Харківський національний університет імені В.Н. Каразіна

Факультет комп'ютерних наук

Безпека інформаційних систем і технологій

«Допущено до захисту»

Зав.кафедрою БІСТ

Сватовський І.І._____

« » червня 2023р.

Пояснювальна записка

до кваліфікаційної роботи бакалавра

спеціальність: 125 Кібербезпека

на тему: «Розробка міжмережевого екрану виділеного серверу на базі вільного програмного забезпечення»

оцінка « »

Голова ЕК

Лемешко О.В._____

Керівник д.т.н. Олійников Р.В.
(прізвище та ініціали/підпис)

Рецензент PhD з комп. наук Родінко М.Ю.
(прізвище та ініціали/підпис)

Виконавець студент групи КБ-41

Стешенко І.Є.
(прізвище та ініціали/підпис)

Харків – 2023

РЕФЕРАТ

Дипломна робота бакалаврського рівня вищої освіти на тему «Розробка міжмережевого екрану виділеного серверу на базі вільного програмного забезпечення» містить 68 сторінок, 1 таблицю, 29 рисунків, 2 додатки та використано 26 джерел.

Метою роботи є аналіз Інтернет мережі, огляд та розробка власного міжмережевого екрану на базі виділеного серверу, як один із способів захисту інформації в мережі.

Протягом написання дипломної роботи розглядається інформаційно-телекомунікаційна система, види мереж та чим вони відрізняються. Розглядається модель існуючих загроз для безпеки в мережі. Також розбирається модель зловмисника в мережі та його можливості. Можно побачити декілька рішень проблеми безпеки в мережі, та короткий опис кожного з них. Проводиться дослідження і повний розбір такої технології захисту в мережі як міжмережевий екран, розглядаються його види, можливості та особливості. Розглядається улітіта командного рядка для створення правил фільтрації трафіку міжмережевого екрану iptables та її можливості. Створюється власний міжмережевий екран на базі ОС Linux, написано власні правила фільтрації трафіку за допомогою утиліти iptables. Всі налаштування міжмережевого екрану протестовано, та зроблені відповідні висновки.

Результат даної роботи може використовуватись при налаштуванні локальних мереж в невеликих компаніях або організаціях які працють в Інтернеті, також можна налаштовувати та захищати власну локальну мережу вдома.

Подальші напрямки в дослідженні даної теми є удосконалення й прибільшення правил фільтрації для міжмережевого екрану, а також розширення його застосування.

Ключові слова: БЕЗПЕКА МЕРЕЖІ, ЗАХИСТ, МІЖМЕРЕЖЕВИЙ ЕКРАН, ФІЛЬТРАЦІЯ ПАКЕТІВ, IPTABLES, СЕРВЕР APACHE2.

ABSTRACT

The bachelor's thesis of university degree on the topic "Development of an inter-network screen of a dedicated server based on free software" contains 68 pages, 1 table, 29 figures, 2 appendices and 26 sources were used.

The purpose of the work is the analysis of the Internet network, review and development of its own inter-network screen based on a dedicated server, as a way to protect information in the network.

During the writing of the thesis, the information and telecommunication system, types of networks and how they differ are considered. The model of existing threats to network security is considered. The model of the attacker in the network and its capabilities are also analyzed. It can be seen several solutions to the network security problem, and a brief description of each of them. A study and full analysis of such a network protection technology as a network screen is conducted, its types, possibilities and features are considered. The iptables command-line utility for creating firewall traffic filtering rules and its capabilities are discussed. A custom firewall based on the Linux OS is created, custom traffic filtering rules are written using the iptables utility. All firewall settings have been tested and appropriate conclusions drawn.

The result of this work can be used when configuring local networks in small companies or organizations that work on the Internet, it can also be configured and protect own local network at home.

Further directions in the research of this topic are the improvement and increase of the filtering rules for the inter-network firewall, as well as the expansion of its application.

Keywords: NETWORK SECURITY, SECURITY, INTERNET FIREWALL, PACKET FILTERING, IPTABLES, APACHE2 SERVER.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ ТА СИМВОЛІВ

АС	-	Автоматизовані системи
ІзОД	-	Інформація з обмеженим доступом
ІТС	-	Інформаційно-технологічний супровід
КСЗІ	-	Комплексна система захисту інформації
МЕ	-	Міжмережевий екран
НД	-	Несанкціонований доступ
BGP	-	Border Gateway Protocol
DoS	-	Denial-of-Service
DNS	-	Domain Name System
FTP	-	File Transfer Protocol
HTTP	-	Hyper Text Transfer Protocol
IDS	-	Intrusion Detection System
IPS	-	Intrusion Prevention System
ICMP	-	Internet Control Message Protocol
IP-address	-	Internet Protocol address
LAN	-	Local Area Network
NAT	-	Network Address Translation
OSI	-	Open Systems Interconnection
OSPF	-	Open Shortest Path First
QoS	-	Quality of Service
TCP	-	Transmission Control Protocol
UDP	-	User Datagram Protocol
VPN	-	Virtual Privat Network
WAN	-	Wide Area Network

ВСТУП

З розвитком технологій, стрімким зростанням глобальної мережі Інтернет та бурхливим розвитком інформаційних технологій сформувалося інформаційне середовище, яке впливає на всі сфери діяльності людини. Нові технологічні можливості сприяють поширенню інформації, підвищенню ефективності виробничих процесів та розширенню ділових відносин. Проте, незважаючи на інтенсивний розвиток комп'ютерних засобів та інформаційних технологій, вразливість сучасних інформаційних систем і комп'ютерних мереж, на жаль, не зменшується. Тому проблеми забезпечення інформаційної безпеки привертають пильну увагу як фахівців у галузі комп’ютерних систем і мереж, так і численних користувачів, у тому числі компаній, що працюють у сфері електронного бізнесу [1].

Сьогодні безпечна мережа стала потребою будь-якої організації. Щодня загрози для інформації та її безпеки зростають, дротові або бездротові послуги інтернету стають більш небезпечними та ненадійними. Потреба також спрямована на такі сфери, як оборона, де безпечний доступ до ресурсів є ключовим питанням, пов’язаним з інформаційною безпекою. У сучасному світі комп’ютерні мережі використовуються в різних сферах життя, включаючи бізнес, освіту, науку, медицину та багато інших галузей. Однак, разом із зростанням залежності від мережі зростає і загроза її безпеці.

Основами мережевої безпеки є захист робочих місць від шпигуньского та шкідливого програмного забезпечення. Також це забезпечення багаторівневого захисту даних, запобігання різним атакам, розбиваючи інформації на чисельні частини та подальше їх шифрування. Безпека – це насамперед питання транспортування інформації мережею, а саме потреба в створенні нелазелжних шляхів для передачі даних. Щодня кількість користувачів ПК, смарфтонів та

Інетрнету збільшується, а отже й збільшується можливість потенційних атак на користувачів глобальних мереж.

Кожного року компанії втрачають багато коштів саме через недостатню захищеність своїх комп’ютерів та інформації, тільки у 2023 році згідно з розрахунками Canalys, компанії з досліджень у сфері технологій та аналітики ринку, витрати на кібербезпеку зростуть на 13% [2]. Через це приходе розуміння що потрібно збільшувати фінансування сфери безпеки своїх даних і поліпшення засобів захисту. Це питання наразі гостро постає не тільки в роботі окремих компаній або інших об’єднань, а й цілиї країн які також зрозуміли важливість цієї проблеми.

Одним з основних засобів захисту мережі є міжмережевий екран (англ. Firewall). Його основна функція – контроль доступу до мережі та забезпечення безпеки мережевого трафіку [3].

Міжмережевий екран (МЕ) працює на основі правил, які визначають, який трафік дозволено для проходження в мережі, а який заблоковано. Ці правила можна налаштувати для фільтрації трафіку на основі різних критеріїв, таких як IP-адреса, порти, протоколи тощо. Крім того, МЕ може використовувати різні методи автентифікації, шифрування та інші методи безпеки, щоб запобігти несанкціонованому доступу до мережі.

Використання міжмережевих екранів для безпеки мережі має кілька важливих переваг. По-перше, вони дозволяють виявляти та блокувати небажаний мережевий трафік, наприклад віруси, хробаки, шпигунські програми тощо. По-друге, вони забезпечують контроль і моніторинг мережевого трафіку, що допомагає виявляти атаки та інші загрози безпеці мережі.

Підсумовуючи вищесказане, можна зазначити, що комерційні рішення по захисту інформації дуже затратні, а вільні рішення потребують великої і клопіткої роботи для налаштування. В даній роботі буде розроблятися саме власне налаштування міжмережевого екрану для користування для захисту веб серверу.

1 ОПИС ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ І МОДЕЛІ ЗАГРОЗ

1.1 Опис інформаційної технології

Інформаційно-телекомунікаційна система - це технологічна система, в якій використовуються та показують свою ефективність технології для створення, зберігання та передачі інформації з використанням програмних і апаратних засобів та додатків для досягнення цього ефекту [4]. Всі файли, додатки та програмне забезпечення є спільною інформацією в мережі. Переваги мережевого зв'язку з точки зору безпеки, ефективності, контролю та економічності гостро відчуваються, оскільки він забезпечує широкий спектр взаємодій між користувачами. В основному, мережі складаються з апаратних компонентів, таких як комп'ютери, комутатори та маршрутизатори, які відіграють важливу роль у передачі даних з одного місця в інше за допомогою різних технологій радіохвиль і дротових з'єднань.

Наразі існує декілька типів мереж, і найпоширенішими є локальна мережа (LAN) і глобальна мережа (WAN) [5].

LAN (Local Area Network) – це локальна мережа, а саме – це невелика мережа, часто всередині дому чи підприємства або, можливо, у більшому середовищі, як корпоративний офіс. Пристрої в локальній мережі часто використовують інфраструктуру локальної мережі для підключення до загальнодоступного Інтернету, але вони часто можуть спілкуватися один з одним безпосередньо через локальну мережу швидше. Наприклад, зазвичай не потрібно надсилати файл у загальнодоступний Інтернет, щоб отримати його на принтер у тій же локальній мережі. Локальна мережа може використовувати бездротовий зв’язок, або дротове з’єднання.

WAN (Wide Area Network) – це широкодоступна, або глобальна комп'ютерна мережа. Це важлива комп’ютерна мережа, яка охоплює велику географічну

територію. WAN порівняно набагато більша за LAN і відносно дорожча. Через вартість і складне налаштування WAN зазвичай не належать одній організації. Глобальні мережі створюються за допомогою кількох локальних мереж, з'єднаних телефонними лініями або радіохвилями. Як правило, ці типи мереж організовані за допомогою телекомунікаційних каналів високого класу. Інтернет, різновид загальнодоступної мережі, є прикладом найбільшої глобальної мережі.

Безпека інформації в телекомунікаційних мережах – це спроможність цих самих мереж забезпечити інформацію від знищення, перекручення, її блокування, від несанкціонованого витоку, порушення цілісності або від порушення встановленого порядку маршрутизації [6].

Для прикладу можна уявити велику компанію, яка в основному працює в мережі, має власний офіс, а отже має й власну локальну мережу, кожен працівник має власний ПК, який під'єднаний до неї. Всі працівники обмінюються між собою інформацією через мережу, також переглядають Інтернет в пошуках нової потрібної інформації, також завантажують різноманітні файли. Саме на цьому етапі зловмисники й приступають до роботи, саме основна загроза безпеці компанії і всій її конфіденційній інформації йде через недостатню безпеку локальної мережі.

Також можна привести в приклад компанію яка має власний веб сервер, на якому створений електронний довідник, через який всі працівники отримують цінну інформацію, тобто він виступає сховищем всієї інформації компанії.

Рішенням наведених вище проблем безпеки може стати міжмережевий екран, який повністю контролює трафік, як вихідний так і той, що надходить на сервер або ПК в локальній мережі. Є два варіанти такого захисту, або комерційний, який потребує великі кошти на покупку й обслуговування, або вільне рішення яке потребує великої роботи задля налаштування правил міжмережевого екрану. Далі буде розглянуто саму будову мережі.

Використання комп'ютерних мереж означає, перш за все, використання моделі ISO/OSI, стека TCP/IP для передачі даних і веб-технології для їх представлення.

Модель ISO/OSI (International Organization for Standardization/Open Systems Interconnection) є стандартною моделлю, яка описує, як комп'ютерні мережі повинні взаємодіяти між собою [5]. Вона складається з семи рівнів, кожен з яких виконує певні функції для обробки і передачі даних. Основними рівнями моделі ISO/OSI є фізичний, канальний, мережевий, транспортний, сеансовий, рівень представлення та прикладний рівні. Кожен рівень відповідає за певну функцію для забезпечення ефективної та надійної передачі даних через мережу.Модель ISO/OSI є важливим інструментом для розробки та впровадження стандартів мережевої комунікації. Модель зображена на рисунку 1.1.

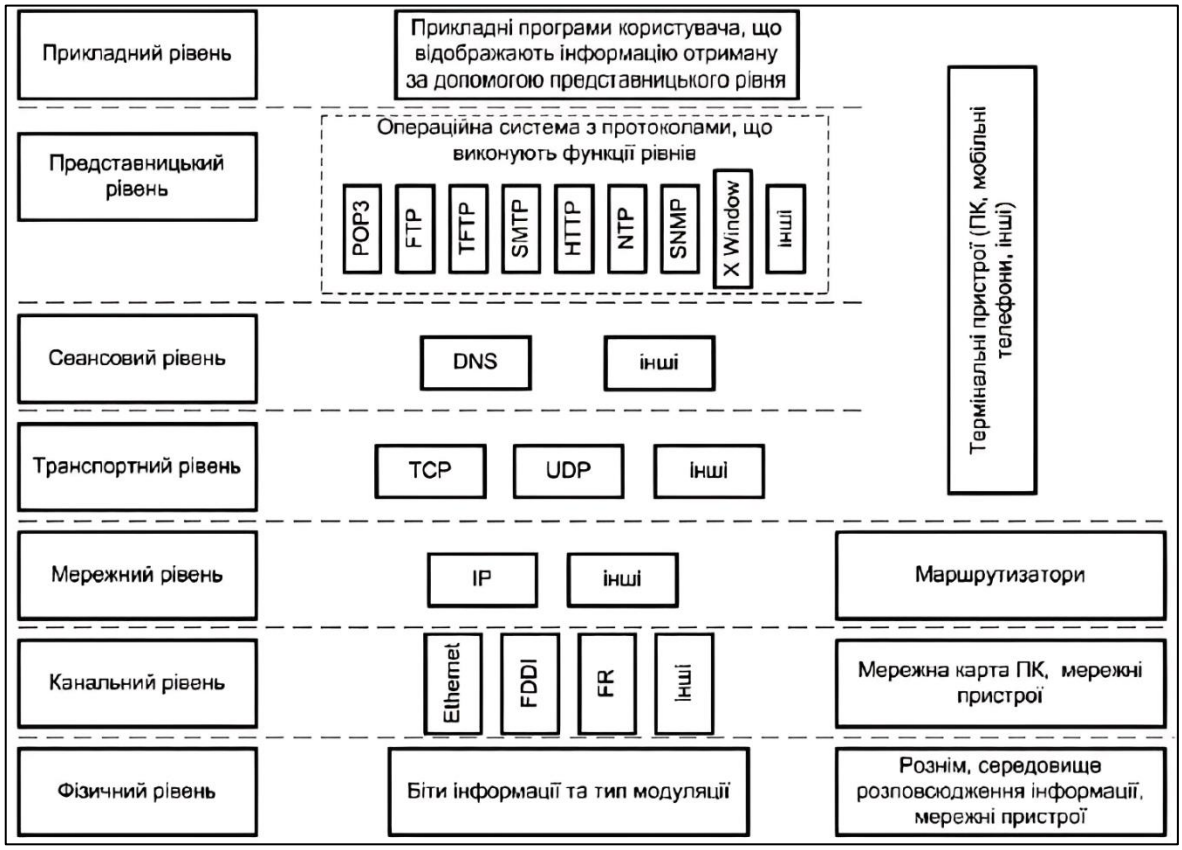

Рисунок 1.1 – Реалізація моделі ISO/OSI

- Прикладний рівень (Application Layer). Цей рівень представляє вищий рівень абстракції і визначає специфічні протоколи та служби, які використовуються для комунікації між програмами або службами. На цьому рівні відбувається взаємодія з додатками, такими як веб-браузери, електронна пошта, файлові сервіси тощо.
- Представницький рівень (Presentation Layer). Цей рівень відповідає за перетворення даних у формат, який може бути розуміним для отримувача. Він забезпечує стандартизований спосіб представлення даних, щоб різні системи могли взаємодіяти.
- Сеансовий рівень (Session Layer). Цей рівень встановлює, підтримує та закриває зв'язок між двома пристроями у мережі. Він відповідає за управління сеансами зв'язку, синхронізацію та керування обміном даними між пристроями.
- Транспортний рівень (Transport Layer). Цей рівень забезпечує надійну доставку даних відправнику до одержувача. Він розділяє дані на менші частини, які потім передаються через мережу, і переконується, що всі частини були доставлені успішно та в правильному порядку.
- Мережевий рівень (Network Layer). Цей рівень відповідає за маршрутизацію пакетів даних у мережі. Він визначає, які шляхи повинні бути вибрані для передачі даних від одного пристрою до іншого.
- Канальний рівень (Data Link Layer). Цей рівень забезпечує безперервну передачу даних між пристроями в мережі. Він використовує фрейми (кадри) для передачі даних і включає методи виявлення та корекції помилок.
- Фізичний рівень (Physical Layer). Цей рівень визначає фізичні аспекти передачі даних, такі як електричні сигнали, кабелі, роз'єми та фізичні характеристики мережевих пристроїв. Він встановлює правила для передачі бітів через фізичну мережу.

Стек TCP/IP (Transmission Control Protocol/Internet Protocol) є набором протоколів, які використовуються для забезпечення комунікації в Інтернеті та в багатьох

локальних мережах [7]. Вона була розроблена для стандартизації передачі даних між комп'ютерами та різними мережевими пристроями. Модель TCP/IP складається з чотирьох рівнів, які описують різні аспекти комунікації . Короткий огляд кожного рівня:

- Рівень мережного інтерфейсу (Network Access Layer). Цей рівень відповідає за фізичний доступ до мережі та передачу даних через нього. Він включає різні протоколи, такі як Ethernet, Wi-Fi, PPP (Point-to-Point Protocol) та інші, які забезпечують передачу бітів даних через мережеве середовище.
- Міжмережевий рівень (Internet Layer). Цей рівень відповідає за маршрутизацію пакетів даних через мережу. Він використовує протокол Internet Protocol (IP) для адресації та маршрутизації пакетів у мережі. Крім IP, на цьому рівні також використовуються протоколи, такі як ICMP (Internet Control Message Protocol) для обміну повідомленнями про стан мережі та протоколи маршрутизації, такі як OSPF (Open Shortest Path First) та BGP (Border Gateway Protocol).
- Транспортний рівень (Transport Layer). Цей рівень відповідає за надійну доставку даних між програмами або службами, що працюють на різних пристроях в мережі. Два найвідоміші протоколи на цьому рівні – це Transmission Control Protocol (TCP) та User Datagram Protocol (UDP). TCP забезпечує з'єднання з контролем помилок та контролем потоку, тоді як UDP надає простий передачу пакетів без підтримки з'єднання.
- Прикладний рівень (Application Layer). Цей рівень відповідає за забезпечення послуг для користувачів та програм, які працюють в мережі. Він включає в себе багато різноманітних протоколів, які забезпечують передачу даних від різних програм. До таких протоколів належать HTTP (Hypertext Transfer Protocol) для передачі веб-сторінок, FTP (File Transfer Protocol) для передачі файлів, SMTP (Simple Mail Transfer Protocol) для передачі електронної пошти та інші.

Модель TCP/IP по відношенню до моделі OSI [8] зображена на рисунку 1.2.

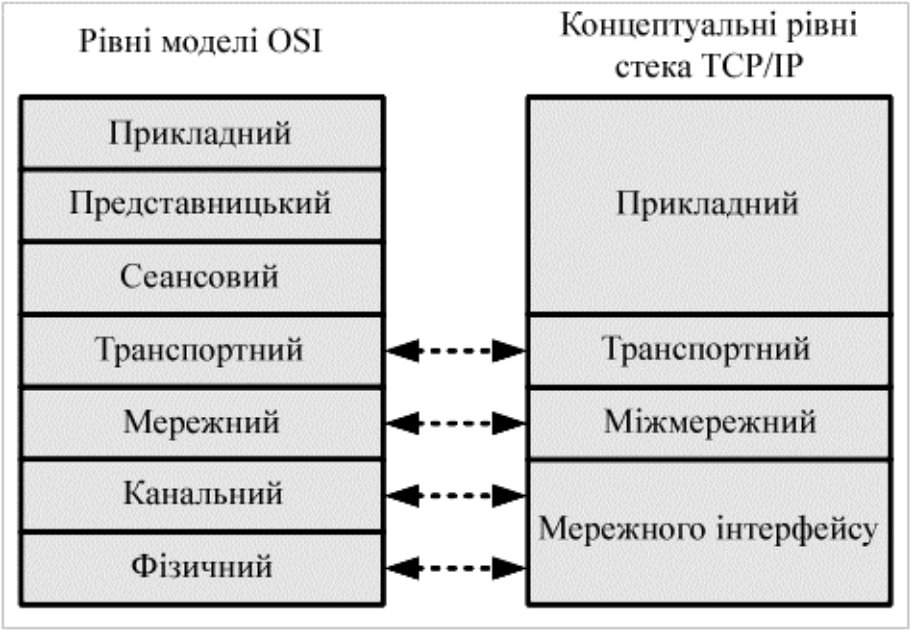

Рисунок 1.2 – Рівні моделі TCP/IP

Модель TCP/IP є більш універсальною, оскільки вона була розроблена для специфічних потреб Інтернету та може бути використана для комунікації в різних типах мереж [8]. Модель OSI була створена як загальна модель мережі, але її застосування більш обмежене, тому модель TCP/IP є більш поширеною, та широко використовується у світі комп'ютерних мереж.

1.2 Модель загроз

Несанкціонований доступ до інформації – доступ до інформації з порушенням правил розмежування доступу з використанням стандартних засобів, що забезпечуються засобами обчислювальної техніки, або автоматизованими системами (АС) .

Стандартні засоби – це сукупність технічного та програмного забезпечення комп’ютерної техніки та АС. Незважаючи на це, дія впровадженої програмної закладки, яка призвела до доступу зловмисника до захищеної інформації, також може розглядатися як факт несанкціонованого доступу (НД).

Серед основних загроз безпеки інформації: [9]

- Загрози проникнення в операційне середовище комп'ютера за допомогою стандартного програмного забезпечення (засоби операційної системи або прикладні програми загального призначення).
- Загрози створення позаштатних режимів роботи програмних (програмно-апаратних) засобів за рахунок навмисної зміни службових даних, ігнорування обмежень щодо складу та характеристик оброблюваної інформації, передбачених штатними умовами, спотворення (модифікації) самі дані тощо.
- Загрози впровадження шкідливих програм (програмно-математичний вплив).

Крім того, можливі комбіновані загрози, що представляють собою комбінацію зазначених загроз. Наприклад, впровадження шкідливих програм може створити умови для несанкціонованого доступу до операційного середовища комп'ютера.

Джерелом загроз може бути зловмисник, носій шкідливої програми або апаратна закладка. Порушники залежно від наявності доступу до інформаційної системи поділяються на внутрішніх і зовнішніх.

Носієм шкідливої програми може бути як апаратний елемент комп’ютера (флешка, диск тощо), так і програмний контейнер (наприклад, пакети повідомлень, що передаються комп’ютером).

Також 88% порушень даних є прямим результатом людської помилки. Саме людський фактор, є найбільшою проблемою при захисті даних. [10]

Далі буде розглянуто основі типи атак в мережах на основі TCP/IP. Якщо ОС підключена до загальнодоступних мереж, то проти неї можуть бути здійснені мережеві атаки. До публічних мереж на основі стеку протоколів TCP/IP відноситься Інтернет, на прикладі якого розглянуто найпоширеніші атаки сьогодні. Мережа Інтернет створена для зв'язку державних установ і університетом з метою сприяння навчальному процесу.

Стек сьогодні має багато вразливостей, які зловмисники успішно використовують для здійснення атак. Уразливості стеку протоколів TCP/IP пов'язані з певними слабкими місцями або недоліками у дизайні та реалізації цих протоколів, також зі слабкою автентифікацією та обмеженням розміру буфера. Короткий опис найбільш небезпечних вразливостей наведено в таблиці 1.1.

Таблиця 1.1 — Уразливості протоколу стека TCP/IP [11]

Найменування протоколу	Рівень стека в протоколі	Найменування уразливості	Зміст порушення безпеки інформації
FTP (File Transfer Protocol) - протокол для передачі файлів по мережі.	Прикладний, представницький, сеансовий	1. Автентифікація на базі відкритого тексту (паролі пересилаються в незашифрованому вигляді) 2. Доступ за замовчуванням 3. Наявність двох відкритих портів	Можливість перехоплення даних облікового запису (імен зареєстрованих користувачів, паролів) Отримання віддаленого доступу до хостів
Telnet – протокол управління віддаленим терміналом	Прикладний, представницький, сеансовий.	Автентифікація на базі відкритого тексту (паролі в незашифрованому вигляді)	Можливість перехоплення даних облікового запису користувача
UDP - це протокол для передачі даних без встановлення з'єднання	Транспортний	Відсутність механізму запобігання перевантажень буфера	Можливість реалізації UDP-шторму. В результаті обміну пакетами відбувається істотне зниження продуктивності сервера
ARP – протокол перетворення IP-адреси в фізичну адресу	Мережевий	Автентифікація на базі відкритого тексту	Можливість перехоплення трафіку користувача зловмисником

Продовження таблиці 1.1 — Уразливості протоколу стека TCP/IP [11]

RIP – протокол маршрутної інформації	Транспортний	Відсутність автентифікації керуючих повідомлень про зміну маршруту	Можливість перенаправлення трафіку через хост зловмисника
TCP – протокол управління передачею	Транспортний	Відсутність механізму перевірки коректності заповнення заголовків пакету	Істотне зниження швидкості обміну і навіть повний розрив довільних з'єднань за протоколом TCP
DNS - протокол для встановлення відповідності між мнемонічними іменами та мережевими адресами	Прикладний, представницький, сеансовий	Відсутність засобів перевірки автентифікації отриманих даних від джерела	Фальсифікація відповіді DNS-сервера
IGMP – протокол передачі повідомлень про маршрутизації	Мережевий	Відсутність автентифікації повідомлень про зміну параметрів маршруту	Зависання системи Windows
SMTP – протокол забезпечення сервісу доставки повідомлень по електронній пошті	Прикладний, представницький, сеансовий	Відсутність підтримки автентифікації заголовків повідомлень	Можливість підробки повідомлень електронної пошти, а також адреси відправника повідомлення
SNMP – протокол управління маршрутизаторами в мережах	Прикладний, представницький, сеансовий	Відсутність підтримки автентифікації заголовків повідомлень	Можливість переповнення пропускної спроможності мережі

Загрози, що реалізуються через мережу, класифікуються за 6 основними ознаками: [12]

1) Характер загрози.

- Пасивні загрози спрямовані на неправомірне здобування чутливої інформації без зміни її цілісності. Це означає, що зловмисник намагається перехопити інформацію, таку як паролі, конфіденційні дані, ключі шифрування тощо, без знання або згоди власника цієї інформації. Приклади пасивних загроз включають перехоплення мережевого трафіку, аналіз пакетів і перехоплення бездротових сигналів.
- Активні загрози передбачають зміну або руйнування даних, атаки на системи або перешкоджання звичайній роботі мережі. Зловмисник активно втручається в мережу та виконує дії, які можуть завдати шкоди системам або даним. Це може включати в себе впровадження шкідливого програмного забезпечення (наприклад, віруси, черв'яки, троянські програми), виконання атак на вразливості систем або навіть фізичні атаки на інфраструктуру мережі.

2) Мета реалізації загрози.

- Незаконне отримання конфіденційної інформації. Зловмисники можуть намагатися отримати доступ до конфіденційних даних, таких як особисті дані клієнтів, фінансова інформація, корпоративні секрети тощо. Ці дані можуть бути використані для шахрайства, шпигунства або вимагань.
- Пошкодження або зруйнування даних. Зловмисники можуть спробувати виконати атаки, які призведуть до пошкодження або втрати даних. Це може негативно вплинути на роботу організацій, завдати фінансової шкоди або порушити нормальне функціонування систем.
- Впровадження шкідливого програмного забезпечення. Зловмисники можуть намагатися впровадити в систему шкідливе програмне забезпечення, таке як

віруси, черв'яки, троянські програми. Це дозволить їм здійснювати різні дії, такі як шпигунство, контроль над системою або викрадення даних.

3) Умова початку нападу.

- На вимогу зловмисника. Тобто зловмисник розраховує передати запит певного типу, який буде умовою для початку несанкціонованого доступу.
- При настанні очікуваної події на атакуваному об'єкті.
- Безумовний вплив – зловмисник нічого не чекає, тобто загроза реалізується негайно і незалежно від стану атакуваного об’єкта.

4) Наявність зворотного зв'язку з об'єктом атаки:

- Зі зворотним зв'язком, тобто зловмиснику необхідно отримати відповідь на деякі запити. Таким чином, між тим, кого атакують, і тим, хто атакує, існує зворотний зв'язок, що дозволяє зловмисникові стежити за станом об'єкта, що атакується, і адекватно реагувати на його зміни.
- Без зворотного зв'язку – відповідно, відсутній зворотний зв'язок і необхідність реагування зловмисника на зміни в об'єкті атаки.

5) Розташування порушника по відношенню до атакуваної інформаційної системи: внутрішньосегментні та міжсегментні. Сегмент мережі — це фізична сукупність хостів, технічних засобів та інших компонентів мережі, які мають мережеву адресу.

6) Рівень еталонної моделі ISO/OSI, на якому реалізована загроза: фізичний, канальний, мережевий, транспортний, сеансовий, представницький, прикладний.

1.3 Модель зловмисника

Зловмисник в мережі — це особа або група, яка займається злочинною діяльністю в інтернеті або комп'ютерних мережах [13]. Вони мають різні мотивації і цілі, а їхні дії можуть бути спрямовані на отримання незаконного доступу до інформації, викрадення даних, шкоду комп'ютерним системам або мережам,

фінансову вигоду, шпигунство, вандалізм, політичні або ідеологічні цілі, а також завдання шкоди окремим особам, компаніям або урядовим структурам.

Порушники безпеки інформації в мережі володіють різними навичками та знаннями з областей комп'ютерної науки, криптографії, мережевої безпеки, програмування та соціальної інженерії. Вони використовують різноманітні методи та технології для здійснення своїх злочинних дій, серед яких можуть бути:

- Використання шкідливого програмного забезпечення. Зловмисники розробляють або використовують вже існуюче шкідливе програмне забезпечення, таке як віруси, черв'яки, троянські коні, розповсюджувачі шпигунського програмного забезпечення тощо.
- Фішинг (Phising). Це метод соціальної інженерії, при якому зловмисники намагаються видурити конфіденційні дані (наприклад, паролі, номери кредитних карток) шляхом відправлення підроблених повідомлень або електронних листів
- Витік інформації (Data Breach). Зловмисники можуть намагатися проникнути в систему або мережу з метою викрадення конфіденційної інформації, такої як особисті дані користувачів, фінансова інформація, комерційні та технічні дані підприємств.
- Соціальна інженерія. Зловмисники можуть використовувати маніпулятивні техніки та обман, щоб отримати несанкціонований доступ до системи або інформації.

Згідно з пунктом 2.8 «Наказу Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 20 липня 2007 року №141» [14], залежно від вірогідних умов експлуатації засобів КЗІ та відповідно до цінності інформації, що захищається, визначаються чотири рівні можливостей порушника, а саме:

- Нульовий рівень – користувач мережі Інтернет випадково порушив конфіденційність або цілісність інформації.

- Перший рівень – користувач усвідомлено, маючи невеликі фінанси самостійно розробляє та створює способи атаки на ПК жертви, а також на інформаційно-телекомунікаційні системи через використання загальнодоступних програмних засобів.
- Другий рівень – порушник має гарні фінансові можливості задля створення спеціальних технічних засобів, які після застосування, та порушенню цілісності та конфіденційності, також у разі втрати жертвою всієї інформації, нанесуть рівнозначне фінансове ураження жертві. У випадку таких атак можуть використовуватися локальні комп'ютерні мережі.
- Третій рівень – порушник має необмежений доступ до науково-технічних ресурсів, тобто має найрізномітніший арсенал задля атак, як технічний так і фінансовий.

У даній роботі при розробці й тестуванні технології міжмережевого екрану буде йти взаємодія лише зі зловмисниками нульового та першого рівня.

1.4 Види захисту інформації в мережі

Захист інформації, що обробляється автоматизованими системами (АС), є ключовим питанням для забезпечення конфіденційності, цілісності та доступності даних і вимагає створення та підтримки системи як технічних заходів, таких як інженерні, програмні та апаратні засоби, так і нетехнічних заходів, таких як правові та організаційні. В автоматизованих системах для досягнення цих цілей використовуються різноманітні заходи безпеки. [15]

Нижче на рисунку 1.3 зображено основні види програмно-апаратних методів захисту в мережі.

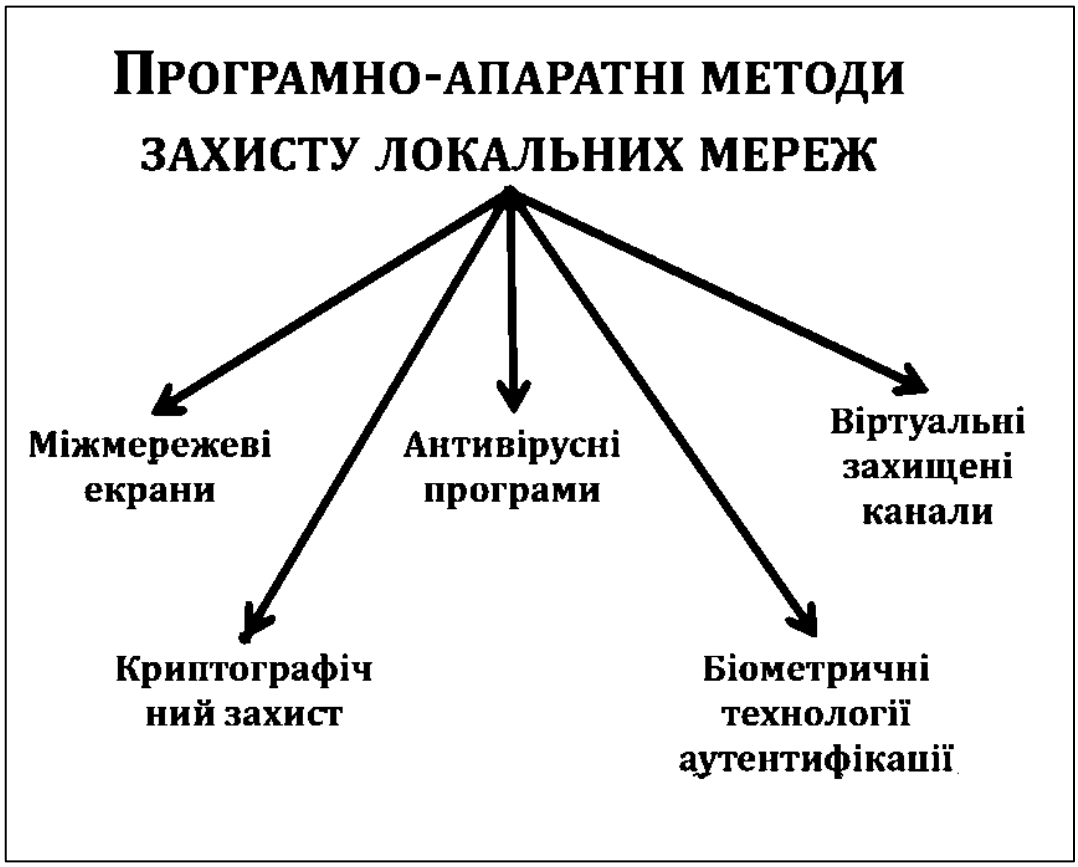

Рисунок 1.3 – Програмно-апаратні методи захисту мережі

Розбір кожного методу окремо:

1) Міжмережеві екрани можуть працювати на різних рівнях протоколу моделі OSI.
   - На мережевому рівні вхідні та вихідні пакети фільтруються за IP-адресами (наприклад, не пропускаються пакети з Інтернету, які надсилаються на сервери, доступ до яких іззовні заборонений). На транспортному рівні також відбувається фільтрація за номерами портів і прапорами, що містяться в пакетах (наприклад, запити на підключення).
2) Віртуальна приватна мережа створюється на основі загальнодоступного Інтернету.
   - Спілкування через Інтернет має свої недоліки, головний з яких полягає в тому, що воно може призвести до потенційних порушень безпеки та конфіденційності, VPN можуть гарантувати, що трафік, який

надсилається через Інтернет, є таким же безпечним, як і передача в локальній мережі.

- При цьому віртуальні мережі забезпечують значну економію коштів у порівнянні з вмістом власної мережі в глобальному масштабі.

3) Антивірус - це програмний засіб, призначений для боротьби з вірусами. Виходячи з визначення, основними завданнями антивіруса є:
   - Запобігання проникненню вірусів в комп'ютерну систему
   - Виявлення наявності вірусів в комп'ютерній системі
   - Видалення вірусів з комп'ютерної системи без нанесення шкоди іншим об'єктам системи
   - Мінімізація шкоди від дії вірусів
   - Технології виявлення вірусів
4) Криптографічний захист інформації.
   - Використовує методи шифрування та розшифрування для забезпечення конфіденційності, цілісності та автентичності даних під час їх передачі та зберігання.
   - Криптографія використовує математичні алгоритми та ключі шифрування для перетворення зрозумілої інформації в криптографічний вигляд, який може бути розшифрований лише з використанням правильних ключів.
5) Біометричний захист інформації.
   - Біометричні системи безпеки використовують вимірювані фізіологічні характеристики, унікальні для кожної людини, щоб перевірити особу особи. Цей процес називається електронною автентифікацією. Його суть полягає в тому, щоб визначити, чи справді людина є тим, за кого себе видає. Це відрізняє автентифікацію від ідентифікації та авторизації. Мета ідентифікації — перевірити, чи відома особа системі, наприклад, шляхом перевірки пароля, а авторизація — надати користувачеві доступ до певних ресурсів на основі його особистості.

- Інформація вважається захищеною від конкретних видів загроз або їх класифікації, коли забезпечені основні властивості інформації та систем в якій вона обробляється, а саме: цілісність, конфіденційність та доступність. [16]

1.5 Постановка задач

Основною темою моєї роботи є розробка міжмережевого екрану для захисту інформації в мережі, тому визначено саме такі тези постановки задачі :

- Розглянути технологію міжмережевого екрану, визначити його різновиди, переваги та недоліки.
- Розглянути утиліту командного рядка iptables, за допомогою якої можна налаштовувати міжмережеві екрани в операційних системах на базі ядра Linux.
- Розробка правил міжмережевого екрану, написання їх за допомогою утиліти iptables, перевірка й використання функціоналу даної утиліти, й послідуюче правильне налаштування міжмережевого екрану.
- Тестування створеного МЕ різними атаками , перевірка надійності захисту.
- Зробити висновки за результатами роботи розробленого мною налаштування правил захисту й фільтрації трафіку міжмережевого екрану.

2 ОГЛЯД ТЕХНОЛОГІЇ МІЖМЕРЕЖЕВИХ ЕКРАНІВ

2.1 Принципи роботи міжмережевих екранів.

Міжмережевий екран (МЕ) це система, яка використовується для захисту комп'ютерних мереж від несанкціонованого доступу та заборонених зовнішніх впливів [17]. Технологія виступає в ролі захисної стіни між локальною мережою (LAN) та зовнішньою мережою (WAN) і запобігає будь-яким загрозам. Він використовується задля контролю вхідного та вихідного трафіку на пристроях в локальній мережі, дає змогу припиняти фактично всі види мережевих атак, видаляти рекламу, відключати банери, рекламні скрипти сайтів, вспливаючі вікна, тощо. Ще одним з плюсів МЕ є те, що він контролює безпеку інформації про ваш пристрій, а саме щоб та не була надіслана «чужим» серверам, також робить даремною роботу програм-троянів, вірусів і засобів віддаленого адміністрування.

Міжмережевий екран працює на основі набору правил, які визначають, які типи з'єднань, пакетів даних або мережевих протоколів можуть проходити через нього. Він перевіряє кожен пакет даних, що надходить ззовні, і приймає рішення щодо його допуску або блокування в залежності від встановлених правил. Таким чином, він допомагає запобігти несанкціонованому доступу до мережі і захищає внутрішні ресурси від зловмисників.

Робота цього застосунку полягає в аналізі структури і вмісту інформаційних пакетів, що надходять з зовнішньої мережі, та в залежності від результатів аналізу вмісту даних пропускає ці пакети у внутрішню мережу, або є повністю їх відфільтровує, тобто не пропускає. Перевага міжмережевого екрану, що працює під управлінням Windows, полягає в тому, що він повністю замінює реалізований стек протоколів TCP/IP, унеможливлюючи спотворення хакерських даних протоколу зовнішньої мережі і втручання в її роботу.

Загальну схему роботи МЕ можна розглянути на рисунку 2.1 .

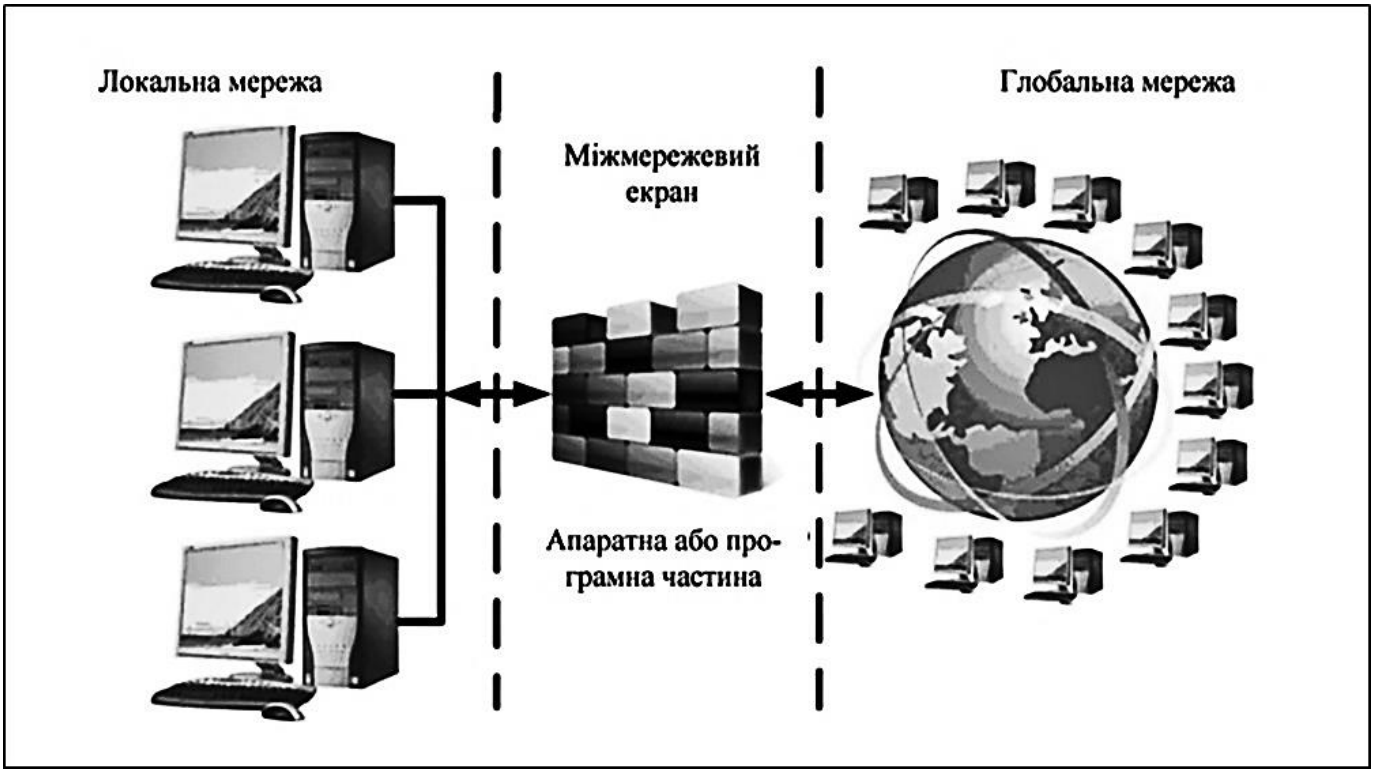

Рисунок 2.1 – Загальна схема роботи міжмережевого екрану

2.2 Характеристика функцій МЕ

Основні функції міжмережевих екранів включають:

- Фільтрація пакетів. Міжмережевий екран може фільтрувати вхідні та вихідні пакети даних, що пролягають через нього, на основі заданих правил. Це дозволяє блокувати несанкціонований доступ до мережі та контролювати доступ до конкретних ресурсів.
- NAT (Network Address Translation). Міжмережевий екран може використовувати NAT для перетворення внутрішніх IP-адрес на одну зовнішню IP-адресу. Це дозволяє обмежити прямий доступ до внутрішньої мережі та забезпечити її безпеку.
- VPN (Virtual Private Network) проксі. Міжмережевий екран може надавати можливість встановлення захищених тунелів VPN для з'єднання внутрішньої

мережі з зовнішніми мережами. Це дозволяє забезпечити безпеку підключення до мережі з дистанції та забезпечити захист конфіденційної інформації.

- Інспекція стану. Міжмережевий екран може аналізувати стан пакетів даних, що проходять через нього, та виявляти підозрілі дії, такі як спроби атак або зловмисного програмного забезпечення. Це дозволяє оперативно реагувати на загрози та запобігати можливим інцидентам.

Існують два типи міжмережевих екранів: програмний і апаратний. Апаратний тип представляє собою пристрій, який фізично підключається до мережі. Цей пристрій відстежує всі аспекти вхідного і вихідного обміну даними, а також перевіряє адреси джерела і призначення кожного оброблюваного повідомлення, що забезпечує захист, допомагаючи запобігти небажаним проникненням в мережу або пристрій користувача. Наступний тип який буде розглянуто – програмний, він виконує ті ж самі функції, що й апаратний, але використовує не зовнішній пристрій, а програмний продукт, який запущений на кінцевому комп’ютері або шлюзі. Найбільшого розповсюдження отримав саме програмний тип реалізації. [18]

Основні властивості міжмережевих екранів включають:

- Безпека. Міжмережевий екран є основним засобом захисту мережі від несанкціонованого доступу та зловмисних атак.
- Сумісність. Міжмережевий екран повинен бути сумісним з різними мережевими пристроями та програмним забезпеченням, що використовуються в організації. Це означає, що він повинен підтримувати різні мережеві протоколи, такі як TCP/IP, IPv4 або IPv6, а також різні типи підключень, включаючи проводові та бездротові.
- Легкість налаштування. Міжмережевий екран повинен мати інтуїтивний і зручний інтерфейс для налаштування правил і параметрів безпеки. Налаштування має бути зрозумілим для адміністраторів мережі, щоб вони

могли ефективно керувати його роботою та забезпечувати відповідний рівень захисту.

- Масштабованість. Міжмережевий екран повинен бути здатним працювати в різних мережевих середовищах, включаючи малий офіс, підприємства або навіть хмарні інфраструктури. Він повинен бути гнучким і масштабованим для відповіді на зростаючі потреби мережі.
- Моніторинг та журналювання. Міжмережевий екран повинен забезпечувати можливість моніторингу мережевої активності та реагувати на події пов’язані з безпекою. Він повинен мати можливість реєструвати події, створювати журнали та забезпечувати аналіз мережевої активності для виявлення аномальних або підозрілих здійснень.
- Висока продуктивність. Міжмережевий екран повинен мати достатню продуктивність для обробки великого обсягу мережевого трафіку, забезпечуючи швидкодію та низьку затримку. Важливо, щоб він мав достатні ресурси, такі як швидкоплинні процесори, достатню оперативну пам'ять і швидкість мережевих інтерфейсів, щоб забезпечити ефективну обробку трафіку навіть при високих навантаженнях.
- Виявлення загроз і запобігання. Міжмережевий екран повинен мати можливості виявлення загроз і запобігання їхній реалізації. Це може включати вбудовані механізми виявлення вторгнень (Intrusion Detection System - IDS) або вторгнень і запобігання їхній реалізації (Intrusion Prevention System - IPS), що допомагають виявити та блокувати зловмисні атаки.
- Веб-фільтрація та контроль доступу. Міжмережевий екран може мати вбудовані функції веб-фільтрації, які дозволяють контролювати доступ до веб-сайтів, блокувати небажані або небезпечні ресурси, а також обмежувати доступ до певних типів контенту. Це дозволяє підтримувати політики безпеки та обмежувати ризик доступу до небезпечних веб-ресурсів.

- Журналювання та аудит. Міжмережевий екран може забезпечувати журналювання всіх подій та активності, що стосуються мережі, включаючи спроби вторгнень, блокування пакетів, аналіз трафіку тощо. Це надає можливість для аналізу подій, виявлення аномалій та вирішення інцидентів безпеки.
- Постійне оновлення і підтримка. Міжмережевий екран повинен мати можливість регулярного оновлення свого програмного забезпечення та визначення нових правил безпеки. Це дозволяє забезпечити захист від нових загроз і вразливостей, а також впроваджувати нові функції і покращення безпеки.
- Гнучкість в конфігурації. Міжмережевий екран повинен мати гнучкі можливості конфігурації для відповідності потребам та політикам безпеки організації. Це включає можливість створення і налаштування правил доступу, блокування конкретних портів або протоколів, встановлення обмежень швидкості трафіку та інші параметри безпеки.

Міжмережевих екрани також мають власну політику щодо трафіку в мережі, яка будується на основі :

- Мережевих протоколів.
- Вмісту трафіка.
- Спрямованість трафіку в мережі.
- Програм що діють в мережі.
- Усіх діапазонів IP-адрес в мережі.

Міжмережеві екрани працюють на декількох рівнях протоколів моделі OSI. На мережевому рівні моделі вхідні та вихідні пакети фільтруються за IP-адресами (наприклад, не пропускаються пакети з мережі Інтернет, які відправляються на сервери, доступ до яких ззовні заборонено). Для прикладу на прикладному рівні аналізуються прикладні протоколи такі як FTP, SMTP,HTTP і потік даних взагалі. [19]

2.3 Види міжмережевих екранів та їх відмінності

Існує кілька типів міжмережевих екранів, основні з яких включають: [20]

1) Пакетні фільтри (Packet Filtering Firewalls). Цей тип екранів працює на найнижчому рівні мережевої структури і аналізує заголовки пакетів даних, що протікають через них. Вони приймають рішення щодо допустимості чи блокування пакета на основі правил, які визначаються адміністратором мережі. Ці екрани є простими у налаштуванні, але мають обмежену можливість аналізувати дані поза заголовками пакетів. Як влаштований даний тип міжмережевого екрану зображено на рисунку 2.2.

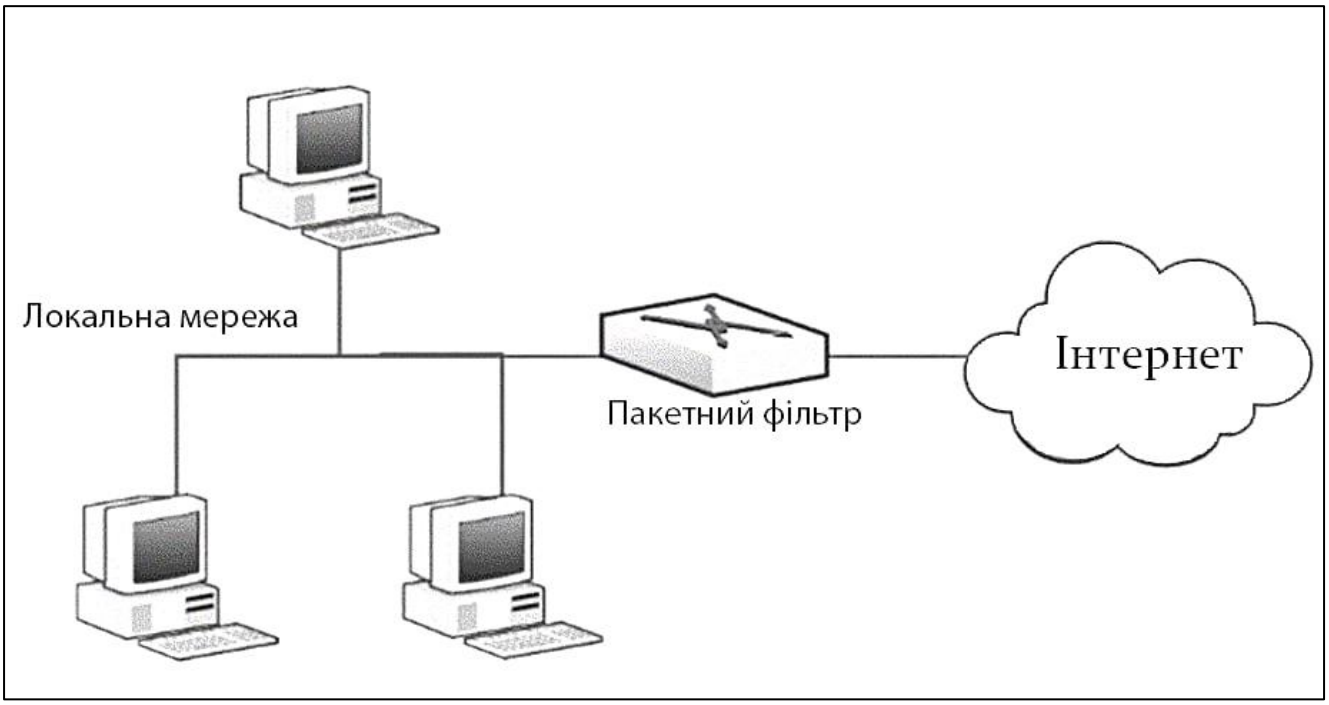

Рисунок 2.2 – Пакетний фільтр

Пакетні фільтри, як міжмережеві екрани, мають свої переваги і недоліки.

Переваги пакетних фільтрів:

- Простота. Пакетні фільтри є відносно простими в реалізації та налаштуванні. Вони аналізують заголовки пакетів даних і відповідають на запити згідно з налаштованими правилами безпеки.

- Швидкодія. Пакетні фільтри працюють на низькому рівні мережевого стеку, що дозволяє їм обробляти велику кількість пакетів даних зі швидкістю, не впливаючи на продуктивність мережі.
- Ефективність. Використання пакетних фільтрів дозволяє блокувати небажаний трафік та неправильні запити на ранніх етапах, що допомагає зменшити ризики зовнішніх атак і загроз безпеці.

Недоліки пакетних фільтрів:

- Обмежена функціональність. Пакетні фільтри аналізують тільки заголовки пакетів даних, а не їхнє вміст, тому вони можуть бути менш ефективними у виявленні складних загроз, які можуть приховуватися у вмісті пакетів.
- Відсутність контексту. Пакетні фільтри працюють на основі правил, встановлених на основі заголовків пакетів, і не мають контексту або інформації про стан пакета. Це може призводити до неправильної інтерпретації трафіку та виникнення ложно-позитивних або ложно-негативних результатів.
- Недостатня гнучкість. Пакетні фільтри можуть бути обмежені у своїй гнучкості і можливостях налаштування.

2) Інспекція стану пакета (Stateful Inspection Firewalls). Ці екрани спостерігають за потоком мережевого трафіку і здатні виявляти стан кожного пакета. Вони зберігають інформацію про попередні пакети в потоці, що дозволяє їм аналізувати діалоги та забезпечувати більш докладний контроль. Інспекція стану пакета також дозволяє екранам виявляти й блокувати підозрілі або небезпечні пакети, які не відповідають правилам безпеки. Як влаштовані межмережеві екрани іспекції стану пакета зображено на рисунку 2.3.

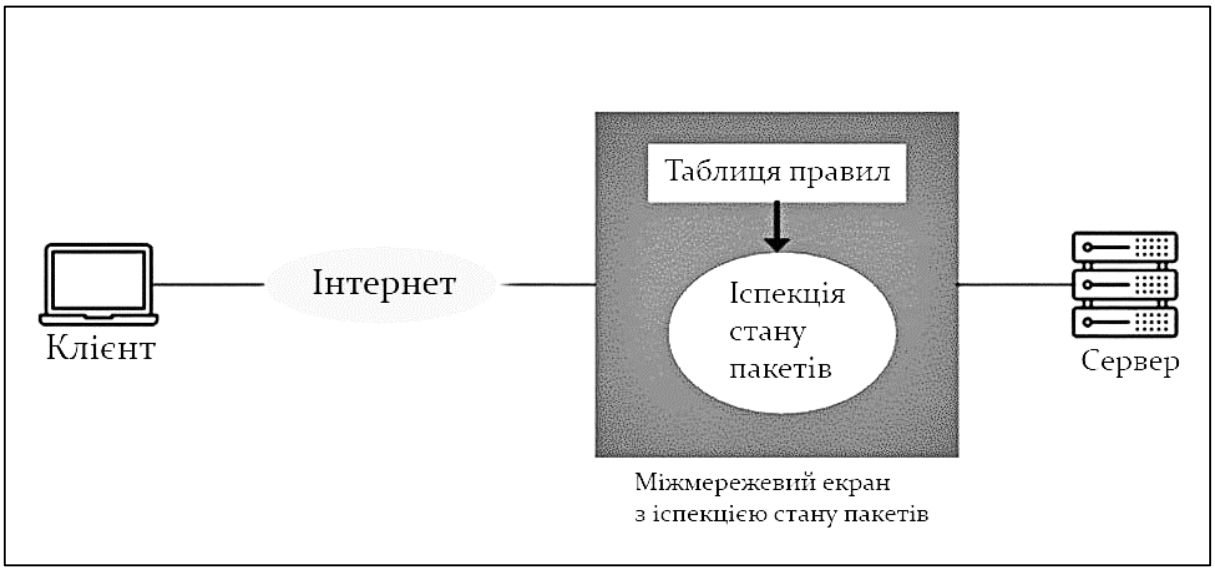

Рисунок 2.3 – МЕ інспекції стану пакета

Інспекція стану пакетів (Stateful Inspection Firewalls) є покращеною версією пакетних фільтрів, яка має свої переваги і недоліки.

Переваги Інспекції стану пакетів:

- Розуміння контексту. Інспекція стану пакета здатна розуміти контекст і стан кожного пакета, а не просто аналізувати заголовки. Вона зберігає інформацію про попередні пакети в потоці, що дозволяє виявляти підозрілу активність і вести діалоговий аналіз.
- Вищий рівень безпеки. Інспекція стану пакета дозволяє виявляти і блокувати складні загрози, такі як атаки на вміст, фрагментацію пакетів, вторгнення та інші. Вона може аналізувати не лише заголовки, але і вміст пакетів для виявлення потенційно шкідливого змісту.
- Зменшення ложних срабатувань. Інспекція стану пакета збирає контекстуальну інформацію про стан комунікації між джерелом і призначенням. Це допомагає зменшити кількість хибних спрацьовувань і покращує ефективність фільтрації трафіку.

Недоліки інспекції стану пакетів:

- Більший обсяг обробки. Інспекція стану пакета вимагає зберігання і обробки більшої кількості інформації про стан пакетів. Це може впливати на продуктивність мережі і потребувати більшої обчислювальної потужності для роботи.
- Складність налаштування. Інспекція стану пакета потребує докладнішого налаштування та конфігурації порівняно з простими пакетними фільтрами. Це означає, що вона може вимагати більше часу та експертизи з боку адміністратора мережі для належної роботи та оптимального налаштування.
- Потреба у великій кількості пам'яті. Інспекція стану пакета потребує зберігання даних про стан пакетів в пам'яті. Це може вимагати значної обсягу пам'яті, особливо для великих мереж або потоків даних з великою пропускною здатністю.
- Обмеження на швидкість. Враховуючи більший обсяг обробки та потребу в пам'яті, інспекція стану пакета може стикатися з обмеженнями швидкості при обробці великого обсягу трафіку. Це може вплинути на продуктивність мережі, особливо в великих мережевих середовищах з високим рівнем трафіку.
- Вартість. В порівнянні з простими пакетними фільтрами, інспекція стану пакета зазвичай є більш складною технологією з більшою функціональністю, що може призвести до вищої вартості. Залежно від постачальника та рівня функціональності, вона може бути більш коштовною для впровадження та підтримки.

Не дивлячись на недоліки, інспекція стану пакета є більш розширеною та ефективною технологією міжмережевої безпеки, яка здатна забезпечувати більш високий рівень контролю та захисту мережі.

3) Проксі-сервери (Proxy Firewalls). Проксі-сервери працюють на рівні програмного забезпечення і діють як посередник між внутрішньою мережею та

зовнішніми ресурсами. Коли комп'ютер з внутрішньої мережі намагається отримати доступ до зовнішнього ресурсу, запит спочатку надсилається проксі-серверу, який виконує запит від свого імені. Проксі-сервер перевіряє запит на відповідність правилам безпеки та фільтрації, а також може здійснювати додаткову функціональність, таку як кешування веб-сторінок для поліпшення продуктивності мережі. Використання проксі-серверів дозволяє забезпечити вищий рівень контролю та конфіденційності для внутрішніх мереж.

На рисунку 2.4 зображено схему роботи Проксі-серверу.

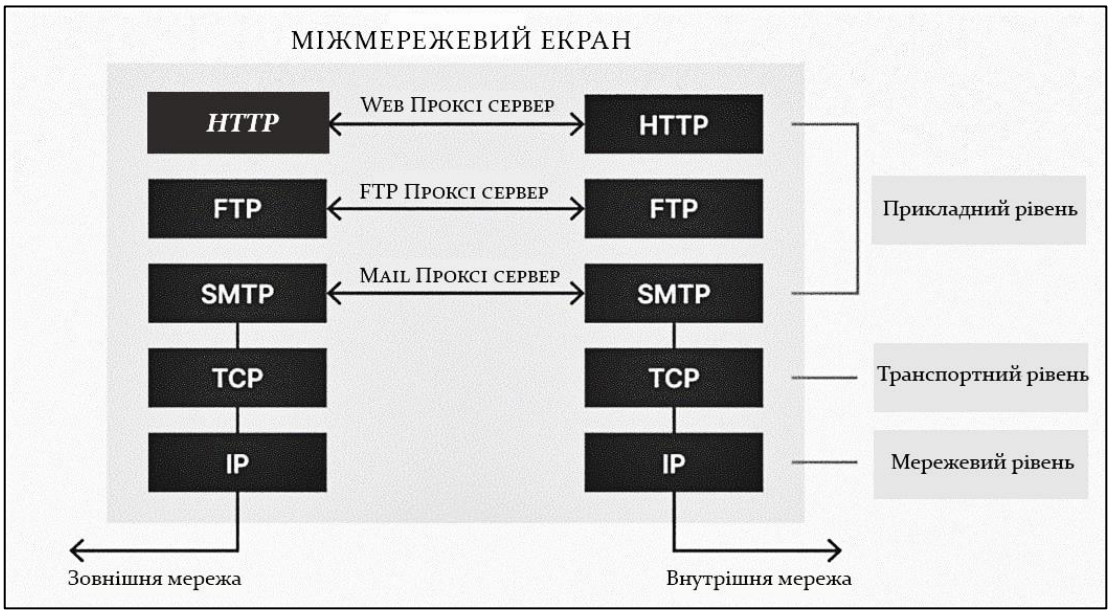

Рисунок 2.4 – Проксі-сервер

Переваги Проксі-серверів:

- Контроль доступу. Проксі-сервери здатні забезпечити детальний контроль доступу до різних мережевих ресурсів. Вони можуть використовувати правила та політики, щоб обмежувати або дозволяти доступ до веб-сайтів, додатків чи інших сервісів на основі різних параметрів, таких як IP-адреси, порти, протоколи та ідентифікатори користувачів.
- Захист від атак. Проксі-сервери можуть використовуватись для фільтрації трафіку та виявлення потенційно шкідливих дій або загроз. Вони можуть

використовувати механізми перевірки безпеки, такі як виявлення вторгнень або антивірусні сканери, для забезпечення безпеки мережі та запобігання атакам.

- Кешування та прискорення. Проксі-сервери можуть кешувати веб-сторінки, що дозволяє зберігати копії популярних ресурсів та надавати їх користувачам безпосередньо з кешу. Це покращує швидкодію і ефективність мережі, зменшує пропускну здатність Інтернету і знижує завантаження на сервери.
- Анонімність. Проксі-сервери можуть допомогти забезпечити анонімність та конфіденційність користувачів. Вони можуть приховувати реальні IP-адреси користувачів, замість цього передаючи запити через свою власну адресу. Це може бути корисно для користувачів, які хочуть зберегти приватність або обійти обмеження доступу до деяких веб-ресурсів.

Недоліки проксі-серверів:

- Збільшення затримки. Використання проксі-серверів може спричинити збільшення затримки в обробці мережевого трафіку. Кожен запит від клієнта повинен пройти через проксі-сервер, що може вплинути на час передачі даних та загальну швидкодію мережі.
- Одна з точок вразливості. Проксі-сервери стають однією точкою вразливості в мережевій інфраструктурі. Якщо проксі-сервер компрометований або некоректно налаштований, це може призвести до компрометації всієї мережі та витоку конфіденційної інформації.
- Обмеження функціональності. Проксі-сервери можуть бути обмежені у своїй функціональності порівняно з безпосереднім підключенням до ресурсів. Вони можуть не підтримувати деякі протоколи, функції або взаємодію з деякими типами додатків, що може обмежувати доступ користувачів до певних ресурсів або послуг.

- Складність налаштування та управління. Проксі-сервери вимагають налаштування та управління для забезпечення безпеки і ефективності. Це може бути складним завданням, особливо для великих мереж або мереж зі складною структурою.
- Вплив на швидкість трафіку. У разі великого обсягу трафіку або неправильного налаштування проксі-серверів може виникати затримка або втрата пакетів, що може вплинути на продуктивність та якість обслуговування мережі.

Важливо враховувати ці плюси і мінуси проксі-серверів при розгляді проксі-серверів для впровадження мережевої інфраструктури. Незважаючи на недоліки, проксі-сервери є важливими інструментами для забезпечення безпеки, контролю та оптимізації мережі. Правильно налаштовані та керовані проксі-сервери можуть допомогти покращити безпеку мережі, зменшити ризики атак та забезпечити контроль доступу до ресурсів.

4) Програмно-апаратні міжмережеві екрани (Next-Generation Firewalls). Цей тип екранів поєднує в собі функціональність пакетних фільтрів, інспекції стану пакета та проксі-серверів, а також надає додаткові можливості. Вони здатні виявляти та блокувати загрози на різних рівнях, включаючи вразливості застосунків, вторгнення, атаки на вміст та інші сучасні загрози. Програмно-апаратні міжмережеві екрани зазвичай мають більш розширені можливості управління та моніторингу, що дозволяє адміністраторам здійснювати детальний контроль над мережею.

Кожен вид міжмережевого екрана має свої переваги та недоліки, і вибір певного типу залежить від конкретних потреб, обмежень та цілей мережі. У деяких випадках комбінація різних видів міжмережевих екранів може бути доцільною для забезпечення більш повного рівня безпеки та контролю. [21]

3 ФУНКЦІОНАЛ УТИЛІТИ IPTABLES

3.1 Ключові поняття iptables

Основним засобом через який можно реалізувати й протестувати функціонал міжмережевого екрану на ОС Linux є утиліта iptables. Далі буде розглянуто основні можливості цього інтерфейсу управління міжмережевими екранами.

Iptables є інструментом для налаштування правил міжмережеовго екрану в операційних системах на базі ядра Linux [22]. Iptables дозволяє адміністраторам мережі контролювати та фільтрувати мережевий трафік, що входить і виходить з системи, забезпечуючи безпеку та захист мережевих ресурсів.

Основні складові утиліти iptables:

1) Таблиці (Tables). Iptables використовує різні таблиці для організації правил фільтрації пакетів. В iptables визначаються чотири основні таблиці, які мають різний функціонал:
    - FILTER. Використовується для фільтрації пакетів і прийняття рішення про їх долю.
    - NAT. Використовується для зміни мережевих адрес пакетів, таких як Network Address Translation (NAT).
    - MANGLE. Використовується для зміни певних атрибутів пакетів, таких як TTL (Time to Live) або TOS (Type of Service).
    - RAW. Використовується для налаштування правил фільтрації пакетів, які обходять деякі ланцюжки таблиці filter.
2) Ланцюжки (Chains). Ланцюжки є послідовністю правил фільтрації, які застосовуються до пакетів. В iptables існують п'ять основних ланцюжків, кожен з яких виконує свою роль в системі, далі буде перераховано кожен з них і їх задачі:

- INPUT. Контролює вхідний трафік до системи.
- OUTPUT. Контролює вихідний трафік з системи.
- FORWARD. Контролює пересилання пакетів між мережевими інтерфейсами системи.
- PREROUTING. Використовується в таблиці NAT для обробки пакетів до маршрутизації.
- POSTROUTING. Використовується в таблиці NAT для обробки пакетів після маршрутизації.

3) Правила (Rules). Правила визначають умови фільтрації пакетів та дії, які потрібно виконати, якщо пакет відповідає цим умовам. Кожне правило має свої параметри, які визначають, які пакети вони застосовуються до, та ціль, що виконується, якщо пакет відповідає умовам правила. Користувач може налаштовувати безліч правил для функціонування свого ME, але потрібно відноситися до цієї задачі уважно.
4) Ціль (Target). Ціль визначає, що робити з пакетом, якщо він відповідає правилу. Цілі можуть бути різними діями, такими як ACCEPT (прийнятий), DROP (відкинутий), REJECT (відхилений), LOG (запис в журнал) або перенаправлення на інший ланцюжок або правило. Ціль встановлює долю пакета після його перевірки з урахуванням умов правила.
5) Маркерування (Marking). Iptables дозволяє маркувати пакети з додатковими атрибутами для подальшого використання. Це може бути корисно для маркування пакетів для спеціального оброблення, наприклад, встановлення QoS (Quality of Service) або для ідентифікації певного трафіку.
6) Модулі (Modules). Iptables підтримує різні модулі, які розширюють його функціональність і дозволяють застосовувати різноманітні правила фільтрації. Деякі з популярних модулів включають модуль state (для відстеження стану з'єднання), модуль connlimit (для обмеження кількості одночасних з'єднань) та модуль recent (для відстеження останніх подій).

3.2 Ланцюги і таблиці.

На рисунку 3.1 зображено схему проходження пакетів з інформацією в ланцюжках iptables.

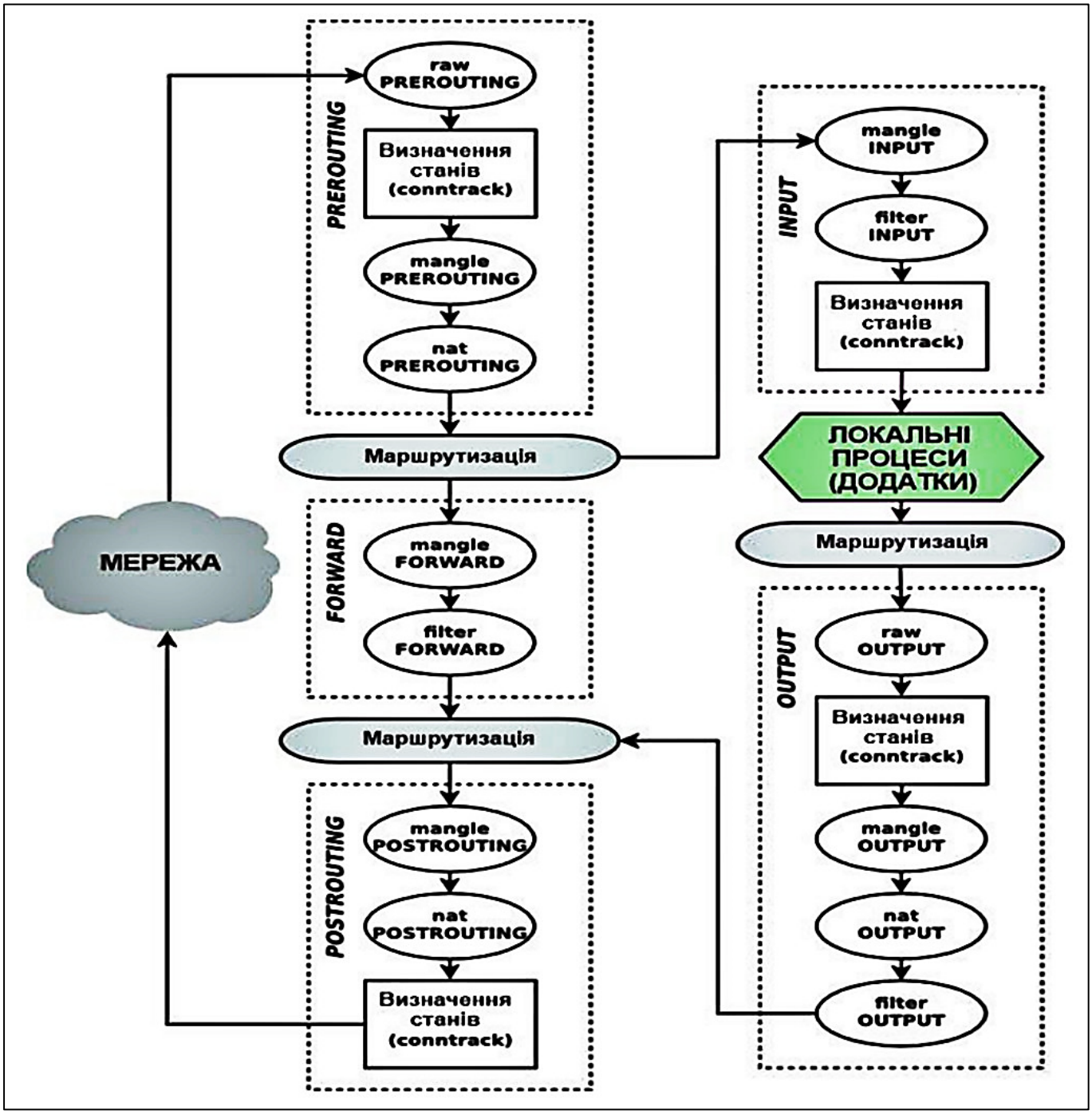

Рисунок 3.1 - Схема проходження пакетів з інформацією в ланцюжках iptables

Iptables використовує різні таблиці для організації правил фільтрації пакетів. Далі буде розглянуто кожну таблицю в iptables окремо: [23]

1) Filter (фільтр):
    - Ланцюжки: INPUT, OUTPUT, FORWARD.
    - Призначення. Ця таблиця використовується для фільтрації пакетів і прийняття рішення про їх долю.
    - Ланцюжок INPUT застосовує правила до вхідного трафіку, що надходить на систему.
    - Ланцюжок OUTPUT застосовує правила до вихідного трафіку з системи.
    - Ланцюжок FORWARD застосовується до пакетів, які проходять через систему і мають бути переслані на інший мережевий інтерфейс.
2) NAT (Network Address Translation, мережеве перетворення адреси):
    - Ланцюжки: PREROUTING, POSTROUTING, OUTPUT.
    - Призначення. Ця таблиця використовується для зміни мережевих адрес пакетів, наприклад, для здійснення Network Address Translation (NAT).
    - Ланцюжок PREROUTING використовується для обробки пакетів перед маршрутизацією.
    - Ланцюжок POSTROUTING використовується для обробки пакетів після маршрутизації.
    - Ланцюжок OUTPUT застосовується до пакетів, що генеруються локальною системою перед відправкою.
3) Mangle (зміна):
    - Ланцюжки: PREROUTING, INPUT, FORWARD, OUTPUT, POSTROUTING.
    - Призначення. Ця таблиця використовується для зміни певних атрибутів пакетів, таких як TTL (Time to Live) або TOS (Type of Service).

- Ланцюжки PREROUTING, INPUT та FORWARD використовуються для обробки пакетів до маршрутизації.
- Ланцюжок OUTPUT застосовується до пакетів, що генеруються локальною системою перед відправкою.
- Ланцюжок POSTROUTING використовується для обробки пакетів після маршрутизації.

4) Raw (сирі дані):
   - Ланцюжки: PREROUTING, OUTPUT.
   - Призначення. Ця таблиця використовується для обробки пакетів до того, як вони пройдуть перевірку з правилами фільтрації.
   - Ланцюжок PREROUTING використовується для обробки пакетів до маршрутизації.
   - Ланцюжок OUTPUT застосовується до пакетів, що генеруються локальною системою перед відправкою.
5) Security (безпека):
   - Ланцюжки: INPUT, OUTPUT, FORWARD.
   - Призначення. Ця таблиця використовується для виконання різних заходів безпеки, таких як захист від атак, контроль доступу тощо.
   - Ланцюжок INPUT застосовується до вхідного трафіку, що надходить на систему.
   - Ланцюжок OUTPUT застосовується до вихідного трафіку з системи.
   - Ланцюжок FORWARD застосовується до пакетів, які проходять через систему і мають бути переслані на інший мережевий інтерфейс.

Вибір таблиці в iptables залежить від конкретних потреб і сценаріїв використання. Для кожної таблиці можуть бути визначені різні ланцюжки та правила фільтрації [24].

Ланцюги є важливою частиною iptables і використовуються для організації правил фільтрації пакетів. Кожен пакет, що надходить або виходить з системи, проходить

через певні ланцюжки в певному порядку. Далі буде розглянуто основі ланцюжки в iptables:

1) PREROUTING:
    - Ланцюжок PREROUTING застосовує правила до пакетів до маршрутизації.
    - Це означає, що правила в цьому ланцюжку застосовуються до пакетів, щойно вони надійшли на мережевий інтерфейс системи, ще до того, як буде визначено, куди направити ці пакети.
2) INPUT:
    - Ланцюжок INPUT застосовує правила до вхідного трафіку, що надходить на систему.
    - Це означає, що правила в цьому ланцюжку виконуються для пакетів, призначених для самої системи.
3) FORWARD:
    - Ланцюжок FORWARD застосовується до пакетів, які проходять через систему і мають бути переслані на інший мережевий інтерфейс.
    - Це означає, що правила в цьому ланцюжку виконуються для пакетів, які промаршрутізовані через систему.
4) OUTPUT:
    - Ланцюжок OUTPUT застосовує правила до вихідного трафіку з системи.
    - Це означає, що правила в цьому ланцюжку виконуються для пакетів, що генеруються локальною системою перед відправкою.
5) POSTROUTING:
    - Ланцюжок POSTROUTING використовується для обробки пакетів після маршрутизації.
    - Це означає, що правила в цьому ланцюжку застосовуються до пакетів перед їх відправкою з мережевого інтерфейсу системи.

Ці ланцюжки в iptables надають гнучкість і можливості для налаштування обробки пакетів у мережі. Вибір конкретного ланцюжка залежить від сценаріїв використання і потреб в мережі користувача.

3.3 Побудова правил для iptables

Побудова правила в iptables включає в себе визначення таблиці, ланцюжка, матчів, дій та додаткових параметрів. [25]

Далі буде розглянуто елементи побудови правила в iptables:

iptables [-t table] command [match] [target/jump] – загальна будова правила.

A – додавання нового правила ;

D – видалення вказаного правила;

R – заміна виділеного правила;

I – додавання в вказаний ланцюг нового правила;

L – вивести список всіх наявних правил в даному ланцюгу;

F – відкинути всі задані правила в даному ланцюгу;

Z – відкинути усі лічильники в даному ланцюгу;

N – створити новий ланцюг;

X – видалення вказаного ланцюга;

P – задання нової політики для вказаного ланцюга;

E – задання нового ім'я для вказаного ланцюга;

Match – показати критерії для перевірки.

Target/jump – дія (DROP, REJECT, ACCEPT etc.).

Розділ match в iptables відповідає за визначення критеріїв, за якими будуть зіставлятися пакети для визначення відповідних правил фільтрації. Він дозволяє адміністраторам встановлювати умови, які повинні виконуватися, щоб пакет відповідав певному правилу.

Розділ target відноситься до дії, яка виконується над пакетами, що відповідають певним правилам фільтрації. Кожне правило має свою ціль, яка вказує, що робити з пакетом, коли воно відповідає умовам правила.

Основні файли конфігурації iptables:

«/etc/sysconfig/iptables» - файл конфігурації

Лог-файли iptables знаходятся в директоріях:

«/var/log/messages»

«/var/log/syslog»

«/var/log/iptables.log»

Отже, iptables допомагає захистити мережу від різноманітних мережевих атак, таких як SYN-флуд атаки, атаки на відмову в обслуговуванні (DoS), атаки на переповнення буфера та інші. Застосування правильних правил фільтрації пакетів дозволяє обмежити атакуючий трафік і забезпечити надійність мережі.

Iptables надає широкі можливості для налаштування правил фільтрації пакетів і працює з різними типами мережевих протоколів. Він також може бути розширений за допомогою різноманітних модулів, що дозволяють налаштовувати специфічні правила для певних сценаріїв. Наприклад, модуль "conntrack" дозволяє відстежувати стан мережевих з'єднань і приймати рішення на основі цього стану.

Також iptables підтримує можливість логування подій, пов'язаних з фільтрацією пакетів. Це дозволяє адміністраторам відстежувати трафік, аналізувати потенційні загрози та здійснювати моніторинг мережі.

4 ПОБУДОВА І РОЗГОРТАННЯ ВЛАСНОГО РІШЕННЯ

4.1 Встановлення програмного забезпечення.

Для подальшого налаштування власного міжмережевого екрану виділеного серверу на базі вільного програмного забезпечення, буде використовуватися операційна система Ubuntu версії 22.04.2 LTS, найпопулярніший у світі дистрибутив Linux, який встановлений на віртуальній машині Vmware Workstation 17 Player. На рисунку 4.1. зображено встановлену операційну систему з актуальною версією.

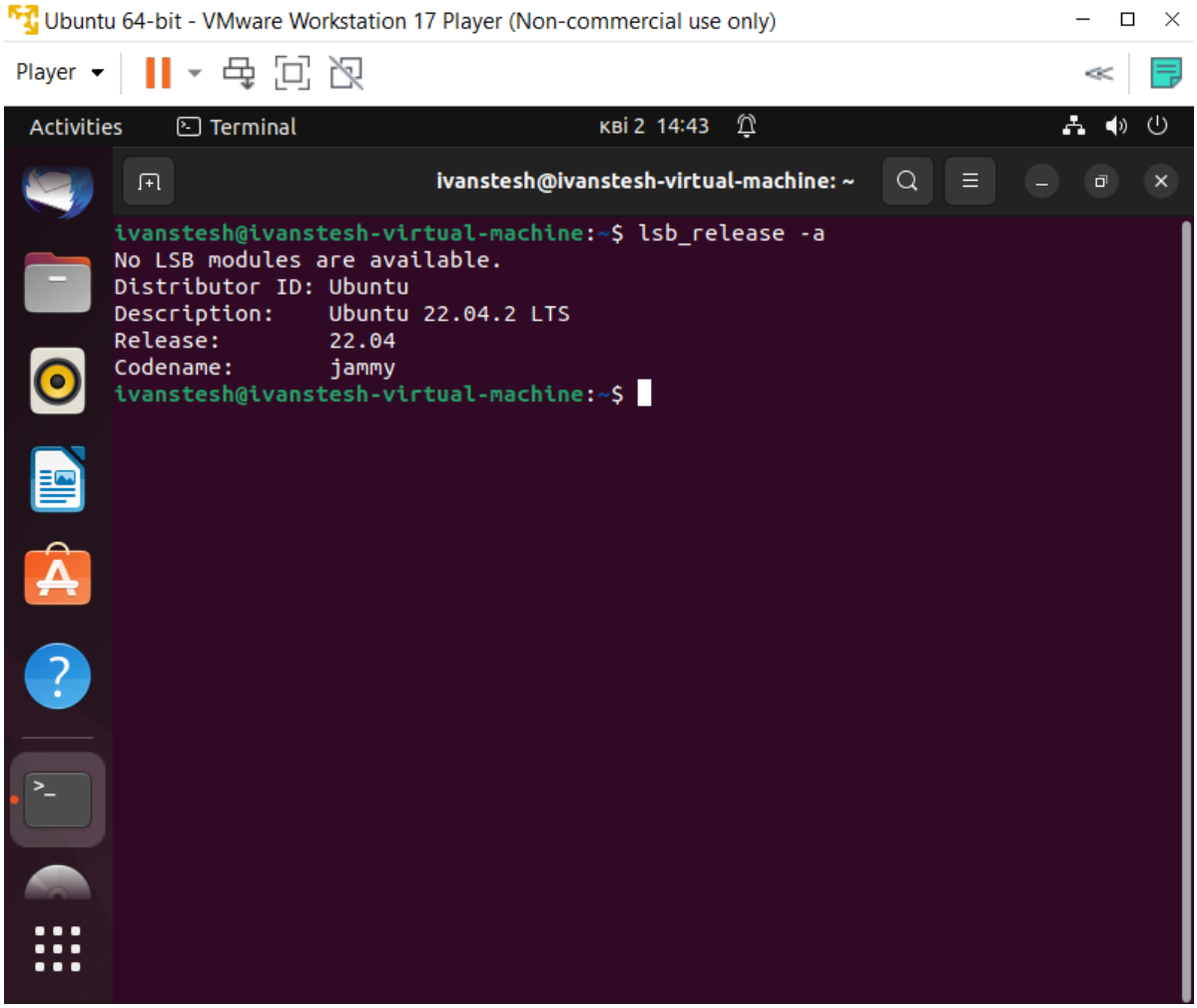

Рисунок 4.1. – встановлена Ubuntu 22.04.2 LTS на Wmware Workstation 17

Далі для роботи також потрібно встановити утиліту iptables для подальшого налаштування міжмережевого екрану. Для цього в терміналі Ubuntu використано команду « sudo apt-get install iptables », результат зображено на рисунку 4.2.

```
ivanstesh@ivanstesh-virtual-machine:~$ sudo apt-get install iptables
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
iptables is already the newest version (1.8.7-1ubuntu5).
The following packages were automatically installed and are no longer required:
  chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi i965-va-driver intel-media-va-driver
  libaacs0 libaom3 libass9 libavcodec58 libavformat58 libavutil56 libbdplus0 libblas3
  libbluray2 libbs2b0 libchromaprint1 libcodec2-1.0 libdav1d5 libflashrom1 libflite1
  libftdi1-2 libgme0 libgsm1 libgstreamer-plugins-bad1.0-0 libigdgmm12 liblilv-0-0
  libllvm13 libmfx1 libmysofa1 libnorm1 libopenmpt0 libpgm-5.3-0 libpostproc55
  librabbitmq4 librubberband2 libserd-0-0 libshine3 libsnappy1v5 libsord-0-0
  libsratom-0-0 libsrt1.4-gnutls libssh-gcrypt-4 libswresample3 libswscale5 libudfread0
  libva-drm2 libva-wayland2 libva-x11-2 libva2 libvdpau1 libvidstab1.1 libx265-199
  libxvidcore4 libzimg2 libzmq5 libzvbi-common libzvbi0 mesa-va-drivers
  mesa-vdpau-drivers pocketsphinx-en-us va-driver-all vdpau-driver-all
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 15 not upgraded.
```

Рисунок 4.2 – Встановлення утиліти iptables

Відразу йде перевірка стандартних налаштувань міжмережевого екрану за допомогою команди « sudo iptables -L », результат зображено на рисунку 4.3.

```
ivanstesh@ivanstesh-virtual-machine:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Рисунок 4.3. – Стандартні налаштування iptables

Як можно побачити за замовчуванням немає жодних правил для міжмережевого екрану і проходимість пакетів по мережі не обмежується.

На цьому встановлення програмного забезпечення, необходіного для практичної частити завершено.

4.2 Встановлення та налаштування власного серверу Apache2.

Для роботи з пакетами та налаштування свого міжмережевого екрану обрано HTTP-сервер Apache2. Він має дуже багато функцій, включаючи динамічно загружаючі модулі, надійну підтримку медіаформатів і інтеграцію з іншим програмним забезпеченням. Apache2 доступний в репозиторіях програмного забезпечення Ubuntu за замовчуванням, тому є змога встановити його за допомогою стандартних інструментів управління пакетами. Посилання для документації налаштування серверу Apache2: [26] https://httpd.apache.org/docs/2.4/en/ . Команда для установки серверу виглядає наступним чином і використовує привілегії sudo:

« sudo apt install apache2 », на рисунку 4.4 зображено процес встановлення серверу на мою операційну систему.

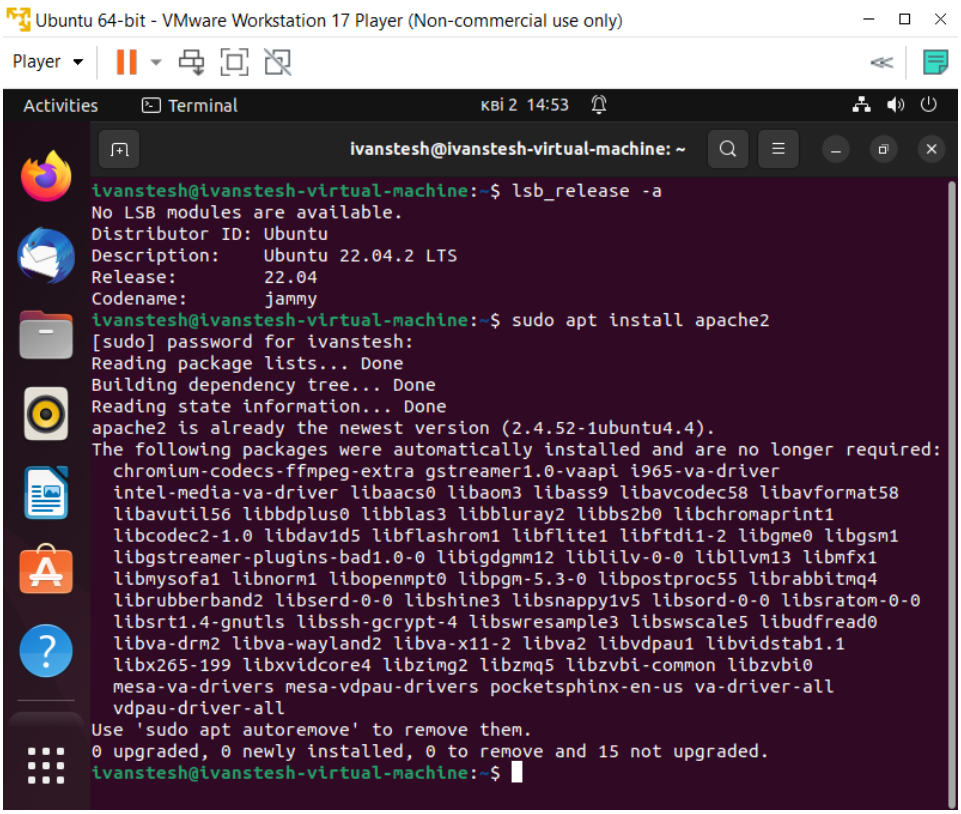

Рисунок 4.4 – установка серверу Apache2 на Ubuntu

Потрібно зробити перевірку чи правильно виконано установку, скориставшись командою:

« sudo systemcl status apache2 », результат відображено на рисунку 4.5

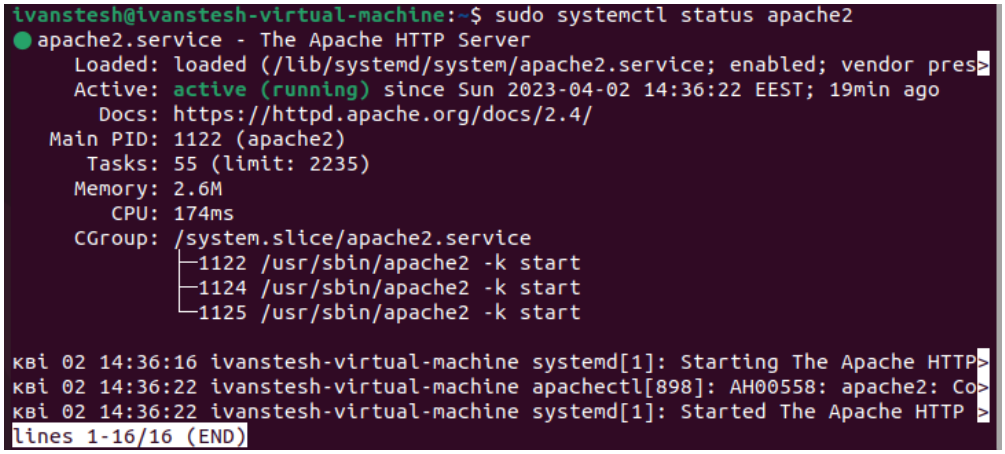

Рисунок 4.5 – Системна перевірка правильності встановлення Apache2

На рисунку добре видно, що сервер встановлено та вже запущено, щоб перейти до нього, потрібно відкрити браузер, в даному випадку це Firefox Web Browser, та ввівши в адресний рядок айпі системи (192.168.31.220), повинен відкритися запущений сервер Apache2. На рисунку 4.6 виконаю дані дії.

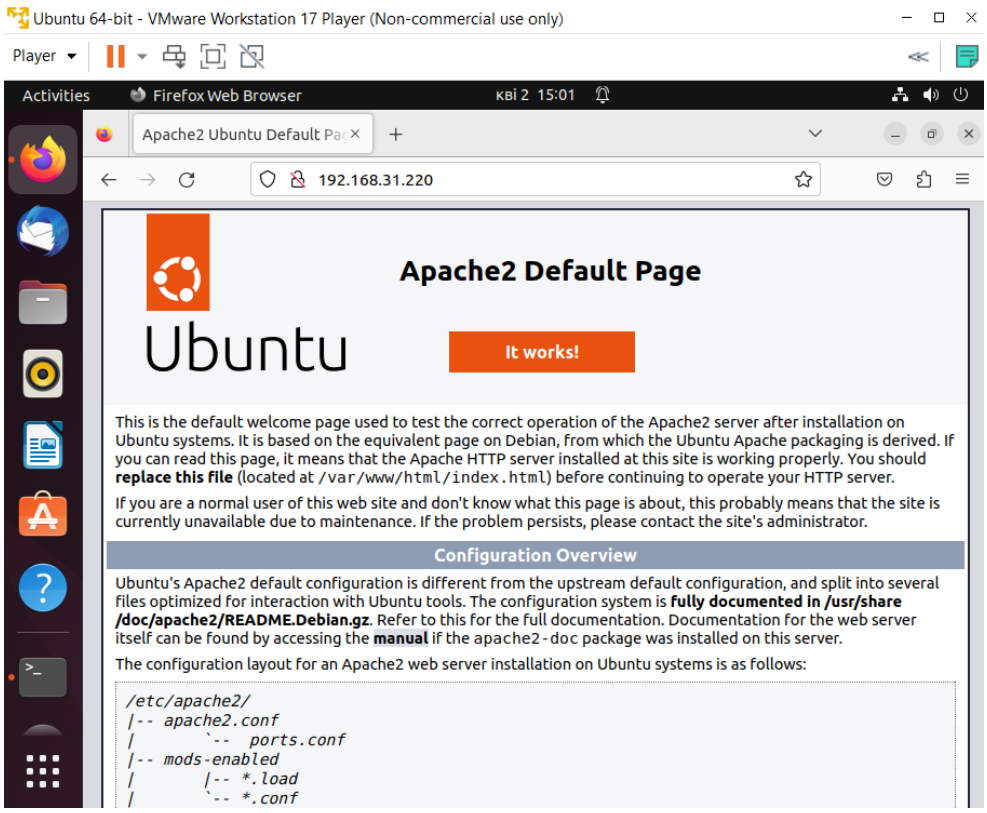

Рисунок 4.6 – перегляд сторінки запущеного серверу в браузері

Наступним кроком видозмінено саму сторінку на свій розгляд, написано власний HTML-код. Для цьогозмінено директорію в терміналі на /var/www/html, та відкрито файл index.html налаштування сторінки на сервері Apache2 командою:

« sudo nano index.html »

Результат створення нової сторінки показано на рисунку 4.7.

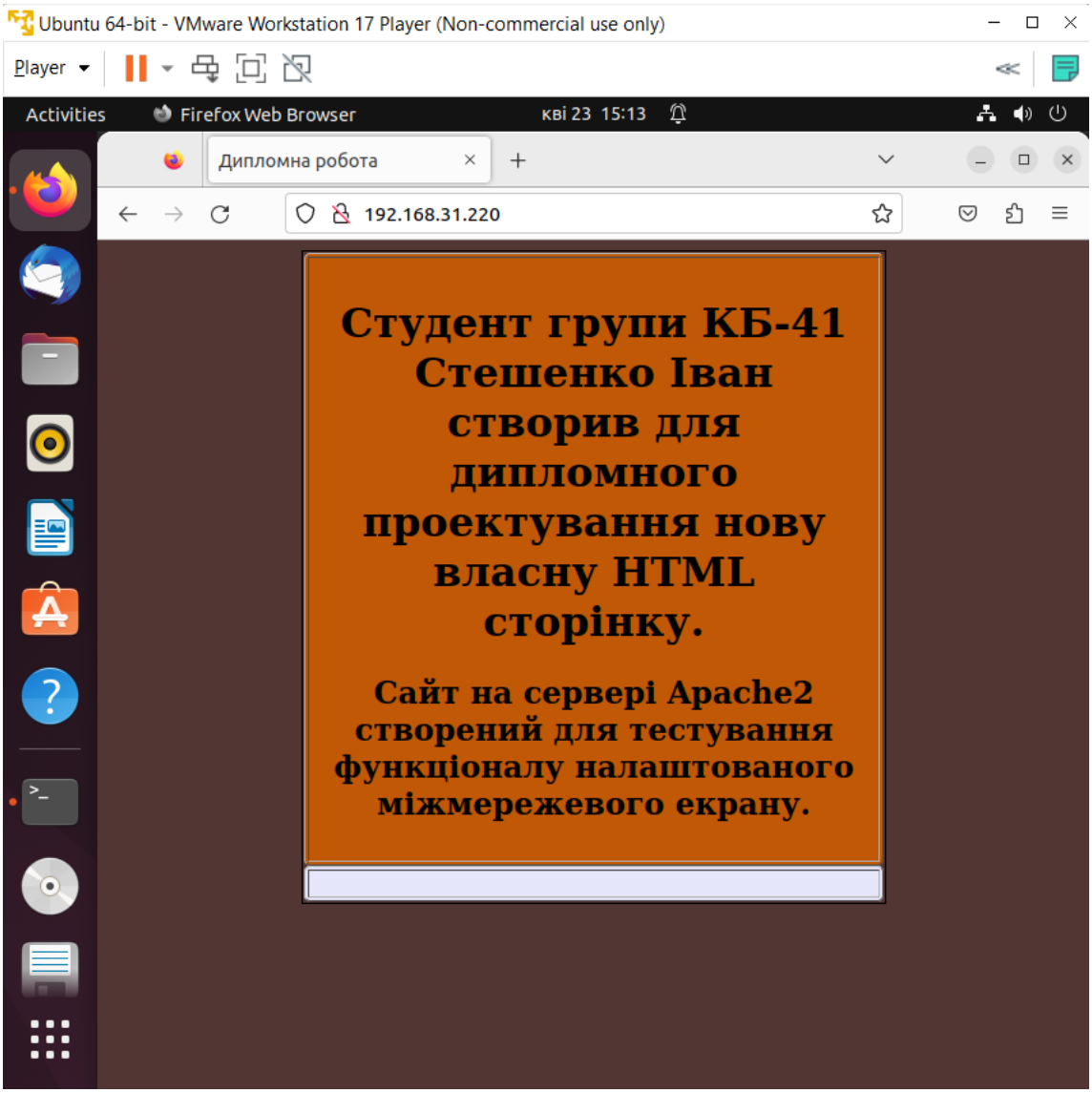

Рисунок 4.7 – Змінена на власний розгляд сторінки запущеного серверу

Виконавши всі дії, можно вважати, що встановлення і налаштування серверу завершено.

4.3 Налаштування таблиці Filter власного міжмережевого екрану.

В даному проекті роль міжмережевого екрану буде виконувати операційна система Ubuntu, тобто основна операційна система Windows буде здійснювати обмін пакетами з запущеним сервером Apache2 через налаштований на Ubuntu міжмережевий екран. Спочатку потрібно перевірити з'єднання однієї операційної системи з іншою, щоб дізнатися чи вони взаємодіють скористувавшися командою «ping 192.168…(ip)» з однієї системи до іншої та навпаки.

IP (Ubuntu): 192.168.31.220, назва інтерфейсу «ens37»

IP (Windows): 192.168.31.14

На рисунках 4.8 та 4.9 зображено перевірку взаємодії двух ОС.

```
ivanstesh@ivanstesh-virtual-machine:~$ ping 192.168.31.14
PING 192.168.31.14 (192.168.31.14) 56(84) bytes of data.
64 bytes from 192.168.31.14: icmp_seq=1 ttl=64 time=0.286 ms
64 bytes from 192.168.31.14: icmp_seq=2 ttl=64 time=0.364 ms
64 bytes from 192.168.31.14: icmp_seq=3 ttl=64 time=0.201 ms
64 bytes from 192.168.31.14: icmp_seq=4 ttl=64 time=0.255 ms
^C
--- 192.168.31.14 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3051ms
rtt min/avg/max/mdev = 0.201/0.276/0.364/0.058 ms
```

Рисунок 4.8 – Відправка пакетів з Ubuntu до Windows

```
C:\WINDOWS\system32>ping 192.168.31.220

Pinging 192.168.31.220 with 32 bytes of data:
Reply from 192.168.31.220: bytes=32 time<1ms TTL=64
Reply from 192.168.31.220: bytes=32 time<1ms TTL=64
Reply from 192.168.31.220: bytes=32 time<1ms TTL=64
Reply from 192.168.31.220: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.31.220:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Рисунок 4.9 – Відправка пакетів з Windows до Ubuntu

Після перевірки можно зробити висновок, що ОС коректно взаємодіють, тому можна починати налаштування міжмережевого екрану за допомогою утиліти iptables.

Спочатку відключається стандартна політика налаштування взаємодії з пакетами «INPUT», «OUTPUT» та «FORWARD», а саме пропуск всіх пакетів. Потрібно змінити параметр з ACCEPT на параметр DROP в ланцюгу INPUT,OUTPUT та FORWARD за допомогою команд:

«sudo iptables –P INPUT DROP»,

«sudo iptables –P OUTPUT DROP»,

«sudo iptables –P FORWARD DROP»,

Після виконання буде заборонена будь-яка взаємодія з пакетами в Ubuntu.

На рисунку 4.10 зображено введення команд та перевірка взаємодії з пакетами шляхом використання команд типу ping.

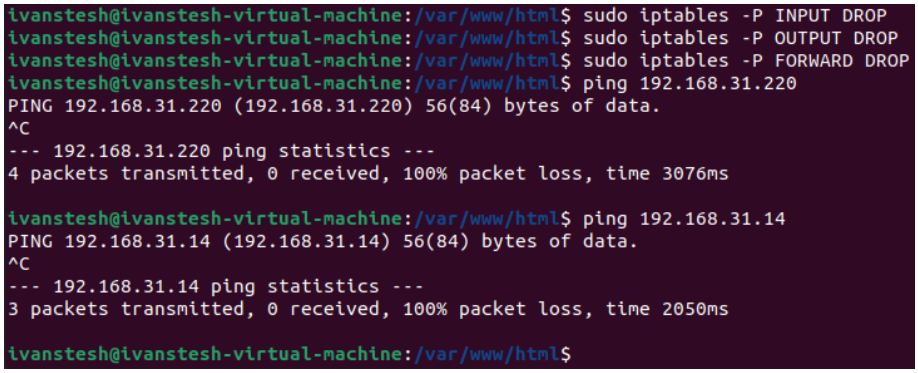

Рисунок 4.10 – Спроба відправити пакети на ОС Ubuntu після зміни правил

На рисунку 4.11 зображено відправка пакетів з ОС Windows до Ubuntu після зміни правил.

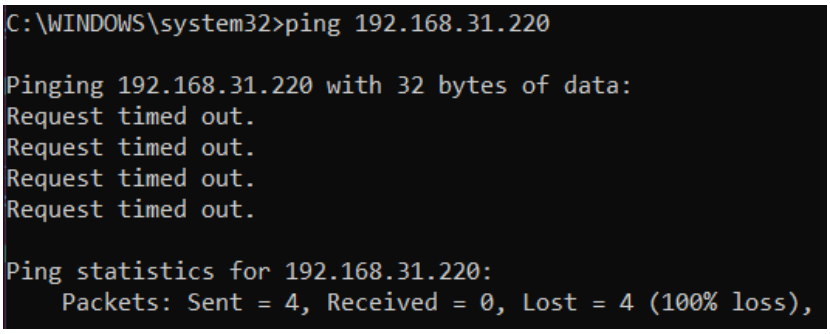

Рисунок 4.11 – Спроба відправити пакети на ОС Windows після зміни правил

Можна зробити висновок, що й справді ОС Ubuntu тепер не приймає і не відправляє пакети даних. Після цього почиється створення нових правил міжмережевого екрану для взаємодії з пакетами даних . Спочатку дозволяється ОС Ubuntu взаємодіяти самою з собою, тобто додається правило прийнятя пакетів з власного локального інтерфейсу. На рисунку 4.12 зображено перевірку як взаємодія налаштована зараз. (IP local iterface : 127.0.0.1)

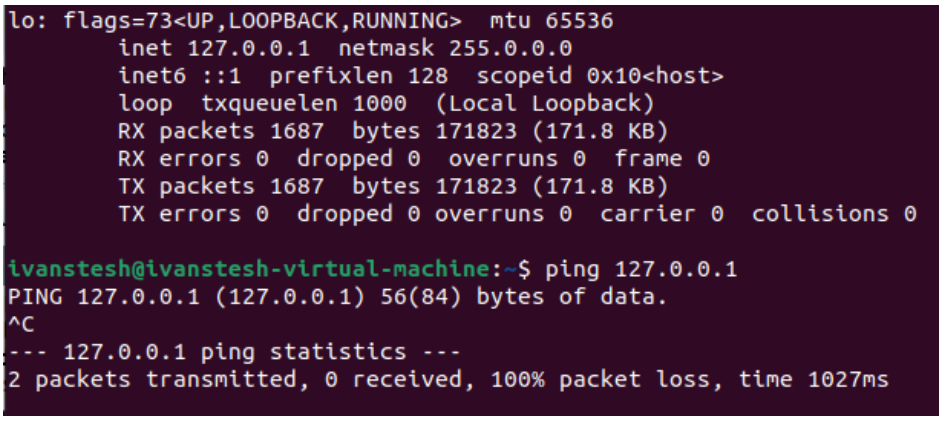

Рисунок 4.12 – перевірка зв’язку Ubuntu з власним локальним інтерфейсом

З рисунку видно що й насправді пакети не відправляються, виправлено цю ситуацію за допомогою наступних команд:

« sudo iptables –A INPUT –i lo –j ACCEPT »,

« sudo iptables –A OUTPUT –o lo –j ACCEPT »,

Результат зображено на рисунку 4.13. Пакети відправляються.

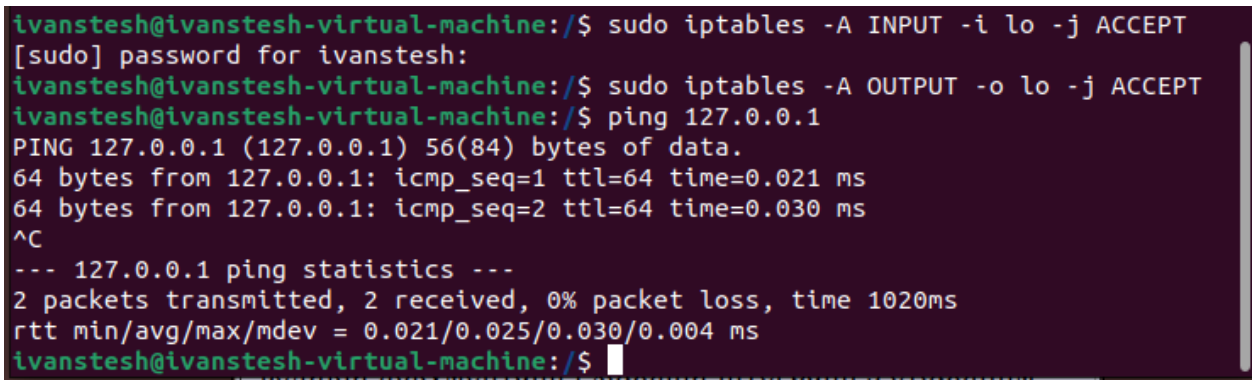

Рисунок 4.13 – Перевірка взаємодії після додання нових правил

Тепер додається правило для того щоб можно було відправляти пакети з даними до ОС Windows, в разі виконання команди «ping» буде надіслано echo запит на IP 192.168.31.14 (Windows), і ОС Ubuntu повинна отримати відповідь в вигляді пакету на цей запит, але політика INPUT наразі не приймає пакети з даними, а політика OUTPUT не відправляє. Тому щоб відправляти пакети, отримувати відповіді на згенеровані нами запити потрібно дозволити ОС Ubuntu приймати пакети зі станом RELATED або ESTABLISHED. Також потрібно додати дозвіл на взаємодії з IP-адресою ОС Windows. Дозволяється трафік, пов'язаний із встановленими з'єднаннями, використано наступні команди :

« sudo iptables –A INPUT –m conntrack –ctstate RELATED, ESTABLISHED –j ACCEPT ».

« sudo iptables –A OUTPUT –m conntrack –ctstate RELATED, ESTABLISHED –j ACCEPT ».

«sudo iptables -A INPUT -s 192.168.31.14 -j ACCEPT»

«sudo iptables -A OUTPUT -d 192.168.31.14 -j ACCEPT»

І відразу перевіряється спроможность робити запити на інший IP. Результат зображено на рисунку 4.14.

```
ivanstesh@ivanstesh-virtual-machine:/$ sudo iptables -A INPUT -s 192.168.31.14 -j ACCEPT
ivanstesh@ivanstesh-virtual-machine:/$ sudo iptables -A OUTPUT -d 192.168.31.14 -j ACCEPT
ivanstesh@ivanstesh-virtual-machine:/$ ping 192.168.31.14
PING 192.168.31.14 (192.168.31.14) 56(84) bytes of data.
64 bytes from 192.168.31.14: icmp_seq=1 ttl=64 time=0.259 ms
64 bytes from 192.168.31.14: icmp_seq=2 ttl=64 time=0.272 ms
^C
--- 192.168.31.14 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1026ms
rtt min/avg/max/mdev = 0.259/0.265/0.272/0.006 ms
```

Рисунок 4.14 – Спроба зробити запит до Windows після додання правил

Для того щоб дозволити трафік від встановлених з’єднань і пов’язаний з ними трафік потрібно встановити додаткові правила, а саме:

«sudo iptables -A INPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT»

«sudo iptables -A OUTPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT»

Також створю блокування визначеної IP-адреси, а саме заблоковано IP-адресу 192.168.1.100. Наступними командами:

«sudo iptables -A INPUT -s 192.168.1.100 -j DROP»

«sudo iptables -A OUTPUT -d 192.168.1.100 -j DROP»

Тепер потрібно налаштувати доступ ОС Windows до сервера Apache2 який був раніше запущений на ОС Ubuntu. Спочатку потрібно спробувати відвідати цей сервер без додання нових правил. Результат зображено на рисунку 4.15.

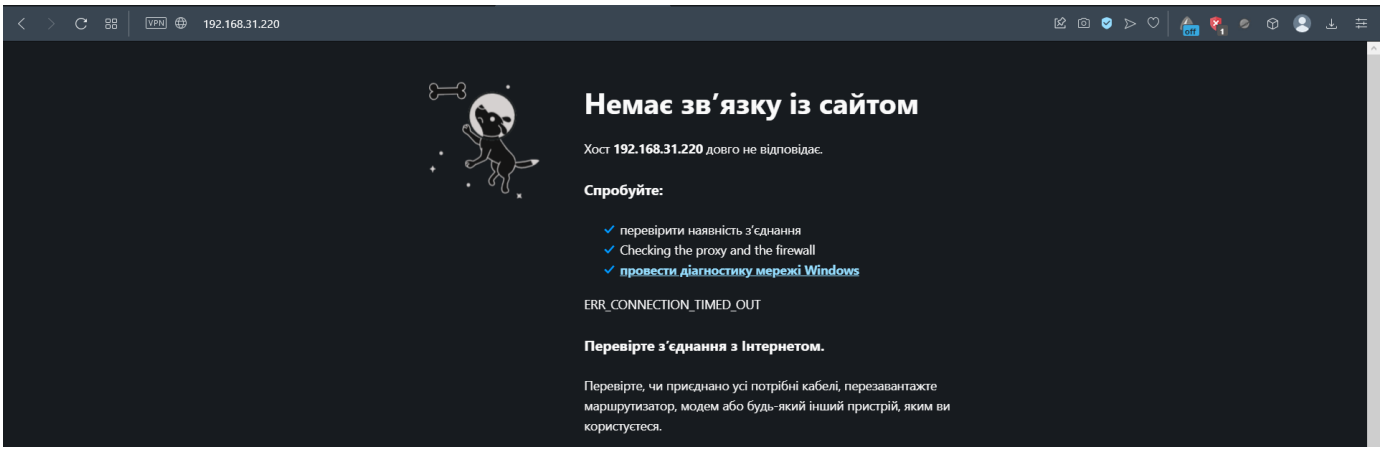

Рисунок 4.15 – Спроба зайти на Apache2 з Windows

З рисунку видно що немає доступу до цього серверу, щоб вирішити цю проблему потрібно створено правило для того щоб дозволити доступ до кокретного порту, а саме 80 порту, який відповідає за загальнопризнаний протокол інтернет зв'язку, протокол передачі гіпертексту HTTP-протокол. Також відкрито порти 22 та 443 для захищеного веб зв'язку та гарної роботи з TCP протоколом. До налаштувань міжмережевого екрану додаються наступні правила. після чого знову перевіряється доступ до серверу Apache2 з Windows. Встановлення нових правил:

«sudo iptables -A INPUT -p tcp -m multiport --dports 22,80,443 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT»

«sudo iptables -A OUTPUT -p tcp -m multiport --sports 22,80,443 -m conntrack --ctstate ESTABLISHED -j ACCEPT»

Після чого знову перевіряється доступ до серверу Apache2 з Windows.

Резльтат показаний на рисунку 4.16.

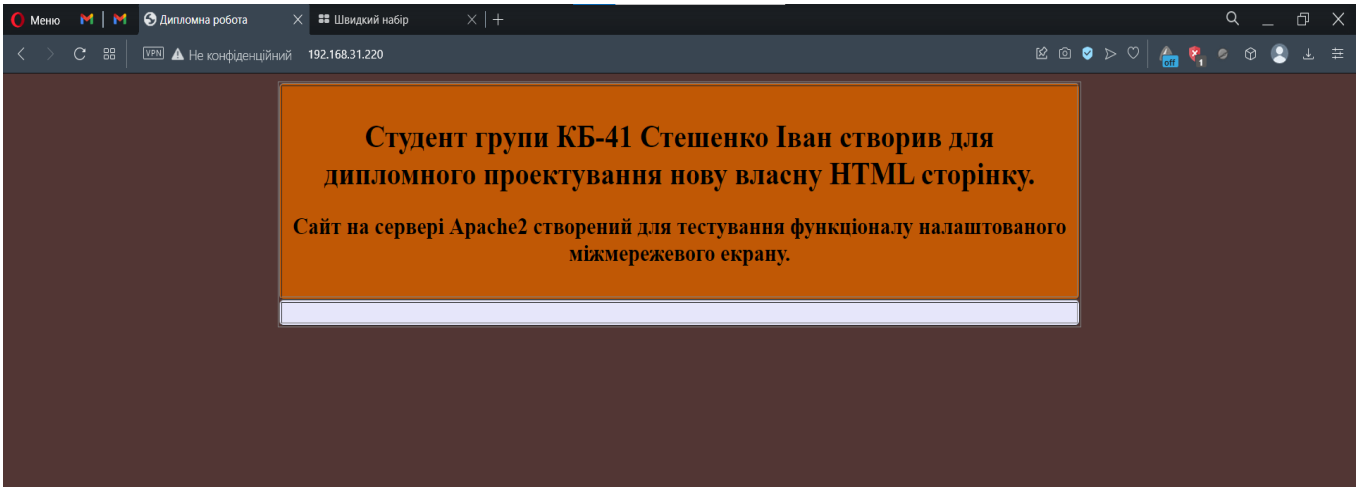

Рисунок 4.16 – Спроба зайти на Apache2 з Windows після додання правила

Можна зробити висновок що налаштування виконано правильно, адже з’єднання встановлено.

Щоб переглянути проміжковий список створених нами правил міжмережевого екрану iptables, та відразу побачити скільки пакетів по ним вже пройшло, потрібно скористуватися командою:

« sudo iptables –L -v » , на рисунку 4.17 зображено результат.

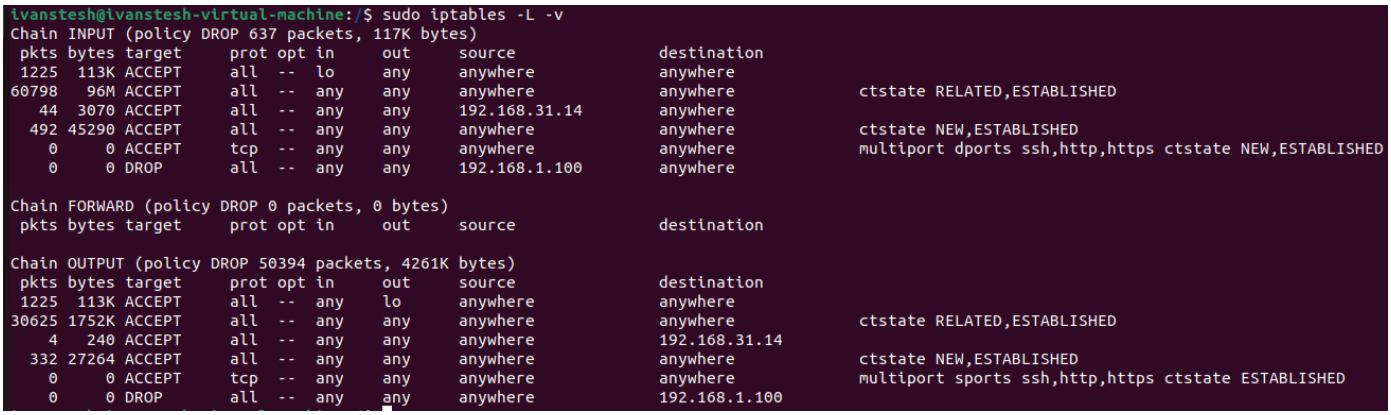

Рисунок 4.17 – Проміжковий перегляд створених правил в ME

Для того щоб зберегти створені налаштування міжмережевого екрану, потрібно використати наступні команди:

«sudo mkdir /etc/iptables»

«sudo iptables-save > /etc/iptables/firewall.conf»

На рисунку 4.18 зображено файл зі збереженими правилами налаштування міжмережевого екрану iptables.

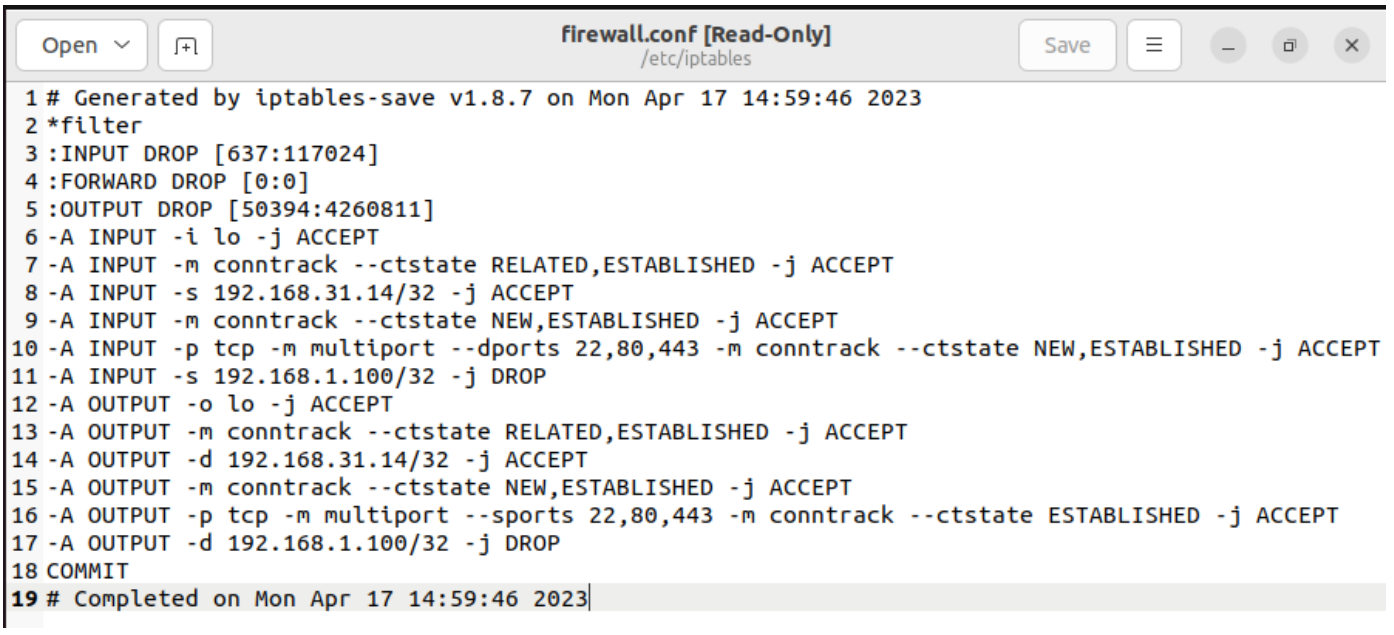

Рисунок 4.18 – Вміст файлу firewall.conf

В Додатку Б розміщено вміст даного файлу.

Для того щоб після перезапуску системи відновити налаштування iptables, потрібно скористатися наступною командою:

«sudo iptables-restore > /etc/iptables/firewall.conf»

Для того щоб налаштовані правила iptables були встановлені за замовчуванням, та завантажувалися відразу після запуску системи, потрібно відкрити файл з налаштуваннями системи наступною командою:

«vi /etc/network/interfaces»

Після відкриття додати в цей файл після рядка «iface lo net loopback», рядок з наступним змістом:

«pre-up iptables-restore < /etc/iptables/firewall.conf»

Таким чином, налаштування міжмережевого екрану завершено. Розібрано покрокове налаштування МЕ та написання правил для нього, також наявна інструкція по впровадженню в систему на постіній основі. Після цього потрібно обов’зяково провести тестування даного налаштування та правильність роботи. Потрібно також зазначити що даний функціонал МЕ не максимальний, його можно розвивати та робити більш гнучким та готовим до окремих, потрібний користувачу ситуацій.

5 ТЕСТУВАННЯ РОЗРОБЛЕНОГО РІШЕННЯ

5.1 Вимоги до функціонування міжмережевого екрану.

У даній роботі розроблено міжмережевий екран за допомогою утиліти командного рядка iptables. Створено правила налаштування МЕ, які дозволяють вільно проводити найпростіші взаємодії між сервером та ОС Windows, відфільтровуючи весь непотрібний трафік.

Нижче перечислені вимоги до створеного міжмережевого екрану:

- Насамперед це відкидання всього трафіку, який не дозволений написаними правилами фільтрації.
- Відкритий доступ ОС Ubuntu до власного локального інтерфейсу та змога запиту до нього задля обміну пакетами з даними.
- Вільна взаємодія ОС Ubuntu та ОС Windows, безперебійний та швидкий обмін даними.
- Стабільний доступ до відкритих правилами портів , а саме SSH (порт 22), HTTP (порт 80), HTTPS (порт 443).
- Блокування будь-якого з’єднання з забороненою правилами IP-адресою, а саме 192.168.1.100.

Після визначення вимог до створеного міжмережевого екрану, можно переходити до розробки методики тестування.

5.2 Проведення тестування міжмережевого екрану.

Щоб перевірити міжмережевий екран, буде використовано різні інструменти, такі як «ping» і «curl», щоб побачити, чи дозволено чи заблоковано трафік відповідно до створених правил. Тестування розробленного міжмережевого екрану:

1) Ping локального інтерфейсу (lo), перевірка доступу до нього. На рисунку 5.1 відображено результат перевірки.

```
ivanstesh@ivanstesh-virtual-machine:/$ ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.071 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.034 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.036 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.034 ms
^C
--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3077ms
rtt min/avg/max/mdev = 0.034/0.043/0.071/0.015 ms
```

Рисунок 5.1 – Ping локального інтерфейсу

2) Запит на веб сторінку в інтернеті (наприклад Google.com). (Очікуваний результат: Буде встановлено з’єднання та відображено вміст сторінки.) Результат тесту відображено на рисунку 5.2

```
ivanstesh@ivanstesh-virtual-machine:/$ curl google.com
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.com/">here</A>.
</BODY></HTML>
```

Рисунок 5.2 – Запит на сайт Google

3) Запит на заблоковану правилами міжмережевого екрану IP-адресу.
(192.168.1.100) (Очікуваний результат: з’єднання має бути розірвано через правило блокування.)
Результат відображено на рисунку 5.3

```
ivanstesh@ivanstesh-virtual-machine:/$ sudo curl 192.168.1.100
curl: (28) Failed to connect to 192.168.1.100 port 80 after 130049 ms: Connection timed out
```

Рисунок 5.3 – Запит на заблоковану адресу

5.3 Результати тестування.

Виконано тестування правильності роботи налаштованого міжмережевого екрану за допомогою запропонованої методики, зроблено декілька висновків за результатами даного тестування:

- Дві операційні системи Windows та Ubuntu які взаємодіють в практичній демонстрації налаштування міжмережевого екрану, вільно взаємодіють, при взаємному обміні пакетів з даними помилок не виникає.
- ОС Ubuntu вільно взаємодіє з локальним інтерфейсом (IP:127.0.0.1) запити надходять і Ubuntu отримує відповіді, отже дане правило написане вірно.
- При спросбі запиту на сайт Google.com за допомогою команди «curl» з’єднання встановлено та вміст сторінки відображено вірно, отже в налаштуваннях міжмережевого екрану правильно відкриті порти.
- Спроба з’єднання з заблокованою правилами міжмережевого екрану IP-адресою виявилася невдалою, було розірвано з’єднання. Отже, заборонне правило ME написане правильно.
- Всі розроблені правила для міжмережевого екрану працюють коректно, отже ME налаштований правильно.
- В перспективі можно додавати нові правила фільтрації трафіку та легко тестувати правильність реалізації.

ВИСНОВКИ

У сучасному світі, де мережеві технології використовуються в усіх сферах діяльності, міжмережеві екрани стають все більш необхідними для забезпечення безпеки мережі. Міжмережеві екрани є центральним елементом безпеки мережі, який відповідає за контроль руху даних між мережами, захист від зовнішніх загроз та захист від внутрішніх атак.

У дипломній роботі було досліджено модель мережі, її будову, основні загрози для безпеки інформаціїї в мережі та модель зловмисника. Розглянуто також основні принципи роботи міжмережевих екранів, їх функції та можливості. Були розглянуті різні типи міжмережевих екранів, такі як пакетні фільтри, мережеві екрани з іспекцією стану пакетів, проксі-сервери та програмно-апаратні міжмережеві екрани. Розглянуто переваги й недоліки кожного з типів міжмережевого екрану.

Було виявлено, що міжмережеві екрани є важливим засобом захисту мережі від зловмисників та загроз зовнішнього середовища. Вони забезпечують контроль доступу до ресурсів мережі, моніторинг та аналіз мережевого трафіку, фільтрацію небезпечних даних та виявлення атак на мережу.

У практичній демонстрації було запущено власний сервер Apache2, створено нову головну сторінку сайту. Розроблено й налаштовано міжмережевий екран зі збереженням стану (Stateful Firewall) за допомогою утиліти командного рядку iptables. Визначино правила дозволу або блокування трафіку на основі різних критеріїв, таких як IP-адреси джерела/одержувача, порти чи служби та стан з’єднання. Насамперед це відкидання всього трафіку, який не дозволений написаними правилами фільтрації. Налаштована також вільна взаємодія ОС Ubuntu та ОС Windows, безперебійний та швидкий обмін даними. Крім того, налаштовано стабільний доступ до відкритих правилами портів, а саме SSH (порт 22), HTTP (порт 80), HTTPS (порт 443).

Тестування міжмережевого екрану за допомогою різних інструментів і сценаріїв підтвердило, що правила працюють належним чином.

Важливо відзначити, що у ході подальшої роботи планується розробити комплексну модель міжмережевого екрану, і багато інших правил і конфігурацій можна додати, щоб створити більш складний функціональний профіль міжмережевого екрану. Крім того, iptables може бути складним і заплутаним для тих, хто з ним не знайомий, тому важливо ретельно перевірити та зрозуміти правила, перш ніж застосовувати їх у робочому середовищі.

ПЕРЕЛІК ПОСИЛАНЬ

1. Мясіщев О. А., Мартинюк О. О., Гіневська Н. М. Аналіз захищеності комп'ютерних мереж на основі побудови дерева атак. *Вісник Хмельницького національного університету*. 2016. С. 243.
2. Cybersecurity investment to grow by 13% in 2023. Canalys Newsroom. URL: https://canalys.com/newsroom/cybersecurity-forecast-2023 (дата звернення: 13.03.2023).
3. ISO/IEC 27001:2015. Інформаційні технології. методи захисту системи управління інформаційною безпекою. вимоги. Чинний від 2017-01-01. Вид. офіц. 2015.
4. НД ТЗІ 2.7-009-09. Методичні вказівки з оцінювання функціональних послуг безпеки в засобах захисту інформації від несанкціонованого доступу. Чинний від 2009-07-24. Вид. офіц. Київ, 2009. 231 с.
5. ISO/IEC 7498-1:1994. Information technology – open systems interconnection – basic reference model: the basic model. Чинний від 1994-11-01. Вид. офіц. 1994.
6. Про електронні комунікації : Закон України від 16.12.2021 № 3549-1. URL: https://ips.ligazakon.net/document/JH2QL1AA?an=3 (дата звернення: 15.03.2023)
7. Meghanathan N. Network security. *Network security technologies*. 2014. С. 174–203. URL: https://doi.org/10.4018/978-1-4666-4789-3.ch011 (дата звернення: 20.03.2023).
8. Олещенко Л. М. Організація комп'ютерних мереж. Конспект лекцій. Київ : КПІ ім. Ігоря Сікорського, 2018. 225 с.
9. Матов О. Я. Модель загроз у розподілених мережах. Київ : Національний авіаційний університет, 2008. 92 с.

10. Firewall Statistics 2023 - Everything You Need to Know. URL: https://webinarcare.com/best-firewall-software/firewall-statistics/ (дата звернення: 05.04.2023).
11. Жураковський Б.Ю., Зенів І.О. Розробка та реалізація мережних протоколів. Київ : КПІ ім. Ігоря Сікорського, 2020. 462 с.
12. Заплотинський Б. О. Основи інформаційної безпеки. Конспект лекцій. Київ : КІІВіП НУ “ОЮА”, 2017. 128 с.
13. Бурячок В.Л., Толубко В.Б., Хорошко В.О., Толюпа С.В. Інформаційна та кібербезпека : підручник. Київ : Державний університет телекомунікацій, 2015. 288 с.
14. Положення про порядок розроблення, виробництва та експлуатації засобів криптографічного захисту конфіденційної інформації та відкритої інформації з використанням електронного цифрового підпису. *Liga 360*. URL: https://ips.ligazakon.net/document/view/re14129?an=188&ed=2007_07_20 (дата звернення: 15.04.2023).
15. НД ТЗІ 1.1-002-99. Загальні положення щодо захисту інформації в комп'ютерних системах від несанкціонованого доступу. Чинний від 1999-04-28. Вид. офіц. Київ : ДСТСЗІ СБ України, 1999. 21 с.
16. Корпань Я. В. Класифікація загроз інформаційній безпеці в комп’ютерних системах при віддаленій обробці даних. Черкаси, 2015. С. 1-8. URL: http://dspace.nbuv.gov.ua/bitstream/handle/123456789/131565/04-Korpan.pdf?sequence=1 (дата звернення: 16.04.2023).
17. Рондалєв Д., Нарежній О., Мелкозьорова О. Особливості функціонування корпоративного міжмережевого екрану та питання взаємодії з системою IDS. Харків : ХНУ ім. В.Н. Каразіна, 2019. 11 с.
18. Інформаційна безпека в комп’ютерних мережах / О. А. Смірнов та ін. Кропивницький : ЦНТУ, 2020. 295 с.

19. В. Б. Вишня, О. С. Гавриш, Е. В. Рижков. Основи інформаційної безпеки : навч. посібник. Дніпро : ДДУВС, 2020. 128 с.
20. Технології забезпечення безпеки мережевої інфраструктури / В. Л. Бурячок та ін. Київ : Київ. ун-т ім. Бориса Грінченка, 2019. 218 с.
21. Сучасні інформаційно-комунікаційні технології / Г. Г. Швачич та ін. Дніпро : НМетАУ, 2017. 230 с.
22. Тестування та контроль якості (QA) вбудованих систем / І. А. Клименко та ін. Київ : КПІ ім. Ігоря Сікорського, 2022. 75 с.
23. Iptables tutorial: ultimate guide to linux firewall. *Knowledge Base by phoenixNAP*. URL: https://phoenixnap.com/kb/iptables-tutorial-linux-firewall (дата звернення: 24.04.2023).
24. Iptables tutorial - securing ubuntu VPS with linux firewall. *Hostinger Tutorials*. URL: https://www.hostinger.com/tutorials/iptables-tutorial (дата звернення: 25.04.2023).
25. Жилін А. В., Шаповал О. М., Успенський О. А. Технології захисту інформації в інформаційно-телекомунікаційних системах. Київ : КПІ ім. Ігоря Сікорського, 2020. 213 с.
26. Apache HTTP server version 2.4documentation - apache HTTP server version 2.4. *Welcome! - The Apache HTTP Server Project*. URL: https://httpd.apache.org/docs/2.4/en/ (дата звернення: 28.04.2023).

ДОДАТОК А

HMTL-код для власної створеної сторінки на сервері Apache2.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Дипломна робота</title>
</head>
<body bgcolor="523634">


<table
border="1"
align="center"
rules="rows"
style="width:60%;">
<tr>
<td>


<table
border="1"
bgcolor="#C05805"
cellpadding="10"
```

```
style="width:100%; border-radius:5px;">

<tr>

<th>

<h1> Студент групи КБ-41 Стешенко Іван створив для дипломного проектування нову власну HTML сторінку.</h1>

<h3> Сайт на сервері Apache2 створений для тестування функціоналу налаштованого міжмережевого екрану. </h3>

</th>

</tr>

</table>


<table

border="1"

bgcolor="#e6e6fa"

cellpadding="10"

style="width:100%; border-radius:5px;">

<tr>

<td

rowspan="2"

style="width:80%">


</body>

</html>
```

ДОДАТОК Б

Вміст файлу firewall.conf

```
# Generated by iptables-save v1.8.7 on Mon Apr 17 14:59:46 2023
*filter
:INPUT DROP [637:117024]
:FORWARD DROP [0:0]
:OUTPUT DROP [50394:4260811]
-A INPUT -i lo -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.31.14/32 -j ACCEPT
-A INPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
-A INPUT -p tcp -m multiport --dports 22,80,443 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.1.100/32 -j DROP
-A OUTPUT -o lo -j ACCEPT
-A OUTPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A OUTPUT -d 192.168.31.14/32 -j ACCEPT
-A OUTPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -p tcp -m multiport --sports 22,80,443 -m conntrack --ctstate ESTABLISHED -j ACCEPT
-A OUTPUT -d 192.168.1.100/32 -j DROP
COMMIT
# Completed on Mon Apr 17 14:59:46 2023
```