Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“____” _____ 2025 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація системи протидії мережевим DDoS-атакам”

Виконав здобувач вищої освіти
ІІ курсу, групи КІ-24М
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Талмазан С.Д.
«____» _____ 2025 р.

Керівник проекту
доктор філософії (PhD)
_____ Усік П.С.
«____» _____ 2025 р.
Рецензент _____
_____

м. Кропивницький

АНОТАЦІЯ

Талмазан С.Д. Дослідження та програмна реалізація системи протидії мережевим DDoS-атакам. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2025.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи протидії мережевим DDoS-атакам.

Метою розробки є дослідження та програмна реалізація системи протидії мережевим DDoS-атакам.

Об'єктом дослідження є процес протидії мережевим DDoS-атакам.

Предметом дослідження є методи протидії мережевим DDoS-атакам.

Методи дослідження базуються на методах захисту інформації у комп'ютерних мережах, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи протидії мережевим DDoS-атакам.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерна інженерія, DDoS-атаки

ABSTRACT

Talmazan S.D. Research and software implementation of a system for countering network DDoS attacks. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2025.

In this final qualification work for the second (master's) level of higher education, software has been developed, which is intended for a system for countering network DDoS attacks.

The purpose of the development is the research and software implementation of a system for countering network DDoS attacks.

The object of the research is the process of countering network DDoS attacks.

The subject of the research is methods for countering network DDoS attacks.

The research methods are based on methods of information protection in computer networks, methods of mathematical statistics, and methods of software development.

The result of the work is a software implementation of a system for countering network DDoS attacks.

In the process of working on the software model, an analysis of existing hardware and software tools was performed. All components of the developed software are fully described.

A user-friendly user interface has been developed. Instructions for working with the software are provided.

The program can be used on a PC with Windows 10/11.

The program is developed in the Python environment.

Keywords: computer engineering, DDoS attacks

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ 3
ВСТУП..... 4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ 6
1.1 Призначення системи..... 6
1.2 Область застосування..... 7
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ 9
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти..... 9
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування..... 22
2.3 Розгорнута постановка завдання 24
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ 26
3.1 Опис функціонування системи 26
3.2 Розробка структурної схеми..... 30
3.3 Розробка функціональної схеми 42
3.4 Розробка діаграми процесів..... 55
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ..... 57
4.1 Розробка блок-схем та опис алгоритмів функціонування системи..... 57
4.2 Захист розробленого програмного забезпечення..... 73
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ..... 75
6 НАУКОВА НОВИЗНА 79

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ ІТ-ПРОЄКТУ 80
7.1 Визначення цільової аудиторії кінцевого готового продукту 80
7.2 Оцінка привабливості шляхом застосування методів експертних оцінок ... 81
7.3 Вибір методу оцінки вартості ПЗ 81
7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості 82
7.5 Пропозиція алгоритму просування проєкту розробки ПЗ 84
7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ 85
7.7 Визначення ключових факторів успіху конкретного проєкту 85
8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ 87
8.1 Вступ 87
8.2 Пожежна безпека 89
8.3 Пропозиції щодо підвищення працездатності ІТ-фахівців 90
8.4 Розробка заходів з умов поліпшення охорони праці 92
8.5 Розрахункова частина 93
9 ОСНОВНІ ВИСНОВКИ 96
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 98

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ЕОМ	–	електрона обчислювальна машина
КВ	–	коефіцієнт варіації
КЗ	–	канал зв'язку
НСД	–	несанкціонований доступ
ПС	–	програмна середа
СВВ	–	система виявлення вторгнень
СеМО	–	експонентна мережа масового обслуговування
СМО	–	система масового обслуговування
СПД	–	система передачі даних

ВСТУП

Актуальність теми. Забезпечення захищеності мережі від DDoS-атак вимагає добре структурованої захисної стратегії. Мережа часто містить численні ресурси – веб-сайти, програми та сервіси – не лише для своїх власників, але й для їхніх клієнтів.

Масштабна DDoS-атака, спрямована лише на один із цих ресурсів, може створити величезне навантаження на мережеву інфраструктуру. Різке збільшення незаконного трафіку може перевантажити навіть найпотужніші маршрутизатори, що призведе до перебоїв у роботі та потенційних збоїв.

І це не просто теорія – DDoS-атаки, що досягають сотень гігабіт на секунду, зараз є поширеним явищем. Нещодавно ми навіть зафіксували атаку потужністю 1,5 Тбіт/с.

Зрозуміло, що само по собі обладнання та програмне забезпечення на периферії ледве справляються з такими масованими атаками, що робить хмарні рішення для боротьби з DDoS-атаками необхідністю.

Проблему посилює те, що мережеві оператори зазвичай керують великими пулами IP-адрес, які зловмисники використовують, одночасно запускаючи численні менші DDoS-атаки. Ці атаки низької інтенсивності можуть залишитися непоміченими традиційними засобами захисту, але їхній сукупний вплив на периферійні пристрої може бути серйозним, що призводить до зниження продуктивності, операційної нестабільності або навіть повного виходу з ладу вузла.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи протидії мережевим DDoS-атакам.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

– Огляд існуючих систем протидії мережевим DDoS-атакам.

– Дослідження системи протидії мережевим DDoS-атакам.

– Програмна реалізація системи протидії мережевим DDoS-атакам.

Об’єктом дослідження є процес протидії мережевим DDoS-атакам.

Предметом дослідження є методи протидії мережевим DDoS-атакам.

Методи дослідження базуються на методах захисту інформації у комп’ютерних мережах, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод протидії мережевим DDoS-атакам.

– Розроблено вітчизняний продукт протидії мережевим DDoS-атакам, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі протидії мережевим DDoS-атакам.

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти LV науково-технічної конференції «Наука в ЦНТУ: основні досягнення та перспективи розвитку» (2025 р.), основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №15.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи протидії мережевим DDoS-атакам, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Аудит вашої мережі та розробка стратегії захисту від DDoS-атак

Перш ніж впроваджувати захист мережі, важливо провести ретельний аудит мережевої та інформаційної безпеки (ІБ). Це дасть вам чітке уявлення про вашу інфраструктуру та допоможе вам побудувати ефективну стратегію захисту від DDoS-атак.

Ось на що слід звернути увагу під час аудиту:

1. Зрозумійте архітектуру вашої мережі:

– Визначте, як структурована та налаштована ваша мережа.

– Аналізуйте розподіл трафіку, потужність сервера та продуктивність периферійних пристроїв.

– Визначте, як периферійні пристрої впливають на решту вашої інфраструктури.

– Використовуйте бази даних керування конфігурацією (CMDB), якщо вони доступні – вони надають цінну інформацію та зазвичай підтримуються операторами центрів обробки даних та мереж.

2. Перевірте відкриті служби та порти:

– Визначте, які мережеві служби та порти активні на периферійних пристроях.

– Вимкніть ті, що не потрібні, щоб зменшити потенційні поверхні для атак.

3. Оцініть існуючі заходи безпеки:

– Перегляньте наявні інструменти безпеки, як вони керуються та як вони взаємодіють.

– Виявити прогалини або неефективність у координації захисту від DDoS-

атак.

4. Визначте критичні активи, які потребують захисту:

– Визначте ключові пристрої, служби, порти та IP-адреси, які потребують захисту від DDoS-атак.

– Майте на увазі: часткового захисту недостатньо. Дійсно ефективна стратегія повинна забезпечувати комплексне покриття на всіх рівнях:

- Мережевий та транспортний рівні (L3/L4) – захист від волюметричних та протокольних атак.
- Рівень застосунків (L7) – захист служб DNS, HTTP та HTTPS від складніших DDoS-загроз.

Щойно ви повністю зрозумієте, як захистити свою мережу від DDoS-атак, ви зможете розробити структуровану стратегію захисту від DDoS-атак і відповідно спланувати наступні кроки.

1.2 Область застосування

Зберіть мережеву інформацію для вашого постачальника послуг захисту від DDoS-атак

Чим більше інформації ви надасте своєму постачальнику послуг захисту від DDoS-атак, тим швидше та ефективніше він зможе допомогти захистити вашу мережу. Гарний постачальник активно запитуватиме детальні дані про мережу – це ознака професійного підходу та добре продуманої стратегії захисту.

Використовуючи дані аудитів вашої мережі та безпеки, підготуйте комплексний огляд мережі. Це допоможе вам і вашому провайдеру вибрати найкращий варіант захисту мережі від DDoS-атак і розробити індивідуальну стратегію безпеки на основі вашої інфраструктури, моделей трафіку та вразливостей.

Вашому провайдеру знадобляться конкретні відомості про вашу мережу, такі як IP-адреси ваших DNS-серверів, де розташовані ваші VPN-шлюзи та які IP-

адреси обробляють великі обсяги трафіку (наприклад, пули NAT). Якщо ви використовуєте кешувальні проксі-сервери, такі як Squid або BlueCoat, вам також слід надати їхні IP-адреси.

Деякі служби кешування, такі як Google Global Cache (GGC), Facebook Network Appliance (FNA), Netflix caching та Akamai, час від часу генерують великі сплески трафіку. Якщо ваш провайдер не знає про це, він може помилково заблокувати їх як DDoS-атаки. Надання цієї інформації забезпечує правильне налаштування захисту від DDoS.

Також важливо уточнити, які IP-адреси належать повноцінним інтернет-клієнтам, яким потрібен як вхідний, так і вихідний трафік (наприклад, VDS/VPS, кінцеві користувачі), а які належать веб-серверам, спільному хостингу або іншим сервісам, яким не потрібен повний доступ до протоколу. Це допомагає створювати точні профілі захисту та дозволяє вашому провайдеру впроваджувати кращі заходи проти спроб розвідки мережі.

Інформування вашого постачальника про ваші маршрутизатори допомагає йому оцінити продуктивність і за потреби рекомендувати оновлення. Наприклад, низькоякісні маршрутизатори MikroTik справляються навіть зі слабкими DDoS-атаками (100-200 тис. пакетів/сек), тому можуть знадобитися суворіші правила фільтрації. Маршрутизатори серії Cisco ASR повинні обробляти 5-6 мільйонів пакетів/сек, але якщо вони стикаються з проблемами при нормальному навантаженні, можуть знадобитися коригування для покращення продуктивності та стійкості.

Ділячись детальною аналітикою мережі, ви дозволяєте своєму постачальнику анти-DDoS-захисту створити надійну, ефективну та індивідуальну стратегію захисту, адаптовану до ваших потреб.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи протидії мережевим DDoS-атакам, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Корпоративні брандмауери повинні обробляти високі навантаження трафіку, забезпечуючи безпеку мереж. Нижче наведено найкращі варіанти на 2025 рік, кожен з яких підходить для різних потреб.

Пристрій безпеки Meter's – найкращий засіб для повністю керованої та безпроблемної безпеки

Брандмауер безпеки Meter створений для компаній, яким потрібен надійний захист, але які не хочуть витрачати години на його керування.

На відміну від традиційних брандмауерів, які потребують ручного оновлення та постійної уваги, Meter керує цим за вас. Ніякого налаштування, жодних проблем із прошивкою та жодної поспіху зі встановленням патчів безпеки. Це частина мережі Meter, яка включає захист брандмауера, Wi-Fi та комутацію в одній простій системі.

Багато корпоративних брандмауерів потребують регулярного налаштування, щоб залишатися ефективними. Meter знімає це з IT-відділу, займаючись налаштуванням, моніторингом та оновленнями, тож компанії отримують надійний захист, не додаючи додаткової роботи своїм командам.

Особливості:

– Запобігання вторгненням для виявлення та зупинки загроз у режимі реального часу.

– Вбудований SD-WAN для безпечного та стабільного підключення до кількох точок розташування.

– Безпека Zero Trust з детальним контролем доступу.

– Аналітика на базі штучного інтелекту для глибшого розуміння мережі.

– Автоматичні оновлення безпеки без простоїв.

Найкраще підходить для компаній, яким потрібен корпоративний рівень безпеки без зайвих клопотів. IT-команди, які мають обмежений ресурс або не хочуть керувати брандмауерами вручну, оцінять підхід без участі сторонніх осіб.

Переваги:

– Зменшує ризики безпеки, автоматизуючи оновлення та запобігаючи загрозам.

– Звільняє IT-команди, займаючись керуванням та моніторингом брандмауера.

– Знижує витрати, усуваючи потребу в додатковому апаратному чи програмному забезпеченні безпеки.

– Працює як частина повністю керованої мережі, що зменшує складність роботи з постачальниками.

Недоліки:

– Він розроблений для найкращої роботи з мережею Meter, тому поєднання з іншими постачальниками може додати складності.

– Досвідчені користувачі, які шукають глибокі ручні налаштування, можуть віддати перевагу більш настроюваним опціям.

Підсумок

Пристрій безпеки Meter забезпечує підприємствам високий рівень безпеки без навантаження на IT-інфраструктуру. Він повністю керований, масштабується за потреби та безперебійно працює з мережевою інфраструктурою Meter, що робить його чудовим вибором для компаній, що розвиваються.

Fortinet FortiGate – найкращий варіант для високопродуктивної безпеки

Fortinet FortiGate створений для швидкості. Деякі брандмауери сповільнюють роботу зі збільшенням трафіку, але цей використовує

спеціалізовані процесори безпеки для забезпечення швидкого передавання даних та блокування загроз.

Це один із найкращих варіантів для компаній, які обробляють великі обсяги трафіку. Завдяки безпеці на базі штучного інтелекту та глибокій перевірці пакетів він зупиняє атаки, не уповільнюючи мережу.

Особливості:

– Спеціальні процесори безпеки для швидкого виявлення загроз.

– Захист від шкідливих програм на базі штучного інтелекту.

– Вбудований SD-WAN для безпечної мережі з кількох локацій.

– Інтегрований контроль доступу VPN та Zero Trust.

Найкраще підходить для компаній з високими потребами в трафіку, включаючи фінанси, охорону здоров'я та хмарні бізнеси. Це надійний варіант для тих, кому потрібна безпека без затримок.

Переваги:

– Використовує спеціалізовані процесори безпеки для обробки високого трафіку без затримок.

– Захист від шкідливих програм на базі штучного інтелекту допомагає швидко виявляти та зупиняти загрози.

– Глибока перевірка пакетів забезпечує детальний аналіз безпеки без уповільнення продуктивності.

– Вбудована SD-WAN покращує безпечне підключення між кількома місцями розташування.

Недоліки:

– Для налаштування розширених параметрів безпеки можуть знадобитися додаткові навички.

– Деякі функції вимагають окремих ліцензій, що може збільшити витрати.

Підсумок

Fortinet FortiGate – один із найшвидших брандмауерів на ринку. Це чудовий вибір для компаній, яким потрібен високий рівень безпеки та висока швидкість одночасно.

Брандмауери нового покоління від Palo Alto Networks – найкращі для виявлення загроз на основі штучного інтелекту

Брандмауери нового покоління від Palo Alto Networks використовують штучний інтелект та машинне навчання для зупинки кіберзагроз, перш ніж вони завдадуть шкоди. Замість того, щоб дотримуватися попередньо встановлених правил, ці брандмауери аналізують трафік у режимі реального часу та коригуються в міру розвитку загроз.

Кібератаки швидко змінюються. Брандмауер, який блокує лише відомі загрози, не може за ними встигати. Система Palo Alto на базі штучного інтелекту виявляє нові ризики, навчається на них і зупиняє атаки до їх поширення. Це робить її чудовим вибором для компаній, яким потрібна проактивна безпека, а не реактивні рішення.

Особливості:

– Безпека на базі штучного інтелекту, яка виявляє та блокує загрози в режимі реального часу.

– Глибока перевірка пакетів для блокування шкідливого програмного забезпечення, фішингу та підозрілого трафіку.

– Автоматизовані політики безпеки, що зменшують ручну роботу IT-команд.

– Хмарні оновлення, які підтримують захист у актуальному стані

Найкраще підходить для компаній, яким потрібен надійний та постійний захист, наприклад, у сфері фінансів, охорони здоров'я чи державного управління. Це особливо корисно для компаній, які обробляють конфіденційні дані або стикаються з частими кіберзагрозами.

Переваги:

– Адаптується до нових загроз у режимі реального часу, не чекаючи ручних оновлень.

– Автоматизовані політики безпеки зменшують потребу в ручному управлінні.

– Хмарні оновлення підтримують актуальність захисту від загроз без додаткових зусиль з боку IT-відділу.

– Інтегрується з іншими інструментами безпеки Пало-Альто для створення єдиної системи захисту.

Недоліки:

– Розширені функції на основі штучного інтелекту можуть вимагати від IT-команд адаптації до нового робочого процесу.

– Деякі функції безпеки залежать від хмарного підключення, яке може не підходити для всіх середовищ.

Підсумок

Брандмауери Palo Alto забезпечують найвищий рівень безпеки на основі штучного інтелекту. Вони створені для компаній, яким потрібен захист у режимі реального часу та автоматичне виявлення загроз без постійних ручних оновлень. Якщо безпека не може дозволити собі відставати, цей брандмауер – надійний варіант.

Check Point Quantum – найкращий варіант для багаторівневої, багатохмарної безпеки

Check Point Quantum розроблено для компаній, які використовують як хмарні, так і локальні мережі. Багато брандмауерів мають проблеми з гібридними налаштуваннями, але Quantum захищає все в рамках однієї системи. Він пропонує багаторівневий захист, виявлення загроз і просте керування для хмарних мереж.

Керування безпекою на різних платформах може бути складним завданням. Check Point Quantum вирішує цю проблему, створюючи єдину політику безпеки як для хмарних, так і для локальних систем. Він також

використовує виявлення загроз на основі штучного інтелекту та засоби контролю Zero Trust для ранньої зупинки атак.

Особливості:

– Хмарна безпека із запобіганням загрозам у режимі реального часу.

– Сегментація Zero Trust для обмеження несанкціонованого доступу.

– Безпека на основі штучного інтелекту, яка адаптується до нових кіберзагроз.

– Централізоване управління гібридними та мультихмарними мережами.

Найкраще підходить для компаній із гібридними хмарними налаштуваннями, яким потрібні узгоджені політики безпеки в різних місцях та у різних постачальників хмарних послуг. Це чудовий варіант для великих підприємств, фінансових установ та компаній, що мають справу з суворими правилами безпеки.

Переваги:

– Забезпечує стабільну безпеку в локальних та хмарних середовищах в рамках однієї системи.

– Виявлення загроз на основі штучного інтелекту допомагає зупинити кібератаки до їх поширення.

– Централізоване управління спрощує нагляд за безпекою складних мереж.

– Сегментація Zero Trust додає додатковий рівень захисту від несанкціонованого доступу.

Недоліки:

– Розширені функції можуть вимагати додаткового налаштування для підприємств з унікальними потребами безпеки.

– Деякі можливості залежать від підключення до хмари, яке може не працювати для повністю офлайн-систем.

– Вартість ліцензування може зрости для компаній, яким потрібен повнофункціональний багатохмарний захист.

– Керування гібридною політикою безпеки може вимагати від IT-команд коригування існуючих робочих процесів.

Підсумок

Check Point Quantum пропонує надійний захист як у хмарних, так і в локальних середовищах. Він створений для компаній, яким потрібно керувати безпекою на кількох платформах без додаткової складності. Якщо гібридний хмарний захист є пріоритетом, цей брандмауер – чудовий вибір.

Cisco Secure Firewall – найкращий варіант для компаній, які вже використовують інфраструктуру Cisco

Cisco Secure Firewall розроблено для підприємств, які покладаються на мережеве обладнання Cisco. Він працює з комутаторами, маршрутизаторами та хмарними сервісами Cisco, що робить його логічним вибором для підприємств, які вже інвестували в екосистему Cisco. Замість того, щоб жонглювати кількома постачальниками, IT-команди можуть керувати безпекою та мережею в рамках однієї системи.

Cisco Secure Firewall підтримується Cisco Talos, однією з найбільших команд розвідки загроз у світі. Це означає захист у режимі реального часу від нових і виникаючих кіберзагроз. Він також підтримує безпеку Zero Trust, VPN-доступ і автоматизоване застосування політик, що робить його сильним вибором для великих мереж, яким потрібен централізований контроль.

Особливості:

– Аналітика загроз у режимі реального часу на базі Cisco Talos

– Інтегрований VPN та безпека Zero Trust для безпечного віддаленого доступу

– Масштабований для підприємств зі складними мережами та кількома локаціями

– Автоматизовані політики безпеки для зменшення ручного налаштування

Найкраще підходить для компаній, які вже використовують маршрутизатори, комутатори або хмарні сервіси Cisco. Це особливо корисно для великих підприємств та IT-команд, які віддають перевагу глибокій інтеграції між безпекою та мережею.

Переваги:

– Легко інтегрується з мережевим обладнанням та програмним забезпеченням Cisco для створення єдиної системи.

– Аналітика загроз Cisco Talos забезпечує безперервний захист від нових кіберзагроз.

– Легко масштабується для підтримки великих та складних корпоративних мереж.

Недоліки:

– Деякі розширені функції безпеки вимагають додаткових витрат на ліцензування.

– Для управління системою може знадобитися IT-персонал, знайомий з екосистемою Cisco.

– Параметри налаштування можуть бути складнішими порівняно з іншими рішеннями брандмауера.

Підсумок

Cisco Secure Firewall – це очевидний вибір для компаній, які вже використовують інфраструктуру Cisco. Він спрощує керування безпекою та забезпечує надійний захист без зайвої складності. Якщо ваша мережа побудована на Cisco, цей брандмауер – найпростіший спосіб додати безпеку корпоративного рівня.

Брандмауер Barracuda CloudGen – найкращий варіант для гібридних хмарних середовищ

Брандмауер Barracuda CloudGen створений для компаній, які використовують як хмарні, так і локальні мережі. Багато брандмауерів захищають одне або інше, але Barracuda забезпечує безпеку обох в одній системі. Він

пропонує SD-WAN, хмарну безпеку та розширене виявлення загроз, що робить його чудовим варіантом для компаній з віддаленими командами або кількома локаціями.

Деякі брандмауери уповільнюють роботу хмарних програм або ускладнюють віддалений доступ. Barracuda забезпечує безперебійну передачу трафіку, швидке з'єднання та надійний захист. Він також захищає від кіберзагроз за допомогою запобігання вторгненням та сканування на шкідливе програмне забезпечення.

Особливості:

– Повна підтримка SD-WAN для безпечної та високопродуктивної хмарної мережі.

– Запобігання вторгненням та сканування на шкідливе програмне забезпечення для зупинки кіберзагроз.

– Розширена оптимізація трафіку для запобігання уповільненням та покращення продуктивності програм.

– Уніфіковані політики безпеки як для хмарних, так і для локальних середовищ.

Найкраще підходить для компаній, які використовують поєднання хмарної та фізичної інфраструктури. Це чудовий варіант для компаній з віддаленими офісами, гібридними хмарними налаштуваннями або команд, які залежать від хмарних додатків.

Переваги:

– Забезпечує повну підтримку SD-WAN для безпечного та оптимізованого хмарного підключення.

– Централізоване управління спрощує контроль безпеки в кількох місцях.

– Включає вбудоване запобігання вторгненням та сканування на шкідливе програмне забезпечення для посилення захисту.

Недоліки:

– Деякі розширені функції безпеки можуть вимагати додаткового налаштування.

– Бізнеси без гібридної хмарної конфігурації можуть не повною мірою скористатися її функціями.

– Початкове налаштування може зайняти більше часу для організацій, які не знайомі з екосистемою Barracuda.

Підсумок

Брандмауер Barracuda CloudGen захищає гібридні мережі без додавання складності. Він захищає як хмарні, так і локальні середовища, забезпечує безперебійний потік трафіку та допомагає компаніям керувати безпекою в кількох місцях. Якщо вам потрібен брандмауер, який працює як для хмарних, так і для фізичних мереж, це найкращий вибір.

Інші корпоративні брандмауери, які варті згадки

Деякі брандмауери пропонують надійний захист, але не потрапили до нашого списку найкращих через певні обмеження або нішеві випадки використання. Ці брандмауери все ще забезпечують надійний захист і можуть добре підійти для певних видів бізнесу. Ось чому їх не включили до нашого списку найкращих і кому вони все ще можуть бути корисними.

Брандмауер Sophos XG – простий захист для малого та середнього бізнесу

Брандмауер Sophos XG розроблений для простоти використання, що робить його популярним серед малого бізнесу, якому потрібна надійна безпека без зайвих складнощів. Він забезпечує глибоку перевірку пакетів, хмарне управління та контроль Zero Trust.

Хоча він добре працює для невеликих мереж, він не масштабується так ефективно, як інші корпоративні брандмауери. Він підходить для малого та середнього бізнесу, який шукає доступний та простий в управлінні захист. Функції безпеки надійні, але не такі просунуті, як у Palo Alto чи Fortinet. Компанії

з обмеженим IT-персоналом оцінять його просте налаштування та хмарні засоби керування.

Серія Juniper Networks SRX – надійний варіант для мереж на базі Juniper

Серія SRX від Juniper – це швидкий та надійний брандмауер, який працює з мережевим обладнанням Juniper. Він пропонує глибоку перевірку пакетів, безпеку Zero Trust та підтримку SD-WAN, що робить його сильним вибором для компаній, які вже використовують продукти Juniper.

Хоча він і потужний, він вимагає більше ручного налаштування, ніж конкуренти. Інтерфейс керування менш інтуїтивно зрозумілий, ніж у Fortinet або Palo Alto, що ускладнює його використання командами, незнайомими з екосистемою Juniper.

Чудово підходить для підприємств, які вже використовують мережеві продукти Juniper і бажають легко інтегрованої безпеки. Найбільшу користь отримають IT-команди з досвідом управління обладнанням Juniper.

Брандмауер Forcepoint наступного покоління – надійний захист даних, але не повноцінне рішення для безпеки мережі

Брандмауер Forcepoint створено з урахуванням безпеки даних. Він зосереджений на запобіганні витокам даних, внутрішнім загрозам та порушенням нормативних вимог, що робить його надійним вибором для компаній, які працюють з конфіденційною інформацією.

Його сильна сторона полягає в захисті даних, а не в повній безпеці мережі. Хоча він запобігає витоку даних, він не зрівняється з виявленням загроз у режимі реального часу та оптимізацією продуктивності Fortinet чи Palo Alto.

Ідеально підходить для компаній, які обробляють висококонфіденційні дані, наприклад, у сфері фінансів, охорони здоров'я чи уряду. Компаніям, яким потрібні потужні інструменти запобігання втраті даних, він буде корисним.

Розподілений брандмауер VMware NSX – чудово підходить для віртуалізації, але не є традиційним брандмауером

VMware NSX – це програмний брандмауер, розроблений для віртуалізованих та хмарних середовищ. Замість захисту трафіку на межі мережі він захищає внутрішній трафік усередині віртуальних машин.

Він корисний лише для компаній, що використовують платформу VMware. Він не захищає фізичну інфраструктуру, а це означає, що не може замінити стандартний брандмауер для більшості підприємств. Добре працює для центрів обробки даних та хмарних компаній, але не підходить для традиційних потреб мережевої безпеки.

WatchGuard Firebox – бюджетний вибір для малого бізнесу

WatchGuard Firebox – це економічно ефективний брандмауер, який включає базовий захист від загроз, підтримку VPN та можливості Zero Trust. Він простий у використанні та добре підходить для невеликих компаній, яким потрібна надійна безпека без зайвих складнощів.

Він не призначений для великих підприємств. Йому бракує розширеного виявлення загроз, високошвидкісної продуктивності та масштабованості, властивих потужнішим брандмауерам. Менші підприємства, які шукають доступний та простий захист, можуть отримати від нього користь. Організації без спеціалізованих IT-команд оцінять його прості інструменти управління.

Серія SonicWall NSa – хороший брандмауер середнього класу, але не корпоративного рівня.

Брандмауери SonicWall NSa забезпечують надійний захист за нижчою ціною, ніж у багатьох конкурентів. Вони включають запобігання вторгненням, підтримку VPN та хмарне управління, що робить їх розумним вибором для середнього бізнесу.

Хоча він пропонує надійний захист, йому бракує безпеки на основі штучного інтелекту, розширеної автоматизації та високошвидкісної обробки, як у провідних корпоративних брандмауерах. Це гарне рішення середнього рівня, але

не ідеальне для складних мереж. Здебільшого використовується середніми підприємствами, яким потрібен доступний захист без шкоди для багатьох функцій. Компанії з помірними потребами в безпеці знайдуть у ньому надійний баланс між вартістю та захистом.

Як брандмауери вписуються в повну стратегію кібербезпеки

Брандмауер – це гарний початок, але він не повинен бути єдиним інструментом безпеки, на який покладається бізнес. Він найкраще працює як частина багаторівневого захисту, блокуючи загрози, тоді як інші інструменти безпеки виявляють, розслідують та зупиняють атаки, що проходять крізь них. Ось як брандмауери вписуються в сильніший план кібербезпеки.

Зупинити погрози біля вхідних дверей недостатньо

Брандмауери блокують небажаний трафік, перш ніж він досягне мережі, але це лише перший крок. Кіберзагрози все ще можуть прослизнути. Щоб залишатися захищеними, підприємствам також потрібні системи виявлення вторгнень (IDS), засоби захисту кінцевих точок та інструменти SIEM для виявлення атак, які проходять повз брандмауер.

Нульова довіра не пускає недобросовісних людей

Брандмауер допомагає захистити мережу, але довіряти всьому, що знаходиться всередині неї, ризиковано. Безпека Zero Trust додає ще один рівень, дозволяючи доступ лише схваленим користувачам і пристроям. Брандмауери підтримують це, забезпечуючи суворий контроль доступу, зменшуючи ризик як зовнішніх атак, так і внутрішніх загроз.

Віддалена робота робить брандмауери ще важливішими

Оскільки все більше співробітників працюють віддалено та використовують хмарні додатки, брандмауери повинні захищати не лише офісні мережі. Брандмауери з вбудованими VPN та інструментами хмарної безпеки допомагають захистити віддалених працівників, філії та хмарні сервіси, забезпечуючи безпеку даних незалежно від того, де до них здійснюється доступ.

Правила люблять хороший журнал брандмауера

Такі галузі, як фінанси, охорона здоров'я та роздрібна торгівля, повинні дотримуватися суворих правил безпеки. Брандмауери допомагають компаніям дотримуватися таких норм, як **HIPAA**, **PCI-DSS** та **GDPR**, відстежуючи мережеву активність, забезпечуючи дотримання політик безпеки та створюючи звіти для аудитів.

Штучний інтелект робить брандмауери розумнішими

Старі брандмауери дотримувалися встановлених правил, але кіберзагрози змінюються занадто швидко для цього. Сучасні брандмауери використовують штучний інтелект та машинне навчання для виявлення дивної поведінки, зупинки загроз у режимі реального часу та адаптації до нових атак. У поєднанні з моніторингом мережі та безпекою кінцевих точок брандмауери на базі штучного інтелекту допомагають компаніям випереджати кіберризики.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Як мова програмування обрана Python. Python – високорівнева мова програмування загального призначення з акцентом на продуктивність розроблювача й читаність коду. Синтаксис ядра Python мінімалістичний. У той же час стандартна бібліотека включає великий обсяг корисних функцій.

Python підтримує кілька парадигм програмування, у тому числі структурне, об'єктно-орієнтоване, функціональне, імперативне й аспектно-орієнтоване. Основні архітектурні риси – динамічна типізація, автоматичне керування пам'яттю, повна інтроспекція, механізм обробки виключень, підтримка багатопоточні обчислень і зручні високорівневі структури даних. Код у Python організовується у функції й класи, які можуть поєднуватися в модулі (які у свою чергу можуть бути об'єднані в пакети).

Еталонною реалізацією Python є інтерпретатор CPython, що підтримує більшість активно використовуваних платформ. Він поширюється вільно під дуже ліберальною ліцензією, що дозволяє використовувати його без обмежень у будь-яких застосунках, включаючи пропрієтарні. Є реалізації інтерпретаторів для JVM (з можливістю компіляції), MSIL (з можливістю компіляції), LLVM і інших. Проект PyPy пропонує реалізацію Python на самому Python, що зменшує витрати на зміни мови й постановку експериментів над новими можливостями.

Python – мова програмування, що активно розвивається, нові версії (з додаванням/зміною язикових властивостей) виходять приблизно раз у два з половиною року. Внаслідок цього й деяких інших причин на Python відсутні ANSI, ISO або інші офіційні стандарти, їхня роль виконує CPython.

Python портований і працює майже на всіх відомих платформах – від КПК до мейнфреймів. Існують порти під Microsoft Windows, практично всі варіанти UNIX (включаючи FreeBSD і Linux), Plan 9, Mac OS і Mac OS X, iPhone OS 2.0 і вище, Palm OS, OS/2, Amiga, AS/400 і навіть OS/390, Symbian і Android.

При цьому, на відміну від багатьох портуємих систем, для всіх основних платформ Python має підтримку характерних для даної платформи технологій (наприклад, Microsoft COM/DCOM). Більше того, існує спеціальна версія Python для віртуальної машини Java – Jython, що дозволяє інтерпретаторові виконуватися на будь-якій системі, що підтримує Java, при цьому класи Java можуть безпосередньо використовуватися з Python й навіть бути написаними на Python. Також кілька проектів забезпечують інтеграцію із платформою Microsoft.NET, основні з яких – IronPython і Python.Net.

Python підтримує динамічну типізацію, тобто тип змінної визначається тільки під час виконання. Тому замість «присвоювання значення змінної» краще говорити про «зв'язування значення з деяким ім'ям». У Python є убудовані типи: бульові, рядки, Unicode-рядки, цілі числа довільної точності, числа із плаваючої коми, комплексні числа й деякі інші. З колекцій Python підтримує кортежі

(tuples), списки, словники (асоціативні масиви) і, починаючи з версії 2.4, безлічі. Всі значення в Python є об'єктами, у тому числі функції, методи, модулі, класи.

Додати новий тип можна або написавши клас (class), або визначивши новий тип у модулі розширення (наприклад, написаному мовою C). Система класів підтримує спадкування (одиночне й множинне) і метапрограмування. Можливе спадкування від більшості убудованих типів і типів розширень.

Всі об'єкти діляться на посилальні й атомарні. До атомарного ставляться int, long, complex і деякі інші. При присвоюванні атомарних об'єктів копіюється їхнє значення, у той час як для посилальних копіюється тільки покажчик на об'єкт, таким чином, обидві змінні після присвоювання використовують те саме значення. Посилальні об'єкти бувають змінювані й незмінні. Наприклад, рядки й кортежі є незмінними, а списки, словники й багато інших об'єктів – змінюваними. Кортеж у Python є, по суті, незмінним списком. У багатьох випадках кортежі працюють швидше списків, тому якщо ви не плануєте змінювати послідовність, то краще використовувати саме їх.

Мова має чіткий і послідовний синтаксис, продуману модульність й масштабованість, завдяки чому вихідний код написаних на Python програм легко читаємий.

Python – стабільна й розповсюджена мова. Він використовується в багатьох проектах і в різних якостях: як основна мова програмування або для створення розширень і інтеграції застосунків. На Python реалізоване велика кількість проектів, також він активно використовується для створення прототипів майбутніх програм. Python використовується в багатьох великих компаніях.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускну кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи протидії мережевим DDoS-атакам.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформувати висновки про виконаний обсяг робіт та одержані результати.

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Закрийте невикористовувані порти та приховайте невикористовувані IP-адреси

Ваша мережа повинна виглядати якомога більше схожою на «чорну скриньку» з точки зору зловмисника. Хакери часто шукають вразливості, слабкі місця та незахищені ресурси – іноді навіть ті, які ви могли пропустити під час аудиту – щоб розпочати DDoS-атаки.

Щоб мінімізувати цей ризик, створіть детальний список активних та неактивних мережевих служб і ресурсів. Потім закрийте все, що не використовується, щоб запобігти їхньому використанню зловмисниками.

Ви також захочете ускладнити зловмисникам аналіз вашої мережі. Одна з ефективних стратегій – приховати IP-адреси вашого пірингу від Traceroute, як зовні, так і зсередини. Майте на увазі, що загрози можуть виходити не лише від зовнішніх зловмисників, а й від інсайдерів – це включає ваш власний персонал або співробітників компаній-клієнтів, які розміщують свої ресурси у вашій мережі.

Для адрес, які неможливо приховати, захистіть їх за допомогою списків контролю доступу (ACL). Зверніться до свого постачальника послуг захисту від DDoS, щоб налаштувати це.

Забезпечення належної продуктивності периферійних пристроїв

Одна з найпоширеніших причин, чому мережі страждають від DDoS-атак, – це недостатньо потужні периферійні пристрої – маршрутизатори, брандмауери, балансувальники навантаження та інші компоненти інфраструктури. Ці пристрої можуть добре обробляти звичайний трафік, але навіть відносно невелика DDoS-атака може вивести їх з ладу.

Ми часто стикаємося з брандмауерами Cisco ASA (особливо старішими моделями) та маршрутизаторами MikroTik, які виходять з ладу під час атак. Багато мереж також покладаються на апаратне забезпечення для невеликих офісів або застаріле обладнання, яке колись було ефективним, але тепер нездатним впоратися із сучасними DDoS-загрозами.

Навіть найкращі рішення проти DDoS-атак не можуть відфільтрувати 100% атаки. Якщо лише 1% масштабної атаки (наприклад, 50 Гбіт/с) обійде фільтрацію, це може перевантажити периферійні пристрої в десятки або навіть сотні разів.

З увімкненою перевіркою стану пакетів (SPI) продуктивність падає ще швидше. У важких випадках пристрої перевантажуються настільки, що не можуть обробляти легітимний трафік або навіть повідомляти про свій стан, що робить усунення несправностей неможливим.

– Визначте слабкі пристрої за допомогою мережевого аудиту та замініть їх на більш продуктивні альтернативи. Без цього навіть найкращий захист від DDoS-атак не запобіжить збоям.

– Розширте пропускну здатність – сильна атака може перевантажити не лише пристрої, але й ваші інтернет-канали. Масштабування існуючої пропускної здатності та додавання резервних каналів допоможе підтримувати стабільність.

Модернізуючи периферійну інфраструктуру та забезпечуючи достатню пропускну здатність, ви можете значно покращити стійкість вашої мережі до DDoS-атак та підтримувати її працездатність навіть під час серйозних атак.

Проведіть стрес-тест

Ви можете оцінити стійкість вашої мережі до DDoS-атак, провівши стрес-тести за допомогою загальнодоступних інструментів, таких як hping3, який входить до складу багатьох дистрибутивів Linux. Ця утиліта дозволяє імітувати різні типи атак з регульованими параметрами. Однак важливо використовувати її обережно, поступово збільшуючи навантаження, щоб уникнути ненавмисних збоїв.

Стрес-тестування також слід регулярно проводити після впровадження захисту від DDoS-атак. Це допомагає оцінити, що відбувається, коли навіть невелика кількість атакуючого трафіку досягає вашої мережі. Крім того, це дозволяє оцінити швидкість реагування вашого постачальника анти-DDoS-захисту – наскільки швидко вони реагують на атаку, чи надають вони підтримку поза робочим часом і наскільки ефективно вони усувають загрозу.

Читайте також: Чому стрес-тести та інші перевірки захисту від DDoS-атак такі важливі.

Захистіть свої DNS-сервери

У першій половині 2025 року DNS- атаки стали другим за поширеністю типом DDoS-атак, одразу після HTTP-флуду. Це робить захист DNS головним пріоритетом – без нього цілеспрямована атака може спричинити нестабільність, що призведе до проблем з доступністю ресурсів для користувачів.

Якщо ваш DNS-сервер розміщено у вашій мережі, ви можете захистити його за допомогою оголошень BGP, але лише якщо ваш постачальник послуг захисту від DDoS-атак підтримує фільтрацію DNS-атак (не всі вони підтримують!). Якщо ваш постачальник пропонує цю функцію, поділіться з ним адресами своїх DNS-серверів і запросіть налаштовану фільтрацію трафіку для забезпечення стабільної роботи.

Інтегруйте захист від DDoS-атак у свою загальну стратегію безпеки

Захист від DDoS-атак має бути безперешкодно інтегрований у стратегію інформаційної безпеки (ІБ) вашої організації та загальні плани управління кіберризиками. Це не окремий захід – він має працювати синхронно з іншими процесами безпеки, щоб забезпечити комплексний захист від загроз, що постійно змінюються.

Для досягнення цієї мети захист від DDoS-атак має бути тісно узгоджений з:

– Управління вразливостями – виявлення та виправлення слабких місць, перш ніж зловмисники їх скористаються.

– Управління конфігурацією – забезпечення оптимізації налаштувань безпеки для протистояння атакам.

– Реагування на інциденти – наявність чіткого плану виявлення та зменшення DDoS-загроз у режимі реального часу.

– Моніторинг та аудит – безперервна оцінка мережевої активності для виявлення ранніх ознак атаки.

З розвитком DDoS-загроз ваша мережа також повинна адаптуватися:

– Регулярно проводити аудити безпеки та оцінки вразливостей.

– Проаналізуйте минулі атаки, щоб зрозуміти закономірності та запобігти майбутнім загрозам.

– Проведіть стрес-тести, щоб оцінити стійкість вашої мережі під навантаженням.

– Зверніться до свого постачальника анти-DDoS-захисту, щоб проактивно усунути вразливості та покращити конфігурації безпеки.

Постійно вдосконалюючи та узгоджуючи свою стратегію захисту від DDoS-атак із ширшими процесами ІБ, ви забезпечуєте довгострокову стійкість до постійно зростаючої загрози кібератак.

Як запобігти DDoS-атакам у вашій мережі: остаточний контрольний список

1. Проведіть аудит мережі та розробіть стратегію захисту від DDoS-атак.

2. Зберіть ключові мережеві дані для вашого постачальника послуг захисту від DDoS-атак.

3. Закрийте невикористовувані порти та приховайте невикористовувані IP-адреси.

4. Переконайтеся, що периферійні пристрої можуть обробляти DDoS-трафік.

5. Проведіть стрес-тести для оцінки стійкості мережі.

6. Впровадити надійний захист для DNS-серверів.

7. Інтегруйте захист від DDoS-атак у свою загальну стратегію безпеки.

Дотримуючись цих кроків, ви можете захистити свою мережу від DoS-атак і значно зменшити ризик простою через ці загрози.

3.2 Розробка структурної схеми

DDoS-атаки можуть мати шкідливі наслідки для вашої організації, зокрема:

– Фінансові невдачі: Успішна DDoS-атака може призвести до зниження продуктивності, простоїв, втрати доходів та значних витрат на пом'якшення наслідків атаки та відновлення після неї.

– Збої в роботі: DDoS-атаки можуть паралізувати основні операції вашої організації або перешкодити клієнтам у доступі до послуг.

– Шкода репутації: DDoS-атаки можуть потенційно підірвати довіру та лояльність клієнтів, змушуючи їх обирати конкурентів через неможливість доступу до бажаного веб-сайту або сіючи сумніви щодо його надійності.

– Підвищені ризики для безпеки системи**:** DDoS-атаки можуть виявити існуючі слабкі місця в корпоративній мережі, які потім можуть бути використані зловмисниками для здійснення додаткових атак або отримання несанкціонованого доступу до системи.

Хоча кінцевою метою DDoS-атаки є зробити онлайн-сервіс недоступним для користувачів, методи, що використовуються для досягнення цієї мети, можуть відрізнятися. Різні типи DDoS-атак спрямовані на різні частини мережі та класифікуються на основі рівнів мережевого з'єднання, які вони використовують. Три основні типи: об'ємна атака, атака протоколу та атака прикладного рівня.

Об'ємні атаки

Найпоширенішим типом DDoS-атаки є волюметрична атака. Цей тип атаки зосереджений на перевантаженні мережі хибними запитами даних та виснаженні пропускної здатності мережі та можливостей обробки, що призводить до відмови в обслуговуванні для законних користувачів. Це часто досягається за допомогою

ботнетів. Прикладом волюметричної атаки є посилення системи доменних імен (DNS), яке використовує відкриті DNS-сервери для надсилання багатьох DNS-запитів до цілі, що призводить до перевантаження трафіку. Атака перевантаження протоколу користувацьких дейтаграм (UDP) – це ще один тип волюметричної DDoS-атаки, метою якої є затоплення певного сервера пакетами інтернет-протоколу (IP) за допомогою UDP. Оскільки сервер не може визначити пункт призначення або цільову програму для цих пакетів, він відповідає повідомленнями «пункт призначення недоступний». Цей потік UDP-трафіку може перевантажити сервер, що призведе до перебоїв у обслуговуванні або простоїв.

Атаки на протоколи

Атаки на протоколи – це тип DDoS-атаки, спрямованої на порушення роботи сервісу шляхом використання вразливостей у протоколах, що використовуються для передачі даних. Мета полягає в перевантаженні ресурсів сервера та/або ресурсів мережевого обладнання, такого як брандмауери та балансувальники навантаження. На щастя, цей тип атаки зазвичай має чіткий слід і його легко виявити.

Прикладом протокольної атаки є атака синхронізації (SYN) перевантаженням, коли зловмисник надсилає цілі надмірну кількість запитів на з'єднання протоколу керування передачею (TCP), використовуючи підроблені вихідні IP-адреси. Цільові сервери намагаються виконати ці запити на з'єднання, але замість успішних з'єднань ціль отримує велику кількість запитів на з'єднання. Це перевантаження запитами виснажує ресурси цілі, фактично зв'язуючи систему та перешкоджаючи їй приймати легітимні з'єднання.

Атаки на рівні додатків

Атаки на рівні додатків спрямовані на слабкі місця в додатку. Ці атаки зосереджені переважно на прямому веб-трафіку та їх може бути важко виявити, оскільки машині може бути важко відрізнити їх від звичайного інтернет-трафіку з великим обсягом.

Поширеною формою атаки на рівні додатків є перевантаження через протокол передачі гіпертексту (HTTP), яке нагадує багаторазове оновлення веб-браузера на кількох комп'ютерах одночасно. Ця надмірна кількість HTTP-запитів перевантажує сервер, що призводить до відмови в обслуговуванні.

Прикладом HTTP-флуду є Slowloris, який в першу чергу націлений на веб-сервери. Під час атаки Slowloris зловмисник надсилає HTTP-запити на веб-сервер, але насправді ніколи не завершує їх. Періодично та повільно зловмисник додає додаткові заголовки, щоб продовжувати обробку запиту, так і не завершивши його. Ця стратегія змушує веб-сервер підтримувати відкриті з'єднання для цих частково завершених HTTP-запитів, що зрештою запобігає прийняттю будь-яких нових з'єднань.

Ще одним прикладом атаки на рівні додатків є ін'єкція структурованої мови запитів (SQL). За допомогою цієї форми SQL-ін'єкції зловмисники маніпулюють полями введення на веб-сайті, щоб виконувати шкідливі SQL-запити до бази даних, що споживатиме потужність веб-сервера та бази даних, а також виснажуватиме ресурси сервера.

3 мотивації DDoS-атаки

DDoS-атаки можуть бути ініційовані окремими особами, компаніями та навіть державами, кожна з яких має свої власні мотиви. Ось деякі можливі мотиви DDoS-атак:

1. Хактивізм: Хактивісти використовують DDoS-атаки як метод протесту та привернення уваги до своїх соціальних чи політичних проблем. Їхні цілі можуть включати уряди, політиків та великі бізнес-організації.

2. Вимагання: Вимагання стало популярною мотивацією для DDoS-атак, коли зловмисники вимагають викуп від своїх жертв, щоб зупинити DDoS-атаку.

3. Ідеологічні причини: Деякі зловмисники можуть ініціювати DDoS-атаки, керуючись своїми ідеологічними переконаннями. Це може включати осіб, які прагнуть порушити роботу та завдати шкоди компаніям чи організаціям, які вони вважають неетичними.

4. Кібервійна: Кібервійна зазвичай асоціюється з використанням державами DDoS-атаок, що спонсоруються ними, для отримання політичної та військової переваги. Вони спрямовані на руйнування життєво важливих фінансових, медичних та інфраструктурних систем у країнах, на які спрямовані дії. Ці стратегії передбачають залучення добре навчених фахівців з технологій та пов'язані з урядовими військовими або терористичними організаціями. Багато урядів у всьому світі інвестували значні ресурси для здійснення атак, які порушують роботу онлайн- та критично важливої інфраструктури їхніх супротивників.

5. Конкуренція в бізнесі: DDoS-атаки все частіше використовуються як стратегічний інструмент для конкурентних підприємств. Основною метою застосування такої тактики є завдання фінансової та репутаційної шкоди конкуруючим компаніям з метою порушення їхніх послуг для отримання конкурентної переваги на ринку. Ці атаки можуть приймати різні форми, починаючи від запобігання участі конкурента в онлайн-заходах і закінчуючи повним порушенням їхньої онлайн-операцій на тривалий час.

6. Помста: Деякі особи або групи, які відчувають розчарування через уявну несправедливість, можуть розпочинати DDoS-атаки як помсту проти особи чи організації.

Як виявити DDoS-атаку?

Виявлення DDoS-атаки передбачає розпізнавання ознак, які можуть свідчити про те, що ваша мережа піддається атаці. Наступні ознаки можуть потенційно вказувати на DDoS-атаку:

– Раптове та неочікуване зростання веб-трафіку з певного місця або IP-адреси

- У більшості випадків ці запити на підключення неможливо виконати, оскільки справжнє джерело IP-пакетів приховане.

– Повільна або нестабільна робота мережі, наприклад, затримка завантаження веб-сайту

- Це трапляється, коли зловмисник перевантажує сервер надмірним обсягом запитів, що призводить до помітного уповільнення роботи системи.

– Незрозумілі повідомлення про помилки сервера, тайм-аути або неможливість доступу до вашого веб-сайту

- Це трапляється, коли зловмисник завантажує ваш сервер великою кількістю запитів, що призводить до його перевантаження та виникнення помилки 503 "Служба недоступна", яка зазвичай пов'язана зі збоями в роботі сервісу. Зазвичай це вирішується самостійно, коли вхідний трафік зменшується. Однак, якщо проблема не зникає, це може свідчити про серйознішу проблему, таку як DDoS-атака.

– Працівники скаржаться на повільне з'єднання

- Це особливо актуально, якщо вони використовують те саме мережеве з’єднання, що й ваш вебсайт. У такому випадку це свідчить про те, що продуктивність мережі може бути порушена та пов’язана з DDoS-атакою.

– Зниження продуктивності інших служб, що використовують ту саму мережу

- Часто це відбувається через те, що запити зловмисника перевищують доступну пропускну здатність мережі, що призводить до уповільнення або перебоїв в роботі інших служб.

– Сповіщення від постачальника інтернет-послуг (ISP), постачальника хмарних послуг (CSP) або іншого постачальника послуг

5 стратегій пом'якшення наслідків DDoS-атак

Основною проблемою у запобіганні DDoS-атаці є розрізнення легітимного трафіку та шкідливого трафіку. Проблема виникає через безліч різних типів DDoS-атак в Інтернеті. Ці атаки можуть набувати різних форм, починаючи від атак з одного джерела до складних атак з кількох джерел.

Складні DDoS-атаки можуть використовувати кілька шляхів для перевантаження цілі, одночасно використовуючи різні методи для перенаправлення зусиль щодо пом'якшення наслідків між цими різними

маршрутами. Прикладом є одночасне націлювання на кілька рівнів стеку протоколів, наприклад, поєднання атаки посилення DNS з HTTP-флудом. Загалом, чим складніша атака, тим важче відрізнити трафік атаки від легітимного трафіку.

Зловмисники прагнуть залишатися непоміченими, щоб перешкоджати зусиллям щодо пом'якшення наслідків. Щоб ефективно протидіяти цим складним DDoS-атакам, слід впровадити багаторівневе захисне рішення для боротьби з різноманітними маршрутами атак. Ваше рішення має бути розроблене з урахуванням масштабованості, інтегрованого резервування, а також мати можливість моніторити трафік і ефективно керувати вразливостями.

Навчайте своїх співробітників

Навчання ваших співробітників є важливою частиною загальної стратегії кібербезпеки. DDoS-ботнет – це тактика, яку використовують зловмисники для компрометації мережі пристроїв шляхом дистанційного маніпулювання ними, щоб затопити ціль величезним обсягом трафіку. Зловмисники можуть використовувати пристрої нічого не підозрюючих співробітників як частину цього ботнету. Вкрай важливо навчити своїх співробітників, щоб вони розуміли, як захистити свої пристрої від такого використання.

Працівники можуть значно зменшити ризик стати учасником ботнету, дотримуючись наступних запобіжних заходів та впроваджуючи рекомендації, описані у відповідних інструкціях, зазначених нижче.

– Забезпечте регулярне оновлення ваших пристроїв та програмного забезпечення.

– Використовуйте багатофакторну автентифікацію для захисту своїх облікових записів.

– Будьте пильними щодо підозрілих електронних листів та їхніх вкладень.

– Використовуйте надійне рішення для захисту своїх пристроїв від шкідливих програм.

– Використовуйте надійну віртуальну приватну мережу (VPN).

– Резервне копіювання ваших пристроїв та інформації.

Реалізація маршрутизації чорних дір

Чорні діри – це контрзахід для пом'якшення DDoS-атаки шляхом відкидання вхідного трафіку, спрямованого на певну IP-адресу. За допомогою вашого інтернет-провайдера ваш мережевий адміністратор може встановити маршрут чорної діри, який спрямовує весь мережевий трафік на нульовий маршрут. Однак, якщо фільтрація чорних дір не має конкретних критеріїв обмеження, вона може направляти як легітимний, так і шкідливий мережевий трафік у чорну діру, назавжди видаляючи його з мережі. Маршрутизація чорних дір DDoS далеко не ідеальна, оскільки вона по суті досягає мети зловмисника, яка полягає в тому, щоб зробити мережу недоступною та потенційно спричинити втрати для бізнесу. Отже, її слід розглядати як крайній засіб, коли альтернативні методи пом'якшення виявляються неефективними. Незважаючи на свій потенціал допомогти зловмиснику досягти своїх цілей, маршрутизація чорних дір все ще може служити цінній меті, коли ціллю атаки є менший сайт у більшій мережі. У таких ситуаціях перенаправлення трафіку з цільового сайту за допомогою чорних дір може ефективно захистити більшу мережу від негативних наслідків атаки.

Впровадження обмеження швидкості

Обмеження швидкості – це ще один метод зменшення DDoS-атак, який передбачає встановлення обмежень на кількість запитів, які сервер може прийняти до певної IP-адреси протягом певного періоду часу. Це обмежить мережевий трафік і допоможе запобігти перевантаженню системних ресурсів з боку зловмисників. Впровадження обмеження швидкості – це хороший спосіб гарантувати, що законні користувачі все ще можуть отримати доступ до системних ресурсів, не перешкоджаючи загальній продуктивності програми. Хоча цей підхід сам по собі може не забезпечити повного захисту від складних DDoS-атак, він може служити цінним компонентом більш комплексної стратегії зменшення DDoS-атак.

Встановлення брандмауера веб-застосунку

Брандмауер веб-застосунків(WAF) – це захисний інструмент, який використовується для зменшення DDoS-атак на рівні додатків. Він служить зворотним проксі-сервером і створює щит між Інтернетом і вашими додатками. Він допомагає експертам з безпеки виявляти будь-який шкідливий трафік, який намагається порушити роботу ваших сервісів. WAF дозволяє вам контролювати вхідний трафік, дозволяючи або забороняючи доступ на основі попередньо визначеного набору правил безпеки. Ви можете почати з базового набору правил і налаштовувати їх у міру виявлення підозрілих закономірностей, пов’язаних з DDoS-атаками.

Забезпечити постійний моніторинг мережевого трафіку

Безперервний моніторинг (СМ) та аналіз мережевого трафіку в режимі реального часу пропонують кілька переваг для виявлення та зменшення потенційних DDoS-атак. Впровадження виявлення вторгненьСистеми виявлення вторгнень (IDS) та системи запобігання вторгненням (IPS) для постійного моніторингу мережевого трафіку ефективні у розпізнаванні та блокуванні підозрілих моделей трафіку, пов'язаних з DDoS-атаками. Використання цих інструментів аналізу трафіку дозволяє раннє виявлення DDoS-атаок, що дозволяє швидко реагувати до ескалації атак. Моніторинг допомагає створити базовий рівень нормальної активності в мережі або комп'ютерних системах. Цей базовий рівень повинен охоплювати як дні із середнім, так і з високим трафіком. Він також допомагає зрозуміти нормальну мережеву активність та моделі трафіку, що полегшує розрізнення легітимного та шкідливого трафіку та виявлення незвичайної або підозрілої активності. Цілодобовий моніторинг також дозволить виявити майбутню атаку навіть у неробочий час та вихідні.

Реалізація розповсюдження мережі anycast

Одноадресна маршрутизація, що широко використовується в мережевому зв'язку завдяки своїй простоті та універсальності, обслуговує різні програми, такі як перегляд веб-сторінок, електронна пошта та передача файлів. В моделі

одноадресної розсилки кожен мережевий вузолабо пристрою призначається унікальна IP-адреса, що забезпечує прямий та ефективний зв'язок по мережі. Однак, незважаючи на свою простоту, одноадресний зв'язок не є стійким до DDoS-атак. Оскільки трафік спрямовується безпосередньо до певного центру обробки даних, DDoS-атака може перевантажити це місце або навколишню інфраструктуру надмірним трафіком. Такий сплеск може призвести до відмови в обслуговуванні, що ускладнить виконання законних запитів. На відміну від одноадресної маршрутизації, мережеве поширення Anycast є більш стійким завдяки своїм унікальним характеристикам маршрутизації та адресації. Anycast розподіляє вхідний трафік по мережі серверів, розподілених у різних місцях, використовуючи одну й ту саму IP-адресу. Цей метод розширює покриття мережі, запобігаючи перевантаженню будь-якого місця шкідливими запитами. Коли на адресу надсилається надзвичайно великий обсяг трафіку, наприклад, під час DDoS-атаки, трафік автоматично перенаправляється до найближчого доступного місця в мережі, тим самим мінімізуючи вплив на основну інфраструктуру.

Маршрутизація Anycast підвищує стійкість мережі, роблячи атаки більш керованими та зменшуючи їхній потенціал для збоїв, тим самим забезпечуючи безперебійну доступність послуг. Широко розповсюджена конфігурація мережі ускладнює для зловмисників виконання DDoS-атаки, оскільки це вимагає значних ресурсів для ефективного надсилання шкідливого трафіку через ботнет.

Проведіть оцінку ризиків

Оцінка ризиків дозволить вам оцінити вразливість вашої організації до DDoS-атак. Вам слід регулярно проводити оцінки ризиків та аудити вашої мережевої інфраструктури, щоб виявити вразливості. Хоча повністю запобігти DDoS-атаці неможливо, повне розуміння апаратних та програмних активів вашої організації, включаючи їхні сильні та слабкі сторони, має вирішальне значення для забезпечення належного захисту. Визначення найбільш вразливих зон у вашій мережі є важливим для визначення найефективнішої стратегії пом'якшення впливу DDoS-атаки.

Проводячи оцінку ризиків, ви:

– Визначте критично важливі активи вашої організації та їх важливість для забезпечення безперервної роботи.

– Аналізуйте та оцінюйте потенційні загрози, що стосуються діяльності вашої організації.

– Визначте вразливості мережі вашої організації, включаючи слабкі місця, які можуть використовувати зловмисники, та оцініть вплив і ймовірність DDoS-атаки на основі історичних даних, розвідки загроз та галузевих тенденцій.

– Визначте різні шляхи, які зловмисники можуть використовувати для ініціювання DDoS-атаки, включаючи такі методи, як UDP-флуд, SYN-флуд або HTTP-флуд.

– Розставте пріоритети між виявленими ризиками, враховуючи такі фактори, як ймовірність здійснення атаки, потенційні наслідки атаки та ймовірність як виявлення, так і пом'якшення наслідків атаки.

Розробка плану реагування на DDoS-атаку

Для ефективної підготовки до DDoS-атаки вкрай важливо мати добре структурований план реагування. Цей план повинен містити чіткі кроки, які допоможуть виявити, пом'якшити та відновитися після атаки. Ваш план також має бути спрямований на мінімізацію впливу на вашу організацію та забезпечення безперебійного або мінімального простою у вашій бізнес-операції.

У рамках цього плану слід враховувати такі компоненти:

– Чітко окресліть та задокументуйте ролі та обов'язки всіх членів команди, які реагуватимуть на DDoS-атаку, включаючи внутрішніх зацікавлених сторін, керівників організації та мережевих адміністраторів, а також будь-яких залучених постачальників послуг.

– Розробіть вичерпний контрольний список, який визначає процеси та дії, необхідні під час DDoS-атаки. Вкажіть необхідні інструменти та ресурси, які будуть потрібні, та визначте осіб, з якими потрібно зв’язатися.

– Розробіть надійний план комунікації, який окреслює заздалегідь визначений ланцюжок зв'язку, якого слід дотримуватися у разі DDoS-атаки.

– Регулярно проводите навчання з реагування на інциденти та переконайтеся, що ваш план реагування на DDoS-атаки є невід'ємною частиною загальної стратегії аварійного відновлення та плану забезпечення безперервності бізнесу вашої організації.

Зверніться до постачальника послуг захисту від DDoS-атак

Якщо ваша організація має обмежені ресурси для управління кібербезпекою, ви можете розглянути можливість співпраці зі сторонніми організаціями для посилення захисту від кіберзагроз. Вони можуть пропонувати різні послуги захисту, включаючи очищення DDoS-трафіку, яке може допомогти захистити ваш інтернет-трафік від DDoS-атаки. Очищення DDoS-трафіку передбачає фільтрацію вхідного трафіку для виявлення та видалення шкідливих даних, дозволяючи лише легітимному трафіку потрапляти до цільової мережі. Це дозволить вам підтримувати онлайн-присутність під час атак, не втрачаючи зв'язку.

Більшість інтернет-провайдерів та постачальників послуг зв'язку пропонують певний рівень захисту від DDoS-атак. Вам слід дізнатися про захисні заходи, які вони надають, та переглянути угоду про надання послуг, щоб визначити будь-які потенційні обмеження їхнього покриття.

Використання хмарних рішень для запобігання DDoS-атак також може запропонувати багато переваг. До них належать спеціалізований персонал, який забезпечує швидший час реагування у разі атаки, та висока пропускна здатність мережі, що робить їх більш стійкими до DDoS-атак на основі обсягів. Ці рішення також можуть забезпечувати автоматичні варіанти реплікації або резервного копіювання, що дозволяє вам запускати свої сервіси, не перериваючи роботу користувачів.

Якщо вам потрібне ще надійніше рішення для захисту від DDoS-атак, зверніться до постачальника керованих послуг (MSP), щоб дослідити рішення,

адаптовані до потреб вашої організації для захисту від DDoS-атак. Ці служби вміють активно моніторити ваш мережевий трафік, виявляти будь-які ознаки атаки, визначати її походження та вживати заходів для перенаправлення шкідливого трафіку з вашої мережі.

Звернення до постачальника керованих послуг (MSP) для захисту від DDoS-атак пропонує численні переваги. MSP, що спеціалізуються на кібербезпеці, надають досвід, передові технології та цілодобовий моніторинг для раннього виявлення загроз. Вони швидко реагують на атаки та масштабують послуги відповідно до потреб мережі. MSP постійно оновлюють системи, щоб випереджати нові загрози, забезпечуючи стратегічний та ефективний підхід до захисту онлайн-сервісів. Довіривши захист від DDoS-атак постачальнику керованих послуг (MSP), ваша внутрішня IT-команда може зосередитися на основних бізнес-операціях, а не постійно моніторити та реагувати на потенційні кіберзагрози.

Рисунок 3.1 – Структурна схема системи

3.3 Розробка функціональної схеми

Функціональна схема розробленої системи зображена на рисунку 3.2. Так, як функціональна схема є більш подобним описом функціональних можливостей структурної схеми, то вона буде представляти собою, більш детальний варіант структурної схеми.

З рисунку видно, що розроблена система складається з наступних частин:

– Блок моніторингу мережі.

– Блок аналізу мережної статистики.

– Блок визначення топології мережі.

– Блок виявлення аномального поводження трафіку.

– Блок визначення виду атаки.

– Блок зберігання результатів.

Блок аналізу мережної статистики

Блок збирання наступної інформації:

– Основна статистика (Summary).

– Ієрархія протоколу (Protocol Hierachy).

– Сеанси обміну пакетами (Conversations).

– Точки призначення (Endpoints).

– Графіки I/O (IO Graphs).

– Список сеансів обміну пакетами (Conversation List).

– Список точок призначення (Endpoint List).

– Час чекання відповіді від сервісу (Service Response Time).

– RTP.

– SIP.

– Виклики VoIP (VoIP Calls).

– Призначення (Destination).

– Графік потоку (Flow Graph).

– HTTP.

– IP-адреса (IP address).

– Довжина пакету (Packet Length).

– Тип порту (Port Type).

Розпишемо їх більш детально.

1. Основна статистика. Доступні такі елементи основної статистики, як:

– Властивості захоплених файлів.

– Час захвату.

– Інформація про фільтр захвату.

– Інформація про фільтр відображення.

2. Ієрархія протоколу. Статистика ієрархії протоколу допомагає аналізувати пакети, розбиваючи відображені дані, які належать чинному рівню OSI.

3. Сеанси обміну пакетами. Якщо ви використовуєте протокол TCP/IP або програму, яка працює із цим протоколом, ви маєте побачити чотири активних вкладок для обміну пакетами за допомогою Ethernet, IP, TCP та UDP. «Діалог» між комп'ютерами відображає трафік між двома активними хостами. Номер, зазначений на вкладці після назви протоколу, означає кількість «діалогів» між хостами. Номер, зазначений на вкладці після назви протоколу, означає кількість «діалогів» між хостами, наприклад, «Ethernet:6».

4. Точки призначення. Точки призначення забезпечують статистику даними про відправку та прийом пакетів. Номер, зазначений на вкладці після назви протоколу, вказує на кількість точок призначення. Наприклад, «Ethernet:6».

5. Графіки I/O. Основний графік може бути отриманий за допомогою команди «IO graphs» (Графіки I/O). Ще декілька графіків можуть бути додані у тому ж вікні на основі фільтрів відображення.

6. Час чекання відповіді від сервісу. 13 протоколів доступні для глибокого аналізу.

7. RTP. RTP (Real-time Transport Protocol, протокол передачі у реальному часі, RFC 3550) – це протокол для передачі звука та відео через IP-мережу. Він працює у початку протоколу дейтаграм користувача (User Datagram Protocol,

UDP). Він часто використовується у сукупності з протоколами SIP або H.233, забезпечуючи виконання сигнальних завдань.

8. SIP. SIP (Session Initiation Protocol, протокол встановлення сесії, RFC 3261) – це сигнальний протокол, який оголошує відео– або VoIP-сесії. Він працює разом із протоколом RTP, який використовується для передачі мультимедійних даних.

9. Виклики VoIP.

VoIP (Voice over IP, голосовий зв'язок за допомогою Інтернету) взагалі використовує два типи протоколів:

– сигнальні протоколи, такі, як SIP або H.323

– переносні протоколи, наприклад, RTP

10 Призначення. Відображення усіх IP-адрес призначення мережевих пакетів.

11. Графік потоків. Графіки потоків забезпечує послідовний аналіз TCP-з'єднань. Перші три строки містять оголошення TCP– з'єднання з послідовностями «SYN», «SYN ACK» та «ACK».

12 HTTP. HTTP (Hypertext Transfer Protocol, протокол передачі гіпертексту) – це протокол типу «клієнт-сервер», який використовується для передачі HTML-файлів. HTTP-клієнт (у більшості випадків це web-браузер) відсилає HTTP-запит до web-серверу із полем «URL», який допомагає знайти потрібний файл. Web-сервер відповідає HTTP-пакетом та забезпечує клієнт необхідною web-сторінкою.

Меню «HTTP» містить три підменю:

– «Load Distribution» (Розподіл пакетів).

– «Packet Counter» (Лічильник пакетів).

– «Requests» (Запити).

14 IP-адреса. Відображення IP-адреси джерела або призначення мережевих пакетів.

15. Довжина пакету.

16. Тип порту. Відображення статистики портів TCP або UDP.

Блок визначення топології мережі

Блок визначення топології мережі:

– Блок використання відомостей із загальної системи моніторингу мережі, а не опитування пристрою додатково.

– Блок складання списку пристроїв у мережі, автоматично, ґрунтуючись на дані системи моніторингу.

– Блок побудови топології мережі, за станом на задану дату й відстеження змін у топології протягом часу.

– Блок автоматичного визначення рівнів ієрархії пристроїв у мережі, з виділенням периферійних, проміжних і центральних вузлів;

– Блок побудови топології мережі, незалежно від використовуваної системи моніторингу й програмно-апаратних платформ;

– Блок комбінувати показників, на основі яких визначаються зв'язки між пристроями, і при їхньому обчисленні виконувати перевірку на значимість із використанням статистичних критеріїв.

Блок виявлення аномального поводження трафіку

Блок виявлення аномального поводження трафіку:

– Блок визначення профілю поводження нормального трафіку.

– Блок заміни направлення трафіку.

– Блок усунення аномального трафіку.

При первісному розгортанні рішення по DDoS адміністратор створює профіль поводження нормального трафіку. Цей процес іменується навчанням. Компанія використовує додатки звичайним образом протягом 24 годин протягом одного тижня, і трафік додатка проходить через Детектор аномалій трафіку. У період навчання Детектор аномалій трафіку збирає базову інформацію для розуміння нормальної роботи мережі, куди входять:

– Інтенсивність пакетів для кожного типу пакетів, обмірювана як кількість пакетів у секунду (pps).

– Співвідношення пакетів, наприклад, співвідношення пакетів SYN і пакетів FIN.

– Кількість одночасних TCP-з'єднань, відкритих одним джерелом.

Базова інформація збирається по кожній цільовій адресі хост-ПК, цільовій підмережі, вихідній адресі хост-ПК і вихідній підмережі.

Після закінчення періоду навчання Детектор аномалій трафіку переводиться в режим моніторингу, а Блок усунення аномального трафіку – у резервний режим готовності. Доти, поки немає атаки, що активно розвивається, вхідний трафік з мережі Інтернет проходить через комутатор без якого-небудь втручання з боку Блоку усунення аномального трафіку. Копія вхідного трафіку посилає для аналізу на Детектор аномалій трафіку через зовнішній аналізатор протоколів (SPAN) або віртуальні списки ACL. Якщо Детектор аномалій трафіку виявляє аномальне в порівнянні з базовою інформацією поводження трафіку, починається процес усунення:

– Детектор аномалій трафіку направляє в Блок усунення аномального трафіку команду почати процес зміни напрямку.

– Блок усунення аномального трафіку відхиляє (“захоплює”) трафік, адресований на атакуєму IP-адресу, переадресуючи його на самого себе.

– Блок усунення аномального трафіку піддає трафік багатоступінчастому аналізу й застосовує контрзаходи для відділення благонадійних джерел від джерел атаки. Цей процес іменується очищенням або вичищенням.

– Блок усунення аномального трафіку скидає трафік атаки й пересилає благонадійний трафік назад на нормальний маршрут проходження трафіку до мети. Цей процес іменується ін'єкцією.

Детектор аномалій трафіку

Детектор аномалій трафіку – це пасивний пристрій моніторингу, що постійно виявляє ознаки, що вказують на присутність атаки DDoS, спрямованої проти захищеного місця призначення, також іменованого зоною. Це може бути сервер, інтерфейс міжмережного екрана або інтерфейс маршрутизатора. Детектор

аномалій трафіку аналізує копії всього вхідного трафіку, адресуємого в захищені зони, через SPAN або відгалуження пасивної мережі. Цей аналіз включає зіставлення поточного поводження трафіку з базовими граничними параметрами, які також іменуються зональною політикою, для виявлення аномального поводження трафіку. Якщо аномальне поводження виявлене й виглядає як можлива атака, Детектор аномалій трафіку через позаполосну управлінську мережу Ethernet посилає в Блок усунення аномального трафіку сигнал про початок аналізу й усунення атаки.

Блок усунення аномального трафіку

Блок усунення аномального трафіку – це автономний пристрій аналізу й фільтрації трафіку. Починаючи прийом трафіку, адресованого в конкретну зону, що, очевидно, піддається атаці, Блок усунення аномального трафіку проводить точний аналіз цього трафіку. Якщо результати аналізу підтверджують, що трафік злочинний, Блок усунення аномального трафіку застосовує контрзаходи, наприклад, механізми анти-спуфінга й фільтрацію різного рівня (таблиця 3.1). Кінцевий результат полягає в тому, що трафік зі злочинних джерел скидається, а трафік із благонадійних джерел пересилається в передбачений пункт призначення.

Можливі варіанти зміни напрямку трафіку

Фахівці з IT можуть використовувати описані нижче варіанти зміни напрямку трафіку з його пересиланням з мережі, розташованого вище лежачого оператора зв'язку, на Блок усунення аномального трафіку. Цей процес також іменується “захватом” трафіку:

– Повідомлення прикордонного шлюзового протоколу (Border Gateway Protocol, BGP) із Блок усунення аномального трафіку на маршрутизатори, розташовані у вище лежачого оператора зв'язку, з інформацією про те, що трафік, адресований на захищену адресу призначення, буде переспрямований на Блок усунення аномального трафіку.

– Використання зовнішніх механізмів зміни напрямку трафіку, наприклад, маршрутизаторів віддаленого відновлення BGP.

– Повідомлення про ін'єкцію очищеного трафіку на маршруті (Route Health Injection, RHI) від Блок усунення аномального трафіку для процесу маршрутизації в Catalyst серії 6500 або в систему нагляду серії 7600. Ці повідомлення поміщають статичний маршрут у глобальну таблицю маршрутизації, у якій модуль Блок усунення аномального трафіку позначений як наступний вузол.

Можливі варіанти ін'єкції трафіку

Ін'єкція трафіку – це процес, застосовуваний у Блок усунення аномального трафіку для пересилання очищеного благонадійного трафіку в точку призначення, що піддається атаці. Рішення підтримує різні варіанти ін'єкції трафіку. У варіанті 2-ого рівня топології, очищений трафік пересилається із Блок усунення аномального трафіку на статично-конфігуруєму наступну адресу заходу. Ця адреса перебуває на маршрутизаторі, розташованому нижче й з'єднаним з тої ж VLAN або підмережею, що й інтерфейс/VLAN ін'єкції трафіку. Ін'єкцію трафіку на 2-му рівні найпростіше конфігурувати, оскільки тут не потрібно вносити які-небудь істотні зміни в конфігурацію маршрутизатора, розташованого нижче.

Варіанти ін'єкції трафіку 3-го рівня:

– Маршрутизація й пересилання по VPN (VPN Routing and Forwarding, VRF).

– Маршрутизація на основі політики (Policy-Based Routing, PBR).

– Транкинг VLAN (VLAN Trunking).

– Інкапсуляція по загальній маршрутизації (GRE) або інкапсуляція IP у тунелі IP (IPIP).

Атаки DDoS – Виявлення й усунення

У таблиці 3.1 перераховані типи атак DDoS, які може виявляти й усувати Блок усунення аномального трафіку.

Таблиця 3.1 – Категорії й особливі типи атак DDoS

Категорія атаки	Особливі типи атак
Атаки із заповненням смуги пропущення	Лавинні атаки зі спуфінгом або без спуфінга: – Прапор TCP (SYN, SYN-ACK, ACK, FIN). – Протокол керування повідомленнями в Інтернет (ICMP). – Протокол користувальницьких датаграмм (UDP). Приклади: лавинна атака SYN, smurf, LAND і UDP – лавинні атаки.
	Атаки зомбі-комп'ютерів/мереж зомбі-комп'ютерів, у яких кожний вихідний зомбі-ПК або мережа відкриває множинні TCP-з'єднання й, у деяких випадках, видає багаторазові запити HTTP.
	Атаки DNS, наприклад, лавинна атака із запитами DNS.
Атаки з дефіцитом ресурсів	Атаки пакетного розміру, характерна риса яких – фрагментированні або великі пакети. Приклади: teardrop і ping-of-death.
	Атаки зомбі-комп'ютерів/мереж зомбі-комп'ютерів з низькою інтенсивністю схожі на атаки із заповненням смуги пропущення за тим виключенням, що кожне джерело атаки посилає множинні запити з невеликим обсягом в одиницю часу.
	Атаки DNS з рекурсивним переглядом DNS.

Блок визначення виду атаки

Блок визначення виду атаки:

– Атака ARP-spoofing на таблицю mac-адрес комутаторів.

– Широкомовний шторм.

– Додатки, що роблять інтенсивне широкомовне розсилання, наприклад: широкомовні чати й мережні ігри.

ARP-spoofing

ARP-spoofing – техніка атаки в Ethernet мережах, що дозволяє перехоплювати трафік між хостами. Заснована на використанні протоколу ARP.

При використанні в розподіленій обчислювальній системи (РВМ) алгоритмів віддаленого пошуку існує можливість здійснення в такій мережі типової віддаленої атаки «помилковий об'єкт РВМ». Аналіз безпеки протоколу ARP показує, що, перехопивши на атакуючому хості усередині даного сегмента мережі широкомовний ARP-запит, можна послати помилкову ARP-відповідь, у якій оголосити себе шуканим хостом (наприклад, маршрутизатором), і надалі активно контролювати мережний трафік дезінформованного хосту, впливаючи на нього за схемою «помилковий об'єкт РВМ».

Протокол ARP призначений для перетворення IP-адрес в MAC-адреси. Найчастіше мова йде перетворенні в адреси Ethernet, але ARP використовується й у мережах інших технологій: Token Ring, FDDI і інших.

Алгоритм роботи ARP

Протокол може використовуватися в наступних випадках:

1. Хост А хоче передати IP-пакет вузлу В, що перебуває з ним в одній мережі.

2. Хост А хоче передати IP-пакет вузлу В, що перебуває з ним у різних мережах, і користується для цього послугами маршрутизатора R.

У кожному із цих випадку вузлом А буде використовуватися протокол ARP, тільки в першому випадку для визначення MAC-адреси вузла В, а в другому – для визначення MAC-адреси маршрутизатора R. В останньому випадку пакет буде переданий маршрутизатору для подальшої ретрансляції.

Далі для простоти розглядається перший випадок, коли інформацією обмінюються вузли, що перебувають безпосередньо в одній мережі. (Випадок коли пакет адресований вузлу, який знаходиться за маршрутизатором,

відрізняється тільки тим, що в пакетах переданих після того як ARP-перетворення завершене, використовується IP-адреса одержувача, але MAC-адреса маршрутизатора, а не одержувача.)

Проблеми ARP

Протокол ARP є абсолютно незахищеним. Він не має ніякого способу перевірки дійсності пакетів: як запитів, так і відповідей. Ситуація стає ще більш складною, коли може використовуватися мимовільний ARP (gratuitous ARP).

Мимовільний ARP – таке поводження ARP, коли ARP-відповідь надсилається, коли в цьому (з погляду одержувача) немає особою необхідності. Мимовільна ARP-відповідь це пакет-відповідь ARP, присланий без запиту. Він застосовується для визначення конфліктів IP-адрес у мережі: як тільки станція одержує адресу по DHCP або адреса привласнюється вручну, розсилається ARP-відповідь gratuitous ARP.

Мимовільний ARP може бути корисний у наступних випадках:

– Відновлення ARP-таблиць, зокрема, у кластерних системах.

– Інформування комутаторів.

– Повідомлення про включення мережного інтерфейсу.

Незважаючи на ефективність мимовільного ARP, він є особливо небезпечним, оскільки з його допомогою можна запевнити віддалений вузол у тому, що MAC-адреса якої-небудь системи, що перебуває з нею в одній мережі, змінилася й указати, яка адреса використовується тепер.

До виконання ARP-spoofing'а в ARP-таблиці вузлів A і B існують записи з IP- і MAC-адресами один одного. Обмін інформацією виробляється безпосередньо між вузлами A і B.

У ході виконання ARP-spoofing'а комп'ютер **C**, що виконує атаку, відправляє ARP-відповіді (без одержання запитів):

– вузлу A: з IP-адресою вузла B і MAC-адресою вузла C;

– вузлу B: з IP-адресою вузла A і MAC-адресою вузла C.

У силу того що комп'ютери підтримують мимовільний ARP (gratuitous ARP), вони модифікують власні ARP-таблиці й поміщають туди записи, де

замість справжніх MAC-адрес комп'ютерів A і B коштує MAC-адреса комп'ютера C.

Після того як атака виконана, коли комп'ютер A хоче передати пакет комп'ютеру B, він знаходить в ARP-таблиці запис (він відповідає комп'ютеру C) і визначає з її MAC-адресу одержувача. Відправлений по цьому MAC-адресу пакет приходить комп'ютеру C замість одержувача. Комп'ютер C потім ретранслює пакет тому, кому він дійсно адресований – тобто комп'ютеру B.

Широкомовний шторм

Широкомовний шторм – лавина (сплеск) широкомовних акетів (на другому рівні моделі OSI – кадрів). Розмноження некоректно сформованих широкомовних повідомлень у кожному вузлі приводить до експонентного росту їхнього числа й паралізує роботу мережі. Звичайно такі пакети використовуються мережними сервісами для оповіщення станцій про свою присутність. Вважається нормальним, якщо широкомовні пакети становлять не більше 10% від загального числа пакетів у мережі.

Також досить часто до шторму приводять кільця в мережі при некоректному налаштуванні протоколу Spanning Tree, оскільки в заголовку пакетів Ethernet немає інформації про час життя кадру, як, наприклад, у пакетів IP. Крім цього широкомовний шторм застосовується (навмисно) зломщиками.

Відповідно до галузевого стандарту де-факто число широкомовних і багатоадресних кадрів у мережі не повинне перевищувати 8-10% від загального числа кадрів.

Широкомовний кадр – це кадр, адресований всім станціям у домені мережі. Багатоадресний кадр – це кадр, адресований групі станцій у домені мережі. Оскільки широкомовний кадр адресований всім станціям, то, одержавши його, станції повинні перервати свою роботу й обробити такий кадр. Це сповільнює роботу всієї мережі.

Якщо відношення числа широкомовних кадрів до загального числа кадрів більше 10%, то такий ефект називається "широкомовним штормом".

Рисунок 3.2 – Функціональна схема системи

Широкомовний шторм може бути наслідком дефектів устаткування або неправильного налаштування параметрів активного встаткування.

Найчастіше це явище спостерігається в розподілених мережах NetWare, побудованих на основі комутаторів, або коли дані між сегментами або доменами мережі можуть передаватися більш ніж по одному потенційному шляху. Якщо один з комутаторів такої мережі не підтримує протокол Spanning Tree (звичайно IEEE 802.1d) або останній неправильно настроєний або збоїть, то в мережі починається некерована циркуляція широкомовних кадрів.

Виявлення "широкомовного шторму" є не настільки тривіальним завданням, як це може здатися на перший погляд. Для його виявлення недостатньо взяти загальне число широкомовних кадрів і поділити його на загальне число кадрів, що пройшли по мережі.

Для цього ви повинні визначити: яку частку становлять широкомовні кадри в кожний інтервал часу (наприклад, за одну мінуту) і яка при цьому утилізація каналу зв'язку. Якщо, наприклад, за одну мінуту по мережі пройшло 4 кадри, а 2 з них були широкомовними, то це ще не виходить, що ви спостерігаєте "широкомовний шторм".

Захист від широкомовних штормів (broadcast storm)

Одна з характерних несправностей мережного програмного забезпечення – мимовільна генерація з високою інтенсивністю широкомовних пакетів. Широкомовним штормом вважається ситуація, у якій відсоток широкомовних пакетів перевищує 20% від загальної кількості пакетів у мережі. Звичайний комутатор або міст сліпо передає такі пакети на всі свої порти, як того вимагає його логіка роботи, засмічуючи, таким чином, мережу.

Боротьба із широкомовним штормом у мережі, з'єднаної комутаторами, жадає від адміністратора відключення портів, що генерують широкомовні пакети. Маршрутизатор не поширює такі ушкоджені пакети, оскільки в коло його завдань не входить копіювання широкомовних пакетів в усі поєднувані їм мережі.

Тому маршрутизатор є прекрасним засобом боротьби із широкомовним штормом, щоправда, якщо мережа розділена на достатню кількість підмереж.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма взаємодії процесів системи, розробленої у результаті виконання магістерської роботи, наведена на рисунку 3.3.

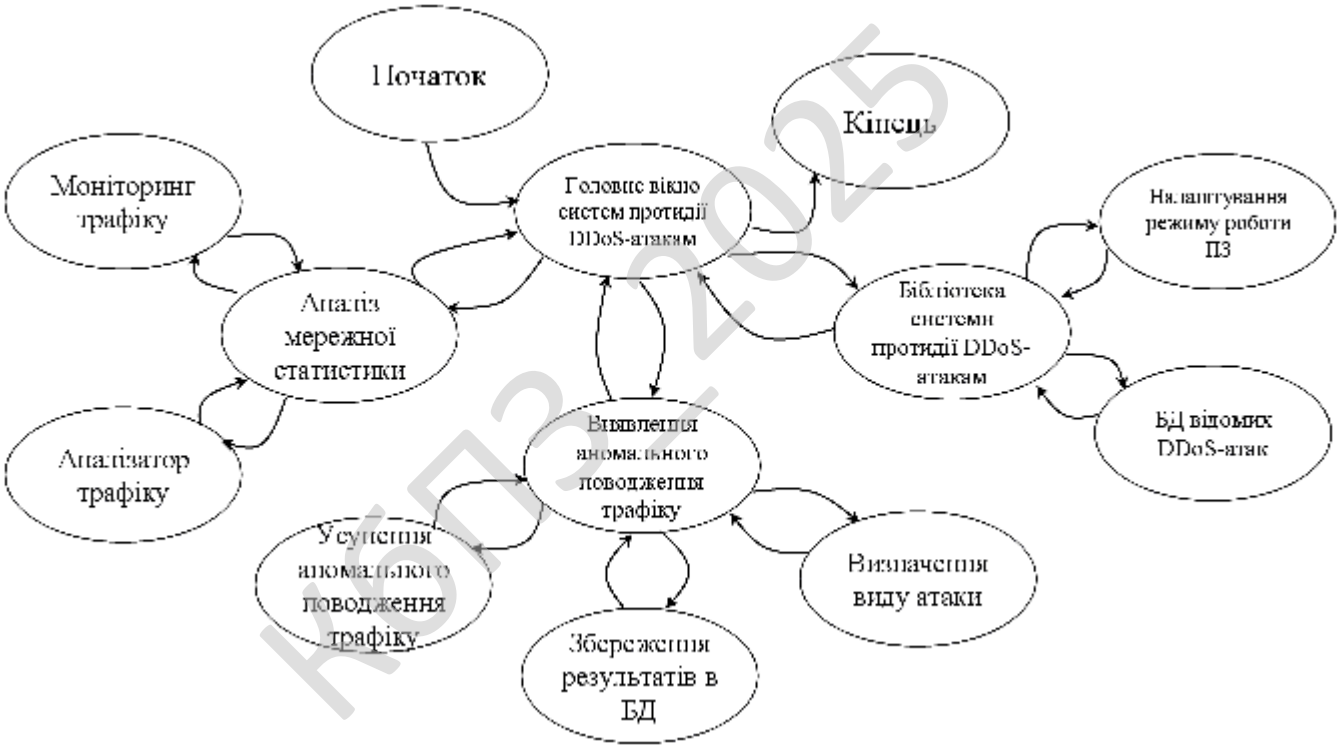

Рисунок 3.3 – Діаграма взаємодії процесів

При детальному її розгляді можна побачити як саме проходе взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування). Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи. Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

Діаграми потоків даних містять чотири типи елементів:

– Процеси які являють собою трансформацію даних в рамках описуваної системи.

– Сховища даних (репозиторії).

– Зовнішні по відношенню до системи сутності.

– Потоки даних між елементами трьох попередніх типів.

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Блок-схеми є першоджерелами стратегії розвитку ПЗ. Тому від точності і детальної блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об’єктно-орієнтована що вимагає оптимізації програми високого рівня, також те, що при розробці програми слід надати особливу увагу модулю систем протидії DDoS-атакам.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підпрограми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ. При роботі підпрограми виконується основний функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

Було використано підходи з використанням UML, це уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування. Є невід'ємною частиною уніфікованого процесу розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, називаної UML-моделлю. UML був створений для визначення,

візуалізації, проектування й документування в основному програмних систем. UML не є мовою програмування, але в засобах виконання UML-моделей як інтерпретованого коду можлива кодогенерація.

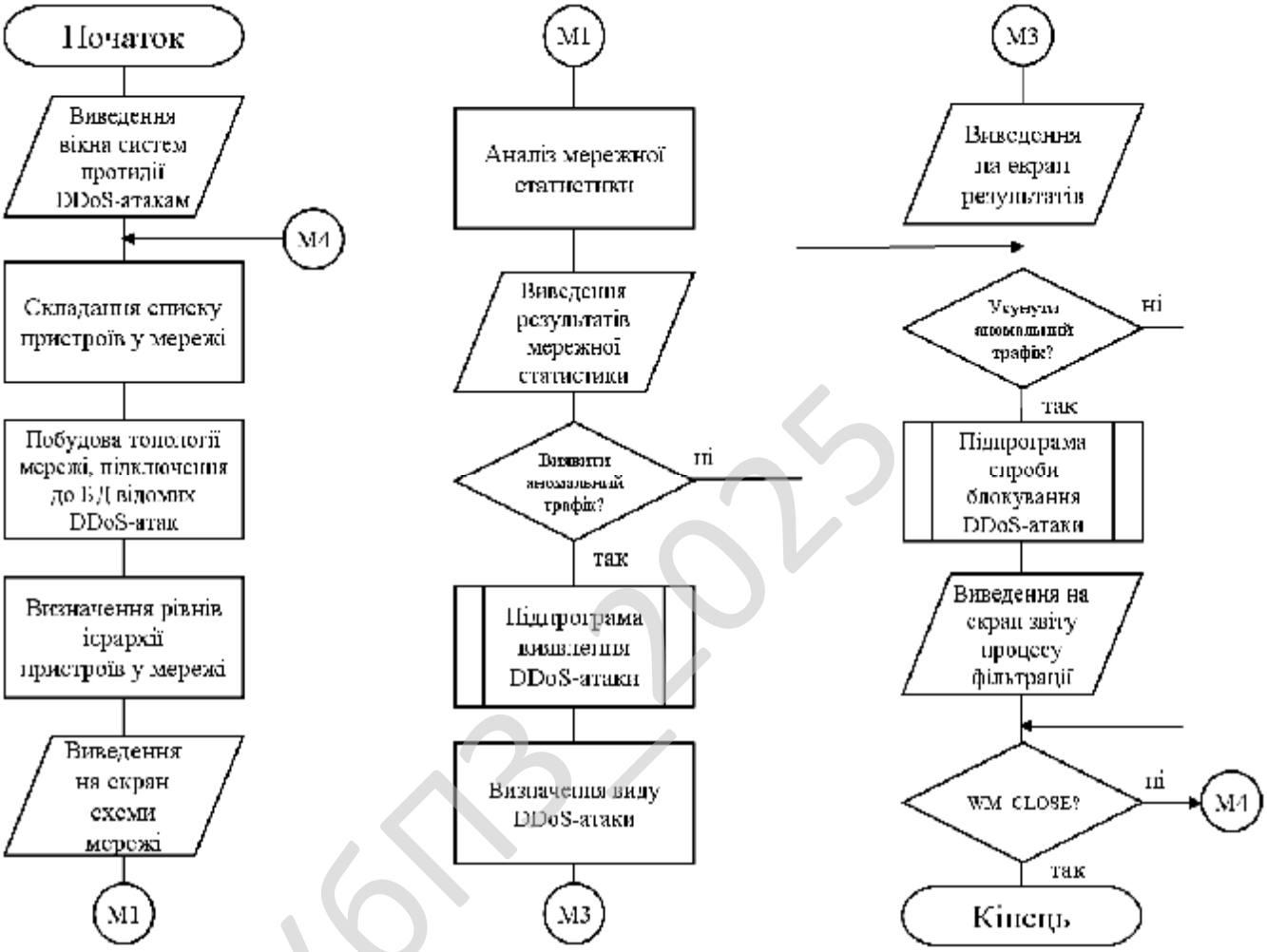

Рисунок 4.1 – Блок-схема основної програми

UML може бути застосовано на всіх етапах життєвого циклу аналізу бізнес-систем і розробки прикладних програм. Різні види діаграм які підтримуються UML, і найбагатший набір можливостей представлення певних аспектів системи робить UML універсальним засобом опису як програмних, так і ділових систем.

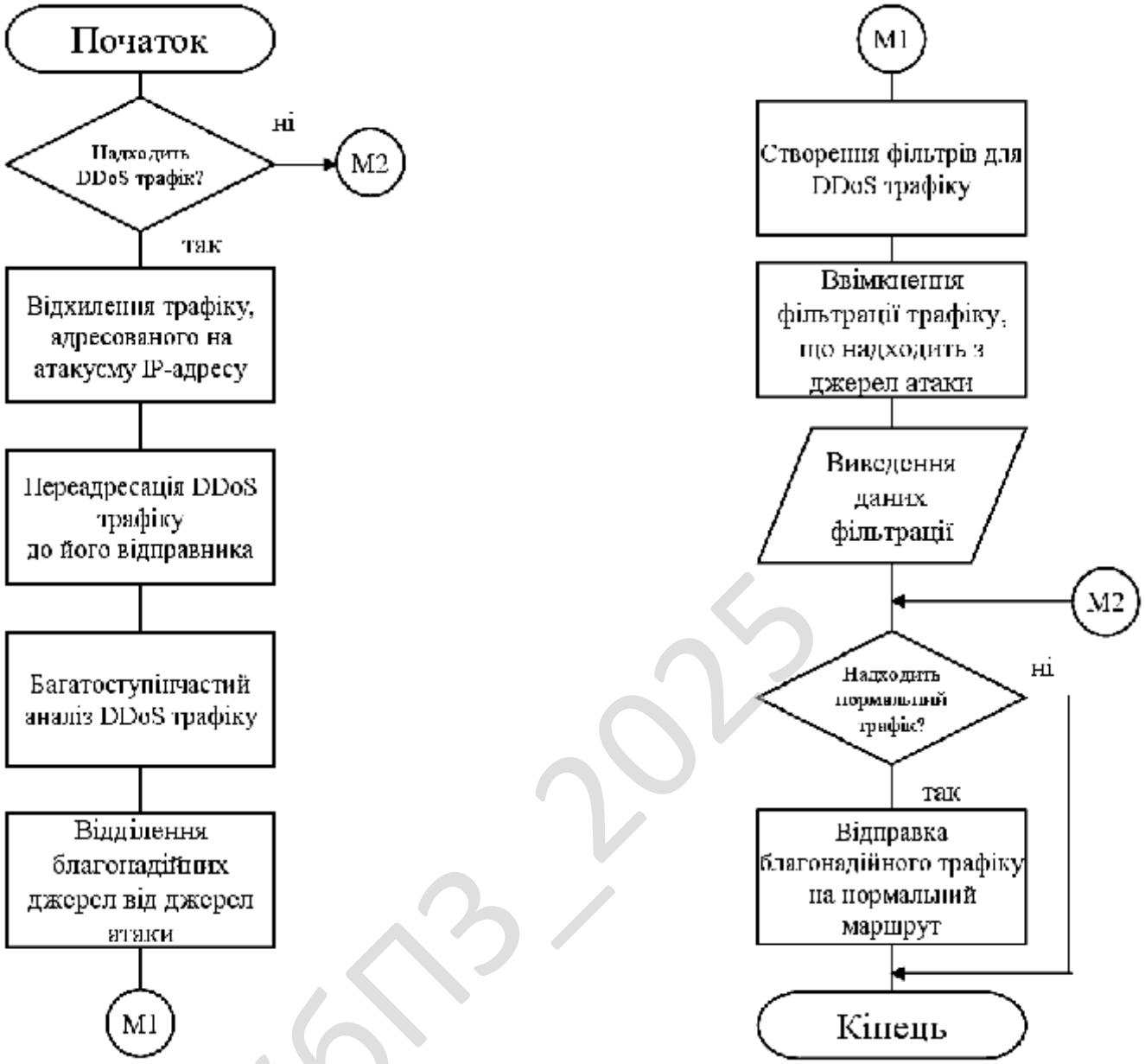

Рисунок 4.2 – Блок-схема роботи підпрограми

Діаграми дають можливість представити систему (як ділову, так і програмну) у такому вигляді, щоб її можна було легко перевести в програмний код. Основною причиною використання мови UML є спілкування розробників між собою.

Крім того, UML спеціально створювалася для оптимізації процесу розробки програмних систем, що дозволяє збільшити ефективність їх реалізації у кілька разів і помітно поліпшити якість кінцевого продукту.

UML прекрасно зарекомендувала себе в багатьох успішних програмних проектах. Засоби автоматичної генерації кодів дозволяють перетворювати моделі мовою UML у вихідний код об’єктно-орієнтованих мов програмування, що ще більш прискорює процес розробки. Практично усі CASE-засоби (програми автоматизації процесу аналізу і проектування) мають підтримку UML. Моделі розроблені в UML, дозволяють значно спростити процес кодування і направити зусилля програмістів безпосередньо на реалізацію системи.

Діаграми підвищують супроводжуваність проекту і полегшують розробку документації.

UML необхідний:

– Керівникам проектів, які керують розподілом завдань і контролем за проектом.

– Проектувальникам інформаційних систем які розробляють технічні завдання для програмістів.

– Бізнес-аналітикам, які досліджують реальну систему і здійснюють інжиніринг і реінжиніринг бізнесу компанії.

– Програмістам які реалізовують модулі інформаційної системи.

При модифікації системи об'єктний підхід дозволяє легко включати в систему нові об'єкти і виключати застарілі без істотної зміни її життєздатності. Використання побудованої моделі при модифікаціях системи дає можливість усунути небажані наслідки змін, оскільки вони не ламають структури системи, а тільки змінюють поведінку об'єктів.

Також при розробці магістерської роботи було використано наступні підходи UML: діаграма діяльності (діаграми поведінки типу); діаграма прецедентів (діаграми поведінки типу); Діаграма класів.

Діаграма діяльності. Це візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій. Дія є фундаментальною

одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних сигналів.

Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності. Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності.

Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку.

Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Підсумок даної діаграми полягає в наступному: проектована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання (use case) використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором.

При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

– асоціації (association relationship);

– включення (include relationship);

– розширення (extend relationship);

– узагальнення (generalization relationship).

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме – за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення (include) у мові UML – це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності (dependency) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення розширення (extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

Діаграма класів це статичне представлення структури моделі. Відображає статичні (декларативні) елементи, такі як: класи, типи даних, їх зміст та відношення.

Діаграма класів, також, може містити позначення для пакетів та може містити позначення для вкладених пакетів. Також, діаграма класів може містити позначення деяких елементів поведінки, однак їх динаміка розкривається в інших типах діаграм.

Діаграма класів (class diagram) служить для представлення статичної структури моделі системи в термінології класів об'єктно-орієнтованого

програмування. На цій діаграмі показують класи, інтерфейси, об'єкти й кооперації, а також їхні відносини.

В UML існують наступні типи зв'язків які використовуються у діаграмі класів: Асоціації; Агрегація; Композиція.

Асоціації це якщо між двома класами визначена асоціація, то можна переміщатися від об'єктів одного класу до об'єктів іншого. Цілком припустимі випадки, коли обидва кінці асоціації відносяться до одного і того ж класу. Це означає, що з об'єктом деякого класу дозволено зв'язати інші об'єкти з того ж класу. Асоціація, що зв'язує два класи, називається бінарної. Можна, хоча це рідко буває необхідним, створювати асоціації, що зв'язують відразу кілька класів. Графічно асоціація зображується у вигляді лінії, що з'єднує клас сам з собою або з іншими класами.

Асоціації може бути присвоєно ім'я, яке описує природу відносини. Зазвичай ім'я асоціації не вказується, якщо тільки ви не хочете явно задати для неї рольові імена або у вашій моделі настільки багато асоціацій, що виникає необхідність посилатися на них і відрізняти один від одного. Ім'я буде особливо корисним, якщо між одними і тими ж класами існує кілька різних асоціацій.

Клас, що бере участь в асоціації, грає в ній деяку роль. По суті, це "обличчя", яким клас, що знаходиться на одній стороні асоціації, звернений до класу з іншого її боку. Можна явно позначити роль, яку клас грає в асоціації.

Часто при моделюванні буває важливо вказати, скільки об'єктів може бути пов'язано допомогою одного примірника асоціації. Це число називається кратністю (Multiplicity) ролі асоціації та записується або як вираз, значенням якого є діапазон значень, або в явному вигляді.

Вказуючи кратність на одному кінці асоціації, ви тим самим говорите, що на цьому кінці саме стільки об'єктів повинно відповідати кожному об'єкту на протилежному кінці. Кратність можна задати рівною одиниці (1), можна вказати діапазон: "нуль або одиниця" (0..1), "багато" (0.. *), "одиниця або більше" (1.. *). Дозволяється також вказувати певне число (наприклад, 3). За допомогою списку

можна задати і більш складні кратності, наприклад 0.. 1, 3..4, 6.. *, що означає "будь-яке число об'єктів, крім 2 і 5".

Агрегація це проста асоціація між двома класами відображає структурний відношення між рівноправними сутностями, коли обидва класу знаходяться на одному концептуальному рівні і ні один не є більш важливим, ніж інший. Але іноді доводиться моделювати відношення типу «частина/ціле», в якому один з класів має більш високий ранг (ціле) і складається з декількох менших за рангом (частин).

Ставлення такого типу називають агрегацією; воно зараховане до відносин типу «має» (з урахуванням того, що об'єкт-ціле має кілька об'єктів-частин). Агрегація є окремим випадком асоціації і зображується у вигляді простої асоціації з незафарбованим ромбом з боку «цілого». Графічно агрегація представляється порожнім ромбом на блоці класу, і лінією, яка від цього ромба до міститься класу.

Композиція це більш суворий варіант агрегації. Відома також як агрегація за значенням.

Композиція має жорстку залежність часу існування екземплярів класу контейнера та примірників містяться класів. Якщо контейнер буде знищений, то весь його вміст буде також знищено. Графічно представляється як і агрегація, але з зафарбовані ромбиком.

Опис системи та структури

Система протидії мережевим DDoS атакам у межах магістерської випускної кваліфікаційної роботи реалізується у вигляді модульного програмного комплексу на мові Python. Програма моделює реальний процес обробки мережевого трафіку, виявлення аномальної активності і застосування заходів реагування до зловмисних джерел.

Така реалізація підходить для експериментів у лабораторних умовах і для подальшої інтеграції з реальними засобами захоплення трафіку на рівні операційної системи або мережевого обладнання.

Архітектура системи має кілька логічних підсистем. Конфігураційний модуль зберігає основні параметри роботи, такі як розмір часової ковзної вибірки, порогові значення швидкості трафіку для окремих джерел, час блокування підозрілих адрес, максимальну глобальну інтенсивність пакетів та шляхи до файлів журналу і метрик.

Модуль моделювання трафіку генерує нормальний і атакувальний трафік, що дає можливість відтворити сценарії DDoS атак без доступу до реальної мережі. Підсистема статистичного аналізу накопичує пакети у часовому вікні, рахує кількість пакетів від кожної IP адреси та визначає поточну інтенсивність трафіку.

Підсистема виявлення аномалій аналізує статистику, порівнює її з порогами та формує рішення про наявність DDoS атаки. Модуль реагування управляє списком заблокованих адрес та реалізує дії з блокування і розблокування джерел. Підсистема збереження метрик накопичує інформацію про події та зберігає її у форматі JSON для подальшого аналізу. Основний клас системи координує роботу усіх модулів та запускає демонстраційні сценарії.

Конфігураційний клас SystemConfig оформлюється як dataclass. У ньому зберігаються параметри window_size_seconds, rate_threshold_per_ip, block_duration_seconds, max_global_packets_per_second, а також імена файлів для журналу log_file і метрик metrics_file. Окремо задається список whitelist_ips, у якому знаходяться IP адреси, що не підлягають блокуванню. Така структура конфігурації дає змогу легко змінювати налаштування під час експериментів і пояснювати вплив кожного параметра на чутливість системи до аномалій.

Модель мережевого пакета реалізується у вигляді dataclass Packet. Вона містить час надходження timestamp, адреси джерела і призначення src_ip і dst_ip, розмір пакета size_bytes, тип протоколу protocol і додаткове поле flags для позначення спеціальних режимів, наприклад SYN трафіку.

Окремі dataclass структури DetectionResult і BlockRecord описують результат аналізу пакета та інформацію про блокування IP адреси. DetectionResult

містить ознаку is_ddos, текстове пояснення reason і за потреби адресу offending_ip. BlockRecord містить IP, час блокування і причину, що дозволяє зберігати історію рішень.

Журналювання подій реалізується функцією setup_logging. Вона налаштовує модуль logging на вивід повідомлень у файл і одночасний показ у консолі. Формат повідомлень включає час, рівень важливості і текст події. Така конфігурація дозволяє використовувати журнали як джерело даних для пояснювальної записки і для аналізу роботи алгоритмів. У коді система фіксує початок і кінець демонстрації, блокування і розблокування IP адрес, а також коротку статистику про швидкість трафіку і найактивніші джерела.

Генерація трафіку реалізується класом TrafficGenerator. Конструктор приймає список нормальних адрес normal_ips і адресу сервера target_ip. Метод generate_normal_packet створює випадковий пакет із однієї з нормальних адрес до сервера, випадковим розміром і протоколом. Метод generate_ddos_packet приймає список botnet_ips і створює пакет із ботнет адреси до цілі з позначкою SYN-FLOOD у полі flags. Завдяки цьому в системі присутні окремі сценарії нормального та атакувального трафіку, які легко комбінуються у демонстраційних режимах.

Модуль StatisticsWindow відповідає за ковзне вікно статистики. Він зберігає двосторонню чергу пакета, лічильники пакетів за IP адресами і загальну кількість пакетів у вікні. Метод add_packet додає новий пакет, оновлює лічильники і викликає допоміжний метод _expire_old_packets. Останній видаляє з черги пакети, час яких виходить за межі заданого інтервалу window_size_seconds.

Функція get_rate_for_ip обчислює інтенсивність трафіку для конкретної адреси як відношення кількості пакетів до тривалості вікна. Аналогічно get_global_rate повертає загальну інтенсивність трафіку, а get_top_talkers повертає список найактивніших джерел.

Такий модуль дозволяє будувати просту, але наочну модель для пояснення принципів статистичного виявлення DDoS.

Підсистема виявлення DDoS атаки реалізується класом DDoSDetector. Він використовує конфігурацію і статистичне вікно. Метод analyze_packet отримує черговий пакет і виконує послідовність перевірок. Спочатку відкидаються адреси зі списку whitelist_ips. Потім оцінюється глобальна інтенсивність трафіку через get_global_rate.

Якщо значення перевищує max_global_packets_per_second, формується DetectionResult з позначкою атаки і адресою найактивнішого джерела. Далі обчислюється швидкість для конкретної адреси через get_rate_for_ip. Якщо вона перевищує rate_threshold_per_ip, пакет позначається як частина атаки. Окремо аналізується поле flags. Якщо в ньому присутній індикатор SYN-FLOOD, система сприймає трафік як потенційну атаку. Якщо жодна умова не виконується, формується результат з поясненням, що аномалія не виявлена. Така послідовність правил дає зрозумілий приклад евристичного виявлення DDoS.

Модуль реагування і блокування реалізується класом MitigationEngine. Він зберігає словник blocked_ips, у якому ключем є IP адреса, а значенням запис BlockRecord. Метод is_blocked перевіряє, чи знаходиться адреса під блокуванням, і перед перевіркою викликає _cleanup_expired_blocks. Останній проходить по словнику блокувань і видаляє записи, для яких перевищено час block_duration_seconds. Метод block_ip створює новий запис BlockRecord і додає його до словника, а також записує попереджувальне повідомлення у журнал. Така реалізація дозволяє у пояснювальній записці показати приклад простої системи блокування на основі часових обмежень без прив’язки до конкретних міжмережевих екранів.

Модуль MetricsStorage відповідає за накопичення і збереження інформації про події. У внутрішній структурі data зберігаються списки events і blocked_ips. Метод add_event додає словник з атрибутами пакета, результатом виявлення і поясненням причини у список подій. Метод update_blocked_ips синхронізує список заблокованих адрес з актуальним станом MitigationEngine.

Метод save записує структуру у JSON файл з використанням кодування

UTF 8. У пояснювальній записці такий модуль демонструє, як система підтримує можливість подальшого аналізу, побудови графіків і статистичних звітів для оцінювання ефективності алгоритмів.

Основний клас DDoSProtectionSystem зв'язує усі модулі в єдину систему. У конструкторі створюються екземпляри StatisticsWindow, DDoSDetector, MitigationEngine, MetricsStorage і TrafficGenerator.

Також визначається список botnet_ips для моделювання атак. Метод process_packet є центральною точкою обробки трафіку. Спочатку система перевіряє, чи не заблокований джерельний IP. Якщо адреса знаходиться у списку блокування, пакет відкидається, а у журнал записується відповідне повідомлення.

Якщо блокування відсутнє, пакет додається у статистичне вікно, аналізується модулем DDoSDetector і записується у метрики. У разі виявлення атаки і наявності offending_ip викликається блокування адреси через MitigationEngine.

Після цього оновлюється список заблокованих адрес у MetricsStorage. Додатково метод періодично виводить у журнал поточну глобальну швидкість і список найактивніших джерел.

```
# Програмна реалізація системи протидії мережевим DDoS атакам

import time
import random
import logging
import json
from dataclasses import dataclass, field
from typing import Dict, List, Optional
from collections import deque

# Клас конфігурації системи
@dataclass
class SystemConfig:
    window_size_seconds: int = 10
    rate_threshold_per_ip: int = 100
    block_duration_seconds: int = 300
    max_global_packets_per_second: int = 5000
    metrics_file: str = "ddos_metrics.json"
```

```
    log_file: str = "ddos_system.log"
    whitelist_ips: List[str] = field(default_factory=list)

# Структура мережевого пакета
@dataclass
class Packet:
    timestamp: float
    src_ip: str
    dst_ip: str
    size_bytes: int
    protocol: str
    flags: str = ""

# Результат виявлення аномалії
@dataclass
class DetectionResult:
    is_ddos: bool
    reason: str
    offending_ip: Optional[str] = None

# Запис про блокування
@dataclass
class BlockRecord:
    ip: str
    blocked_at: float
    reason: str


# Ініціалізація журналювання
def setup_logging(config: SystemConfig) -> None:
    logging.basicConfig(
        filename=config.log_file,
        level=logging.INFO,
        format="%(asctime)s [%(levelname)s] %(message)s",
    )
    logging.getLogger().addHandler(logging.StreamHandler())


# Модуль генерації імітованого трафіку
class TrafficGenerator:
    def __init__(self, normal_ips: List[str], target_ip: str) -> None:
        self.normal_ips = normal_ips
```

```
        self.target_ip = target_ip

    def generate_normal_packet(self) -> Packet:
        src_ip = random.choice(self.normal_ips)
        dst_ip = self.target_ip
        size_bytes = random.randint(64, 1500)
        protocol = random.choice(["TCP", "UDP", "HTTP"])
        timestamp = time.time()
        return Packet(
            timestamp=timestamp,
            src_ip=src_ip,
            dst_ip=dst_ip,
            size_bytes=size_bytes,
            protocol=protocol,
        )

    def generate_ddos_packet(self, botnet_ips: List[str]) -> Packet:
        src_ip = random.choice(botnet_ips)
        dst_ip = self.target_ip
        size_bytes = random.randint(64, 512)
        protocol = random.choice(["TCP", "UDP"])
        timestamp = time.time()
        flags = "SYN-FLOOD"
        return Packet(
            timestamp=timestamp,
            src_ip=src_ip,
            dst_ip=dst_ip,
            size_bytes=size_bytes,
            protocol=protocol,
            flags=flags,
        )


# Вікно статистики за часом
class StatisticsWindow:
    def __init__(self, config: SystemConfig) -> None:
        self.config = config
        self.packets = deque()
        self.count_by_ip: Dict[str, int] = {}
        self.total_packets: int = 0

    def add_packet(self, packet: Packet) -> None:
```

```
                self.packets.append(packet)
                self.total_packets += 1
                if packet.src_ip not in self.count_by_ip:
                    self.count_by_ip[packet.src_ip] = 0
                self.count_by_ip[packet.src_ip] += 1
                self._expire_old_packets(packet.timestamp)

            def _expire_old_packets(self, current_time: float) -> None:
                while self.packets and current_time - self.packets[0].timestamp >
self.config.window_size_seconds:
                    old_packet = self.packets.popleft()
                    self.total_packets -= 1
                    count = self.count_by_ip.get(old_packet.src_ip, 0)
                    if count <= 1:
                        self.count_by_ip.pop(old_packet.src_ip, None)
                    else:
                        self.count_by_ip[old_packet.src_ip] = count - 1

            def get_rate_for_ip(self, ip: str) -> float:
                count = self.count_by_ip.get(ip, 0)
                if not self.packets:
                    return 0.0
                window_duration = max(
                    0.001,
                    self.packets[-1].timestamp - self.packets[0].timestamp,
                )
                return count / window_duration

            def get_global_rate(self) -> float:
                if not self.packets:
                    return 0.0
                window_duration = max(
                    0.001,
                    self.packets[-1].timestamp - self.packets[0].timestamp,
                )
                return self.total_packets / window_duration

            def get_top_talkers(self, limit: int = 5) -> List[str]:
                sorted_ips = sorted(
                    self.count_by_ip.items(),
                    key=lambda item: item[1],
                    reverse=True,
```

```
            )
            return [ip for ip, _ in sorted_ips[:limit]]


    # Модуль виявлення DDoS
    class DDoSDetector:
        def __init__(self, config: SystemConfig, window: StatisticsWindow) ->
None:
            self.config = config
            self.window = window

        def analyze_packet(self, packet: Packet) -> DetectionResult:
            if packet.src_ip in self.config.whitelist_ips:
                return DetectionResult(
                    is_ddos=False,
                    reason="IP у білому списку",
                )

            global_rate = self.window.get_global_rate()
            if global_rate > self.config.max_global_packets_per_second:
                top_ip = self._find_most_active_ip()
                return DetectionResult(
                    is_ddos=True,
                    reason="Перевищення глобального порогу пакетів за секунду",
                    offending_ip=top_ip,
                )

            ip_rate = self.window.get_rate_for_ip(packet.src_ip)
            if ip_rate > self.config.rate_threshold_per_ip:
                return DetectionResult(
                    is_ddos=True,
                    reason="Перевищення порогу трафіку для джерела",
                    offending_ip=packet.src_ip,
                )
            if "SYN-FLOOD" in packet.flags:
                return DetectionResult(
                    is_ddos=True,
                    reason="Виявлений підозрілий SYN трафік",
                    offending_ip=packet.src_ip,
                )
            return DetectionResult(
                is_ddos=False,
```

```
                reason="Аномалія не виявлена",
            )
        def _find_most_active_ip(self) -> Optional[str]:
            if not self.window.count_by_ip:
                return None
            return max(self.window.count_by_ip,
key=self.window.count_by_ip.get)
```

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм ДСТУ 28147:2009, що є класичним алгоритмом симетричного шифрування на основі мережі Фейстеля (рисунок 4.3). Даний алгоритм шифрує інформацію блоками по 64 біта (такі алгоритми називаються "блоковими"). Зміст мережі Фейстеля полягає в тому, що блок шифруемої інформації розбивається на два або більше субблоків, частина яких обробляється за певним законом, після чого результат цієї обробки накладається (операцією побітового додавання за модулем 2) на необроблювані субблоки. Потім субблоки міняються місцями, після чого обробляються знову й т.д. певне для кожного алгоритму число раз – раундів.

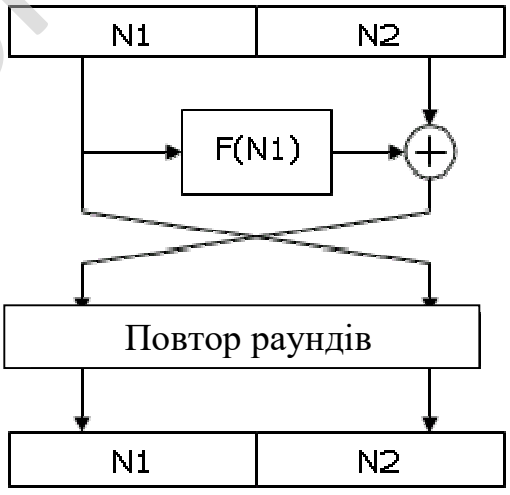

Рисунок 4.3 – Мережа Фейстеля

Основна відмінність алгоритмів симетричного шифрування друг від друга складається саме в різних функціях обробки субблоків. Дана функція часто називається "основним криптографічним перетворенням", оскільки саме вона несе основне навантаження при шифруванні інформації. Основне перетворення алгоритму ДСТУ 28147:2009 є досить простим, що забезпечує високу швидкодію алгоритму; у ньому виконуються наступні операції (рисунок 4.4).

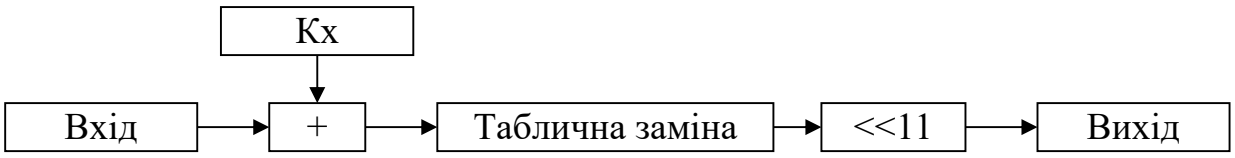

Рисунок 4.4 – Основне перетворення алгоритму ДСТ 28147:2009

1. Додавання субблоку з певним фрагменом ключа шифрування за модулем 2^{32}. Кх – це 32-бітна частина ("підключ") 256-бітного ключа шифрування, якому можна представити як конкатенацію 8 підключів: К = $К_0К_1К_2К_3К_4К_5К_6$ $К_7$. Залежно від номера раунду й режиму роботи алгоритму (про їх – нижче), для даної операції вибирається один з підключів.

2. Таблична заміна. Для її виконання субблок розбивається на 8 4-бітних фрагментів, кожний з яких прогоняється через свою таблицю заміни. Таблиця заміни містить у певній послідовності значення від 0 до 15 (тобто всі варіанти значень 4-бітні фрагменти даних); на вхід таблиці подається блок даних, числове подання якого визначає номер вихідного значення. Наприклад, подається значення 5 на вхід наступної таблиці: "13 0 11 74 91 10 143 5 122 15 8 6". У результаті на виході виходить значення 9 (оскільки 0 заміняється на 13, 1 – на 0, 2 – на 11 і т.д.).

3. Побітове циклічне зрушення даних усередині субблока на 11 біт уліво.

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено інтерфейс програмного забезпечення, розробленого у результаті виконання магістерської роботи.

Розроблене програмне забезпечення систем протидії DDoS-атакам складається з наступних функціональних блоків:

– Навігаційне меню: Файл; Вид; Інструменти; Параметри; Довідка.

– Підрозділу представлення інформаційних даних.

– Вікна обрання типу.

– Вікно виведення результату роботи системи.

– Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.

– Графічних функціональних кнопок ПЗ.

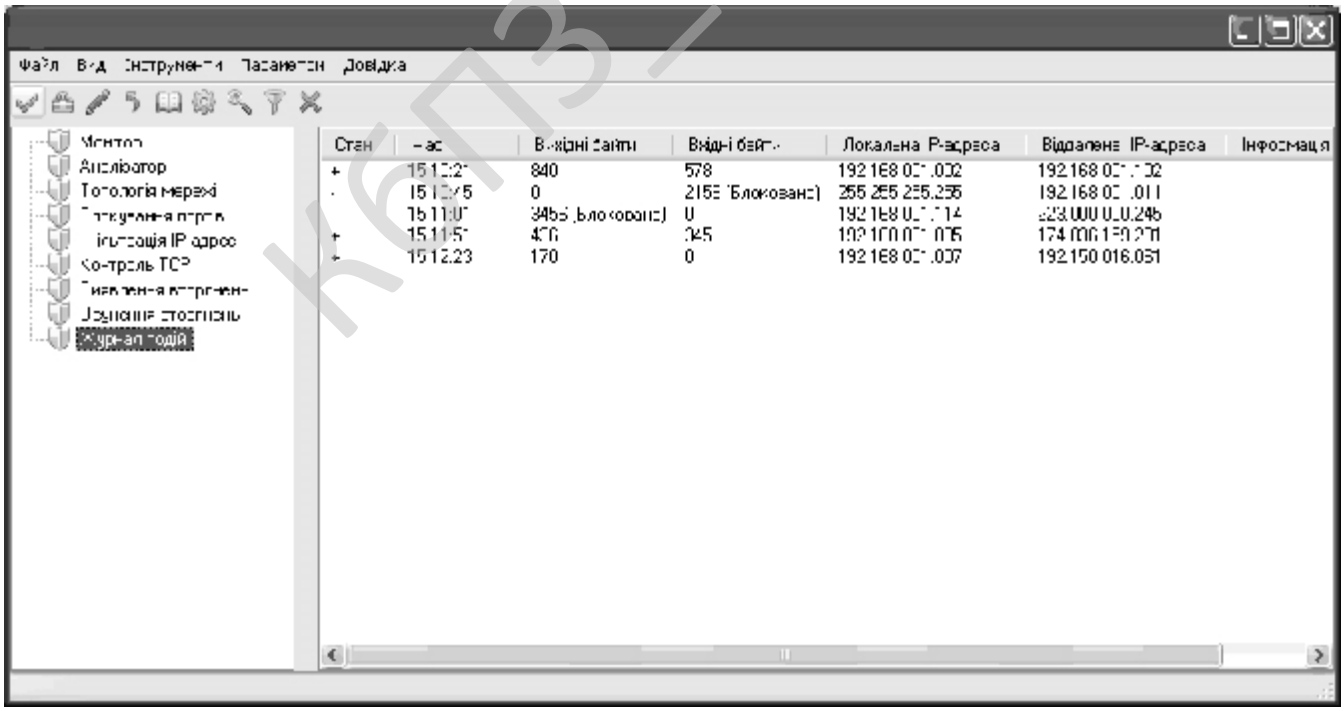

Рисунок 5.1 – Головне вікно розробленого ПЗ

Для перегляду короткої довідки про програму слід натиснути на основному вікні кнопку авторського права, після чого на екрані з’явиться вікно показане на рисунку 5.2.

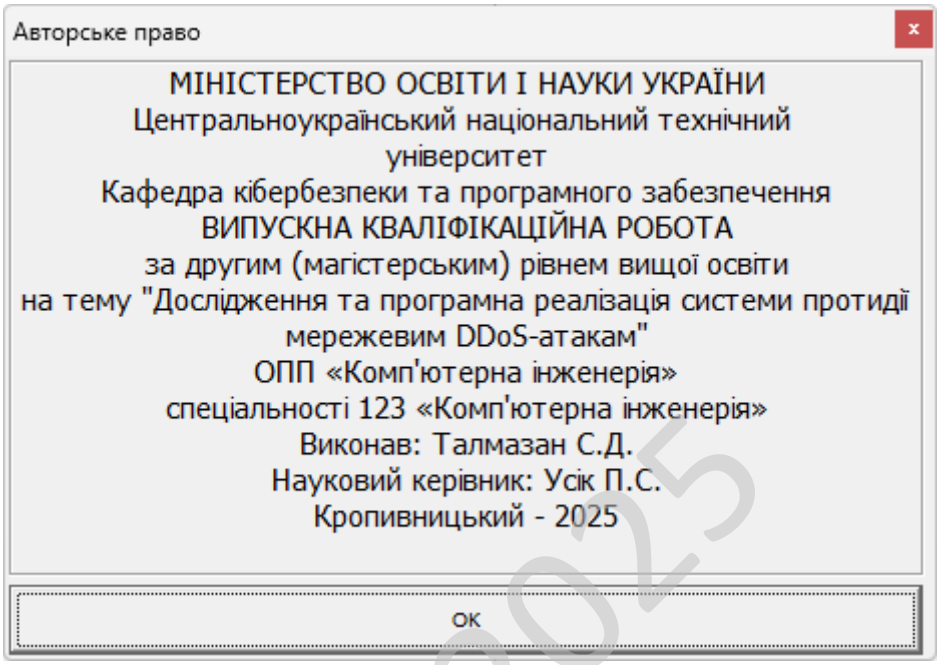

Рисунок 5.2 – Вікно розробника ПЗ

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

– відповідності поставленим вимогам;

– правильна відповідь для усіх можливих вхідних даних;

– виконання функцій за прийнятний час;

– практичність;

– сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом чорної скриньки

Основне місце програми тестів «чорної скриньки» – інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

– Як виконуються функції програми.

– Як приймаються вихідні дані.

– Як виробляються результати.

– Як зберігається цілісність зовнішньої інформації.

При тестуванні «чорної скриньки» розглядаються системні характеристики програм, ігнорується їхня внутрішня логічна структура. Вичерпне тестування, як правило, неможливе.

Наприклад, якщо в програмі 10 вхідних величин і кожна приймає по 10 значень, то кількість тестових варіантів становитиме 10^10. Тестування «чорної скриньки» не реагує на багато особливостей програмних помилок.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, які забезпечують повну перевірку всіх функціональних вимог до програми.

Програмний виріб тут розглядається як «чорна скринька», чию поведінку можна визначити тільки дослідженням його входів та відповідних виходів. При такому підході бажано мати:

– Набір, утворений такими вхідними даними, які призводять до аномалій у поведінці програми (назвемо його ITc).

– Набір, утворений такими вхідними даними, які демонструють дефекти програми (назвемо його ОТ).

Будь-який спосіб тестування «чорної скриньки» повинен:

– Виявити такі вхідні дані, які з високою ймовірністю належать набору ITс;

– Сформулювати такі очікувані результати, які з високою імовірністю є елементами набору ОТ.

Принцип «чорної скриньки» не альтернативний принципу «білої скриньки». Скоріше це доповнює підхід, який виявляє інший клас помилок.

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

– Некоректних чи відсутніх функцій.

– Помилок інтерфейсу.

– Помилок у зовнішніх структурах даних або в доступі до зовнішньої бази даних.

– Помилок характеристик (необхідна ємність пам'яті і т.д.).

– Помилок ініціалізації та завершення.

Обрано умови розповсюдження – Freeware.

Це власницьке програмне забезпечення, котре можна Безоплатно використовувати протягом необмеженого терміну без обмежень у функціональності, і поширюване без сирцевих кодів.

Автори такого програмного забезпечення, як правило, хочуть «дати щось спільноті», але хочуть також контролювати його подальшу розробку. Іноді, коли програмісти вирішують припинити розробку, вони передають сирцевий код іншим програмістам, або ж спільності як вільне програмне забезпечення.

Дуже часто плутають поняття «безплатне програмне забезпечення» та «вільне програмне забезпечення», хоча вони суттєво відрізняються.

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи протидії мережевим DDoS-атакам.

Метою розробки є дослідження та програмна реалізація системи протидії мережевим DDoS-атакам.

Об’єктом дослідження є процес протидії мережевим DDoS-атакам.

Предметом дослідження є методи протидії мережевим DDoS-атакам.

Методи дослідження базуються на методах захисту інформації у комп’ютерних мережах, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод протидії мережевим DDoS-атакам.

– Розроблено вітчизняний продукт протидії мережевим DDoS-атакам, який має більш широкі можливості, на відміну від існуючих аналогів.

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати дослідження та впровадження системи протидії мережевим DDoS-атакам можуть бути насамперед цікавими підприємствам, чия діяльність безпосередньо залежить від безперервного функціонування вебресурсів. Це можуть бути банки, державні установи, торговельні платформи, компанії з надання онлайн-послуг і навіть навчальні заклади, які активно використовують цифрові сервіси. Для таких організацій навіть короткочасна зупинка сайтів або серверів означає втрату прибутку, клієнтів і репутації.

Крім того, результати розробки становлять інтерес для ІТ-компаній, які займаються аутсорсингом технічної підтримки або кіберзахисту. Такі підприємства можуть інтегрувати напрацьовані алгоритми та методики у власні продукти, надаючи замовникам нові сервіси захисту від атак. Це відкриває додаткові можливості для розширення ринку й підвищення конкурентоспроможності.

Наукові установи та освітні заклади також можуть зацікавитися розробкою, адже тема DDoS-атак і засобів протидії є надзвичайно актуальною для сучасної підготовки спеціалістів у галузі кібербезпеки. Використання програмної реалізації в навчальному процесі дозволить моделювати різні сценарії атак і вивчати поведінку мережі під навантаженням у контрольованих умовах.

Загалом, подібне дослідження має міждисциплінарний характер, адже воно корисне не лише для фахівців із безпеки, а й для управлінців, аналітиків і розробників, які прагнуть створювати стійкі та надійні інформаційні системи.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Щоб визначити привабливість проєкту з розробки системи протидії DDoS-атакам, можна застосувати метод експертних оцінок, який базується на колективній думці фахівців у сфері кібербезпеки. Для цього формується група з 8–10 експертів, серед яких мережеві адміністратори, системні аналітики та спеціалісти із захисту даних. Їм пропонують оцінити систему за критеріями: ефективність виявлення атаки, швидкість реакції, рівень автоматизації, вартість впровадження, масштабованість і простота інтеграції.

Кожен експерт виставляє оцінки за десятибальною шкалою. У результаті середній бал системи може становити, наприклад, 8,9. Найвищі оцінки отримує показник ефективності захисту (9,5), а найнижчі – складність початкового налаштування (7,8). Це свідчить про високий рівень технологічної зрілості проєкту та його значну ринкову привабливість.

Експерти також можуть надати свої рекомендації щодо вдосконалення. Наприклад, вони можуть порадити інтегрувати механізм поведінкового аналізу або штучного інтелекту для підвищення точності виявлення атак. Такі пропозиції дозволяють не лише оцінити, а й покращити проєкт ще на етапі розробки.

Таким чином, метод експертних оцінок допомагає комплексно визначити конкурентоспроможність системи на ринку й сформувати обґрунтовану стратегію її розвитку, спираючись на думку професіоналів із практичним досвідом.

7.3 Вибір методу оцінки вартості ПЗ

Для оцінки вартості розробки системи протидії DDoS-атакам доцільно використати витратно-функціональний метод, який враховує витрати на обладнання, програмне забезпечення, заробітну плату розробників, а також супутні операційні витрати на підтримку системи. Цей метод дозволяє

об’єктивно визначити реальні фінансові потреби проєкту, враховуючи як первинні інвестиції, так і подальші витрати на експлуатацію.

Додатково варто застосувати метод оцінки «вартість–ефективність», який показує, наскільки кожна гривня, вкладена в систему, приносить користь у вигляді зменшення збитків від атак. Для цього порівнюють вартість простоїв до і після впровадження системи. Якщо щорічні втрати досягали кількох мільйонів гривень, а після встановлення рішення знизилися в десятки разів, це є наочним підтвердженням економічної доцільності.

Також може бути використаний метод дисконтування грошових потоків, щоб оцінити майбутні вигоди проєкту у теперішніх цінах. Це особливо важливо для великих організацій, які розглядають впровадження таких систем як довгострокову інвестицію у стабільність бізнесу.

Поєднання цих підходів дозволяє побачити повну картину вартості та вигод від реалізації проєкту, що допомагає ухвалити обґрунтоване управлінське рішення про його впровадження.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Підприємство середнього масштабу працює у сфері електронної комерції та надає онлайн-доступ до своїх послуг 24/7. За останній рік компанія кілька разів зазнала DDoS-атак, через що сайт і платіжна система були недоступні в середньому по 8–10 годин. Кожна атака призводила до прямих фінансових втрат через зупинку продажів, а також до репутаційних збитків і відтоку клієнтів.

Для зменшення ризиків прийнято рішення впровадити автоматизовану систему виявлення та фільтрації DDoS-трафіку, яка інтегрується на рівні мережевого шлюзу. Система здатна ідентифікувати підозрілу активність, блокувати аномальний трафік у реальному часі та забезпечувати стабільну роботу веб-ресурсів навіть під час атаки. Вхідні дані зафіксовано в таблиці 7.1.

Таблиця 7.1 – Вихідні дані для розрахунку

Показник	**До впровадження**	**Після впровадження**	**Економічний ефект**
Кількість DDoS-інцидентів на рік	5	1	–4
Середня тривалість одного інциденту	10 год	1 год	–9 год
Вартість простою 1 години (втрата прибутку + відшкодування клієнтам)	50 000 грн	10 000 грн	–40 000 грн
Середній обсяг збитків за рік	5 × 10 × 50 000 = **2 500 000 грн**	1 × 1 × 10 000 = **10 000 грн**	**–2 490 000 грн**
Початкові інвестиції у систему (обладнання, ліцензії, впровадження)	—	900 000 грн	—
Щорічні витрати на підтримку системи	—	120 000 грн	—

Розрахунок економічного ефекту демонструє наступне: економія на усуненні наслідків атак – 2 490 000 грн/рік, економія на технічних роботах і

людських ресурсах (скорочення витрат на відновлення систем після атак, близько 200 000 грн/рік), сукупний річний економічний ефект – 2 690 000 грн, чистий економічний ефект – 2 570 000 грн/рік, термін окупності – 0,35 року (~4 місяці), рентабельність інвестицій – 285 %.

Додаткові нефінансові переваги: підвищення безперервності бізнесу – вебресурси залишаються доступними навіть під час атак, збереження репутації – клієнти не стикаються з перебоями в роботі сервісів, автоматизація безпеки – знижується навантаження на ІТ-персонал, масштабованість – систему можна розширювати з ростом компанії чи обсягу трафіку, відповідність стандартам безпеки – забезпечується відповідність вимогам ISO/IEC 27001, GDPR.

Таким чином, система протидії DDoS-атакам не лише підвищує рівень кіберзахисту, а й стає економічно обґрунтованим елементом стратегії інформаційної безпеки, який безпосередньо впливає на прибутковість і стійкість компанії у цифровому середовищі.

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Просування проєкту системи протидії DDoS-атакам має ґрунтуватися на поєднанні технічної демонстрації та побудови довіри до бренду. На першому етапі важливо створити докладну демонстраційну версію або кейс, який покаже реальні результати роботи системи під час імітації атаки. Потенційні клієнти повинні побачити наочно, як система блокує шкідливий трафік і забезпечує стабільну роботу мережі.

Далі необхідно налагодити інформаційну кампанію. Це може бути серія статей, відео чи вебінарів, присвячених актуальним кіберзагрозам. Акцент варто робити не на технічних деталях, а на бізнес-перевагах – зниженні фінансових ризиків, збереженні довіри клієнтів, підвищенні рівня безпеки сервісів.

Важливо залучати партнерів – провайдерів хостингу, дата-центрів і телеком-компаній, які можуть пропонувати систему своїм клієнтам як додаткову

послугу. Така стратегія дозволить охопити більшу аудиторію без значних маркетингових витрат.

На завершення можна проводити спільні пілотні проєкти з великими компаніями, результати яких стануть найкращою рекламою продукту. Практичні кейси завжди мають більшу довіру, ніж рекламні гасла.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для ефективного збуту подібної системи слід поєднати кілька напрямів реалізації. По-перше, варто запровадити модель «рішення як послуга» (DDoS Protection as a Service), яка дозволяє клієнтам користуватися захистом без купівлі обладнання – за передплатою. Це спрощує входження на ринок і робить продукт доступним для малого та середнього бізнесу.

По-друге, доцільно створити партнерську мережу серед хостинг-компаній і провайдерів. Вони можуть інтегрувати систему у свої сервіси, пропонуючи її як додатковий рівень захисту клієнтам. Це не лише розширить географію збуту, а й підвищить впізнаваність бренду.

Також варто забезпечити наявність онлайн-демо, де користувачі зможуть протестувати систему на прикладі симульованої атаки. Такий формат інтерактивного залучення допоможе переконати потенційних замовників у реальній користі рішення.

У підсумку, оптимізація збуту полягає в створенні гнучкої моделі продажів, де клієнт отримує можливість вибору – від короткострокової підписки до повного корпоративного впровадження з технічною підтримкою.

7.7 Визначення ключових факторів успіху конкретного проєкту

Успіх такого проєкту визначається передусім його надійністю та реальними результатами. Якщо система здатна швидко реагувати на загрози,

ефективно фільтрувати трафік і запобігати збоям, користувачі самі стануть її головними промоутерами. У сфері кібербезпеки саме довіра й підтверджена ефективність є найкращою рекламою.

Не менш важливим є аспект підтримки клієнтів. Постійне оновлення баз даних атак, моніторинг у режимі 24/7 і швидке реагування на інциденти створюють відчуття захищеності, що значно підвищує лояльність користувачів.

Фактором успіху є також доступність системи для різних категорій клієнтів. Якщо рішення можна адаптувати як для великої корпорації, так і для невеликого бізнесу, це відкриває ширші можливості на ринку.

І, нарешті, ключову роль відіграє інноваційність. Використання штучного інтелекту, машинного навчання та поведінкової аналітики для прогнозування атак забезпечує конкурентну перевагу та дозволяє компанії залишатися на крок попереду кіберзлочинців.

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

В охорону праці включають санітарно-гігієнічні, лікувально-профілактичні та організаційно-технічні системи правових і соціально-економічних заходів.

В кожній ІТ компанії є трудові відносини з працівниками. Згідно закону України “Про охорону праці” [3] кожна компанія впроваджує заходи з охорони праці. Реалізується трудові відносини з вживанням необхідних засобів з охорони праці та розробки відповідних документів:

– Інструкцій з охорони праці по кожній професії і загальні.

– Положення про охорону праці.

– Накази з охорони праці.

– Журнали реєстрації та інструктажу.

Роботодавець створює відділ який працює відповідно до типового положення, яку затверджується центральним органом виконавчої влади і забезпечує виконання вимог державної політики у сфері охорони праці.

За недотриманням вимог, керівники ІТ компаній можуть бути притягнуті до відповідальності, яка виглядає у виді накладання штрафу. Якщо в результаті порушення умов охорони праці є постраждалі працівники то керівні особи ІТ компаній притягуються до кримінальної відповідальності.

Законом України “Про охорону праці” [3] регламентуються загальні положення державної політики в галузі охорони праці, а конкретизуються ці положення нормативно-правовими актами про охорону праці, зокрема Наказом Міністерства соціальної політики України 14.02.2018 № 207, який зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за №508/31960 «Про затвердження Вимог щодо безпеки та захисту здоров’я працівників під час роботи

з екранними пристроями» [5], яким затверджено нормативно-правовий акт з охорони праці НПАОП 0.00-7.15-18, «Правила охорони праці під час експлуатації електронно-обчислювальних машин», та «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98 [2].

Програмісти у процесі роботи мають негативний вплив на органи зору, а також мають значну розумову напругою і нервово-емоційне навантаження. Руки (суглоби пальців та м'язи рук) при роботі з клавіатурою мають теж істотне навантаженням. До шкідливих факторів, які впливають на робітників галузі інформаційних технологій (ІТ) спеціалісти відносять високочастотні електромагнітні коливання (випромінювання) роботи апаратної частини ЕОМ та виділення шкідливих газів.

Ці шкідливі фактори можуть привести до професійних захворювань.

Розглянемо шкідливі чинники роботи програмістів керуючись наступними нормативно-правовими актами: «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98 [2], та «Вимоги щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями» НПАОП 0.00-7.15-18.

Умови праці програміста включають наступні фактори:

– параметри повітряного середовища в приміщенні;

– вентиляція приміщення;

– освітлення приміщення;

– параметри повітряного середовища в приміщенні, тощо.

Щоб запропонувати заходи щодо зменшення негативного впливу комп’ютера на організм людини визначемо фактори, які можуть викликати професійне захворювання і впливають на працездатність програміста.

8.2 Пожежна безпека

Вимог пожежної безпеки на підприємстві неухильно повинен дотримуватися кожен співробітник, а організаційна складова при цьому покладається на посадових осіб за відповідним рішенням керівництва і прописується в посадових інструкціях і положеннях по структурним підрозділам. Зокрема, вказуються конкретні території, ділянки, зони, об’єкти, цілі будівлі і їх частини, поверхи, на яких відповідального співробітника повинне проводити такі організаційні роботи.

Відповідальні особи зобов’язуються розробити, впровадити та підтримувати в певному інструкцією і положенням на ввірених їм об’єктах протипожежний режим і інструкції відповідно до вимог, викладених в нормативних актах.

Передбачено також створення підрозділу добровільної пожежної охорони та пожежно-рятувальної команди в його складі.

Встановлений режим включає порядки з описом місць спеціального призначення та правила їх користування та утримання, наприклад:

– евакуаційних шляхів;

– місць для паління;

– місць складування продукції та сировини;

– стоянки транспорту.

Також встановлюється порядок роботи та технічного обслуговування:

– вентиляційного устаткування;

– засобів пожежогасіння і захисту від загорянь;

– нагрівальних приладів;

– електрообладнання.

Розробляються і впроваджуються правила роботи з відкритим вогнем і горючими матеріалами. Створюються графіки проходження інструктажів з пожежної безпеки співробітників, а також порядок і терміни перевірок знань

пожежно-технічного мінімуму, в тому числі, тих працівників, які відповідальні за цю ділянку роботи на підприємстві. При цьому можуть передбачатися внутрішні лекції, семінари, тренінги та практичні заняття на підприємстві, а також зовнішні – на базі спеціалізованих навчальних центрів з професійними викладачами. Важливою складовою протипожежного режиму на будь-якому об'єкті є розробка і впровадження порядку дій при виникненні пожежі. Неодмінно має бути план евакуації, описано, як повинні відключатися електроустановки, що і в якій послідовності необхідно робити співробітникам.

Відповідно, для кожного об'єкта, кожного приміщення (крім коридорів, санвузлів, басейнів і подібних приміщень), окремих видів робіт складаються інструкції, за якими повинен працювати персонал, залучений на певних ділянках і в виконанні окремих видів робіт. За інструкціями проводиться навчання (інструктаж) персоналу з подальшим контролем знань. Детально про те, як розробити протипожежний режим, прописати порядки та інструкції, пояснюють на тематичних курсах і семінарах. [4]

8.3 Пропозиції щодо підвищення працездатності ІТ-фахівців

Поява та впровадження нових інформаційно-комунікаційних технологій зумовлює необхідність подальшого вдосконалення охорони праці фахівців ІТ-індустрії. Все це потребує розробки нових нормативно-правових актів з регламентації праці та відпочинку фахівців ІТ-індустрії і стандартів підприємств, центрів комп'ютерної техніки, центрів інформаційних технологій, сучасних комп'ютерних класів. Для підвищення розумової працездатності то зорової роботи повинна здійснюватися ергономічна оптимізація в рамках системи «оператор-термінал», яка сприятиме результативній фізичній та інтелектуальний працездатності і відновленню психосоматичного здоров'я фахівців ІТ-індустрії. Всі наведені заходи щодо вдосконалення охорони праці фахівців ІТ-індустрії повинні контролюватися службою охорони праці та комісією з охорони праці

підприємства. Особливе значення у соціальному захисті цієї категорії працівників належить прийняття комплексного договору, який може забезпечити фахівців додатковими пільгами та компенсаціями.

Пропозиції щодо підвищення працездатності ІТ-фахівців, розіб'ємо на декілька категорій:

– Середовище і розпорядок праці. Для мінімізації негативних ефектів, що пов'язані з перевтомленням ІТ-фахівців, потрібно чітко прописати і реалізувати графік періодів праці-відпочинку, щоб фахівець міг можливість переключити увагу, дати можливість відпочити очам, мозку, елементарно, встати розім'яти ноги. Також потрібно зробити максимально комфортними умови мікроклімату у офісному приміщенні, де працюють ІТ-фахівці. Мається на увазі встановлення і експлуатація, коли виникає необхідність, кондиціонерів, опалення, та системи вентиляції, задля попередження перегрівання, переохолодження ІТ-фахівців, і подальшої неможливості ними виконувати свої функції. Також, за можливості, нами пропонується введення практики віддаленої праці ІТ-фахівцями, якщо роботодавець не може забезпечити оптимальні і безпечні умови в офісному приміщенні, або якщо фахівця вони не влаштовують із певних причин.

– Фізичні і психоемоційні чинники. Першим і найважливішим чинником, що впливає на працездатність ІТ-фахівців є робоче місце, і саме тому, роботодавець має забезпечити максимальний його комфорт і безпеку. Гарантією цих факторів може слугувати сертифікація меблів, що використовуються на підприємстві ІТ-галузі. Тому нами пропонується закупівля тільки меблів , які пошли сертифікацію на відповідність. Під психоемоційними чинниками ми розуміємо гарне самопочуття фахівців, позитивний настій, гарний психологічний клімат у колективі, тощо. Задля того, щоб психоемоційні чинники мали максимально позитивний ефект, керівництву слід поводити заходи, які сприятимуть укріпленню і покращенню міжособистісних стосунків у колективі, таких як психологічні тренінги, тімбілдінг, спортивні змагання і естафети. Також, сюди можна віднести розробку і впровадження системи мотивації працівників, як фінансової, так моральної і адміністративної.

8.4 Розробка заходів з умов поліпшення охорони праці

Згідно аналізу умов праці в розглянутому приміщенні, ми одержали наступні результати:

– розмірі приміщення, у розрахунку на одному працюючого, відповідають нормативам;

– мікроклімат відповідає нормативному значенню;

– акустичні умови роботи не перевищують нормативних значень;

Таким чином можна припустити, що основною причиною можливого зниження працездатності програміста є психофізіологічний фактор, тому основна пропозиція буде така: дотримання позитивної психологічної атмосфери в колективі та регламентованого режиму праці та відпочинку, організація робочого місця з урахуванням ергономічних вимог.

Рекомендовані заходи: регулярні періодичні наочні огляди персоналом шляхів для евакуації людей із приміщення, відповідно до плану евакуації (який повинен розташовуватись на видному місці у приміщенні), включення до колективного договору мінімально можливого вмісту аптечок з обов'язковою наявністю масок-клапанів, або іншого спорядження для штучного дихання. Регулярна періодична перевірка параметрів заземлення та занулення (вимірювання опору ланцюга) [9].

Регулярна наочне знайомство персоналу із шляхами для евакуації людей із приміщення відповідно до плану евакуації, забезпечення розподільних щитів спеціальними розетками з заземлюючими контактами; організація заземлення всіх приладів і пристроїв, які працюють при напрузі вище 36 В.

Так як при ураженні електричним струмом у людини може статися фібриляція шлуночків серця, в організації бажано мати дефібрилятор і підготовлений персонал для роботи з ним.

8.5 Розрахункова частина

Для захисного штучного заземлення будемо застосовувати вертикальні електроди з сталевого прокату круглого перерізу діаметром 45 мм, довжиною L=2 м., та горизонтальний електрод – металева полоса з перетином 45·5 мм. Напруга – 220/380 В. Розрахункова схема розташування заземлюючих електродів – по контуру (прямокутником).

Розрахунок проводиться за допустимим опором розтіканню струму заземлювача.

Початкові дані для розрахунку захисного заземлення:

– тип верхнього шару ґрунту – чорнозем, нижнього шару грунту – глина (питомий опір ρ_2 = 40 Ом·м);

– умовна товщина верхнього шару ґрунту: H=0,55 м;

– відстань між вертикальними заземлювачами (електродами) A=3 м;

– глибина закладення горизонтального контуру заземлення t=0,75 м;

– опір заземлювача, який нормується: $R_{ЗН}$ = 4 Ом.

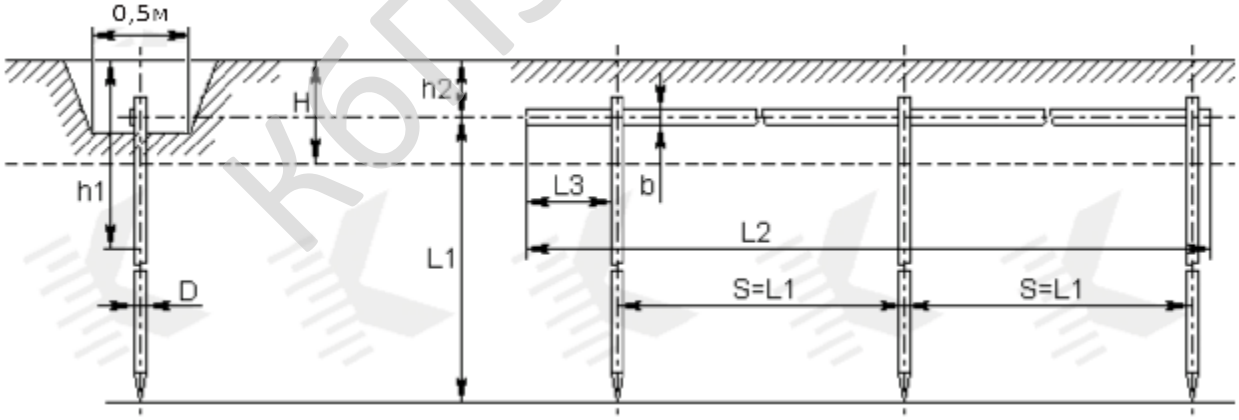

Рисунок 8.1 – Схема штучного заземлення

Необхідно визначити потрібну кількість вертикальних заземлювачів та довжину полоси (горизонтального заземлювача) (рис. 8.1).

Виконаємо розрахунок.

Відстань від центра вертикального заземлювача до поверхні землі:

$$T = t + L/2 = 0,75 + 2/2 = 1,75 \text{ м.}$$

Розрахунковий питомий опір ґрунту (з врахуванням того, що фактично вся конструкція заземлювача розташовується у нижньому шарі ґрунту):

$$\rho = \psi \rho = 1,36 \cdot 40 = 54,5 \text{ Ом·м.}$$

де

$\psi = 1,36$ – табличне значення коефіцієнту сезонності для відповідної кліматичної зони у багатошаровому ґрунті [11];

$\rho_2 = 40$ Ом·м. – табличне значення питомого опору нижнього шару ґрунту (глина) [11].

Діаметр вертикального електроду (заданий) $D_в = 45$ мм = 0,045 м.

Відношення A/L=3/2=1,5.

Опір розтіканню електричного струму одного електрода вертикального заземлювача з урахуванням заглиблення заземлювача [11]:

$$R_O = 0,366 \cdot (\rho/L) \cdot [\lg(2L/D_в) + (1/2) \cdot \lg((4T+L)/(4T-L))] =$$

$$= 0,366 \cdot (54,5/2) \cdot [(\lg(2 \cdot 2/0,045) + (1/2) \cdot \lg((4 \cdot 1,75+2)/(4 \cdot 1,75-2))] = 20,6 \text{ Ом.}$$

Визначаємо коефіцієнт екранування вертикальних електродів Кев=0,53 при орієнтовній кількості вертикальних електродів, яке дорівнює 5 [11].

Визначаємо необхідну кількість вертикальних електродів заземлювача (без врахування горизонтального заземлювача), при $R_{ЗН} = 4$ Ом:

$$N = R_O / (К_{ев} \cdot R_{ЗН}) = 20,6 / (0,53 \cdot 4) = 9,75 \approx 10 \text{ шт.}$$

Визначаємо довжину з'єднуючої полоси:

$$L_П = 1,05 \cdot A \cdot N = 1,05 \cdot 3 \cdot 10 = 30,7 \approx 31 \text{ м.}$$

Опір розтіканню електричного струму з'єднуючої полоси з урахуванням кліматичного коефіцієнта питомого опору ґрунту $K_П$ [11]:

$$R_П = 0,366 \cdot (\rho \cdot K_П / L_П) \cdot \lg(2(L_П \cdot L_П)/(B \cdot t)) =$$

$$= 0,366 \cdot (40 \cdot 5/40) \cdot \lg((2 \cdot 40^2)/(0,045 \cdot 0,75)) = 11,57 \text{ Ом.}$$

де $K_П = 5$ – табличне значення кліматичного коефіцієнта питомого опору ґрунту для відповідної кліматичної зони для з'єднуючої полоси [11]:

B = 45 мм = 0,045 м.- ширина з'єднуючої полоси (задана).

Загальний опір розтіканню електричного струму заземлювача [11]:

$$R = (R_O \cdot R_П)/(R_O \cdot \eta_П + N \cdot R_П \cdot Кев) =$$

$$= (20,6 \cdot 11,57)/(20,6 \cdot 0,55 + 10 \cdot 11,57 \cdot 0,53) = 3,3 \text{ Ом}.$$

де $\eta_П = 0,55$ – табличне значення коефіцієнта екранування з'єднуючої полоси [11].

Умова $R \leq R_{ЗН}$ виконується ($3,3 \leq 4$).

Оскільки R суттєво більше $R_{ЗН}$, зменшимо кількість вертикальних електродів до 8 і виконаємо перерахунок. У результаті остаточно отримали: кількість вертикальних електродів дорівнює 8 при R =3,9 Ом.

Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз приміщення, призначеного для праці програмістів, проведено розгляд питань пожежної безпеки, небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи.

Тільки повна усвідомленість працівника про можливі небезпеки, що можуть підстерігати його на робочому місці та дотримання вимог нормативних актів о питань охорони праці та відповідних рекомендацій фахівців, дозволять значною мірою знизити негативний вплив шкідливих та небезпечних факторів при роботі з комп'ютером на організм людини.

Виконано розрахунок захисного штучного заземлення, як одного з ключових факторів безпеки програміста.

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи протидії мережевим DDoS-атакам.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів протидії мережевим DDoS-атакам.

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем протидії мережевим DDoS-атакам.

– Досліджена система протидії мережевим DDoS-атакам.

– На основі отриманих результатів досліджень створена програмна реалізація системи протидії мережевим DDoS-атакам.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання протидії мережевим DDoS-атакам.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створені програмного забезпечення було використано об’єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм ДСТУ 28147:2009.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Проведено маркетингове та економічне обґрунтування ІТ-проєкту, що дозволило визначити ключові фактори успіху даного проекту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Талмазан С.Д. Дослідження та програмна реалізація системи протидії мережевим DDoS-атакам // Збірник праць молодих науковців ЦНТУ. – Вип. 15. – Кропивницький: ЦНТУ, 2025.

2. Mark S. Merkow. Practical Security for Agile and DevOps. CRC Press. 2022. 236 p

3. Derek Fisher. Application Security Program Handbook. Manning Publications. 2021. 155 p.

4. Cameron Wyatt PH.D. Kali Linux Tutorial. Independently published. 2021. 60 p.

5. Alex Matrosov, Eugene Rodionov, Sergey Bratus. Rootkits and Bootkits. No Starch Press. 2019. 450 p.

6. Петрик В.М., Присяжнюк М.М., Аль-Файюмі Халед та ін. «Системи інформаційної зброї та технології інформаційної війни»: підручник / Петрик В.М., Присяжнюк М.М., Аль-Файюмі Халед, Жарков Я.М., Смірнов О.А, Буравченко К.О., Давидюк А.В., Кононович В.Г., Корчинский В.В., Кудирко В.М., Фесенко А.О.; за заг. ред. В.М. Петрика, М.М. Присяжнюка.– К.: Видавничий центр “Кафедра”, 2025.– 320 с.

7. Усік, П.С., Смірнова, Т.В., Буравченко, К.О., Смірнов, О.А., Улічев, О.С., Смірнов, С.А. «Дослідження технологій забезпечення кібербезпеки банківських систем з використанням штучного інтелекту». Кібербезпека: освіта, наука, техніка. 2025. Том 1 № 29. С.704–716, 2025

8. Kuznetsov, O., Frontoni, E., Kuznetsova, K., Arnesano, M., Smirnov, O. «A secure biometric authentication architecture for blockchain-driven cyber-physical systems». Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications, 2025, pp. 193–224.

9. Kuznetsov, O., Atzeni, G., Arnesano, M., Randieri, C., Smirnov, O. «Secure IoT-based smart wheelchair system: From implementation to security enhancement strategy». Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications, 2025, pp. 225–257.

10. Kuznetsov, O., Smirnov, O., Kuznetsova, T., Shaikhanova, A., Svatowsky, I. «Privacy-utility trade-offs in IoT networks: A comparative analysis of differential privacy mechanisms for sensor data aggregation». Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications, 2025, pp. 589–622.

11. Lakhno, V., Malyukov, V., Smirnov, O., Bebeshko, B., Chubaievskiy, V., Zhumadilova, M., Malyukova, I., Smirnov, S. «Multifactorial Model for Targeted Attacks Counteracting Within the Framework of a Multi-Step Quality Game with Fuzzy Information». 8th International Symposium on Intelligent Informatics, ISI 2023, 2025. vol 389. pp 377-389. Springer, Singapore.

12. Kuznetsov O., Frontoni E., Kuznetsova Y., Smirnov O., Moskovchenko I. «Trust-Based Security Architecture for Edge Computing: A Simulation Study of Dynamic Trust Evolution and Attack Detection». CEUR Workshop Proceedings, 2024, 3909, pp. 227–241.

13. Kuznetsov, O., Frontoni, E., Kryvinska, N., Chevardin, V., Smirnov, O. «Wireless Network Encryption Stream Ciphers, Computational Modeling, and Security Analysis». Computational Modeling and Simulation of Advanced Wireless Communication Systems, 2024, pp. 379–402.

14. Kuznetsov, O., Frontoni, E., Kryvinska, N., Smirnov, O., Imoize, G.L. «Computational Modeling of Enhanced Spread Spectrum Codes for Asynchronous Wireless Communication». Computational Modeling and Simulation of Advanced Wireless Communication Systems, 2024, pp. 403–447.

15. Ткаченко, О., Ільєнко, А., Улічев, О., Мелешко, Є., Смірнов, О. «Правові засади поширення інформаційних впливів в соціальних мережах».

Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка», 2024. № 2(26), С. 170–188.

16. Смірнова Т.В., Конопліцька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». Кібербезпека: освіта, наука, техніка. 2024. №4(24), С. 6-27.

17. Вінтенко, Б., Миронець, І., Смірнов, О., Кравчук, О., Козірова, Н., Савеленко, Г., Коваленко, А. «Дослідження вимог та аналіз кібербезпеки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». Кібербезпека: освіта, наука, техніка. 2024. №3(23), С. 111-131.

18. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». Підводні технології, 2024, № 13, с. 28-35.

19. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». CEUR Workshop Proceedings, 2023, 3628, pp. 106-115.

20. Akhalaia, G., Iavich, M., Iashvili, G., Prysiazhnyy, D., Smirnova, T. «Secure Encrypted Connection on Georgian Website». CEUR Workshop Proceedings, 2023, 3550, pp. 313-320.

21. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». Advanced Information Systems, 2023, 7(2), pp. 49-56

22. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». CEUR Workshop Proceedings, Volume 3530, 2023, pp. 256-265.

23. Kuznetsov, O., Kandiy, S., Frontoni, E., Smirnov, O. «Trade-offs in Post-Quantum Cryptography: A Comparative Assessment of BIKE, HQC, and Classic McEliece». CEUR Workshop Proceedings, Volume 3504, 2023, pp. 1-11.

24. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskyi, V., Khorolska, K., Bebeshko, B. «Selection of a Rational Composition of İnformation Protection Means Using a Genetic Algorithm». In: Rajakumar, G., Du, KL., Vuppalapati, C., Beligiannis, G.N. (eds) Intelligent Communication Technologies and Virtual Mobile Networks. Lecture Notes on Data Engineering and Communications Technologies, vol 131. 2023. Springer, Singapore. pp. 21-34.

25. Смірнова Т.В., Гнатюк С.О., Бердибаєв Р.Ш., Сидоренко В.М., Жигаревич О.К., «Система корелювання подій та управління інцидентами кібербезпеки на об’єктах критичної інфраструктури». Кібербезпека: освіта, наука, техніка, №3(19), 2023, С. 176-196.

26. Смірнов О.А. Козлов Я.О., Смірнова Т.В. «Дослідження застосування SIEM-систем для забезпечення кібербезпеки та захисту інформації». II Міжнародна науково-практична Інтернет-конференція «Інновації та перспективні шляхи hозвитку інформаційних технологій (ІПШРІТ-2023)» м.Черкаси 6 грудня 2023 року – Черкаси: ЧДТУ.– 2023. – С.251-252.

27. Козлов Я.О., Смірнова Т.В., Смірнов О.А. «Дослідження SIEM-систем для забезпечення кібербезпеки». VII міжнародна науково-практична конференція “Інформаційна безпека та комп’ютерні технології” до 30-ти річчя кафедри кібербезпеки та програмного забезпечення, м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 26.

28. Козлов Я.О., Козірова Н.Л., Смірнов О.А. «Дослідження структури та принципу роботи SIEM-системи». VII міжнародна науково-практична конференція “Інформаційна безпека та комп’ютерні технології” до 30-ти річчя кафедри кібербезпеки та програмного забезпечення, м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 59.

29. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А., Коваленко А.С. «Дослідження нормативних документів та галузевих стандартів розробки програмного забезпечення комп’ютерних систем управління АЕС,

важливих для безпеки». Системи управління, навігації та зв’язку, 2023, вип. 2(72), С. 170-178.

30. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» CEUR Workshop Proceedings, Volume 3187, 2022,

31. Smirnov O.A., Al-Oraiqat A.M., Ulichev O.S., Meleshko Ye.V., Al-Rawashdeh H.S., Polishchuk L.I. «Modeling strategies for information influence dissemination in social networks». Journal of Ambient Intelligence and Humanized Computing Volume 13, Issue 5. Springer, Cham. 2022, pp. 2463-2477.

32. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». Проблеми інформатизації та управління, № 2(70). 2022. С. 28-37.

33. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» Системи управління, навігації та зв'язку, 2022, № 3(69). С. 93-98.

34. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» Вісник Хмельницького національного університету. Серія: «Технічні науки», № 2 (307). С. 46-52. 2022.

35. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» Системи управління, навігації та зв'язку, 2022, № 1(67). С. 84-89.

36. Smirnov O., Kuznetsov A., Zhora V., Onikiychuk A., Pieshkova O. «Hiding Messages in Audio Files Using Direct Spread Spectrum». 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, Cracow, Poland, 22-25 September 2021. P. 414-418

37. Smirnov O., Kuznetsov A., Lokotkova I., Kuznetsova T., Florov S., Lebid O. «Using Orthogonal Signals to Hide Information in Images». 4 IEEE International Conference on Advanced Information and Communication Technologies (AICT) – 2021, Lviv, Ukraine, September 21-25, 2021. P. 255-260.

38. Smirnov O., Kuznetsov A., Girzheva O., Kiian A., Nakisko O., Kuznetsova T. «Advanced Code-Based Electronic Digital Signature Scheme». 2020 IEEE International Conference on Problems of Infocommunications Science and Technology, PIC S and T 2020, Kharkiv, 6 October 2020-9 October 2020, P. 358-362.

39. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova K. «Data hiding scheme based on spread sequence addressing». CEUR Workshop Proceedings Volume 2805, 2020, Pages 44-58.

40. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». International Journal of Computing; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

41. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.

42. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». Journal of theoretical and applied information technology Vol.98. No 21, 2020, P. 3334-3346.

43. Smirnov O., Kuznetsov A., Arischenko A., Chepurko I., Onikiychuk A., Kuznetsova T. «Pseudorandom sequences for spread spectrum image steganography». CEUR Workshop Proceedings Volume 2654, 2020, Pages 122-131.

44. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New technique for data hiding in cover images using adaptively generated pseudorandom sequences». CEUR Workshop Proceedings Volume 2654, 2020, Pages 1-14.

45. Smirnov O., Lutsenko M., Kuznetsov A., Kiian A., Kuznetsova T., «Biometric cryptosystems: overview, state-of-the-art and perspective directions». Lecture Notes in Networks and Systems, vol 152. Springer, Cham. 2021, pp 66-84.

46. Smirnov O., Kuznetsov A., Onikiychuk A., Makushenko T., Anisimova O., Arischenko A. «Adaptive pseudo-random sequence generation for spread spectrum image steganography». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 161-165.

47. Smirnov O., Kuznetsov A., Kiian A., Babenko V., Perevozova I.,Chepurko I. «New Approach to the Implementation of Post-Quantum Digital Signature Scheme». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 166-171.

48. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

49. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

50. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». CEUR Workshop Proceedings Volume 2616, 2020, Pages 125-136.

51. Smirnov, O., Shekhanin, K., Kuznetsov, A., Krasnobayev, V. «Detecting Hidden Information in FAT». International Journal of Computer Network and Information Security (IJCNIS). Vol. 12, No. 3, 2020. PP.33-43.

52. Smirnov, O., Kuznetsov, A., Gorbacheva, L., Babenko, V., «Hiding data in images using a pseudo-random sequence», CEUR Workshop Proceedings Volume 2608, 2020, Pages 646-660.