотечі з рани кінцівки без можливості її чіткої візуалізації:

а) Накладіть кровоспинний джгут якомога вище на кінцівку;

б) Заспокойте постраждалого та поясніть подальші кроки;

в) Якщо можливо, розріжте одяг на кінцівці;

г) Оцініть ефективність накладання кровоспинного джгута:

Якщо кровотеча зупинилась, зафіксуйте точний час накладання джгута на самому джгуті або видимому місці. Якщо неможливо зафіксувати час, повідомте медичному персоналу та переконайтеся, що ця інформація буде внесена до медичних записів.

Якщо у вас є навик перевірки пульсу на кінцівці нижче джгута, перевірте його. Якщо пульс присутній, збільште тиск кровоспинного джгута або накладіть додатковий джгут.

Якщо кровотеча не зупинилась, збільште тиск на кровоспинному джгуті або накладіть ще один джгут в залежності від місця рани. Якщо другий джгут не є ефективним або неможливо його накласти, продовжуйте чинити прямий тиск на рану руками до прибуття медичної бригади або тампонуйте рану.

Не знімайте або не послабляйте кровоспинний джгут до прибуття медичної бригади.

6) При кровотечі з рани, розташованої в пахвових ділянках, сідницях або основі шиї:

а) Застосуйте максимальний тиск на рану;

б) Заспокойте постраждалого та поясніть подальші дії;

в) Тампонуйте рану тугим гемостатичним засобом або марлевим бинтом. Після тампонування продовжуйте здійсніть прямий тиск на рану протягом 3 хвилин (з гемостатиком) або 10 хвилин (з марлевим бинтом);

г) Оцініть ефективність тампонування рани.

Якщо кровотеча зупинилась, продовжуйте надавати іншу домедичну допомогу, передбачену процедурою.

Якщо кровотеча не зупинилась, спробуйте повторно тампонувати рану. Якщо це неможливо, продовжуйте чинити максимальний тиск на рану руками до прибуття швидкої медичної допомоги.

Це загальна послідовність дій, яку слід виконати, але завжди важливо дотримуватись інструкцій медичних фахівців та адаптувати допомогу до конкретної ситуації. Виконання цих кроків допоможе забезпечити постраждалому першу необхідну допомогу та зберегти його життя до прибуття медичних фахівців.

4.2 Зниження стресу та покращення психологічного благополуччя працівників

Психофізіологічне розвантаження є одним з варіантів зменшення стресу.

Дана практика включає в себе застосування методу аутогенного тренування, що передбачає свідоме використання комплексу прийомів психічної саморегуляції та виконання простих фізичних вправ зі словесним самонавіюванням. Основна увага приділяється розслабленню м'язів (релаксації).

Під час сеансів психофізіологічного розвантаження рекомендується використовувати три періоди, що відповідають фазам відновлення:

Перший період - абстрагування від виробничого середовища, що відповідає фазі залишкового збудження. В цей час відтворюється повільна мелодійна музика та звуки пташиного співу. Працівники знаходять зручну позу та психологічно готуються до наступних періодів.

Другий період - заспокоєння, що відповідає фазі відновлювального гальмування. Показуються фотослайди з зображеннями природи, таких як квітучі луки, березові гаї, ставки і т.д. Звуковий супровід включає спокійну музику та заспокійливі формули аутогенного тренування.

Третій період - активізація, що відповідає фазі підвищеної збудженості. Спочатку світло повністю вимикається, а потім на екрані появляється червона пляма, розмір і яскравість якої поступово збільшуються. В кінці періоду звучить бадьора музика, а працівники виконують мобілізуючі формули аутогенного тренування, попередньо зробивши глибоке вдихання та видихання.

Сеанси психофізіологічного розвантаження можуть проводитись за єдиною програмою через індивідуальні навушники і складатись із двох періодів по 5 хвилин кожний: повне розслаблення та активізація працездатності. При необхідності, на фоні музики можуть використовуватись фрази, що сприяють відпочинку, покращенню самопочуття та бадьорості на заключному етапі. Після сеансів психофізіологічного розвантаження працівники відчувають зменшення втоми, з'являється бадьорість та гарний настрій, а загальний стан помітно поліпшується.

Додатково до сеансів психофізіологічного розвантаження, працівники також можуть скористатись іншими методами для зниження стресу та покращення психологічного благополуччя.

Одним з ефективних підходів є впровадження регулярних перерв під час робочого дня. Це можуть бути короткі паузи, під час яких працівники займаються розслаблюючими вправами, дихальними техніками або просто відпочивають. Це допомагає знизити напругу і покращити фокусування під час робочого процесу.

Також важливо створити комфортне робоче середовище для працівників. Це може включати забезпечення комфортних стільців і столів, добре освітлення та

достатню вентиляцію. Природне освітлення та наявність рослин у приміщенні також можуть позитивно вплинути на настрій та самопочуття працівників.

Підтримка від керівництва та колег також має важливе значення. Створення сприятливого та підтримуючого робочого середовища, де працівники можуть відчути підтримку та співпрацю, сприяє зниженню стресу та покращує загальний настрій в колективі.

Крім того, особисте самоуправління і здібність до саморегуляції є важливими навичками для працівників. Це включає уміння регулювати власні емоції, реагувати на стресові ситуації та знаходити способи їх подолання, наприклад, за допомогою медитації, йоги або інших релаксаційних технік.

Усі ці підходи сприяють створенню здорової та продуктивної робочої атмосфери, де працівники можуть ефективно керувати стресом, забезпечуючи своє фізичне та емоційне благополуччя.

ВИСНОВКИ

У ході кваліфікаційної роботи було досліджено важливість та роль брандмауерів у забезпеченні безпеки мережі та захисту комп'ютерних систем, використовуючи операційну систему FreeBSD як основу. Робота висвітлює ключові принципи функціонування брандмауерів, включаючи фільтрацію пакетів, контроль доступу та виявлення вторгнень.

Було показано, як встановлювати та налаштовувати брандмауери, які включають IPFW, PF та IPFilter (IPF). Робота також представляє різні сценарії використання брандмауерів у реальних умовах, включаючи налаштування фільтрації пакетів, налаштування правил контролю доступу та виявлення вторгнень.

На конкретних прикладах було доведено що PF є потужним інструментом мережевої безпеки. Використовуючи PF можна встановлювати тонко налаштовувані правила для керування трафіком мережі, включаючи блокування, обмеження швидкості, перенаправлення та багато іншого.

Особливо цінним є те, що PF дозволяє встановлювати динамічні правила, які реагують на зміни трафіку в реальному часі. Наприклад, ви можете використовувати PF для захисту від атак brute force та для пом'якшення DDoS-атак, відслідковуючи велику кількість одночасних з'єднань або велику швидкість нових з'єднань від кожного джерела. Додатково, можливість використання SYNPROXY дозволяє захистити сервер від DDoS-атак, використовуючи SYN flood, що є поширеним способом атаки.

Отримані результати свідчать про успішну реалізацію поставлених завдань та досягнення поставленої мети..

Подані у роботі матеріали можуть бути корисними для адміністраторів мереж, системних інженерів, а також у навчальних цілях. Дослідження, проведене у цій роботі, сприяє розвитку мережевої безпеки, поліпшенню методів захисту та забезпеченню безпеки мережевих інфраструктур. В результаті цієї роботи було досягнуто кращого розуміння брандмауерів у контексті FreeBSD та їх впливу на загальну безпеку мереж.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Брандмауер [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. –Режим доступу: https://www.eset.com/ua/support/information/entsiklopediya-ugroz/brandmauer/
2. What Is a Firewall [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – Режим доступу: https://www.cisco.com/c/en/us/products/security/firewalls/what-is-a-firewall.html
3. FreeBSD Documentation [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – Режим доступу: https://docs.freebsd.org/en/
4. Firewalls [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – Режим доступу: https://docs.freebsd.org/en/books/handbook/firewalls/
5. Packet filter configuration file [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – https://man.freebsd.org/cgi/man.cgi?pf.conf(5)
6. IPFW [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – Режим доступу: https://man.freebsd.org/cgi/man.cgi?ipfw(8)
7. Ipfilter [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – Режим доступу: https://man.freebsd.org/cgi/man.cgi?ipfilter
8. Ipnat [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – Режим доступу: https://man.freebsd.org/cgi/man.cgi?query=ipnat
9. Kali Linux Official Documentation [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – Режим доступу: https://www.kali.org/docs/
10. Kali Linux Tool Documentation: hydra [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – Режим доступу: https://www.kali.org/tools/hydra/
11. Про затвердження порядків надання домедичної допомоги особам при невідкладних станах [Електронний ресурс] : [Інтернет-портал]. – Електронні дані. – Режим доступу: https://zakon.rada.gov.ua/laws/show/z0356-22#n769