ІНФОКОМУНІКАЦІЙНІ ТЕХНОЛОГІЇ ТА ЕЛЕКТРОННА ІНЖЕНЕРІЯ
INFORMATION AND COMMUNICATION TECHNOLOGIES,
ELECTRONIC ENGINEERING

№ 2 (1), 2022

ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ

https://doi.org/10.23939/ictee2022.01.001

ІНТЕЛЕКТУАЛЬНА СИСТЕМА МОНІТОРИНГУ ТА АНАЛІЗУ ТРАФІКУ ДЛЯ ВИЯВЛЕННЯ АТАК У ПРОГРАМНО-КОНФІГУРОВАНИХ МЕРЕЖАХ

М. Бешлей, А. Прислупський, М. Медвецький, Г. Бешлей

Національний університет "Львівська політехніка", вул. С. Бандери 12, Львів, 79013, Україна

Відповідальний за рукопис: М. Бешлей (e-mail: mykola.i.beshlei@lpnu.ua)

(Подано 31 липня 2022)

Сьогодні завдання моніторингу, безпеки, управління, пріоритезації трафіку й оптимізації загального навантаження на мережеву інфраструктуру стає вкрай важливим не тільки для корпоративного сектору, а й для операторів мобільного та фіксованого зв'язку. Поточні тенденції у сфері послуг інтернет-контенту вказують, що вимоги до управління трафіком непередбачувані. Тому постачальники послуг повинні реалізовувати нові алгоритми моніторингу та аналізу мережевого трафіку для виявлення атак у сучасних мережах, де програмне забезпечення може бути оновлено для підтримки мінливих вимог до управління і контролю трафіку в міру необхідності. Саме тому ми розробили унікальний алгоритм моніторингу та аналізу мережевого трафіку, що дав змогу підвищити ефективність використання стандартних систем виявлення та запобігання вторгненню завдяки застосуванню інтелектуальних мультифрактальних процесів аналізу вхідного трафіку. Експериментально доведено, що використання розроблених алгоритмів у комунікаційній інфраструктурі дає можливість в умовах наявності шкідливого трафіку в каналах зв'язку зменшити втрати даних до 5 %.

Ключові слова: *SDN; мережеві аномалії; параметр Херста; DPI; виявлення вторгнень.*
УДК: 621.391

1. Вступ

Виявлення мережевих аномалій нині є одним із напрямів дослідження у галузі забезпечення кібербезпеки [1–3], які активно розвиваються. Це пов'язано з тим, що аномалії здебільшого є початковою стадією мережевих атак, що може спричиняти як негативні нематеріальні наслідки, так і фінансові втрати для організацій. Розуміння природи аномалій мережевого трафіку є важливим завданням і для традиційних інфокомунікаційних мереж, і для майбутніх автоматизованих програмно-конфігурованих інтенційно-орієнтованих мереж. Незалежно від того, чи є ці аномалії шкідливими, важливо проаналізувати їх з двох причин:

- аномалії можуть спричинити перевантаження мережі та збільшити використання ресурсів мережевих вузлів;

© 2022 Національний університет "Львівська політехніка"

- деякі аномалії не обов'язково впливають на мережу, але можуть істотно впливати на якість надання сервісу кінцевому користувачеві.

Істотною проблемою виявлення аномалій є те, що форми аномалій можуть змінюватися залежно від причин їх появи, зокрема, створювані від простіших DoS-атак до неправильних конфігурацій адміністратора. Відмова в обслуговуванні, зловмисний контроль, зловмисна робота, сканування, шпигунство та неправильне налаштування – це такі атаки та аномалії, які можуть призвести до виходу із ладу всієї мережевої інфраструктури.

Виявлення та класифікація аномалій передбачають постійний процес моніторингу подій в інформаційних системах та мережах. Саме для цього використовують більшість систем DPI. Відомі сьогодні рішення для виявлення мережевих аномалій перешкоджають розробленню єдиного універсального механізму виявлення раніше не відомих типів атак. DPI (Deep Packet Inspection) – це комерційні продукти, які дають змогу аналізувати трафік на наявність аномалій та загроз у реальному часі. Проте обмежувальними факторами використання таких систем є висока вартість і закрита архітектура, що ускладнює їх адаптацію до організаційної інфраструктури. Використання сучасних методів та систем виявлення аномалій дає змогу посилити політику безпеки та надати більшу гнучкість експлуатації мережевих ресурсів. Але будь-яка система має певні переваги, а також деякі недоліки. Аналіз роботи цих засобів показує, що більшість із них виконують лише одну або кілька специфічних функцій, які не можуть забезпечити тією чи іншою мірою складності захисту інформації, необхідної для майбутніх інфокомунікаційних мереж. Також, згідно із [4–7], встановлено, що наявні DPI системи потребують нових інтелектуальних алгоритмів виявлення аномалій та атак для забезпечення необхідного рівня якості обслуговування та безпеки в перспективних програмно-конфігурованих мережах. Для ефективнішого виявлення нових атак у роботі [8] запропоновано модель виявлення аномалії із використанням вектора показника Херста та мультифрактального спектра. Показано, що мультифрактальний аналіз визначає чутливість до будь-яких відхилень властивостей мережевого трафіку в результаті аномалій. Запропоновані методи аналізу трафіку можуть бути ідеальними для захисту критично важливих даних та підтримки безперервності роботи інтернет-послуг, зокрема IoT. У результаті аналізування методів і систем виявлення аномалій виявлено, що лише завдяки поєднанню декількох методів і алгоритмів у одній системі DPI вдається досягти ефективного захисту та забезпечити протидію загрозам.

Саме тому ми розробили власну унікальну програмно-орієнтовану систему DPI, з можливістю розгортати у ній свої унікальні алгоритми управління трафіком, виявлення аномалій, розпізнавання та блокування мережевих атак, що детально описано нижче.

2. Розроблення алгоритму моніторингу та аналізу трафіку для автоматизованого виявлення аномалії та запобігання атакам у програмно-конфігурованих інтенційно-орієнтованих мережах

Виявлення та класифікація аномалій передбачають безперервний процес моніторингу подій в інформаційних системах і мережах, у зв'язку з чим необхідне оброблення великих обсягів даних, що генерують ці джерела. Одне із основних обмежень сучасних систем полягає у тому, що вони потребують тривалого навчання на великих наборах даних, що не завжди можливо в реальних робочих програмах. Для цього використовують автоматизовані системи виявлення вторгнень у системі DPI. DPI поєднує в собі функціональність системи виявлення вторгнень (IDS) та системи запобігання вторгненням (IPS) із традиційним брандмауером.

Один з основних алгоритмів, використовуваних у розробленій DPI-платформі, – перевірка сигнатур протоколів і додатків. Під сигнатурою розуміють шаблон опису даних, який вибирають для унікальної ідентифікації пов'язаного із ним програми/протоколу. Розроблена DPI-платформа зберігає бібліотеку сигнатур, яка поповнюється у разі появи нових версій або додатків. Крім сигнатурного методу, використовують і аналіз мережевого трафіку, який теж може мати специфічні для кожного з додатків і протоколів характеристики (розмір корисного навантаження, кількість і

розміри пакетів у відповідь на запит, позиція фіксованих рядків або байт всередині пакета тощо). В розробленому арсеналі DPI [9] є методи, основані на статистичному і поведінковому аспектах потоку даних й інші евристичні методи. Зрозуміло, що вилучення інформації з пакета і її аналіз потребують значних обчислювальних ресурсів, а одна із основних вимог до DPI-платформ для програмно-конфігурованих інтенційно-орієнтованих мереж – виконувати сканування пакетів на швидкості каналу передавання даних та гнучкість розгортання на різних сегментах мережі, що показано на рис. 1.

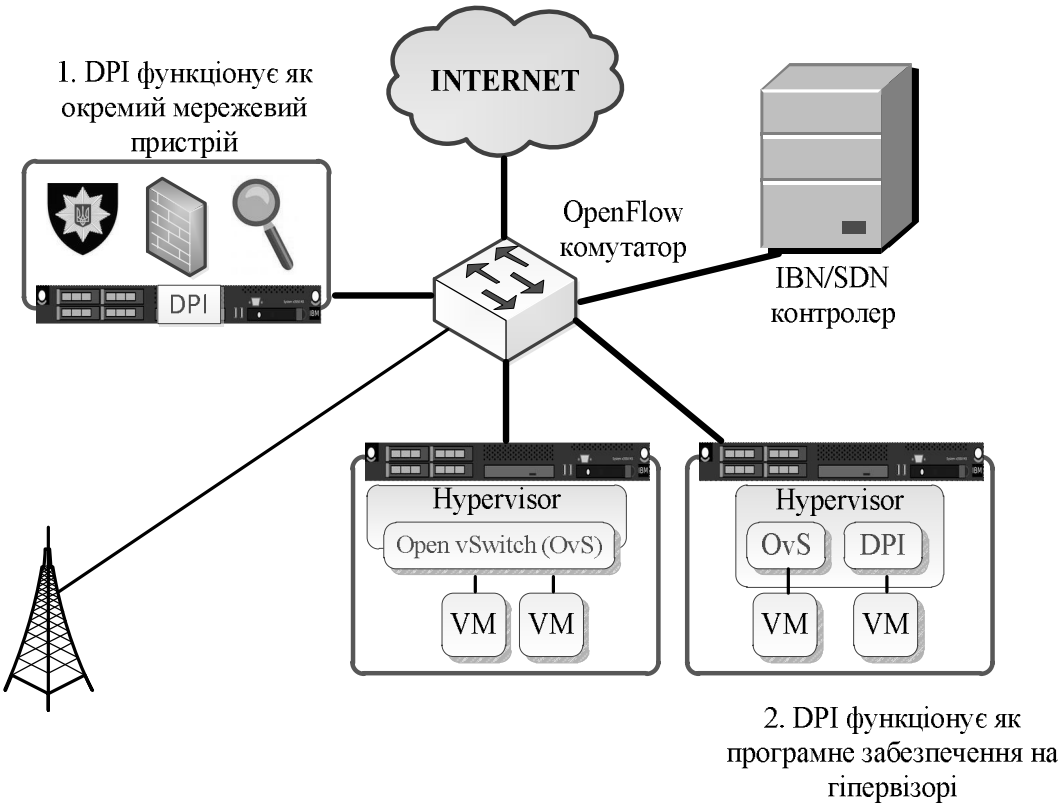

Рис. 1. Гнучкість розгортання пропонованої програмної DPI системи для програмно-конфігурованих мереж

У роботі запропоновано алгоритм для DPI системи (рис. 2), оснований на глибокому навчанні, що дає змогу вивчати адекватну поведінку мережі та виявляти будь-які аномалії без тривалого навчання на великих наборах даних. Після навчання нейронна мережа запускається у режимі розпізнавання. У ситуації, коли вхідний потік не розпізнає нормальну поведінку, атака реєструється. Отже, поєднавши дві різні нейронні мережі, можна визначити та розпізнати інформаційні атаки із доволі високим ступенем точності [10]. Основні переваги використання підходів, основаних на нейронних мережах, – можливість адаптації до динамічних умов та показників продуктивності, що особливо важливо, коли система працює у режимі реального часу Ключовою особливістю запропонованої IDS є оптимальне вилучення сигнатур для зменшення обчислювальної складності та покращення виявлення раніше не відомих атак.

Запропонований алгоритм може бути реалізований або як самостійне рішення, або як частина складнішої IDS з іншими моделями глибокого навчання та DPI. Продуктивність запропонованої моделі перевіряється на різних наборах даних із погляду розподілених атак відмови в обслуговуванні (DDoS). Зазвичай рішення на основі глибокого навчання для виявлення атак DDoS ґрунтуються на контрольованому навчанні великого набору даних відомих атак зі звичайним мережевим трафіком. Ці набори даних складаються із вхідних параметрів, які передаються у глибоку нейронну мережу, а вихідні мітки націлюються, щоб модель могла дізнатися будь-які приховані залежності між входом і виводом.

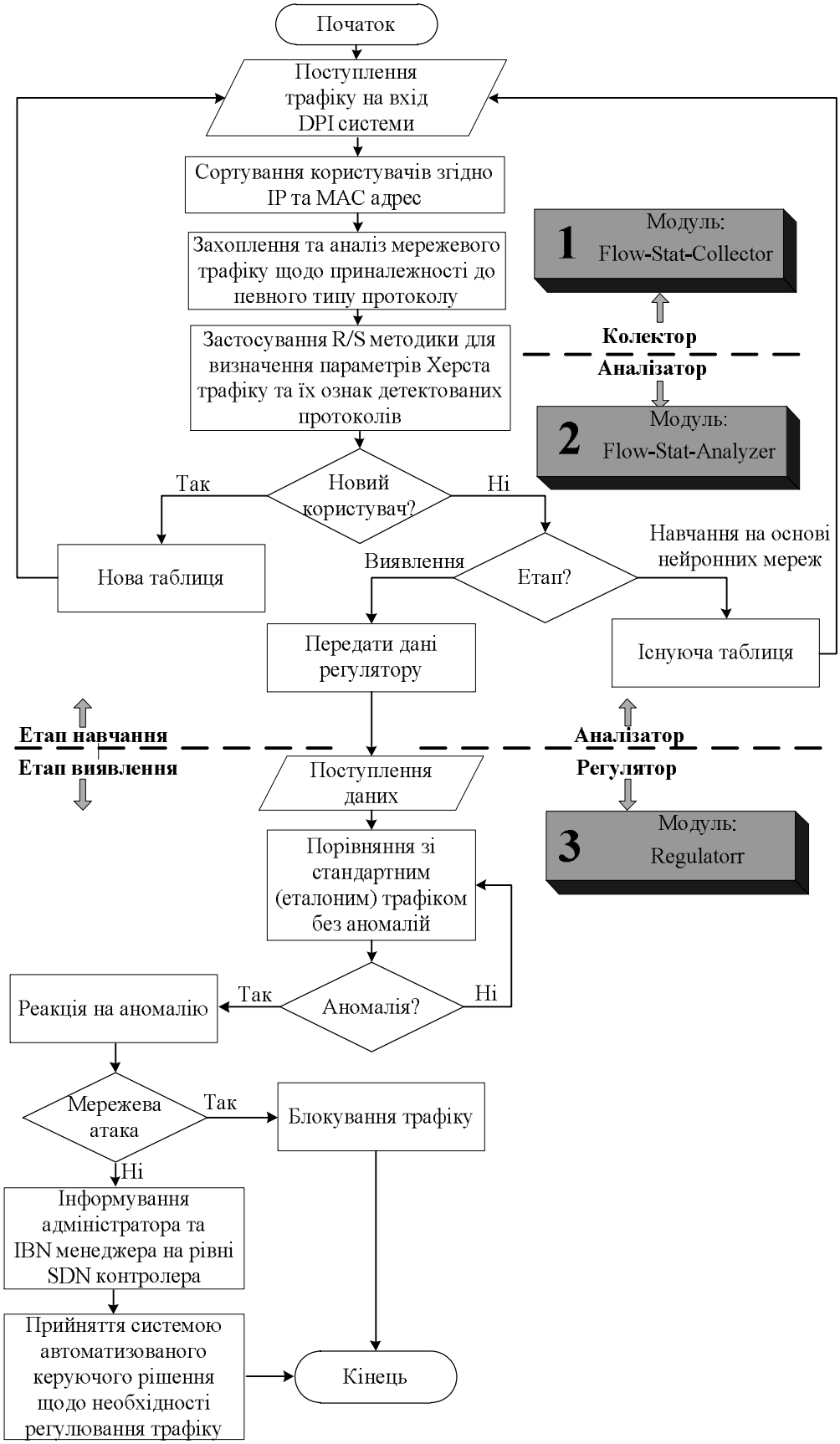

Рис. 2. Алгоритм роботи системи виявлення та блокування вторгнень на основі глибокого навчання для інтелектуальних програмно-конфігурованих мереж

Під час навчання нейронна мережа оцінює різні вхідні характеристики, зокрема параметр Херста трафіку, в результаті чого прогнозує вихідні значення. Потім вихідні значення порівнюються з цільовими значеннями й обчислюється відповідна функція втрат. З кожною ітерацією процесу навчання (епохи) вага між нейронами оновлюється, щоб зменшити функцію втрат. Цей процес навчання триває доти, доки функція втрат не буде зведена до прийнятного порога.

Запропонований алгоритм DPI системи складається із трьох модулів, розміщених у контролері інтелектуальної мережі. Ці три модулі описано нижче:

1. Модуль Flow-Stat-Collector відповідає за збирання статистичних даних про потоки із таблиць потоків комутаторів OpenFlow і зберігає їх у базі даних статистики потоку (FSD). FSD зберігає статистику активних потоків, таких як кількість пакетів, кількість байтів, тривалість потоків, IP-адреси джерела-призначення та номери портів, і періодично оновлює їх новою статистикою потоку. Ці статистичні дані потоку можна використати для виявлення потенційних DDoS-атак. Модуль Flow-Stat-Collector збирає дані із комутаторів через захищений канал і надсилає їх до модуля Flow-Stat-Analyzer.

2. Модуль Flow-Stat-Analyzer запускає алгоритм виявлення статистичної аномалії та виконує аналіз зібраних даних, що зберігаються у базі даних FSD. Це найважливіший крок, оскільки контролеру потрібно з'ясувати, чи є атака, чи це законний трафік. Цей модуль додатково розділений на два компоненти. Перший компонент Analyzer 1 видає попередження про підозрілий трафік, коли інтенсивність трафіку перевищує поріг контролю. Подальший аналіз виконує другий компонентний Аналізатор 2, щоб з'ясувати, чи відповідає цей шаблон трафіку законному трафіку або DDoS-атаці. У такому разі може відбуватися навчання та порівняння еталона за певними сигнатурами. Якщо здійснюється атака, Flow-Stat-Analyzer видає попередження модулю Regulator Attack.

3. Регулятор атаки негайно вживає заходів після отримання попередження. Regulator формує правила OpenFlow до комутаторів на основі типу атаки. Ці правила можуть варіюватися від зниження трафіку атаки до дозволу законного трафіку з певними IP-адресами джерела або призначення.

Для прикладу розглянемо режим виявлення атак типу SYN. Механізм виявлення визначає закономірності трафіку аномалій як ознаки атак за параметром Херста і поділяється на три фази виявлення атак. Перша фаза виявлення атак використовує вимірювання статистики трафіку на основі вікон часових рядів. Алгоритм виявлення постійно контролює швидкість надходження пакетів SYN у кожне фіксоване вікно розміром Δw та оцінює параметр Херста трафіку, після чого порівнює його з пороговим значенням у кінці кожного часового вікна. Алгоритм виявлення швидко видає попередження, коли швидкість пакетів SYN перевищує порогове значення. Однак сплеск законного трафіку також може збільшити швидкість SYN-пакетів, що надходять на контролер. Отже, на другій фазі виявлення атак подальший аналіз виконують, порівнюючи вихідні IP-адреси усіх потоків до бази даних, що містить дійсні IP-адреси джерела. Інформаційні потоки, яким притаманна певна відповідність, вважаються законними. Після цього контролер переходить до третьої фази: обчислює відношення однопакетних потоків до загальної кількості потоків і порівнює це відношення з пороговою величиною. Ідею використання одиничних пакетних потоків для виявлення атак запропонував автор [12], який зауважив, що потоки, які містять менше ніж три пакети, є аномальними, отже, можуть бути генеровані DDoS-атакою. У нашому методі виявлення потоки, що містять один пакет, вважаються аномальними, оскільки в DDoS-атаках з підробленими адресами, наприклад атаках потоку SYN, генерується лише один пакет на потік. Якщо протягом трьох послідовних вікон коефіцієнт відсотка перевищує поріг, оголошується атака SYN, і контролер вживає відповідних заходів.

Для виявлення DDoS є два основні параметри: розмір вікна Δw та пороги параметра Херста. У роботі прийнято, що розмір вікна фіксований. Розмір вікна моніторингу впливає на роботу

алгоритму виявлення. Якщо розмір вікна занадто великий, система повільно реагуватиме на атаки. Якщо розмір вікна замалий, не буде достатньо зібраних зразків даних для точного аналізу трафіку. Крім того, він також використовує більшу пропускну здатність і потребує більше обробних ресурсів. У цій роботі вибрано розмір вікна 3 с. Здається, це значення забезпечує хороший компроміс.

Запропонований алгоритм виявлення потребує двох адаптивних порогів. Замість використання заздалегідь визначених порогових значень алгоритм виявлення адаптивно обчислює порогові значення. Ці порогові параметри адміністратор мережі може налаштувати для адаптації до мінливих умов трафіку в мережі. Для того, щоб зробити порогові значення адаптивними до характеристик трафіку мережі, ми використаємо середнє та стандартне відхилення параметрів трафіку та оцінюватимемо за R/S методикою параметр Херста для виведення відповідних порогових еталонних значень. Для обчислення середнього та стандартного відхилення контролер збирає конкретні параметри трафіку в кожному вікні та використовує їх для обчислення.

На першій фазі виявлення алгоритм виявлення безперервно відстежує кількість SYN-пакетів, що надходять до контролера, у трисекундне вікно, і підраховує кількість SYN-пакетів. Нехай $\overline{X}_n$ – кількість пакетів SYN, що надійшли у вікно n^{th}. Тоді алгоритм оновлює середнє та стандартне відхилення у кінці вікна, використовуючи такі рівняння:

$$MEAN = EWMA = \overline{X}_n = \overline{X}_{n+1} + \alpha\left(X_n - \overline{X}_{n-1}\right), \tag{1}$$

$$Std.Dev. = EWMSTD = \mathrm{S_n} = \sqrt{\alpha\left(X_n - \overline{X}_{n-1}\right)^2 + (1-\alpha)\left(S_{n-1}\right)^2}, \tag{2}$$

де параметр налаштування $0 < \alpha \le 1$ – коефіцієнт згладжування. Вибір значення α істотно впливає на ефективність виявлення. Значення $\alpha = 1$ надає більшу вагу останнім даним і меншу вагу старим даним; невелике значення α надає більшої ваги старим даним. У роботі приймемо значення α – 0,5, що надає однакову вагу поточним та минулим даним $\overline{X}_n$ та S_n – це середнє та стандартне відхилення, розраховане відповідно у вікні n^{th}.

Динамічний поріг δ_n, обчислений у вікні n^{th}, встановлюється відповідно до рівняння (3):

$$\delta_n = \overline{X}_n + k\mathrm{S_n}, \tag{3}$$

де параметр налаштування k є константою. Вибір k – це компроміс між хибно позитивним та помилково негативним. Більше значення k збільшує частоту хибно негативних, тоді як менше значення k – хибно позитивних. У роботі для k встановлено значення 1. Для цього вибираємо невелике значення, оскільки воно використовується на першій фазі виявлення. Будь-які помилкові спрацьовування, введені на цій стадії, можна виправити на пізніших етапах виявлення. З іншого боку, вибираючи менше значення, алгоритм може виявити атаку раніше і зменшити коефіцієнт помилково негативних наслідків.

Для того, щоб отримати другий поріг, який алгоритм виявлення використовує на третій фазі виявлення, система виявлення шукає потоки із одиничними пакетами даних. Визначимо параметр P_r як відношення кількості однопакетних потоків до загальної кількості потоків. Нехай C_{Single} – кількість однопакетних потоків, а C_{Total} – загальна кількість потоків. Потім коефіцієнт відсотка (P_r) обчислюється за допомогою рівняння (4):

$$P_r = \left(\frac{C_{Single}}{C_{Total}}\right) \cdot 100. \tag{4}$$

Параметр P_r обчислюється у кінці кожного часового вікна. За допомогою P_r алгоритму виявлення визначають середнє та стандартне відхилення для отримання адаптивного другого порога. Нехай P_{r_n} – процентне співвідношення потоків одного пакета у часовому вікні; тоді алгоритм оновлює середнє та стандартне відхилення у кінці вікна n^{th}, використовуючи такі рівняння:

$$MEAN = \overline{P_{r_n}} = \overline{P_{r_{n-1}}} + \alpha\left(P_n - \overline{P}_{r\ n-1}\right), \tag{5}$$

$$Std.Dev. = \mathrm{S}_{\mathrm{r_n}} = \sqrt{\alpha\left(P_{r_n} - \overline{P}_{r\ n-1}\right)^2 + (1-\alpha)\left(S_{r_{n-1\,prev}}\right)^2}. \tag{6}$$

У наведених рівняннях (5) та (6) $\overline{P_{r_n}}$ та S_{r_n} – середнє та стандартне відхилення відповідно.

Другий поріг ϕ_n, обчислений у вікні n^{th}, встановлюється відповідно до такого рівняння (7):

$$\phi_n = \overline{P}_{r\ n} + S_{r_n}. \tag{7}$$

Суть використання параметра Херста у тому, що під час дослідження характеристик інформаційних потоків можна знайти унікальний взаємозв'язок між конкретним протоколом, яким передається трафік, і характерним для нього параметром Херста. Отже, виконавши кілька контрольних вимірювань та записавши для кожного користувальницького трафіку певну таблицю, можемо робити висновки про нормальність або аномальність трафіку на підставі відстані отриманих фактичних значень параметра Херста від значень еталонного трафіку без аномалії, які заздалегідь навчені та задані у подальших спостереженнях. Вимірювання параметра Херста відбуваються із певним встановленим вікном моніторингу, зокрема отримані значення записуються у таблиці користувачів кожних 3, 15 та 60 с. Вікно спостереження не обмежується лише 60 с, може бути і більшим. Проте в разі зростання вікна спостереження виявлення аномалій триватиме довше [11].

Важливою частиною методу є концепція зберігання довідкових значень в таблицях. У таблицях передбачається зберігати дані, отримані після застосування математичної формули визначення параметра Херста. Кожен користувач, ідентифікований DPI системою, наприклад, за IP-адресою, має свою таблицю моніторингу значень інформативних сигнатур трафіку згідно із визначеним критерієм Херста. Таблиці вважають протермінованими, якщо пристрій не отримує даних від абонента протягом заданого часу очікування. У таблиці абонента зберігається його IP-адреса, час створення таблиці, а також інші значення, які не змінюються під час функціонування системи моніторингу. Крім того, в таблиці користувача зберігаються дані із трьох різних вікон моніторингу (20 рядків трисекундного вікна, чотири рядка 15-секундного вікна). У табл. 1 наведено приклад таблиці абонента, заповненої в результаті роботи на етапі навчання.

Як видно з прикладу таблиці користувачів, у кінці кожного блока записується середнє значення для кожного контрольованого об'єкта. Якщо вікно моніторингу дорівнює 60 с, то ці значення не враховуються (не обов'язково). Стандартне відхилення також враховується і записується в таблицю. Стандартне відхилення враховується для вікон моніторингу тривалістю 3 і 15 с. Сенс використання трьох різних типів вікон моніторингу полягає у тому, щоб на етапі виявлення мати точніші дані для алгоритму.

Для кожного вікна спостереження визначається дисперсія S_N та середнє значення M_N параметра Херста. Після цього роблять висновок про наявність у трафіку аномалії, порівнюючи його із еталонним трафіком. Наприклад, якщо трафік еталонного трафіку характеризується наближеним нормальним розподілом, для порівняння відхилень значень параметра Херста можна використати правило "трьох сигм".

Таблиця 1

Приклад таблиці користувача для DPI (вікно моніторингу 3 с)

IP.addr = XX.XX.XX.XX			Another.data		
Int., s.	H1	H2	H3	H4	Hn
Window.size = 3 sec.					
1–3	H1-31	H1-32	H1-33	H1-34	H1-3n
…–…	…	…	…	…	…
58–60	H58-601	H58-602	H58-603	H58-604	H58-60n
Havg (3 sec)	Havg 1	Havg 2	Havg 3	Havg 4	Havg n
Sr_n (3 sec)	Sr_n (3 sec)1	Sr_n (3 sec)1	Sr_n (3 sec)1	Sr_n (3 sec)1	S (3 sec)1

Таблиця 2

Приклад таблиці користувача для DPI (вікно моніторингу 15 с)

IP.addr = XX.XX.XX.XX			Another.data		
Int., s.	H1	H2	H3	H4	Hn
Window.size = 15 sec					
1–15	H1-151	H1-152	H1-153	H1-154	H1-15n
16–30	H16-301	H16-302	H16-303	H16-304	H16-30n
31–45	H31-451	H31-452	H31-453	H31-454	H31-45n
46–60	H46-601	H46-602	H46-603	H46-604	H46-60n
Havg (15 sec)	Havg 1	Havg 2	Havg 3	Havg 4	Havg n
Sr_n (15 sec)	Sr_n (15 sec)1	Sr_n (15 sec)1	Sr_n (15 sec)1	Sr_n (15 sec)1	Sr_n (15 sec)1

У цьому випадку видається логічним запропонувати використовувати віддаленість від діапазону $\left(-1S_{r_n};+1S_{r_n}\right)$ як критерій аномалії трафіку. Тоді значення трафіку в діапазонах $\left(-\infty;-1S_{r_n}\right)$ або $\left(+1S_{r_n};+\infty\right)$ можна вважати аномальними. Якщо умова виконується у вікні протягом 3 с, це вважається нормальним. Якщо в трисекундному вікні умова не виконується (значення H виходять за межі нормального діапазону), система має підстави вважати, що існує аномалія. Однак система чекатиме значення 15-секундного вікна і порівнюватиме його з таблицями, розрахованими на етапі навчання. Якщо на 15-секундному вікні немає аномалії, це вважається нормою, і система ніяких дій не виконує. Якщо і тут є аномалія, необхідно з'ясувати, наскільки значення відхиляється від умовної норми. Якщо діапазон відхилень – $\left(+1S_{r_n};+2S_{r_n}\right)$ або $\left(-2S_{r_n};-1S_{r_n}\right)$, будуть вжиті обмежувальні заходи, і система продовжуватиме чекати значення хвилинного вікна [11].

Якщо діапазон становить $\left(+2S_{r_n};+\infty\right)$ або $\left(-\infty;-2S_{r_n}\right)$, система негайно вживатиме обмежувальних заходів. У хвилинному вікні можна отримати лише одне значення, тому в цьому вікні як критерій буде використано просте порівняння значень H для $H \le H_{avg1-60}$. Якщо ця умова виконується, трафік вважається нормальним, якщо не виконується, трафік є аномальним.

Як видно з алгоритму, в разі виявлення аномалії у найменшому вікні моніторингу (3 с) система позначить це значення як підозріле, але не виконає ніяких дій (оскільки не можна із повною впевненістю сказати, що трафік аномальний, враховуючи вид "сплесковості" мережевого

трафіку), можливо, це всього лише “точкова” аномалія. Якщо аномальна поведінка зберігається у вікні моніторингу протягом 15 с, то будуть вжиті відповідні заходи (обмежувальні або заборонні). Якщо аномальна поведінка також зберігається у вікні хвилинного моніторингу, то вже будуть вжиті заборонні заходи (блокування трафіку). Зазначимо, що застосування заходів можливе лише в “серійному режимі”. В іншому випадку DPI система пасивно інформує адміністратора про проблему (повідомлення в syslog, електронна пошта тощо або автоматично обмежує пропускну здатність цього трафіку для забезпечення QoS у режимі реального часу).

3. Дослідження ефективності виявлення аномалій на основі запропонованого алгоритму

У роботі здійснено дослідження запропонованого алгоритму виявлення та блокування мережевих атак за допомогою програмної реалізації системи DPI. Для цього було вимушено згенеровано мережеву атаку від кінцевого пристрою, після чого відбувалось захоплення мережевого трафіку від різних користувачів.

Навчання відбувається в умовах відсутності будь-яких аномалій та атак у мережі з метою надання достовірної інформації, що відповідає за так званні еталоні значення трафіку, згідно із якими відбуватиметься їх порівняння. Перейшовши на етап виявлення, система порівнюватиме розрахунки в режимі реального часу функціонування за певні інтервали із еталонними значеннями в таблиці, отриманими під час навчання. Після цього будуть зроблені висновки про наявність аномальної активності. Для кожного абонента ведеться окрема таблиця і для кожного ж абонента є свої етапи навчання і виявлення. Сама система володіє гнучкістю точки зору розгортання. Розроблена система DPI створена у вигляді програмного забезпечення, що встановлюється на серверах та аналізує трафік вибраного інтерфейсу та відповідно до заданих параметрів здійснює управління потоком даних. Основні логічні компоненти розроблюваної системи – інтелектуальний аналізатор, колектор статистичних даних та прийняття рішень, а також інтелектуальний регулятор трафіку.

Зокрема, під час аналізу мережевого трафіку виявлено шкідливий аномальний трафік, що є одним із різновидів DDoS атак. На рис. 3 крива зеленого кольору показує замасковану атаку під торент-трафік та його основні статистичні характеристики.

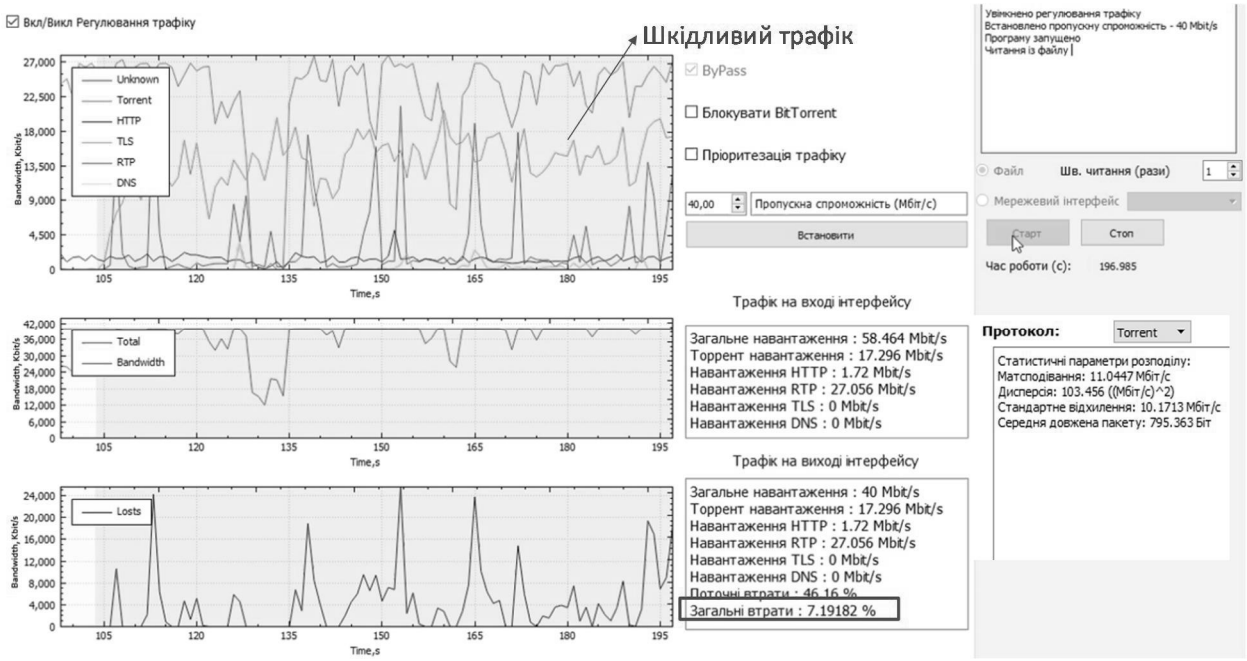

Рис. 3. Демонстрація процесу роботи алгоритму моніторингу та аналізу мережевого трафіку на етапі виявлення мережевих атак у програмно-конфігурованих мережах

У цьому підрозділі виконано повторне дослідження із використанням того самого файла із тими самими вихідними параметрами моделі, але із увімкненим режимом блокування трафіку, функцію якого реалізовує модуль регулятора. На рис. 4 показано проміжок, на якому відсутній торент навантаження. Це пов'язано із тим, що шкідливий трафік – атака зеленого кольору, успішно заблокована системою. Як бачимо, рівень загальних втрат за навантаження на вхід інтерфейсу 53,736 Мбіт/с, знизився від 7,19 % (в умовах наявності шкідливого трафіку) до 2,2 % (в умовах блокування шкідливого трафіку), а втрати змінилися на стрибкоподібні, а не потокові.

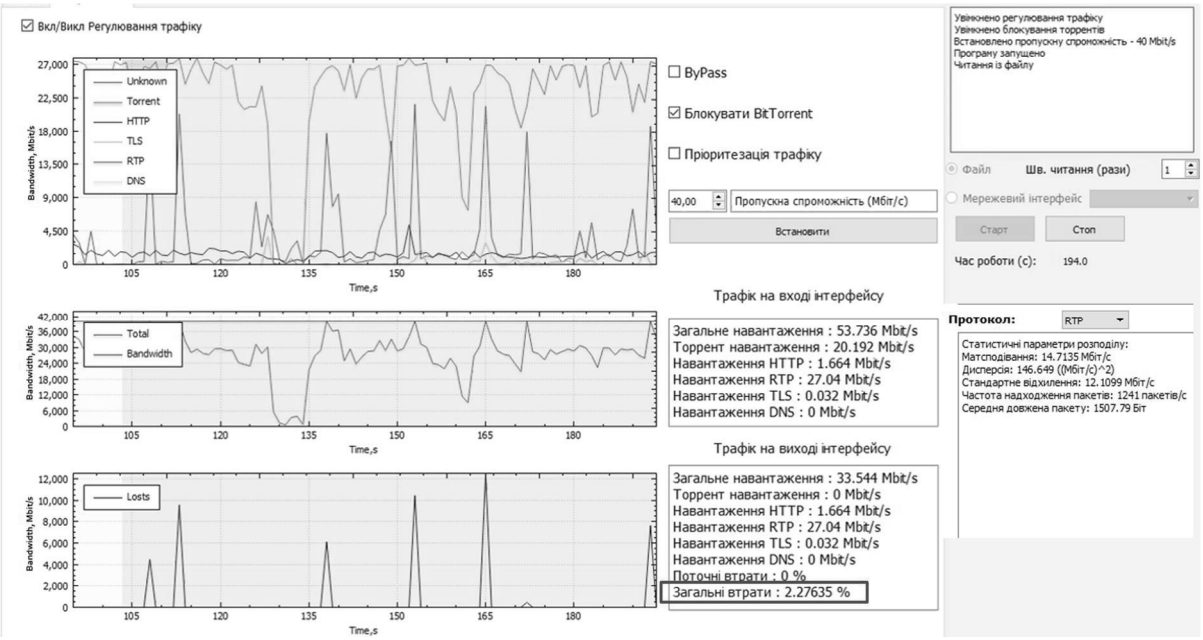

Рис. 4. Демонстрація процесу роботи алгоритму моніторингу та аналізу мережевого трафіку на етапі блокування мережевих атак у програмно-конфігурованих мережах

Якщо порівняти дані, отримані у результаті виконаних дій, то очевидно, що за високого навантаження шкідливого трафіку в каналі доцільно автоматично блокувати цей трафік, щоб забезпечити рівень Qo, відповідний іншим потокам.

Висновок

Розроблено алгоритми для системи виявлення вторгнень (IDS), основані на статистичному аналізі та глибокому навчанні, які спрямовані на виявлення підозрілої поведінки в сучасних та майбутніх програмно-конфігурованих мережах. Запропонований підхід ґрунтується на вивченні часових рядів нормальної поведінки мережі та виявляє помітні аномалії мережі, одночасно зменшуючи час навчання на порядок. Дослідження довели, що використання запропонованого алгоритму виявлення та блокування шкідливого трафіку дало змогу зменшити на 5 % втрати в загальному каналі зв'язку та відповідно покращити якість обслуговування та сприйняття послуг для користувачів законного трафіку.

Список використаних джерел

[1] *He D., Chan S., Ni X. and Guizani M. (2017), "Software-Defined-Networking-Enabled Traffic Anomaly Detection and Mitigation", in IEEE Internet of Things Journal, Vol. 4, No. 6, pp. 1890–1898, Dec. 2017.*

[2] *Hong J., Liu C. and Govindarasu M. (2014), "Integrated Anomaly Detection for Cyber Security of the Substations", in IEEE Transactions on Smart Grid, Vol. 5, No. 4, pp. 1643–1653, July 2014.*

[3] *Yang J., Zhou C., Yang S., Xu H. and Hu B. (2018), "Anomaly Detection Based on Zone Partition for Security Protection of Industrial Cyber-Physical Systems", in IEEE Transactions on Industrial Electronics, vol. 65, no. 5, pp. 4257–4267, May 2018.*

[4] *Kao D. (2020), "Using the Actionable Intelligence Approach for the DPI of Cybercrime Insider Investigation", 2020 22nd International Conference on Advanced Communication Technology (ICACT), pp. 1218–1224.*

[5] *Siwach M. and Mann S. (2022), "Anomaly Detection for Web Log based Data: A Survey", 2022 IEEE Delhi Section Conference (DELCON), pp. 1–5. DOI: 10.1109/DELCON54057.2022.9753130.*

[6] *Alabadi M. and Celik Y. (2020), "Anomaly Detection for Cyber-Security Based on Convolution Neural Network : A survey", 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), pp. 1–14. DOI: 10.1109/HORA49412.2020.9152899.*

[7] *Wang S., Balarezo J. F., Kandeepan S., Al-Hourani A., Chavez K. G. and Rubinstein B. (2021), "Machine Learning in Network Anomaly Detection: A Survey", in IEEE Access, vol. 9, pp. 152379–152396. DOI: 10.1109/ACCESS.2021.3126834.*

[8] *Dymora P. and Mazurek M. (2019), "Anomaly Detection in IoT Communication Network Based on Spectral Analysis and Hurst Exponent", Applied Sciences, Vol. 9, No. 24, Art. no. 24, Jan. 2019. DOI: 10.3390/app9245319.*

[9] *Song W., Beshley M., Przystupa K., Beshley H., Kochan O., Pryslupskyi A., Pieniak D., Su (2020), "A Software Deep Packet Inspection System for Network Traffic Analysis and Anomaly Detection", Sensors, Vol. 20, No. 6, p. 1637-1–1637-41, March 2020.*

[10] *Lutsiv N., Maksymyuk T., Beshley M., Lavriv O., Andrushchak V., Sachenko A., Vokorokos L., Gazda J. (2021). "Deep Semisupervised Learning-Based Network Anomaly Detection in Heterogeneous Information Systems", Computers, Materials & Continua, Vol. 70, No. 1, pp. 413–431.*

[11] *Бешлей М. І. (2021). "Синтез та реалізація інтенційно-орієнтованих інфокомунікаційних мереж для адаптивного надання сервісів": дис. ... д-ра техн. наук: 05.12.02 – телекомунікаційні системи та мережі / Бешлей Микола Іванович; Міністерство освіти і науки України, Національний університет "Львівська політехніка". Львів, 450 с.*

[12] *El Moussaid N., Toumanari A. and El Azhari M. (2017), "Security analysis as software-defined security for SDN environment", 2017 Fourth International Conference on Software Defined Systems (SDS), pp. 87–92. DOI: 10.1109/SDS.2017.7939146.*

INTELLIGENT TRAFFIC MONITORING AND ANALYSIS SYSTEM TO DETECT ATTACKS IN SOFTWARE-DEFINED NETWORKS

M. Beshley, A. Pryslupskyi, M. Medvetskyi, H. Beshley

Lviv Polytechnik National University, 12, S. Bandery str., Lviv, 79013, Ukraine

Today, the task of monitoring, security, management, traffic prioritization and optimization of the overall load on the network infrastructure is becoming extremely important not only for the corporate sector, but also for mobile and fixed-line operators. Current trends in Internet content services indicate that the requirements for traffic management are unpredictable. Therefore, service providers must implement new algorithms to monitor and analyze network traffic to detect attacks in today's networks, where software can be updated to support changing traffic management and control requirements as needed. That is why this paper developed a unique algorithm for monitoring and analyzing network traffic, which improved the efficiency of standard intrusion detection and prevention systems by using intelligent multifractal processes for analyzing incoming traffic. It has been experimentally proved that the use of the developed algorithms in the communication infrastructure allowed to reduce data losses up to 5 % if malicious traffic was present in the communication channels.

Key words: *SDN; network anomaly; Hurst parameter; DPI; intrusion detection.*