МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Кафедра Комп'ютерних систем та мереж

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри комп'ютерних систем та мереж

_____(Жуков І.А.)

« ____ »
2021 р.

КВАЛІФІКАЦІЙНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ
"БАКАЛАВР"

Тема: Система виявлення вторгнення в комп'ютерну мережу на основі аналізу трафік

Виконавець: студент групи КС-431 Калініченко Данило Ігорович

Керівник: к.т.н., доцент Малярчук Олександр Володимирович

Нормоконтролер: Журавель С.В.

Київ 2023

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук та технологій
Кафедра комп'ютерних систем та мереж

Напрям (спеціальність) 123 "Комп'ютерна інженерія"
(шифр, найменування)

ЗАТВЕРДЖУЮ
Завідувач кафедри
комп'ютерних систем та
мереж

_____(Жуков
І.А.)

« ____ »
2021 р.

З А В Д А Н Н Я
на виконання дипломного проекту

Калініченку Данилу Ігоровичу
(прізвище, ім'я, по батькові випускника в родовому відмінку)

1. Тема проекту (роботи): Система виявлення вторгнення в комп'ютерну мережу на основі аналізу трафіку
затверджена наказом ректора від "26" квітня 2023 року № 591/ст.

2.Термін виконання проекту (роботи): з 22.05.2023 до 25.06.2023

3.Вихідні дані до проекту (роботи):
1. Методи моніторингу та аналізу трафіку.
2. Системи виявлення вторгнення в мережу

4.Зміст пояснювальної записки (перелік питань, що підлягають розробці):
1. Моніторинг та аналіз роботи корпоративних мереж
2. Методи і засоби експерементального дослідження та аналізу трафіку.
3. Сучасні системи виявлення вторгненнь в комп'ютерну мережу

5.Перелік обов'язкового графічного матеріалу:
Презентація *PowerPoint*

6. Календарний план-графік

№ пор.	Завдання	Термін виконання	Відмітка про виконання
1	Узгодити технічне завдання	16.05.2023	
2	Оглянути літературу по темі дипломного проекту	17.05.2023-22.05.2023	
3	Проаналізувати особливості роботи та організації корпоративних комп'ютерних мереж	22.05.2023-24.05.2023	
4	Дослідити методи та засоби аналізу мережевого трафіку	25.05.2023-27.05.2023	
5	Розглянути принципи та особливості функціонування систем виявлення вторгнень в комп'ютерну мережу	28.05.2023-31.05.2023	
6	Підготувати дослідне середовище, зібрати зразки аномального трафіку, провести експериментальне виявлення мережевих атак	01.06.2023-03.06.2023	
7	Зробити висновок щодо отриманих результатів	04.06.2023-06.06.2023	
8	Оформити пояснювальну записку та презентацію	06.06.2023-10.06.2023	
9	Захистити дипломний проект	20.06.2023	

7. Дата отримання завдання « » 2023 р.

Керівник дипломного проекту Малярчук О.В.
(підпис)

Завдання прийняв до виконання Калініченко Д.І.
(підпис студента)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи «Виявлення вторгнення в комп’ютерну мережу на основі аналізу трафіку», 79 сторінок, 42 рисунки, 7 таблиць, 6 використані джерела.

Об’єкт дослідження – система захисту комп’ютерної мережі.

Мета дипломної роботи полягає у проведенні аналізу мережевого трафіку з метою виявлення аномалій, що можуть свідчити про можливі вторгнення в мережу. Додатково, робота спрямована на вивчення та оцінку сучасних систем виявлення атак в локальній мережі.

Методи дослідження – імітаційне комп’ютерне моделювання трафіку, методи математичної статистики.

Матеріали даної дипломної роботи можна використовувати при проведенні наукових досліджень, навчальному процесі та в практичній діяльності фахівців у сфері комп’ютерних мереж.

ОРГАНІЗАЦІЙНА МЕРЕЖА, ПЕРЕДАЧА ДАНИХ, АНАЛІТИЧНІ МЕТОДИ, СПОСТЕРЕЖЕННЯ, НАПАДИ, КЕРУВАННЯ, КОНЦЕПЦІЇ.

Зміст

ВИСНОВКИ

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

API – Application Programming Interface

FLM – Fractional Levi Motion

IDS – Intrusion Detection System,

IPS – Intrusion Prevention System,

NCP – The Network Control Protocol

SNA – Systems Network Architecture

UTM – Unified Threat Management

АРМ – автоматизоване робоче місце

ККМ – корпоративна комп'ютерна мережа

ММЕ – міжмережевий екран

МРМ – марківська рандомізована модель

СВА – система виявлення атак

СЗІ – система захисту інформації

ВСТУП

Активний прогрес мережевих технологій призводить до з'яви нових форм атак на комп'ютерні мережі. Розмаїтість методів вторгнень та їх використання в атаках ставлять під загрозу ефективність існуючих захисних технологій та засобів у сфері захисту даних в корпоративних комп'ютерних мережах. Отже, необхідно постійно вдосконалювати технології та інструменти захисту з метою забезпечення безпеки цих мереж.

Використання передових інформаційних технологій відіграє ключову роль у ефективному керуванні різноманітними системами та об'єктами. Корпоративні комп'ютерні мережі є незамінними інструментами, які забезпечують успішне функціонування цих систем. Однак, з ростом комп'ютерних мереж збільшується кількість користувачів та обсяг передаваної інформації. Тим не менш, збільшення мережевого трафіку може призвести до погіршення якості надання мережевих послуг. З цієї причини необхідно вдосконалювати інструменти моніторингу та аналізу мережевого трафіку, щоб забезпечити його надійність та якість.

Проблема аналізу мережевого трафіку вивчається протягом тривалого часу, і було проведено значну кількість досліджень, спрямованих на пошук ефективних рішень в різних умовах та обмеженнях. Особлива увага була приділена останніми роками, оскільки мережевий ландшафт зазнає швидких змін. Методи і алгоритми, які колись демонстрували добрі результати, втратили свою ефективність або стали непридатними в нових умовах.

Зміни в мережевому середовищі, такі як значне збільшення обсягу передаваного трафіку та пропускної здатності каналу зв'язку, суттєво впливають на придатність різних методів аналізу. У таких умовах виникає потреба у розробці алгоритмів, які зменшують обсяг обчислень.

Механізм виявлення вторгнень в систему базується на припущенні про стаціонарність мережевого трафіку, де відхилення від стаціонарних характеристик розглядаються як атаки. З цим уявленням важливо

продовжувати дослідження проблем аналізу мережевого трафіку та виявлення вторгнень в комп'ютерні мережі.

Метою дипломної роботи є проведення аналізу мережевого трафіку з метою виявлення можливих аномалій, які можуть свідчити про вторгнення в мережу, а також вивчення сучасних систем виявлення атак у локальній мережі.

Об'єктом дослідження є безпека передачі даних в корпоративній мережі, а предметом дослідження є сам трафік, який перетікає в цій мережі.

Для досягнення поставленої мети використовуються методи імітаційного комп'ютерного моделювання трафіку, а також методи математичної статистики. Ці методи дозволяють аналізувати та визначати особливості трафіку, шукати аномалії та потенційні загрози безпеці мережі.

За допомогою програмного інструменту OPNET Modeler були створені та змодельовані мережі, на основі яких були побудовані відповідні графіки. Ці графіки використовуються для аналізу характеристик корпоративних мереж та допомагають отримати уявлення про їхнє функціонування та ефективність.

РОЗДІЛ 1

АНАЛІЗ СИСТЕМ АНАЛІЗУ ТРАФІКУ ТА СИСТЕМ ЗАПОБІГАЮЧІ ВТОРГЕННЯ

1.1. Характеристики корпоративних комп'ютерних мереж

Перед тим, як зануритись у світ боротьби між вторгненням у мережу та програмами виявлення та усунення вторгнення необхідно почати з розбору самого поняття як комп’ютерна мережа, з чим може допомогти існуюча література від знавців свого діла.

Комп'ютерна мережа, або КМ, складається з взаємопов'язаних комп'ютерів, які сполучені каналами передачі даних. Ця мережа надає користувачам можливість обмінюватись інформацією та спільно використовувати апаратні, програмні та інформаційні ресурси[4].

Використання персонального комп'ютера без доступу до інформаційних ресурсів у мережі є неефективним, оскільки його функціональність суттєво обмежується. Ці ресурси можуть бути розташовані локально, у внутрішній мережі офісу або підприємства, або в глобальних мережах, включаючи Інтернет.

Комп'ютерна мережа складається з таких складових елементів:

1. Мережеве обладнання - це спеціальні пристрої, які з'єднують комп'ютерне обладнання (або будь-яке інше обладнання, що може працювати в мережі) у одну або кілька взаємодіючих систем.

2. Лінії зв'язку або канали передачі даних - це проміжне обладнання і фізичне середовище, через яке передаються інформаційні сигнали (дані). В залежності від фізичного середовища передачі даних, лінії зв'язку можуть бути провідними, кабельними або бездротовими.

3. Мережеве програмне забезпечення - це програми, які дозволяють організувати роботу користувачів у мережі. Воно включає загальне, системне та спеціальне програмне забезпечення.

Комп'ютерні мережі можуть бути класифіковані залежно від наступних характеристик:

1. За територією охоплення: локальні (*LAN*), регіональні (*WAN*), глобальні (*GAN*);
2. За топологією: шина, кільце, зірка, змішана топологія;
3. За середовищем передачі: вита пара, коаксіальний кабель, оптоволокно, телефонний кабель, радіозв'язок, супутниковий зв'язок;
4. За методом доступу до середовища передачі: конкурентний, детермінований з опитуванням або маркерним доступом;
5. За технологією: *Ethernet, Archnet, Token Ring, FDDI, SNA, Internet* та інші.

Комп'ютерні мережі можна умовно поділити на три групи залежно від їх універсальності та масштабу поширення:

1. Глобальна комп'ютерна мережа, відома як Інтернет, представляє собою всесвітню систему взаємопов'язаних комп'ютерних мереж, призначену для зберігання та передачі інформації.
2. Мережі Інтранет є локальними або регіонально розподіленими мережами, що відокремлені від зовнішнього доступу до Інтернету. Вони дозволяють використовувати публічні канали зв'язку, які належать до Інтернету, але забезпечують захист передаваних даних і приймають заходи для запобігання проникненню зовнішніх осіб до корпоративних вузлів.

Основні особливості корпоративних мереж:

1. Використання тих самих засобів і технологій, що і для локальних мереж загального призначення. Це означає, що корпоративна мережа може використовувати стандартні протоколи і технології, такі як *Ethernet, Wi-Fi* тощо.

2. Відокремлення внутрішньої мережі організації від глобальних мереж за допомогою міжмережевого екрану (*МЕ*) або інших методів безпеки. Це дозволяє контролювати доступ до інформації і забезпечувати безпеку внутрішньої мережі.
3. Різні типи інформації, що передається в мережі: офіційна, групова та неофіційна. Офіційна інформація загальна для всієї організації і призначена для всіх співробітників. Групова інформація призначена для конкретної групи або відділу в межах організації. Неофіційна інформація включає особисту інформацію співробітників і може бути обмеженою доступністю.
4. Централізована система управління мережею. Корпоративна мережа має централізовану систему управління, яка забезпечує ефективність функціонування, безпеку та життєздатність мережі. Це може включати системи моніторингу, управління доступом, резервне копіювання даних та інші функції.

Загалом, корпоративні мережі створюються для задоволення потреб і вимог конкретної організації та є важливим елементом для забезпечення її виробничо-господарської діяльності, управління та обміну інформацією.

1.2. Дослідження сучасних систем виявлення атак і захисту від вторгнень

Системи виявлення вторгнень (СВВ) є невід'ємною частиною захисту інформації в корпоративних мережах. Завдяки їм можна виявити мережеві атаки, спроби несанкціонованого доступу та використання мережевих ресурсів. З огляду на зростаючу складність, обширність і інтенсивність атак, розробка та вдосконалення СВВ є актуальною задачею.

СЗІ складаються з програмного та апаратного забезпечення, які базуються на математичних методах та моделях. Важливим елементом їх надійності є перевірка програмного коду та методів, що використовуються в системі. Верифікація здійснюється за допомогою математичних методів відповідно до заданих критеріїв ефективності СЗІ. Аналіз цих критеріїв дозволяє оцінити якість реалізації окремих елементів та всієї системи.

Проте, розвиток шкідливого програмного забезпечення та складність атак вимагають постійного вдосконалення СВВ. Порівняльний аналіз існуючих систем виявлення атак та протидії вторгненням є важливим кроком для визначення найефективніших механізмів захисту інформаційних активів. Враховуючи постійну зміну атак, системи виявлення вторгнень повинні бути гнучкими та вміти адаптуватись до нових загроз шляхом оновлення правил та моделей.

Узагальнюючи, для забезпечення безпеки інформації в корпоративних мережах використовуються СВВ, які постійно вдосконалюються і адаптуються до змінних загроз. Це дозволяє виявляти атаки та запобігати несанкціонованому доступу до інформації компаній.

З огляду на насиченість ринку інформаційних технологій такими системами, користувачам стає необхідно вибрати оптимальну систему для виявлення атак та запобігання вторгненням. Однак це можливо лише шляхом аналізу поточного стану і прогнозу найближчого розвитку цих систем. На сьогоднішній день існує велика кількість систем, які позиціонуються як системи виявлення вторгнень (IDS) або системи запобігання вторгненням (IPS).

На практиці, зловмисники використовують різні комбінації атак. Наприклад, вони можуть використовувати мережеві сканери для виявлення топології мережі і сканери вразливостей для виявлення уразливих хостів. Знайдені уразливості на хостах використовуються для віддаленого виконання коду. Тому системи виявлення вторгнень повинні мати механізми для виявлення різних типів атак.

Виявлення атаки означає процес ідентифікації та реагування на підозрілі дії, спрямовані проти комп'ютерних або мережевих ресурсів, тоді як сама атака відноситься до будь-якої дії зловмисника, що створює загрозу через вразливості комп'ютерної системи.

Аналіз базових концепцій систем виявлення вторгнень (СВВ) дійсно допомагає зрозуміти, що ці системи зазвичай базуються на методах виявлення

аномалій та зловживань. Ці методи використовують моделі шаблонів або профілів нормальної поведінки для виявлення незвичайних дій або атак.

Аномалійні методи виявлення орієнтовані на виявлення невідомих атак або втручань у комп'ютерну систему, використовуючи моделі нормальної поведінки. Вони використовують статистичні методи, нейронні мережі, теорію масового обслуговування та інші підходи для створення цих моделей. Однак, одним з недоліків моделей шаблонів нормальної поведінки, які базуються на статистичних методах виявлення, є велика кількість помилково позитивних результатів системи.

Помилково позитивні результати можуть бути спричинені помилками першого типу, коли система пропускає атаку або незвичайну дію, а також помилками другого типу, коли система неправильно спрацьовує і виявляє нормальну дію як аномальну. Це може виникати через складність створення точних моделей нормальної поведінки, різноманітність мережевих активностей та змінність у шаблонах атак.

Для зменшення помилково позитивних результатів і поліпшення ефективності СВВ використовуються додаткові техніки, такі як комбінація різних методів виявлення, використання спеціалізованих алгоритмів та аналіз додаткової інформації, наприклад, контекстуальних даних. Важливим є також постійне оновлення правил і моделей СВВ з урахуванням нових загроз і атак, що постійно змінюються.

Загалом, системи виявлення вторгнень є важливим компонентом комплексних заходів забезпечення безпеки інформаційних ресурсів системи. Вони доповнюють стандартні засоби захисту, забезпечуючи виявлення незвичайних або шкідливих дій, які можуть бути пропущені іншими захисними засобами.

В табл. 1.1 наведено основні механізми реалізації різних видів атак .

Таблиця 1.1

Основні механізми реалізації атак

№ з/п	Вид нападу	Процес здійснення нападу

1	Віддалене проникнення	Віддалений виклик командного рядка шляхом переповнення буфера
2	Аналіз топології мережі	Передача мережних пакетів, що містять запити ECHO_REQUEST
3	Пошук вразливості	Сканування хосту
4	Відмова в обслуговуванні	Передача великої кількості мережних пакетів
5	Злам паролів	Багаторазові спроби аутентифікації в системі
6	Аналіз мережного трафіка	Перемикання мережного інтерфейсу в “режим прослуховування” і перехоплення мережного трафіка
7	Несанкціонована аутентифікація	Порушення прав доступу і незаконне використання ресурсів
8	Шкідливе ПЗ	Приховане встановлення програмних модулів, прихований запуск процесів

Ризик атак можна виявити або зменшити, розпізнавши характеристики недозволених дій або механізмів атаки, таких як:

- Повторюваність певних подій у системі;
- Некоректні або суперечливі процеси та команди;
- Використання уразливих місць;
- Несприятливі параметри мережевого трафіку;
- Непередбачувані атрибути;
- Додаткові знання про порушення.

Таблиця 1.2 містить найважливіші механізми виявлення вторгнень, специфічні для різних типів атак.

Таблиця 1.2

Основні механізми виявлення атак

№ з/п	Механізми виявлення атаки	Клас атак, що виявляються

1	Моніторинг процесу аутентифікації в системі	Зовнішні (внутрішні) мережні (локальні) активні
2	Моніторинг зловмисного перехоплення мережевого трафіку	Зовнішні мережні активні
3	Аналіз мережевого трафіку	Зовнішні мережні пасивні
4	Моніторинг виконання процесів та доступу до файлової системи й реєстру	Внутрішні локальні активні

1.3. Приклади систем аналізу та захисту

Отже залишається питання, які програми здатні аналізувати трафік, захищати мережу від несанкціонованого доступу та аналіз мережі на вразливість.

Можна почати з систем аналізу трафіку , завдяки якому можна займатися моніторингом вхідного та вихідного трафіку, що дозволяє виявити «шкідливий» або трафік з-зовні комп'ютерної мережі, а цих програм є декілька:

1) SolarWinds Network Bandwidth Analyzer

Рис.1.1. Головне вікно програми Solarwinds

Network Performance Monitor є базовим рішенням, спрямованим на моніторинг продуктивності мережі. Він надає загальне уявлення про стан та ефективність мережі шляхом збору та аналізу різноманітних статистичних даних. За допомогою Network Performance Monitor ви зможете контролювати швидкість та надійність передачі даних та пакетів у вашій мережі. Це дозволить виявляти несправності та проблеми, які можуть виникати в роботі мережі.

Програма також пропонує просунуті інтелектуальні можливості для виявлення потенційних проблем. Вона забезпечує візуальне представлення результатів у вигляді таблиць та графіків, що дозволяє зрозуміти стан мережі та отримати чіткі попередження про можливі проблеми. Це полегшує процес аналізу та виявлення проблем в мережі.

NetFlow Traffic Analyzer є модульним розширенням, яке доповнює можливості Network Performance Monitor. Він спеціалізується на аналізі мережевого трафіку, зокрема використовує протокол NetFlow для збору даних про трафік. Цей модуль дозволяє отримати детальну інформацію про типи трафіку, використання пропускної здатності, статистику передачі даних та інші параметри, що стосуються трафіку в мережі. Застосування NetFlow Traffic Analyzer допомагає виявляти аномалії, витоки даних, надмірне навантаження та інші проблеми, пов'язані з мережевим трафіком.

Спільне використання Network Performance Monitor та NetFlow Traffic Analyzer дозволяє отримати комплексну інформацію про стан мережі та аналізувати як загальну продуктивність, так і деталізовану інформацію щодо мережевого трафіку.

Використовуючи ці два продукти разом, ви зможете отримати більш повне уявлення про функціонування вашої мережі та швидше виявляти потенційні проблеми.

2) Wireshark

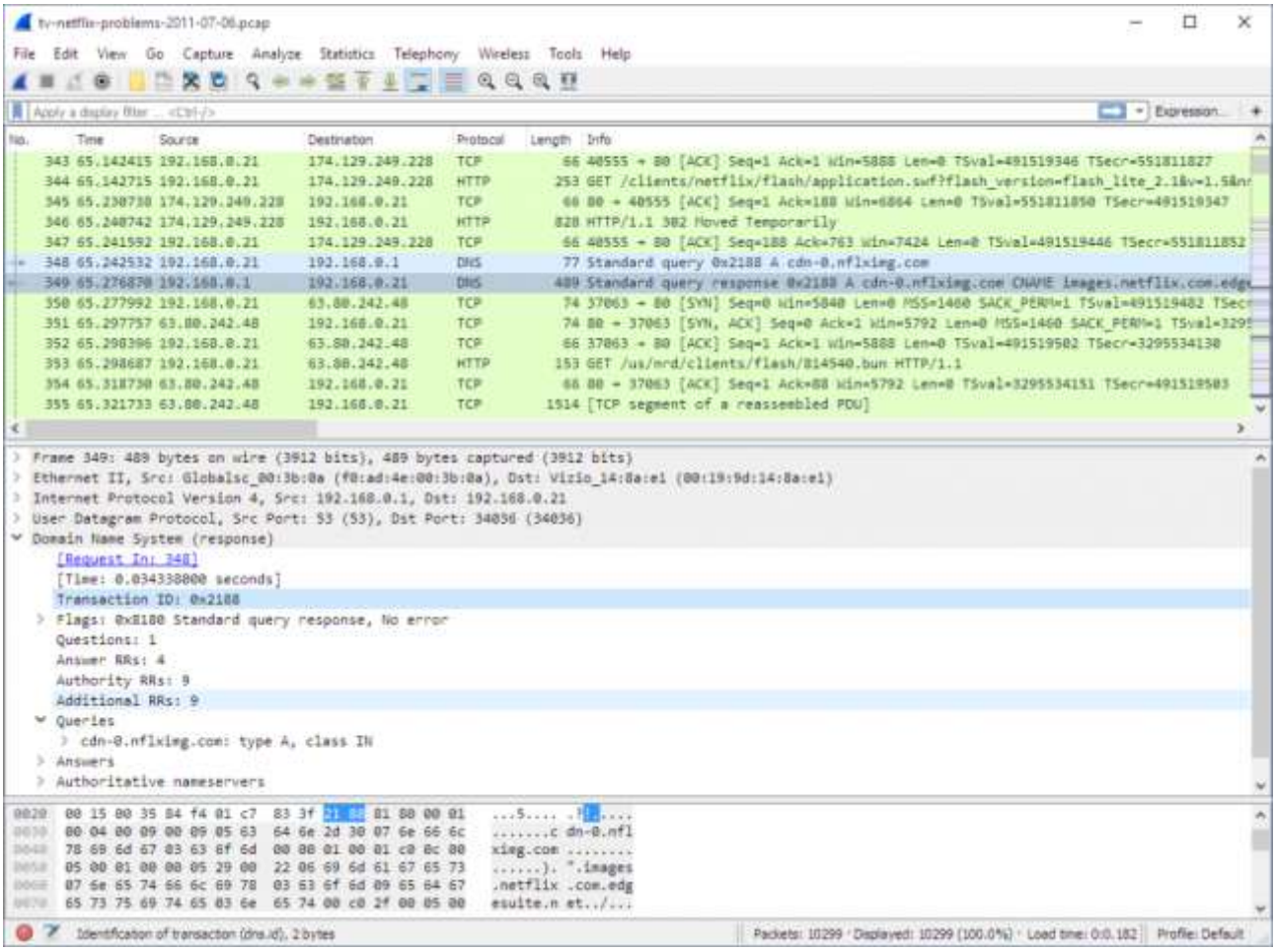

Рис. 1.2. Головне вікно програми WireShark

WireShark є потужним інструментом для аналізу мережевого трафіку і отримав визнання серед IT-професіоналів. Він є відносно новим у сім'ї рішень для мережної діагностики, але вже здобув популярність завдяки своїм функціональним можливостям.

WireShark надає безліч опцій для фільтрації та сортування даних, що дозволяє зосередитися на необхідній інформації. Користувачі з різних сімейств операційних систем, таких як *NIX, Windows і macOS, можуть успішно використовувати WireShark, оскільки він підтримує всі ці платформи.

Завдяки своїй простоті, сумісності та портативності, WireShark стає зручним інструментом для аналізу трафіку. Використовуючи WireShark, ви можете швидко отримати необхідну інформацію та аналізувати трафік вашої мережі з високою точністю.

Загалом, WireShark є потужним, зручним і надійним інструментом для аналізу мережевого трафіку, який зарекомендував себе серед IT-професіоналів. Його широкі можливості і сумісність з різними операційними системами роблять його привабливим вибором для вивчення та аналізу мережевої активності.

Програми для запобігання вторгнення вже мають більш складну систему, адже на відміну від систем виявлення вторгнень вони мають окрім того, що аналізувати трафік, так ще й запобігти йому потрапляння у локальну мережу в тих випадках , коли цей трафік є шкідливим, та намагається авторизуватися у мережу, скориставшись вразливістю мережі. Через об’єм роботи, що необхідно виконувати програмі, це дається взнаки на швидкодії самої мережі, але задля захисту своєї мережі воно того варте.

Загалом, для домашніх користувачів ПК звичайні антивіруси виступають у ролі систем IPS, тобто, системи запобігання вторгненню. Але для корпоративних мереж окрім системного антивірусу, має бути спеціальний IPS, який буде рятувати мережу у випадках, коли зловмисники користуються вразливістю мережі.

1) Snort

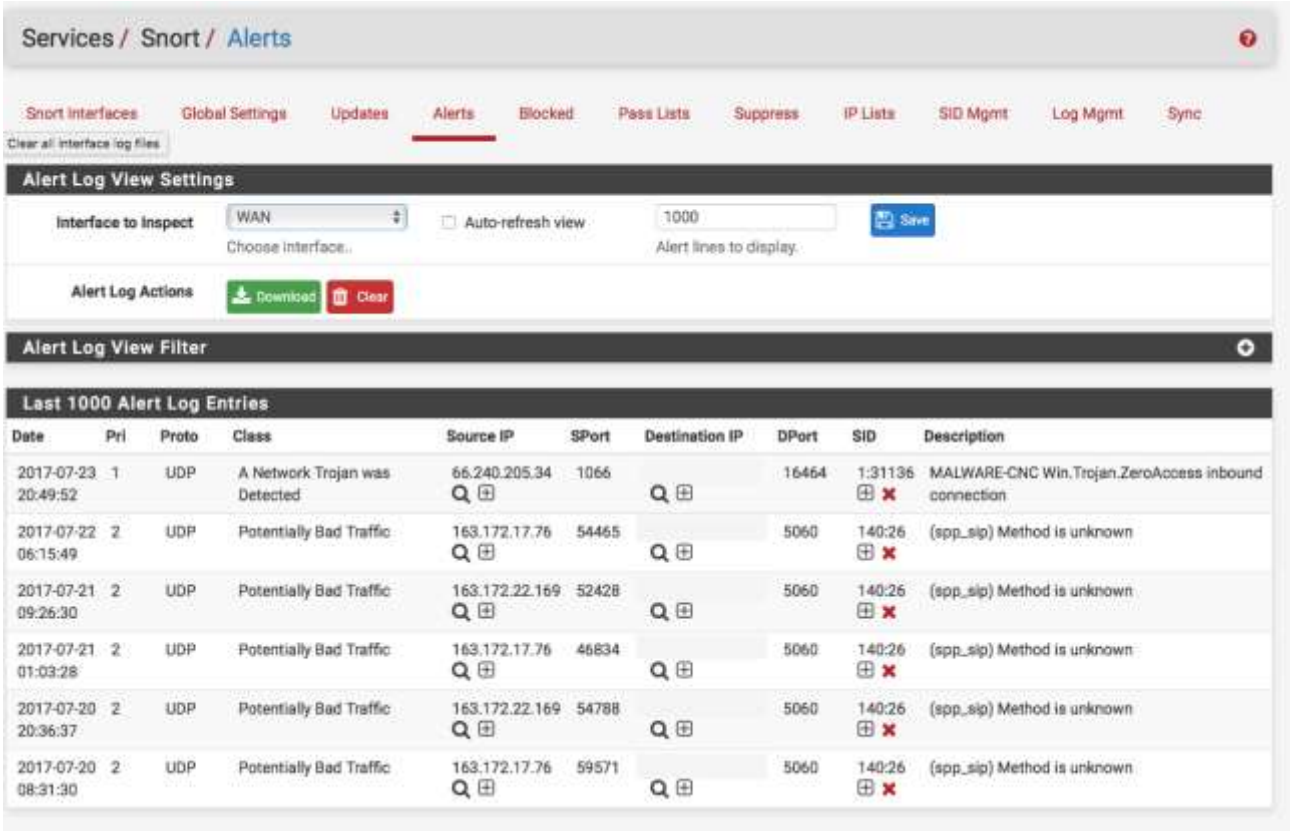

Рис. 1.3. Вікно попереджень програми Snort

Snort, який зображений на ілюстрації 1.3, є відкритим системою запобігання вторгнень (IPS) і вважається однією з найкращих у світі. Він використовує набір правил, що допомагають виявити зловмисну мережеву активність, та використовує ці правила для пошуку пакетів, що збігаються з ними, і генерує сповіщення для користувачів.

Snort може бути розгорнутий в мережі для зупинки зловмисних пакетів. Ця система має три основних функції. По-перше, вона може працювати як сніффер пакетів, що дозволяє перехоплювати та аналізувати мережевий трафік, подібно до програми tcpdump. По-друге, Snort може функціонувати як реєстратор пакетів, що корисно для збереження даних про мережевий трафік для подальшого аналізу та налагодження. Нарешті, Snort може використовуватись як повномасштабна система запобігання вторгнення в мережу, де він аналізує пакети та приймає заходи для запобігання потенційним атакам.

Snort доступний для завантаження та налаштування для особистого використання або використання в бізнесовому середовищі. Він має відкритий код, що дозволяє користувачам налаштовувати його згідно зі своїми потребами. Завдяки своїм можливостям і гнучкості, Snort є потужним інструментом для захисту мережі від потенційних загроз та вторгнень.

Загалом, Snort є високоефективною системою запобігання вторгнень з великим спектром функціональності, яка заслужила свою популярність серед користувачів, що займаються безпекою мережі.

2) Suricata

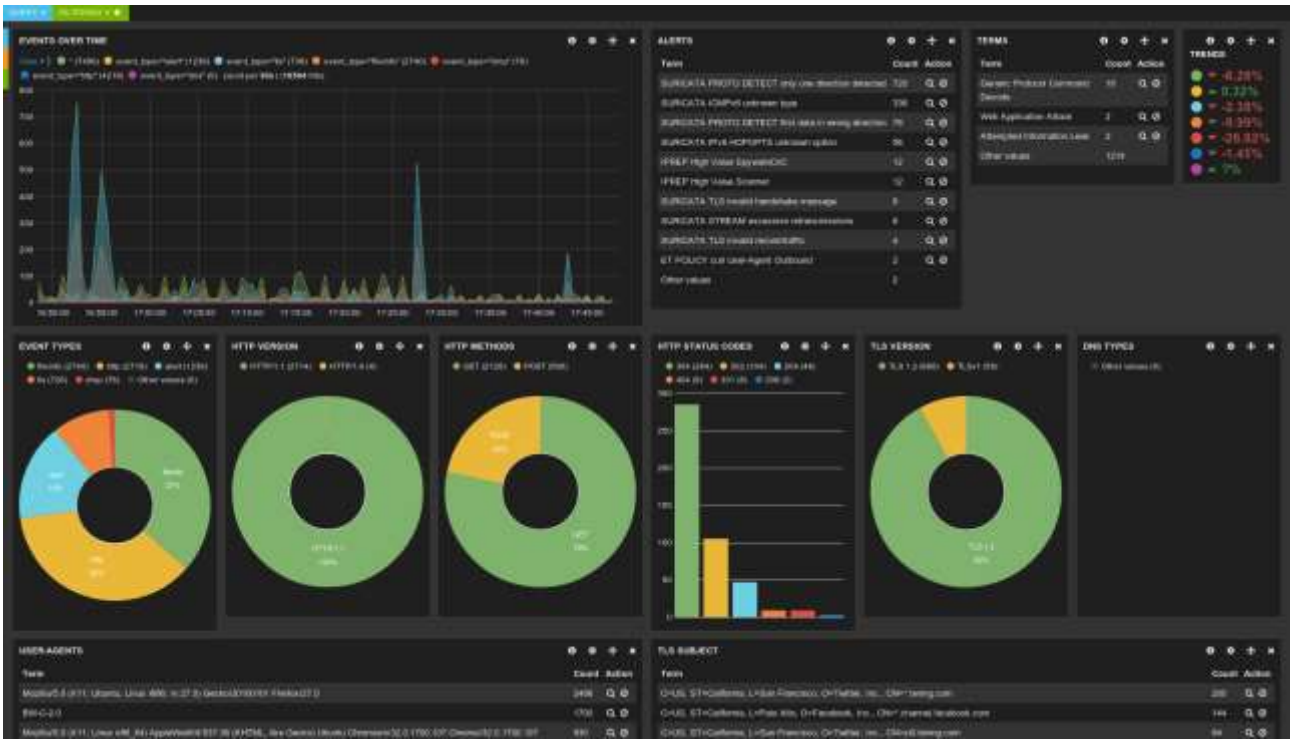

Рис. 1.4. Головне вікно програми Suricata

Suricata (на ілюстрації 1.4) - open source IPS/IDS система. Заснована розробниками, які працювали над IPS версією Snort. Основна відмінність Suricata від Snort - можливість використання GPU в режимі IDS, більш просунута система IPS, багатозадачність, як наслідок, висока продуктивність, що дозволяє обробляти трафік до 10Gbit на звичайному устаткуванні, і багато іншого, у тому числі повна підтримка формату правил Snort.

Висновки за розділом

Аналізуючи основні методи аналізу мережевого трафіку та моделі мережевого трафіку, можна зробити висновок, що ці методи і моделі широко використовуються в готових апаратних та програмно-апаратних комплексах для запобігання вторгненням у комп'ютерну мережу.

Один з прикладів систем моніторингу трафіку - сніфери, проте вони мають свої недоліки. Серед них можна відзначити складність використання для недосвідчених користувачів, складний інтерфейс, обмежену сумісність з різними операційними системами, а також відсутність графічного інструменту для відображення стану мережі у певний момент часу. Відсутність графічного представлення інформації ускладнює процес аналізу мережевого трафіку.

Не існує універсальних вимог щодо вибору систем виявлення та запобігання вторгненням (IDPS); вибір системи залежить від конкретних вимог безпеки у кожному випадку. Проведений аналіз систем виявлення атак і запобігання вторгненням дозволяє розглянути їх особливості, відмінності та властивості різних типів таких систем, а також механізми реалізації кібератак. З урахуванням цих даних важливо вдосконалювати та налаштовувати СВВ відповідно до особливостей функціонування та реалізації конкретної комп'ютерної мережі, де забезпечується захист.

РОЗДІЛ 2

РОЗРОБКА ТА ВИПРОБОВУВАННЯ IPS-СИСТЕМИ, ПОБУДОВАНІ НА SNORT ТА SURICATA

2.1. Дослідження методів виявлення вторгнень в комп'ютерну мережу

Проаналізували методи виявлення вторгнень у комп'ютерну мережу і відібрали основні з них[5]:

Аналіз сигнатур. Метод аналізу сигнатур виявлення атак базується на ідентифікації специфічних параметрів, подій або дій, що вказують на спробу атаки. Цей метод порівнює поточний стан системи та її дії з відомими сигнатурами, які зберігаються у базі даних. Аналіз сигнатур має переваги в швидкості роботи і малій ймовірності помилок, оскільки він використовує наявні дані про відомі атаки. Проте, його недолік полягає у неможливості виявлення нових атак, які не мають відповідних сигнатур у базі даних.

Статистичний аналіз. Метод статистичного аналізу виявлення атак ґрунтується на створенні статистичних профілів системи, які містять набір параметрів та допустимі значення для нормальної поведінки системи. Цей метод спирається на виявлення атаки, якщо поведінка захищеної системи відхиляється від статистичного профілю або моделі. Метод статистичного аналізу має перевагу у своїй адаптивності, оскільки може виявляти невідомі атаки. Однак, недоліками цього методу є висока ймовірність помилкових спрацьовувань, а також той факт, що зміни в діяльності об'єкта не обробляються, що може призводити як до помилкових спрацьовувань, так і до пропущених атак.

Аналіз систем станів. Метод аналізу систем станів виявлення атак базується на описі роботи захищеної системи як послідовності станів та переходів між ними. Стан роботи системи інтерпретується у вигляді спрямованого графа, який часто має нескінченну кількість вершин. У графі

існують недопустимі шляхи, які вказують на небезпечний кінцевий стан для захищеної системи. Метод аналізу системи станів спрямований на виявлення відомих неприпустимих шляхів у графі станів системи для виявлення атак. Успішне виявлення атаки передбачає розпізнавання послідовності переходів, яка призводить до небезпечного стану. Проте, обмеженням цього методу є неможливість виявити атаку, якщо послідовність системних станів перекривається.

Графи сценаріїв атаки. Метод графів сценаріїв атаки включає створення графа, який описує всі відомі сценарії атак на основі характеристик стану системи. Цей метод використовує формальний опис захищеної системи та визначає властивість коректності системи. Метод графів сценаріїв атаки створює повний набір можливих неприйнятних поведінок для конкретної системи, що дозволяє описати потенційні шляхи атаки. Цей метод корисний для виявлення слабких місць у структурі системи. Проте через високу обчислювальну складність, він не є ефективним для виявлення фактичних вторгнень.

Експертні системи. Метод експертних систем для виявлення атак ґрунтується на описі роботи системи через низку фактів та правил висновку. Коли введені дані про спостережувані події у системі у вигляді фактів, експертна система використовує ці факти та правила, щоб визначити, чи має місце атака. У загальному, цей метод вимагає великих обчислювальних зусиль, оскільки потребує розгляду багатьох можливих варіантів.

Методи, засновані на специфікаціях. Методи, засновані на специфікаціях, використовують опис обмежень, які визначають недозволену поведінку об'єктів у захищеній системі у вигляді специфікацій атаки. Специфікація може містити обмеження на завантаження ресурсів, заборонені операції та їхній спосіб роботи, а також обмеження щодо часу, коли діють певні обмеження. Якщо об'єкти порушують ці специфікації, це вважається атакою. Головним недоліком цього методу є необхідність постійного оновлення специфікацій у зв'язку з еволюцією загроз та нових атак.

Нейронні та імунні мережі. Нейромережі та імуномережі застосовуються для розв'язання проблеми виявлення вторгнень, яка може бути розглянута як завдання розпізнавання патернів або класифікації. У випадку імуномереж, система створює раніше невідомі сигнатури, які емулюють негативний механізм відбору, і порівнює їх зі звичайним профілем. Нейромережі, з іншого боку, моделюють захищену систему та її зовнішні об'єкти як траєкторії в числовому просторі атрибутів. У якості методу виявлення зловживань, нейромережі навчаються на прикладах атак кожного класу, а потім використовуються для визнання того, до якого класу атак належить спостережувана поведінка.

Груповий аналіз. Груповий аналіз використовується для виявлення атак шляхом розподілу властивостей спостережуваних векторів системи на кластери. Кластери, які відповідають нормальній поведінці системи, виділяються серед цих кластерів. Кожен метод кластерного аналізу використовує свою метрику для оцінки, чи належить системний вектор властивостей до одного з кластерів, чи виходить за межі відомих кластерів. Метод кластерного аналізу подібний до методу статистичного аналізу. За допомогою результатів кластерного аналізу можна визначити, чи спостерігається в системі аномальна поведінка, яка може свідчити про атаку. Таблиця 2.1, яка наведена у вашому запиті, містить результати порівняльного аналізу методів ідентифікації різних класів атак. Ця таблиця надає порівняння ефективності різних методів виявлення атак для різних класів.

Згідно з аналізом, як показано в таблиці 2.1, адаптивні методи виявлення вторгнень, такі як статистичний аналіз, графи сценаріїв атак, експертні системи, нейронні мережі, імунні мережі, кластерний аналіз та біометрична поведінка, виявилися ефективними у виявленні різних класів атак. Ці методи можуть адаптуватися до змін у вторгненнях та постійно покращувати свою продуктивність. Застосування адаптивних методів виявлення вторгнень є важливим, оскільки кількість нових та раніше невідомих атак зростає щороку. Ці методи дозволяють системам виявляти невідомі атаки та пристосовуватися до нових загроз. З урахуванням цих результатів, можна стверджувати, що

використання комбінації різних адаптивних методів виявлення вторгнень може бути ефективним підходом для забезпечення безпеки системи та виявлення нових атак.

Таблиця 2.1

Результати аналізу методів виявлення вторгнень

Методи	Рівень спостереження	Верифікація	Адаптивність	Стійкість	Обчислювальна складність
Аналіз сигнатур	Хост, мережа, додатки	Так	Ні	глобальна	O(log n)
Статичний аналіз	Хост, мережа	Ні	Так	локальна	O(n)
Аналіз систем станів	Хост, мережа, додатки	Так	Ні	локальна	O(n)
Графи сценаріїв атаки	Хост, мережа, додатки	Так	Так	локальна	NP
Експертні системи	Хост, мережа	Так	Так	глобальна	NP
Методи, засновані на специфікаціях	Мережа	Так	Ні	локальна	O(log n)
Нейронні мережі	Хост, мережа, додатки	Так	Так	локальна	O(n)
Імунні мережі	Хост, мережа	Ні	Так	локальна	O(n)
Кластерний аналіз	Хост, мережа, додатки	Ні	Так	локальна	O(n)
Поведінкова біометрія	Хост	Ні	Так	локальна	O(n)

Методи, які базуються на створенні графів сценаріїв атак та використанні експертних систем, практично не використовуються в існуючих системах для виявлення та запобігання вторгненням через великі обчислювальні витрати, які необхідно вкласти у їх реалізацію. Методи, які базуються на нейронних

мережах, мають перевагу адаптивності та низької обчислювальної складності, але для виявлення нових типів атак потрібно формувати навчальний набір для навчання системи. Якщо навчальний набір містить помилки або неправильні дані, це може призвести до неефективної роботи системи виявлення вторгнень. З інших методів адаптивного виявлення вторгнень метод статистичного аналізу та аналогічний метод кластерного аналізу виявляються досить ефективними, але існує ймовірність отримання помилково позитивних результатів.

2.2. Аналіз використання статистичних методів для виявлення вторгнень у комп'ютерні мережі з метою оцінки їх ефективності та можливостей застосування.

При виконанні аналізу статистичних характеристик аномальних вторгнень, потрібно обчислити різні статистичні показники для кожного нового набору даних. Ці показники включають наступні статистичні характеристики:

– вибіркове середнє

$$m_i = \frac{1}{n}\sum_{j=i}^{i+n} S_j,$$

де S – показник активності мережевого трафіка;

– вибіркова дисперсія

$$D = \frac{1}{n-1}\sum_{j=1}^{i+n}(S_j m_i)^2;$$

– коефіцієнт асиметрії

$$K_a = \frac{\frac{1}{n}\sum_{j=i}^{i+n}(S_j m_i)^3}{D^3};$$

– коефіцієнт ексцесу

$$K_a = \frac{\frac{1}{n}\sum_{j=i}^{i+n}(S_j m_i)^4}{D^4} - 3;$$

– контрексцес η

$$K_o = \frac{1}{\sqrt{\eta}},$$

де η – параметр ексцесу

$$\eta = Ke.$$

Для порівняння розподілів, які були створені для кожної статистичної характеристики, застосовується критерій згоди Пірсона. Цей критерій дозволяє визначити, чи існує лінійна залежність між двома розподілами. Він оцінює ступінь схожості чи відхилення між цими розподілами, що дозволяє зробити висновок про їхню подібність або різницю.

$$r_{xy} = \frac{\sum_{i=1}^{m}(x_j - \bar{x})(y_j - \bar{y})}{\sqrt{\sum_{i=1}^{m}(x_j - \bar{x})^2}\sqrt{\sum_{i=1}^{m}(y_j - \bar{y})^2}} = \frac{\mathrm{cov}\,(x,y)}{\sqrt{S_x^2 S_y^2}}.$$

Мережева атака виявляється шляхом виявлення збільшення варіації мережевої активності, коли заражені пристрої суттєво змінюють характеристики вхідного мережевого трафіку, що в свою чергу проявляється у змінах статистичних показників.

Під час наступного порівняння з визначеними еталонними значеннями статистичних характеристик виявляється незвичайна поведінка мережевого трафіку, що вказує на можливість мережевої атаки.

Атака типу HTTP-flood може бути використана для прикладу, щоб проілюструвати поведінку статистичних характеристик. В цьому прикладі атака починається о 12-й секунді і триває 45 секунд. Статистичний аналіз виконується протягом цього періоду. Дані статистичних характеристик оновлюються на 57-й секунді. Ці характеристики можуть відображати зміни, пов'язані з інтенсивністю та обсягом мережевого трафіку під час атаки HTTP-flood.

На основі рисунку 2.1 можна спостерігати, що показник вибіркового середнього у момент атаки починає зростати лінійно і досягає свого максимуму. Таке зростання та наступне стабілізування показника можуть свідчити про зміну у середньому значенні мережевого трафіку під час атаки HTTP-flood.

З рисунку 2.2 видно, що показник вибіркової дисперсії під час атаки починає квадратично зростати, досягає свого максимального значення, а потім починає зменшуватися до середнього показника між моментом атаки і повним

оновленням даних статистичного аналізу. Така залежність свідчить про збільшення розсіювання значень мережевого трафіку під час атаки, а потім його поступове зменшення до нормального рівня після атаки.

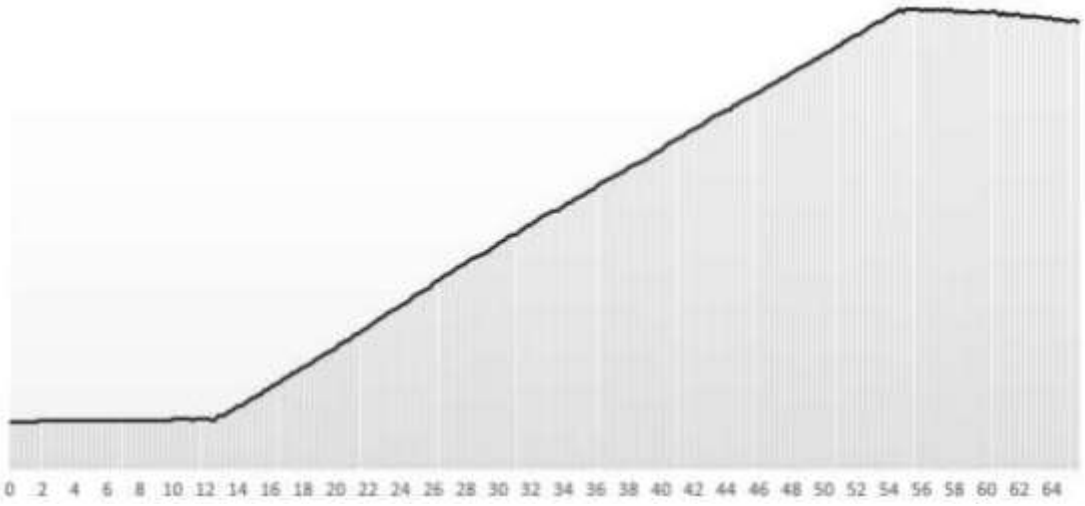

Рис. 2.1. Розподіл значень вибіркового середнього

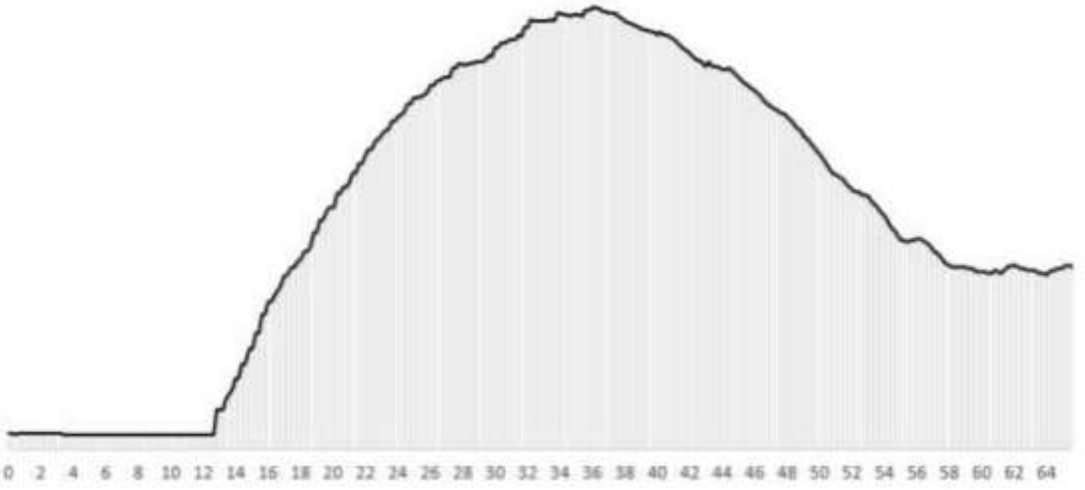

Рис. 2.2. Розподіл значень вибіркової дисперсії

З рисунку 2.3 видно, що в момент здійснення атаки показник коефіцієнта асиметрії різко зростає до свого максимального значення. Після цього спостерігається експоненціальне зменшення показника, і він досягає свого мінімального значення до моменту повного оновлення даних статистичного аналізу.

Згідно з рисунком 2.4, можна помітити, що під час атаки коефіцієнт контрексцесу різко знижується і досягає свого мінімального значення. Після

цього спостерігається квадратичне зростання показника, яке досягає свого максимального значення, а потім поступово повертається до початкового рівня.

Для оцінки розподілу статистичних параметрів необхідно мати функцію розрахунку коефіцієнта кореляції Пірсона. Ця функція дозволяє порівняти сформований розподіл кожної статистичної характеристики з відомими розподілами випадкових величин. Це особливо важливо при аналізі мережевих атак, оскільки вони часто проявляються через зміни у розподілі цих характеристик. Вірне порівняння допомагає виявити аномальність розподілу та встановити наявність мережевої атаки.

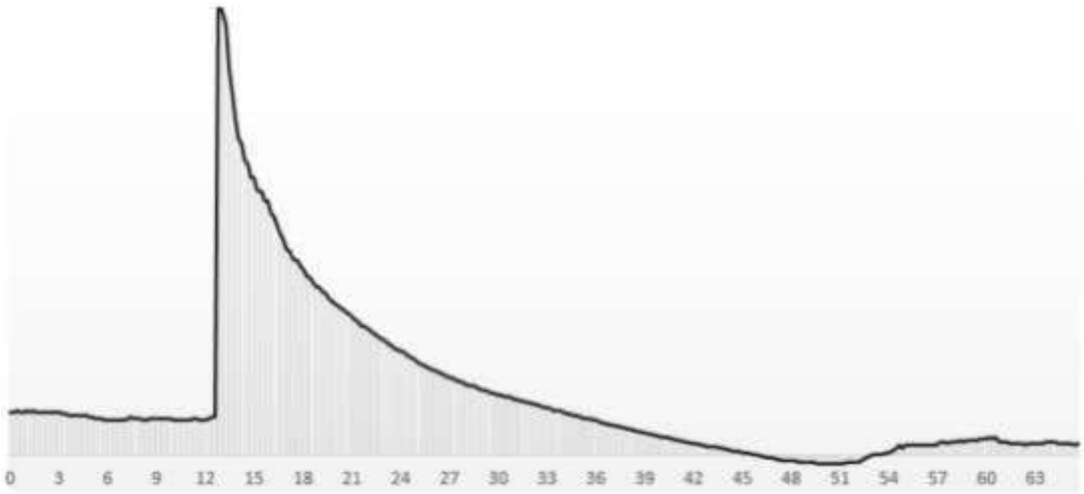

Рис. 2.3. Розподіл значень коефіцієнта асиметрії

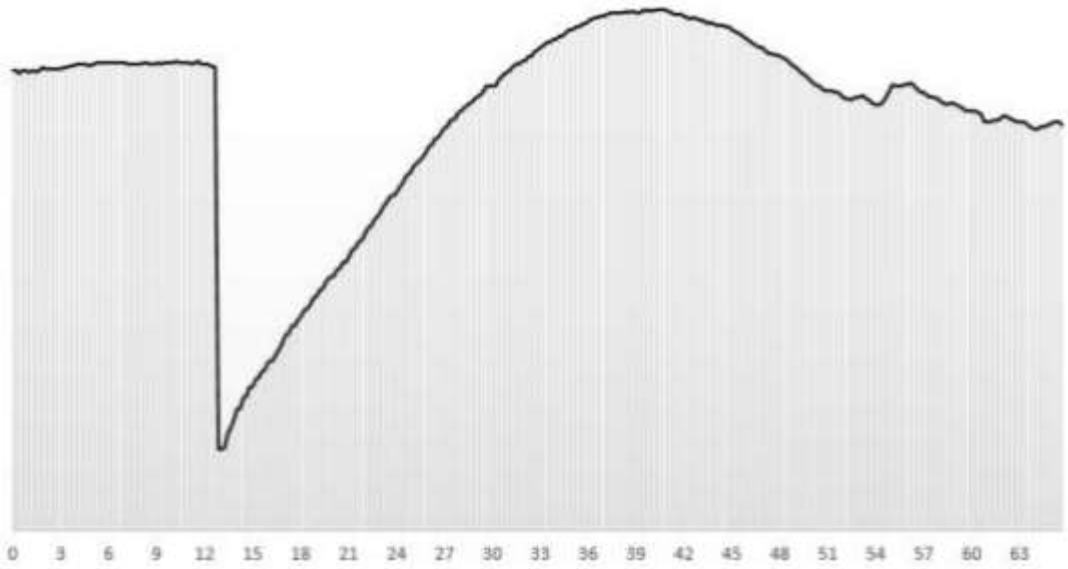

Рис. 2.4. Розподіл значень коефіцієнта контрексцесу

На рисунку 2.5 можна спостерігати розподіл значень дисперсії під час атаки типу HTTP-flood. Атака на мережу розпочалася 28 секунд після початку спостереження і тривала до 147-ої секунди. Виявлення атаки відбулося на 120-ій секунді, що становить 92 секунди після початку атаки.

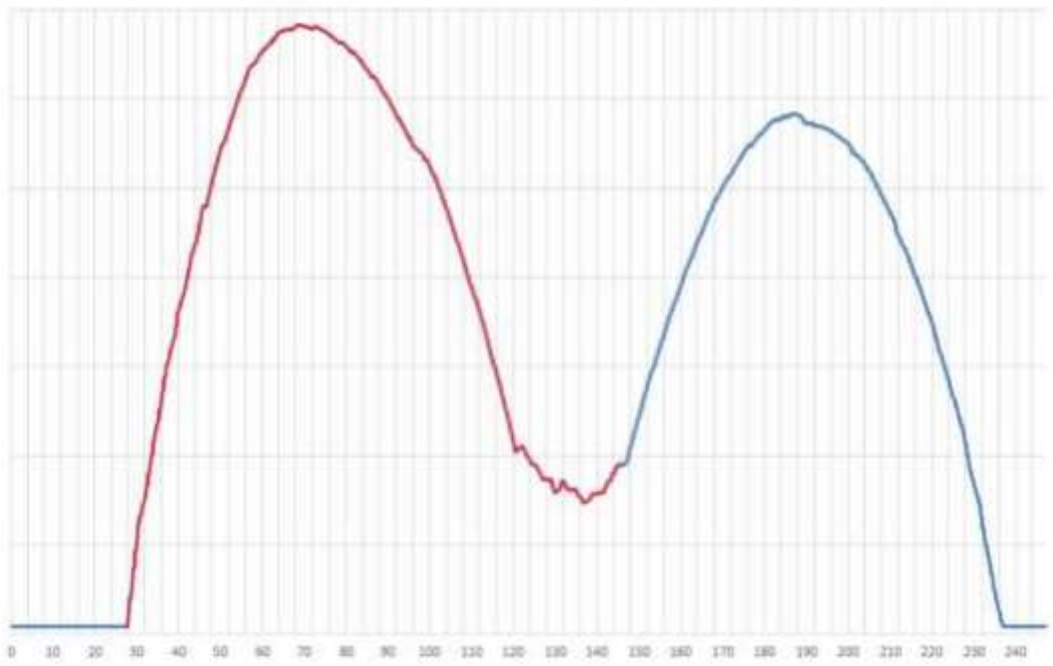

Рис. 2.5. Значення вибіркової дисперсії

Аналізуючи отримані результати, можна зробити висновок про ефективність статистичних методів при виявленні мережевих атак. За допомогою вивчених статистичних характеристик, таких як вибіркове середнє, дисперсія, коефіцієнт асиметрії та контрексцесу, було можливо виявити відхилення в поведінці мережевого трафіку, що свідчить про наявність мережевих атак. Розрахунки коефіцієнта кореляції Пірсона також підтвердили це, показуючи високу кореляцію між спостережуваним розподілом та нормальним розподілом. Таким чином, статистичні методи виявлення мережевих атак виявилися ефективними, оскільки здатні виявляти відхилення в статистичних характеристиках, які є показниками незвичайної активності мережі. Це дозволяє вчасно виявляти та реагувати на мережеві атаки, забезпечуючи більшу безпеку та захист мережевої інфраструктури.

2.3. Обґрунтування вибору систем виявлення вторгнень

На ринку інформаційної безпеки доступний широкий спектр продуктів і послуг, які пропонуються різними виробниками. Цей асортимент включає як невеликі програмні проекти з відкритим вихідним кодом, так і комплексні системи захисту інформації від відомих постачальників. Однак, деякі виробники продуктів інформаційного захисту впроваджують підхід, коли вони додають додаткові модулі, які виконують функції системи виявлення вторгнень (IDS), до продукту, який зазвичай використовується як брандмауер. Ці модулі, як правило, мають обмежену функціональність та продуктивність, оскільки вони не є самостійними системами.

Здійснено порівняльний аналіз систем виявлення вторгнень (IDS), що в даний момент доступні на ринку інформаційної безпеки, як загальнодоступні рішення або продукти, що пропонуються виробниками. Результати цього порівняння наведені в таблиці 2.2.

Таблиця 2.2

Порівняльний аналіз існуючих СВВ

Назва	ПЗ/ПАК	Тип сенсора	Спосіб збору даних	Аналіз результатів	Повнота документації	Вартість
KFSensor	ПЗ	HIDS	Сигнатурний	Ні	Ні	Платно
OSSEC HIDS	ПЗ	HIDS	Сигнатурний	Ні	Так	Безкоштовно
Snort	ПЗ	NIDS	Сигнатурний	Ні	Так	Безкоштовно
Suricata	ПЗ	HIDS/ NIDS	Сигнатурний	Ні	Так	Безкоштовно
EasyIDS	ПЗ	NIDS	Сигнатурний	Ні	Ні	Безкоштовно
Bro	ПЗ	NIDS	Сигнатурний	Ні	Так	Безкоштовно
Cisco IPS	ПАК	NIDS/ HIDS	Сигнатурний евристични	Так	Так	Платно

			й			
ViPNet IDS	ПАК	NIDS/ HIDS	Сигнатурний евристичний	Так	Так	Платно
McAfee IPS	ПЗ/ПАК	NIDS/ HIDS/ APIDS	Сигнатурний евристичний	Так	Так	Платно
Open Source Tripwire	ПЗ	NIDS/ HIDS	Сигнатурний	Ні	Так	Безкоштовно
IBM ISS Proventia IPS	ПЗ/ПАК	NIDS/ HIDS/ APIDS	Сигнатурний евристичний	Так	Так	Платно
OSSIM	ПЗ	HIDS	Сигнатурний евристичний	Так	Так	Платно

Апаратно-програмні комплекси мають вбудовані системи для аналізу даних та виведення результатів роботи. Проте, вартість таких систем може бути високою, що створює проблеми для їх використання в малих компаніях та підприємствах. Безкоштовне програмне забезпечення, у свою чергу, зазвичай не має вбудованої бази для аналізу даних та графічного інтерфейсу. Однак, для реалізації цих функцій можна використовувати сторонні програми.

2.4. Аналіз ефективності Snort і Suricata, інструментів виявлення і запобігання вторгнення

Snort є безкоштовним програмним забезпеченням з відкритим кодом, що випускається під ліцензією GPL. Воно було розроблене в 1998 році відомим експертом в галузі кібербезпеки, Мартіном Роше, який також є автором численних книг з цієї тематики. Головною метою створення системи IDS Snort було заповнення прогалини на той момент у сфері безпеки, шляхом розробки ефективного і безкоштовного інструменту для виявлення атак та передавання відповідних сповіщень[2].

Snort є потужним інструментом, який здатний виявляти різноманітні типи атак і аномального трафіку. Ось лише кілька прикладів того, що Snort може виявити:

- Поганий трафік, який може вказувати на незвичайну або підозрілу активність в мережі.
- Використання експлойтів та ідентифікація Shell-коду, що може свідчити про спроби злому системи.
- Сканування системи, включаючи сканування портів, операційної системи, користувачів тощо.
- Атаки на різні послуги, такі як Telnet, FTP, DNS і інші, які можуть виявити спроби несанкціонованого доступу до цих послуг.
- DoS/DDoS-атаки, які спрямовані на перевантаження системи або мережі шкідливим трафіком.
- Атаки, пов'язані з веб-серверами, такі як CGI, PHP, FrontPage, IIS та інші, які можуть ставити під загрозу безпеку веб-додатків.
- Атаки на бази даних, такі як SQL або Oracle, які можуть спробувати використати вразливості у системі керування базами даних.
- Атаки через протоколи SNMP, NetBIOS та ICMP, що можуть виявляти спроби зловмисників отримати несанкціонований доступ до мережевих пристроїв.
- Атаки на протоколи SMTP, IMAP, POP2, POP3, які можуть спробувати скомпрометувати поштові сервери або отримати несанкціонований доступ до поштових скриньок.
- Виявлення різноманітних "задніх дверей" або шкідливого програмного забезпечення, які можуть бути встановлені без дозволу користувача.
- Виявлення та блокування веб-фільтрів, що містять матеріали порнографічного характеру, забезпечуючи контроль над доступом до такого вмісту.

Це лише деякі з можливостей Snort для виявлення різноманітних загроз у мережевому середовищі.

Принцип роботи Snort показано на рис. 2.6.

Рис. 2.6. Принцип роботи Snort

Завдяки бібліотеці libpcap можна перехоплювати мережеві пакети, які надходять на мережеву карту до того, як вони будуть оброблені стеком протоколів. Ця бібліотека використовується при розробці програм, призначених для моніторингу та тестування мережі, таких як Snort, а також для реалізації сніферів, на приклад, WireShark [3].

Після перехоплення пакету він проходить через процес декодування, який відповідає за розбір протоколів канального рівня, таких як Ethernet або 802.11. Головна мета декодера полягає в розпакуванні даних, які містяться на мережевому і транспортному рівнях (наприклад, IP, TCP, UDP, ICMP). В результаті цього процесу пакет стає доступним для подальшого аналізу та обробки.

Препроцесор виконує підготовку даних протоколів на рівнях транспорту та мережі для подальшої обробки детектором. У системі Snort існують налаштування препроцесорів та їх правил, що дозволяє оптимізувати продуктивність системи в цілому. Це дозволяє забезпечити швидку обробку пакетів і збільшити ефективність виявлення потенційних загроз.

Детектор виконує аналіз перехоплених даних шляхом пошуку в пакетах певних правил або сигнатур, які знаходяться в базі даних. Ці правила включають опис самого правила, сигнатури, опис потенційної загрози та реакції, яка має бути вжита при виявленні такої загрози. Шляхом порівняння перехоплених даних з правилами детектор визначає наявність можливої атаки або вразливості в мережі [21].

Після завершення аналізу даних, Snort генерує відповідну інформацію, таку як журнали (Logs) або попередження (Alerts), і форматує її в потрібний

вигляд. Це дозволяє зберегти необхідну інформацію для подальшого аналізу або сповіщення про виявлені загрози.

Snort здатна працювати в трьох режимах:

1. Режим сніфера: в цьому режимі Snort просто перехоплює дані з мережі і виводить їх на екран, не проводячи додаткового аналізу.
2. Режим реєстратора пакетів: у цьому режимі Snort перехоплює дані з мережі і реєструє їх у відповідних файлах. Цей режим дозволяє зберегти дані для подальшого аналізу і зберігання.
3. Режим системи виявлення вторгнень: у цьому режимі Snort перехоплює дані з мережі і проводить їх аналіз з використанням правил і сигнатур. Якщо в системі виявляється потенційна загроза, автоматично включається реєстрація пакетів. Проте, користувач може налаштувати систему, щоб змінити цю поведінку.

Barnyard2 є інтерпретатором з відкритим вихідним кодом для двійкових файлів, створюваних Snort.

Стандартний метод запису подій в Snort, який включає виведення на консоль або запис до файлу, вимагає значних обчислювальних ресурсів. У випадку, коли потрібно забезпечити більш ефективне зберігання подій, найкращим варіантом є використання бази даних MySQL. Запис подій до бази даних MySQL дозволяє зберігати дані у структурованому форматі, забезпечуючи швидкий доступ та зручне керування. Використання MySQL забезпечує надійне зберігання подій з використанням мінімуму ресурсів.

PulledPork є скриптом, призначеним для зручного завантаження, комбінування, встановлення та оновлення правил для Snort з різних джерел. Цей скрипт може працювати з різними наборами правил, які можна завантажити з різних джерел. Один з наборів правил, який PulledPork може завантажити, є безкоштовний набір правил спільноти Snort. Для завантаження цього набору правил не потрібно створювати обліковий запис на Snort.org, що є зручним варіантом налаштування. PulledPork дозволяє автоматизувати процес завантаження та оновлення правил для Snort, спрощуючи керування правилами без необхідності вручну завантажувати і встановлювати кожне правило окремо.

BASE (Basic Analysis and Security Engine) - це базовий двигун аналізу і безпеки. Це програмне забезпечення призначене для візуалізації виявлених атак і подій, що допомагає аналізувати та відстежувати безпекові події. BASE надає інтерфейс, який дозволяє користувачам переглядати, аналізувати та відстежувати виявлені атаки і події, забезпечуючи візуальне представлення цих даних. Він допомагає зрозуміти характер атак, ідентифікувати загрози та приймати відповідні заходи щодо забезпечення безпеки мережі. BASE може використовуватися як доповнення до системи IDS (інтрузійна система виявлення) або IPS (інтрузійна система запобігання), допомагаючи адміністраторам мережі відслідковувати та аналізувати відхилення від норми та вживати відповідних заходів для забезпечення безпеки.

Suricata, подібно до Snort, складається з кількох модулів, таких як захоплення, збір, декодування, виявлення і виведення. За замовчуванням, у Suricata трафік декодується одним потоком після його захоплення. Це підходить для ефективного виявлення атак, але може створювати більше навантаження на систему. Однак, у відмінність від Snort, Suricata надає гнучкі налаштування, які дозволяють перевизначати поведінку трафіку. Це означає, що за допомогою налаштувань можна розділити потоки вже після їх захоплення, а також вказати, як розподіляти потоки між процесорами системи. Це надає широкі можливості для оптимізації обробки трафіку на конкретному обладнанні в конкретній мережі. Такий підхід дозволяє адаптувати роботу Suricata до потреб конкретної мережі і забезпечує гнучкість в розподілі навантаження на процесори.

Suricata має підтримку для вилучення та перевірки переданих файлів через протокол HTTP. Він може розпізнавати та аналізувати стиснений контент, а також ідентифікувати його за допомогою різних параметрів, таких як URI, cookie, заголовки, user-agent, тіло запиту і відповіді. У деяких мережах Suricata використовується для протоколювання HTTP-трафіку без здійснення виявлення атак. Це означає, що він може виділяти контент у потоці за певними шаблонами або використовуючи регулярні вирази. Крім того, Suricata має можливість ідентифікації файлів за їх ім'ям, типом або контрольною сумою MD5. Ці

можливості Suricata дозволяють ефективно аналізувати та обробляти HTTP-трафік, а також забезпечують гнучкість у виявленні та ідентифікації файлів, що передаються через протокол HTTP.

Suricata підтримує декодування протоколу IPv6, включаючи різні типи тунелювання, такі як IPv4-in-IPv6, IPv6-in-IPv6, Teredo та інші. Завдяки модульній архітектурі движка, легко можна додавати нові компоненти для захоплення, декодування, аналізу або обробки пакетів. Для перехоплення трафіку Suricata використовує різні інтерфейси, такі як NF Queue, IPF Ring, Lib Pcap, IPFW, AF_PACKET, PF_RING. Це надає гнучкість у виборі інтерфейсу залежно від потреб мережі та наявного обладнання. Крім того, Suricata підтримує режим Unix Socket, що дозволяє автоматично аналізувати PCAP-файли, які були захоплені попередньо іншими програмами, наприклад, сніферами. Ці можливості Suricata забезпечують ефективне перехоплення і обробку трафіку, а також дозволяють працювати з різними типами мереж та отримувати дані з різних джерел.

2.5. Випробування системи виявлення та запобігання вторгненню

Критичне порівняння проводиться між системами виявлення та запобігання вторгненням *Suricata* та *Snort*.

Для вимірювання ефективності систем виявлення та запобігання вторгнень використовуються різні показники, включаючи:

1. Швидкість виявлення атак: Цей показник відображає, наскільки швидко система може виявити атаки і сповістити про них. Вимірюється у вигляді часу, який потрібен системі для виявлення нової атаки після її появи в мережі. Чим швидше система виявляє атаки, тим ефективніше вона захищає мережу.
2. Помилкові спрацьовування: Цей показник вказує на кількість помилкових спрацьовувань системи, коли вона помилково виявляє нормальний трафік як шкідливий. Високі значення помилкових спрацьовувань можуть привести до надмірного сповіщення про потенційні загрози, що вимагає додаткового часу та зусиль для аналізу та перевірки ложних сигналів.

3. Обмеження потужності: Цей показник вказує на максимальну потужність системи виявлення та запобігання вторгнень. Якщо система досягає своєї граничної потужності, вона може не мати достатніх ресурсів для обробки всього трафіку. Це може призвести до відкидання пакетів і пропуску шкідливого вмісту без виявлення. Тому важливо мати систему, яка може масштабуватися і працювати на високій потужності.

Ці показники допомагають оцінити ефективність систем виявлення та запобігання вторгнень і дозволяють визначити, наскільки ефективно система захищає мережу від потенційних загроз.

Для кількісної оцінки метрик, що використовуються для оцінки точності системи виявлення та запобігання вторгнень, можна використовувати наступні показники:

1. Охоплення: Це показник, який вказує на кількість атак, які можуть бути виявлені системою. Вимірюється шляхом підрахунку кількості успішно виявлених атак відносно загальної кількості атак, які присутні в мережі. Чим вище значення охоплення, тим ефективніше система виявляє потенційні загрози.

2. Ймовірність помилкових спрацьовувань: Цей показник вказує на ймовірність того, що система помилково виявить нормальний трафік як шкідливий. Високі значення ймовірності помилкових спрацьовувань можуть призвести до надмірного сповіщення про потенційні загрози, що потребує додаткового часу та зусиль для аналізу та перевірки ложних сигналів.

3. Ймовірність виявлення резистивних атак: Цей показник вказує на ймовірність виявлення складних атак, які можуть уникати типових методів виявлення. Вимірюється відношенням кількості успішно виявлених резистивних атак до загальної кількості таких атак, які присутні в мережі.

4. Здатність обслуговувати канал з високою пропускною здатністю і ємністю: Цей показник вказує на здатність системи обробляти великий обсяг трафіку з високою швидкістю і без втрати пакетів. Вимірюється шляхом тестування системи на максимальному навантаженні і визначення межі її оброблювальної здатності.

Щодо продуктивності, вона включає різні компоненти і не може бути представлена як окрема метрика. У таблиці 2.3 наведено деякі показники, які відображають ємність системи для обробки трафіку.

Для реєстрації показників ефективності систем виявлення та запобігання вторгнень, рекомендується враховувати наступні метрики:

1. Байти в секунду: Цей показник вимірює обсяг даних, що передаються через мережевий канал за одну секунду. Він дає уявлення про пропускну здатність системи та її здатність ефективно обробляти великі обсяги даних.

2. Пакети в секунду: Ця метрика вказує на кількість пакетів, які обробляються системою за одну секунду. Вона відображає швидкість обробки пакетів і може бути важливою при розгляді потужності системи.

3. Кількість мережевих атак: Цей показник відображає загальну кількість виявлених мережевих атак системою. Він може слугувати показником ефективності системи в розпізнаванні та блокуванні потенційних загроз.

4. Кількість втрачених пакетів: Ця метрика вказує на кількість пакетів, які були втрачені або відкинуті системою. Зменшення цього показника є показником покращення ефективності системи, оскільки підтверджує здатність системи ефективно обробляти всі пакети.

5. Тригери, помилкові спрацьовування, негативні тригери та загальна кількість тривог: Ці показники відображають результати аналізу системою, включаючи виявлення правильних тригерів, помилкових спрацьовувань, негативних тригерів (коли система не виявила потенційну загрозу) та загальну кількість зареєстрованих тривог.

6. Використання центрального процесора та пам'яті: Ці метрики відображають навантаження на центральний процесор та використання пам'яті системою. Вони можуть свідчити про ресурсоємність системи та допомогти виявити можливі обмеження продуктивності.

7. Пропускна здатність інтерфейсу та статистика файлів підкачки: Ці показники вказують на швидкість передачі даних через мережевий інтерфейс та статистику використання файлів підкачки, які можуть вплинути на продуктивність системи.

Реєстрація цих показників дозволяє здійснювати моніторинг та оцінку ефективності систем виявлення та запобігання вторгнень.

Таблиця 2.3

Оцінка потенціалу

Показник, що перевіряється	Використання ресурсів
Пакетів в секунду	Цикли *CPU*, пропускна здатність інтерфейсів, пропускна здатність шини
Байт в секунду (середній розмір пакета)	Цикли *CPU*, пропускна здатність інтерфейсів, пропускна здатність шини
Протоколи	Цикли *CPU* і пропускна здатність шини
Кількість унікальних хостів	Розмір пам'яті, цикли *CPU*, пропускна здатність шини
Кількість нових з'єднань в секунду	Цикли *CPU* і пропускна здатність шини
Кількість одночасних з'єднань	Розмір пам'яті, цикли *CPU*, пропускна здатність шини
Попередження в секунду	Розмір пам'яті, цикли *CPU*, пропускна здатність шини

Для забезпечення гнучкості та безпеки експерименту було налаштовано тестове середовище у віртуальній обстановці. Це дозволило здійснювати часте повторення та реконфігурацію експериментальних випробувань. Використання віртуального середовища забезпечує переносимість та зручність проведення експериментів без необхідності фізичної наявності окремого апаратного забезпечення. Такий підхід дозволяє зосередитися на важливих аспектах експерименту та здійснювати швидкі зміни в конфігурації з мінімальними затратами часу та ресурсів.

Для віртуалізації було використано платформу VMware Workstation 15, оскільки вона має високу продуктивність вводу-виводу та жорсткий диск порівняно з іншими засобами віртуалізації. В якості операційної системи було обрано 64-розрядну версію Ubuntu 22.04 LTS. Ubuntu є популярною операційною системою Linux зі значною підтримкою спільноти, і вона регулярно оновлюється. Такий вибір забезпечує доступ до широкого спектру функцій та ресурсів, які підтримуються спільнотою Ubuntu.

Апаратна конфігурація для системи виявлення та запобігання вторгнень в мережу включала чотириядерний процесор Intel Xeon (E5462) з тактовою частотою 2,8 ГГц і 4-ядерною пам'яттю DDR2 800 МГц об'ємом 3 ГБ, яка була повністю буферизованою. Кожна система також мала жорсткий диск максимальним об'ємом 20 ГБ. Мережевий трафік передавався окремо для кожної системи. Для відтворення мережевого трафіку використовувалась окрема система, яка працювала на одному ядрі процесора і мала 1 ГБ оперативної пам'яті. Операційна система хосту VMware також використовувала 2 ГБ оперативної пам'яті і одне ядро процесора, що забезпечувало ізоляцію хосту від тестового стенду.

Snort і Suricata були налаштовані для використання однакових правил. Однак, Suricata використовує різні конфігураційні класифікатори в порівнянні з Snort. Snort використовує 134 декодери та 174 правила препроцесора, тоді як Suricata має свою власну класифікацію. Для реєстрації подій в обидвох системах використовувалися однакові методи, такі як Barnyard, MySQL і AcidBase, як для системи виявлення вторгнень, так і для системи запобігання. Версії Snort і Suricata були відповідно v2.9.8.3 і v4.1.2. Обидві системи використовували набір правил VRT Snort v2.9.8.3, доповнений набором правил для нових загроз. Після завантаження всіх правил, Suricata мав 11039 визначень правил, порівняно з 11065 в Snort. Ця розбіжність пов'язана з тим, що Suricata не може аналізувати деякі правила VRT.

При виборі мережевого трафіку для тестування систем виявлення та запобігання вторгнення в мережу важливо враховувати деякі особливості. По-перше, атакуючий трафік може використовуватись самостійно або в поєднанні з контекстним фоновим трафіком. Фоновий трафік може бути реальною або симульованою активністю. У разі реальної активності фонового трафіку можна залишити недоторканим або, навпаки, здійснити дезінфікуючі заходи, що означає видалення конфіденційних користувацьких даних та інформації про IP-адреси.

У процесі тестування було корисно та бажано використовувати реальний мережевий трафік у фоновому режимі. Однак повторення експериментів з

таким трафіком в реальному часі було непередбачуваним через його динамічність. Тому було прийнято рішення використовувати захоплений з файлу pcap трафік. Це дозволило обробляти трафік системою виявлення та запобігання вторгненню в автономному режимі, а також відтворювати його в мережі з різною швидкістю за допомогою інструменту TCPReplay. Такий підхід дозволив уникнути будь-яких ризиків для критично важливих мереж.

Існує багато джерел тестового трафіку, які можуть бути використані для навантаження систем виявлення та запобігання вторгненню в мережу (NIDPS). Однак, багато з цих джерел трафіку піддаються дезінфікації, що робить їх непридатними для оцінки NIDPS, які виконують глибокий аналіз пакетів. Існують інструменти, наприклад, TCPdump Randomiser, які можуть додавати корисне навантаження до очищених даних, але реалістичність таких змінених даних може бути сумнівною.

Таблиця 2.4

Вивчення атак

Код	Ім’я	Опис
1	2	3
ms03_026_dcom	Microsoft RPCDCOM Interface Overflow	Модуль використовуваного стеку переповнення буфера в службі RPCSS
ms05_039_pnp	MicrosoftServer Service NetpwPathCan onicalize Overflow	Стек переповнення буфера в службі Windows Plug and Play
ms05_047_pnp	Microsoft Plugand Play Service Registry Overflow	Стек переповнення буфера в службі Windows PnP. Причина перезавантажень.
ms06_040_netapi	MicrosoftServer Service NetpwPathCan onicalize Overflow	Стек переповнення буфера в NetApi32 CanonicalizePathName () використовуючи функцію NetpwPathCanonicalize RPC виклик служби Server
ms05_017_msmq	MicrosoftMessage Queueing Service Path	Використовуваний стек переповнення буфера в

	Overflow	RPC інтерфейсі в службі Microsoft Messgae Queueing
ms01_033_idq	Microsoft IIS5.0 IDQ Path Overflow	Використовуваний стек переповнення буфера в IDQ ISAPI обслуговування для Microsoft Index Server

Був зафіксований трафік, який використовувався для проведення атак Metasploit на комп'ютері з операційною системою Microsoft Windows 2000. Windows 2000 було вибрано як операційна система, найбільш підходяща для використання з Metasploit порівняно з іншими варіантами. Чим більше служб і додатків було встановлено та припинено їх функціонування, тим більше атак було можливо здійснити. На жаль, не всі атаки були доступні. Атаки, перелічені в таблиці 2.4, були зареєстровані за допомогою програми Wireshark. Захоплений трафік включав як фоновий, так і атакуючий типи трафіку. Для вирівнювання часових міток використовуваного трафіку з фоновим трафіком була використана частина програми Wireshark, відома як Edicap. Це дозволило встановити хронологічний порядок трафіку, зміщуючи атакуючий трафік на другий план.

Ефективність роботи систем виявлення та запобігання вторгнень в мережу (NIDPS) в значній мірі залежить від продуктивності центрального процесора. Тому для оцінки роботи Snort та Suricata в умовах великого навантаження важливо навантажувати центральний процесор. Це означає, що при тестуванні функціональності та продуктивності Snort та Suricata доцільно створювати стресові умови, які докладно навантажують центральний процесор. Це дозволить оцінити, наскільки ефективно системи впораються з великим обсягом обробки данних та здатні виявляти потенційні загрози. Завантаження центрального процесора в ході тестування Snort та Suricata у стресових умовах допоможе встановити їхню продуктивність та здатність працювати під великим навантаженням. Це є важливим етапом в оцінці ефективності та надійності систем виявлення та запобігання вторгнень в мережу.

Використання VMware дозволило зменшити кількість доступних логічних та фізичних ядер у системі. Шляхом регулювання навантаження на ці ядра створювалися контрольовані та вимірювані потоки роботи. Для досягнення цього був використаний інструмент cpulimit, який надає можливість налаштування навантаження на центральний процесор та обмеження загального використання ресурсів кожним потоком до певного відсотка потужності. Таким чином, за допомогою cpulimit було створено контрольоване середовище, в якому керувалася робота центрального процесора та розподіл ресурсів між потоками. Це дозволило ефективно управляти навантаженням на процесор та встановлювати обмеження щодо використання ресурсів для кожного потоку. Такий підхід дозволяє забезпечити більш точне контролювання та вимірювання роботи системи в умовах обмеженого ресурсу центрального процесора.

Як Snort, так і Suricata підтримують внутрішнє відтворення файлів pcap. Цей процес здійснюється з максимальною можливою швидкістю, що дозволяє провести ефективну оцінку продуктивності системи NIDPS. Однак при використанні цього методу не враховуються максимальні швидкості без втрат (MLFR). З метою перевірки швидкості передачі трафіку без втрат використовувався інструмент TCPReplay. Це дозволило провести стрес-тести під навантаженням на мережу та перевірити, наскільки система може справлятися з високим обсягом трафіку без втрати даних. Таким чином, комбінація внутрішнього відтворення файлів pcap в Snort та Suricata та використання TCPReplay надає можливість оцінити продуктивність системи NIDPS як з максимальною швидкістю, так і з урахуванням максимальних швидкостей без втрат.

Було проведено експеримент для вивчення впливу великої пропускної здатності та високої напруги процесора на NIDPS з підходом MLFR. Цей експеримент мав на меті зібрати дані, як кожна система може впоратися зі збільшеною пропускною здатністю при високому навантаженні процесора.

Атакуючий трафік був направлений через обидві системи NIDPS з різними конфігураціями процесора для проведення експерименту.

Використовувалися наступні конфігурації: 2-ядерна конфігурація процесора, 1-ядерна конфігурація з навантаженням 50% та 75%. Під час експерименту було оцінено здатність NIDPS до обробки пакетів і точність виявлення, з особливим акцентом на помилково негативних результатах. Тестовий трафік було відтворено у середовищі з використанням TCPReplay з масштабуванням у 40 разів. Це означає, що трафік відтворювався у 40 разів швидше, ніж в реальному часі при захопленні. Цей підхід дозволив отримати швидкість передачі даних на рівні 3,1 Мбіт/с і виявити лише 2% скинутих пакетів. Така швидкість забезпечила вчасне завершення експериментів перед втратою пакетів.

Під час кожного тестування були реєстровані початок і кінець трафіку, що протікає через систему NIDPS. Це становило важливу вихідну точку для подальшого аналізу ефективності системи попередження та збору статистики. Інформація про попередження була реєстрована для кожного тестового запуску та зберігалася у вихідному форматі unified2, також відомому як acidbase, для зручності подальшого використання. В статистиці продуктивності NIDPS були враховані такі показники, як кількість згенерованих попереджень, кількість оброблених пакетів і співвідношення цих показників до різних мережевих протоколів. Весь трафік проходив через хости з IP-адресами 192.168.16.2 та 192.168.16.128, але був відзначений як небажаний трафік, який потребує уваги та аналізу з боку системи NIDPS.

Для оцінки точності був використаний контроль попереджень, який отримувався без використання стресових тестів і слугував базовим еталоном. Зміни точності виявлення відображалися відхиленням від цього базового рівня під час стресового навантаження. У таблиці 2.5 наведено кількість різних типів попереджень, які генерувалися при атаках на кожну систему NIDPS. На рисунку 2.7 показано попередження, що генерувалися системою Suricata для кожного експлоіта в різних конфігураціях, проте було помічено втрату деяких попереджень, що призвело до скорочення діапазону виявлення.

Таблиця 2.5

Попередження згенеровані Snort і Suricata

Попередження	Snort	Suricata
ms05_040_pnp	4	4
ms05_047_pnp	1	1
ms05_039_pnp	1	6
ms03_026_dcom	1	2
ms01_033_1dq	2	4
ms05_017_msmq	2	3

На рисунку 2.8 показано невдачі в попередженнях системи Snort для атаки ms01_033_idq. Ці помилкові негативні результати виникають через надмірне навантаження системи.

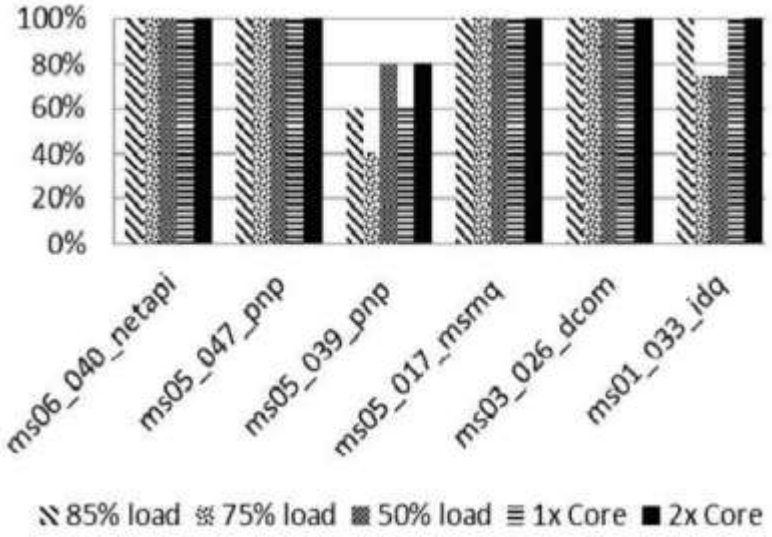

Рис. 2.7. Попередження у Suricata

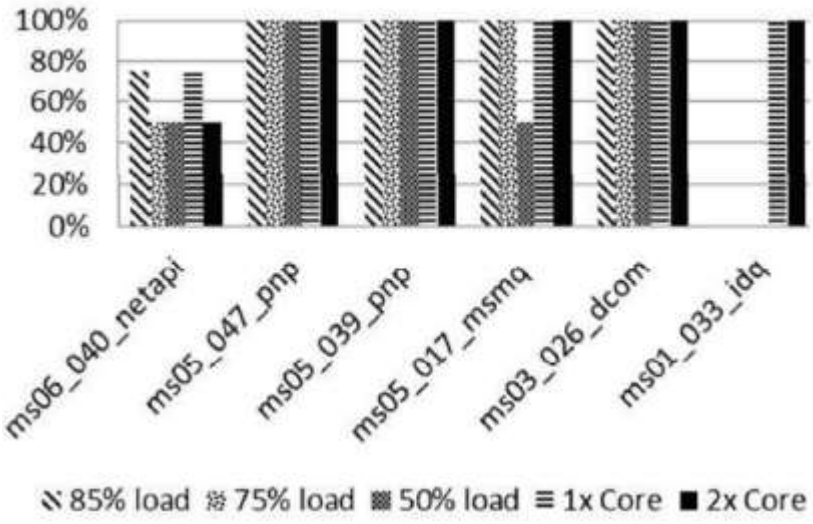

Рис.2.8. Попередження у Snort

На рисунку 2.9 представлено кількість помилкових позитивних і справжніх позитивних результатів для обох NIDPS, порівняно з кількістю втрачених попереджень для кожної системи.

Неправильно виявлені негативні результати можуть призвести до втрати пакетів. На діаграмі 2.10 показано, як залежність між кількістю відхилених пакетів Snort та Suricata змінюється залежно від доступних ресурсів центрального процесора. У випадку Snort, відсоток втрат зазвичай збільшується лінійно, тоді як продуктивність Suricata суттєво падає лише тоді, коли ресурси центрального процесора обмежуються одним ядром. На діаграмі 2.11 показано, як зменшення кількості ядер та використання центрального процесора впливає на обидві системи, а також вплив неправильно виявлених негативних результатів на їх продуктивність.

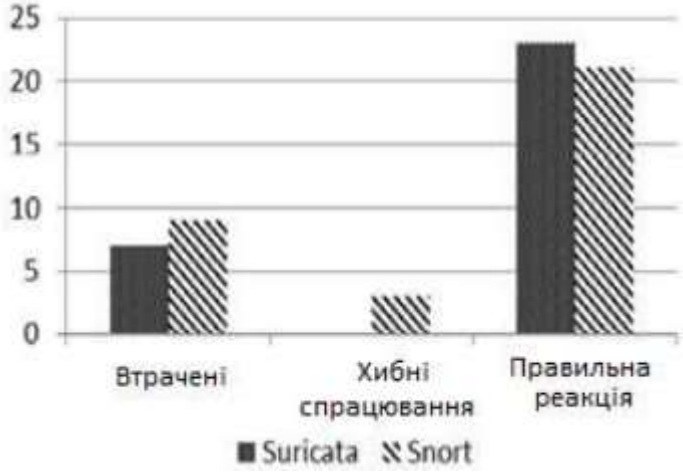

Рис. 2.9. Точність вимірювання атак

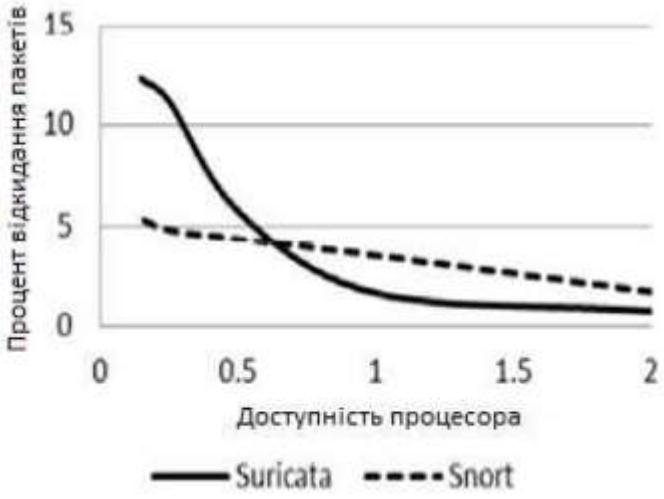

Рис. 2.10. Графік втрати пакетів в 3,2 Мб/с

На рисунку 2.12 зображено взаємозв'язок між використанням процесора та пропускною здатністю мережі для систем Suricata та Snort. Графік демонструє, як збільшення використання центрального процесора впливає на пропускну здатність мережі. Цей ефект особливо помітний у випадку Suricata. Snort також виявляє подібну тенденцію, але в значно меншій мірі.

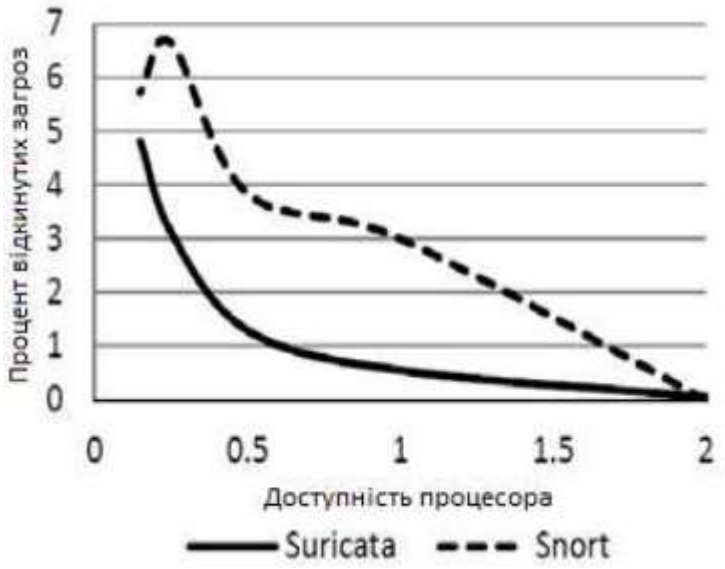

Рис. 2.11. Графік помилкових відкинутих попереджень

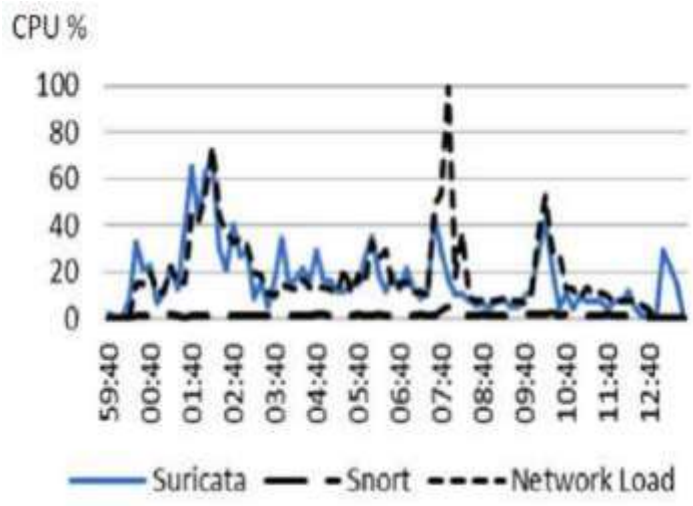

Рис. 2.12. Пропускна здатність мережі і використання CPU для одного ядра

Для отримання відповіді на питання щодо здатності обох систем використовувати двоядерні процесори, були розглянуті дані з рисунків 2.13 і

2.14. На рисунку 2.13 показано, як система Snort використовує двоядерний процесор, а на рисунку 2.14 - система Suricata.

На основі даних з рисунку 2.13 можна зробити висновок, що система Suricata рівномірно розподіляє навантаження на обидва ядра процесора, у порівнянні з системою Snort, де спостерігається менш стабільне балансування. Це співпадає з очікуваними результатами, оскільки Suricata має багатопотокову архітектуру, що дозволяє краще використовувати ресурси обох ядер процесора, ніж у випадку Snort. Детальніше про це можна побачити на рисунку 2.14.

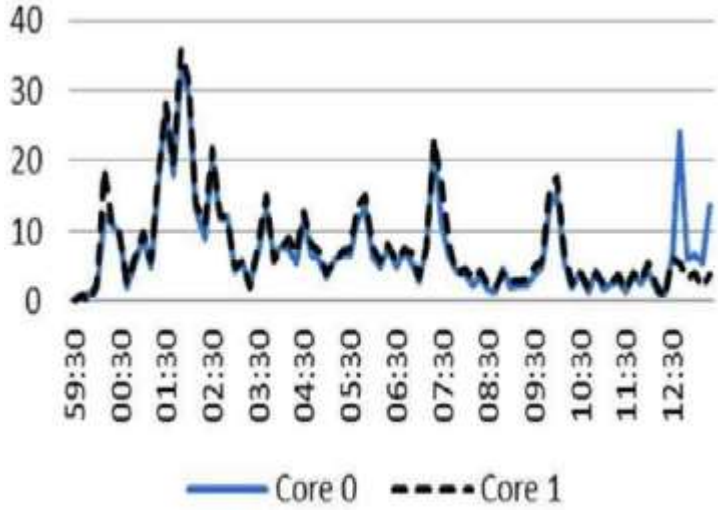

Рис. 2.13. Використання Suricata двох ядер

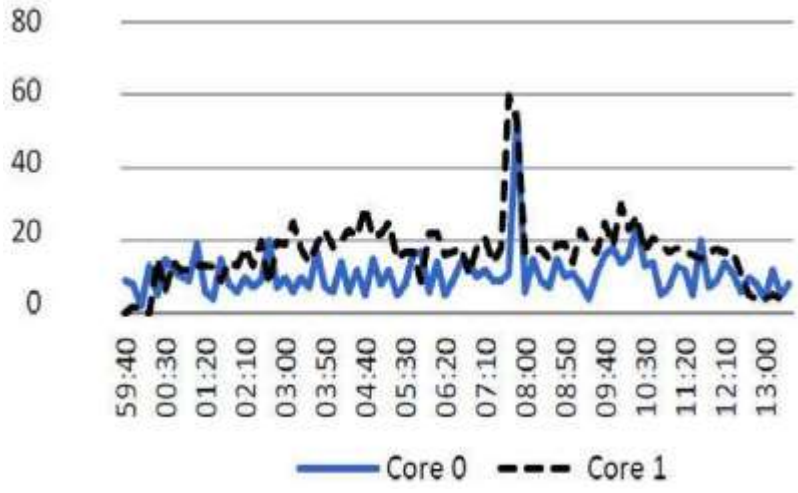

Рис. 2.14. Використання Snort двох ядер

Кожна з систем NIDPS, Snort і Suricata, має здатність автономно обробляти трафік, приймаючи файл pcap і обробляючи його на найбільшій можливій швидкості. Це було виконано з метою визначення пропускної

здатності обох систем при обробці трафіку. Проведено тестування обох NIDPS, використовуючи один і той самий файл pcap. Результати цього тестування, включаючи час, необхідний для обробки кожною системою, показані на рисунку 2.15.

Незважаючи на використання додаткових ядер, час обробки Snort не покращився, в той час як продуктивність Suricata збільшилась на 220% при використанні 4 ядер порівняно з одним. Це результат відповідає очікуванням, оскільки Suricata має багатопотокову архітектуру, що сприяє покращенню продуктивності при використанні більшої кількості ядер.

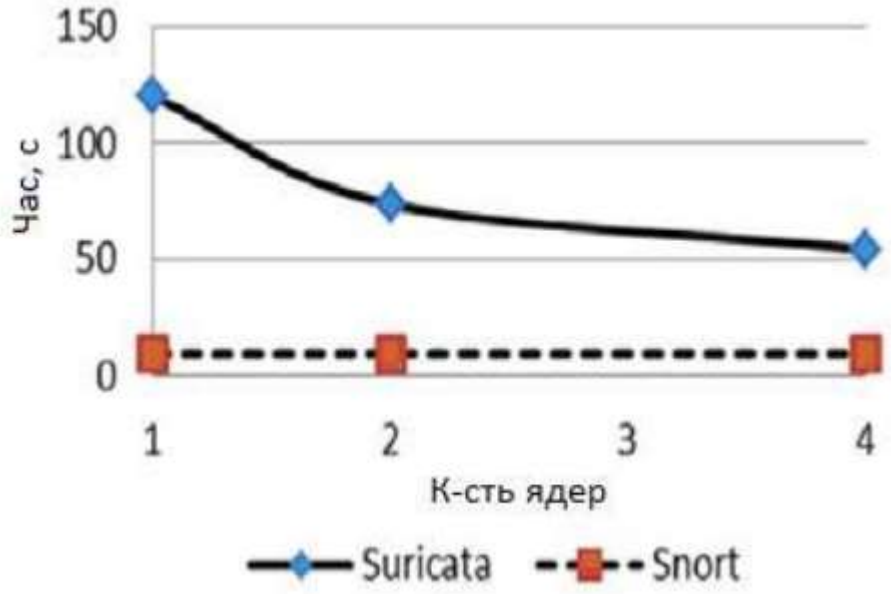

Рис. 2.15. Час обробки pcap файлу

Оцінка точності є одним з найважливіших показників для IDPS. Вона включає в себе оцінку покриття атак, помилкових позитивних та помилкових негативних результатів, потенційних загроз, а також здатність системи обробляти великий обсяг трафіку, тобто мати високу пропускну здатність.

Розробники Suricata акцентували свою увагу на підвищенні точності NIDPS і, судячи з результатів, вони досягли певного успіху. На рисунках 2.7, 2.8 і 2.9 видно, що Suricata демонструє більшу точність порівняно з Snort. Наприклад, це стає очевидним з того, що Snort не зміг виявити експлойт ms01_033_idq, коли центральний процесор був навантажений менше 50%.

Частково це пов'язано з тим, що Snort менш контролює механізм сповіщень під час атаки, в порівнянні з Suricata (два проти чотирьох). Snort не зміг виявити ms01_033_idq за допомогою двох правил з набору правил VRT, які мали ідентифікатори 1245 та 1244. З іншого боку, Suricata був успішним, і ці сповіщення були ефективними.

У багатоядерній конфігурації Suricata демонструє меншу втрату пакетів в порівнянні з Snort. Це видно на рисунках 2.13 і 2.14, які показують, що Suricata розподіляє навантаження рівномірніше між доступними ядрами. Тести в автономному режимі показують, що Suricata працює повільніше, ніж Snort. Однак Suricata показує більшу масштабованість завдяки ефективному використанню багатоядерної системи (див. рисунки 2.10, 2.11 і 2.15). Якщо Snort досягає задовільної пропускної здатності, рекомендується запускати кілька екземплярів Snort на різних ядрах. Це може забезпечити подібну масштабованість, що й Suricata, але при цьому вимагатиме додаткових зусиль для обробки однопотокових додатків на кількох ядрах.

Висновки за розділом

Було проведено аналіз методів виявлення вторгнень у комп'ютерній мережі. Дослідження показали, що найефективнішими рішеннями можуть бути статистичні методи та методи, що використовують нейронні мережі, за умови належної навчальної вибірки.

Було проведено оцінку ефективності статистичного методу у виявленні вторгнень у комп'ютерну комерційну мережу (ККМ). Результати підтвердили його високу ефективність, тому цей метод може бути використаний у системах виявлення вторгнень. Для подальшого дослідження було проведений аналіз систем виявлення вторгнень (СВВ) з різних критеріїв, і були обрані СВВ Snort та Suricata для подальшого дослідження.

Після докладного аналізу функціонування систем виявлення вторгнень (СВВ), можна зробити висновок, що Suricata має вищу точність порівняно з Snort. Це в частині обумовлено збільшеним навантаженням на центральний

процесор. Результати дослідження показали, що при більш рівномірному розподілі ядер Suricata проявляє більшу масштабованість та ефективність в разі використання декількох ядер. Однак варто зазначити, що при використанні Suricata з одним ядром точність може зменшуватися через збільшені вимоги до ресурсів.

РОЗДІЛ 3

ПРОЦЕС ВСТАНОВЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ. ІНСТРУКТАЖ ТА КРОКИ НАЛАШТУВАННЯ

Отже, коли розроблена система з двох безкоштовних IDPS систем розроблена та випробувана, є необхідність залишити пам’ятку для спеціалістів, що встановлюють захист корпоративної мережі, та модерувати цим програмним забезпеченням, а серед обов’язкових до встановлення та конфігурації є : операційна система Ubuntu (бажано останньої версії) , Wireshark, Snort та Suricata.

3.1. Завантаження операційної системи Ubuntu 22.04 LTS

Ubuntu – операційна система (ОС), що побудована на основі Debian GNU/Linux, вибір якої обґрунтовується вузької спеціалізацією, направлену на роботу з мережами та досить легка в освоєнні користувачами ПК завдяки численним форумам присвяченим цій ОС. Вона має досить простий графічний інтерфейс, що забезпечує меншу вибагливість до ресурсів ПК, а це, в свою чергу, забезпечує велику швидкодію та малу затримку в мережі.

Для того, щоб поставити на ПК цю операційну систему необхідно виконати декілька кроків :

1. Зайти на офіційний сайт ubuntu.com, обрати вкладку Download та вибрати “Get Ubuntu Desktop”, як проілюстровано на рисунку 3.1.

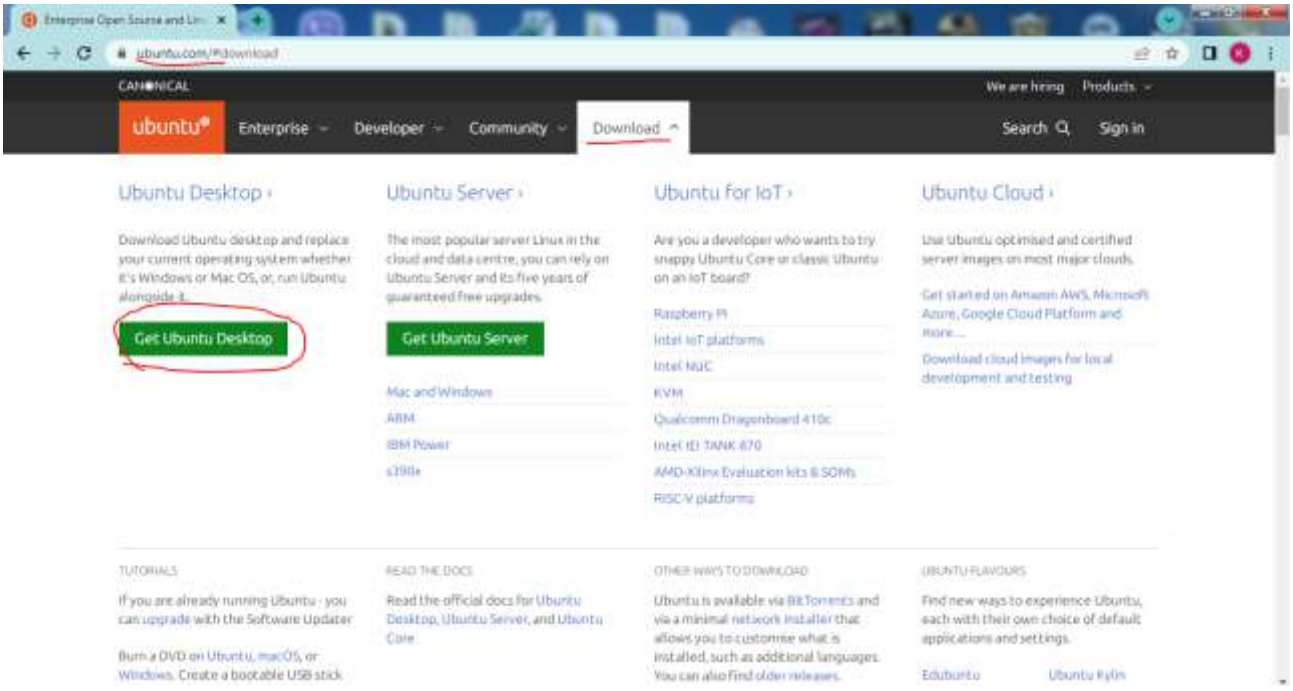

Рис. 3.1. Сайт для завантаження дистрибутива Ubuntu

Буде представлений список дистрибутивів різної версії, обираємо версію Ubuntu 22.04, продемонстрована на рисунку 3.2 (остання версія на момент проєктування дипломної роботи).

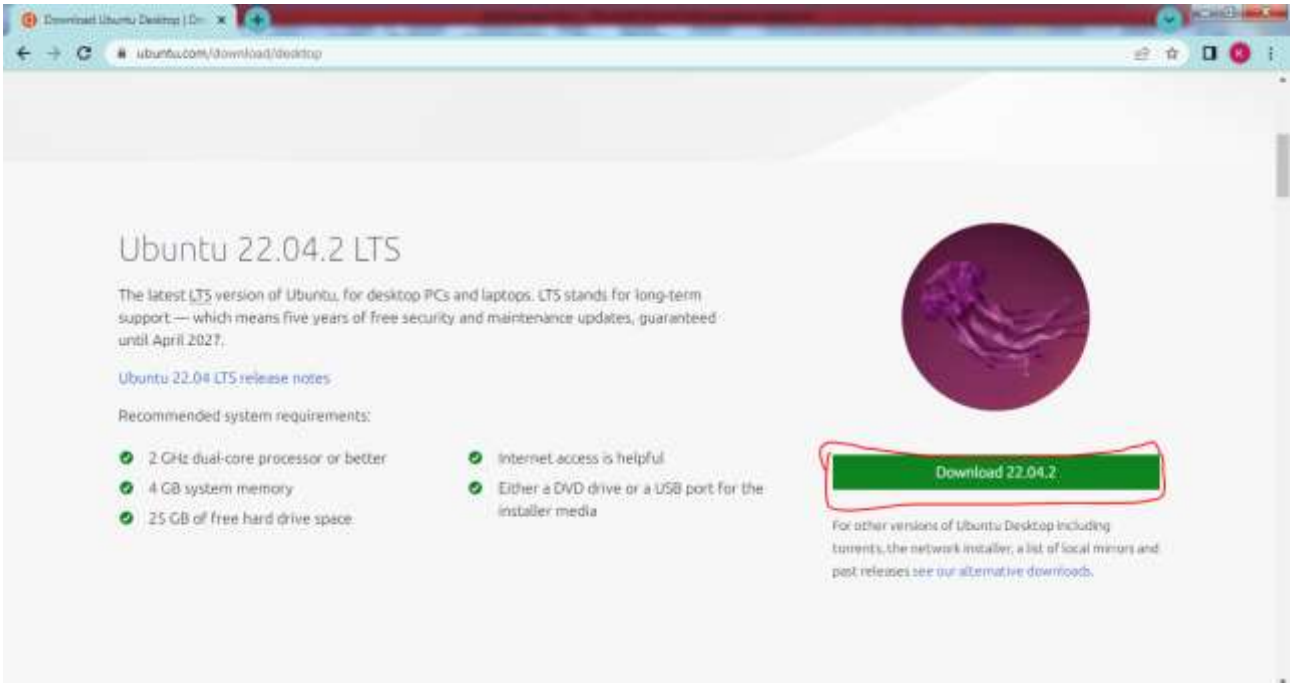

Рис. 3.2. Завантаження версії Ubuntu

2. Після завантаження дистрибутива на ваш ПК необхідно зробити завантажувальну USB-флеш-накопичувач з програмою завантаження Ubuntu. Для цього необхідно мати пустий флеш-накопичувач та програму утиліту, яка дозволяє зробити завантажувальний флеш-накопичувач, наприклад, SuperISO, який ілюстрований на рисунку 3.3.

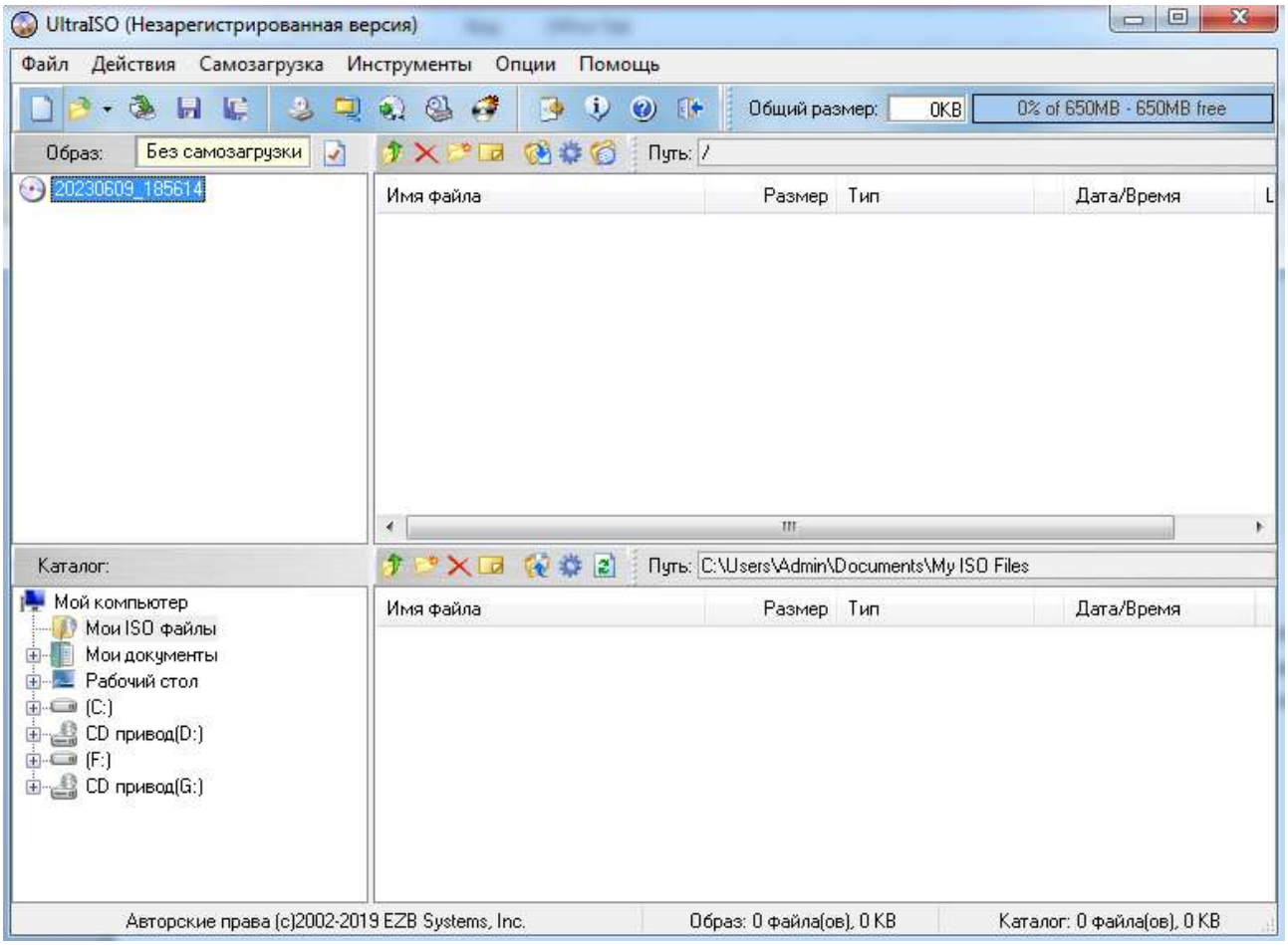

Рис. 3.3. Головне вікно програми-утиліти SuperISO

3.Після створення завантажувального флеш-накопичувача необхідно вставити у USB-гніздо та ввімкнути ПК, на який збираєтесь встановити цю ОС та з моменту появи завантаження системи увійти у BIOS, щоб поставити в налаштуваннях черги запуску накопичувачів свій завантажувальний флеш-накопичувач на перше місце, після чого , зберігаєте налаштування та перезавантажуєте ПК (рисунок 3.4).

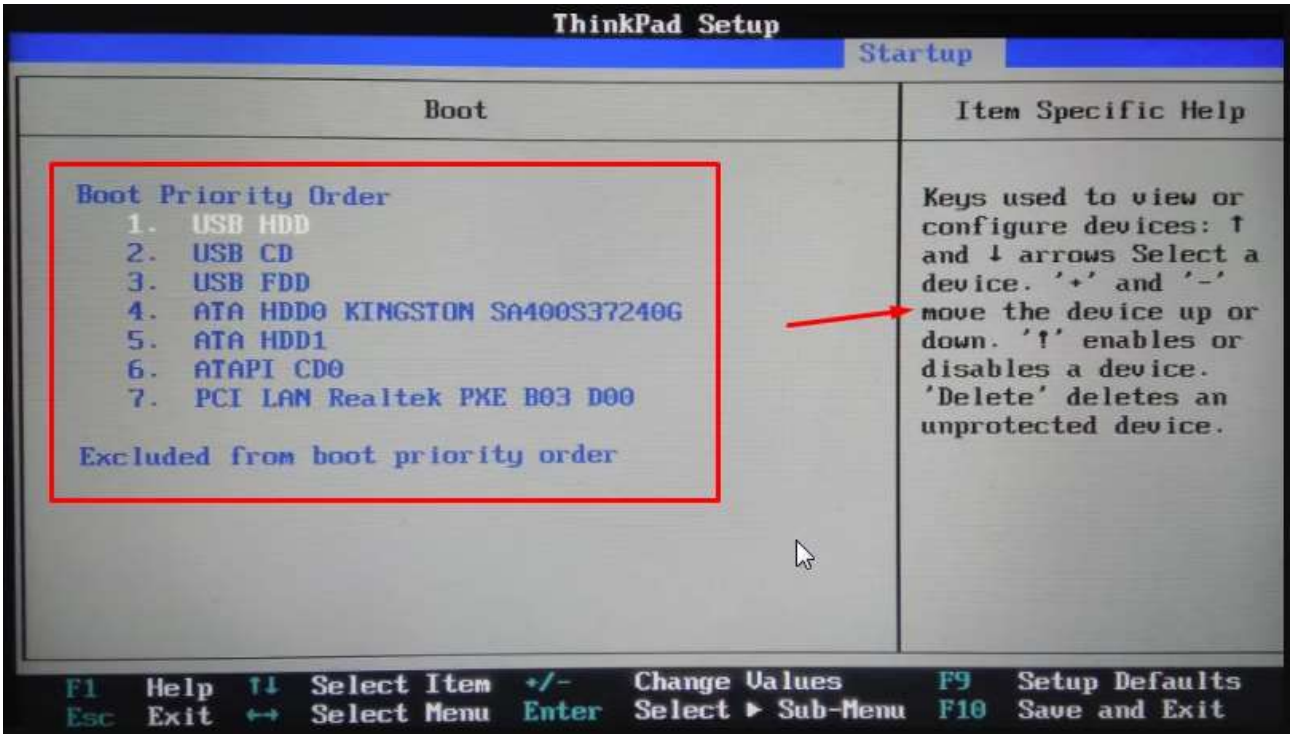

Рис. 3.4. Налаштування черги запуску накопичувачів.

4. Якщо Ви правильно зробили попередні кроки, то при перезавантаженні ПК з’явиться таке вікно (рисунок 3.5).

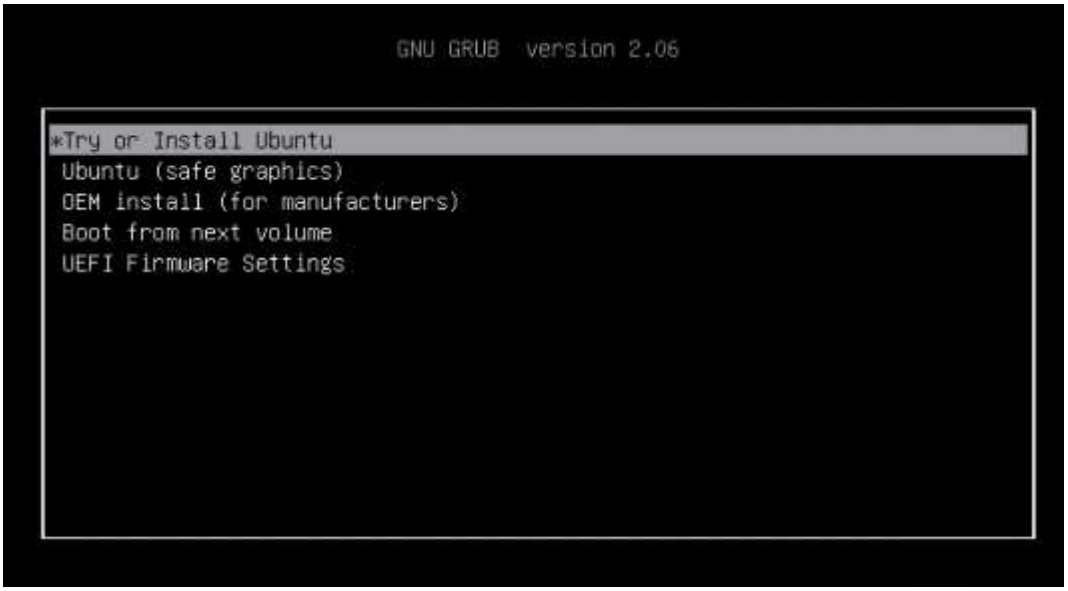

Рис. 3.5. Меню GNU GRUB

Натискаєте клавішу “Enter” і далі йде налаштування цієї ОС.

Кроки налаштування самої Ubuntu будуть проілюстровані в наступних рисунках.

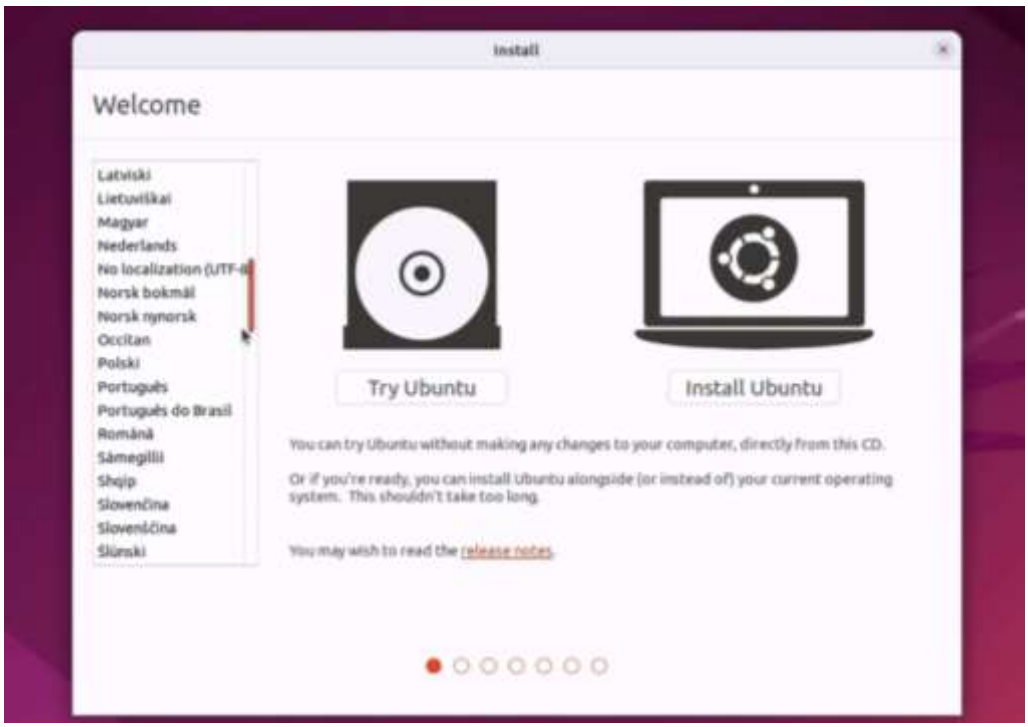

Рис. 3.6. Вибір мови та вибір чи встановлювати ОС на ПК чи працювати з флеш-накопичувача

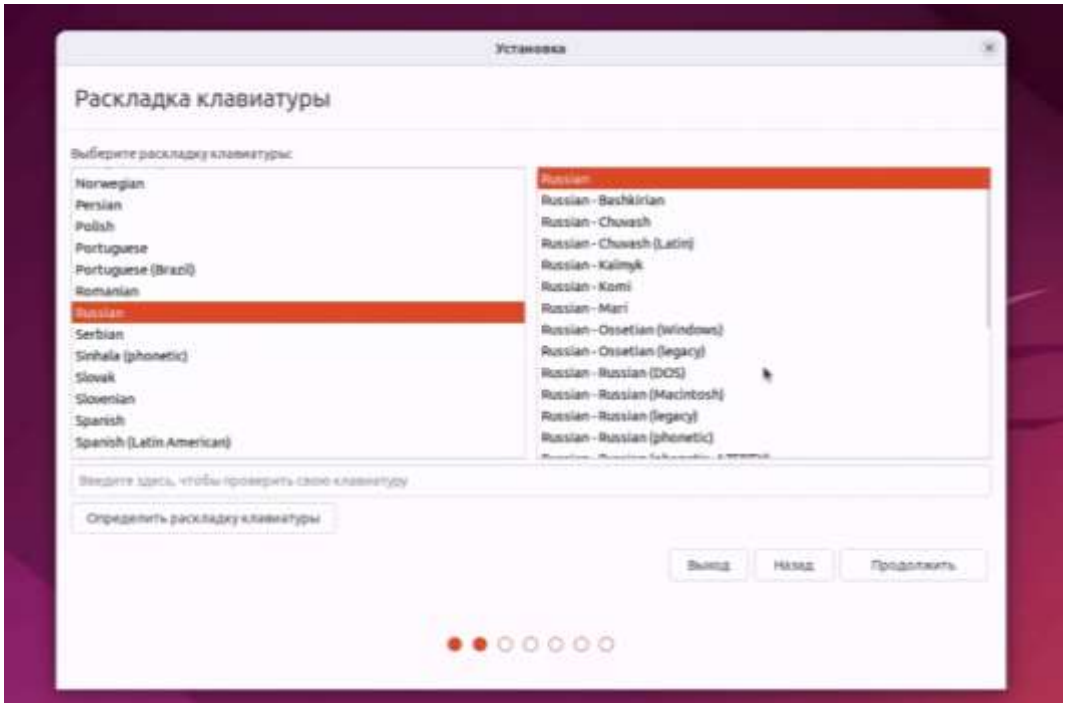

Рис. 3.7. Вибір мови інтерфейсу та розкладки клавіатури

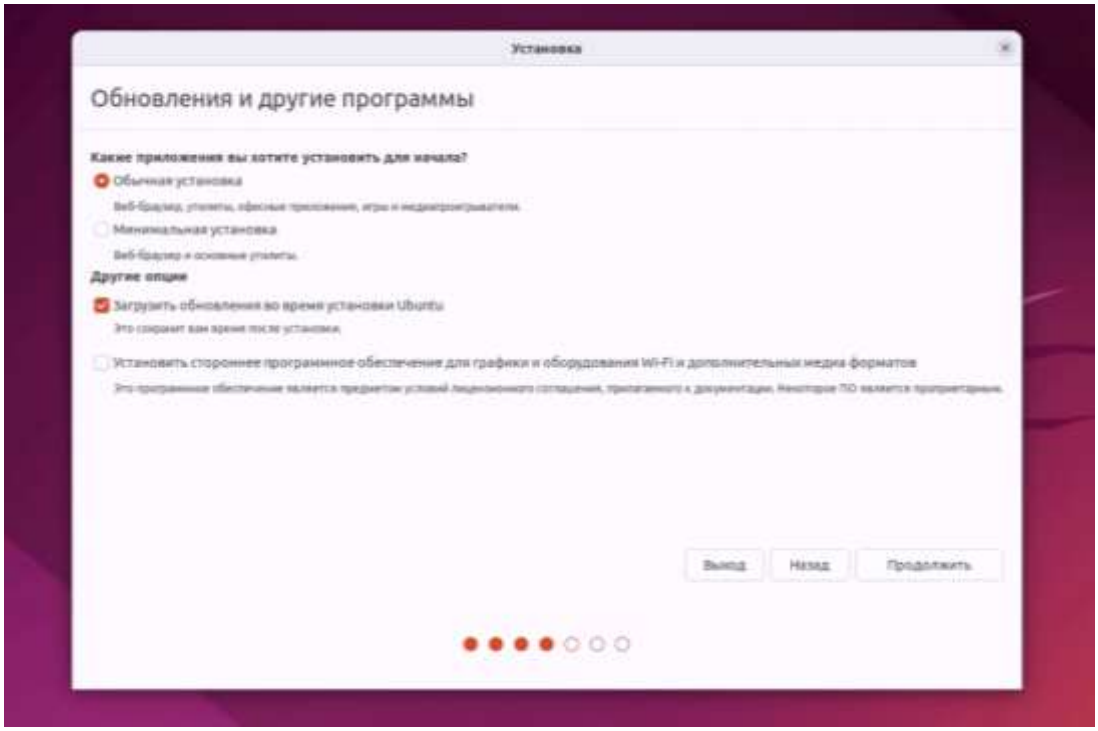

Рис. 3.8. Вибір типу завантаження додаткових програм у ОС

Рекомендується залишити прапорці на своїх місцях як на рисунку 3.8, так як потрібні програми можна встановити вручну потім.

Далі вам запропонують обрати диск, на якому буде розташовуватись ОС, якщо використовуваний ПК буде слугувати у якості “щита” для корпоративної мережі, то можна стерти існуючу операційну систему і замінити тією, що встановлюємо(рисунок 3.9).

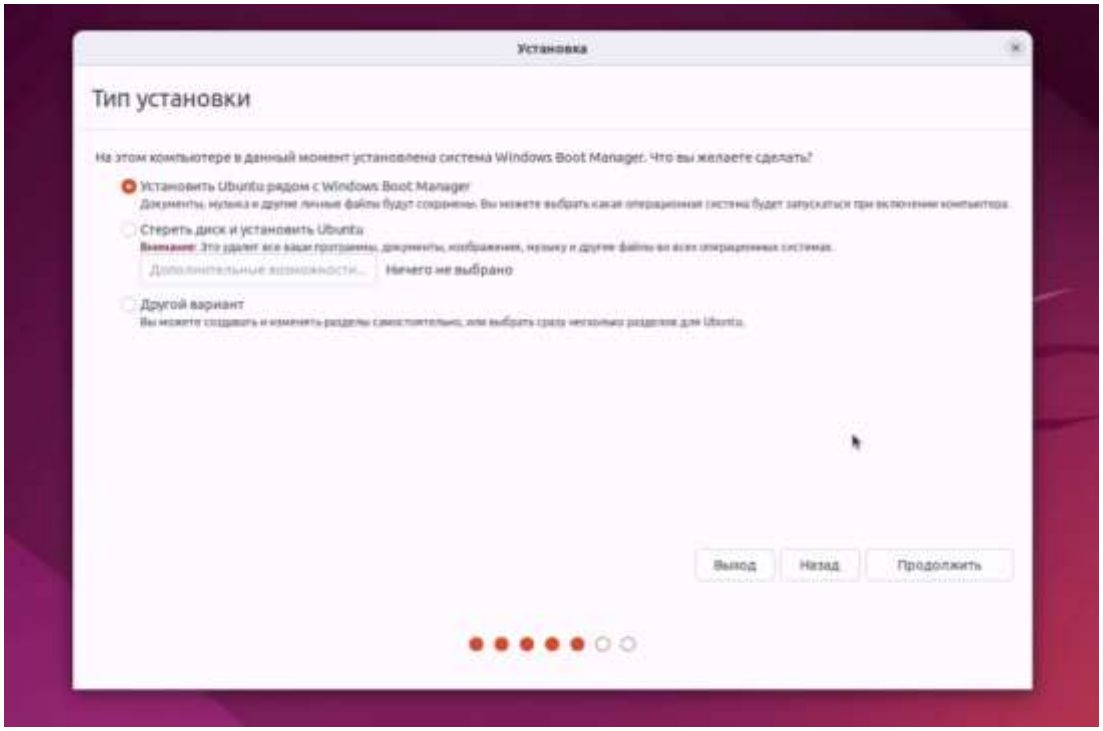

Рис. 3.9. Вибір розташування ОС

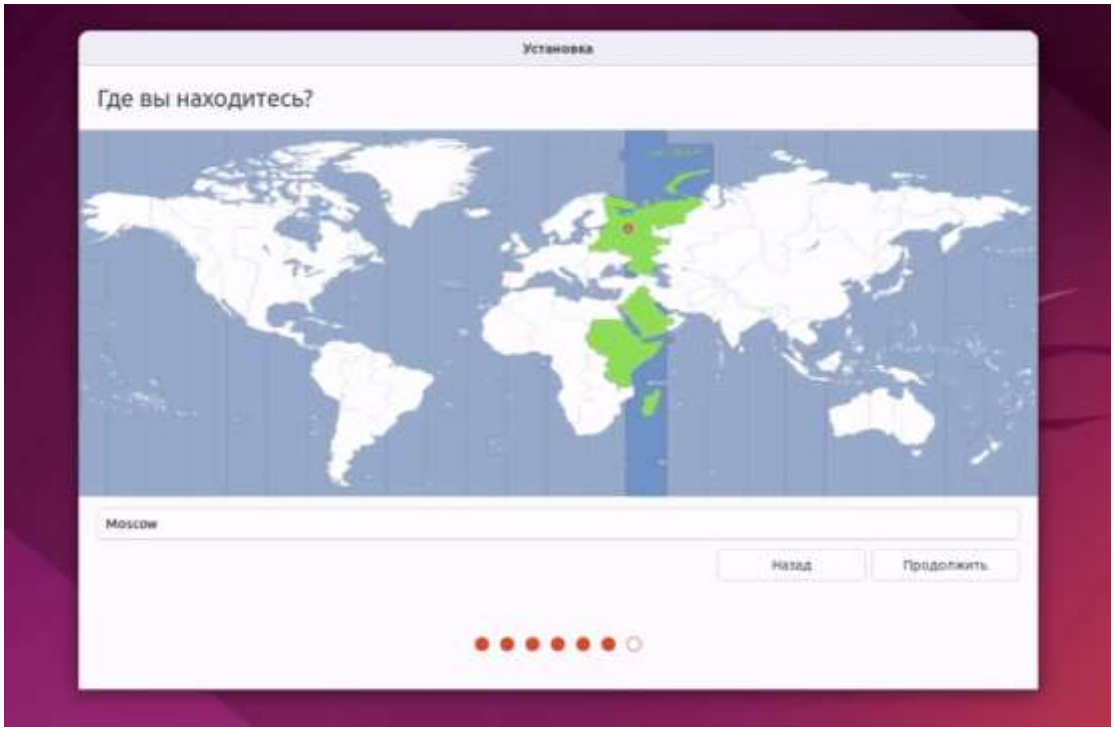

Рис. 3.10. Вибір часового поясу

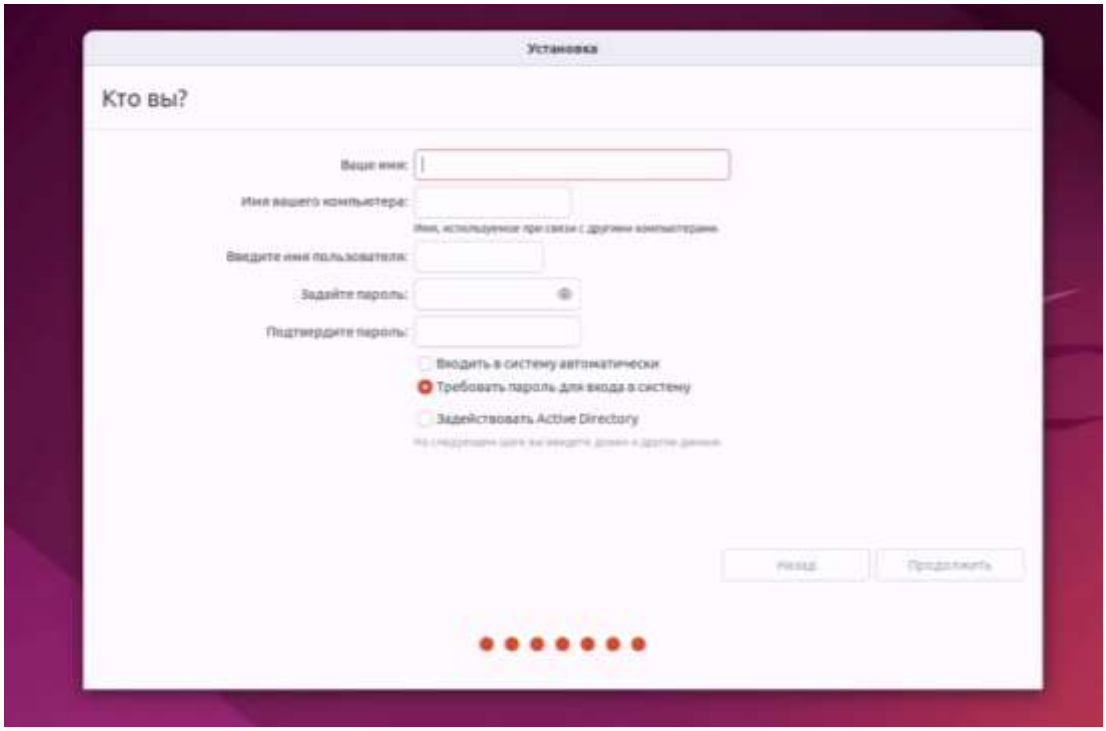

Рис. 3.11. Створення нового користувача

Рис. 3.12. Завантаження ОС

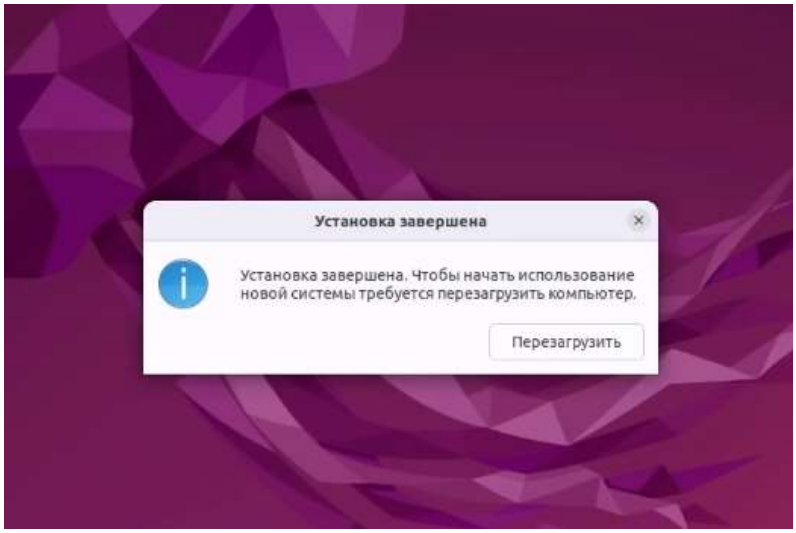

Рис. 3.13. Вікно пропозиції перезавантажити ПК

Отже, операційна система завантажена, а значить завантажувальний флеш-накопичувач можна виймати та відновити порядок завантаження накопичувачів у BIOS.

Якщо у вас відкрився Робочий стіл як на рисунку 3.14, то вітаю, Ви правильно встановили Ubuntu.

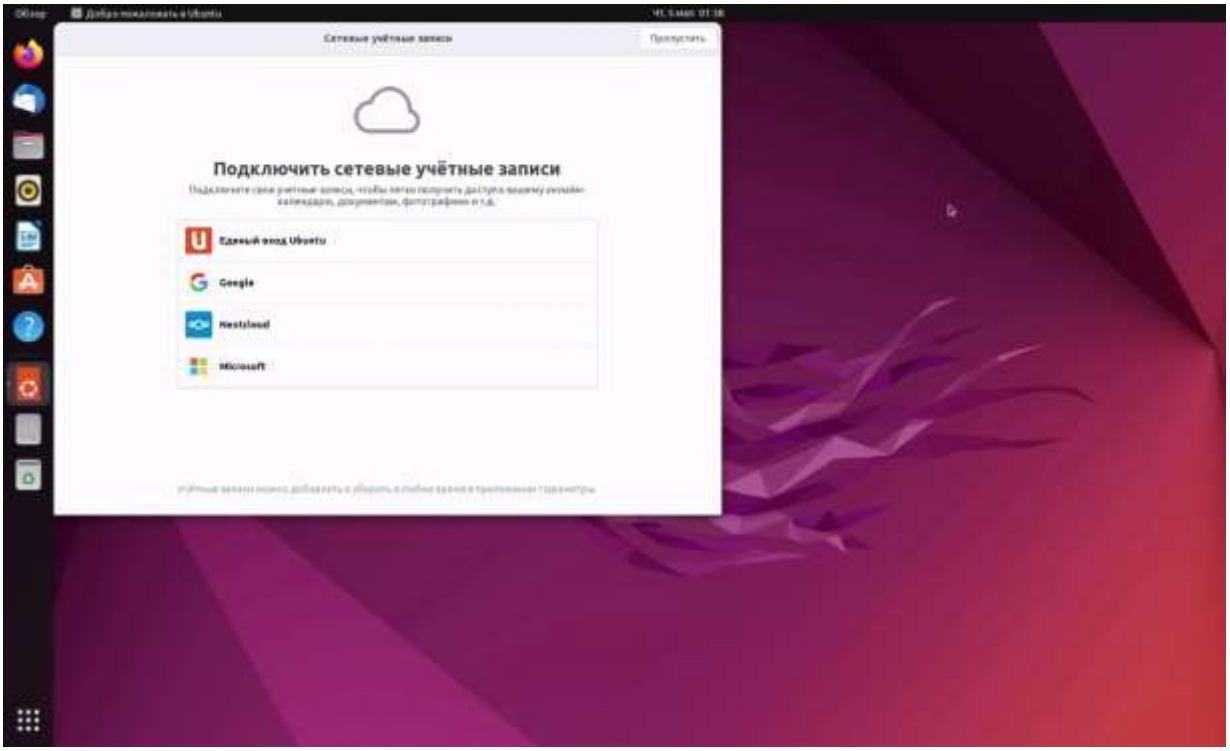

Рис. 3.14. Робочий стіл

3.2. Встановлення IPS-програми Snort

Для встановлення Snort на наш ПК для початку необхідно відкрити термінал як на рисунку 3.15 або його аналог.

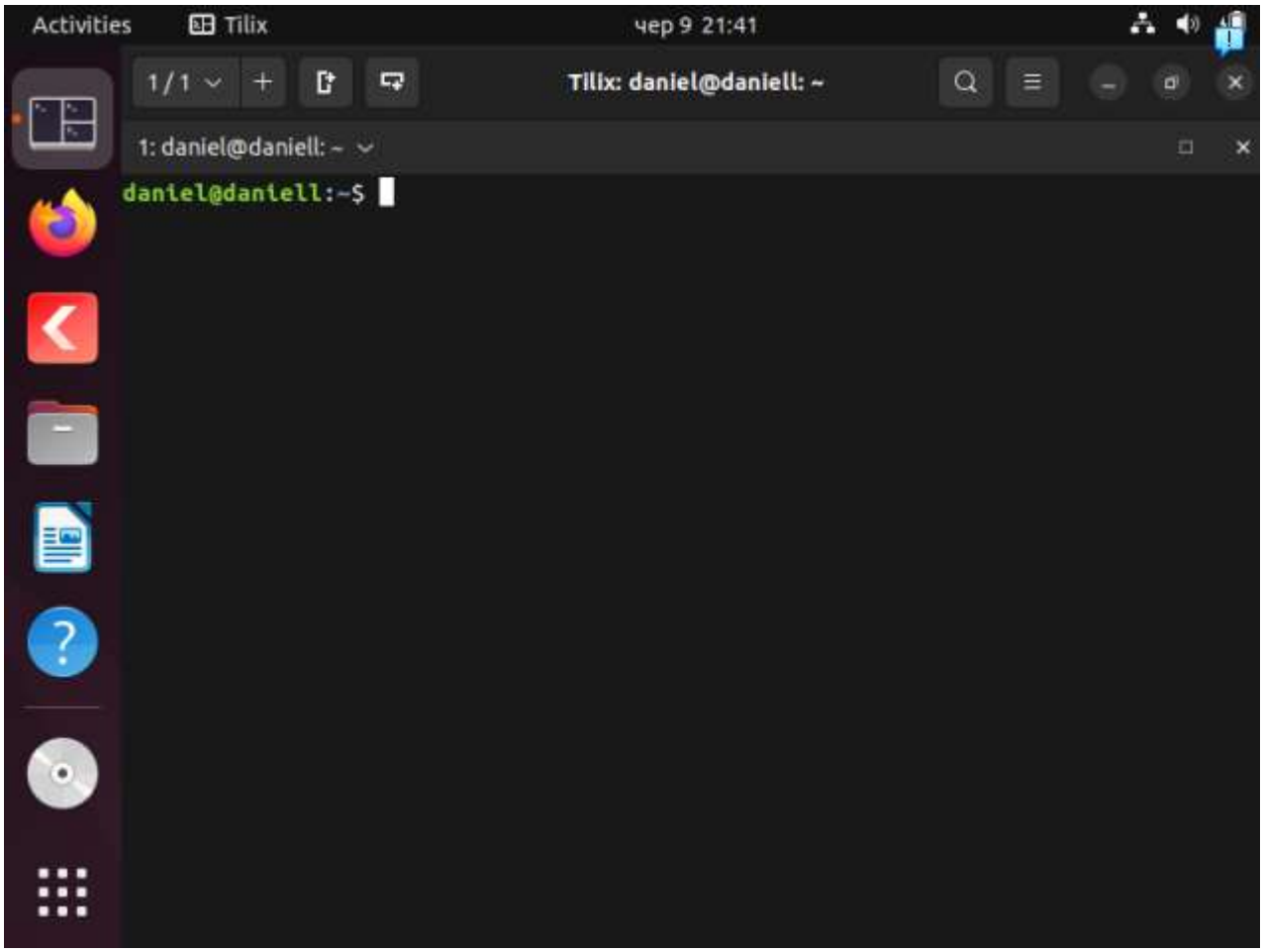

Рис. 3.15. Вікно терміналу

Далі необхідно ввести команду:

sudo apt install snort

Після введення команди та її підтвердження клавішею “Enter” термінал нас запитує пароль від нашого облікового запису, який необхідно ввести та підтвердити, і після цього почнеться завантаження, попередньо запитавши Вас, чи необхідно встановити. Натискаєте клавішу “y” і підтверджуєте операцію.

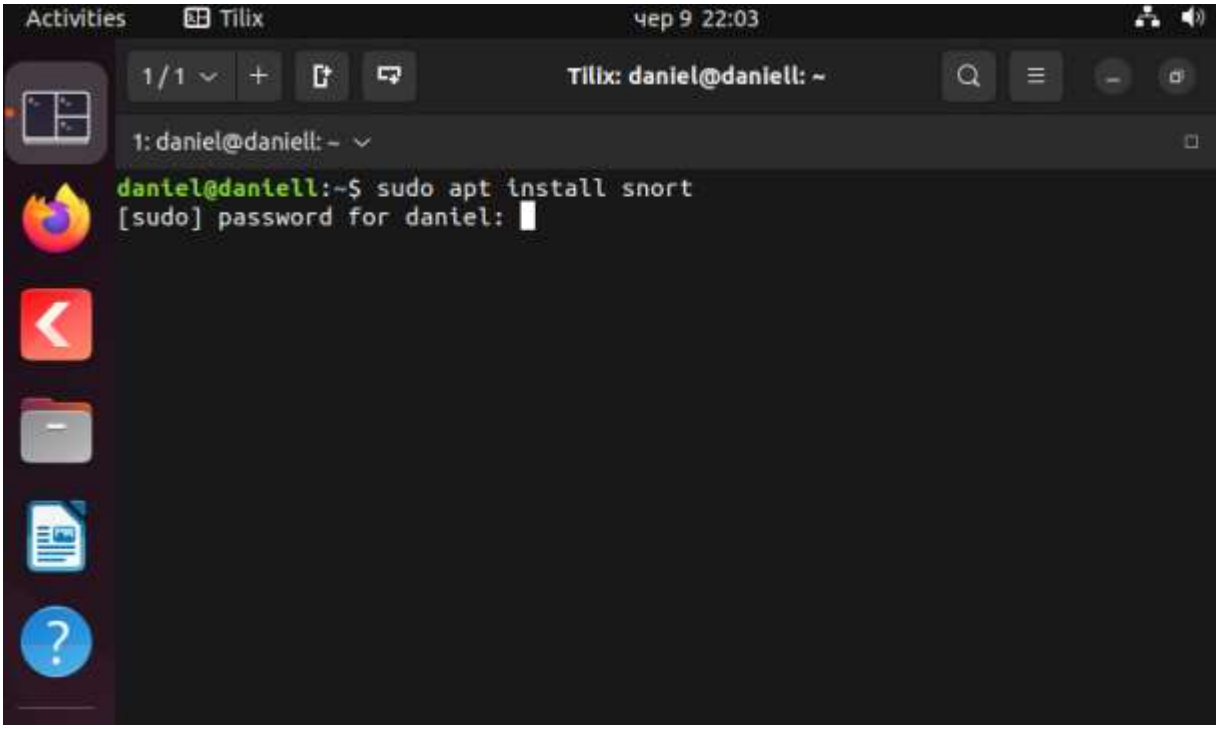

Рис. 3.16. Підтвердження завантаження Snort

Після недовгого очікування програма буде встановлена на ПК і можна приступати до його конфігурації.

Після завантаження Snort автоматично відкриється та запропонує вам налаштувати його інструментарій, як показано на рисунку 3.17.

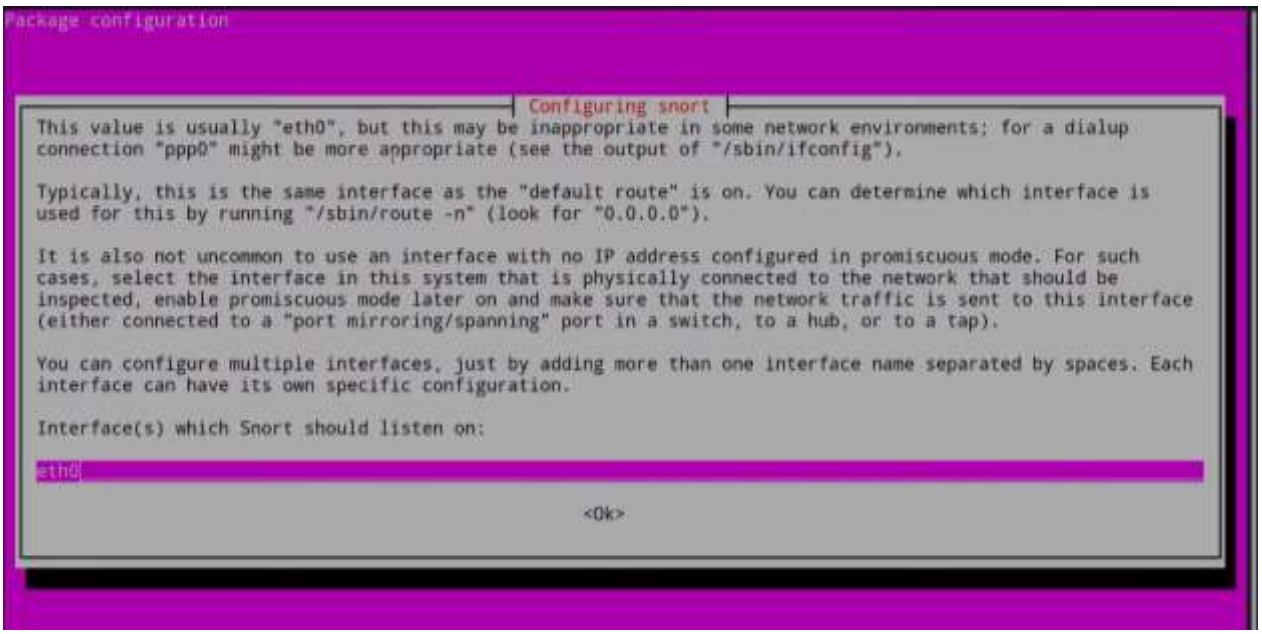

Рис. 3.17. Початкове вікно конфігурації Snort

У рядку необхідно ввести інтерфейс своєї мережі, і продовжити далі. Після ведення, програма приведе IP-адресу разом з діапазоном вашої мережі(рисунок 3.18), тож міняти немає необхідності, тому просто продовжити.

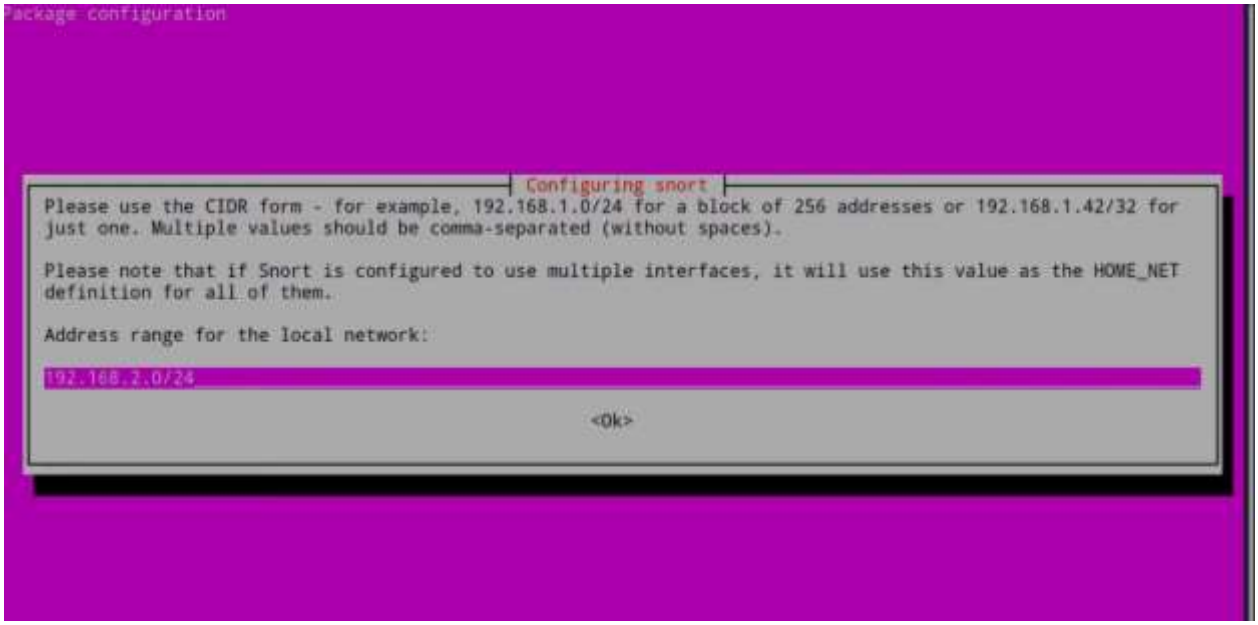

Рис. 3.18. Вікно вибору IP-адреси

Після виконання цієї операції буде відбуватися завантаження додаткових конфігурацій, після чого, програма буде налаштована повністю.

3.3. Встановлення IPS-програми Suricata

Завантаження Suricata не відрізняється від завантаження Snort окрім назви програми, тому після ввімкненого терміналу, пишемо команду:

sudo apt install suricata

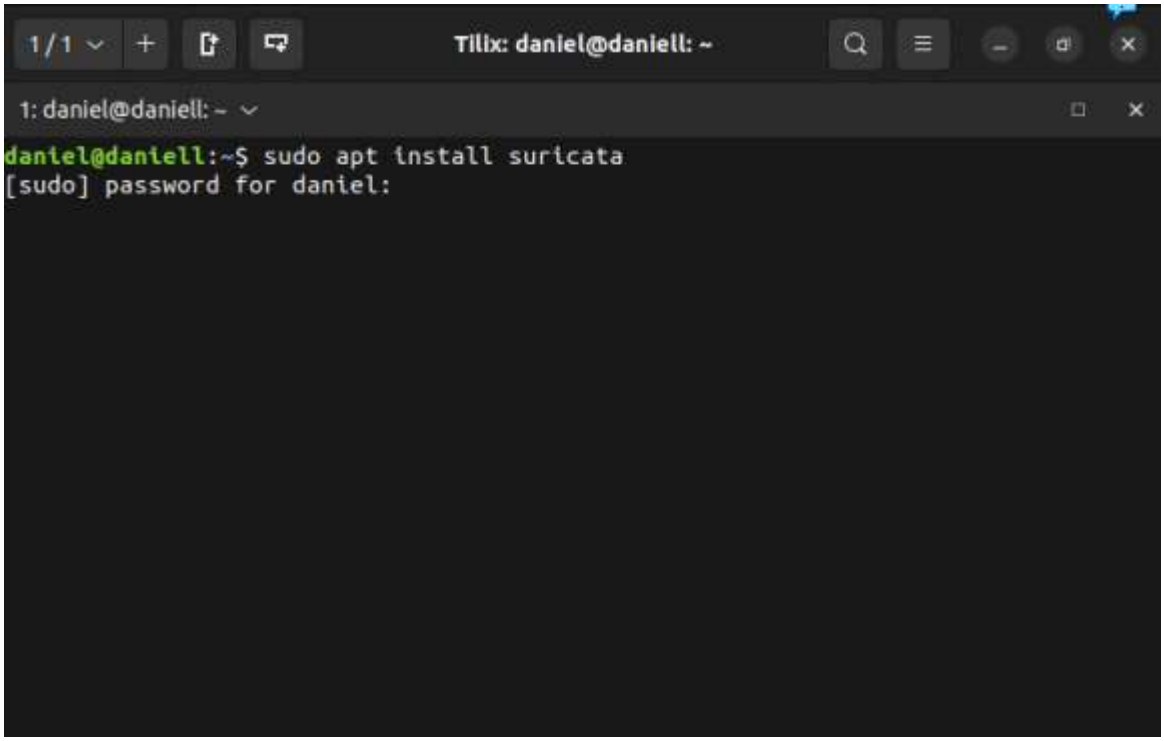

Рис. 3.19. Завантаження Suricata

Після введеного паролю та підтвердження завантаження (рисунок 3.19), програма встановлюється на ваш ПК, а значить можна приступати до її конфігурації.

Для початку конфігурації програми необхідно знайти файл конфігурації з форматом .yaml за допомогою команди:

ls –al /etc/suricata

Після підтвердження термінал має вивести список файлів, розташованих у папці suricata як на рисунку 3.20

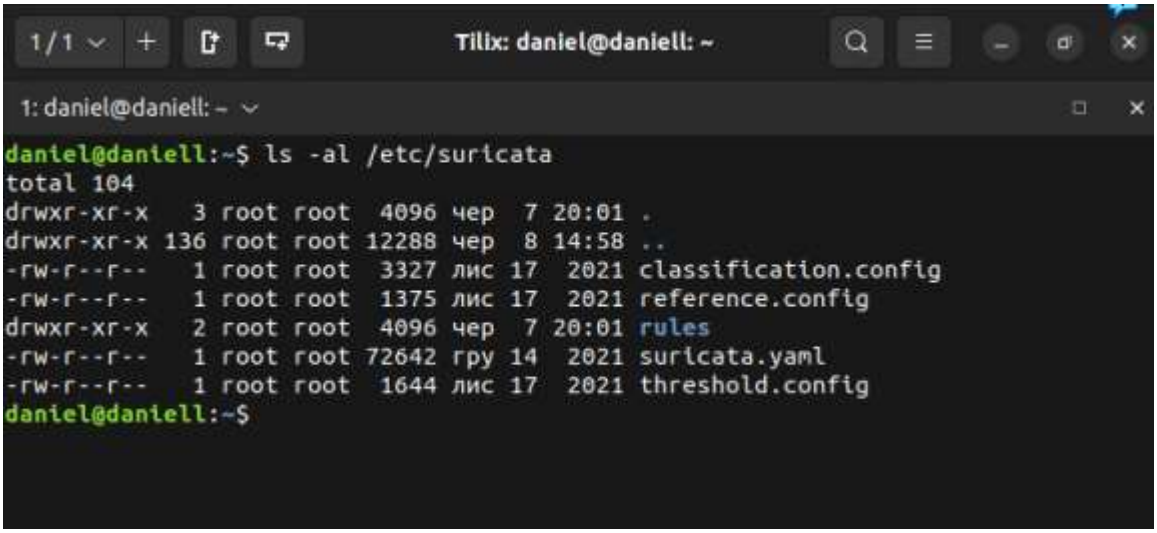

Рис. 3.20. Результат введення команди

Серед цих файлів помічаємо потрібний для виконання налаштування програми, а саме “suricata.yaml”, тому вводимо наступну команду, яка показана на рисунку 3.21 :

sudo vim /etc/suricata/suricata.yaml

Рис. 3.21. Команда редагування файлу конфігурації

Якщо все зроблено правильно, перед Вами має відкритись наступне, як на рисунку 3.22 :

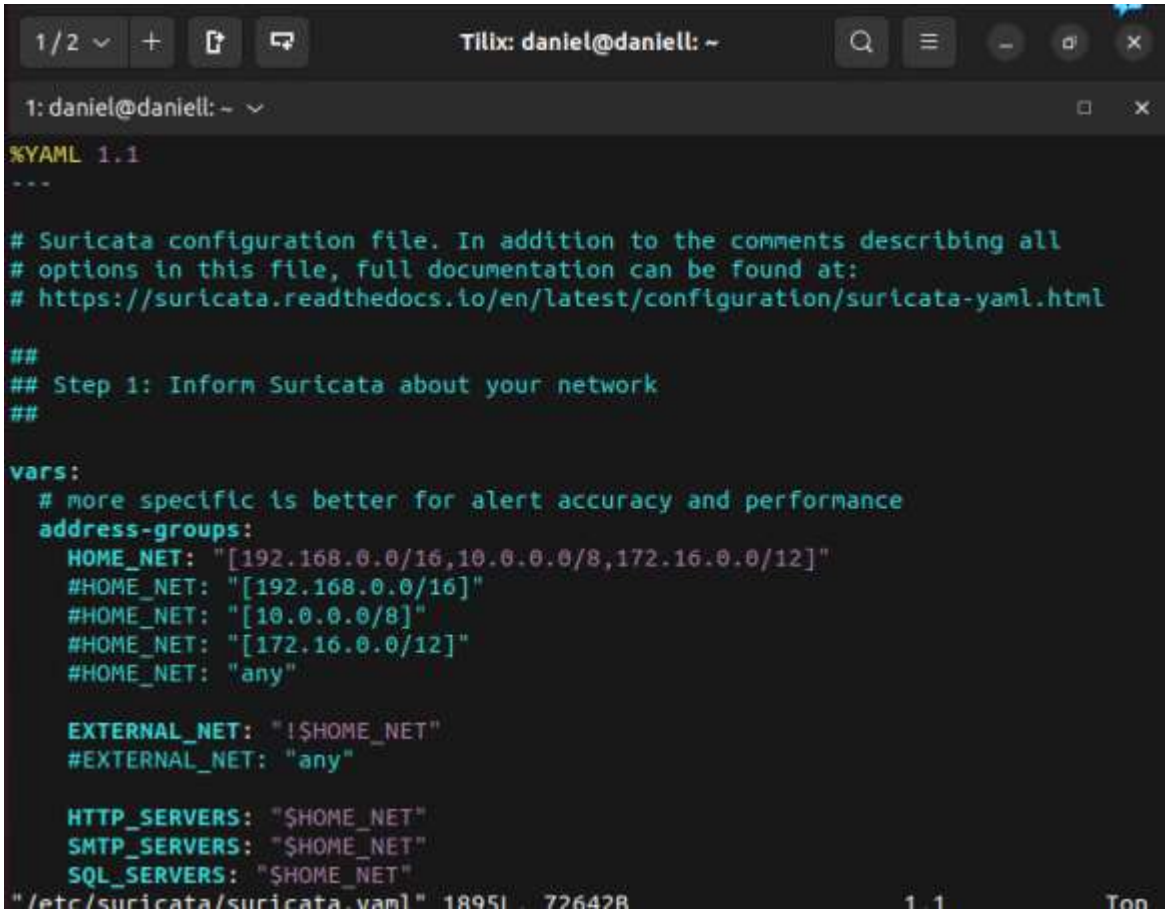

Рис. 3.22. Файл конфігурації в режимі редагування

Далі натискаємо клавішу “Insert” на клавіатурі, та знаходимо рядок “HOME_NET” та редагуємо IP-адресу на ту, яка стоїть в нашій мережі (проілюстровано на рисунку 3.23).

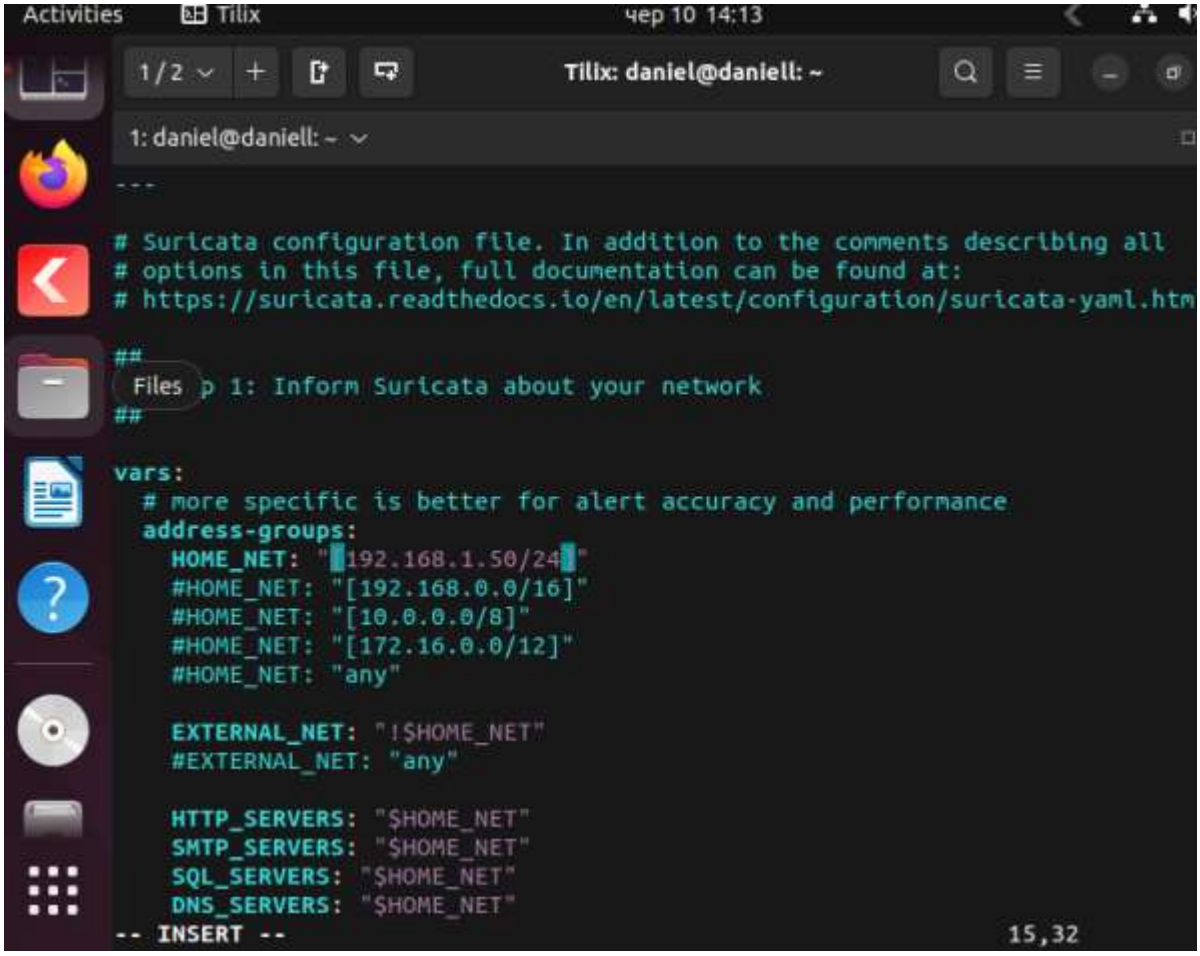

Рис. 3.23. Редагування IP-адреси

Після проведення цих операцій, можна натиснути на клавішу “Escape” для виходу з режима редагування та застосувати комбінацію клавіш “Esc+Shift” та два рази букву “Z” для закриття файлу конфігурації зі збереженими нововведеннями у файлі.

Висновки за розділом

У цьому розділі було детально описано завантаження операційної системи Ubuntu та програм для виявлення та запобігання вторгненню в мережу. Завдяки налаштуванням програм IPS мережа буде відстежувати увесь трафік, що протікає через обрану мережу та виявляти “шкідливий” трафік, ціль якого проникнення в мережу, але IPS-система, що складається з двох безкоштовних

програм, буде блокувати хакерські атаки на мережу, тим самим, запобігши великим збиткам компанії, яку захищає IPS-система.

Під час встановлення та налаштування Ubuntu, Snort та Suricata для системи виявлення вторгнень, було зроблено декілька спостережень.

По-перше, Ubuntu є популярним та надійним дистрибутивом операційної системи Linux, який забезпечує стабільну платформу для встановлення та роботи з СВВ. Встановлення Ubuntu здійснюється шляхом стеження за кроками установки та налаштування основних параметрів.

По-друге, Snort та Suricata є двома популярними системами виявлення вторгнень. Snort є легким у використанні та має низькі системні вимоги, що робить його доступним для широкого кола користувачів. З іншого боку, Suricata має вищі вимоги до ресурсів, але його масштабованість та точність виявлення вторгнень вищі, особливо при використанні багатоядерних систем.

У загальному висновку можна сказати, що вибір між Snort та Suricata залежить від вимог до системи та обраної архітектури. Якщо важлива легкість використання та низькі вимоги до ресурсів, Snort може бути практичним варіантом. Однак, для більш точного виявлення вторгнень та можливості масштабування, Suricata є більш привабливим варіантом, зокрема при використанні багатоядерних систем.

Враховуючи ці фактори, вибір між Snort та Suricata повинен бути здійснений з урахуванням конкретних вимог та обмежень системи виявлення вторгнень.

ВИСНОВКИ

Швидкий розвиток мережевих технологій призводить до появи нових загроз для комп'ютерних мереж. Існує велика кількість різних методів вторгнень та атак, що потребує постійного вдосконалення технологій та засобів захисту даних в корпоративних комп'ютерних мережах.

У сучасних умовах постає важливе питання про безпеку корпоративних мереж у підприємств і компаній. Корпоративні мережі є ключовим компонентом мережевої інфраструктури підприємств, оскільки вони передають як внутрішню інформацію, доступну лише співробітникам, так і інформацію для клієнтів, яка передається через захищені канали зв'язку глобальної мережі. Незаконне втручання в ці мережі може призвести до збоїв у роботі, погіршення якості надання послуг компанією, витоку конфіденційної інформації (яка є головною метою кіберзлочинців), що може призвести до втрати клієнтів і значних фінансових втрат.

Для забезпечення безпеки корпоративних мереж вивчалося питання контролю та аналізу мережевого трафіку в цих мережах. Контроль мережі є важливою практичною задачею. Моніторинг корпоративної мережі є важливою функцією в галузі IT, яка може призвести до зниження витрат, підвищення продуктивності інфраструктури та покращення ефективності роботи співробітників.

Для моніторингу та аналізу мережевого трафіку існує різноманітні програмні та апаратні засоби, як-то інтегровані системи діагностики та управління, аналізатори протоколів, експертні системи та мережеві аналізатори.

Для аналізу мережевого трафіку існують два способи: аналіз у реальному часі (під час роботи) та ретроспективний аналіз мережевого трафіку. Ретроспективний аналіз трафіку передбачає запис всього або частини трафіку на диск для подальшого аналізу.

Використання моделей мережевого трафіку для прогнозування можливих станів захищеної системи може бути обмеженим, оскільки відомі моделі спрямовані на конкретний тип трафіку. Це вимагає значних досліджень для

адаптації моделі до параметрів мережевої конфігурації та трафіку. Такий процес стає складним, оскільки різноманітні джерела та конфігурації мережі мають великий вплив на її ефективність.

Очевидно, що перед проведенням аналізу мережевого трафіку необхідно спершу перехопити цей трафік. Для цього використовуються програмні інструменти, які називаються сніферами. Сніфери слідкують за конкретним мережевим інтерфейсом і можуть перехоплювати трафік, а при необхідності зберігати його в архіві. Однак, загалом вони не мають вбудованої системи для аналізу отриманих результатів, тому використання сніферів самостійно для забезпечення безпеки комп'ютерних мереж є недостатнім.

Крім перехоплення вхідного мережевого трафіку, системи виявлення вторгнень у комп'ютерну мережу також проводять його аналіз. Однак вибір системи виявлення вторгнень має ґрунтуватися на вимогах безпеки конкретної мережі. Для правильного вибору захисної системи, яка ефективно працюватиме у корпоративній мережі підприємства, було проаналізовано методи виявлення вторгнень у комп'ютерну мережу.

Алгоритми, що базуються на нейронних мережах, відзначаються своєю адаптивністю та низькою складністю обчислень. Однак, для ефективного виявлення невідомих типів атак необхідно створити правильний навчальний набір для системи. Якщо навчальний набір містить помилки, це може призвести до неефективної роботи системи виявлення вторгнень. Серед інших методів адаптивного виявлення вторгнень, метод статистичного аналізу та аналогічний метод кластерного аналізу показують добру ефективність, хоча існує ймовірність помилково позитивних результатів.

Для оцінки ефективності статистичного методу виявлення вторгнень у комп'ютерну мережу за допомогою віртуалізації було створено модель атаки типу HTTP-flood на локальній мережі. Виявлення атаки ґрунтувалося на порівнянні статистичних характеристик трафіку в нормальному та аномальному стані. Для цього були використані такі характеристики, як середнє значення, дисперсія, коефіцієнт асиметрії, ексцес та контрексцес. Для порівняння розподілів, що були сформовані для кожної статистичної характеристики,

застосовувався критерій згоди Пірсона, який відображає наявність лінійної залежності між двома розподілами.

Значення дисперсії під час атаки типу HTTP-flood були розподілені на проміжок від 28 до 147 секунд після початку відліку. Виявлення атаки на мережу стало можливим через 120 секунд, тобто 92 секунди після початку самої атаки.

Коефіцієнт кореляції Пірсона для цього розподілу становить 0,9141, що свідчить про те, що обрана ділянка на 91% корелює з прикладом нормального розподілу. Це вказує на наявність мережевої атаки.

Отже, перед нами стоїть завдання розробити методику захисту корпоративної комп'ютерної мережі підприємства, яка поєднуватиме ефективність захисту, універсальність застосування і низьку вартість. Для досягнення цієї мети було прийнято рішення використовувати одну з наявних систем виявлення вторгнень як основу для захисту корпоративної комп'ютерної мережі.

Були розглянуті декілька безкоштовних програмних комплектів для виявлення вторгнень, зокрема Snort, Suricata, EasyIDS, Bro і Openwind Tripwire. Серед цих систем IDS виділяються Snort і Suricata, які можна вважати стандартами в цій галузі. Suricata відрізняється більшою гнучкістю порівняно з Snort. Однак багато постачальників вибирають Snort як основу для розробки та впровадження систем виявлення вторгнень.

У дослідженні були порівняні дві системи - Suricata та Snort. Вибір цих систем обумовлений їх функціональністю, оскільки обидві використовують не лише сигнатурний аналіз трафіку, а й статистичні методи. Також важливим чинником є їх відкритий вихідний код та безкоштовна ліцензія, що дозволяє їх використання. Крім того, обидві системи можуть бути інтегровані з іншими додатками, оскільки вони не залежать від графічного подання інформації.

Для оцінки ефективності систем виявлення вторгнень було створено тестове середовище на платформі віртуалізації VMware Workstation. Були налаштовані дві системи для розміщення систем виявлення вторгнень (СВВ) і

одна система для генерації мережевого трафіку. Аномальний трафік було створено за допомогою Metasploit Framework.

Атакуючий трафік був направлений через обидві системи виявлення вторгнень та інтрузії з різними конфігураціями процесора. У цих конфігураціях враховувалися наступні параметри: наявність двох ядер процесора, використання одного ядра, навантаження на процесор у розмірі 50% та 75%. Було виміряно здатність систем виявлення вторгнень та інтрузій до читання пакетів, а також точність їх попереджень. Особлива увага була приділена виявленню помилково негативних результатів.

Suricata виявляється більш точним в порівнянні з Snort. Під час атаки, коли центральний процесор працював з меншим навантаженням (менше 50%), Snort не зміг передбачити експлойт ms01_033_idq. Це частково пов'язано з тим, що Snort має меншу кількість контрольованих оповіщень під час атаки, порівняно з Suricata (два оповіщення проти чотирьох). Два правила з набору правил VRT, що використовує Snort, також не дозволили йому попередити ms01_033_idq.

Suricata вимагає високої обробки, що дозволяє йому досягати більш високої продуктивності порівняно з Snort. Це пояснюється значною кількістю відкинутих пакетів при великому навантаженні. Зрівнюючи це з Snort, який має нижчі системні вимоги, він не може обробляти пакети з такою втратою під максимальним навантаженням системи. У багатоядерній конфігурації Suricata демонструє менші втрати пакетів в порівнянні з Snort. Suricata також краще розподіляє роботу між доступними ядрами.

Аналіз тестів в автономному режимі, коли трафік зберігається у файлі pcap, показує, що Suricata працює значно повільніше, ніж Snort. Незважаючи на те, що Suricata ефективніше використовує багатоядерну систему порівняно з Snort. З цього можна зробити висновок, що Suricata має кращу масштабованість. Однак, якщо Snort показує гарну продуктивність пропускної здатності, рекомендується запускати декілька екземплярів Snort на кількох ядрах. Це може забезпечити подібну масштабованість, як у Suricata, але з додатковими витратами на обробку однопотокових додатків на кількох ядрах.

Тому, для забезпечення безпеки корпоративних мереж підприємств і компаній, рекомендується використовувати комбінацію сигнатурного та статистичного методів виявлення вторгнень в мережу. Використання цих методів окремо не є найбільш ефективним підходом, тому краще використовувати системи запобігання вторгненням в комп'ютерну мережу, оскільки вони успішно поєднують обидва методи. Зокрема, для мереж з обмеженими апаратними та обчислювальними ресурсами рекомендується використовувати СВВ Snort, а для потужних ресурсів – Suricata.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Корячко В.П. Корпоративные сети: технологии, протоколы, алгоритмы / В.П. Корячко, Д.А. Перепелкин. – М.: Гор. линия-Телеком, 2013. – 219 с.

2. Ложковский А.Г. Модель мультисервисного трафика и метод расчета параметров QoS при его обслуживании / А.Г. Ложковский // Радиотехника. – 2009. – Вып. 157. – С. 48 – 52.

3. Добровольский Е.В., Нечипорук О.Л. Имитационное моделирование источников нагрузки в сетях передачи данных с коммутацией пакетов / Е.В. Добровольский, О.Л. Нечипорук // Наукові праці ОНАЗ ім. О.С. Попова. – 2000. – № 3. – С. 19 – 23.

4. Олифер В.Г. Компьютерные сети. Принципы, технологии, протоколы: учебник для вузов. 4-е изд. / В.Г. Олифер, Н.А. Олифер. – СПб.: Питер, 2010. – 944 с.: ил.

5. Методи аналізу та моделювання безпеки розподілених інформаційних систем: навч. посіб. / В.В. Литвинов, В.В. Казимир, І.В. Стеценко та ін. – Чернігів: Чернігівський національний технологічний університет, 2016. – 254 с.

6. *Feldmann A., Gilbert A.C., Willinger W. Data Networks as Cascades: Investigating the multifractal nature of Internet WAN traffic. / A. Feldmann, A.C. Gilbert, W. Willinger // ACM SIGCOM*. – 1998. – p. 42 – 55.