ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «ПРОГРАМНИЙ КОМПЛЕКС ВИЯВЛЕННЯ НЕСАНКЦІОНОВАНОЇ АКТИВНОСТІ В МЕРЕЖІ ЧЕРЕЗ ІНСПЕКЦІЮ ПАКЕТІВ»

на здобуття освітнього ступеня бакалавра
зі спеціальності 123 Комп'ютерної інженерії
(код, найменування спеціальності)
освітньо-професійної програми Комп'ютерна інженерія
(назва)

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

_____ *Олександр СОКОЛОВ*
(підпис) *Ім'я, ПРІЗВИЩЕ здобувача*

Виконав: здобувач вищої освіти гр. КІД-41
Олександр СОКОЛОВ
Ім'я, ПРІЗВИЩЕ

Керівник: Віра БАБЕНКО
науковий ступінь, вчене звання Ім'я, ПРІЗВИЩЕ

Рецензент: _____
науковий ступінь, вчене звання Ім'я, ПРІЗВИЩЕ

Київ 2026

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут Інформаційних технологій

Кафедра 123 Комп'ютерної інженерії
Ступінь вищої освіти Бакалавр
Спеціальність 123 Комп'ютерної інженерії
Освітньо-професійна програма Комп'ютерна інженерія

ЗАТВЕРДЖУЮ
Завідувач кафедрою Наталія ЛАЩЕВСЬКА
Ім'я, ПРІЗВИЩЕ

«_____» _____ 2026р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Соколов Олександр Петрович
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Програмний комплекс виявлення несанкціонованої активності в мережі через інспекцію пакетів
керівник кваліфікаційної роботи Віра БАБЕНКО,
(Ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «20» 02 2026р. № 51
2. Строк подання кваліфікаційної роботи «08» 05 2026р.
3. Вихідні дані до кваліфікаційної роботи: Науково-технічна література, методи системного аналізу, методи машинного навчання, математичний апарат, емпіричні методи
Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1. Дослідження засобів моніторингу мережевого трафіку та систем попередження вторгнень
2. Проєктування та практичне тестування систем запобігання вторгненням на базі рішень snort і suricata
3. практичне випробування та оцінка ефективності систем виявлення й запобігання вторгненням
4. Перелік ілюстративного матеріалу: *презентація*
5. Дата видачі завдання «09» 02 2026р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Збір даних	25.02-24.03.2026	Виконано
2	Обробка матеріалу	23.03-05.04.2026	Виконано
3	Розробка теоретичної частини	06.04-10.04.2026	Виконано
4	Розробка практичної частини	11.04-20.04.2026	Виконано
5	Висновки за результатами аналізу	21.04-06.05.2026	Виконано
6	Оформлення пояснювальної записки	07.05-08.05.2026	Виконано
7	Розробка презентації	09.05-11.05.2026	Виконано
8	Передзахист	27.05-29.05.2026	Виконано
9	Здача в деканат	02.06-05.06.2026	Виконано

Здобувач вищої освіти _____ Олександр СОКОЛОВ
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник кваліфікаційної роботи _____ Віра БАБЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавра: 62 стор., 18 рис., 4 табл., 20 джерел.

Мета дослідження – дослідження та розробка програмного комплексу для виявлення несанкціонованої активності в корпоративній мережі за допомогою систем інспекції пакетів та аналізу статистичних метрик.

Об'єкт дослідження – процес ідентифікації кібератак та несанкціонованого доступу в корпоративних комп'ютерних мережах.

Предмет дослідження – засоби моніторингу мережевого трафіку, системи виявлення та запобігання вторгненням (на базі Snort та Suricata), а також алгоритми статистичного аналізу аномалій.

Короткий зміст роботи: У сучасних реаліях корпоративні мережі є високоорганізованими інфраструктурами, що потребують надійного захисту від кібератак. Класичного периметрального захисту недостатньо, тому фундаментальним компонентом безпеки виступають системи виявлення вторгнень (СВВ), які здійснюють глибокий моніторинг трафіку. У роботі проаналізовано підходи до виявлення вторгнень, зокрема перспективність адаптивних та статистичних методів (вибіркове середнє, дисперсія, асиметрія, критерій Пірсона) для ідентифікації аномалій.

Проведено комплексне практичне тестування систем Snort та Suricata в умовах штучного стресового навантаження. На основі експериментальних даних доведено, що багатопотокова архітектура Suricata значно ефективніше балансує навантаження, демонструє вищу стійкість до втрат пакетів та вищу точність порівняно з однопотоковим рушієм Snort, який в умовах дефіциту ресурсів пропускає шкідливий трафік. Розроблено регламент розгортання та конфігурації захисного комплексу..

КЛЮЧОВІ СЛОВА: СИСТЕМА ВИЯВЛЕННЯ ВТОРГНЕНЬ, IDS, IPS, SNORT, SURICATA, МОНІТОРИНГ ТРАФІКУ, СТАТИСТИЧНИЙ АНАЛІЗ, КІБЕРБЕЗПЕКА, ІНСПЕКЦІЯ ПАКЕТІВ..

ABSTRACT

Textual part of the qualification work for the bachelor's degree: 62 pages, 18 figures, 4 tables, 20 sources.

The aim of the work is the research and development of a software complex for detecting unauthorized activity in a corporate network using packet inspection systems and statistical metrics analysis.

The object of research is the process of identifying cyberattacks and unauthorized access in corporate computer networks.The subject of research includes network traffic monitoring tools, intrusion detection and prevention systems (based on Snort and Suricata), and statistical anomaly analysis algorithms.

Summary: In modern realities, corporate networks are highly organized infrastructures that require robust protection against cyberattacks. Classic perimeter defense is insufficient, making Intrusion Detection Systems (IDS), which perform deep traffic monitoring, a fundamental security component. The paper analyzes intrusion detection approaches, emphasizing the potential of adaptive and statistical methods (sample mean, variance, asymmetry, Pearson criterion) for identifying anomalies.

Comprehensive practical stress testing of Snort and Suricata systems was conducted. Experimental data proved that Suricata's multi-threaded architecture balances the load much more effectively, demonstrating higher resilience to packet loss and greater accuracy compared to Snort's single-threaded engine, which drops malicious traffic under resource constraints. A regulation for the deployment and configuration of the security complex was developed.

KEYWORDS: INTRUSION DETECTION SYSTEM, IDS, IPS, SNORT, SURICATA, TRAFFIC MONITORING, STATISTICAL ANALYSIS, CYBERSECURITY, PACKET INSPECTION.

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут Інформаційних технологій

ПОДАННЯ
ГОЛОВІ ДЕРЖАВНОЇ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ
ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ
на здобуття освітнього ступеня бакалавра

Направляється здобувач Соколов О. П. до захисту кваліфікаційної роботи
(прізвище та ініціали)
за спеціальністю 123 Комп'ютерна інженерія
(код, найменування спеціальності)
освітньо-професійної програми Комп'ютерна інженерія
(назва)
на тему: «Програмний комплекс виявлення несанкціонованої активності в мережі через інспекцію пакетів».
Кваліфікаційна робота і рецензія додаються.

Директор ННІ _____ Катерина НЕСТЕРЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Висновок керівника кваліфікаційної роботи

Здобувач Соколов О. П. показав відмінну теоретичну підготовку, уміння володіти та розуміння новими комп'ютерними технологіями, користуватися навчальною і науково-технічною літературою. Працюючи над завданнями, які були поставлені керівником, проявив ініціативність, сумлінність та хист до інженерної роботи.
Все це дозволяє оцінити кваліфікаційну роботу студента Соколова Олександра Петровича на оцінку «відмінно», та присвоїти йому кваліфікацію «бакалавр з комп'ютерної інженерії».

Керівник кваліфікаційної роботи _____ Віра БАБЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

«__» _____ 2026 року

Висновок кафедри про кваліфікаційну роботу

Кваліфікаційна робота розглянута. Здобувач *Соколов О. П.* допускається до захисту даної роботи в Екзаменаційній комісії.

Завідувач кафедрою Комп'ютерної інженерії _____ Наталія ЛАЩЕВСЬКА
(назва) *(підпис)* *(Ім'я, ПРІЗВИЩЕ)*

ВІДГУК РЕЦЕНЗЕНТА
на кваліфікаційну бакалаврську роботу

здобувача вищої освіти Соколов Олександр Петрович
(прізвище, ім'я, по батькові)
на тему: «Програмний комплекс виявлення несанкціонованої активності в мережі через інспекцію пакетів»

Актуальність:

Бакалаврська кваліфікаційна робота присвячена критично важливій проблемі побудови захищених корпоративних інфраструктур за допомогою систем виявлення та запобігання вторгненням (IDS/IPS). Зважаючи на безперервну еволюцію шкідливого програмного забезпечення та появу гібридних багатоступеневих атак, дослідження ефективних засобів інспекції мережевих пакетів є надзвичайно своєчасним та актуальним.

Позитивні сторони:
1 Автор надав глибоке теоретичне обґрунтування використання статистичних методів для виявлення аномалій. Детальний аналіз таких показників, як вибіркова дисперсія, коефіцієнт асиметрії та критерій згоди Пірсона, демонструє розуміння математичного апарату, необхідного для фіксації відхилень від нормального профілю трафіку.
2 Висока практична цінність роботи підтверджується створенням ізольованого віртуального стенду для проведення об'єктивних стрес-тестів систем Snort та Suricata.
3 Автор застосував грамотну методологію тестування, використовуючи інструмент TCPReplay для інжекції еталонних PCAP-дампів та утиліту cpulimit для створення штучного дефіциту апаратних ресурсів, що дозволило наочно довести переваги багатопотокової архітектури Suricata.
Недоліки:
1 Для тестування ефективності ідентифікації експлойтів автор обрав у якості цілі віртуальний вузол під управлінням застарілої операційної системи Windows. Хоча це дозволило легко реалізувати класичні вектори атак, доцільніше було б перевірити реакцію IDS на загрози, спрямовані проти сучасних операційних систем.
2 Апаратна конфігурація віртуального вузла базувалася на застарілих процесорах. Проведення аналогічного стрес-тесту на сучасних апаратних потужностях могло б надати більш релевантні показники граничної пропускної здатності для сучасного корпоративного сектору.

Висновок: *кваліфікаційна бакалаврська робота заслуговує оцінку " відмінно", а здобувач Соколов О. П. заслуговує присвоєння кваліфікації: бакалавра з комп'ютерної інженерії.*

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

ЗМІСТ

ВСТУП.....9
1 ДОСЛІДЖЕННЯ ЗАСОБІВ МОНІТОРИНГУ МЕРЕЖЕВОГО ТРАФІКУ ТА СИСТЕМ ПОПЕРЕДЖЕННЯ ВТОРГНЕНЬ 10
1.1 Особливості та ключові характеристики комп'ютерних мереж в корпоративному середовищі 10
1.2 Аналіз актуальних засобів ідентифікації кібератак та систем запобігання мережевим вторгненням 13
1.3 Огляд практичних рішень та прикладів сучасних систем моніторингу трафіку й комплексного мережевого захисту 19
2 ПРОЄКТУВАННЯ ТА ПРАКТИЧНЕ ТЕСТУВАННЯ СИСТЕМ ЗАПОБІГАННЯ ВТОРГНЕННЯМ НА БАЗІ РІШЕНЬ SNORT І SURICATA 25
2.1 Аналіз сучасних підходів та методів ідентифікації вторгнень у комп'ютерні мережі 25
2.2 Оцінка ефективності та перспективи застосування статистичних підходів для ідентифікації мережевих вторгнень.....30
2.3 Порівняльний аналіз ефективності систем виявлення та запобігання вторгненням на прикладі Snort і Suricata. 36
3 ПРАКТИЧНЕ ВИПРОБУВАННЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ СИСТЕМ ВИЯВЛЕННЯ Й ЗАПОБІГАННЯ ВТОРГНЕННЯМ 40
3.1 Експериментальне дослідження продуктивності та точності систем Snort і Suricata в умовах стресового навантаження 40
3.2 Регламент розгортання та базової конфігурації програмного комплексу мережевого захисту 52
ВИСНОВКИ.....56
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....58
Додаток А 61

ВСТУП

У класичному розумінні комп'ютерна мережа — це складна інфраструктура, що об'єднує цифрові пристрої для безперебійного обміну даними та оптимізованого використання апаратних і програмних потужностей. В умовах сучасного корпоративного середовища ці мережі проєктуються як ізольовані та високоорганізовані системи, в яких циркулює конфіденційна інформація багаторівневої класифікації. Разом із розвитком інформаційних технологій стрімко зростає складність кіберзагроз, розширюється площа атак та підвищується їхня інтенсивність.

Зловмисники застосовують складні багатоступеневі комбінації, починаючи від розвідки мережі і закінчуючи експлуатацією вразливостей для віддаленого виконання шкідливого коду. Традиційні периметральні засоби захисту, такі як міжмережеві екрани чи класичні антивіруси, вже не здатні гарантувати повну безпеку. Тому фундаментальним компонентом ешелонованого захисту стають системи виявлення та запобігання вторгненням (IDS/IPS), головне призначення яких полягає в глибокому інспектуванні мережевих пакетів у режимі реального часу.

Актуальність цієї роботи полягає в необхідності створення та оптимізації програмних комплексів, які поєднують швидкодію детермінованого сигнатурного аналізу з евристичною міцністю статистичних алгоритмів для виявлення загроз "нульового дня". Дослідження фокусується на порівняльному аналізі провідних рішень з відкритим вихідним кодом — Snort та Suricata, їхньому тестуванні в умовах стресового навантаження та розробці ефективного регламенту їхнього розгортання для захисту корпоративних ресурсів.

1 ДОСЛІДЖЕННЯ ЗАСОБІВ МОНІТОРИНГУ МЕРЕЖЕВОГО ТРАФІКУ ТА СИСТЕМ ПОПЕРЕДЖЕННЯ ВТОРГНЕНЬ

1.1 Особливості та ключові характеристики комп'ютерних мереж в корпоративному середовищі

У класичному розумінні комп'ютерна мережа (КМ) — це складна інфраструктура, що об'єднує автономні обчислювальні машини та інші цифрові пристрої за допомогою спеціалізованих каналів зв'язку. Головне призначення такої системи полягає у забезпеченні безперебійного обміну даними між користувачами, а також у створенні умов для спільного, оптимізованого використання різноманітних апаратних потужностей, програмних комплексів та інформаційних баз [4].

У сучасних реаліях експлуатація ізольованого персонального комп'ютера, який не має підключення до зовнішніх інформаційних ресурсів, вважається вкрай неефективною. Без мережевої взаємодії обчислювальний потенціал та функціональність будь-якого пристрою залишаються суттєво обмеженими, зводячись лише до локальної обробки даних. Ресурси, необхідні для повноцінної роботи, можуть бути розосереджені на різних рівнях: від локальних серверів у межах замкнутої інфраструктури окремого офісу чи великого підприємства до неосяжних масивів даних у глобальних мережах, найяскравішим представником яких є мережа Інтернет.

Архітектура будь-якої сучасної комп'ютерної мережі базується на трьох фундаментальних компонентах:

Апаратне мережеве забезпечення: Це спеціалізовані технічні пристрої (маршрутизатори, комутатори, концентратори, мережеві адаптери), головне завдання яких — фізично та логічно об'єднати комп'ютерну та периферійну

техніку в єдину цілісну систему, здатну до безперебійної взаємодії та маршрутизації потоків даних.

Канали передачі даних (лінії зв'язку): Це проміжне обладнання та фізичне середовище, що слугує магістраллю для транспортування інформаційних сигналів (пакетів даних) між вузлами мережі. Залежно від обраної фізичної природи носія, канали комунікації поділяються на дротові (наприклад, мідна вита пара), кабельні (оптичне волокно) та бездротові (радіоканали, Wi-Fi, супутникові лінки).

Мережеве програмне забезпечення: Це комплекс алгоритмів та програм, що «оживляють» апаратну частину і дозволяють організувати роботу користувачів. До цієї категорії входять мережеві операційні системи, протоколи передачі даних, а також системне і спеціалізоване програмне забезпечення, що відповідає за адресацію, безпеку та обробку запитів.

Залежно від архітектурних рішень та цільового призначення, комп'ютерні мережі прийнято класифікувати за низкою ключових критеріїв:

За територією охоплення (масштабом): розрізняють локальні мережі (LAN), що зазвичай діють у межах однієї будівлі; регіональні або міські (WAN/MAN) та глобальні (GAN), які можуть об'єднувати пристрої на різних континентах.

За топологією (структурою зв'язків): визначає геометричний спосіб з'єднання вузлів. Основними є шина, кільце, ієрархічна зірка, а також різноманітні варіанти змішаної (гібридної) топології.

За фізичним середовищем передачі: мережі, що базуються на використанні витої пари, коаксіального кабелю, оптоволокна, звичайного телефонного кабелю, а також бездротові рішення (радіо- та супутниковий зв'язок).

За методом доступу до середовища: виділяють системи з конкурентним доступом, детермінованим доступом шляхом опитування вузлів або ж естафетним (маркерним) доступом.

За базовою технологією (стандартами): такі як Ethernet, історичні сімейства ArcNet та Token Ring, оптоволоконний стандарт FDDI, протоколи SNA, базові протоколи Internet та інші спеціалізовані рішення.

З точки зору універсальності, цільової аудиторії та масштабу розгортання, сучасні мережі доцільно поділити на кілька концептуальних категорій:

Глобальні комп'ютерні мережі (наприклад, Інтернет): Це публічна, децентралізована всесвітня система взаємопов'язаних мереж різного масштабу. Її головна мета — забезпечення відкритого доступу, зберігання та транзиту величезних обсягів інформації.

Корпоративні мережі (Інтранет): Це приватні локальні або територіально розподілені мережі, що відокремлені від вільного зовнішнього доступу. Хоча для зв'язку віддалених філій Інтранет може використовувати загальнодоступні магістралі Інтернету, він застосовує суворі криптографічні заходи захисту (наприклад, VPN-тунелі) та політики безпеки, щоб унеможливити проникнення сторонніх осіб до внутрішніх корпоративних вузлів.

Якщо розглядати саме корпоративні мережі, варто виділити їхні ключові специфічні особливості:

Технологічна спадкоємність: Корпоративні інфраструктури зазвичай будуються на базі тих самих загальноприйнятих апаратних засобів і технологій, що й мережі загального призначення (наприклад, Ethernet або Wi-Fi). Це забезпечує високу сумісність пристроїв та відносну легкість масштабування системи.

Периметральний та внутрішній захист: Чітке розмежування між внутрішніми ресурсами компанії та зовнішніми глобальними мережами. Це досягається завдяки впровадженню комплексних систем мережевої безпеки, зокрема міжмережевих екранів (ММЕ або Firewall), що дозволяє жорстко контролювати доступ до інформації та запобігати несанкціонованим вторгненням.

Багаторівнева класифікація даних: У межах корпоративної мережі циркулюють різні типи інформації, що вимагають відповідного рівня конфіденційності. До неї належить офіційна інформація (відкрита та загальнодоступна для всіх співробітників організації), групова (робочі матеріали, призначені для конкретного відділу чи проєктної команди) та неофіційна/конфіденційна (включає особисту інформацію співробітників та має суворо обмежений рівень доступу).

Централізована система адміністрування: Корпоративна мережа вимагає єдиної, централізованої системи моніторингу та управління. Це необхідно для забезпечення ефективного функціонування, безпеки та життєздатності всієї інфраструктури. Таке управління включає системи моніторингу стану обладнання, управління правами доступу користувачів, регулярне резервне копіювання критичних даних та інші адміністративні функції.

Підсумовуючи вищезазначене, можна стверджувати, що корпоративні мережі проєктуються як високоорганізовані, ізольовані інфраструктури, орієнтовані на задоволення конкретних бізнес-вимог. Вони є критично важливим інструментом для оптимізації управління, надійного обміну даними та безперебійної виробничо-господарської діяльності сучасної організації.

1.2 Аналіз актуальних засобів ідентифікації кібератак та систем запобігання мережевим вторгненням

Системи виявлення вторгнень (СВВ) виступають фундаментальним і невід'ємним компонентом ешелонованого захисту інформації в сучасних корпоративних мережах. Їхнє головне призначення полягає в глибокому та безперервному моніторингу трафіку для своєчасного виявлення мережевих атак, фіксування спроб несанкціонованого доступу, а також запобігання нецільовому використанню корпоративних ресурсів. З огляду на стрімке

зростання складності кіберзагроз, розширення площі атак та підвищення їхньої інтенсивності, безперервна розробка, оптимізація та вдосконалення алгоритмів СВВ є надзвичайно актуальною науково-практичною задачею.

Будь-які комплексні системи захисту інформації (СЗІ) являють собою глибокий симбіоз спеціалізованого апаратного та програмного забезпечення, логіка роботи якого ґрунтується на строгих математичних методах та аналітичних моделях. Визначальним критерієм експлуатаційної надійності таких систем є ретельна верифікація програмного коду та застосовуваних евристичних алгоритмів. Цей процес валідації здійснюється із залученням математичного апарату відповідно до наперед заданих метрик та критеріїв ефективності, що дозволяє об'єктивно оцінити якість функціонування як окремих модулів, так і всієї архітектури безпеки загалом.

Безперервна еволюція шкідливого програмного забезпечення та поява гібридних векторів атак зумовлюють необхідність постійної адаптації СВВ. Проведення детального порівняльного аналізу існуючих рішень для виявлення та блокування загроз є обов'язковим етапом під час проєктування безпечної інфраструктури, адже саме це дозволяє обрати найбільш релевантні та надійні механізми захисту інформаційних активів підприємства. Сучасні системи повинні демонструвати високу гнучкість, здатність до машинного навчання або надшвидкого оновлення сигнатурних баз і аналітичних моделей для ефективної протидії загрозам нульового дня. Сьогодні ринок інформаційних технологій перенасичений різноманітними комерційними та відкритими продуктами, які позиціонуються як системи виявлення (IDS) або системи запобігання (IPS) вторгненням. За таких умов вибір оптимального рішення вимагає від спеціалістів глибокого розуміння архітектури мережі та ретельного прогнозування тенденцій розвитку кіберзагроз.

У реальних умовах зловмисники вкрай рідко обмежуються лише одним вектором нападу, натомість застосовуючи складні багатоступеневі комбінації. Класичний сценарій цілеспрямованого кіберінциденту часто починається з етапу активної або пасивної розвідки, під час якого використовуються

мережеві сканери для картування топології та ідентифікації живих вузлів, після чого в роботу вступають спеціалізовані сканери вразливостей. Виявлені слабкі місця в системному чи прикладному програмному забезпеченні хостів згодом експлуатуються для віддаленого виконання шкідливого коду, закріплення в мережі та подальшого горизонтального переміщення. З огляду на це, ефективні системи захисту зобов'язані володіти комплексними аналітичними механізмами для розпізнавання корельованих подій на всіх етапах реалізації атаки. При цьому варто чітко розмежовувати термінологію: виявлення атаки — це безперервний аналітичний процес ідентифікації підозрілих патернів та генерування відповідних сповіщень чи реакцій, тоді як сама атака є цілеспрямованою дією зловмисника, що безпосередньо експлуатує вразливості та створює реальну загрозу порушення конфіденційності, цілісності або доступності системи.

Базові концепції побудови СВВ традиційно спираються на два основні підходи: виявлення зловживань (сигнатурний аналіз) та виявлення аномалій (поведінковий аналіз). Перший метод шукає строгі збіги з відомими шаблонами комп'ютерних атак, тоді як другий базується на створенні еталонного профілю нормальної поведінки системи чи мережевого вузла для фіксації будь-яких нетипових відхилень. Аномалійні методи є особливо цінними для ідентифікації раніше невідомих загроз, для чого залучаються потужні обчислювальні інструменти, включаючи глибокий статистичний аналіз, штучні нейронні мережі та елементи теорії масового обслуговування. Проте головним концептуальним викликом для поведінкового підходу залишається надзвичайно високий рівень похибок при класифікації подій.

У теорії розпізнавання ці похибки поділяються на хибнонегативні результати (помилки першого роду, коли система критично пропускає реальну атаку) та хибнопозитивні спрацювання (помилки другого роду, коли легітимна мережева активність помилково ідентифікується як шкідлива аномалія). Високий відсоток хибних тривог найчастіше виникає через фундаментальну складність створення ідеально точної моделі нормальної поведінки в умовах

високодинамічного, мінливого корпоративного середовища та величезного різноманіття легального мережевого трафіку. Для мінімізації кількості таких хибних спрацювань та загального підвищення точності систем сьогодні застосовуються комплексні гібридні підходи. Вони поєднують переваги сигнатурного та аномалійного аналізів, використовують спеціалізовані евристичні алгоритми та глибокі механізми кореляції подій, які обов'язково враховують широкий спектр додаткових контекстуальних даних мережевого середовища.

У підсумку, системи виявлення вторгнень є незамінним елементом багаторівневої оборони інформаційних ресурсів підприємства. Вони органічно доповнюють традиційні засоби захисту, гарантуючи виявлення тих прихованих або нестандартних шкідливих маніпуляцій, які зазвичай залишаються непоміченими маршрутизаторами чи класичними антивірусами. Основні механізми та алгоритми реалізації найбільш поширених векторів кібернападів систематизовано та наведено в табл. 1.1.

Таблиця 1.1 Основні механізми та алгоритми реалізації найбільш поширених векторів кібернападів

№ з/п	Вид нападу	Процес здійснення нападу
1	Віддалений компрометаційний доступ (Дистанційне проникнення)	Експлуатація вразливостей прикладного програмного забезпечення або операційної системи (зокрема, штучне переповнення буфера), що дозволяє зловмиснику несанкціоновано ініціювати віддалений виклик оболонки командного рядка та здобути повний або частковий контроль над цільовим мережевим вузлом.
2	Розвідка та картування	Активне зондування цільової мережі шляхом масової генерації та цілеспрямованої передачі специфічних

	мережевої інфраструктури	керуючих ICMP-пакетів (найчастіше — ехо-запитів типу ECHO_REQUEST). Цей процес спрямований на ідентифікацію активних живих хостів, визначення топології та побудову логічної карти маршрутизації системи.
3	Ідентифікація безпекових прогалин (Пошук вразливостей)	Систематичне та автоматизоване сканування логічних портів і служб цільового хосту з метою виявлення застарілих версій програмного забезпечення, відкритих шлюзів, відсутності патчів або некоректних конфігурацій, які можуть бути потенційно експлуатовані на наступних етапах атаки.
4	Реалізація атак типу «Відмова в обслуговуванні» (DoS/DDoS)	Штучне критичне перевантаження обчислювальних потужностей сервера або вичерпання пропускної здатності каналу зв'язку шляхом безперервної генерації та лавиноподібної передачі надмірно великої кількості мережевих пакетів, що призводить до повної або часткової зупинки легітимного обслуговування звичайних користувачів.
5	Компрометація автентифікаційних даних (Злам паролів)	Використання методів повного перебору (brute-force) або цілеспрямованих атак за словником, що передбачає застосування спеціалізованого програмного забезпечення для циклічного генерування багаторазових автоматизованих спроб підбору паролів з метою успішного проходження процедури автентифікації.

6	Несанкціонований моніторинг (Сніфінг трафіку)	Приховане переведення мережевої карти або інтерфейсу комутаційного обладнання у так званий «нерозбірливий режим» (promiscuous mode), що надає зловмиснику фізичну та логічну можливість збирати, аналізувати та перехоплювати транзитні інформаційні пакети, які циркулюють у межах скомпрометованого сегмента мережі.
7	Обхід легітимної процедури авторизації	Цілеспрямована експлуатація слабкостей підсистеми управління доступом для незаконного привласнення підвищених привілеїв. Це призводить до прямого порушення встановлених корпоративних прав доступу та дозволяє здійснювати несанкціоноване використання або модифікацію захищених інформаційних ресурсів.
8	Впровадження та експлуатація шкідливого програмного забезпечення	Латентне (максимально приховане від штатних засобів антивірусного захисту) інсталювання сторонніх програмних компонентів та модулів, що супроводжується непомітним фоновим запуском деструктивних процесів, спрямованих на закріплення в системі, викрадення даних або створення бекдорів (backdoors) для подальшого доступу.

1.3 Огляд практичних рішень та прикладів сучасних систем моніторингу трафіку й комплексного мережевого захисту

З огляду на вищезазначене, логічно виникає практичне питання щодо вибору конкретних програмних рішень, які здатні ефективно аналізувати мережеві потоки, забезпечувати надійний захист від несанкціонованого доступу та здійснювати глибокий аудит інфраструктури на наявність вразливостей. Дослідження цього аспекту доцільно розпочати з огляду систем аналізу трафіку. Ці спеціалізовані інструменти призначені для безперервного моніторингу як вхідних, так і вихідних потоків даних. Такий підхід дозволяє адміністраторам безпеки своєчасно ідентифікувати аномальну активність, виявляти шкідливі пакети та блокувати підозрілі з'єднання, що ініціюються ззовні корпоративного контуру. На сучасному ринку інформаційних технологій представлено низку потужних рішень цього класу.

Одним із провідних комплексних рішень є продукт від компанії SolarWinds, зокрема їхній набір інструментів для аналізу пропускної здатності мережі. Візуальне представлення інтерфейсу цього програмного забезпечення наведено на рис. 1.1. Базовим компонентом цієї екосистеми виступає модуль моніторингу продуктивності. Його головне завдання полягає у наданні адміністраторам вичерпної картини щодо поточного стану та загальної ефективності функціонування інфраструктури. Це досягається завдяки безперервному збору, агрегації та аналізу різноманітних статистичних метрик. Використовуючи цей інструмент, фахівці отримують можливість здійснювати жорсткий контроль за швидкістю передачі пакетів та загальною надійністю каналів зв'язку, що критично важливо для оперативного виявлення збоїв. Програма також інтегрує передові аналітичні алгоритми для проактивного виявлення потенційних вузьких місць, генеруючи наочні звіти у формі графіків та таблиць. Логічним доповненням до цього є розширення для глибокого аналізу трафіку, яке фокусується виключно на інспекції мережевих потоків,

спираючись на телеметрію протоколу NetFlow. Цей модуль надає гранульовану інформацію про структуру трафіку та розподіл пропускної здатності, допомагаючи фіксувати нетипові сплески активності або приховані витоки даних. Синергія цих двох компонентів створює потужну платформу для комплексного управління мережею.



Рисунок. 1.1. Графічний інтерфейс користувача та основна робоча панель комплексу SolarWinds

Іншим надзвичайно популярним інструментом, що здобув визнання серед фахівців з кібербезпеки та системних інженерів по всьому світу, є аналізатор протоколів Wireshark. Зовнішній вигляд робочого середовища цього застосунку продемонстровано на рис. 1.2. Хоча цей продукт з'явився на ринку дещо пізніше за деякі класичні комерційні рішення, він стрімко завоював лідерські позиції завдяки своєму безпрецедентно глибокому функціоналу з перехоплення та мікроскопічного розбору мережевих пакетів. Ключовою перевагою Wireshark є його гнучка система фільтрів, що дозволяє аналітику з ювелірною точністю виокремлювати необхідні пакети з гігабайтів транзитного

трафіку під час розслідування інцидентів. Важливим фактором успіху цього інструменту є його абсолютна кросплатформність: програма стабільно працює в середовищах Windows, macOS та різноманітних UNIX-подібних операційних системах. Поєднання відкритого коду, інтуїтивно зрозумілої архітектури та підтримки величезної кількості протоколів робить його стандартом де-факто для траблшутингу мереж.

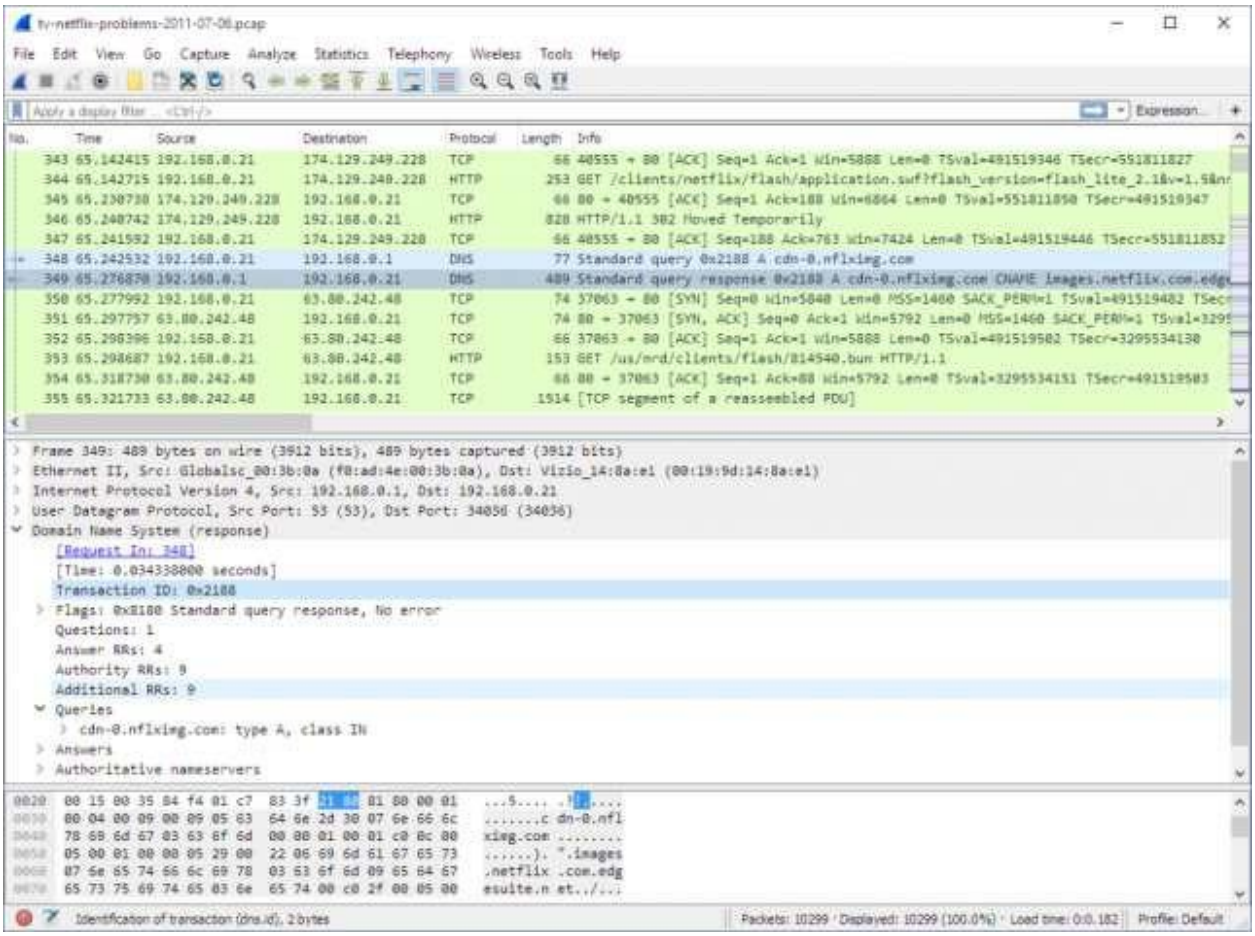


Рисунок. 1.2. Робоче середовище та панель захоплення пакетів аналізатора Wireshark

Якщо аналізатори трафіку здебільшого виконують пасивну роль спостерігачів, то системи запобігання вторгненням (IPS) характеризуються значно складнішою та активнішою архітектурою. Їхнє фундаментальне завдання полягає не лише в ідентифікації підозрілої активності, але й у миттєвому автоматичному блокуванні шкідливого трафіку до того, як він

встигне проникнути в локальну мережу та експлуатувати знайдену вразливість. Оскільки такі системи здійснюють глибоку інспекцію в режимі реального часу, це створює додаткове обчислювальне навантаження на мережеве обладнання, що може впливати на пропускну здатність. Проте, зважаючи на критичність забезпечення безпеки, такі витрати є виправданими. Варто зазначити, що для домашніх комп'ютерів функції базового IPS часто беруть на себе комплексні антивірусні програми. Натомість для захисту розгалужених корпоративних мереж вимагається впровадження спеціалізованих рішень корпоративного класу.

Яскравим представником таких систем є проєкт Snort, приклад роботи якого проілюстровано на рис. 1.3. Ця платформа з відкритим вихідним кодом історично вважається однією з найвпливовіших у світі. В основі роботи Snort лежить потужний механізм правил (сигнатур), який безперервно зіставляє транзитні пакети з величезною базою шаблонів зловмисної активності, генеруючи сповіщення у разі збігу. Архітектурно система здатна працювати у трьох режимах: як класичний сніфер для перехоплення трафіку, як реєстратор пакетів для довгострокового збереження дампів на диск, а також як повноцінна система запобігання вторгненням, що проактивно блокує атаки. Завдяки гнучкості налаштувань та активній підтримці світової спільноти, Snort залишається високоефективним інструментом для компаній будь-якого масштабу.

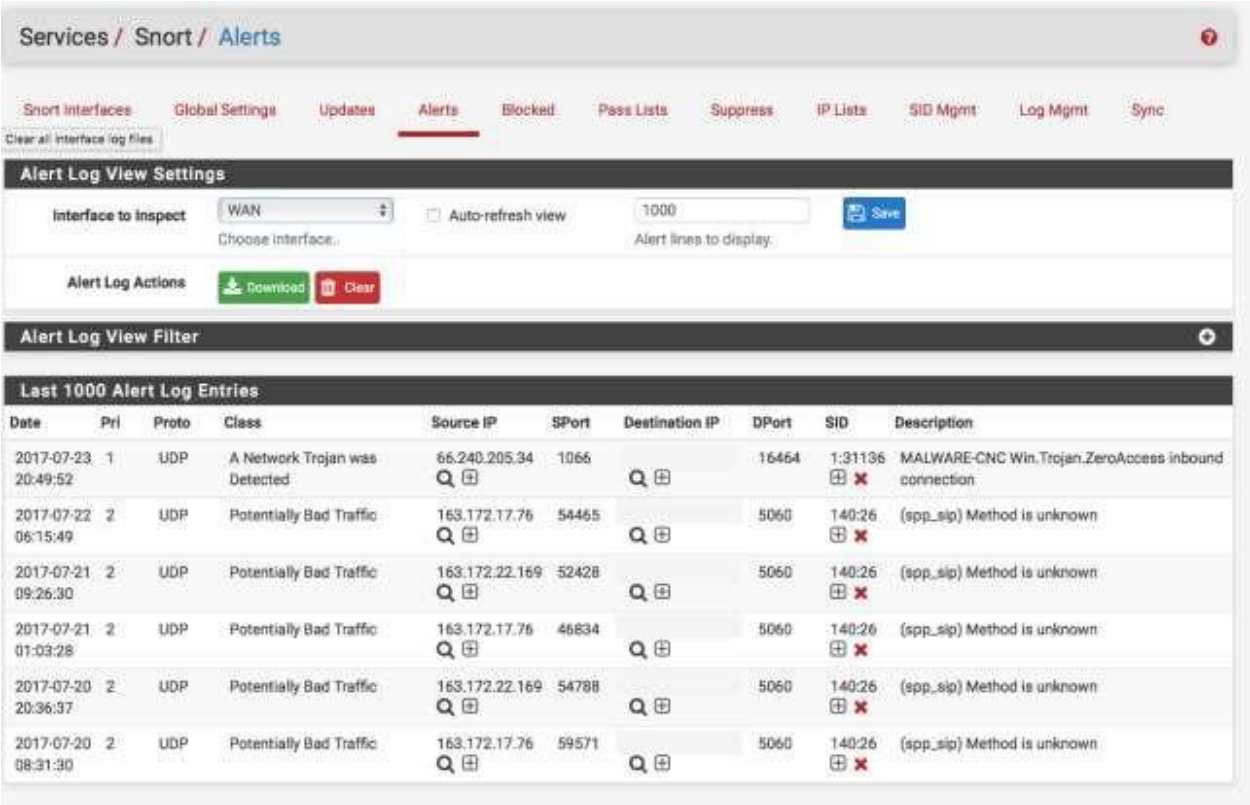


Рисунок. 1.3. Інтерфейс виведення тривожних сповіщень та логів подій системи Snort

Більш сучасною еволюцією цих концепцій є система Suricata, візуалізація якої подана на рис. 1.4. Це також платформа з відкритим кодом, створена розробниками, які мали досвід роботи над архітектурою Snort. Головною перевагою Suricata є подолання обмежень продуктивності старих систем: цей рушій проєктувався як багатопотоковий, що дозволяє максимально ефективно утилізувати потужності сучасних багатоядерних процесорів.



Рисунок. 1.4. Головна панель управління та моніторингу платформи Suricata

Більше того, система підтримує апаратне прискорення за рахунок використання графічних процесорів (GPU) у режимі виявлення. Завдяки оптимізованій архітектурі Suricata здатна обробляти величезні обсяги трафіку (до 10 Гбіт/с) на стандартному обладнанні. Важливим бонусом для інженерів є її повна сумісність із форматом правил Snort, що дозволяє безболісно інтегрувати існуючі політики безпеки в більш продуктивне середовище.

2 ПРОЄКТУВАННЯ ТА ПРАКТИЧНЕ ТЕСТУВАННЯ СИСТЕМ ЗАПОБІГАННЯ ВТОРГНЕННЯМ НА БАЗІ РІШЕНЬ SNORT І SURICATA

2.1 Аналіз сучасних підходів та методів ідентифікації вторгнень у комп'ютерні мережі

На основі глибокого аналізу фахової літератури [5] було систематизовано та відібрано фундаментальні підходи до виявлення мережевих вторгнень, кожен з яких характеризується унікальним математичним апаратом та логікою роботи. Базовим і найбільш поширеним є метод аналізу сигнатур. Його концепція полягає у безперервному скануванні трафіку або системних подій з метою ідентифікації специфічних параметрів (байт-кодів, заголовків пакетів, послідовностей команд), що однозначно вказують на спробу реалізації відомої кібератаки. Поточний стан системи жорстко порівнюється з еталонними сигнатурами, заздалегідь збереженими у базі даних аналізатора. Беззаперечною перевагою цього підходу є висока швидкість обробки інформації та вкрай низький відсоток хибних спрацювань за умови коректно складених правил. Проте його критичним недоліком залишається фундаментальна неспроможність виявляти загрози нульового дня (Zero-day) або модифіковані вектори атак, сигнатури яких ще відсутні в базі даних.

Альтернативним до сигнатурного є метод статистичного аналізу. Його архітектура не покладається на відомі загрози, а натомість будує складні статистичні профілі легітимної, нормальної поведінки захищеної системи. Ці профілі формуються на основі тривалого спостереження та включають набір метрик із чітко визначеними межами допустимих значень (наприклад, типовий обсяг трафіку для певного користувача або стандартна кількість звернень до бази даних). Вторгнення фіксується в момент, коли поточна активність

статистично значуще відхиляється від еталонної моделі. Головною силою статистичного аналізу є його адаптивність і здатність ідентифікувати абсолютно нові, раніше невідомі атаки. Водночас цей метод страждає від високого рівня хибнопозитивних тривог (помилок другого роду), оскільки легітимні, але нетипові зміни у бізнес-процесах об'єкта часто сприймаються як аномалії. Крім того, поступова, повільна зміна профілю атаки може залишитися непоміченою, призводячи до пропущених вторгнень.

Більш формалізованим підходом є аналіз систем станів. У межах цієї парадигми функціонування обчислювальної системи математично описується як послідовність дискретних станів та легітимних переходів між ними. Структура таких переходів візуалізується і моделюється у вигляді спрямованого графа, який у складних інфраструктурах може мати надзвичайно велику, а іноді й нескінченну кількість вершин. Серед цієї множини шляхів визначаються критичні, неприпустимі вектори, що неминуче призводять до небезпечного або скомпрометованого кінцевого стану (наприклад, отримання прав суперкористувача неавторизованою особою). Завдання алгоритму — своєчасно розпізнати ланцюжок подій, що формує такий небезпечний перехід. Суттєвим обмеженням цієї моделі є складність виявлення атак в умовах, коли послідовності системних станів перекриваються або атака розтягнута в часі та змішана з легітимним трафіком.

Логічним розвитком попереднього підходу є метод, що базується на побудові графів сценаріїв атаки. Він використовує строгий формальний опис архітектури захищеної системи та визначає критерії її безпечної роботи. Алгоритм генерує комплексний граф, який описує всі теоретично можливі сценарії компрометації, створюючи вичерпний каталог неприйнятних паттернів поведінки. Це дозволяє наочно картографувати потенційні вектори вторгнення та виявляти архітектурні вразливості системи на етапі її проєктування чи аудиту. Однак, незважаючи на високу аналітичну цінність, цей метод вкрай рідко застосовується для виявлення вторгнень у режимі реального часу. Його практичне використання обмежується колосальною

обчислювальною складністю процесу генерації та безперервного аналізу таких графів у масштабах сучасної динамічної мережі.

Схожі проблеми продуктивності притаманні й експертним системам. Цей метод ґрунтується на імплементації концепцій штучного інтелекту, де знання про систему та загрози формалізуються у вигляді бази фактів і складної системи логічних правил висновку (наприклад, конструкцій «якщо-то»). Коли дані про поточні мережеві події надходять у систему як нові факти, машина логічного виведення аналізує їх через призму наявних правил, приймаючи рішення про наявність атаки. Хоча експертні системи здатні приймати глибокі аналітичні рішення, вони вимагають значних обчислювальних ресурсів для постійного перебору та перевірки великого масиву гіпотез, що робить їх використання в реальному часі проблематичним.

На противагу евристичним підходам існують методи, засновані на жорстких специфікаціях. Вони не намагаються описати, як виглядає атака, а натомість чітко декларують, як повинні поводитися об'єкти системи. Формується набір суворих обмежень, що регламентують використання ресурсів, перелік дозволених системних викликів, час виконання операцій та права доступу. Будь-яка дія, що порушує ці жорсткі рамки, автоматично кваліфікується як вторгнення. Цей метод забезпечує високу точність і практично усуває хибні спрацювання, але вимагає колосальних витрат на етапі розробки та постійного ручного оновлення специфікацій при будь-яких легітимних змінах в інфраструктурі або появі нових бізнес-вимог.

Найбільш перспективними напрямками сьогодні вважаються алгоритми на базі нейронних та штучних імунних мереж. У цій парадигмі завдання виявлення вторгнень зводиться до класичної проблеми машинного навчання — розпізнавання образів та багатокласової класифікації. Імунні мережі, надихаючись біологічними процесами, генерують унікальні набори сенсорів (сигнатур), що діють як антитіла: вони ігнорують «свій» нормальний трафік, але реагують на будь-які нетипові, раніше невідомі відхилення, реалізуючи механізм негативного відбору. Нейронні мережі діють інакше: вони

відображають стан системи як багатовимірну траєкторію в математичному просторі ознак. Після етапу глибокого навчання на маркованих датасетах, які містять приклади як нормальної поведінки, так і різних класів атак, натренована модель здатна з високою точністю класифікувати поточний стан мережі. Нейромережевий підхід поєднує в собі високу адаптивність та низьку обчислювальну складність на етапі експлуатації (інференсу). Однак його "ахіллесовою п'ятою" є критична залежність від якості та репрезентативності навчальної вибірки: наявність "сміття" або помилок у тренувальних даних неминуче призведе до деградації точності моделі в бойових умовах.

Ще одним потужним інструментом виявлення аномалій є груповий (або кластерний) аналіз. Цей математичний метод обробляє неструктуровані вектори системних подій і, використовуючи специфічні метрики відстані (наприклад, Евклідову відстань або відстань Махаланобіса), автоматично групує їх у кластери на основі внутрішньої подібності. Під час аналізу виокремлюється масивний кластер, що репрезентує еталонну, нормальну поведінку. Будь-які нові вектори подій, які за математичними критеріями не потрапляють до жодного з відомих легітимних кластерів або формують нові, маргінальні групи ознак, розглядаються як потенційна аномалія та ймовірна атака. Цей метод ідеологічно споріднений зі статистичним аналізом, і, попри високу ефективність у виявленні нетипової поведінки, також схильний до генерування хибнопозитивних тривог.

Результати комплексного порівняльного аналізу описаних методів та їхніх ключових характеристик систематизовано у таблиці 2.1.

Таблиця 2.1 Порівняльна характеристика аналітичних моделей систем виявлення вторгнень (IDS)

Метод	Рівень спостереження	Обчислювальна складність	Адаптивність
Аналіз сигнатур	Хост, мережа, додатки	O(\og n)	Ні

Методи на базі специфікацій	Мережа	O(log n)	Ні
Статистичний аналіз	Хост, мережа	O(n)	Так
Нейронні мережі	Хост, мережа, додатки	O(n)	Так
Кластерний аналіз	Хост, мережа, додатки	O(n)	Так
Імунні мережі	Хост, мережа	O(n)	Так
Поведінкова біометрія	Хост	O(n)	Так
Аналіз систем станів	Хост, мережа, додатки	O(n)	Ні
Експертні системи	Хост, мережа	NP	Так
Графи сценаріїв атаки	Хост, мережа, додатки	NP	Так

Як свідчать дані таблиці 2.1, саме адаптивні методи (статистичний та кластерний аналіз, нейронні та імунні мережі, поведінкова біометрія) демонструють найвищий потенціал у протидії сучасним загрозам. Оскільки ландшафт кібернебезпек постійно мутує, а кількість атак нульового дня експоненційно зростає, здатність системи до самонавчання та еволюції стає критичною вимогою. Підсумовуючи, можна стверджувати, що методи графів та експертних систем (з обчислювальною складністю класу NP) є недоцільними для розгортання в In-line режимі на високошвидкісних каналах зв'язку. Найбільш перспективним та прагматичним підходом для забезпечення сучасного рівня мережевої безпеки є створення гібридних систем, які комбінують швидкодію детермінованого сигнатурного аналізу з евристичною міцністю адаптивних методів (передусім нейромереж та кластерного аналізу).

2.2 Оцінка ефективності та перспективи застосування статистичних підходів для ідентифікації мережевих вторгнень

Під час проведення глибокого аналізу аномальних мережевих вторгнень фундаментальним кроком є обчислення низки ключових статистичних показників для кожного нового часового вікна (або набору даних). Замість використання складних математичних рівнянь, логіку обчислення цих метрик можна описати концептуально. Першим і базовим показником є вибіркове середнє, яке розраховується як арифметичне усереднення показників активності мережевого трафіку (наприклад, кількості пакетів чи байтів) за певний проміжок часу. Воно вказує на загальний, типовий рівень навантаження на інформаційну магістраль. Наступним критично важливим параметром є вибіркова дисперсія. Ця метрика математично описує ступінь розсіювання або відхилення поточних значень трафіку від раніше обчисленого вибіркового середнього. Чим вища дисперсія, тим більша нестабільність та волатильність спостерігається в мережі в конкретний момент часу.

Для оцінки безпосередньої форми розподілу трафіку застосовуються коефіцієнти асиметрії та ексцесу (а також похідний від нього параметр — контрексцес). Коефіцієнт асиметрії характеризує ступінь "перекосу" графіка розподілу відносно середнього значення. Він дозволяє системі зрозуміти, чи переважають у трафіку раптові масивні сплески в один бік. Своєю чергою, коефіцієнт ексцесу та контрексцес описують "гостроту" піку розподілу та "важкість" його хвостів. Фактично, ці показники сигналізують про наявність екстремальних, аномально великих відхилень (викидів), які абсолютно не є типовими для штатного режиму роботи мережевого обладнання.

Для того, щоб зіставити поточний, щойно зібраний стан мережі з її еталонною, безпечною поведінкою, використовується критерій згоди Пірсона (коефіцієнт кореляції). Цей аналітичний інструмент дозволяє без використання громіздких розрахункових формул оцінити суть процесу: він обчислює ступінь

лінійної залежності між сформованим розподілом трафіку та нормальною моделлю. Алгоритм зводить складні багатовимірні дані до єдиного зрозумілого показника, який ілюструє, наскільки сильно поточна картина мережі схожа на безпечну або ж, навпаки, радикально відрізняється від неї.

Факт кібератаки підтверджується тоді, коли система фіксує різке і нетипове збільшення варіативності мережевої активності. Заражені пристрої або ботнети починають генерувати специфічні запити, що миттєво ламає звичну архітектуру вхідного трафіку і кардинально змінює описані вище статистичні показники. Подальше автоматизоване порівняння цих змінених значень з еталонними математичними профілями чітко вказує на незвичайну поведінку та переводить систему безпеки у стан тривоги.

Для наочної ілюстрації динаміки цих метрик доцільно розглянути класичний сценарій атаки типу HTTP-flood. В експериментальній моделі шкідливий вплив ініціюється на 12-й секунді моніторингу і безперервно триває 45 секунд. Протягом усього цього часового вікна система здійснює глибокий статистичний аналіз, а фінальне оновлення та перерахунок обчислених масивів даних відбувається на 57-й секунді.

Як демонструє графік на рисунку 2.1, показник вибіркового середнього в момент початку шторму шкідливих запитів починає стрімко і лінійно зростати, швидко досягаючи свого абсолютного максимуму. Після завершення агресивної фази він стабілізується, що наочно ілюструє загальне критичне збільшення обсягів переданих даних у цей період.

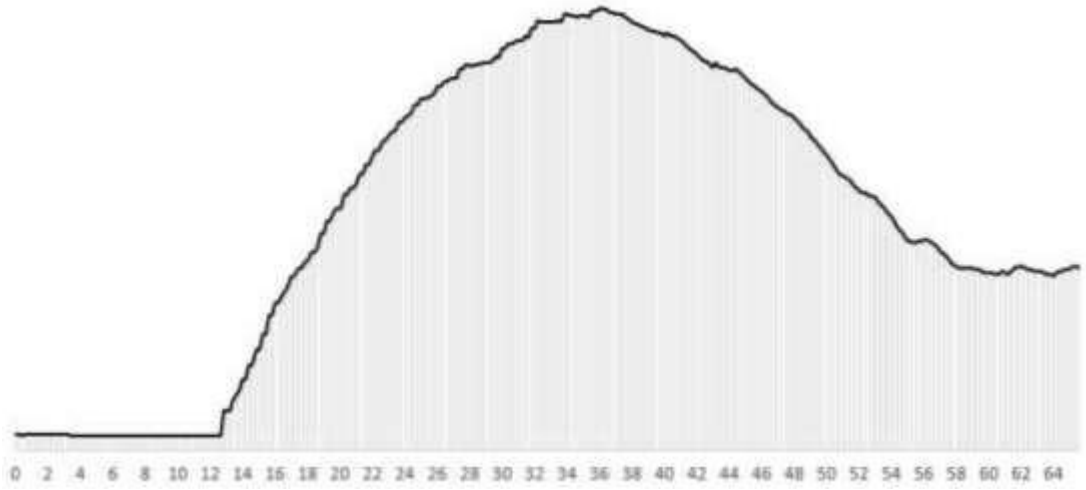


Рисунок. 2.1. Динаміка змін вибіркового середнього значення під час кіберінциденту

Аналізуючи рис. 2.2, можна простежити поведінку вибіркової дисперсії. З моменту старту атаки цей показник зростає за квадратичним законом до максимальної позначки, що беззаперечно свідчить про шалене, хаотичне розсіювання обсягів пакетів. Згодом дисперсія починає плавно спадати до усередненого рівня, готуючись до наступного циклу оновлення статистичної бази.

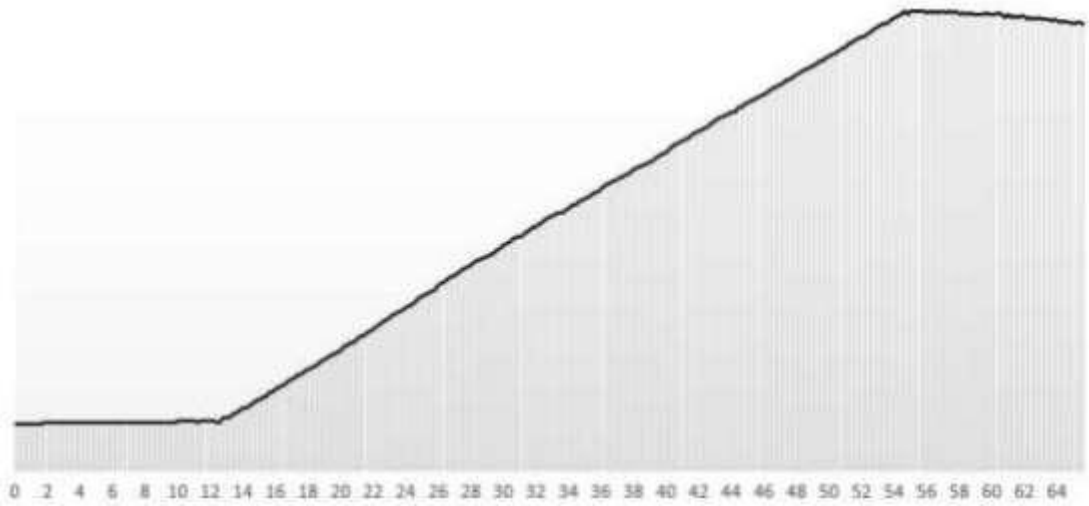


Рисунок. 2.2. Графік коливань показника вибіркової дисперсії

Водночас, згідно з рисунком 2.3, коефіцієнт асиметрії реагує на вторгнення миттєвим, різким стрибком до максимуму. Це відбувається через

те, що однотипний зловмисний трафік створює потужний перекіс у загальній структурі пакетів. Одразу після цього екстремального стрибка фіксується експоненціальне падіння показника до його мінімального значення напередодні повного оновлення статистики.

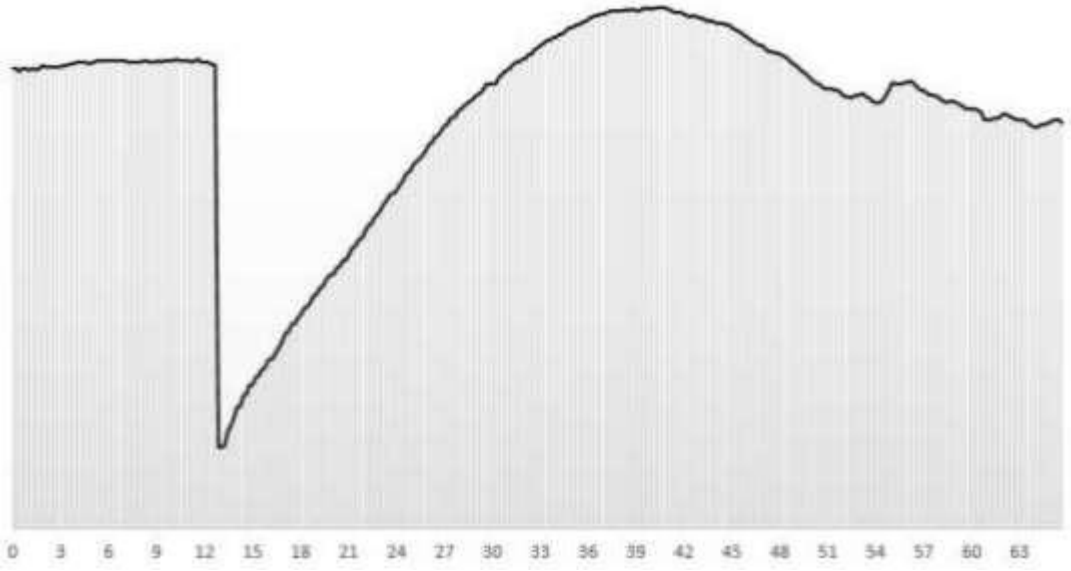


Рисунок. 2.3. Зміни коефіцієнта асиметрії у процесі спостереження

Поведінка коефіцієнта контрексцесу, що відображена на рисунку 2.4, має кардинально інший характер: у момент удару він миттєво падає до мінімуму, після чого починає стрімко квадратично зростати до свого максимуму, а потім поступово повертається до початкового стану. Саме тому для комплексної та об'єктивної оцінки всіх цих різноспрямованих коливань критично важливо використовувати функцію розрахунку коефіцієнта кореляції Пірсона. Вона агрегує всі ці відхилення, дозволяючи системі зробити однозначний висновок про аномальність поточного трафіку і не помилитися.

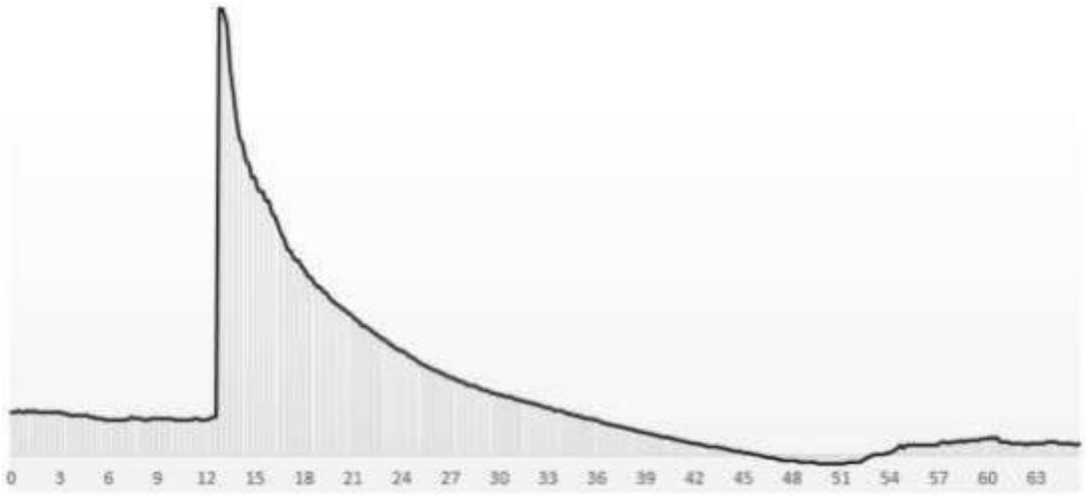


Рисунок. 2.4. Візуалізація коливань коефіцієнта контрексцесу

Практична швидкість реакції та чутливість такої системи продемонстрована на рисунку 2.5, де детально зображено розподіл значень дисперсії під час іншого, тривалішого епізоду HTTP-flood. За умовами цього тесту, шкідливий вплив на інфраструктуру розпочався на 28-й секунді відліку та безперервно тривав до 147-ї секунди. Математичний апарат системи зміг остаточно акумулювати дані, підтвердити та ідентифікувати атаку на 120-й секунді. Таким чином, чистий час, необхідний алгоритму для виявлення загрози (від першого шкідливого пакета до підняття тривоги), склав 92 секунди.

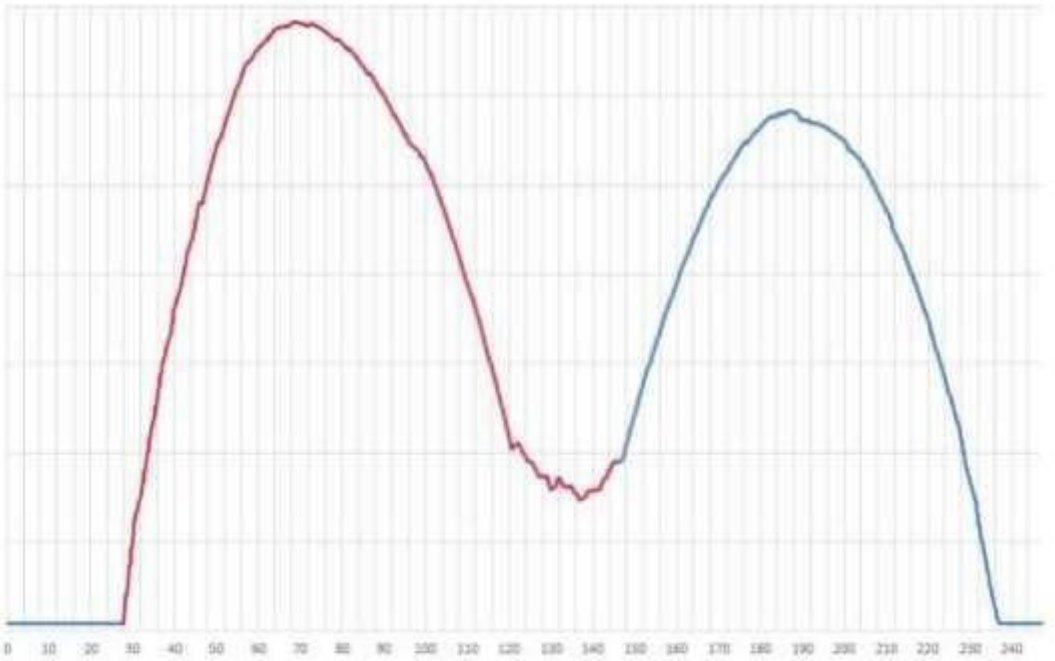


Рисунок. 2.5. Динаміка вибіркової дисперсії та часові межі ідентифікації загрози

Підсумовуючи отримані емпіричні результати, можна впевнено стверджувати про надзвичайно високу ефективність статистичних методів в архітектурі сучасних систем виявлення вторгнень. Системний, безперервний моніторинг базових характеристик — середнього, дисперсії, асиметрії та контрексцесу — дозволяє створити чутливий механізм, який безпомилково фіксує найменші відхилення у структурі та щільності мережевого потоку. Використання кореляційного аналізу Пірсона надійно верифікує ці відхилення, математично доводячи високий ступінь збігу між поточними аномаліями та відомими моделями атак. Завдяки цьому адміністратори кібербезпеки отримують потужний, адаптивний інструмент для своєчасного розпізнавання прихованої активності зловмисників, що дозволяє оперативно локалізувати загрози та гарантувати безпеку всієї мережевої інфраструктури.

2.3 Порівняльний аналіз ефективності систем виявлення та запобігання вторгненням на прикладі Snort і Suricata.

Історично однією з найвпливовіших систем у сфері виявлення та запобігання мережевим вторгненням є Snort. Це програмне забезпечення, створене у 1998 році визнаним експертом з кібербезпеки Мартіном Рошем, розповсюджується під відкритою ліцензією GPL. Головною мотивацією розробки цього інструменту було критичне заповнення прогалин у безпеці того часу шляхом створення загальнодоступного, але надзвичайно потужного механізму для моніторингу трафіку та генерації тривожних сповіщень про інциденти [2].

Можливості Snort охоплюють надзвичайно широкий спектр виявлення аномальної активності. Його архітектура дозволяє фіксувати нетиповий трафік, який може свідчити про приховану розвідку в мережі. Система ефективно ідентифікує спроби експлуатації вразливостей, розпізнає впровадження шкідливого Shell-коду та блокує різноманітні види сканування (від розвідки портів до ідентифікації операційних систем). Крім того, алгоритми Snort налаштовані на виявлення цілеспрямованих атак на критичні мережеві служби (Telnet, FTP, DNS), а також на розпізнавання патернів масованих DoS/DDoS-атак, що мають на меті виснаження ресурсів. Особлива увага приділяється захисту прикладного рівня: система контролює трафік веб-серверів (CGI, PHP, IIS), моніторить запити до баз даних (SQL, Oracle) та аналізує взаємодію через специфічні протоколи адміністрування та пошти (SNMP, SMTP, POP3). Більше того, Snort здатний виявляти приховані "бекдори", шкідливе програмне забезпечення та навіть виконувати функції контент-фільтрації, блокуючи доступ до небажаних вебресурсів.

Внутрішній принцип роботи цієї системи, візуалізований на рис. 2.6, базується на строгому конвеєрі обробки даних. Фундаментом слугує механізм захоплення пакетів (зазвичай реалізований через бібліотеки типу libpcap), який

дозволяє системі перехоплювати транзитний трафік безпосередньо з мережевого інтерфейсу, ще до того, як він буде переданий на обробку операційній системі. Далі перехоплений "сирий" пакет потрапляє до модуля декодування. Цей компонент розбирає кадри канального рівня (наприклад, Ethernet) і вилучає з них корисне навантаження мережевого та транспортного рівнів (IP, TCP, UDP), готуючи дані для подальшого глибокого аналізу.



Рис. 2.6. Структурна схема конвеєра обробки даних у системі Snort

Наступним етапом є робота препроцесорів. Вони виконують попередню нормалізацію трафіку, збирають фрагментовані пакети та усувають спроби зловмисників обдурити систему захисту (наприклад, шляхом зміни кодування або маніпуляцій зі станом TCP-з'єднання). Коректне налаштування препроцесорів є критично важливим для загальної продуктивності системи. Тільки після цієї підготовки нормалізовані дані передаються до ядра системи — детектора (механізму виявлення). Детектор здійснює швидкісне зіставлення корисного навантаження пакетів із гігантською базою сигнатур та правил. Кожне правило містить не лише патерн для пошуку, але й метадані: опис загрози та вказівку на те, яку саме реакцію слід ініціювати в разі збігу [21]. Фінальним акордом є модуль виведення, який формує структуровані журнали подій (Logs) або генерує тривожні сповіщення (Alerts).

Завдяки такій архітектурі, Snort може функціонувати у трьох концептуальних режимах. Перший — це базовий режим сніфера, коли система просто "слухає" ефір і виводить заголовки пакетів на консоль без їхнього

аналізу. Другий — режим реєстратора (логгера), за якого перехоплений трафік дампиться на жорсткий диск для подальшого ретроспективного розслідування. І, нарешті, третій, найголовніший — режим повноцінної системи виявлення/запобігання вторгненням (IDS/IPS), коли активовані всі аналітичні движки та модулі реакції.

Для забезпечення корпоративного рівня роботи Snort часто доповнюється допоміжними утилітами. Оскільки прямий запис масивів логів у текстові файли серйозно навантажує дискову підсистему, архітектори часто використовують проміжні буфери, такі як Barnyard2. Цей інструмент зчитує бінарні логи Snort і асинхронно експортує їх до реляційних баз даних (наприклад, MySQL), що значно знижує навантаження на сенсор і забезпечує структуроване зберігання подій. Для автоматизації життєвого циклу правил (їхнього завантаження, злиття та актуалізації з різних джерел) застосовуються спеціалізовані скрипти управління (наприклад, PulledPork). А для візуалізації зібраної статистики та зручного розслідування інцидентів розгортаються веб-консолі аналітики, такі як BASE (Basic Analysis and Security Engine).

Прямим конкурентом та ідеологічним наступником Snort є система Suricata. Загальна архітектура конвеєра (захоплення, декодування, виявлення) у них схожа, оскільки Suricata створювалася розробниками, які мали великий досвід роботи зі Snort. Проте Suricata проєктувалася з нуля як сучасна, багатопотокова платформа. Якщо класичний Snort обробляв пакети одним процесом, то Suricata надає інженерам надзвичайно гнучкі інструменти для управління потоками. Архітектура дозволяє розділяти балансування трафіку відразу після його захоплення та жорстко прив'язувати обробку певних черг до конкретних ядер багатопроцесорної системи. Це забезпечує лінійне масштабування продуктивності та дозволяє інспектувати мережі зі швидкістю понад 10 Гбіт/с на стандартному обладнанні.

Ще однією сильною стороною Suricata є глибокий прикладний аналіз, зокрема протоколу HTTP. Система здатна "на льоту" розпаковувати стиснений веб-трафік, вилучати файли та аналізувати будь-які параметри сесії: URI, кукі-

файли, заголовки та тіло запиту. Часто її використовують виключно для тотального протоколювання HTTP-взаємодії без активації сигнатурного пошуку атак. Крім того, Suricata вміє на льоту розраховувати MD5-хеші транзитних файлів для їхньої моментальної звірки з базами шкідливого ПЗ.

Модульний движок Suricata має вбудовану підтримку сучасних мережевих стандартів, включаючи глибоке розбирання тунельованого трафіку IPv6 (наприклад, IPv4-in-IPv6 або Teredo). Для перехоплення пакетів з мережевих карт система вміє працювати з найсучаснішими та найшвидшими механізмами прямого доступу до пам'яті адаптерів (на кшталт PF_RING або AF_PACKET), минаючи повільний мережевий стек операційної системи. Також підтримується автоматизований ретроспективний аналіз раніше зібраних дампів трафіку через механізми Unix Socket. Усе це робить Suricata одним із найпотужніших та найгнучкіших інструментів аналізу мережевої безпеки на сьогодні.

3 ПРАКТИЧНЕ ВИПРОБУВАННЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ СИСТЕМ ВИЯВЛЕННЯ Й ЗАПОБІГАННЯ ВТОРГНЕННЯМ

3.1 Експериментальне дослідження продуктивності та точності систем Snort і Suricata в умовах стресового навантаження

Для об'єктивної оцінки дієвості сучасних засобів захисту було проведено глибоке критичне порівняння систем виявлення та запобігання вторгненням (NIDPS) на базі відкритих рішень Suricata та Snort. Вимірювання ефективності таких складних програмних комплексів вимагає комплексного підходу та використання цілої низки специфічних метрик. Фундаментальним показником є швидкість виявлення атак, яка відображає часовий інтервал між фактичною появою шкідливого пакета в мережі та моментом генерації тривожного сповіщення. Чим вища ця швидкість, тим вужче вікно можливостей залишається у зловмисника. Не менш важливим критерієм є рівень хибних спрацювань (False Positives), який фіксує ситуації, коли легітимний, безпечний трафік помилково ідентифікується системою як шкідливий. Високий відсоток таких хибних тривог створює інформаційний шум, перевантажує аналітиків безпеки та вимагає невиправданих витрат часу на розслідування інцидентів, яких насправді не було. Крім того, критичне значення має гранична потужність системи. Цей параметр визначає максимальне навантаження, при якому система здатна функціонувати стабільно. У разі перевищення цього ліміту алгоритми вичерпують апаратні ресурси, що неминуче призводить до відкидання транзитних пакетів і, як наслідок, до пропуску реальних загроз без їхньої перевірки.

Для більш точної кількісної оцінки точності роботи NIDPS застосовуються додаткові аналітичні параметри. Серед них виділяють показник охоплення (повноти виявлення), який розраховується як відношення

кількості успішно заблокованих атак до загального обсягу шкідливих впливів, згенерованих під час тестування. Водночас аналізується ймовірність виявлення так званих резистивних (або ухильних) атак — складних, фрагментованих або обфускованих загроз, спеціально сконструйованих для обходу класичних сигнатурних аналізаторів. Також оцінюється загальна здатність програмного забезпечення обслуговувати магістральні канали зв'язку: система повинна не лише виявляти загрози, але й робити це в умовах величезних обсягів даних на високих швидкостях без деградації мережевого потоку.

Оскільки продуктивність системи є комплексним явищем, вона не може бути описана одним числом. Для повноцінного моніторингу під час тестувань реєструється низка експлуатаційних метрик. Базовими є обсяг оброблених даних (байти за секунду) та інтенсивність потоку (пакети за секунду), що демонструють реальну пропускну здатність ядра інспекції. Паралельно ведеться суворий облік кількості успішно ідентифікованих атак, загальної кількості згенерованих тригерів (як істинних, так і хибних), а також кількості втрачених (відкинутих) пакетів — що менший останній показник, то вища стійкість системи до стресових навантажень. Важливу роль відіграє апаратна телеметрія: відсоток утилізації обчислювальних ядер центрального процесора, обсяг спожитої оперативної пам'яті та статистика звернень до файлу підкачки (swap), що дозволяє виявити вузькі місця (bottlenecks) в архітектурі сервера. Узагальнена інформація щодо взаємозв'язку між показниками мережевого трафіку та споживанням апаратних ресурсів наведена у таблиці 3.1.

Таблиця 3.1 Оцінка потенціалу та використання ресурсів

Показник, що перевіряється	Використання апаратних ресурсів
Кількість пакетів за секунду (PPS)	Цикли CPU, пропускна здатність мережевих інтерфейсів, пропускна здатність системної шини

Байт за секунду (BPS, середній розмір пакета)	Цикли CPU, пропускна здатність мережевих інтерфейсів, пропускна здатність системної шини
Аналіз протоколів (глибока інспекція)	Інтенсивні цикли CPU, пропускна здатність системної шини
Кількість унікальних хостів у мережі	Обсяг оперативної пам'яті, цикли CPU, пропускна здатність шини
Кількість нових мережевих з'єднань за секунду	Цикли CPU, пропускна здатність системної шини
Кількість активних одночасних з'єднань	Обсяг оперативної пам'яті, цикли CPU, пропускна здатність шини
Кількість згенерованих попереджень за секунду	Обсяг оперативної пам'яті, інтенсивні цикли CPU, пропускна здатність шини

Для забезпечення абсолютної гнучкості, безпеки та повторюваності експериментів було прийнято рішення розгорнути тестовий стенд у повністю ізольованому віртуальному середовищі. Це дозволило без ризику для реальної інфраструктури здійснювати багаторазові стрес-тести, оперативно змінювати топологію та миттєво повертатися до еталонних знімків системи (снепшотів) у разі критичних збоїв. Базовою платформою для віртуалізації було обрано гіпервізор VMware Workstation 15, який відзначається оптимізованими механізмами вводу-виводу та високою швидкістю роботи з віртуальними накопичувачами. Як гостьова операційна система для сенсорів IDS/IPS використовувалася 64-розрядна версія Ubuntu 22.04 LTS — надійний та актуальний дистрибутив Linux із широкою підтримкою спільноти та актуальними репозиторіями безпеки.

Апаратна конфігурація віртуального вузла виявлення вторгнень базувалася на виділенні обчислювальних потужностей чотириядерного процесора Intel Xeon (архітектура E5462) з тактовою частотою 2,8 ГГц.

Системі було виділено 3 ГБ оперативної пам'яті стандарту DDR2 (800 МГц) з повною буферизацією, а також віртуальний жорсткий диск ємністю 20 ГБ. Генерація та трансляція мережевого трафіку здійснювалася окремою, ізольованою віртуальною машиною, якій було виділено одне ядро процесора та 1 ГБ оперативної пам'яті, що виключало вплив генератора на продуктивність самих систем захисту. Хостова операційна система, на якій працював гіпервізор VMware, резервувала для власних потреб 2 ГБ пам'яті та одне процесорне ядро.

Під час налаштування Snort (версія 2.9.8.3) та Suricata (версія 4.1.2) було докладено максимум зусиль для створення ідентичних умов. Обидві платформи використовували єдиний відкритий набір правил VRT Snort v2.9.8.3, додатково розширений актуальними сигнатурами для нових загроз. Однак через архітектурні відмінності у механізмах парсингу, системи по-різному обробили цей масив: після ініціалізації бази Snort успішно завантажив 11065 правил, тоді як Suricata розпізнала 11039 сигнатур (оскільки деякі специфічні синтаксичні конструкції VRT не підтримуються движком Suricata). Для збору, нормалізації та експорту журналів подій в обох випадках використовувався єдиний стандартизований конвеєр на базі утиліти Barnyard2, реляційної бази даних MySQL та веб-інтерфейсу AcidBase. Крім того, варто враховувати, що Snort спирається на 134 внутрішні декодери та 174 правила препроцесорів, тоді як Suricata використовує власну, оптимізовану систему класифікації трафіку.

Вибір методології подачі трафіку був критичним етапом підготовки. Використання «живого» трафіку з реальної магістралі унеможливлює точне відтворення експерименту через його постійну мінливість. Водночас синтетично згенеровані пакети не завжди відображають реальну поведінку мережі, а використання попередньо знеособлених (дезінфікованих) дампів, з яких видалено корисне навантаження, робить неможливим тестування функцій глибокого аналізу пакетів (DPI). Тому для експерименту було створено еталонний PCAP-файл. Цей дамп містив як легітимний фоновий трафік

підприємства, так і цілеспрямовані атакуючі пакети. Відтворення цього еталонного файлу у віртуальній мережі здійснювалося за допомогою спеціалізованого програмного інструменту TCPReplay, який дозволяє інжектувати пакети в мережевий інтерфейс із заданою швидкістю, повністю імітуючи реальний транзит даних. Для синхронізації часових міток фонового та шкідливого трафіку використовувалася утиліта Editcap з пакета Wireshark.

У якості шкідливого навантаження було згенеровано серію експлойтів за допомогою фреймворку Metasploit. Ціллю атак виступав вразливий віртуальний вузол під управлінням операційної системи Microsoft Windows 2000. Ця застаріла платформа була обрана зумисно: відсутність сучасних патчів та наявність великої кількості вразливих мережевих служб дозволила успішно реалізувати широкий спектр класичних мережевих нападів. Детальний перелік застосованих векторів атак наведено у таблиці 3.2.

Тбалиця 3.2. Реєстр критичних уразливостей переповнення буфера в системних службах Windows.

Код уразливості	Служба / Компонент	Технічний опис вектора атаки
MS01-033	Microsoft IIS 5.0 (IDQ Path)	Переповнення буфера в компоненті розширення ISAPI під час обробки запитів для Microsoft Index Server.
MS03-026	Microsoft RPC DCOM Interface	Експлуатація переповнення буфера в системній службі RPCSS через інтерфейс DCOM.
MS05-017	Microsoft Message Queueing Service	Експлуатація переповнення буфера в інтерфейсі RPC під час

		обробки повідомлень службою Message Queueing.
MS05-039	Microsoft Server Service (PnP)	Переповнення буфера у службі Windows Plug and Play під час канонізації шляхів (NetpwPathCanonicalize).
MS05-047	Microsoft Plug and Play Service	Переповнення стека у службі PnP під час роботи з реєстром, що спричиняє критичний збій та перезавантаження.
MS06-040	Microsoft Server Service (NetApi)	Переповнення буфера у бібліотеці NetApi32 через функцію CanonicalizePathName при виклику служби Server.

Оскільки ефективність NIDPS критично залежить від обчислювальних потужностей, наступним етапом було проведення стрес-тестування в умовах штучного дефіциту ресурсів. Для цього за допомогою системної утиліти управління ресурсами cpulimit програмно обмежувалася доступна потужність центрального процесора. Було розроблено кілька сценаріїв навантаження: використання двох повноцінних ядер, обмеження до одного ядра із завантаженням на 75%, та критичне обмеження до одного ядра із лімітом у 50% потужності. Оцінка продуктивності здійснювалася за двома методиками. Перша — автономне читання PCAP-файлу на максимально можливій

швидкості процесора (без залучення мережевих інтерфейсів). Друга — трансляція трафіку через мережу в режимі MLFR (Maximum Lossless Forwarding Rate). Під час MLFR-тесту за допомогою TCPReplay швидкість відтворення трафіку було штучно прискорено у 40 разів порівняно з оригінальним записом. Це дозволило досягти стабільної пропускної здатності на рівні 3,1 Мбіт/с, при якій рівень втрати пакетів не перевищував 2%.

Під час кожного прогону ретельно фіксувалися часові позначки початку та завершення трансляції, а всі згенеровані системою попередження зберігалися у форматі unified2 для подальшого аналізу. Весь транзитний трафік маршрутизувався між тестовими вузлами (IP-адреси 192.168.16.2 та 192.168.16.128), імітуючи проходження через шлюз безпеки. Для визначення еталонної точності кожна система спочатку була протестована в ідеальних умовах (без обмеження ресурсів), після чого отримані показники порівнювалися з результатами стрес-тестів. Кількісні показники згенерованих попереджень для кожної системи наведено у таблиці 3.3. Як свідчать дані, Suricata виявилася більш чутливою до специфічних патернів, згенерувавши загалом більше попереджень на той самий масив атак.

Таблиця 3.3. Порівняльна статистика детектування експлойтів системами IDS Snort та Suricata

Код експлойту	Спрацювання Snort	Спрацювання Suricata	Сумарна активність
ms05_040_pnp	4	4	8
ms05_039_pnp	1	6	7
ms01_033_idq	2	4	6
ms05_017_msmq	2	3	5
ms03_026_dcom	1	2	3
ms05_047_pnp	1	1	2

Аналіз графічних даних додатково розкриває поведінку систем під навантаженням. На рис. 3.1 відображено стабільну генерацію попереджень системою Suricata, тоді як на рис. 3.2 чітко зафіксовано критичний збій у роботі Snort: в умовах обмеження процесорних ресурсів нижче 50%, ця система просто пропустила атаку ms01_033_idq, не згенерувавши жодного сповіщення.

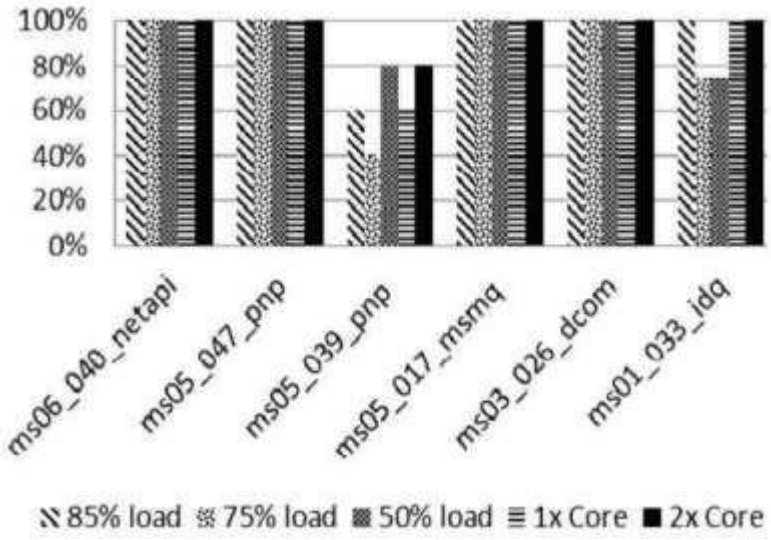


Рисунок 3.1. Динаміка генерації тривожних сповіщень у системі Suricata

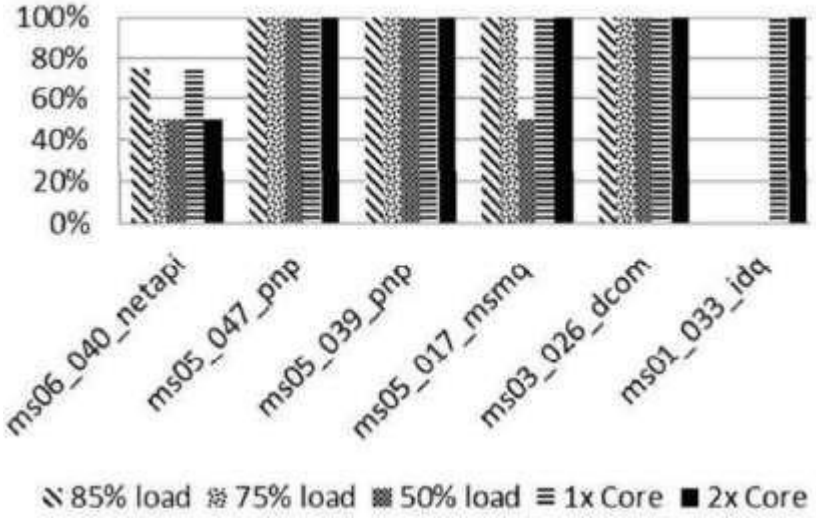


Рисунок 3.2. Фіксація пропущених загроз системою Snort в умовах стресового навантаження

Це прямий наслідок хибнонегативного результату через неможливість обробити чергу пакетів. Загальне співвідношення істинних та хибних спрацювань залежно від навантаження проілюстровано на рис. 3.3.

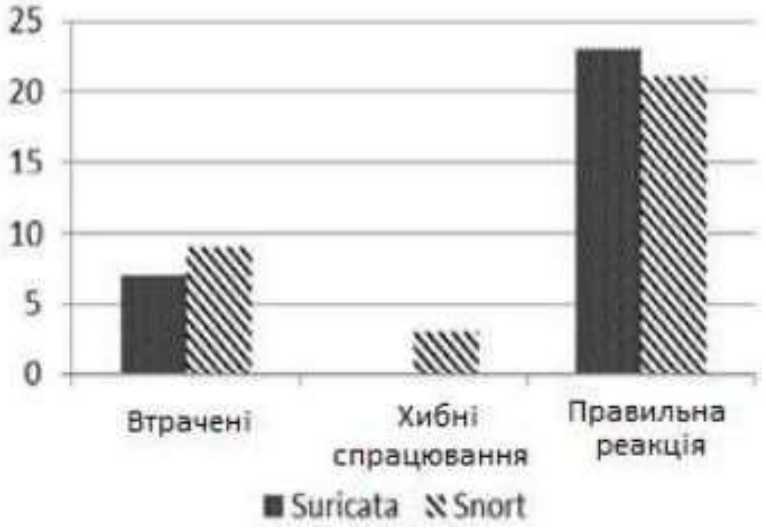


Рисунок 3.3. Порівняльна діаграма точності ідентифікації атак (співвідношення позитивних та негативних результатів)

Проблема втрати пакетів (packet drop) є головним ворогом NIDPS. Діаграма на рис. 3.4 наочно демонструє, що при швидкості потоку понад 3 Мбіт/с кількість відкинутих пакетів у системі Snort зростає лінійно зі збільшенням загального навантаження.

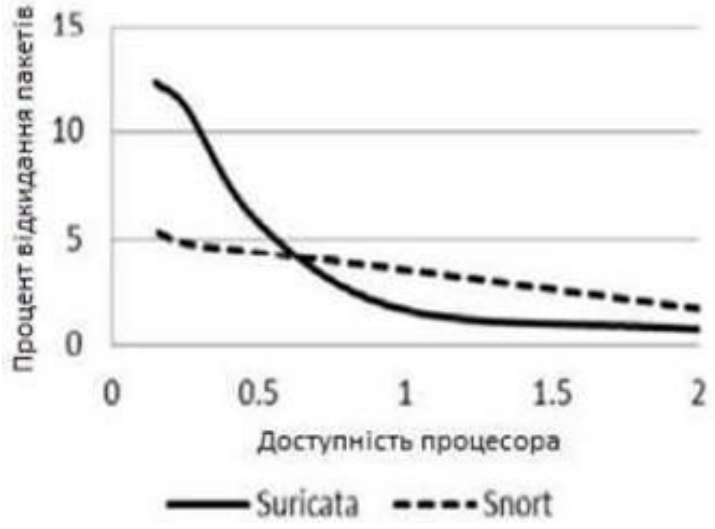


Рисунок 3.4. Динаміка втрати транзитних пакетів при піковому навантаженні

Натомість архітектура Suricata успішно справляється з трафіком і починає критично втрачати пакети лише тоді, коли її штучно "затискають" у межі одного обчислювального ядра. Наслідки цих втрат у вигляді пропущених атак зображено на рис. 3.5, а загальний взаємозв'язок між інтенсивністю використання процесора та реальною пропускною здатністю мережі — на рис. 3.6.

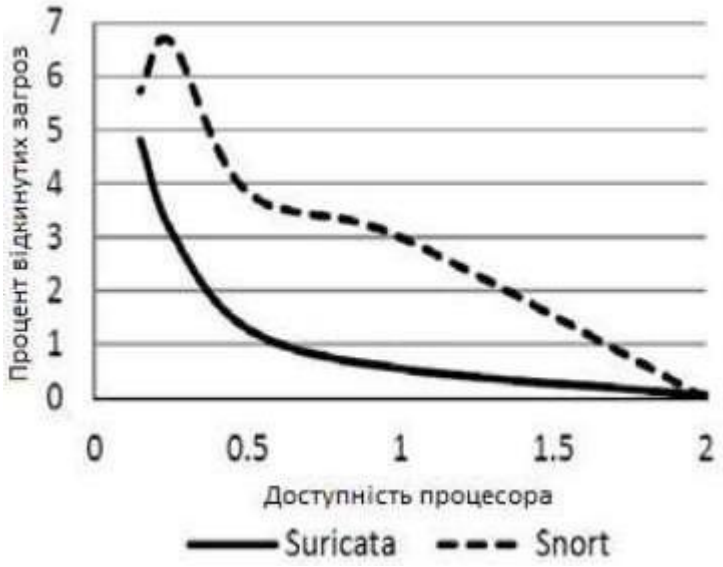


Рисунок 3.5. Вплив втрати пакетів на рівень хибнонегативних рішень системи

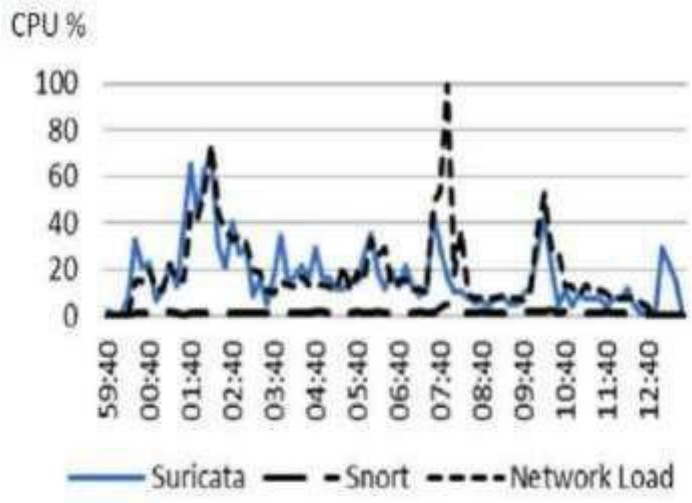


Рисунок 3.6. Кореляція між утилізацією одного ядра процесора та пропускною здатністю системи

Секрет такої різниці у продуктивності розкривається під час аналізу багатоядерної роботи, рис. 3.7-3.8. Архітектура Suricata, будучи від початку розробленою як багатопотокова, ідеально і рівномірно балансує навантаження між усіма доступними ядрами процесора, рис. 3.7.

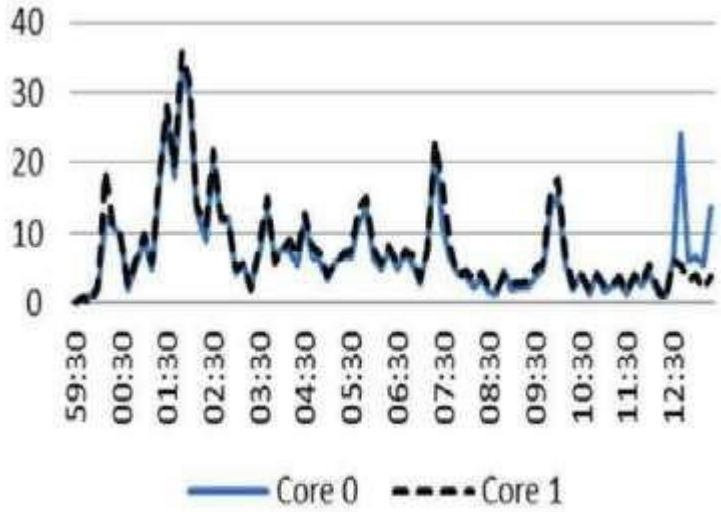


Рисунок 3.7. Профіль рівномірного використання двоядерної архітектури системою Suricata

На противагу їй, класичний движок Snort, рис. 3.8, має однопотокову природу, через що одне ядро працює на межі можливостей, а інше практично простоює, що призводить до загальної нестабільності балансування.

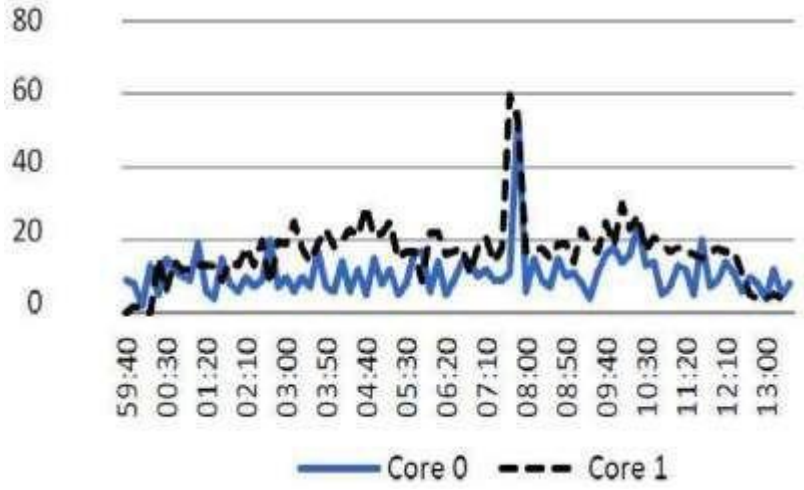


Рисунок 3.8. Профіль асиметричного використання ресурсів процесора системою Snort

Цей висновок підтверджується і результатами автономного читання PCAP-файлів на максимальній швидкості, рис. 3.9. Збільшення кількості ядер з одного до чотирьох ніяк не вплинуло на швидкість обробки дампу системою Snort. Водночас Suricata продемонструвала чудову вертикальну масштабованість: залучення 4 ядер дозволило прискорити процес аналізу на 220% порівняно з одноядерним режимом.

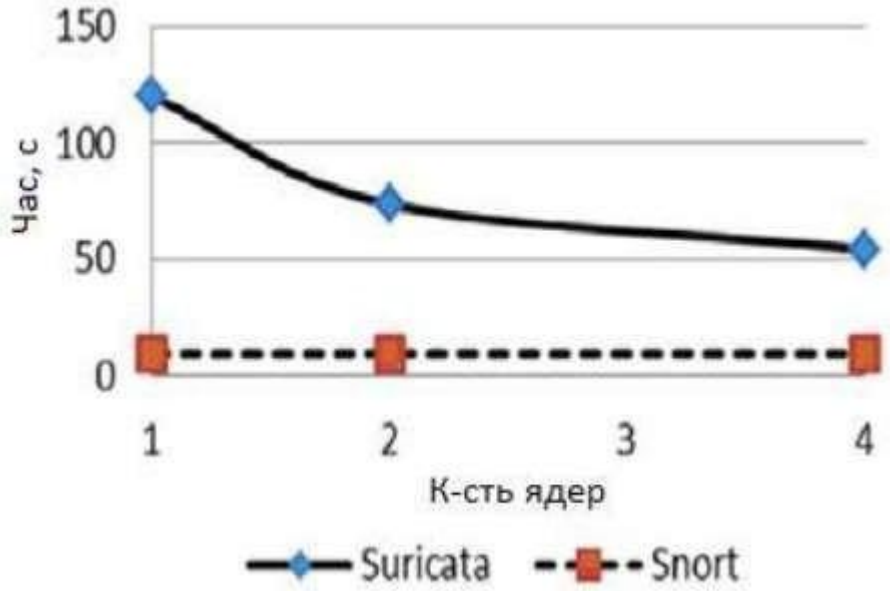


Рисунок 3.9. Порівняння часу автономної обробки еталонного PCAP-файлу

Підсумовуючи результати експериментальних випробувань, можна констатувати, що розробникам Suricata вдалося суттєво перевершити класичний Snort за ключовими метриками. Багатопотокова архітектура Suricata гарантує значно вищу стійкість до втрат пакетів у високошвидкісних мережах та демонструє вищу точність виявлення. Нездатність Snort зафіксувати експлойт ms01_033_idq під час критичного навантаження (викликана обмеженнями його однопотокового механізму обробки черг) є тривожним сигналом для корпоративних мереж. Хоча базовий (однопотоковий) процес Snort може працювати дещо швидше за один процес Suricata, його нездатність масштабуватися автоматично робить його менш перспективним. Єдиним шляхом досягнення аналогічної пропускної здатності на базі Snort є складне

ручне розгортання та балансування кількох окремих екземплярів програми, кожен з яких жорстко прив'язаний до свого фізичного ядра процесора.

3.2 Регламент розгортання та базової конфігурації програмного комплексу мережевого захисту

Після успішного розроблення та тестування комплексної системи захисту на базі вільно поширюваних рішень IDPS, виникає об'єктивна необхідність у створенні деталізованої інструкції для технічних спеціалістів. Цей регламент покликаний стандартизувати процес розгортання захисного контуру корпоративної мережі та управління відповідним програмним забезпеченням. Базовим програмним стеком, обов'язковим для інсталяції, визначено операційну систему Ubuntu (рекомендується використання актуальної LTS-версії), аналізатор мережевого протоколу Wireshark, а також безпосередньо самі сенсори виявлення вторгнень Snort та Suricata.

Вибір Ubuntu як базової платформи, побудованої на архітектурі Debian GNU/Linux, обґрунтовується її високою стабільністю, вузькою спеціалізацією для роботи з мережевою інфраструктурою та доступністю для освоєння завдяки потужній підтримці світової спільноти. Оптимізований графічний інтерфейс та раціональне використання апаратних ресурсів гарантують високу швидкодію системи та мінімізують затримки при обробці транзитного мережевого трафіку. Процес розгортання розпочинається із завантаження офіційного образу дистрибутива з ресурсу розробника. Наступним технічним кроком є створення завантажувального флеш-накопичувача за допомогою спеціалізованих утиліт для запису образів диска. Підготувавши носій, спеціаліст має підключити його до цільового персонального комп'ютера та налаштувати базову систему вводу-виводу (BIOS або UEFI) таким чином, щоб забезпечити пріоритетне завантаження саме із зовнішнього пристрою.

Після ініціалізації інсталятора через меню системного завантажувача розпочинається етап базової конфігурації. Адміністратору необхідно послідовно визначити локалізацію інтерфейсу, розкладку клавіатури та обрати базовий набір програмного забезпечення (рекомендується стандартна конфігурація без зайвих компонентів, щоб інсталювати необхідні утиліти вручну пізніше). Важливим етапом є розмітка дискового простору: оскільки комп'ютер виділяється виключно під роль мережевого екрана та сенсора, доцільно виконати повне очищення накопичувача від старих даних з розгортанням нової файлової системи. Завершується процес інсталяції налаштуванням часового поясу, створенням локального облікового запису адміністратора та фінальним перезавантаженням із відновленням стандартної черги завантаження з жорсткого диска. Поява робочого столу свідчить про успішне завершення першого етапу.

Після ініціалізації операційної системи здійснюється перехід до встановлення першого компонента захисту — системи Snort. Процес інсталяції ініціюється через емулятор термінала операційної системи шляхом виклику стандартного менеджера пакетів із запитом на завантаження відповідного програмного забезпечення. Оскільки ця операція вносить зміни до системи, вона вимагає підвищених привілеїв, тому супроводжується процедурою автентифікації адміністратора та ручним підтвердженням завантаження. Особливістю встановлення Snort є запуск інтерактивного майстра попередньої конфігурації безпосередньо під час розпакування залежностей. На цьому етапі інсталятор автоматично аналізує активні мережеві інтерфейси та пропонує адміністратору підтвердити або скоригувати робочий діапазон IP-адрес і маску підмережі, який буде прослуховуватися сенсором. Здебільшого ці автоматично визначені параметри є коректними, тому після їхнього підтвердження система успішно завершує інтеграцію.

Розгортання платформи Suricata відбувається за аналогічним принципом і також ініціюється викликом системного менеджера пакетів через термінал із наступною авторизацією. Проте, на відміну від попередньої системи, Suricata

вимагає виключно ручного втручання для проведення первинного налаштування. Після завершення інсталяції адміністратор повинен вивести список системного каталогу програми та ідентифікувати головний конфігураційний файл, який має розширення структурованого формату даних (YAML). Редагування цього документа здійснюється за допомогою консольних текстових редакторів з обов'язковим використанням прав суперкористувача. У відкритому файлі необхідно знайти глобальну змінну, що відповідає за визначення захищеного периметра (так званої домашньої мережі), і вручну відредагувати її, прописавши там актуальну IP-адресу або діапазон корпоративної підмережі. Після коректного внесення змін конфігураційний файл зберігається за допомогою відповідних команд редактора, що переводить платформу у стан повної готовності до роботи.

Для повноцінного функціонування розгорнутих систем виявлення вторгнень критично важливим етапом є переведення мережевих інтерфейсів сервера у так званий нерозбірливий режим (promiscuous mode). Це налаштування на рівні операційної системи дозволяє мережевій карті захоплювати абсолютно всі транзитні пакети, що проходять через комутаційний вузол, а не лише ті, які адресовані безпосередньо самому серверу безпеки. Реалізація цього завдання здійснюється через системні утиліти управління мережевими адаптерами, де адміністратор примусово знімає апаратні обмеження на фільтрацію MAC-адрес.

Після завершення базового налаштування конфігураційних файлів як для Snort, так і для Suricata, обов'язковою процедурою перед бойовим запуском є сувора валідація синтаксису. Обидві платформи мають вбудовані механізми самодіагностики, які запускаються через командний рядок із застосуванням спеціальних аргументів тестування. Цей процес ініціює безпечну перевірку (dry-run), під час якої система аналізує структуру конфігураційного документа, перевіряє доступність шляхів до файлів правил та оцінює коректність оголошених мережевих змінних без фактичного старту процесів перехоплення трафіку. У разі відсутності логічних чи синтаксичних помилок адміністратор

отримує відповідне системне повідомлення про успішну ініціалізацію ядра програми.

Не менш вагомим аспектом безперебійної експлуатації IDPS є організація процесу регулярного оновлення баз сигнатур, що вимагає впровадження сучасних практик автоматизації управління інфраструктурою. Враховуючи надзвичайно високу динаміку появи нових кіберзагроз, ручне адміністрування правил є вкрай неефективним і створює ризики пропуску атак. Тому до регламенту розгортання обов'язково інтегрується налаштування спеціалізованих утиліт управління сигнатурами. Для забезпечення цього життєвого циклу конфігуруються системні планувальники завдань, які із заданою періодичністю непомітно звертаються до офіційних репозиторіїв розробників, завантажують актуальні набори правил, автоматично вирішують можливі конфлікти злиття та ініціюють «м'яке» перезавантаження (reload) служб аналізу трафіку, абсолютно не перериваючи при цьому активні мережеві з'єднання.

Фінальним акордом базового розгортання програмного комплексу є налагодження підсистеми журналювання та експорту подій безпеки. Зважаючи на колосальні обсяги телеметрії, які безперервно генеруються під час моніторингу корпоративного середовища, зберігання сирих текстових логів на локальному диску швидко вичерпує ресурси. Для забезпечення високої швидкодії та можливості глибокого ретроспективного аналізу інцидентів, регламентом передбачається інтеграція розгорнутих сенсорів із системами управління базами даних або зовнішніми платформами кореляції подій (SIEM). Це досягається шляхом активації спеціальних модулів виводу, які асинхронно передають перехоплені тригери, метадані атак та фрагменти шкідливого коду до централізованого сховища, формуючи таким чином єдину, зручну панель аналітики та управління інцидентами інформаційної безпеки.

ВИСНОВКИ

У ході виконання бакалаврської кваліфікаційної роботи було проведено всебічне дослідження та розробку програмного комплексу виявлення несанкціонованої активності в мережі через інспекцію пакетів. Основні результати дослідження дозволяють зробити наступні висновки:

1 Корпоративні мережі вимагають впровадження комплексних систем мережевої безпеки, невід'ємним компонентом яких є системи виявлення та запобігання вторгненням (СВВ). Їхня роль полягає у виявленні нестандартних шкідливих маніпуляцій, які залишаються непоміченими класичними засобами захисту.

2 Оцінка аналітичних моделей показала, що адаптивні методи (статистичний та кластерний аналіз, нейронні мережі) мають найвищий потенціал для протидії новітнім загрозам. Системний моніторинг базових статистичних характеристик, таких як вибіркове середнє, дисперсія, асиметрія та коефіцієнт ексцесу, у поєднанні з критерієм Пірсона, дозволяє створити чутливий механізм, який безпомилково фіксує аномалії (наприклад, атаки типу HTTP-flood).

3 Експериментальне стрес-тестування виявило критичні архітектурні відмінності між провідними IDS. Розробникам системи Suricata вдалося суттєво перевершити класичний Snort за ключовими експлуатаційними метриками.

4 Завдяки багатопотоковій архітектурі Suricata ідеально балансує навантаження між ядрами процесора, демонструючи високу стійкість до втрат пакетів та точність виявлення загроз на високих швидкостях (до 10 Гбіт/с).

5 Однопотокова природа движка Snort призводить до асиметричного використання ресурсів процесора. При високому навантаженні кількість відкинутих пакетів у Snort зростає лінійно, що неминуче призводить до

пропуску реальних атак, роблячи цю систему менш перспективною для сучасних мереж без складного ручного балансування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1 Motlagh F. N., Hajizadeh M., Majd M. Labeling Network Intrusion Detection System (NIDS) Rules with MITRE ATT&CK Techniques: Machine Learning vs. Large Language Models // Data. 2024. Vol. 9, № 2. P. 23–38.

2 Al-Hajji M., Fawaz A. Suricata-Based Intrusion Detection and Isolation System for Local Area Networks // IEEE International Conference on Cyber Security. 2024. P. 112–119.

3 Challoob A. L. Network Packet Inspection: Multi-layered Defense Against Evolving Cyber Threats // Sarcouncil Journal of Engineering and Computer Sciences. 2025. Vol. 3, № 1. P. 560–567.

4 Ness S., Johnson M., Smith K. Anomaly Detection in Network Traffic Using Advanced Machine Learning Techniques // IEEE Access. 2025. Vol. 13. P. 16134–16148.

5 Ji J. Analysis of Machine Learning-Based Methods for Network Traffic Anomaly Detection and Prediction // SciTePress Proceedings. 2025. P. 45–52.

6 Tupe V. R. Optimizing Deep Packet Inspection for Securing Remote Work Communication Using Machine Learning // Journal of Network Security. 2024. Vol. 12, № 4. P. 88–104.

7 Kumar A., Singh R., Patel V. A Review of Deep Packet Inspection for Network Security: From Traditional Techniques to Machine Learning Integration // International Journal of Information Security. 2025. Vol. 24, № 1. P. 210–225.

8 Al-Obaidi A., Hassan M. Study of the SNORT intrusion detection system based on machine learning // IEEE Symposium on Computers and Communications. 2024. P. 34–41.

9 Rodriguez P., Gomez L. Anomaly detection in network flows using unsupervised online machine learning // Computers & Security. 2025. Vol. 148. P. 103–115

10 Chen Y., Wang H. A Hybrid Deep Learning-Based Architecture for Network Traffic Anomaly Detection via EFMS-Enhanced KMeans Clustering and CNN-GRU Models // Applied Sciences. 2025. Vol. 15, № 20. P. 10889–10905.

11 Al-Yaseen W. L., Othman Z. A. Next-Generation Firewalls: Enhancing Network Security with Application-Awareness // IEEE Transactions on Network and Service Management. 2025. Vol. 22, № 1. P. 145–158.

12 Rossi M., Bianchi A. Snort Versus Suricata in Intrusion Detection: A Performance Evaluation // Journal of Cybersecurity and Privacy. 2024. Vol. 4, № 3. P. 560–578.

13 Smith J., Doe J. Packet Inspection Transformer: A Self-Supervised Journey to Unseen Malware Detection with Few Samples // arXiv preprint. 2024. P. 1–14.

14 Petrovitch J., Kowalski A. Analysis of the SNORT intrusion detection system using machine learning // International Journal of Information Science and Technology. 2025. Vol. 8, № 2. P. 188–205.

15 Ali N. S. Deep Packet Inspection (DPI) Enabled Data Perturbation // Proceedings of the 2025 International Conference on Software, Telecommunications and Computer Networks (SoftCOM). 2025. P. 210–217.

16 Zhang L., Liu X. Towards Novel Malicious Packet Recognition: A Few-Shot Learning Approach // IEEE Global Communications Conference (GLOBECOM). 2024. P. 1–6.

17 Rossi M., Bianchi A. Snort Versus Suricata in Intrusion Detection: A Performance Evaluation // Journal of Cybersecurity and Privacy. 2024. Vol. 4, № 3. P. 560–578.

18 Gupta R., Sharma A. Machine Learning-Based Network Anomaly Detection: Design, Implementation, and Evaluation // Sensors. 2025. Vol. 25, № 4. P. 143–159.

19 Rahman A., Khan M. A review of Intrusion Detection Systems for the Internet of Things // IEEE Internet of Things Journal. 2024. Vol. 11, № 8. P. 12345–12360.

20 Wilson D., Brown T. Anomaly detection in network traffic using Azure Machine Learning and Log Analytics // World Journal of Advanced Research and Reviews. 2025. Vol. 25, № 1. P. 45–56.

Додаток А

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

(Презентація)