Хмельницький національний університет
Факультет програмування та комп'ютерних і телекомунікаційних систем
Кафедра кібербезпеки та комп'ютерних систем і мереж

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Метод виявлення розподілених атак на інформаційну систему
Назва теми

Галузь знань 12 – Інформаційні технології

Спеціальність 123 –Комп'ютерна інженерія

КРМКІ.015085.19.01.09 ПЗ

Виконав: студент 2 курсу, група КІ1м-19-1 _____ Соколюк Я.В.
Підпис

Керівник доц., к. т. н, доцент кафедри КбКСМ _____ Муляр І.В.
Підпис

Нормоконтролер доц., к. т. н, доцент кафедри КбКСМ _____ Муляр І.В.
Підпис

До захисту допускаю:
Зав. кафедри КбКСМ, к.т.н., доц _____ Кльоц Ю.П.
Підпис

10 12 2020 р.

Хмельницький, 2020

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ПРОГРАМУВАННЯ ТА КОМП'ЮТЕРНИХ І ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ

Кафедра КІБЕРБЕЗПЕКИ ТА КОМП'ЮТЕРНИХ СИСТЕМ І МЕРЕЖ

Освітній рівень МАГІСТР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА ПІДГОТОВКИ МАГІСТРА

ЗАТВЕРДЖУЮ

Зав. кафедри Ю.П. Кльоц

"10" 12 2020 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Соколюк Я.В.

Прізвище, ім'я, по батькові студента

Тема проекту (роботи) Метод виявлення розподілених атак на інформаційну систему

1. Керівник проекту (роботи) к.т.н., доц. Міляр І.В.

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом № 118 ректора університету додаток №23 від 01.09.2020

2. Строк подання студентом проєкту (роботи) на кафедру 1.12.2020

3. Вихідні дані до проекту (роботи) Моніторинг сучасних DDoS-атак, собливості DDoS-атак регіонального рівня, вимоги до методики та засоби з виявлення

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Аналіз сучасних DDoS - атак, методів і засобів протидії.

Математична модель виявлення початку атаки і шкідливого трафіку.

Комплексний метод фільтрації трафіку.

Методика визначення вразливостей до DDoS - атакам систем управління

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) Тема, мета магістерської роботи, задачі дослідження, наукова новизна, практична цінність,. Типи DDoS-атак. Запропонована класифікація. Модель виявлення початку атаки з урахуванням сезонності. Математичне обгрунтування вибору актуальних сезонних періодів. Метод кластеризація. Алгоритми по визначенню початку атаки і виділенню шкідливого трафіку. Структура програмного комплексу Висновки.

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прий
Відповідальний за оформлення КРМ	Муляр І.В.		

7. Дата видачі завдання «1» лютого 2020 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) кваліфікаційної роботи	Термін виконання етапів проекту (роботи)	Прим
1	Вибір напряму дослідження та узгодження тематики КРМ з керівником	2.02.2020	
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	2.03.2020	
3	Робота над розділом 1 – аналіз відомих моделей, методів за темою; постановка задачі	1.04.2020	
4	Робота над розділом 2 – розробка моделей і методів для вирішення поставленої задачі	1.05.2020	
5	Робота над науковою статтею	1.06.2020	
6	Робота над розділом 3 – розробка алгоритмів та технологій, їх аналіз	1.09.2020	
7	Робота над розділом 4 – проектування ПЗ для вирішення поставленої задачі	1.10.2020	
8	Узгодження отриманих; оформлення пояснювальної записки згідно вимог	1.11.2020	
9	Оформлення графічної частини	11.11.2020	
10	Попередній захист КРМ	15.11.2020	
11	Захист ДРМ на засіданні ЕК	10.12.2020	

Студент _____ Сколоз І.В.
Підпис Ініціали, прізвище

Керівник проекту (роботи) _____ Муляр
Підпис Ініціали, прізвище

АНОТАЦІЯ

Тема кваліфікаційної роботи: Метод виявлення розподілених атак на інформаційну систему

Автор роботи: Соколюк Я.В.

Керівник роботи: к.т.н., доц. Муляр І.В.

Пояснювальна записка: 92 с., 22 рис., 3 табл., 3 дод., 53 джерелf.

МЕРЕЖНА ІНФОРМАЦІЙНА СИСТЕМА, РОЗПОДІЛЕНІ АТАКА, ІНФОРМАЦІЙНІ ПОТОКИ

Метою магістерської роботи є розроблення методу для раннього виявлення DDoS атак, і подальшого блокування загрозливого трафіку на стороні ресурсу, що атакується.

Запропонований метод виявлення та блокування шкідливого трафіку DDoS-атак на ранніх стадіях базується на аналізі сезоності мережного трафіку. Розроблено алгоритм визначення точок початку атаки та алгоритм поділу змішаного трафіку на надійний та загрозливий, який враховує сезонну кількість мережного навантаження. Розроблено програмний засіб для виявлення початку атаки та подальшого виявлення та блокування нелегітимних звернень. Його особливістю є модульність та універсальність для забезпечення безпеки різних мережних ресурсів та захисту їх від атак різного типу.

Дата 9.12.20 Підпис студента

ANNOTATION

a master's degree work of Sokoliuk Yaroslav
entitled «Method of Identifying Distributed Attacks on an Information System».

Mentor: Ihor Muliar

Total volume of work: 92 pages, 22 figures, 3 tables, 3 appendices, 53 references.

NETWORK INFORMATION SYSTEM, DISTRIBUTED ATTACK, INFORMATION FLOWS

The purpose of the master's thesis is to develop a method for early detection of DDoS attacks, and subsequent blocking of threatening traffic on the side of the attacked resource.

The proposed method of detecting and blocking malicious traffic DDoS-attacks in the early stages is based on the analysis of seasonality of network traffic. An algorithm for determining the starting points of the attack and an algorithm for dividing mixed traffic into reliable and threatening, which takes into account the seasonal amount of network load. A software tool has been developed to detect the onset of an attack and the subsequent detection and blocking of illegitimate appeals. Its feature is modularity and versatility to ensure the security of different network resources and protect them from attacks of different types.

Date 9.12.20. Signature

ЗМІСТ

ВСТУП

Актуальність роботи. У сучасному світі використання комп’ютерів та комп’ютерних мереж зростає з кожним днем. Все частіше зустрічаються як мобільні пристрої - телефони і планшети, так і розумна побутова електроніка - телевізори, холодильники, сучасні ігрові приставки. Однією з найпоширеніших загроз є атаки відмови в обслуговуванні. Ця атака робить систему неможливою, частково або повністю блокує ресурси та послуги, необхідні користувачеві.

Атаки відмови в обслуговуванні можна розділити на 2 основні групи - атаки відмови в обслуговуванні та розподілені атаки відмови в обслуговуванні. Останній характеризується використанням декількох мережних вузлів для здійснення атаки, як правило, досить великої кількості, що дуже ускладнює виявлення такої атаки та захист від неї.

Для полегшення виявлення та захисту від таких атак необхідно мати чітку класифікацію за різними критеріями. В даний час існує велика кількість класифікацій DoS-атак [11], але відсутня така, яка максимально характеризує всі сучасні особливості DoS-атак, з можливістю застосування в реальних системах.

Розподілена відмова в обслуговуванні (DDoS) – це одна з найнебезпечніших загроз захисту інформації в комп’ютерних мережах. DDoS-атака - це напад на комп'ютерну систему з метою зробити недоступними комп'ютерні ресурси для користувачів, яким призначена комп'ютерна система.

Кількість інцидентів та потужність DDoS-атак постійно зростають, у березні 2018 року була зареєстрована найпотужніша DDoS-атака в історії з піковою пропускною здатністю 400 Гбіт / с. На той момент результати атаки відчули мільйони користувачів Інтернету, в основних центрах обміну трафіком (Internet Exchange Point) курс обміну даними впав у 3 рази, доступ до більшості сайтів був обмежений.

Традиційно DDoS-атаки можна розділити на два підтипи: мережні та

прикладні. Кілька років тому DDoS-атаки на рівні мережі, спрямовані на перевантаження каналів, були найефективнішими, але сьогодні вони не представляють серйозної загрози через розроблені методи боротьби з ними. У той же час DDoS-атаки на рівні додатків продовжують залишатися ефективним інструментом для перевантаження інформаційних ресурсів. DDoS-атаки на рівні додатків - це атаки, націлені на мережеві служби на рівні програми згідно з класифікацією довідкової моделі OSI / ISO. Зазвичай вони призводять до завантаження великої кількості масових файлів або переповнення ресурсу ресурсоємними запитами. При використанні протоколу HTTP у DDoS-атаках (HTTP-повені) рівня додатків синтаксис HTTP-запитів та характеристики рівня трафіку нічим не відрізняються від легального трафіку, а тому їх виявлення в реальному часі є цілком виклик. Рішенням проблеми DDoS-атак на рівні додатків є раннє виявлення факту їх початку, виявлення джерел атак, а також повідомлення адміністраторам зомбі. У цьому випадку атаку можна ефективно контролювати.

Нещодавно були запропоновані різні методи та способи зменшення впливу DDoS-атак на мережеве середовище. Jie-Hao та Ming [42] використовували штучні нейронні мережі для виявлення DDoS-атак. Автори порівнюють ефективність виявлення DDoS-атак за допомогою таких методів, як дерево рішень, аналіз ентропії, метод наївного байєсівського класифікатора та метод штучних нейронних мереж. Лю та Гу [34] використовували нейронну мережу з квантуванням векторів у процесі навчання. Це опція керованого квантування, яка може бути використана для розпізнавання зразків, багатокласової класифікації та стиснення даних. Akilandeswari та Shalinie [38] представили класифікатор атак на основі імовірнісної нейронної мережі. Сіатерліс та Магларіс [45] провели експерименти з багатошаровим персептроном як алгоритмом прийняття рішень у класифікації трафіку. Бхаранідхаран Шанмугам та Норбік Баша Ідріс [41] запропонували використовувати гібридну систему виявлення вторгнень на основі нечіткої логічної апаратури. Система складається з таких компонентів: аналізатор даних, модуль

аналізу даних та модуль нечіткого виводу. Аналізатор даних перевіряє трафік і виконує агрегування пакетів. Сукупна інформація надходить у модуль аналізу даних, який генерує правила фільтрації. Правила фільтрації та мережевий трафік надходять до нечіткого модуля виводу, який вирішує, чи існує загрозливий вплив на систему. На практиці метод не може бути використаний для виявлення атак у режимі реального часу через його високу обчислювальну складність та високий рівень помилкових спрацьовувань.

Метою магістерської роботи є розроблення методу для раннього виявлення DDoS атак, і подальшого блокування загрозливого трафіку на стороні ресурсу, що атакується.

Для досягнення зазначеної мети в магістерській роботі поставлено і вирішено такі завдання:

1. Проаналізовано сучасний стан DDoS атак, та проведено аналіз нинішнього стану технологій для вирішення проблем захисту інформації

2. Розроблено математичну модель атак з врахуванням опису сезонності мережного навантаження.

3 Розроблено алгоритм визначення точок початку атаки та алгоритм поділу змішаного трафіку на надійний та загрозливий, який враховує сезонну кількість мережного навантаження.

4. Вирішено завдання по створенню методу і програмного комплексу по виявленню DDoS-атакам малої інтенсивності.

5. Розроблено програмний засіб для виявлення початку атаки та подальшого виявлення та блокування нелегітимних звернень. Його особливістю є модульність та універсальність для забезпечення безпеки різних мережних ресурсів та захисту їх від атак різного типу.

Об'єктом дослідження є розподілені атаки на інформаційну систему.

Предметом дослідження виступають моделі та методи виявлення DDoS атак, і виділення зловмисного трафіку.

Методи дослідження, використані в магістерській роботі: апарат теорії алгоритмів, теорії захисту інформації, системного аналізу, теорії імовірності та математичної статистики, кластерного і системного аналізу.

Наукова новизна досліджень полягає в наступному:

1. Розроблено математичну модель атак, яка враховує опис сезонності мережного трафіку для різної періодичності

2. Вдосконалено метод раннього виявлення та протидії DDoS атакам, особливостями якого є врахування сезонних періодів

Практичне значення магістерської роботи полягає у створенні методу та алгоритмів захисту мережних ресурсів від DDoS-атак, що дозволяють проводити активну протидію безпосередньо на стороні атакованого ресурсу. Це підтверджується розробкою та подальшою реалізацією розробленого програмного пакету для виявлення DDoS-атак, та подальше блокування нелегітимних звернень на різних рівнях.

Апробація результатів роботи. За темою магістерської опубліковано 1 наукова стаття і 1теза доповідей.

1 АНАЛІЗ РОЗПОДІЛЕНИХ АТАК, МЕТОДІВ І ЗАСОБІВ БОРОТЬБИ З НИМИ

1.1 Дослідження сучасного стану DDoS-атак

Розподілена атака на відмову в обслуговуванні (DDoS) - це зловмисна спроба порушити нормальний трафік цільового сервера, послуги або мережі, переповнюючи ціль або навколишню її інфраструктуру потоком Інтернет-трафіку

DDoS-атаки досягають ефективності завдяки використанню кількох скомпрометованих комп'ютерних систем як джерел трафіку атак. Експлуатовані машини можуть включати комп'ютери та інші мережеві ресурси, такі як пристрої IoT [14].

Ці мережі складаються з комп’ютерів та інших пристроїв (наприклад, пристроїв IoT), інфікованих шкідливим програмним забезпеченням , що дозволяє віддалено керувати ними зловмисником. Ці окремі пристрої називаються ботами, а група ботів називається бот-мережею. Після встановлення бот-мережі зловмисник може керувати атакою, надсилаючи віддалені інструкції кожному боту.

Коли бот-мережа націлена на сервер або мережу жертви, кожен бот надсилає звернення на IP-адресу цілі , що потенційно може спричинити перевантаження сервера або мережі, що призводить до відмови в обслуговуванні звичайного трафіку. Оскільки кожен бот є законним Інтернет-пристроєм, відокремити трафік атаки від звичайного може бути важкою.

Найбільш очевидним симптомом DDoS-атаки є те, що сайт або служба раптово стають повільними або недоступними. Але оскільки низка причин - такий законний стрибок трафіку - може створити подібні проблеми з продуктивністю, як правило, потрібне подальше розслідування. Інструменти аналізу трафіку можуть допомогти вам виявити деякі з цих виразних ознак DDoS-атаки:

- підозрілий обсяг трафіку, що надходить з однієї IP-адреси або діапазону IP;
- потік трафіку від користувачів, які мають єдиний поведінковий профіль, наприклад, тип пристрою, геолокацію або версію веб-браузера;
- незрозумілий сплеск звернень до однієї сторінки чи кінцевої точки;
- дивні схеми руху, такі як стрибки в непарні години доби або схеми, які здаються неприродними (наприклад, стрибок кожні 10 хвилин).

Хоча майже всі DDoS-атаки передбачають переповнення цільового пристрою або мережі трафіком, атаки можна розділити на три категорії [16]. Зловмисник може використовувати один або декілька різних векторів атаки або циклювати вектори атаки у відповідь на контрзаходи.

Атаки на основі об’єму [32]. Потоки UDP (User Datagram Protocol) атакують випадкові порти на віддаленому сервері із запитами, що називаються UDP-пакетами. Хост перевіряє порти на відповідні програми. Коли жодної програми не вдається знайти, система відповідає на кожен запит пакетом “недоступний для призначення”. Отриманий трафік може переповнити послугу.

Поток ICMP (ping): Поток протоколу керування Інтернетом (ICMP) надсилає пакети ехо-запиту (pings) ICMP хосту. Пінги - це загальні звернення, що використовуються для вимірювання підключення двох серверів. Коли відправляється пінг, сервер швидко реагує. Однак під час повені пінгу зловмисник використовує велику серію пінгів, щоб вичерпати вхідну та вихідну пропускну здатність цільового сервера.

При об'ємних DDoS-атаках зловмисники зазвичай засипають жертву великим обсягом пакетів або з'єднань, переважною мережевим обладнанням, серверами або ресурсами пропускної здатності. Це найбільш типові DDoS-атаки. У минулому об'ємні атаки здійснювали численні компрометовані системи, які були частиною бот-мережі; зараз хакери не тільки використовують звичайні методології атак, але й набирають добровольців для здійснення цих атак зі своїх машин. Крім того, нові

центри величезних об'ємних атак зараз запускаються з центрів обробки даних провайдерів хмарних послуг, коли зловмисники або орендують, або компрометують хмарні системи, що мають величезну пропускну здатність Інтернету.

Ботнет - це група підключених до Інтернету компрометованих систем, яка може використовуватися для надсилання повідомлень електронної пошти зі спамом, участі в DDoS-атаках або виконання інших незаконних завдань. Слово ботнет походить від слів робот і мережа. Скомпрометовані системи часто називають зомбі. Зомбі може бути скомпрометовано шляхом обману користувачів, щоб вони здійснили "проїжджаюче" завантаження, використали вразливості веб-браузера або переконали користувача запустити іншу шкідливу програму, таку як програма троянських коней. На рисунку 2 наведено приклад типового ботнету.

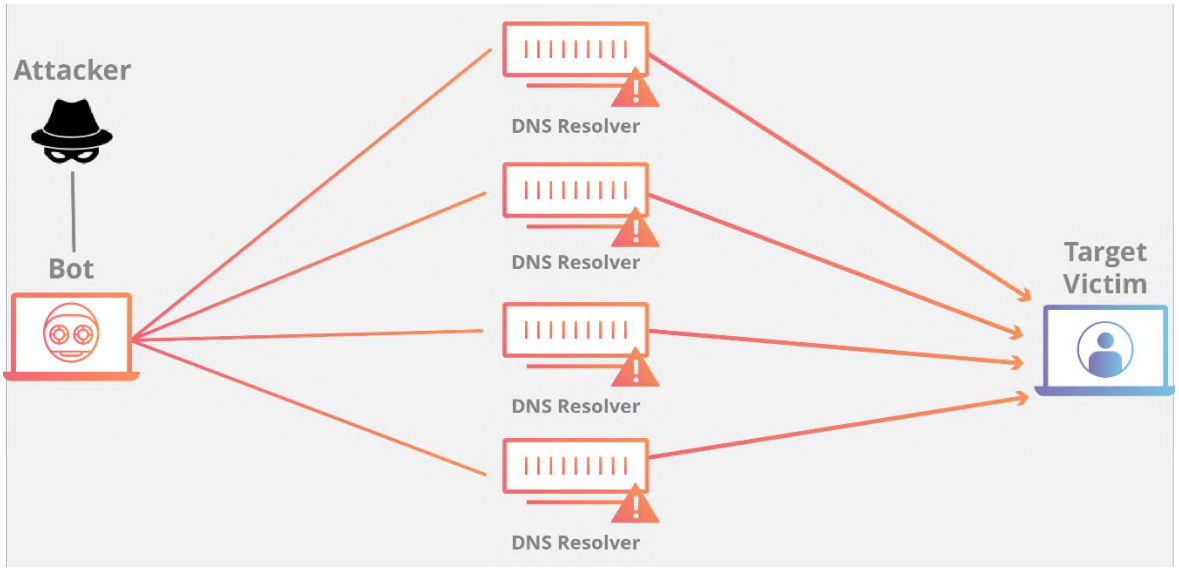


Рисунок 1.1 - Атаки на основі ботнету

У цьому прикладі зловмисник контролює зомбі для запуску DDoS-атаки на інфраструктуру жертви. Ці зомбі запускають прихований канал для зв’язку з сервером управління і управління, яким керує зловмисник. Це спілкування часто

відбувається через Інтернет-ретрансляційний чат (IRC), зашифровані канали, певні однорангові мережі та навіть Twitter.

З появою хмарних послуг та провайдерів з'явилася нова тенденція. Зловмисники орендують або компрометують великі машини обробки даних / хмари для запуску DDoS-атак. Хмарні обчислення не лише створюють нові можливості для законних організацій; він також забезпечує чудову платформу для кіберзлочинців, оскільки недорого та зручно дозволяє їм використовувати потужні обчислювальні ресурси для вчинення поганих дій. Ця концепція проілюстрована на рис. 1.2.

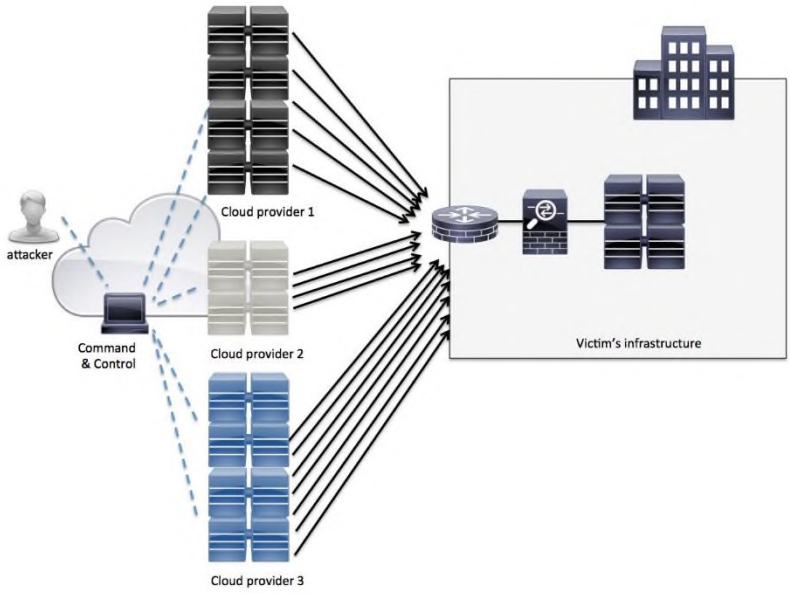


Рисунок 1.2 - Компрометовані хмарні сервери

Атаки посилення DNS [22]. Запит системи доменних імен (DNS) може бути рекурсивним або нерекурсивним (або ітеративним). Клієнтські програми, такі як Інтернет-браузери, зазвичай вимагають, щоб сервер DNS виконував рекурсію, встановлюючи прапорець бажаної рекурсії (RD) у пакеті звернень DNS. Якщо DNS-сервер не може відповісти на запит ні з кешу, ні з інформації про зону, сервер буде запитувати допомогу від інших DNS-серверів. На жаль, багато рекурсивних

серверів імен приймають звернення DNS із будь-якого джерела. Крім того, багато реалізацій DNS за замовчуванням дозволяють рекурсію, навіть коли передбачається, що сервер імен обслуговує лише авторитетні звернення. Це відоме як відкритий вирішувач. Відкриті вирішувачі DNS вразливі до численних шкідливих атак, таких як отруєння кешу DNS та атаки DDoS.

Атака посилення DNS - це найпоширеніша атака DDoS, яка використовує рекурсивні сервери імен, хоча для деяких атак посилення DNS рекурсивний сервер може не вимагати успіху. Атаки посилення DNS подібні до атак smurf. Під час атаки smurf зловмисник може надсилати підроблені ICMP-ехо-звернення (тип 8) для створення умови DoS. У підсилювачі DNS-атаки DDoS-зловмисник зловмисник надсилає невеликі підроблені звернення адреси до відкритого розподільника, змушуючи його надсилати значно більші відповіді на ціль підробленої адреси. Згодом вирішувач сприяє DDoS-атаці на підроблені адреси. Рисунок 1.3 ілюструє основні етапи DDoS-атаки посилення DNS.

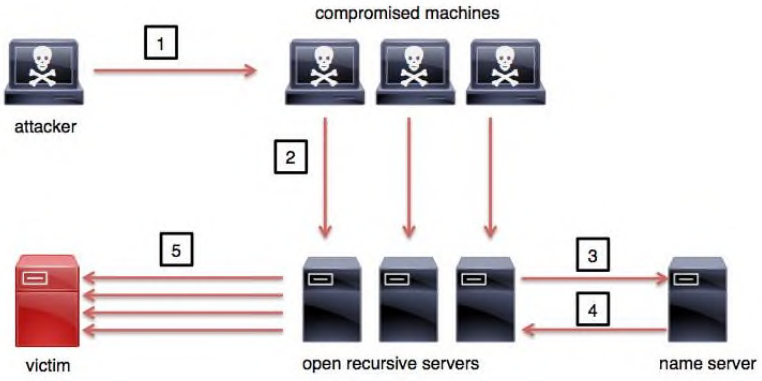


Рисунок 1.3 - Атака посилення DNS

Наступні кроки проілюстровані рис.1.3:

1. Зловмисник запускає та скеровує скомпрометовані машини для початку атаки

2. Взломані машини надсилають DNS-запит для домену *example.com* і встановлюють вихідну IP-адресу IP-адресою жертви

3. Відкриті сервери розпізнавача запитують вищі сервери імен про розташування *example.com*

4. Сервер імен надсилає відповідь назад на відкриті рекурсивні сервери

5. Відкриті рекурсивні сервери надсилають DNS-відповіді жертві

Атаки на рівні додатків [24]. Поток HTTP - це атака додатків рівня 7, яка використовує бот-мережі, які часто називають "армією зомбі". При цьому типі атаки стандартні звернення GET та POST заповнюють вебсервер або додаток. Сервер переповнений запитами та може вимкнутись. Ці атаки може бути особливо важко виявити, оскільки вони виглядають як цілком дійсний трафік.

З іншого боку, інші програми, такі як Voice over IP (VoIP), DNS та інші, часто націлені.

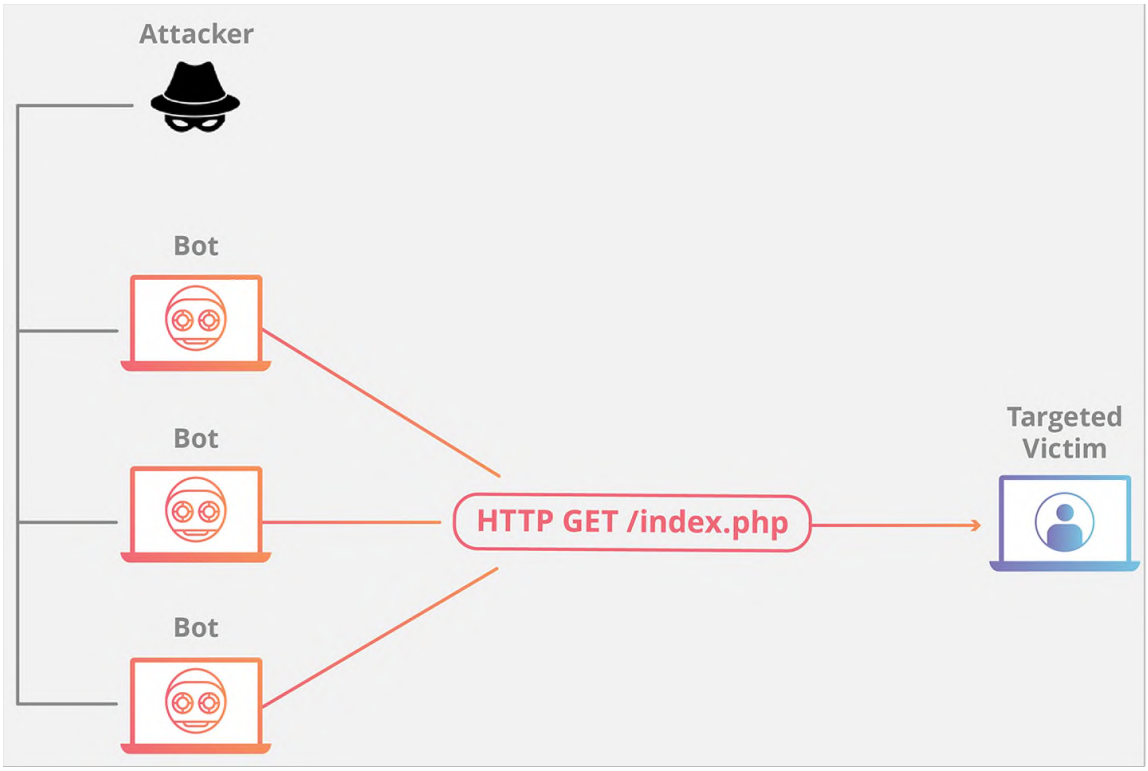


Рисунок 1.4 - Атаки на рівні додатків

DDoS-атаки на програми можуть бути націлені на багато різних додатків; однак найпоширеніший цільовий HTTP, спрямований на вичерпання веб-серверів

та служб. Деякі з цих атак характерно ефективніші за інші, оскільки для досягнення своєї мети їм потрібно менше мережних з'єднань. Наприклад, зловмисник може запустити численні HTTP GET або POSTS, щоб вичерпати веб-сервер або веб-програму [19].

Можливість реалізації повільної DoS-атаки обумовлюється особливостями роботи інтернет протоколу TCP, а саме реалізованому механізму тайм-ауту та можливістю повторної передачі пакету. Цей механізм функціонує наступним чином: після відправлення пакету відбувається пакеочікування пакету-відповіді протягом певного інтервал часу RTO (Retransmission TimeOut). Саме цю особливість використовує зловмисник для реалізації атаки. Він надсилає імпульс трафіку у кінці інтервалу RTO. Через це, канал зв'язку переповнюється в момент, надходження пакету - відповіді, по цій причині вони не отримуються. Потім він ще раз повторює свої дії. Таким чином, виникає з'являється стійка непрацездатність системи. Представлення повільної DoS-атаки в графічному вигляді, отримане при її експериментальному дослідженні, зображено на рис. 1.5.

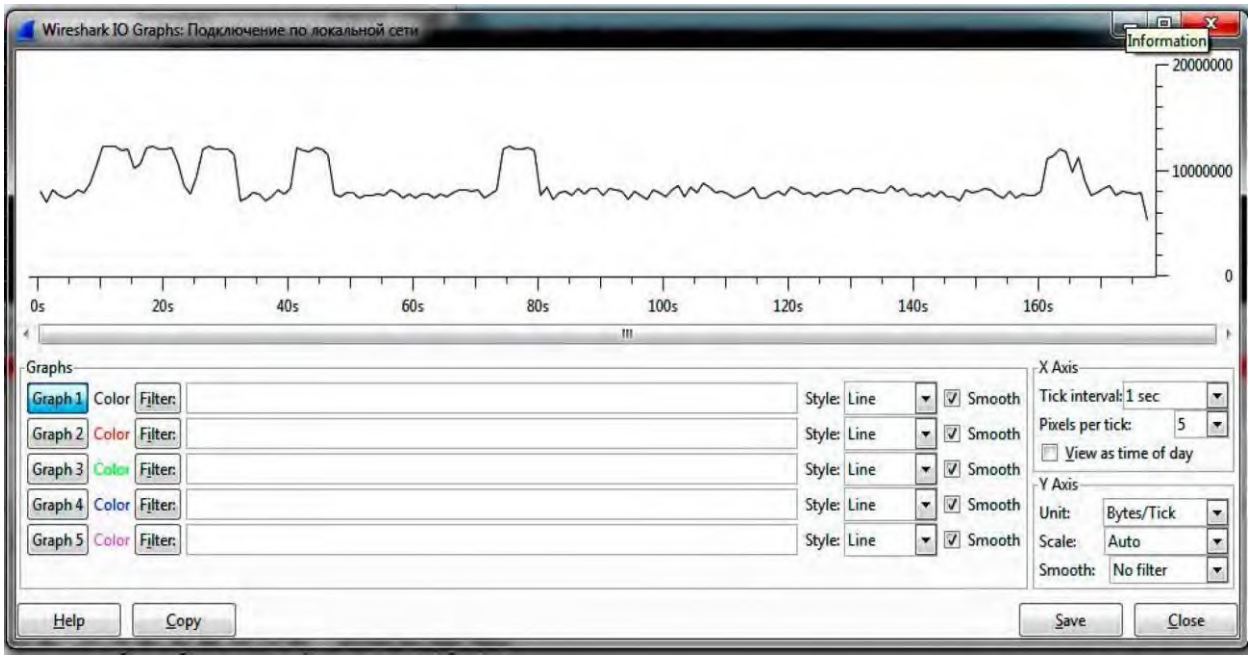


Рисунок 1.5 - Графічне зображення повільної DoS-атаки

Фундаментальною можливістю для виявлення повільних DoS-атак може бути відслідковування контрольних характеристик трафіку в штатному режимі з максимальною завантаженістю мережі і подальше виявлення аномалій у структурі трафіку. Під явищем аномалії розуміється подія, яка характеризується відхиленням від стандартної структури трафіку, що отримана раніше. Але сама аномалія ще не говорить про те, що точно йде атака. Для прийняття рішення про наявність загрози необхідний аналіз отриманої аномалії. Під час повільних DoS-атаках аномалії можуть мати вигляд короткочасних невеликих сплесків трафіку, як зображено на рис. 1.5.

Отримання декількох піків трафіку з вказаними інтервалами однозначно свідчить про те, що імовірно йде повільна DoS-атака. Але інтервал часу, за який проходить один цикл атаки, насправді досить великий. Тому очікувати довгий час, щоб змогти зробити висновок про наявність повільної DoS-атаки, досить нераціонально і небезпечно , зважаючи на нормальну працездатність системи. Тому з’являється необхідність вибору меншого інтервалу часу спостереження.

Ще одним видом атак є низькошвидкісні атаки DoS (LDoS). Вони часто використовують недоліки реалізації програми та недоліки дизайну [38]. Slowloris: Названий на честь азіатського примата, Slowloris рухається повільно. Атака надсилає невеликі частини HTTP-запиту на сервер [34]. Ці частини надсилаються із інтервалами, тому запит не вичерпується, та сервер чекає його завершення. Ці незавершені звернення вичерпують пропускну здатність та впливають на здатність сервера обробляти законні звернення.

Slowloris - це інструмент атаки, створений RSnake (Роберт Хансен), який намагається тримати численні з'єднання відкритими на вебсервері. Загроза працює, відкриваючи з'єднання на сервері жертви та надсилаючи частковий запит. Із перервами атака надсилає наступні заголовки HTTP [32]. Однак атака не завершує запит на підтримку цих зв’язків як відкритих, поки жертва не зможе обробити звернення від законних клієнтів.

Існують подібні засоби та методології атаки. Нижче наведено кілька її прикладів:

- PyLoris
- QSlowloris (варіант Slowloris для Windows)
- повільний

Іонна гармата з низькою орбітою (LOIC) та Canon Ion з високою орбітою (HOIC) стали популярними інструментами DDoS нападів для таких хакерських груп, як Anonymous чи Syrian Electronic Army. Ці інструменти дозволяють навіть нетехнічним людям створювати DDoS-атаку за допомогою декількох клацань за допомогою власних комп'ютерів замість традиційних ботових атак.

DDoS-атаки з нульовим днем (їх часто називають вбивцями одного пакета) - це вразливості в системах, які дозволяють зловмиснику відправляти один або кілька пакетів до ураженої системи, що спричиняє стан DoS (збій або перезавантаження пристрою). Ці атаки часто є найбільш прихованими та важкими для виявлення, оскільки вони часто невідомі постачальникам, і жодних патчів чи обхідних шляхів не існує. Зазвичай такі уразливості та експлойти продаються на підпільному ринку, що робить їх однією з найбільших загроз для будь-якої організації. Озброєння цих типів подвигів стає новою нормою для кіберзлочинців.

Атаки на рівні протоколів [17]. Затоплення SYN: під час атаки SYN-потоку зловмисник надсилає, здавалося б, звичайні звернення SYN на сервер, який відповідає запитом SYN-ACK (синхронізоване підтвердження). Зазвичай клієнт відправляє назад запит ACK, і встановлюється з'єднання. Під час нападу SYN-повінь зловмисник не відповідає остаточним ACK. На сервері залишається велика кількість незавершених звернень SYN-ACK, які обтяжують систему.

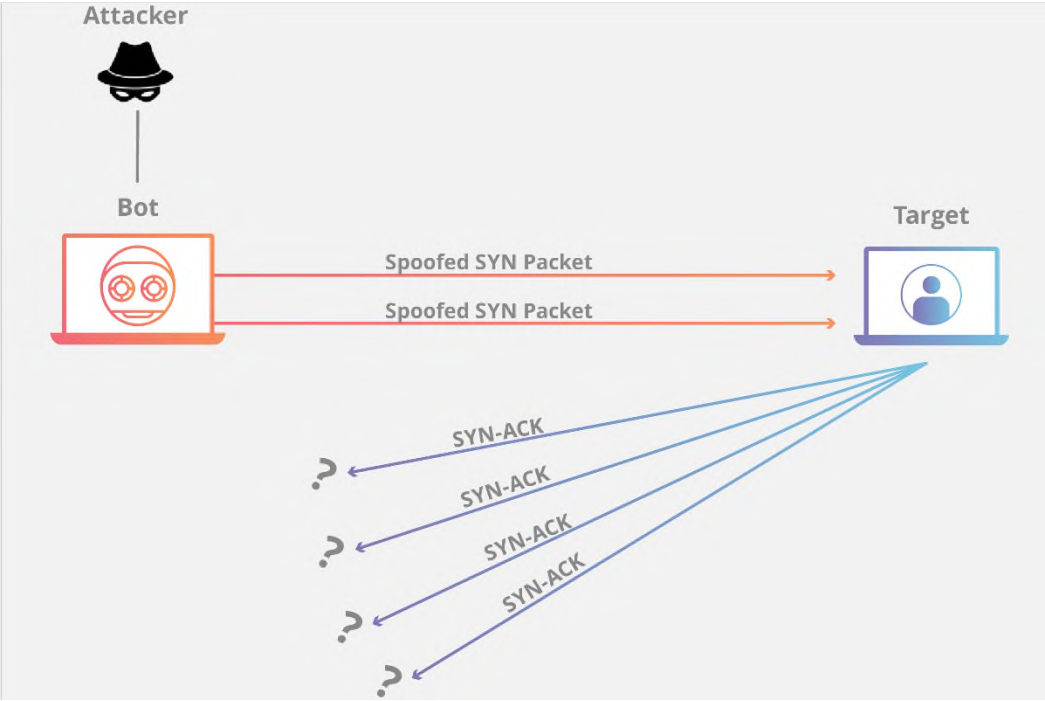


Рисунок 1.6 - Протокольні атаки

Ping of Death: під час атаки Ping of Death зловмисник намагається зірватися або заморозити сервер, надіславши звичайний запит ping, який або фрагментований, або негабаритний. Стандартний розмір заголовка IPv4 становить 65 535 байт. Коли надсилається більший пінг, цільовий сервер буде фрагментувати файл. Пізніше, коли сервер сформулює відповідь, повторна збірка цього більшого файлу може спричинити перевантаження буфера та збій.

Потоки протоколів повідомлень керування Інтернетом. Атаки повені через протокол ICMP (Internet Control Message Protocol) існують багато років. Вони є одними і, чи флуд атаки – це один з найдавніших типів атак DoS. При атаках повені ICMP зловмисник переповнює цільовий ресурс пакетами exo-запиту (ping) ICMP, великими пакетами ICMP та іншими типами ICMP, щоб значно наситити та уповільнити мережеву інфраструктуру жертви. Це показано на рис 1.7.

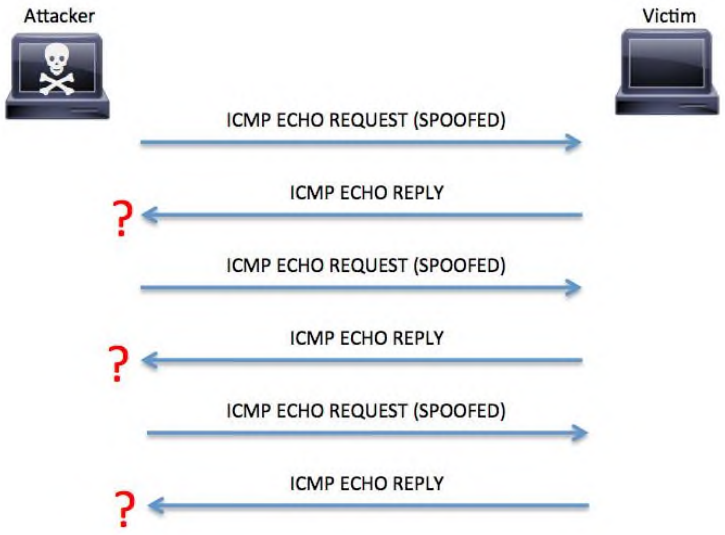


Рисунок 1.7 - Приклад повені ICMP

Інший тип атак на основі ICMP - це атака smurf [26]. Назва *smurf* походить від оригінального вихідного коду інструмента exploit *smurf.c* , створеного особою під назвою TFreak у 1997 році. Під час атаки smurf зловмисник передає велику кількість ICMP-пакетів із підробленим IP-адресою жертви в мережу за допомогою IP-адреса трансляції. Це змушує пристрої в мережі реагувати, надсилаючи відповідь на вихідну IP-адресу. Цей обмін показано на рис 1.8.

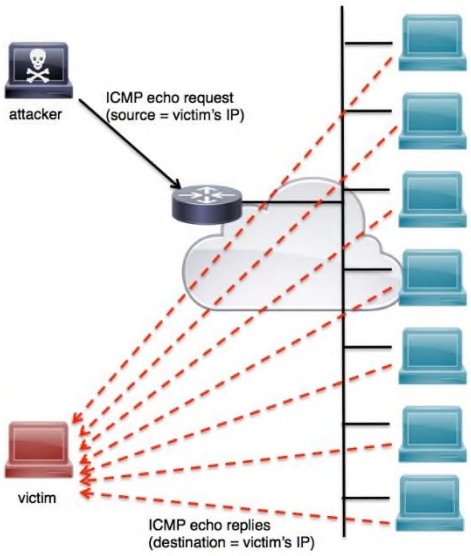


Рисунок 1.8 - Смурф атака

Цю атаку можна легко пом'якшити на пристрої Cisco IOS, використовуючи команду підінтерфейсу no ip спрямованої трансляції , як показано в наступному прикладі:

- маршрутизатор (конфігурація) # інтерфейс GigabitEthernet 0;
- маршрутизатор (конфігурація-якщо) # немає ip спрямованої трансляції;
- SYN flood атаки.

Коли хост (клієнт) ініціює TCP-з'єднання з сервером, клієнт і сервер обмінюються низкою повідомлень для встановлення з'єднання. Це встановлення зв'язку називається тристороннім рукостисканням TCP. Це показано на рис 1.9.

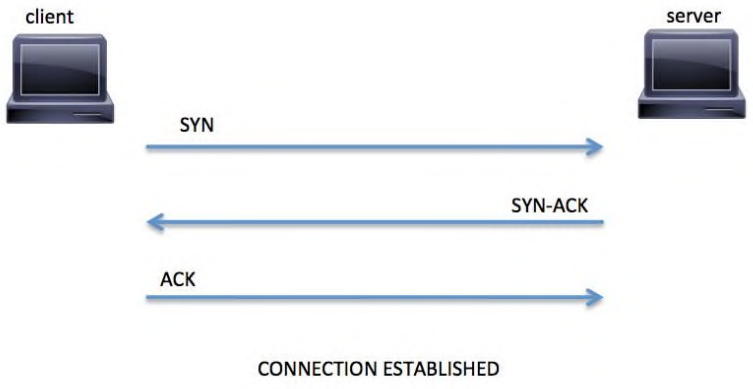


Рисунок 1.9 - Тристороннє рукостискання TCP

- Клієнт запитує з'єднання, надсилаючи на сервер безпосережньо повідомлення SYN (синхронізація)
- Сервер підтверджує цей запит, надсилаючи SYN-ACK назад клієнту
- Клієнт відповідає ACK (підтвердженням), тп зв’язок встановлюється.

Атаки потоку UDP. Подібно атакам потоку TCP, основною метою зловмисника при виконанні атаки потоку UDP є спричинення голодування системних ресурсів. Атака потоку UDP запускається шляхом надсилання великої кількості пакетів UDP до випадкових портів системи жертви. Система помітить, що жодна програма не прослуховує цей порт і не відповідає пакетом недоступного призначення ICMP. Згодом, якщо надсилається велика кількість пакетів UDP,

жертва буде змушена відправити численні пакети ICMP. У більшості випадків саме ці атаки здійснюються завдяки підробці джерела IP-адреси зловмисника. Більшість сучасних операційних систем тепер обмежують швидкість надсилання відповідей ICMP, мінімізуючи вплив та трохи пом'якшуючи такий тип DDoS-атаки.

Краплеві атаки. Атаки сльози передбачають надсилання створених пакетів із накладанням великих корисних навантажень на систему жертв. Сучасні операційні системи навіть тепер не застраховані від цього виду атаки, але через дефіцит фрагментації TCP і повторне збірку реалізації застарілих операційних систем цей вид атак може спричинити збій цих систем.

Підсумовуючи розглянуті типи DDoS-атак пропонується класифікацію по наступних критеріях (рис. 1.10).

За кількістю задіяних пристроїв - за цією ознакою атаку можна розділити на: прості DoS-атаки; групові DDoS-атаки - за допомогою малої кількості добровільно задіяних комп’ютерів (до 100 пристроїв) та масових DDoS-атак - понад 100 задіяних пристроїв.

Тому, потрібно розрізняти методи боротьби з такими видами атак. У випадку простих або групових атак можна зробити блокування пакети із відповідних машин вручну на підставі чорного списку. У випадку масових атак досить проблематично блокувати всі вірогідні джерела атак вручну, та здійснити ідентифікацію законних користувачів від зловмисників.

Відповідно до джерела атаки на зловмисника, напади можна поділити на: добровільні атаки безпосередньо з машин зловмисників, напади з використанням ботнетів, загрози з використанням фізичних і віртуальних виділених серверів, напади з використанням проміжних машин і інструментів тунелювання, і також атаки з використанням випадкових користувачів.

Слід зазначити, що якщо атака здійснюється з виділених серверів, трафік може бути значним навіть на невеликій кількості пристроїв, оскільки такі сервери зазвичай мають мультигігабітні канали. Забезпечуючи захист від атак із проміжних

інструментів тунелювання, слід взяти до уваги, що один інструмент тунелювання може бути використаний як для атак, так і для здійснення легального доступу до служби. І визнаючи цю різницю може бути досить важко.

Рисунок 1.10 - Класифікація DDoS-атак

Атаки з використанням бот-мереж розділяються на атаки з використанням мобільних пристроїв, інфікованих серверів [12], та домашніх комп'ютерів.

За кількістю атакуючих машин DDoS-атаки можна позділити на статичні (фіксована кількість пристроїв), з динамічним керуванням (кількість і розташування атакуючих пристроїв може змінюватися з часом, та існує

формальний список джерел нападу, наприклад користувачів каналів IRC, вузлів Tor, список IP-адрес) та динамічні без контролю - характеризуються відсутністю технічної можливості, щоб створити список атакуючих машин.

За способом управління атаки можна поділити на: керовані (віддалене керування нападом), ручні (зловмисник вручну відправляє пакети), та автоматичні (виконання атаки без безпосереднього втручання людини).

Відповідно, засоби управління атакою можна розділити засобами зв'язку на пряме управління (централізоване управління, де пристрої мають відкритий порт для ідентифікації вихідних машин) та непряме керування (машини не мають відкритого порту, керуються за допомогою зворотних з'єднань чи додаткових протоколів) [5].

Безпосередньо керовані атаки можна розділити за мережним підключенням: випадкове сканування - нападник випадково сканує IP-адреси для пошуку інфікованих машинах, сканує по списку - зловмисник має список інфікованих машин і зворотній зв'язок - інфіковані машини таємно повідомляють його про своє зараження.

За вразливістю атаки діляться на семантичні - використання певних служб чи протоколу та «паводкові» - спроби "дурного" перевантаження розміром пакетів або їх величезною кількістю.

За правильністю вихідної адреси атаки можна розділити на: атаки з правильним джерелом - ви можете чітко визначити джерело атаки в пакеті даних, атаки з підробленим джерелом - адреса джерела в пакеті відсутня або вказана некоректно, та зворотні атаки - здається, що атака здійснюється легальною службою, яка насправді просто відповідає на неправильно сформовані запити (DNS, Google).

За силою напади можна поділити на: атаки з постійною силою - атака виконується з фіксованою силою, з випадковою потужністю - сила змінюється хаотично, з певною змінною силою - потужність змінюється за відомим

алгоритмом, атаки з коливальною силою - сила атаки коливається або пульсує, атаки зі збільшенням потужності - сила атаки постійно зростає.

За рівнем реалізації напади діляться на атаки на фізичному рівні - фізичні перешкоди в комп'ютерній системі, наприклад обрив кабелю, підвищення напруги, випромінювання, атаки на рівні каналу [7] - напади на рівні фізичної, атаки на мережному рівні - напад на рівні IP-пакетів, атаки на транспортному рівні - напад на рівні сегментів і дейтаграм, атаки на рівні сесії - напад в межах логічних з'єднань, атаки на рівні програми - напад з використанням протоколів прикладного рівня [3] та атака на рівні сервісу - напад з використанням особливостей певної програми чи послуги [6].

Зрозуміло, що чим вищий рівень атаки, тим менший рівень обслуговування він вражає. Відповідно, атака на фізичному або мережному рівні може вивести з ладу всю корпоративну мережу, атака на рівні програми - вебсервер з усіма розміщеними вебресурсами, а атака на рівні служби лише заблокує використання певної послуги, наприклад певного вебресурсу.

За впливом на жертву атаки можна розділити на напади, що блокують доступ - результатом є блокування ділових відносин із клієнтами, напади, що збільшують споживання ресурсів - не блокує доступ повністю, а лише значно збільшує споживання ресурсів, напади, які ведуть до знищення - атака руйнує компоненти системи - втрата даних через переповнення жорсткого диска, перегрів та несправність обладнання.

Блокуючі атаки можна розділити на напади з можливістю відновлення - підключення клієнта стане можливим, як тільки атака припиниться, і напади без можливості відновлення - підключення клієнта не відбудеться, як тільки атака припиниться - потрібно втручання вручну, з огляду на можливе відновлення. За типом атаки вразливість можна розділити на напади на протокол та напад на реалізацію.

1.2 Життєвий цикл DDoS-атаки

Мотиви, цілі та масштаби DDoS-атаки змінилися за останнє десятиліття. Однак основна мета атаки - заборонити користувачам мережі доступ до ресурсів - не склалася. Компоненти, що складають атаку, теж не сильно змінилися. Щоб зрозуміти життєвий цикл DDoS, важливо спочатку зрозуміти компоненти, що складають інфраструктуру атаки. Описаний тут життєвий цикл зосереджений насамперед на ботнеті або колекції зомбі-машин, що звітують про один або кілька серверів командно-адміністративного управління [14].

Розвідка. Початок DDoS-атаки характеризується ручними або автоматизованими спробами знайти вразливі хости, що виступають в ролі серверів C2 або клієнтів ботнетів. Розвідка може надходити від зловмисника у вигляді IP-зондів (їх також називають пінг-розгортками). Ці зонди можуть створити менший список хостів для подальшого зондування за допомогою сканування портів. Сканування портів надає більше інформації про хост, наприклад, пропоновані послуги та версію операційної системи. Зловмисник використовує цю інформацію для визначення найпростішого способу використання вразливості [16].

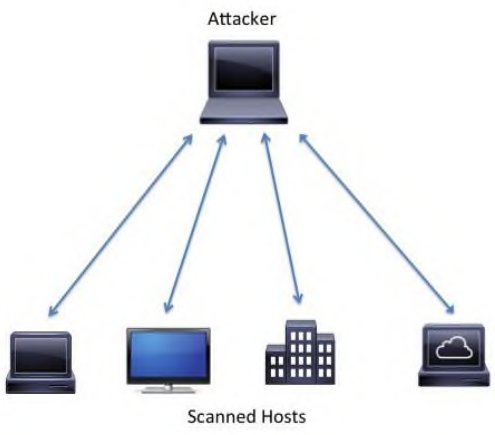


Рисунок 1.11 - Розвідка DDoS

Експлуатація та розширення. Після виявлення потенційних жертв їх націлюють на експлуатацію, щоб зловмисник міг контролювати конкретну систему. Тепер експлуатована система може бути використана, як частина інфраструктури DDoS. Залежно від потреб зловмисника, машина- жертва нападу може стати сервером, відправити DDoS-трафік чи розповсюдити експлойти на інші пристрої. З часом ботнет може розростися до тисяч, або навіть мільйонів хостів [19].

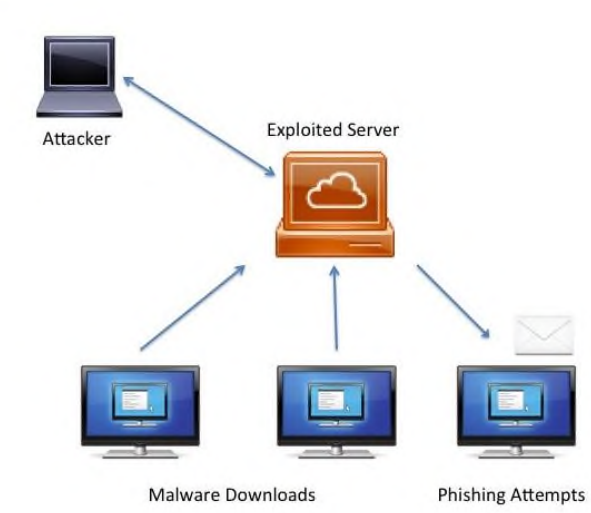


Рисунок 1.12 - Компоненти інфраструктури DDoS

Важливо зазначити, що не всі хости, які беруть участь в DDoS-атаці, є жертвами експлойту. Іноді користувачі, прихильні до політичної справи, самі охоче встановлюють програмне забезпечення DDoS, аби завдати шкоди конкретній цілі. Схожим чином, бот-мережі використовуються для інших цілей, крім DDoS-атак.

Командування та управління. Ботнети вимагають свого обслуговування. Internet Relay Chat (IRC), це форма обміну текстовими повідомленнями в реальному часі, використовує модель клієнт / сервер, і також є загальним протоколом ботнет-зв'язку. Пристрої-зомбі та сервери повинні взаємодіяти, аби надавати клієнтам інструкції, такі як хронометраж атаки або оновлення загрозливого програмного забезпечення. Модель однорангового зв'язку (P2P) важче виявити чи порушити,

оскільки з'єднання багато-до-багатьох, зменшуючи ризик порушення роботи офлайн-сервером [33].

Тестування. Ботнет досягає критичної маси, коли достатньо хостів, щоб генерувати трафік з достатньою пропускною здатністю, щоб наситити жертву. Коли ботнет досягне цієї точки, швидше за все, буде період тестування. Жертви тестування побачать великий обсяг трафіку протягом декількох секунд або хвилин. Зловмисник може оцінити ефективність атаки та внести корективи до створення стійкої атаки. Часто трафік при тривалій атаці з часом змінюється, і зловмисник перевіряє ці зміни, щоб максимізувати вплив на жертву [30].

Стійкий напад. Зловмисник визначає, коли доручити клієнтам ботнету почати надсилати трафік до цільової інфраструктури. Основна частина DDoS-атаки може тривати від годин до тижнів, залежно від мотивів зловмисника. Атаки рівня 7 стають все більш популярними, і вони здебільшого відбуваються у формі HTTP GET повені, SSL GET повені і HTTP POST повені. Посилення атак посилення зростає [29].

1.3 Особливості DDoS атак

Використовуючи Wi-Fi, об’єктами даних атак можливі як точки доступу, так і їх клієнти. За наявності великої кількості керованих атакуючою стороною пристроїв біля обраного в якості жертви пристрою (або біля ТД, до якої він під’єднаний) - можливе проведення DDoS-атаки. Дані атаки характеризуються масовим виконанням одних дій, які призводять до вичерпання певного ресурсу жертви.

Типові DoS-атаки [14]:

- ICMP Flood
- UDP Flood
- SYN Flood

- Tear Drop
- Smurf
- Deauth (деавтентифікації)
- Глушіння (Jamming)

Розглянемо детальніше атаку деавтентифікації клієнтів. Ця атака вперше була продемонстрована як частина атаки на протокол WAP на конференції PacSec 2008, Eric Tews виступив з темою «Gone in 900 Seconds, Some Crypto Issues with WPA». З того часу з'явилось декілька дуже популярних утиліт на ПЕС для її проведення, скористатись якими може будь-хто, не маючи хорошого технічного підґрунтя. Ось деякі з них:

- Aireplay-ng
- WiFite
- Airgeddon
- Scapy
- Zulu

Мета порушення безпеки КС не може відмовити в підтримці атаки, і вона може бути спрямована на отримання проміжного результату, необхідного для подальшої реалізації загрози. У разі такої невідповідності порушення безпеки КС розглядається як етап підготовки до реалізації загрози. Результатом порушення безпеки є наслідки, що сприяють здійсненню атаки [13].

Ці загрози впливають на КС та його складові, які забезпечують передачу інформації відповідно до функціональних характеристик кожної системи асоціацій. Загальна структура КС представлена на рис. 1.13

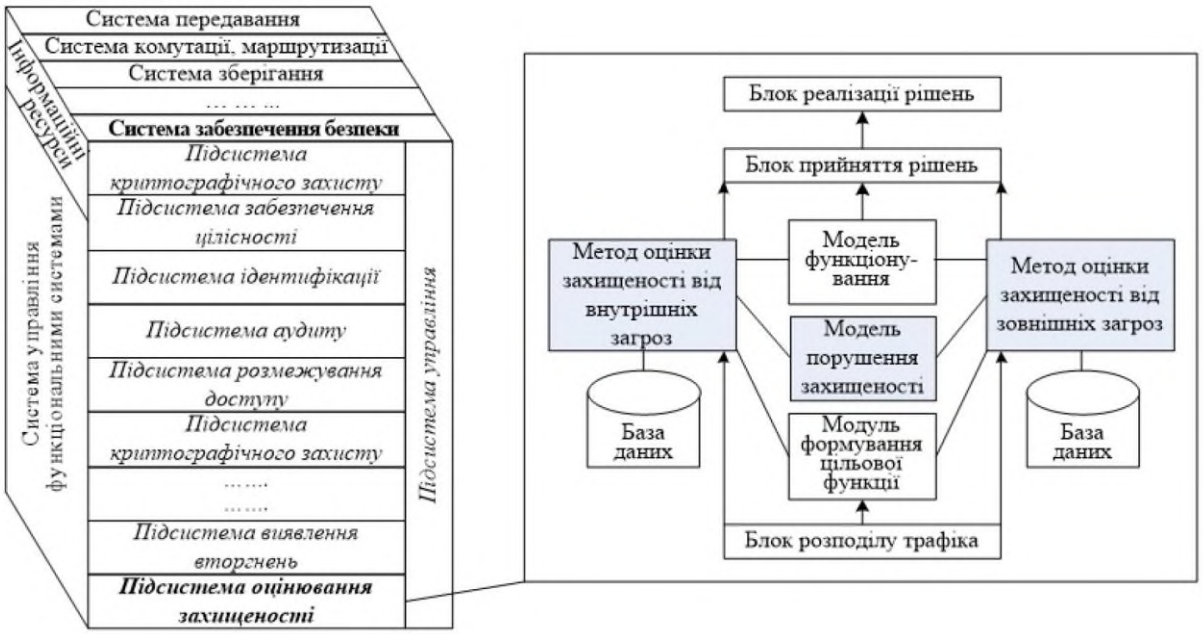


Рисунок 1.13 - Структура КС

З рис. 1.13 показано, що КС складається з багатьох взаємопов'язаних функціональних підсистем, серед яких СЗІ. Загалом можна зазначити, що для ефективного функціонування СЗБ підсистема оцінки безпеки повинна проводити власну оцінку, засновану на даних про реалізацію основних видів атак, спрямованих на систему ззовні (зовнішня) або система елементи вже посередині (внутрішні) та з урахуванням набору стратегій порушення його безпеки.

Загрози реалізовані на всіх рівнях мережевої моделі OSI і можуть впливати на об'єкти КС ззовні (потік даних, вузол мережі, кінцевий пристрій), а також зсередини (трафік даних, програмне та апаратне забезпечення КС).

Вторгнення реалізуються різними способами (вплив загрози на один або кілька об'єктів; безліч загроз на один об'єкт або кілька об'єктів). Ці методи спрямовані на досягнення проміжної або кінцевої мети, результатом якої є: відмова в обслуговуванні, віддалений моніторинг, блокування або захоплення частини системи або КС в цілому [5].

Розглядаючи практичну реалізацію порушень або атак на інформацію,

програмне та апаратне забезпечення КС, слід зазначити, що об’єктами атак є правила та технічні процедури, які підключають та обмінюються даними на КС і належать до різних рівнів мережі OSI. модель. Можуть існувати такі типи впливу атак на різних рівнях мережевої моделі OSI [27]:

- рівень застосунків - відмова у доступі до програм, отримання або зміна пріоритету обслуговування певних видів трафіку, відмова в обслуговуванні, відмова в обслуговуванні, порушення мережевого з'єднання;

- транспортний рівень - порушення надходження великих пакетів даних, створення підроблених пакетів, переповнення буфера, перебої у роботі служб із частим надсиланням звернень, надсилання великої кількості апеляційних пакетів;

- рівень мережі - порушення доставки повідомлень, порушення маршрутизації, відмова в обслуговуванні певного класу трафіку, відправка помилкових повідомлень, атака ICMP-запитами, підробка адрес;

- рівень каналу - порушення синхронізації, відмова в доступі, відмова в обслуговуванні, заміна MAC-адреси, незалежний розподіл даних;

- фізичний рівень - відмова в обслуговуванні, вимкнення, шум, відмова у перетворенні сигналу, перехоплення та прослуховування.

Швидкий розвиток інформаційних технологій обмежує можливість стрімкої адаптації існуючих на теперішній час засобів захисту до нових загроз, і взаємодію елементів систем захисту з компонентами інфраструктури КС, що значно розширює можливості їх впливу на КС та системи інформаційної безпеки [14], підкреслює актуальність питань моделювання порушення безпеки КС

1.4 Сучасні тенденції захисту від DDoS-атак

Для належної підготовки до захисту мережевої інфраструктури від DDoS-атак надзвичайно важливо якомога швидше знати, що в мережі відбувається аномальна поведінка, зловмисна чи інша. Попереднє усвідомлення зловмисної чи

підлої поведінки та інших випадків у мережі допоможе звести до мінімуму будь-які простої, що впливають на дані мережі, ресурси та кінцевих користувачів.

Далі наводиться частковий перелік доступних інструментів та технологій - деякі з яких, ймовірно, вже присутні в мережі - для допомоги у виявленні, ідентифікації та подальшій класифікації аномальних мережних подій. Ці інструменти та технології допоможуть зосередитися на показниках компромісу.

Виклик користувача / клієнта. Незалежно від особливостей сценарію, ми хочемо перешкодити кінцевому користувачеві повідомити нам про проблему. Незважаючи на те, що звернення від кінцевих користувачів іноді ми вперше дізнаємось про проблему мережі, ми волієм попереджувати попередження про проблему раніше, ніж користувачі її виявлять. Баланс нашого списку допоможе нам це зробити.

Виявлення аномалії. Як і у випадку з багатьма з цих методів, нам потрібні встановлені базові показники для роботи мережі. Вони можуть включати, але не обмежуючись цим, використання пропускної спроможності, використання центрального процесора пристрою і поломки типу трафіку. Просто неможливо виявити зміни в базовій лінії мережі, якщо ми не встановили ці базові лінії.

Мережі та пристрої з підтримкою мережі постійно створюють трафік. Однак цей трафік дотримується певних шаблонів відповідно до програми та поведінки користувачів. Аналіз цих закономірностей дозволяє нам побачити, що *не* є нормальним. Головне - збирати інформацію про рух (NetFlow) та обчислювати різні статистичні дані для порівняння з базовим рівнем. Потім відхилення від норми аналізуються більш докладно.

Cisco IOS NetFlow - це форма мережної телеметрії, яку маршрутизатори та комутатори Cisco можуть збирати локально [16].

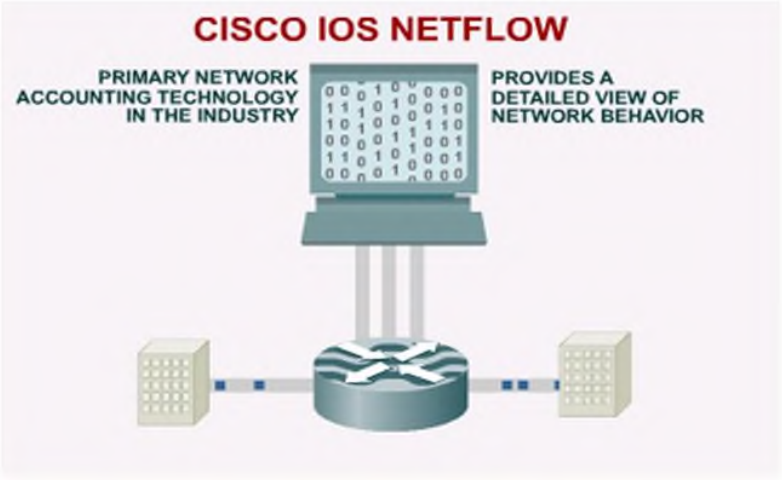


Рисунок 1.14 - **Cisco IOS NetFlow**

Дані, що надходять через NetFlow, подібні до інформації в телефонному рахунку. Користувач може переглядати, хто розмовляє (IP-адреса джерела та пункту призначення) та скільки триває розмова (кількість трафіку в байтах та пакетах).

На рис. 1.15 висвітлено сім ключових параметрів (як використовуються у версії 5 NetFlow), які перевіряються в кожному пакеті, щоб визначити, чи слід створювати новий потік. Якщо якесь із семи полів відрізняється від потоків, які були створені раніше, створюється новий потік, який додається до кешу NetFlow. Сім полів є такими:

- IP-адреса джерела
- IP-адреса призначення
- Протокол 3 рівня
- Байт TOS
- Порт джерела
- Порт призначення
- Вхідний інтерфейс

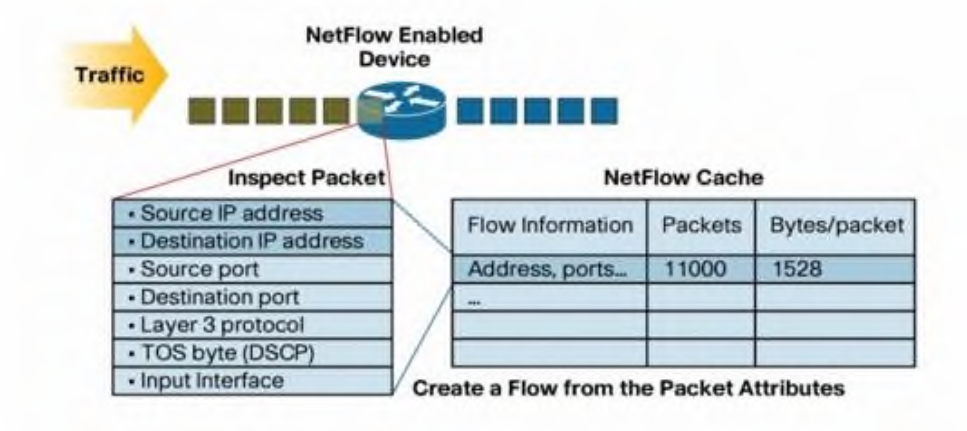


Рисунок 1.15 - Ключові параметри NetFlow

Дані NetFlow можна експортувати з мережних пристроїв до різноманітних інструментів з відкритим кодом та комерційних інструментів NetFlow Collection. Рішення захисту від кіберзагроз Cisco є ефективним методом збору та аналізу даних NetFlow [15]. Захист від кіберзагрози об'єднує роботу Cisco та Lancope для швидкого та ефективного виявлення аномальної поведінки в мережі та надання розуміння того, як можна вирішити деякі з цих питань.

Проблема запобігання DDoS-атакам полягає в природі трафіку та природі "атаки", оскільки найчастіше трафік є законним, як визначено протоколом. Отже, не існує прямолінійного підходу або методу фільтрації або блокування порушувального трафіку. Крім того, також слід розуміти різницю між об'ємним трафіком і трафіком на рівні програми.

При об'ємних атаках використовується збільшений слід атаки, який намагається перевантажити ціль. Цей трафік може бути специфічним для програми, але найчастіше це просто випадковий трафік, що надсилається з високою інтенсивністю для надмірного використання наявних ресурсів цілі. Об'ємні атаки зазвичай використовують бот-мережі для посилення сліду атаки. Додатковими прикладами об'ємних атак є атаки посилення DNS та SYN потоку [11].

Атаки на рівні додатків використовують конкретні програми або служби цільової системи. Зазвичай вони бомбардують протокол та порт, який використовує

певна служба, щоб зробити послугу марною. Найчастіше ці атаки мають своєю ціллю загальнодоступні служби та порти, такі, наприклад, як HTTP (TCP-порт 80) чи DNS (TCP / UDP-порт 53). .

Пристрої з підтримкою стану. Пристрої з підтримкою стану не забезпечують повного покриття та пом'якшення для DDoS-атак через їх здатність контролювати стан з'єднання та підтримувати таблицю стану. Зберігання такої інформації вимагає значного процесора та пам'яті. Коли бомбардується припливом трафіку, пристрій, що перебуває у стані, витрачає більшість, якщо не всі, свої ресурси, відстежуючи стани та додаткові деталі, орієнтовані на підключення. Ці зусилля часто призводять до того, що пристрій, що перебуває у стані, стає "точкою задухи" або піддається атаці.

Типовими пристроями перевірки стану та їх роллю у зменшенні загрози є брандмауери, пристрої IDS / IPS, балансири навантаження та брандмауери веб-додатків.

Брандмауери є найпоширенішими засобами перевірки стану в сучасному арсеналі пом'якшення загроз. У рішеннях брандмауера з виправленням є компонент, широко відомий як механізм перевірки пакетних пакетів (SPI). Це також називається DPI (глибока перевірка пакетів). Цей механізм забезпечує інтелект, вивчаючи потік пакетів, щоб визначити та визначити інформацію про з'єднання та деталі на рівні програми [27].

Пристрої IDS / IPS часто розгортаються в ядрі мережі та / або на краю і забезпечують інтелектуальні можливості прийняття рішень, використовуючи DPI для аналізу та пом'якшення масиву атак та загроз. Більше того, DPI дозволяє пристрою IDS / IPS реагувати на мережні події та трафік в режимі реального часу, забезпечуючи попередження або вбудовані пом'якшення

Балансири навантаження використовують SPI для прийняття рішень на основі з'єднань, які перетинають функцію балансування навантаження.

Брандмауери веб-додатків використовують SPI для оцінки веб-потоків програм, таких як звернення GET. Докладніше про SPI у брандмауерах веб-додатків див.

Методи фільтрації маршрутів [16]. Віддалено спрацьована фільтрація чорних дір (RTBH) може скинути небажаний трафік, перш ніж потрапити в захищену мережу.

Переадресація зворотного шляху одноадресної передачі [32]. Мережеві адміністратори можуть використовувати одноадресну переадресацію зворотного шляху (uRPF), щоб допомогти обмежити зловмисні потоки трафіку, що відбуваються в мережі, як це часто буває при DDoS-атаках.

URPF захищає від підробки IP, забезпечуючи, що всі пакети мають вихідну IP-адресу, яка відповідає правильному вихідному інтерфейсу відповідно до таблиці маршрутизації [32]. Зазвичай пристрій захисту перевіряє лише адресу призначення, коли визначає, куди переслати пакет. uRPF доручає приладу безпеки також звертати увагу на адресу джерела.

Географічна дисперсія (Anycast Global Resources) [33]/ Нове рішення для пом'якшення DDoS-атак розріджує ефекти атак, розподіляючи відбиток DDoS-атак, щоб цілі не були індивідуально насичені обсягом трафіку атак. Це рішення використовує концепцію маршрутизації, відому як Anycast.

Жорсткі обмеження та час очікування [8]. Заходи боротьби з недопущенням, такі як обмеження з'єднань та примусове встановлення часу очікування в мережевому середовищі, спрямовані на те, щоб DDoS-атаки не запускалися та не поширювалися зсередини мережі ні навмисно, ні ненавмисно.

Блокування на основі репутації. Блокування на основі репутації стало важливим компонентом сучасного арсеналу веб-фільтрації. Типовою тенденцією зловмисного програмного забезпечення, активності ботнетів та інших веб-загроз є надання URL-адреси, яку користувачі повинні відвідати для досягнення

компромісу. Найчастіше такі методи, як спам, віруси та фішингові атаки, спрямовують користувачів на шкідливу URL-адресу [6].

Технологія, що базується на репутації, забезпечує аналіз URL-адрес та встановлює репутацію кожної URL-адреси. Технологія репутації має два аспекти. Аспект розвідки поєднує в собі телеметрію загроз по всьому світу, інженерів розвідки та аналітику / моделювання.

Списки контролю доступу [17]. ACL надають гнучку можливість для різноманітних загроз безпеці та використання, включаючи DDoS. ACL забезпечують нульове або реактивне пом'якшення для DDoS-атак, а також пом'якшення першого рівня для атак на рівні додатків. ACL - це впорядкований набір правил, що фільтрує трафік. Кожне правило задає набір умов, яким повинен відповідати пакет, щоб відповідати правилу.

ACL часто використовуються для захисту мереж та певних хостів від непотрібного або небажаного трафіку за допомогою фільтрації протоколів / портів, хоча фільтрація також може базуватися на параметрах TCP та прапорах. Наприклад, ACL можуть заборонити HTTP-трафік із мережі високого рівня захисту до Інтернету. Ви також можете використовувати ACL, щоб дозволити HTTP-трафік лише на певні сайти, використовуючи IP-адресу сайту, щоб ідентифікувати його в IP-адресі ACL [34].

DDoS Run Books [35]. Передумовою книги DDoS є просто надання "посібника" для організації у випадку виникнення DDoS-атаки. По суті, посібник забезпечує управління кризисними ситуаціями (більш відомий як план реагування на аварії) у випадку атаки DDoS.

Відповіді вручну на DDoS-атаки [30]. Нічого не варте того, що ручні відповіді на DDoS-атаки зосереджуються на заходах та рішеннях, які базуються на деталях, які адміністратори дізнаються про атаку.

Процес реагування часто залишається поза увагою. Як згадується в DDoS Run Books , організації часто не мають процесу або плану, і, отже, покладаються

виключно на ручні відповіді. Проактивні рішення та постійний моніторинг та оновлення конфігурації повинні бути загальноприйнятою практикою, а ручні відповіді вважаються рідкісними рішеннями.

Очищення та перенаправлення дорожнього руху [12]. Через поширеність DDoS-атак в останні роки, численні організації та компанії зараз надають захист DDoS як послугу. Хоча існують різні способи досягти захисту від DDoS та пом'якшення атак, більшість постачальників пропонують вбудоване рішення, за допомогою якого трафік організації може надсилатися до або через сервісну установу.

Рішення AT&T Internet Protect: розподілене відмову в обслуговуванні призначене для клієнтів AT&T, які шукають захист від DDoS. Оскільки AT&T вже управляє мережею, через яку проходить трафік клієнта, AT&T використовує свою експертизу та інтелектуальні рішення в магістралі для фільтрації будь-якого зловмисного або необдуманого трафіку до того, як потрапить у середовище клієнта. Крім того, захисне рішення аналізує мережевий потік. Служба оборони Verizon DoS працює так само, як і раніше обговорені, тим, що вона контролює трафік і маршрутизує трафік через середовище Verizon для очищення, дозволяючи хороший трафік повертати назад до захищеного середовища клієнта [21].

Рішення Arbor Networks Pravail System Availability Protection System (APS) є прикладом рішення на місці (на місці). Пристрій знаходиться в рядку в середовищі клієнта і має зв’язок назад із серверною базою Arbor. Ця служба включає розвідувальні дані та інформацію, отриману від команди інженерії безпеки Arbor (ASERT). У поєднанні з такими методами, як базове виділення та виявлення аномалій, Arbor APS є видатним рішенням DDoS.

1.5 Постановка задачі

У першому розділі розглянуті різні типи атак, їх категорії та прийоми, які вони використовують. Він представив класичні та сучасні методології ідентифікації, класифікації та пом'якшення DDoS-атак. Оскільки мережі різняться, ми не прагнемо надати всеохоплюючий документ щодо пом'якшення DDoS, який застосовується до кожної організації, але ми спробували описати інструменти, доступні для боротьби з DDoS-атаками.

Використання шестифазної моделі пом’якшення DDoS від Cisco є гарним початком, і його також можна постійно переглядати при створенні обґрунтованої політики DDoS. Підготовка є ключовою частиною будь-якої стратегії DDoS. Переконайтесь, що інструменти, які будуть використовуватися для ідентифікації DDoS, перевірені, функціонують та працюють у відповідних місцях, а також, що мережевий персонал навчений та здатний експлуатувати необхідні інструменти для ідентифікації DDoS.

В магістерській роботі необхідно вирішити наступні завдання:

1. Провести аналіз сучасний стан DDoS атак, та аналіз нинішнього стану технологій для вирішення проблем захисту інформації

2. Розробити математичну модель атак з врахуванням опису сезонності мережного навантаження.

3 Вдосконалити метод раннього виявлення та протидії DDoS-атакам малої інтенсивності.

5. Розробити алгоритм визначення точок початку атаки та алгоритм поділу змішаного трафіку на надійний та загрозливий, який враховує сезонну кількість мережного навантаження.

6. Запропоновати критерії успішності, які дозволяють оцінювати успішність роботи алгоритму роботи, і сторонні засоби за фільтрацією трафіку.

7. Описати інформаційну систему та розроблено її функціональну модель.

8. Спроєктувати програмний засіб для виявлення початку атаки та подальшого виявлення та блокування нелегітимних звернень. Його особливістю є модульність та універсальність для забезпечення безпеки різних мережних ресурсів та захисту їх від атак різного типу.

9. Дослідити ефективність та результативність розробленого програмного забезпечення для протидії DDoS-атакам та їх виявлення.

2 МАТЕМАТИЧНА МОДЕЛЬ ВИЗНАЧЕННЯ ПОЧАТКУ АТАКИ

2.1 Аналіз аномалій станів атакованої системи

Атака HTTP Get Flood відома як найпоширеніша DDOS-атака на прикладний рівень з частотою 21% у всіх атаках. Оскільки величезна кількість звернень надсилається на Веб-сервер для отримання сторінок, а також обсяг відповідей, виданих сервером, набагато більше, ніж обсяг, отриманий зомбі в такому вигляді атаки, отже, це може бути зроблено за допомогою невеликих ботнетів; з іншого боку, оскільки кожен зомбі намагається видати запит, використовуючи його справжню адресу, виконує всі етапи триступеневих рукостискань, а контекст звернень повністю відповідає протоколу HTTP, методи підробки виявлення адрес та виявлення аномалійв тексті не може бути використаний. Механізми, які використовуються для боротьби з цією атакою, не тільки мають великі перевантаження процесора, але також можуть спричинити два типи «помилково негативного» (неправильно усвідомити фальшивий трафік як реальний трафік) та «хибно позитивного» (неправильно зрозуміти реальний трафік як помилковий трафік).

Підхід, заснований на аналізі аномалій, який веде до порівняння поточного стану системи з нормальним станом, буде оптимальним для виявлення початку атаки та подальшого виявлення загрозливого руху. Одночасно порівнюються різні властивості мережної діяльності. Ці властивості контексту DDoS-атак можуть включати: тип та кількість пакетів, кількість пакетів певного протоколу, IP-адресу джерела, час і швидкість запитів тощо.

Нехай $A(a_1, a_2, a_3, ..., a_n)$ - множина всіх можливих характеристик для всіх мережних клієнтів. Вектор $B(b_1, b_2, b_3, ..., b_m)$ - множина допустимих клієнтів певного мережного ресурсу. Перетин цих підмножин характеризує клієнтів мережного ресурсу, за якими вони можуть бути класифіковані. Так само нелегальні

клієнти отримають свій персональний набір властивостей, за яким вони також можуть бути класифіковані.

Атака типу HTTP GET flood використовується нападниками для атаки веб-серверів та серверів вебзастосунківв. Атака - це сукупність на перший погляд законних звернень GET або POST до сервера. В результаті вони можуть призвести до стану відмови в обслуговуванні, без необхідності переповнювати канал великим обсягом трафіку. Такі звернення у випадку розподіленої атаки DoS надсилаються з десятків тисяч інфікованих (зомбі) вузлів. На рис. 2.1 схематично показує послідовність пакетів у запиті HTTP GET після з'єднання TCP.

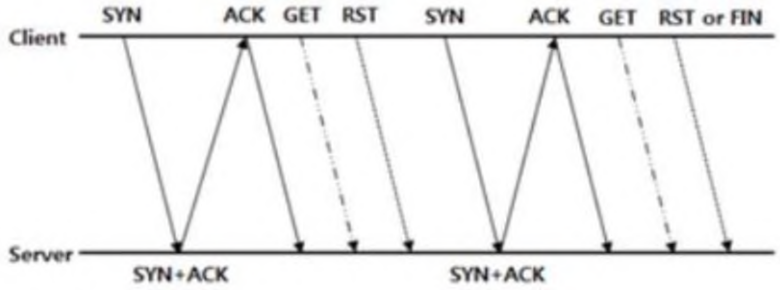


Рисунок 2.1- Послідовність пакетів при атаці типу HTTP GET

В процесі атаки зловмисник постійно відправляє звернення, створюючи при цьому нові TCP з’єднання. Останнім часом [8] також стали розповсюдженими HTTP GET flood атаки в рамках одного TCP з’єднання (див. рис. 2.2). Цей тип атаки не можливо виявити методом оцінки кількості SYN звернень.

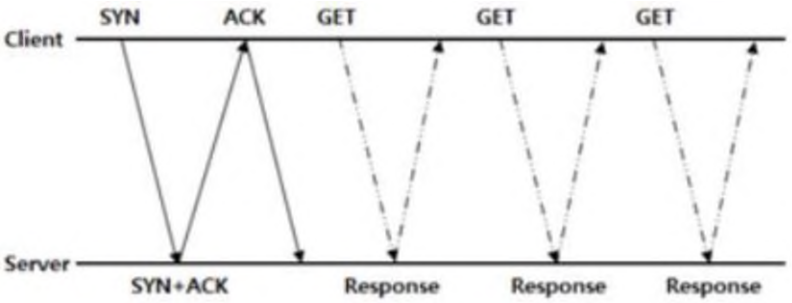


Рисунок 2.2 - Послідовність пакетів при атаці типу HTTP GET в рамках одного TCP з’єднання

Зараз атака HTTP-потоків є однією з найдосконаліших загроз інформаційній безпеці, яка безпосередньо не пов'язана із вразливістю програмного забезпечення. Для обладнання безпеки відрізнити зловмисні HTTP-звернення від законних надзвичайно складно, а неправильні методи або налаштування призводять до великої кількості помилкових спрацьовувань. Використання метрик, заснованих лише на оцінці інтенсивності запиту, не є оптимальним методом виявлення DDoS-атак, таких як HTTP flood, оскільки обсяг трафіку може бути нижче порогового. Тому доцільно використовувати багатокритеріальний метод виявлення DDoS-атак з показниками, які залежать від інтенсивності звернень, і тими, які не залежать від цього показника.

MapReduce - модель проведення розподіленої паралельної обробки великих об’ємів даних з використанням кластерів (великої кількості обчислюваних блоків). Робота моделі MapReduce складається із двох етапів: Map і Reduce [37].

Для роботи багатокритеріального методу виявлення DDoS-атак необхідно провести попередній аналіз та розрахунки: визначити показники (критерії), за якими буде виявлено наявність або відсутність атаки; побудувати модель для звичайного мережевого трафіку; встановити пороги для вибраних показників.

В якості критеріїв оцінки були обрані наступні показники:

- рівень завантаження процесора сервера;
- обсяг зайнятої оперативної пам'яті сервера;
- розмір упаковки;
- поточний рівень трафіку (Мбіт / с);
- розподіл значення адреси джерела звернень (source ip);
- користувач-агент у запиті;
- URI (ієрархічна частина та фрагменти URL-адреси звернення);

Наявність атаки потоку HTTP GET flood може характеризуватися кількістю звернень від джерела за секунду (GET запит за секунду, GRPS) [8]. Зрозуміло, що

законний користувач не постійно робить велику кількість звернень на один і той же ресурс, як це може вузол, керований зловмисником (зомбі). Тому деякий час At надходить з IP-адреси джерела x надходить s. звернень.

Таким чином, завдання виявлення зловмисних звернень у контексті цієї роботи полягає в їх класифікації на основі властивостей мережної діяльності.

Для цього необхідно правильно визначити початкову точку атаки. Це дозволить класифікувати весь попередній трафік як законний та відкриє додаткові можливості для розподілу змішаного трафіку, що надходить після атаки, на законний та загрозливий. У цьому випадку метод виявлення загрозливого трафіку, у першому наближенні, буде зведений до наступних етапів:

1. Визначте поточні сезонні періоди.

2. Беручи до уваги сезонність, визначте початкову точку нападу.

3. Ми відносимо весь попередній трафік до початку атаки до законного.

4. Класифікуємо змішаний трафік на законний та загрозливий.

5. Порівняйте законний трафік, вибраний із змішаного, з трафіком, отриманим до атаки.

6. На основі результатів, отриманих на попередньому кроці, та розроблених критеріїв успіху, скоригуємо вибірки.

7. Весь вхідний трафік аналізується на основі отриманих даних.

2.2 Модель визначення початку атаки з урахуванням сезонності

Для протидії розподіленим атакам, спрямованим на відмову у обслуговуванні, потрібно виконання двох таких завдань [3].

1. Діагностувати DDOS-атаку на самих початкових стадіях. Чим раніше буде виявлена DDOS-атака, тим раніше зможе включитися у гру мережний адміністратор та тим раніше можна буде розпочати проводити анти DDOS-захід. Крім того, при виявленні DDOS-атаки можна буде, не чекаючи реагування

адміністратора, автоматично запустити заходи щодо протидії: задіяти резервні канали зв'язку, включити фільтри і т. д.

2. Друге завдання пов'язане з поділом загального потоку трафіку на загрозливий і звичайний. Зрозумівши, які з клієнтських звернень є результатом DDOS-атаки, можна буде створити відповідне правило для брандмауера, правила для маршрутизатора або ж, в випадку масштабної атаки, перенаправити ці дані на вищі маршрутизатори.

Перша з цих задач є досить новою. Кілька років тому основним було завдання по «сортуванні» трафіку. Однак зловмисники постійно вдосконалюють способи проведення атак такого типу. І сучасні атаки відрізняються складністю і наявністю етапу подготовки. Під час підготовчого етапу зловмисник намагається виявити найбільш уразливі для атаки місця. Наприклад, для вебсервера такими місцями можуть бути певні скрипти, які здійснюють велику кількість звернень до бази даних або надмірно використовують процесорний час. Знайшовши вразливе місце, зловмисник зможе паралізувати роботу сервера, використовуючи бот-мережу меншого розміру. З іншого боку, якщо діагностувати атаку вдасться вже на цьому етапі, можна буде задіяти автоматичні засоби запобігання атаки, а у системного адміністратора буде час підготуватися - оптимізувати скрипти, надмірно завантажують ресурси комп’ютера, створити фільтри і т.д.

Для виявлення DDOS-атак і створення спеціальних фільтрів для відсікання загрозливого трафіку застосовуються різноманітні методи і підходи.

Серед основних методів можна виділити методи, що базуються на статистичному аналізі. Це кількісний аналіз, аналіз середньоквадратичних відхилень, кластерний аналіз і т.д. Всі ці види аналізу можуть оцінювати різні параметри мережевої активності і діагностувати початок атаки або визначати загрозливий трафік.

Основними параметрами, за якими проводиться аналіз, можуть бути:

• Кількість звернень за певний період.

• Кількість звернень з певного джерела або з певною мережі.

• Кількість звернень до певного пункту призначення (для вебсервера це конкретний скрипт).

• Швидкість надходження звернень.

• Час між запитами.

• Інші різні параметри мережної активності.

За допомогою середньоквадратичного відхилення можна розрахувати допустиму межу для одного з параметрів мережевої активності, наприклад, для кількості звернень за якийсь період часу. У разі якщо межа буде порушена, це стане свідченням початку атаки. Так як в різний час навантаження на мережевий ресурс, так само може бути різною, то для раннього виявлення атаки необхідний постійний моніторинг і перерахунок кордонів для кожного тимчасового кроку. Постійний моніторинг дозволить визначити атаку, якщо вона почнеться в період невеликої мережевої активності, або, якщо зловмисник шукає потенційно вразливі місця на сервері, проводячи міні- DDOS-атаки і вивчаючи поведінки сервера. У разі якщо верхня межа задана строго і зловмисник проводить міні-атаки в період найменшої мережевої активності, він може не порушувати задану кордон, і його дії будуть не виявлені. Атака буде виявлена тоді, коли зловмисник знайде потенційно вразливе місце, і зробить на нього атаку. Постійний моніторинг активності і перерахунок допустимих меж дозволяє цього уникнути. У період меншою мережевий активності верхня межа знизиться. Однак і цей метод має ряд мінусів.

По-перше, зловмисник може почати атаку поступово. Показники активності на кожному кроці будуть плавно підвищуватися, але при цьому не будуть порушувати кордонів. Так як при розрахунку середньоквадратичного відхилення використовуються останні п інтервалів, в тому числі і ті, які вже містять дані атаки, то зловмисник, поступово збільшуючи інтенсивність атаки, буде відсовувати кордон.

По-друге, вибір розміру періоду *n* для розрахунку середньоквадратичного відхилення, не є однозначним.

Якщо *n* буде занадто велике, отримана межа буде занадто високо, якщо використовується мале значення *n*, то можливі часті спрацьовування. Вибрати ж оптимально значення *n* у цій ситуації буде неможливо, так як будь-яке його значення може захоплювати два різних періоду. Наприклад, навіть мале значення *n*, що розраховується на початку робочого дня, буде захоплювати дані ще й нічного періоду, пов'язаного із низькою активністю, і денного інтервалу, який характеризується більшим навантаженням на мережні ресурси.

Для того щоб запобігти помилковому спрацьовуванні, пов'язане з початком робочого дня, потрібно використовувати таке значення *n*, в якому будуть дані за кілька днів. Або контролювати відразу кілька тимчасових періодів - при спрацьовуванні на хвилинних інтервалах, розглянути часові або добові періоди. Але це в свою чергу призведе до зменшення точності, і атака буде визначена з запізненням [14].

В якості гіпотези можна припустити, що більш високу точність в цих випадках може дати облік різних періодів активності і порівняння схожих між собою періодів.

Для вирішення описаних проблем необхідно зробити змінну оцінку, яка буде характеризувати точну активність нитки. На етапі оцінки встановлюють динамічний кордон, відповідний періоду наймолодшої атаки. Для формування ковзаючої оцінки використаєм середньоквадратичне відхилення:

$$\sigma = \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} \left(x_i - \bar{x} \right)^2}, \qquad (2.1)$$

де σ - середньоквадратичне відхилення; *n* - кількість часових діапазанів; x_i - кількість звернень за певний період; $\bar{x}$ - середнє арифметичне звернень по всіх періодах.

Нехай x_i, кількість звернень до сервера за одну годину. Сервер отримує стабільне добове навантаження. Кількість добових інтервалів n. Тоді звернення до сервера запишемо у вигляді матриці:

$$x_{11},\ x_{12},\ x_{13}, \dots, x_{124}$$
$$x_{21},\ x_{22},\ x_{23}, \dots, x_{224}$$
$$\dots\dots\dots\dots\dots\dots\dots\dots$$
$$x_{n1},\ x_{n2},\ x_{n3}, \dots, x_{n24}$$

Кожен рядок матриці містить добові дані про кількість звернень. Перший рядок містить дані поточної доби, томі він може бути заповнений не до кінця. Розрахунок середньоквадратичного відхилення в цьому випадку може проводитися двома способами:

1. Звичайним способом - з урахуванням певного числа останніх значень, наприклад, так:

$$x_{21},\ x_{22},\ x_{23}, \dots, x_{224}, x_{11},\ x_{12},\ x_{13}, x_{14}$$

Значення беруться з ***рядків*** матриці.

2. З урахуванням критерію сезонності - значення беруться зі ***стовбців*** матриці.

$$x_{n1}, \dots, x_{21},\ x_{11}$$

Для розрахунку граничного значення для $i + 1$ періоду необхідно використати значення $i + 1$ стовпчика. Якщо аналіз мережного ресурсу показує, що він зазнає циклічне навантаження, то можна виключити рядки, які відповідають святковим і вихідним дням. Або наприклад використовувати рядок, який відповідає за певний період часу, наприклад з 10:00 до 11:00 для кожного понеділка.

Це дозволить підвищити точність формування границі, порушення якої буде ідентифікувати можливість початку DDOS-атаки, і відповідно скоротити час на її фіксацію.

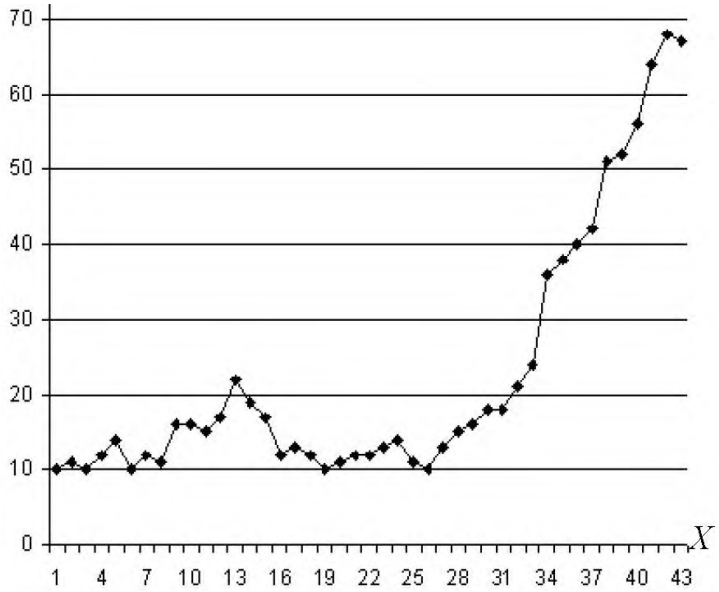


Рисунок 2.3 - Кількість звернень у період початку DDOS-атаки, 10-хвилинний інтервал

На графіку відображається кількість звернень до сервера за секунду, відповідні періоду початку DDOS-атаки. Ось X - часовий інтервал. Одну поділку відповідає 10 хв. Ось Y - кількості звернень до сервера за секунду.

Для отримання статистичних даних використовувався один з найбільших сервісів статистики Liveinternet.

Одна з найбільших в інтернеті блог-платформа, що дає кожному користувачеві можливість вести власний щоденник, який інтегрує блоги різних користувачів, а також підписку на оновлення в блогах цікавих йому людей.

Сервіс надає такі можливості:

- Додавати допис в щоденник, безпосередньо через вебінтерфейс, або за допомогою програмних клієнтів, електронною поштою або SMS.
- Коментувати дописи в щоденниках ресурсу, а також інших системах щоденників, щопідтримують OpenID.
- Додавати в свої дописи зображення, інформери (погода, ТБ-програма), голосування, завантажувати різноманітні типи файлів найбільш зручним способом, використовувати до 12 власних аватарів.

- Читати оновлення в ораних щоденниках, а також будь-яких інших RSS-каналах: через рядок друзів у Інтернеті, через підписку електронною поштою, в програмних клієнтах.
- Фрмува свої, а також брати участь в роботі вже існуючих тематичних груп.

Наприклад, якщо розглядати місячну відвідуваність одного з найбільших новинних порталів – «Українська правда» (https://www.pravda.com.ua/, то чітко прослідковується періодичність (рис. 2.4). Найбільша кількість перегляду сторінок на понеділок і низька активність користувачів на вихідних.

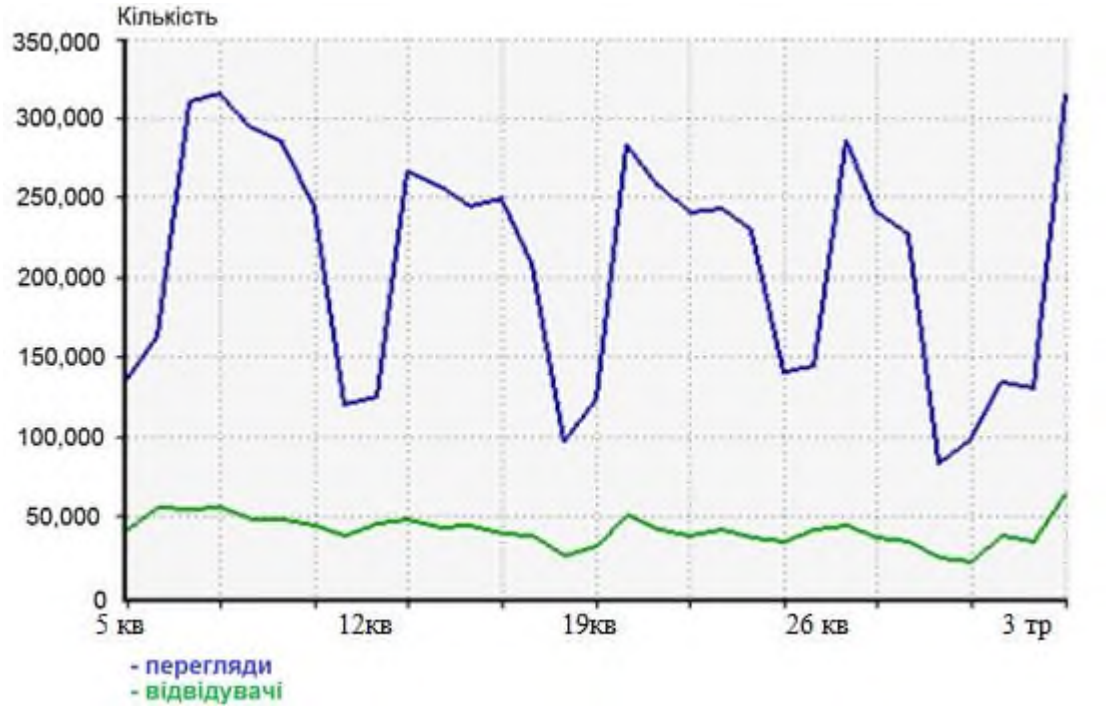


Рисунок 2.4 - Графік звернень до ресурсу «Українська правда»

На протязі доби кількість переглядів зростає з початком робочого дня і зменшується до вечора (рис. 2.5).

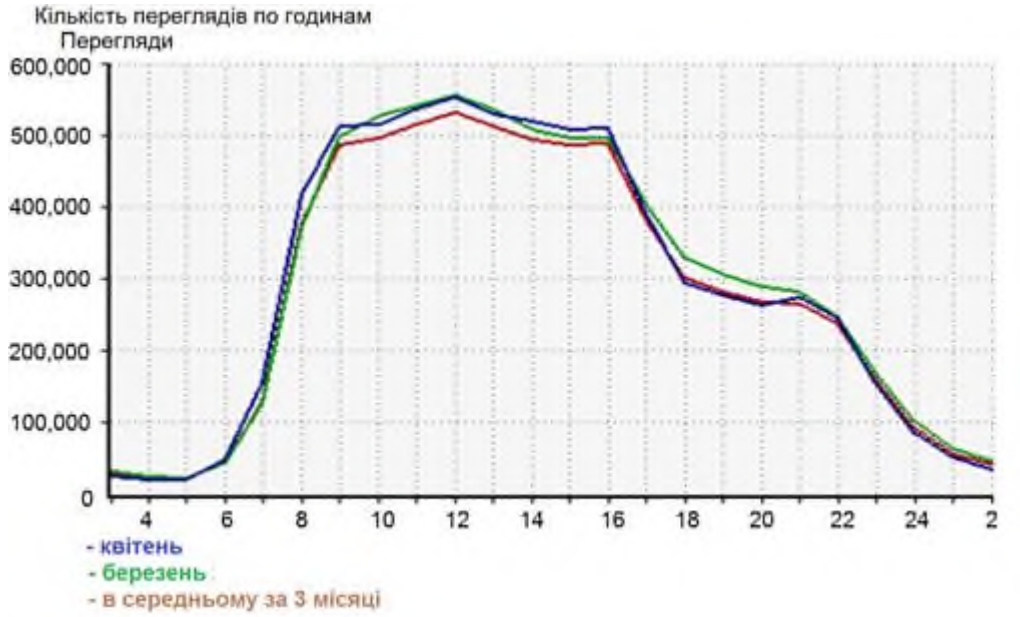


Рисунок 2.5 - Графік звернень до ресурсу «Українська правда» по годинах

Очевидно, що, основними відвідувачами таких сайтів є співробітники офісів, які отримують до них доступ в робочі дні.

Однак є ресурси, користувачі яких активні весь тиждень, незалежно від буднів та вихідних. Це можуть бути соціальні мережі або служби обміну повідомленнями.

Для моделювання процесу множини актуальних сезонних періодів був використаний метод Херста.

Ключовий параметр фрактального аналізу - показник Херста. Це міра, яку використовують при аналізі часових рядів. Чим більше затримка між двома однаковими парами значень в часі ряду, тим менше коефіцієнт Херста [17].

Цей важливий показник ввів Гарольд Едвін Херст - видатний британський гідролог, який займався проектом греблі на Нілі в Єгипті. Для будівництва потрібно оцінити приплив води та, відповідно, потреби у відтоку. Спочатку вважалося, що приплив води - величина випадкова, стохастичний процес. Однак Херст вивчив записи по розливах Нілу за дев'ять століть та знайшов в цьому явищі закономірності. Цей стало відправною точкою у дослідженні. Було виявлено,закономірність, що розливи більше середнього часто змінювалися ще

більшими розливами. Після цього процес міняв свій напрям, і розливи за рівнем менше середнього змінювали ще менші. Були в наявності цикл із неперіодичною тривалістю.

Основою статистичної моделі Херста стала робота Альберта Ейнштейна про броунівському русі, яка по суті є моделлю випадкових блукань частинки.

Для формування свого методу гідролог використовував тимчасової ряд $X1..Xn$ значень розливу річки. Далі проводився наступний алгоритм, названий в подальшому методом нормованого розмаху або R / S-аналізом:

Розрахунок середнього значення, Xm, ряду $X1..Xn$

Розрахунок стандартного відхилення ряду, S

Нормалізація ряду, шляхом вирахування з кожного значення середнього значення, Zr, де $r = 1..n$

Створення кумулятивного тимчасового ряду $Y1 = Z1 + Zr,$ де $r = 2..n$

Розрахунок розмаху кумулятивного тимчасового ряду $R = max\,(Y1..Yn) - min\,(Y1..Yn)$

Розподіл розмаху кумулятивного тимчасового ряду на стандартне відхилення S.

В загальній формі критерій Херста описується:

$$(R/S)_n = c \times n^H$$

У загальному випадку значення R / S змінює масштаб у міру збільшення приросту часу, відповідно до значення ступеня залежності, що дорівнює H, яка зазвичай і називається показником Херста.

Метод вперше був застосований Херстом при проєктуванні греблі. Гідролог прийняв показник H за 0,5, якби процес розливу був випадковим. У процесі спостережень він виявив, що H = 0.91! Виходить, що нормований розмах змінюється швидше, ніж квадратний корінь з часу, тобто система проходить більшу відстань, ніж імовірнісний процес. Даний факт був передумовою моменту, коли

можна стверджувати, що події минулого істотно впливають на сьогодення і майбутнє [17].

$r_1 \quad r_2 \quad r_3 \quad r_4 \quad r_5 \quad r_6 \quad r_7 \quad \dots \quad r_{T-3} \quad r_{T-2} \quad r_{T-1}$

$$\begin{array}{cccc} \overline{r_1^2}, S_1^2, R_1^2 & \overline{r_2^2}, S_2^2, R_2^2 & \overline{r_3^2}, S_3^2, R_3^2 & \dots \quad \overline{r_{T_2}^2}, S_{T_2}^2, R_{T_2}^2 \\ R_1^2 / S_1^2 & R_2^2 / S_2^2 & R_3^2 / S_3^2 & R_{T_2}^2 / S_{T_2}^2 \end{array} \quad \left(R/S\right)_2 = \frac{\sum_{i=1}^{T_2} R_i^2 / S_i^2}{T_2}$$

$$\begin{array}{ccc} \overline{r_1^3}, S_1^3, R_1^3 & \overline{r_2^3}, S_2^3, R_2^3 & \overline{r_{T_3}^3}, S_{T_3}^3, R_{T_3}^3 \\ R_1^3 / S_1^3 & R_2^3 / S_2^3 & R_{T_3}^3 / S_{T_3}^3 \end{array} \quad \left(R/S\right)_3 = \frac{\sum_{i=1}^{T_3} R_i^3 / S_i^3}{T_3}$$

$$\dots$$

$$\begin{array}{ccc} \overline{r_1^N}, S_1^N, R_1^N & \dots & \overline{r_{T_N}^N}, S_{T_N}^N, R_{T_N}^N \\ R_1^N / S_1^N & & R_{T_N}^N / S_{T_N}^N \end{array} \quad \left(R/S\right)_N = \frac{\sum_{i=1}^{T_N} R_i^N / S_i^N}{T_N}$$

де $\overline{r_j}$ - середнє значення, S_j^2 - дисперсія значень, $X(t,i) = \sum_{i=1}^{t}\left(r_i - \overline{r_j}\right)$, $t < j$ - адитивне відхилення, $R(j) = \max_{1 \le t \le j} X(t,i) - \min_{1 \le t \le j} X(t,i)$ -розмах, $T_n = \left[\left(T-1\right)/n\right]$ - кількість блоків, при $n = 2 \dots \left[T/2\right]$. Для кожного значення *t* формується та лінійно апроксимується графік залежності $Ln\left(R/S\right)_n$ від $Ln\,(n)$.

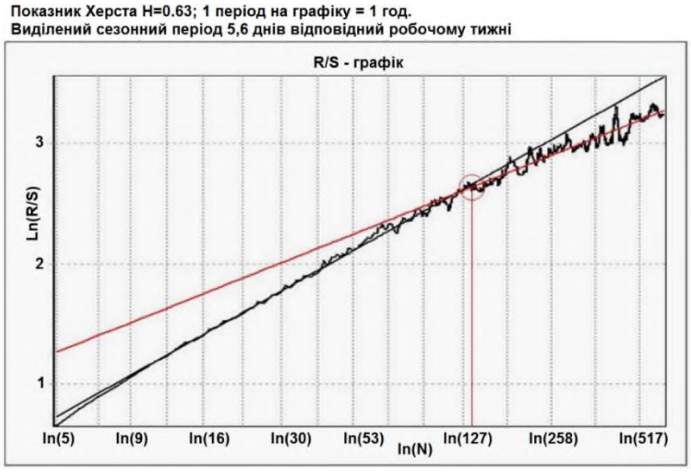


Рисунок 2.6 - Оцінка показника Херста (H)

Коефіцієнт нахилу кривої дає можливість сформувати оцінку показника Херста (H) (рис. 2.6).

Запропонований підхід дає можливість аналізувати мережний трафік і виявляти сезонну циклічність в умовах невизначеності.

2.3 Метод раннього визначення та протидії DDoS атакам на підставі кластеризації

Аналітикам часто простіше визначити групи подібних об’єктів, вивчити їх особливості та побудувати окрему модель для кожної групи, ніж створити єдину загальну модель для всіх даних. Цей прийом постійно використовується в маркетингу, виявляючи групи споживачів, покупців, товари та розробляючи окрему стратегію для кожного з них.

Для порівняння двох об’єктів необхідно мати критерій, на основі якого відбуватиметься порівняння. Як правило, цим критерієм є відстань між об'єктами.

Існує багато критеріїв відстані, розглянемо декілька з них [25]:

Евклідова відстань - найпоширеніша відстань. Це геометрична відстань у багатовимірному просторі.

Квадрат евклідової відстані. Іноді може бути бажано, щоб квадратична стандартна евклідова відстань надавала більшої ваги більш віддаленим об'єктам.

Відстань кварталів міста (відстань Манхеттена). Ця відстань - це просто середнє значення різниці координат. У більшості випадків ця міра відстані призводить до тих самих результатів, що і для звичайної евклідової відстані. Однак слід зазначити, що для цього заходу вплив деяких великих різниць (викидів) зменшується (оскільки вони не мають квадрата) [32].

Відстань Чебишев. Ця відстань може бути корисною, коли ви хочете визначити два об'єкти як "різні", якщо вони відрізняються в якійсь одній координаті (будь-якому одному вимірі) [14].

Статична відстань. Іноді бажано поступово збільшувати або зменшувати вагу, пов'язану з розміром, для якого відповідні предмети дуже різні. Цього можна досягти, використовуючи статичну відстань.

Вибір відстані (критерій подібності) повністю залежить від дослідника. При виборі різних заходів результати кластеризації можуть значно відрізнятися.

Алгоритм k-середніх [35]. Найпростіший, але в той же час досить неточний метод кластеризації в класичному здійсненні. Він ділить набір елементів векторного простору на заздалегідь визначену кількість кластерів k. Ефект алгоритму полягає в тому, що він прагне мінімізувати стандартне відхилення в точках кожного кластера. Основна ідея полягає в тому, що на кожному кроці ітерації центр маси перераховується для кожного з кластерів, одержаних на попередньому кроці, потім вектори знову розділяються на кластери відповідно до того, який із нових центрів був ближче до обраної метрики. Алгоритм закінчується, при умові, що на будь-якому кроці ітерації не відбувається зміни кластера.

Проблеми та недоліки алгоритму k-середніх:

- необхідно заздалегідь знати число кластерів. В кваліфікаційній роботі запропоновано метод визначення кількості кластерів, що базувався на пошуку кластерів, розподілених за якимось законом (в нашому випадку події зведено до

нормального закону). Після цього був застосовано класичний алгоритм k-середніх значень, який дав більш точні результати.

- Алгоритм дуже чутливий до вибору початкових центрів кластерів. Класична версія передбачає випадковий вибір кластерів, що часто було джерелом помилок. Як рішення необхідно вивчити об'єкт, щоб точніше визначити центри початкових скупчень. В нашрму випадку на початковому етапі пропонується взяти участь у якості центрів найвіддаленіші точки скупчень.

- Не справляється із завданням, коли об'єкт належить однаково до різних кластерів або не належить жодному. Вибір і обгрунтування методу кластеризації. Позначимо *T* - множина клієнтських звернень до початку атаки, множин *H* - зловмисні клієнтські звернення і множина Т * - надійних клієнтських звернень. Трафік після початку атаки, потрібно на дві групи, щоб виділити зловмисний (кластеризувати на основі алгоритму k-середніх).

Для цього обрахуємо їх центри мас,

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} \left(x_j - \mu_i \right)^2 \qquad (2.2)$$

де *k*- число кластерів, S_i - отримані кластери, i – 1,2, … , k_i, μ_i - центри мас векторів $x_j \in S_i$.

Після цього для аналізу і обробки будуть доступні три групи трафіку:

1. Передує початку атаки - *T*.

2. Виділений із змішаної - *T**.

3. Зловмисний трафік, виділений з - *H*.

Для оцінки ефективності запропонованого поділу запишемо систему рівнянь стаціонарних ймовірностей:

$$
\begin{aligned}
&p_0\lambda = p_1\mu \\
&\left(\lambda + i\mu\right) p_i = \lambda p_{i-1} + \left(i+1\right)\cdot \mu p_{i+1},\ i = 1,\ldots, K-2, \\
&\left(\lambda + \left(K-1\right)\mu\right) p_{K-1} = \lambda p_{K-2} + KN\cdot \mu p_K, \\
&\left(\lambda^* + KN\mu\right) p_K = \lambda p_{K-1} + KN\cdot \mu p_{K+1}, \\
&\left(\lambda^* + i\mu^*\right) p_i = \lambda^* p_{i-1} + \left(i+1\right)\cdot \mu^* p_{i+1},\ i = K+1,\ldots, N-1
\end{aligned}
\qquad (2.3)
$$

де, λ - інтенсивність завантаження, λ_L - інтенсивність завантаження допустимими користувачами, S - інтенсивність зловмисного трафіку, μ - інтенсивність зменшення черги звернень, μ^* - інтенсивність зменшення черги звернень при використанні фільтру, E_1, E_2 - похибки, K - границі активації фільтра, N - величина черги звернень, $\lambda = \lambda_L + S$ - навантаження під час здійснення нападу, $\lambda^* = \lambda_L + S\left(1 - E_2\right)$ - навантаження при використанні фільтру, $Z\mu$, де, Z - коефіцієнт уповільнення роботи серверу.

Імовірність блокування звернення.

$$
p_{BLK} = \frac{\dfrac{1}{N!}\left(\dfrac{\lambda}{\mu}\right)^{K-1} \dfrac{\lambda}{\mu^*}\left(\dfrac{\lambda^*}{\mu^*}\right)^{N-K}}{\displaystyle\sum_{i=0}^{K-1} \frac{\left(\dfrac{\lambda}{\mu}\right)^i}{i!} + \sum_{i=K}^{N} \frac{1}{i!}\left(\frac{\lambda}{\mu}\right)^{K-1} \frac{\lambda}{\mu^*}\left(\frac{\lambda^*}{\mu^*}\right)^{i-K}}
\qquad (2.4)
$$

Тому ефективність кластеризації трафіку можна оцінити, як $R = \left(1 - E_1\right)\cdot\left(1 - p_B\right)$.

Далі відбувається корекція отриманих вибірок з врахуванням критеріїв успішності:

- розмірності отриманих кластерів;

- подільність надійних вибірок;

- відповідності вибраних центрів мас.

Кластерний аналіз - завдання поділу даної вибірки об’єктів (ситуацій) на підмножини, які називаються кластерами, щоб кожен кластер складався із подібних об’єктів, а об’єкти різних кластерів суттєво відрізнялися. Завдання кластеризації відноситься до категорії статистичної обробки,[13].

Кластерний аналіз - це не єдиний алгоритм, а загальна проблема, для вирішення якої використовуються різні підходи. наприклад, алгоритми побудови кластерів можуть суттєво відрізнятися у розумінні того, що призначити одному кластеру і як ефективно їх шукати. Серед популярних концепцій кластерів застосовуються групи з елементами, які формуються, базуючись на відстані між ними, щільності розділів в просторі даних, інтервалів чи конкретних статистичних розподілів. Тому кластеризацію можна подати як проблему багатокритеріальної оптимізації.

Для розрахунку подібності надійних кластерів і надалі для класифікації звернень, що надходять можна скористатися «Байєсвським класифікатором» [12].

Наївний «Байєсвським» класифікатор - імовірнісний класифікатор, який використовує теорему Баєса для обраховування ймовірності приналежності спостереження (елементів вибору) до одного із класів при приєднанні (наївному) незалежності змін.

Потрібно, якщо на основі значущих змін можна однозначно визначити, до якого класу слід керуватись спостереженням, байтовий класифікатор визначає ймовірність приналежності до цього класу.

У проміжних же варіантах, коли спостереження може бути різною імовірністю, що належать до різних класів, результати роботи класифікатора будуть векторними, компоненти яких є імовірними можливостями доступу до того чи іншого класу.

Можна бачити, що ідеальний бай класифікатор у будь-якому сенсі є оптимальним. Його результат не може бути поліпшень, тому що в усіх випадках,

коли можлива однозначна відповідь, він його дасть - і в тих випадках, коли відповідь неоднозначна, результат кількісно визначає міру цієї неоднозначності.

У той же час основний недолік класифікатора полягає в оптимальності: для його побудови потрібна вибірка, що містить усі можливі комбінації змін - і розмір таких варіантів зростає експоненціально із збільшенням кількості змін. Ознайомитись із вищеописаною проблемою практичного використання збройних бойових класифікаторів - класифікаторів, побудованих на основі підготовки до незалежності змін, що передбачається встановити, що використання цього вступу не може вивчити взаємодію всіх можливих сукупних змін, обмежених лише кожною змінною окрема ситуація для одного з класів.

Перевага цього підходу полягає в тому, що вимоги до розміру зменшуються з впливу на лінійні. Недоліком є те, що модель є точною лише тоді, коли вносяться зміни до незалежності. В іншому випадку, строго кажучи, обчислювальна ймовірність вже не така (і навіть більше, їх сума може бути не дорівнює одиниці, саме тому вам потрібно нормалізувати результат). Однак на практиці незначне відхилення від незалежності призводить лише до незначного зниження точності, і навіть у випадку значного зв’язку між різними результатами класифікатор продовжує корелювати із справжньою приналежністю форми до класів. Згідно з цим класифікатором переваг (висока швидкість, масштаб простірі, обсяг вимог до пам'яті), загалом переважають недоліки.

Згідно теореми Байєса, маємо:

$$p(C \mid F_1,...,F_n) = \frac{p(C) \cdot p(F_1,...,F_n \mid C)}{p(F_1,...,F_n)}$$

Умовний розподіл виражено так:

$$p(C \mid F_1,...,F_n) = \frac{1}{Z} p(C) \prod_{i=1}^{n} p(F_i \mid C)$$

Отже, для класифікації трафіку за двома ознаками маємо вираз:

$P(T \mid D) = \frac{P(T)}{P(D)} \prod_{i=1}^{n} P(w_i \mid T)$ - для легальних користувачів;

$P(H \mid D) = \frac{P(H)}{P(D)} \prod_{i=1}^{n} P(w_i \mid H)$ - для нелегальних користувачів

Після завершення етапу компоненти з групи *T**, які описують загрозливий трафік, міняються з компонентами групи *H* з врахуванням описаних критеріїв. Цей крок повторюється, поки всі елементи множини *T* не будуть позначені як надійні.

Отримані зразки, що відповідають надійному та шкідливому трафіку, а також механізм їх підтримання в поточному стані дозволяють використовувати їх з різними класифікаторами.

На рис. 2.7 зображено блок схеми алгоритмів по визначенню початку атаки і виділенню загрозливого трафіку (а - виділення загрозливого трафіку, б і в – ідентифікація початку розподіленої атаки).

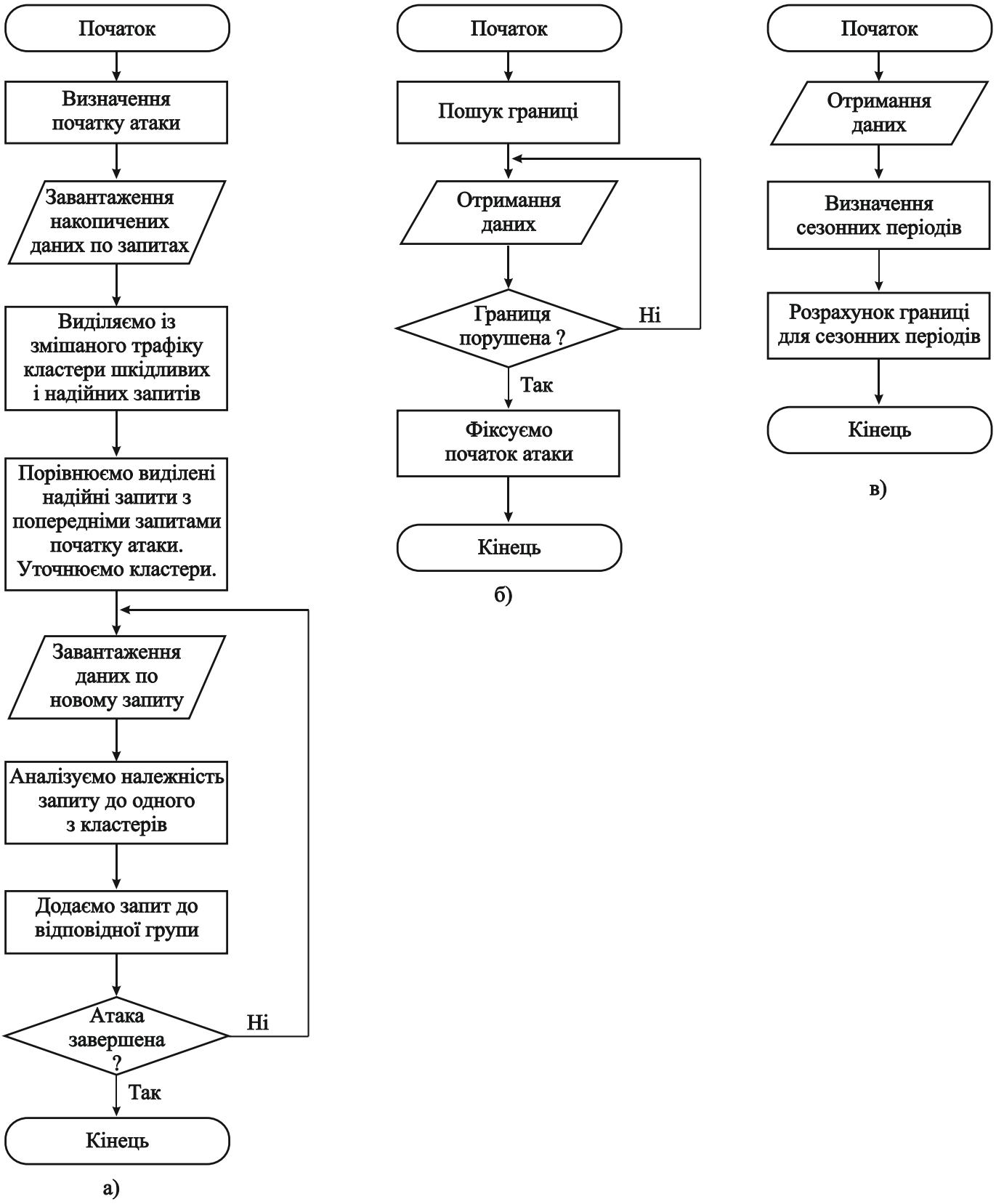


Рисунок 2.7 - Алгоритм по ідентифікації початку розподіленої атаки і виділенню загрозливого трафіку

Першим кроком є виклик підпрограм для визначення циклічних періодів, обчислення допустимого обмеження кількості звернень та визначення початку атаки. У разі атаки змішаний трафік поділяється на два кластери, один з яких містить зловмисні зверненя, а другий надійні. Нові звернення аналізуються на належність кластеру, а результат додається у відповідний кластер.

2.4 Висновки

Розроблено метод отримання зразків навчання та класифікації вхідного трафіку на групи шкідливих та надійних додатків. Алгоритм кластеризації k-середніх використовується для розділення змішаного трафіку. Вибір цього алгоритму виправданий, проведено доказ його ефективності. Для алгоритму вибираються оптимальні характеристики та розмірність даних, розробляються критерії успіху. Розроблені алгоритми складають основу узагальненої методики виявлення DDoS-атак та загрозливого трафіку, яку загалом можна записати наступним чином:

1.Ввизначаємо існуючі сезонні періоди і границю кількості звернень під час них.

2. Порушення границі сигналізує про початок атаки.

3. Кластеризуємо трафік.

5. Розділяємо змішаний трафік на надійний та загрозливий (алгоритм k-середніх).

6. Порівнюємо трафік, який передує початку атаки, із кластером, надійного трафіку, безпосередньо виділеного із змішаного трафіку.

7. Коригуємо кластери, із врахуванням критеріїв їх успішності.

8. Повторюємо етапи.

3 КОМПЛЕКСНИЙ МЕТОД ПРОТИДІЇ HTTP-FLOOD DDOS-АТАКАМ СЕРЕДНЬОЇ І МАЛОЇ ІНТЕНСИВНОСТІ

3.1 Аналіз даних та засобів реалізації методу фільтрації трафіку

Найбільше зростання кількості розподілених атак, на 836% в порівнянні з попереднім періодом, було зафіксовано в Україні на ринку онлайн-кас [23]. Атаки йдуть не на самі каси, а на ті сервери, на які вони відправляють дані. Реальне зростання числа атак спостерігається в страхуванні, букмекерських конторах, онлайн-іграх, а також у банків (рис.3.1).

Рисунок 3.1 – Статистика DDoS-атак у 2019 р

Причому DDoS-атака була б неможлива без десятків тисяч зламаних по всьому інтернету пристроїв, які без відома їх власників відправляють по команді зловмисників безглузді звернення на сайт обраної ними жертви. Останнім часом все

частіше цими пристроями стають всілякі пристрої інтернету речей (Internet of Things, IoT): IP-камери, онлайн-каси, Wi-Fi-маршрутизатори та ін

У даній кваліфікаційній роботі автор ставить перед собою мету - розробити алгоритм протидії HTTP-flood DDOS атак середньої і малої інтенсивності. Алгоритм повинен відповідати наступним вимогам:

- крос-платформеність;
- швидкість розгортання;
- робота в автоматичному режимі;
- робота тільки з тими даними, які є в наявності у адміністратора вебресурсу;
- прийнятна вартість впровадження.

Основна частина. Для отримання актуальної вибірки, відповідної благонадійності трафіку, оптимально використовувати алгоритм раннього виявлення DDOS атак, що враховує сезонні коливання навантаження на мережний ресурс. Використання даного алгоритму дозволяє досить точно оцінити момент початку атаки, а також визначити початок атаки на ранніх періодах, коли зловмисник може дозовано почати підмішувати загрозливий трафік для негативного навчання фільтрів.

Суть алгоритму виявлення початку атаки зводиться до розрахунку середньоквадратичного відхилення основних кількісних властивостей мережевої активності і подальшого порівняння прогнозного і фактичного значень. Для розрахунку середньоквадратичного відхилення використовуються останні п періодів, актуальних сезонів. Наприклад, період з 9-00 до 10-00, кожного понеділка [4].

Трафік, що приходить після початку атаки, буде включати в себе як загрозливий, так і легітимний трафік.

У першому наближенні виділити зловмисний трафік можна за допомогою алгоритму кластеризації к-середніх. Даний алгоритм дає можливість проводити кластеризацію по заздалегідь відомій кількості кластерів.

Суть методу полягає в тому, що на кожному кроці ітерації переобчислюють центр мас для всіх кластерів, отриманих на попередньому етапі. Далі вектори розбиваються на кластери ще раз, відповідно до того, який із нових центрів мас виявився ближчим по вибраній метриці [5].

Крок завершується, коли на якомусь етапі не проходить зміна кластерів. У разі аналізу лог-файлів, кластеризації можна здійснювати окремо по кожній групі кластерів, наприклад, за кількістю звернень з якоїсь адреси або за числом звернень до певної сторінки. В цьому випадку остаточна вибірка буде являти собою звернення, що потрапляють ц перетин різних груп даних, по кожному кластеру.

Враховуючи значне переважанням серед атак «відмова в обслуговувані» атак типу HTTP-flood, проєктований заіб протидії орієнтована на відбивання атак цього типу.

У зв'язку з тим, що лог-файли мають свій специфічний формат, їх аналіз стандартними методами є скрутним. Для проведення аналізу був створений скрипт, який видобуває з лог- файлу необхідні дані і експортує їх в базу даних. Скрипт був реалізований з допомогою мови PHP (Hypertext Preprocessor). Перевага даної мови програмування віддано в зв'язку з наявністю багатого інструментарію по роботі з рядками і регулярними виразами [6]. В якості системи управління базами даних обрана вільно поширювана СУБД MySQL версії 5.5.23. Використання СУБД дозволило прискорити процес обробки і аналізу даних і зробити його більш гнучким.

Для проведення власне самого аналізу також була розроблена окрема програма. Мова реалізації програми PHP [18]. ЇЇ використання дозволило витримати створений програмний комплекс в одному ключі і дало можливість реалізувати вебінтерфейс. Вебінтерфейс може бути доступний для системного адміністратора з будь-якого комп'ютера, що дозволяє в віддаленому режимі діагностувати атаку і виявляти у вхідному трафіку різні аномалії. У подальшому планується доопрацювати програму для роботи в повністю автоматичному режимі. У цьому

випадку дані з лог-файлу будуть експортіро¬ваться в базу даних в режимі реального часу. Аналіз повинен відбуватися після кожного нового додавання даних. У разі виявлення початку атаки програма буде розсилати необхідні повідомлення і автоматично задіяти заходи щодо протидії нападу.

PHP (Hypertext Preprocessor - гіпертекстовий препроцесор) - мова програмування, створена для організації HTML-сторінок із сторони вебсервера. PHP є однією фз найпоширеніших мов, які використовуються в галузі веброзробок разом з Java, .NET, Perl, Python і Ruby [20]. Сервер інтерпретує код PHP в код HTML, і передає клієнту. Також код PHP можна інтегрувати безпосередньо в код html. На відміну від мови програмування JavaScript, користувач не бачить PHP-коду, атже браузер клієнта отримує готовий html-код.

За допомогою PHP ви можете робити все, що роблять програми CGI (Common Gateway Interface) - стандарт інтерфейсу, який використовується для взаємодії з програмою сервера і зовнішньою програмою. Наприклад, обробка форм, генерація хмісту динамічних сторінок, надсилання і отримання файлів cookie. Але PHP має набагато більший функціонал.

Основні програми PHP-коду.

1. Написання сценаріїв для виконання на стороні сервера. PHP традиційно та найбільш часто використовується саме таким чином. Вам потрібно запустити вебсервер зз встановленим на ньому PHP. Ви можете завантажити сторінку з PHP-кодом на вебсервер і переглянути результати її функціонування через веббраузер. Все це можна виконати навіть на домашньому комп’ютері. Також тестувати PHP-код можна з допомогою онлайн-сервісу.

2. Створення сценарії для запуску в командному рядку без будь-якого сервера чи браузера. Цей тип засьосування ідеально підходить для сценаріїв, які регулярно запускаються через cron (на * nix або Linux) або схожий планувальник завдань (на Windows). Ці сценарії також можна використовувати для зручної обробки текстів.

3. Написання програм із графічним інтерфейсом. Хоча мова PHP не

оптисальна для такого використання.

PHP також може використовуватися в більшості операційних систем: Linux, різних варіантах Unix (включаючи HP-UX, Solaris та OpenBSD), Mac OS X, Microsoft Windows, RISC OS та, можливо, інших.

PHP також має підтримку більшості розповсюджених вебсерверів: IIS, Apache, та багатьох інших. Також, підтримка доступна на будь-якому вебсервері, що використовує двійкові коди FastCGI, чи протокол клієнт-сервер для взаємодії вебсервера та додатків.

PHP не обмежується створенням лише HTML-коду: можна представляти зображення, файли PDF і навіть Flash-відео, які генеруються "на льоту". Ви можете легко вивести будь-який текст XHTML та будь-який інший XML-файл. PHP може генерувати такі файли та зберігати їх у файловій системі, а не просто відображати текст, створюючи кеш на динамічному вмісті на стороні сервера.

Для роботи багатокритеріального методу виявлення DDoS-атак необхідно провести попередній аналіз та розрахунки: визначити показники (критерії), за якими буде ідентифіковано наявність або відсутність атаки; побудувати модель для звичайного мережного трафіку; встановити пороги для вибраних показників.

В якості критеріїв оцінки були обрані наступні показники:

- рівень завантаження процесора сервера;
- обсяг зайнятої оперативної пам'яті сервера;
- розмір упаковки;
- поточний рівень трафіку (Мбіт / с);
- розподіл значення адреси джерела звернень (source ip);
- користувач - агент у запиті;
- URI (ієрархічна частина та фрагменти URL-адреси запиту);

Наявність атаки потоку GET може характеризуватися кількістю звернень від джерела в секунду [8]. Зрозуміло, що законний користувач не постійно робить велику кількість звернень на той самий ресурс, як це може вузол, керований

зловмисником (зомбі). Тому за деякий час At надходить із IP-адреси джерела x надходить s. звернень.

Необхідно визначити поріг, при якому звернення від джерела вважаються зловмисними.

Низький поріг може призвести до помилок першого виду (помилково позитивних) при визначенні наявності атаки, а занадто високий – може призвести до помилок другого виду (пропуск події). Враховуючи вірогідність помилки через неоднорідність мережного середовища та відносну складність розрахунку значення границі, у разі неоднозначності порогового значення H, необхідно враховувати оцінку інших критеріїв.

Запропонований процес виявлення DDoS-атак прикладного рівня складається із наступних етапів (рис. 3.2):

1.Захоплення трафіку і накопичення лог-файлу із даними по вхідних зверненнях на сервер.

2. Надсилання лог-файлу до обчислювального кластеру.

3. Визначення кількості ентропії GRPS за допомогоювикористання моделі MapReduce.

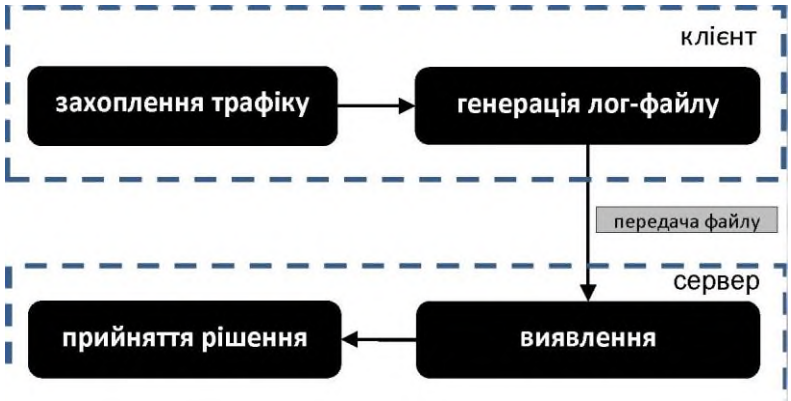


Рисунок 3.2 - Схема роботи методу виявлення HTTP GET flood

Лог-файл містить інформацію про звернення, значення обраних для оцінки критеріїв, адресу відправника, адресу отримувача, а також мітку часу і протокол:

139875; 10.1.12.73; 10.1.12.101;

HTTP/1.1; 730; GET; /posts/23877/324/req?k=12877182746;

UA=Mozilla/5.1 (Macintosh; U; en-US) Apple WebKit/533.4 (KHTML, like Gecko) Chrome/5.1.375.86 Safari/533.4;

CPU=11; LOAD=4; RAM=66;

Необхідно задати інтервал часу вимірювання (time interval), порогове значення числа звернень на сервер (threshold) та порогове значення числа звернень між окремим клієнтом та сервером (peer threshhold). Після перевищення заданого рівня числа звернень до сервера (threshold) система знаходить звернення з завищеним peer treshold і блокує відправника цих звернень.

Результати дослідження. Щоб згенерувати HTTP GET звернення при емуляції DDoS-атаки використовується ряд утиліт, які доступні у відкритому доступі Mausezahn [23], LOIC [9], Scapy [11], Iperf [32]. Для проведення експерименту було обрано Mausezahn через можливість генерувати потік звернень з підставною адресою відправника.

3.2 Комплексний метод виявлення розподілених атак на інформаційну систему

Як джерело аналізованих даних виступає стандартний access log одного з найпопулярніших вебсерверів – Apache.

Реалізація програмного застосунку буде відбувається на прикладі аналізу інформації, отриманої з log-файлів цього сервера. Дані запитів до серверу Apache зберігаються у log-файлі access. log, що містить:

$\%\ H\%\ l\%\ u\%\ t\ \backslash"\%\ r\ \backslash»\%>\ s\%\ b\ \backslash"\%\ \{Referer\}\ i\ \backslash"\ \backslash"\%\ \{User-Agent\}\ i\ \backslash»,$

в котрому:

$\%\ H$ - хост / IP-адреса, з якого зробленр зверненнядо сервера;

$\%\ T$ - час запиту до сервера і часовий пояс сервера;

% R - тип запиту, його версія і вміст;

% S - код стану HTTP запиту;

% B - кількість переданих сервером байт;

% $\{Referer\}$ - URL-джерело звернення;

% $\{User-Agent\}$ - HTTP-заголовок, який містить інформацію про запит (мову, клієнтську програму і т. д.).

Комплексний метод виявлення розподілених атак на інформаційну систему включає наступні кроки:

1. Програма постійно аналізує трафік на предмет початку атак, дані отримуються з файлу access.log, проходять обробку та завантажуються у базу даних.

2. У разі початку нападу фіксується точка початку атаки та в базі даних створюються дві додаткові таблиці, відповідні благонадійності трафіку та, відповідно трафіку, який прийшов після початку нападу.

3. За допомогою алгоритму k-середніх трафік, прийшовший після початку нападу, ділиться на дві групи.

4. Трафік, позначений як загрозливий, блокується.

5. На підставі загрозливого трафіку створюються необхідні правила заборони для файєрволу.

Для створення гнучкості кросплатформеності ПЗ розділене на три блоки:

- блок обробки даних і їх завантаження в базу даних;

- блок виявлення початку атаки;

- блок фільтрації трафіка.

Всі присутні в log-файлі дані можуть бути використані для аналізу. На підставі блоку GeoIP, можна виділити звернення з певного регіону, країни і т.д.

Структура розробленого програмного комплексу представлена на рис. 3.3:

1. В результаті опрацювання звернень, що надходять до вебсерверу вони додаються в журнал.

2. Блок завантаження даних направляє їх в базу даних.

3. Відбувається аналіз звернень за допомогою модулю виявлення початку атаки. В разі його спрацювання в базі даних формуються дві порожні таблиці. Одну для дозволених звернень, другу для нелегітимних.

4. Блок виявлення загрозливого трафіку відстежує їх стан і проводить кластеризацію і первинне заповнення таблиць. Якщо в таблицях вже є дані, блок аналізує звернення, шо надійшли на предмет приналежності до груп надійних або нелегітимних звернень, і добавляю інформацію про зверненення у відповідну таблицю.

5. Модуль блокування запиту одержує паролі IP-адрес з таблиці, що містить нелегітимні звернення і вносить їх в «чорні списки» файєрволу.

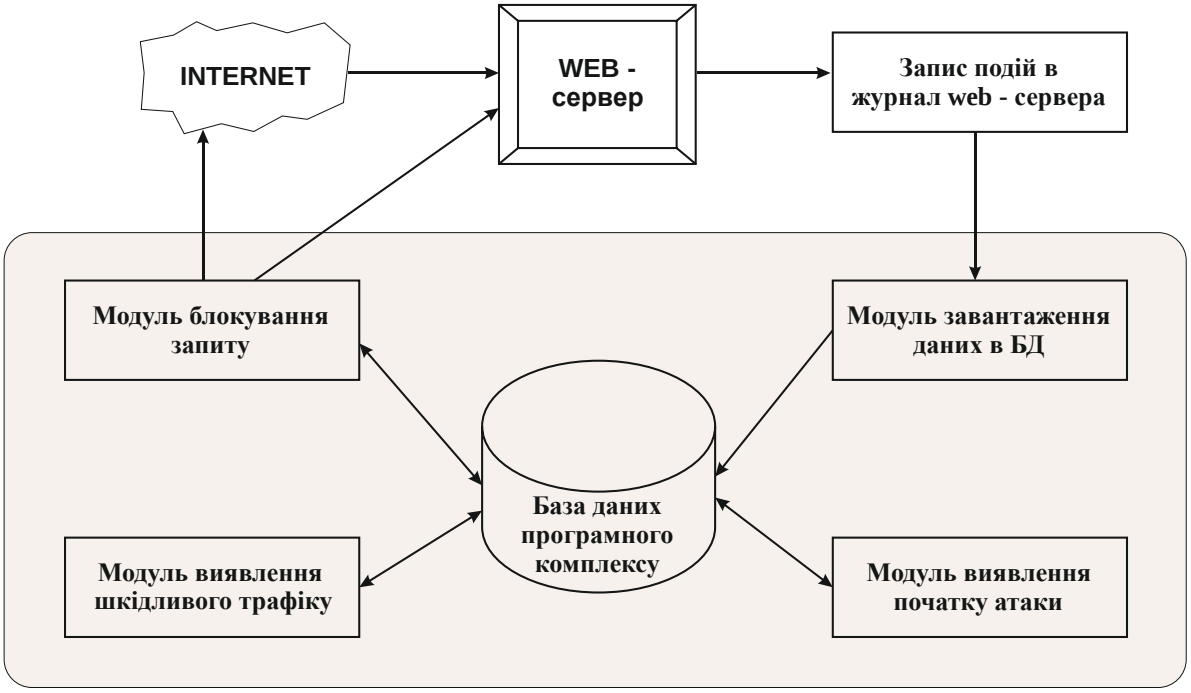


Рисунок 3.3 - Структура програмного комплексу

3.3 Висновки

Метод сезонного аналізу показав вищу точність виявлення DDoS-атаки та менший час, що пройшов від початку атаки до її діагностики.

На основі IP-адрес, що належать до ботових мережних комп'ютерів, які були виявлені під час аналізу файлів журналів, можна було точно визначити час початку атаки.

В середньому, за всіма тестами, час виявлення DDoS-атаки методами з урахуванням сезонності скорочувався в 4 рази, а також зменшувалась кількість помилкових спрацьовувань.

Досить великою складністю, яка виникає при використанні цього методу, є правильний вибір подібних періодів. Дані з таких серверів були відібрані для тестування, періоди роботи яких були однозначно визначені і не викликали жодних сумнівів. Однак важко визначити різні періоди роботи, наприклад, великого магістрального маршрутизатора. Його діяльність може не підпорядковуватися щоденним або тижневим періодам, але може також мати свої періоди, які можуть бути складними періодами, що виникають в результаті додавання діяльності різних груп користувачів, таких як користувачі з різних часових поясів. Крім того, вже існуючі сезонні періоди можуть змінюватися, до них можуть додаватися нові періоди, тому при постійному моніторингу руху пакетів потрібно буде проводити його кластеризацію та визначати нові сезонні періоди в роботі.

Отримані засоби протидії відповідають цілям, встановленим у дослідженнях.

Робота цієї програми, а також теоретична частина алгоритму була перевірена в лабораторії з використанням даних, що відповідають реальним DDoS-атакам.

Універсальність програмного пакету для виявлення DDoS-атак в можливості його використання не тільки для виявлення http-флуду, але і для DDoS - атак різних типів. При незначних змінах, програмне забезпечення може використовуватися для аналізу даних, що містяться в log-файлах різноманітних мережних сервісів.

4 ДОСЛІДЖЕНЯ МЕТОДУ ВИЗНАЧЕННЯ ВРАЗЛИВОСТЕЙ ДО РОЗПОДІЛЕНИХ АТАК

4.1 Функціональна модель системи

Для аналізу особливостей системи виявлення мережних вторгнень необхідно створити опис сисктми у вигляді функціональної моделі.

Результатом використання методології є функціональна модель (рис 4.1), що складається із діаграм, фрагментів тексту і глосарію, пов'язаних між собою посиланнями. Основними компонентами моделі є схеми, всі функції і інтерфейси яких представлені у вигляді блоків і дуг.

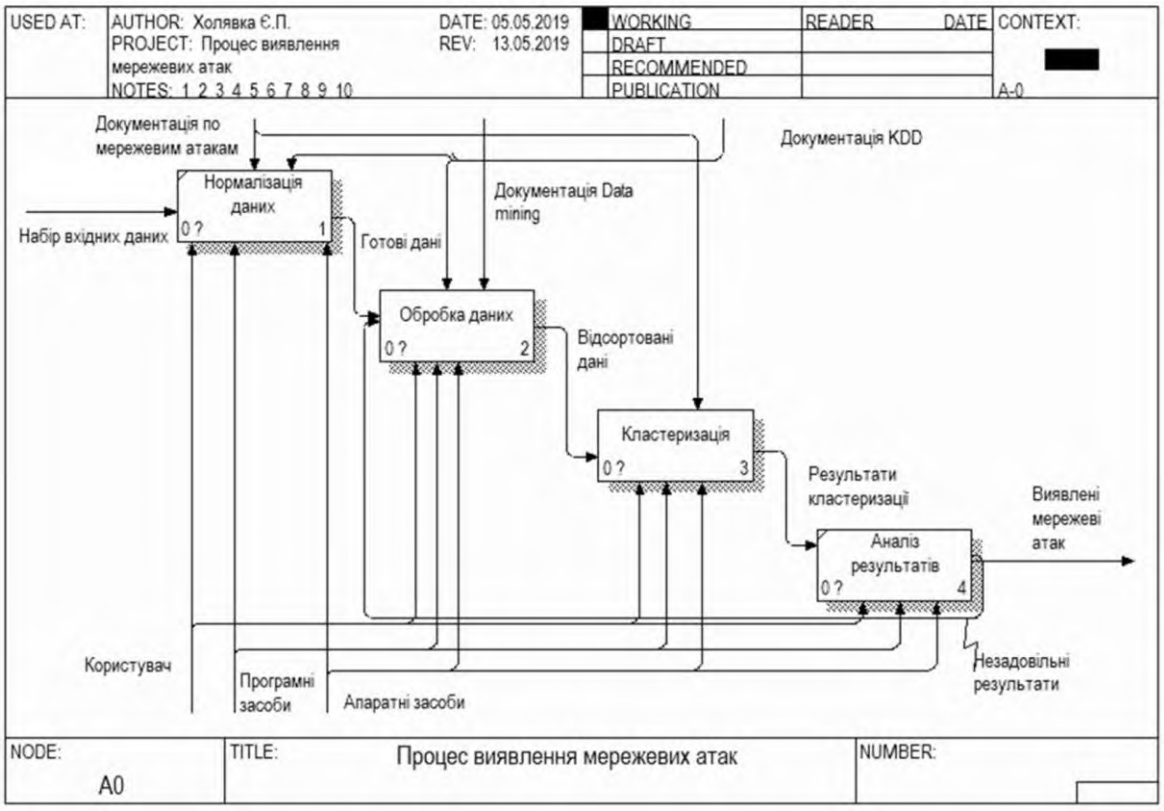


Рисунок 4.1 - Функціональна модель інформаційної системи

Точка з'єднання дуги з блоком визначає тип інтерфейсу:

- контрольна інформація включена у блок вище;

- оброблена інформація відображається на лівій стороні блоку;

- результати функції відображаються в правій частині блоку;

- механізм (автоматизована система або особа), який виконує операцію, відображається у вигляді дуги, яка включена в блок нижче [27].

Робота системи здійснюється шляхом виконання відповідних функцій відповідних підсистем:

- блок нормалізації даних: здійснює обробку даних і перетворення текстових даних у числові;

- блок кластеризації: забезпечує обробку та подання інформації;

- блок обробки даних: є алгоритмом вибору атрибутів для зменшення кількості обчислювальних операцій;

- блок аналізу результатів: призначений для перевірки належної роботи компонентів системи;

На практиці склад підсистеми захисту може відрізнятися залежно від конкретного здійснення, виду діяльності та відповідних вимог щодо інформаційної безпеки нормативних актів.

Він має такі блоки:

- ранжування атрибутів: алгоритми сортування даних використовуються у порядку зростання важливості;

- відмова від зайвих атрибутів: алгоритми вибору атрибутів застосовуються для зменшення розмірності вхідних даних.

Наступним кроком є створення діаграми використання. Діаграми випадків використання показують взаємозв'язок між випадками використання, які представляють функції системи, та суб'єктами, які представляють людей або системи, які приймають або передають інформацію до цієї системи. Діаграми використання показують багато інформації про систему. Цей тип діаграм описує всю функціональність системи. Користувачі, аналітики, розробники, менеджери

проектів, фахівці з контролю якості та всі, хто цікавиться системою в цілому, можуть зрозуміти, що повинна робити система, переглянувши схеми використання [28].

Першим кроком опису функціональних можливостей системи є моделювання вимог до неї.

Завданнями аналізу та моделювання вимог є:

- досягнення порозуміння між розробниками, замовниками та користувачами щодо того, що повинна робити система;

- обмеження її функціональності;

- досягнення кращого розуміння поведінки ;

- створення бази для планування перед розробкою проєкту;

- визначення типу інтерфейсу користувача.

Результатом цього кроку є розробка схеми випадків використання для інформаційної системи (рис. 4.2)

Рисунок 4.2 - Діаграма варіантів використання для інформаційної системи

4.2 Дослідження процесу створення навантажувальної мережі

Компанія CISCO, лідер у виробництві і реалізації мережних рішень, визнає, що на сьогоднішній день немає достатніх коштів для боротьби з DDoS атаками. Це пов'язано з тим, що атаки такого типу виникають несподівано і у системних адміністраторів немає можливості проаналізувати атаку до моменту її початку. Підбирати заходи протидії атакам доводиться вже в момент проведення атаки, коли мережевий ресурс вже відчуває труднощі. Крім того, в кожному конкретному випадку досвід минулих атак може бути недостатній для відображення нової атаки. Це пов'язано з тим, що зловмисники постійно розвивають засоби атак, змінюють стиль, конфігурацію пакетів і т.д. [2].

Дослідження DDoS атак схоже з вивченням природних явищ, таких як землетруси, виверження вулканів, розряди блискавки, тобто які можуть бути безпосередньо досліджені тільки в момент їх настання. Можна створити математичну або комп'ютерну модель DDOS атаки і проводити її дослідження, але немає ніяких гарантій, що ця модель буде відображати всі нюанси наступної DDOS атаки, яку зможуть придумати і реалізувати зловмисники. Аналогічно і з засобами захисту, зловмисник може знайти елемент мережного ресурсу, атака на який призведе до відмови в обслуговуванні всього ресурсу [13].

Для дослідження DDoS атак необхідний механізм, який би в лабораторних умовах міг повторювати реальні DDoS атаки як угодне кількість разів, емулювати нові атаки, максимально відповідають реальним, вносити зміни в основні параметри атаки і відстежувати результат. За своєю суттю механізм повинен представляти розподілену ,jn мережу, максимально наближену до реальних чинним бот-мереж.

В якості спеціалізованого програмного забезпечення для створення навантажувальної мережі виступає Apache JMetr.

Застосунок Apache JMeter - це програмне забезпечення із відкритим кодом, 100% чиста програма Java, призначена для завантаження тестової функціональної поведінки та вимірювання продуктивності. Спочатку він був розроблений для тестування вебпрограм, але потім розширився до інших тестових функцій.

Apache JMeter може використовуватися для тестування продуктивності статичних, динамічних ресурсів, вебдинамічних додатків.
Він може бути використаний для модедювання великого навантаження на сервер, групу серверів, щоб перевірити його міцність та проаналізувати загальну продуктивність при різних типах навантаження на нього.

Особливості Apache JMeter включають:

- Можливість змоделювання та перевірка продуктивності багатьох різних типів програм / серверів / протоколів:
  - Інтернет - HTTP, HTTPS (Java, NodeJS, PHP, ASP.NET,…)
  - Веб-сервіси SOAP / REST
  - LDAP
  - FTP
  - База даних через JDBC
  - Проміжне програмне забезпечення, орієнтоване на повідомлення (MOM) через JMS
  - Власні команди або сценарії оболонки
  - TCP
  - Пошта - SMTP (S), POP3 (S) та IMAP (S)
  - Об'єкти Java
- Повнофункціональна тестова середовище, яке дозволяє швидко записувати план тесту (з браузерів або власних застосунків), створювати та налагоджувати.
- Рrежим командного рядка для завантаження тесту з будь-якої сумісної з Java операційної системи (Linux, Windows, Mac OSX,…)

- Змістовний і готовий до презентації динамічний звіт HTML
- Простота кореляції завдяки властивості витягувати дані з найпопулярніших форматів відповідей, JSON, XML, HTML, або будь-якого текстового формату
- Повна багатопотокова структура спрощує одночасну вибірку для багатьох потоків і одночасну вибірку різних функцій між окремими групами потоків.
- Кешування і офлайн-аналіз / відтворення результатів проведеного тесту.
- Високорозширюване ядро:

Як атакуємий сервер використовується виділений фізичний сервер із зазначеними вище технічними та системними характеристиками. На сервері повинен бути таким набір серверно¬го програмного забезпечення:

- вебсервер: Apache;

- сервер баз даних: MySQL.

При проведенні експериментів ставилося завдання виробити методику пошуку потенційно вразливим-мих місць у системі управління змістом - CMS - Content Management System [16]. В якості системи керування вмістом була застосована CMS Wordpress. На сьогоднішній день це одна з найпопулярніших CMS із відкритим вихідним кодом.

Узагальнюючи, CMS Wordpress це вебдодаток, що дозволяє власникам сайтів, редакторам, авторам управляти їх сайтами і публікувати вміст без жодних знань програмування.

WordPress використовує PHP і MySQL, вони підтримуються практично всіма хостинг-провайдерами. Однак спеціальні тарифні плани хостингу для WordPress можуть значно поліпшити показники швидкості, продуктивності та надійності сайту.

Зазвичай, саме цю CMS використовують для створення блогу. Проте сайт на WordPress може бути легко модифікований в інтернет-магазин, новинний сайт, портфоліо або будь-який інший тип ресурсу.

Одне з прекрасних властивостей платформи - інтуїтивно зрозумілий та зручний у використанні інтерфейс.

Експеримент проводився в кілька етапів, в ході кожного етапу комп'ютери зомбі-мережі посилали звернення до певного скрипту системи управління вмістом. При цьому фіксувалася навантаження, яку генерував скрипт. Оцінка навантаження проводилася за наступними критеріями: завантаження процесора, використання ресурсу пам'яті, час відгуку сервера. Дані про завантаження процесора і використанні памяті фіксувалися безпосередньо на комп’ютері, який атакувався. Час відгуку фіксувався з нейтрального комп'ютера, що знаходився в тій же підмережі, що і комп'ютери бот-мережі. Сценарій проведення атаки був розроблений на підставі даних лог-файлів вебсервера відповідних однією з реальних DDoS атак.

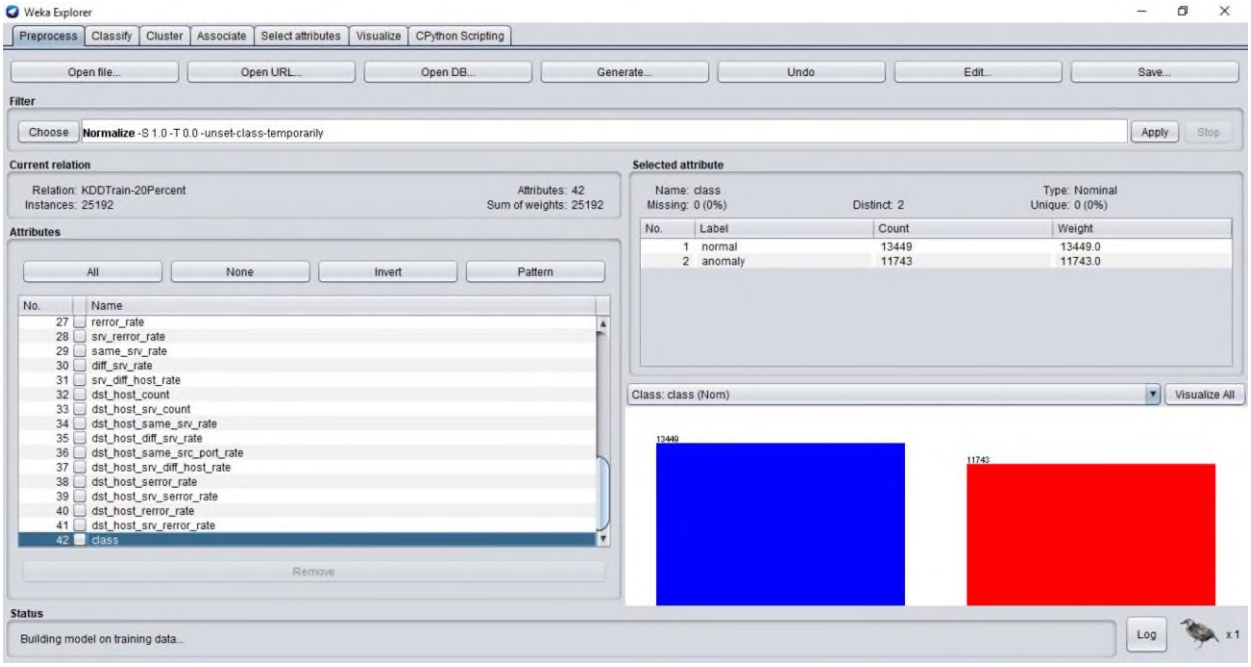


Рисунок 4.3 – Отримання данних для кластеризації

Основною трудністю при проведенні експерименту стало виявлення цілей-скриптів, на які необхідно було проводити атаки. Для створення списку всіх можливих цілей була використана бібліотека mnoGoSearch для мови PHP, яка представляє собою модуль для повнотекстового пошуку - реаліцію пошукової машини.

Результати оцінки ефективності виявлення початку атаки та виявлення загрозливого трафіку в цілому відповідають результатам, отриманим під час випробувань в навантажувальній мережі.

Відхилення для похибки першого виду було не більше 3%, відхилення для похибки другого виду - не більше 6%.

Зведені показники ефективності розробленого програмного забезпечення за результатами впровадження представлені в табл. 4.1.

Таблиця 4.1 - Оцінка ефективності розробленого програмного забезпечення

№ п/п	Назва показника	Результат
1	Зменшення часу ідентифікації початку нападу	в 4-5 разів
2	Зменшення економічних витрат на створення системизахисту	на 18%
3	Збільшення часу працездатності об’єкту під час атаки	в 4 рази

Після проведення зазначеного дослідження необхідно оптимізувати роботу отриманих вразливих сценаріїв і повторити дослідження, необхідну кількість разів. Це дозволяє виявляти нові увразливості на кожному кроці та підвищувати стійкість сервера до майбутніх атак.

4.3 Висновки

Мережні атаки стають все більшою небезпекою для економіки України і світу.

Зручним інструментом вирішення завдання задачі кластеризації проблемних ситуацій наявності кібератак, може бути розроблене програмне забезпечення.

Дані тестів підтверджують ефективність та результативність розробленого програмного забезпечення для протидії DDoS-атакам та їх виявлення.

В рамках експериментів проведено порівняння розробленого програмного забезпечення з існуючими аналогами. Встановлено, що розроблений засіб є більш ефективним. Крім того, розроблений інструмент показує кращі результати у виявленні атак порівняно з результатами, досягнутими в подібних дослідженнях.

ВИСНОВКИ

У магістерській роботі вирішено наукове завдання раннього виявлення початку DDoS-атаки і подальшого визначення загрозливих звернень.

Основними результатами дослідження є:

1. Проаналізовано сучасний стан DDoS атак, та запропоновано їх класифікацію. Проведено аналіз нинішнього стану технологій для вирішення проблем захисту інформації, проаналізовано методи, що застосовуються для рішення задач пов’язаних з доступністю ресурсів. Виділено новий вид атак малої інтенсивності, спрямовану на регіональні ресурси. Виявлено відсутність засобів, протидії для даної групи.

2. Розроблено математичну модель атак з врахуванням опису сезонності мережного навантаження.

3 Використовуючи модель атаки запропоновано метод раннього визначення та протидії DDoS-атакам малої інтенсивності.

5. Розроблено алгоритм визначення точок початку атаки ы алгоритм поділу змішаного трафіку на надійний та загрозливий, який враховує сезонну кількість мережного навантаження.

6. Запропоновано критерії успішності, які дозволяють оцінювати успішність роботи алгоритму роботи, і сторонніх засобыв за допомогою фільтраціь трафіку.

7. Описано інформаційну систему та розроблено її функціональну модель. Розроблено діаграму варіантів використання інформаційної системи виявлення мережних атак

8. Розроблено програмний засіб для виявлення початку атаки та подальшого виявлення та блокування нелегітимних звернень. Його особливістю є модульність та універсальність для забезпечення безпеки різних мережних ресурсів та захисту їх від атак різного типу.

9. Проведено ряд тестів, які підтверджують ефективність та результативність розробленого програмного забезпечення для протидії DDoS-атакам та їх виявлення.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Басалаєва, Ю.С. Вибір інструментів Data Mining для аналізу результатів дистанційної освіти / Ю.С. Басалаєва // Сучасні матеріали, техніка та технологія. - 2015. - № 2. - С. 22 - 25.

2. Бабаш, А.В. Информационная безопасность. Лабораторный практикум : учеб. пособие / А. В. Бабаш, Е. К. Баранова, Ю. Н. Мельников. - 2-е изд., стер. - М. : КНОРУС, 2016. - 132 с.

3. Бабаш, А.В. Криптографические методы защиты информации : учебник для студ. вузов / А. В. Бабаш, Е. К. Баранова. - М. : КНОРУС, 2016. - 190 с.

4. В. Л. Бурячок, “Сучасні системи виявлення атак в інформаційно-телекомунікаційних системах і мережах. Модель вибору раціонального варіанта реагування на прояви стороннього кібернетичного впливу”, Інформаційна безпека, № 1, с. 33-40, 2013.

5. Гремякіна О. А. Вибір платформи інтелектуального аналізу даних для застосування в академічних цілях // Молодий вчений. - 2015. - №22. - С. 26-29.

6. Глинская, Е. В. Информационная безопасность конструкций ЭВМ и систем : учеб. пособие для студ. вузов / Е. В. Глинская, Н. В. Чичварин. - М. : ИНФРА-М, 2016. - 118 с.

7. Даниленко, А.Ю. Безопасность систем электронного документооборота: Технология защиты электронных документов / А. Ю. Даниленко. - М. : ЛЕНАНД, 2015. - 232 с.

8. Д. Б. Мехед, Ю. М. Ткач, В. М. Базилевич, В. І. Гур’єв, та Я. Ю. Усов, “Аналіз вразливостей корпоративних інформаційних систем”, Захист інформації, т. 20, № 1, с. 61-66, 2018. doi: 10.18372/2410-7840.20.12453.

9. Джулій В.М. Метод виявлення та протидії розподіленим атакам, спрямованим на відмову в обслуговуванні / В.М. Джулій, В.І. Чорненький, О.О.

Савіцька, -Хмельницький: Вісник Хмельницького національногоуніверситету, 2019. - Вип. №1. – С.127-134

10. І. Яковів, “Інформаційно-телекомунікаційна система, концептуальна модель кіберпростору і кібербезпека”, Information Technology and Security, № 5 (9), с. 134-144, 2017.

11. Зінченко В. В. Виявлення DDoS-атак прикладного рівня / В. В. Зінченко, М. В. Зінченко // Міжнародна науково-технічна конференція «Радіотехнічні поля, сигнали, апарати та системи» (Київ, 16-22 березня 2015). - К., 2015. - С. 262-264.

12. Кабакова, Н. В. Система защиты информации ViPNet: курс лекций : учеб. пособие / Н. В. Кабакова [и др.] ; под ред. А. О. Чефрановой. - М. : ДМК Пресс, 2014. - 392 с.

13. Мережеві аномалії [Електронний ресурс]. URL: http://nag.ru/ articles/revievvs/15588/setevyie-anomalii .html

14. Організація комп'ютерних мереж: конспект лекцій/ Л.М. Олешенко ; КПІ ім. Ігоря Сікорського. - Електронні текстові дані. - Київ : КПІім. Ігоря Сікорського, 2018. - 225 с.

15. Платонов, В. В. Програмно-апаратні засоби захисту інформації. - М .: Видавничий центр «Академія», 2013. - 336 с.

16. Р. Грищук, В. Охрімчук, та В. Ахтирцева, “Джерела первинних даних для розроблення шаблонів потенційно небезпечних кібератак”, Захист інформації, №18 (1), с. 21-29, 2016. doi: 10.18372/2410-7840.18.10109.

17. Савіцька О.О. Архітектура програмного комплексу забезпечення безпеки виявлення і протидії DDoS-атакам / О.О. Савіцька, В.М. Джулій. - «Інтелектуальний потенціал – 2018» - збірник наукових праць молодих науковців і студентів з нагоди 30-річчя підготовки ІТ- фахівців в ХНУ/ Колектив авторів – Хмельницький: ПВНЗ УЕП, 2018. – Ч.3: Кібербезпека та актуальні проблеми комп’ютерних систем і мереж — С. 44 - 47.

18. С. В. Сальник, О. Я. Сова, та Д. А. Міночкін, “Аналіз методів виявлення вторгнень у мобільні радіомережі класу MANET”, Сучасні інформаційні технології у сфері безпеки та оборони, № 1 (22), с. 103-112, 2015. P-ISSN 2411-1031. Information Technology and Security. January-June 2019. Vol. 7. Iss. 1 (12) 32

19. Соколюк Я.В. Процес визначення початку атаки типу HTTP GET flood/ Я.В. Соколюк, І. В. Муляр // «Інтелектуальний потенціал – 2020» - збірник наукових праць молодих науковців і студентів / Колектив авторів – Хмельницький: ПВНЗ УЕП, 2020. – Частина 2. С. 69-73

20. Системи і методи виявлення вторгнень: сучасний стан і напрями вдосконалення [Електронний ресурс]. URL: http: / / citfomm.ni/security/ intemet/ids_overview/#3

21. Хайкін, Р. Нейронні мережі: повний курс, 2-е видання. Пер. з англ. / Р.Хайкін. - М .: Видавничий дім «Вільямс», 2006. - 1104 с

22. Холявка Є. П.; Метод виявлення мережних атак в комп’ютеризованих системах управління: наукова робота, Хмельницький національний університет. - Хмельницький, 2019, [Електронний ресурс]. URL: http://konkurs.khnu.km.ua/wp-content/uploads/sites/25/2019/04/DP3 Eugen.pdf

23. Ю. Васильєв, “Класифікація та аналіз загроз інформаційній безпеці в ключових системах інформаційної інфраструктури”, Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні, № 1 (29), с. 51-61, 2015.

24. Я. В. Корпань, “Класифікація загроз інформаційній безпеці в комп’ютерних системах при віддаленій обробці даних”, Реєстрація, зберігання і обробка даних, т. 17 № 2, с. 39- 46, 2015.

25. Akilandeswari V. Probabilistic neural network based attack traffic classification / V. Akilandeswari, S. Shalinie // Proceedings of the Fourth International Conference on Advanced Computing (Chennai, 13-15 Dec. 20182). - Chennai, 2018. - P. 1-8

26. A. A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning approach for Internet of Things," Future Gener. Comput. Syst., vol. 82, pp. 761-768, May 2018.

27. Baddar S.A.-H., Merlo A., Migliardi M. Anomaly Detection in Computer Networks: A State-of-the-Art Review // Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. 2014. vol. 5. no. 4. pp. 29-64.

28. DDoS Definitions - DdoSPedia [Electronic resource] - Access mode : http://security.radware.com/knowledge-center/DDoSPedia/http-flood/. - Access data : July 2019.- The title of the screen.

29. D. L. Nazareth, and J. Choi, “A system dynamics model for information security management”, Information & Management, vol. 52, iss. 1, pp. 123-134, 2015. doi:10.1016/j.im.2014.10.009.

30. Garg, T.; Khurana, S.S. Comparison of Classification Techniques for Intrusion Detection Dataset Using WEKA. In Proceedings of the International Conference on Recent Advances and Innovations in Engineering, Jaipur, India, 9-11 May 20Bharath, R.R., Thanigaivel, K., Alfahath, A., Prasanth, T.: Feature extraction based dynamic recommendation for analogous users. Int. J. Comput. Sci. Inf. Technol. 5(2), 1358-1362(2014)

31. Hall M., Witten I., Frank E. Data Mining: Practical Machine Learning Tools and Tech-niques. Morgan Kaufmann Publishers, 2011. 629 p.

32. Iperf : network performance measurement tool [Electronic resource] - Access mode : https://iperf.fr/ - Access data : August 2015. - The title of the screen.

33. J. Yu. A detection and offense mechanism to defend against application layer DDoS attacks / J. Yu, Z. Li, H. Chen, X. Chen // Networking and Services. - 2017. - Режим доступу до ресурсу: https://www.researchgate.net/profile/Xiaoming_Chen17/publication/4314603_A_Detection_and_Offense_Mechanism_to_Defend_Against_Application_Layer_DDoS_Attacks/links/546ee4da0cf29806ec2ebfeb.pdf.

34. Jie-Hao C. DDoS defense system with test and neural network / C. Jie-Hao, Z. Ming,

35. C. Feng-Jiao, Z. An-Di // Proceedings of the IEEE International Conference on Granular Computing (Hangzhou, China, 11-13 Aug. 2018). - Hangzhou, 20182 - P. 38-43.

36. K. Alrawashdeh and C. Purdy, "Toward an online anomaly intrusion detection system based on deep learning," in Proc. 15th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA), Anaheim, CA, USA, Feb. 2017, pp. 195-200.

37. K. M. Ting, T. Washio, J. R. Wells, F. T. Liu, S. Aryal, “DEMass: a new density estimator for big data,” Knowledge & Information Systems, Vol. 35, pp. 493524, 2013.

38. Lammel R. Google’s mapreduce programming model revisited / R. Lammel // Science of Computer Programming. - January 2018. - No. 70 (1). - P. 1-30.

39. Li J. DDoS attack detection based on neural network / J. Li, Y. Liu, L. Gu // Proceedings of the 2nd International Symposium on Aware Computing (Tainan, 1-4 November, 2010). - Tainan, 2010. - P. 196-199.

40. Madbouly A. Relevant Feature Selection Model Using Data Mining for Intrusion Detection System / Madbouly, A. // Gody, A. // Barakat, T International Journal of Engineering Trends and Technology (IJETT) - Volume 9 Number 10 - Mar 2014. - P. 12

41. LOIC (Low Orbit Ion Cannon) : A network stress testing application [Electronic resource] - Access mode : http://sourceforge.net/projects/loic/. - Access data : July 2015. - The title of the screen.

42. Mikrovik J. A taxonomy of DDoS attack and DDoS defense mechanisms / J J Mirkovic, P Reiher // ACM SIGCOMM Computer Communication. - 2018. - Режим доступу до ресурсу : https://www.researchgate.net/profile/ Peter_Reiher/publication/2879658_A_taxonomy_of_DDoS_attack_andDDoS_defense _mechanisms/links/02e7e51d1ce0432910000000.pdf.

43. M. Srivatsa / Mitigating application-level denial of service attacks on Web servers: A clienttransparent approach / M. Srivatsa, A. Iyengar, J. Yin, L. Liu // ACM Transactions on the Web. - 2018. - Режим доступу до ресурсу: http://researcher.ibm.com/files/us-aruni/TWEBDos.pdf

44. NSL-KDD dataset [Електронний ресурс]. URL: http://wwvv.unb.ca/cic/research/datasets/nsl.html

45. P. Aggarwal, and S. K. Sharma "Analysis of KDD dataset attributes-class wise for intrusion detection", Procedia Computer Science. vol. 57, pp. 842–851, 2015

46. Sah JJ. Impact of DDOS attacks on cloud environment / JJ Sah, DLJ Malik // IJRCCT. - 2013. - Режим доступу до ресурсу: http://ijrcct.org/index.php/ojs/article/download/276/pdf.

47. Siaterlis C. Detecting incoming and outgoing DDoS attacks at the edge using a single set of network characteristics / C. Siaterlis, V Maglaris // Proceedings of the 10th IEEE Symposium on Computers and Communications (Washington, 27-30 June 2015). Washington, 2015. - P. 469-475.

48. N. S. Safa, R. V. Solms, and S. Furnell, "Information security policy compliance model in organizations", Computers & Security, vol. 56, pp. 70-82, 2016. doi:10.1016/j.cose.2015.10.006.

49. V. Gupta. Denial of service attacks at the MAC layer in wireless ad hoc networks / V. Gupta, S. Krishnamurthy // MILCOM. - 2019. - Режим доступу:https://www.cs.wmich.edu/wise/doc/spins/dos/denial-of-service-attacks.pdf.

50. V.J. Nirmal and D.I.G. Amalarethinam, "Parallel Implementation of Big Data Pre-Processing Algorithms for Sentiment Analysis of Social Networking Data," International Journal of Fuzzy Mathematical Archive, Vol. 6, No. 2, pp.149-159, 2015.

51. Venkatachalam. V. Performance comparison of intrusion detection system classifiers using various feature re daction techniques // SELVAN S Erode Sengunthar Engineering College. 31 March 2015. - P. 19

52. WEKA Classification Algorithms [Електронний ресурс]. URL: http: //wekaclassalgos.sourceforge.net/

53. Y. Alshboul, and K. Streff, “Analyzing Information Security Model for Small-Medium Sized Businesses”, in Proc. 21st Americas Conference on Information Systems, Puerto Rico, 2018.

ДОДАТОК А

(обов’язковий)

Фрагмент коду програмного забезпечення аналізу трафіка

```
#!/usr/bin/python
from __future__ import division
import pandas as pd
from pandas.io import sql
import psycopg2
import base64
import numpy as np
import struct
from socket import inet_ntoa
import time
import datetime
from sklearn import preprocessing
from sklearn import neighbors
from sklearn import svm
from sklearn import tree
from sklearn.naive_bayes import GaussianNB
from sklearn.model_selection import train_test_split
start_time = time.time()
SIZE_OF_HEADER = 24
SIZE_OF_RECORD = 48
ACCUR = 4
PROTOCOL = { 1: "ICMP", 6: "TCP", 14: "Telnet", 17: "UDP" }
# формування словника з інформацією про потоки
def nfdata_new():
 nfdata = {}
 nfdata['flow_count'] = 0
 nfdata['pcount'] = 0
 nfdata['bcount'] = 0
 nfdata['protocol'] = { 'TCP': 0, 'ICMP': 0, 'Telnet': 0, 'UDP': 0 }

 return nfdata

# додавання даних про потік в словник
def nfdata_add(nfdata, flow):
 nfdata['pcount'] += flow['pcount']
 nfdata['bcount'] += flow['bcount']
 nfdata['flow_count'] += 1
 for p in flow['protocol']:
 nfdata['protocol'][p] += flow['protocol'][p]

return nfdata
# групування даних про трафік
```

```
def get_group(flow, t=None):
 if t == None or t == "all":
 return "all"
 s = ""
 for x in t.split('_'):
 if 'd' in x:
 s += "_"
 s += flow[x[0] + "addr"]
 if 'p' in x:
 s += ":" + str(flow[x[0] + "port"])
 return s
# вибірка трафіку з БД
def select(conn, t1=None, t2=None):
 limit = ""
 if t1 != None and t2 != None:
 limit = "WHERE time > %s AND time < %s" % (t1, t2)
 elif t1 != None:
 limit = "WHERE time == %s" % t1
 # формування запиту до БД
 sqlquery = "SELECT * FROM flows" + limit
 data = sql.read_sql(sqlquery, conn, index_col='id')
 return data
# формування DataFrame для обробки
def get_data(conn, t1=None, t2=None, nf_group_type=None):
 data = select(conn, t1, t2)
 result = { }
 print('* get_data() starts ', datetime.datetime.now())
 df = pd.DataFrame(
 columns=(
 'saddr',
 'sport',
 'daddr',
 'dport',
 'pcount',
 'bcount',
 'first',
 'last',
 'duration',
 'bpp',
 'bps',
 'pps',
 'protocol',
 'label',
 )
)
 col_names = list(df)
 for i in range(len(data)):
 nfc_group = { }
 #X-threads:
```

```
 for j in range(len(data.iloc[i]["data"])):
 #1-thread:
 s_buf = data.iloc[i]["data"][j]
 if s_buf == "":
 continue
 buf = s_buf.decode('base64')
 flow_count = struct.unpack('B', buf[3:4])[0]
 for index in xrange(flow_count):
 offset = SIZE_OF_HEADER + (index * SIZE_OF_RECORD)
 flow = { }
 # розбір запису netflow
 if len(buf) - offset > 47:
 # розпакування структури
 d = struct.unpack('!IIIIHH',buf[offset + 16:offset + 36])
flow['saddr'] = inet_ntoa(buf[offset + 0:offset + 4])
flow['sport'] = d[4]
 flow['daddr'] = inet_ntoa(buf[offset + 4:offset + 8])
flow['dport'] = d[5]
 flow['pcount'] = d[0]
flow['bcount'] = d[1]
 flow['protocol'] = { }
flow['protocol'][PROTOCOL[ord(buf[offset + 38])]] = 1
flow['upairs'] = set()
flow['upairs'].add((buf[offset + 0:offset + 4], d[4],
buf[offset + 4:offset + 8], d[5]))
flow['first'] = d[2]
flow['last'] = d[3]
flow['duration'] = d[3] - d[2]
 flow['proto'] = ord(buf[offset + 38])
 flow['bpp'] = round(flow['bcount'] / flow['pcount'], ACCUR)
 flow['bps'] = 0 if not flow['duration'] else
round((flow['bcount'] * 8) / flow['duration'], ACCUR)
flow['pps'] = 0 if not flow['duration'] else
round(flow['pcount'] / flow['duration'], ACCUR)
 flow['label'] = 'benign' if d[3] - d[2] == 0 else
'irc_botnet'
 temp_row = pd.Series(
 [
 flow['saddr'],
flow['sport'],
flow['daddr'],
flow['dport'],
flow['pcount'],
flow['bcount'],
flow['first'],
flow['last'],
flow['pps'],
flow['proto'],
flow['label'],
```

```
 ], index=col_names
 )
 df = df.append(temp_row, ignore_index=True)
s = get_upair_group(flow, nf_group_type)
if s not in nfc_group:
 nfc_group[s] = nfdata_new()
 nfc_group[s] = nfdata_add(nfc_group[s], flow)
 for k in nfc_group:
 nfc_group[k]['ucount'] = len(nfc_group[k]['upairs'])
 del nfc_group[k]['upairs']
 result[data.iloc[i]["time"]] = nfc_group
 # return result
 df.to_csv('~/traf.csv')
 return df
def analyze(df):
 print('analyze', datetime.datetime.now())
 new_df = pd.concat(
 [
 df['pcount'],
 df['bcount'],
 df['duration'],
 df['bpp'],
 df['bps'],
 df['pps'],
df['protocol'],
 df['label']
 ],
 axis=1,
 )
 X = np.array(new_df.drop(['label'], 1))
 y = np.array(new_df['label'])
 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
 gnb = GaussianNB()
 y_pred = gnb.fit(X_test, y_test)
 gnb_accuracy = clf.score(X_test, y_test)
 print('NB Accuracy: {}'.format(gnb_accuracy))
 tree_clf = tree.DecisionTreeClassifier()
 tree_clf.fit(X_train, y_train)
 tree_accuracy = tree_clf.score(X_test, y_test)
 print('Tree Accuracy: {}'.format(tree_accuracy))
 knn_clf = neighbors.KNeighborsClassifier()
 knn_clf.fit(X_train, y_train)
 knn_accuracy = clf.score(X_test, y_test)
 print('KNN Accuracy: {}'.format(knn_accuracy))
 clf = svm.SVC()
 clf.fit(X_train, y_train)
 svm_accuracy = clf.score(X_test, y_test)
 print('SVM Accuracy: {}'.format(svm_accuracy))
```

ДОДАТОК Б

(обов’язковий)

Копії наукових праць

Процес визначення початку атаки типу HTTP GET flood

Соколюк Я.В.
Науковий керівник – к.т.н.,доц. Муляр І.В.
Хмельницький національний університет

Для виявлення початку атаки та подальшого виявлення шкідливого трафіку оптимальним буде підхід, який базується на аналізі аномалій, що призводить до порівняння поточного стану системи з її нормальним станом.

При цьому порівнюються різні властивості мережної активності. Ці властивості контексті DDoS-атак можуть включати: тип та кількість запитів, кількість запитів певного протоколу, IP-адресу джерела, час та швидкість запитів, тощо [1].

Атака типу HTTP GET flood використовується нападниками для атаки веб-серверів та серверів веб-додатків. Атака - це сукупність на перший погляд законних запитів GET або POST до сервера [2]. Це спеціально розроблені запити на споживання значної кількості серверних ресурсів. В результаті вони можуть призвести до стану відмови в обслуговуванні, без необхідності переповнювати канал великим обсягом трафіку. Такі запити у випадку розподіленої атаки DoS надсилаються з десятків тисяч заражених вузлів. На рис. 1 схематично показує послідовність пакетів у запиті HTTP GET після з'єднання TCP.

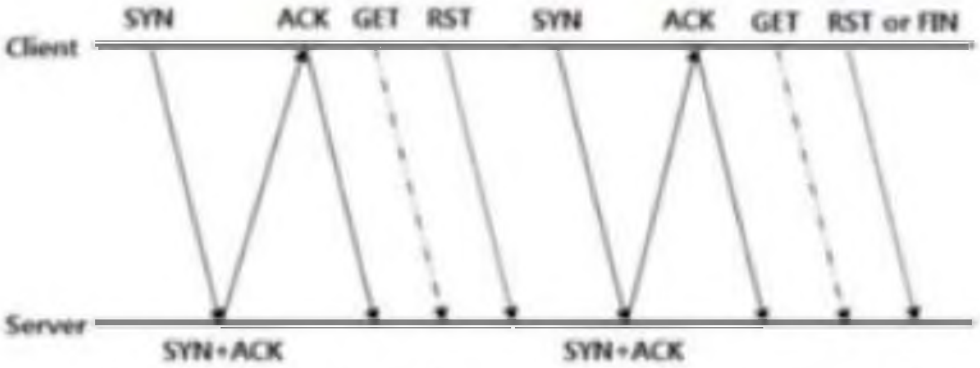


Рисунок 1- Послідовність пакетів при атаці типу HTTP GET

В процесі атаки зловмисник постійно відправляє запити, створюючи при цьому нові TCP з'єднання. Останнім часом [3] також стали розповсюдженими HTTP GET flood атаки в рамках одного TCP з'єднання (див. рис. 2). Цей тип атаки не можливо виявити методом оцінки кількості SYN запитів.

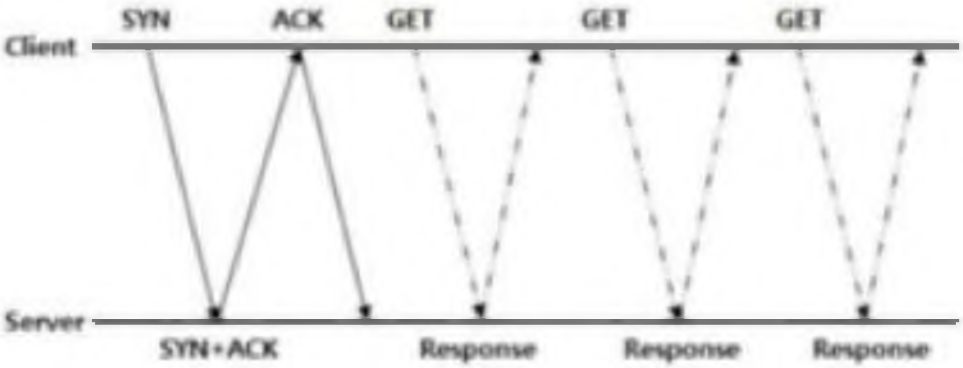


Рисунок 2 - Послідовність пакетів при атаці типу HTTP GET в рамках одного TCP з'єднання

Нині атака HTTP-потоків є однією з найдосконаліших загроз інформаційній безпеці, яка безпосередньо не пов'язана з уразливістю програмного забезпечення. Для обладнання безпеки відрізнити зловмисні HTTP-запити від законних надзвичайно складно, а неправильні методи або налаштування призводять до великої кількості помилкових спрацьовувань. Використання метрик, заснованих лише на оцінці інтенсивності запиту, не є оптимальним методом виявлення DDoS-атак, таких як повеня HTTP, оскільки обсяг трафіку може бути нижче порогового. Тому доцільно використовувати багатокритеріальний метод виявлення DDoS-атак з показниками, які залежать від інтенсивності запитів, і тими, які не залежать від цього показника.

MapReduce - модель проведення розподіленої паралельної обробки великих масивів даних з використанням кластерів (великої кількості обчислюваних блоків). Робота MapReduce складається із двох етапів: Map і Reduce [4].

На етапі Map виконується попередня обробка вхідних даних. Для цього один із обчислювальних елементів кластеру (головний вузол, master node) отримає вхідні дані для розрахунку і розподіляє дані серед робочих вузлів.

На етапі Reduce попередньо оброблені дані обертаються. Основний вузол отримує відповіді від робочих вузлів і на їх основі формує результат - рішення проблеми.

Перевага моделі MapReduce полягає в тому, що вона дозволяє виконувати операції попередньої обробки та згортки паралельно і незалежно, а також горизонтально масштабувати обчислювальну потужність кластера. Операції попередньої обробки діють незалежно одна від одної і можуть виконуватися паралельно (хоча на практиці це обмежується джерелом вхідного сигналу та / або кількістю використовуваних обчислювальних блоків). Аналогічно, група робочих вузлів може конвертувати - для цього потрібно лише те, щоб усі результати попередньої обробки з одним конкретним значенням ключа оброблялися одним робочим вузлом одночасно.

Для роботи багатокритеріального методу виявлення DDoS-атак необхідно провести попередній аналіз та розрахунки: визначити показники (критерії), за якими буде ідентифіковано наявність або відсутність атаки; побудувати модель для звичайного мережевого трафіку; встановити пороги для вибраних показників.

В якості критеріїв оцінки були обрані наступні показники:

- рівень завантаження процесора сервера;
- обсяг зайнятої оперативної пам'яті сервера;
- розмір упаковки;
- поточний рівень трафіку (Мбіт / с);

- розподіл значення адреси джерела запитів (source ip);

- користувач-агент у запиті;

- URI (ієрархічна частина та фрагменти URL-адреси запиту);

Наявність атаки потоку HTTP GET flood може характеризуватися кількістю запитів від джерела за секунду [4]. Зрозуміло, що законний користувач не постійно робить велику кількість запитів на один і той же ресурс, як це може вузол, керований зловмисником (зомбі). Тому деякий час At надходить з IP-адреси джерела x надходить 5. запитів.

Для цього необхідно правильно визначити початкову точку атаки. Це дозволить класифікувати весь попередній трафік як законний та відкриє додаткові можливості для розподілу змішаного трафіку, що надходить після атаки, на законний та шкідливий [5]. У цьому випадку метод виявлення шкідливого трафіку, у першому наближенні, буде зведений до наступних етапів:

1. Визначте поточні сезонні періоди.

2. Беручи до уваги сезонність, визначте початкову точку нападу.

3. Ми відносимо весь попередній трафік до початку атаки до законного.

4. Класифікуємо змішаний трафік на законний та шкідливий.

5. Порівняйте законний трафік, вибраний із змішаного, з трафіком, отриманим до атаки.

6. На основі результатів, отриманих на попередньому кроці, та розроблених критеріїв успіху, скоригуємо вибірки.

7. Весь вхідний трафік аналізується на основі отриманих даних.

Серед основних методів можна виділити методи, що базуються на статистичному аналізі. Це допомагає оцінювати різні параметри мережної активності і діагностувати початок атаки або визначати шкідливий трафік.

Основними параметрами, за якими проводиться аналіз, можуть бути:

- Кількість запитів за певний період.
- Швидкість надходження запитів.
- Кількість запитів з певного джерела або з певною мережі.
- Кількість запитів до певного пункту призначення (для вебсервера це конкретний скрипт).
- Час між запитами.
- Інші різні параметри мережевої активності.

За допомогою середньоквадратичного відхилення можна розрахувати допустиму межу для одного з параметрів мережевої активності, наприклад, для кількості запитів за якийсь період часу. У разі якщо межа буде порушена, це стане свідченням початку атаки. Так як в різний час навантаження на мережевий ресурс, так само може бути різною, то для раннього виявлення атаки необхідний постійний моніторинг і перерахунок кордонів для кожного тимчасового кроку. Постійний моніторинг дозволить

визначити атаку, якщо вона почнеться в період невеликої мережевої активності, або, якщо зловмисник шукає потенційно вразливі місця на сервері, проводячи міні- DDOS-атаки і вивчаючи поведінки сервера. У разі якщо верхня межа задана строго і зловмисник проводить міні-атаки в період найменшої мережевої активності, він може не порушувати задану кордон, і його дії будуть не виявлені. Атака буде виявлена тоді, коли зловмисник знайде потенційно вразливе місце, і зробить на нього атаку. Постійний моніторинг активності і перерахунок допустимих меж дозволяє цього уникнути. У період меншою мережевий активності верхня межа знизиться.

Перелік посилань

1. Долішний В.С. Аналіз і моніторинг сучасних DDoS - атак / В.С. Долішний, В.М. Чешун. - Тези доповідей Всеукраїнської науково-практичної конференції молодих вчених, ад'юнктів, слухачів, курсантів і студентів "Молодіжна військова наука у Київському національному університеті імені Тараса Шевченка" [Текст] / за заг. редакцією І.В. Толока. – К. : ВІКНУ, 2018. – С. 147 - 148.

2. DDoS Definitions - DdoSPedia, [Електронний ресурс]. http://security.radware.com/knowledge- center/DDoSPedia/http-flood

3. Zinchenko, V. V, Zinchenko, M. V (2017), Viyavlennya ddos-atak prikladnogo rivnya [Detection of application layer DDoS attacks], Mizhnarodna naukovo-tehnichna konferentsiya «Radiotehnichni polya, signali, aparati ta sistemi», Kyiv, pp. 262-264.

4. Системи і методи виявлення вторгнень: сучасний стан і напрями вдосконалення [Електронний ресурс]. URL:http://citfomm.ni/security/ intemet/ids_overview/#3

5. Холявка Є. П.: Метод виявлення мережевих атак в комп'ютеризованих системах управління: наукова робота, Хмельницький національний університет. - Хмельницький, 2019, [Електронний ресурс]. URL: http://konkurs.khnu.km.ua/wp-content/uploads/sites/25/2019/04/ DP3Eugen.pdf

АНАЛІЗ І МОНІТОРИНГ СУЧАСНИХ DDoS – АТАК

***Муляр І.В.**. (ХмНУ),*
***Соколюк Я.В.** (ХмНУ)*

DDoS-атака - розподілена атака, спрямована на відмову в обслуговуванні. Атаки такого типу можуть швидко виснажити мережеві ресурси або потужності сервера, що призведе до неможливості отримати доступ до ресурсу і викличе серію негативних наслідків: втрачений прибуток, неможливість скористатися послугами і зробити різні транзакції і т.д. У DDoS-атаці в ролі атакуючого виступає так звана бот-мережа, або зомбі-мережа. Зомбі-мережа може налічувати від декількох десятків до тисяч хостів. Зазвичай це нейтральні комп'ютери, які в силу якихось причин (відсутність файрволу, застарілі бази антивіруса і т.д.) були заражені, шкідливими програмами. Програми, працюючи у фоновому режимі, безперервно посилають запити на атакуємий сервер, виводячи його таким чином з ладу. На даний момент не існує якогось універсального засобу для протидії DDoS-атакам. Навіть такі великі компанії, як Microsoft, eBay, Amazon, Yahoo, страждають від DDoS-атак і не завжди можуть з ними впоратися.

В результаті проведеного аналізу відмічено, що в даний час значно збільшилася кількість DDoS-атак середньої і малої інтенсивності, спрямованих, як правило, на регіональні ресурси. Це збільшення цілком прогнозовано - з розвитком мережі збільшується потенційна кількість можливих жертв. Крім того, вдосконалюється сам механізм проведення атак. Для зловмисника проведення атаки вже не є настільки складним. А зомбі-комп'ютери намагаються емулювати дії самих користувачів. Все це веде до загального збільшення числа атак.

Аналіз засобів протидії показав, що в даний час більший розвиток отримала група засобів протидії, призначена для відбиття потужних атак. У цю групу входять, як правило, дорогі засоби, призначені для великих провайдерів або компаній. Засоби протидії невеликим і середнім атакам, що базуються на сервер, що атакується, представлені в незначній кількості. Це пов'язано з незначною кількістю таких атак в минулому.

При цьому аналіз вхідного трафіку на рівні додатків може бути більш ефективним. З одного боку, проведення такого аналізу економічно затратно, з іншого - може бути цілком достатнім для відображення малих і середніх атак, тенденція домінування яких вже намітилася.

Оптимальним рішенням для виявлення початку атаки і подальшого виявлення шкідливого трафіку буде рішення, засноване на аналізу аномалій, в результаті якого відбувається порівняння поточного стану системи з її нормальним станом. Порівняння станів системи в контексті DDoS-атак можна проводити шляхом порівняння різних властивостей мережевої активності. До цих властивостей можуть бути віднесені: кількість запитів, тип запитів, кількість запитів певного типу або протоколу, IP адреса джерела, швидкість надходження запитів, їх час і т.д

ДОДАТОК В

(обов’язковий)

Презентація

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Соколюк Ярослав Вікторович

МЕТОД ВИЯВЛЕННЯ РОЗПОДІЛЕНИХ АТАК НА ІНФОРМАЦІЙНУ СИСТЕМУ

Науковий керівник
к.т.н., доцент Муляр І.В.

кафедра кібербезпеки та комп'ютерних систем і мереж

Тема Метод виявлення розподілених атак на інформаційну систему

Метою магістерської роботи є розроблення методу для раннього виявлення DDoS атак, і подальшого блокування шкідливого трафіку на стороні ресурсу, що атакується

Об'єкт дослідження: розподілені атаки на інформаційну систему

Предметом дослідження: є моделі та методи виявлення DDoS атак, і виділення зловмисного трафіку

Задачі досліджень у роботі формулюються наступним чином:

1. Проаналізовано сучасний стан DDoS атак, та проведено аналіз нинішнього стану технологій для вирішення проблем захисту інформації

2. Розроблено математичну модель атак з врахуванням опису сезонності мережного навантаження.

3 Розроблено алгоритм визначення точок початку атаки та алгоритм поділу змішаного трафіку на надійний та шкідливий, який враховує сезонну кількість мережного навантаження.

4. Вирішено завдання по створенню методу і програмного комплексу по виявленню DDoS-атакам малої інтенсивності.

5. Розроблено програмний засіб для виявлення початку атаки та подальшого виявлення та блокування нелегітимних запитів. Його особливістю є модульність та універсальність для забезпечення безпеки різних мережних ресурсів та захисту їх від атак різного типу.

Наукова новизна роботи полягає в:

1. Розроблено математичну модель атак, яка враховує опис сезонності мережного трафіку для різної періодичності

2. Вдосконалено метод раннього виявлення та протидії DDoS атакам, особливостями якого є врахування сезонних періодів

. Методи дослідження, використані в магістерській роботі: апарат теорії алгоритмів, теорії захисту інформації, системного аналізу, теорії імовірності та математичної статистики, кластерного і системного аналізу.

Практична цінність полягає у створенні методу та алгоритмів захисту мережних ресурсів від DDoS-атак, що дозволяють проводити активну протидію безпосередньо на стороні атакованого ресурсу. Це підтверджується розробкою та подальшою реалізацією розробленого програмного пакету для виявлення DDoS-атак, та подальше блокування нелегітимних звернень на різних рівнях.

Публікації. По темі магістерської роботи опубліковано 1 стаття у нефаховому журналі (збірник НПК МНІС ІП-2020), 1 - теза доповідей на всеукраїнській конференції (Тези доповідей XVI Міжнародної науково-практичної конференції "Військоваосвіта і наука: сьогодення та майбутнє" [Текст] / за заг. редакцією Ігоря Толока.– К. : ВІКНУ, 2020)

Класифікація DDoS- атак

За кількістю задіяних пристроїв
- Прості
- Групові
- Масовані

За приналежністю джерел до зловмисника
- Добровільні
- З використанням бот-мереж
  - З використанням серверів
  - З використанням домашніх комп'ютерів
  - З використанням мобільних пристроїв
- З використанням виділених серверів
- З використанням засобів тунелювання та проміжних машин
- З використанням випадкових користувачів

За складом джерел
- Сттичні
- Динамічні
  - З керуванням
  - Без керування

За засобом керування
- Ручні
- Керовані
  - З прямим керуванням
    - Випадкове сканування
    - Сканування за списком
    - Зворотній зв'язок
  - З непрямим керуванням
- Автоматичні

За типом вразливості
- Семантичні
- Flood

За коректністю адреси джерела
- З вказаннм коректного джерела
- З вказанням некоректного джерела
- Реверс-атаки

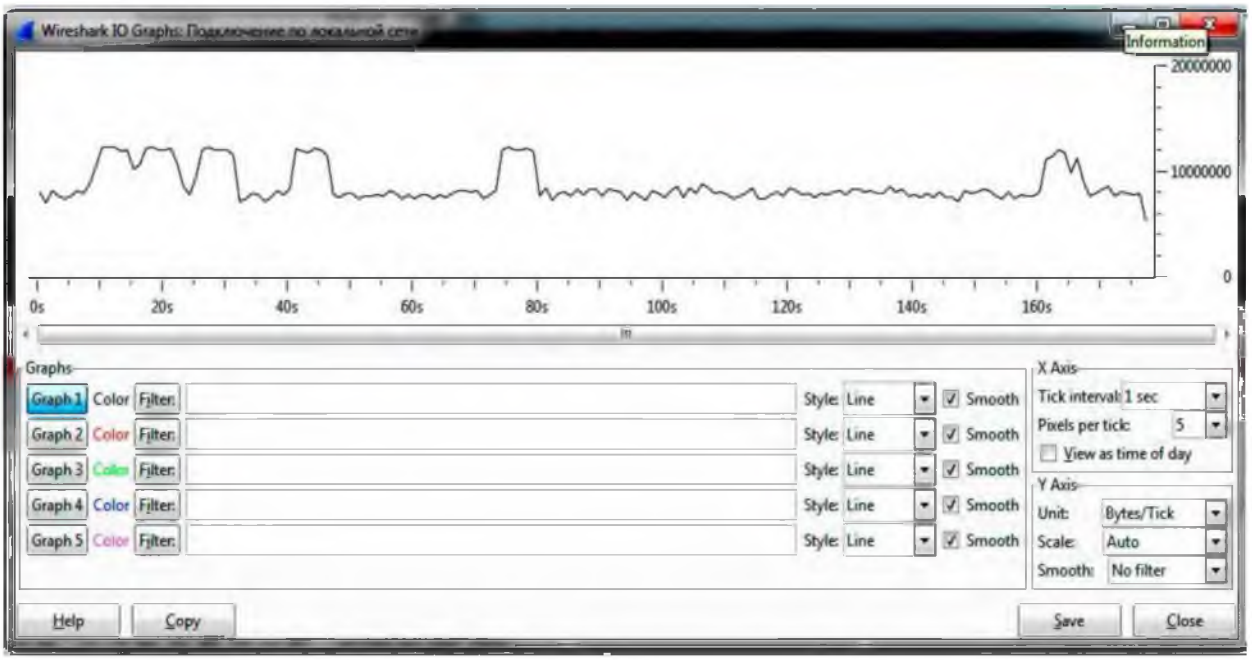
За типом вразливості
- Атака на протокол
- Атака на реалізацію

За динаміою потужності
- Постійна потужність
- Випадкова змінна потужність
- Визначена змінна потужність
- Потужність, що коливається
- Потужність, що збільшується

За рівнем реалізації
- Фізичний рівень
- Канальний рівень
- Мережний рівень
- Транспортний рівень
- Сеансовий рівень
- Прикладний рівень
- Рівень сервісів

За впливом
- Атака, що блокує доступ
  - З можливістю відновлення
  - Без можливості відновлення
- Атака, що збільшує споживання ресурсів
- Атака, що приводить до руйнування

Графічне представлення повільної DDoS-атаки



Перший науковий результат

Модель виявлення початку атаки статистичними методами з урахуванням сезонності

$A\left(a_1, a_2, a_3, ..., a_n\right)$ - множина всіх можливих властивостей для всіх мережних клієнтів. $B\left(b_1, b_2, b_3, ..., b_m\right)$ - множина легальних клієнтів конкретного мережного ресурсу. клієнт b_1 має властивості $A1\left(a_4, a_8, a_{10}, a_{14}\right)$,

клієнт b_2 має властивості $A2\left(a_3, a_8, a_{11}, a_{14}\right)$ і т.д.

Для формування ковзаючої оцінки використаємл середньоквадратичне відхилення:

$$\sigma = \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} \left(x_i - \bar{x}\right)^2}$$

де σ - середньоквадратичне відхилення; n - кількість часових діапазанов; x_i - кількість звернень за певний період; $\bar{x}$ - середнє арифметичне звернень по всіх періодах.

Звернення до запишемо у вигляді матриці:

$$x_{11},\ x_{12},\ x_{13}, \dots, x_{124}$$
$$x_{21},\ x_{22},\ x_{23}, \dots, x_{224}$$
$$\dots\dots\dots\dots\dots\dots\dots\dots$$
$$x_{n1},\ x_{n2},\ x_{n3}, \dots, x_{n24}$$

Кожен рядок матриці містить добові дані про кількість звернень. Перший рядок містить дані поточної доби, томі він може бути заповнений не до кінця. Розрахунок середньоквадратичного відхилення в цьому випадку може проводитися двома способами.

Розрахунок середньоквадратичного відхилення в цьому випадку може проводитися двома способами:

1. Звичайним способом - з урахуванням певного числа останніх значень, наприклад, так:

$$x_{21},\ x_{22},\ x_{23}, \dots, x_{224}, x_{11},\ x_{12},\ x_{13}, x_{14}$$

Значення беруться з ***рядків*** матриці.

2. З урахуванням критерію сезонності - значення беруться зі ***стовбців*** матриці.

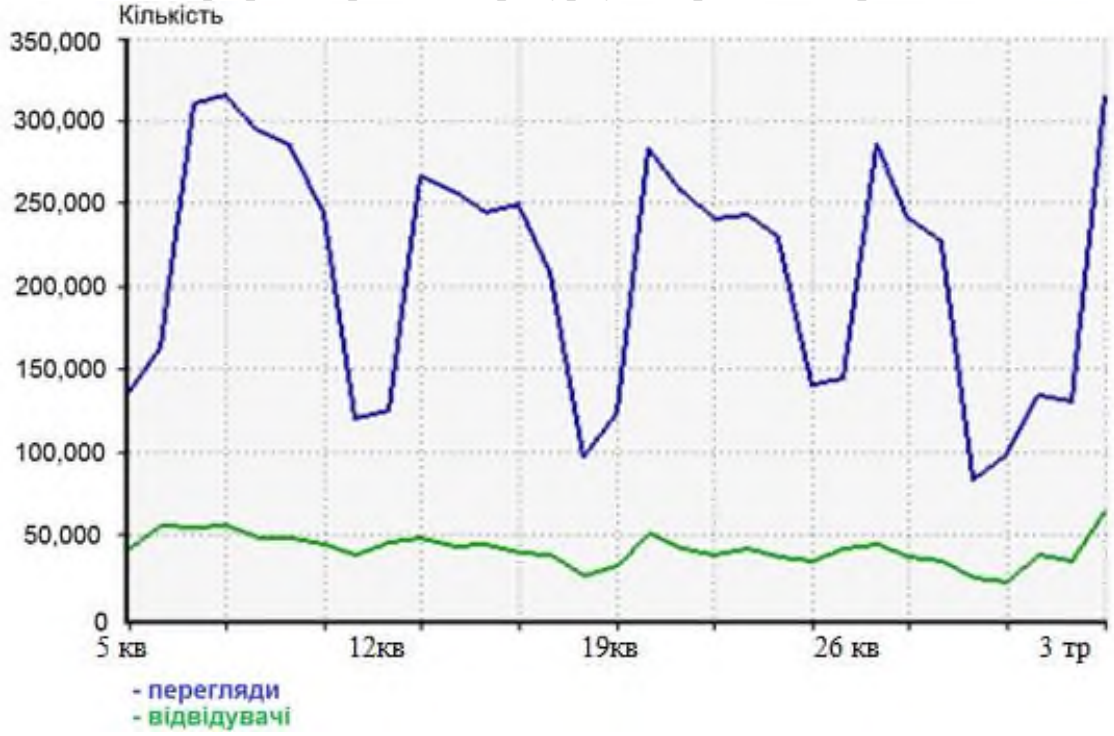
$$x_{n1}, \dots, x_{21},\ x_{11}$$

Такий підхід дозволяє підвищити точність формування границі, порушення якої буде ідентифікувати можливість початку DDOS-атаки, і відповідно зменшити час на її фіксацію.

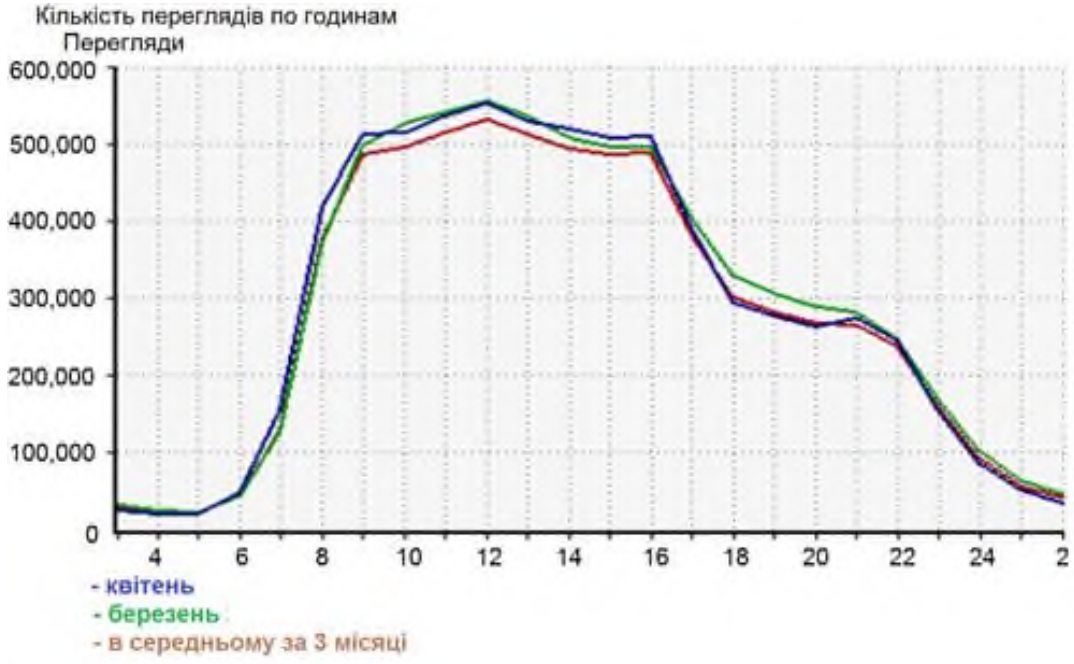
Другий науковий результат

Метод раннього виявлення та протидії DDoS атакам, особливостями якого є врахування сезонних періодів

Графік звернень до ресурсу «Українська правда»



Графік звернень до ресурсу «Українська правда» по годинах



В результаті дослідження було встановлено наявність сезонних періодів в роботі вебресурсів.

Другий науковий результат

Моделювання процесу вибору актуальних сезонних періодів на базі методу Херста

r_1	r_2	r_3	r_4	r_5	r_6	r_7	…	r_{T-3}	r_{T-2}	r_{T-1}	
$\overline{r_1^2}, S_1^2, R_1^2$ R_1^2 / S_1^2		$\overline{r_2^2}, S_2^2, R_2^2$ R_2^2 / S_2^2		$\overline{r_3^2}, S_3^2, R_3^2$ R_3^2 / S_3^2		…			$\overline{r_{T_2}^2}, S_{T_2}^2, R_{T_2}^2$ $R_{T_2}^2 / S_{T_2}^2$		$(R/S)_2 = \frac{\sum_{i=1}^{T_2} R_i^2 / S_i^2}{T_2}$
$\overline{r_1^3}, S_1^3, R_1^3$ R_1^3 / S_1^3			$\overline{r_2^3}, S_2^3, R_2^3$ R_2^3 / S_2^3					$\overline{r_{T_3}^3}, S_{T_3}^3, R_{T_3}^3$ $R_{T_3}^3 / S_{T_3}^3$			$(R/S)_3 = \frac{\sum_{i=1}^{T_3} R_i^3 / S_i^3}{T_3}$
…											
$\overline{r_1^N}, S_1^N, R_1^N$ R_1^N / S_1^N			…					$\overline{r_{T_N}^N}, S_{T_N}^N, R_{T_N}^N$ $R_{T_N}^N / S_{T_N}^N$			$(R/S)_N = \frac{\sum_{i=1}^{T_N} R_i^N / S_i^N}{T_N}$

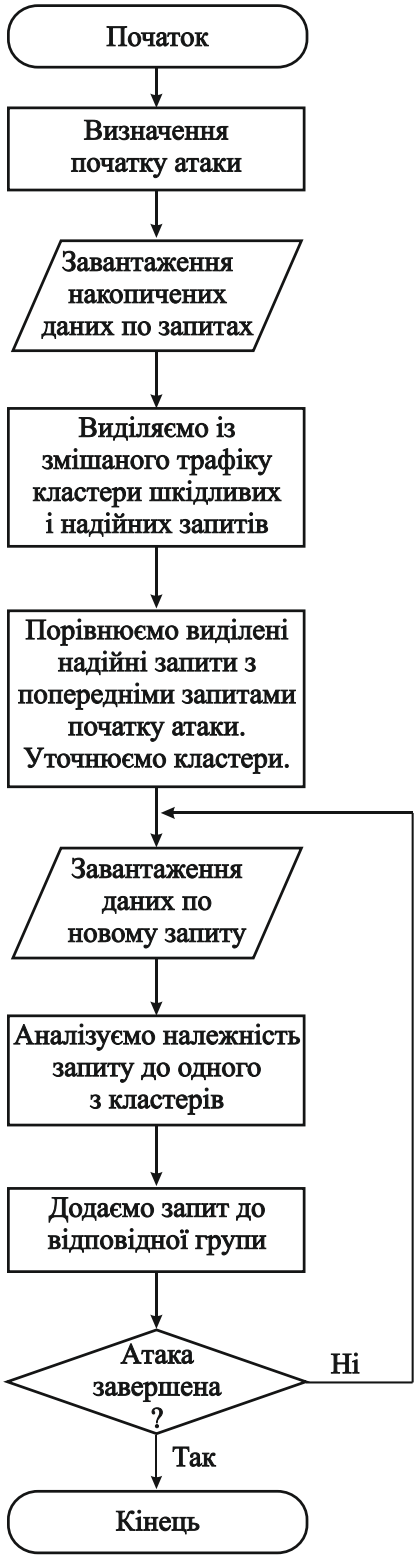
де $\overline{r_j}$ - середнє значення, S_j^2 - дисперсія значень, $X(t,i) = \sum_{i=1}^{t}\left(r_i - \overline{r_j}\right)$, $t < j$ - адитивне відхилення, $R(j) = \max_{1 \le t \le j} X(t,i) - \min_{1 \le t \le j} X(t,i)$ -розмах, $T_n = \left[(T-1)/n\right]$- кількість блоків, при $n = 2...\left[T/2\right]$. Для кожного значення t формується та лінійно апроксимується графік залежності $Ln(R/S)_n$ від Ln (n).

графік залежності $Ln(R/S)_n$ від Ln (n). Графік лінійно апроксимується.

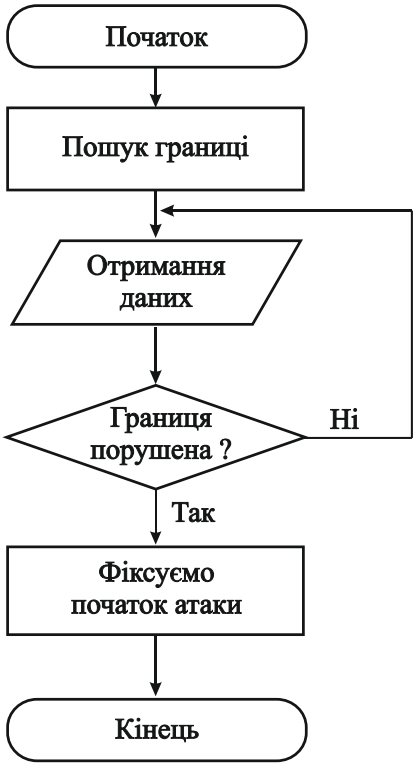
Метод дає можливість виявляти сезонні періоди в умовах невизначеності

Другий науковий результат

Алгоритми по визначенню початку атаки і виділенню шкідливого трафіку



а)



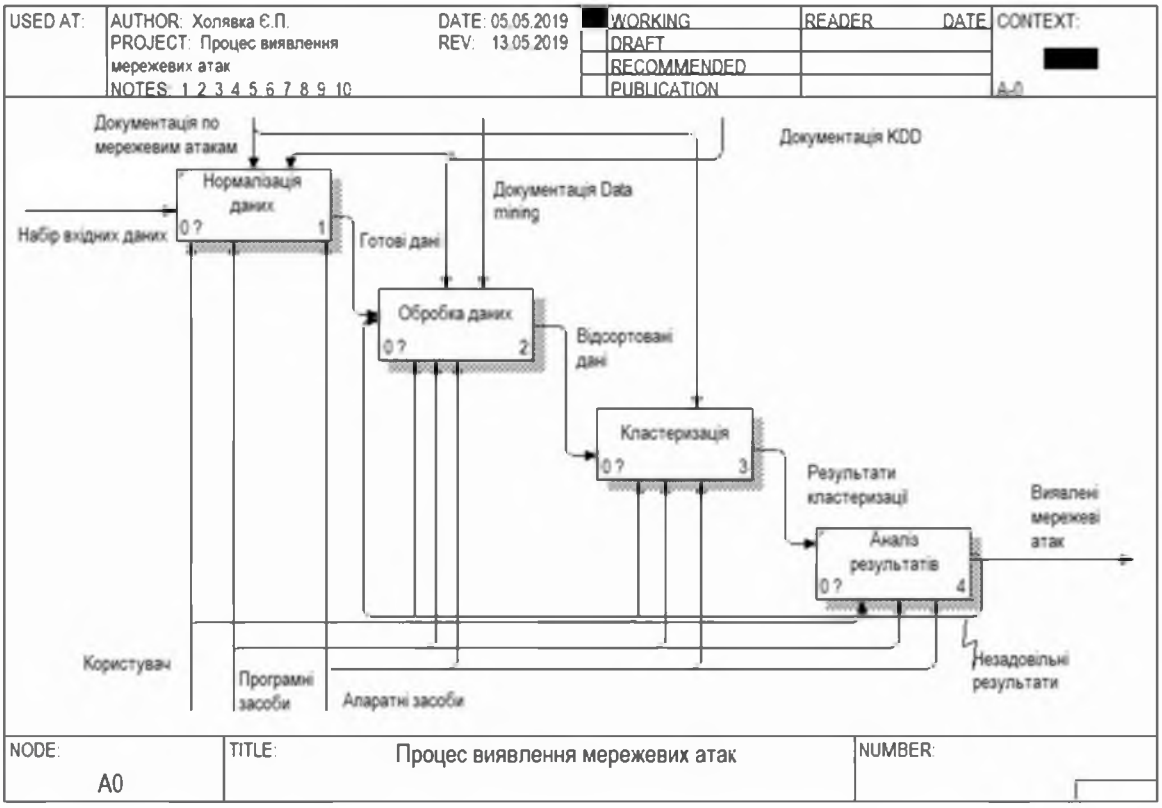
б)

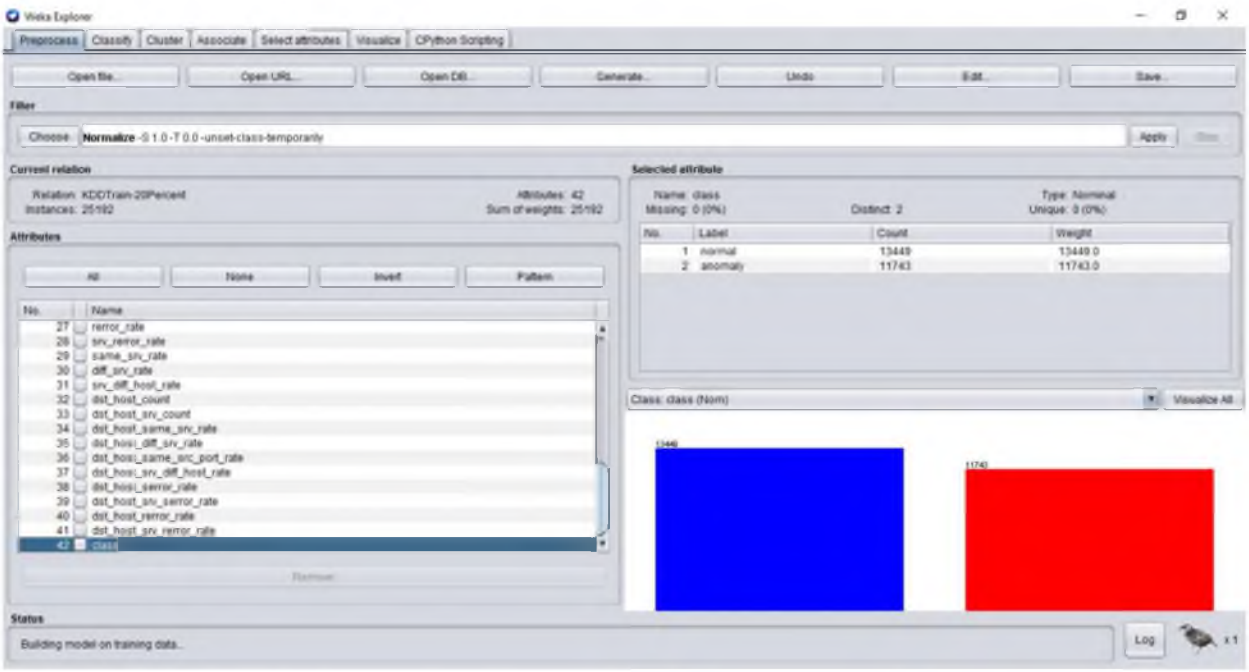
в)

Блок схеми алгоритмів по визначенню початку атаки і виділенню загрозливого трафіку (а - виділення загрозливого трафіку, б і в – ідентифікація початку розподіленої атаки).

Відмінною рисою алгоритмів є облік сезонних коливань мережного навантаження.

Функціональна модель інформаційної системи





Отримання даних для кластеризації

ВИСНОВКИ

У магістерській роботі вирішено наукове завдання раннього виявлення початку DDoS-атаки і подальшого визначення загрозливих запитів.

Основними результатами дослідження є:

1. Проаналізовано сучасний стан DDoS атак, та запропоновано їх класифікацію. Проведено аналіз нинішнього стану технологій для вирішення проблем захисту інформації, проаналізовано методи, що застосовуються для рішення задач пов'язаних з доступністю ресурсівю. Виділено новий вид атак малої інтенсивності, спрямовану на регіональні ресурси. Виявлено відсутність засобів, протидії для даної групи.

2. Розроблено математичну модель атак з врахуванням опису сезонності мережного навантаження.

3 Використовуючи модель атаки розроблено метод раннього виявлення та протидії DDoS-атакам малої інтенсивності.

5. Розроблено алгоритм визначення точок початку атаки та алгоритм поділу змішаного трафіку на надійний та шкідливий, який враховує сезонну кількість мережного навантаження.

6. Запропоновано критеріїв успішності, які дозволяють оцінювати успішність роботи алгоритму роботи, і сторонні засоби за фільтрацією трафіку.

7. Описано інформаційну систему та розроблено її функціональну модель. Розроблено діаграмау варіантів використання інформаційної системи виявлення мережних атак

8. Розроблено програмний засіб для виявлення початку атаки та подальшого виявлення та блокування нелегітимних запитів. Його особливістю є модульність та універсальність для забезпечення безпеки різних мережних ресурсів та захисту їх від атак різного типу.

9. Проведено ряд тестів, які підтверджують ефективність та результативність розробленого програмного забезпечення для протидії DDoS-атакам та їх виявлення.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Соколюк Ярослав Вікторович

МЕТОД ВИЯВЛЕННЯ РОЗПОДІЛЕНИХ АТАК НА ІНФОРМАЦІЙНУ СИСТЕМУ

Науковий керівник
к.т.н., доцент Муляр І.В.

кафедра кібербезпеки та комп'ютерних систем і мереж

Тема Метод виявлення розподілених атак на інформаційну систему

Метою магістерської роботи є розроблення методу для раннього виявлення DDoS атак, і подальшого блокування шкідливого трафіку на стороні ресурсу, що атакується

Об'єкт дослідження: розподілені атаки на інформаційну систему

Предметом дослідження: є моделі та методи виявлення DDoS атак, і виділення зловмисного трафіку

Задачі досліджень у роботі формулюються наступним чином:

1. Проаналізовано сучасний стан DDoS атак, та проведено аналіз нинішнього стану технологій для вирішення проблем захисту інформації

2. Розроблено математичну модель атак з врахуванням опису сезонності мережного навантаження.

3 Розроблено алгоритм визначення точок початку атаки та алгоритм поділу змішаного трафіку на надійний та шкідливий, який враховує сезонну кількість мережного навантаження.

4. Вирішено завдання по створенню методу і програмного комплексу по виявленню DDoS-атакам малої інтенсивності.

5. Розроблено програмний засіб для виявлення початку атаки та подальшого виявлення та блокування нелегітимних запитів. Його особливістю є модульність та універсальність для забезпечення безпеки різних мережних ресурсів та захисту їх від атак різного типу.

Наукова новизна роботи полягає в:

1. Розроблено математичну модель атак, яка враховує опис сезонності мережного трафіку для різної періодичності

2. Вдосконалено метод раннього виявлення та протидії DDoS атакам, особливостями якого є врахування сезонних періодів

. Методи дослідження, використані в магістерській роботі: апарат теорії алгоритмів, теорії захисту інформації, системного аналізу, теорії імовірності та математичної статистики, кластерного і системного аналізу.

Практична цінність полягає у створенні методу та алгоритмів захисту мережних ресурсів від DDoS-атак, що дозволяють проводити активну протидію безпосередньо на стороні атакованого ресурсу. Це підтверджується розробкою та подальшою реалізацією розробленого програмного пакету для виявлення DDoS-атак, та подальше блокування нелегітимних звернень на різних рівнях.

Публікації. По темі магістерської роботи опубліковано 1 стаття у нефаховому журналі (збірник НПК МНІС ІП-2020), 1 - теза доповідей на всеукраїнській конференції (Тези доповідей XVI Міжнародної науково-практичної конференції "Військоваосвіта і наука: сьогодення та майбутнє" [Текст] / за заг. редакцією Ігоря Толока.– К. : ВІКНУ, 2020)

Класифікація DDoS- атак

За кількістю задіяних пристроїв
- Прості
- Групові
- Масовані

За приналежністю джерел до зловмисника
- Добровільні
- З використанням бот-мереж
  - З використанням серверів
  - З використанням домашніх комп'ютерів
  - З використанням мобільних пристроїв
- З використанням виділених серверів
- З використанням засобів тунелювання та проміжних машин
- З використанням випадкових користувачів

За складом джерел
- Сттичні
- Динамічні
  - З керуванням
  - Без керування

За засобом керування
- Ручні
- Керовані
  - З прямим керуванням
    - Випадкове сканування
    - Сканування за списком
    - Зворотній зв'язок
  - З непрямим керуванням
- Автоматичні

За типом вразливості
- Семантичні
- Flood

За коректністю адреси джерела
- З вказаннм коректного джерела
- З вказанням некоректного джерела
- Реверс-атаки

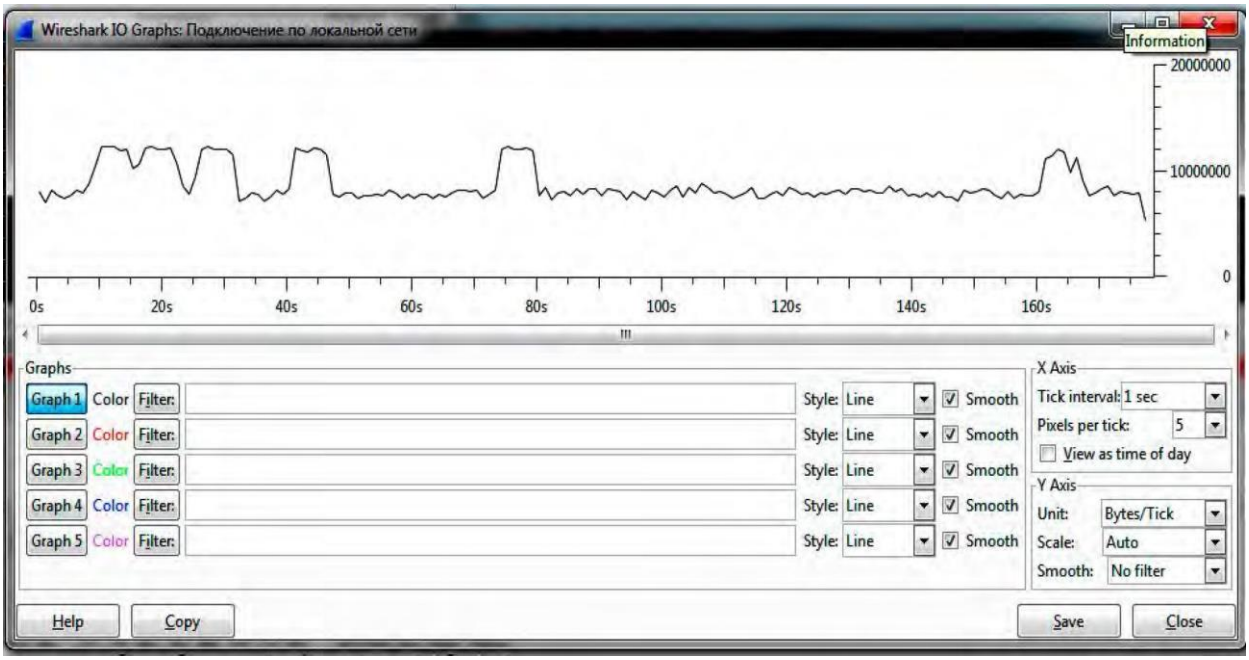
За типом вразливості
- Атака на протокол
- Атака на реалізацію

За динаміою потужності
- Постійна потужність
- Випадкова змінна потужність
- Визначена змінна потужність
- Потужність, що коливається
- Потужність, що збільшується

За рівнем реалізації
- Фізичний рівень
- Канальний рівень
- Мережний рівень
- Транспортний рівень
- Сеансовий рівень
- Прикладний рівень
- Рівень сервісів

За впливом
- Атака, що блокує доступ
  - З можливістю відновлення
  - Без можливості відновлення
- Атака, що збільшує споживання ресурсів
- Атака, що приводить до руйнування

Графічне представлення повільної DDoS-атаки



Перший науковий результат

Модель виявлення початку атаки статистичними методами з урахуванням сезонності

$A\left(a_1, a_2, a_3, ..., a_n\right)$ - множина всіх можливих властивостей для всіх мережних клієнтів. $B\left(b_1, b_2, b_3, ..., b_m\right)$ - множина легальних клієнтів конкретного мережного ресурсу. клієнт b_1 має властивості $A1\left(a_4, a_8, a_{10}, a_{14}\right)$,

клієнт b_2 має властивості $A2\left(a_3, a_8, a_{11}, a_{14}\right)$ і т.д.

Для формування ковзаючої оцінки використаємл середньоквадратичне відхилення:

$$\sigma = \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} \left(x_i - \bar{x}\right)^2}$$

де σ - середньоквадратичне відхилення; n - кількість часових діапазанов; x_i - кількість звернень за певний період; $\bar{x}$ - середнє арифметичне звернень по всіх періодах.

Звернення до запишемо у вигляді матриці:

$$x_{11},\ x_{12},\ x_{13}, \dots, x_{124}$$
$$x_{21},\ x_{22},\ x_{23}, \dots, x_{224}$$
$$\dots\dots\dots\dots\dots\dots\dots\dots$$
$$x_{n1},\ x_{n2},\ x_{n3}, \dots, x_{n24}$$

Кожен рядок матриці містить добові дані про кількість звернень. Перший рядок містить дані поточної доби, томі він може бути заповнений не до кінця. Розрахунок середньоквадратичного відхилення в цьому випадку може проводитися двома способами.

Розрахунок середньоквадратичного відхилення в цьому випадку може проводитися двома способами:

1. Звичайним способом - з урахуванням певного числа останніх значень, наприклад, так:

$$x_{21},\ x_{22},\ x_{23}, \dots, x_{224}, x_{11},\ x_{12},\ x_{13}, x_{14}$$

Значення беруться з ***рядків*** матриці.

2. З урахуванням критерію сезонності - значення беруться зі ***стовбців*** матриці.

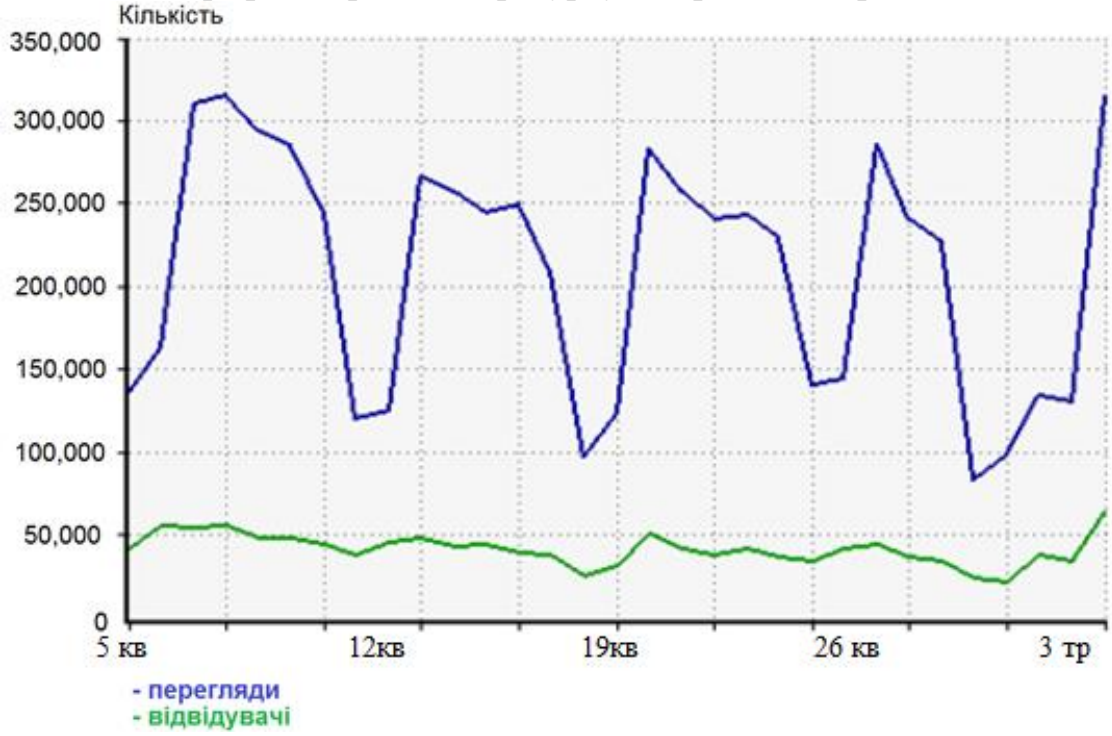
$$x_{n1}, \dots, x_{21},\ x_{11}$$

Такий підхід дозволяє підвищити точність формування границі, порушення якої буде ідентифікувати можливість початку DDOS-атаки, і відповідно зменшити час на її фіксацію.

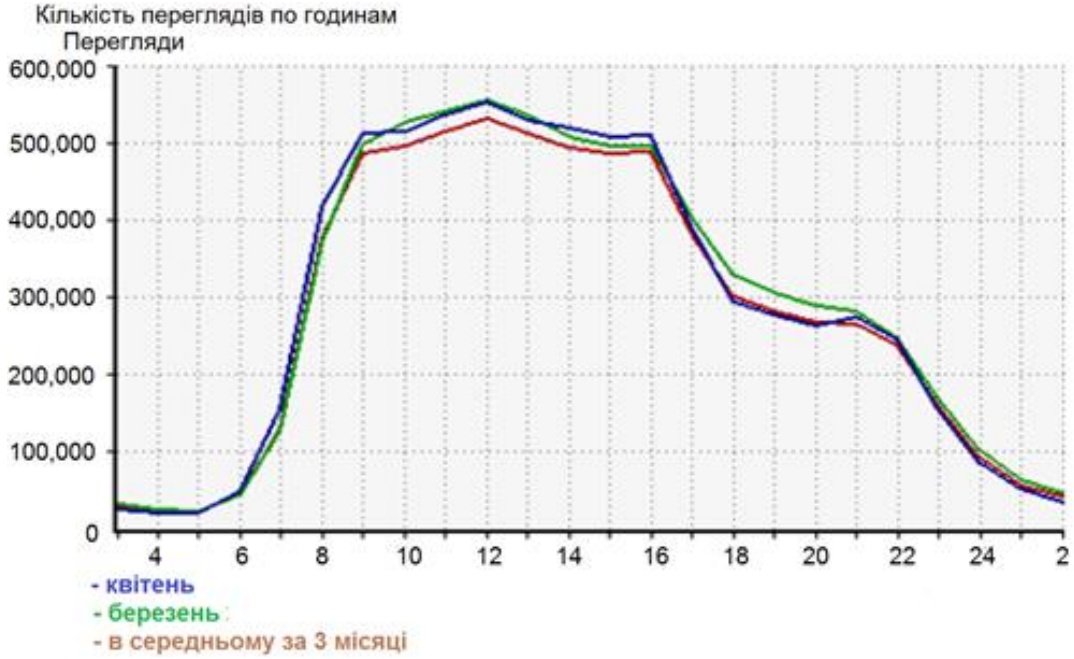
Другий науковий результат

Метод раннього виявлення та протидії DDoS атакам, особливостями якого є врахування сезонних періодів

Графік звернень до ресурсу «Українська правда»



Графік звернень до ресурсу «Українська правда» по годинах



В результаті дослідження було встановлено наявність сезонних періодів в роботі вебресурсів.

Другий науковий результат

Моделювання процесу вибору актуальних сезонних періодів на базі методу Херста

r_1	r_2	r_3	r_4	r_5	r_6	r_7	…	r_{T-3}	r_{T-2}	r_{T-1}	
$\overline{r_1^2}, S_1^2, R_1^2$ R_1^2 / S_1^2		$\overline{r_2^2}, S_2^2, R_2^2$ R_2^2 / S_2^2		$\overline{r_3^2}, S_3^2, R_3^2$ R_3^2 / S_3^2		…			$\overline{r_{T_2}^2}, S_{T_2}^2, R_{T_2}^2$ $R_{T_2}^2 / S_{T_2}^2$		$(R/S)_2 = \frac{\sum_{i=1}^{T_2} R_i^2 / S_i^2}{T_2}$
$\overline{r_1^3}, S_1^3, R_1^3$ R_1^3 / S_1^3			$\overline{r_2^3}, S_2^3, R_2^3$ R_2^3 / S_2^3					$\overline{r_{T_3}^3}, S_{T_3}^3, R_{T_3}^3$ $R_{T_3}^3 / S_{T_3}^3$			$(R/S)_3 = \frac{\sum_{i=1}^{T_3} R_i^3 / S_i^3}{T_3}$
…											
$\overline{r_1^N}, S_1^N, R_1^N$ R_1^N / S_1^N			…					$\overline{r_{T_N}^N}, S_{T_N}^N, R_{T_N}^N$ $R_{T_N}^N / S_{T_N}^N$			$(R/S)_N = \frac{\sum_{i=1}^{T_N} R_i^N / S_i^N}{T_N}$

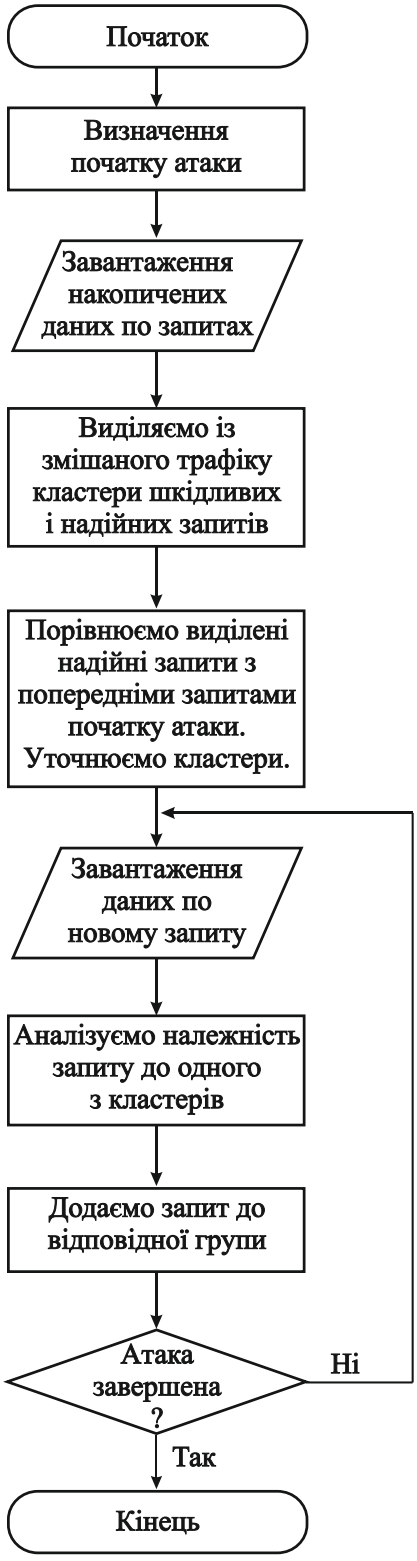
де $\overline{r_j}$ - середнє значення, S_j^2 - дисперсія значень, $X(t,i) = \sum_{i=1}^{t}\left(r_i - \overline{r_j}\right)$, $t < j$ - адитивне відхилення, $R(j) = \max_{1 \le t \le j} X(t,i) - \min_{1 \le t \le j} X(t,i)$ -розмах, $T_n = \left[(T-1)/n\right]$- кількість блоків, при $n = 2...\left[T/2\right]$. Для кожного значення t формується та лінійно апроксимується графік залежності $Ln(R/S)_n$ від Ln (n).

графік залежності $Ln(R/S)_n$ від Ln (n). Графік лінійно апроксимується.

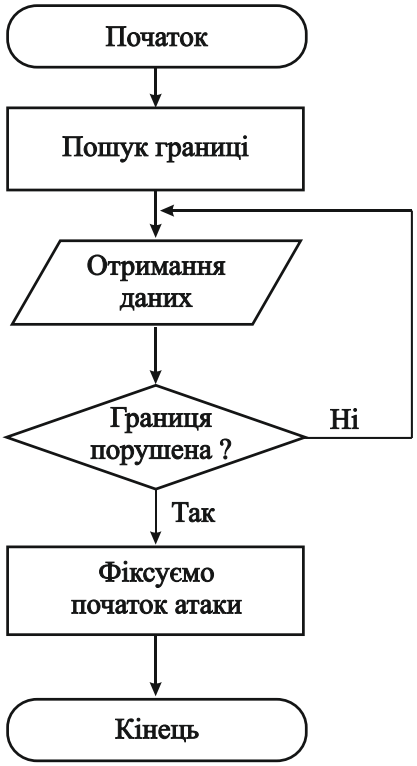
Метод дає можливість виявляти сезонні періоди в умовах невизначеності

Другий науковий результат

Алгоритми по визначенню початку атаки і виділенню шкідливого трафіку



а)



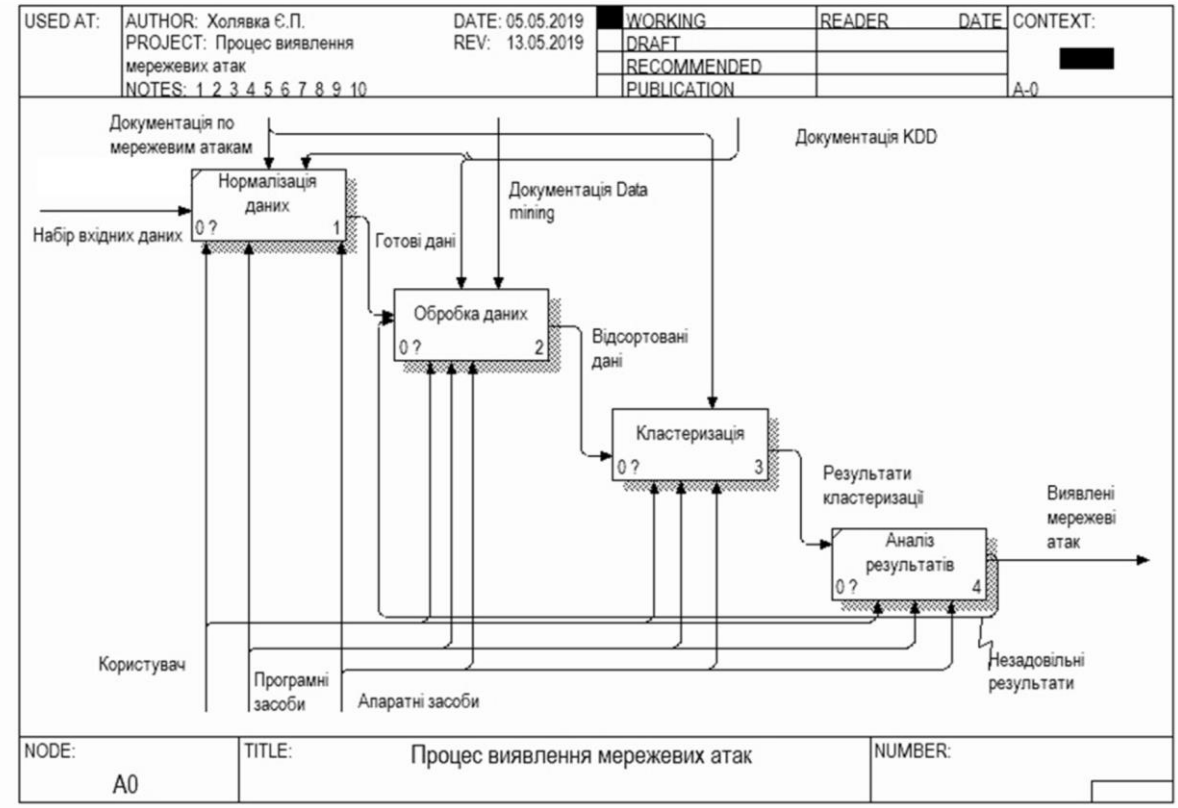
б)

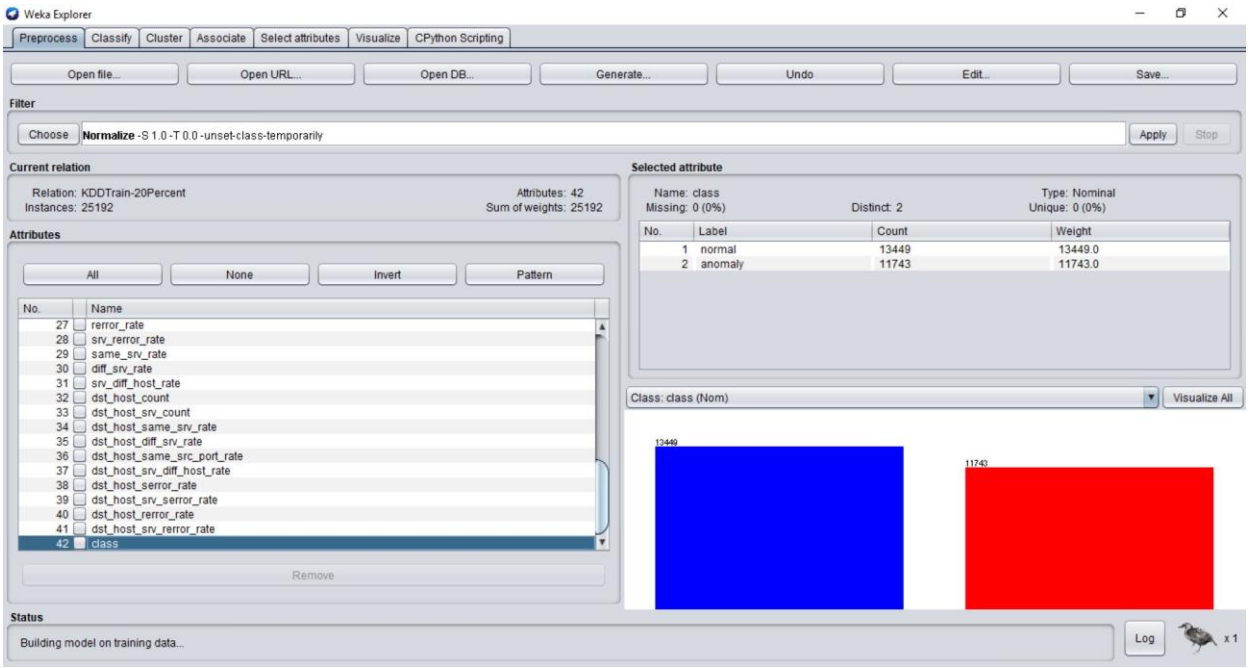
в)

Блок схеми алгоритмів по визначенню початку атаки і виділенню загрозливого трафіку (а - виділення загрозливого трафіку, б і в – ідентифікація початку розподіленої атаки).

Відмінною рисою алгоритмів є облік сезонних коливань мережного навантаження.

Функціональна модель інформаційної системи





Отримання даних для кластеризації

ВИСНОВКИ

У магістерській роботі вирішено наукове завдання раннього виявлення початку DDoS-атаки і подальшого визначення загрозливих запитів.

Основними результатами дослідження є:

1. Проаналізовано сучасний стан DDoS атак, та запропоновано їх класифікацію. Проведено аналіз нинішнього стану технологій для вирішення проблем захисту інформації, проаналізовано методи, що застосовуються для рішення задач пов'язаних з доступністю ресурсівю. Виділено новий вид атак малої інтенсивності, спрямовану на регіональні ресурси. Виявлено відсутність засобів, протидії для даної групи.

2. Розроблено математичну модель атак з врахуванням опису сезонності мережного навантаження.

3 Використовуючи модель атаки розроблено метод раннього виявлення та протидії DDoS-атакам малої інтенсивності.

5. Розроблено алгоритм визначення точок початку атаки та алгоритм поділу змішаного трафіку на надійний та шкідливий, який враховує сезонну кількість мережного навантаження.

6. Запропоновано критеріїв успішності, які дозволяють оцінювати успішність роботи алгоритму роботи, і сторонні засоби за фільтрацією трафіку.

7. Описано інформаційну систему та розроблено її функціональну модель. Розроблено діаграмау варіантів використання інформаційної системи виявлення мережних атак

8. Розроблено програмний засіб для виявлення початку атаки та подальшого виявлення та блокування нелегітимних запитів. Його особливістю є модульність та універсальність для забезпечення безпеки різних мережних ресурсів та захисту їх від атак різного типу.

9. Проведено ряд тестів, які підтверджують ефективність та результативність розробленого програмного забезпечення для протидії DDoS-атакам та їх виявлення.

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КІБЕРБЕЗПЕКИ ТА КОМП'ЮТЕРНИХ СИСТЕМ І МЕРЕЖ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:
Назва: Метод виявлення розподілених атак на інформаційну систему

Автор: Соколюк Ярослав Вікторович
Спеціальність: 123 – Комп'ютерна інженерія
Освітня програма: Програмування та захист комп`ютерних систем і мереж
Науковий керівник: Муляр Ігор Володимирович, к.т.н., доцент
Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних). Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи.	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
5	Інше:	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
3) окремі виявлені збіги є загальновживаними фразами або виразами, про що свідчить посилання системи на збіг з 15-30 джерелами на один фрагмент речення;
4) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості, складає 6.54% і адресується до 389 першоджерела, що, з урахуванням наведених обгрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи І.В. Муляр

Завідувач кафедри КбКСМ, гарант ОП Ю.П. Кльоц

Дата: 08.12.2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

освітнього ступеня «магістр»

Магістр Соколюк Я.В.

Тема Метод виявлення розподілених атак на інформаційну систему

Галузь знань 12 – Інформаційні технології

Спеціальність 123 –Комп'ютерна інженерія

Обсяг кваліфікаційної роботи освітньо-кваліфікаційного рівня «магістр»:

кількість листів креслень 9; кількість сторінок записки 93

1. Короткий зміст кваліфікаційної роботи та прийнятих рішень в рамках роботи Запропонований метод виявлення та блокування шкідливого трафіку DDoS-атак на ранніх стадіях базується на аналізі сезоності мережного трафіку. Розроблено алгоритм визначення точок початку атаки та алгоритм поділу змішаного трафіку на надійний та загрозливий, який враховує сезонну кількість мережного навантаження. Розроблено програмний засіб для виявлення початку атаки та подальшого виявлення та блокування нелегітимних звернень. Його особливістю є модульність та універсальність для забезпечення безпеки різних мережних ресурсів та захисту їх від атак різного типу.

2. Висновок про відповідність кваліфікаційної роботи завданню Кваліфікаційна робота ОС «магістр» у повній мірі відповідає поставленому завданню як в теоретичній, так і в практичній частині роботи.

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У вступі висвітлюється актуальність теми роботи, дається аналіз досліджуваної проблеми і обгрунтовується застосовуваний підхід до її вирішення, формулюються цілі і завдання дослідження, описується наукова новизна і практична значимість отриманих результатів. У першому розділі проведено аналіз розподілених атак, та методів боротьби з ними. Наступні розділи присвячені розробці математичної моделі атак, яка враховує опис сезонності мережного трафіку для різної періодичності та вдосконалено метод раннього виявлення та протидії DDoS атакам, особливостями якого є врахування сезонних періодів Розглянуто питання застосування розробленого методу.

4. Позитивні сторони роботи Кваліфікаційна робота містить ряд інноваційних рішень, зокрема запропонований метод дозволяє дозволяють проводити активну протидію атакам безпосередньо на стороні атакованого ресурсу

5. Негативні сторони роботи Запропонований метод раннього визначення та протидії DDoS-атакам доцільно використовувати тільки при атаках малої інтенсивності

6. Оцінка графічного оформлення та пояснювальної записки роботи Графічне оформлення виконане відповідно до теми кваліфікаційної роботи з дотриманням стандартів. В загальному графічне оформлення виконане якісно. Пояснювальна записка відповідає нормам для її оформлення.

7. Відгук про роботу в цілому В загальному кваліфікаційної роботи заслуговує позитивної оцінки. Весь матеріал дипломної роботи структурований, чіткий та послідовний. Усі розділи роботи послідовні та логічні, що дозволяє чітко розуміти викладений матеріал в рамках тематики кваліфікаційної роботи. Графічний матеріал дозволяє наочно побачити доцільність та ефективність рішень, які були прийняті за основу для досягнення поставленої задачі

8. Інші зауваження

9. Оцінка кваліфікаційної роботи Розглянувши позитивні та негативні сторони представленої кваліфікаційної роботи, можна зробити висновок, що вона заслуговує оцінку „добре" В (4,50)

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи) проф. д. ф.-м. н. Бебронюк К.П., зав. кафедри ІПЗ

« 8 » 12 2020. (підпис)

ХНУ. Зам. 54, тир. 6000, 2007.

UNICHECK

User name:
Кафедра кибербезпеки

Check ID:
1005406216

Check date:
08.12.2020 21:50:22 EET

Check type:
Doc vs Internet

Report date:
08.12.2020 21:52:21 EET

User ID:
100005590

File name: **ПЗ_Сококлюк_ю**

Page count: **85** Word count: **15659** Character count: **115893** File size: **7.85 MB** File ID: **1005698081**

6.54% Matches

Highest match: **2.48%** with Internet source (**https://ela.kpi.ua/bitstream/123456789/15772/1/8.pdf**)

6.54% Internet sources	389	..... Page 87

No Library search was conducted

0% Quotes

Exclusion of quotes is off

Exclusion of references is off

0% Exclusions

No exclusions

Modifind

Text modifications detected. Find more details in the online report.

Replaced characters	1