Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Інститут електроенергетики
Факультет інформаційних технологій
Кафедра безпеки інформації та телекомунікацій

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня бакалавра

студента *Нашиванько Романа Олексійовича*

академічної групи 125-18-1

спеціальності *125 Кібербезпека*

спеціалізації[1]

за освітньо-професійною програмою *Кібербезпека*

на тему Захист веб-сервісів від DDoS-атак на прикладному рівні

Керівники	**Прізвище, ініціали**	**Оцінка за шкалою**		**Підпис**
		рейтинговою	**інституційною**	
кваліфікаційної роботи	д.ф.- м.н. проф. Кагадій Т.С.	90	відмінно	
розділів:				
спеціальний	ст. викл. Тимофєєв Д.С.	90	відмінно	
економічний	к.е.н., доц. Пілова Д.П.	90	відмінно	

Рецензент				

Нормоконтролер	ст. викл. Тимофєєв Д.С.	90	відмінно	

Дніпро

2022

ЗАТВЕРДЖЕНО:
завідувач кафедри
безпеки інформації та телекомунікацій
_____ д.т.н., проф. Корнієнко В.І.

«_____»_____20___року

ЗАВДАННЯ
на кваліфікаційну роботу ступеня бакалавра

студенту *Нашиванько Р.О.* **академічної групи** *125-18-1*
(прізвище та ініціали) (шифр)

спеціальності *125 Кібербезпека*

спеціалізації

за освітньо-професійною програмою *Кібербезпека*

на тему Захист веб-сервісів від DDoS-атак на прикладному рівні

Затверджену наказом ректора НТУ «Дніпровська політехніка» від 18.05.2022 № 268-с

Розділ	**Зміст**	**Термін виконання**
РОЗДІЛ 1	Провести загальний аналіз актуальності веб-сервісів, DDoS-атак, та методів захисту	07.05.22
РОЗДІЛ 2	Виконати аналіз DDoS-атак прикладного рівня, запропонувати методи та засоби захисту	25.05.22
РОЗДІЛ 3	Виконати техніко-економічне обгрунтування доцільності запровадження методу захисту веб-сервісу	07.06.22

Завдання видано _____ Кагадій Т.С.
(підпис керівника)

Дата видачі завдання: 10.01.2022

Дата подання до екзаменаційної комісії: 14.06.2022

Прийнято до виконання _____ Нашиванько Р.О.
(підпис студента)

РЕФЕРАТ

Пояснювальна записка: 52 с., 29 рис., 2 табл., 5 додатків, 16 джерел.

Об’єкт розробки: методи захисту веб-сервісів від DDoS-атак.

Предмет розробки: рекомендації з захисту веб-сервісів від DDoS-атак на прикладному рівні моделі OSI.

Мета розробки: забезпечення достатнього рівня захисту веб-сервісів від DDoS-атак на прикладному рівня моделі OSI.

Методи розробки: аналіз, опис, проектування та розрахунки

У першому розділі було розглянуто актуальність веб-сервісів їх роль та функції в сучасному світі. Було приведено статистичні дані щодо DDoS-атак. Проаналізовано методи DDoS-атак та методи захисту від DDoS-атак.

У другому розділі розглянуто DDoS-атаки на прикладному рівні. Розроблено рекомендації з забезпечення захисту веб-сервісу від DDoS-атак на прикладному рівні.

У третьому розділі проведено розрахунок капітальних та експлуатаційних витрат на проектування та впровадження системи захисту від DDoS-атак прикладного рівня.

ІНФОРМАЦІЙНА БЕЗПЕКА, ВЕБ-СЕРВІС, ВРАЗЛИВІСТЬ, DDoS-АТАКА, СХЕМА МЕРЕЖІ, ЗАХИСТ ІНФОРМАЦІЇ, ІНФОРМАЦІЙНІ ПОТОКИ, АНАЛІЗ МЕРЕЖІ, ОБ’ЄКТ ІНФОРМАЦІЙНОЇ ДІЯЛЬНОСТІ, МОНІТОРИНГ МЕРЕЖІ, ВЕБ-ВРАЗЛИВОСТІ.

Abstract

Explanatory note: 52 pages, 29 images, 2 tables, 5 applications, 16 sources.

Object of development: methods of protecting web-service from DDoS-attacks

Subject of development: recommendations for protecting web services from DDoS attack at the application level of the OSI model.

The purpose of development: provide a sufficient level of protection web-services from application layer DDoS-attacks

Methods of developed: analyses, descripting, designing and calculations.

The first section discusses the relevance of web services, their role and functions in the modern world. Statistics on DDoS attacks were provided. DDoS attack methods and DDoS attack protection methods are analysed.

The second section discusses DDoS attacks at the application level. Recommendations have been developed to protect the web service from DDoS attacks at the application level.

In the third section, the calculation of capital and operating costs for the design and implementation of a system of protection against DDoS-attacks of the application level.

The practical significance of the work is to increase the level of security information to minimize the potential costs of information threats.

INFORMATION SECURITY, WEB SERVICE, VULNERABILITY, DDoS ATTACK, NETWORK SCHEME, INFORMATION PROTECTION, INFORMATION FLOWS, ANALYSIS OF NETWORKS, OBJECT OF INFORMATION ACTIVITY, MONITORING OF NETWORKS, WEB VULNERABILITY.

СПИСОК УМОВНИХ СКОРОЧЕНЬ

DDoS – Distributed Denial of Service

HTTP – HyperText Transfer Protocol

DNS – Domain Name System

TCP – Transmission Control Protocol

UDP – User Datagram Protocol

WAF – Web Application Firewall

IDS – Intrusion Detection System

OSI – Open System Interconnection

DOM – Document Object Model

SOAP – Simple Object Access Protocol

XML – eXtensible Markup Language

SQL – Structure Query Language

ACL – Access Control List

АС – автоматизована система

ІТ – інформаційні технології

ІТС – інформаційно-телекомунікаційна система

КС – комп'ютерна мережа (або система)

НД ТЗІ – нормативний документ технічного захисту інформації

НСД – несанкціонований доступ

ОС – операційна система

ПЗ – програмне забезпечення

ПК – персональний комп'ютер

ЗМІСТ

с.

Вступ

Предмет розробки: рекомендації з захисту веб-сервісів від DDoS-атак на прикладному рівні моделі OSI.

Об’єкт розробки: методи захисту веб-сервісів від DDoS-атак.

Мета розробки: забезпечення достатнього рівня захисту веб-сервісів від DDoS-атак на прикладному рівня моделі OSI.

Розподілена відмова в обслуговуванні (DDoS) – це спроба зловмисника порушити нормальний трафік сервера, сервіса або мережі, яка переповнює ціль або мережеві пристрої, наприклад маршрутизатор, фальсифікованим трафіком. Атака націлена на те щоб зробити ресурс в інтернеті недоступним для користувачів. Це порушує один з найважливіший принцип кібербезпеки – доступність. Компанія, яка критична пов’язана з використанням веб-сервісів, може стати жертвою DDoS, та як наслідок призвести до втрати клієнтів, що стане причиною фінансових проблем. Використовуючи рішення для захисту від DDoS атак, компанія має шанс, але не 100 відсотковий, не стати жертвою зловмисника.

При DDoS атаках використовуються мережі комп’ютерів, які називаються – зомбі або боти. Скомпрометовані машини, які утворюють бот-нет, можуть включати в себе як персональні комп’ютери користувачів так і інші мережеві ресурси, наприклад IoT пристрої.

Найочевиднішим симптомом DDoS атак є те, що сайт або служба раптово починають працювати дуже повільно або стають недоступними. Але зазвичай потрібно подальше розслідування. Інструменти аналізу трафіка можуть допомогти виявити такі ознаки DDoS-атак:

- Підозрілий обсяг трафіка, що надходить з одної IP адреси або з діапазону IP-адрес.
- Потік трафіка від користувачів, які мають єдиний поведінковий профіль, наприклад тип пристрою, геолокацію або версію веб-переглядача.

- Незрозумілий сплеск запитів до однієї сторінки або кінцевої точки.
- Дивні моделі руху, такі як стрибки в непарні години дня або моделі, які здаються неприродними (наприклад, стрибок кожні 10 хвилин).

Існують інші, більш специфічні ознаки DDoS-атаки, які можуть відрізнятися в залежності від типу атаки.

РОЗДІЛ 1. СТАН ПИТАННЯ. ПОСТАНОВКА ЗАДАЧІ

1.1 Аналіз актуальності задачі

На сьогоднішній день веб-сервіси є найпоширенішим додатком в мережі Інтернет. Через веб-додатки можна зробити майже будь-яку дію, наприклад: отримувати онлайн-освіту, переглядати новини, робити онлайн покупки, замовляти квитки та ще багато іншого.

На рис. 1.1 наведено графік [1], який показує динаміку зростання веб-сервісів в світі:

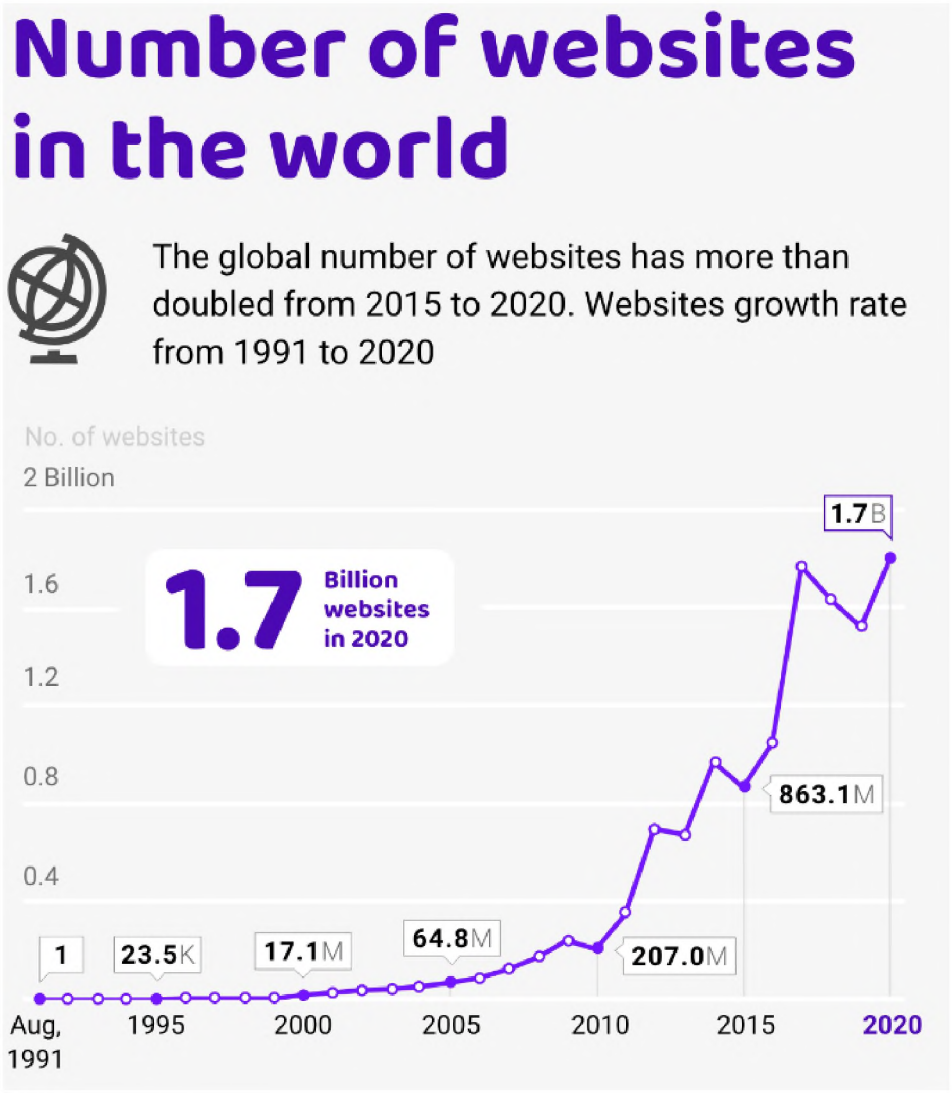

Рисунок 1.1 – Кількість веб-сервісів в світі

Різні структури використовують веб-сервіси, включаючи державні органи та банки. Така популярність цікавить зловмисників, які розробили велику кількість атак на веб-ресурси.

Далі приведено графік [2], які галузі найбільш піддаються веб-атакам:

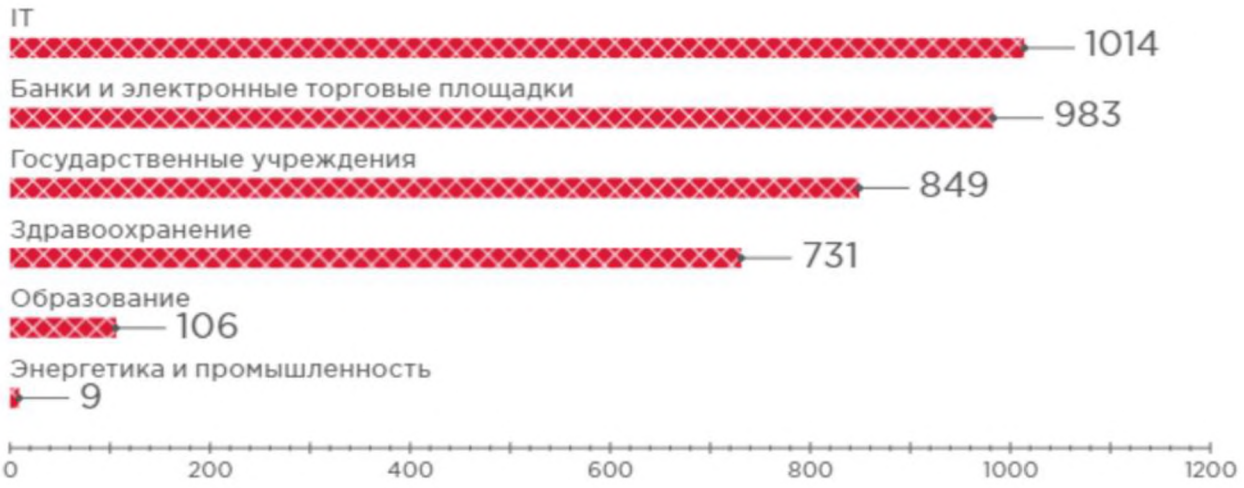

Рисунок 1.2 – Галузі які найбільш піддаються веб-атакам

Існують багато технік та методів для завдання шкоди веб-ресурсам. Одна з головних проблем безпеки, яку потрібно вирішити – це розподілена атака на відмову в обслуговуванні (DDoS-атаки).

DDoS — це скоординована атака, яка запускається за допомогою великої кількості скомпрометованих хостів. На початковому етапі зловмисник визначає вразливості в одній або кількох мережах для встановлення шкідливих програм на кількох машинах, щоб контролювати їх з віддаленого розташування. На пізнішому етапі зловмисник використовує ці скомпрометовані хости, щоб надіслати пакети на цільову машину, яка зазвичай знаходиться за межами вихідної мережі інфікованих хостів, без відома цих скомпрометованих машин. Залежно від інтенсивності пакетів атаки та кількості хостів, які використовуються для атаки, у мережі жертви виникає відповідна шкода. Якщо зловмисник може використати велику кількість скомпрометованих хостів, мережа або веб-сервер можуть бути порушені протягом короткого часу.

DDoS одна з найпоширеніших та небезпечних атак, яка дуже популярна среди зловмисників. Так за оцінкою Cisco кількість атак подвоїлось з 7,9 мільйона у 2018 році до 15,4 мільйона атак у 2022 році.

Компанії несуть збитки з точки зору фінансів. За оцінками різноманітних дослідницьких фірм, з поточною вартістю 2,4 мільярда доларів у 2019 році ринок захисту та пом'якшення DDoS-атак майже подвоїться до 4,7 мільярда доларів до 2024 року. Це сукупний річний темп зростання (CAGR) у 14 відсотків.

За даними Кожного року ми можемо бачити як DDoS-атаки стають більш потужними. Ми можемо переконатися в цьому завдяки статистиці DDoS-атак на інфраструктуру підконтрольну Google з 2010 по 2020 роки [3]. На рис. 1.3 видно експонентне зростання потужності атак. Найбільші пікові показники атак спрямовані на мережеві вузли.

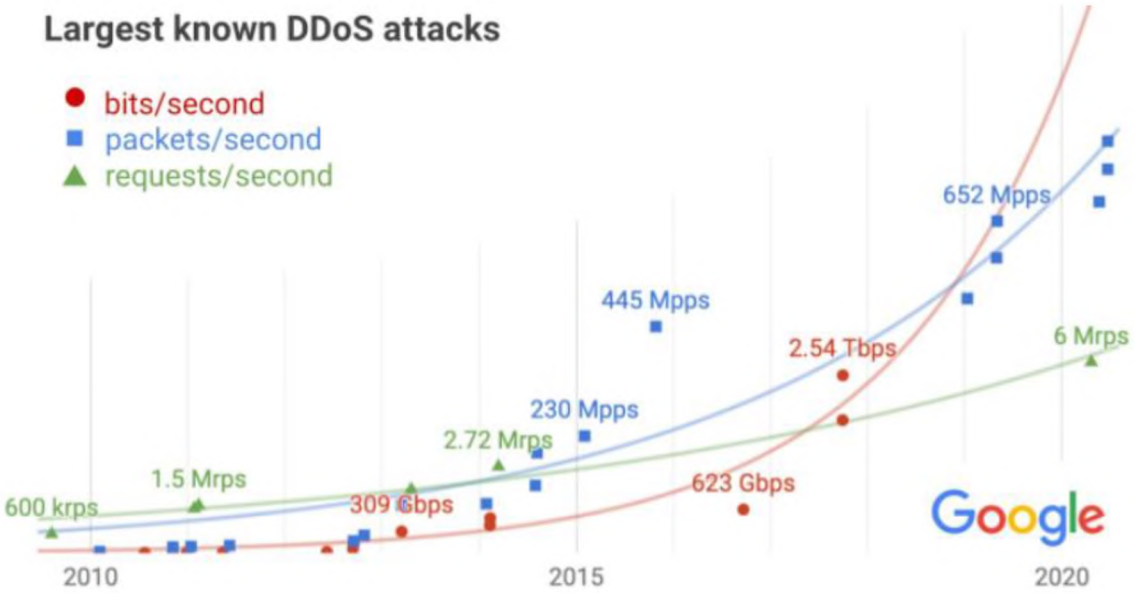

Рисунок 1.3 – Динаміка характеристик відомих DDoS-атак

bits/second – атаки, спрямовані на мережеві вузли

packets/second – атаки, спрямовані на мережеве обладнання або DNS-сервери

request/second – атаки, спрямовані на серверні програми

DDoS-атаки продемонстрували помітне зниження з початку 2021 року до кінця року, хоча частота атак залишалася дещо постійною протягом останніх двох років: у 2021 році було лише на 3% менше, ніж у 2020 році. Але хоча на малюнку 1.4 показано загальне зниження атак, частота протягом 2021 року, це також показує, що розміри атак значно зросли. Хоча пікові розміри атаки залишалися незмінними протягом 2020 року, приблизно 200 Мбіт/с, все змінилося в лютому 2021 року, коли було виявлено найбільшу атаку, вагою 500 Мбіт/с. Однак цей рекорд протримався недовго, оскільки в 2021 році спостерігалися все більші й масштабніші атаки, кульмінацією яких стала атака 1,4 Тбіт/с у листопаді. Крім пікових розмірів атаки, середній розмір атаки також зріс. Середній розмір атаки в першому кварталі 2020 року становив 5 Гбіт/с і понад 21 Гбіт/с у четвертому кварталі 2021 року.

На рис. 1.4 [4] представлена динаміка зростання потужності DDoS-атак за 2020 та 2021 роки:

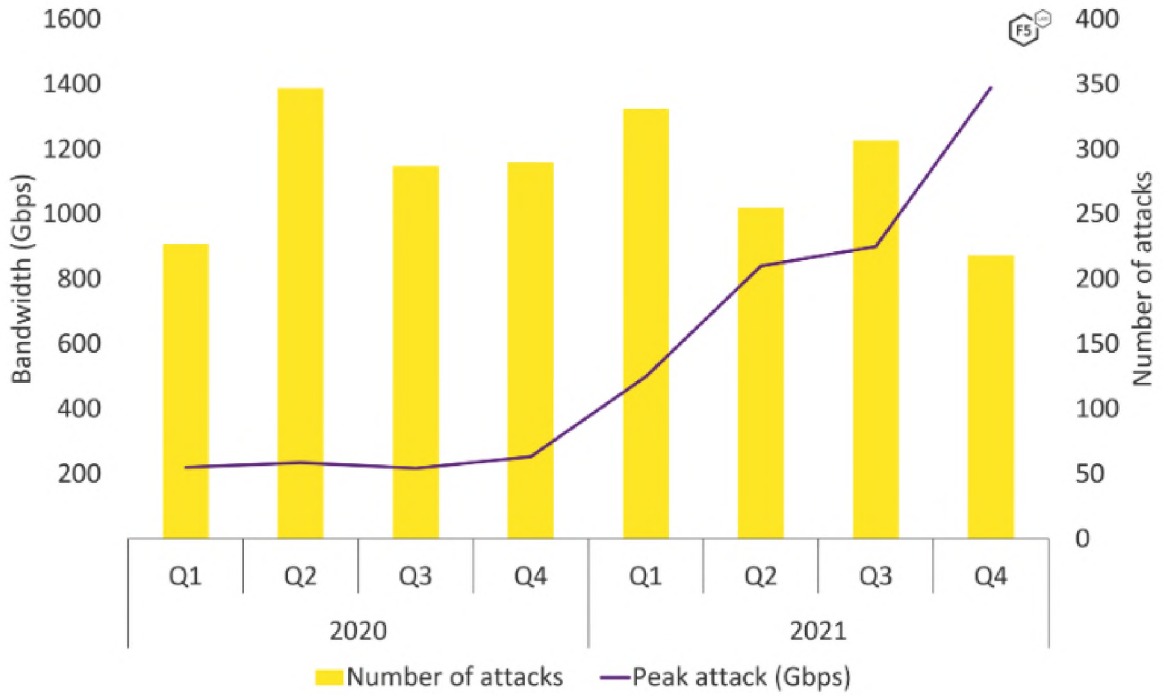

Рисунок 1.4 – Динаміка зростання потужності DDoS-атак

Нещодавні гучні [5] атаки показують масштаб та потужність порушень які вже обчислюється терабітами за секунду.

- Атака на GitHub

Ця атака примітна тим, що в ній для генерації трафіка не знадобився ботнет. В цьому порушенні використовувалася вразливість Memcached-серверів.

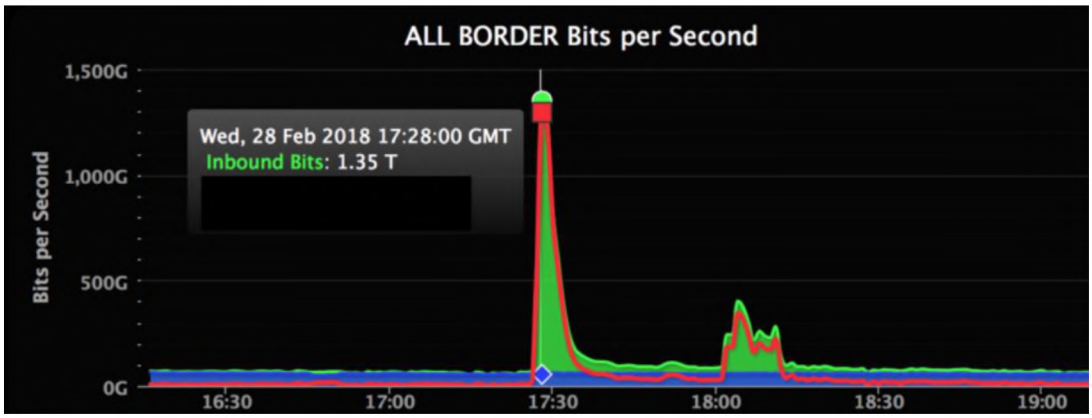

Рисунок 1.5 – Рівень трафіку під час атаки

- Атака на Cloudflare

Рекордна DDoS-атака в 15 мільйонів запитів була зафіксована в квітні 2022 року, та була націлена на клієнта сервісу Cloudflare. Експерти назвали цю атаку, одну з крупніших HTTPS DDoS-атак в історії. Тривалість цієї атаки дорівнює 15 секундам.

Як відмітили ІБ-експерти – «HTTPS DDoS-атаки обходяться дорожче з погляду необхідних обчислювальних ресурсів через вищу вартість установки безпечного зашифрованого TLS-з'єднання. Тому зловмиснику дорожче розпочати атаку, а жертві – зупинити її.»

1.2 Класифікації DDoS-атак

Хоча майже всі порушення типу DDoS передбачають перевантаження цільового пристрою або мережі трафіком, атаки можна розділити на три

категорії. Зловмисник може використовувати один або кілька різних векторів атаки або циклічні вектори атаки у відповідь на контрзаходи, які вживає ціль. Виділяються саме такі рівні:

- Атаки на прикладний рівень
- Атаки на протоколи
- Об’ємні атаки

На рис. 1.6 [4] продемонстровано частота використання типів атак за 2021 рік:

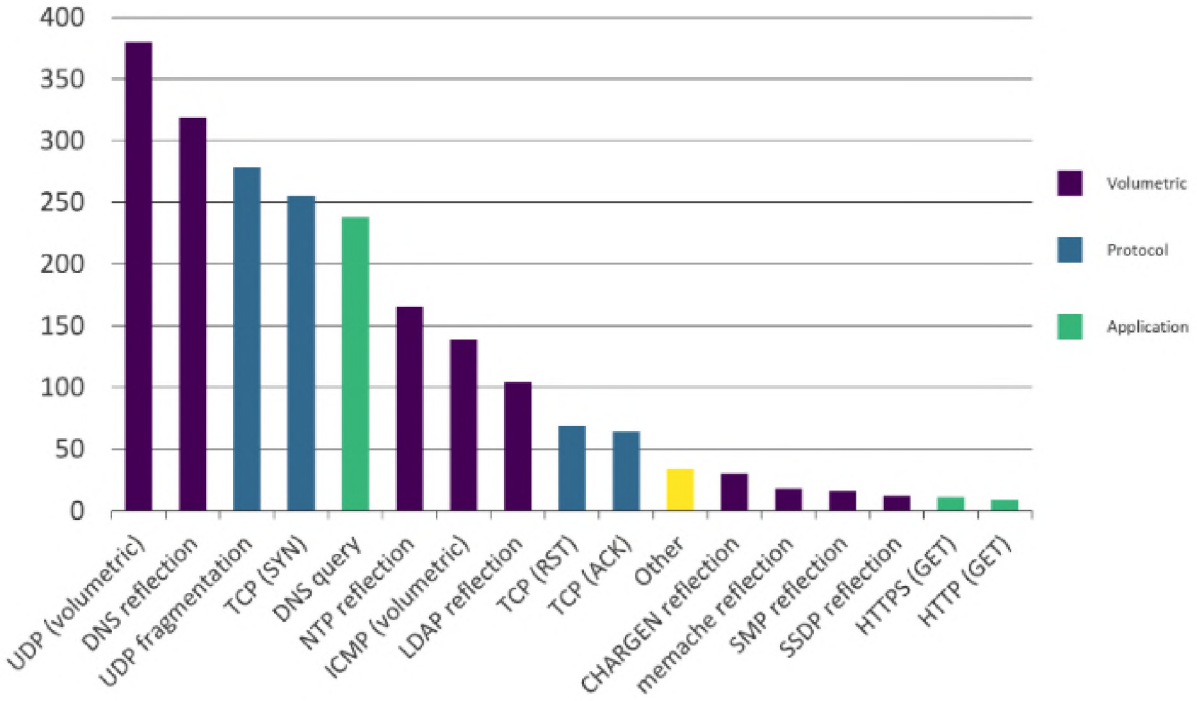

Рисунок 1.6 – Частота використання типів атак за 2021 рік

Далі приведена статистика яка показує процент використання різних видів атак в 2020 та 2021 роках:

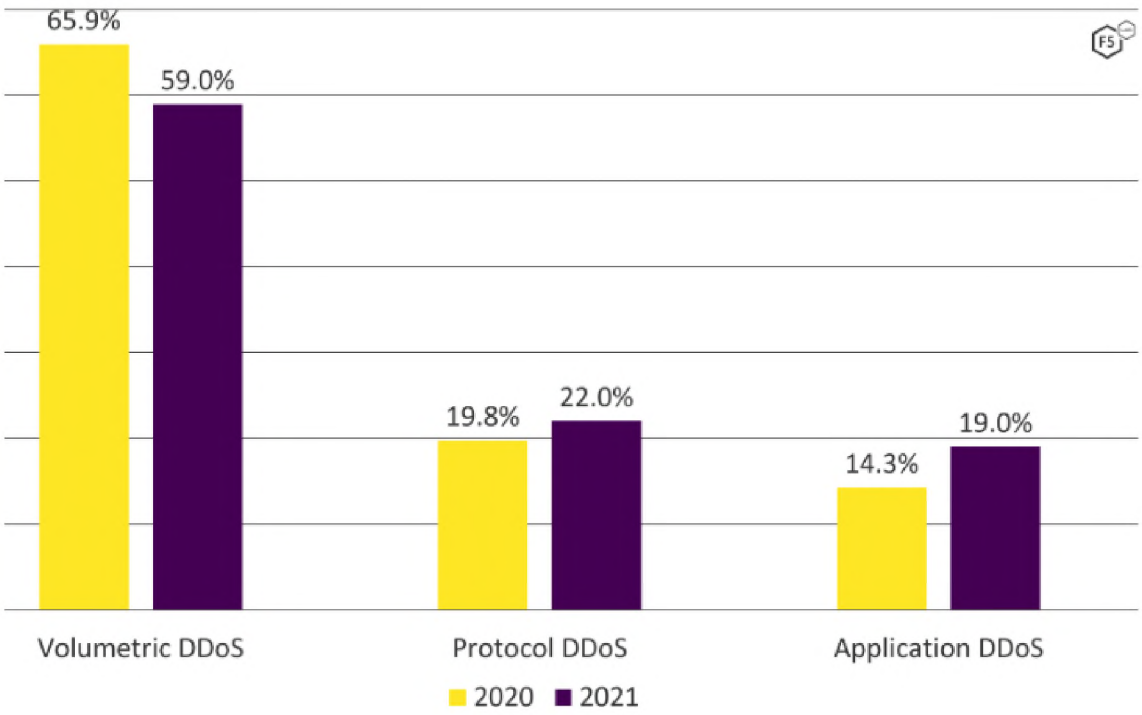

Рисунок 1.7 – Графік порівняння типів атак за 2020 та 2021 рік

У першому кварталі DDoS-атаки на прикладному рівні зросли на 164% р/к та на 135% за квартал – найактивніший квартал за останній рік.

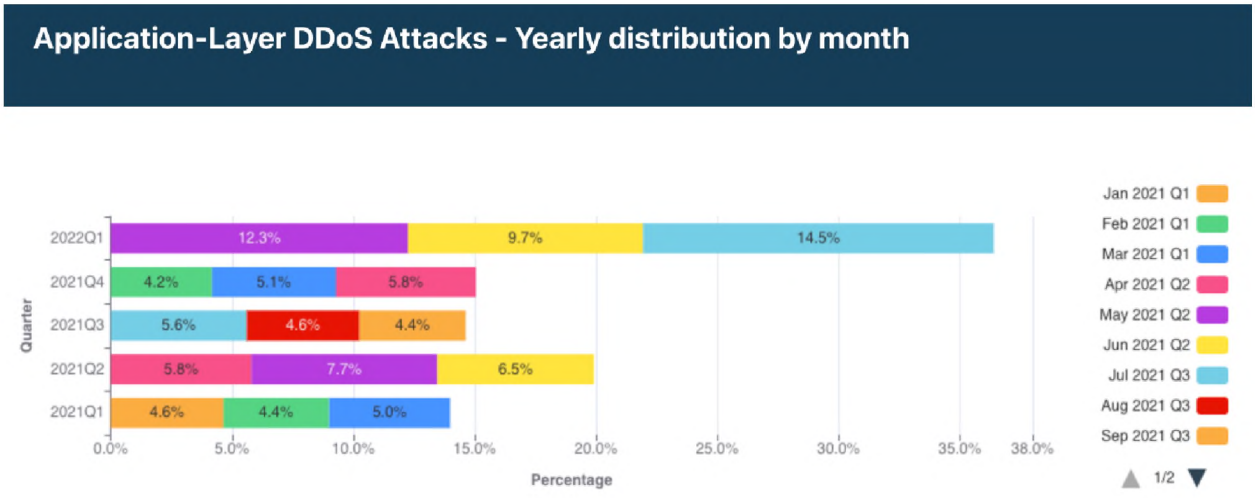

Рисунок 1.8 – Річний розподіл по місяцях

У той час як HTTP DDoS-атаки різко зросли в першому кварталі, DDoS-атаки мережевого рівня фактично зменшилися на 58% за квартал, але все ще зросли на 71% р/р.

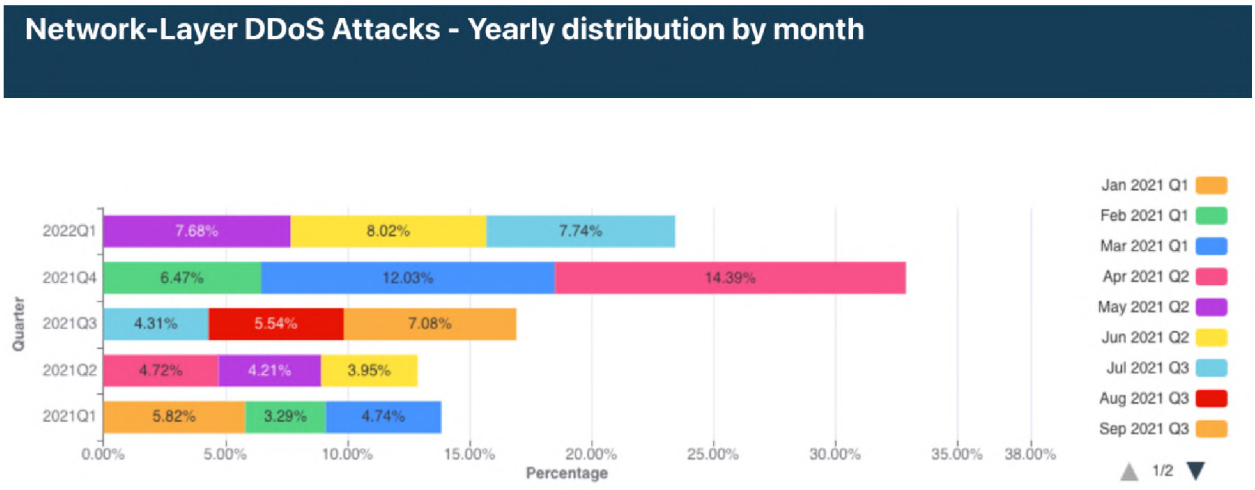

Рисунок 1.9 – Річний розподіл по місяцях

Атаки спрямовані на 7 рівень моделі OSI, де веб-сторінки генеруються на сервері та доставляють у відповідь на запити HTTP. Виконання одного HTTP-запиту на стороні клієнта з точки зору обчислень дешеве, але відповідати на нього цільовому серверу може бути дорого, оскільки сервер часто завантажує кілька файлів і виконує запити до бази даних, щоб створити веб-сторінку.

Приклад:

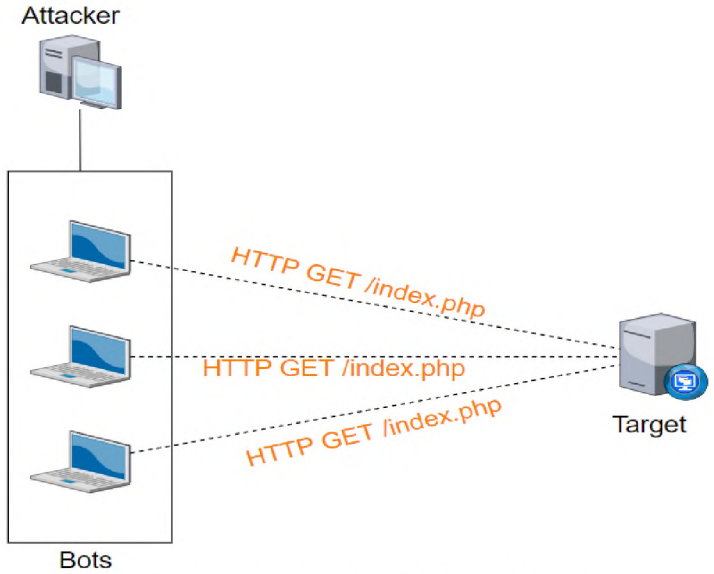

Рисунок 1.10 – HTTP Flood

HTTP Get/Post Flood – в цьому методі генерується велика кількість HTTP запитів на максимальні великі елементи сайтів. Таким чином, не потрібно мати велику армію ботів для здійснення цього методу атаки. Окрім GET запитів також можуть посилатися запити POST і здійснюватися інші HTTP дії, що призводять до одного і того ж результату - перевантаження веб-сервера жертви.

Атаки протоколів, викликають порушення роботи служби через надмірне споживання ресурсів сервера та/або ресурсів мережевого обладнання, таких як брандмауери та машини для балансування навантаження. Протокольні атаки використовують слабкі місця в рівнях 3 і 4 стеку протоколів, щоб зробити ціль недоступною.

Приклад:

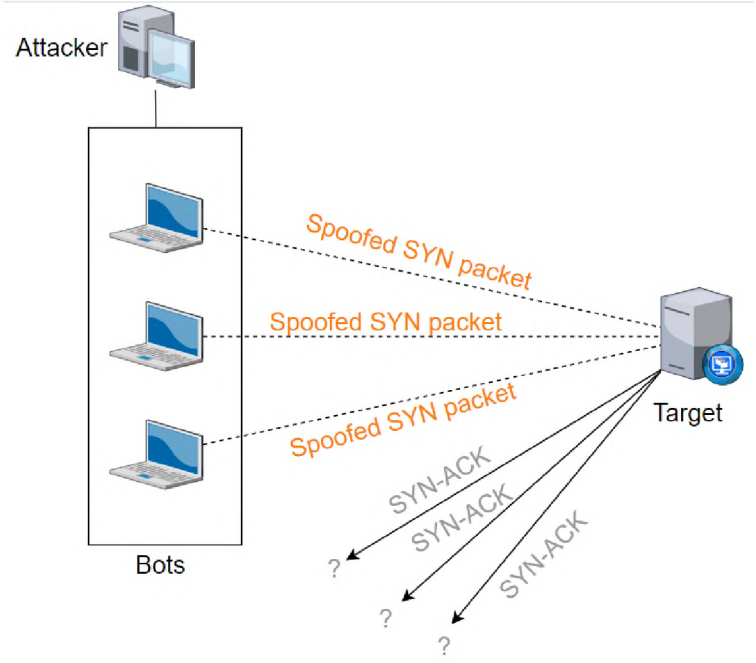

Рисунок 1.11 – SYN Flood

SYN Flood – в цьому методі ми надсилаємо пакети з встановленим прапором SYN на сервер підміняє IP адресу відправника. Згідно рукостисканні TCP, сервер відповідає пакетом з прапорами SYN-ACK, але надсилає його на неіснуючу IP адресу. Як результат в черзі підключень

з’являються полу відкриті з’єднання, які чекають підтвердження від клієнта. Після закінчення певного тайм-ауту ці підключення відкидаються. Метод дуже ефективний та актуальний досі.

При об’ємних атаках зловмисник намагається створити перевантаження, споживаючи всю доступну пропускну спроможність між цільовим елементом і більшим Інтернетом. Великі обсяги даних надсилаються до цілі за допомогою форми посилення або іншого засобу створення масивного трафіку, наприклад запитів від ботнету.

Приклад:

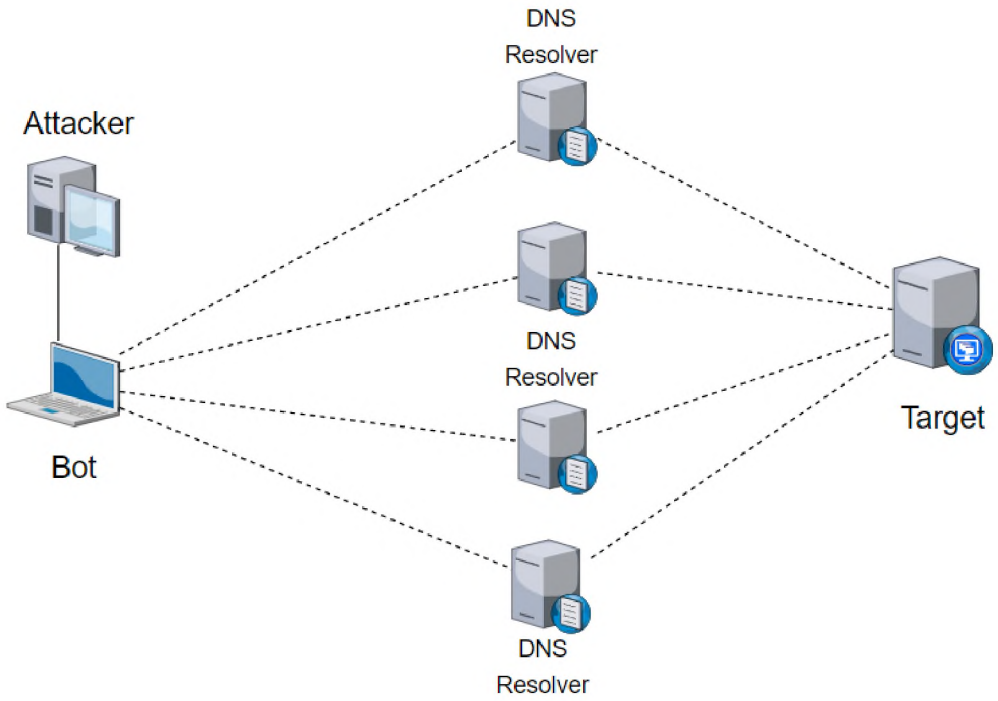

Рисунок 1.12 – DNS Ампліфікація

DNS ампліфікація – ця атака транспортного рівня (4^{й} рівень моделі OSI) використовує специфіку роботи DNS служби в мережі. Суть полягає в тому, щоб запросити у публічного DNS-сервера дані про домен і направити його відповідь на сервер, що атакується. При реалізації даного виду атаки ми формуємо та надсилаємо запит, у відповідь на який DNS-сервер повертає якнайбільше даних. Наприклад, запит списку всіх DNS-записів у певної зони. Даний метод використовує 53 порт.

1.3 Класифікація методів захисту від DDoS-атак

Наразі не існує комплексного методу захисту від усіх відомих форм DDoS-атак. Крім того, багато похідних DDoS-атак постійно розробляються зловмисниками, щоб обійти кожен новий використаний контрзахід.

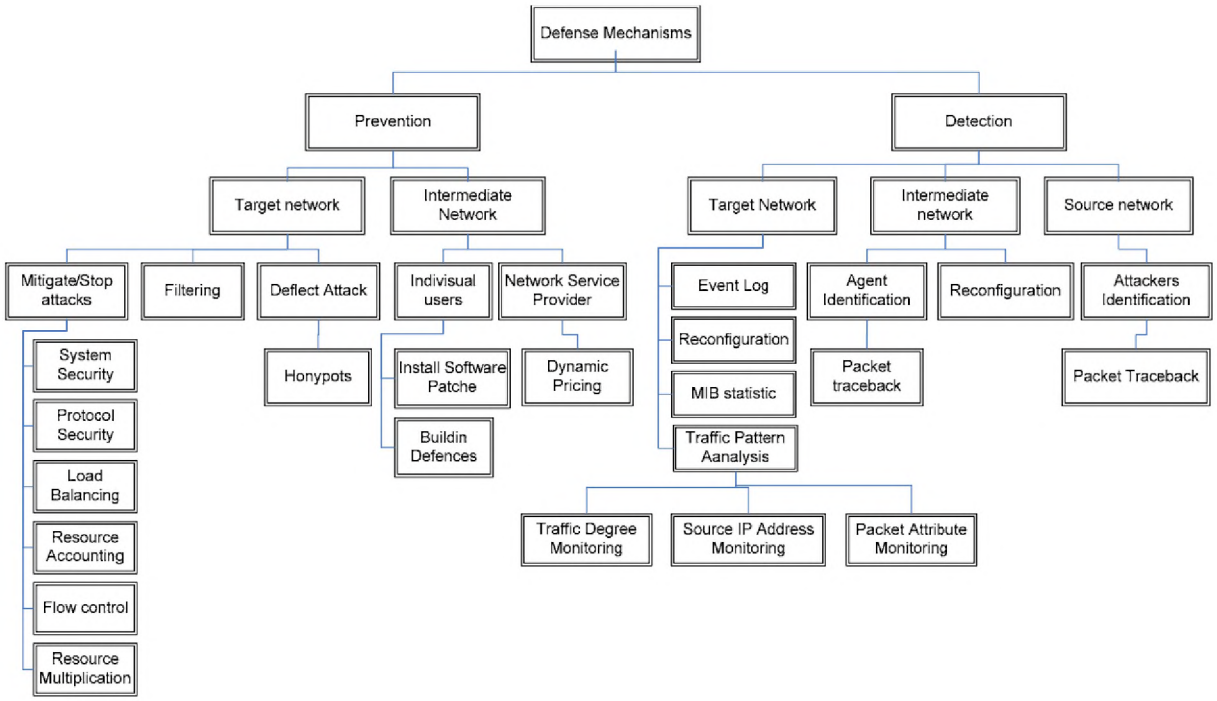

Рисунок 1.13 – Класифікація методів захисту від DDoS-атак

На рисунку 1.13 показана класифікація. DDoS-атаки мають кілька особливостей, які перешкоджають їх успішному виявленню та захисту:

1) DDoS-атаки генерують великий потік обсягів, щоб перевантажити цільову мережу.
2) Важко відрізнити атакуючі пакети від легітимних пакетів.
3) Більшість DDoS-атак використовують підроблені IP-адреси.
4) Велика кількість атакуючих машин і використання вихідної підробка IP-адреси робить зворотне відстеження важким або неможливим.

5) Розподілений характер атак вимагає розподіленої реакції, але співпраці між адміністративними доменами важко досягти.

Весь захисний механізм був розділений на дві категорії:

- Запобігання
- Виявлення

При підході запобігання, дослідники намагаються зупинити атаку на початку. Запобігання може здійснюватися в цільовій мережі або в проміжній мережі.

Приклад:

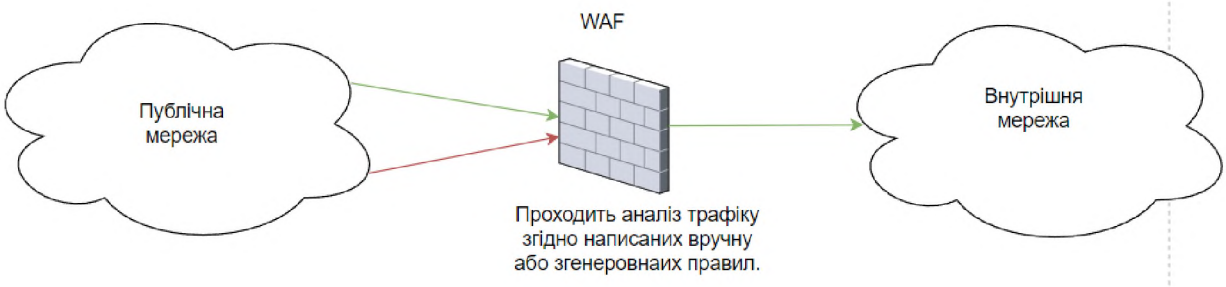

Рисунок 1.14 – WAF

Приклади методів запобігання:

- Механізми безпеки підвищують загальну безпеку системи, захищаючи від нелегального доступу до машини, видаляючи помилки додатків та оновлення протоколу встановлення для запобігання вторгнень та неправомірного використання системи.
- Механізми безпеки протоколу захищають систему на рівні мережевих протоколів (3 та 4 рівні моделі OSI).
- Балансування навантаження може покращити як нормальну продуктивність, так і пом’якшити DDoS-атаки. Якщо на сервері буде помітна високе навантаження, трафік буде переведений на резервні, спеціально для цього виділенні, сервери.

- Контроль потоку – це ще один метод, запропонований для запобігання виходу серверів з ладу. При цьому методі запобігання, налаштовується маршрутизатори, які звертаються до сервера за допомогою логіки для налаштування вхідного трафіку до рівнів, які будуть безпечними для обробки сервером.

Метод виявлення використовує сигнатури атак або навчання нормальної поведінки мережі для виявлення атак. Багато систем виявлення вторгнень написані на основі цього підходу і використовуються штучний аналіз даних техніки розвідки. Його можна використовувати для виявлення атак у цільовій мережі або проміжній мережі.

Приклад:

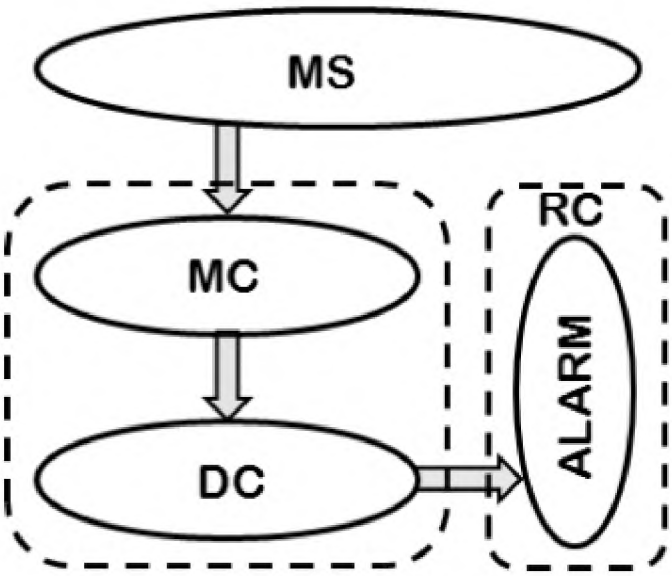

Рисунок 1.15 – Система виявлення вторгнень: загальний вигляд

Чотири основні компоненти – це керування, моніторинг, виявлення та створення тривоги.

- Управляючий компонент контролює потік трафіку в мережі. Він надає інформацію про трафік компоненту моніторингу для аналіз.

- Компонент моніторингу відстежує трафік і аналізує поведінку мережі.
- Компонент виявлення виявляє будь-яку підозрілу поведінку з повагою до нормального робочого характеру мережі. Якщо виявлено будь-яку ненормальну поведінку, він зв'язується з компонентом генерації сигналів.
- Якщо виявлено будь-які відхилення в потоці трафіка, компонент генерації тривоги генерує сигнал, щоб повідомити адміністратора, та можна було б належним чином обробити вторгнення.

Приклади методів виявлення:

- Шаблон трафіку завдяки якому спеціалісти можуть контролювати мережевий трафік на предмет підозрілої активності.
- Аналіз інформації MIB — це ще один метод визначення, коли відбувається DDoS-атака. Дані MIB включають параметри, які вказують різну статистику маршрутизації і мережевих пакетів.
- Механізми реконфігурації змінюють топологію цільової мережі, щоб додати більше ресурсів до цільової мережі.
- Механізми ідентифікації агента надають жертві інформацію про ідентичність машин, які здійснюють атаку. Ідентифікація агента використовує методи зворотного відстеження, які дозволяють використовувати поле вихідної адреси для ідентифікації агента.

1.4 Постановка задачі

Розглянувши актуальність задачі, проаналізувавши статистику та відомі класифікації атаки та відомі методи захисту, була поставлена задача, розробити систему захисту та надати рекомендації для типового веб-сервісу від DDoS-атак на прикладному рівні.

Для цього потрібно вирішити наступні задачі:

- Проаналізувати види атак прикладного рівня
- Проаналізувати методи захисту від атак прикладного рівня
- Обрати метод захисту для типового об’єкту
- Розробити рекомендації із застосування засобів захисту
- Зробити економічне обґрунтування вибраного метода

1.5 Висновки

В розділі було детально розглянуто актуальність задачі, були приведені роль, розповсюдженість та функції веб-сервісів в сучасному світі. Проаналізовано статистику, за якими можна зробити висновок, що DDoS-атаки це велика проблема сучасного Інтернету. Так з 2018 року кількість атак збільшилась у 2 рази. Також зростає і потужність атак, яка вже обчислюється в терабайтах на секунду. Також в розділі наведена класифікація атак та класифікація методів захисту.

РОЗДІЛ 2. СПЕЦІАЛЬНА ЧАСТИНА

2.1 Аналіз DDoS-атак прикладного рівня

DDoS-атаки сьомого рівня або атаки рівня прикладних програм відносяться до типу шкідливої активності, призначеної для націлювання на «верхній» рівень у моделі OSI, де користувач має доступ до служб, таких як, обробка запитів к базі даних, доступ к файлам, пересилка електронної пошти та звичайні може робити інтернет-запити, такі як HTTP GET і HTTP POST. Ці атаки сьомого рівня, на відміну від об'ємних атак рівня, таких як DNS Amplification, є особливо ефективними через споживання ресурсів сервера на додаток до ресурсів мережі.

Основна ефективність більшості атак DDoS полягає в невідповідності між кількістю ресурсів, необхідних для запуску атаки, і кількістю ресурсів, необхідних для її поглинання або пом'якшення. Хоча це все ще стосується атак сьомого рівня, ефективність впливу як на цільовий сервер, так і на мережу вимагає меншої загальної пропускної спроможності для досягнення такого ж руйнівного ефекту; атака прикладного рівня завдає більшої шкоди з меншою загальною пропускною здатністю.

DDoS-атаки прикладного рівня призначені для атаки на конкретні програми, найпоширенішими є веб-сервери, але можуть включати будь-які програми, такі як голосові послуги SIP та BGP.

Причина що робить DDoS-атаки прикладного рівня найбільш небезпечними, так це те, що навіть коли багатовекторні атаки містять ідентифіковані шаблони, зловмисник відслідковуватиме результати своєї атаки та модифікує її, щоб перешкодити захиснику. Оскільки відомо, що активні зловмисники постійно змінюють шаблони корисного навантаження, щоб уникнути спрощеного пом'якшення DDoS-атак, підтримувати поточний

список відомих моделей атак швидко стає недоцільним через проблеми з масштабом і швидкістю, з якою цей список повинен оновлюватися. Крім того, оскільки моделі корисного навантаження мають високий ризик заподіяння побічної шкоди, підтримувати довговічний набір шаблонів корисного навантаження може бути нерозумним.

Проблема при зупиненні чи пом'якшені DDoS-атаки на прикладному рівні в тому що, відрізнити трафік атаки та звичайний трафік важко. Оскільки кожен бот у ботнеті робить, здавалося б, законні запити мережі, трафік не підроблений і може виглядати «звичайним» за походженням.

Атаки прикладного рівня вимагають адаптивної стратегії, включаючи здатність обмежувати трафік на основі певних наборів правил, які можуть регулярно змінюватися. Такі інструменти, як правильно налаштований WAF, можуть зменшити кількість фіктивного трафіку, який передається на вихідний сервер, значно зменшуючи вплив спроби DDoS.

З іншими атаками, такими як SYN-flood або атаками на відображення, такими як посилення NTP, можна використовувати стратегії для досить ефективного зниження трафіку за умови, що сама мережа має пропускну здатність для їх отримання. На жаль, більшість мереж не можуть отримати атаку посилення 300 Гбіт/с, і ще менше мереж можуть правильно маршрутизувати та обслуговувати обсяг запитів прикладного рівня, які може створити атака L7.

Далі приведено деякі методи та техніки DDoS-атак прикладного рівня

- DDoS-атака Slow Loris

Slow Loris — це DDoS-атака прикладного рівня, яка використовує часткові HTTP-запити для відкриття з'єднань між одним комп'ютером і цільовим веб-сервером, а потім залишає ці з'єднання відкритими якомога довше, таким чином перевантажуючи та сповільнюючи ціль. Цей тип DDoS-атаки вимагає мінімальної пропускної спроможності для запуску і впливає лише на цільовий веб-сервер, залишаючи без впливу інші служби та порти.

DDoS-атаки Slowloris можуть бути спрямовані на багато типів програмного забезпечення веб-серверів, але виявилися високоефективними проти Apache 1.x і 2.x.

Ця атака складається з 4 кроків:

1) Зловмисник спочатку відкриває кілька підключень до цільового сервера, надсилаючи кілька часткових заголовків HTTP-запитів.
2) Ціль відкриває потік для кожного вхідного запиту з наміром закрити потік після завершення з'єднання. Щоб бути ефективним, якщо з'єднання займає занадто багато часу, сервер затримує надто довге з'єднання, звільняючи потік для наступного запиту.
3) Щоб запобігти тайм-ауту з'єднань, зловмисник періодично надсилає часткові заголовки запиту цілі, щоб підтримувати запит.
4) Цільовий сервер ніколи не може звільнити жодне з відкритих часткових з'єднань, очікуючи завершення запиту. Як тільки всі доступні потоки будуть використані, сервер не зможе відповідати на додаткові запити, зроблені від звичайного трафіку, що призведе до відмови в обслуговуванні.

Ключем до Slow Loris є його здатність створювати багато проблем із дуже невеликим споживанням пропускної здатності.

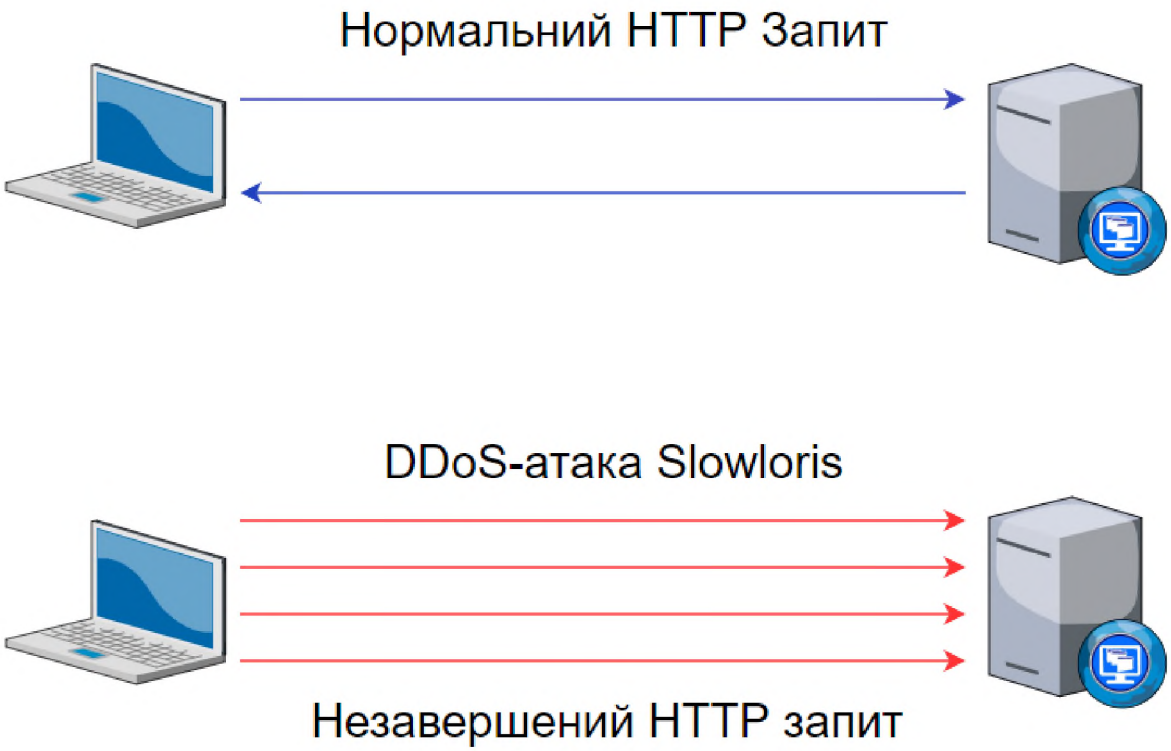

Рисунок 2.1 – Slowloris DDoS-атака

- Slow Post

Під час атаки Slow Post DDoS зловмисник надсилає легітимні заголовки HTTP POST на веб-сервер. У цих заголовках правильно вказано розміри тіла повідомлення. Однак тіло повідомлення надсилається з дуже низькою швидкістю. Ці швидкості можуть становити один байт кожні дві хвилини.

Оскільки повідомлення обробляється нормально, цільовий сервер зробить усе можливе, щоб дотримуватися вказаних правил. Як і під час атаки Slow Loris, сервер згодом сповільниться до сканування. Ще гірше, коли зловмисники одночасно запускають сотні або навіть тисячі атак Slow POST, ресурси сервера швидко споживаються, що робить легітимні з'єднання недосяжними.

- Низька та повільна DDoS-атака

Низька і повільна атака — це тип DDoS-атаки, яка полягає на невеликий потік дуже повільного трафіку, націленого на ресурси програми або сервера. На відміну від більш традиційних атак грубої сили, низькі та повільні атаки вимагають дуже малої пропускної здатності, і з ними важко боротися, оскільки вони створюють трафік, який дуже важко відрізнити від звичайного. У той час як широкомасштабні DDoS-атаки, швидше за все, будуть помічені швидко, низькі та повільні атаки можуть тривалий час залишатися непоміченими, при цьому відмовляючи або сповільнюючи обслуговування реальних користувачів.

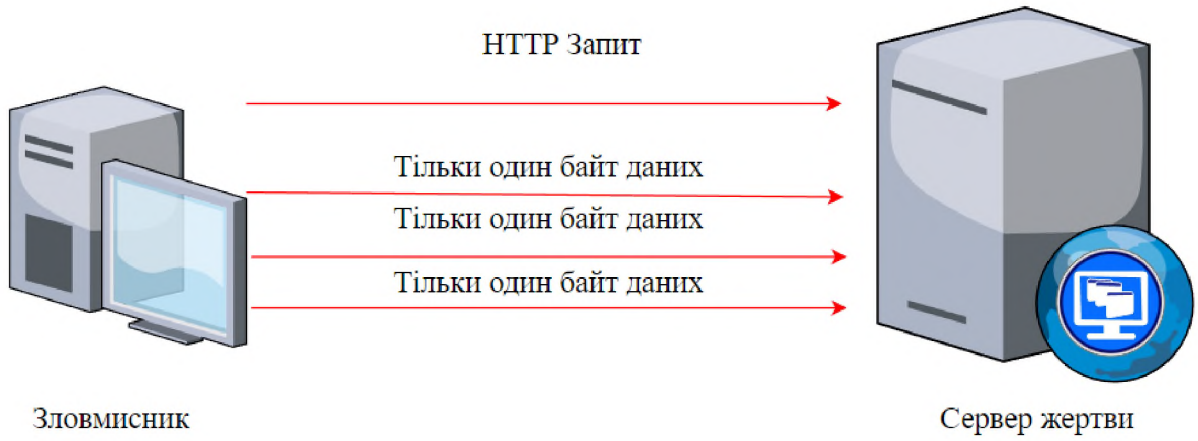

Рисунок 2.2 – Низька та повільна DDoS-атака

Оскільки для їх здійснення не потрібно багато ресурсів, низькі та повільні атаки можна успішно запускати за допомогою одного комп'ютера, на відміну від більш розподілених атак, для яких може знадобитися ботнет. Два з найпопулярніших інструментів для запуску низької та повільної атаки називаються Slow Loris і R.U.D.Y.

Низькі та повільні атаки націлені на веб-сервери на основі потоків з метою зв'язати кожен потік повільними запитами, таким чином не дозволяючи справжнім користувачам отримати доступ до служби. Це досягається шляхом передачі даних дуже повільно, але достатньо швидко, щоб запобігти тайм-ауту сервера. Зловмисники можуть використовувати

HTTP-заголовки, HTTP-запити POST або TCP-трафік для здійснення низьких і повільних атак. Ось 3 поширені приклади атак:

- Інструмент Slow Loris підключається до сервера, а потім повільно надсилає часткові заголовки HTTP. Це змушує сервер залишати з'єднання відкритим, щоб він міг отримати решту заголовків, зв'язуючи потік.
- Інший інструмент під назвою R.U.D.Y. (R-U-DEAD-YET?) генерує HTTP-запити POST для заповнення полів форми. Він повідомляє серверам, скільки даних очікувати, але потім надсилає ці дані дуже повільно. Сервер підтримує з'єднання відкритим, оскільки очікує більше даних.
- Ще одним типом низької та повільної атаки є атака Sockstress, яка використовує вразливість у 3-сторонньому рукостисканні TCP/IP, створюючи невизначене з'єднання.

- Large Payload DDoS-атаки

Це клас HTTP DDoS-атаки, коли зловмисник зловживає кодуванням XML, що використовується веб-серверами. У цьому типі DDoS-атаки веб-серверу надсилається структура даних, закодована в XML, яку потім сервер намагається декодувати, але змушений використовувати надмірну кількість пам'яті, таким чином перевантажуючи систему та аварійну роботу служби.

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Header>
        <oversize>
            Вставити контент великого розміру
        </oversize>
    </soap:Header>
    <soap:Body>
    </soap:Body>
</soap:Envelope>
```

Рисунок 2.3 – XML файл

Ця атака використовує SOAP та DOM.

SOAP – це протокол обміну повідомленнями, заснований на XML, для передачі інформацією між комп'ютерами. SOAP – це додаток специфікації XML.

DOM – це програмний інтерфейс для HTML та XML документів. Ця технологія визначає логічну структуру документа та дає доступ для маніпуляції з елементами документа.

DDoS-атаки з великим навантаженням виникають, коли веб-служби використовують синтаксичний аналізатор DOM для створення представлення в пам'яті повідомлення SOAP. Під час цього процесу розмір повідомлення SOAP може подвоїтися, а в деяких випадках збільшитися до 30 разів. Отримані великі документи призводять до виснаження пам'яті. Варіанти цієї атаки можуть включати вміст надмірного розміру, що міститься в заголовку повідомлення SOAP, в тілі SOAP або в конверті SOAP, але за межами заголовка SOAP і тіла SOAP.

- HTTP Flooding DDoS-атака

Атака HTTP Flood DDoS використовує те, що здається легітимними запитами HTTP GET або POST для атаки на веб-сервер або програму. Ці атаки DDoS-атак часто покладаються на ботнет, який є групою підключених до Інтернету комп'ютерів, які були зловмисно привласнені за допомогою зловмисного програмного забезпечення, такого як троянський кінь.

Ці типи DDoS-атак покликані змусити цільовий сервер або додаток виділити якомога більше ресурсів у прямій відповіді на кожен запит. Таким чином зловмисник сподівається перевантажити сервер або додаток, «заваливши» його якомога більшою кількістю інтенсивних запитів.

Є два варіанта цього метода:

  - Get атака

    У цій формі атаки кілька комп'ютерів або інших пристроїв координуються для надсилання запитів на зображення, файли чи інші активи від цільового сервера. Коли ціль переповнена вхідними

запитами та відповідями, відмова в обслуговуванні буде відбуватися на додаткові запити від законних джерел трафіку.

- Post атака

Зазвичай, коли форма надсилається на веб-сайті, сервер повинен обробляти вхідний запит і передавати дані на рівень збереження, найчастіше в базу даних. Процес обробки даних форми та виконання необхідних команд бази даних є відносно інтенсивним порівняно з обсягом обчислювальної потужності та пропускної здатності, необхідної для відправки запиту POST. Ця атака використовує нерівність у відносному споживанні ресурсів, надсилаючи багато запитів на пошту безпосередньо на цільовий сервер, поки його потужність не буде насичена та не відбудеться відмова в обслуговуванні.

2.2 Аналіз методів захисту

Захиститися від атак сьомого рівня мережевої моделі OSI складною задачею. Один із методів полягає в тому, щоб реалізувати перевірку машини, яка робить запит, щоб перевірити, чи є це ботом, наприклад, як тест Captcha, який зазвичай зустрічається під час створення облікового запису в Інтернеті. Використовуючи такі вимоги, як обчислювальна проблема JavaScript, можна захиститися від багато атак.

На відміну від атак мережевого рівня, атаки прикладного рівня не можна пом’якшити лише потужністю вашої мережі. Замість цього компанії зазвичай покладаються на брандмауери веб-додатків (WAF), ручну фільтрацію IP-адреси та спеціальний аналіз мережі. Проблема з цими підходами двояка:

- Тепер зловмисники можуть легко розповсюджувати ботів за допомогою сотень тисяч різних IP-адрес, що робить фільтрацію на основі IP значною мірою неефективною.
- Фільтрація вручну займає дуже багато ресурсів і, як правило, занадто повільна, щоб ефективно пом'якшувати великі атаки.

Найефективніший спосіб захистити свої програми від DDoS-атак прикладного рівня – це точно профільтрувати вхідний трафік. Це дозволить вам відрізнити ботів від людей і блокувати будь-який небажаний або підозрілий трафік, не порушуючи роботу користувачів для цільової аудиторії.

Далі приведені методи та техніки з захисту веб-сервісів від DDoS-атак на прикладному рівні, методи які можуть допомогти знизити ризик DDoS-атак.

- IDS та IPS

IDS – аналізуйте та відстежуйте мережевий трафік на наявність ознак того, що зловмисники використовують відому кіберзагрозу для проникнення або крадіжки даних з вашої мережі. Системи IDS порівнюють поточну мережеву активність з відомою базою даних загроз, щоб виявити кілька видів поведінки, як порушення політики безпеки, зловмисне програмне забезпечення та сканери портів.

На рис. 2.4 зображена типова схема мережі яка в себе включає IDS:

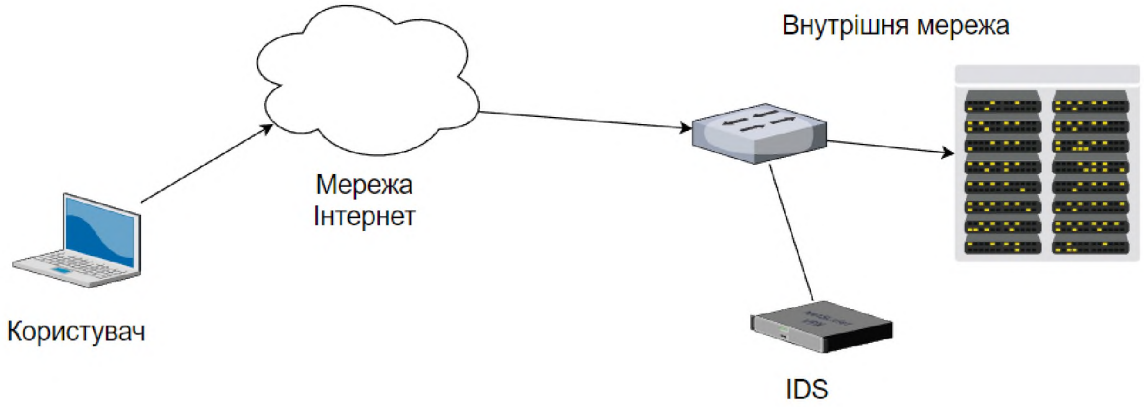

Рисунок 2.4 – схема мережі з IDS

IPS – встановлюються в тій же зоні мережі, що і брандмауер, між зовнішнім світом і внутрішньою мережею. IPS забороняє мережевий трафік на основі профілю безпеки, якщо цей пакет представляє відому загрозу безпеці.

На рис. 2.5 зображена типова схема мережі яка в себе включає IPS:

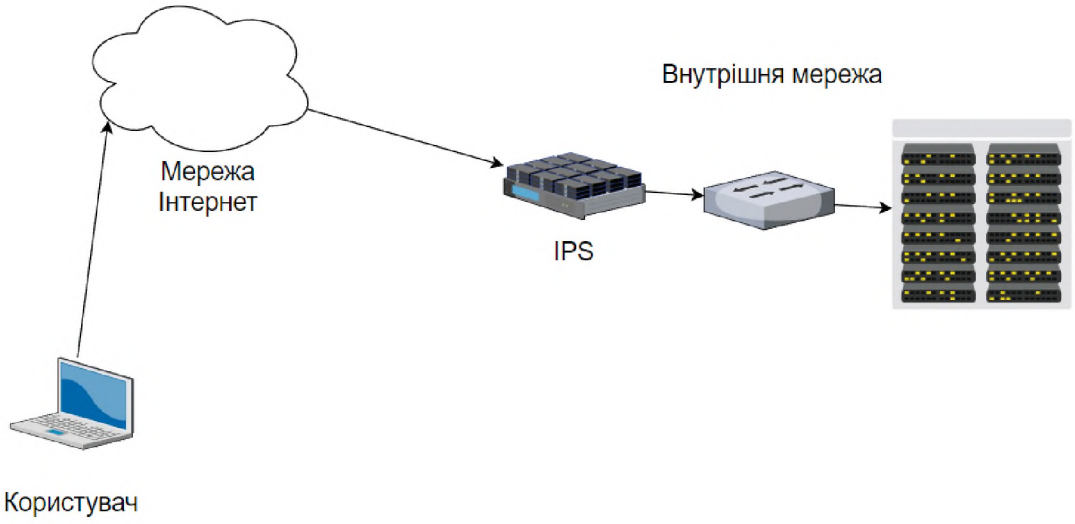

Рисунок 2.5 – схема мережі з IPS

IDS жодним чином не змінює мережеві пакети, а тільки спостерігає і сповіщає про можливі загрози, тоді як IPS запобігає доставці пакету на основі вмісту пакета, це є їх відмінність.

Різницю між IPS чи IDS і будь-яким типом брандмауера дуже легко помітити. Брандмауер захищає межі системи, тоді як IPS та IDS відстежує трафік всередині мережі.

- Брандмауер

Брандмауер – це інструмент, який відстежує, фільтрує та контролює трафік, що входить або виходить із вашої мережі. Його робота полягає в тому, щоб дозволити довіреному трафіку протікати і не дозволяти недовіреному трафіку отримати доступ або залишити вашу внутрішню мережу.

На рис. 2.6 зображена типова схема мережі яка в себе включає брандмауер:

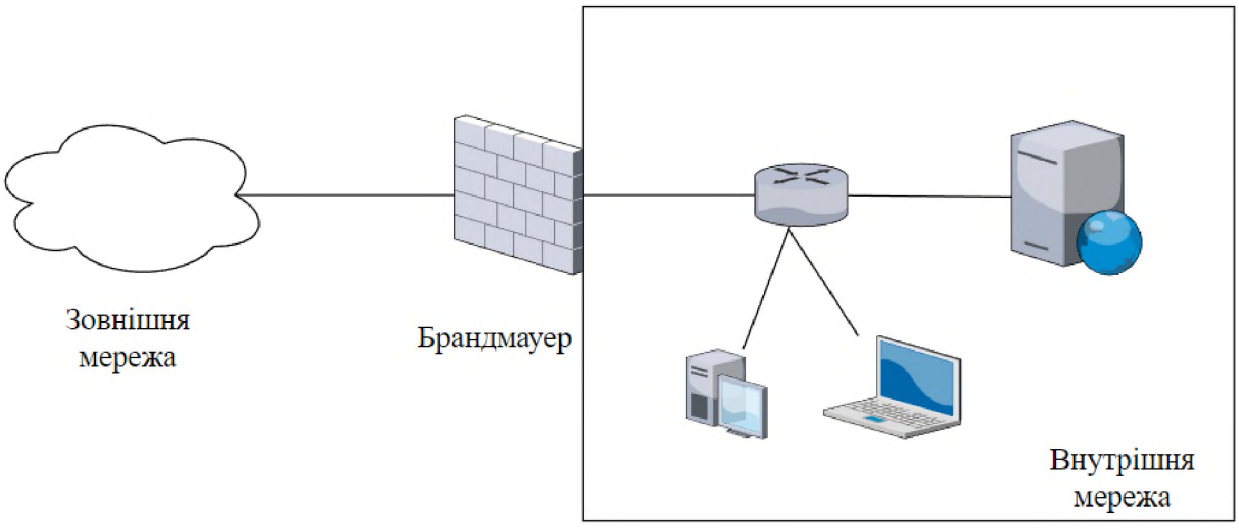

Рисунок 2.6 – Схема мережі з брандмауером

Багато антивірусних програм мають брандмауери, операційні системи зазвичай мають вбудовані брандмауери як частину заходів безпеки

інформації. Брандмауери можуть працювати на ваших маршрутизаторах або серверах. Вони також можуть бути повністю окремою комп'ютерною системою. В результаті існує багато варіантів, які відповідають вашим потребам і бюджету.

Визначення апаратного брандмауера полягає в тому, що це окремий пристрій, який відстежує, фільтрує та контролює трафік, що входить або виходить із вашої мережі. Апаратний брандмауер можна стратегічно розмістити в мережі, щоб оптимізувати його функціональність. Цей тип обладнання часто поєднується з маршрутизатором, щоб почати з кореня потоку введення/виведення. Це відрізняється від програмного брандмауера, який ви встановлюєте на наявних кінцевих пристроях, серверах, маршрутизаторах тощо, щоб регулювати мережевий трафік для цього пристрою. У більшості випадків використання обох є відмінним варіантом, оскільки вони насправді роблять речі по-різному.

Також брандмауери діляться на два типи:

- З збереженням стану – цей тип брандмауера використовує всю інформацію з підключення; замість того, щоб перевіряти окремий пакет, цей брандмауер визначає поведінку пристрою на основі всього з'єднання. Цей тип брандмауера споживає багато ресурсів у порівнянні з брандмауерами без збереження стану, оскільки прийняття рішень є динамічним. Наприклад, брандмауер може дозволити перші частини рукостискання TCP, які згодом вийдуть з ладу. Якщо з'єднання з хоста погане, воно заблокує весь пристрій.
- Без збереження стану – цей тип брандмауера використовує статичний набір правил, щоб визначити, чи легітимні окремі пакети чи ні. Наприклад, пристрій, який надсилає поганий пакет, не обов'язково означає, що весь пристрій буде заблоковано. Хоча ці брандмауери використовують набагато менше ресурсів, ніж

альтернативні, вони більш примітивні. Наприклад, ці брандмауери діють точно згідно правилу, визначеному в них. Якщо правило не точно відповідає, воно фактично марне. Однак ці брандмауери чудово підходять під час отримання великої кількості трафіку від набору хостів, наприклад при DDoS-атаках.

Брандмауери цього виду працюють з IP-адресами та мережевими портами, тобто на третьому та четвертому рівні моделі OSI.

- WAF

WAF захищає веб додатки відстежуючи, фільтруючи та блокуючи будь-який зловмисний HTTP та HTTPS трафік, що надходить до веб додатків, і запобігає виходу будь-яких неавторизованих даних із програми. Слід вказати, що гарною практикою перед WAF встановлювати звичайний Firewall.

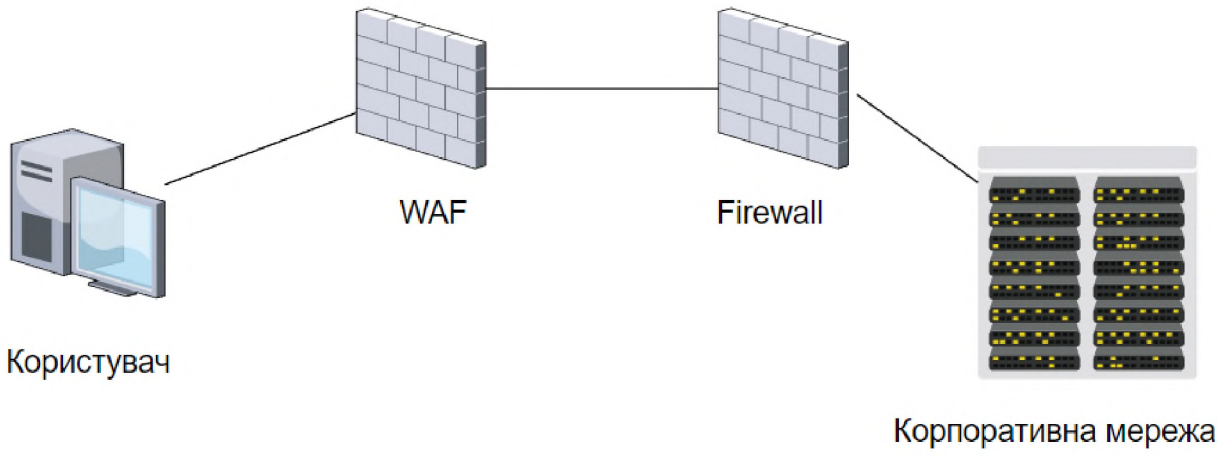

Рисунок 2.7 – Схема мережі з WAF

Це робиться, дотримуючись набору правил, які допомагають визначити, який трафік є шкідливим, а який безпечним.

У таблиці 2.1 наведено основні види правил які використовуються в WAF:

Таблиця 2.1 – Основні правила які використовується в WAF

Назва	Опис
Географічне порівняння	Перевіряє країну походження запиту
Збіг по IP	Перевіряє запит щодо набору IP-адрес і діапазонів адрес
Відповідність міток	Перевіряє запит на мітки, додані іншими правилами в той самий веб-список керування доступом
Оператор правила відповідності регулярного виразу	Порівнює шаблон регулярного виразу з зазначеним компонентом запиту
Набір шаблонів регулярного виразу	Порівнює шаблони регулярних виразів із зазначеним компонентом запиту
Обмеження розміру	Перевіряє обмеження розміру щодо зазначеного компонента запиту
Збіг рядків	Порівнює рядок із зазначеним компонентом запиту

Також, правила можна комбінувати та використовувати логічні оператори: AND, OR, NOT. Таким чином можна вкладати правила чи групу правил у нове правило.

WAF підтримує оператори на основі ставок і груп правил. Ви не можете вкладати ці типи правил в інші правила. Для деяких із цих операторів ви можете звузити сферу запитів, які вони перевіряють, додавши оператор зменшення.

Таблиця 2.2 – Основні види груп правил

Назва	Опис
Керована група правил	Виконує правила, визначені у вказаній групі керованих правил. Ви можете звузити сферу запитів, які ви оцінюєте за допомогою групи правил, додавши оператор зменшення. Ви не можете вкладати оператор групи керованих правил всередині будь-якого іншого типу оператора.
Група правил	Запускає правила, визначені в групі правил, якою ви керуєте. Ви не можете вкладати оператор групи правил всередині будь-якого іншого типу оператора.
На основі ставок	Відстежує швидкість запитів з окремих IP-адрес і тимчасово блокує адреси, поки вони

	надсилають занадто багато запитів. Ви можете звузити сферу запитів, які ви оцінюєте за допомогою оператора на основі ставок, додавши оператор зменшення обсягу. Ви не можете вкладати оператор на основі ставок під жодним оператором правила. Ви можете визначити оператор на основі ставок у групі правил, якою ви керуєте.

WAF стоїть перед усіма пристроями корпоративної мережі, тому він має бути метою вашої URL-адреси. Це означає, що ви більше не маєте прямого контролю над трафіком, оскільки всі записи DNS спочатку спрямовують відвідувачів веб-сайту до хмарної інфраструктури.

Подібно до того, як проксі-сервер діє як посередник для захисту особистості клієнта, WAF працює подібним чином, але навпаки – так званий зворотний проксі-сервер – діє як посередник, який захищає сервер веб-додатка від потенційно шкідливого клієнта. Ця технологія може захищає веб від наступних вразливостей:

- DDoS
- SQL Ін'єкції – це вразливість, завдяки якої зловмисник надає запит у вигляді мови структурованих запитів (SQL) через веб-форму безпосередньо веб-додатку, щоб отримати доступ до серверної бази даних та даних програми. Це може призвести до ненавмисної та шкідливої поведінки цільової програми. Зазвичай цей тип атаки є успішним через відсутність у веб-додатку перевірки введених даних користувача, що дозволяє користувачам надавати код програми SQL у формах HTML замість звичайних запитів користувача, наприклад.

- Порушена аутентифікація – вразливість у механізмі аутентифікації веб-додатка. Завдяки цьому зловмисник може дізнатися ім’я користувачів веб-ресурса або отримати несанкціонований доступ до облікового запису тим самим получивши доступ до даних та подальший вектор атаки.
- Видалення конфіденційних даних – відбувається, коли організація несвідомо розкриває конфіденційні дані або коли інцидент безпеки призводить до випадкового чи незаконного знищення, втрати, зміни або несанкціонованого розкриття чи доступу до конфіденційних даних. Таке розкриття даних може статися в результаті неналежного захисту бази даних, неправильної конфігурації при створенні нових екземплярів сховищ даних, неналежного використання систем даних тощо.
- XXE – це класифікація атак, яку часто дуже просто виконати, але з руйнівними результатами. Ця класифікація атак спирається на неправильно налаштований синтаксичний аналізатор XML у коді програми.
- Порушений контроль доступу – виникають, коли параметри безпеки не визначені, не реалізовані, а значення за замовчуванням підтримуються. Зазвичай це означає, що параметри конфігурації не відповідають галузевим стандартам безпеки, які є критичними для підтримки безпеки та зниження бізнес-ризиків. Неправильна конфігурація зазвичай відбувається, коли адміністратор системи або бази даних або розробник не налаштовує належним чином структуру безпеки програми, веб-сайту, робочого столу або сервера, що призводить до небезпечних відкритих шляхів для хакерів.
- XSS – це атака на веб-додаток, яка використовується для отримання доступу до приватної інформації клієнта шляхом доставки шкідливого коду, зазвичай JavaScript, кінцевим

користувачам через поля вводу веб-додатка. Як правило, цей тип атаки є успішним через відсутність у веб-додатку валідації користувацьких введених даних, що дозволяє зловмиснику надавати код програми у формах HTML замість звичайних текстових рядків.

- Небезпечна десеріалізація – це коли дані, які контролюються користувачем, десеріалізуються веб-сайтом. Це потенційно дає змогу зловмиснику маніпулювати серіалізованими об'єктами, щоб передати шкідливі дані в код програми. Можна навіть замінити серіалізований об'єкт на об'єкт зовсім іншого класу. Насторожує те, що об'єкти будь-якого класу, доступного для веб-сайту, будуть десеріалізовані та створені, незалежно від того, який клас очікувався. З цієї причини небезпечну десеріалізацію іноді називають уразливістю «ін'єкції об'єктів».

WAF можуть поставлятися у вигляді пристрою або надаватися як послуга. Політику можна налаштувати відповідно до унікальних потреб вашого веб-додатка або набору веб-програм. Хоча багато WAF вимагають регулярного оновлення політики для усунення нових вразливостей, прогрес у машинному навчанні дозволяє деяким WAF оновлюватися автоматично. Ця автоматизація стає все більш важливою, оскільки ландшафт загроз продовжує зростати у складності та неоднозначності.

- NGFW

Брандмауер наступного покоління – це брандмауер третього покоління, який включає в себе функціонал традиційного firewall та других мережевих пристроїв для фільтрації. Також може фільтрувати трафік на основі програм або типів трафіку, що проходить через ці порти. Наприклад, ви можете відкрити порт 80 лише для вибраного HTTP-трафіку для тих конкретних програм, сайтів або служб, які ви дозволяєте. Можна думати про це як

поєднання брандмауера та функцій якості обслуговування (QoS) в одне рішення.

Далі приведені загальні риси більшості NGFW:

- Стандартні функції брандмауера: вони включають традиційні функції брандмауера, такі як перевірка портів чи протоколів із заповненням стану, NAT і VPN.
- Ідентифікація та фільтрація додатків: це головна характеристика NGFW. Вони можуть ідентифікувати та фільтрувати трафік на основі конкретних програм, а не просто відкривати порти для будь-якого трафіку. Це запобігає використання шкідливих програм і діяльності нестандартних портів, щоб уникнути брандмауера.
- Перевірка SSL та SSH: NGFW можуть навіть перевіряти зашифрований трафік SSL та SSH. Вони можуть розшифрувати трафік, переконатися, що це дозволена програма, а також перевірити інші політики, а потім повторно зашифрувати його. Це забезпечує додатковий захист від шкідливих програм і діяльності, які намагаються приховати за допомогою шифрування, щоб уникнути брандмауера.
- Запобігання вторгненням: будучи більш розумними та з більш глибокою інспекцією трафіку, вони також можуть виконувати виявлення та запобігання вторгненню. Деякі брандмауери наступного покоління можуть мати достатню функціональність IPS, так що автономний IPS може не знадобитися.
- Інтеграція каталогу: більшість NGFW включають підтримку каталогів, тобто Active Directory. Наприклад, для керування авторизованими програмами на основі користувачів і груп користувачів.

- Фільтрація шкідливого програмного забезпечення: NGFW також може забезпечити фільтрацію на основі репутації, щоб блокувати програми, які мають погану репутацію. Це, можливо, може перевірити фішингові, шкідливі та інші сайти та програми зловмисного програмного забезпечення.

- Машинне навчання

Сьогодні технологія машинного навчання все більш використовується в різних галузях, включаючи інформаційну безпеку. Ця технологія дозволяє автоматично генерувати правила, за якими аналізується мережевий трафік, сигнатури програм. Використовується в антивірусах, системах моніторингу, WAF, NGFW.

На наступних рисунках [6] приведено приклад реагування на HTTP Flood DDoS-атаку:

На рис. 2.8 видно як , виявлено 17700 запитів на секунду (RPS), що представляло аномальне збільшення порівняно зі звичайними обсягами трафіку для цієї кінцевої точки. Проаналізувавши запити, виділяються дві сигнатури регіональний код та User-Agent. Це приведено на рис. 2.9 та 2.10. На рис. 2.11 продемонстровано правило, яке було створено в результаті роботи технології машинного навчання.

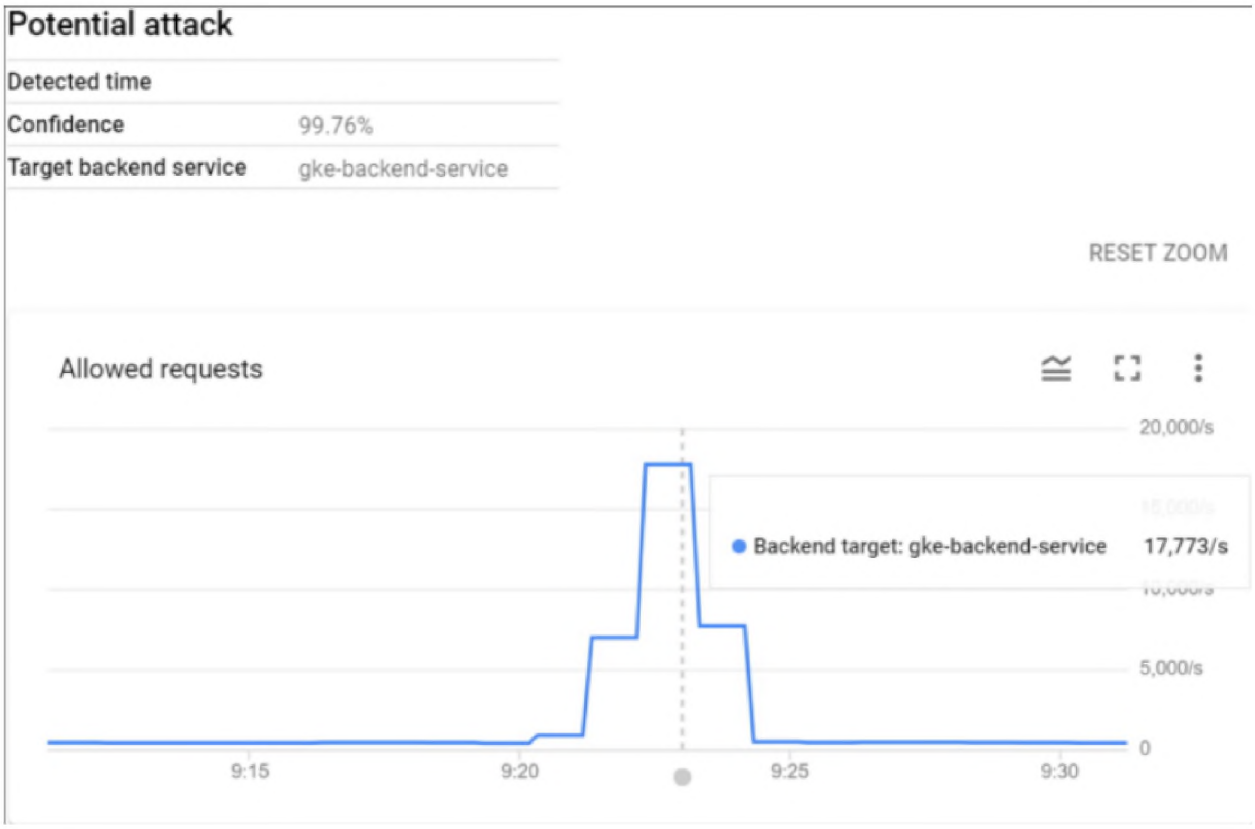

Рисунок 2.8 – Потенційна атака

Value	Attack likelihood	Attack	Baseline
ID	100%	10.58%	≤ 1%
CN	100%	9.61%	≤ 1%
AT	100%	9.16%	≤ 1%
RU	100%	8.6%	1.05%
TH	100%	5.85%	≤ 1%
BR	99.09%	5.54%	1.14%

Рисунок 2.9 – Регіональний код

Value	Attack likelihood	Attack	Baseline
Mozilla/5.0 (Linux; Android 7.0; SM-G930V Build/NRD90M) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.125 Mobile Safari/537.36	100%	18.87%	≤ 1%
Mozilla/5.0 (Linux; Android 8.0.0; SM-G960F Build/R16NW) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/62.0.3202.84 Mobile Safari/537.36	100%	12.67%	≤ 1%
Mozilla/5.0 (Linux; Android 7.1.2; DSCS9 Build/NHG47L; wv) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/80.0.3987.149 Safari/537.36	100%	11.9%	≤ 1%
Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.75 Safari/537.36	100%	11.85%	≤ 1%
Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0	100%	9.77%	≤ 1%
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.75 Safari/537.36	100%	9.36%	≤ 1%
Mozilla/5.0 (iPhone; CPU iPhone OS 14_0_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0 Mobile/15E148 Safari/604.1	100%	8.85%	≤ 1%
Mozilla/5.0 (iPhone; CPU iPhone OS 10_3_1 like Mac OS X) AppleWebKit/603.1.30 (KHTML, like Gecko) Version/10.0 Mobile/14E304 Safari/602.1	100%	8.75%	≤ 1%
Mozilla/5.0 (Linux; Android 7.0; SM-A310F Build/NRD90M) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/55.0.2883.91 Mobile Safari/537.36 OPR/42.7.2246.114996	100%	7.99%	≤ 1%

Рисунок 2.10 – User-Agent

Policy	allow-all
Recommended priority	2147483646
Action	Deny (403): preview only
Backend services using the policy	gke-8080-backend-service, gke-backend-service, **4 MORE**
Impacted attack's traffic	100%
Impacted baseline's traffic	0%

Рисунок 2.11 – Результат роботи технології машинного навчання

2.3 Рекомендації з захисту типового веб-сервісу від DDoS-атак на прикладному рівні

Були розглянуті методи які допомагають пом'якшити DDoS-атаку. Кожна з цих технік має свої позитивні та негативні сторони.

IDS діє у середині мережі, аналізуючи та повідомляючи про підозрілу активність. IPS також діє у середині мережі, але може втручатися в трафік та приймати рішення самостійно, згідно з правилами. IDS та IPS це потужні пристрої але вони працюють на рівні пакетів та не підходять для аналізу даних на прикладному рівні.

Брандмауер – девайс або програмне забезпечення, використовується для того щоб зупинити або знизити неавторизований доступ до приватної мережі. Працює завдяки політикам, наприклад блокування по IP-адресі або мережевому порту, любий другий трафік, який буде пробувати получити доступ, буде блокуватися. Девайс встановлюють перед мережею, між зовнішніми та внутрішніми девайсами. Можна зробити висновок що брандмауер працює на третьому та четвертому рівні мережевої моделі OSI.

WAF захищає веб-сервери від веб-атак включаючи DDoS-атаки прикладного рівня, тому що може аналізувати дані сьомого рівня моделі OSI, включаючи HTTP заголовки. WAF включає в себе техніки IDS та IPS, а також може використовувати машинне навчання для більш швидкого реагування та створення правил.

NGFW включають типові функції традиційних брандмауерів, такі як фільтрація пакетів, NAT, перевірка стану та VPN. Мета брандмауерів наступного покоління полягає в тому, щоб включити більше шарів моделі OSI, покращуючи фільтрацію мережевого трафіку, який залежить від вмісту пакета.

Для того щоб надати рекомендації з захисту типового веб-сервісу від DDoS-атак на прикладному рівні створимо мережу з наступними характеристиками:

Веб-сервер:

OS: Linux Fedora Server, 64-bit

ПЗ: Apache2.4.18, MySQL 5, PHP 7

CPU: 8-core Intel Xeon Silver 4215 (2.5 - 3.5 ГГц)

RAM: 32 Gb

HDD: 2 x 2 Tb

Ports: 2 x USB 3.0, 2 x RJ-45 GbE LAN

LAN Speed: 1 Gb/s

Маршрутизатор:

OS: FreeBSD 6.4 m0n0wall

Інтерфейси:
5 x Ethernet 10/100 Мбит/с
5 x Ethernet 1000 Мбит/с
1 x SFP
1 x Серийный порт RJ45
1 x microUSB type AB
WAN-порт: Ethernet

На рис. 2.12 зображено схему мережі:

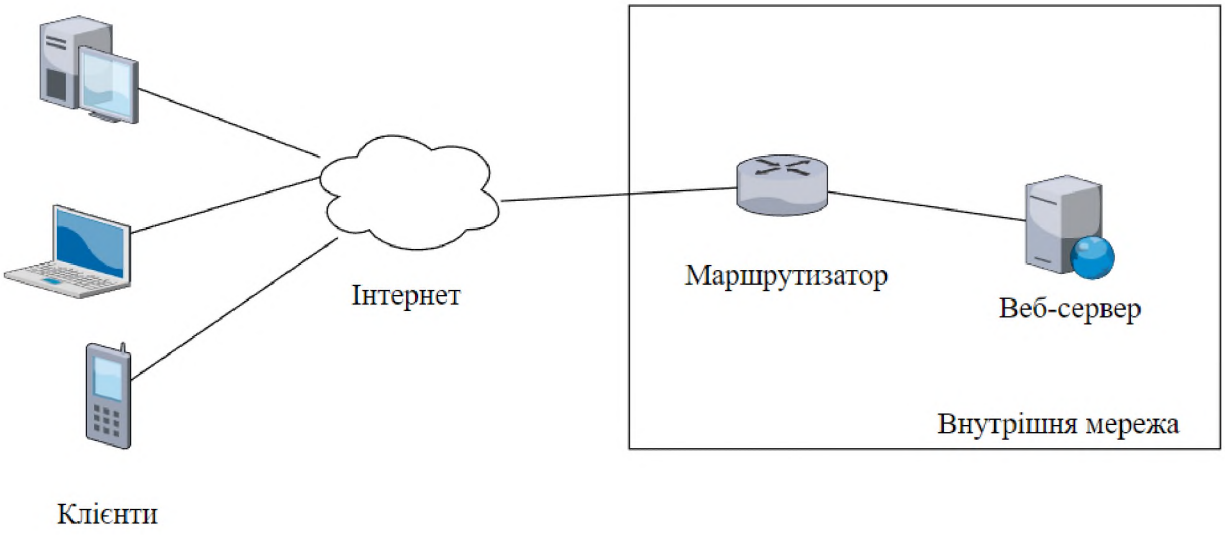

Рисунок 2.12 – Схема мережі

Проаналізувавши технології для захисту веб-сервісу від DDoS-атак на прикладному рівні, та враховуючи приведено мережу, обираємо WAF по наступним причинам:

- Може працювати на прикладному рівні моделі OSI
- Має вбудований функціонал IDS та IPS
- Має технологію машинного навчання
- WAF більш дешевий ніж NGWF (середня ціна за WAF 3000 доларів, коли NGWF коштує 140000 доларів), та для вирішення нашої задачі не потрібен такий великий функціонал який є у NGWF.

Далі наведено рекомендації з захисту від DDoS-атак на прикладному рівні:

1) Проводити фільтрацію поганих запитів, наприклад від ботів
    - Встановлення CAPTCHA
2) Встановити параметри для комунікації з веб-сервером
    - Регулярне оновлення та перевірки встановленого програмного забезпечення
    - Відключення ресурсів та програмного забезпечення яке не використовується
    - Збільшення максимальної кількості клієнтів, яких може прийняти веб-сервер
    - Встановлення обмеження на мінімальну швидкість передачі даних під час підключення
    - Обмеження кількості часу, протягом якого клієнту дозволяється залишатися на зв'язку
    - Збільшення об'єму кеша веб-серверу. Вже перевірені та дозволені адреси не обробляються
3) Порівняння параметрів веб запитів з правилами в WAF

- Встановлення ліміта на максимальний розмір документа який надсилається на веб-сайт
- Створення бази даних репутації IP для відстеження та блокування ненормальної активності
- Обмеження кількості з'єднань, які можуть бути встановлені з однієї IP-адреси
- Встановлення мінімальної швидкості вхідних даних, після чого відключення будь-якого з'єднання, яке повільніше за цю швидкість

4) Використання технології машинного навчання

На рис. 2.13 приведена схема мережі зі встановленим WAF.

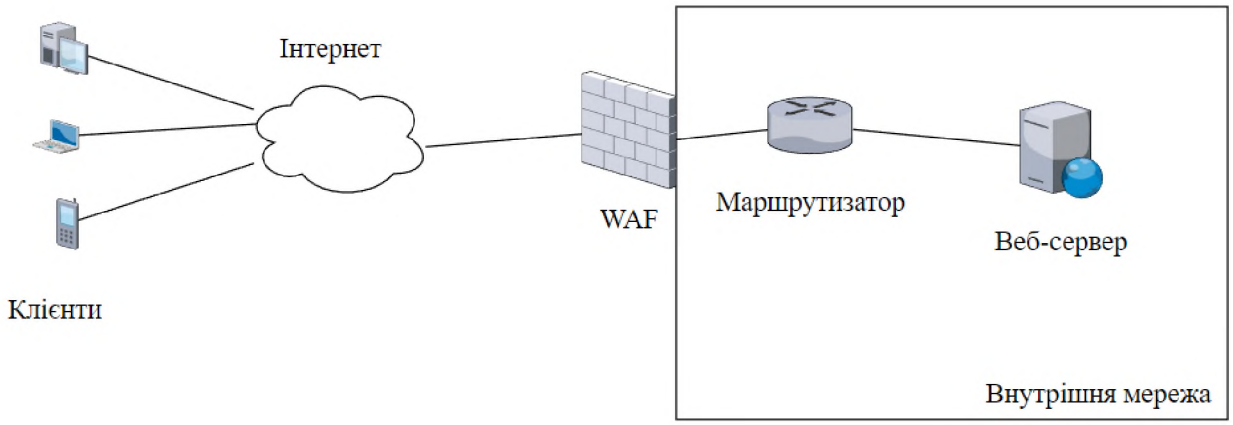

Рисунок 2.13 – Схема мережі зі встановленим WAF

Як можна бачити, WAF встановлюється на границі з внутрішньої мережею та зовнішньою, для аналізу всього вхідного трафіку.

На рис. 2.14 приведені рекомендовані рівні захисту через які проходить кожен запит:

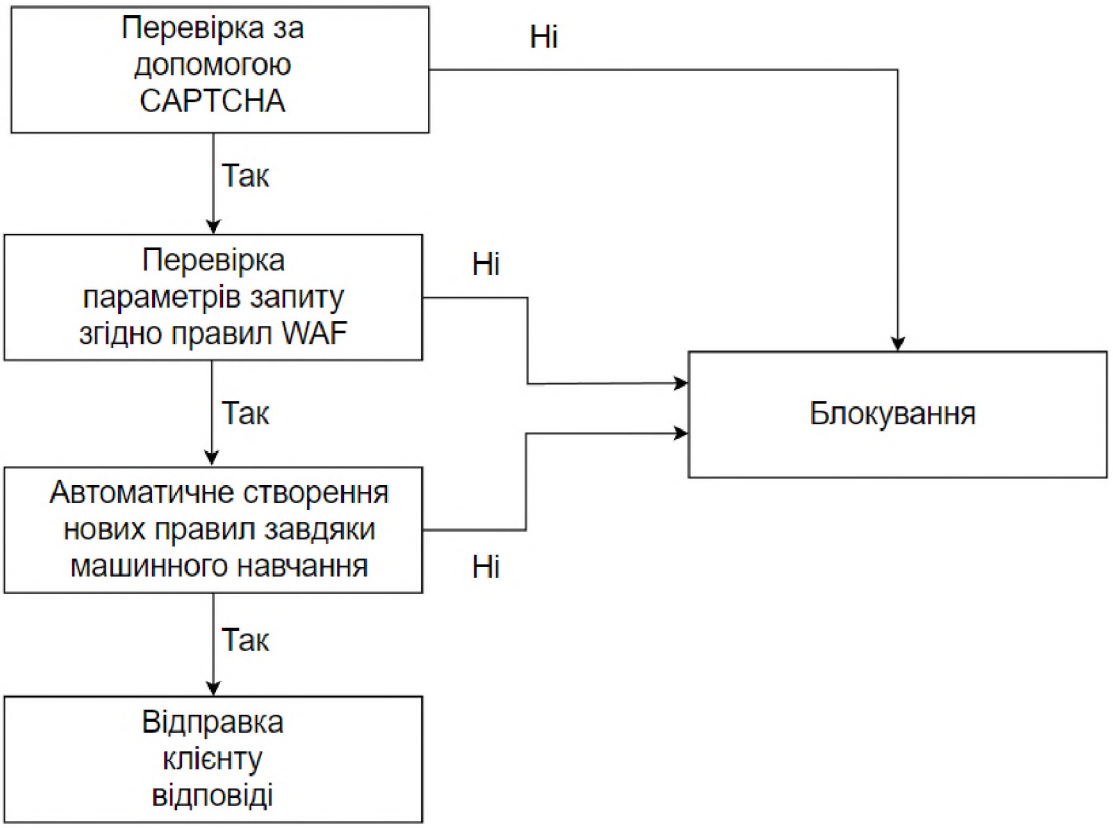

Рисунок 2.14 – Рівні захисту через які проходить кожен запит

2.4Висновки

В спеціальній частині було проаналізовано DDoS-атаки прикладного рівня. Було приведено методи захисту, різниця між ними та обрана технологія для забезпечення стійкості мережі проти DDoS-атак прикладного рівня. Було розроблено рекомендації з захисту веб-сервісу від DDoS-атак на прикладному рівні для типового об’єкту.

ЕКОНОМІЧНИЙ РОЗДІЛ

Розробка системи захисту веб-сервісу від DDoS-атак прикладного рівня на базі WAF потребує економічне обґрунтування її доцільності, виходячи з аналізу витрат на розробку та впровадження. Тому метою економічного розділу є здійснення відповідних розрахунків, які дозволять встановити економічного ефекту від впровадження та налагодження системи захисту від DDoS-атак.

3.1 Розрахунок капітальних (фіксованих) витрат

Капітальні інвестиції – це кошти, призначені для створення і придбання основних фондів і нематеріальних активів, що підлягають амортизації.

До капітальних витрат належать витрати на розробку системи захисту, написання правил і конфігураційних файлів та її запровадження.

3.1.1 Визначення трудомісткості розробки рекомендацій з захисту від DDoS-атак на прикладному рівні

Трудомісткість розробки рекомендацій з захисту від DDoS-атак на прикладному рівні визначається тривалістю кожної робочої операції, починаючи з складання технічного завдання і закінчуючи оформленням документації (за умови роботи одного спеціаліста з інформаційної безпеки):

$$t = t_{тз} + t_{в} + t_{а} + t_{вз} + t_{озб} + t_{овр} + t_{д}, \text{ год.}, \qquad (3.1)$$

де:

$t_{тз}$ - тривалість складання технічного завдання на розробку;

$t_{в}$ - тривалість розробки концепції безпеки інформації у організації;

$t_{а}$ - тривалість процесу аналізу ризиків;

$t_{вз}$ - тривалість визначення вимог до заходів, методів та засобів захисту;

$t_{озб}$ - тривалість вибору основних рішень з забезпечення безпеки інформації;

$t_{овр}$ – тривалість організації виконання відновлюваних робіт і забезпечення неперервного функціонування організації;

$t_{д}$ - тривалість документального оформлення політики безпеки.

Визначено, що відповідно до етапів розробки політики безпеки інформації, тривалість операції склала наступні величини:

$t_{тз}$ = 20 годин, $t_{в}$ = 35 годин, $t_{тз}$ = 20 годин, $t_{вз}$ = 18 годин, $t_{озб}$ = 10 годин, $t_{овр}$ = 8 годин, $t_{д}$ = 6 годин.

Згідно формули 3.1, трудомісткість розробки політики безпеки інформації дорівнює:

$$t = 20 + 35 + 20 + 18 + 10 + 8 + 6 = 117 \text{ годин.}$$

3.1.2 Розрахунок витрат на створення системи захисту від DDoS-атак

$$К_{рп} = З_{зп} + З_{мч} \qquad (3.2)$$

Витрати на розробку системи захисту $К_{рп}$ складається з:

$З_{зп}$ – витрати на заробітну плату спеціаліста з інформаційної безпеки;

$З_{мч}$ – вартість витрат машинного часу, що необхідні для розробки системи захисту.

Витрати на заробітну плату спеціаліста ІБ розраховуються за формулою:

$$З_{зп} = t * З_{іб} \text{ грн.,} \qquad (3.3)$$

де:

t – загальна тривалість розробки, годин;

$З_{іб}$ – средньогодинна заробітна плата спеціаліста з інформаційної безпеки з нарахуваннями, грн/годину.

Средньогодинна заробітна плата спеціаліста з інформаційної безпеки становить – 150 грн/ годину.

Заробітна плата виконавця враховує основну і додаткову заробітну плату, а також відрахування на соціальні потреби та визначається за формулою 3.3 :

$$З_{зп} = 117 * 150 = 17550 \text{ грн.}$$

Вартість машинного часу для розробки системи захисту від DDoS-атак визначається за формулою:

$$З_{мч} = t * С_{мч} \qquad (3.4)$$

де:

t – трудомісткість підготовки документів, годин;

$С_{мч}$ – вартість 1 години машинного часу ПК, грн./година.

Вартість 1 години машинного часу ПК визначається за формулою:

$$Смч = P \cdot t_{нал} \cdot C_e + \frac{Ф_{зал} \cdot Н_a}{F_p} + \frac{К_{лпз} \cdot Н_{апз}}{F_p}, \text{ грн,} \qquad (3.5)$$

де:

P – встановлена потужність ПЕС, кВт;

$t_{нал}$ – кількість задіяних робочих станцій при створенні системи захисту;

$С_е$ – тариф на електроенергію, грн./кВт*година;

$Ф_{зал}$ – залишкова вартість ПК на поточний рік, грн.;

$Н_а$ – річна норма амортизації на ПК, частки одиниці;

$Н_{апз}$ – річна норма амортизації на ліцензійне програмне забезпечення, частки одиниці;

$К_{лпз}$ – вартість ліцензійного програмного забезпечення, грн;

$F_р$ – річний фонд робочого часу (за 40-годинного робочого тижня $F_р$=1920).

$$С_{мч}=0,7*2*1,51+((6879*0,3)/1920)+((1958*0,1)/1920)=3,29грн.$$

$$З_{мч} = 117 * 3,29 = 384,93 \text{ грн}$$

$$К_{рп}= 17550 + 384,93 =17934,93 \text{ грн.}$$

Капітальні (фіксовані) витрати на створення системи захисту WAF від DDoS-атак:

$$К = К_{рп} + К_{аз} + К_{зпз} + К_{навч} \quad (3.6)$$

де:

$К_{рп}$ – витрат на створення системи захисту від DDoS-атак

$К_{аз}$ – вартість закупівлі апаратного забезпечення та допоміжних матеріалів

$К_{зпз}$ – вартість закупівлі ліцензійного програмного забезпечення

$К_{навч}$ – витрати на навчання технічних фахівців в обслуговуючого персоналу

$$К = 17934,93 + 90000 + 3000 + 3000 = 113934,93$$

3.2Розрахунок поточних (експлуатаційних) витрат

Експлуатаційні витрати – поточні витрати на обслуговування об’єкта проектування за визначений період.

Річні поточні витрати на функціонування системи інформаційної безпеки:

$$С=С_{в} + С_{к} + С_{ак}, \text{ грн.} \quad (3.7)$$

де:

$С_{в}$ – вартість відновлення й модернізації системи ($С_{в} = 0$);

$С_к$ – витрати на керування та функціонал системи;

$С_{ак}$ – витрати, викликані активністю користувачів ($С_{ак} = 0$).

Витрати на керування системи ($С_к$) складають:

$$С_к = С_н + С_а + С_з + С_{єв} + С_{ел} + С_{тос}, \text{ грн.} \quad (3.8)$$

де:

$С_н$ – це витрати на навчання адміністративного персоналу й кінцевих користувачів визначаються за даними організації з проведення тренінгів персоналу, курсів підвищення кваліфікації – 3000 грн

$С_а$ – річний фонд амортизаційних відрахувань

$$С_а = С_{а1} + С_{а2}, \text{ грн} \quad (3.9)$$

де:

$С_{а1}$ - це річний фонд амортизаційних відрахувань ПЗ (програмного забезпечення)

$С_{а2}$ - це річний фонд амортизаційних відрахувань АЗ (апаратного забезпечення)

$$С_{а1/2} = \frac{Ф_п}{Т}, \text{ грн} \quad (3.10)$$

де:

$Ф_п$ – первісна вартість придбаного ПЗ/АЗ

Т – мінімальний срок корисного використання (2 роки для ПЗ, 5 - АЗ)

$$С_{а1} = 3000/2=1500 \text{ грн};$$

$$С_{а2} = 30000/5=15000 \text{ грн};$$

$$С_а = 1500+15000 = 16500 \text{ грн.}$$

$С_з$ – це річний фонд заробітної плати інженерно-технічного персоналу, котрий обслуговує систему ІБ, вираховується за формулою:

$$С_з = З_{осн} + З_{дод}, \text{ грн} \quad (3.11)$$

Де основна заробітна плата ($З_{осн}$) визначається, виходячи з місячного посадового окладу, а додаткова заробітна плата ($З_{дод}$) в розмірі 8-10% від основної заробітної плати. Основна заробітна плата спеціаліста з інформаційної безпеки 15318грн./місяць.

Виконання робіт вимагає залучення спеціаліста з інформаційної безпеки на 0,25 ставки.

$$С_з = (15318*12+15318*12*0,1)*0,25=50549,40 \text{ грн.}$$

В 2022 році ЄСВ є 22% від фонду заробітної плати і становить:

$$С_{єв} = 50549,40*0,22 = 11120,87 \text{ грн.}$$

$С_{ел}$ - це вартість електроенергії, що споживається апаратурою системи ІБ протягом року, вираховується за формулою:

$$С_{ел} = Р * F_р * Ц_е, \text{ грн} \quad (3.12)$$

де:

Р – встановлена потужність апаратури інформаційної безпеки.

Р = 0.7 кВт.

$F_р$- це річний фонд робочого часу системи інформаційної безпеки.

$F_р$ = 1920 год.

$Ц_е$ – це тариф на електроенергію, 1,51грн/кВт годин.

$С_{ел}$ = 0.7*1920*1,51= 2029,44 грн;

$С_{тос}$ – це витрати на технічне та організаційне адміністрування та сервіс системи ІБ визначаються за даними організації. Або 1% від суми капітальних інвестицій – $С_{тос}$ = 573,38 грн.

$С_о$ – це витрати на залучення сторонніх організацій для виконання деяких видів обслуговування та сертифікацію обслуговування персоналу.

$С_о$ = 0 грн.

$С_к$ = 3000+16500+50549,40+11120,87+2029,44+573,38 = 83773,09 грн.

Отже, річні поточні витрати на функціонування системи інформаційної безпеки становлять:

С = 0+84373,49+0 = 83773,09 грн.

3.3Оцінка можливого збитку від DDoS-атак на типовий веб-сервіс

3.3.1 Оцінка величини збитку

Упущена вигода від простою атакованого вузла або сегмента корпоративної мережі становить:

$$U = П_п + П_в + V, \text{ грн.} \quad (3.13)$$

де:

$П_п$ – оплачувані втрати робочого часу та простою співробітників атакованого веб-сервісу, грн;

$П_в$ – вартість відновлення працездатності веб-сервісу (переустановлення системи, зміна конфігурації), грн;

V – втрати від зниження обсягу продажів за час простою атакованого веб-сервісу, грн.

Втрати від зниження продуктивності веб-сервісу являють собою втрати їхньої заробітної плати (оплата непродуктивної праці) за час простою внаслідок атаки:

$$П_п = \frac{\sum З_с}{F} * t_п = 15000 * \frac{10}{176} * 10 = 6818,1 \quad (3.14)$$

де:

$З_с$ – заробітна плата співробітника атакованого веб-сервісу, 12000 грн/міс

F – місячний фонд робочого часу (при 40-а годинному робочому тижні становить 176 ч)

Витрати на відновлення працездатності веб-сервісу включає кілька складових:

$$П_в = П_{ви} + П_{пв} + П_{зч}$$

де:

$П_{ви}$ – витрати на повторне уведення інформації

$П_{пв}$ – вартість на відновлення веб-сервісу вцілому, грн

$П_{зч}$ – вартість заміни устаткування або запасних частин, грн

Витрати на повторне введення інформації розраховуються виходячи з розміру заробітної плати співробітника атакованого веб-сервісу, які зайняті повторним введенням втраченої інформації, з урахуванням необхідного для цього часу $t_{ви}$:

$$П_{ви} = \frac{\sum З_с}{F} * t_п = 12000 * \frac{10}{176} * 8 = 5454{,}5 \text{ грн}$$

Витрати на відновлення веб-сервісу визначаються часом відновлення після атаки і розміром середньогодинної заробітної плати обслуговуючого персоналу (адміністраторів):

$$П_{пв} = \frac{\sum З_с}{F} * t_п = 12000 * \frac{2}{176} * 5 = 1090{,}9 \text{ грн}$$

$$П_в = 5454{,}5 + 1090{,}9 = 6545{,}4 \text{ грн}$$

Втрати від зниження очікуваного обсягу прибутків за час простою атакованого веб-сервісу визначаються виходячи із середньогодинного обсягу прибутку і сумарного часу простою сегмента корпоративної мережі:

$$V = \frac{О}{Fг} * (t_п + t_в + t_{ви})$$

де:

Fг – річний фонд часу роботи філії (52 робочих тижнів, 5-ти денний робочий тиждень, 8-ми годинний робочий день) становить близько 2080 год

$$V = \frac{15000000}{2080} * (10 + 5 + 8) = 16586{,}5 \text{ грн}$$

$$U = 6818{,}1 + 6545{,}4 + 16586{,}6 = 29950 \text{ грн}$$

Таким чином, загальний збиток від атаки на типовий веб-сервіс становить:

$$B = \sum i + \sum n * U = 1 * 40 * 16586{,}6 = 663460 \text{ грн}$$

3.3.2 Загальний ефект від впровадження системи захисту від DDoS-атак

Загальний ефект від впровадження системи захисту веб-сервісу визначається з урахуванням ризиків порушення інформації:

$$E = B * R - C$$

де:

B – загальний збиток від DDoS-атаки на веб-сервіс;

R – вірогідність успішної реалізації атаки на веб-сервіс, частки одиниці (35%);

C – щорічні витрати на експлуатацію всіх заходів, грн.

Загальний ефект від впровадження системи захисту веб-сервісу від DDoS-атак на прикладному рівні визначається з урахуванням ризиків порушення інформації безпеки:

$$E = 663460 * 0{,}35 - 83773{,}09 = 148437{,}91 \text{ грн}$$

3.4 Визначення та аналіз показників економічної ефективності системи інформаційної безпеки

Коефіцієнт повернення інвестицій ROSI показує, скільки гривень додаткового прибутку приносить одна гривня капітальних інвестицій на впровадження системи інформаційної безпеки.

Щодо інформаційної безпеки говорять не про прибуток, а про запобігання можливих втрат від атаки на сегмент або вузол корпоративної мережі, а отже:

$$ROSI = \frac{E}{K}$$

де:

E – Загальний ефект від впровадження системи захисту веб-сервісу

K – Капітальні (фіксовані) витрати на створення системи захисту WAF від DDoS-атак

$$ROSI = \frac{148437,91}{113934,93} = 1.3, \text{ частки одиниці}$$

Проект визначається економічно доцільним, якщо розрахункове значення коефіцієнта повернення інвестицій перевищує величину річної депозитної ставки з урахуванням інфляції:

$$ROSI > (N_{деп} – N_{інф})/1000$$

де:

$N_{деп}$ – річна депозитна ставка, (18%)

$N_{інф}$ – річний рівень інфляції, (11%)

Розрахункове значення коефіцієнта повернення інвестицій:

$$1,3 > (18\text{-}11)/100 = 1,3 > 0,7$$

Термін окупності капітальних інвестицій T_o показує, за скільки років капітальні інвестиції окупляться за рахунок загального ефекту від впровадження системи захисту:

$$T_o = \frac{K}{E} = \frac{1}{ROSI} = \frac{1}{1.3} = 0,76 \text{ років} = 9 \text{ місяців}$$

3.5Висновок

Надані рекомендації з захисту веб-сервісу від DDoS-атак на прикладному рівні є економічно доцільними, оскільки капітальні та експлуатаційні витрати будуть меншими за можливий відвернений збиток. Капітальні витрати складають 113934,93 грн, експлуатаційні 83773,09 грн. Величина річного економічного ефекту складає 148437,91 грн. Коефіцієнт повернення інвестицій ROSI складає 1,3. Порівнюючи ефект від реалізації системи та упущених витрат від негативного впливу DDoS-атак можна сказати, що

результат від впровадження рекомендацій з захисту буде максимально ефективним та окупиться через 9 місяців.

ВИСНОВКИ

На сьогоднішній день веб-сервіси стали невід'ємною частиною сучасного інтернету. Майже кожну операцію можна виконати не виходячи з дому. Також веб-сервісами користуються такі галузі, як державні структури та фінансові підприємства. Такі сфери в першу чергу потребують в захисті та цілісності даних. Актуальною проблемою для компаній стає захист веб-сервісів від DDoS-атак.

DDoS атаки залишаються одними з найпоширенішим видом атак які з кожним роком стають все масштабніше та потужнішими. Зловмисники вигадують нові тактики та техніки і комбінують вже існуючі.

Під захистом веб-сервісів від DDoS-атак розуміють налаштування веб-ресурсів та використання додаткового програмного забезпечення чи пристроїв, які допомагають знизити ризик від DDoS-атак.

В кваліфікаційній роботі було розглянуто актуальність проблем захисту веб-сервісів враховуючи їх роль та функції в сучасному світі. Було приведено статистичні дані щодо DDoS-атак. Проаналізовано методи DDoS-атак та методи захисту від DDoS-атак.

У другому розділі розглянуто DDoS-атаки на прикладному рівні. Представлені методи та техніки захисту від DDoS-атак на прикладному рівні. Розглянута типова конфігурація веб-сервісу. Розроблено рекомендації з забезпечення захисту веб-сервісу від DDoS-атак на прикладному рівні.

В економічному розділі, отримали данні щодо підтвердження економічної доцільності запропонованих проектних рішень.

ПЕРЕЛІК ПОСИЛАННЬ

1. DDoS Attacks Evolution, Detection, Prevention, Reaction and Tolerance [Книга] –Dhruba Kumar Bhattacharyya, Jugal Kumar Kalita.
2. Статистичні дані [Електронний ресурс] – Режим доступу https://docplayer.com/92549752-Statistika-atak-na-veb-prilozheniya-itogi-2017-goda.html
3. Статистичні дані від Google [Електронний ресурс] – https://cloud.google.com/blog/products/identity-security/identifying-and-protecting-against-the-largest-ddos-attacks
4. Статистичні дані [Електронний ресурс] – Режим доступу https://www.f5.com/labs/articles/threat-intelligence/2022-application-protection-report-ddos-attack-trends.
5. Статистичні дані [Електронний ресурс] – Режим доступу https://www.cloudflare.com/en-gb/learning/ddos/famous-ddos-attacks/
6. Приклад від Google [Електронний ресурс] – Режим доступу https://cloud.google.com/blog/products/identity-security/improve-your-ddos--waf-defense-with-new-cloud-armor-features
7. Налаштування системи захисту WAF від DDoS-атак прикладного рівня [Електронний ресурс] – Режим доступу https://docs.aws.amazon.com/waf/index.html.
8. Топ 10 вразливостей web-ресурсів [Електронний ресурс] – Режим доступу https://owasp.org/www-project-top-ten/.
9. Web-сервер очима хакера 3 видання, Проблеми безпеки web-серверів [Книга] – Михайло Фленов.
10. Модель DDoS-атак [Електронний ресурс] – Режим доступу https://www.cloudflare.com/learning/ddos/what-is-a-ddos-attack/.
11. Бізнес під загрозою кібератаки. Як захистити компанію? [Електронний ресурс] – Режим доступу https://biz.ligazakon.net/news/208297_bznes-pd-zagrozoyu-kberataki-yak-zakhistiti-kompanyu.

12. Джерела загроз інформаційної безпеки [Електронний ресурс] – Режим доступу https://studfile.net/preview/9649913/.
13. Типи DDoS-атак прикладного рівня та їх опис [Електронний ресурс] – Режим доступу https://www.netscout.com/what-is-ddos/application-layer-attacks.
14. A Guide to Web Application Firewall [Електронний ресурс] – Режим доступу https://www.trustradius.com/buyer-blog/web-application-firewall-vs-network-firewall.
15. Методичні рекомендації до виконання кваліфікаційних робіт бакалаврів спеціальності 124 Кібербезпека/ Упоряд.: О.В. Герасіна, Д.С. Тимофєєв, О.В. Кручінін, Ю.А. Мілінчук – Дніпро: НТУ «ДП», 2020. – 47 с.
16. Методичні вказівки до виконання економічної частини дипломного проекту зі спеціальності 125 Кібербезпека / Упорядн.: Д.П. Пілова – Дніпро: Національний технічний університет «Дніпровська політехніка», 2019. – 16 с

ДОДАТОК А. Відомість матеріалів кваліфікаційної роботи

№	Формат	Найменування	Кількість листів	Примітка
Документація				
1	А4	Реферат	2	
2	А4	Список умовних скорочень	1	
3	А4	Зміст	2	
4	А4	Вступ	2	
5	А4	Стан питання. Постановка задачі.	15	
6	А4	Спеціальна частина	27	
7	А4	Економічний розділ	10	
8	А4	Висновки	1	
9	А4	Перелік посилань	2	
10	А4	ДОДАТОК А	1	
11	А4	ДОДАТОК Б	6	
12	А4	ДОДАТОК В	1	
13	А4	ДОДАТОК Г	1	
14	А4	ДОДАТОК Ґ	1	

ДОДАТОК Б. Конфігураційний файл Apache 2.4.18

```
# This is the main Apache server configuration file.  It contains the
# configuration directives that give the server its instructions.
# See http://httpd.apache.org/docs/2.4/ for detailed information about
# the directives and /usr/share/doc/apache2/README.Debian about Debian
specific
# hints.
#
# Summary of how the Apache 2 configuration works in Debian:
# The Apache 2 web server configuration in Debian is quite different to
# upstream's suggested way to configure the web server. This is because
Debian's
# default Apache2 installation attempts to make adding and removing modules,
# virtual hosts, and extra configuration directives as flexible as possible,
in
# order to make automating the changes and administering the server as easy
as
# possible.

# It is split into several files forming the configuration hierarchy outlined
# below, all located in the /etc/apache2/ directory:
#
#     /etc/apache2/
#     |-- apache2.conf
#     |     `--  ports.conf
#     |-- mods-enabled
#     |     |-- *.load
#     |     `-- *.conf
#     |-- conf-enabled
#     |     `-- *.conf
#     `-- sites-enabled
#           `-- *.conf
#
#
# * apache2.conf is the main configuration file (this file). It puts the
pieces
```

```
#   together by including all remaining configuration files when starting up
the
#   web server.
#
# * ports.conf is always included from the main configuration file. It is
#   supposed to determine listening ports for incoming connections which can
be
#   customized anytime.
#
# * Configuration files in the mods-enabled/, conf-enabled/ and sites-
enabled/
#   directories contain particular configuration snippets which manage
modules,
#   global configuration fragments, or virtual host configurations,
#   respectively.
#
#   They are activated by symlinking available configuration files from their
#   respective *-available/ counterparts. These should be managed by using
our
#   helpers a2enmod/a2dismod, a2ensite/a2dissite and a2enconf/a2disconf. See
#   their respective man pages for detailed information.
#
# * The binary is called apache2. Due to the use of environment variables, in
#   the default configuration, apache2 needs to be started/stopped with
#   /etc/init.d/apache2 or apache2ctl. Calling /usr/bin/apache2 directly will
not
#   work with the default configuration.
# Global configuration
#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
# NOTE!  If you intend to place this on an NFS (or otherwise network)
# mounted filesystem then please read the Mutex documentation (available
# at <URL:http://httpd.apache.org/docs/2.4/mod/core.html#mutex>);
# you will save yourself a lot of trouble.
#
# Do NOT add a slash at the end of the directory path.
#
```

```
#ServerRoot "/etc/apache2"
# The accept serialization lock file MUST BE STORED ON A LOCAL DISK.
#
#Mutex file:${APACHE_LOCK_DIR} default
# The directory where shm and other runtime files will be stored.

DefaultRuntimeDir ${APACHE_RUN_DIR}

# PidFile: The file in which the server should record its process
# identification number when it starts.
# This needs to be set in /etc/apache2/envvars
#
PidFile ${APACHE_PID_FILE}

# Timeout: The number of seconds before receives and sends time out.
#
Timeout 300

# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 10000

# KeepAliveTimeout: Number of seconds to wait for the next request from the
# same client on the same connection.
#
KeepAliveTimeout 3
# These need to be set in /etc/apache2/envvars
User ${APACHE_RUN_USER}
Group ${APACHE_RUN_GROUP}

# HostnameLookups: Log the names of clients or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132 (off).
```

```
# The default is off because it'd be overall better for the net if people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
#
HostnameLookups Off

# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here.  If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog ${APACHE_LOG_DIR}/error.log
# LogLevel: Control the severity of messages logged to the error_log.
# Available values: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the log level for particular modules, e.g.
# "LogLevel info ssl:warn"
#
LogLevel warn
# Include module configuration:
IncludeOptional mods-enabled/*.load
IncludeOptional mods-enabled/*.conf
# Include list of ports to listen on
Include ports.conf
# Sets the default security model of the Apache2 HTTPD server. It does
# not allow access to the root filesystem outside of /usr/share and /var/www.
# The former is used by web applications packaged in Debian,
# the latter may be used for local directories served by the web server. If
# your system is serving content from a sub-directory in /srv you must allow
# access here, or in any related virtual host.
<Directory />
      Options FollowSymLinks
      AllowOverride None
      Require all denied
</Directory>
```

```
<Directory /usr/share>
      AllowOverride None
      Require all granted
</Directory>

<Directory /var/www/>
      Options Indexes FollowSymLinks
      AllowOverride None
      Require all granted
</Directory>

#<Directory /srv/>
#     Options Indexes FollowSymLinks
#     AllowOverride None
#     Require all granted
#</Directory>

# AccessFileName: The name of the file to look for in each directory
# for additional configuration directives.  See also the AllowOverride
# directive.
#
AccessFileName .htaccess
#
# The following lines prevent .htaccess and .htpasswd files from being
# viewed by Web clients.
#
<FilesMatch "^\.ht">
      Require all denied
</FilesMatch>

# The following directives define some format nicknames for use with
# a CustomLog directive.
#
# These deviate from the Common Log Format definitions in that they use %O
# (the actual bytes sent including headers) instead of %b (the size of the
# requested file), because the latter makes it impossible to detect partial
# requests.
#
```

```
# Note that the use of %{X-Forwarded-For}i instead of %h is not recommended.
# Use mod_remoteip instead.
#
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-
Agent}i\"" vhost_combined
LogFormat "%h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\""
combined
LogFormat "%h %l %u %t \"%r\" %>s %O" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf
```

ДОДАТОК В. Перелік матеріалів на оптичному носії

Нашиванько_РО_125_18_1_ПЗ.docx

Нашиванько_РО_125_18_1_ПЗ.pdf

Нашиванько_РО_125_18_1_ДМ.pptx

Нашиванько_РО_125_18_1_ПЗ.pdf

Нашиванько_РО_125_18_1_ПЗ.pdf.p7s

ДОДАТОК Г. Відгук керівник економічного розділу

Відгук керівника економічного розділу:

Економічний розділ виконаний відповідно до вимог, які ставляться до кваліфікаційних робіт, та заслуговує на оцінку 90 б. («відмінно»).

Керівник розділу _____ доц. Пілова Д.П

ДОДАТОК Ґ. Відгук керівника кваліфікаційної роботи

В І Д Г У К

на кваліфікаційну роботу студента групи 125-18-1

Нашиванько Романа Олексійовича

на тему: «Захист веб-сервісів від DDoS-атак на прикладному рівні»

Пояснювальна записка складається зі вступу, трьох розділів і висновків, викладених на 52 сторінках.

Метою кваліфікаційної роботи є забезпечення достатнього рівня захисту веб-сервісів від DDoS-атак на прикладному рівня моделі OSI.

Тема кваліфікаційної роботи безпосередньо пов'язана з об'єктом діяльності бакалавра спеціальності 125 «Кібербезпека». Для досягнення поставленої мети в кваліфікаційній роботі вирішуються наступні задачі: аналіз статистичних даних щодо DDoS-атак; аналіз методів DDoS-атак та методів захисту від DDoS-атак.

Розроблено рекомендації з захисту веб-сервісу типового виду від DDoS-атак на прикладному рівні.

Практичне значення результатів кваліфікаційної роботи полягає у підвищенні ефективності процесу захисту від DDoS-атак, за рахунок розробки рекомендацій з захисту.

Оформлення пояснювальної записки до кваліфікаційної роботи виконано з незначними відхиленнями від стандартів.

За час дипломування Нашиванько Р.О. проявила себе фахівцем, здатним самостійно вирішувати поставлені задачі та заслуговує присвоєння кваліфікації бакалавра за спеціальністю 125 Кібербезпека, освітньо-професійна програма «Кібербезпека» .

Рівень запозичень у кваліфікаційній роботі не перевищує вимог "Положення про систему виявлення та запобігання плагіату".

Кваліфікаційна робота заслуговує оцінку «відмінно»

Керівник кваліфікаційної роботи _____ Кагадій Т.С.

Керівник спец. Розділу _____ Тимофєєв Д.С.