ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ
КАФЕДРА ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ ТА МЕРЕЖ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «ДОСЛІДЖЕННЯ ВПЛИВУ DDOS-АТАК НА ТЕЛЕКОМУНІКАЦІЙНІ МЕРЕЖІ ТА МЕТОДИ ЇХНЬОГО ВИЯВЛЕННЯ»

на здобуття освітнього ступеня магістра
зі спеціальності 172 Електронні комунікації та радіотехніка
(код, найменування спеціальності)
освітньо-професійної програми Телекомунікаційні системи та мережі
(назва)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

_____ Мирослава НІГРУЦА
(підпис) *(Ім'я ПРІЗВИЩЕ)*

Виконала: здобувачка вищої освіти гр. ТСДМ-62
Мирослава НІГРУЦА
Ім'я, ПРІЗВИЩЕ

Керівник: Вікторія КОРЕЦЬКА
к.п.н., доцент *Ім'я, ПРІЗВИЩЕ*

Рецензент: _____
Ім'я, ПРІЗВИЩЕ

Київ 2025

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут Телекомунікацій

Кафедра ___Телекомунікаційних систем та мереж_____
Ступінь вищої освіти __магістр_____
Спеціальність ___172 Електронні комунікації та радіотехніка _____
Освітньо-професійна програма _____Телекомунікаційні системи та мережі_____

ЗАТВЕРДЖУЮ
Завідувач кафедри __ТСМ_____
_____ Віктор ЗАЇКА
«____» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Нігруці Мирославі Валентинівні
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Дослідження впливу DDoS-атак на телекомунікаційні мережі та методи їхнього виявлення
керівник кваліфікаційної роботи Вікторія КОРЕЦЬКА, к.п.н., доцент
(Ім'я, ПРІЗВИЩЕ науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «30» 10. 2025 р. № 467

2. Строк подання кваліфікаційної роботи «___» _____ 2025 р.

3. Вихідні дані до кваліфікаційної роботи:
1. DDoS атаки.
2. Вплив DDoS-атак на телекомунікаційні мережі.
3. Захист від DDoS атак.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1. База DDoS атак.
2. Компоненти протидії DDoS.
3. Захист від DDoS атак з точки зору операторів зв'язку.
4. Комплексні методи захисту.

5. Перелік ілюстративного матеріалу: ___презентація на ____слайдах_____

6. Дата видачі завдання «___» _____ 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Підбір та аналіз науково-технічної літератури	15.10.2025	Викон.
2	Розробка першого розділу	22.10.2025	Викон.
3	Розробка другого розділу	02.11.2025	Викон.
4	Розробка третього розділу	12.11.2025	Викон.
5	Розробка четвертого розділу	22.11.2025	Викон.
6	Висновки, вступ, реферат	02.12.2025	Викон.
7	Розробка презентації	12.12.2025	Викон.

Здобувачка вищої освіти _____ Мирослава НІГРУЦА
(підпис) *(Ім'я ПРІЗВИЩЕ)*

Керівник кваліфікаційної роботи _____ Вікторія КОРЕЦЬКА
(підпис) *(Ім'я ПРІЗВИЩЕ)*

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут телекомунікацій
ПОДАННЯ
ГОЛОВІ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ
ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ
на здобуття освітнього ступеня магістра

Направляється здобувачка Нігруца М.В. до захисту кваліфікаційної роботи
(прізвище та ініціали)
за спеціальністю 172 Електронні комунікації та радіотехніка
(код, найменування спеціальності)
освітньо-професійної програми Телекомунікаційні системи та мережі
(назва)
на тему: «Дослідження впливу DDoS-атак на телекомунікаційні мережі та методи їхнього виявлення»

Кваліфікаційна робота і рецензія додаються.

Директор ННІТ _____ Владислав КРАВЧЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Висновок керівника кваліфікаційної роботи

Здобувачка Нігруца Мирослава Валентинівна показала гарну теоретичну та інженерну підготовку, уміння володіти новими комп'ютерними технологіями, користуватися навчальною, довідковою і науково-технічною літературою. Працюючи над завданнями, які були поставлені керівником, проявила ініціативність, сумлінність та хист до наукової роботи.

Все це дозволяє оцінити виконану кваліфікаційну роботу здобувачки Нігруци Мирослави Валентинівни на оцінку «відмінно» та присвоїти їй кваліфікацію магістр з електронних комунікацій та радіотехніки.

Керівник кваліфікаційної роботи _____ Вікторія КОРЕЦЬКА
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

«____» _____ 20___ року

Висновок кафедри про кваліфікаційну роботу

Кваліфікаційна робота розглянута. Здобувачка Нігруца М.В. допускається до захисту даної роботи в Екзаменаційній комісії.

Завідувач кафедри телекомунікаційних систем та мереж _____ Віктор ЗАЇКА
(назва) *(підпис)* *(Ім'я, ПРІЗВИЩЕ)*

«____» _____ 20___ року

ВІДГУК РЕЦЕНЗЕНТА
на кваліфікаційну роботу
на здобуття освітнього ступеня магістра

здобувачки вищої освіти Нігруци Мирослави Валентинівни
(прізвище, ім'я, по батькові)
на тему «Дослідження впливу DDoS-атак на телекомунікаційні мережі та методи їхнього виявлення»

Актуальність.
У сучасних умовах інтенсивного розвитку цифрової економіки, хмарних сервісів та критичних інформаційних систем проблема протидії DDoS-атакам набуває особливої ваги. Відмова в обслуговуванні навіть на короткий час може призвести до суттєвих фінансових збитків, втрати клієнтів і підриву репутації компаній та державних установ. З огляду на збільшення масштабів, складності та багатовекторності DDoS-кампаній, традиційні методи фільтрації трафіку вже не забезпечують достатнього рівня безпеки.

Позитивні сторони.
1. У проаналізовано основні архітектурні підходи до протидії DDoS.
2. У роботі досліджено різновиди атак — від класичних ICMP- та UDP-flood до більш витончених методів на основі протоколів TCP, SSL/TLS та HTTP, включно з «повільними» Low & Slow-атаками, Wordpress Pingback DDoS, Full Browser Stack-атаками.

Недоліки.
1. У роботі бракує таблиць, графіків або статистики (наприклад, частоти атак, обсягів трафіку, часу реагування систем захисту)..
2. У роботі було б доцільно навести дані CERT-UA, Держспецзв'язку чи провайдерів щодо DDoS-атак в Україні 2022–2025 років.

Відзначені зауваження не впливають на загальну позитивну оцінку магістерської кваліфікаційної роботи

Висновок: *кваліфікаційна робота на здобуття ступеня магістра заслуговує оцінку "відмінно", а здобувачка Нігруца Мирослава Валентинівна заслуговує присвоєння кваліфікації: магістр з електронних комунікацій та радіотехніки.*

Рецензент _____ _____
науковий ступінь, вчене звання (підпис) *(Ім'я ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 78 стор., 17 рис., 23 джерел.

Мета роботи – аналіз механізмів реалізації DDoS-атак у телекомунікаційних мережах та розроблення узагальненої моделі багаторівневої системи протидії, яка забезпечує своєчасне виявлення, фільтрацію шкідливого трафіку та підвищення стійкості інформаційної інфраструктури до кібератак.

Об’єкт дослідження – процес забезпечення доступності та безперервності функціонування телекомунікаційних мереж в умовах впливу DDoS-атак.

Предмет дослідження – методи, засоби та архітектурні рішення протидії DDoS-атакам, зокрема механізми фільтрації трафіку, системи очищення (scrubbing centers), міжмережеві екрани, Web Application Firewall (WAF) та технології поведінкового аналізу й машинного навчання.

Короткий зміст роботи: у кваліфікаційній роботі встановлено, що DDoS-атаки є однією з найсерйозніших загроз для телекомунікаційних мереж, особливо на прикладному рівні (L7). Проаналізовано основні типи атак — від ICMP- та UDP-flood до HTTP- і SSL/TLS-флудів, а також розглянуто сучасні методи протидії. Найефективнішою визначено багаторівневу систему захисту, яка поєднує фільтрацію трафіку, Центри очищення, WAF і поведінковий аналіз. Перспективним напрямом розвитку є застосування машинного навчання для автоматичного виявлення аномалій і швидкого реагування на нові форми атак, що забезпечує безперервність роботи мереж.

КЛЮЧОВІ СЛОВА: DDOS-АТАКИ, ТЕЛЕКОМУНІКАЦІЙНІ МЕРЕЖІ, КІБЕРБЕЗПЕКА, ФІЛЬТРАЦІЯ ТРАФІКУ, WEB APPLICATION FIREWALL, ПОВЕДІНКОВИЙ АНАЛІЗ, МАШИННЕ НАВЧАННЯ, ВИЯВЛЕННЯ АНОМАЛІЙ, БАГАТОРІВНЕВИЙ ЗАХИСТ, ЦЕНТР ОЧИЩЕННЯ ТРАФІКУ

ABSTRACT

Text part of the master's qualification work: 78 pages, 17 figures, 23 sources.

The purpose of the work is to analyze the mechanisms of implementing DDoS attacks in telecommunication networks and to develop a generalized model of a multi-level countermeasure system that ensures timely detection, filtering of malicious traffic and increasing the resilience of the information infrastructure to cyberattacks.

Object of the study is the process of ensuring the availability and continuity of operation of telecommunication networks under the influence of DDoS attacks.

Subject of the study is methods, tools and architectural solutions for countering DDoS attacks, in particular traffic filtering mechanisms, scrubbing centers, firewalls, Web Application Firewall (WAF) and behavioral analysis and machine learning technologies.

Summary of the work: the qualification work established that DDoS attacks are one of the most serious threats to telecommunication networks, especially at the application level (L7). The main types of attacks are analyzed - from ICMP and UDP floods to HTTP and SSL/TLS floods, and modern countermeasure methods are also considered. The most effective is a multi-level protection system that combines traffic filtering, Cleaning Centers, WAF and behavioral analysis. A promising direction of development is the use of machine learning for automatic detection of anomalies and rapid response to new forms of attacks, which ensures the continuity of network operations.

KEYWORDS: DDOS-ATTACKS, TELECOMMUNICATION NETWORKS, CYBERSECURITY, TRAFFIC FILTERING, WEB APPLICATION FIREWALL, BEHAVIOR ANALYSIS, MACHINE LEARNING, ANOMALIES DETECTION, MULTI-LEVEL PROTECTION, TRAFFIC CLEANING CENTER

ЗМІСТ

ВСТУП 9
РОЗДІЛ 1 БАЗА DDOS АТАК 11
1.1 DoS та DDoS-атаки: суть, різновиди та класифікація 11
1.2 Атаки мережевого та транспортного рівнів 15
1.3 Допоміжні види атак 20
РОЗДІЛ 2 КОМПОНЕНТИ ПРОТИДІЇ DDOS 34
2.1 Основні механізми захисту 34
2.2 Методи інтеграції компонентів захисту 36
2.3 Підходи до архітектури системи 40
РОЗДІЛ 3 ЗАХИСТ ВІД DDOS АТАК З ТОЧКИ ЗОРУ ОПЕРАТОРІВ ЗВ'ЯЗКУ 46
3.1 Рішення захисту для операторів 46
3.2 Детектування DDoS-атак 47
3.3 Фільтрація DDoS атак 55
РОЗДІЛ 4 КОМПЛЕКСНІ МЕТОДИ ЗАХИСТУ 61
4.1 Використання машинного навчання для протидії DDoS 61
4.2 Захист від DDOS атак засобами BGP 67
4.3 Захист від DDoS на рівні веб-сервера 70
ВИСНОВКИ 77
ПЕРЕЛІК ПОСИЛАНЬ 79
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ 81

ВСТУП

Сучасні телекомунікаційні мережі є фундаментом цифрового суспільства та забезпечують безперервний обмін інформацією в бізнесі, фінансовій сфері, державному управлінні та повсякденному житті. Зростання кількості онлайн-сервісів, електронної комерції, хмарних технологій та мобільних застосунків робить інфраструктуру зв’язку критично важливою для функціонування економіки та соціальних систем. Водночас, це зумовлює підвищення рівня загроз, серед яких особливе місце займають розподілені атаки на відмову в обслуговуванні (DDoS-атаки).

DDoS-атаки спрямовані на створення умов, за яких легітимні користувачі втрачають можливість отримати доступ до ресурсів чи послуг. Їхня небезпека полягає не лише у перевантаженні каналів чи серверів, але й у здатності вражати прикладний рівень (L7) моделі OSI, де атаки стають більш витонченими та важко відслідковуваними. Застосування ботнетів із десятків чи навіть сотень тисяч заражених пристроїв, використання методів маскування та багатовекторність роблять такі атаки серйозним викликом для операторів і власників інформаційних систем.

Проблема DDoS-атак набула глобального масштабу. Відомі випадки атак із піковими значеннями понад 1 Тбіт/с, що призводили до зупинки роботи відомих сервісів, зокрема GitHub. Крім того, спостерігається зростання кількості атак на веб-застосунки та протоколи SSL/TLS, які потребують значно більш складних методів виявлення та протидії. Сучасні зловмисники активно використовують як «сирі» flood-атаки (ICMP, UDP, SYN flood), так і складні прикладні атаки (HTTP flood, Low & Slow, Wordpress Pingback DDoS, Full Browser Stack тощо).

Важливим аспектом проблеми є швидкість реагування та ефективність засобів захисту. Традиційні «ручні» або напівавтоматичні методи вже не забезпечують належного рівня протидії, оскільки характер атак змінюється за лічені хвилини. Саме тому актуальним є розвиток багаторівневих систем захисту, які поєднують методи фільтрації трафіку, застосування Центрів очищення (ЦОТ),

використання механізмів Flow Specification, Web Application Firewall (WAF), а також алгоритмів поведінкового аналізу.

Окрему увагу привертають перспективи впровадження машинного навчання. Алгоритми ML, зокрема Decision Tree, Random Forest та нейронні мережі, довели свою ефективність у завданнях класифікації трафіку, дозволяючи відрізняти легітимні запити від зловмисних навіть у складних умовах. Це відкриває нові можливості для побудови адаптивних систем протидії, здатних оперативно реагувати на атаки «нульового дня».

Таким чином, тема дослідження впливу DDoS-атак на телекомунікаційні мережі та методів їхнього виявлення є надзвичайно актуальною. Вона має не лише теоретичне, але й важливе практичне значення, оскільки ефективна протидія цим загрозам є запорукою стабільної роботи інформаційної інфраструктури та гарантією безперервності цифрових сервісів.

1 БАЗА DDOS АТАК

1.1 DoS та DDoS-атаки: суть, різновиди та класифікація

DoS (Denial of Service — відмова в обслуговуванні) — це тип атаки на обчислювальну систему, основною метою якої є виведення з ладу або значне зниження доступності її сервісів. Під час такої атаки створюються умови, за яких легальні користувачі не мають можливості отримати доступ до необхідних ресурсів або цей доступ стає настільки обмеженим, що фактично унеможливлює нормальне функціонування сервісу.

DDoS (Distributed Denial of Service — розподілена відмова в обслуговуванні) є більш небезпечною формою DoS-атаки. У цьому випадку для досягнення ефекту використовується не один атакуючий вузол, а велика кількість заражених комп'ютерів або пристроїв (так звані «ботнети»). DDoS-атака полягає у виснаженні ресурсів атакованої системи або зовнішніх компонентів, від яких вона залежить: каналів зв'язку, систем зберігання даних, транзитних операторів тощо.

Метою таких атак є не лише втрата доступності, а й порушення здатності сервісу своєчасно й адекватно реагувати на запити легітимних клієнтів. У деяких випадках це призводить до повного паралічу сервісу. Серед основних мотивів зловмисників варто виділити: недобросовісну конкуренцію, хактивізм (що зумовлений ідеологічними чи політичними мотивами), фінансове вимагання, кібершпигунство, а також використання DDoS як «димової завіси» для маскування більш складних атак на інфраструктуру.

Класифікація DoS/DDoS-атак

Сучасні атаки цього класу охоплюють усі рівні моделі OSI — від мережевого (L3) до прикладного (L7). Найпоширеніші різновиди можна поділити на такі групи:

1. **Атаки на мережеві ресурси (L3–L4).**

   Вони спрямовані на вичерпання канальної ємності мережі як у самої жертви, так і в її провайдера чи транзитних операторів. Класичними прикладами є ICMP- та UDP-flood, Smurf-атака, Ping of Death. Ці атаки є технічно

простими, оскільки часто не вимагають встановлення TCP-з’єднання. В окремий підклас виділяють **Amplification-атаки**, що ґрунтуються на використанні проміжних уразливих вузлів, які здатні підсилювати атаку в кілька разів. Найбільш відомі приклади: **DNS Amplification**, **NTP Amplification**, **SSDP Amplification**.

2. **Атаки на інфраструктурне обладнання.**

   Вони спрямовані на порушення роботи критично важливих мережевих пристроїв — маршрутизаторів, міжмережевих екранів, систем балансування навантаження тощо. Серед прикладів — **SYN flood**, **Route Loop DDoS**, а також помилкові або навмисно спотворені BGP-анонси, здатні викликати масштабні перебої в роботі інтернет-маршрутизації.

3. **Атаки на транспортний рівень (L4).**

   Тут експлуатуються обчислювальні складнощі TCP-протоколу. Атакуючі можуть навмисно створювати велику кількість неповних або некоректних TCP-з’єднань (наприклад, **SYN Flood**, **ACK Flood**, **TCP Connection Flood**), що призводить до виснаження ресурсів серверів. Цей тип атак вважається одним із найпоширеніших завдяки відносній простоті реалізації.

4. **Атаки на протоколи SSL/TLS.**

   Протоколи шифрування даних ще складніші за TCP і тому відкривають додаткові вектори атак. Зловмисники можуть атакувати процес встановлення TLS-з’єднання (TLS Handshake), надсилати серверу псевдовипадкові «сміттєві» пакети або зловживати механізмами узгодження ключів. Це створює значне навантаження і може призвести до відмови в обслуговуванні навіть у захищених сервісах.

5. **Атаки на прикладний рівень (L7).**

   Це найбільш складні й небезпечні атаки, оскільки вони спрямовані безпосередньо на прикладні сервіси та протоколи. Вони охоплюють HTTP/HTTPS, DNS, VoIP, SMTP, FTP та інші протоколи. Особливістю є можливість маскування шкідливого трафіку всередині SSL/TLS-сеансів.

- **Малі та повільні атаки («Low and Slow»).** Вони діють непомітно, поступово виснажуючи ресурси серверів, що ускладнює їх виявлення.
- **Атаки великого обсягу «сміттєвих» запитів.** Вони призводять до миттєвого перевантаження сервісу. Приклад — **Wordpress Pingback DDoS**, який здатен генерувати трафік у десятки гігабіт на секунду.
- **Атаки з імітацією легітимної активності.** Вони базуються на використанні ботів, що повторюють дії звичайних користувачів. Це дозволяє експлуатувати уразливості веб-додатків і часто супроводжується скануванням мережі, яке слугує підготовчим етапом.

Таким чином, DoS- і DDoS-атаки є багаторівневим явищем, що охоплює як прості технічні дії, так і високотехнологічні маніпуляції. Їх небезпека полягає не лише в тимчасовій втраті доступності сервісів, а й у можливості прихованого проникнення до інфраструктури та подальших більш руйнівних атак.

Приклади DDoS-атак на різних рівнях моделі OSI

Сучасні DDoS-атаки охоплюють усі рівні мережевої взаємодії, проте найбільший інтерес становлять атаки прикладного рівня (L7). Вони є більш небезпечними та деструктивними, оскільки спрямовані безпосередньо на веб-застосунки та сервіси. Їх особливість полягає в тому, що зловмисник не просто перевантажує канали чи серверні ресурси, а фактично втручається у логіку роботи застосунку. Унаслідок цього можуть виникати критичні збої: некоректна обробка запитів, спотворення результатів бізнес-процесів, втрата клієнтських транзакцій або навіть суттєве зниження прибутку компанії.

Атаки рівня L7 також надзвичайно складні для виявлення та нейтралізації. Стандартні засоби фільтрації, які ефективні на мережевому (L3) чи транспортному (L4) рівнях, тут практично безсилі. У багатьох випадках для протидії таким атакам необхідне використання Web Application Firewall (WAF), який здатний аналізувати структуру HTTP/HTTPS-запитів, застосовувати контекстні правила, виконувати глибоку інспекцію пакетів і відсіювати підозрілий трафік ще на рівні логіки застосунку.

На відміну від цього, атаки L3 та L4 зазвичай мають інший характер. Вони не впливають безпосередньо на прикладну логіку, проте призводять до глобальної недоступності сервісу. У таких випадках система або зовсім не відповідає на запити користувачів, або повертає стандартні серверні помилки класу 5xx, що вказують на перевантаження чи неспроможність обробити звернення. Особливістю прикладних атак є те, що вони можуть тривалий час не призводити до очевидних збоїв чи помилок, маскуючи свій вплив у звичайному трафіку. Це робить їх особливо небезпечними, адже компанія може вчасно не помітити шкідливий вплив.

Пояснення термінів L3 / L4 / L7

L3 – мережевий рівень (Network Layer).

Цей рівень відповідає за логічну адресацію та маршрутизацію даних у мережі. Саме тут здійснюється трансляція IP-адрес у фізичні адреси та реалізується робота основних протоколів маршрутизації (OSPF, BGP, RIP тощо). Метою атак цього рівня є виснаження пропускної здатності каналів зв’язку, перевантаження маршрутизаторів або створення ситуацій, коли пакети даних не можуть бути доставлені адресатові. Класичними прикладами є ICMP Flood, Smurf-атака, UDP Flood.

L4 – транспортний рівень (Transport Layer).

На цьому рівні працюють протоколи TCP і UDP, які відповідають за організацію доставки даних між клієнтом і сервером. Він забезпечує встановлення з’єднання, контроль послідовності пакетів і перевірку їхньої цілісності. Атаки на транспортному рівні експлуатують слабкі місця у механізмах TCP-з’єднань. Серед них — SYN Flood, який створює велику кількість «напіввідкритих» сесій, або ACK Flood, що спрямований на виснаження обчислювальних ресурсів сервера. Такі атаки здатні швидко вивести з ладу навіть потужні сервери, оскільки вони споживають значні ресурси обробки.

L7 – прикладний рівень (Application Layer).

Найближчий до користувача рівень, на якому працюють прикладні протоколи та сервіси, зокрема HTTP, HTTPS, DNS, SMTP, FTP, VoIP тощо. Саме тут реалізуються всі веб-застосунки, які бачать користувачі у браузерах та клієнтських програмах.

Атаки на цьому рівні вважаються найскладнішими, оскільки вони імітують звичайну користувацьку активність: надсилання великої кількості запитів до серверу, створення «повільних» запитів (Low and Slow), що навмисне утримуються відкритими, або генерацію великого обсягу сміттєвого трафіку (наприклад, Wordpress Pingback DDoS). Їхня небезпека полягає у прихованості та здатності обходити класичні системи захисту.

Таким чином, різні рівні моделі OSI мають власні специфічні загрози й уразливості. Якщо атаки L3/L4 у більшості випадків призводять до прямої відмови у доступі до сервісу, то атаки L7 здатні завдати значно більшої шкоди, оскільки вони впливають на логіку прикладних систем і бізнес-процеси. Саме тому сучасні підходи до захисту повинні поєднувати багаторівневу фільтрацію трафіку, використання міжмережевих екранів, балансування навантаження, системи WAF та комплексний моніторинг поведінки користувачів у реальному часі.

1.2 Атаки мережевого та транспортного рівнів

ICMP Flood

Атака типу ICMP Flood належить до класу мережевих (L3) атак і вважається однією з найпростіших, проте ефективних технік порушення доступності сервісів. Сутність атаки полягає у масовому надсиланні на вузол-жертву стандартних ехо-запитів (ICMP Echo Request). Кожен такий пакет змушує цільовий вузол виконати низку дій: перевірити коректність запиту, виділити обчислювальні ресурси для його обробки та згенерувати ехо-відповідь (ICMP Echo Reply).

Особливість ICMP Flood полягає в асиметрії навантаження: хоча один запит має невеликий розмір, його обробка потребує від системи значно більше ресурсів — як процесорних, так і мережевих. Це означає, що зловмисник може, використовуючи відносно малу кількість трафіку, спричинити критичне перевантаження цільового вузла за кількістю оброблюваних пакетів. У результаті система починає втрачати пакети, відповідати із затримками або повністю виходить із робочого стану.

Ще одним фактором, який робить цю атаку небезпечною, є легкість реалізації. Генерація великої кількості коректних ICMP Echo-запитів (у тому числі зі спуфінгом IP-адреси, коли підробляється адреса відправника) не вимагає складних алгоритмів. Завдяки цьому зловмисник здатний сформувати значний потік трафіку, що призведе до вичерпання канальної ємності атакованої системи.

Для підвищення ефективності часто використовується ботнет — розподілена мережа заражених комп'ютерів та IoT-пристроїв, які одночасно надсилають ICMP-запити. У такому разі атакуючий отримує можливість генерувати мільйони пакетів за секунду з тисяч різних джерел, що робить атаку важковиявною та практично некерованою для системи-захисника.

Окремим різновидом є так звана Smurf-атака, яка належить до підсилювальних (Amplification) атак. Її суть полягає у відправленні ICMP Echo-запиту на широкомовну адресу мережі із підробленою IP-адресою джерела, яка вказує на жертву. Усі вузли, що отримують такий широкомовний запит, автоматично відповідають ехо-відповіддю саме жертві. Це створює ситуацію, коли один запит здатний згенерувати десятки або навіть сотні відповідей. Таким чином відбувається багаторазове посилення трафіку, і жертва отримує лавиноподібний потік ICMP-відповідей, який значно перевищує її реальні можливості щодо обробки вхідних даних.

Наслідки ICMP Flood і Smurf-атак можуть бути різними: від часткової втрати пакетів та зниження швидкодії сервісів до повного виведення з ладу серверів, маршрутизаторів чи інших критично важливих вузлів мережі. Особливу небезпеку становить той факт, що ICMP-трафік є службовим і традиційно використовується для діагностики та моніторингу мереж. Це ускладнює фільтрацію, адже блокування ICMP-пакетів повністю може вплинути на коректність роботи мережевих механізмів.

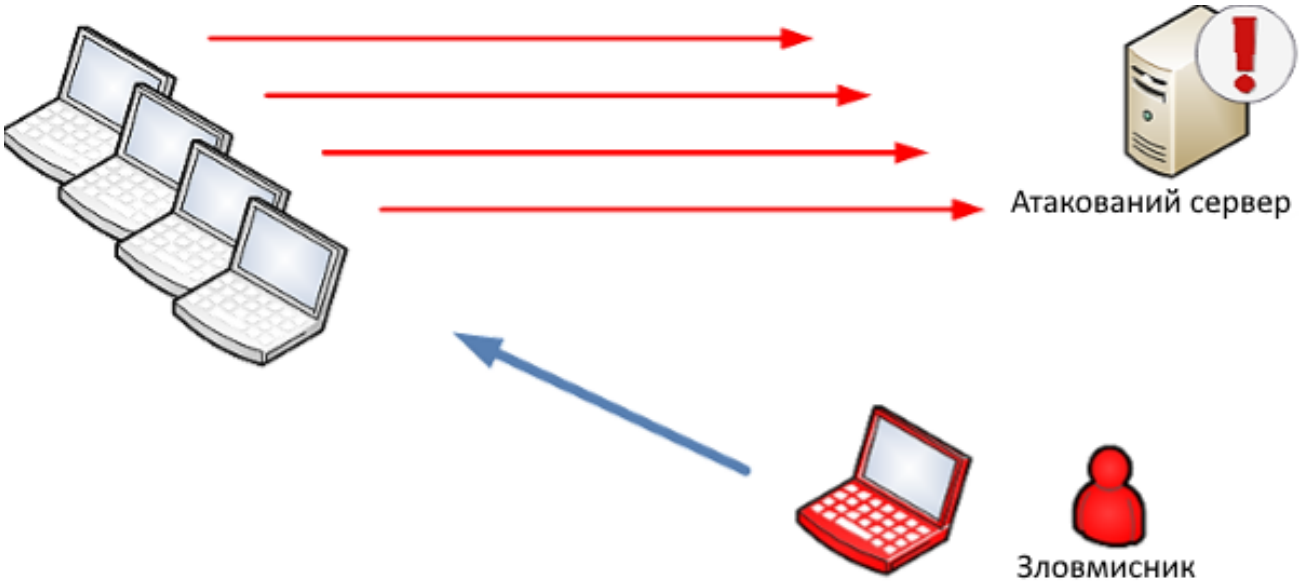


Рис. 1.1 ICMP Flood

Отже, ICMP Flood та його різновид Smurf залишаються класичними, але й досі актуальними інструментами DDoS-атак. Їх простота, дешевизна та можливість використання механізму підсилення роблять ці атаки особливо привабливими для зловмисників, а їх ефективна протидія потребує комплексних методів захисту — від правильної конфігурації мережевого обладнання до застосування спеціалізованих систем фільтрації та моніторингу аномалій трафіку.

UDP Flood

Атака типу UDP Flood належить до класу DDoS-атак транспортного рівня та за своєю сутністю є близькою до ICMP Flood, однак використовує інші механізми перевантаження. Її принцип полягає у масовій відправці великої кількості UDP-пакетів на різні порти вузла-жертви. Кожен отриманий пакет змушує систему виконати аналіз, щоб визначити, яке саме застосування має обробити даний трафік. Якщо на конкретному порту немає жодного активного сервісу, система повинна сформувати ICMP-повідомлення з кодом «порт недоступний» та відправити його назад відправнику. Якщо ж порт є відкритим, пакет передається безпосередньо користувацькому застосунку, який витрачає ресурси на його опрацювання та формування відповіді.

Цей механізм створює значне додаткове навантаження на обчислювальні ресурси атакованої системи. Унаслідок атаки канальна ємність швидко

виснажується, оскільки шкідливі пакети захоплюють пропускну здатність мережі, не залишаючи ресурсів для легітимного трафіку. Саме це й є кінцевою метою атакуючого — унеможливити доступ легальних користувачів до сервісів. Для ускладнення виявлення та маскування своєї активності зловмисники часто застосовують підміну IP-адреси відправника. У такому випадку ICMP-відповіді спрямовуються на сторонні адреси, а реальні джерела атаки залишаються працездатними та прихованими.

Важливо зазначити, що UDP Flood, як і ICMP Flood, не використовує конкретних уразливостей операційних систем чи прикладних сервісів. Він базується виключно на стандартних принципах роботи TCP/IP-стека, що робить цю атаку універсальною і простою в реалізації. Саме через свою технічну простоту та ефективність вона широко застосовується у практиці зловмисників.

Окремої уваги заслуговують так звані атаки з підсиленням (Amplification). У цьому випадку початковий UDP-трафік спрямовується не безпосередньо на жертву, а на проміжні уразливі сервери, наприклад, відкриті DNS, NTP чи SSDP-сервери. Такі сервери здатні згенерувати відповідь, яка у десятки чи навіть сотні разів перевищує початковий обсяг пакета-запиту. У результаті навіть невеликий вихідний трафік з боку атакуючого може перетворитися на потужний лавиноподібний потік, що багаторазово перевищує пропускні можливості мережі жертви. Це робить UDP Amplification однією з найнебезпечніших та найскладніших для протидії технік DDoS.

Атаки на транспортному рівні

Атаки транспортного рівня здебільшого спрямовані на експлуатацію особливостей протоколу TCP, зокрема механізмів встановлення та підтримки з’єднань. Основними цілями для атакуючих є початковий етап ініціалізації TCP-сесії або таблиці активних з’єднань, які зберігають інформацію про встановлені сеанси. Найпоширенішим прикладом є атака SYN Flood.

SYN Flood

Атака SYN Flood експлуатує базовий механізм TCP — процедуру встановлення з’єднання, відому як three-way handshake («потрійне

рукостискання»). Вона передбачає три послідовні кроки: по-перше, клієнт надсилає серверу сегмент SYN (Synchronize), по-друге, сервер відповідає сегментом SYN-ACK (Synchronize-Acknowledge), і по-третє, клієнт підтверджує отримання сегментом ACK (Acknowledge). Після цього з'єднання вважається встановленим.

Ключова проблема полягає в тому, що сервер виділяє ресурси вже на етапі отримання SYN-пакета, тобто до моменту завершення підтвердження. Зловмисник може скористатися цією особливістю: він підмінює IP-адреси джерел та надсилає величезну кількість SYN-пакетів. Сервер, отримуючи їх, резервує пам'ять і ресурси для кожного нового з'єднання, але остаточного підтвердження ніколи не отримує. Як наслідок, таблиця з'єднань переповнюється, ресурси вичерпуються, і сервер стає неспроможним обробляти легітимні запити. Таким чином досягається повна відмова в обслуговуванні.

Методи захисту від SYN Flood передбачають використання спеціалізованих механізмів, таких як SYN Cookies та SYN Proxy. У випадку SYN Cookies сервер при отриманні SYN-пакета не виділяє ресурси одразу, а формує спеціальний токен, який містить інформацію про з'єднання та секретний ключ. Цей токен відправляється клієнту у складі SYN-ACK-відповіді. Лише легітимний клієнт із реальним IP зможе правильно підтвердити отримання, надіславши ACK із токеном. Лише після цього сервер виділить ресурси й встановить з'єднання. У випадку з SYN Proxy використовується проміжний вузол, який приймає SYN-запити та завершує триетапне рукостискання замість основного сервера, передаючи підтверджені сесії далі.

Однак обидва методи мають свою складність. Для формування SYN Cookies або роботи SYN Proxy потрібні значні обчислювальні ресурси та мережеві канали з високою пропускною здатністю. Сучасна інфраструктура, що позиціонує себе як захищена, повинна бути здатна обробляти трафік на рівні 100 Гбіт/с і більше. Лише в такому випадку можна ефективно протидіяти атакам SYN Flood, які у сучасних умовах можуть досягати масштабів сотень гігабіт у секунду.

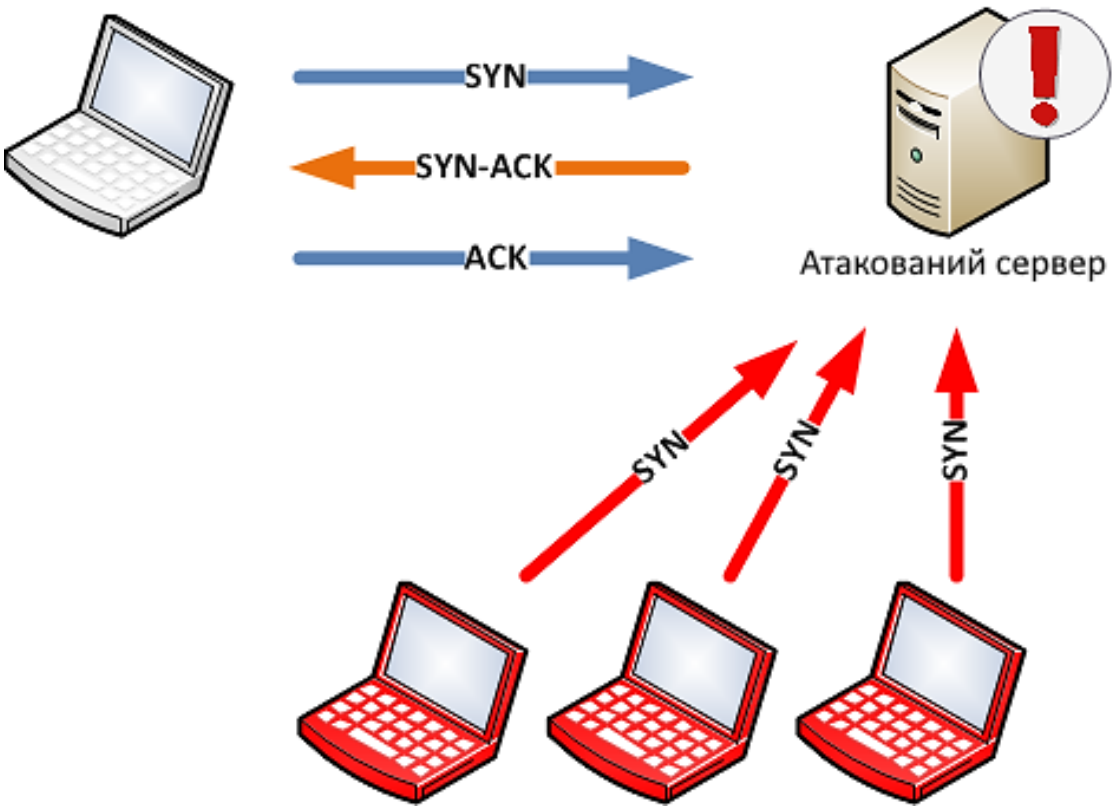


Рис. 1.2 SYN Flood

1.3 Допоміжні види атак

TCP Connection Flood

Атака TCP Connection Flood є одним із найбільш поширених різновидів перевантажувальних атак на транспортному рівні, адже вона експлуатує одне з найвразливіших місць протоколу TCP — таблицю активних з'єднань. Її особливість у тому, що зловмисник, використовуючи ботнет, може ініціювати велику кількість TCP-з'єднань з реальних IP-адрес, що робить атаку правдоподібною і значно ускладнює її фільтрацію. Для атакуючого вартість таких дій є мінімальною — кожне нове з'єднання потребує лише двох простих TCP-пакетів. Натомість для сервера-жертви наслідки є набагато серйознішими: він змушений виділяти ресурси для обробки кожного з'єднання та підтримувати їх у своїй таблиці TCP-сесій.

Поступово таблиця з'єднань починає переповнюватися, і сервер втрачає можливість створювати нові сеанси для легітимних клієнтів. Більш того, атака зачіпає не лише основні таблиці TCP-з'єднань, а й інші рівні інфраструктури:

- нижні (наприклад, таблиці відстеження з’єднань connection tracking при використанні NAT, які можуть перевантажувати маршрутизатори та міжмережеві екрани),
- верхні (ресурси, що відповідають за кількість файлових дескрипторів у HTTP-серверах чи кількість одночасно відкритих процесів).

Усе це робить атаку багатогранною: вона виснажує ресурси одразу на кількох рівнях системи й поступово знижує якість обслуговування, аж до повної відмови в роботі сервісу.

Для протидії TCP Connection Flood традиційно використовують обмеження таймаутів на встановлення та підтримку з’єднань, особливо у стані неактивності (*idle*). Практика показує, що оптимальним є значення таймауту близько 60 секунд, однак воно може змінюватися залежно від характеру роботи сервера та вимог до часу утримання з’єднання. Чим коротший таймаут — тим швидше звільняються ресурси для нових клієнтів, але при цьому зростає ризик обриву легітимних сесій.

У разі повного встановлення TCP-з’єднання і підтвердження IP-адреси клієнта ефективними є евристичні методи захисту. Вони передбачають аналіз поведінки клієнтів та виявлення нетипових патернів:

- занадто велика кількість нових з’єднань за одиницю часу,
- підозріло повільні запити або відповіді,
- нетипові переходи між станами TCP-автомата,
- незвичні інтервали активності.

У разі виявлення таких аномалій IP-адреса заноситься до «чорного списку», і подальші спроби підключення від цього джерела блокуються на певний час.

Водночас важливо враховувати технічні обмеження. Масове створення величезної кількості TCP-з’єднань із підроблених IP-адрес сьогодні є складним завданням, оскільки для завершення триетапного рукостискання потрібне підтвердження від клієнта. Проте зловмисник може відкривати тисячі з’єднань із ботнету, використовуючи реальні адреси заражених пристроїв, що робить атаку реалістичною і небезпечною.

«Чорні списки» є зручним інструментом для блокування таких клієнтів, однак їх застосування вимагає обережності. Вони ефективні саме проти атак, заснованих на повноцінних TCP-з'єднаннях, але абсолютно не підходять для протидії атакам, що не передбачають встановлення сесії, як-от SYN Flood чи ICMP Flood. Використання чорних списків у таких випадках може призвести до хибних блокувань легітимних користувачів і навіть до виснаження пам'яті системи фільтрації, що лише погіршить ситуацію.

Таким чином, TCP Connection Flood — це складний і багаторівневий вид атаки, що не лише виснажує ресурси сервера, а й створює тривале навантаження на всю інфраструктуру. Для ефективної протидії потрібне поєднання класичних методів (зменшення таймаутів, використання чорних списків, обмеження кількості одночасних з'єднань) та сучасних підходів, що базуються на аналізі поведінкових характеристик клієнтів і гнучкому налаштуванні політик доступу.

Атаки на основі протоколів SSL/TLS

У міру того як захищені з'єднання на базі протоколів SSL (Secure Socket Layer) та його сучасної версії TLS (Transport Layer Security) стали стандартом у сфері веб-комунікацій, вони привернули значну увагу з боку кіберзловмисників. SSL/TLS широко застосовуються у фінансових системах, державних електронних сервісах, хмарних платформах, електронній комерції та будь-яких веб-додатках, де потрібна конфіденційність даних, автентифікація сторін та цілісність переданої інформації. Але той факт, що ці протоколи використовують складні криптографічні алгоритми та ресурсоємні процедури, зробив їх зручним інструментом для реалізації асиметричних DDoS-атак.

Одним із найвідоміших прикладів є HTTPS Flood, при якому сервер-безпека піддається лавиноподібному потоку запитів через протокол HTTPS. Кожен такий запит виглядає легітимним, проте його обробка вимагає від сервера значно більше обчислювальних ресурсів, ніж від клієнта. Для користувача-атакуючого витрати мінімальні, тоді як сервер має виконати низку криптографічних операцій: перевірити сертифікати, згенерувати ключі, забезпечити шифрування та підтримку

сесії. Це створює асиметрію навантаження — невеликий обсяг зусиль атакуючого може призвести до багаторазового зростання витрат ресурсів на стороні жертви.

Атаки на основі SSL/TLS можуть проявлятися в кількох формах. Найчастіше вони націлені на процес встановлення сесії, коли сервер змушений виконати складні обчислення для підтвердження з'єднання та генерації ключів. Інший варіант передбачає надсилання великої кількості «сміттєвих» або некоректно сформованих пакетів, які сервер повинен перевірити та відхилити. Особливу небезпеку становить багаторазове повторне узгодження ключів (re-handshake), яке потребує значних ресурсів, а отже може швидко виснажити навіть потужний сервер.

Прикладом є атака THC-SSL-DoS. Її алгоритм надзвичайно простий, але ефективний: атакуючий ініціює встановлення SSL/TLS-сесії, після чого негайно запитує повторне узгодження ключів. Такий запит повторюється знову і знову, доки сервер не вичерпає свої ресурси. У результаті навіть один потужний домашній комп'ютер здатен паралізувати роботу веб-сервера, що використовує SSL/TLS, а кілька десятків комп'ютерів можуть вивести з ладу цілу інфраструктуру, включно з масштабними серверними фермами, які підтримують роботу міжнародних онлайн-платформ.

Важливо враховувати, що встановлення однієї SSL/TLS-сесії вимагає у 15 разів більше ресурсів на стороні сервера, ніж на стороні клієнта. Це й пояснює ефективність такого роду атак: витрати атакуючого є незначними, тоді як сервер у короткі терміни може повністю вичерпати свої обчислювальні можливості. У масштабних DDoS-кампаніях це створює ситуацію, коли навіть добре захищені системи, оснащені сучасними апаратними рішеннями, стають вразливими.

Асиметричність SSL/TLS-атак робить їх особливо небезпечними. Якщо звичайні атаки на рівні L3 чи L4 здебільшого зводяться до перевантаження каналу або споживання пропускної здатності, то атаки на рівні SSL/TLS спрямовані на виснаження саме обчислювальних потужностей сервера. У результаті виникає ситуація, коли система формально має вільний канал для передачі даних, але не здатна обробити запити користувачів через надмірне навантаження на процесор, пам'ять або криптографічні бібліотеки.

Ефективна протидія таким атакам потребує комплексного підходу. Використання лише традиційних мережевих фільтрів не дає результату, оскільки трафік виглядає легітимним і проходить усі базові перевірки. У сучасних рішеннях застосовують комбінацію заходів: балансування навантаження між кількома вузлами, оптимізацію процесів встановлення сесій, впровадження спеціалізованих алгоритмів кешування криптографічних операцій, використання Web Application Firewall (WAF) із підтримкою SSL-інспекції та глибокої перевірки пакетів. Додатково важливими стають механізми поведінкового аналізу, які дозволяють виявляти аномальні патерни у використанні SSL-з'єднань і своєчасно відсіювати підозрілий трафік.

Таким чином, атаки на основі SSL/TLS становлять одну з найсерйозніших сучасних загроз для інформаційної інфраструктури. Їх сила криється в простоті реалізації та колосальній асиметрії навантаження, що дає змогу навіть відносно слабким клієнтським пристроям створювати критичні проблеми для великих серверних систем. Це пояснює, чому сьогодні в галузі кіберзахисту саме SSL/TLS Flood розглядаються як один із головних викликів, що потребує подальших досліджень і впровадження нових методів протидії.

Атаки на веб-застосунки

Серед усіх видів DDoS-атак особливо небезпечними вважаються ті, що спрямовані на прикладний рівень, а саме на веб-застосунки. Найбільш поширеною формою є атаки на основі протоколу HTTP, які в літературі узагальнено називають HTTP Flood. Вони належать до найбільш витончених методів впливу на інфраструктуру, адже спрямовані не стільки на перевантаження каналу чи апаратних ресурсів, скільки на виснаження програмної логіки самого застосунку.

Як правило, для проведення HTTP Flood використовується ботнет — мережа заражених комп'ютерів, смартфонів чи пристроїв IoT. Проте нерідко до атак долучаються й добровольці, наприклад, у випадках, коли йдеться про хактивізм. У цьому разі атака має не лише технічний, а й ідеологічний або політичний підтекст.

Механізм дії базується на масовому надсиланні HTTP-запитів, здебільшого методів GET і POST. Кожен запит може виглядати абсолютно коректним і

легітимним, що значно ускладнює його відокремлення від реального трафіку. Більш того, атака може здійснюватися у зашифрованих TLS-сесіях, що ще більше ускладнює фільтрацію, адже інспекція шифрованого трафіку вимагає додаткових ресурсів і спеціалізованих засобів захисту.

Ключова загроза HTTP Flood полягає у створенні асиметрії навантаження. Атакуючий надсилає невеликий за обсягом запит, який для нього практично нічого не коштує. Однак сервер у відповідь змушений генерувати значно більший обсяг даних або виконувати складні обчислення (наприклад, формування HTML-сторінки, опрацювання запиту до бази даних, перевірка користувацьких сесій тощо). Таким чином навіть при наявності вільної пропускної здатності мережевого каналу та значного запасу апаратних ресурсів «вузьким місцем» стає саме прикладне програмне забезпечення. Унаслідок цього веб-застосунок або починає обробляти легітимні запити з великими затримками, або взагалі перестає функціонувати.

Особливістю таких атак є також можливість масштабування. Підміна IP-адрес джерел робить неможливим ефективне повернення відповіді до атакуючих вузлів. У результаті навантаження концентрується виключно на сервері-жертві. Таких вузлів може бути тисячі або навіть сотні тисяч, що робить атаку багатопотоковою та надзвичайно потужною.

Загрози HTTP Flood мають кілька вимірів. По-перше, це виснаження канальної ємності. Навіть якщо атака здійснюється з використанням невеликих пакетів, їхня масова кількість може заповнити канал і зробити його недоступним для легітимних користувачів. По-друге, це відмова веб-застосунку на рівні обробки запитів. Будь-який сервер чи веб-додаток має межу продуктивності, обумовлену архітектурою, параметрами конфігурації, ефективністю системи автентифікації та обмеженнями бази даних. Під час атаки ці підсистеми перевантажуються, що призводить до зриву роботи всієї системи. У результаті легітимний користувач не може отримати потрібний цифровий контент чи доступ до сервісу.

Варто підкреслити, що HTTP Flood часто використовується у поєднанні з іншими видами атак, наприклад, із TCP Connection Flood чи TLS Flood. Це робить

їх ще більш небезпечними, адже система змушена одночасно боротися з перевантаженням як мережевих, так і прикладних ресурсів.

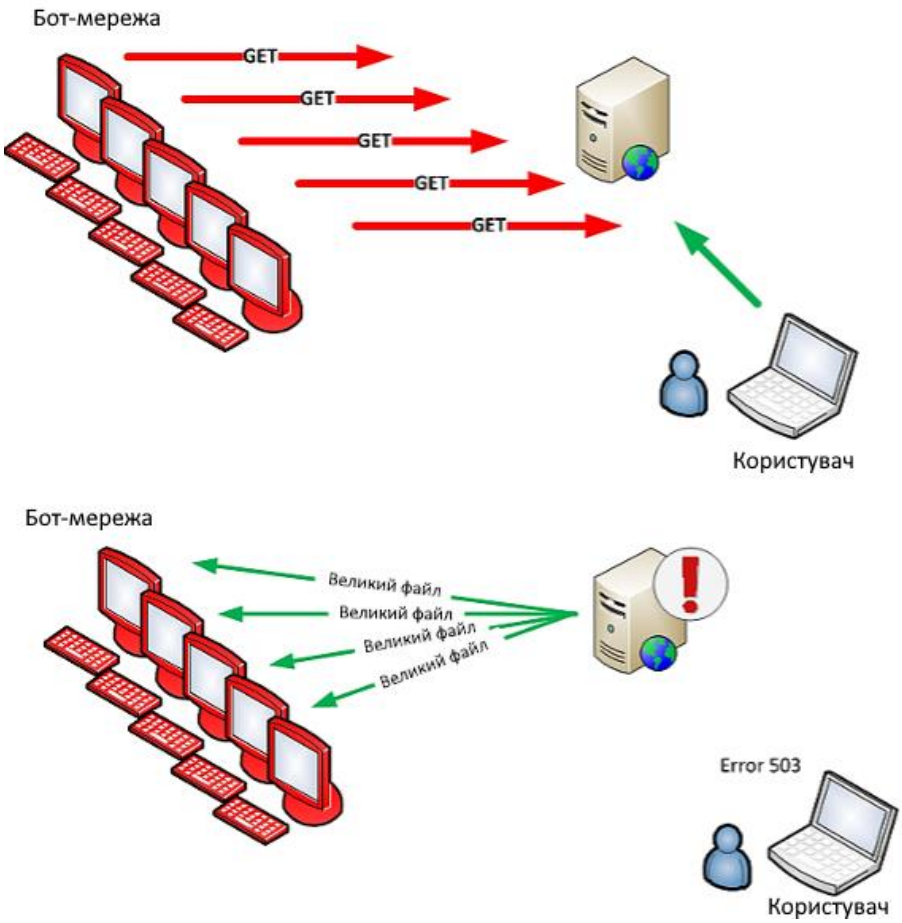


Рис. 1.3 Атаки на веб-застосунки

Таким чином, атаки на веб-застосунки є одним із найсерйозніших викликів сучасної кібербезпеки. Вони базуються на простому принципі — використати звичайний функціонал протоколу HTTP для створення непропорційного навантаження на сервер. Проте наслідки таких атак можуть бути катастрофічними: від зниження швидкодії та переривання роботи сервісу до повної його недоступності й значних фінансових втрат для організації.

HTTP-атаки із застосуванням шифрування (HTTPS Flood)

У сучасних умовах практично всі веб-застосунки працюють із використанням протоколів SSL/TLS, що забезпечують шифрування даних і гарантують безпеку їхньої передачі. Це створює додатковий рівень захисту для користувачів і сервісів, проте водночас відкриває нові вектори атак для зловмисників. Будь-яка DDoS-атака, яка здійснюється на веб-застосунок, здатний працювати із зашифрованим трафіком,

може бути запущена поверх HTTPS-з’єднання. Виявлення та нейтралізація таких атак є набагато складнішими, адже більшість класичних механізмів протидії DoS не здійснюють повноцінну інспекцію SSL/TLS-потоку. Для цього потрібне його дешифрування, що вимагає значних обчислювальних ресурсів і часто стає вузьким місцем для самого захисного обладнання. Саме тому HTTPS Flood вважається одним із найнебезпечніших і водночас найскладніших для блокування різновидів DDoS.

WordPress Pingback DDoS

До категорії відбивних (Reflection) L7-атак належить і WordPress Pingback DDoS — яскравий приклад використання архітектурних особливостей популярного веб-фреймворку WordPress. До певного часу WordPress надавав публічне API, яке дозволяло здійснювати HTTP-запити на будь-які сторонні ресурси без автентифікації. Це дало змогу атакуючим надсилати маленький запит (до 400 байт) на уразливий WordPress-сервер і змушувати його завантажувати великі веб-сторінки чи файли із цільового сервера-жертви.

У ході такої атаки частина навантаження припадає на уразливі WordPress-сайти, проте воно розподіляється між тисячами серверів. Для кожного окремого сайту це додаткове навантаження не є критичним і може навіть залишитися непоміченим адміністратором. Основний удар припадає саме на сервер-жертву, який отримує лавиноподібний потік трафіку від тисяч джерел одночасно.

Попри те, що розробники WordPress усунули уразливість, подібні атаки залишалися актуальними ще у 2021–2022 роках. Це пояснюється тим, що в Інтернеті на той час все ще існували мільйони сайтів на старих версіях WordPress, і до 350 тисяч із них могли брати участь в атаці одночасно. Оскільки WordPress зазвичай розгортається не на звичайних домашніх машинах, а на потужних серверах із великою пропускною здатністю, масштаб таких атак був вражаючим. Потік трафіку від WordPress Pingback DDoS міг досягати 10 Гбіт/с і навіть більше, що робило атаку особливо небезпечною для ресурсів із недостатньою канальною ємністю.

Ще більш складною ця атака стає у випадках, коли використовується HTTPS. Якщо запити й відповіді відбуваються всередині зашифрованих SSL/TLS-сесій, то без механізмів аналізу зашифрованого трафіку їх практично неможливо відфільтрувати. Це змушує адміністраторів застосовувати рішення, що передбачають розшифрування TLS-потоку та його глибоку інспекцію.

Необхідно зазначити, що Pingback-атаки характерні не лише для WordPress. Будь-яке застосування, яке дозволяє неконтрольоване перенаправлення запитів (через архітектурні особливості або помилки конфігурації), може стати мішенню для подібної експлуатації. Тому важливо максимально обмежувати можливості неавторизованих користувачів для роботи з API-запитами та стежити за своєчасним оновленням систем.

Full Browser Stack-атаки та імітація користувацької поведінки

Одним із найскладніших для виявлення різновидів прикладних атак вважаються Full Browser Stack (FBS)-атаки. Їхня специфіка полягає в тому, що замість використання простих бібліотек чи скриптів, які імітують запити браузера, атакуючі застосовують повноцінні браузери або високоякісні емулятори. Вони здатні виконувати весь цикл дій, притаманних реальному користувачеві: завантажувати з сервера стилі CSS, зображення, виконувати JavaScript-код, обробляти редиректи та підтримувати динамічну взаємодію.

Через це кожен окремий запит у такій атаці виглядає абсолютно легітимним. Відрізнити його від дій реального користувача традиційними методами фільтрації практично неможливо. Навіть додаткові перевірки у вигляді редиректів чи JavaScript-тестів не дають бажаного ефекту. Для виявлення FBS-атак потрібні складні статистичні й поведінкові алгоритми, що аналізують історію дій клієнтів і визначають нетипові закономірності. Це вимагає потужної інфраструктури, здатної обробляти значні масиви даних у реальному часі.

Цікаво, що такі атаки можуть виконуватися не лише ботнетами. Іноді в них беруть участь живі люди, зокрема хактивісти, що діють із політичних чи ідеологічних мотивів. У таких випадках навіть CAPTCHA або інші перевірки «людина чи робот» можуть виявитися малоефективними.

Повільні атаки малого обсягу (Slow and Low)

DDoS-атаки не завжди передбачають величезні обсяги трафіку. Існує клас так званих повільних атак малого обсягу (Slow and Low), які працюють на прикладному рівні (L7) і можуть бути здійснені навіть одним комп'ютером. Їхня небезпека полягає у використанні стандартних можливостей прикладних протоколів, але з нетиповими параметрами.

Принцип роботи полягає в тому, що зловмисник надсилає невеликий запит і підтримує з'єднання максимально довго, передаючи дані вкрай повільно або з великими затримками. У цей час сервер утримує ресурси, очікуючи завершення запиту. Якщо таких підключень буде багато, сервер швидко вичерпає свій пул ресурсів і не зможе обробляти нові звернення.

Типовими прикладами є атаки Slowloris та RUDY (R-U-Dead-Yet?). Вони не створюють великого мережевого навантаження, тому звичайні системи захисту, що орієнтуються на аномально високий трафік, тут не допоможуть. Незважаючи на малі масштаби трафіку, ефект таких атак може бути руйнівним — аж до повної зупинки роботи сервісу.

Атака типу Are You Dead Yet (RUDY)

Атака RUDY (Are You Dead Yet?) належить до сімейства «повільних» прикладних DDoS-атак і характеризується використанням цілком легітимних можливостей HTTP POST-запитів, проте в нетиповій формі. Її ефективність ґрунтується не на величезних обсягах трафіку чи сотнях тисяч одночасних підключень, а на створенні штучних умов, за яких веб-сервер вимушений витрачати значні ресурси для підтримки кожної сесії протягом максимально довгого часу.

У звичайному сценарії взаємодії користувач, наприклад, на сайті банку, онлайн-магазину чи системи бронювання квитків, заповнює форму з особистими даними й відправляє її на сервер. При цьому надсилається обмежена кількість пакетів, сесія швидко завершується, ресурси звільняються і сервер безперешкодно обслуговує інших клієнтів. У випадку з атакою RUDY ситуація розвивається інакше. Зловмисник, використовуючи спеціалізовані інструменти, розбиває дані на безліч дрібних частин і надсилає їх на сервер із довільними затримками. Кожен

пакет може містити лише один байт чи кілька символів, однак сервер, дотримуючись протоколу, продовжує вважати, що передача даних ще триває. Це змушує його утримувати з’єднання відкритим і витрачати ресурси на обробку неповного запиту.

Проблема стає критичною, коли кількість таких сесій сягає тисяч. Навіть при відсутності великого загального трафіку сервер вичерпує свій пул ресурсів: заповнюються таблиці з’єднань, використовується оперативна пам’ять, обмежується кількість файлових дескрипторів, що унеможливлює обслуговування нових легітимних клієнтів. У підсумку для справжніх користувачів сайт стає повністю недоступним.

Унікальність RUDY-атаки полягає у використанні «маскування» під легітимного користувача. Повільне надсилання даних імітує поведінку клієнта з низькошвидкісним каналом інтернет-зв’язку. Для системи це виглядає як звичайна, хоч і дещо тривала, взаємодія, тому сервер не має підстав закривати таку сесію достроково. Завдяки цьому атака є надзвичайно складною для виявлення: кожен запит формально відповідає стандарту, немає ознак підробки чи масового ботнет-трафіку.

На практиці для досягнення бажаного ефекту достатньо кількох тисяч повільних POST-запитів, щоб зупинити роботу навіть потужного сервера. При цьому атакуючому не потрібно володіти ані значними ресурсами, ані бот-мережею — іноді одного комп’ютера достатньо для паралізації критично важливого онлайн-сервісу.

Уразливими до RUDY виявляються практично всі веб-ресурси, що використовують форми введення даних: банківські портали, платіжні системи, ресурси електронної комерції, системи бронювання тощо. Це робить атаку універсальною та потенційно небезпечною для будь-якої організації.

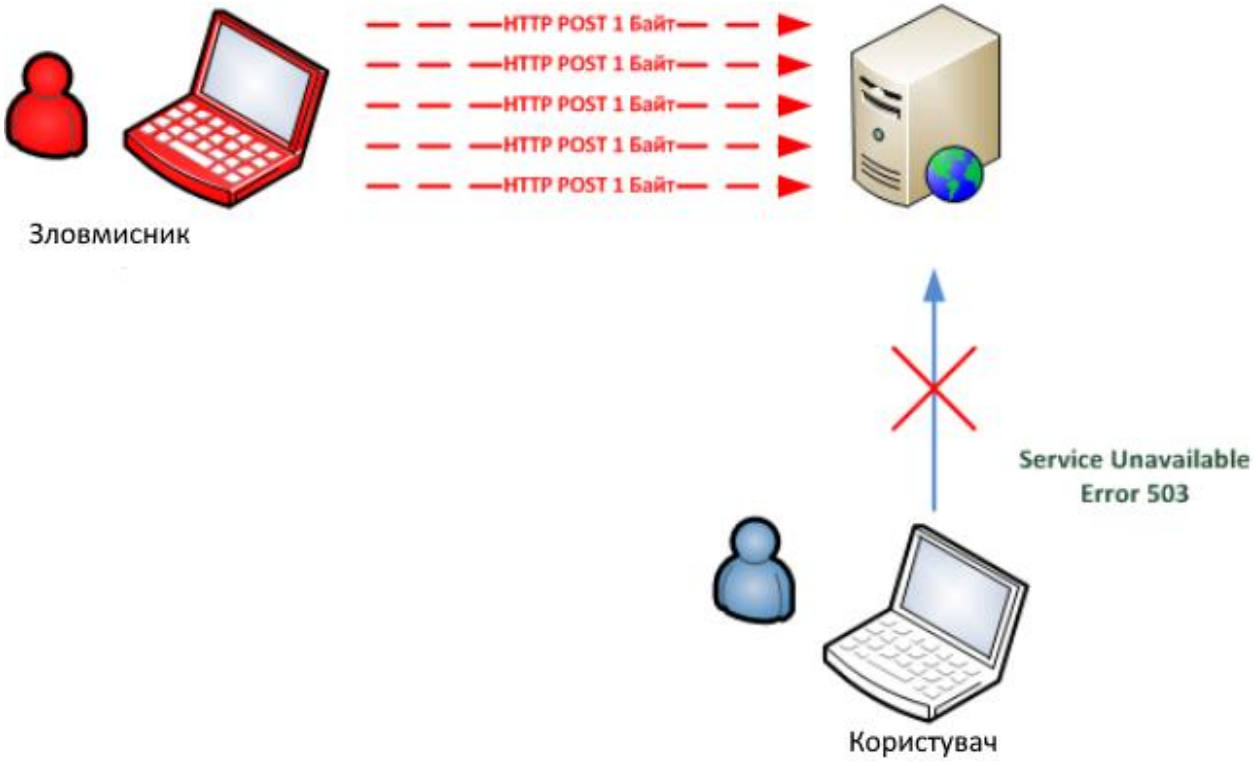


Рис. 1.4 RUDY-атака

Боротьба з RUDY-атаками є складним завданням. Традиційні методи виявлення DDoS, засновані на обсягах трафіку чи інтенсивності запитів, тут не працюють, оскільки загальний рівень навантаження може залишатися незначним. Для протидії використовують системи очищення трафіку та поведінковий аналіз клієнтів. Вони здатні виявити аномально тривалі сесії, надто повільну передачу даних чи нетипово велику кількість активних підключень від одного джерела. Водночас реалізація таких захисних механізмів потребує значних обчислювальних потужностей і ретельного налаштування.

Атака SlowLoris

Назви LOIC, HOIC, HULK, SlowLoris для досвідчених спеціалістів з інформаційної безпеки можуть звучати як відлуння минулого — вони довгий час асоціювалися з примітивними інструментами, якими користувалися початківці у сфері кіберзлочинності. Однак практика показує, що навіть такі техніки не втратили своєї актуальності. Вже у 2021–2022 роках низка авторитетних компаній, зокрема Radware, у своїх звітах відзначали реєстрацію нових хвиль атак із використанням цих давно відомих методів. Це підтверджує, що принципи їхньої роботи варто знати

й сьогодні, оскільки навіть відносно старі атаки можуть перетворитися на серйозну загрозу для веб-ресурсів, які не мають належного захисту.

Атака SlowLoris належить до класу «повільних» прикладних DDoS-атак, що експлуатують особливості стандартного функціонування HTTP-протоколу. У нормальних умовах легітимний клієнт формує GET-запит до веб-сервера, який складається з повної структури запиту та завершується спеціальною послідовністю символів, що сигналізує серверу про кінець сесії. Після цього сервер закриває з'єднання, звільняючи ресурси для обробки нових клієнтів.

Принцип атаки SlowLoris полягає у свідомому порушенні цієї логіки. Зловмисник, використовуючи однойменний інструмент або його модифікації, ініціює велику кількість HTTP-з'єднань із цільовим сервером. Але на відміну від легітимних користувачів, він не завершує запити: у HTTP-повідомленнях відсутня завершальна послідовність символів, яка дозволяє серверу закрити сесію. Таким чином сервер утримує кожне з'єднання у відкритому стані, вважаючи, що клієнт усе ще передає дані.

У міру накопичення таких «незавершених» підключень ресурси сервера швидко вичерпуються. Заповнюється пул доступних TCP-з'єднань, збільшується навантаження на оперативну пам'ять і файлові дескриптори. Коли кількість відкритих сесій перевищує можливості сервера, він перестає відповідати на нові запити. У підсумку легітимні користувачі не можуть отримати доступ до веб-ресурсу, а сам сайт стає фактично непрацездатним.

Ключова особливість SlowLoris полягає в тому, що для його реалізації не потрібен великий обсяг трафіку чи численний ботнет. Достатньо навіть одного комп'ютера середнього рівня, щоб створити сотні чи тисячі незавершених з'єднань і «покласти» вразливий веб-сервер. Це робить атаку доступною навіть для малодосвідчених зловмисників.

Захист від SlowLoris на сучасному рівні не становить значних труднощів для інфраструктури, яка має комплексні механізми протидії DDoS. Спеціалізовані системи можуть розпізнавати нетипові запити, які надсилаються з аномально повільною швидкістю або без завершальної послідовності, і блокувати їх ще на

початкових етапах. Проте для веб-серверів із базовими налаштуваннями чи застарілими конфігураціями ця атака досі може бути серйозною проблемою.

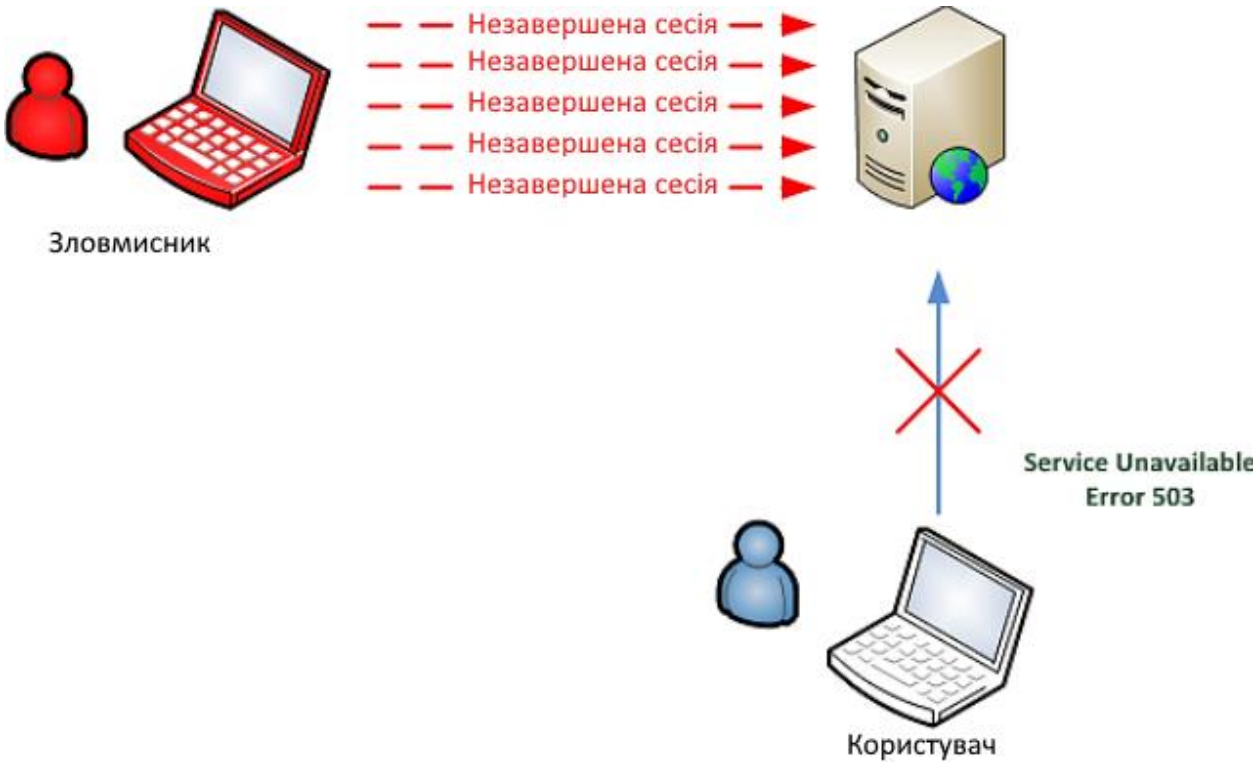


Рис. 1.5 SlowLoris атака

Небезпека SlowLoris полягає в поєднанні простоти реалізації, низьких витрат для атакуючого та ефективності проти вразливих систем. Вона не вимагає великих каналів зв’язку чи потужного обладнання, але здатна вивести з ладу навіть доволі продуктивний сервер. У поєднанні з іншими методами DDoS ця атака може стати складовою частиною більш масштабної кампанії, що робить її ще більш небезпечною.

2 КОМПОНЕНТИ ПРОТИДІЇ DDOS

2.1 Основні механізми захисту

Сучасні підходи до протидії DDoS-атакам та універсальні системи фільтрації

Як показує аналіз попередніх розділів, наслідки DDoS-атаки можуть бути вкрай руйнівними: від істотного уповільнення роботи корпоративних і публічних сервісів до повної їхньої недоступності. При цьому важливо розуміти, що об’єктом нападу може стати абсолютно будь-яка організація — незалежно від її масштабів, галузі діяльності чи рівня захищеності. Атакуючі можуть використовувати як давно відомі уразливості, так і нові, складні комбінації методів, що поєднують кілька типів впливу одночасно. Часто відбувається ситуація, коли проти одного ресурсу одночасно здійснюється декілька різних атак, що значно ускладнює завдання захисту. У випадку потужних DDoS-кампаній зловмисники зазвичай використовують ботнети, які складаються з десятків або навіть сотень тисяч заражених пристроїв, здатних створювати трафік гігантського масштабу.

З огляду на це, для забезпечення безперервної роботи легітимних користувачів і підтримання стабільної доступності сервісів критично важливо впроваджувати рішення, здатні не лише ефективно протистояти класичним DoS/DDoS-атакам, але й автоматично реагувати на так звані атаки «нульового дня», а також блокувати специфічні загрози, спрямовані на уразливості веб-застосунків.

Архітектура комплексної системи фільтрації

Гіпотетична «універсальна система протидії атакам» повинна забезпечувати автоматизований багаторівневий захист від атак, спрямованих як на мережеву інфраструктуру, так і на веб-застосунки, охоплюючи діапазон від L2 до L7 моделі ISO/OSI. Такі системи вже існують на ринку й зазвичай складаються з кількох ключових компонентів, які взаємодіють між собою для створення багаторівневої лінії оборони.

1. Апаратно-програмний комплекс для очищення трафіку (Anti-DDoS appliance).

Цей модуль є основою будь-якої системи захисту. Він поєднує апаратні й програмні засоби, спеціально оптимізовані для відбиття масованих flood-атак на рівнях L3 і L4, а також для протидії більш витонченим загрозам, таким як повільні (*Low and Slow*) атаки прикладного рівня. Сучасні комплекси повинні мати можливість обробляти і SSL/TLS-атаки, що особливо важливо в умовах масового переходу сервісів на HTTPS.

Якщо власні канальні потужності дата-центру (наприклад, до 100 Гбіт/с) є недостатніми для ефективної локальної фільтрації широкомасштабних атак, то використовується інтеграція з зовнішніми центрами очищення трафіку. Це можуть бути ресурси провайдерів зв'язку або спеціалізовані хмарні сервіси. У такому випадку весь вхідний трафік автоматично або вручну перенаправляється на зовнішній центр фільтрації, де й відбувається його очищення від шкідливих потоків, перш ніж він потрапить до інфраструктури ЦОД. Це дозволяє мінімізувати вплив атаки на внутрішні канали та сервери.

2. Web Application Firewall (WAF).

WAF є критично важливим інструментом для захисту веб-застосунків від атак на рівні L7. Його завданням є аналіз HTTP/HTTPS-трафіку в реальному часі, виявлення підозрілих або аномальних запитів та блокування спроб експлуатації вразливостей (наприклад, SQL Injection, XSS, HTTP Flood). Ринок WAF-рішень сьогодні надзвичайно широкий: від простих інтегрованих модулів у хмарних середовищах до високопродуктивних корпоративних систем із підтримкою глибокого аналізу та SSL-інспекції. Наявність WAF особливо важлива для організацій, де основна загроза зосереджена саме на прикладному рівні.

3. Система управління, моніторингу та звітності.

Цей компонент відповідає за централізоване керування всіма елементами захисної інфраструктури. Він забезпечує моніторинг у реальному часі, аналіз подій, побудову звітності й формування аналітичних моделей атак. Додатковою перевагою є інтеграція з іншими системами безпеки, зокрема SIEM (Security Information and

Event Management), що дозволяє об’єднувати дані з різних джерел і проводити кореляційний аналіз інцидентів. Таким чином, адміністратори отримують не лише засоби контролю, а й інструменти для прогнозування та запобігання подальшим атакам.

4. Служба підтримки постачальника рішення.

Нарешті, важливою складовою будь-якої захисної системи є сервісна підтримка з боку постачальника. Йдеться про цілодобовий режим роботи (24/7/365), оперативну допомогу у налаштуванні системи, консультації в реальному часі та навіть написання індивідуальних сигнатур під час складних або нестандартних атак. Цей компонент має радше організаційний, ніж технічний характер, проте його значення не можна недооцінювати: доступність фахівців значно зменшує час відновлення сервісу та забезпечує адаптивність системи до нових загроз.

2.2 Методи інтеграції компонентів захисту

Установка систем захисту «в розрив» (Inline)

Метод впровадження систем безпеки «в розрив» (Inline) передбачає розміщення обладнання для захисту від атак безпосередньо на шляху проходження трафіку. Іншими словами, будь-які дані, що надходять або виходять із корпоративної мережі, спочатку проходять через фільтраційний модуль, де відбувається їхній аналіз і перевірка, а вже після цього потрапляють до кінцевих вузлів. Такий підхід вважається одним із найбільш дієвих у протидії DoS/DDoS-атакам, оскільки дозволяє виконувати повноцінну інспекцію трафіку в режимі реального часу, оперативно блокуючи аномалії чи шкідливу активність.

Разом з тим, цей метод має й уразливі місця. Основна проблема полягає у тому, що вихід із ладу самого модуля безпеки може призвести до повної втрати доступу до трафіку — тобто фактично відключити компанію від мережі. Щоб уникнути такого сценарію, заздалегідь реалізують резервування рішень та впроваджують механізм bypass. У цьому випадку трафік може автоматично або

вручну спрямовуватися напряму, минаючи несправний вузол, що забезпечує безперервність роботи сервісів навіть у випадку збоїв.

Перевагою використання схеми «в розрив» є те, що система захисту має доступ до всіх потоків трафіку, а не лише до його вибіркових частин. Це створює умови для максимально точної аналітики, своєчасного виявлення шкідливих патернів і ефективного блокування як зовнішніх, так і внутрішніх атак. Крім того, такий підхід захищає не тільки прикладні сервіси, а й саме мережеве обладнання — маршрутизатори, міжмережеві екрани та інші вузли, які також можуть стати мішенню зловмисників.

Не варто забувати, що більшість маршрутизаторів та міжмережевих екранів використовують зовнішні «білі» IP-адреси, які доступні для виявлення за допомогою простих сканувальних інструментів. Це означає, що атакуючий може проводити попередню розвідку: визначати відкриті порти, версії операційних систем і служб, а потім експлуатувати відомі вразливості TCP/IP-стеку. Саме розташування системи «в розрив» дозволяє блокувати подібні спроби на ранньому етапі, захищаючи всю інфраструктуру.

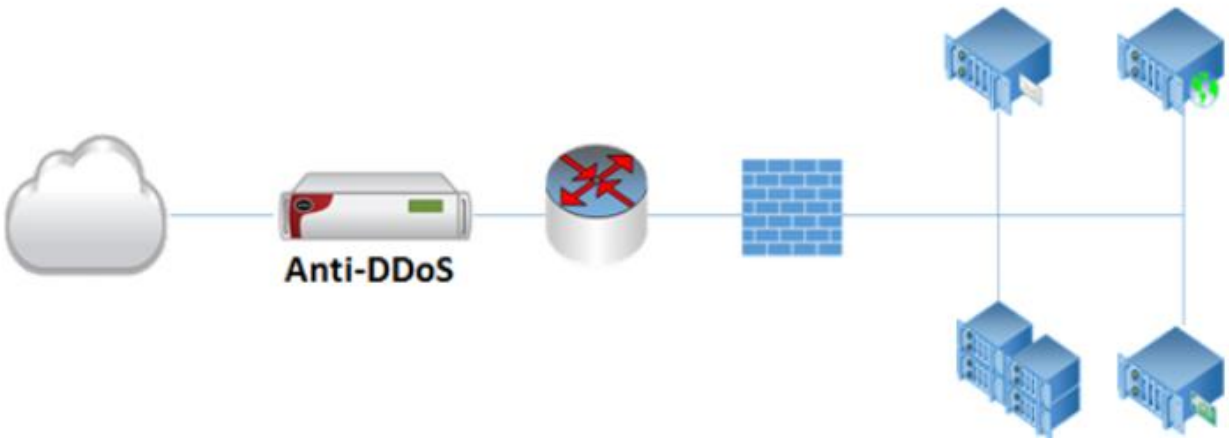


Рис. 2.1 Метод установки «в розрив» (Inline)

Алгоритм роботи модуля захисту

Робота модулів протидії DDoS у конфігурації «в розрив» зазвичай реалізується у вигляді двох взаємопов’язаних фаз, які функціонують паралельно та незалежно одна від одної:

Фаза 1 — навчання.

До початку атаки система збирає статистику та формує еталонні характеристики нормального трафіку. Це включає аналіз обсягів, частоти пакетів, характеру запитів, середніх часових затримок тощо. На основі цих параметрів формується «база нормальності», яка у майбутньому слугує точкою порівняння. При цьому важливо, що у випадку виявлення аномалії процес навчання негайно зупиняється, щоб уникнути спотворення еталонних даних.

Фаза 2 — виявлення та реагування.

На постійній основі система аналізує трафік, що проходить через неї, зіставляючи його з еталонними характеристиками. Якщо виявлено аномалію, система автоматично визначає її джерело і створює динамічні фільтри, що адаптуються під конкретний тип атаки. Такі фільтри блокують лише шкідливі пакети, дозволяючи легітимним користувачам продовжувати роботу з сервісами (за умови, що сам канал не перевантажений). Процес очищення триває доти, доки система не зафіксує стабільну відсутність аномалій.

Після завершення атаки автоматично відновлюється призупинений процес навчання, що дозволяє системі пристосовуватися до нових змін у мережі та підтримувати актуальність контрольних характеристик. Таким чином, ефективність захисного рішення з часом лише зростає.

Метод «Out-of-Path»

Одним із варіантів розгортання систем протидії DDoS-атакам є метод «Out-of-Path», тобто встановлення засобів захисту поза прямим шляхом проходження трафіку. На відміну від моделі «в розрив» (Inline), у цьому випадку весь потік даних не проходить постійно через фільтраційний модуль. Трафік перенаправляється до системи очищення лише тоді, коли виникає потреба у виявленні чи відбитті атаки. Подібний підхід найбільш часто використовується саме телекомунікаційними провайдерами, які обслуговують велику кількість клієнтів, тоді як для корпоративних центрів обробки даних (ЦОД) він застосовується рідше через низку обмежень.

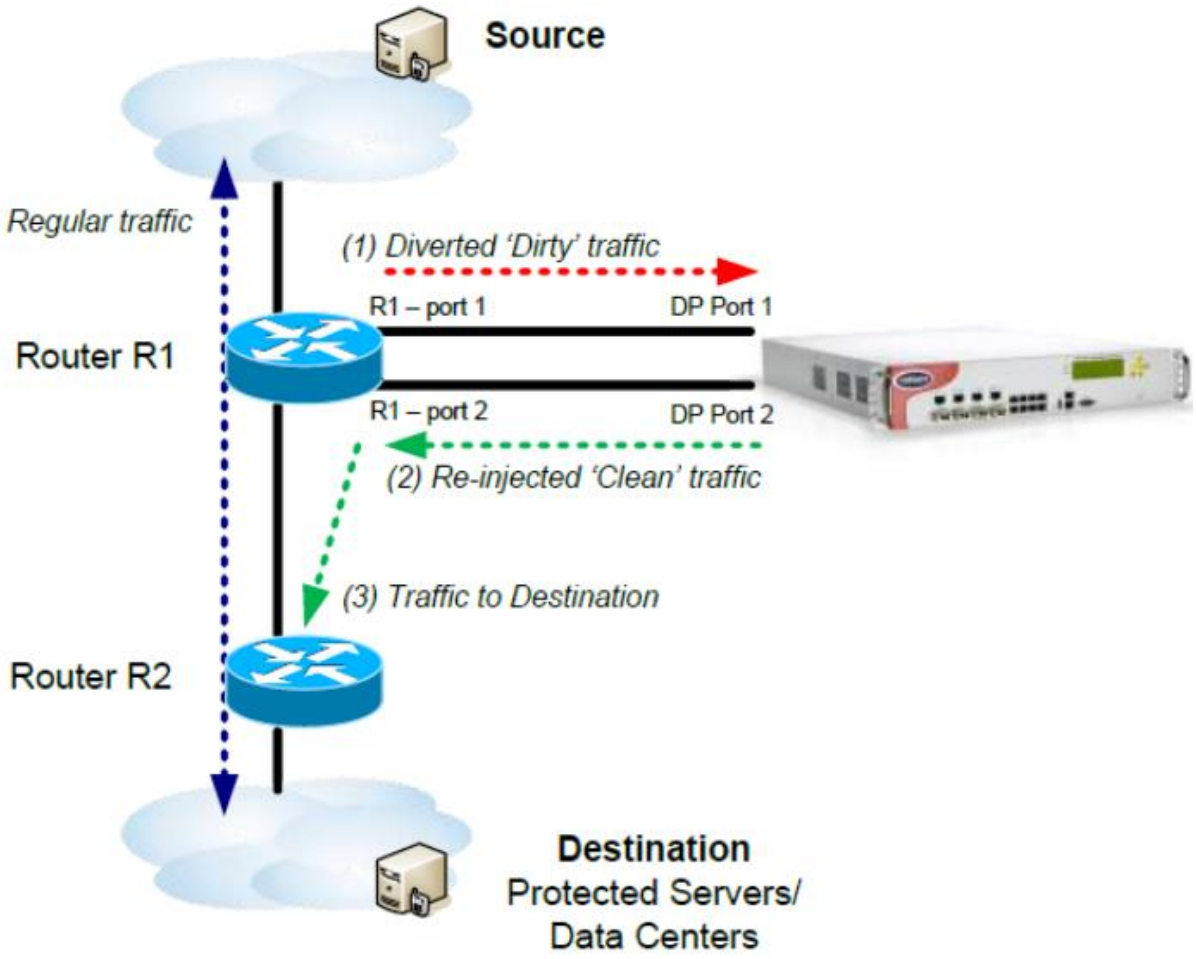


Рис. 2.2 Метод установки «Out-of-Path»

Принцип роботи Out-of-Path

У роботі такого рішення зазвичай виділяють дві ключові фази:

- **Фаза 1.** Маршрутизатор організації перенаправляє увесь вхідний трафік на обладнання очищення.
- **Фаза 2.** Пристрій фільтрує шкідливі потоки й повертає «чисті» дані на наступний маршрутизатор, після чого трафік надходить до кінцевих ресурсів.

Важливо, що два маршрутизатори з'єднані між собою таким чином, щоб у будь-який момент можна було увімкнути режим bypass. У цьому випадку трафік піде напряму, минаючи систему захисту, наприклад, якщо остання вийде з ладу або потребує технічного обслуговування.

Недоліки методу

Попри очевидну гнучкість, рішення Out-of-Path має низку недоліків, які обмежують його ефективність у корпоративних середовищах:

1. **Затримка виявлення та фільтрації атак.**

На відміну від Inline-рішень, які здійснюють перевірку безпосередньо в потоці, Out-of-Path потребує додаткових сенсорів або детекторів. Це означає, що атака буде виявлена із певним запізненням, що може бути критичним у випадку високошвидкісних DDoS-кампаній.

2. **Потреба у перенаправленні трафіку.**

   При виявленні загрози весь трафік необхідно спрямовувати до центрів очищення, що створює додаткові затримки та навантаження. Деякі організації вирішують проблему за рахунок постійного пропускання даних через центри очищення, однак це призводить до підвищених витрат і зниження продуктивності.

3. **Велика частка ручного управління.**

   Багато Out-of-Path-рішень не мають повністю автоматизованого механізму захисту й покладаються на дії адміністратора. Це знижує швидкість реагування у випадку багатовекторних атак.

4. **Нездатність аналізувати певні типи атак.**

   Найбільш критичним обмеженням є неможливість ефективно фільтрувати SSL/TLS-атаки, у тому числі широкосмугові Pingback DDoS. Причина криється у використанні сучасними веб-сервісами сесійних шифрів із підтримкою властивості Perfect Forward Secrecy (PFS), зокрема ефемерного протоколу Диффі-Геллмана. PFS гарантує, що після узгодження ключа між клієнтом і сервером цей ключ неможливо відновити навіть маючи приватний ключ сервера. Це робить дешифрування трафіку на рівні Out-of-Path практично неможливим.

2.3 Підходи до архітектури системи

Комплексний підхід до розгортання систем протидії атакам та вторгненням

Захист сучасних корпоративних інформаційних систем від DDoS-атак та інших кібератак вимагає не застосування поодиноких інструментів, а впровадження

комплексної багаторівневої архітектури безпеки. Лише інтеграція різних засобів і технологій дозволяє досягти ефективності, адже загрози постійно розвиваються й комбінують різні вектори впливу — від масових flood-атак на рівнях L3/L4 до витончених атак прикладного рівня (L7), спрямованих на конкретні веб-застосунки.

У такому підході модуль захисту від мережевих атак розташовується перед прикордонним маршрутизатором дата-центру (ЦОД). Це забезпечує відсікання значної частини паразитного трафіку ще до того, як він зможе вплинути на продуктивність серверів чи мережевого обладнання. Додатковою перевагою є можливість захисту самого прикордонного міжмережевого екрана, який теж може бути ціллю атак. У випадку використання зовнішніх або хмарних рішень ця вимога реалізується автоматично, адже очищення відбувається на стороні провайдера або в центрах очищення.

Засоби протидії прикладним атакам (HTTP Flood, Slowloris, RUDY та ін.) рекомендується розміщувати після модуля мережевого захисту, що створює додатковий бар’єр і дозволяє відсікати ті загрози, які обходять базові механізми фільтрації.

Окрему роль відіграє IPS (Intrusion Prevention System — система запобігання вторгненням). Її основне завдання полягає у виявленні й нейтралізації атак із мінімальним трафіком, але високим рівнем загрози, а також у блокуванні спроб експлуатації відомих уразливостей у протоколах, сервісах та операційних системах. IPS у комплексі з іншими модулями виконує роль «інтелектуального фільтра», здатного реагувати на поведінкові аномалії та підозрілі патерни взаємодії.

Особливе місце займає WAF (Web Application Firewall), який відповідає за захист веб-застосунків від атак прикладного рівня, включно з тими, що входять до переліку OWASP TOP-10 (SQL Injection, XSS, CSRF тощо). WAF може інтегруватися як у зв’язку з прикладним модулем, так і встановлюватися безпосередньо перед веб-серверами. У другому випадку забезпечується точковий контроль і захист найуразливіших ресурсів компанії.

Завдяки такій архітектурі формується багаторівнева система захисту, здатна ефективно протидіяти майже всім відомим типам атак. Єдиним винятком

залишаються загрози, пов’язані з протоколом BGP, для яких на сьогодні ще не створено надійних універсальних методів захисту.

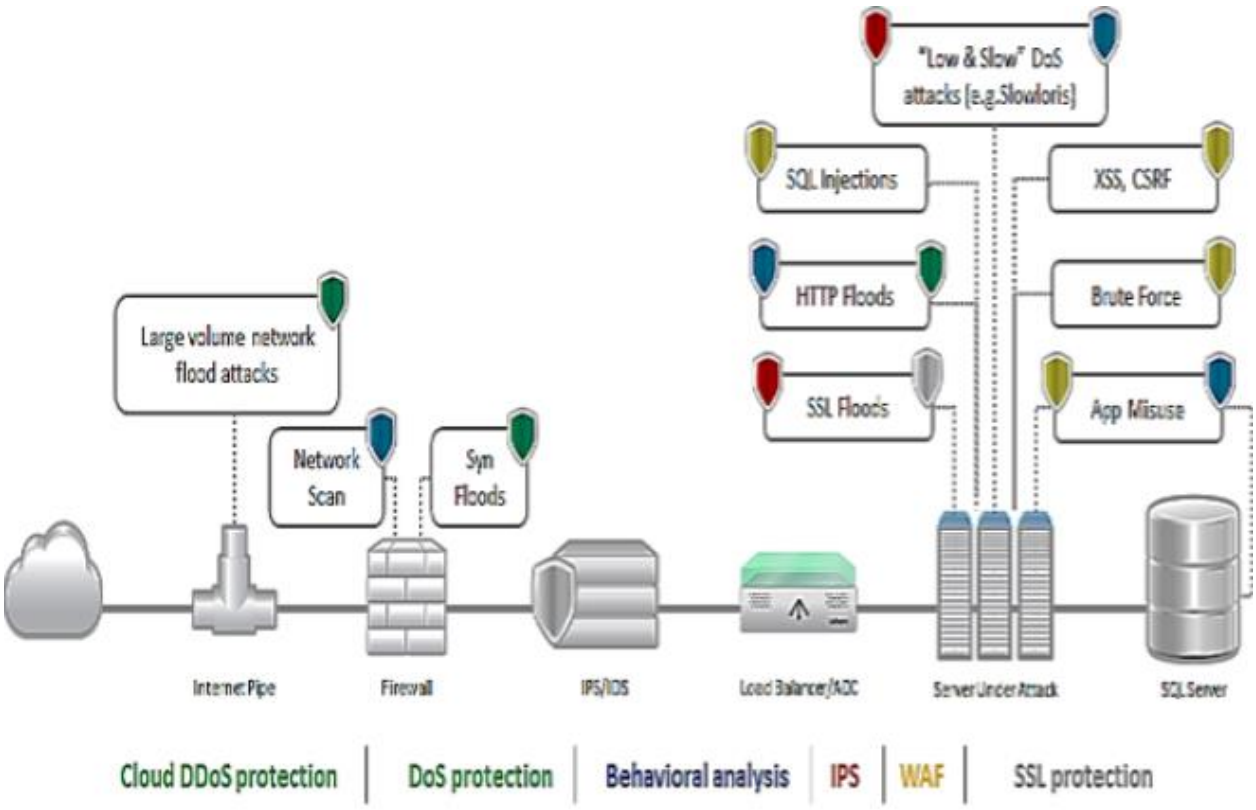


Рис. 2.3 Орієнтовна архітектура рішення захисту від DDoS

Компоненти комплексного рішення

- **Cloud DDoS protection** — хмарний сервіс для нейтралізації масових DDoS-атак за межами корпоративної мережі.
- **DDoS protection (CPE)** — локальне обладнання (Customer Premises Equipment), яке розгортається у клієнта для відсікання паразитного трафіку безпосередньо в мережі організації.
- **Behavioral analysis** — рішення на основі поведінкового аналізу, реалізовані у вигляді IPS чи WAF, що дозволяють виявляти аномалії та адаптувати захист до нових типів загроз у реальному часі.

Час реагування на атаку

Однією з ключових характеристик сучасних атак є їхня динамічність. Інструменти, що знаходяться у розпорядженні зловмисників, дозволяють змінювати вектор атаки буквально за лічені хвилини. Це стосується навіть

масштабних ботнетів, які складаються з тисяч або десятків тисяч вузлів: у будь-який момент вони можуть перебудувати свою поведінку та використати новий механізм впливу. У таких умовах використання ручних чи навіть напівавтоматичних методів протидії стає малоефективним. Швидкість виявлення й блокування атаки людиною завжди буде суттєво відставати від швидкості зміни алгоритмів деструктивних дій. Для цього потрібні значні людські ресурси — висококваліфіковані фахівці з інформаційної безпеки, доступні у режимі 24/7/365, що не завжди можливо організаційно та економічно.

Через це фільтрація вручну майже завжди виявляється несвоєчасною і неповною. Єдиний дієвий підхід полягає у застосуванні рішень, які працюють автоматично й у режимі реального часу, реагуючи на аномалії протягом кількох секунд. Система має функціонувати незалежно від втручання персоналу, виконуючи ідентифікацію та блокування атак без затримок. Лише в такому випадку захисне рішення може вважатися ефективним і дійсно здатним забезпечувати стабільність роботи сервісів під час масованих атак.

Захист веб-застосунків: виклики підтримки й супроводу

Сучасні веб-застосунки розвиваються надзвичайно швидко. Їхня головна перевага — гнучкість та динамічність, що дозволяє легко додавати нові функції, інтегрувати сторонні сервіси або масштабувати ресурси під потреби бізнесу. Проте ця ж особливість створює значні ризики. Наприклад, запуск нового розділу на сайті може принести прибуток, але якщо він не захищений коректно, це перетворюється на нову точку вразливості для атакуючих.

Для зниження ризиків сучасні рішення мають підтримувати динамічне оновлення політик безпеки. В ідеалі — у режимі реального часу, синхронно з розгортанням нових функціональних можливостей застосунку. Такий підхід реалізується двома шляхами:

- через постійний моніторинг запитів, що дозволяє відслідковувати аномалії та підозрілу активність;

- шляхом інтеграції з рішеннями DAST (Dynamic Application Security Testing), які здійснюють автоматичний пошук вразливостей у процесі роботи застосунку.

У цьому контексті важливу роль відіграє WAF (Web Application Firewall). Він має вміти автоматично відслідковувати зміни у веб-застосунку, пропонувати нові політики захисту та ініціювати DAST-сканування у випадку модифікації коду або додавання нових сервісів. Результати перевірок одразу перетворюються у звіти та рекомендації, а згодом інтегруються в політики у автоматичному або ручному режимі. Такий підхід часто називають системою віртуальних патчів, адже він дозволяє швидко «закривати» нові уразливості без очікування на випуск офіційних оновлень.

Мінімізація шкоди для легітимних користувачів

Найважливіше завдання будь-якої системи протидії атакам полягає не лише у блокуванні трафіку зловмисників, а й у забезпеченні комфортної роботи легітимних користувачів. Якщо користувач не може отримати доступ до ресурсу або робота з ним значно ускладнена, то незалежно від того, чи була атака формально відбита, фактична ціль зловмисників досягнута.

Сучасні рішення мають гарантувати, що навіть у період активної атаки сервіс залишатиметься доступним, а затримки у роботі будуть мінімальними. При цьому слід уникати методів, які надмірно ускладнюють доступ, таких як CAPTCHA або JavaScript-перевірки, що забирають ресурси клієнта й можуть відштовхнути реальних користувачів. Ці інструменти доречні лише як допоміжні, але не як основні.

Таким чином, пріоритетом є баланс: одночасний захист від атак і збереження безперервної роботи легальних клієнтів.

Моніторинг, звітність і візуалізація атак

Щоб система протидії була максимально ефективною, вона повинна включати єдину платформу управління, моніторингу та звітності. Така платформа має забезпечувати централізоване керування всіма компонентами з однієї консолі та давати можливість повної видимості атак у реальному часі.

Особливо важливою є здатність інтегрувати кілька центрів очищення трафіку та координувати їхню роботу через єдиний інтерфейс. Це знімає потребу у розгортанні багатьох систем управління та значно спрощує адміністрування.

Дані з різних пристроїв повинні збиратися у консолідовані звіти й панелі візуалізації, що дозволяє ІТ- та ІБ-фахівцям оперативно аналізувати ситуацію, виявляти закономірності та приймати рішення у стислі терміни.

Бажано, щоб система управління надавала розширений набір інструментів для контролю:

- доступності сервісів;
- продуктивності інфраструктури;
- безпеки ресурсів.

Це дозволяє не лише відбивати атаки, але й підтримувати довгострокову стабільність роботи інформаційної системи.

Комплексний підхід до організації кіберзахисту має базуватися на чотирьох ключових складових:

1. **Н миттєве автоматичне реагування** у режимі реального часу.
2. **Адаптивний захист веб-застосунків**, що оновлюється разом із розвитком сервісів.
3. **Мінімізація шкоди легітимним користувачам**, з акцентом на зручність і доступність.
4. **Централізований моніторинг та звітність**, які забезпечують прозорість і швидкість прийняття рішень.

Лише поєднання цих складових дозволяє створити надійну багаторівневу систему захисту, здатну протистояти складним багатовекторним DDoS-атакам і вторгненням у сучасному кіберпросторі.

3 ЗАХИСТ ВІД DDOS АТАК З ТОЧКИ ЗОРУ ОПЕРАТОРІВ ЗВ'ЯЗКУ

3.1 Рішення захисту для операторів

Розробка та впровадження систем протидії DDoS-атакам для операторів зв'язку має свою специфіку, яка безпосередньо залежить від архітектури магістральних мереж та функціональних можливостей мережевого обладнання. На відміну від корпоративних або дата-центрових мереж, де можна чітко визначити «периметр» захисту, у масштабних операторських мережах такого периметру в класичному розумінні не існує. Це створює додаткові труднощі при побудові рішень для моніторингу та фільтрації.

Розглянемо приклад спрощеної архітектури магістральної IP/MPLS-мережі українського оператора (USER1). У цьому випадку можна виділити три ключові типи підключень:

Upstream — вищестоящий оператор зв'язку, що забезпечує доступ до глобального Інтернету.

Peer — рівноправний оператор або великий постачальник контенту, з яким організовано прямий обмін трафіком.

Customer — клієнти автономної системи AS48210, які отримують послуги доступу або транзиту.

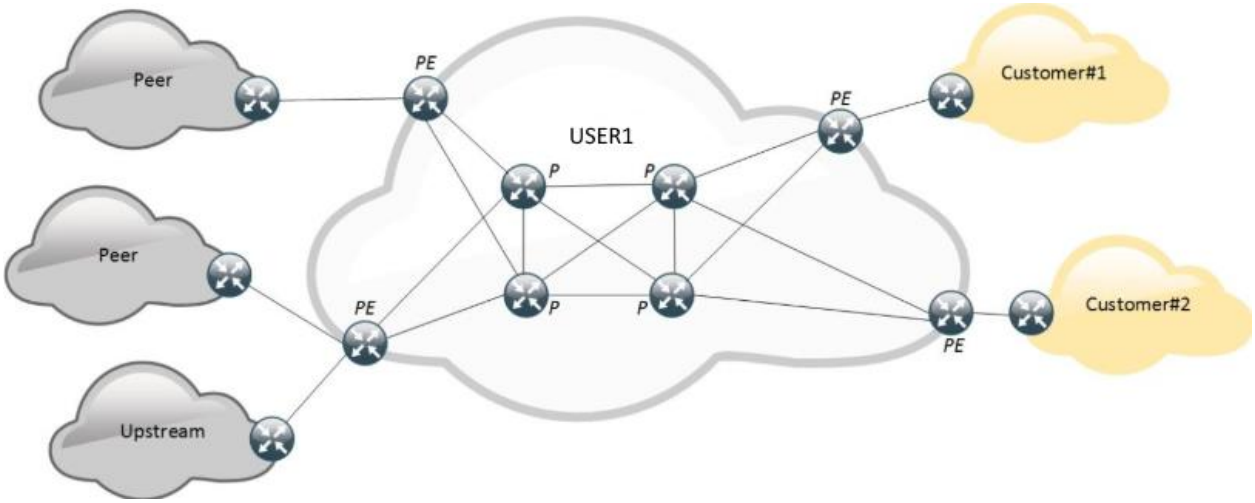


Рис. 3.1 Спрощена архітектура магістральної IP/MPLS-мережі

Якщо «накласти» цю архітектуру на карту України, стає очевидно, що мережа оператора має десятки точок стику з різними провайдерами та контент-провайдерами, а також численних клієнтів у різних регіонах. При цьому слід враховувати, що маршрутизація трафіку є асиметричною: пакети, які йдуть у напрямку до певного IP-префіксу, можуть проходити одним маршрутом, а зворотній трафік — зовсім іншим.

Це ключова відмінність від корпоративних мереж. Якщо у дата-центрі можна поставити систему моніторингу чи фільтрації на кордоні, то в операторській інфраструктурі такий підхід не працює. Тому для надійного захисту українських операторів необхідно створювати більш гнучку та масштабовану архітектуру AntiDDoS.

Архітектурний підхід: дві підсистеми

Ефективна AntiDDoS-система для оператора зв’язку складається з двох базових підсистем, які тісно взаємодіють між собою:

Підсистема виявлення аномалій.

Виконує збір статистики та аналітику мережевого трафіку в реальному часі. Завдяки розподіленому моніторингу у різних вузлах магістралі система може оперативно виявляти ознаки DDoS-атак — від різкого стрибка кількості пакетів до нетипових аномалій у використанні протоколів.

Підсистема фільтрації.

На основі виявлених аномалій запускається блокування паразитного трафіку. Це може бути реалізовано через механізми ACL, чорні списки IP-адрес, rate-limiting, перенаправлення трафіку у спеціалізовані центри очищення або навіть застосування методів поведінкової фільтрації.

3.2 Детектування DDoS-атак

Аналіз трафіку та виявлення аномалій у межах автономної системи

Для того щоб забезпечити можливість виявлення аномалій та проведення повноцінного аналізу трафіку стосовно будь-якої IP-адреси — незалежно від того,

чи є вона адресою безпосередньо самої мережі, клієнта, підключеного напряму, або ж використовується для транзитного проходження трафіку через USER1 (умовний приклад українського оператора), — необхідно мати змогу аналізувати весь масив трафіку. Це включає інформацію з усіх маршрутизаторів і кожного IP-інтерфейсу в межах автономної системи.

Якщо спробувати реалізувати таке завдання прямолінійно, без застосування спеціальних технологій оптимізації, то це призведе до величезних витрат обчислювальних і фінансових ресурсів. Саме тому оптимальним рішенням із практичної та економічної точки зору є використання протоколів мережевої телеметрії, серед яких найбільш поширеними є:

- **J-Flow (версії 5/9)** — протокол збору статистики, сумісний із NetFlow;
- **NetStream** — аналогічний механізм телеметрії, застосовуваний на обладнанні різних виробників;
- **IPFIX (IP Flow Information Export)** — сучасний та більш гнучкий стандарт, розроблений на основі NetFlow.

Для спрощення подальшого викладу всі ці протоколи об’єднують під умовною назвою **NetFlow**.

Особливості NetFlow

Протоколи родини NetFlow дозволяють операторові отримувати велику кількість статистичних даних про роботу мережі, проте вони обмежені в глибині аналізу і забезпечують інформацію лише до 4-го рівня моделі OSI. Це означає, що вони не надають даних про прикладний рівень (наприклад, вміст HTTP-запитів або SQL-команд), але водночас формують критично важливу базу для виявлення мережевих аномалій і DDoS-активності.

За допомогою NetFlow можна відстежувати такі ключові параметри:

- IP-адреси джерела та призначення,
- порти джерела та отримувача,
- використовуваний транспортний протокол (TCP, UDP, ICMP тощо),
- кількість переданих пакетів і байтів,
- тривалість з’єднання та часові мітки початку і завершення потоку.

Наприклад, структура заголовка у J-Flow v5 містить набір полів, що дозволяють чітко фіксувати ці характеристики. Завдяки цьому можна формувати детальну статистику потоків, яка у подальшому передається в аналітичні системи для виявлення підозрілих змін у поведінці трафіку.

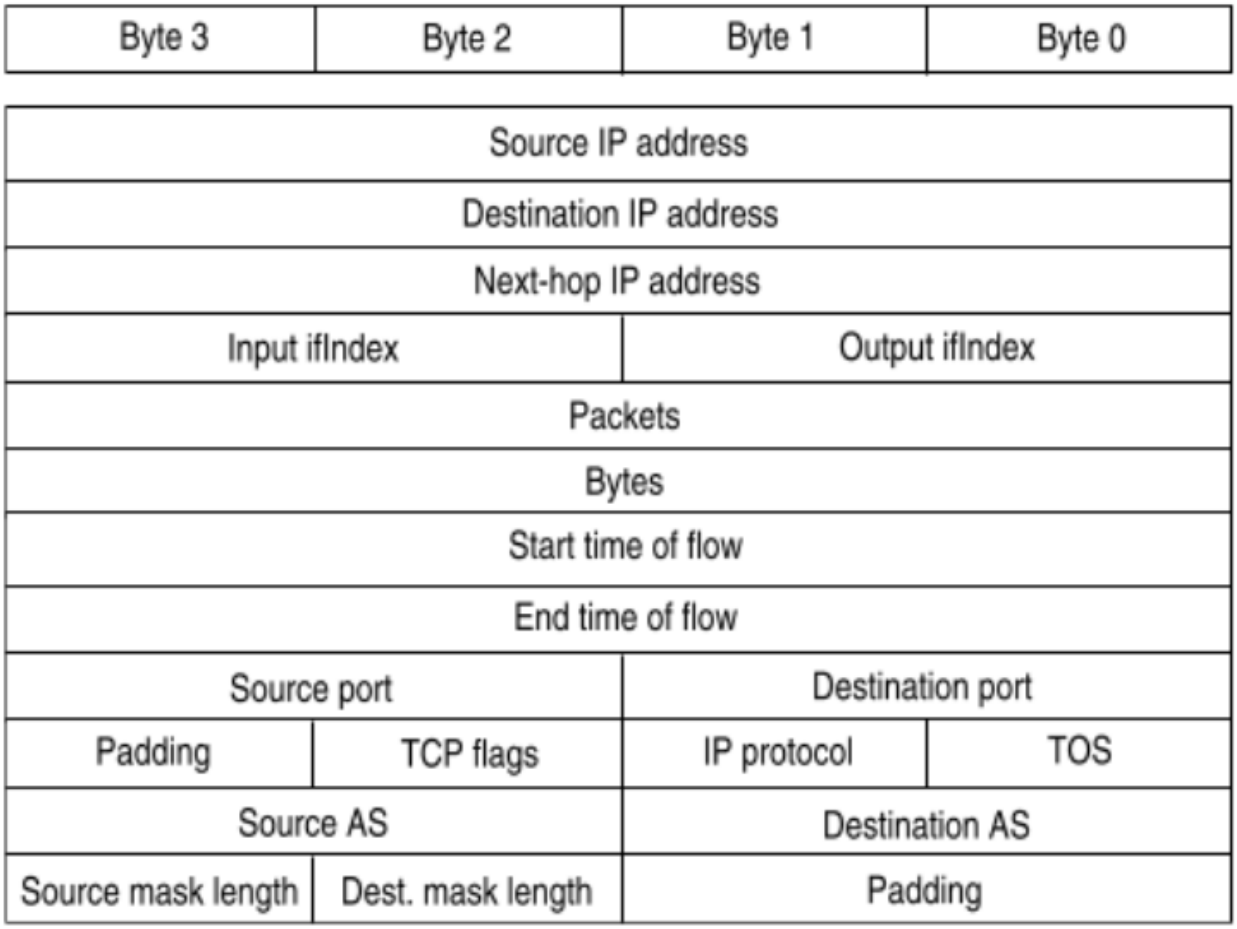


Рис. 3.2 Структура заголовку

Основні параметри, що фіксуються при зборі статистики потоків

Під час аналізу мережевого трафіку за допомогою протоколів телеметрії (NetFlow, J-Flow, IPFIX) формується детальний опис потоку, який включає набір ключових полів. Кожне поле має своє призначення і дозволяє оператору одержати вичерпне уявлення про характеристики мережевого з'єднання.

- **Source IP address** — IP-адреса джерела. Вказує на відправника трафіку, тобто вузол, який ініціював передачу даних.
- **Destination IP address** — IP-адреса призначення, що ідентифікує отримувача потоку.

- **Next-Hop IP address** — IP-адреса наступного маршрутизатора, через який трафік буде передано далі. Цей параметр допомагає зрозуміти маршрут руху потоку всередині мережі.
- **Input ifIndex** — SNMP-індекс інтерфейсу, через який маршрутизатор отримав потік. Дозволяє визначити вхідну точку руху даних у маршрутизатор.
- **Output ifIndex** — SNMP-індекс інтерфейсу, через який потік було передано далі. Це поле дозволяє простежити вихідну точку маршрутизатора для конкретного потоку.
- **Packets** — загальна кількість пакетів, що належать до даного потоку. Це поле дає змогу оцінити інтенсивність передачі.
- **Bytes** — обсяг інформації у байтах, переданий у межах потоку. Використовується для аналізу навантаження на канали зв’язку.
- **Start time of flow** — час початку потоку. Допомагає точно фіксувати момент ініціювання передавання даних.
- **End time of flow** — час завершення потоку. Використовується для оцінки тривалості сесії.
- **Source port** — порт джерела, який дозволяє визначити прикладний процес або службу, що згенерувала трафік.
- **Destination port** — порт призначення, на який надсилаються дані. Це поле дає можливість встановити, для якого сервісу або застосунку призначений потік.
- **TCP Flags** — набір службових бітів (флагів) TCP, що відображають стан з’єднання (SYN, ACK, FIN та ін.). Використовується для виявлення типів взаємодії або атак, що експлуатують TCP.
- **IP protocol** — номер IP-протоколу, який використовується для даного потоку (наприклад, TCP, UDP чи ICMP).
- **ToS (Type of Service)** — параметр, що визначає клас обслуговування пакету та дозволяє пріоритезувати трафік.

- **Source AS** — номер автономної системи, з якої походить IP-джерело. Дає змогу визначити, з якої мережі надходить трафік.
- **Destination AS** — номер автономної системи, до якої належить IP-призначення. Використовується для аналізу міжмережевих з'єднань.
- **Source Mask** — маска підмережі для IP-джерела. Визначає діапазон адрес, з яких формується трафік.
- **Destination Mask** — маска підмережі для IP-призначення. Дозволяє описати цільову підмережу.
- **Padding** — службове поле, що використовується для вирівнювання довжини заголовка, забезпечуючи ефективність його обробки.

Розширені можливості J-Flow v9 та IPFIX

Більш сучасні протоколи збору статистики, такі як J-Flow v9 і IPFIX, окрім перелічених вище параметрів, дозволяють збирати додаткові дані, які суттєво розширюють можливості для аналізу:

- **ICMP type/code** — відображає тип і код повідомлення ICMP, що важливо для діагностики та виявлення ICMP-флудів.
- **IPv6** — підтримка збору статистики для трафіку нового покоління, що ґрунтується на протоколі IPv6.
- **MPLS** — інформація про використання багатопротокольної комутації міток (Multiprotocol Label Switching), яка часто застосовується в операторських мережах.
- **BGP Peer AS** — номер автономної системи BGP-піра, що дозволяє точніше відслідковувати міжоператорські зв'язки.

Ключова відмінність між версіями NetFlow v9 та IPFIX у порівнянні з v5 полягає в їхній значно більшій гнучкості. Якщо у v5 набір полів був жорстко зафіксований і оператор міг отримувати лише обмежений набір параметрів, то у v9 та IPFIX користувач має можливість самостійно визначати, які саме поля необхідні для збору та подальшого аналізу. Це реалізується шляхом створення шаблонів, завдяки чому оператор може адаптувати процес збору статистики під конкретні завдання, наприклад, виявлення атак на транспортному рівні, оцінку навантаження

на канали зв’язку або моніторинг взаємодії між різними автономними системами. Такий підхід робить систему гнучкою і дає змогу максимально ефективно використовувати обчислювальні ресурси.

Що стосується технології NetStream, то наразі вона ще не застосовується у продуктивному середовищі, однак найближчим часом планується її інтеграція. Очікується, що використання NetStream дозволить отримувати ще ширший спектр даних та підвищить деталізацію аналізу потоків у магістральних мережах.

Візьмемо умовний приклад оператора USER1, інфраструктура якого включає понад 300 маршрутизаторів. Для такої масштабної мережі надзвичайно важливо мати централізовану систему збору та обробки статистики, тому було розгорнуто спеціалізовану інфраструктуру NetFlow-колекторів. Ці колектори здатні приймати величезні обсяги даних на високих швидкостях, обробляти їх у режимі реального часу та записувати результати у бази даних для подальшого аналізу. Враховуючи, що крізь мережу щосекунди проходять терабіти інформації, обсяг телеметрії також є колосальним: навіть при використанні механізму семування з коефіцієнтом понад 1:4000 маршрутизатори генерують більше ніж 300 тисяч NetFlow-записів щосекунди.

Механізм семування (sampling) у цьому випадку відіграє критично важливу роль. Він дозволяє значно зменшити навантаження на маршрутизатори та їхні обчислювальні ресурси. Замість того, щоб аналізувати кожен пакет трафіку, обладнання обробляє лише вибіркову частину даних, дотримуючись спеціальних алгоритмів, реалізованих вендорами. Це дозволяє суттєво знизити навантаження на Control Plane або сервісні карти маршрутизаторів, водночас зберігаючи достатній рівень точності для виявлення аномалій та атак. Завдяки такому підходу оператор отримує можливість ефективно аналізувати величезні обсяги трафіку, не створюючи надмірного тиску на свою інфраструктуру.

У результаті така архітектура дозволяє українським провайдерам формувати сучасні системи моніторингу, здатні не лише збирати статистику у масштабах цілої автономної системи, а й у режимі реального часу виявляти DDoS-атаки та інші види аномальної активності. Це створює основу для інтеграції NetFlow-даних із більш

потужними системами аналітики — SIEM-рішеннями або спеціалізованими AntiDDoS-платформами, що забезпечує комплексний рівень захисту телекомунікаційної інфраструктури.

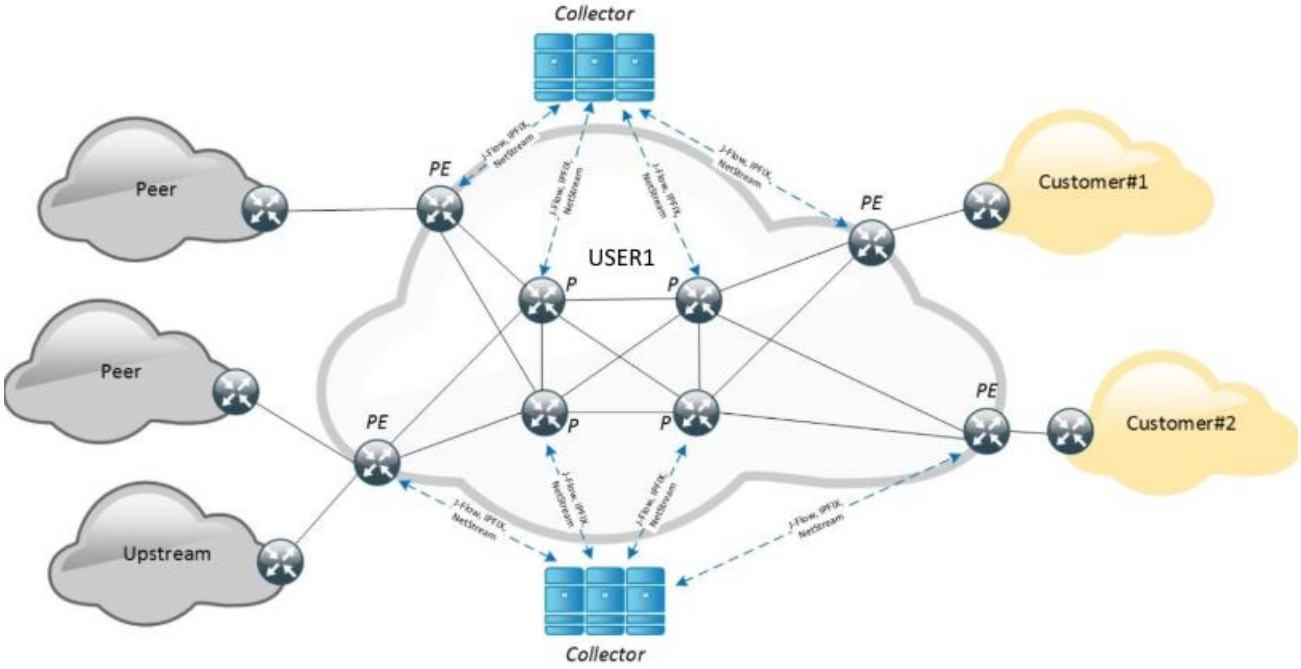


Рис. 3.3 Інфраструктура колекторів

Використання Binning Table для поглибленого аналізу трафіку.

У системах збору та обробки NetFlow-даних одним із ключових інструментів виступають так звані Binning Table — спеціалізовані таблиці, у які мапляться дані потоків та формується агрегована статистика для подальшого аналізу. Їхнє завдання полягає в тому, щоб структурувати величезні обсяги телеметрії, які надходять із сотень маршрутизаторів, і на основі цієї інформації будувати профілі поведінки трафіку.

Об’єктом захисту в межах таких систем може вважатися будь-яка сутність мережі, описана за одним або кількома критеріями. Найчастіше для цього використовуються:

- IP prefix list — списки IP-префіксів, CIDR-блоки та групи адрес, які дозволяють формувати агреговані правила контролю;
- ASN — автономні системи, причому оператор має можливість враховувати додаткові атрибути, такі як AS-Path чи community, що дає змогу розрізняти легітимний і підозрілий міжоператорський трафік;

- Мережеві інтерфейси — як фізичні, так і логічні, через які проходить трафік, що дозволяє локалізувати джерело аномалії на рівні обладнання;
- Flow Filter — складні логічні вирази для комбінації різних параметрів заголовків IP і транспортних протоколів. Наприклад: dst host 1.1.1.1 and proto tcp and dst port 80 описує трафік, спрямований на конкретний сервер із зазначенням протоколу та порту.

Поля для збору статистики

Для аналізу доступний широкий спектр параметрів, що дозволяє створити детальну картину стану мережевого трафіку. До них належать:

- Average packet lengths — середня довжина пакетів у потоці, що дозволяє виявляти відхилення, наприклад, при атаках, де використовуються специфічні розміри пакетів;
- Destination addresses — IP-адреси призначення, що допомагають визначити, на які ресурси спрямований основний обсяг трафіку;
- Destination ports — цільові порти, які вказують, які сервіси піддаються навантаженню;
- ICMP codes/types — коди та типи ICMP-повідомлень, що важливо для виявлення ICMP flood або Smurf-атак;
- Protocols — транспортні протоколи (TCP, UDP, ICMP тощо), що дозволяє виділяти характерні шаблони атак;
- Source addresses — IP-адреси джерел, необхідні для локалізації потенційних ботнетів;
- Source ports — порти джерел, які допомагають відрізнити легітимний трафік від підробленого;
- TCP flags — службові біти TCP, які можуть сигналізувати про специфічні види атак (наприклад, SYN flood);
- ToS bits — біти поля типу сервісу, які відображають пріоритетність пакетів.

Формування профілів нормальної поведінки

На основі зібраної статистики система будує динамічні профілі нормальної поведінки трафіку для кожного об’єкта захисту. Це дозволяє відстежувати характерні закономірності у роботі мережі та виявляти відхилення. У разі потреби адміністратори можуть задати статичні профілі, визначивши конкретні порогові значення для найбільш популярних сигнатур атак.

Такий підхід особливо добре працює для виявлення DDoS-атак типу Amplification (NTP, DNS, Chargen, SSDP тощо). Ці атаки легко ідентифікуються завдяки різкому перевищенню заздалегідь встановлених порогових параметрів. Якщо система виявляє, що інтенсивність або структура трафіку виходять за межі допустимих значень, вона негайно генерує повідомлення про аномалію.

Класифікація аномалій за критичністю

Для зручності реагування всі аномалії розподіляються на три рівні за ступенем небезпеки: low, medium та high.

Low-аномалії найчастіше зумовлені сплесками легітимного трафіку. Наприклад, після запуску масштабної рекламної кампанії кількість користувачів, які відвідують сайт, може істотно зрости, перевищивши середні показники.

Medium-аномалії сигналізують про ситуації, які потребують більш пильної уваги операторів, оскільки можуть бути викликані як нетиповою поведінкою клієнтів, так і початковими фазами атаки.

High-аномалії зазвичай свідчать про активну фазу DDoS-атаки або іншого цілеспрямованого деструктивного впливу і вимагають негайного реагування.

Таким чином, система дозволяє не тільки виявляти підозрілу активність, але й правильно інтерпретувати її, щоб не плутати реальні атаки з природними піковими навантаженнями. Завдяки цьому оператор може оптимізувати роботу чергових команд та сфокусувати ресурси на найбільш небезпечних загрозах.

3.3 Фільтрація DDoS атак

Перенаправлення та фільтрація трафіку після виявлення аномалії

Після того як система виявила аномалію у трафіку, який стосується захищеного ресурсу, наступним кроком є перенаправлення потоків даних на фільтрацію. Цей процес може виконуватися у ручному режимі, коли рішення приймає оператор, або у повністю автоматичному режимі, що дозволяє скоротити час реакції до секунд і мінімізувати участь людини.

У сучасних системах захисту від DDoS застосовуються кілька базових методів фільтрації:

- **Flow Specification-фільтри.** Цей метод базується на застосуванні спеціальних політик до потоків трафіку, які описуються набором правил. Це дозволяє швидко відсікати підозрілі або явно шкідливі потоки, не блокуючи легітимний трафік.
- **Black-Hole Routing.** Класичний метод «чорної діри», коли весь трафік до певного вузла або підмережі повністю відкидається. Він ефективний з точки зору простоти, але непридатний у рамках комплексного AntiDDoS-сервісу, оскільки фактично призводить до відмови сервісу для легітимних користувачів. У професійних рішеннях цей спосіб застосовується рідко.
- **Інтелектуальна фільтрація у Центрах очищення трафіку (ЦОТ).** Найбільш сучасний та дієвий метод, що передбачає перенаправлення усього трафіку на спеціалізовані апаратно-програмні комплекси, де відбувається його глибокий аналіз і очищення від аномалій.

Архітектура Центрів очищення трафіку

У розглянутій інфраструктурі оператора функціонують два гео-резервованих Центри очищення трафіку (ЦОТ). Вони реалізовані у відмовостійкому варіанті (GR + HA), що гарантує їх безперервну роботу навіть у разі відмови одного з майданчиків.

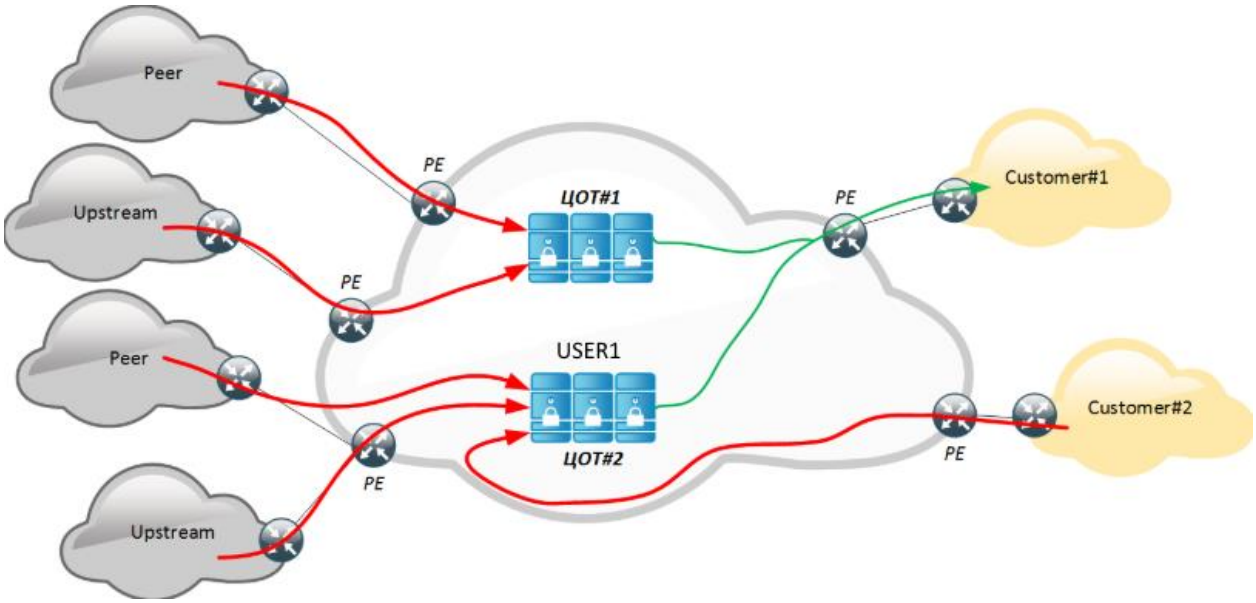


Рис. 3.4 Центри очищення трафіку

Перенаправлення трафіку відбувається за допомогою анонсування більш специфічного маршруту через ЦОТ у межах автономної системи USER1. У результаті весь трафік, включно з легітимним і паразитним, стягується у ЦОТ. Там застосовуються різноманітні методи фільтрації, після чого «чистий» трафік повертається назад у мережу клієнта. Для уникнення проблем із петлями маршрутизації використовується технологія MPLS з передачею маршрутних міток через механізм BGP Labeled-Unicast. Важливо, що достатньо один раз налаштувати обладнання, після чого клієнт отримує захист без необхідності будь-яких додаткових дій зі свого боку.

Це робить сервіс масштабованим: будь-який абонент, підключений до мережі оператора, може отримати захист без внесення змін до власної інфраструктури. Зворотний трафік при цьому маршрутизується за принципом best-path, минаючи ЦОТ. Утворюється асиметрія трафіку, яка має як переваги, так і недоліки. До мінусів належить обмеження у можливості аналізу відповідей від застосунків, а також часткове ускладнення застосування контрзаходів. До переваг можна віднести відсутність додаткової затримки для клієнтських відповідей і кращу продуктивність сервісів.

Особливості фільтрації та технічна реалізація

Наявна асиметрія маршрутизації змушує розробників системи будувати методи фільтрації таким чином, щоб вони ґрунтувалися лише на вхідному трафіку. Це складне завдання, однак сучасні технології дозволяють визначати паразитні потоки і ботів без аналізу вихідного трафіку клієнта.

Детектування аномалій зазвичай зосереджене на мережевих і транспортних рівнях, однак фільтрація здійснюється до прикладного рівня (L7) моделі OSI. Тут застосовуються як сигнатурні методи (виявлення заздалегідь відомих шаблонів атак), так і поведінкові алгоритми, здатні виявляти нетипову активність у роботі веб-додатків чи сервісів. Це особливо важливо у випадку складних DDoS-кампаній, які маскуються під легітимний користувацький трафік.

Архітектурно ЦОТ побудовані на базі спеціалізованого обладнання з використанням ATCA-платформ, що забезпечує високу продуктивність очищення навіть при роботі з багатогігабітними потоками на рівні L7. Однак технічний прогрес у сфері мережевих технологій відкрив нові можливості. Завдяки появі таких рішень, як Intel DPDK, HyperScan, сучасних мережевих карт зі швидкістю 10G і 40G, а також багатоядерних процесорів, стало можливим ефективно паралелізувати обробку потоків. Це робить систему більш масштабованою і стійкою до зростання обсягів трафіку.

У найближчій перспективі планується поступовий перехід від використання ATCA-рішень до більш гнучких та економічно вигідних серверів на архітектурі x86. Такий крок дозволить підвищити ефективність, знизити вартість утримання інфраструктури та прискорити впровадження нових функцій захисту.

Flow Specification (FlowSpec) є одним із ключових інструментів сучасного захисту від DDoS-атак, оскільки дозволяє централізовано управляти фільтрацією трафіку у великих операторських мережах. Усі маршрутизатори операторського класу мають вбудовані механізми контролю трафіку до рівня L4 моделі OSI, які зазвичай реалізуються у вигляді списків контролю доступу (Access Control List, ACL). Такі ACL виконуються апаратно на лінійних картах і здатні фільтрувати як транзитні пакети, так і пакети, призначені самому маршрутизатору, при цьому швидкість роботи практично не відрізняється від швидкості лінії (line-rate). Це

робить ACL ефективними у випадках, коли потрібно максимально швидко відсікати паразитний трафік біля джерела атаки, тобто на кордоні мережі. Однак серйозним обмеженням ACL є те, що їх потрібно налаштовувати локально на кожному маршрутизаторі. Для українських операторів, мережі яких можуть налічувати сотні маршрутизаторів, це перетворює оперативне реагування на атаку на складне завдання.

Для вирішення цієї проблеми було створено протокол BGP Flow Specification (RFC 5575), що дозволяє централізовано створювати, поширювати й видаляти правила фільтрації у масштабах усієї автономної системи. Саме він забезпечує можливість швидко реагувати на появу атак, формувати політики блокування і негайно поширювати їх на всі критичні вузли мережі. Деякі українські оператори вже впроваджують FlowSpec як сервіс для клієнтів, надаючи можливість компаніям самостійно налаштовувати захист своїх ресурсів. Це дозволяє за відносно невеликі кошти отримати базовий рівень захисту від атак, орієнтованих на перевантаження каналів зв’язку, і уникнути витрат на дорогі комплексні AntiDDoS-сервіси.

Сучасні системи інтегрують FlowSpec із платформами моніторингу та управління. Це дозволяє створювати правила безпосередньо з web-інтерфейсу або навіть автоматично, на основі зафіксованих аномалій. У результаті час реакції на атаку суттєво скорочується, а процес захисту стає менш залежним від людського фактора. Реалізація цієї технології залежить від виробника обладнання: в одних випадках правила застосовуються глобально до всіх інтерфейсів маршрутизатора, в інших — лише до вибраних, що дозволяє балансувати між рівнем безпеки та продуктивністю.

FlowSpec найчастіше використовується як перший ешелон захисту, оскільки він добре підходить для відсікання атак, які легко шаблонізуються до L4. Насамперед це UDP-based Amplification-атаки (DNS, NTP, SSDP, Chargen та інші). Такий підхід дає змогу ще на ранньому етапі «зрізати» основний обсяг паразитного трафіку й не допускати його до Центрів очищення трафіку. У результаті ЦОТ отримує вже суттєво зменшений обсяг підозрілих даних і може зосередитися на «тонкій» фільтрації складних сценаріїв атак.

Таким чином, FlowSpec є дієвим інструментом для українських операторів, що дозволяє гнучко управляти фільтрацією на мережевому рівні, підвищувати ефективність захисту від масованих DDoS-кампаній і одночасно оптимізувати використання інфраструктурних ресурсів.

4 КОМПЛЕКСНІ МЕТОДИ ЗАХИСТУ

4.1 Використання машинного навчання для протидії DDoS

Сьогодні у світі вже фіксуються реальні DDoS-атаки з піковою інтенсивністю понад 1 Тбіт/с, що ще кілька років тому здавалося майже недосяжним. Одним із найбільш відомих випадків стала атака на ресурс GitHub у 2018 році, яка досягла небаченого раніше рівня — близько 1,35 Тбіт/с. Цей інцидент став знаковим, оскільки продемонстрував новий масштаб загроз і можливості зловмисників, здатних організовувати атаки небувалої сили. Водночас, окрім простого нарощування обсягів трафіку, зростає частота використання більш витончених атак, орієнтованих на прикладні протоколи та сервіси. Подібні атаки небезпечні тим, що вони впливають безпосередньо на ключовий принцип інформаційної безпеки — забезпечення доступності інформаційних ресурсів. Якщо класичні flood-атаки виводять із ладу канали або обладнання, то атаки на рівні додатків порушують коректність роботи сервісів, що може призводити до суттєвих фінансових втрат для організацій.

У цих умовах дедалі більшої актуальності набувають технології машинного навчання (ML), які сьогодні активно застосовуються не лише в комерційних чи наукових задачах, а й у сфері кібербезпеки. Використання алгоритмів ML відкриває нові можливості для ідентифікації та відсічі DDoS-атак, оскільки традиційні методи (наприклад, сигнатурні фільтри) часто виявляються малоефективними проти динамічних та багатовекторних загроз.

Принцип дії таких систем базується на тому, що модулі виявлення DDoS-атак аналізують величезні масиви даних про мережеві з’єднання та трафік у цілому. На основі цієї інформації формується оцінка ризику для кожного конкретного ресурсу або сегмента мережі. Тут і починають працювати алгоритми машинного навчання: вони «навчаються» на попередніх випадках атак, враховують зворотний зв’язок від аналітиків або автоматичних систем і поступово вдосконалюють свої моделі.

Ключова перевага полягає в адаптивності. Якщо звичайні статичні методи фільтрації швидко застарівають, то ML-алгоритми здатні змінювати свої підходи у реальному часі, розпізнавати нові типи шкідливого трафіку та виявляти навіть невеликі відхилення від «нормальної» поведінки мережі. Це дозволяє зменшити кількість хибнопозитивних спрацювань і водночас значно підвищити точність виявлення.

Таким чином, впровадження машинного навчання у сфері протидії DDoS-атакам стає необхідною умовою для збереження працездатності інформаційних систем. Такі технології дають змогу не лише швидше реагувати на загрози, а й здійснювати прогнозування можливих атак, формувати більш стійкі політики безпеки та забезпечувати роботу критично важливих сервісів навіть у період масштабних атак.

Рівні OSI та роль ML у захисті від DDoS

DDoS-атаки можуть здійснюватися на різних рівнях моделі OSI, проте найбільш критичними для захисту є рівні з третього (мережевого) по сьомий (прикладний). На нижчих рівнях, зокрема на L3 та L4, атаки зазвичай мають більш передбачуваний і примітивний характер: це floods-потоки однакових пакетів, спрямованих на перевантаження каналу чи протоколу. У таких випадках достатньо застосувати класичні методи фільтрації — наприклад, блокування повторюваних пакетів або відсіювання трафіку за простими правилами. Це дозволяє відбити атаку без складних обчислювальних процедур.

Модель відкритих систем OSI					
	Рівень	Тип даних	Функції рівня	Особливість адресації	Приклади протоколів
Рівень хоста	7. Прикладиний	Дані (байти)	Доступ до мережевих служб	URL	HTTP(S), FTP(S), RPC, POP3
	6. Представлення		Представлення та шифрування даних		ASCII, EBCDIC
	5. Сеансовий		Управління сеансом зв'язку		PAP
	4. Транспортний	Сегменти Дейтаграми	Прямий зв'язок між кінцевими пунктами та надійність	Порт	TCP, UDP, SCTP, PORTS
Рівень зв'язку	3. Мережевий	Пакети	Визначення маршруту та логічна адресація	IP-адреса	IPv4, IPv6, IPsec, AppleTalk
	2. Канальний	Біти Кадри	Фізична адресація	MAC-адреса	PPP, IEEE 802.22, Ethernet, DSL, ARP, L2TP,
	1. Фізичний	Електричні сигнали, біти	Робота з середовищами передачі, сигналами та двійковими даними		USB, кабель (оптоволокно)

Рис. 4.1 Структурна схема OSI

Однак справжній виклик для систем безпеки постає на сьомому рівні OSI, де функціонують прикладні протоколи (HTTP, HTTPS, DNS, SMTP та інші) та відбувається взаємодія з користувачем. Саме тут атаки стають витонченішими та найнебезпечнішими, оскільки вони часто імітують легітимну поведінку користувачів. Запити можуть мати різні параметри, надходити з великої кількості ботів або навіть від справжніх клієнтів, і при цьому зовні виглядати абсолютно звичайними. З цієї причини традиційні механізми виявлення вже не дають бажаного результату.

У таких умовах на перший план виходить машинне навчання (ML), яке дозволяє аналізувати трафік не лише за формальними ознаками, а й за поведінковими патернами. ML здатне розпізнати приховані закономірності у запитах, відстежити логіку дій користувачів, виявити нетипові відхилення та відрізнити реального користувача від автоматизованої програми. Це відкриває можливість вчасно реагувати на атаки навіть тоді, коли вони майстерно замасковані під легітимний трафік.

Труднощі впровадження ML у протидію DDoS

Незважаючи на очевидні переваги, застосування ML у сфері захисту від DDoS супроводжується серйозними викликами.

По-перше, впровадження подібних рішень вимагає економічного обґрунтування. Вартість побудови й підтримки ML-систем не повинна перевищувати потенційних збитків від можливої атаки. Це робить використання ML доцільним насамперед для організацій із критично важливими або високонавантаженими ресурсами, де навіть короткочасна недоступність може призвести до мільйонних втрат.

По-друге, реалізація алгоритмів ML потребує висококваліфікованих спеціалістів, які володіють як знаннями в галузі кібербезпеки, так і досвідом у сфері штучного інтелекту та аналізу даних. Пошук і утримання таких кадрів пов’язані зі значними фінансовими витратами.

По-третє, надзвичайно складним завданням є збір якісних даних для навчання моделей. Щоб алгоритм ефективно розпізнавав DDoS-атаки, він повинен працювати саме з тими даними, що відображають реальні особливості конкретної мережі чи застосунку. Це означає, що потрібно організувати масштабний процес збору, зберігання та маркування даних, що створює додаткові труднощі.

Крім того, моделі машинного навчання потребують постійного оновлення та перенавчання, адже методи атак безперервно змінюються. Зловмисники розробляють нові тактики обходу захисту, комбінують кілька векторів атак, змінюють параметри запитів та трафіку. У результаті система захисту має перебувати у стані постійної адаптації. Це збільшує як часові, так і обчислювальні витрати, ускладнює адміністрування й вимагає високопродуктивних обчислювальних потужностей.

Сучасні рішення у сфері застосування ML для виявлення DDoS-атак

На сьогодні у науковій літературі та практичних дослідженнях описано велику кількість рішень, спрямованих на застосування машинного навчання для боротьби з DDoS-атаками. Автори різних робіт наводять найрізноманітніші результати тренування моделей, що свідчить про багатогранність підходів і водночас підтверджує актуальність цієї проблематики. Серед найбільш відомих і широко використовуваних алгоритмів можна виокремити «Дерево рішень» та

«Випадковий ліс», які відзначаються високою ефективністю в задачах класифікації трафіку.

Алгоритм «Дерево рішень»

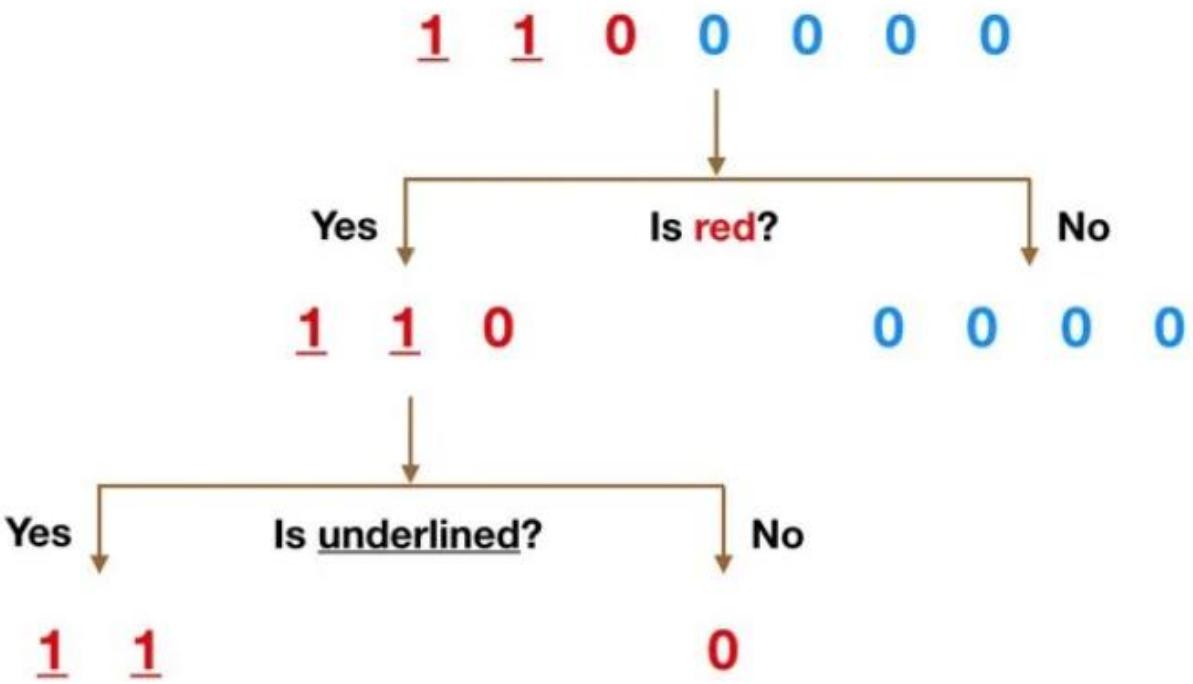


Рис. 4.2 Алгоритм «Дерево рішень»

«Дерево рішень» є одним із базових методів машинного навчання, який поєднує простоту, універсальність та зрозумілу інтерпретацію результатів. Воно створює ієрархічну структуру, що нагадує дерево, де кожна внутрішня вершина відповідає певній ознаці, а кожна гілка — можливому результату перевірки цієї ознаки. У підсумку формується набір правил if–else, які ведуть до класифікації чи прогнозування цільової змінної.

Принцип роботи алгоритму зводиться до рекурсивного розбиття даних. На кожному кроці обирається така ознака, яка найкраще відділяє дані на підгрупи. У результаті отримані групи максимально відрізняються одна від одної, а всередині кожної підгрупи елементи є найбільш подібними. Наприклад, на початковому етапі може використовуватися ознака кольору, на наступному — підкреслення, а далі інші характеристики, залежно від того, які параметри дають найкраще відділення.

Перевагою цього методу є його наочність і можливість пояснити, чому модель зробила той чи інший вибір. Водночас існує серйозний недолік: дерева

рішень мають тенденцію до перенавчання (overfitting), особливо у випадках, коли даних дуже багато або вони є складними й багатовимірними. Це обмежує ефективність їхнього застосування у задачах із динамічно змінюваними умовами, наприклад, у випадках DDoS-атак, що постійно видозмінюються.

Алгоритм «Випадковий ліс»

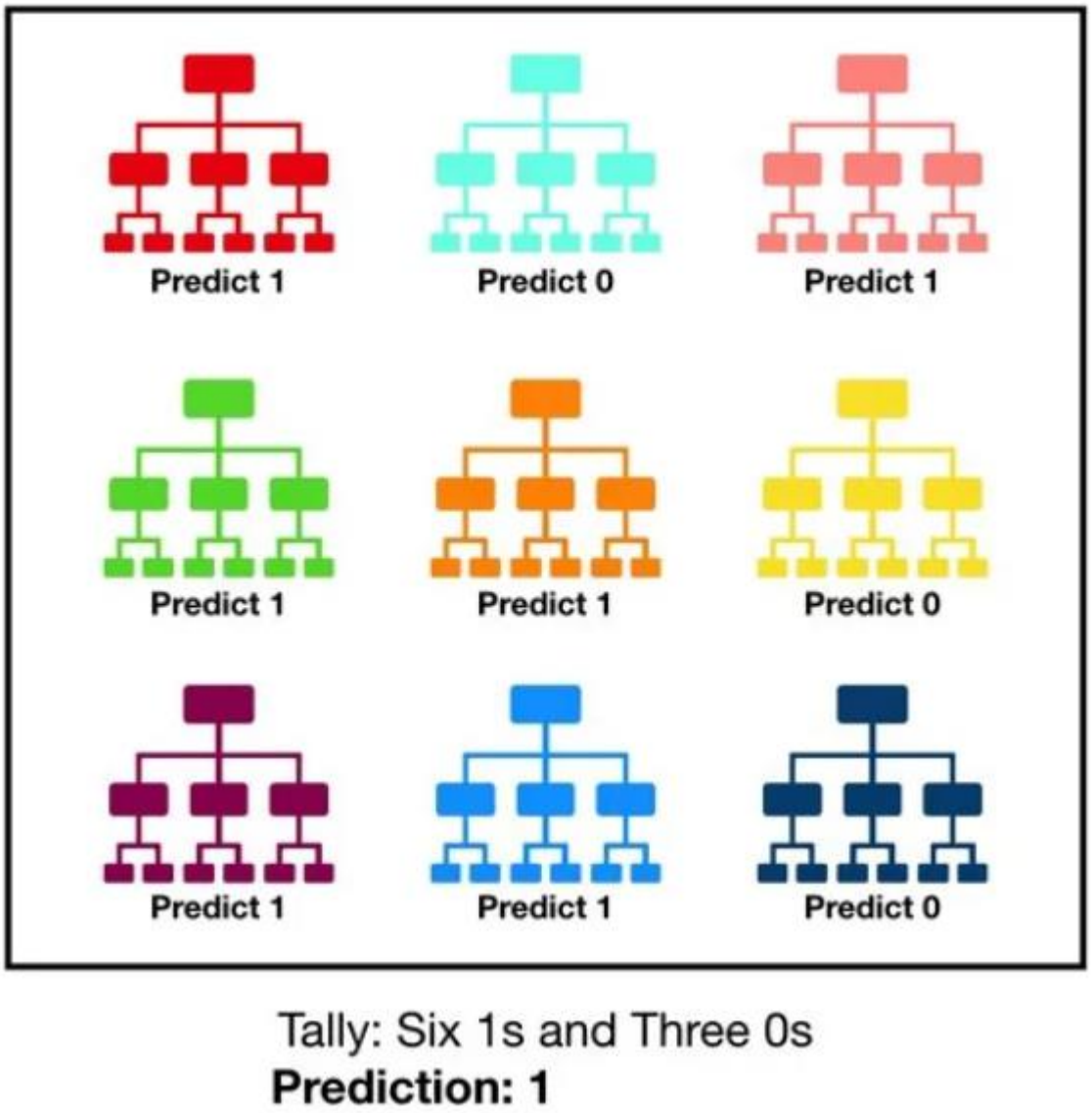


Рис. 4.3 Алгоритм «Випадковий ліс»

Більш просунутим і точним рішенням є ансамблевий метод «Випадковий ліс» (Random Forest). У низці досліджень зафіксовано, що його точність при виявленні DDoS-атак може сягати 99%, що робить його одним із найбільш перспективних інструментів у практичних системах кіберзахисту.

Принцип роботи «Випадкового лісу» полягає у створенні великої кількості дерев рішень, які працюють паралельно, але на різних підмножинах даних і ознак.

Кожне дерево будується незалежно від інших, використовуючи випадкові вибірки, а фінальний результат формується за допомогою механізму голосування: кожне дерево «віддає свій голос» за певний клас, і той клас, що отримав найбільшу кількість голосів, стає остаточним прогнозом.

Фундаментальна концепція цього підходу ґрунтується на принципі «мудрості натовпу». У випадковому лісі слабко корельовані дерева у сукупності забезпечують значно кращий результат, ніж кожне окремо взяте дерево. Це робить модель більш стійкою до шумних даних, менш схильною до переобучення та здатною узагальнювати результати навіть за складних і нестабільних умов.

Порівняння підходів

Таким чином, якщо «Дерево рішень» можна вважати простим і зрозумілим алгоритмом, придатним для базового аналізу та швидкого отримання результатів, то «Випадковий ліс» є більш потужним і надійним інструментом, який дозволяє ефективно виявляти складні закономірності у великих масивах даних. Для завдань кібербезпеки, зокрема для ідентифікації DDoS-атак, ансамблеві методи на кшталт Random Forest виявляються значно результативнішими, адже вони здатні враховувати багатовекторність атак і забезпечувати високу точність класифікації.

Завдяки цьому в сучасних системах захисту саме Random Forest дедалі частіше використовується як основа для побудови модулів виявлення аномального трафіку, тоді як дерево рішень застосовується у ролі допоміжного або навчального інструмента.

4.2 Захист від DDOS атак засобами BGP

Механізм GP Blackhole є одним із найбільш поширених та дієвих інструментів у практиці боротьби з масованими DDoS-атаками, що спрямовані на перевантаження каналів зв’язку. Його головна особливість полягає в тому, що управління трафіком здійснюється ще на стороні провайдера, тобто до того, як атака досягає мережі цільової автономної системи (AS). Це дозволяє суттєво знизити

навантаження на канали та обладнання жертви й забезпечити більш масштабний та ефективний захист.

Основна мета Blackhole — відкидання небажаного трафіку у «чорну діру» ще до того, як він потрапить у захищену мережу. У класичній схемі метод передбачає використання спеціального next-hop для маршруту — IP-адреси з приватного діапазону. У більшості магістральних провайдерів маршрути до приватних підмереж мають бути налаштовані на Null0 (у термінології Cisco) або discard (у Juniper). Це означає, що пакети, спрямовані на такі адреси, автоматично знищуються без подальшої обробки. Проте на практиці не всі оператори суворо дотримуються вимог RFC: деякі використовують приватні адреси для власних завдань маршрутизації або просто нехтують рекомендаціями.

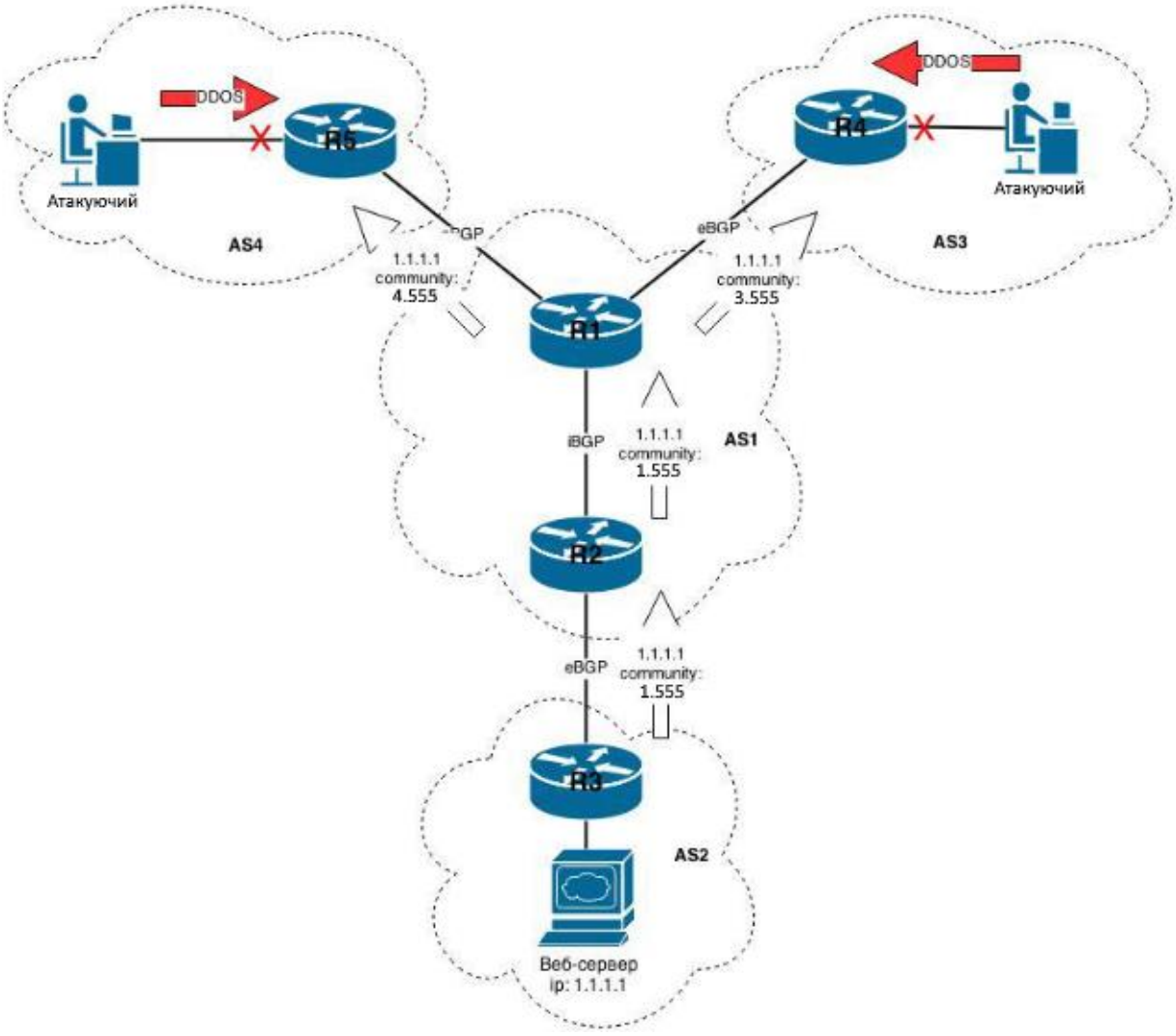


Рис. 4.4 Принцип роботи GP Blackhole

Використання BGP community

Для реалізації Blackhole зазвичай застосовують розширені можливості маршрутизації за допомогою BGP community. Суть методу полягає у створенні спеціальних міток (community), які вказують, що трафік до певного маршруту має бути «відкинутий». У разі початку атаки адміністратор атакованої автономної системи може анонсувати маршрут із довжиною маски /32, доповнивши його відповідним community. Це дає сигнал маршрутизаторам провайдера про необхідність блокування трафіку на зазначену IP-адресу.

Фільтрація може виконуватися різними методами: за допомогою списків доступу (ACL), Null-інтерфейсів або більш ефективного підходу — рекурсивного Blackhole. Саме останній варіант вважається оптимальним, оскільки він дозволяє реалізувати блокування на декількох рівнях, поширюючи «чорну діру» у напрямку до джерел атаки.

Приклад роботи рекурсивного Blackhole

Припустимо, зловмисники з автономних систем AS3 і AS4 організували DDoS-атаку на вебсервер із IP-адресою 1.1.1.1, що належить мережі AS2. Щоб зупинити атаку, адміністратор AS2 оголошує маршрут до адреси сервера з міткою community, наприклад 555, яка сигналізує про активацію Blackhole. Отримавши цей маршрут, провайдер AS1 починає автоматично відкидати всі пакети, спрямовані на 1.1.1.1.

Для подальшого зниження навантаження AS1 може передати цей маршрут своїм вищестоящим провайдерам, використовуючи їхні власні позначки community (наприклад, 3:555 і 4:555). У результаті відкидання трафіку починається ще ближче до джерел атаки, на магістральних вузлах, які безпосередньо підключають клієнтів-атакувальників.

Таким чином, механізм рекурсивного Blackhole дозволяє поступово «чистити» паразитний трафік на різних рівнях, знімаючи навантаження не лише з цільової мережі, але й із каналів магістральних операторів.

Переваги GP Blackhole

1. **Захист каналів зв’язку** — відкидання трафіку ще на стороні провайдера зменшує навантаження на канали між різними автономними системами.
2. **Масштабованість** — завдяки механізму BGP community Blackhole може поширюватися далі в мережу, що дозволяє блокувати атаки на глобальному рівні.
3. **Простота реалізації** — налаштування Blackhole вимагає мінімальних змін на стороні клієнта, оскільки основна робота виконується провайдером.
4. **Гнучкість** — можливість використовувати як локальне відкидання трафіку (на своєму маршрутизаторі), так і рекурсивне, коли фільтрація поширюється на вищі рівні інфраструктури.

Отже, GP Blackhole є ефективним методом протидії масштабним DDoS-атакам, що дозволяє значно знизити ризик перевантаження каналів і захистити критично важливу інфраструктуру. Його основна сила полягає у відкиданні шкідливого трафіку ще на магістральному рівні, що забезпечує розвантаження не лише цільової AS, але й усієї інфраструктури провайдерів зв’язку.

4.3 Захист від DDoS на рівні веб-сервера

Статистика DDoS-атак демонструє постійне зростання їхньої кількості, а також поступове зміщення вектора з мережевого рівня на рівень прикладних сервісів. Це робить проблему ще більш гострою, адже якщо раніше головним об’єктом атак були канали зв’язку чи мережеве обладнання, то тепер у фокусі опиняються веб-додатки, які набагато складніше захищати.

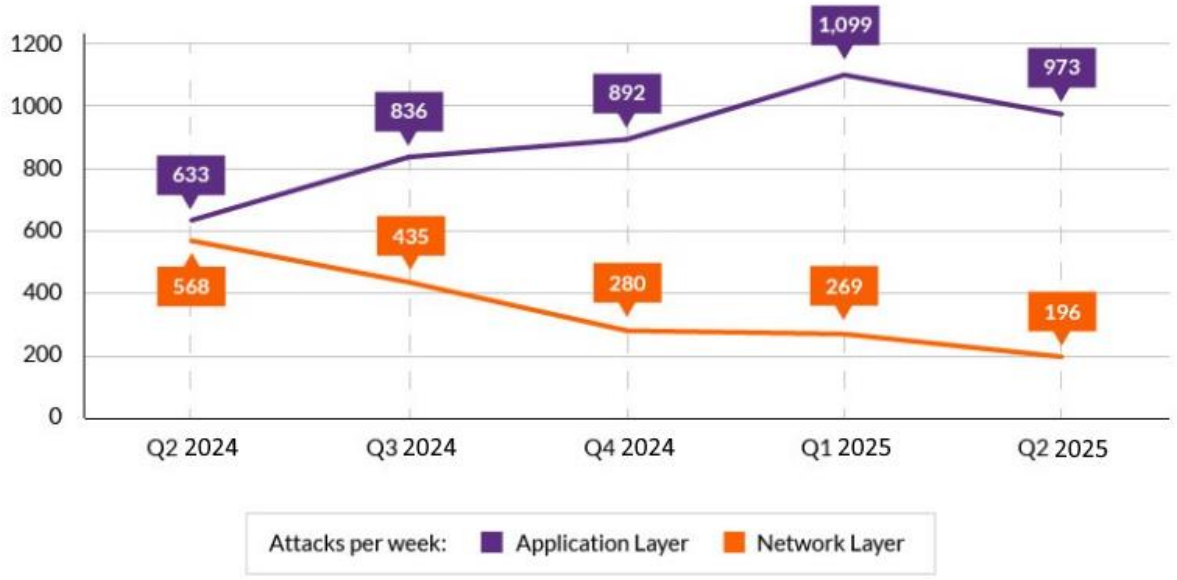


Рис. 4.5 Статистика DDoS атак

Невеликі вебсайти, що працюють на серверах із мінімальними характеристиками, особливо вразливі. Їх можна вивести з ладу навіть за допомогою легальних інструментів стрес-тестування, які створювались для перевірки стійкості ресурсів. Однак у руках зловмисників такі засоби перетворюються на ефективну зброю. При цьому варто пам’ятати, що IP-адреса сервера легко відстежується, і будь-яка спроба «експерименту» може закінчитися відповідальністю за завдання шкоди. У результаті сайт без захисту від DDoS сьогодні виглядає настільки ж безпорадним, як комп’ютер із Windows 98 без антивірусного захисту.

Першим і обов’язковим кроком у підвищенні безпеки є налаштування брандмауера iptables. Це базовий інструмент для фільтрації трафіку на сервері, який дозволяє обмежувати кількість з’єднань і блокувати небажані пакети. У практиці часто використовують готові налаштування, рекомендовані постачальниками рішень захисту від DDoS. Змінювати доводиться лише деякі параметри, наприклад, кількість допустимих одночасних підключень, щоб адаптувати правила до реальних умов роботи сервера. Водночас важливо завжди мати можливість швидко скинути iptables у початковий стан, адже невірно задані правила можуть повністю заблокувати доступ до сервера, включно для адміністратора.

Проте iptables контролює лише мережевий рівень. Наступним кроком є налаштування вебсервера, наприклад nginx, який безпосередньо обробляє HTTP-запити. У конфігураційному файлі nginx.conf потрібно підвищити ліміти на кількість відкритих файлів і з'єднань, що дозволяє серверу витримувати більше одночасних підключень. Таке налаштування не завжди повністю зупинить атаку, проте дасть змогу суттєво знизити ризик відмови у роботі сервісу та підвищити його стійкість під навантаженням:

```
# Збільшення максимальної кількості використовуваних файлів
 worker_rlimit_nofile 90000;
 events {
    # Збільшення максимальної кількості з'єднань
    worker_connections 34732;
    # Використання ефективного методу epoll для обробки з'єднань
    ...
 }
```

Наступним кроком слід сконфігурувати сервер за замовчуванням, який обмежуватиме доступ для пристроїв (наприклад, IoT), що намагаються підключатися безпосередньо за IP-адресою замість використання доменного імені:

```
    # Default server configuration
    #
    server {
       listen 100 default_server;
       listen 222 ssl default_server;
       deny  all;
    }
```

У nginx можна встановити певні обмеження на кількість запитів, проте такі правила зазвичай є надто загальними та не відрізняються гнучкістю. Наша мета — налаштувати захист на рівні вебсервера так, щоб він ефективно блокував трафік від зловмисників, але не заважав легітимним користувачам отримувати доступ до ресурсу.

Багато сучасних провайдерів послуг DDoS-захисту використовують OpenResty — серверне рішення, що об’єднує nginx та Lua (розроблене компанією Taobao). Хоча виконання коду на Lua трохи повільніше за добре оптимізований код на C, розробка на Lua набагато простіша й швидша. Важливо й те, що скрипти можна змінювати без необхідності перекомпіляції сервера: достатньо зробити перезапуск, після чого Lua-скрипти будуть зчитані, скомпільовані LuaJIT і готові до використання.

Для встановлення OpenResty існує детальна покрокова інструкція. Після інсталяції наступним кроком є налаштування nginx, де у секції http потрібно задати параметри, необхідні для роботи Lua-скриптів та подальшого гнучкого управління обробкою запитів:

lua_shared_dict whitelist 10m;
lua_shared_dict banlist 100m;

lua_package_path '/home/username/antiddos/?.lua;;';

init_by_lua '
  local whitelist = ngx.shared.**whitelist**
  whitelist:**add(**"1.2.3.4", **true)**
  whitelist:**add(**"5.6.7.8", **true)**
';

access_by_lua_file /home/username/antiddos/main.lua;

Директива **lua_shared_dict** використовується для створення словника у форматі «ключ–значення». Такий словник є спільним для всіх запитів до сервера, що робить його зручним інструментом для зберігання «білих» та «чорних» списків. Крім того, він може мати параметр **time-to-live**, завдяки якому зручно вести лічильники запитів, коли необхідно обмежити їх кількість за певний проміжок часу.

Команда **lua_package_path** задає шляхи пошуку модулів Lua і повинна включати каталог зі скриптами. Якщо наприкінці рядка вказані дві крапки з комою, це означає, що новий шлях додається до вже існуючих, а не замінює їх повністю.

Директива **init_by_lua** визначає код, який виконується один раз під час запуску сервера, а не при кожному новому запиті. Тут зазвичай задається «білий список» IP-адрес. Другий параметр функції **add** (true) виступає як значення, що використовується далі в операторі **if**. Оскільки третій параметр **time-to-live** відсутній, дані зберігатимуться без обмеження за часом.

Директива **access_by_lua_file** встановлює шлях до Lua-скрипта, що виконується для кожного запиту до сервера. У цьому скрипті зосереджена основна логіка захисту — саме тут відбувається аналіз і приймається рішення про обробку чи блокування запиту.

Далі можна розглянути приклади перевірок, які реалізуються за допомогою Lua-скриптів:

```
    -- if client IP is in whitelist, pass
local whitelist = ngx.shared.whitelist
in_whitelist = whitelist:get(ngx.var.remote_addr)
if in_whitelist then
   return
end

-- HTTP headers
local headers = ngx.req.get_headers();

-- wp ddos
if type(headers["User-Agent"]) ~= "string"
   or headers["User-Agent"] == ""
   or ngx.re.find(headers["User-Agent"], "^WordPress", "ioj") then
   ngx.log(ngx.ERR, "ddos")
   ngx.exit(444)
```

```
    return
end

local banlist = ngx.shared.banlist
local search_bot = "search:bot:count:request:per:10:s"
if ngx.re.find(headers["User-Agent"], "Google Page Speed
Insights|Googlebot|baiduspider|twitterbot|facebookexternalhit|rogerbot|linkedinbot|embe
dly|quora link
preview|showyoubot|outbrain|pinterest|slackbot|vkShare|W3C_Validator", "ioj") then
   local count, err = banlist:incr(search_bot, 1)
    if not count then
        banlist:set(search_bot, 1, 10)
        count = 1
    end
    if count >= 50 then
        if count == 50 then
            ngx.log(ngx.ERR, "bot banned")
        end
        ngx.exit(444)
        return
    end
    return
end
```

Мова Lua за своєю структурою та принципами дуже подібна до JavaScript, тому розібратися з її синтаксисом не складе труднощів тим, хто вже працює з JS.

Глобальна змінна ngx використовується для взаємодії з контекстом сервера nginx. Якщо оператор return розміщено поза межами функції, це означає вихід із модуля. Наприклад, коли IP-адреса користувача входить до «білого списку», скрипт одразу завершує виконання, а подальша обробка запиту відбувається стандартними механізмами nginx.

Додатково вбудовано механізм виявлення атак, пов’язаних із особливостями роботи CMS WordPress. У разі виявлення атаки сервер повертає спеціальний код 444 (унікальний для nginx) за допомогою команди ngx.exit(444).

Для пошукових ботів передбачено виняток, однак з огляду на те, що зловмисники часто видають себе за них, використовується лічильник. Наприклад:

- banlist:set(search_bot, 1, 10) — створює лічильник, який автоматично скидається через 10 секунд;
- banlist:incr(search_bot, 1) — збільшує значення лічильника на одиницю.

Подальша ідентифікація ботів і шкідливих клієнтів може здійснюватися за кількома критеріями: перевіркою підтримки редиректів, здатності працювати з cookies чи виконувати JavaScript-код. За потреби можна додавати й інші перевірки, орієнтуючись на специфіку вебдодатка.

У більшості випадків такий вебсервер використовується як проксі-рішення, а основний ресурс розміщується на іншій IP-адресі. Це дозволяє створити додатковий рівень захисту та мінімізувати ризик прямого впливу атак на цільовий сервер.

ВИСНОВКИ

У ході дослідження впливу DDoS-атак на телекомунікаційні мережі та методів їхнього виявлення було підтверджено, що розподілені атаки на відмову в обслуговуванні є однією з найсерйозніших загроз для сучасних інформаційно-комунікаційних систем. Вони можуть уражати як інфраструктурний рівень (L3–L4), так і прикладний рівень (L7) моделі OSI, причому з кожним роком простежується тенденція зміщення вектора атак саме на рівень додатків. Це зумовлено зростанням кількості вебсервісів та їхньою критичною роллю у функціонуванні бізнесу, фінансових організацій та державних структур.

Досліджено різновиди атак — від класичних ICMP- та UDP-flood до більш витончених методів на основі протоколів TCP, SSL/TLS та HTTP, включно з «повільними» Low & Slow-атаками, Wordpress Pingback DDoS, Full Browser Stack-атаками. Встановлено, що прикладні атаки є особливо небезпечними, оскільки вони здатні не лише перевантажити ресурси системи, а й порушити логіку роботи додатків, що призводить до значних фінансових та репутаційних втрат.

У роботі проаналізовано основні архітектурні підходи до протидії DDoS: локальні CPE-рішення, хмарні сервіси, послуги захисту від провайдерів зв’язку та гібридні моделі. Доведено, що найбільш ефективною є комплексна система багаторівневого захисту, яка поєднує фільтрацію трафіку «в розрив» (inline) та «out-of-path», застосування Центрів очищення трафіку (ЦОТ), механізмів Flow Specification, Web Application Firewall (WAF), IPS/IDS, а також систем поведінкового аналізу. Така архітектура дозволяє мінімізувати наслідки атак різних класів, зокрема й «нульового дня».

Особливу увагу приділено питанням виявлення аномалій. Розглянуто застосування протоколів мережевої телеметрії (NetFlow, J-Flow, IPFIX) та механізмів формування профілів нормальної поведінки трафіку з подальшим порівнянням і виявленням відхилень. Визначено, що одним із перспективних напрямів є впровадження алгоритмів машинного навчання (Decision Tree, Random

Forest, нейронні мережі), які здатні підвищувати точність класифікації та оперативність реагування на нові форми атак.

Проведений аналіз показав, що ефективність протидії атакам значною мірою залежить від швидкості реакції та автоматизації процесів фільтрації. Ручні або напівавтоматичні методи виявлення вже не відповідають вимогам сучасних багатовекторних атак, адже зміна тактики ботнетів може відбуватися за лічені хвилини. Тому ключовими є системи, здатні до обробки великих обсягів даних у реальному часі, до самонавчання та швидкого оновлення сигнатур.

У результаті дослідження сформульовано висновок, що боротьба з DDoS-атаками повинна ґрунтуватися на поєднанні кількох стратегій: архітектурної стійкості мережі, багаторівневих методів очищення трафіку, інтелектуальних алгоритмів аналізу даних та використання можливостей машинного навчання. Лише комплексний підхід здатний забезпечити безперервність роботи телекомунікаційних мереж, стабільність сервісів та захист інтересів легітимних користувачів.

Таким чином, DDoS-атаки залишаються динамічною та еволюційною загрозою, яка потребує постійного вдосконалення засобів захисту. Подальші дослідження у цій сфері мають бути спрямовані на розвиток автоматизованих систем моніторингу та впровадження інноваційних алгоритмів машинного навчання, що дозволить ще більш ефективно протидіяти атакам нового покоління.

ПЕРЕЛІК ПОСИЛАНЬ

1. Gupta, B.; Dahiya, A. Distributed Denial of Service (DDoS) Attacks – Classification, Attacks, Challenges and Countermeasures. CRC Press, 2021. – класифікація DDoS-атак, виклики та методи захисту.
2. Bhattacharyya, D. K.; Kalita, J. DDoS Attacks: Evolution, Detection, Prevention, Reaction, and Tolerance. Chapman & Hall, 2016. – еволюція DDoS, протидія та стратегії стійкості.
3. Amiri, I. S., et al. Theoretical and Experimental Methods for Defending Against DDoS Attacks. Elsevier, 2015. – теоретичні та практичні підходи до захисту.
4. Zhou, W.; Jia, W. Distributed Denial of Service (DDoS) Attacks: Detection, Prevention and Mitigation. Springer, 2018. – сучасні техніки запобігання і зменшення впливу атак.
5. Osanaiye, O.; Choo, K.-K. R. DDoS Attack Detection and Mitigation in Cloud and Mobile Networks. Springer, 2019. – методи для телеком і хмарних середовищ.
6. DDoS Attacks Impact on Network Traffic and its Detection Approach. 2022. – статистичний аналіз впливу на трафік та методи детекції.
7. A Comprehensive Study of DDoS Attacks over IoT Network and Their Detection. 2023. – моделі атак та ML-методи для IoT-середовищ.
8. Advancements in Detecting, Preventing, and Mitigating DDoS Attacks in Cloud Environments: A Systematic Review. 2024. – систематичний огляд захисних стратегій у хмарі.
9. DDoS Attacks Detection Using Machine Learning and Deep Learning Techniques: Analysis and Comparison. 2023. – порівняння ефективності ML та DL підходів.
10. Distributed Denial of Service (DDoS) Attack Detection in Network Traffic. 2022. – дослідження на основі CICDDoS2019 датасету.
11. Classification of DDoS Attack Traffic on SDN Network Environment Using Deep Learning. 2024. – детекція атак у SDN за допомогою DL.
12. Detection and Mitigation of DDoS Attacks Based on Multi-Dimensional Characteristics in SDN. 2024. – підхід із багатовимірними характеристиками.

13. Improving Network Resilience against DDoS Attacks: A Fuzzy TOPSIS-Based Quantitative Assessment Approach. 2024. – оцінка стійкості мережі до DDoS.
14. Distributed Denial of Service Attack and Defense Mechanisms in Various Web-Enabled Computing Platforms: Issues, Challenges and Future Research. 2022. – огляд захисних механізмів у різних платформах.
15. Network Traffic Analysis for DDoS Attack Detection. 2022. – класифікація трафіку і методи виявлення.
16. DDoS Attack and Detection Methods in Internet-Enabled Networks. 2023. – вплив на продуктивність та практичні методи детекції.
17. Detection of Distributed Denial of Service Attacks Based on Network Traffic Entropy and Time Series Models. 2020. – використання ентропії та часових рядів.
18. SDN-Based Intrusion Detection System for Early Detection and Mitigation of DDoS Attacks. 2021. – інтеграція IDS у SDN для раннього виявлення.
19. Predicting the Impact of DDoS Attacks in LTE-M Networks Using a Continuous-Time Markov Chain Model. 2024. – CTMC-модель для LTE-M.
20. High-Speed Network DDoS Attack Detection: A Survey. 2023. – огляд методів у високошвидкісних телеком-мережах.
21. Analysis of the Effects of DDoS Attacks on Multiprotocol Label Switching Networks. 2013. – дослідження впливу на MPLS-мережі.
22. Advancing DDoS Detection in 5G Networks Through Machine Learning and Deep Learning Algorithms. 2024. – ML/DL для 5G-мереж.
23. Dual-Level Attack Detection, Characterization, and Response for DDoS. 2022. – дворівнева модель виявлення та реагування (D-LAD).

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ

КВАЛІФІКАЦІЙНА РОБОТА

ДОСЛІДЖЕННЯ ВПЛИВУ DDOS-АТАК НА ТЕЛЕКОМУНІКАЦІЙНІ МЕРЕЖІ ТА МЕТОДИ ЇХНЬОГО ВИЯВЛЕННЯ

Виконала:
студентка групи ТСДМ-62
Мирослава НІГРУЦА

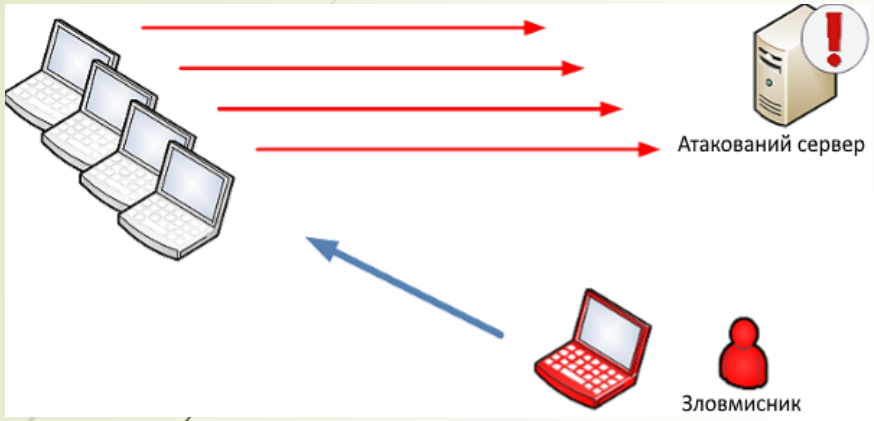
МЕТА РОБОТИ ТА АКТУАЛЬНІСТЬ

- Мета роботи – аналіз механізмів реалізації DDoS-атак у телекомунікаційних мережах та розроблення узагальненої моделі багаторівневої системи протидії, яка забезпечує своєчасне виявлення, фільтрацію шкідливого трафіку та підвищення стійкості інформаційної інфраструктури до кібератак.
- Об’єкт дослідження – процес забезпечення доступності та безперервності функціонування телекомунікаційних мереж в умовах впливу DDoS-атак.
- Предмет дослідження – методи, засоби та архітектурні рішення протидії DDoS-атакам, зокрема механізми фільтрації трафіку, системи очищення (scrubbing centers), міжмережеві екрани, Web Application Firewall (WAF) та технології поведінкового аналізу й машинного навчання.

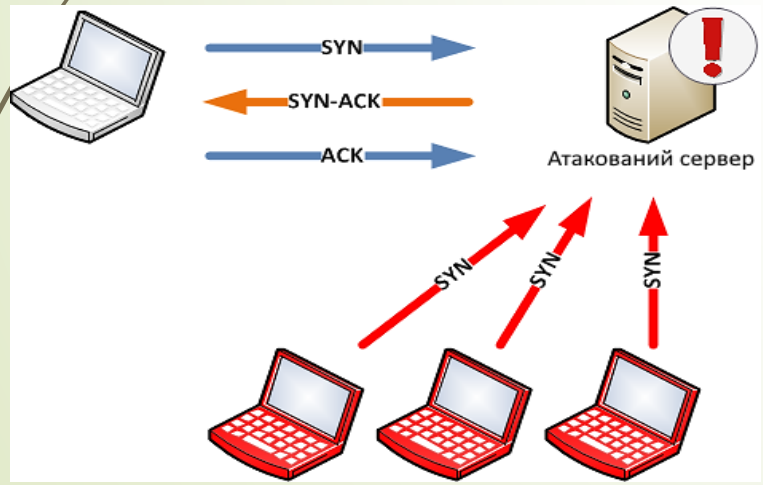
Актуальність.

- У сучасних умовах інтенсивного розвитку цифрової економіки, хмарних сервісів та критичних інформаційних систем проблема протидії DDoS-атакам набуває особливої ваги. Відмова в обслуговуванні навіть на короткий час може призвести до суттєвих фінансових збитків, втрати клієнтів і підриву репутації компаній та державних установ. З огляду на збільшення масштабів, складності та багатовекторності DDoS-кампаній, традиційні методи фільтрації трафіку вже не забезпечують достатнього рівня безпеки.

Атаки мережевого та транспортного рівнів



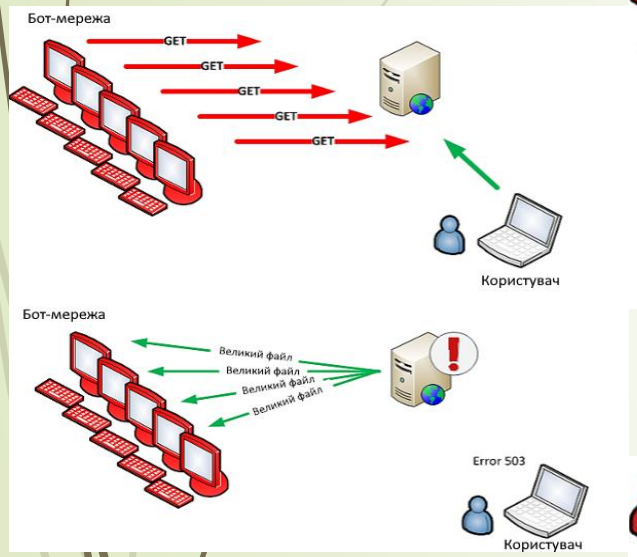
ICMP Flood



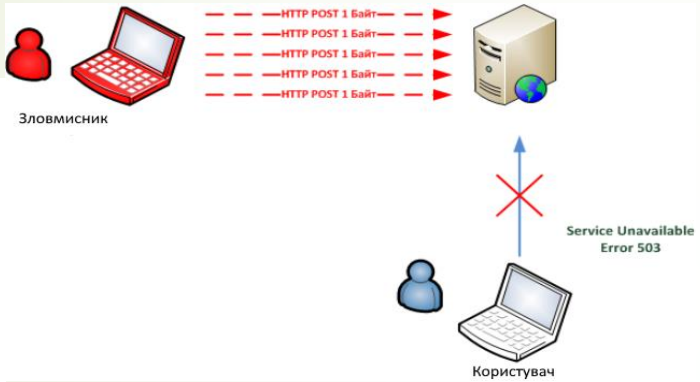
SYN Flood

- ICMP Flood — мережна (L3) атака, що полягає в масовій відправці ICMP Echo Request, що перевантажує CPU і мережеві буфери; Smurf-варіант посилює ефект через широкомовні відповіді зі спуфінгом джерела. UDP Flood надсилає багато UDP-пакетів на різні порти, змушуючи систему генерувати ICMP «порт недоступний» або навантажувати застосунки. Amplification-атаки використовують відкриті сервіси (DNS, NTP тощо) для багаторазового збільшення трафіку. SYN Flood експлуатує three-way handshake, переповнюючи таблиці з’єднань. Захист: моніторинг NetFlow/IPFIX, rate-limiting і ACL, anti-spoofing (RPF), блокування широкомовних запитів, FlowSpec/blackholing і cloud-scrubbing, а також SYN Cookies/SYN Proxy.

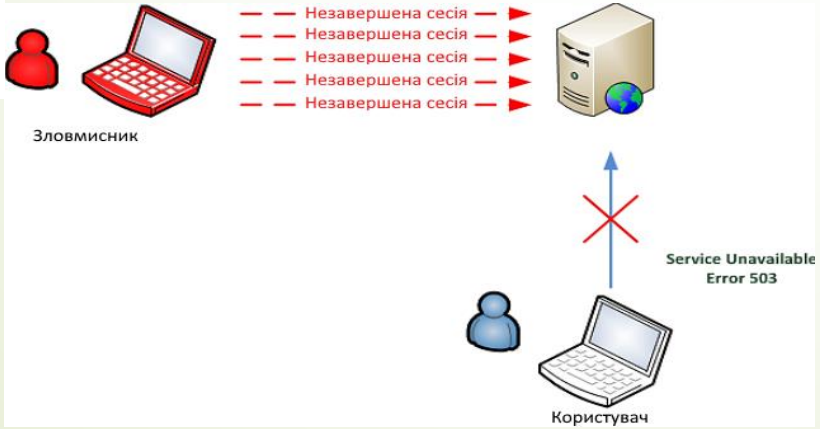
Допоміжні види атак



Атаки на веб-застосунки



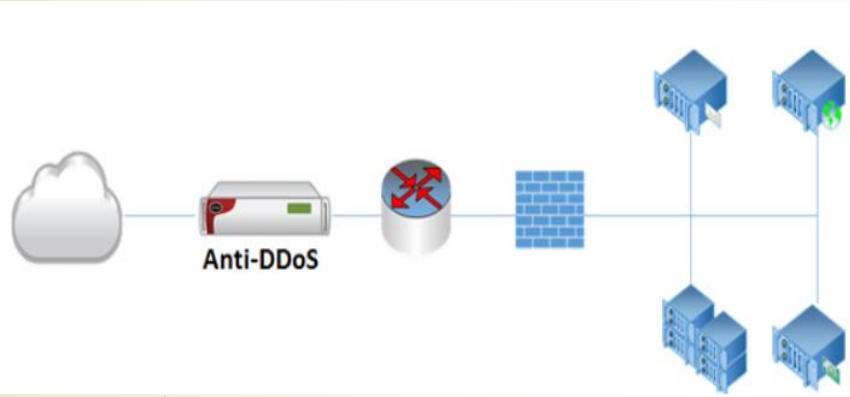
RUDY-атака



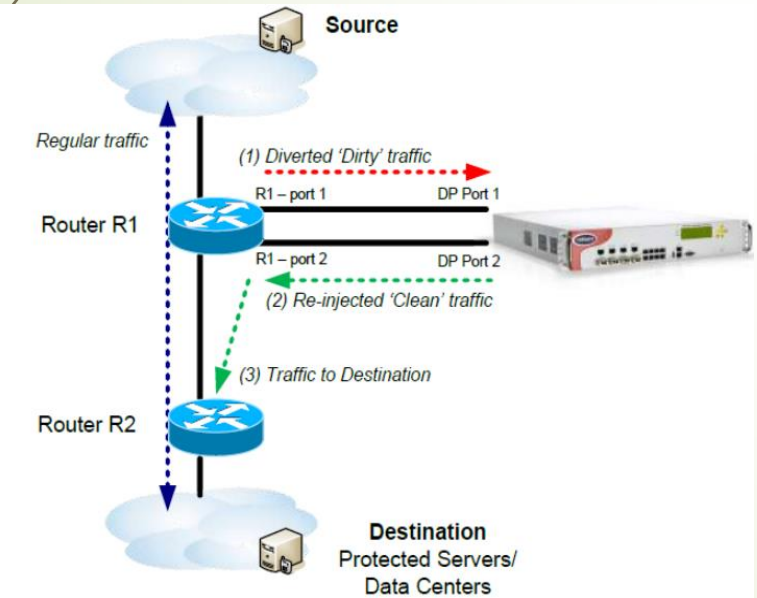
SlowLoris атака

- Атаки на веб-застосунки є одними з найнебезпечніших, адже вони спрямовані не лише на мережеві ресурси, а безпосередньо на логіку роботи веб-сервісів. Найпоширеніші серед них — **RUDY** та **SlowLoris**.
- **RUDY-атака (Are You Dead Yet?)** базується на повільному надсиланні даних через HTTP POST-запити. Сервер, очікуючи завершення передачі, довго утримує з’єднання відкритим, витрачаючи ресурси. Навіть невелика кількість таких запитів може паралізувати роботу сайту.
- **SlowLoris-атака** діє подібно, але використовує неповні HTTP-запити, які ніколи не завершуються. Це змушує сервер тримати тисячі відкритих сесій, швидко вичерпуючи пул ресурсів.
- Обидві атаки потребують мінімум трафіку, але ефективно блокують доступ користувачів. Для захисту необхідні системи поведінкового аналізу, обмеження часу з’єднань і спеціалізовані WAF-рішення.

Методи інтеграції компонентів захисту



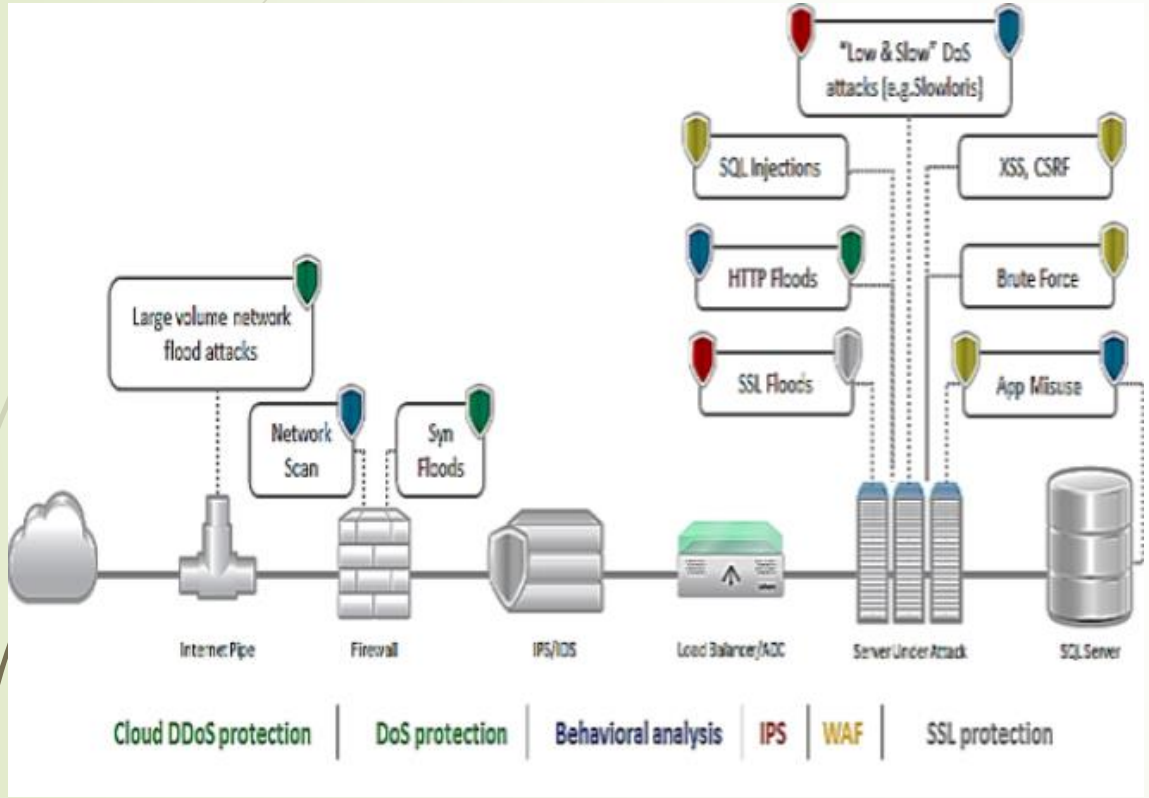
Метод установки «в розрив» (Inline)



Метод установки «Out-of-Path»

- Метод «в розрив» (Inline) розміщує фільтраційний модуль безпосередньо на шляху трафіку, забезпечуючи реальну інспекцію та швидке блокування аномалій; для надійності застосовують резервування і bypass. Система працює у фазах: навчання (побудова профілів нормального трафіку) та виявлення/реагування (динамічні фільтри). Альтернатива — «Out-of-Path»: трафік перенаправляють у центри очищення тільки при атаці. Цей підхід економніший, але має затримку виявлення, потребує перенаправлення трафіку та частіше залежить від ручного управління; також складно обробляти зашифровані SSL/TLS-атаки через відсутність дешифрування.

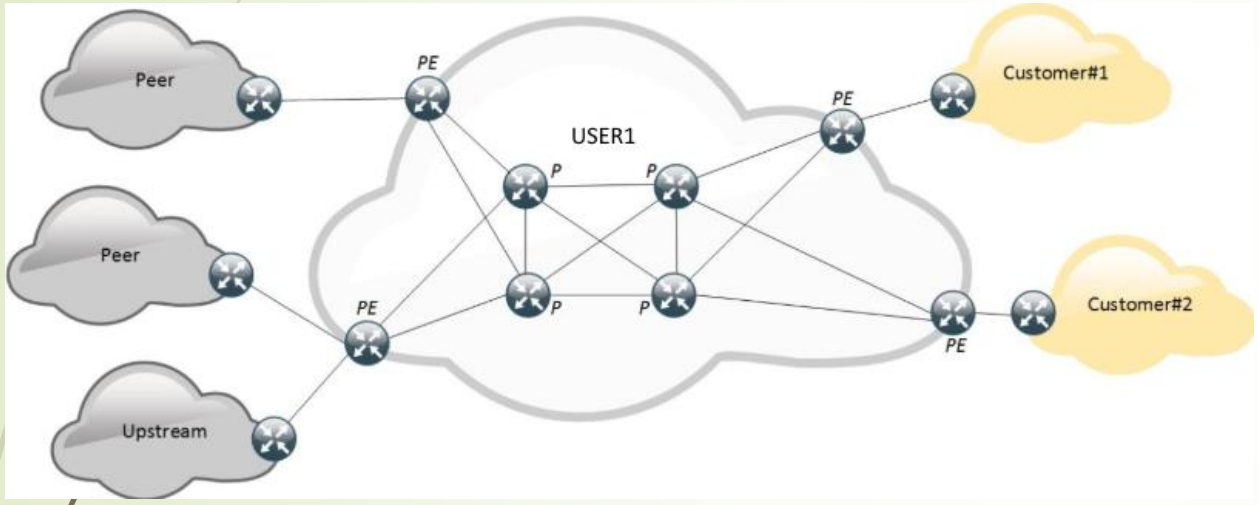
Підходи до архітектури системи



Орієнтовна архітектура рішення захисту від DDoS

- Ефективний захист від DDoS-атак потребує комплексного, багаторівневого підходу, що поєднує мережеві та прикладні методи безпеки. Система має включати модулі фільтрації трафіку, IPS для запобігання вторгненням, WAF для захисту веб-застосунків і хмарні рішення для нейтралізації масових атак. Ключовим є автоматичне реагування у реальному часі, мінімізація впливу на легітимних користувачів, а також централізований моніторинг і звітність. Такий підхід забезпечує стійкість корпоративних систем до складних кібератак, підтримуючи безперебійну роботу сервісів навіть під час активних загроз.

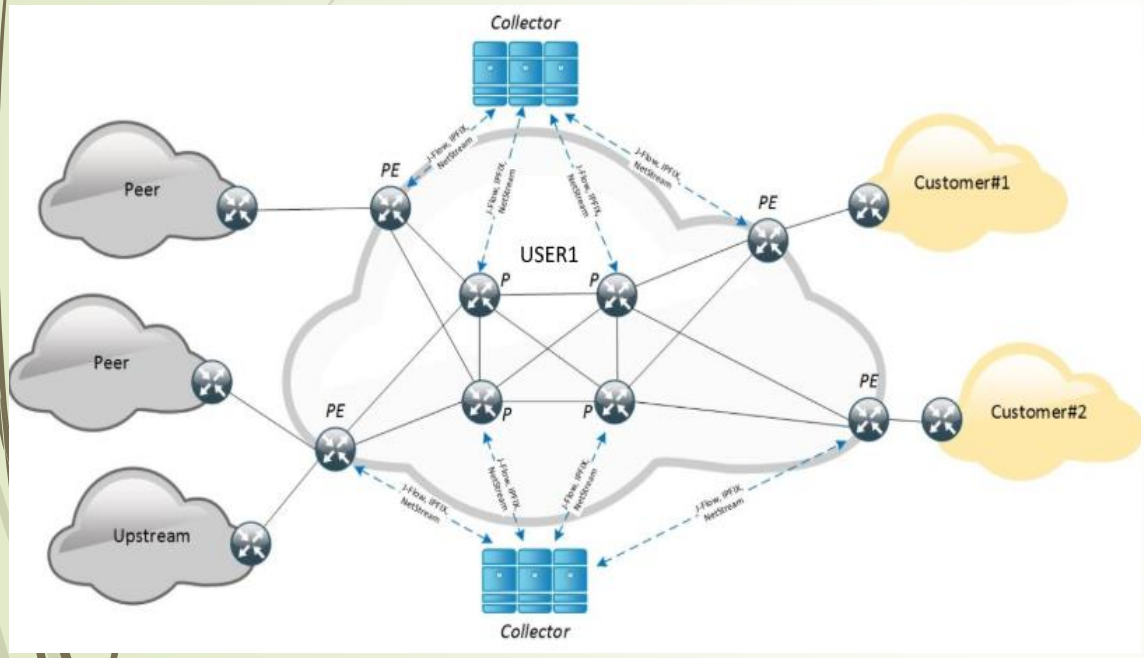
Рішення захисту для операторів



Спрощена архітектура магістральної IP/MPLS-мережі

- Розробка систем протидії DDoS-атакам для операторів зв'язку ускладнюється відсутністю чіткого периметра захисту та асиметрією маршрутизації трафіку в магістральних мережах. На прикладі IP/MPLS-архітектури оператора USER1 виділяють три типи з'єднань: upstream, peer і customer. Через розгалужену структуру мережі класичні методи фільтрації, як у корпоративних системах, неефективні. Тому операторські рішення базуються на двох підсистемах: виявлення аномалій — для аналізу трафіку в реальному часі, та фільтрації — для блокування паразитних потоків через ACL, чорні списки, rate-limiting і центри очищення трафіку.

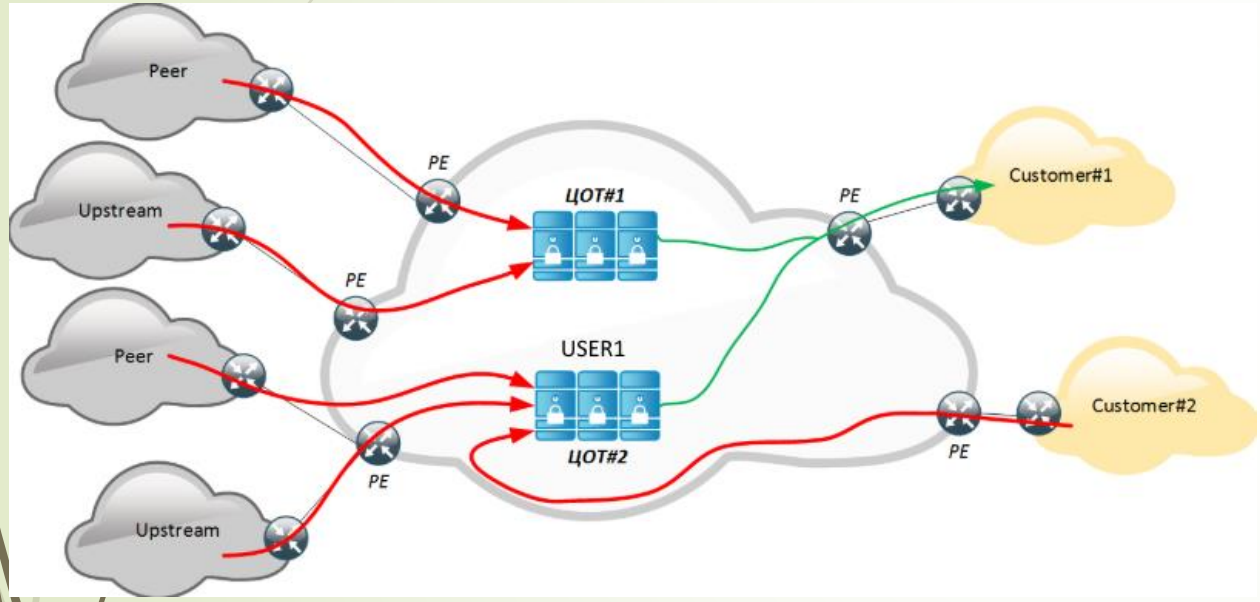
Детектування DDoS-атак



Інфраструктура колекторів

- Для виявлення аномалій у межах автономної системи використовується мережева телеметрія (NetFlow, J-Flow, IPFIX), що дозволяє аналізувати трафік до рівня L4 моделі OSI. Ці протоколи збирають ключові параметри — IP-адреси, порти, кількість пакетів, протоколи, часові мітки тощо, формуючи статистику потоків. На основі даних створюються профілі нормальної поведінки трафіку, які допомагають виявляти відхилення та потенційні DDoS-атаки. Аномалії класифікуються за рівнем критичності (low, medium, high), що дає змогу оперативно реагувати на загрози, мінімізувати хибні спрацювання та підвищити ефективність захисту мережевої інфраструктури.

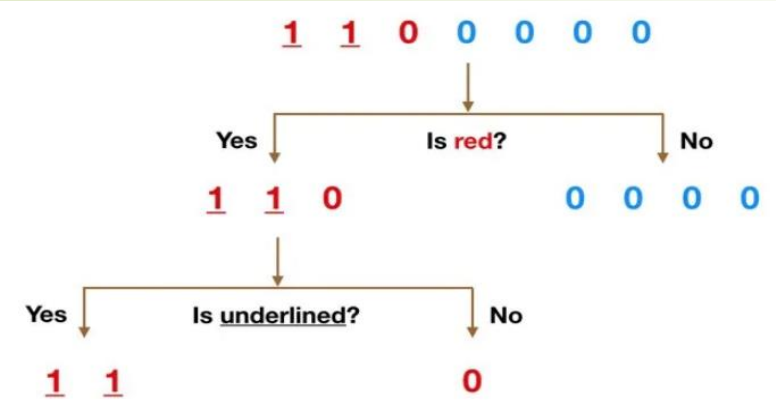
Фільтрація DDoS атак



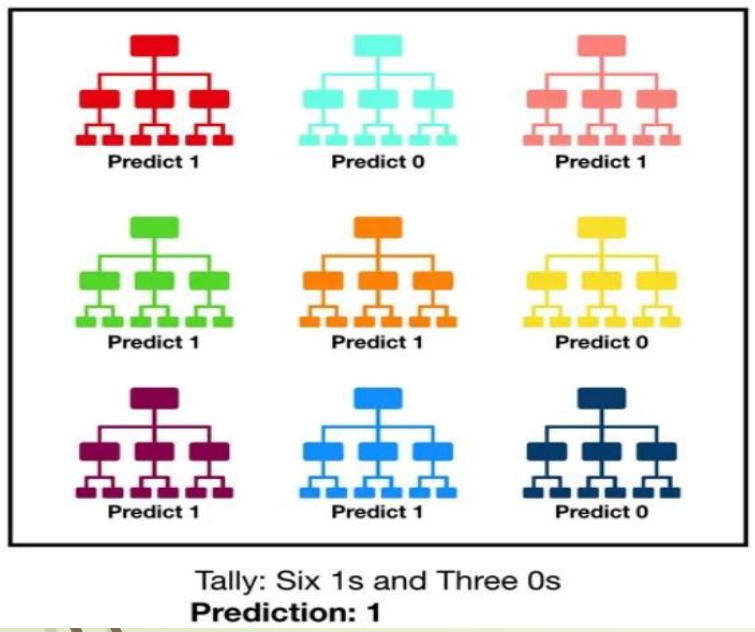
Центри очищення трафіку

- Після виявлення аномалій трафік перенаправляється на фільтрацію — вручну або автоматично. Використовуються три основні методи: Flow Specification, Blackhole Routing та очищення у Центрах очищення трафіку (ЦОТ). Найефективнішими є ЦОТ, які аналізують і фільтрують трафік у реальному часі, повертаючи лише «чисті» дані. Для масштабованості застосовується MPLS та BGP Labeled-Unicast. Ключову роль відіграє протокол BGP Flow Specification (RFC 5575), що забезпечує централізоване поширення правил фільтрації в межах автономної системи, дозволяючи швидко реагувати на атаки та ефективно блокувати шкідливий трафік ще на магістральному рівні.

Використання машинного навчання для протидії DDoS



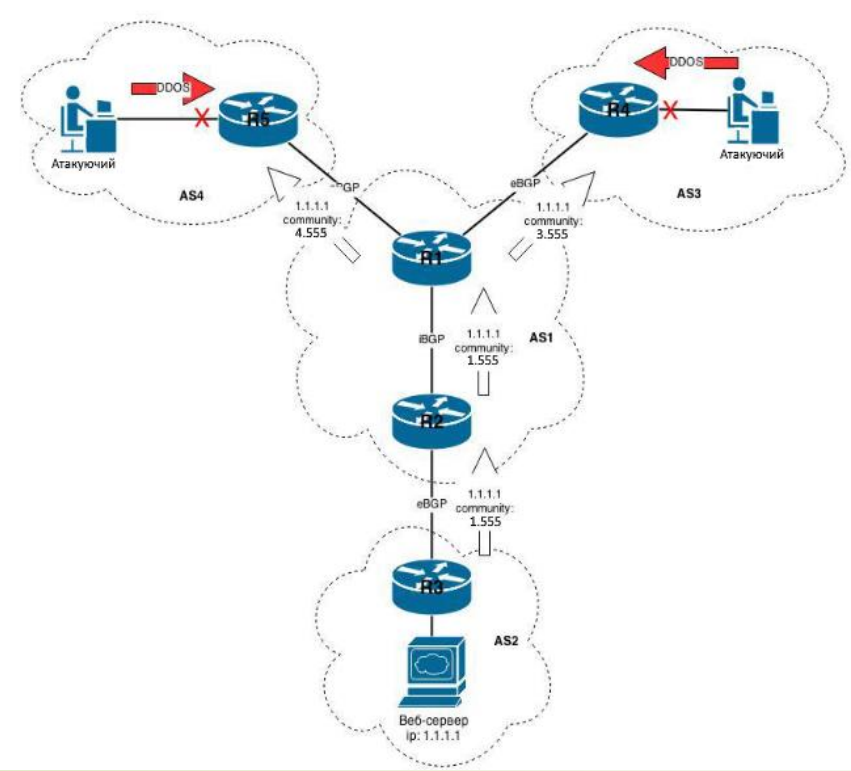
Алгоритм «Дерево рішень»



Алгоритм «Random Forest»

- Сьогодні DDoS-атаки досягають інтенсивності понад 1 Тбіт/с, а їхній вектор зміщується з мережевого на прикладний рівень, що загрожує безперервності роботи сервісів. Для підвищення ефективності захисту активно впроваджуються технології машинного навчання, здатні виявляти приховані закономірності в трафіку та відрізняти легітимних користувачів від ботів. Найбільш результативними виявилися алгоритми «Дерево рішень» і «Випадковий ліс», які забезпечують високу точність класифікації. Використання ML дозволяє адаптивно реагувати на нові вектори атак, зменшувати кількість хибних спрацювань і забезпечувати стабільну роботу інформаційних систем.

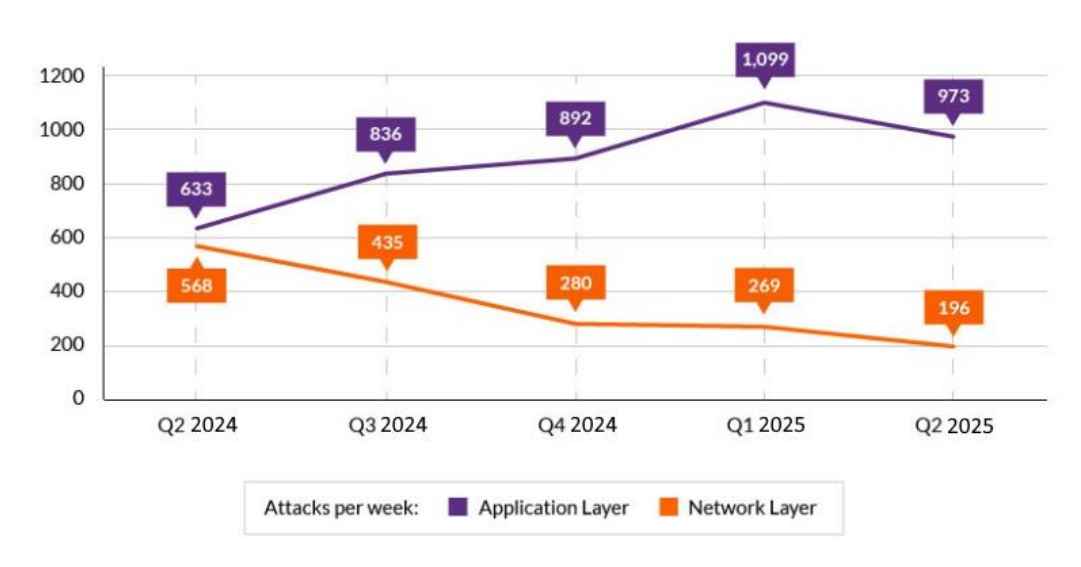
Захист від DDOS атак засобами BGP



Принцип роботи GP Blackhole

- Механізм GP Blackhole є одним із найефективніших засобів протидії масованим DDoS-атакам, спрямованим на перевантаження каналів зв’язку. Його суть полягає у відкиданні шкідливого трафіку ще на стороні провайдера, до потрапляння в мережу жертви. Реалізація здійснюється за допомогою протоколу BGP та спеціальних міток community, що сигналізують про блокування маршруту. Рекурсивний Blackhole дозволяє поширювати фільтрацію у напрямку джерел атаки, зменшуючи навантаження на магістральні вузли. Основні переваги — масштабованість, простота реалізації та ефективність у зниженні ризику перевантаження каналів зв’язку під час DDoS-атак.

Захист від DDoS на рівні веб-сервера



Статистика DDoS атак

- Статистика свідчить про зростання кількості DDoS-атак і зміщення їхнього вектора з мережевого на прикладний рівень, що робить вебдодатки головною ціллю зловмисників. Для базового захисту доцільно налаштувати брандмауер iptables і вебсервер nginx, обмеживши кількість підключень і запитів. Ефективним рішенням є використання OpenResty, який поєднує nginx і Lua для гнучкої обробки трафіку. Lua-скрипти дозволяють реалізувати “білі” та “чорні” списки, виявляти підозрілих ботів і блокувати їх у режимі реального часу, що підвищує стійкість вебресурсу до DDoS-атак.

ВИСНОВКИ

- У ході дослідження підтверджено, що DDoS-атаки є однією з найнебезпечніших загроз для телекомунікаційних мереж. Вони уражають як інфраструктурний (L3–L4), так і прикладний (L7) рівні моделі OSI, причому дедалі частіше спрямовані саме на вебдодатки, критичні для бізнесу й державних структур.
- Проаналізовано різновиди атак — від ICMP- та UDP-flood до складних методів на основі TCP, SSL/TLS і HTTP, включно з Low & Slow та Wordpress Pingback DDoS. Встановлено, що прикладні атаки небезпечні тим, що порушують логіку роботи сервісів і можуть спричиняти значні збитки.
- Розглянуто архітектурні рішення захисту: локальні, хмарні та гібридні моделі. Найефективнішою визначено багаторівневу систему, яка поєднує фільтрацію трафіку, Центри очищення, WAF, IPS/IDS і поведінковий аналіз.
- Перспективним напрямом є застосування машинного навчання (Decision Tree, Random Forest, нейронні мережі) для виявлення аномалій і підвищення швидкості реагування.
- Комплексний підхід, що об’єднує архітектурну стійкість, автоматизацію та інтелектуальний аналіз, є ключем до забезпечення безперервності роботи мереж і протидії атакам нового покоління.

Дякую за увагу!

Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Мирослава НІГРУЦА

Назва: ДОСЛІДЖЕННЯ ВПЛИВУ DDOS-АТАК НА ТЕЛЕКОМУНІКАЦІЙНІ МЕРЕЖІ ТА МЕТОДИ ЇХНЬОГО ВИЯВЛЕННЯ

Координатор: Вікторія КОРЕЦЬКА

Підрозділ: ННІТ

Коефіцієнт подібності 1:0.32

Коефіцієнт подібності 2:0

Тривога: 0

Після аналізу Звіту подібності констатую наступне:

☐ виявлені в роботі запозичення є сумлінними і не мають ознак плагіату. Тому робота визнається самостійною і допускається до захисту;

☐ виявлені в роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і самостійності її автора. Роботу направити на доопрацювання;

☐ виявлені в роботі запозичення є недобросовісними і мають ознаки плагіату або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень. У зв'язку з чим, робота не допускається до захисту.

…… ……

Дата *Підпис Наукового керівника*