Харківський національний університет імені В.Н. Каразіна

Факультет комп'ютерних наук

Кафедра безпеки 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 архітектури і побудова міжмережевого екрану»

оцінка « » Керівник проф.Олійников Р.В
(прізвище та ініціали/підпис)

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

Лемешко О.В._____ Виконавець студентка групи КБ-41

Данилова В.І.
(прізвище та ініціали/підпис)

Харків – 2023

РЕФЕРАТ

Пояснювальна записка містить 60 сторінок, 20 рисунків та 23 джерела та 1 додаток.

Метою дипломної роботи є дослідженння архітектури та розробка міжмережевого екрану.

Об'єктом дослідження дипломної роботи є архітектура міжмережевих екранів і їх побудова. Міжмережевий екран є центральним елементом захисту мережі, який контролює передачу даних між різними мережами та забезпечує фільтрацію пакетів даних згідно з визначеними правилами та політиками безпеки. Основною метою міжмережевого екрана є захист мережі від несанкціонованого доступу, шкідливих атак та недозволених дій.

Методи дослідження. Для досягнення мети дипломної роботи було використано наступні методи дослідження:

1) Аналіз існуючих архітектур міжмережевих екранів: у цьому етапі проводився огляд та аналіз різних архітектур міжмережевих екранів, що використовуються в сучасних мережах. Цей аналіз дав змогу виявити сильні та слабкі сторони існуючих рішень.
2) Розробка архітектури: на основі отриманих результатів аналізу існуючих архітектур була розроблена архітектура міжмережевого екрана.
3) Реалізація міжмережевого екрана: на основі розробленої архітектури було проведено реалізацію міжмережевого екрана. Для цього були використані сучасні технології та інструменти, що дозволили побудувати ефективну та надійну систему захисту мережі.
4) Тестування та оцінка результатів: розроблений міжмережевий екран був підданий тестуванню з метою оцінки його ефективності та надійності

Результатом даної дипломної роботи є розробка архітектури міжмережевого екрана та побудова системи, що використовує цю архітектуру. Розроблений міжмережевий екран має покращені можливості захисту мережі. Під час тестування було підтверджено ефективність розробленої системи, яка забезпечує надійний рівень захисту мережі від потенційних загроз.

Ключові слова: КОМП’ЮТЕРНА МЕРЕЖА, МІЖМЕРЕЖЕВИЙ ЕКРАН, МОДЕЛЬ ЗЛОВМИСНИКА, АТАКА, КІБЕРБЕЗПЕКА

ABSTRACT

The explanatory note contains 60 pages, 20 figures and 23 references and 1 addition.

The purpose of the thesis is to study the architecture and development of a firewall.

The object of the thesis is the architecture of firewalls and their construction. A firewall is a central element of network security that controls the transmission of data between different networks and provides filtering of data packets according to specific security rules and policies. The main purpose of the firewall is to protect the network from unauthorized access, malicious attacks and unauthorized actions.

Research methods. To achieve the purpose of the thesis, the following research methods were used:

1) Analysis of existing firewall architectures: this stage involved a review and analysis of various firewall architectures used in modern networks. This analysis allowed us to identify the strengths and weaknesses of existing solutions and identify areas for improvement.

2) Architecture development: based on the results of the analysis of existing architectures, a firewall architecture was developed.

3) Implementation of the firewall: based on the developed architecture, the firewall was implemented. For this purpose, modern technologies and tools were used to build an effective and reliable network protection system. The implementation included setting up filtering rules, access control and other functions necessary to ensure security.

4) Testing and evaluation of results: the developed firewall was tested to evaluate its effectiveness and reliability

The result of this thesis is the development of a firewall architecture and the construction of a system that uses this architecture. The developed firewall has improved network security capabilities. During testing, the effectiveness of the developed system was confirmed, which provides a reliable level of network protection against potential threats.

Keywords: COMPUTER NETWORK, FIREWALL, ATTACKER MODEL, ATTACK, CYBERSECURITY

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ІС	-	Інформаційна системи
МЕ	-	Міжмережевий екран
ОС	-	Операційна система
API	-	Application Programming Interface
AWS	-	Платформа хмарних обчислень Amazon Web Services
DDoS	-	Distributed Denial of Service
HTML		Hypertext Markup Language
HTTP	-	Hypertext Transfer Protocol
HTTPS	-	Hypertext Transfer Protocol Secure
LDAP	-	Lightweight Directory Access Protocol
NGF	-	Next-Generation Firewall
OSI	-	Open Systems Interconnection
SQL	-	Structured Query Language
SSL	-	Secure Sockets Layer
SSRF	-	Server-Side Request Forgery
TLS	-	Transport Layer Security
VPN	-	Virtual Private Network
TCP		Transmission Control Protocol

ВСТУП

Технології більше і більше впливають на наше життя, тому кібербезпека стає надзвичайно важливою темою сьогодення. Зростаюча кількість інтернет мереж та розширення їх функціональних можливостей призводять до збільшення кількості потенційних вразливостей, які можуть бути використані для незаконного доступу до системи або викрадення конфіденційної інформації. Саме це вимагає створення ефективних і надійних механізмів захисту, здатних забезпечити безпеку інформації та захистити мережі від потенційних загроз. Одним із ефективних засобів захисту від загроз є використання міжмережевих екранів. Міжмережевий екран (міжмережевий екран) – центральний компонент інфраструктури кібербезпеки, що контролює трафік між різними мережами та фільтрує його за заданими правилами і політикою безпеки. Він функціонує як перша лінія оборони, що забеспечує захист мережі від несанкціонованого доступу, вторгнення зловмисників, вірусів та інших загроз. Його функції - блокування небажаного трафіку та виявлення потенційно небезпечних пакетів даних.[1]

З огляду на сучасну кібербезпеку, розробка архітектури та побудова міжмережевого екрану набуває актуальності та має хороші перспективи для розвитку. Це один з ключових елементів веб-дизайну, що дозволяє створювати багатовимірні та інтерактивні веб-додатки.[2] Міжмережевий екрани (також відомі як адаптивні екрани) можуть адаптувати веб-додатки до різних пристроїв та екранів, таких як комп’ютери, смартфони, планшети, телевізори та переносні пристрої. Вони автоматично змінюються відповідно до характеристик пристрою та розміру екрану, щоб забезпечити оптимальну взіємодію з користувачем.

Зловмисники постійно розвивають нові методи та техніки атак, спрямовані на обхід захисних механізмів і проникнення в систему. Отже для забезпечення ефективного захисту мереж інформаційних систем необхідно розробляти та вдосконалювати міжмережеві екрани, що забеспечують надійну фільтрацію трафіку та виявлення потенційно небезпечних сигналів.

Метою даної роботи є дослідження архітектури та побудова ефективного міжмережевого екрану для забезпечення високого рівня кібербезпеки. В роботі розглянуто існуючі методи та підходи до побудови міжмережевих екранів, проаналізовано їх ефективність та виявлено недоліки.

1 ОПИС ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ТА МОДЕЛЬ ЗАГРОЗ

1.1 Опис інформаційної технології

Багато організацій знаходяться в процесі розгляду можливості інтеграції своїх локальних та корпоративних мереж в глобальну мережу. Оскільки використання глобальної мережі в комерційних цілях та передача конфіденційної інформації стають все більш поширеними, стає важливим розробити ефективну систему захисту інформації. Сучасні глобальні мережі використовуються для передачі комерційної інформації різного рівня конфіденційності, наприклад, для зв'язку з віддаленими офісами центрального офісу організації або для створення веб-сторінок з рекламою та діловими пропозиціями. [1]

Розширення глобальних мереж призвело до значного зростання кількості користувачів і збільшення кількості атак на комп'ютери, які підключені до Інтернету. Щорічні збитки, що виникають внаслідок недостатнього рівня захисту комп'ютерів, оцінюються у мільйонах доларів.[3] При підключенні локальної або корпоративної мережі до Інтернету виникає потреба у забезпеченні безпеки цієї мережі. Інтернет був розроблений як відкрита система для вільного обміну інформацією. Через свою відкритість, Інтернет надає зловмисникам значно більше можливостей порівняно з традиційними інформаційними системами. Тому питання захисту мереж і їх компонентів стає дуже важливим і актуальним. На сьогоднішній день безпека мережі стоїть перед суттєвими викликами. Корпоративні мережі та їх ресурси постійно знаходяться під загрозою мережевих атак та зараження шкідливим ПО. Мета та завдання зловмисників можуть бути різноманітними, а методи вторгнень розмаїтими - від вірусів, що поширюються електронною поштою, до складних атак. Небезпеку становлять не лише хакери, але й власні співробітники.[2]

Для забезпечення ефективного захисту та мінімізації можливих збитків, використовуються спеціальні механізми безпеки. Один з основних інструментів для розв'язання цих проблем - міжмережеві екрани, відомі також як "Firewall". Міжмережевий екран - це комплексна апаратно-програмна система, яка розділяє мережу на безпечні зони та контролює трафік, що пролягає через неї, згідно з заданими правилами. Додатково, для забезпечення безпеки, часто застосовуються програмні технології, такі як IPSec, VPN, Web Proxy, Antivirus та інші.[4]

Головною причиною встановлення міжмережевого екрана в приватній мережі практично завжди є бажання користувача захистити мережу від несанкціонованого проникнення. У більшості випадків, мережа захищається від нелегального доступу до системних ресурсів, а також від передачі інформації без дозволу її власника. Для багатьох підприємств, що підключаються до Інтернету, експорт інформації є першочерговою проблемою.[6]

Деякі організації обирають найпростіший спосіб уникнення подібних проблем - просто не підключатися до Інтернету. Однак, це не є належним рішенням. Якщо мережа є децентралізованою або має недостатнє управління, будь-який співробітник компанії, який має доступ до швидкісного модему, може легко підключитися до Інтернету за допомогою SLIP, що призводить до порушення безпеки всієї мережі.

1.2 Модель зловмисника

Розвиток технологій несподівано стрімко прогресує, і не кожна особа має можливість дістатися та ознайомитися з інформацією щодо своєї безпеки в онлайн-середовищі. Однак кіберзлочинці не стоять на місці, вони постійно розробляють нові методики та виявляють уразливості в обладнанні та програмному забезпеченні, використовуючи їх у своїх користях. На сьогоднішній

день існує значна кількість різноманітних атак, які здійснюються хакерами з метою отримати доступ до конфіденційної інформації, зламати веб-сайти, захопити контроль над мережами та інші негативні дії. Для ефективного захисту від цих різновидів атак важливо розуміти, як вони функціонують та хто саме може намагатися отримати доступ до вашої інформації.[8]

Відповідно до положень, викладених у пункті 8 процедури розроблення, виробництва та експлуатації криптографічних засобів захисту інформації, затверджених Наказом Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 20.07.2007 № 141 (у редакції наказу Адміністрації Державної служби спеціального зв’язку та захисту інформації України від 14.12.2015 № 767), визначаються чотири рівні можливостей порушника:

- нульовий рівень - ненавмисне порушення конфіденційності, цілісності та підтвердження авторства інформації;
- перший рівень - порушник має обмежені кошти та самостійно створює засоби, розробляє методи атак на засоби КЗІ, а також інформаційно-комунікаційні системи із застосуванням поширених програмних засобів та електронно-обчислювальної техніки;
- другий рівень - порушник корпоративного типу має змогу створення спеціальних технічних засобів, вартість яких співвідноситься з можливими фінансовими збитками, що виникатимуть від порушення конфіденційності, цілісності та підтвердження авторства інформації, зокрема при втраті, спотворенні та знищенні інформації, що захищається. У цьому разі для розподілу обчислень при проведенні атак можуть застосовуватися локальні обчислювальні мережі;

- третій рівень - порушник має науково-технічний ресурс, який прирівнюється до науково-технічного ресурсу спеціальної служби економічно розвинутої держави.[2]

Ці зловмисники можуть становити різні загрози для мережі з міжмережевими екранами, зокрема:

- Отримання несанкціонованого доступу до мережі та систем, що може призвести до крадіжки конфіденційної інформації, фінансових даних або інтелектуальної власності;
- Зараження мережі шкідливим програмним забезпеченням, таким як віруси, троянські програми або рандомайзери, що може призвести до пошкодження даних;
- Здійснення DDoS-атак (атак з відмовою в обслуговуванні), спрямованих на перевантаження мережі або серверів, що призводить до недоступності системи для легітимних користувачів;
- Викрадення або підробка ідентифікаційних даних, таких як паролі, для отримання несанкціонованого доступу до систем або приватної інформації;
- Встановлення шпигунського програмного забезпечення або перехоплення трафіку мережі, що дозволяє зловмисникам здійснювати шпигунство, перехоплювати конфіденційну інформацію або зловживати даними користувачів;
- Використання мережі для поширення шкідливого програмного забезпечення, спаму, фішингу або інших видів кібератак на інші мережі або користувачів;

- Порушення конфіденційності, цілісності або доступності даних шляхом злому системи, внесення змін до конфігурацій, пошкодження файлів або видалення даних.[10]

Враховуючи ці загрози, захист мережі з міжмережевими екранами від зловмисників включає в себе впровадження ефективних політик безпеки, використання механізмів автентифікації та авторизації, шифрування даних, моніторинг мережевого трафіку, виявлення та відповідь на кібератаки, резервне копіювання даних і регулярні оновлення програмного забезпечення та обладнання. [3]

1.3 Модель загроз

Міжмережевий екрани (WAF) виконують роль першої лінії захисту від атак на веб-додатки. Вони забезпечують захист трафіку, який передається протоколами передачі гіпертексту (HTTP) і захищеного протоколу передачі гіпертексту (HTTPS) між веб-додатками та Інтернетом. Ось сім найпоширеніших атак, на які призначені WAF для захисту:

1) Ін'єкційні атаки: Ін'єкційні атаки, такі як SQL, NoSQL, атаки на ОС та атаки на протокол доступу до каталогів (LDAP), є популярними серед хакерів. Ці атаки виникають, коли підозрілі дані вставляються в додаток у вигляді команди або запиту. Шкідливі дані можуть обманути інтерпретатор і спричинити виконання небажаних команд або отримання доступу до даних без належного дозволу.

   Серед ін'єкційних атак найпоширенішою є SQL-ін'єкція, яка здійснюється шляхом відправки шкідливого коду на сервер бази даних. Скрипти для SQL-ін'єкцій широко поширені і легко доступні, тому

майже кожен, хто має доступ до Інтернету, може їх використовувати. Ця атака є простою і швидкою в реалізації. [4]

2) Атаки на передбачування розміщення ресурсів - це метод атаки, який використовується для виявлення прихованого вмісту або функціональності чи контенту на веб-сайті. Зловмисники, застосовуючи метод грубої сили та раціональних обгрунтованих припущень, намагаються вгадати імена файлів і каталогів, які не призначені для публічного доступу та перегляду. Це може бути досягнуто шляхом систематичних спроб різних комбінацій імен з використанням загальних конвенцій іменування файлів та стандартних місць розташування.

   Ці файли, які можуть бути розкриті, включають тимчасові файли, файли резервних копій, журнали, адміністративні розділи сайту, конфігураційні файли, демо-версії, файли зразків та інші. Виявлення цих файлів може привести до розкриття конфіденційної інформації про веб-сайт, внутрішні компоненти веб-додатку, інформацію бази даних, паролі, імена комп'ютерів та шляхи до файлів у конфіденційних областях. Це дозволяє зловмисникам визначити структуру сайту, що може призвести до подальших вразливостей сайту, а також розкриття цінної інформації про середовище та користувачів. Цей метод також відомий як примусове сканування, примусовий перегляд, нумерація файлів або нумерація каталогів.[11]

3) HTTP DDoS (флуд) - це вид розподіленої атаки на відмову в обслуговуванні, який зловмисники використовують для нападу на веб-сервери та додатки. Атака HTTP Flood полягає в надсиланні великої кількості HTTP-запитів на цільову веб-сторінку, завантажуючи сервер запитами і перевантажуючи його.

Під час атаки HTTP Flood, HTTP-клієнт (наприклад, веб-браузер) взаємодіє з додатком або сервером, надсилаючи HTTP-запити, такі як "GET" або "POST". Головна мета атаки - позбавити законних користувачів доступу до ресурсів сервера, змушуючи сервер витратити якомога більше ресурсів на обробку атаки. Ці запити часто надсилаються великими обсягами через ботнети, що збільшує загальну потужність атаки.[8]

DDoS-атаки HTTP Flood можуть бути одними з найскладніших і найнебезпечніших загроз, з якими стикаються веб-сервери сьогодні. Важко відрізнити легітимний HTTP-трафік від зловмисного для мережевого обладнання, що забезпечує безпеку, що може призводити до численних помилкових відповідей, якщо не обробляти його належним чином. Механізми виявлення на основі швидкості також не можуть ефективно виявляти цей тип атаки, оскільки обсяг HTTP Flood-трафіку може бути нижчим за поріг виявлення. Тому для ефективного виявлення потрібен багатопараметричний підхід, такий як виявлення на основі швидкості або незалежне від швидкості виявлення. [4]

Сьогодні більшість інтернет-трафіку зашифрована, і більшість атак HTTP Flood відбуваються через HTTPS протокол, що забезпечує шифрування.[4] Зашифровані флуди не тільки є потужнішими через велику кількість серверних ресурсів, необхідних для їх обробки, але також додають додатковий рівень складності для зменшення впливу таких атак. Оскільки DDoS-захист зазвичай не може перевірити вміст HTTPS-запитів без повного дешифрування всього трафіку, це ускладнює процес виявлення та пом'якшення наслідків атаки.

Отже, HTTP DDoS (флуд) - це атака, яка полягає в перенавантаженні веб-сервера шляхом надсилання великої кількості HTTP-запитів. Ця атака може бути особливо складною для виявлення та захисту, особливо якщо вона відбувається через зашифровані канали.[12]

4) Десинхронізація HTTP-запитів є методом атаки, який впливає на обробку послідовностей HTTP-запитів веб-сайтом, які отримані від одного або декількох користувачів. Цей метод дозволяє зловмиснику "переправляти" запити на веб-сервер так, щоб проміжний пристрій між зловмисником і веб-сервером не був відомий про це. Уразливості, пов'язані з переправленням HTTP-запитів, часто дозволяють зловмисникам обійти засоби контролю безпеки, втручатися в сеанси інших користувачів, отримувати несанкціонований доступ до конфіденційної інформації та компрометувати інших користувачів додатку. Це є серйозною вразливістю, яка потребує уваги. [5]
5) Атаки на перегляд шляху до файлів (також відомі як обхід каталогів) представляють собою вразливості веб-безпеки, які дозволяють зловмисникам отримати доступ до файлів і каталогів, які знаходяться за межами кореневої папки веб-сайту. Ці файли можуть містити код і дані додатків, внутрішні системні дані та конфіденційні файли операційної системи. Зловмисники можуть здійснювати атаки обходу шляху до файлів, вимушуючи веб-сервер або веб-додаток, що працює на сервері, повертати файли, які знаходяться за межами кореневої папки веб-сайту. Це представляє серйозну загрозу, яка вимагає ретельного уважного врахування.[15]
6) Підробка запитів на стороні сервера (SSRF) відбувається, коли зловмисник використовує вразливість безпеки веб-додатка, щоб

змусити сервер здійснити HTTP-запит до обраного зловмисником домену. У таких атаках зловмисник може використовувати функції сервера для читання або оновлення внутрішніх ресурсів. Зловмисник може контролювати або змінювати URL-адреси, з яких код, що працює на сервері, читає або відправляє дані, а також отримувати конфігураційні дані сервера, такі як метадані AWS. Зловмисник може підключатися до внутрішніх служб, таких як бази даних, що підтримують HTTP, або до внутрішніх служб, які не призначені для публічного доступу.[4]

Успішна атака SSRF може призвести до несанкціонованого маніпулювання або доступу до даних в організації як у самому вразливому додатку, так і в інших внутрішніх системах, з якими він взаємодіє. У деяких випадках вразливості SSRF можуть дозволити зловмиснику виконувати довільні команди. Також використання SSRF для встановлення з'єднань з зовнішніми сторонніми системами може спричинити подальші зловмисні атаки, які можуть бути сприйняті як дії, що походять від організації, яка використовує вразливий додаток. [6]

7) Перехоплення кліків, також відоме як clickjacking, представляє собою тип атаки, який відбувається на стороні клієнта з метою обману користувача програми, щоб натиснути на елемент, відмінний від того, що він бачить на екрані. Хакери здійснюють цей тип атаки, приховуючи шкідливе програмне забезпечення або шкідливий код за легітимними елементами керування на веб-сайті. Зазвичай вони використовують вразливості в ланцюжку постачання додатків, зокрема в сторонньому JavaScript, який часто не можна контролювати за допомогою стандартних засобів захисту додатків.[4]

   Цей зловмисний метод використовується зловмисниками для запису кліків інфікованих користувачів в Інтернеті. Він може бути

використаний для перенаправлення трафіку на певний сайт або змусити користувачів "вподобати" додаток Facebook або "погодитися" з додатком Facebook. Більш зловмисні цілі можуть включати збір конфіденційної інформації, що зберігається в браузері, наприклад, паролі, або встановлення шкідливого контенту.[16]

1.4 Постановка задач що до захисту

Маючи на увазі, що основна мета встановлення більшості міжмережевий екранів полягає у блокуванні доступу, очевидно, що виявлення будь-яких слабкостей, які дозволяють проникнути в систему, може призвести до повного розколу захисту цієї системи. Якщо несанкціонованому користувачеві вдається проникнути в міжмережевий екран і змінити його конфігурацію, ситуація може стати ще більш небезпечною. Для розрізнення термінології, можна прийняти, що в першому випадку ми маємо справу з вразливістю міжмережевого екрана, а в другому - з повним руйнуванням. Визначити ступінь впливу, який може мати руйнування міжмережевого екрана на систему, є дуже складним завданням. Інформація про діяльність і спроби злому, зібрана самим міжмережевим екраном, може надати найбільш повне розуміння надійності такого захисту. Найгірше трапляється тоді, коли міжмережевий екран повністю руйнується і не залишає жодних слідів, що пояснюють, як це сталося. [21]

У найкращому випадку міжмережевий екран сам виявляє спроби злому і ввічливо повідомляє адміністратора про це. Проте, спроба такого зламу приречена на провал. Один з способів визначити результат спроби злому міжмережевого екрана - перевірити стан речей у так званих зонах ризику. Якщо мережа підключена до Інтернету без міжмережевий екрана, то всій мережі загрожує атака. Проте це само по собі не означає, що мережа стає вразливою до кожної спроби злому. Однак, якщо мережа приєднується до загальної мережі без захисту, адміністратору доведеться забезпечувати безпеку кожного окремого

вузла. У випадку проникнення в міжмережевий екран, зона ризику розширюється і охоплює всю захищену мережу. Зловмисник, який має доступ до входу в міжмережевий екран, може використовувати його як базу для захоплення всієї локальної мережі шляхом методу "захоплення островів". В такому випадку, хоча існує слабка надія, зловмисник може залишити сліди в міжмережевому екрані, що допоможе виявити його. Але якщо міжмережевий екран повністю вийде з ладу, локальна мережа стає вразливою для атаки з будь-якої зовнішньої системи, а визначення характеру цієї атаки стає практично неможливим.[7]

Загалом, міжмережевий екран можна розглядати як засіб зменшення зони ризику до одного центрального пункту. Ініціатива такого роду може здатися не дуже мудрою, оскільки це схоже на складання усіх яєць в одну корзину. Проте, на практиці виявлено, що в більших мережах зазвичай є кілька уразливих вузлів, які можуть бути легко скомпрометовані навіть не дуже досвідченим зловмисником, якщо він має достатньо часу. Багато великих компаній мають організаційну політику щодо забезпечення безпеки вузлів, яка розроблена з урахуванням цих потенційних недоліків. Проте повністю покладатися на правила само по собі не є розумним рішенням. Саме міжмережевий екран дозволяє підвищити надійність вузлів, направляючи зловмисника вузьким тунелем, що збільшує шанси виявити його і вжити заходів до того, як він зможе завдати шкоди. Аналогічно до того, як середньовічні замки мали кілька стін для захисту, застосування міжмережевий екрана створює взаємно блокуючий механізм захисту.[23]

Як WAF захищають від цих типів атак.

WAF використовують різноманітні функції та механізми для захисту додатків від різних типів атак. Це можуть бути динамічні політики безпеки з автоматичним виправленням помилкових спрацьовувань , захист від DDoS-атак на рівні додатків, виявлення та захист API, боротьба з ботами тощо.

Більшість WAF використовують негативну модель безпеки, визначаючи, що заборонено, і неявно дозволяючи все інше. Оскільки сигнатури атак можуть спричиняти хибні спрацьовування, ідентифікуючи легітимний трафік як трафік атаки, такі правила, як правило, спрощуються в спробі виявити очевидні атаки. Це призводить до захисту за найменшим спільним знаменником.[7]

Комплексний захист, коли захист на основі підписів не може компенсувати, вимагає позитивної моделі безпеки, яка визначає набір дозволених типів і значень; у випадку SQL-ін'єкції позитивна модель безпеки порівнює вхідні дані користувача з відомими шаблонами атак і використовує логіку для того, щоб для визначення різниці між легітимним введенням користувача і помилкою ін'єкції. Позитивні моделі безпеки також важливі для успішного зниження ризиків, пов'язаних з SSRF. Ось шість ключових особливостей, які слід враховувати при оцінці WAF, щоб зменшити вплив цих поширених атак і вразливостей:

- Повні можливості виявлення і захисту API, які забезпечують видимість, контроль і пом'якшення всіх форм зловживань і маніпуляцій з API як в локальних, так і в хмарних середовищах
- Вбудований захист від HTTP DDoS, який зупиняє DDoS-атаки на рівні додатків
- Інтегровані засоби управління ботами для виявлення та протидії просунутим ботам 3-го та 4-го поколінь, які імітують людську поведінку
- Механізми запобігання витоку даних, які автоматично маскують конфіденційні дані користувачів, включаючи особисту інформацію (PII)
- Комбінована негативна і позитивна модель безпеки, яка використовує передові методи поведінкового аналізу для виявлення шкідливих загроз
- Механізми вдосконалення політик, які можуть безперервно оптимізувати політики безпеки та адаптуватися до змін у застосунках, трафіку та ландшафті загроз.[4]

2 АНАЛІЗ АРХІТЕКТУРИ МІЖМЕРЕЖЕВИХ ЕКРАНІВ

На основі заздалегідь визначених правил безпеки міжмережевий екрани пропускають лише безпечний трафік і блокують несанкціоновані та шкідливі програми. Міжмережевий екрани встановлюються в мережі для оцінки всіх вхідних пакетів і пропускають тільки безпечні пакети.[10]

Фактори, що враховуються при створенні міжмережевий екрана:

- спроможність конкретної організації впровадити архітектуру міжмережевий екрана.
- прийнятний бюджет організації для впровадження міжмережевий екрана.
- цілі організації щодо мережевої безпеки.
- різні реалізації архітектури міжмережевий екрана[1]

2.1 Архітектура екранованого хоста

Це комбінація маршрутизаторів з фільтрацією пакетів і методів міжмережевого екрана, таких як проксі-сервери додатків. Ці методи включають використання проксі-сервера для обробки трафіку на рівні додатків, що допомагає захисту системи від атак спрямованих на ці додатки. Маршрутизатор, це пристрій або програмне забезпечення, яке фільтрує всі пакети в трафіку за допомогою правил та політик безпеки перед доступом до системних функцій, що

дозволяє заборонити небажаний трафік та захистити хост-систему відпотенційних атак. Послуги надаються хостами, підключеними до внутрішньої мережі а базова безпека забезпечується фільтрацією пакетів, що дозволяє перевіряти та блокувати небажаний трафік. Виконується проксі-обробка. Пакети, що надходять до системи, проходять попередню перевірку для мінімізації мережевого трафіку. Це відповідає за зменшення навантаження на проксі-сервер. Проксі-сервер додатків перевіряє HTTP та інші протоколи прикладного рівня. Цей хост називається бастіонним хостом, він має бути особливо захищеним та забезпечувати найвищий рівень безпеки. Екранований хост має вбудовані механізми моніторингу та аналізу, що дозволяють виявляти потенційні загрози та атаки. Система моніторингу перевіряє активність мереж, реєструє надзвичайні події та надає повідомлення про потенційні загрози. Екранований хост може мати вбудовані засоби аудиту та відстеження, що дозволяють реєструвати та аналізувати активність користувачів, доступ до ресурсів та події в системі. Це допомагає виявляти та реагувати на потенційні загрози та атаки. Екранований хост може бути інтегрований в загальну політику безпеки корпоративної мережі. Це дозволяє забезпечити єдиноцільність політик безпеки, рівень доступу та захисту для всіх систем та ресурсів у межах організації. [19]

Особливості архітектури екранованого хоста:

- Один маршрутизатор з фільтрацією пакетів.
- Один бастіонний хост. [6]

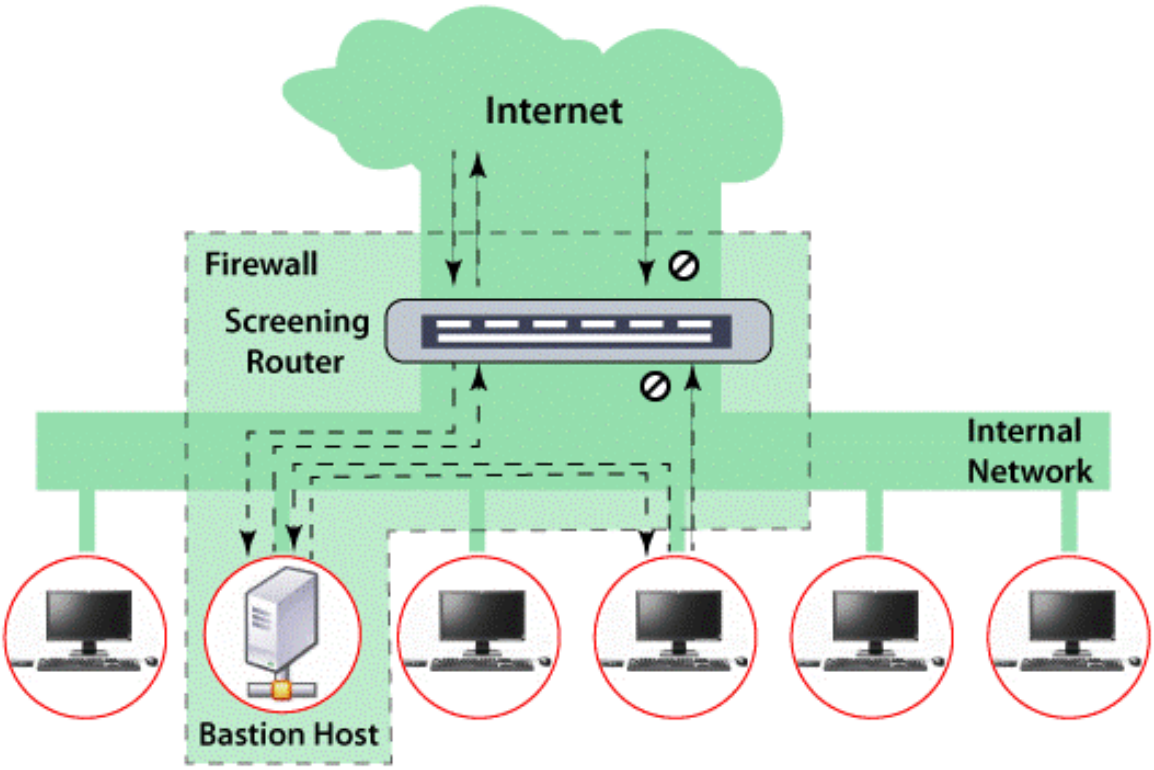

Рисунок 2.1 – Архітектура екранованого хоста

2.2 Архітектура з двома хостами

Подальшим вдосконаленням архітектури з екранованим хостом є архітектура з двома хостами. Основна ідея полягає в тому, що один хост виконує роль активного пристрою, яки обробляє трафік та надає послуги, а другий хост працює в режимі резерву, готовий прийняти обробку трафіку у разі відмови або проблем з активним хостом. Архітектура з двома хостами використовується для захисту системи від несанкціонованого доступу неавторизованих користувачів до конфіденційної інформації. Це архітектура, яка використовує дві мережеві інтерфейсні карти. Архітектура з двома хостами забезпечує безпечний доступ, за допомогою цієї архітектури система стає більш надійною, оскільки має резервний хост. Попередній захист забезпечується налаштуванням однієї мережевої карти в мережі, а іншої - у зовнішній мережі. Це дозволяє ефективно і безпечно маршрутизувати пакети з однієї мережі в іншу та знижує ризик втрати сервісу через непередбачуван ситуації. Сервери, які забезпечують доступ через міжмережевий екрани, шлюзи та проксі-сервери, мають архітектуру з двома хостами. Вона забезпечує дуже високий ступінь контролю, не пропускаючи жодних пакетів із зовнішніх мереж, якщо користувач задає таке правило. Також архітектура з двома хостами дозволяє розподілити навантаження між активним та резервним хостом, що покращує продуктивність та ефективність системи.[20]

Особливості двохостової архітектури

- Один маршрутизатор з функцією фільтрації пакетів.
- Більш сувора форма міжмережевий екрана з екранованим хостом.
- Архітектура Dual Homed Host підтримує Bastion Host з двома мережевими інтерфейсними картами.[6]

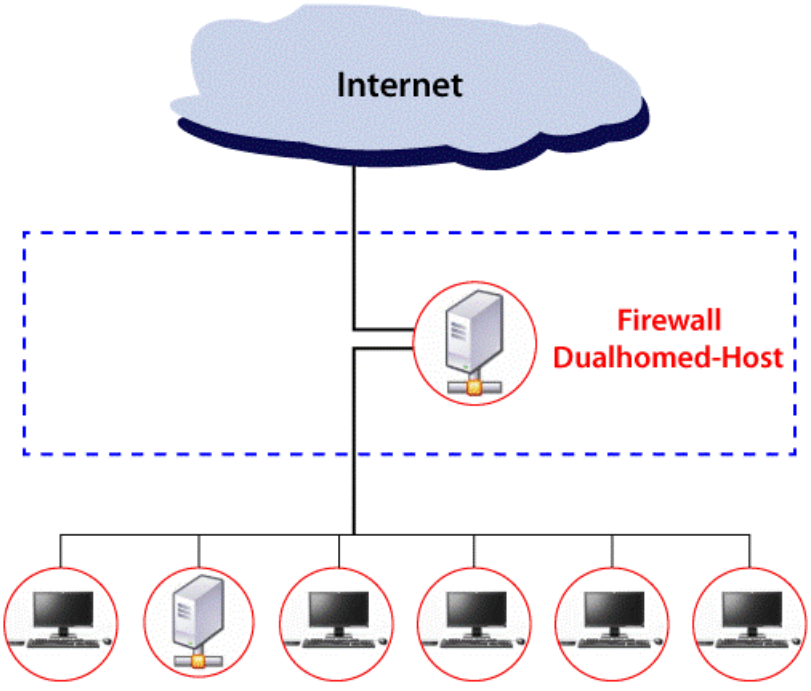

Рисунок 2.2 – Архітектура з двома хостами

2.3 Архітектура з екранованою підмережею

Міжмережевий екран з екранованою підмережею - це комбінація шлюзу з двома хостами і міжмережевого екрана з екранованим хостом, що використовується для забезпечення додаткового рівня безпеки та контролю в мережевому середовищі. Для підвищення гнучкості та пропускної здатності різні компоненти міжмережевого екрана підтримуються в окремих системах. Міжмережеві екрани підмережі менш складні, оскільки кожен компонент виконує окреме завдання. Він може відокремлювати внутрішню мережу від Інтернету; хости Bastion є більш вразливими. Вони піддаються атакам, незважаючи на посилені заходи безпеки. Якщо зловмисникові вдасться атакувати хост Bastion, безпека системи може бути під загрозою. Міжмережевий екрани з екрануванням підмережі використовують два екрановані маршрутизатори для забезпечення безпеки. Вони в основному використовуються в електронній комерції та фінансових системах. Основне призначення - створення демілітаризованої зони (DMZ). В підмережі DMZ розміщуються сервери або системи, до яких потрібен публічний доступ з Інтернету, такі як веб-сервери, поштові сервери або сервери віддаленого доступу. Це дозволяє забезпечити фізичну і логічну ізоляцію цих серверів від внутрішньої мережі, що зменшує ризик вразливості системи. Використання підмережі DMZ дозволяє забезпечити додатковий рівень захисту для внутрішньої мережі організації. У разі, якщо сервери в підмережі DMZ стають жертвою атаки, віддалені зловмисники не

матимуть прямого доступу до внутрішньої мережі, що допомагає обмежити можливі наслідки атаки. Міжмережеві екрани з екрануванням підмережі переважно використовуються в корпоративних мережах. Багато високошвидкісних систем з великим трафіком використовують міжмережевий екран з екранованими підмережами. Архітектура з екранованою підмережею дозволяє встановити додаткові механізми контролю доступу для серверів у підмережі DMZ. Це може включати використання проксі-серверів, фаєрволів, систем автентифікації та авторизації, які дозволяють точно контролювати доступ до цих серверів.[19] Дана архітектура надає більшу гнучкість у налаштуванні та управлінні правилами мережевої безпеки. Різні рівні довіри до підмережі DMZ та внутрішньої мережі дозволяють встановити специфічні правила для кожного сегменту мережі, що відповідає потребам безпеки організації. Ця архітектура складається з двох основних компонентів: хостів-бастіонів і маршрутизаторів з фільтрацією пакетів. Кожен хост відповідає за захист своєї внутрішньої мережі. Існують різні моделі міжмережевий екранів з екранованими підмережами. Одна модель міжмережевий екрана з екранованою підмережею складається з бастіонного хоста з двома хостами між двома фільтруючими маршрутизаторами.Ця архітектура використовує три ключові компоненти:

- Перший компонент підключається до Інтернету і діє як загальнодоступний інтерфейс.

- Другий компонент виступає посередником між першим і третім компонентами. Це середня зона (DMZ).
- Третій компонент працює в основному в поєднанні з інтрамережею.[20]

Особливості міжмережевих екранів з екранованими підмережами

- Міжмережевий екран екранованої підмережі підтримує маршрутизатори з двома рівнями фільтрації пакетів
- Одиночний бастіонний хост.[6]

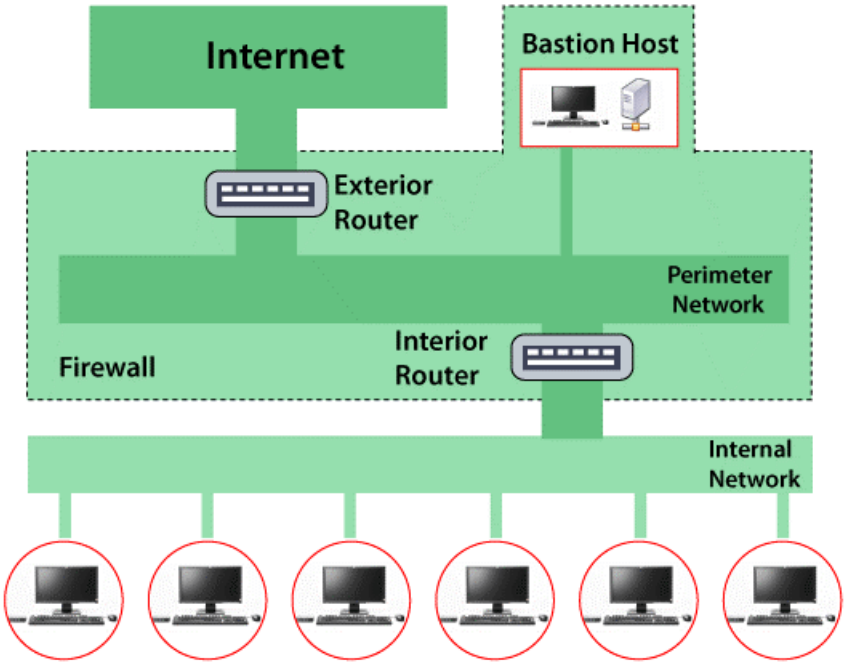

Рисунок 2.3 – Архітектура з екранованою підмережею

2.4 Екрановані маршрутизатори

Працює так само, як і метод міжмережевого екрана з фільтрацією пакетів. Він відповідає за перевірку всього вхідного та вихідного трафіку і маршрутизацію пакетів відповідно до заздалегідь визначених правил. Архітектура міжмережевого екрану з "екранованими маршрутизаторами" є додатковим рівнем безпеки, який забезпечує контроль та фільтрацію мережевого трафіку на рівні маршрутизаторів. У цій архітектурі маршрутизатори виконують функцію міжмережевих екранів. Вони фільтрують та аналізують пакети, що проходять через них, на основі заданих правил безпеки.[19] Це дозволяє здійснювати контроль трафіку на рівні маршрутизації та захищати мережу від шкідливих атак та несанкціонованого доступу. Екрановані маршрутизатори здатні фільтрувати пакети на основі різних параметрів, таких як IP-адреса джерела та призначення, порти, протоколи тощо. Вони можуть встановлювати правила доступу, які контролюють, які пакети можуть пройти через маршрутизатор, а які мають бути блоковані. Це дозволяє захистити мережу від небажаного та шкідливого трафіку. Економічно ефективний. Може використовуватися як захист периметра для внутрішніх мереж. Переважно ефективний для внутрішніх міжмережевих екранів. Може використовуватися в мережах, що надають інтернет-послуги. Постачальники послуг в основному використовують екрануючі маршрутизатори між Інтернетом і хостами послуг.

Якщо маршрутизатор піддається атаці, безпека системи ставиться під загрозу. [18]

Архітектура міжмережевого екрана з екранованим маршрутизатором підходить для ситуацій, коли:

- Кількість протоколів, що використовуються додатком, обмежена.
- Використовувані протоколи прості.
- Мережі з дуже високим рівнем безпеки хоста.
- Потрібна максимальна безпека.
- Потрібна надмірність.[6]

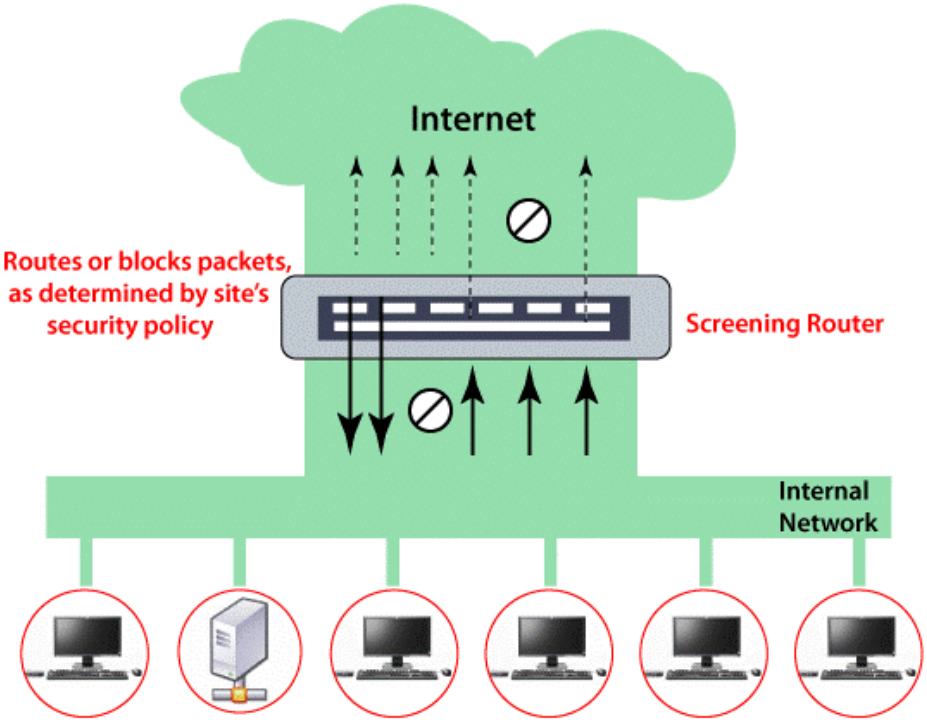

Рисунок 2.4 – Архітектура з екранованою підмережею

3 ВИДИ МІЖМЕРЕЖЕВИХ ЕКРАНІВ

3.1 Класифікація міжмережевих екранів

Міжмережевий екран (англ. "inter-network screen") - це технологія, яка забезпечує захист і безпеку передачі даних між різними мережами, включаючи Інтернет, промислові мережі, корпоративні мережі тощо. Цей екран працює як посередник між зовнішніми мережами та внутрішніми мережами організації, контролюючи трафік, що проходить через нього, та застосовуючи правила фільтрації та перевірки безпеки. Фільтрація здійснюється на підставі набору умов, попередньо завантажених в міжмережевий екран і відображають концепцію інформаційної безпеки корпорації. Міжмережевий екрани можуть бути виконані у вигляді як апаратного, так і програмного комплексу, записаного в комутуючий пристрій або сервер доступу (сервер-шлюз, просто сервер, хосткомп'ютер і т.д.), вбудованого в операційну систему [11].

3.2 Порівняння функціоналу міжмережевого екрану

Робота міжмережевого екрану полягає у забезпеченні безпеки мережі шляхом контролю трафіку, що протікає через нього. Ефективність роботи міжмережевого екрана, працюючого під керівництвом Windows, зумовлена тим, що він цілком заміщає реалізований стік протоколів TCP/IP, і тому порушувати його з допомогою спотворення протоколів зовнішньої мережі (що часто робиться хакерами) неможливо.[22]

Міжмережеві екрани зазвичай виконують такі функції:

- Фізичне відділення робочих станцій та серверів внутрішнього сегмента мережі (внутрішньої підмережі) від зовнішніх каналів зв'язку;

- Міжмережевий екран встановлює правила доступу, які визначають, які типи мережевого трафіку дозволено або заборонено. Він може базуватися на IP-адресах, портах, протоколах або інших параметрах пакетів. Це дозволяє обмежувати доступ до різних мережевих ресурсів залежно від встановлених правил;
- Реєстрацію всіх запитів до компонентами внутрішньої підмережі ззовні;
- Міжмережевий екран може підтримувати віртуальні приватні мережі (VPN) та забезпечувати безпечний доступ до мережевих ресурсів зовнішніх користувачів.
- Контроль цілісності програмного забезпечення та об'єктивності даних;
- Економію адресного простору мережі (у внутрішній підмережі можна використовувати локальна система адресації серверів);
- Міжмережевий екрани можуть працювати різних рівнях протоколів моделі OSI.

- Міжмережевий екран включає механізми захисту від різних типів атак, таких як вторгнення, сканування портів, атаки на відмову в обслуговуванні (DoS) та розподілені атаки на відмову в обслуговуванні (DDoS). Він може виявляти та блокувати небезпечний трафік, що має підозрілі характеристики або відповідає визначеним сигнатурам атак.

Ці функції міжмережевого екрану допомагають забезпечити безпеку мережі, контролювати трафік та захистити ресурси від шкідливого або несанкціонованого доступу. Вони дозволяють організаціям знизити ризики, пов'язані зі зловживанням мережі та зберегти цілісність та конфіденційність даних. [23]

На мережному рівні виконується фільтрація вступників пакетів, джерело якої в IP адреси (наприклад, зупиняти пакунки з Інтернету, створені задля ті сервери,

доступом до яким зовні заборонено; зупиняти пакети з фальшивими зворотними адресами чи IP адресами, занесеними в «чорного списку», тощо.). На транспортному рівні фільтрація припустима ще й з номерам портів TCP і прапорів, які у пакетах (наприклад, запитів встановлення сполуки). На прикладному рівні може виконуватися аналіз прикладних протоколів (FTP,HTTP,SMTP тощо.) контроль над змістом потоків даних (заборона внутрішнім абонентам отримання будь-яких типів файлів: рекламної інформації, або виконуваних програмних модулів, наприклад). Можна в міжмережевий екрані створювати й експертну систему, яка, аналізуючи трафік, діагностує події, які можуть загрожувати безпеки внутрішньої мережі, і сповіщає звідси адміністратора. Експертна система здатна й у випадку небезпеки (спам, наприклад) автоматично посилювати умови фільтрації тощо..[14]

3.3 Порівняння видів міжмережевих екранів

Основна мета міжмережевого екрана - блокувати шкідливі мережеві запити та пакети даних, дозволяючи при цьому законний трафік. Тим не менш, слово "міжмережевий екран" є занадто широким, щоб використовувати покупців IT-безпеки. Існує багато різних типів міжмережевих екранів, кожен з яких функціонує по-різному, як усередині хмари, так і поза нею, щоб захистити різні типи важливих даних.

Є такі типи міжмережевих екранів:

Міжмережевий екран з фільтрацією пакетів (Packet filtering firewall) є типом пристрою для забезпечення мережевої безпеки, який аналізує окремі пакети даних, що проходять через мережевий інтерфейс, і вирішує, чи дозволити чи заблокувати їх на основі заданих правил. Він працює на мережевому рівні моделі OSI (зазвичай на рівні мережевого екрана або маршрутизатора) та надає базові можливості фільтрації шляхом аналізу

заголовків кожного пакета. Мережевий екран аналізує різні поля в заголовку пакета, такі як IP-адреси джерела та призначення, порти джерела та призначення, протоколи та прапорці (флаги), що вказують на тип пакета. За допомогою цих правил мережевий екран вирішує, які пакети допустити, а які заблокувати. Наприклад, він може бути налаштований для блокування пакетів з певних небезпечних IP-адрес або для дозволу лише на певні типи трафіку.[16]

Переваги міжмережевого екрана з фільтрацією пакетів

- Один пристрій може фільтрувати трафік для всієї мережі
- Надзвичайно швидкий і ефективний у скануванні трафіку
- Мінімальний вплив на інші ресурси, продуктивність мережі та досвід кінцевого користувача

Недоліки міжмережевого екрана з фільтрацією пакетів

- Оскільки фільтрація трафіку повністю базується на інформації про IP-адресу чи порт, фільтрації пакетів бракує ширшого контексту, який інформує інші типи міжмережевий екранів
- Не перевіряє корисне навантаження, і його можна легко підробити
- Не ідеальний варіант для кожної мережі
- Списки контролю доступу може бути важко налаштувати та керувати ними

Фільтрування пакетів може не забезпечити рівень безпеки, необхідний для кожного випадку використання, але є ситуації, коли цей недорогий міжмережевий екран є надійним варіантом. Для невеликих організацій або організацій з обмеженим бюджетом фільтрація пакетів забезпечує базовий рівень безпеки, який може забезпечити захист від відомих загроз. Великі підприємства також можуть використовувати фільтрацію пакетів як частину багаторівневого

захисту для відсіювання потенційно шкідливого трафіку між внутрішніми підрозділами.[18]

Шлюз цілісного рівня (Circuit-level gateway) - це тип мережевого пристрою, який працює на сеансовому рівні моделі OSI. Він забезпечує захист мережі, контролюючи та фільтруючи з'єднання між внутрішньою та зовнішньою мережами. Цілісного рівня шлюз встановлює та керує двонаправленими з'єднаннями між внутрішніми та зовнішніми пристроями. Він діє як посередник, аналізуючи пакети даних, що проходять через нього, та приймаючи рішення щодо дозволу або блокування з'єднання залежно від заданих правил безпеки. Одна з особливостей цілісного рівня шлюз полягає в тому, що він не аналізує вміст пакетів даних, але зосереджується на створенні тунелю для з'єднання між двома пристроями. Після встановлення з'єднання, весь трафік, що проходить через цей з'єднання, дозволяється передавати без подальшого перевірки заголовків пакетів. Цілісні рівня шлюзи забезпечують високий рівень безпеки шляхом приховування внутрішніх IP-адрес та даних від зовнішніх мереж. Вони також можуть використовуватися для контролю доступу, аутентифікації користувачів та моніторингу активності в мережі.[20]

Переваги шлюзу на рівні ланцюга

- Обробляє лише запитані транзакції; весь інший трафік відхилено
- Легко налаштувати та керувати
- Низька вартість і мінімальний вплив на досвід кінцевого користувача

Недоліки шлюзу на рівні схеми

- Якщо вони не використовуються в поєднанні з іншими технологіями безпеки, шлюзи на рівні каналу не забезпечують захисту від витоку даних із пристроїв у міжмережевий екрані
- Немає моніторингу прикладного рівня
- Потрібні постійні оновлення, щоб підтримувати правила актуальними

Хоча шлюзи на рівні каналів забезпечують вищий рівень безпеки, ніж міжмережевий екрани з фільтрацією пакетів, їх слід використовувати в поєднанні з іншими системами. Наприклад, шлюзи на рівні схеми зазвичай використовуються поряд із шлюзами на рівні програми. Ця стратегія поєднує атрибути міжмережевий екранів шлюзу на рівні пакетів і каналів із фільтрацією вмісту.[12]

Шлюз прикладного рівня (Application-level gateway) - це тип мережевого пристрою, який працює на рівні додатків моделі OSI. Він забезпечує розширені функції безпеки та контролю доступу до мережі, аналізуючи трафік на основі конкретних додатків або протоколів. Шлюз на рівні додатків виконує більш глибокий аналіз пакетів даних порівняно з іншими типами мережевих пристроїв, такими як фільтруючий мережевий екран або цілісного рівня шлюз. Він розуміє та інтерпретує додаткові заголовки, команди та дані, що передаються в рамках конкретного протоколу додатку. При аналізі трафіку шлюз на рівні додатків може застосовувати різні політики безпеки, контролю доступу та фільтрації, враховуючи контекст додатку. Він може блокувати або дозволяти певні типи запитів або команд, виконувати перевірку автентифікації та авторизації користувачів, а також реалізовувати механізми шифрування для забезпечення конфіденційності даних. Шлюз на рівні додатків може працювати як проксі-сервер для конкретних додатків, що передає трафік між клієнтом та сервером. Це

дозволяє виконувати додаткові функції, такі як кешування, фільтрація вмісту, збір статистики тощо.[21]

Такий пристрій — технічно проксі-сервер, який іноді називають міжмережевим екраном проксі сервера — функціонує як єдина точка входу в мережу та точка виходу з неї. Шлюзи прикладного рівня фільтрують пакети не лише відповідно до служби, для якої вони призначені, як зазначено портом призначення, а й іншими характеристиками, такими як рядок запиту HTTP.

Хоча шлюзи, які фільтрують на прикладному рівні, забезпечують значну безпеку даних, вони можуть значно вплинути на продуктивність мережі та бути складними в управлінні.

Переваги шлюзу прикладного рівня

- Перевіряє всі зв’язки між зовнішніми джерелами та пристроями за міжмережевий екраном, перевіряючи не лише інформацію про адресу, порт і TCP-заголовок, а й сам вміст, перш ніж пропускати трафік через проксі-сервер
- Забезпечує точні елементи керування безпекою, які можуть, наприклад, дозволити доступ до веб-сайту, але обмежити, які сторінки на цьому сайті користувач може відкривати
- Захищає анонімність користувачів

Недоліки шлюзу прикладного рівня

- Може перешкоджати продуктивності мережі
- Дорожче, ніж деякі інші варіанти міжмережевий екрана
- Для отримання максимальної вигоди від шлюзу потрібні великі зусилля
- Працює не з усіма мережевими протоколами

Міжмережевий екрани прикладного рівня найкраще використовувати для захисту корпоративних ресурсів від загроз веб додатків. Вони можуть як блокувати доступ до шкідливих сайтів, так і запобігати витоку конфіденційної інформації з міжмережевий екрана. Однак вони можуть викликати затримку зв'язку.[12]

Міжмережевий екран контролю стану (Stateful inspection firewall) - це тип мережевого пристрою, який здатен аналізувати трафік на рівні пакетів і враховувати стан з'єднання для прийняття рішень щодо перенаправлення, дозволу або блокування пакетів. Однією з ключових особливостей міжмережевий екрана зі становою перевіркою є збереження стану з'єднання між внутрішніми та зовнішніми мережами. При першому встановленні з'єднання міжмережевий екран створює запис про стан з'єднання, який включає інформацію про джерело, призначення, порти та стан (відкрите, закрите, встановлення, завершення тощо). Під час проходження пакетів через міжмережевий екран він порівнює їх з записами стану з'єднання, що дозволяє зробити вирішальне рішення щодо дозволу або блокування на основі попередніх взаємодій.[23]

Завдяки становій перевірці, міжмережевий екран може розпізнавати та контролювати різні типи трафіку, включаючи протоколи на рівні додатків (наприклад, HTTP, FTP, DNS тощо). Він може перевіряти правильність протокольних пакетів, перевіряти трафік на відповідність правилам безпеки, виконувати аутентифікацію та авторизацію користувачів, а також виконувати інші функції контролю доступу.

Міжмережевий екран контролю стану є більш ефективним та безпечним засобом фільтрації трафіку порівняно з простим фільтруючим мережевим екраном. Він може забезпечувати захист від різних видів атак, таких як атаки змі

Пристрої, що перевіряють стан, не тільки перевіряють кожен пакет, але й відстежують, чи є цей пакет частиною встановленого TCP або іншого мережевого

сеансу. Це забезпечує більшу безпеку, ніж фільтрація пакетів або моніторинг каналів, але завдає більшої шкоди продуктивності мережі. Іншим варіантом перевірки стану є багаторівневий інспекційний міжмережевий екран, який розглядає потік поточних транзакцій на кількох рівнях протоколу семирівневої моделі взаємозв'язку відкритих систем (OSI) .[22]

Переваги міжмережевого екрана з перевіркою стану

- Відстежує стан з'єднання протягом усього сеансу, а також перевіряє IP-адреси та корисне навантаження для більш ретельної безпеки
- Пропонує високий ступінь контролю над тим, який вміст пропускається в мережу або виходить з неї
- Не потрібно відкривати численні порти, щоб дозволити вхідний або вихідний трафік
- Надає значні можливості журналювання

Недоліки міжмережевий екрана перевірки стану

- Ресурсомісткий і заважає швидкості мережевих комунікацій
- Дорожче, ніж інші варіанти міжмережевий екрана
- Не надає можливості автентифікації для підтвердження джерел трафіку, чи вони не підроблені

Більшість організацій отримують вигоду від використання міжмережевий екрана перевірки стану. Ці пристрої служать більш повним шлюзом між комп'ютерами та іншими активами в межах міжмережевий екрана та ресурсами за межами підприємства. Вони також можуть бути дуже ефективними для захисту мережевих пристроїв від конкретних атак, таких як DoS.[18]

Міжмережевий екран нового покоління (Next-generation firewall) - це розширена версія мережевого пристрою, який комбінує функціональність традиційного фільтруючого мережевого екрана з рядом додаткових функцій та можливостей для забезпечення безпеки мережі. Основна відмінність міжмережевий екрана нового покоління від традиційного полягає в його здатності проводити більш глибокий аналіз пакетів даних та застосовувати комплексні правила фільтрації та безпеки на різних рівнях мережевого стеку.

Типовий NGFW поєднує інспекцію пакетів із перевіркою стану, а також включає деякі різноманітні глибокі інспекції пакетів (DPI), а також інші системи безпеки мережі, такі як IDS/IPS, фільтрація шкідливих програм і антивірус. У той час як інспекція пакетів у традиційних міжмережевий екранах розглядає виключно заголовок протоколу пакета, DPI розглядає фактичні дані, які несе пакет. Міжмережевий екран DPI відстежує хід сеансу перегляду веб-сторінок і може помітити, чи корисне навантаження пакета, будучи зібрано з іншими пакетами у відповіді HTTP-сервера, становить законну відповідь у форматі HTML.

Переваги NGFW

- Поєднує DPI з фільтрацією зловмисного програмного забезпечення та іншими засобами керування для забезпечення оптимального рівня фільтрації
- Відстежує весь трафік від Рівня 2 до рівня додатків для отримання більш точної інформації, ніж інші методи
- Може бути автоматично оновлено для надання поточного контексту

Недоліки NGFW

- Щоб отримати найбільшу вигоду, організаціям необхідно інтегрувати NGFW з іншими системами безпеки, що може бути складним процесом

- Дорожче, ніж інші типи міжмережевий екранів

NGFW є важливим запобіжним засобом для організацій у жорстко регульованих галузях, таких як охорона здоров’я чи фінанси. Ці міжмережевий екрани забезпечують багатофункціональні можливості, які приваблюють тих, хто добре розуміє, наскільки шкідливим є середовище загроз. NGFW найкраще працюють при інтеграції з іншими системами безпеки, що в багатьох випадках вимагає високого рівня досвіду. [5]

4 ПОБУДОВА МІЖМЕРЕЖЕВОГО ЕКРАНУ

4.1 Вибір архітектури

Мій вибір архітектури "екранування маршрутизатору" для дипломної роботи має свої переваги, оскільки ця архітектура надає деякі важливі привілегії, які можуть бути цікавими для дослідження. Ось кілька переваг архітектури "екранування маршрутизатору" якими я керувалась:

- Єдина точка входу: У цій архітектурі маршрутизатор виступає як центральна точка контролю трафіку. Це дозволяє централізовано керувати та аналізувати весь трафік, що проходить через мережу.
- Простота керування та конфігурації: Архітектура "екранування маршрутизатору" може бути відносно простою у керуванні та конфігурації.
- Вартість: Застосування архітектури "екранування маршрутизатору" може бути вигідним з економічної точки зору, оскільки дозволяє використовувати наявне мережеве обладнання (маршрутизатори) для забезпечення безпеки.
- Інтеграція з існуючими системами: Архітектура "екранування маршрутизатору" легко інтегрується з існуючими системами мережі.

4.2 Вибір програмного забезпечення

Мною було обрано операційні системи Windows і Ubuntu для побудови міжмережевого екрану. Мій вибір обгрунтований такими пунктами:

- Оскільки в роботі використаний інструмент iptables, пов’язаний в основному із системами Linux, існує також можливість використання даного інструменту в системі Windows. Ubuntu ж в свою чергу, як популярний дистрибутив Linux має вбудовану підтримку iptables.

- Windows має дружню інтерфейсну систему, що значно полегшує роботу з даною операційною системою.
- Ubuntu має широкий вибір документації, посібників та онлайн-ресурсів, які допоможуть у розумінні та налаштуванні iptables.
- Ubuntu відома своєю стабільністю та безпекою. Використання Ubuntu в якості операційної системи для міжмережевого екрану дозволило мені отримати переваги безпеки, які надається Linux-платформою, включаючи широкий спектр інструментів та підходів до захисту мережі.
- Великий попередній досвід роботи з цими операційними системами та простота реалізації.
- Доступність програмного забеспеченння.

4.3 Встановлення і налаштування ПЗ міжмережевого екрану

Для подальшого налаштування власного міжмережевого екрану, будуть використовуватися операційні системи Ubuntu nf Wsndows. [17]

Спочатку треба встановити утиліту iptables за допомогою команди

«sudo apt-get install iptables»

Рисунок 4.1 – Встановлюємо утиліту

Далі за допомогою команди ipconfig та ifconfig відображаємо всі поточні мережеві з'єднання. Переконуємось що наші ноутбуки знаходяться в одній мережі, за допомогою IPv4 адрес комп’ютерів.

```
C:\Windows\System32>ipconfig

Windows IP Configuration


Wireless LAN adapter Подключение по локальной сети* 1:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Подключение по локальной сети* 3:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Беспроводная сеть:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::21c0:1c1f:de6e:8496%18
   IPv4 Address. . . . . . . . . . . : 192.168.0.103
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.0.1
```

Рисунок 4.2 – Перевіряємо адресу на ОС Windows

```
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ ifconfig
enp2s0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 8c:16:45:9b:51:2f  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 531  bytes 52367 (52.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 531  bytes 52367 (52.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.104  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::ab5a:47c0:9691:4e1c  prefixlen 64  scopeid 0x20<link>
        ether 50:5b:c2:dd:9f:29  txqueuelen 1000  (Ethernet)
        RX packets 2959  bytes 2825833 (2.8 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2191  bytes 440955 (440.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Рисунок 4.3 – Перевіряємо адресу на ОС Ubuntu

На даних скріншотах видно що ноутбуки знаходяться в одній мережі.

Команди iptables повинні запускатися з привілегіями root, тому переглядаємо перелік поточних правил, які налаштовані для iptables за допомогою команди:

«sudo iptables –L»

```
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ 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 4.4 – Перевіряємо перелік поточних правил

Вимикаємо стандартний міжмережевий екран на ОС Windows, та за допомогою команди типу ping перевіряємо взаємодію пакетів між ноутбуками. Для її виконання вписуємо адресу іншого пристрою з тієї самої мережі, це хост на якому буде виконуватися пінг в мережі.

```
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ ping 192.168.0.103
PING 192.168.0.103 (192.168.0.103) 56(84) bytes of data.
64 bytes from 192.168.0.103: icmp_seq=253 ttl=128 time=7.42 ms
64 bytes from 192.168.0.103: icmp_seq=254 ttl=128 time=13.7 ms
64 bytes from 192.168.0.103: icmp_seq=255 ttl=128 time=7.83 ms
64 bytes from 192.168.0.103: icmp_seq=256 ttl=128 time=22.5 ms
64 bytes from 192.168.0.103: icmp_seq=257 ttl=128 time=90.2 ms
64 bytes from 192.168.0.103: icmp_seq=258 ttl=128 time=7.45 ms
64 bytes from 192.168.0.103: icmp_seq=259 ttl=128 time=7.24 ms
64 bytes from 192.168.0.103: icmp_seq=260 ttl=128 time=6.97 ms
64 bytes from 192.168.0.103: icmp_seq=261 ttl=128 time=19.3 ms
64 bytes from 192.168.0.103: icmp_seq=355 ttl=128 time=5.17 ms
64 bytes from 192.168.0.103: icmp_seq=356 ttl=128 time=9.60 ms
64 bytes from 192.168.0.103: icmp_seq=357 ttl=128 time=11.9 ms
^C
--- 192.168.0.103 ping statistics ---
372 packets transmitted, 105 received, 71,7742% packet loss, time 377556ms
rtt min/avg/max/mdev = 3.348/13.531/296.760/29.442 ms
```

Рисунок 4.5 – Спроба відправки пакетів до зміни правил

Бачимо що йде відправка спеціальних повідомлень по указаній нами адресі. Це означає що по нашій адресі пристрій функціонує. Тож можна приступати до реалізації правил.

Першим правилом перевіримо правило падіння. Спочатку вимкнемо стандартну політику налаштування взаємодії з пакетами “INPUT”, “OUTPUT”, “FORWARD”, а саме пропуск всіх пакетів. Потрібно замінити параметр ACCEPT на параметр DROP в ланцюгу INPUT, OUTPUT, FORWARD. Робимо це за допомогою команд:

«sudo iptables -P INPUT DROP»

«sudo iptables -P OUTPUT DROP»

«sudo iptables -P FORWARD DROP»

```
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -P INPUT DROP
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -P OUTPUT DROP
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -P FORWARD DROP
```

Рисунок 4.6 – Виконання правила падіння

```
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ ping 192.168.0.103
PING 192.168.0.103 (192.168.0.103) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- 192.168.0.103 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3063ms
```

Рисунок 4.7 – Результати відправки пакетів після внесення змін в правила

Після виконання перевіряємо взаємодію з пакетами Ubuntu. Результатом бачимо – 100% втрачених пакетів, отже спроба відправити пакети після зміни правил невдала. Перш за все, ми приходимо до висновку, що ОС Ubuntu наразі не передає або не отримує пакети даних і розпочинає створення нових правил для взаємодії з цими пакетами. Основний пріоритет полягає у тому, щоб дозволити ОС Ubuntu взаємодіяти зі своїм власним локальним інтерфейсом, тобто приймати

пакети з цього інтерфейсу. З цією метою ми перевіримо взаємодію з власним локальним інтерфейсом за допомогою наступної команди:

«ping 127.0.01»

```
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- 127.0.0.1 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2036ms
```

Рисунок 4.8 – Перевірка взаємодії з локальним хостом

За результатами на скріншоті бачимо, що пакети дійсно не відправляються, тож виправляємо цю ситуацію за допомогою команд:

«sudo iptables –A INPUT –I lo ACCEPT»

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

Та ще раз перевіряємо взаємодію з локальним хостом.

```
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -A INPUT -i lo -j ACCEPT
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -A OUTPUT -o lo -j ACCEPT
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ 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=0.069 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.093 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.091 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.093 ms
64 bytes from 127.0.0.1: icmp_seq=5 ttl=64 time=0.091 ms
64 bytes from 127.0.0.1: icmp_seq=6 ttl=64 time=0.089 ms
64 bytes from 127.0.0.1: icmp_seq=7 ttl=64 time=0.093 ms
64 bytes from 127.0.0.1: icmp_seq=8 ttl=64 time=0.087 ms
64 bytes from 127.0.0.1: icmp_seq=9 ttl=64 time=0.088 ms
64 bytes from 127.0.0.1: icmp_seq=10 ttl=64 time=0.081 ms
64 bytes from 127.0.0.1: icmp_seq=11 ttl=64 time=0.074 ms
64 bytes from 127.0.0.1: icmp_seq=12 ttl=64 time=0.096 ms
64 bytes from 127.0.0.1: icmp_seq=13 ttl=64 time=0.088 ms
64 bytes from 127.0.0.1: icmp_seq=14 ttl=64 time=0.110 ms
64 bytes from 127.0.0.1: icmp_seq=15 ttl=64 time=0.072 ms
64 bytes from 127.0.0.1: icmp_seq=16 ttl=64 time=0.093 ms
64 bytes from 127.0.0.1: icmp_seq=17 ttl=64 time=0.088 ms
64 bytes from 127.0.0.1: icmp_seq=18 ttl=64 time=0.082 ms
^C
--- 127.0.0.1 ping statistics ---
18 packets transmitted, 18 received, 0% packet loss, time 17416ms
rtt min/avg/max/mdev = 0.069/0.087/0.110/0.009 ms
```

Рисунок 4.9 – Перевірка взаємодії з локальним хостом після внесссенння змін

Згідно зі скріншотом, можна побачити, що пакети відправляються. Тепер ми додамо правило, щоб можна було відправляти пакети з даними на ОС Windows. При виконанні команди "ping" буде надіслано echo-запит на IP-адресу 192.168.0.103 (Windows). Однак, політика INPUT наразі не дозволяє приймати пакети з даними, а політика OUTPUT не відправляє їх. Тому, для того, щоб відправляти пакети і отримувати відповіді на ці запити на ОС Ubuntu, ми повинні дозволити ОС Ubuntu приймати пакети зі станом RELATED або ESTABLISHED. Крім того, ми також повинні додати дозвіл для взаємодії з IP-адресою ОС Windows. Для досягнення цього ми використаємо наступні команди:

«sudo iptables -A INPUT -m contrack -ctstate RELATED, ESTABLISHED -j ACCEPT»

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

І відразу перевіримо спроможність робити запити на інший IP.

«sudo iptables -A INPUT -s 192.168.0.103 -i ACCEPT»

«sudo iptables -A OUTPUT -s 192.168.0.103 -j ACCEPT»

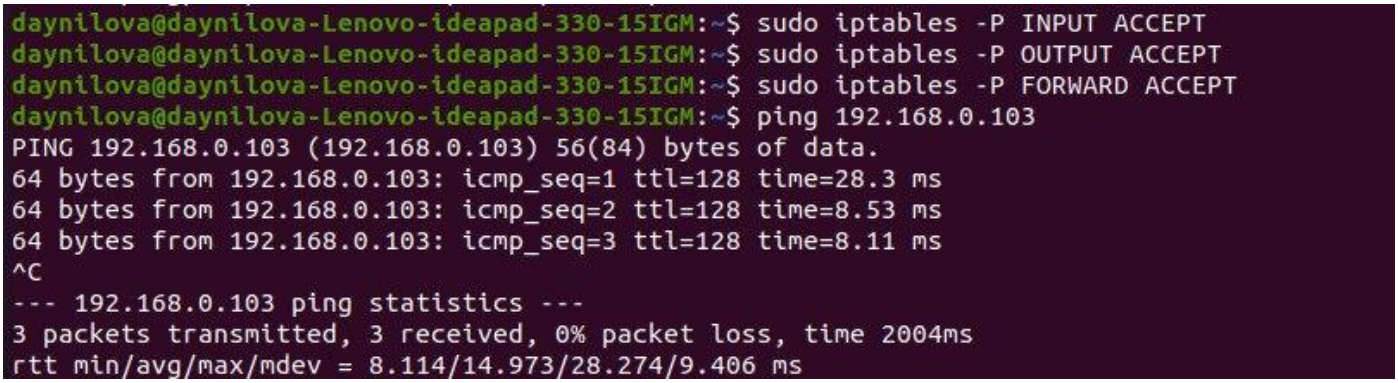

Рисунок 4.10 – Спроба відправки пакетів

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 contrack --ctstate NEW,ESTABLISHED -¡ ACCEPT»

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

Також створю блокування визначеної IP-адреси, а саме заблоковано IP-адресу 192.168.0.103. Наступними командами:

«sudo iptables -A INPUT -s 192.168.0.103 -j DROP»

«sudo iptables -A OUTPUT -d 192.168.0.103 -j DROP»

```
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -A INPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -A OUTPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -A INPUT -s 192.168.0.103 -j DROP
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -A OUTPUT -d 192.168.0.103 -j DROP
```

Рисунок 4.11 – Запит на заблоковану адресу

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

«sudo mkdir /etc/iptables»

«sudo iptables-save > /etc/iptables/rules.v4»

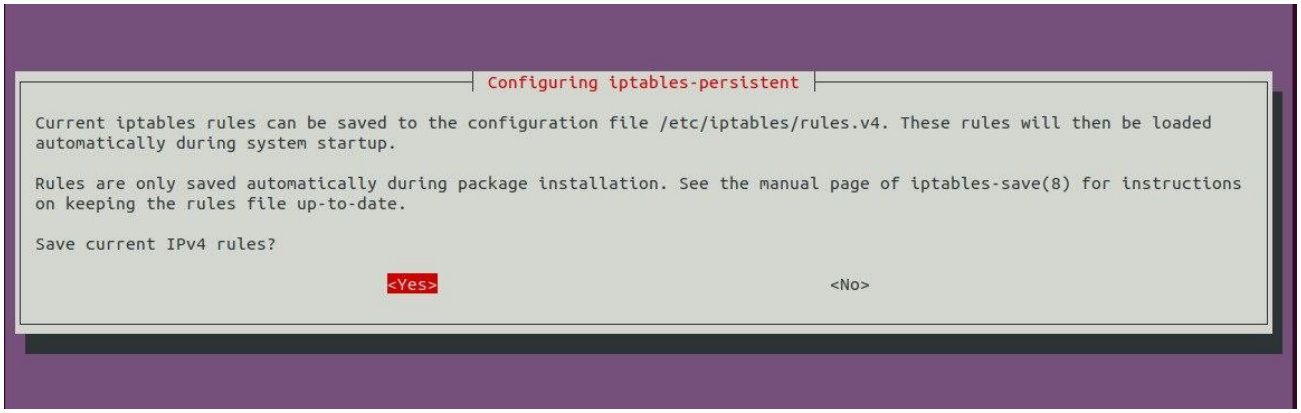

Рисунок 4.12 – Зберігаємо файл

```
1 # Generated by iptables-save v1.8.4 on Sun Jun  4 20:20:51 2023
2 *filter
3 :INPUT DROP [0:0]
4 :FORWARD DROP [0:0]
5 :OUTPUT DROP [0:0]
6 -A INPUT -i lo -j ACCEPT
7 -A INPUT -i lo -j ACCEPT
8 -A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
9 -A INPUT -s 192.168.0.103/32 -j ACCEPT
10 -A INPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
11 -A INPUT -s 192.168.0.103/32 -j DROP
12 -A OUTPUT -o lo -j ACCEPT
13 -A OUTPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
14 -A OUTPUT -d 192.168.0.103/32 -j ACCEPT
15 -A OUTPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
16 -A OUTPUT -d 192.168.0.103/32 -j DROP
17 COMMIT
18 # Completed on Sun Jun  4 20:20:51 2023
```

Рисунок 4.13 – Вміст файлу

Вміст даного файлу, показано в додатку А.

Тепер розглянемо, як, використовуючи засоби IPTables, зробити доступним трафік SSH, щоб можна було підключатися до VPS віддалено. Додаємо правило, що приймає tcp-трафік, який надходить на 22-й порт за допомогою команди:

«iptables -A INPUT -p tcp --dport 22 -j ACCEPT»

Додаємо правило, яке дозволяє трафік на SSH-порт із вказаної IP-адреси з допомогою команди:

«iptables -A INPUT -p tcp -s 192.168.0.103 --dport 22 -j ACCEPT»

Команда установки вихідного з'єднання (пінг та запуск програмного забезпечення):

«iptables -I INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT№

Зберігаємо правила та перевіряємо поточні налаштовані правила

```
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -A INPUT -p tcp -s 192.168.0.103 --dport 22 -j ACCEPT
daynilova@daynilova-Lenovo-ideapad-330-15IGM:~$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             ctstate RELATED,ESTABLISHED
ACCEPT     all  --  192.168.0.103        anywhere
ACCEPT     all  --  anywhere             anywhere             ctstate NEW,ESTABLISHED
DROP       all  --  192.168.0.103        anywhere
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:ssh
ACCEPT     tcp  --  192.168.0.103        anywhere             tcp dpt:ssh

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             ctstate RELATED,ESTABLISHED
ACCEPT     all  --  anywhere             192.168.0.103
ACCEPT     all  --  anywhere             anywhere             ctstate NEW,ESTABLISHED
DROP       all  --  anywhere             192.168.0.103
```

Рисунок 4.14 – Перевірка доступності трафіку SSH

Багато протоколів вимагають двостороннього зв'язку, що означає, що якщо ми хочемо дозволити SSH-підключення до вашої системи, вам потрібно мати вхідні та вихідні правила для цього протоколу. Існують стани з'єднання, які дозволяють двосторонній зв'язок, але дозволяють встановлювати тільки один спосіб з'єднання.

Розглянемо приклад, де ми дозволяємо SSH-з'єднання від IP-адреси 192.168.0.103, але SSH-з'єднання до IP-адреси 192.168.0.103 не дозволено. Однак, системі дозволено надсилати інформацію назад через SSH, коли сеанс вже встановлений, що дозволяє SSH-зв'язок між цими двома хостами. Тому ми дозволяємо SSH-з'єднання від IP-адреси 192.168.0.103, але не дозволяємо SSH-з'єднання до IP-адреси 192.168.0.103, за допомогою наступних команд:

«sudo iptables -A INPUT -p tcp --dport ssh -s 192.168.0.103 -m state --state NEW,ESTABLISHED -j ACCEPT»

«sudo iptables -A OUTPUT -p tcp --sport 22 -d 192.168.0.103 -m state --state ESTABLISHED -j ACCEPT»

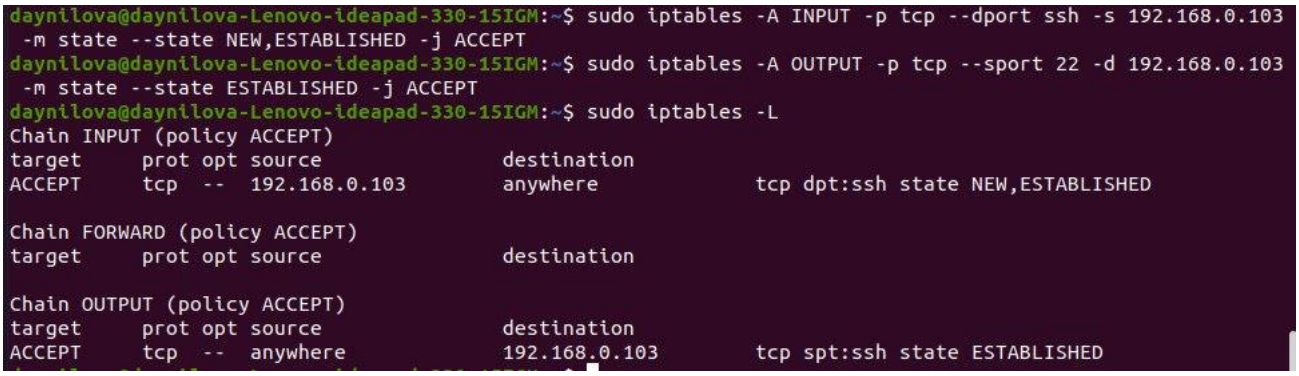

Рисунок 4.15 – Перевіряємо двосторонній зв’язок з одним способом з’єднання

Після всіх проведених маніпуляцій втановлюємо правила нашого міжмережевого екрана за замовчуванням.

Рисунок 4.16 – Встановлюємо наш міжмережевий екран за замовчуванням

ВИСНОВКИ

Дипломна робота бакалавра на тему "Розробка архітектури і побудова міжмережевого екрану" присвячена детальному дослідженню різних архітектур міжмережевих екранів, їх недоліків та переваг. В результаті проведеного аналізу була обрана архітектура "екранування маршрутизатору" для розробки власного міжмережевого екрану. Для реалізації цієї архітектури були використані операційні системи Ubuntu та Windows.

Перевагою обраної архітектури "екранування маршрутизатору" є те, що вона дозволяє контролювати трафік, що проходить через мережевий роутер. Це дає змогу забезпечити більш гранульований рівень безпеки, блокуючи несанкціонований доступ та шкідливі загрози. Крім того, ця архітектура може бути ефективно реалізована за допомогою інструменту iptables, який надає гнучкі налаштування фільтрації пакетів.

Вибір операційних систем Ubuntu та Windows для розробки міжмережевого екрану також має свої переваги. Ubuntu є безкоштовною та відкритою операційною системою, що забезпечує гнучкість та доступність для розробки. Вона має широкий спектр інструментів та пакетів, які допомагають у розробці та налаштуванні міжмережевого екрану. З іншого боку, Windows є поширеною операційною системою, що забезпечує зручний інтерфейс користувача та підтримку великої кількості програмного забезпечення.

Після розробки міжмережевого екрану було проведено тестування з метою перевірки його функціоональностей та досягнення мети дипломної роботи. Під час тестування були виконані різні сценарії, що включали перевірку фільтрації трафіку, блокування небезпечних пакетів і виявлення потенційно шкідливої активності.

Результати тестування показали, що розроблений міжмережевий екран, заснований на архітектурі "екранування маршрутизатору" та побудований на операційних системах Ubuntu та Windows, успішно виконує свої функції. Він забезпечує контроль трафіку, блокує несанкціонований доступ та шкідливі програми, тим самим підвищуючи рівень безпеки мережі.

Під час розробки міжмережевого екрану були враховані переваги та недоліки обраних архітектур і операційних систем. "Екранування маршрутизатору" дозволяє забезпечити гнучкість та ефективність фільтрації пакетів, а Ubuntu та Windows надають зручне середовище розробки та підтримку широкого спектру інструментів.

Загалом, дипломна робота бакалавра з теми "Розробка архітектури і побудова міжмережевого екрану" успішно оглянула різні архітектури міжмережевих екранів, провела дослідження їх переваг і недоліків. На основі цього дослідження було обрано архітектуру "екранування маршрутизатору" та розроблено міжмережевий екран на операційних системах Ubuntu та Windows. Під час тестування досягнута мета дипломної роботи, підтверджено ефективність розробленого міжмережевого екрану в забезпеченні безпеки мережі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1) DeCarlo A. L., Ferrell R. G. The 5 Different Types of Firewalls Explained. *Security*. URL: https://www.techtarget.com/searchsecurity/feature/The-five-different-types-of-firewalls (дата звертання:13.03.2023)
2) Про затвердження Положення про порядок розроблення, виробництва та експлуатації засобів криптографічного захисту інформації : Наказ Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 20.07.2007 р. № № 141.
3) Stallings William NETWORK SECURITY ESSENTIALS: APPLICATIONS AND STANDARDS FOURTH EDITION. 4th ed. Pearson, 2016. 464 p.
4) Radware, “7 Most Common Attack Types Web Application Firewall (WAF) Is Designed To Stop | Radware.” URL: https://www.radware.com/cyberpedia/application-security/7-most-common-attack-types/(дата звертання:20.03.2023)
5) External Firewall Attacks. *ITPro Today: IT News, How-Tos, Trends, Case Studies, Career Tips, More*. URL: https://www.itprotoday.com/security/external-firewall-attacks#close-modal(дата звертання:12.04.2023)
6) Firewall Architecture - TAE. *TAE - A Tutorial Website with Real Time Examples*. URL: https://www.tutorialandexample.com/firewall-architecture(дата звертання:12.04.2023)
7) ISO OSI. *ni.biz.ua*. URL: http://ni.biz.ua/1/1_5/1_54930_tipi-mezhsetevih-ekranov-i-urovni-modeli-ISO-OSI.html(дата звертання:12.04.2023)

8) БЕЗПЕКА INTERNET: БРАНДМАУЕРИ. *ОБРОБКА І ПЕРЕДАЧА ІНФОРМАЦІЇ. СУЧАСНІ КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ*. 2008. Т. 2. С. 5–10.

9) Cheswick W. R., Bellovin S. M., Rubin A. D. Firewalls and Internet Security: Repelling the Wily Hacker. 2nd ed. Addison-Wesley Professional, 2003. 464 p.

10) Stewart J. M., Kinsey D. Network Security, Firewalls, and VPNs. 3rd ed. Jones & Bartlett Learning, 2020. 481 p.

11) Miller L. C. Next Generation Firewalls for Dummies. Wiley Publishing Inc., 2011. 76 p.

12) Maximum Windows Security / Mark Burnett, Chris Dovle, Chris Amaris, M. Burnett et al. Sams Publishing, 2003. 624 p.

13) What Is a Firewall and Why Is it Important in Cyber Security?. *Datto | IT Solutions Built for You*. URL: https://www.datto.com/blog/what-is-a-firewall-and-why-is-it-important-in-cyber-security(дата звертання:01.05.2023)

14) Point C. What is a Firewall. The Different Types of Firewalls. *Bferrite*. 2022.

15) Simplilearn. What Is Firewall ? | Firewall Explained | Firewalls and Network Security | Simplilearn, 2021. *YouTube*. URL: https://www.youtube.com/watch?v=9GZlVOafYTg(дата звертання:02.05.2023)

16) Deshpande C. What Is Firewall: Types, How Does It Work, Advantages & Its Importance. 2022. Vol. 8. P. 5–12.

17) How to configure iptables on Ubuntu. *UpCloud*. URL: https://upcloud.com/resources/tutorials/configure-iptables-ubuntu(дата звертання:16.05.2023)

18) Ellingwood J. How To Set Up a Firewall Using Iptables on Ubuntu. *Digital Ocean*. 2014. Vol. 1, no. 2.

URL: https://www.digitalocean.com/community/tutorials/how-to-set-up-a-firewall-using-iptables-on-ubuntu-14-04(дата звертання:16.05.2023)

19) Introduction of Firewall in Computer Network - GeeksforGeeks. *GeeksforGeeks*.
URL: https://www.geeksforgeeks.org/introduction-of-firewall-in-computer-network/(дата звертання:20.05.2023)

20) Zwicky E. D., Chapman D. B. Building Internet Firewalls: Internet and Web Security. 2nd ed. O'Reilly Media, 2000. 896 p.

21) Paid Program: Stopping Tomorrow's Super-Hacks. *TII*.
URL: https://partners.wsj.com/tii/catalyzing-change/outsmarting-the-super-hackers/?utm_medium=content_discovery&utm_source=google-search&gclid=Cj0KCQjw4NujBhC5ARIsAF4Iv6czvPs7a1CJAjyOWP91y6wJVUjumL2MbbUwaj5EiqZeWLgWaCaL-UgaAgRvEALw_wcB(дата звертання:20.05.2023)

22) Wallander K. Firewall. 8th ed. Vintage Digital, 2008. 547 p.

23) What Is a Firewall? - Definition | Proofpoint US. *Proofpoint*.
URL: https://www.proofpoint.com/us/threat-reference/firewall(дата звертання:21.05.2023)

ДОДАТОК А

```
# Generated by iptables-save v1.8.4 on Sun Jun  4 20:20:51 2023
*filter
:INPUT DROP [0:0]
:FORWARD DROP [0:0]
:OUTPUT DROP [0:0]
-A INPUT -i lo -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.0.103/32 -j ACCEPT
-A INPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
-A INPUT -s 192.168.0.103/32 -j DROP
-A OUTPUT -o lo -j ACCEPT
-A OUTPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A OUTPUT -d 192.168.0.103/32 -j ACCEPT
-A OUTPUT -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
-A OUTPUT -d 192.168.0.103/32 -j DROP
```

COMMIT

Completed on Sun Jun 4 20:20:51 2023