ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Розробка автоматизованої системи виявлення мережевих аномалій засобами штучного інтелекту»

на здобуття освітнього ступеня бакалавра

зі спеціальності *122 Комп'ютерні науки*
(код, найменування спеціальності)

освітньо-професійної програми *штучний інтелект*
(назва програми)

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

_____ Максим МАРКОВСЬКИЙ
(підпис)

Виконав: здобувач вищої освіти групи ШІД-41
Максим МАРКОВСЬКИЙ

Керівник Тетяна КИСІЛЬ

Рецензент _____

Київ 2024

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра Штучний інтелект
Ступінь вищої освіти Бакалавр
Спеціальність 122 Комп'ютерні науки
Освітньо-професійна програма штучний інтелект

ЗАТВЕРДЖУЮ
Завідувач кафедри ШІ
_____ Ольга ЗІНЧЕНКО
"___" _____ 2024 року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Марковський Максим Олексійович

1. Тема кваліфікаційної роботи: «Розробка автоматизованої системи виявлення мережевих аномалій засобами штучного інтелекту»
керівник кваліфікаційної роботи Кисіль Тетяна, старший викладач кафедри.
(прізвище, ім'я, науковий ступінь, вчене звання)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «___» _____ 2024 року № ____.

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи 31.06.2024 р.
3. Вихідні дані до кваліфікаційної роботи
рішення на базі алгоритмів SIEM ELK Stack;
наукова та технічна література, експлуатаційна документація, нормативні документи.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1. Визначення та класифікація аномалій в мережевій інфраструктурі.
2. Загрози та ризики, пов'язані з аномальною активністю в мережі. Використання SIEM-систем для виявлення аномалій.

3. Розроблення рекомендацій щодо проектування оптимізованої мережевої інфраструктури.

4. Перелік графічного матеріалу
Презентація PowerPoint.

6. Дата видачі завдання 15.04.2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ зп	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Визначення та класифікація аномалій в мережевій інфраструктурі	15.03.2024 р.	
2.	Аналіз наукової та технічної літератури.	31.03.2024 р.	
3.	Аналіз методів та засобів оптимізації мережевої інфраструктури.	10.04.2024 р.	
4.	Огляд технології ELK Stack. Переваги та недоліки використання ELK Stack для виявлення аномалій	25.04.2024 р.	
5.	Розроблення рекомендацій використання інструментів ELK Stack для розробки та реалізації алгоритмів.	10.05.2024 р.	
6.	Оформлення результатів дослідження.	27.05.2024 р.	
7.	Підготовка доповіді до захисту.	31.05.2024 р.	

Здобувач вищої освіти _____ Максим МАРКОВСЬКИЙ
(підпис)

Керівник кваліфікаційної роботи _____ Тетяна КИСІЛЬ
(підпис)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи: 71 сторінка, 34 рисунки, 1 таблиця, 25 джерел.

Об'єкт дослідження – процес підвищення продуктивності алгоритмів виявлення аномалій атак інструкторів та небажаних подій.

Предмет дослідження – алгоритм виявлення аномалій в мережевій інфраструктурі SIEM ELK Stack.

Метою даної – підвищення продуктивності алгоритмів виявлення аномалій для захисту мережевої інфраструктури з використанням SIEM ELK Stack.

- Огляд теоретичних аспектів виявлення аномалій у мережевій безпеці.

- Розгляд можливостей SIEM ELK Stack для виявлення та аналізу аномалій.

- Розробка та реалізація алгоритмів виявлення аномалій на основі SIEM ELK Stack.

- Тестування та оцінка ефективності розроблених алгоритмів на реальних або симульованих даних.

Методи дослідження. Для досягнення поставлених цілей використовуватимуться наступні методи дослідження:

- Аналіз наукової літератури та статей, присвячених темі виявлення аномалій у мережевій безпеці.

- Експериментальні дослідження, що включають в себе розробку та тестування алгоритмів виявлення аномалій на реальних або симульованих даних.

- Методи аналізу та порівняння результатів, що використовуються для оцінки ефективності розроблених алгоритмів.

- Використання практичних методів розробки програмного забезпечення для реалізації алгоритмів виявлення аномалій.

Галузь використання – інформаційні технології ресурсів організації.

ВИЯВЛЕКННЯ АНОМАЛІЙ, МЕРЕЖЕВА ІНФРАСТРУКТУРА, SIEM, ELK STACK, ГЛИБОКЕ НАВЧАННЯ, БЕЗПЕКА МЕРЕЖІ, ЖУРНАЛИ ПОДІЙ, LIGHTWEIGHT LOAD

ЗМІСТ

ВСТУП.....8
1 ТЕОРЕТИЧНИЙ АНАЛІЗ АНОМАЛІЙ МЕРЕЖЕВОЇ ІНФРАСТРУКТУРИ..9
1.1 Визначення поняття «аномалії в мережевій інфраструктурі».....9
1.2 Загрози та ризики, пов'язані з аномальною активністю в мережі. 15
1.3 Методи та підходи до виявлення аномалій. 19
Висновок до першого розділу.....24
2. АНАЛІТИЧНИЙ ОГЛЯД ТЕХНОЛОГІЇ SIEM ELK STACK ДЛЯ РОЗРОБКИ АЛГОРИТМІВ ВИЯВЛЕННЯ АНОМАЛІЙ 25
2.1 Огляд технології SIEM ELK Stack.....25
2.2 Приклад використання SIEM ELK Stack для виявлення аномалій.....32
2.3 Переваги та обмеження використання SIEM ELK Stack для виявлення аномалій.....38
2.4 Розроблення та класифікація логістичної регресії……...45
Висновок до другого розділу 43
3. ПРАКТИЧНА РЕАЛІЗАЦІЯ АЛГОРИТМІВ ВИЯВЛЕННЯ АНОМАЛІЙ З ВИКОРИСТАННЯМ SIEM ELK STACK.....45
3.1. Підготовка середовища SIEM ELK Stack та збір даних.....45
3.2. Розробка та тестування алгоритмів виявлення аномалій.....56
Висновок до третього розділу 61
ВИСНОВКИ 62
ПЕРЕЛІК ПОСИЛАНЬ.....64
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ.....67

ВСТУП

Актуальність дослідження. Актуальність дослідження виявлення мережевих аномалій за допомогою систем штучного інтелекту полягає в тому, що з кожним днем загрози для мережевої безпеки стають все більш складними та різноманітними. Традиційні методи виявлення аномалій часто не можуть ефективно впоратися з новими видами загроз, такими як розроблені вдосконалені атаки, фішинг, введення зловмисного коду та інші.

Застосування систем штучного інтелекту виявляється в тому, що вони можуть автоматизувати процес виявлення аномалій у реальному часі, аналізуючи великі обсяги мережевих даних та виявляючи незвичайні або неповторювані патерни, які можуть свідчити про атаки або недоліки в мережі.

Дослідження в галузі виявлення мережевих аномалій за допомогою систем штучного інтелекту має велике значення для підвищення рівня кібербезпеки в організаціях та захисту їх мереж від різноманітних загроз. Такі системи можуть виявити аномальну активність навіть у великих мережах та допомогти адміністраторам реагувати на потенційні загрози швидше та ефективніше.

Об'єкт дослідження – процес підвищення продуктивності алгоритмів виявлення аномалій атак інструкторів та небажаних подій.

Предмет дослідження – алгоритм виявлення аномалій в мережевій інфраструктурі SIEM ELK Stack.

Метою даної – підвищення продуктивності алгоритмів виявлення аномалій для захисту мережевої інфраструктури з використанням SIEM ELK Stack.

Методи дослідження – опрацювання літератури за даною темою, аналіз експлуатаційної документації.

1 ТЕОРЕТИЧНИЙ АНАЛІЗ АНОМАЛІЙ МЕРЕЖЕВОЇ ІНФРАСТРУКТУРИ

1.1 Визначення поняття «аномалії в мережевій інфраструктурі»

Аномалії в мережевій інфраструктурі - це несподівані чи непередбачені події, що виникають у комп'ютерних мережах або мережевих системах. Вони можуть свідчити про можливі проблеми чи загрози безпеці. Причини виникнення таких аномалій можуть бути різноманітні: технічні неполадки, атаки з боку зловмисників, дія вірусів, недоліки програмного забезпечення або навіть недбалість користувачів. [1].

Типові аномалії в мережевій інфраструктурі включають:

1. Атаки з боку зловмисників: Це спроби несанкціонованого доступу до мережевих ресурсів або використання вразливостей в мережевому обладнанні для отримання незаконних привілеїв.
2. Віруси і зловмисне ПЗ: Інфіковані комп'ютери в мережі можуть проявляти аномальну активність, таку як надмірне використання мережевого трафіку або спроби непередбачуваних з'єднань.
3. Неполадки обладнання: Проблеми з мережевим обладнанням, такі як перевантаження комутаторів, роутерів або інших пристроїв, можуть спричинити аномальну поведінку в мережі.
4. Недоліки програмного забезпечення: Помилки в програмах або операційних системах, що керують мережею, можуть призвести до непередбачуваної поведінки або вразливостей, які можуть бути використані для атак.
5. Навігаційні помилки користувачів: Неправильні налаштування або дії користувачів, такі як неправильна конфігурація мережних параметрів або неправильне використання програм, також можуть викликати аномальну поведінку в мережі. [1].

Виявлення та вирішення аномалій в мережевій інфраструктурі є ключовим етапом для забезпечення безпеки та ефективності роботи мережі. Це вимагає

використання спеціалізованих програмних засобів для моніторингу та аналізу мережевого трафіку, а також методів обробки великих обсягів даних для виявлення аномальних відмінностей в звичайному шаблоні поведінки мережі.

Більшість сучасних методів виявлення атак ґрунтуються на аналізі за допомогою правил. Цей аналіз базується на заданому наперед наборі правил, які можуть бути надані адміністратором, автоматично згенеровані системою або поєднанням обох. Експертні системи є одним з найпоширеніших підходів до виявлення атак на основі правил. Ранні дослідження виявлення атак показали, що методи, що вимагають ручного перегляду журналів подій, були неефективними. Інформація, необхідна для виявлення атак, часто представлена у вигляді великих обсягів аудитованих даних, які потребують аналізу з використанням автоматизованих систем для ефективного оброблення. [2].

Експертні системи вбудовані на основі набору правил, які містять знання людей-експертів. Ці правила використовуються системою для оцінки стану безпеки на підставі даних, що надходять від систем виявлення атак. Експертні системи дозволяють інтегрувати обширний досвід експертів у комп'ютерні програми та використовувати ці знання для виявлення аномальної поведінки, що відповідає специфічним ознакам зловживань та атак.

Проте для підтримки адекватності експертні системи потребують постійного оновлення. Вони надають можливість перегляду записаних даних, але в разі відсутності оновлень вони можуть стати неактуальними. Як мінімум, це може знизити ефективність експертної системи. У гіршому випадку, відсутність технічної підтримки може вплинути на загальний рівень безпеки мережі та створити недоліки щодо усвідомлення користувачами рівня безпеки мережі. [2].

Системи, що ґрунтуються на правилах, часто мають обмеження у виявленні сценаріїв атак, що можуть тривати протягом довгого часу. Хоча така система може помітити окремі випадки підозрілої активності, вона може пропустити їх, якщо вони відбуваються в ізоляції. Крім того, сценарії атак, в яких декілька хакерів працюють разом, складно виявити за допомогою цих методів, оскільки система не зосереджена на динаміці атаки, а шукає лише окремі відповідні

елементи. Одиночні атаки, що тривають протягом тривалого періоду або виконуються кількома хакерами, які, здається, не пов'язані між собою, також складно виявити за допомогою цих методів [1, 2].

Коли контролер отримує невідомий пакет через протокол OpenFlow, пакет містить повну інформацію про нього. Після того, як контролер отримує пакет, він розбирає інкапсуляцію пакета. Тобто, він розбирає кожен шар пакета, витягуючи інформацію з кожного шару пакета шар за шаром, ніби зчищаючи аноніон. Таким чином, для пакета може бути створена таблиця потоків на основі витягнутої інформації. Потім, пакет «packet-out» передається на плату обміну через протокол OpenFlow. На основі цього принципу можна обробляти пакети, що надходять від плати обміну, а потім витягувати необхідну інформацію і передавати її навченій моделі для ідентифікації та прогнозування, результати якої показані на рис. 1.1. Спочатку запускається контролер Mininet і RYU. І відправляє пакети, в цей час комутатор пересилає перший пакет контролеру. Контролер передає пакет до моделі виявлення аномалій CNN. Після попередньої обробки даних, навчання та моделювання CNN використовуються для виявлення аномального трафіку. Якщо це нормальний трафік, пакет передається безпосередньо. Якщо виявлено аномальний трафік, блокується, перенаправляється та відкидається - це три способи пом'якшити атаку аномального трафіку.

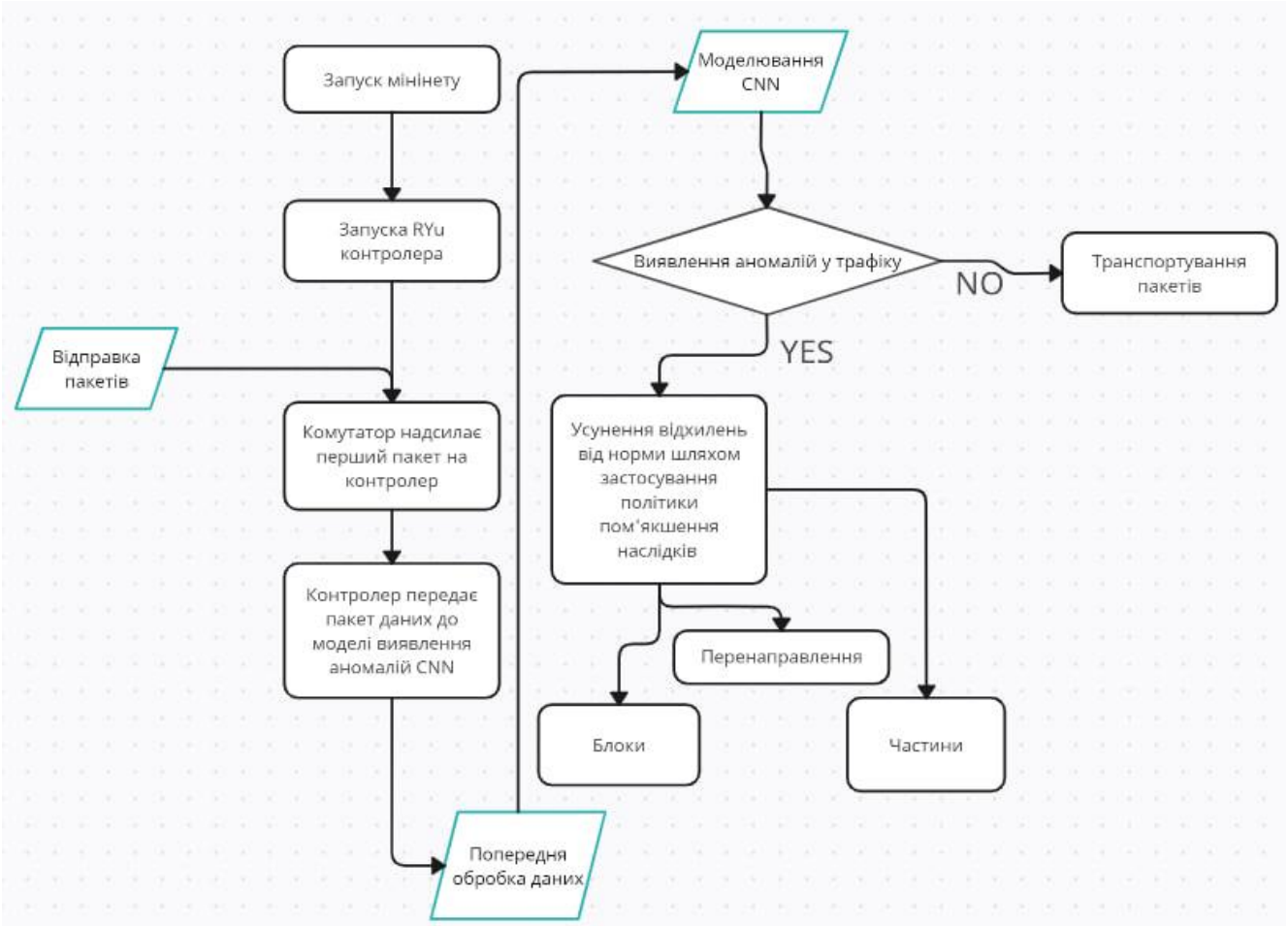


Рис. 1.1 Принципова схема реалізації виявлення аномалій

Існують дві основні технології для виявлення комп'ютерних атак: системи виявлення аномалій (СВА) і системи виявлення вторгнень (СВВ). Обидві ці технології ґрунтуються на аналізі даних з метою виявлення аномальних або підозрілих змін в мережевому трафіку, системних операціях або інших показниках, які можуть свідчити про потенційні атаки або вторгнення.

1. Системи виявлення аномалій (СВА) аналізують типове функціонування мережі або системи та виявляють будь-які відхилення від звичайних шаблонів. Вони використовують методи статистичного аналізу, машинного навчання та інші техніки для виявлення незвичних патернів. Наприклад, якщо зазвичай мережа споживає певний обсяг трафіку протягом певного часу, а потім раптово цей обсяг збільшується або зменшується, це може вказувати на потенційну атаку або проблему з мережевим обладнанням.

2. Системи виявлення вторгнень (СВВ), натомість, використовують

визначені правила або сигнатури, які відомі як методи вторгнення. Вони порівнюють активність в мережі або на комп'ютерах з цими сигнатурами, щоб виявити підозрілі дії, які можуть бути пов'язані з атакою. Наприклад, якщо спостерігається спроба експлуатації відомої вразливості в програмному забезпеченні, система виявлення вторгнень може видачу сповіщення про цю спробу.

Обидві технології мають свої переваги і недоліки, і їх часто використовують разом для більш ефективного виявлення та захисту від комп'ютерних атак.

Важливою частиною процесу виявлення атак є постійне оновлення і покращення систем, щоб вони могли ефективно виявляти нові види загроз і атак.

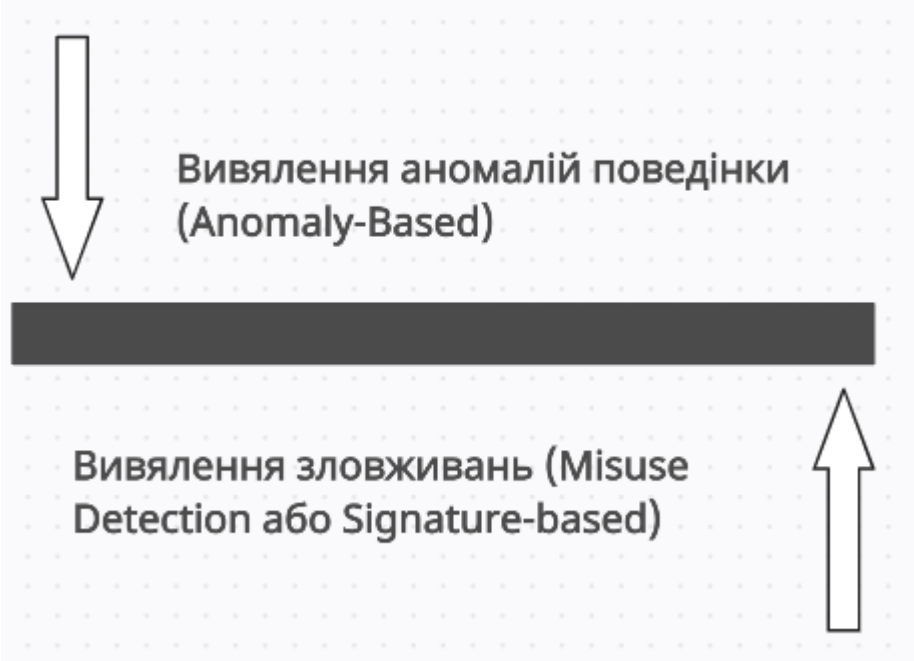


Рис. 1.2. Методи аналізу подій, на яких базується технології побудови СОА

Методика виявлення атак шляхом виявлення аномальної поведінки ґрунтується на припущеннях про те, що аномальна активність користувачів, така як атаки та ворожа поведінка, часто проявляється як відхилення від нормальної поведінки. Ці відхилення відбуваються у формі спроби вторгнення, які відрізняються від звичайних подій взаємодії користувача та вузла мережі. Такі аномалії можна ідентифікувати. Наприклад, велика кількість з'єднань за короткий проміжок часу може свідчити про аномальну активність, таку як високе навантаження на процесор.

Датчики збирають дані про події, створюють шаблони нормальної активності та використовують різні метрики для визначення аномальної активності. Наприклад, якщо можна однозначно описати профіль нормальної поведінки користувача, будь-яке відхилення від цього профілю можна ідентифікувати як аномалію. Проте, важливо зауважити, що аномальна поведінка не завжди є атакою. Наприклад, якщо мережевий адміністратор одночасно надсилає велику кількість запитів, система виявлення атак може помилково ідентифікувати це як атаку "Відмова в обслуговуванні" (DoS).

Методи виявлення аномалій спрямовані на виявлення нових типів атак, але одним з їхніх недоліків є потреба у постійному навчанні. На сьогоднішній день технології виявлення аномалій ще не мають широкого розповсюдження, оскільки вони ще не доступні на ринку. Однак наразі ці технології стають все доступнішими.

Системи виявлення аномалій базуються на тому, що для об'єкта спостереження відомі деякі характеристики, які визначають правильну або прийнятну поведінку. Нормальна або правильна поведінка визначається як та, що відповідає політиці безпеки. Системи виявлення шахрайства базуються на відомих ознаках, які характеризують поведінку об'єкта. [4].

Найпоширенішими реалізаціями методів виявлення зловмисної поведінки є експертні системи (наприклад, Snort, RealSecure IDS, Enterasys Advanced Dragon IDS).

На рис. 1.3 представлені технології систем виявлення атак.

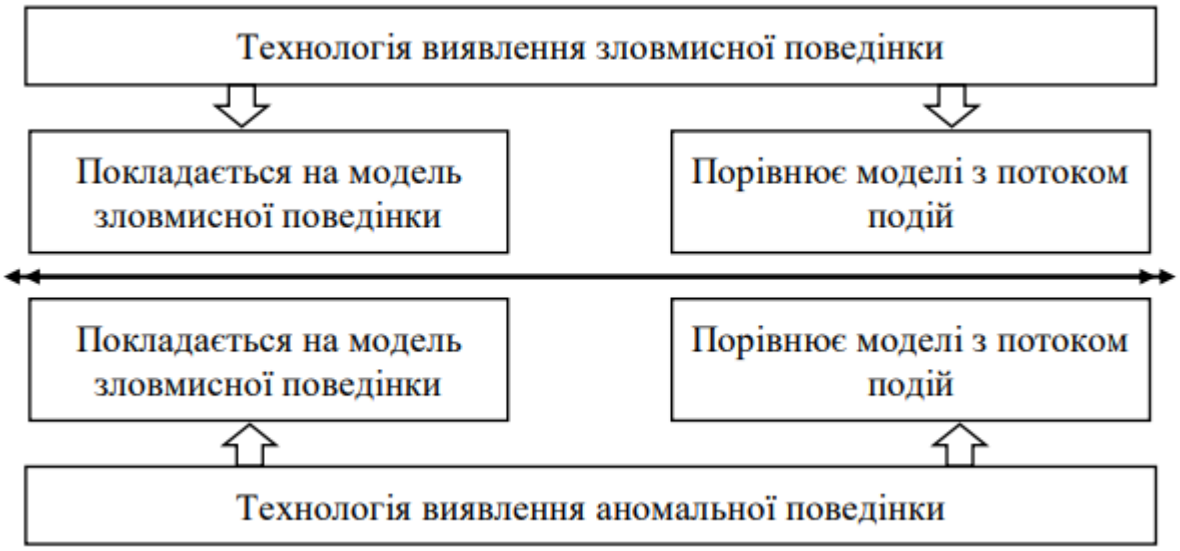


Рис. 1.3. Існуючі технології систем виявлення атак

Датчики аномалій призначені для виявлення незвичайної поведінки або аномалій у функціонуванні окремих об'єктів. Однак їхнє практичне застосування ускладнене через нестабільність самого об'єкта, який захищається, а також зовнішніх об'єктів, що взаємодіють з ними. Об'єктом спостереження може бути мережа в цілому, окремий комп'ютер, мережевий сервіс (наприклад, FTP-сервер), користувач та інше.

1.2 Загрози та ризики, пов’язані з аномальною активністю в мережі.

Аномальна активність в мережі може призвести до різних загроз і ризиків як для індивідів, так і для організацій. На рис. 1.4 зображено, найпоширеніші загрози та ризики, які можна уникнути за допомогою використання шутчного інтелекту.

Рис. 1.4. Найпоширеніші загрози та ризики

1. Шифрування.

В останні роки шифрування стало палицею з двома кінцями. Організації все частіше намагаються захистити конфіденційні дані за допомогою шифрування комунікацій, що, зазвичай, ми сприймаємо як надійний захист. Проте "несхоже шифрування", як повідомило агентство Bloomberg, може створити хибне відчуття комфорту для споживачів.

Справжність полягає в тому, що хакер може здійснювати контроль над пристроями різними способами, включаючи отримання доступу до "повного обговорення", незалежно від заходів безпеки, вбудованих у додаток, який ви використовуєте. Зашифровані дані, по суті, дають хакерам волю дій до того моменту, поки вони не будуть виявлені та усунені.

2. Програми-вимагачі.

Мережеві програми-вимагачі можуть призвести до паралізу систем та втрати даних. Ця загроза особливо тривожна, оскільки вона не потребує значного втручання людини для здійснення атаки та постановки організації на коліна.

Для успішного проведення атаки з вимогою викупу часто достатньо мати активну робочу станцію без необхідних оновлень програмного забезпечення. На заражених пристроях дані можуть бути скомпрометовані або повністю втрачені. Проблему ускладнює той факт, що багато малих і середніх підприємств не повідомляють про атаки з вимогою викупу, коли вони відбуваються [5, 6].

Як підкреслює CPO Magazine, багато атак з використанням програм-вимагачів залишаються неповідомленими, і більшість з них починаються з атак соціальної інженерії.

Профілактичні заходи проти програм-здирників включають:

– Використання різноманітних резервних копій;

– Постійне оновлення антивірусу;

– Постійне оновлення патчів;

– Встановлення програмного забезпечення для моніторингу цілісності файлів і системи;

– Забезпечення відповідності нормативним вимогам.

3. DDoS-атаки.

DDoS-атаки мають реальні наслідки. Серйозність та частота таких атак становлять проблему для багатьох мережевих адміністраторів. Хакери проникають в організації, перенасичуючи їх веб-сайти та мережі підозрілим трафіком. Два основних механізми, що підбадьорюють злочинців, - це послуги "DDoS напрокат", де навички в області злому та атак пропонуються за гроші, та широке використання технології Інтернету речей (IoT). Уразливі пристрої IoT з неадекватним захистом стають легкою мішенню для таких атак. Однак захист від DDoS-атак не повинен бути надто складним завданням.

Нижче наведено способи, які допоможуть захиститися від DDoS-атак [5]:

– Виявлення незвичної активності трафіку;

– Використання необхідного обсягу пропускної здатності;

– Уникнення неправильної реакції на спроби вимагання;

– Негайний контакт з вашим інтернет-провайдером;

– Розробка комплексного підходу до захисту від DDoS-атак.

4. Внутрішні загрози.

Внутрішні загрози продовжують становити проблему для організацій будь-якого розміру. За даними McKinsey & Company, експерти оцінюють, що близько 50% витоків даних виникають через інсайдерські загрози. Частіше за все такі інциденти виникають з метою фінансової вигоди або через недбалість. Хоча усвідомлення про внутрішні загрози зростає в організаціях, більшість підприємств не завжди готові вчасно реагувати, оскільки більшість засобів мережевої безпеки спрямовані на захист від зовнішніх загроз.

Внутрішні зловживання можуть включати в себе, але не обмежуватися ними:

- Віддалений доступ до конфіденційних даних;
- Несанкціоноване видалення даних;
- Несанкціонований доступ до спільних папок;
- Несанкціоноване апаратне/програмне забезпечення.

Організації можуть виявити, що особи, які вже мають законний, санкціонований доступ до конфіденційних даних, діють незаконно, часто з обмеженими або навіть без обмежень на доступ та повноваження. Ця ситуація підкреслює важливість постійного моніторингу підозрілої діяльності. Використання програмного забезпечення для моніторингу цілісності файлів (FIM) може допомогти знизити ризик від дорогих інсайдерських порушень, зокрема інструмент FIM, який забезпечує цілісність та допомагає в реалізації стратегії архітектури нульової довіри (ZTA) [6].

5. Хмарна безпека.

Безпека легальних хмарних сервісів стає все більш проблематичною. Причина полягає в тому, що з ростом популярності хмарних технологій для зберігання та обробки даних, хакери знаходять способи проникнути всередину. Вони використовують ті самі легальні сервіси, але з прихованими мотивами, щоб посіяти хаос.

Організації стають більш вразливими через те, що вони все більше довіряють загальним хмарним платформам і реагують на будь-яку сумнівну діяльність. Це призводить до значних витрат на простої та використання ресурсів для ліквідації наслідків.

Останні інциденти, зокрема той, що стосувався міста Таллахассі, свідчать про це. Наприклад, в результаті фішингової афери, яка використовувала посилання на Dropbox, були завдані великі збитки міському відділу кадрів. Хакери використали вірус, що надійшов з пошти міського голови, щоб отримати доступ до мережі для зміни виплат заробітної плати [6].

З поширенням програмного забезпечення як послуги (SaaS) і переходом послуг до хмари, організаціям слід бути особливо обережними щодо політик і процедур, що можуть викликати помилкове відчуття відповідальності і безпеки даних у хмарі [7].

6. Ін'єкція SQL.

Багато організацій зберігають свої дані на серверах з використанням SQL, що може зробити їх вразливими до SQL-ін'єкцій. Цей тип атак вперше став

відомим у 1998 році, коли зловмисники використовували код для незаконного доступу, зміни або видалення конфіденційної інформації компанії. Вразливості у прикладному програмному забезпеченні можуть дозволити зловмисникам створювати фальшиві облікові записи, маніпулювати даними компанії і навіть анулювати транзакції або змінювати залишки у фінансових записах [7].

Для захисту від таких атак важливо регулярно перевіряти програмне забезпечення на наявність вразливостей. Також необхідно постійно моніторити цілісність файлів, щоб вчасно виявляти та усувати зміни, спричинені атаками SQL-ін'єкцій.**Атаки типу "Людина посередині" (Man-in-the-Middle).**

Атака такого типу відбувається, коли зловмисник "підслуховує" комунікацію, яка повинна бути приватною [7, 8].

У цьому типі атаки зловмисник може перехопити електронну пошту, чат або інше повідомлення між двома сторонами. Потім він може використати свій доступ для підміни повідомлень, зміни даних або здійснення атак соціальної інженерії.

Деякі приклади MIM-атак включають:

- Злам Wi-Fi
- підміна IP-адреси
- перехоплення SSL
- підробка DNS.

1.3 Методи та підходи до виявлення аномалій.

1) Методи, засновані на прогнозуванні

Методи прогнозування використовують навчені моделі для прогнозування майбутніх значень часового ряду на основі його поточного контексту. Зазвичай це означає прогнозування кількох наступних кроків, використовуючи інформацію з попередніх вікон. Прогнозовані значення порівнюються з реальними, щоб визначити, чи є вони аномальними.

Методи в цьому сімействі відрізняються за типом моделі прогнозування, способом її побудови та метрикою для оцінки аномалій. Наприклад, AD-LTI, ARIMA, RBForest, RForest, SARIMA та XGBoosting використовують різні моделі

прогнозування, такі як авторегресійні моделі, ліси випадкових дерев або градієнтний бустінг, і різні метрики, наприклад, відстань між спостережуваними та прогнозованими значеннями.

Ці методи навчаються у напівконтрольованому режимі: моделі навчаються на нормальних даних без аномалій, а потім використовуються для виявлення відхилень у тестових даних. Наприклад, RBForest, RForest, XGBoosting, AD-LTI та LSTM-AD (який може прогнозувати кілька кроків вперед) використовують цей підхід.

2) Методи, що засновані на реконструкції

Засновані на кодуванні методи аналізу часових рядів подібні до методів реконструкції, адже вони також працюють з низькорозмірним представленням даних. Однак, їх основна мета полягає в оцінці аномалій безпосередньо з латентного простору, не вдаючись до спроб реконструювати вихідні підпослідовності.

Наприклад, LaserDBN, PST і MultiHMM використовують імовірнісні моделі для оцінки аномалій. Вони базуються на логарифмі правдоподібності підпослідовностей як міри аномалій. В той час як MultiHMM будує модель з навчального часового ряду, LaserDBN і PST працюють лише з тестовим часовим рядом.

3) Методи, що засновані на кодуванні

Series2Graph використовує низькорозмірне представлення для перетворення підпослідовностей тестового часового ряду у спрямований циклічний граф. Оцінка аномалій базується на частоті переходів між групами підпослідовностей: чим частіше перехід, тим вище оцінка аномалії. Таким чином, точки з низькими оцінками вважаються більш аномальними.

Таким чином, методи на основі кодування пропонують широкий спектр підходів до виявлення аномалій у часових рядах, використовуючи різні стратегії обробки даних та оцінки відхилень від звичайного шаблону.

4) Методи, засновані на особливостях розподілу

Методи розподілу в аналізі часових рядів використовуються для оцінки

розподілу даних або моделювання розподілу на основі наявних даних. Зазвичай розподіл обчислюється за допомогою точок даних або підпослідовностей, отриманих шляхом використання вікон. Подібність між точками або підпослідовностями може враховуватись у процесі оцінки розподілу, проте саме аномалії визначаються за їхньою частотою, а не відстанню до центральних значень.

Зазвичай ці методи вважаються неконтрольованими, оскільки вони спеціалізуються на виявленні аномалій в екстремальних або рідкісних подіях, що можуть виявитися у хвостах розподілів. У напівконтрольованому випадку, де доступно нормальне навчальне середовище, розподіл оцінюється на основі цих даних, і потім використовується для порівняння з тестовим часовим рядом.

Представники методів розподілу, такі як COPOD, DWT-MLEAD, Fast-MCD, HBOS, NF, S-H-ESD, DSPOT і Sub-Fast-MCD, пропонують різноманітні підходи до виявлення аномалій. Наприклад, деякі з них, як DWT-MLEAD, Fast-MCD і Sub-Fast-MCD, оцінюють розподіл Гауса за допомогою дискретного вейвлет-перетворення або інших методів, і визначають аномалії за відстанню до центральних значень розподілу. Інші, як HBOS і COPOD, використовують гістограми або копули для оцінки розподілу та визначення аномалій.

5) Методи дерева ізоляції

Методи ізольованого дерева використовують ансамбль випадкових дерев, щоб виділити аномалії у вибірці тестового часового ряду. Під час побудови кожного дерева випадковим чином обираються функції і значення для розділення, щоб визначити зразки у листках дерева. Довжина шляху, яку проходить зразок від кореня до листка, стає мірою його аномальності, оскільки аномальні зразки в середньому мають коротший шлях.

Представники цього підходу, такі як Extended Isolation Forest (EIF), Hybrid Isolation Forest (HIF), Isolation Forest - Local Outlier Factor (IF-LOF), Isolation Forest (iForest) і Sub-IF, використовують різні варіації методу дерева ізоляції. Наприклад, EIF і HIF є контрольованими варіантами, які розширюють базовий алгоритм iForest. Sub-IF розглядає аномалії у вигляді підпослідовностей, а IF-LOF

комбінує підходи Isolation Forest і Local Outlier Factor.

Ізольовані ліси будуються на ідеї, що аномалії у даних є рідкими та відрізняються від нормальних точок. Зразки, які потрапляють глибше в дерево, мають меншу ймовірність бути аномаліями, оскільки для їх виділення потрібно більше розділень. Таким чином, зразки, які закінчуються коротшими гілками, зазвичай вказують на аномалії, оскільки дереву було легше їх відрізнити від інших спостережень.

Ітеграція SIEM ELK Stack LSTM з автокодувальником.

Автокодувальник LSTM у контексті ELK Stack — це інструмент для обробки послідовностей даних, який використовує архітектуру кодувальника-декодера LSTM. Під час роботи з ELK Stack для даного набору даних послідовностей автокодувальник LSTM налаштований на зчитування, кодування та декодування вхідних послідовностей. Він оцінюється на основі його здатності відтворювати вхідні послідовності. Після досягнення потрібної продуктивності, декодерну частину моделі можна видалити, залишивши лише модель кодера. Цей кодер може бути використаний для кодування вхідних послідовностей у вектор фіксованої довжини, що може бути інтегровано у ELK Stack для подальшого аналізу, візуалізації та виявлення аномалій в часових рядах.

Інтеграція LSTM з ELK Stack може відбуватися за допомогою декількох кроків:

- Збір та підготовка даних: Спочатку дані, з якими ви плануєте працювати, мають бути зібрані та підготовлені для аналізу. Вони можуть бути зібрані з журналів, баз даних або будь-яких інших джерел, а потім оброблені та перетворені в формат, який може бути використаний LSTM.
- Навчання моделі LSTM: Після того, як дані готові, модель LSTM повинна бути навчена на цих даних. Це включає в себе визначення архітектури мережі, вибір гіперпараметрів та навчання моделі на тренувальних даних.
- Прогнозування аномалій з використанням LSTM: Після навчання моделі LSTM вона може бути використана для прогнозування майбутніх значень часового ряду. Значення, які значно відрізняються від прогнозованих

значень, можуть бути визначені як потенційні аномалії.

– Інтеграція з ELK Stack: Прогнозовані значення, а також будь-які відхилення, які можуть бути виявлені як аномалії, можуть бути інтегровані в ELK Stack. Для цього може бути використана Logstash для збору та обробки даних, а потім дані можуть бути індексовані у Elasticsearch для подальшого аналізу та візуалізації в Kibana.
– Відображення результатів: У Kibana можна створити візуалізації та панелі, які демонструють прогнозовані значення, а також будь-які виявлені аномалії. Це дозволяє операторам аналізувати дані та виявляти аномальні відхилення в часових рядах.

Ці методи допомагають виявляти потенційно небезпечні дії або події у системі, використовуючи різні стратегії та підходи до аналізу даних (рис.1.6).

Рис. 1.6 – Процес аналізу та виявлення загроз у SIEM ELK STACK, включаючи збір, збереження, аналіз даних та їх візуалізацію

1.4. Висновок до розділу 1

У вищенаведеному огляді виявлення аномалій у часових рядах розглядається з різних перспектив, що відображає різноманітність методів та підходів до цієї задачі. Від методів, заснованих на прогнозуванні, до підходів, що використовують реконструкцію та кодування, різноманіття стратегій та підходів свідчить про важливість гнучкості в аналізі даних.

Кожен із зазначених підходів має свої переваги та обмеження, залежно від характеристик даних та специфіки завдання. Розуміння цих різноманітних підходів дозволяє вибрати оптимальну стратегію для конкретного випадку та досягти ефективного виявлення аномалій у часових рядах, що має велике значення для різноманітних областей, від фінансів до виробництва та бізнес-аналітики.

Інтеграція моделі LSTM з ELK Stack відкриває нові можливості для аналізу та виявлення аномалій у часових рядах. Цей процес починається зі збору та підготовки даних, йде через навчання моделі та прогнозування аномалій з використанням LSTM, і завершується інтеграцією з ELK Stack для подальшого аналізу та візуалізації результатів. Це сприяє покращенню розуміння динаміки даних і реагуванню на незвичайні події, що може мати важливе значення в широкому спектрі галузей, від моніторингу мережі до прогнозування фінансових тенденцій.

2. АНАЛІТИЧНИЙ ОГЛЯД ТЕХНОЛОГІЇ SIEM ELK STACK ДЛЯ РОЗРОБКИ АЛГОРИТМІВ ВИЯВЛЕННЯ АНОМАЛІЙ

2.1 Огляд технології SIEM ELK Stack

Стек ELK починався як колекція з трьох продуктів з відкритим вихідним кодом - Elasticsearch, Logstash і Kibana - розроблених, керованих і підтримуваних компанією Elastic. Впровадження і подальше додавання Beats перетворило стек на чотириногий проект.

Elasticsearch - це повнотекстова пошукова та аналітична система, заснована на пошуковій системі з відкритим вихідним кодом Apache Lucene.

Logstash - це агрегатор журналів, який збирає дані з різних вхідних джерел, виконує різні перетворення і поліпшення, а потім відправляє дані в різні підтримувані пункти призначення. Важливо знати, що багато сучасних реалізацій ELK не включають Logstash. Щоб замінити його можливості обробки логів, більшість звертаються до легких альтернатив, таких як Fluentd, який також може збирати логи з джерел даних і пересилати їх до Elasticsearch.

Основні можливості Elasticsearch:

1. Повнотекстовий пошук.

Elasticsearch використовує обертальний індекс, що дозволяє ефективно здійснювати повнотекстовий пошук у великих обсягах документів.

Це означає, що користувачі можуть швидко знаходити документи, які містять конкретні слова чи фрази.

2. Розподіленість.

Elasticsearch побудований на архітектурі розподіленої системи, що дозволяє масштабувати його горизонтально.

Дані автоматично розподіляються між вузлами кластера, що забезпечує надійність та масштабованість системи.

3. Мультизапити.

Elasticsearch підтримує різні типи запитів, такі як пошук, агрегація, фільтрація, сортування тощо.

Це дає розробникам можливість працювати з даними у багатьох варіантах і отримувати потрібні результати.

4. Агрегація та аналітика.

Elasticsearch надає широкий набір можливостей для агрегації даних та виконання аналітики.

Це включає групування, обчислення статистичних показників, підсумкові розрахунки тощо, що дозволяє здійснювати різноманітний аналіз даних.

Швидкість:

Завдяки оптимізованому процесу індексування та пошуку даних, Elasticsearch забезпечує високу швидкість обробки запитів.

Це дозволяє користувачам отримувати швидкий доступ до даних, навіть у великих обсягах.

5. Модульність та розширюваність.

Elasticsearch підтримує використання різноманітних плагінів та модулів, які розширюють його функціональність.

Elasticsearch - це система, що забезпечує горизонтально масштабований, багатопотоковий пошук. Вона базується на Apache Lucene і призначена для індексації та пошуку різних типів даних. Elasticsearch надає доступ до всіх функцій Lucene через JSON і Java API, що робить його вкрай гнучким та дозволяє інтегруватися з різноманітними системами.

Elasticsearch може обробляти GET-запити в режимі реального часу і легко інтегрується з Kibana для зручного керування через HTTP-інтерфейси. Його архітектура дозволяє створювати кластери з кількох екземплярів Elasticsearch для масштабування систем великих даних.

Дані в Elasticsearch можуть бути розділені на сегменти, кожен з яких реплікується для забезпечення надійності. Вузол кластера може брати участь як координатор, маршрутизуючи операції на відповідні сегменти з автоматичним

перебалансуванням. Зв'язані дані зазвичай зберігаються в одному індексі, який може містити декілька первинних сегментів та їх репліки.

Індекс може бути довготривалим збереженням завдяки шлюзу, який дозволяє відновлювати дані в разі відмови сервера.

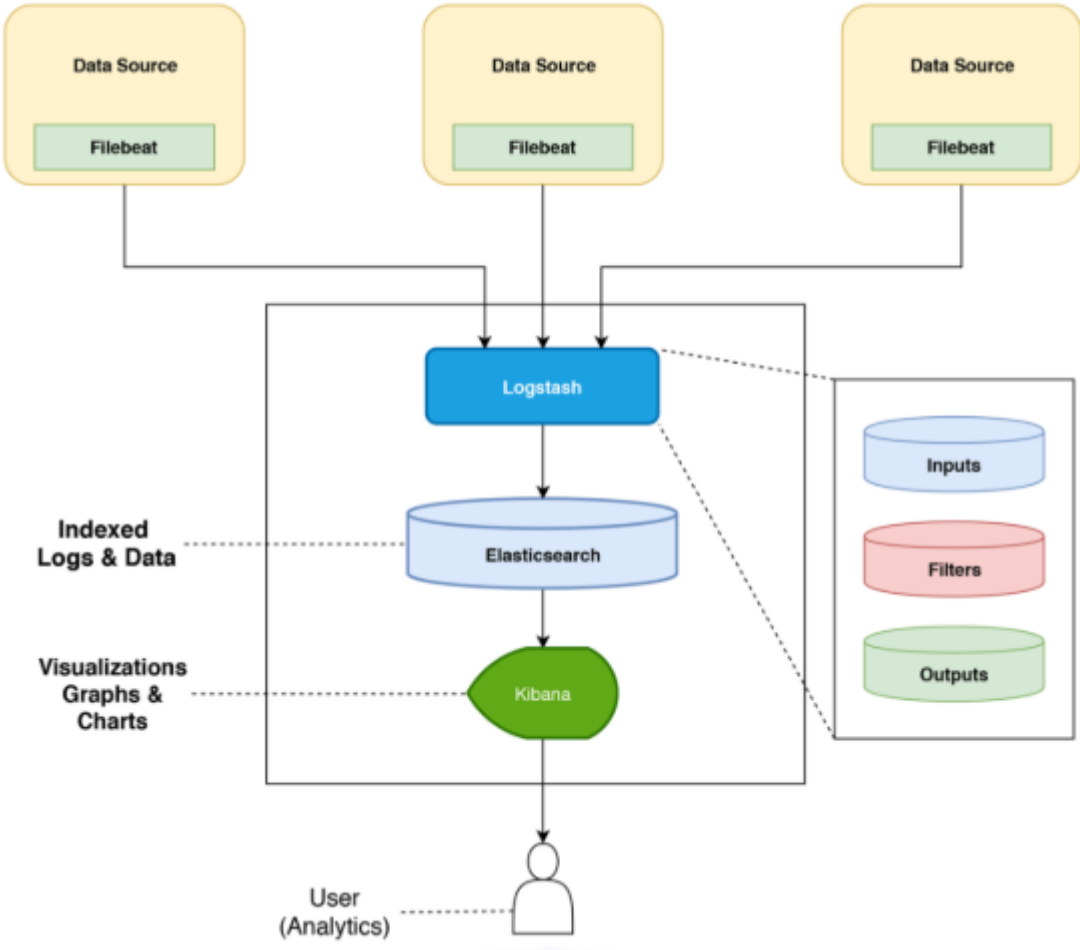


Рис. 2.1 Загальна архітектура ELK

ELK стек, що включає Elasticsearch, Logstash і Kibana, це набір інструментів з відкритим вихідним кодом, призначений для збору, обробки, аналізу та візуалізації даних у реальному часі з різноманітних джерел і форматів. Elasticsearch, як основа стеку, є розподіленою системою пошуку і аналізу, побудованою на базі Apache Lucene.

Logstash виступає як засіб для збору, обробки і передачі даних, забезпечуючи масштабованість, надійність та простоту управління. Він інтегрується з різноманітними джерелами даних і форматами, оброблюючи різні

типи журналів, подій і неструктурованих даних перед їх відправленням до Elasticsearch для подальшого аналізу.

Kibana - це платформа для візуалізації даних, яка дозволяє створювати потужні графіки та інформаційні панелі. Вона допомагає оживити дані за допомогою різних візуальних ефектів, від гістограм до географічних карт, що дозволяє аналізувати інформацію і отримувати з неї цінні інсайти. [13].

Стек ELK функціонує на принципі інтеграції окремих компонентів у єдиний конвеєр даних, як показано на рисунку 2.2.

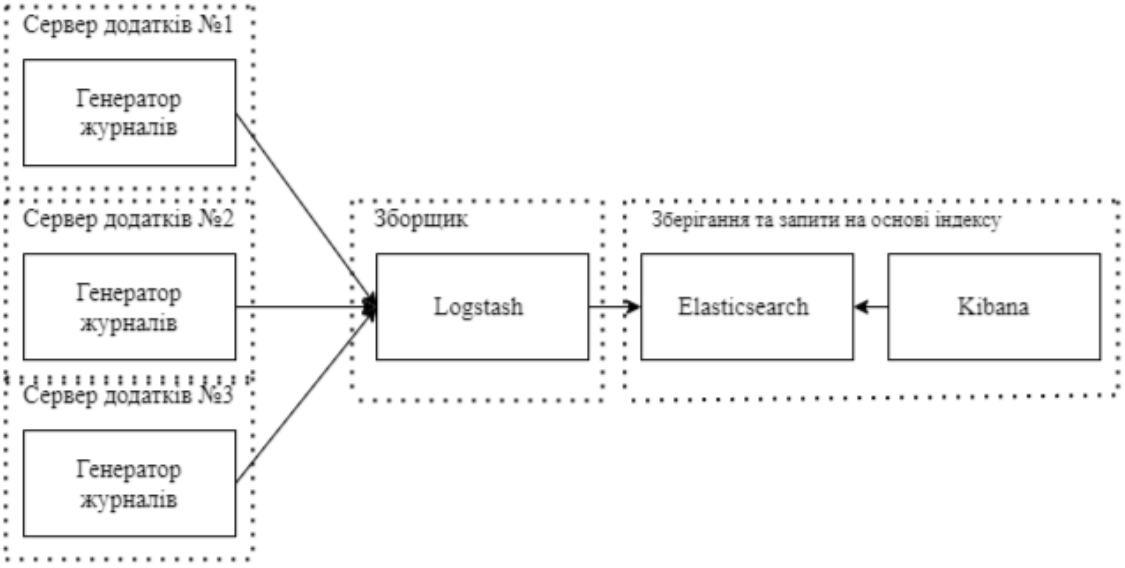


Рисунок 2.2 – Приклад конвеєра даних ELK

Кожна програма починає свою роботу, запускаючи агента Logstash, який відправляє журнали на центральний сервер Logstash, відомий як індексатор. Використання агента на кінцевій програмі дозволяє уникнути необхідності встановлення повної версії Logstash на кожному вузлі. Центральний індексатор Logstash може отримувати логи від кількох додатків одночасно. Потім ці логи передаються до кластера Elasticsearch, де їх можна запитувати за допомогою Kibana для створення візуалізацій і дашбордів.

Elasticsearch — це потужна пошукова система, яка базується на Apache Lucene. Вона пропонує розподілену систему повнотекстового пошуку в режимі реального часу із RESTful API, що працює з JSON документами. Elasticsearch може використовуватися для повнотекстового або структурованого пошуку,

аналітики або їх поєднання. Реалізована на Java, вона має відкритий вихідний код і ліцензується під Apache 2.0. Однією з її ключових переваг є швидкий пошук за допомогою індексації тексту [14].

Існує низка пошукових систем, які можуть проводити пошук за часовими мітками або конкретними значеннями. Однак Elasticsearch вирізняється тим, що вона надає повнотекстовий пошук, враховує синоніми і оцінює релевантність документів. Крім цього, Elasticsearch дозволяє виконувати аналіз і агрегацію в режимі реального часу на тих самих даних, що дає їй перевагу перед іншими пошуковими системами.

Приклади використання:

- Пошук та рекомендації контенту: Elasticsearch допомагає Netflix забезпечити швидкий та ефективний пошук фільмів та серіалів для своїх користувачів. Крім того, він допомагає в аналізі вибору користувачів та їхніх переглядів для надання персоналізованих рекомендацій.
- Моніторинг та аналіз сервісу: Netflix використовує Elasticsearch для моніторингу та аналізу даних про використання платформи, що дозволяє їм вчасно виявляти проблеми та покращувати якість свого сервісу.
- Аналіз відгуків користувачів: Netflix використовує Elasticsearch для аналізу відгуків користувачів, коментарів та оцінок фільмів та серіалів. Це допомагає їм зрозуміти, як користувачі сприймають контент і як вони можуть покращити свою пропозицію.
- Пошук та аналіз даних про вміст: Netflix використовує Elasticsearch для пошуку та аналізу даних про вміст, що допомагає їм вирішувати, які фільми та серіали вони мають ліцензувати або створювати власні виробництва.
- Пошук та аналіз даних про користувачів: Elasticsearch допомагає Netflix аналізувати дані про користувачів, такі як їхні вподобання, звички перегляду та інші характеристики. Це дозволяє їм створювати більш ефективні стратегії залучення та утримання аудиторії.

Хоча приклади використання Elasticsearch великими корпораціями можуть здатися домінуючими, варто зауважити, що ця технологія активно використовується також стартапами та невеликими та середніми підприємствами.

Привабливість Elasticsearch полягає в його гнучкості: він може працювати як на невеликих ноутбуках, так і на сотнях серверів, що дає можливість масштабування для обробки навіть петабайтів даних.

Таким чином, Elasticsearch є відмінним вибором як для великих корпорацій, так і для менших компаній, що шукають потужний інструмент для пошуку, аналізу та управління своїми даними.

Розглянемо деякі з основних можливостей Elasticsearch:

1. Реальний час: Elasticsearch надає можливість отримувати та аналізувати дані у режимі реального часу. Це означає, що користувачі можуть отримувати оновлення даних миттєво, що важливо для багатьох випадків використання.

2. Розподіленість: Elasticsearch є дійсно розподіленою системою, яка може працювати на різних обсягах, починаючи від невеликих ноутбуків і закінчуючи тисячами вузлів. При цьому, система автоматично реорганізує та ребалансує дані при додаванні нових вузлів або виході з ладу існуючих.

3. RESTful інтерфейс: Elasticsearch надає зручний RESTful інтерфейс, який використовує JSON через HTTP. Це дозволяє легко взаємодіяти з Elasticsearch і виконувати різноманітні операції з даними.

4. Базований на Lucene: Elasticsearch побудований на основі Apache Lucene, що є потужним інструментом для пошуку та аналізу даних. Він доступний як відкрите програмне забезпечення з ліцензією Apache 2.0, що дозволяє користувачам використовувати його безкоштовно та вільно модифікувати його вихідний код.

Замість традиційних методів управління журналами, які можуть мати свої обмеження, є сенс перейти до сучасних рішень, таких як Logstash. Ця система надає відкритий фреймворк для збору, аналізу та зберігання журналів. Маючи

відкритий код, вона дозволяє динамічно обробляти дані з різних джерел, нормалізуючи їх за потрібними критеріями.

Logstash здатний обробляти різноманітні типи подій, використовуючи різноманітні плагіни для введення, фільтрації та виведення. Його розширювана архітектура дозволяє розробникам створювати плагіни для різних сфер застосування, включаючи мобільні пристрої, підключені автомобілі та медичні сенсори.

Один із головних компонентів екосистеми Logstash - це Elasticsearch, який дозволяє виконувати широкий спектр операцій зіставлення, агрегації та пошуку. Крім того, для візуалізації та аналізу даних, зібраних Logstash, можна використовувати Kibana - платформу з відкритим вихідним кодом, яка надає інтуїтивно зрозумілі інструменти для створення графіків, гістограм, таблиць та карт. Її зручний інтерфейс у браузері дозволяє реалізувати глибокий аналіз даних та створювати динамічні інформаційні панелі для відображення змін у реальному часі.

Ключові особливості Kibana:

- Поєднання з Elasticsearch: Kibana легко візуалізує інформацію з Elasticsearch, незалежно від того, з якого джерела вона походить.
- Широкий вибір візуалізаційних форматів: Від гістограм до кругових діаграм, Kibana пропонує різноманіття форматів для кращого уявлення даних.
- Аналітичні можливості: Інтеграція з аналітичними засобами Elasticsearch дозволяє проводити глибокий аналіз даних з різних ракурсів.
- Простота використання: Гнучкий інтерфейс Kibana робить процес створення, зберігання, обміну, експорту та вбудовування візуалізованих даних легким та зручним для користувачів. [15].

2.2 Приклад використання SIEM ELK Stack для виявлення аномалій

ELK Stack вважається одним з найпоширеніших інструментів для аналізу логів, завдяки його надійності та масштабованості в зборі, зберіганні та аналізі даних з різних джерел. Цей інструментальний комплекс має широке застосування: від розробки та моніторингу до забезпечення безпеки, відповідності, SEO та бізнес-аналітики.

Перед тим як встановлювати ELK Stack, необхідно чітко визначити конкретні потреби та сценарії його використання. Це впливає на багато аспектів, включаючи місце та спосіб встановлення, налаштування кластера Elasticsearch, розподіл ресурсів, створення конвеєра даних та забезпечення безпеки системи.

Логи грають ключову роль у кризових ситуаціях, надаючи важливу інформацію про помилки та винятки. Також вони корисні на ранніх етапах розробки, коли ведення журналів з першої функції програми і протягом усього життєвого циклу полегшує діагностику та усунення несправностей.

ELK Stack, незалежно від того, чи це монолітна система чи мікросервісна архітектура, спочатку впроваджується як інструмент для розробників. Це допомагає корелювати, виявляти та усувати помилки ще до того, як код потрапляє виробництво. Журнальні повідомлення збираються різними додатками, фреймворками, бібліотеками та диспетчерами, а потім централізовано передаються до ELK Stack для подальшого аналізу та управління.

Після впровадження виробництва Kibana використовується для моніторингу загального стану програми та окремих сервісів. Це значно спрощує процес аналізу та усунення несправностей, оскільки всі журнальні дані доступні в одному місці.

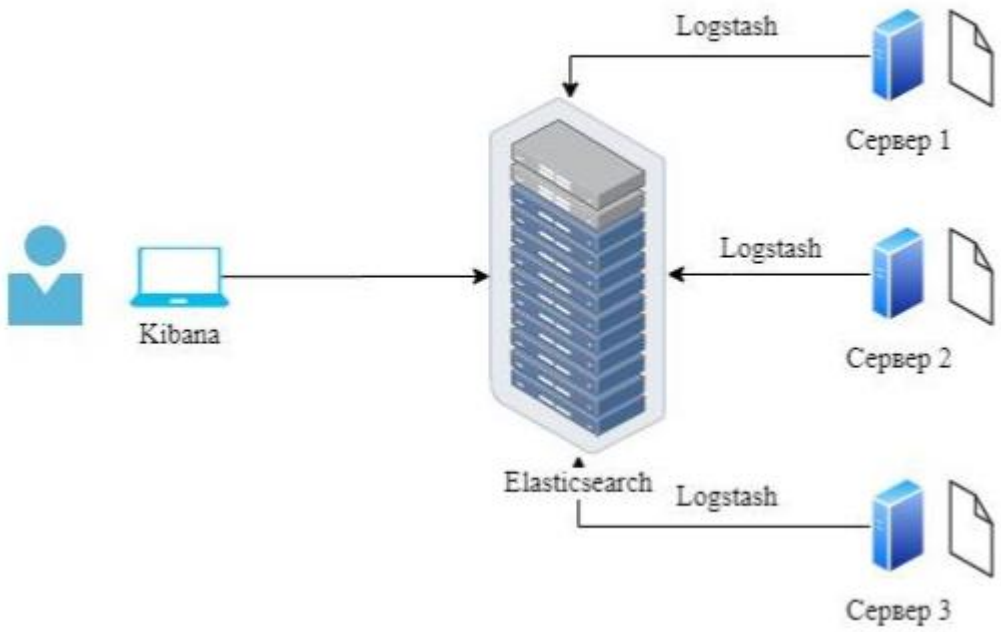


Рисунок 2.3 Загальна структура використання ELK

Сучасне IT-середовище з багаторівневою розподіленою архітектурою є серйозною проблемою для груп, відповідальних за операції та моніторинг. Моніторинг усіх різних систем та компонентів, що складають архітектуру програмного забезпечення, є складним завданням, що вимагає значних ресурсів та часу.

Завдання, що стоять перед командою, включають доступ до окремих машинок, збір та обробку даних, додавання контексту, зберігання та аналіз даних, а також забезпечення безпеки та резервного копіювання.

ELK stack надає організаціям універсальні рішення, надаючи інструменти для вирішення цих проблем. Ви можете розгорнути агент Beats на своєму комп'ютері та імпортувати журнали в Logstash. Logstash налаштований для збору та обробки даних перед індексуванням у elasticsearch. Потім Кібана використовується для аналізу даних, виявлення аномалій, аналізу першопричин та створення інформаційних панелей.

Elasticsearch спочатку був розроблений для повнотекстового пошуку та аналізу і все частіше використовується для аналізу показників Дека. Моніторинг показників продуктивності для кожного компонента архітектури є ключем до

наочності операцій. Ці показники можуть бути змінені за допомогою сторонніх агентів аудиту та моніторингу або за допомогою існуючих бітів (Metricbeat, Packetbeat тощо).) можуть бути зібрані з використанням.).). Kibana пропонує нові типи візуалізації для аналізу часових рядів, включаючи Timelion та Visual Builder.

Моніторинг продуктивності додатків (APM) є одним із найпоширеніших способів вимірювання доступності, часу відгуку та поведінки додатків та служб.Побудований на основі 1 стека ELK, Elastic APM дозволяє відстежувати ключові показники продуктивності, такі як запити, відповіді, операції з базою даних і помилки.

Інтегруючи інструменти розподіленого моніторингу з відкритим кодом, такі як Zipkin та Jaeger, з ELK, ви можете глибше проаналізувати продуктивність додатків.

Безпека завжди була важливою для організації. Збільшення кількості атак і вимог відповідності (HIPAA, PCI, SOC, FISMA і т.д.)) через це безпека в останні роки стала ще більш пріоритетним завданням.).). Безпека стала важливим аспектом використання стека ELK, оскільки журнали містять багато цінної інформації про поточні події. Навіть без вбудованих функцій безпеки здатність elk централізовано зберігати та відстежувати журнали, а також створювати інформаційні панелі, орієнтовані на безпеку, стала популярним інструментом для інтеграції з основними стандартами безпеки.

1. У випадку виявлення DDoS-атаки, кожна секунда має величезне значення. Швидка ідентифікація є ключем до мінімізації збитків, і саме тут моніторинг логів відіграє важливу роль. Журнали містять багато інформації про поточні події, оскільки вони реєструють дії запущених процесів.

2. Використовуючи ELK, організації можуть створювати системи, які збирають дані з різних рівнів IT-середовища (наприклад, веб-серверів, баз даних, брандмауерів), щоб полегшити аналіз та візуалізацію цих даних на потужних дашбордах SIEM.

SIEM - це підхід до управління корпоративною безпекою, спрямований на створення цілісного уявлення про IT-безпеку організації. Основна мета SIEM полягає в наданні одночасного та всебічного уявлення про IT-безпеку, включаючи легку ідентифікацію дій, тенденцій і шаблонів. Використання стеку ELK може сприяти реалізації SIEM, забезпечуючи інтегровану інформаційну панель, яка дозволяє проактивно виявляти загрози, відстежувати онлайн-активність та надавати звіти про реагування на інциденти [16].

Бізнес-аналітика (BI) - це процес аналізу вихідних даних організації за допомогою програмного забезпечення, інструментів і додатків з метою оптимізації процесу прийняття рішень, покращення співпраці та підвищення загальної ефективності. Стек ELK використовується для агрегації та аналізу даних з різних джерел у центральному місці, забезпечуючи можливість аналізу різноманітних даних, включаючи журнали доступу до веб-сервера та дані CRM-системи.Існує багато пропрієтарних інструментів, що використовуються для цих цілей. Проте стек ELK, який має відкритий вихідний код, може виконувати більшість функцій цих інструментів за більш доступну ціну [17].

Технічне SEO є ще одним способом використання стеку ELK, що має велике значення. Основний зв'язок між SEO та ELK полягає у використанні логів.

Логи доступу до веб-сервера (наприклад, Apache, nginx, IIS) надають інформацію про те, хто саме робить запити до веб-сайту, включаючи запити ботів, які належать пошуковим системам і сканують сайт. SEO-фахівці використовують ці дані для визначення кількості запитів, здійснених Baidu, BingBot, GoogleBot тощо, та відстеження активності різних ботів.

Технічні спеціалісти SEO використовують дані журналів для відстеження дати та часу останнього сканування сайту ботами, оптимізації бюджетів на сканування, виявлення помилок і перенаправлень на сайті, визначення пріоритетів сканування, проведення повторних сканувань тощо.

Ми розробляємо алгоритм для виявлення аномальної активності на основі стеку ELK, який працюватиме за такою схемою, як показано на рисунку 2.4.

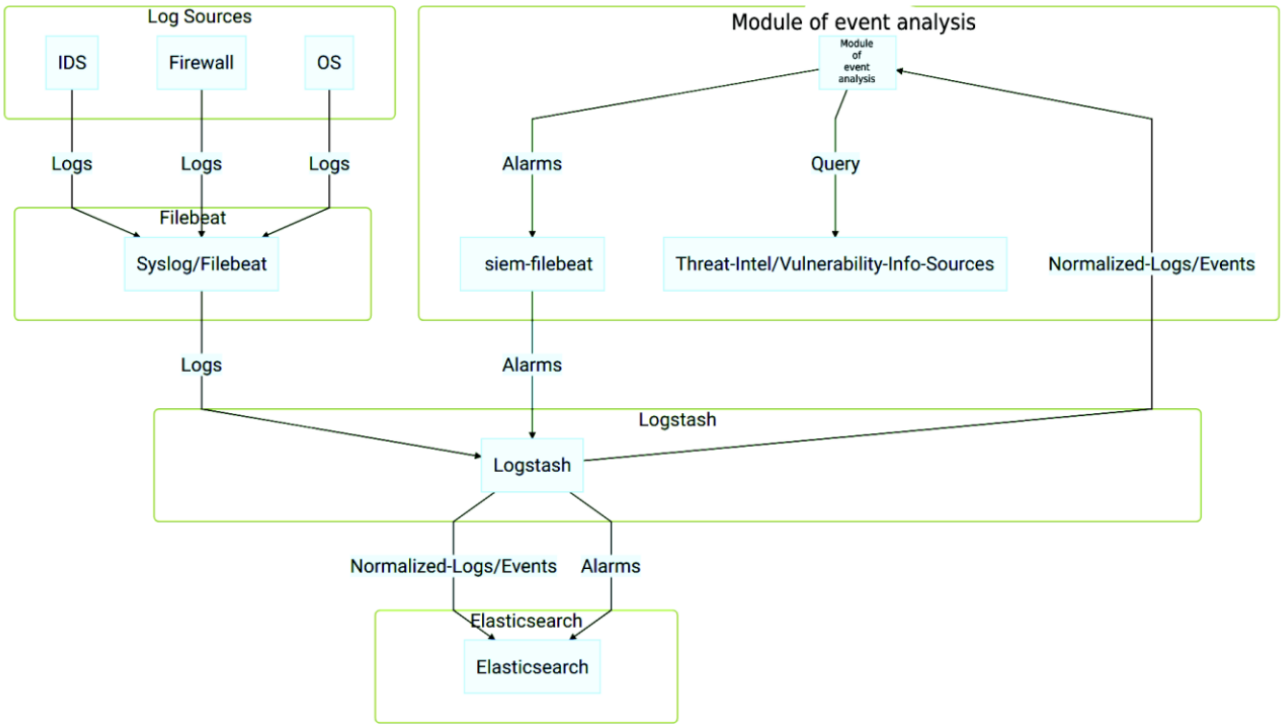


Рис. 2.4 - Алгоритм роботи методу виявлення аномальної активності

На діаграмі вище видно наступне:

1. Джерела журналів передають свої журнали на Syslog / Filebeat, який подальшим чином направляє їх до Logstash з унікальним ідентифікаційним полем. Logstash аналізує журнали за допомогою різних фільтрів, враховуючи тип джерела журналів, та надсилає результати до Elasticsearch, створюючи зазвичай єдиний шаблон індексу для кожного типу журналу (наприклад, ids-* для журналів з IDS, ssh-* для журналів SSH тощо).

2. Метод виявлення аномальної активності використовує спеціальний конфігураційний файл logstash для клонування вхідної події з джерела журналу після аналізу logstash. Нова клонована подія використовується для збору полів, необхідних для модуля аналізу подій ІБ, таких як назва, джерело IP, IP адреса призначення тощо.

3. Вихідні дані цього кроку називаються нормалізованою подією, оскільки вони представляють журнали з різних джерел в одному форматі з загальними полями. Ці події надсилаються до методу виявлення аномальної активності через

вихідний плагін Logstash HTTP та до Elasticsearch під шаблоном імені індексу siem_events.

4. Метод виявлення аномальної активності корелює вхідні нормалізовані події на основі налаштованих правил кореляції подій, виконує пошук інформації про загрози та вразливості, і генерує сигнал загрози в Alarm відповідно до умов правил. Потім цей сигнал записується в локальний файл журналу, який збирається локальним Filebeat, налаштованим на відправлення його вмісту в Logstash.

5. У кінці logstash є ще один спеціальний конфігураційний файл для методу виявлення аномальної активності, який зчитує подані сигнали та надсилає їх до остаточного індексу загроз ІБ в Elasticsearch.

2.3 Переваги та обмеження використання SIEM ELK Stack для виявлення аномалій

Інформація про безпеку і управління подіями (SIEM) включають в себе різні технології та послуги:

1. Система управління журналами (LMS) діє як Центральне сховище для збору та зберігання файлів журналів з різних хостів та систем, що полегшує спрощений доступ до журналів.
2. Системи SLM / SEM (ведення журналу безпеки / управління інцидентами) активно відстежують, аналізують і візуалізують дані, видаючи попередження, що виявляють потенційні загрози безпеці.
3. Система SIM (Security Information Management) спеціалізується на зборі та управлінні даними безпеки з різних джерел, щоб допомогти вам виявляти та реагувати на великі загрози.
4. Система SEC (кореляція подій безпеки) допомагає ідентифікувати та співвідносити події безпеки, які потребують подальшого розслідування та дій.

По суті, технологія SIEM легко інтегрує ці функції. Він збирає та обробляє дані з різних джерел самостійно, централізує сховище, визначає взаємозв'язки між подіями та створює ефективні попередження та звіти на основі цих прогнозів декомунізації.

Основними особливостями технології SIEM є:

1. Збір даних: Це включає збір та об'єднання журналів даних з різних джерел для централізованого моніторингу.

2. Кореляція: si декомунізація допомагає встановлювати зв'язки і виявляти складні взаємозв'язки між подіями з різних джерел.

3. Оповіщення: автоматичний аналіз і генерація попереджень про проблеми дозволяють миттєво реагувати на потенційні загрози.

4. Інформаційна Панель: інструменти візуалізації даних та виявлення шаблонів допомагають зрозуміти стан вашої системи.

5. Сумісність: можливість взаємодії з іншими системами і додатками спрощує інтеграцію і обмін даними.

6. Зберігання: SIEM забезпечує довгострокове зберігання історичних даних для подальшого аналізу та досліджень.

7. Судово-медична експертиза: можливість швидкого пошуку журналів за різними критеріями для розслідування інцид декомунізації та виявлення порушень безпеки.

Система SIEM надає детальну картину стану мережі, знижує ризики, відповідає стандартам безпеки і підвищує продуктивність фахівців з інформаційної безпеки.

Однак для ефективної роботи SIEM необхідно звернути увагу на процеси проектування та складання. Навіть найкращі системи спостереження не завжди можуть ефективно виявляти справжні розширені атаки (APT).

Використання стека SIEM ELK для виявлення аномалій має свої обмеження:

1. Складність встановлення: Встановлення та налаштування стека Elk є складним завданням для організацій, які не мають достатнього досвіду моніторингу та аналізу безпеки.

2. Вимоги до ресурсів: Стек ELK вимагає великих обчислювальних та мережевих ресурсів і є проблематичним для організацій з обмеженими IT-ресурсами.

3. Необхідність навчання персоналу: використання Elk Stack для виявлення аномалій вимагає кваліфікованого персоналу, здатного точно аналізувати дані та реагувати на виявлені загрози.

4. Можливі помилк: неправильна конфігурація або інтерпретація даних може призвести до неправильного виявлення або обходу реальних загроз, що знижує ефективність системи виявлення аномалій.

2.4 Розроблення та класифікація логістичної регресії

Дійсно, назва "логістична регресія" (LR) походить від використання логістичної або сигмоїдної функції, яка є основною частиною цього методу. Логістична функція (рис. 2.1), яка приймає будь-яке значення між 0 та 1, але ніколи не досягає або не перевищує цих меж, була спочатку розроблена для опису росту популяції в екології. Її форма нагадує літеру "S", що стала базою для використання у моделюванні й прогнозуванні в різних областях, зокрема в машинному навчанні. Ця функція в LR використовується для призначення ймовірностей класів, що дозволяє проводити класифікацію даних.

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

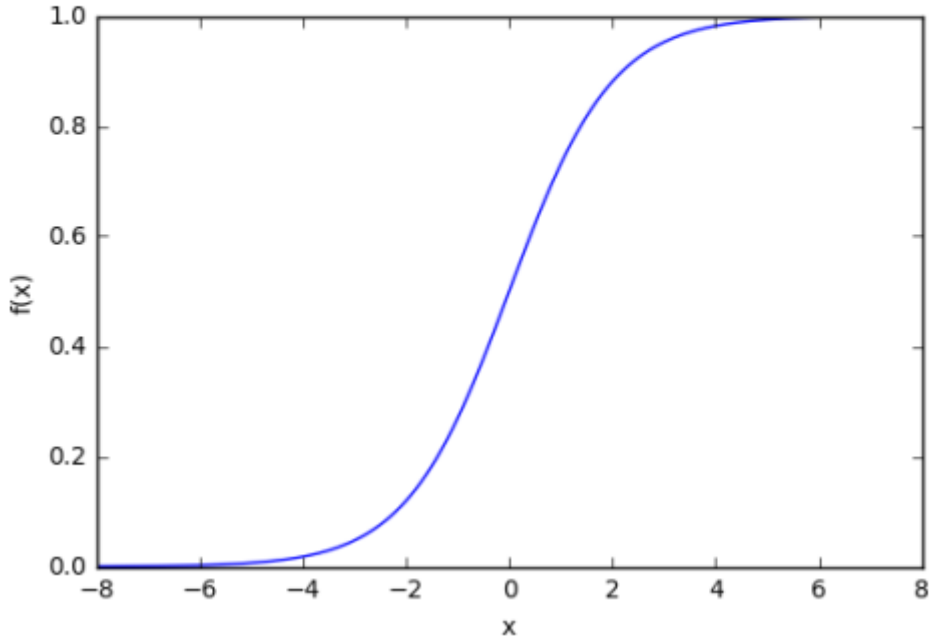


Рис 2.1 – Логістична або сігмоїдна функція

Точно, "логістична регресія" (LR) хоч і має в назві слово "регресія", але фактично це класифікаційна модель, а не модель регресії. Вона використовує логістичну функцію для кадрування бінарної вихідної моделі, де значення функції виражає ймовірність того, що даний запис належить до певного класу. Хоч LR подібна до лінійної регресії, вона не прогнозує конкретне числове значення

виходу, а замість цього використовується для класифікації даних за ймовірністю належності до певних класів.

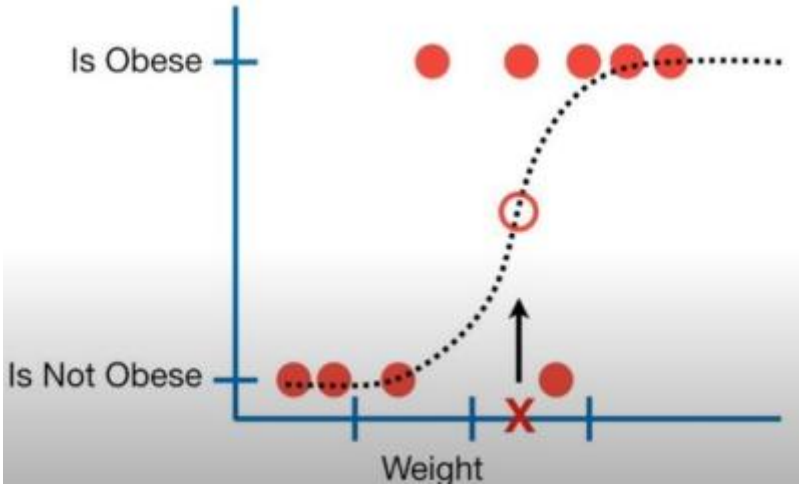


Рис 2.2- Ймовірність приналежності до класу

Точно, логістична регресія (LR) може ефективно працювати з якісними та кількісними даними, такими як генотипи та вік, здатна визначати вплив кожної змінної на прогнозування результату. Це дає можливість визначити, які змінні є корисними для передбачення та які можна виключити. Наприклад, астрологічний знак може бути непотрібною змінною, яка не впливає на результат.

Те, що LR може розрізняти вплив кожної змінної на передбачення, робить його популярним методом машинного навчання. Його здатність працювати з безперервними та дискретними вимірами дозволяє використовувати його в різноманітних сферах, від медицини до фінансів, забезпечуючи точні прогнози та класифікацію нових даних.

Метод опорних векторів

Точно, метод опорних векторів (SVM) є одним з найпоширеніших методів навчання з учителем, особливо у завданнях класифікації та регресії. Він представляє екземпляри даних як набір точок у N-розмірному просторі та створює (N - 1)-розмірний гіперплан, що допомагає ефективно розділити ці точки на дві групи.

Процес навчання SVM зазвичай використовує алгоритм послідовної мінімальної оптимізації для пошуку оптимального гіперплану, який максимально відокремлює класи. Цей метод дає можливість досягнути кусково-лінійної апроксимації даних, що означає, що SVM може ефективно працювати навіть у складних задачах, де взаємозв'язки між класами не є строго лінійними.

$$[\,M < 0] \leq (1 - M) + \qquad (2.1)$$

Основна ідея роботи SVM:

- на початку дані у реально низькому вимірі;

- далі перенесимо дані у простори більш високої розмірності (регуляризація);

- і знаходимо SVC, який розділяє дані більш високих розмірностей на дві групи (навчання);

Для SVM використовуємо регуляризації із квадратичною нормою. Це є обов’язковою умовою.

Навчання виконується на основі спеціалізованих методів квадратичного програмування. Існують ще й нелінійні модифікації SVM. Їх лінійність чи нелінійність визначається типом ядра.

Метод k найближчих сусідів(k-NN)

k-NN - непараметричний метод, одна з найпростіших методик використання МН.

Метричний алгоритм класифікації відшукуємо в наступному вигляді:

$$\alpha(x; X^l) = \arg\max \sum_{i=1}^{l} [y^{(i)} = y]\, w(i, x)$$

ваговий коефіцієнт, він вказує на степінь важливості ітого сусіда об’єкта х,завжди є невід’ємним та по і не зростає.

$$\sum_{i=1}^{l} [y^{(i)} = y]\, w(i, x) -$$

це оцінка близькості об’єкта х до класу. Маємо метод найближчого сусіда.

Так, метод k-найближчих сусідів (k-NN) використовує просту логіку, що полягає у визначенні класу об'єкта на основі більшості його найближчих сусідів у просторі ознак. Основними гіперпараметрами методу є значення k - кількість сусідів, що беруться до уваги, та функція відстані, що визначає міру подібності між об'єктами.

Значення k вибирається з урахуванням помилки перевірки моделі, а функція відстані зазвичай визначається на основі конкретних властивостей даних. Наприклад, може використовуватися евклідова відстань для неперервних ознак, а Манхеттенська відстань - для категоріальних.

На діаграмі представлені навчальні дані класів A та B у вигляді жовтих та фіолетових точок відповідно. Червона зірка позначає тестові дані, які потрібно класифікувати. Застосування методу k-NN з параметрами k = 3 призводить до класифікації вихідного класу як B, тоді як при k = 6 об'єкт класифікується як клас A.

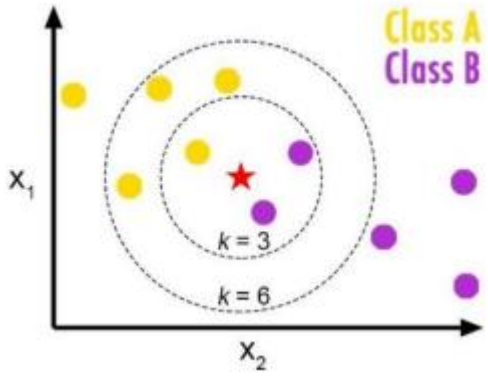


Рисунок 2.3 – Розподілення навчальних даних на діаграмі

Навчання в k-NN немає. Під час тестування k сусідів з мінімальною дистанцією візьмуть участь у класифікації.

Висновок до розділу 2

Проведено детальний аналіз можливостей, переваг та обмежень SIEM ELK Stack у виявленні аномалій у системі. Вивчаючи технологію SIEM ELK Stack, встановлено, що вона складається з Elasticsearch, Logstash та Kibana, що забезпечує повний цикл збору, обробки та візуалізації даних. Це дозволяє отримати повний огляд стану безпеки мережі та виявляти аномалії.

Представлений приклад практичного застосування SIEM ELK Stack для виявлення аномалій підкреслив можливості системи у виявленні несправностей та непередбачених змін у системі, що дозволяє оперативно реагувати на потенційні загрози.

Розглянуті не лише переваги, а й обмеження використання SIEM ELK Stack. Серед них складність налаштування, вимоги до ресурсів та потреба у кваліфікованому персоналі. Ці фактори можуть ускладнити впровадження системи та зменшити її ефективність у деяких сценаріях.

У висновку можна сказати, що SIEM ELK Stack є потужним інструментом для виявлення аномалій у системі, проте впровадження його вимагає ретельного аналізу індивідуальних потреб та можливостей організації. При належному налаштуванні та ефективному використанні, він може значно підвищити рівень безпеки та захисту мережі від потенційних загроз.

3. ПРАКТИЧНА РЕАЛІЗАЦІЯ АЛГОРИТМІВ ВИЯВЛЕННЯ АНОМАЛІЙ З ВИКОРИСТАННЯМ SIEM ELK STACK

3.1. Підготовка середовища SIEM ELK Stack та збір даних.

Репозиторій ***pfELK*** (Packetbeat, Filebeat, Elasticsearch, Logstash, Kibana) — це набір конфігурацій та скриптів, які допомагають налаштувати систему моніторингу логів та аналізу даних за допомогою ELK Stack. ELK Stack складається з Elasticsearch (пошук та аналіз даних), Logstash (збір, обробка та передача логів) і Kibana (інтерфейс для візуалізації та аналізу даних) [18].

pfELK спрощує розгортання та налаштування цих компонентів для аналізу мережевих логів, отриманих від систем моніторингу мережі, таких як Suricata, Bro або Zeek. Цей набір інструментів допомагає створити централізовану систему моніторингу, що дозволяє виявляти аномальну поведінку в мережі та системі.

Використання SIEM ELK Stack (Security Information and Event Management на основі ELK Stack) у вирішенні проблем з аномальною поведінкою в системі полягає в наступному:

1. Збір логів: Packetbeat та Filebeat використовуються для збору різноманітних логів з мережевих пристроїв, таких як файрволи, маршрутизатори, комутатори, інтрудерські виявлення, тощо. Ці логи потім передаються до Logstash для обробки.

2. Обробка логів: Logstash приймає дані від Filebeat і Packetbeat, обробляє їх (наприклад, структурує, фільтрує, збагачує) і передає до Elasticsearch для зберігання та аналізу.

3. Зберігання та аналіз: Elasticsearch використовується для зберігання даних у вигляді індексів, що дозволяє ефективно шукати та аналізувати інформацію. Кібана виступає як інтерфейс користувача, який надає зручний доступ до даних, можливості візуалізації та створення звітів.

4. Виявлення аномальної поведінки: Після встановлення та налаштування системи моніторингу, включаючи правила для виявлення аномальної поведінки,

такі як незвичні запити, підозрілі спроби вторгнення, аномальний трафік [19].

SIEM ELK Stack починає аналізувати дані в реальному часі. Він може виявляти відхилення від типового шаблону поведінки, що може свідчити про потенційні загрози або проблеми в системі.

Тепер розпочнемо всновлення ELK, для цього будемо слідувати наступному алгоритму.

– ***wget https://raw.githubusercontent.com/pfelk/pfelk/main/etc/pfelk/scripts/pfelk-installer.sh***

– ***chmod +x pfelk-installer.sh***

– ***./pfelk-installer.sh***

Під час інсталяції (рис. 3.1 – рис. 3.3) він буде запитувати різні варіанти, скрізь можна вказувати Yes, також він запитає базу даних GeoIP, рекомендується вказати MaxMind, для отримання ключа і пароля потрібно зареєструватися за посиланням (***https://www.maxmind.com/en/geolite2/signup***). Надалі, вказуємо скрізь Yes. В кінцевому підсумку має з'явитися повідомлення про те, що встановлення завершено.

Рис. 3.1. – Встановлення ELK Stack з репозиторію pfELK

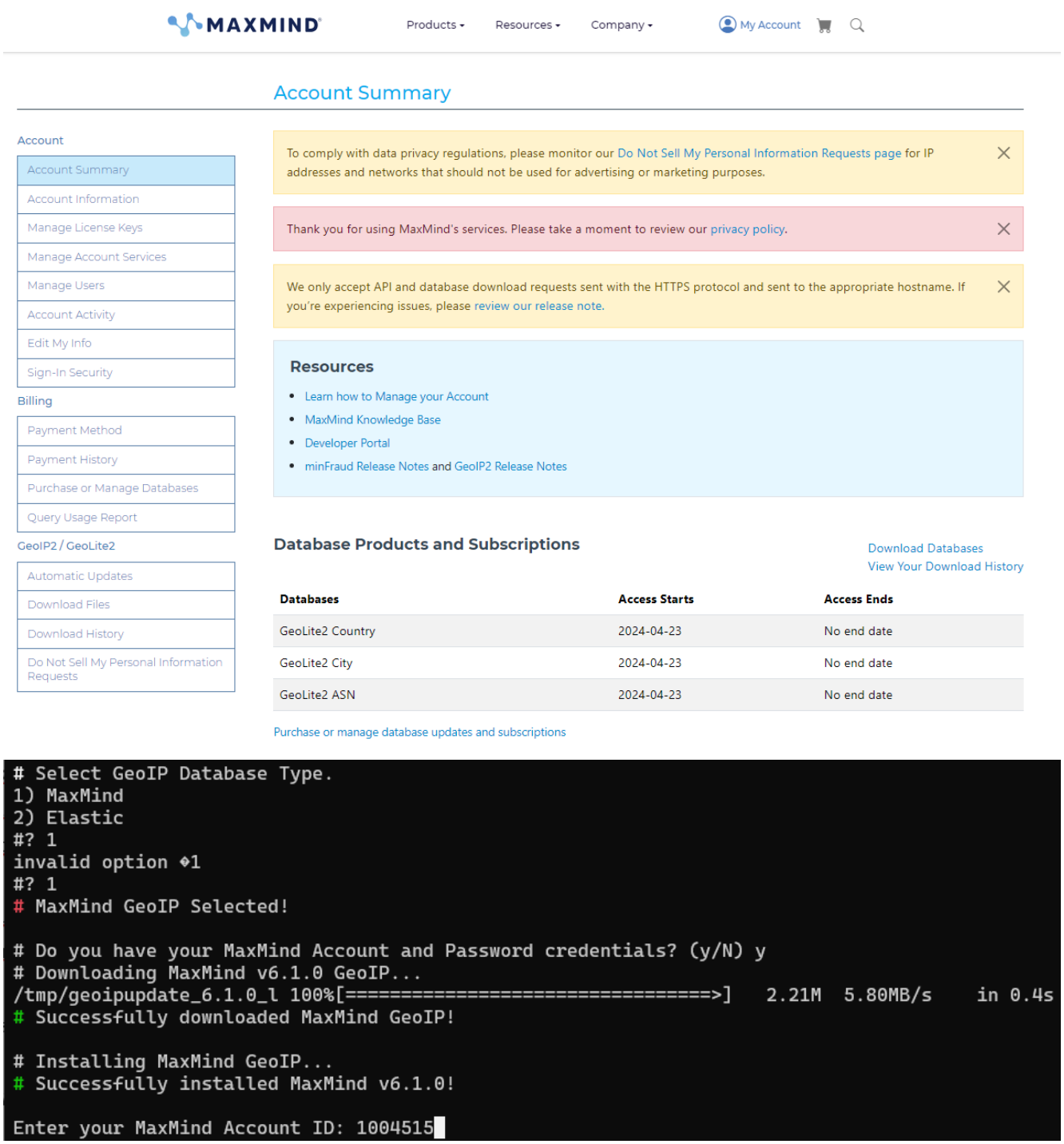


Рис. 3.2. – Реєстрація MaxMind

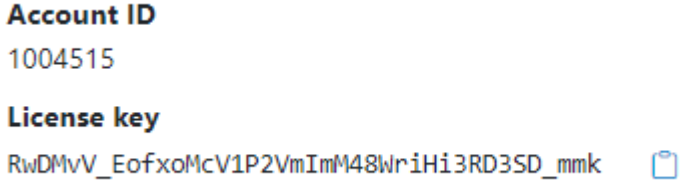


Рис. 3.3. – Отримання даних для генерації оточення

На рис 3.4 проведено інсталяцію необхідних для підтримки середовища демонів.

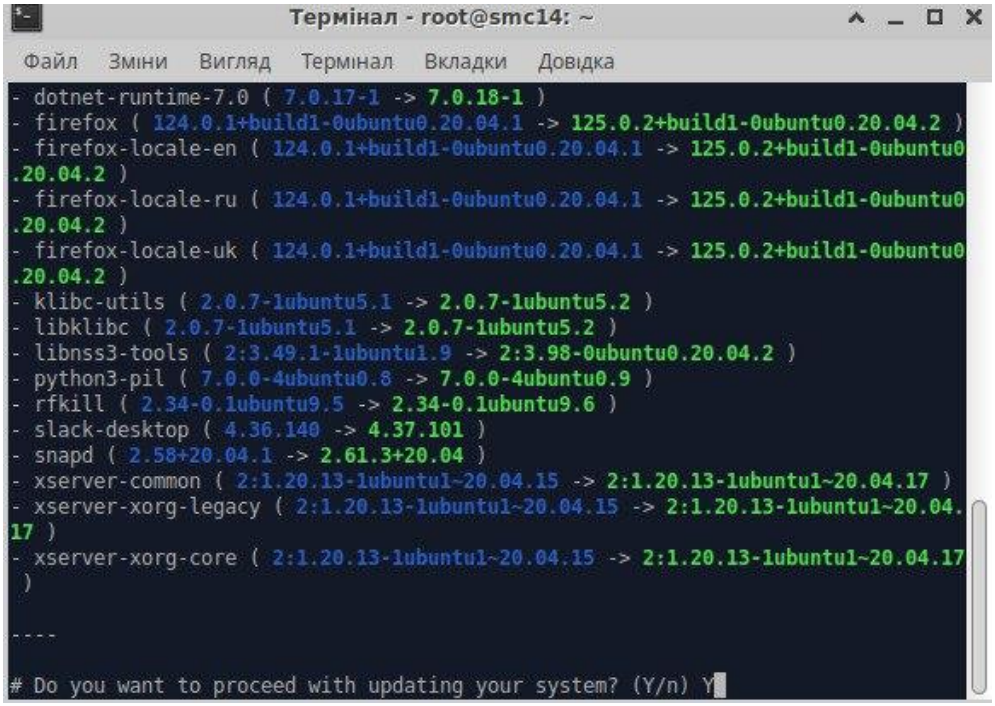


Рис. 3.4. – Інсталяція необхідних для підтримки середовища демонів

На рис 3.5 показано встановлення ліцензійного ключа MaxMind та оновлення пакетів

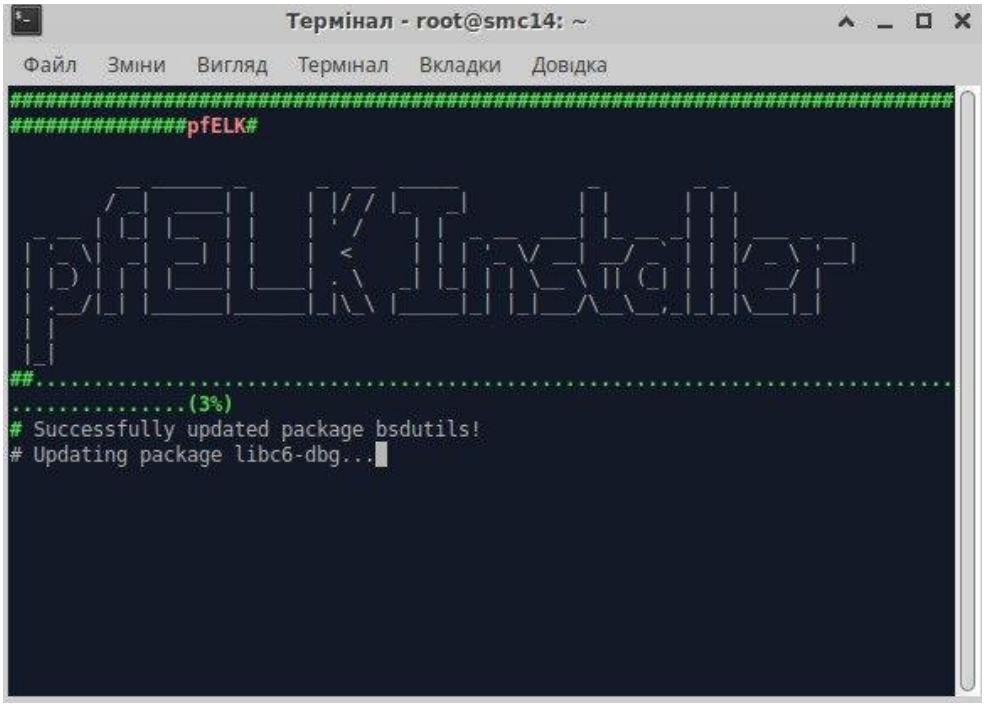


Рис. 3.5. – Встановлення ліцензійного ключа MaxMind та оновлення пакетів

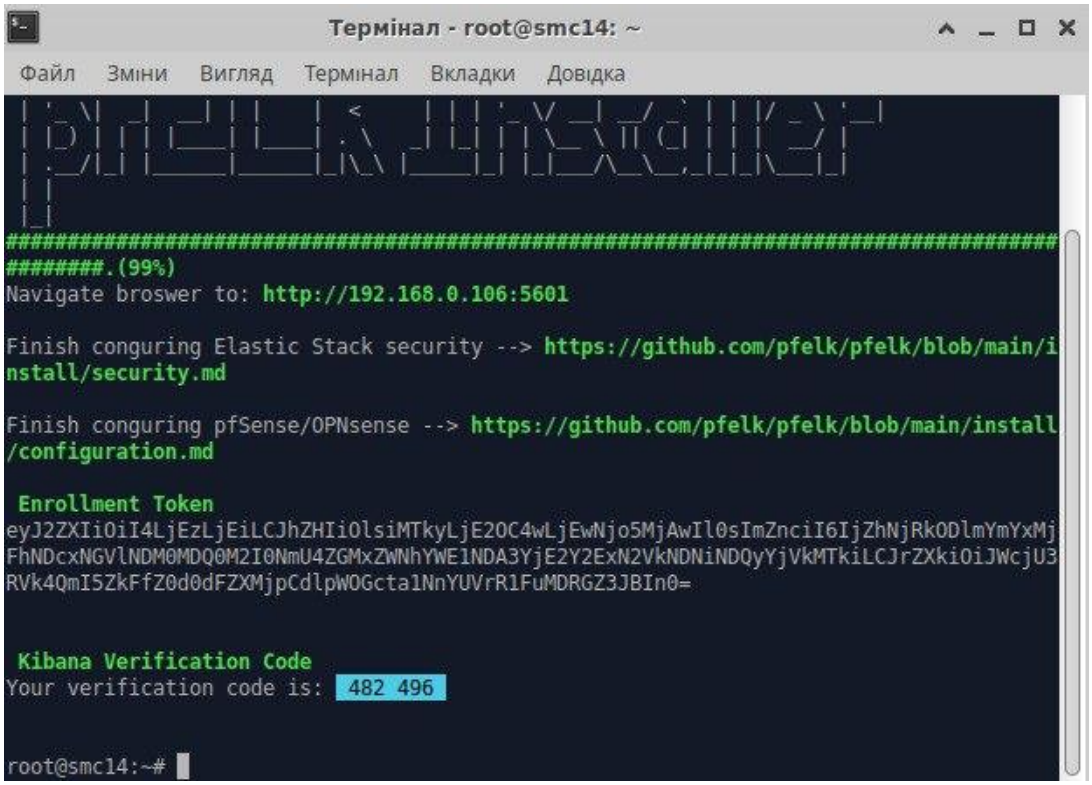


Рис. 3.6. – Інсталяція необхідних для підтримки середовища демонів

Наступним кроком буде завантаження pfelk-templates + dashboard.

Для коректного відображення потрібних нам даних з pfsense, suricata, snort тощо. Необхідно встановити templates. Для цього скористаємося такими командами:

- ***wgethttps://raw.githubusercontent.com/pfelk/pfelk/main/etc/pfelk/scripts/pfelk-template-installer.sh***
- ***chmod +x pfelk-template-installer.sh***
- ***./pfelk-template-installer.sh***

Темплейти повинні правильно завантажитись (рис. 3.7).

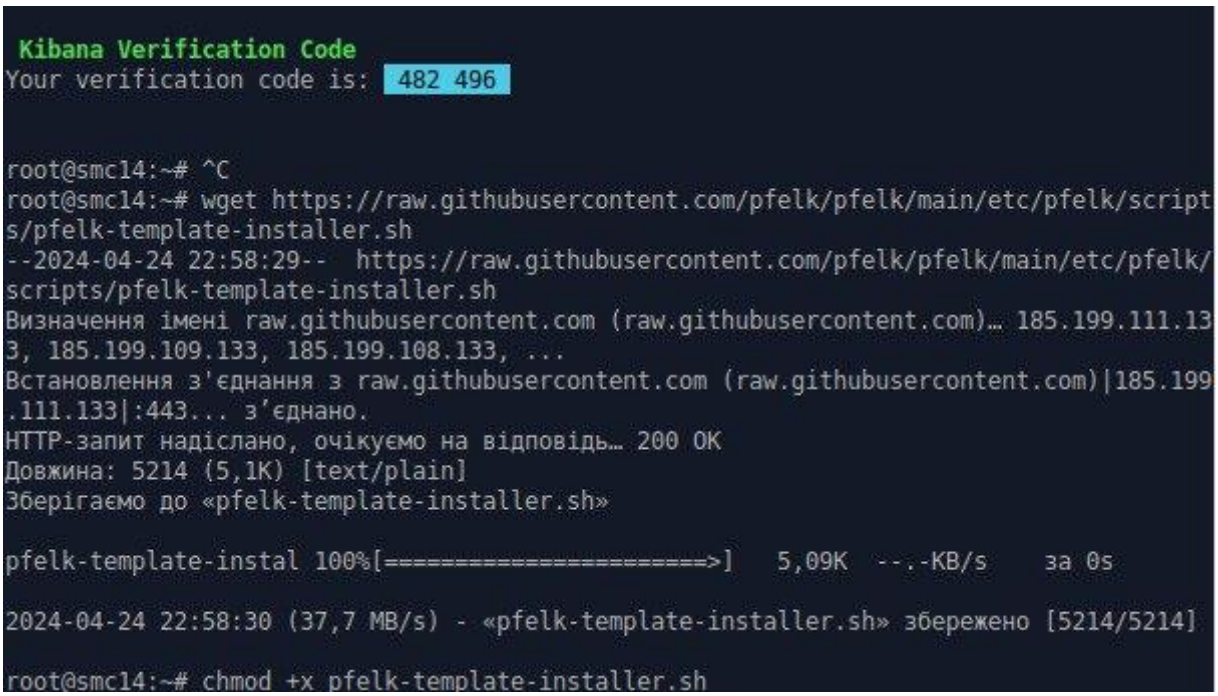


Рис. 3.7. – Завантаження темплейтів pfelk за посиланням wget

Далі встановлюємо дашборди:

- ***wget https://raw.githubusercontent.com/pfelk/pfelk/main/etc/pfelk/scripts/pfelk-dashboadrd-installer.sh***
- ***chmod +x pfelk-template-installer.sh***
- ***./pfelk-dashboard-installer.sh***

Після встановлення рекомендується перезавантажити машину. Після цього ви зможете перейти через браузер у кібану, де можна подивитися дшборди і темплейти (рис. 3.8).

Для переходу в кібану скористаємося посиланням: http://192.168.100.150:5601.

Запасний варіант завантажити образ і залити в EVE-NG: https://drive.google.com/u/0/uc?id=1Eiq51oZNQ_GEXRwskyMQwJS0M5I_dD_J&export=download Comment Suggest edit.

Передача логів pfsense, surica, snort в ELK. Для початку налаштуємо передачу логів власне самих pfsense, для цього заходимо в перший наш pfsense (той, на якому snort). Заходимо на Web-інтерфейс і там переходимо шляхом Status > System Logs > Settings. Гортаємо в кінець і клацаємо галочку в полі Send log messages to remote syslog service, далі вказуємо все як на скріншоті, але зверніть

увагу на ip і порт, їх потрібно вказувати свої, за замовчуванням, для подій від pfsense доступні порти 5140 і 5141, можна вказувати будь-який.

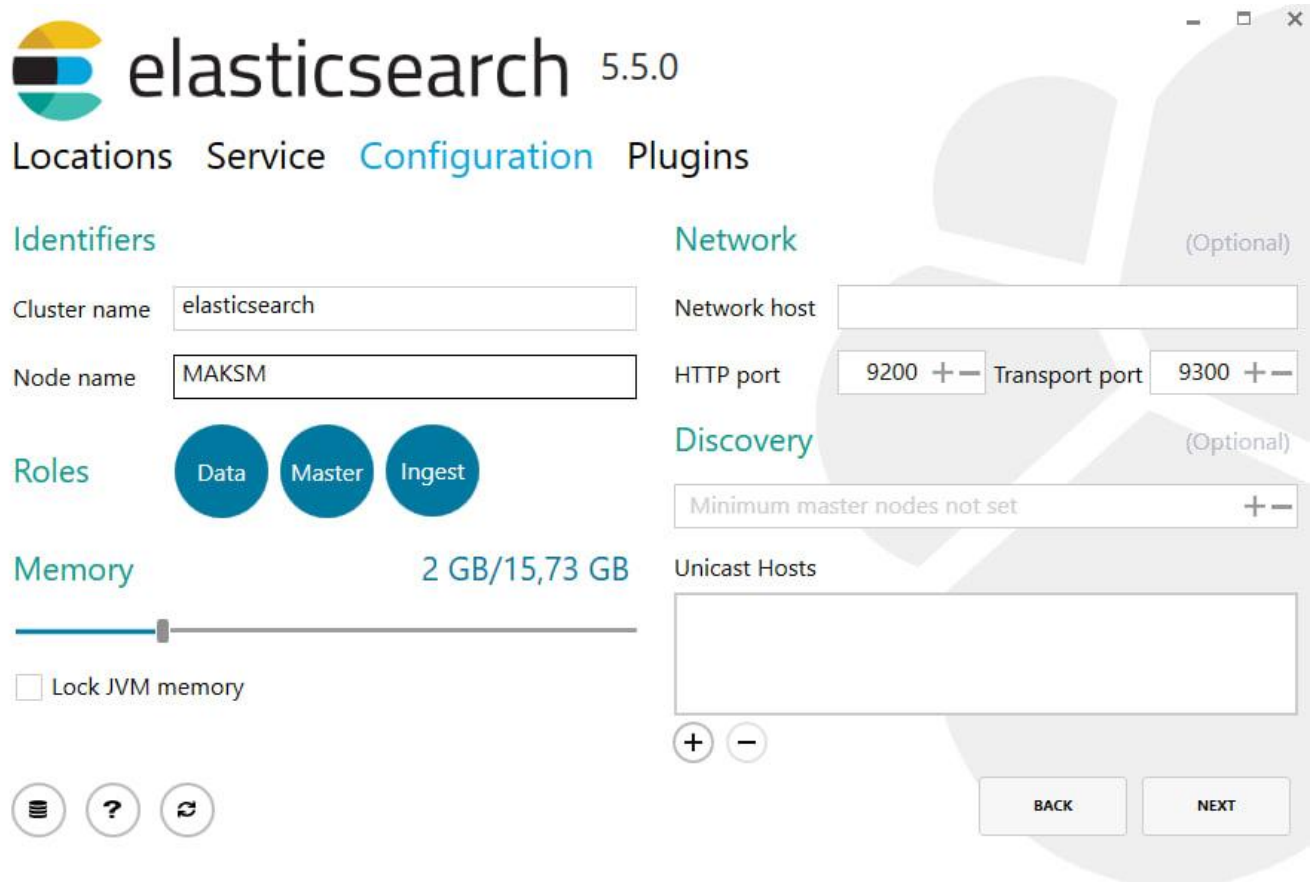


Рис. 3.8. – Конфігурація веб інтерфейсу

Процес інсталяції представлений на рисунку 3.9.

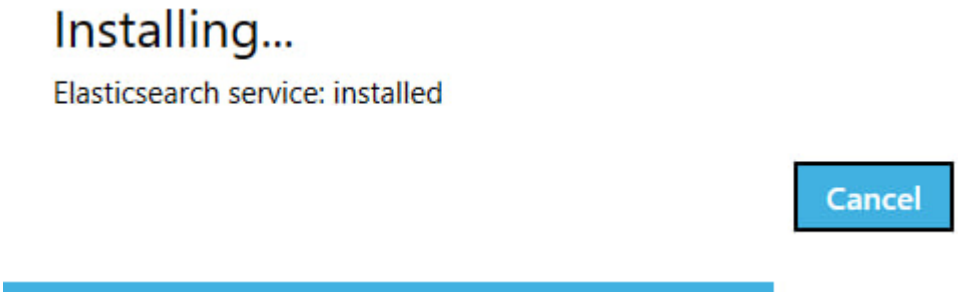


Рис. 3.9. – Процес інсталяції

На рисунку 3.10 показано автоформатування Kibana web-configuration

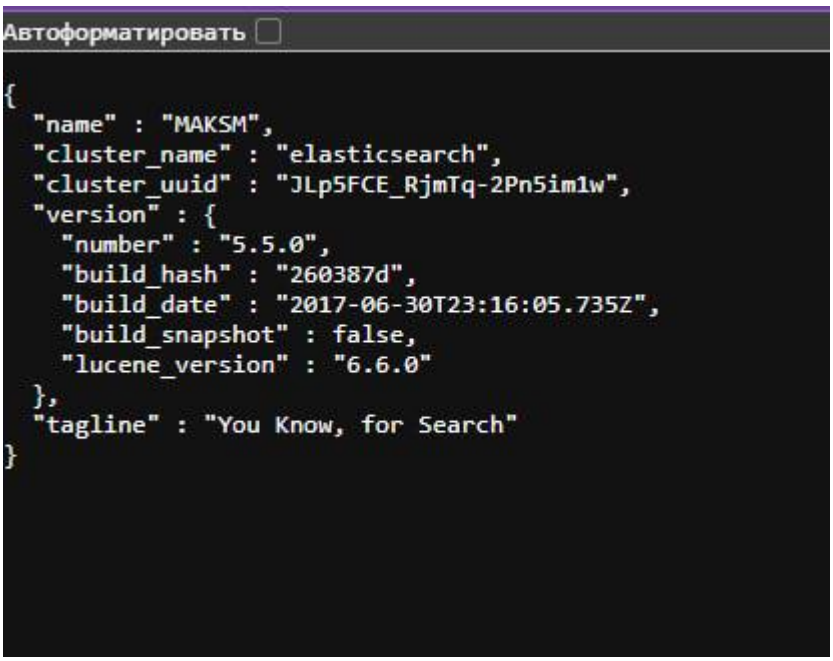


Рис. 3.10. – Автоформатування Kibana web-configuration

Процес логування демону інтерпретації продемонстрований на рисунку 3.11.

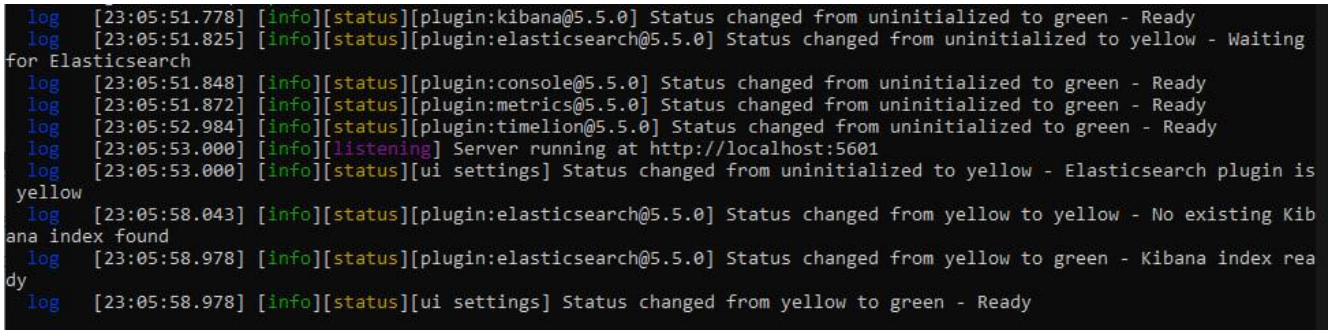


Рис. 3.11. – Логування демону інтерпретації

Налаштування передачі логів suricata, snort (рис. 3.12).

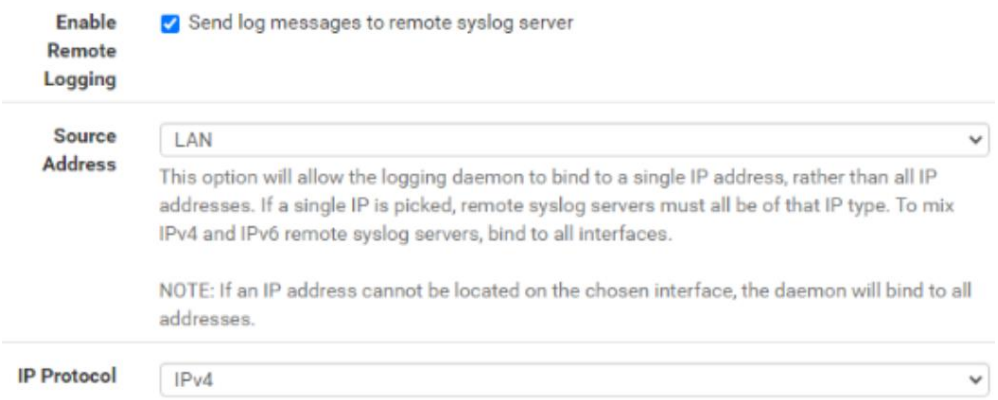


Рис. 3.12. – Налаштування передачі логів suricata, snort

Зберігаємо за кнопкою в кінці сторінки.

Перш ніж передавати логи з другого сервера, необхідно дозволити на цьому сервері, для цього переходимо шляхом Firewall>Rules і у вкладці WAN додаємо правило, що дозволяє.

Після цього заходимо на наш другий pfsense, де suricata і налаштовуємо події аналогічно як на першому. Перевіряємо, що події приходять.

Заходимо в kibana і перевіряємо. Одразу ж рекомендую всередині kibana створити новий index pattern, для цього зліва на вкладці kibana виберіть index pattern, далі на сторінці, що відкрилася, клацніть create index pattern, далі в рядок вводимо * і натискаємо, далі в рядку Time field вибираємо @timestamp.

Даний індекс буде показувати всі події, які приходять в elasticsearch. Тепер переходимо до налаштування suricata і snort.

Отже, для налаштування suricata переходимо на наш зовнішній pfsense і там прямуємо таким шляхом: Services>Suricata, зупиняємо інтерфейс за кнопкою і заходимо в його налаштування.

Там гортаємо нижче і знаходимо поле EVE Output Settings, там ставимо галочку в графі EVE JSON Log, там вказуємо такі поля: EVE Output type: FILE (у разі помилки вибере Syslog) EVE Syslog Output Facility:

AUTH EVE Syslog Output Priority: NOTICE EVE Log Alerts: Suricata буде виводити Alerts через EVE I зберігаємо налаштування (рис. 3.13). Потім виходимо назад і запускаємо інтерфейс для перевірки. В elk повинні приходити події з suricata.

Переходимо до Snort, для того, щоб налаштувати збір подій зі snort, проходимо шляхом Services>Snort, там аналогічно вимикаємо інтерфейс (якщо ввімкнений) і заходимо в його налаштування, там необхідно знайти розділ Alert Settings і в ньому поставити галочку Send Alerts to System Log...

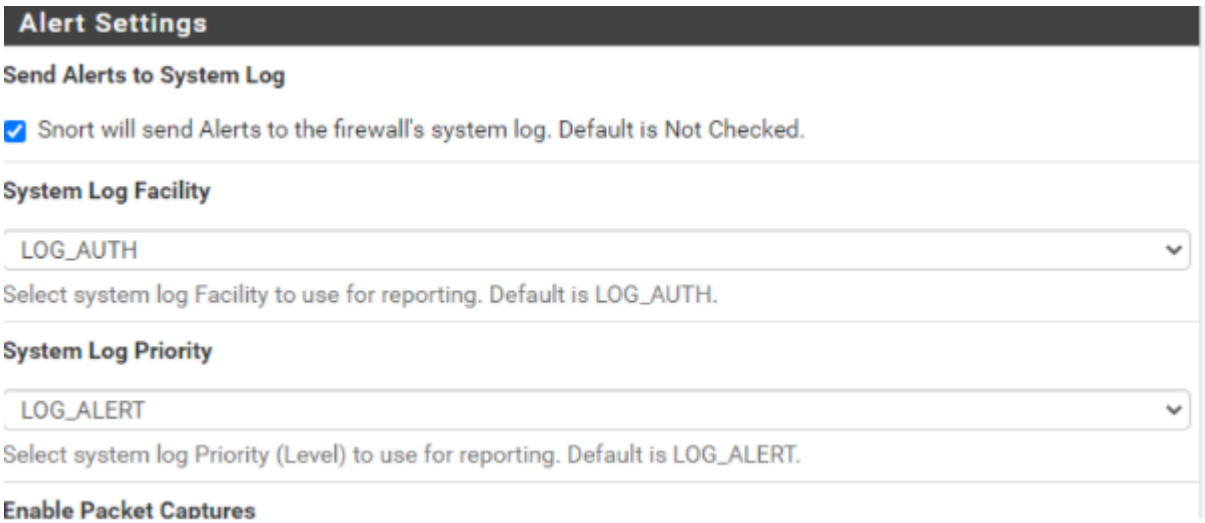


Рис. 3.13. – Включення автентифікації за AUTH

Передача подій із журналів Windows. Тепер нам необхідно налаштувати передачу подій аудиту Windows, який ми розглядали в минулій інструкції.

Як уже було зазначено на початку цієї інструкції, необхідно в інфраструктуру додати ще одну машину Windows 10.

Після того, як додали все необхідне в робочу область. Переходимо безпосередньо до налаштування. Насамперед необхідно поправити налаштування logstash нашого ELK для коректного відображення подій у kibana.

Заходимо в наш ELK і виконуємо таку команду: nano /etc/pfelk/conf.d/50-outputs.conf.

У самому кінці потрібно додати такі рядки: else { elasticsearch { hosts => ["http://localhost:9200"] index => "pthers-%{+YYYY.MM}" } } Кінцевий вигляд конфіга має мати такий вигляд як показано на рисунку 3.14.

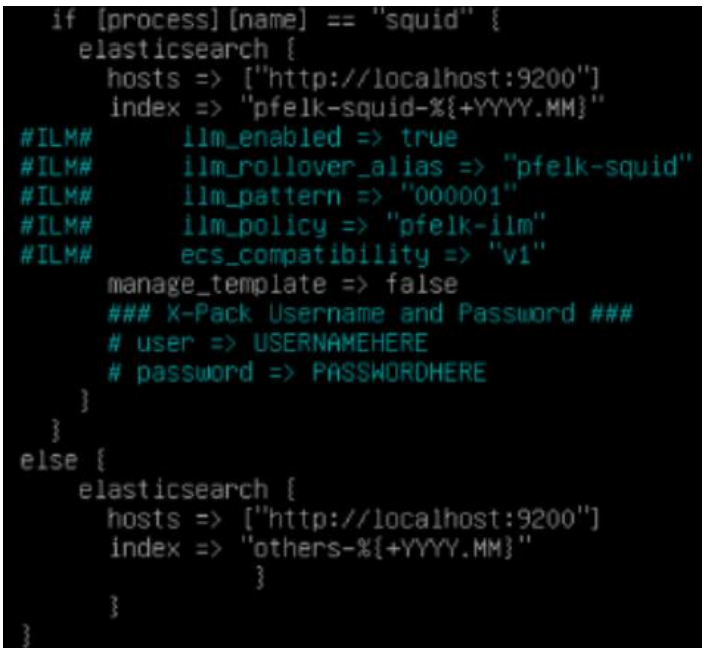


Рис. 3.14. – Кінцевий вид конфігурації ELK

Зберігаємо і виходимо. Перезапускаємо (рис. 3.15) службу logstash і перевіряємо, що все стартануло нормально командою:

tail -f /var/log/logstash/logstash-plain.log

У логах не повинно бути помилок і служба не повинна вирубуватися.

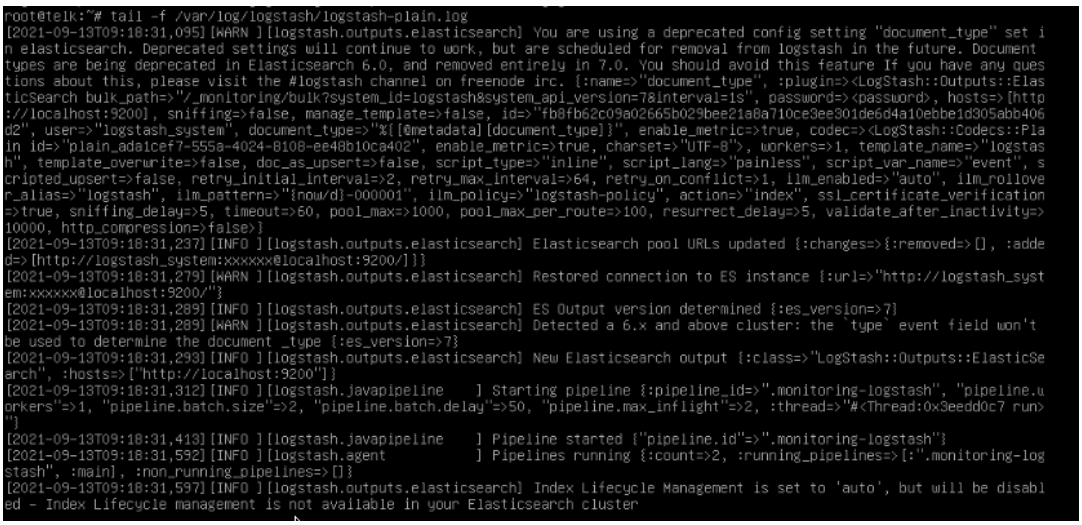


Рис. 3.15. – Перезапуск logstash

3.2. Розробка та тестування алгоритмів виявлення аномалій

Налаштування переходимо безпосередньо до конфігурування машин Windows на відправлення подій в ELK. Для відправлення подій ми використовуватимемо окреме ПЗ для надсилання подій в ELK з Windows під назвою winlogbeat.

Налаштоване ПЗ можна завантажити за посиланням: https://drive.google.com/file/d/1aC4-Dx4nVYS4Y2HMJ00tIEOWbH2CgU60/view?usp=sharing

Переносимо winlogbeat на 3 наші машини (Windows Server, Windows 7 і Windows 10) будь-яким зручним способом, можна завантажити за тим же посиланням вище або завантажити через winscp (рис. 3.16).

Витягуємо з архіву папку winlogbeat на системний диск (C: за умовчанням) у корінь. Заходимо на нього і відкриваємо конфігураційний файл winlogbeat.yml: всередині файлу необхідно змінити тільки останній рядок, а саме output.logstash: hosts: ["192.168.0.106:5601"] замість цього ip необхідно вказати свій.

Зберігаємо і закриваємо файл. Потім запускаємо командний рядок від імені адміністратора і в ньому виконуємо такі команди:

cd 'C:\Winlogbeat' PowerShell.exe -ExecutionPolicy UnRestricted -File .\install-service-winlogbeat.ps1

На запитання чи запустити скрипт вводимо R.

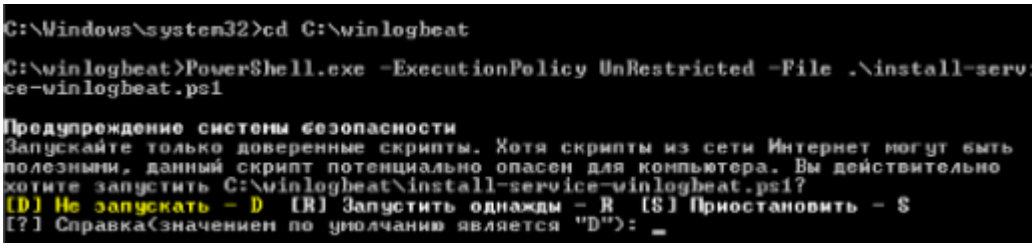


Рис. 3.16. – Завантаження та запуск служби winlogbeat

Після цього скрипт відпрацює.

Далі, там же в командному рядку введіть services.msc і руками запустіть

службу winlogbeat (рис. 3.17).

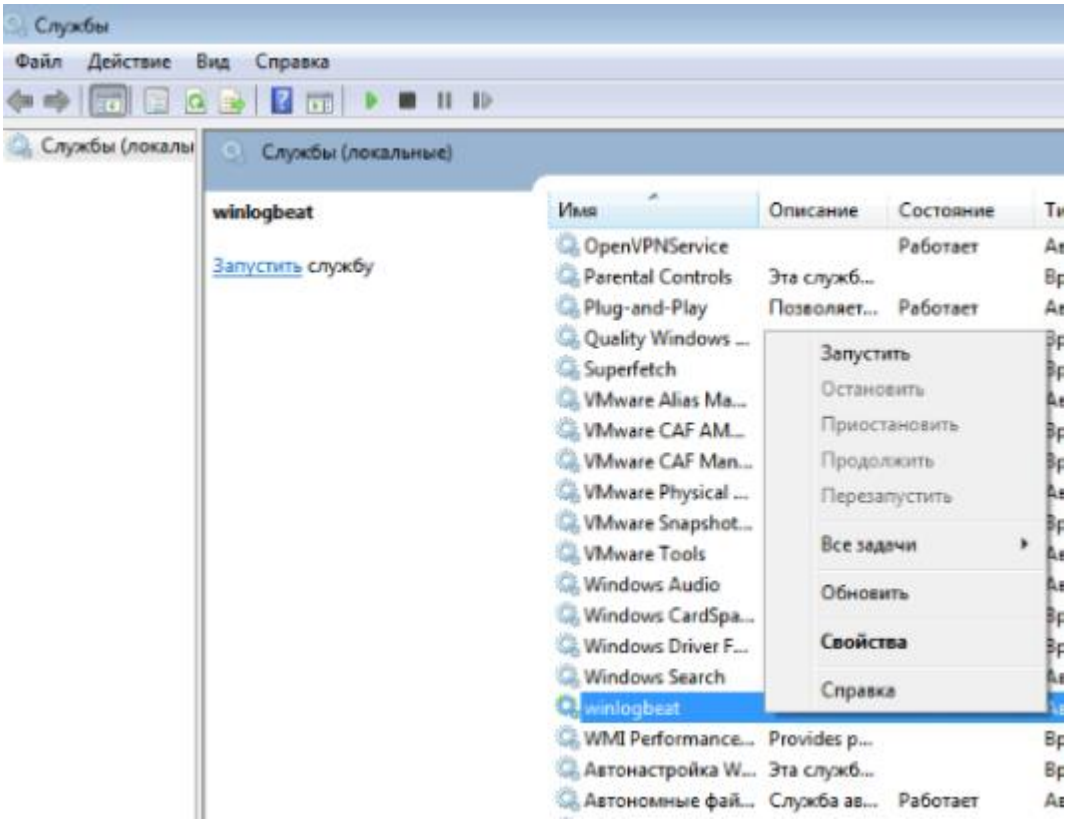


Рис. 3.17. – Перезапуск служби

Аналогічні дії необхідно виконати на інших машинах windows. Якщо ви правильно виконали всі дії в попередній інструкції, то розширений аудит у вас налаштований правильно.

Особливу увагу звертаю на синхронізацію часу машин у домені, якщо час налаштовано невірно, то події не з'являтимуться, вам необхідно синхронізувати час на контролері, на машинах, а також на ELK (спробуйте запустити службу NTP, якщо вона не запущена).

У разі, якщо все налаштовано правильно, то всі події з машин будуть успішно приходити в наш ELK.

Проведення атак на машини та перегляд їх в ELK Перш ніж проводити атаки, налаштуємо дашборд ELK для зручного перегляду подій.

Для налаштування можна відкрити будь-яку подію і вибрати поля, які дозволять нам зручно оглядати події (рис. 3.18).

Для цього зліва від поля, що містить подію, є кнопочка Toogle column in table за допомогою неї можна формувати поля для огляду.

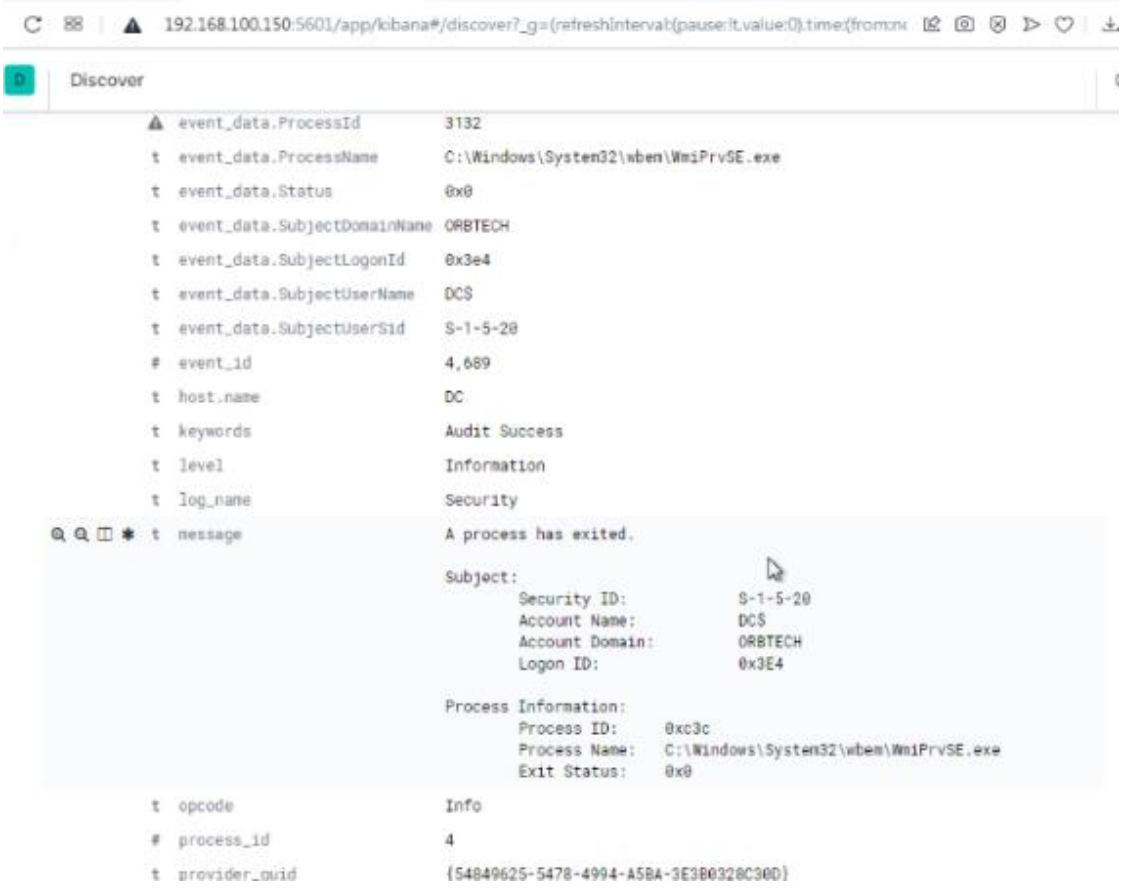


Рис. 3.18. – Перегляд подій на filebeat

Сформуємо список таким чином як показано на рисунку 3.19.

	beat.hostname	event_id	keywords	event_data.SubjectUserSid	event_data.SubjectUserName
09:38:35.816	DC	4,689	Audit Success	S-1-5-20	DC$
09:38:35.784	DC	4,689	Audit Success	S-1-5-18	DC$
09:38:35.201	DC	7,036	Classic	-	-
09:38:35.085	DC	4,689	Audit Success	S-1-5-18	DC$
09:38:35.083	DC	4,689	Audit Success	S-1-5-18	DC$
09:38:35.052	DC	4,688	Audit Success	S-1-5-18	DC$
09:38:35.046	DC	4,688	Audit Success	S-1-5-18	DC$
09:38:34.951	DC	4,688	Audit Su	S-1-5-18	DC$

Рис. 3.19. – Отримані події в режимі реального часу

Спробуємо провести атаку, наприклад, Reverse Shell і подивитися, чи побачимо ми цю подію.

Отже, за інструкцією https://hackmd.io/vviES5n1R7Kd_645sdTiPQ створюємо пейлоад і вмикаємо сервер, файл із пейлоадом передаємо на машину для запуску (рис. 3.20).

Запускаємо файл на машині.

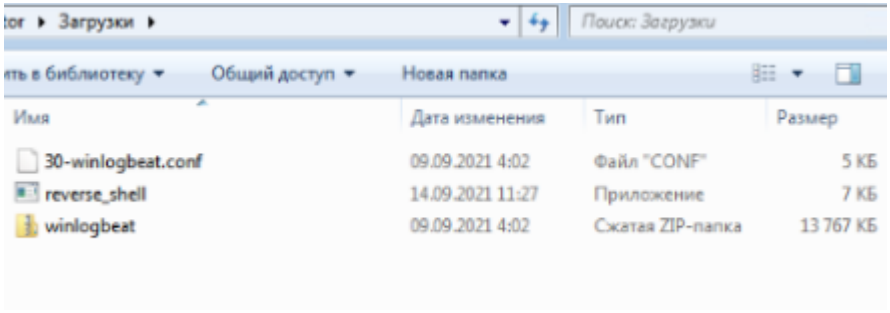


Рис. 3.20. – Запуск файлу на машині

На рисунку 3.21 зображено перегляд мережевого інтерфейсу на зовнішній адресі виходу параметром meterpreter.

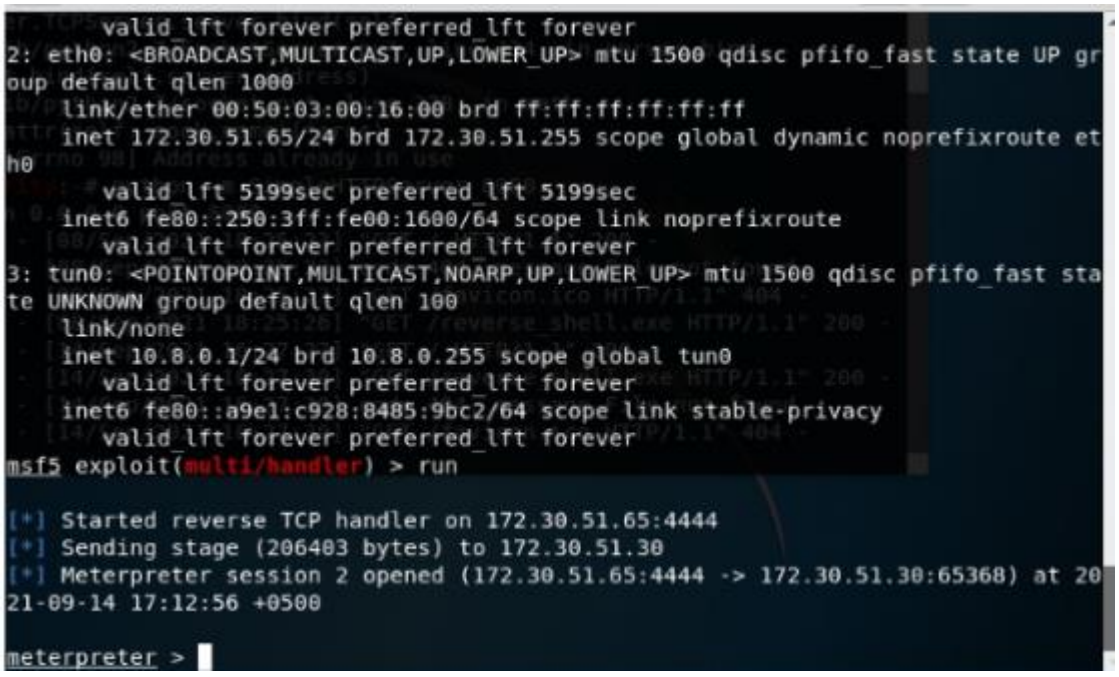


Рис. 3.21. – Перегляд мережевого інтерфейсу на зовнішній адресі виходу параметром meterpreter

Тепер подивимося події в ELK щодо цієї машини (додавши фільтр).

	beat.hostname	event_id	keywords	event_data.SubjectUserSid	event_data.SubjectUserName	event_data.ProcessName
@ 12:23:32.528	WIN7	4,689	Аудит успеха	S-1-5-21-236004864-1829554240-97714527-500	Administrator	C:\Users\administrator\Downloads\reverse_shell.exe
@ 12:23:32.509	WIN7	4,689	Аудит успеха	S-1-5-21-236004864-1829554240-97714527-500	Administrator	C:\Windows\System32\WerFault.exe
@ 12:23:18.414	WIN7	4,688	Аудит успеха	S-1-5-21-236004864-1829554240-97714527-500	Administrator	-
@ 12:19:32.153	WIN7	4,689	Аудит успеха	S-1-5-21-236004864-1829554240-97714527-500	Administrator	C:\Windows\System32\wermgr.exe
@ 12:16:19.295	WIN7	4,689	Аудит успеха	S-1-5-21-236004864-1829554240-97714527-500	Administrator	C:\Windows\System32\slui.exe

Рис. 3.7. – Запуск підозрілого процесу в журналі подій

Як видно зі скріншота (рис. 3.7), з'явилася подія запуску підозрілого процесу. Таким чином, ми можемо бачити різні атаки.

Висновок до розділу 3

Проведено практичну реалізацію алгоритмів виявлення аномалій з використанням SIEM ELK Stack. В рамках підготовки середовища було налаштовано систему збору та обробки даних, що включала налаштування SIEM (Security Information and Event Management) та ELK Stack (Elasticsearch, Logstash, Kibana), а також збір різноманітних даних для подальшого аналізу.

Розроблено та протестовано алгоритми виявлення аномалій. Цей процес включав в себе аналіз різних типів даних, їхню обробку та моделювання нормальної поведінки для виявлення відхилень. Після цього алгоритми були внедрені в середовище SIEM ELK Stack та протестовані на реальних даних для перевірки їхньої ефективності та точності.

Отже, розділ відображає успішну реалізацію практичних аспектів використання SIEM ELK Stack для виявлення аномалій у системі інформаційної безпеки.

ВИСНОВКИ

У першому розділі проведено теоретичному аналізу аномалій у мережевій інфраструктурі. Зокрема, визначено поняття «аномалії в мережевій інфраструктурі» та розглянуто загрози та ризики, пов’язані з такою активністю.

Досліджено різноманітні методи та підходи до виявлення аномалій, включаючи статистичні моделі, машинне навчання та аналіз великих даних. Виявлення та відстеження аномалій у мережевій інфраструктурі має критичне значення для забезпечення безпеки та ефективності роботи мережі.

Застосування відповідних методів дозволяє оперативно виявляти потенційно шкідливу активність та приймати необхідні заходи для її нейтралізації. Даний розділ є важливим етапом у розумінні та боротьбі з аномальною активністю у мережевому середовищі.

Проведено глибокий аналіз функцій, переваг та обмежень SIEM ELK Stack у контексті виявлення аномалій у системі.

Розглянувши технологію SIEM ELK Stack, було з'ясовано, що вона складається з Elasticsearch, Logstash та Kibana, що забезпечує повний цикл збору, обробки та візуалізації данних. Це дозволяє забезпечити повний огляд стану безпеки мережі та виявлення аномалій.

Представлений приклад практичного використання SIEM ELK Stack для виявлення аномалій. Цей приклад підкреслив можливості системи у виявленні несправностей та непередбачених змін у системі, що дозволяє оперативно реагувати на потенційні загрози

Розглянуто не лише переваги, а й обмеження використання SIEM ELK Stack. Серед них складність налаштування, вимоги до ресурсів та потреба у кваліфікованому персоналі. Ці фактори можуть ускладнити впровадження системи та зменшити її ефективність у деяких сценаріях.

Можна сказати, що SIEM ELK Stack є потужним інструментом для виявлення аномалій у системі, проте впровадження його вимагає ретельного аналізу індивідуальних потреб та можливостей організації. При належному

налаштуванні та ефективному використанні, він може значно підвищити рівень безпеки та захисту мережі від потенційних загроз.

Проведено практичну реалізацію алгоритмів виявлення аномалій з використанням SIEM ELK Stack. В рамках підготовки середовища було налаштовано систему збору та обробки даних, що включала налаштування SIEM (Security Information and Event Management) та ELK Stack (Elasticsearch, Logstash, Kibana), а також збір різноманітних даних для подальшого аналізу.

Розроблено та протестовано алгоритми виявлення аномалій. Цей процес включав в себе аналіз різних типів даних, їхню обробку та моделювання нормальної поведінки для виявлення відхилень. Після цього алгоритми були внедрені в середовище SIEM ELK Stack та протестовані на реальних даних для перевірки їхньої ефективності та точності.

ПЕРЕЛІК ПОСИЛАНЬ

1. Heady R., Luger G., Maccabe A. and Servilla M. The Architecture of a Network Level Intrusion Detection System. Computer Science Department, University of New Mexico, Tech. Rep. TR-90. 1990. 21 p. URL: https://www.osti.gov/servlets/purl/425295 (дата звернення: 10.03.2024).

2. Chandola V., Banerjee A., Kumar V. Anomaly Detection: A Survey. ACM Computing Surveys. 2009. Vol. 41, no. 3. P. 15–38.

3. Daniel B., Julia C., Sushil J, Ningning W. ADAM: a test bed for exploring the use of data mining in intrusion detection. ACM SIGMOD Record. 2001. Vol. 30, no. 4, pp. 15–24.

4. Bhuyan M. H., Bhattacharyya D. K., Kalita J. K. Surveying Port Scans and Their Detection Methodologies. The Computer Journal. 2011. Vol. 54, no. 10. P. 1565–1581.

5. Lesot M. J., Rifqi M. Anomaly-based network intrusion detection: Techniques, systems and challenges. International Journal of Knowledge Engineering and Soft Data Paradigms. 2009. Vol. 1, no. 1. P. 63–84.

6. Chio, C.; Freeman, D. Machine Learning and Security, O'Relly Media, 2018, 125-180.

7. Ahmed M, Mahmood A. Network traffic analysis based on collective anomaly detection. In: 2014 IEEE 9th conference on industrial electronics and applications (ICIEA), 2014. p. 1141–46

8. Network Intrusion. URL: https://awakesecurity.com/glossary/network-intrusion/ (дата звернення: 20.03.2024).

9. Микитишин А.Г., Митник М.М., Стухляк П.Д.. Комплексна безпека інформаційних мережевих систем: навчальний посібник – Тернопіль: Вид-во ТНТУ імені Івана Пулюя, 2016. – 256 с.

10. Costin A. Security of cctv and video surveillance systems: Threats, vulnerabilities, attacks, and mitigations. pp. 45–54. URL: https://dl.acm.org/doi/abs/10.1145/2995289.2995290 (дата звернення: 10.04.2024).

11. Kocher P., Jaffe J., Jun B. Introduction to Differential Power Analysis and Related Attacks. URL: https://www.rambus.com/wp-content/uploads/2015/08/DPATechInfo.pdf (дата звернення: 10.04.2024).

12. Новотарський М.А., Нестеренко Б.Б. Штучні нейронні мережі: обчислення // Праці Інституту математики НАН України. – Т50. – Київ: Ін-т математики НАН України, 2004. – 408 с.

13. "Botnet detection". WIRED. Retrieved 2017-05-24. URL: http://jpdias.me/botnetlab//countermeasures/detection.html (дата звернення: 12.04.2024).

14. "Host-based intrusion detection systems". URL: https://gradesfixer.com/free-essay-examples/hostbased-intrusion-detection-systems/ (дата звернення: 15.04.2024).

15. KARTHIK "Tutorial: Visualize historical data with ELK stack". URL: https://www.upnxtblog.com/index.php/2018/08/09/tutorial-visualize-historicaldata-with-elk-stack/ (дата звернення: 20.04.2024).

16. Офіційний сайт інструмента ELK. URL: https://www.elastic.com. (дата звернення: 20.04.2024).

17. Daniel Berman. "Network Security Monitoring with Suricata, Logz.io and the ELK Stack". URL: https://logz.io/blog/network-security-monitoring/ (дата звернення: 25.04.2024).

18. Остапов С. Е. Технології захисту інформації : навчальний посібник / С. Е. Остапов, С. П. Євсеєв, О. Г. Король. – Х. : Вид. ХНЕУ, 2013. – 476 с

19. Гончарова Л.Л., Возненко А.Д., Стасюк О.І., Коваль Ю.О. Основи захисту інформації в телекомунікаційних та комп’ютерних мережах. – К., 2013. – 435 с.

20. Amiri F., Yousefi M. M. R., Lucas C., Shakery A., Yazdani N. Mutual information-based feature selection for intrusion detection systems. Journal of Network and Computer Applications. 2001. Vol. 34, no. 4. P. 1184–1199.

21. Anscombe J., Guttman I. Rejection of outliers. Technometrics. 1960. Vol. 2, no. 2. P. 123–147.

22. Eskin E. Anomaly detection over noisy data using learned probability distributions. ICML 2000: Proc. of the 7th International Conference on Machine Learning, Stanford, CA, USA, 29 June – 2 July, 200. Heidelberg: Springer-Verlag. P. 255–262.

23. Mahoney M. V., Chan P. K. Learning rules for anomaly detection of hostile network traffic. ICDM 2003: Proc. of the 3rd IEEEInternational Conference on Data Mining, Melbourne, FL, USA, 22-22 November, 2003. Washington: IEEECS. P. 145-176.

24. Wattenberg S., Perez J. I. A., Higuera P. C., Fernandez M. M., Dimitriadis I. A. Anomaly Detection in Network Traffic Based on Statistical Inference and Stable Modeling. IEEE Transactions on Dependable and Secure Computing. 2001. Vol. 8, no. 4. P. 494–509.

25. Amiri F., Yousefi M. M. R., Lucas C., Shakery A., Yazdani N. Mutual information-based feature selection for intrusion detection systems. Journal of Network and Computer Applications. 2001. Vol. 34, no. 4. P. 1184–1199.

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут інформаційних технологій
КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ

Кваліфікаційна робота на тему:

«Розробка алгоритмів виявлення аномалій для захисту мережевої інфраструктури»

Виконав: Марковський Максим
Група: ШІД – 41
Спеціальність: Комп'ютерні науки, кафедра штучного інтелекту
Науковий керівник: старший викладач, Кисіль Т.М.

КИЇВ - 2024

Мета роботи – підвищення продуктивності алгоритмів виявлення аномалій для захисту мережевої інфраструктури з використанням SIEM ELK Stack.

Об'єкт дослідження – процес підвищення продуктивності алгоритмів виявлення аномалій атак інструкторів та небажаних подій.

Предмет дослідження – алгоритм виявлення аномалій в мережевій інфраструктурі SIEM ELK Stack.

Методи дослідження:

Експериментальні дослідження, що включають в себе розробку та тестування алгоритмів виявлення аномалій на реальних або симульованих даних.

Методи аналізу та порівняння результатів, що використовуються для оцінки ефективності розроблених алгоритмів.

Використання практичних методів розробки програмного забезпечення для реалізації алгоритмів виявлення аномалій.

Архітектура виявлення аномалій штучним інтелектом в реальному часі

3

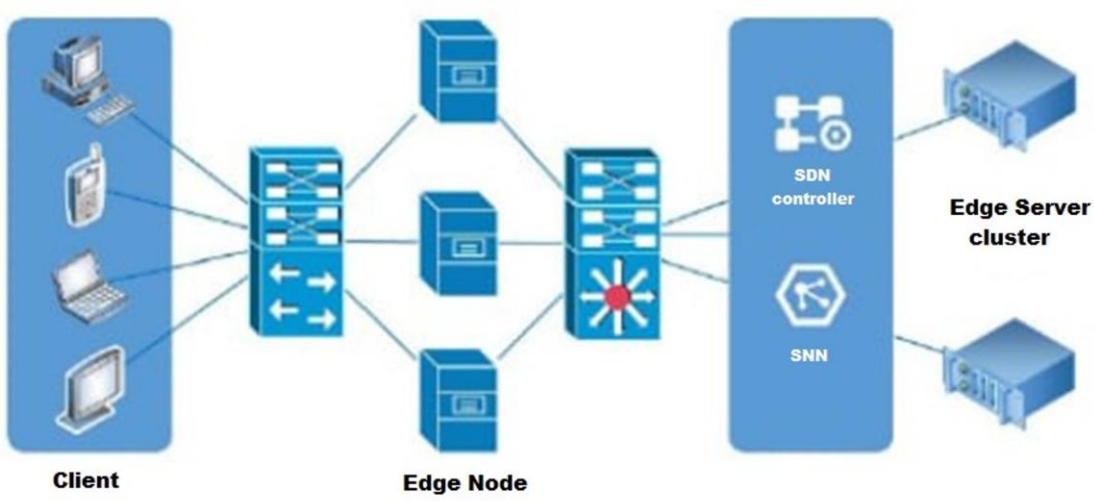


Рисунок 1 – Схематичне зображення архітектури виявлення аномалій штучним інтелектом в реальному часі в SIEM ELK-кластерній мережі на основі SDN та CNN.

Методи штучного інтелекту для виявлення аномалій

4

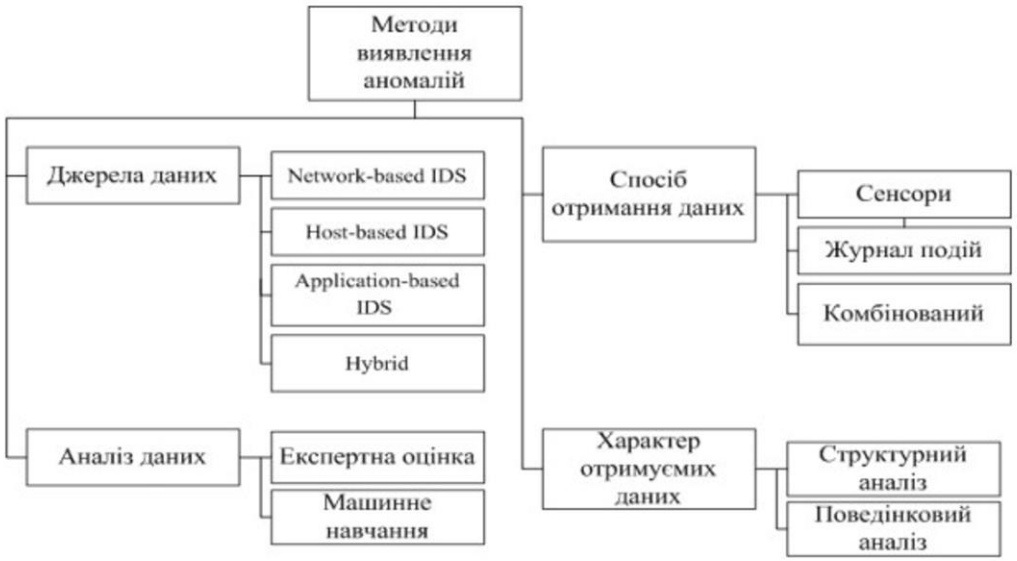


Рисунок 2 – Класифікація методів виявлення аномалій

Виявлення аномалій штучним інтелектом

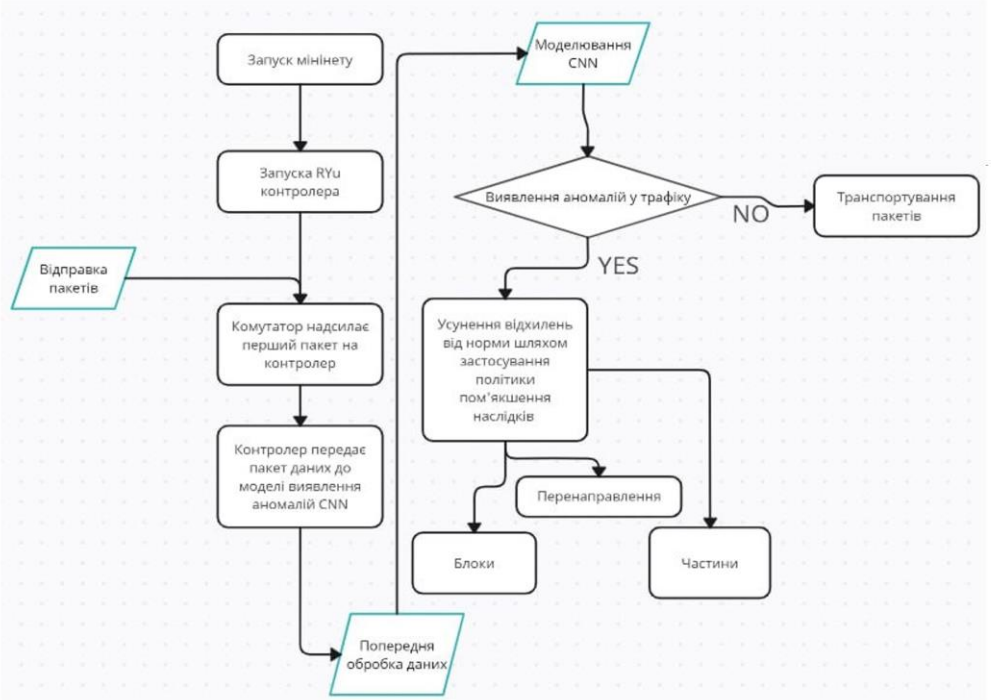


Рисунок 3 – Схема реалізації виявлення аномалій

ВстановленняELK Stack з репозиторіюpfELK

Рисунок 4 – Інсталяція ELK Stack з репозиторію pfELK .

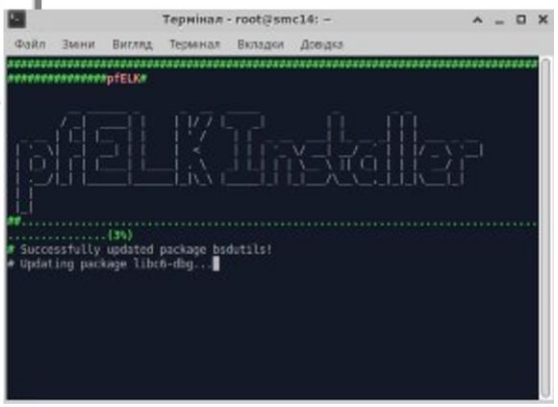


Рисунок 5 – Встановлення ліцензійного ключа MaxMind та оновлення пакетів

Конфігурація веб інтерфейсуELK

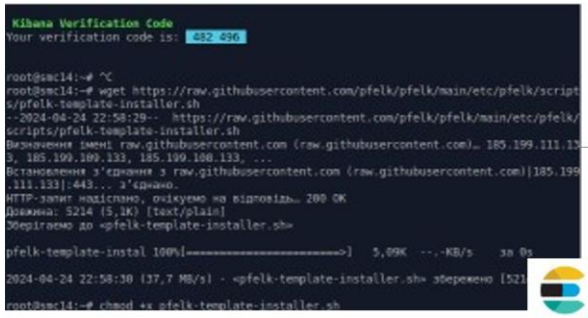


Рисунок 6 – Завантаження темплейтів pfelk за посиланням wget

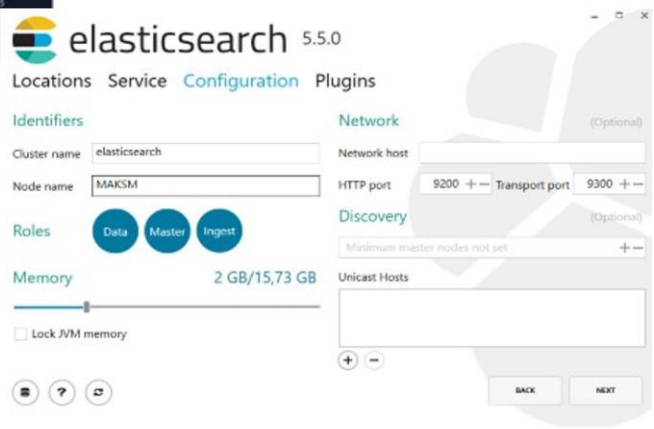


Рисунок 7 – Вхід в Kibana

Розробката тестування алгоритміввиявленняаномалій

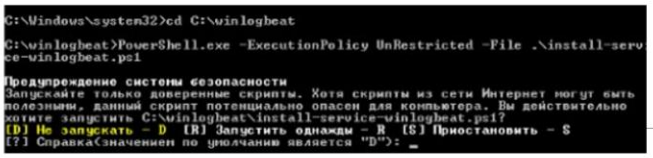
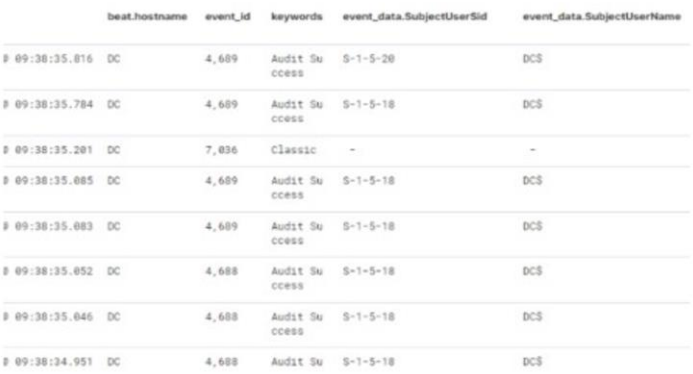


Рисунок 8 – Завантаження та запуск служби winlogbeat



	beat.hostname	event_id	keywords	event_data.SubjectUserSid	event_data.SubjectUserName
09:38:35.816	DC	4,689	Audit Success	S-1-5-20	DC$
09:38:35.784	DC	4,689	Audit Success	S-1-5-18	DC$
09:38:35.201	DC	7,036	Classic	-	-
09:38:35.085	DC	4,689	Audit Success	S-1-5-18	DC$
09:38:35.083	DC	4,689	Audit Success	S-1-5-18	DC$
09:38:35.052	DC	4,688	Audit Success	S-1-5-18	DC$
09:38:35.046	DC	4,688	Audit Success	S-1-5-18	DC$
09:38:34.951	DC	4,688	Audit Su	S-1-5-18	DC$

Рисунок 9 – Отримані події в режимі реального часу

Проведення тестування знаходження підозрілого процесу в журналі подій

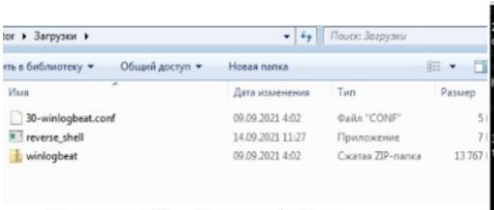


Рисунок 10 – Запуск файлу

Рисунок 11 – Перегляд мережевого інтерфейсу на зовнішній адресі виходу параметром meterpreter

Рисунок 12 – Запуск підозрілого процесу в журналі подій

ВИСНОВКИ

- Проведено глибокий аналіз функцій, переваг та обмежень SIEM ELK Stack у контексті виявлення аномалій у системі.
- Можна сказати, що SIEM ELK Stack є потужним інструментом для виявлення аномалій у системі, проте впровадження його вимагає ретельного аналізу індивідуальних потреб та можливостей організації . При належному налаштуванні та ефективному використанні , він може значно підвищити рівень безпеки та захисту мережі від потенційних загроз.
- Проведено практичну реалізацію алгоритмів виявлення аномалій з використанням SIEM ELK Stack. В рамках підготовки середовища було налаштовано систему збору та обробки даних, що включала налаштування SIEM (Security Information and Event Management) та ELK Stack (Elasticsearch, Logstash, Kibana), а також збір різноманітних даних для подальшого аналізу.
- Розроблено та протестовано алгоритми виявлення аномалій. Цей процес включав в себе аналіз різних типів даних, їхню обробку та моделювання нормальної поведінки для виявлення відхилень . Після цього алгоритми були внедрені в середовище SIEM ELK Stack та протестовані на реальних даних для перевірки їхньої ефективності та точності.