НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

До захисту допущено
В.о. завідувача кафедри

_____ **Микола ГРАЙВОРОНСЬКИЙ**
(підпис)

«_____»_____2021 р.

Дипломна робота

на здобуття ступеня бакалавра
за освітньо-професійною программою «Системи, технології та математичні методи кібербезпеки»
спеціальності: 125 «Кібербезпека»

на тему: «Виявлення аномалій мережного трафіку з метою знешкодження прихованих каналів передачі інформації у мережних протоколах»

Виконала: здобувач вищої освіти **IV** курсу, групи __ФБ-71__
(шифр групи)

Нацвін Катерина Анатоліївна _____
(прізвище, ім'я, по батькові) (підпис)

Керівник к.т.н., доцент Демчинський Володимир Васильович _____
(посада, науковий ступінь, вчене звання, прізвище, ім'я, по батькові) (підпис)

Рецензент _____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище, ім'я, по батькові) (підпис)

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.
Здобувач вищої освіти _____
(підпис)

Київ - 2021 року

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

Рівень вищої освіти – перший (бакалаврський)
Спеціальність – 125 «Кібербезпека»
Освітньо-професійна програма «Системи, технології та математичні методи кібербезпеки»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ Микола ГРАЙВОРОНСЬКИЙ
(підпис)
«___»_____2021 р.

ЗАВДАННЯ
на дипломну роботу здобувачу вищої освіти

Нацвін Катерина Анатоліївна
(прізвище, ім'я, по батькові)

1. Тема роботи «Виявлення аномалій мережного трафіку з метою знешкодження прихованих каналів передачі інформації у мережних протоколах», керівник роботи Демчинський Володимир Васильович к.т.н., доцент,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету від « » _____ 2021 р. №

2. Термін подання здобувачем вищої освіти роботи 07 червня 2021 р.

3. Вихідні дані до роботи: попередні дослідження прихованих каналів передачі інформації в мережних каналах.

4. Зміст роботи: аналіз існуючих методів створення прихованих каналів, дослідження та випробування методів виявлення та знешкодження відомих прихованих каналів, розробка та реалізація власного методу виявлення прихованих каналів.

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо) Виявлення аномалій мережного трафіку з метою знешкодження прихованих каналів передачі інформації у мережних протоколах — презентація.

6. Дата видачі завдання 15.20.2020.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів дипломної роботи	Примітка
1	Отримання завдання	15.10.2020	виконано
2	Вивчення та аналіз літератури	16.10.2020 – 20.01.2021	виконано
3	Аналіз існуючих методів створення прихованих каналів	21.01.2021 – 10.02.2021	виконано
4	Дослідження та випробування методів виявлення та знешкодження відомих прихованих каналів	11.02.2021 – 20.03.2021	виконано
5	Розробка та реалізація власного методу виявлення прихованих каналів	21.03.2021 – 15.05.2021	виконано
6	Проходження переддипломної практики	12.04.2021 – 16.05.2021	виконано
7	Написання дипломної роботи	11.02.2021 – 01.06.2021	виконано
8	Отримання допуску до захисту	02.06.2021	
9	Захист дипломної роботи	15.06.2021	

Здобувач вищої освіти _____ Катерина НАЦВІН
(підпис) (Власне ім'я, ПРІЗВИЩЕ)

Керівник роботи _____ Володимир ДЕМЧИНСЬКИЙ
(підпис) (Власне ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Обсяг роботи 66 сторінок, 37 ілюстрацій, 1 додаток, 40 джерел літератури.

В роботі досліджено методи виявлення та знешкодження відомих прихованих каналів у мережних протоколах. Випробувано готові засоби боротьби із прихованими каналам на прикладі DNS тунелю. Розроблено та реалізовано власний метод виявлення прихованих каналів передачі інформації у мережних протоколах шляхом детектування аномалій мережного трафіку. У ході дослідження створено засоби моніторингу та аналізу мережного трафіку – вочер та дашборд. Доведено доцільність використання комбінації Wireshark та Elastic Stack з метою виявлення потенційного прихованого каналу. Розглянуто превентивні заходи боротьби із прихованими каналами.

Отримані результати можуть бути використані для створення комплексного захисту системи від прихованих каналів передачі інформації у мережних протоколах. Розроблений метод може слугувати для виявлення нових, ще не описаних прихованих каналів, без відомих сигнатур.

Ключові слова: приховані канали у мережних протоколах, таємна передача інформації, стеганографія, моніторинг трафіку, аналіз трафіку, знешкодження прихованих каналів.

ABSTRACT

The volume of work is 66 pages, 37 illustrations, 1 appendice, 40 sources of literature.

Methods of detection and neutralizing of known covert channels in network protocols were investigated. Ready-made tools to eliminate hidden channels have been tested on the example of DNS tunnel. Own method of detecting covert channels of information transmission in network protocols by detecting network traffic anomalies was developed and implemented. In the course of the study, the means of monitoring and analysis of network traffic – watcher and dashboard – were created. The expediency of using the Wireshark and Elastic Stack combination to identify a potential covert channel has been proven. Preventive measures to use as countermeasures against covert channels were investigated.

Obtained results can be used to create a comprehensive system protection against covert channels of information transmission in network protocols. The developed method can serve to detect new, not yet described covert channels, without known signatures.

Keywords: covert channels in network protocols, secret transmission of information, steganography, traffic monitoring, traffic analysis, covert channels elimination.

ЗМІСТ

Вступ..... 9

1 Приховані канали: сутність, історія, застосування 11

1.1 Визначення поняття прихованого каналу передачі інформації в мережному протоколі 12

1.2 Історія розвитку концепції прихованих каналів 13

1.3 Причини та способи використання прихованих каналів 14

Висновок до розділу 1: 16

2 Класифікація прихованих каналів та стратегії захисту.....17

2.1 Класифікація прихованих каналів за рівнем стеку TCP/IP.....18

2.2 Стратегії виявлення та знешкодження прихованих каналів.....29

Висновок до розділу 2: 32

3 Виявлення та знешкодження прихованих каналів передачі інформації у мережних протоколах 33

3.1 Засоби виявлення та знешкодження відомих прихованих каналів ..33

3.2 Виявлення та знешкодження прихованих каналів на основі аналізу мережного трафіку 38

3.3 Превентивні заходи захисту від прихованих каналів 54

3.4 Порівняння ефективності наведених способів захисту 56

Висновок до розділу 3: 56

Висновки.....58

Перелік джерел посилань.....59

Додаток А Розроблений вочер для elastic stack.....63

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Стеганографія (Steganography) — тайнопис, що приховує сам факт існування повідомлення.

Trusted Computer Security Evaluation Criteria (TCSEC) — Критерії оцінки захищеності комп'ютерної системи.

Protocol Data Unit (PDU) — одиниця даних протоколу.

Raw Bit Rate (RBR) — загальна кількість бітів інформації, що передаються протягом секунди.

Packet Raw Bit Rate (PRBR) — загальна кількість бітів прихованої інформації, що передаються з одиницею даних протоколу.

TCP/IP Stack — це систематизований стек протоколів, що поділяється на чотири рівні, які корелюються з еталонною моделлю OSI.

Брандмауер (Firewall) — фізичний пристрій чи програмний застосунок, сконфігурований, щоб допускати, відмовляти, шифрувати, пропускати мережевий трафік між областями різної безпеки мережі згідно з бажаним набором правил безпеки.

Internet Control Message Protocol (ICMP) — протокол, що використовується для передачі повідомлень про помилки й інші виняткові ситуації при передачі даних. На ньому заснована дія утиліт ping та traceroute.

Domain Name System (DNS) — ієрархічна розподілена система перетворення імені хоста в IP-адресу.

Intrusion Prevention System & Intrusion Detection System (IPS & IDS) — програмний або апаратний засіб, для попередження та виявлення фактів несанкціонованого доступу в комп'ютерну систему або мережу або несанкціонованого управління ними в основному через Інтернет.

Internet Group Management Protocol (IGMP) — це протокол керування груповою (multicast) передачею даних в мережах, що базуються на протоколі IP.

Internet Protocol (IP) протокол мережевого рівня для передавання датаграм між мережами.

Dynamic Host Configuration Protocol (DHCP) — протокол мережевого рівня, який дозволяє серверу автоматично призначати IP адресу комп'ютеру з певного діапазону, налаштованого для певної мережі.

Address Resolution Protocol (ARP) — це протокол для визначення MAC адреси за IP адресою.

Transmission Control Protocol (TCP) — протокол, забезпечує послуги надійного зв'язку зі встановленням з'єднання.

User Datagram Protocol (UDP) — протокол, що не потребує встановлення з’єднання, є менш надійним ніж TCP.

Secure Sockets Layer (SSL) і його наступник Transport Layer Security (TLS) — це протоколи шифрування/розшифровки даних при передачі для захищеного зв'язку між двома сторонами.

HyperText Transfer Protocol (HTTP) слугує для передачі та відображення вебсторінок (текстових файлів з розміткою HTML) та файлів.

File Transfer Protocol (FTP) — стандартний мережевий протокол прикладного рівня призначений для пересилання файлів між клієнтом та сервером в комп'ютерній мережі.

Secure Shell (SSH) — криптографічний протокол клієнт-сервера для безпечного обміну даними, віддаленого виконання команд та інших захищених мережевих служб між двома комп'ютерами, які з'єднуються через захищений канал через незахищену мережу.

Ексфільтрація — процес вивантаження даних зі скомпрометованого хоста.

Proof of concept — доказ концепції — реалізація певного методу чи ідеї з метою продемонструвати її доцільність, практичний потенціал.

Вочер (watcher) — скрипт виконання дій в залежності від виконання умов в Elasticsearch.

Дашборд (dashboard) — панель візуалізації даних в Elasticsearch.

Алерт (alert) — сповіщення про настання певної події.

ВСТУП

Актуальність роботи обумовлюється тим, що з початком діджиталізації нашого світу розпочалися спроби скомпрометувати комп’ютерні системи. Поява Інтернету лише збільшила поле для скоєння кіберзлочинів. Як наслідок постійно зростає обсяг інформації про вразливості та шляхи їх знешкодження. Відомі методи атак легко знешкодити, тому злочинці намагаються використати неочевидні способи. Одним із таких є використання прихованих каналів передачі інформації. З іншого боку приховані канали надають можливість безпечнішої передачі інформації, наприклад, системним адміністраторам.

Метою даної роботи є дослідження методів передачі інформації за допомогою прихованих каналів та їх знешкодження.

Для досягнення мети було поставлено такі завдання:

- огляд відомих прихованих каналів
- аналіз методів створення прихованих каналів передачі інформації у мережних протоколах
- огляд методів виявлення та знешкодження прихованих каналів у мережних протоколах
- Розробка та реалізація власного методу виявлення прихованих каналів передачі інформації у мережних протоколах шляхом детектування аномалій мережного трафіку

Об’єктом дослідження є приховані канали передачі інформації у мережних протоколах.

Предметом дослідження є методи створення, виявлення та знешкодження прихованих каналів у мережних протоколах.

Методом дослідження є вивчення літератури, наукових статей та інших інформаційних джерел з теми, опрацювання отриманих результатів,

практична реалізація створення, виявлення та знешкодження прихованих каналів у мережних протоколах.

Наукова новизна одержаних результатів ґрунтується на систематизації та аналізі відомостей про приховані канали передачі інформації в мережних протоколах, що вилилась у розробку та реалізацію власного методу їх виявлення та знешкодження шляхом детектування аномалій мережного трафіку.

Практичне значення результатів роботи полягає у можливості їх використання спеціалістами з кібербезпеки для створення комплексного захисту системи, що дозволить попередити створення прихованих мережних каналів, передачі через них інформації та інших пов’язаних з ними кіберінцидентів. Розроблений метод може слугувати для виявлення нових, ще не описаних прихованих каналів, без відомих сигнатур.

Дослідження було опубліковано на XIX Всеукраїнській науково-практичній конференції студентів, аспірантів та молодих вчених “Теоретичні і прикладні проблеми фізики, математики та інформатики”.

1 ПРИХОВАНІ КАНАЛИ: СУТНІСТЬ, ІСТОРІЯ, ЗАСТОСУВАННЯ

Часто шифрування вважають ключовим елементом захисту інформації. Однак воно лише запобігає розкриттю змісту повідомлення. У багатьох випадках саме існування каналу зв’язку або незначні відхилення від звичної моделі поведінки системи можуть стати підозрілими та привернути небажану увагу. Приховані ж канали маскують сам факт передачі інформації. Зазвичай вони використовують засоби, що не були призначені для цього, майже унеможливлюючи виявлення каналу зв’язку.

Через посилення заходів безпеки проти відкритих каналів передачі інформації зростає частота використання прихованих мережних каналів. Велика кількість даних в інтернеті та різноманіття протоколів є ідеальним середовищем для прихованої передачі інформації. Кількість прихованих каналів значно зростає через появу нових швидкісних мережних технологій. Багато застосувань прихованих каналів є злочинними. Навіть якщо можна таємно передати лише один біт інформації з одним пакетом – загроза залишається значною. Виявлення, знищення та обмеження обсягу прихованих каналів є складним але необхідним кроком для захисту майбутнього комп’ютерних мереж.

Приховані канали передачі інформації у мережі можна вважати певною формою стеганографії. Стеганографія потребує звуковий, візуальний чи текстовий контент для прикриття. У прихованих каналах таким прикриттям слугують мережні протоколи. Деякі мережні протоколи присутні всюди, що робить приховані канали широкодоступними. Вони можуть використовуватись навіть, якщо класична стеганографія не може бути застосована.

1.1 Визначення поняття прихованого каналу передачі інформації в мережному протоколі

Відкритий канал — це канал зв'язку всередині комп'ютерної системи або мережі, призначений для авторизованої передачі даних. Тоді прихований канал — це будь-який канал зв'язку, що може бути використаний процесом для передачі інформації таким чином, що порушує політику безпеки систем [1]. Будь-який спільний ресурс потенційно можна використовувати як прихований канал. Таємні канали можуть бути розділені за визначенням TCSEC на канали зберігання і таймінгу. У випадку з каналами зберігання, як правило, один процес записує інформацію (прямо або опосередковано) на спільний ресурс, в той час як інший процес читає її з нього. Канал таймінгу — це по суті будь-яка техніка, що передає інформацію за часом подій, в цьому випадку процес прийому потребує годинника. Особливий канал таймінгу несе дані за допомогою підрахунку випадків певних подій [2]. Крім того, канали таймінгу можуть бути активними, якщо вони генерують додатковий трафік або пасивними, якщо вони маніпулюють термінами існуючого трафіку. Як і будь-який інший канал зв'язку, прихований канал може бути шумним або безшумним. За кількістю інформаційних потоків між відправником і приймачем — кілька або один, існують агреговані і неагреговані таємні канали [3]. За наявністю або відсутністю проміжного вузла в комунікації, приховані канали можуть бути непрямі або прямі. Таємні канали вивчаються як частина стеганографії, а різні стеганографічні методи, що використовуються в телекомунікаційних мережах, відомі як мережна стеганографія.

Інколи дві сторони хочуть спілкуватися конфіденційно і непомітно через небезпечний канал. Цей канал може контролюватися пасивно — трафік відстежується на предмет несанкціонованої комунікації, або активно — зі зміною повідомлень з метою усунення будь-якої форми

прихованого зв'язку. Сіммонс ввів термін subliminal channel — варіант прихованого каналу, який використовує криптографічний алгоритм або протокол для приховування повідомлень. Композиція з прихованих каналів у криптографічному алгоритмі та в мережному протоколі утворює гібридний канал. [4]

Таємні канали можуть бути проаналізовані за загальною кількістю бітів прихованої інформації, що передаються протягом секунди (Raw Bit Rate — RBR), або за загальною кількістю бітів прихованої інформації, що передаються з одиницею даних протоколу (наприклад, Packet Raw Bit Rate — PRBR) [5]. В ідеалі, якщо немає втрат пакетів, ємність деяких каналів зберігання може бути виражена як C = PRBR × N, де N - кількість пакетів, що надсилаються за одну секунду.

1.2 Історія розвитку концепції прихованих каналів

Термін «прихований канал» був вперше використаний та описаний Батлером Лемпсоном у 1973 в своїй праці «A Note on the Confinement Problem» [6].

Мережні протоколи ідеально підходять для приховування даних. Більшість мережних методів стеганографії для каналів зберігання базуються на факті, що в заголовках протоколів існують опціональні поля, які можуть бути використані для таємного передавання даних і створення прихованих каналів.

Манфред Вольф [7] запропонував використовувати для створення прихованих каналів зарезервовані поля, поле розширення пакету або невизначені поля кадрів мереж стандарту IEEE 802.2, 3, 4 та 5. Тим часом Теодор Хандел і Максвел Сендворд [8] дійшли висновку, що зарезервовані та невизначені поля заголовків інших протоколів також можна експлуатувати з такою метою.

Однак для створення прихованих каналів передачі у мережних протоколах підходять не лише необов’язкові поля. Іншим засобом є поля, заповнені «випадковими» даними, такі як IP Identification (ID) або TCP Initial Sequence Number (ISN). Проте не варто бездумно наповнювати їх даними, оскільки це легко виявляється пасивним контролем каналу, тому що ці поля демонструють достатню структуру і неоднорідність, щоб ефективно та надійно виявляти PDU (Protocol Data Unit — одиниця даних протоколу) зі зміненими полями.

Робота Крейга Роуленда [9] є першим доказом можливості створення та експлуатації прихованих каналів у наборі протоколів TCP/IP, шляхом їх конкретної реалізації. З точки зору зловмисника, канали зберігання є кращими, ніж канали таймінгу, через проблеми синхронізації, що присутні в останніх, їх складність використання та значно меншу пропускну здатність у порівнянні з каналами зберігання.

1.3 Причини та способи використання прихованих каналів

Приховані канали передачі інформації в мережних протоколах можуть використовуватися для широкого спектру задач: координації DDOS (Distributed Denial Of Service) атак, поширення комп'ютерних вірусів і хробаків, таємного зв'язку між терористами і злочинцями, безпечних комунікацій з метою управління мережею [10], обходу брандмауера організації, передачі даних аутентифікації [11] (наприклад, port knocking), обходу обмеження у використанні Інтернету в деяких країнах (Infranet [12], Collage [13], FreeWave [14]), покращення безпеки [15] чи забезпечення QoS (Quality of Service) [16] VoIP-трафіку, маркування мережних потоків (RAINBOW [17], SWIRL [18]), відстеження зашифрованого трафіку атаки або відстеження анонімних peer-to-peer VoIP-дзвінків [19, 20] тощо.

Широкий спектр осіб та угрупувань з різних причин використовує приховані канали для комунікації та координування дій. Зазвичай це умотивовано протистоянням двох сторін (влада – злочинці, хакери – компанії). Зрозуміло, що кожна зі сторін має бажання приховати свої комунікації. Однак просте шифрування не забезпечить неможливість відслідковування шаблонів комунікації. Часто навіть сам факт передачі інформації може викрити багато таємниць. [21]

Після того як хакери скомпрометували комп'ютерну систему, вони зазвичай намагаються викрасти дані або ж завантажити свої інструменти для подальших злочинних дій. Така поведінка генерує мережний трафік, який без належного маскування моментально приверне увагу системних адміністраторів. Як наслідок вони одразу дізнаються про проникнення у систему. Для вивантаження даних не потребує навіть зламу самого комп'ютера. Достатньо скомпрометувати пристрій вводу (наприклад клавіатура) або ж програмне забезпечення (наприклад веб браузер). [22]

Важливо зазначити, що навіть пересічний працівник може бути зацікавлений у використанні прихованих каналів для уникнути блокування брандмауером компанії, щоб отримати доступ до певних ресурсів в інтернеті. Також спроби урядів деяких країн обмежити свободу слова спричинили появу пропозицій щодо використання прихованих каналів передачі інформації у мережі для обходу даних заходів. Крім того приховані канали можуть застосовуватися для захисту інформації, що передається, при забороні використання стійких криптографічних алгоритмів для шифрування.

Адміністратори мережі можуть використовувати приховані канали для безпечного передавання інформації пов'язаної з управлінням мережею з метою заховати її від зловмисників. Ханіпоти (комп'ютерні системи, що є приманкою для хакерів) також можуть використовувати приховані мережні канали для передачі даних логування в реальному часі таємно від злочинця.

Комп’ютерні віруси та хробаки можуть використовувати приховані канали для непомітного поширення та обміну інформацією, призначеною для аналізу на багатьох хостах.

Приховані канали також можуть використовуватись для передавання даних автентифікації. Існують техніки, що дозволяють авторизованим зовнішнім користувачам отримувати доступ до відкритих портів брандмауера, хоча для інших користувачів ці порти будуть відображатися закритими. Одна конкретна техніка, що називається Port Knocking, використовує приховані канали для надсилання даних автентифікації. Дослідники також розробили технологію для відслідковування зворотного шляху пакету за допомогою прихованих каналів. Вона забезпечує наступні вузли зв’язку інформацією про попередні. Таким чином можна захиститися від атаки на відмову в обслуговуванні (англ. Denial-of-service attack) фільтруючи трафік у ближчих до границі мережі вузлах або ж навіть ізолюючи атакуючого. [23]

Висновок до розділу 1:

Прихований канал — це комунікаційний канал, що передає інформацію способом, який не був для цього призначений. Із розвитком мережевих технологій розвивалася й концепція прихованих каналів передачі інформації. Використання прихованих каналів не обмежується злочинними діями. Крім поширення зловмисного програмного забезпечення та викрадення даних, вони можуть застосовуватися для віддаленого керування та адміністрування. Пересічні ж користувачі можуть отримувати за допомогою прихованих каналів доступ до заблокованих ресурсів. Систематизація та аналіз відомостей дозволили поглибити знання про явище прихованого каналу передачі інформації у мережних протоколах, що допомогло краще зрозуміти поставлені у дослідженні завдання.

2 КЛАСИФІКАЦІЯ ПРИХОВАНИХ КАНАЛІВ ТА СТРАТЕГІЇ ЗАХИСТУ

Найбільш поширений підхід до створення прихованих каналів передбачає включення необхідної інформації в протоколи з полями, достатньо великими для забезпечення суттєвої пропускної здатності. Для цього варто вибрати протоколи вищого рівня, проходження яких через брандмауери, як правило, буде дозволено. Корисне навантаження може бути закодоване у невикористовуваних або ж зарезервованих бітах заголовка кадру чи пакету. Це стає ще простішим, якщо стандарти протоколу не регламентують їх значення або отримувачі не перевіряють їх відповідність стандарту. Наприклад, ICMP є хорошим вибором через майже стандартний дозвіл вихідних echo request та вхідних echo reply. [24] Це відкриває грандіозні можливості для прихованих каналів, оскільки корисне навантаження протоколу може містити будь-які дані в будь-якій кількості. Протокол DNS також є чудовим варіантом. [25]

Після вибору протоколу користувач повинен налаштувати локального клієнта та віддаленого слухача, котрий зможе “витягти” тунельний трафік і далі переслати його в інше місце, якщо це необхідно. Щоб запобігти перехопленню інформації, інструменти тунелювання часто використовують автентифікацію (як правило, на основі пароля) для керування доступом до віддаленого слухача. Щоб запобігти виявленню, деякі інструменти використовують методи шифрування та обфускації, приховуючи вміст трафіку під час передачі, щоб обійти аналіз IDS (розшифрувати) на основі вмісту. Інструменти тунелювання також забезпечують можливість уповільнення швидкості передачі пакетів, щоб зменшити обсяг підозрілого трафіку, що протікає мережею, задля зменшення ризику виявлення. Однак це також уповільнює комунікацію і робить тунель менш ефективним, тому існує очевидний компроміс між непомітністю і зручністю використання. [26]

Розуміння існуючих прихованих каналів передачі інформації є надзвичайно важливим для розробки заходів протидії їх використанню.

2.1 Класифікація прихованих каналів за рівнем стеку TCP/IP

Далі буде наведено всебічне дослідження прихованих каналів у стеку протоколів TCP/IP, класифікованих за рівнем і протоколом, з описом переваг, недоліків та методами захисту, якщо такі існують.

Для кращого розуміння класифікації слід розглянути стек протоколів TCP/IP. Він ділиться на 4 рівні: прикладний (application), транспортний (transport), мережевий (internet) та канальний (link). Як і в моделі OSI, дані нижніх рівнів інкапсулюють блоки даних верхніх рівнів (рисунок 2.1).

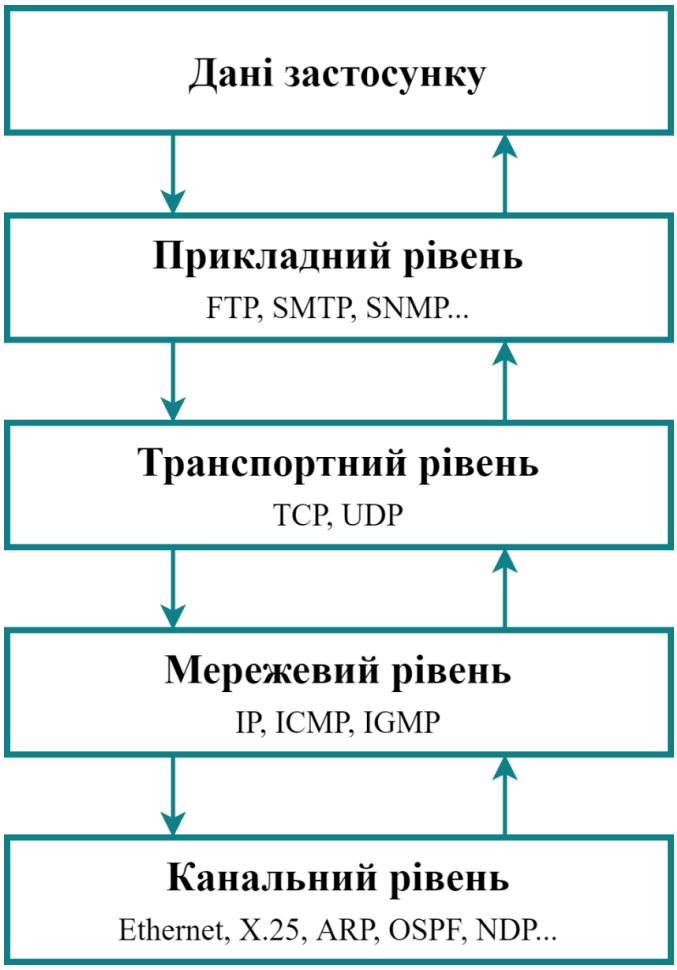

Рисунок 2.1 – Стек TCP/IP

2.1.1 Приховані канали мережевого рівня

Internet Protocol (IP) є основним протоколом у стеку протоколів TCP/IP, який працює на мережевому рівні. IP інкапсулює отримані сегменти транспортного рівня в пакети з IP заголовками і доставляє їх до заданого місця призначення за допомогою IP адрес. IP існує в двох версіях - версія 4 (IPv4) і версія 6 (IPv6). На рисунках 2.2, 2.3 і 2.4 узагальнено інформацію про приховані канали в IP протоколі: переваги, недоліки і механізми захисту.

Paper	Year	Method	PRBR	Advantages	Disadvantages	Defence
Rowland [98]	1997	*Identification* field	16	Big covert rate	Easy elimination	Traffic normalizer and active warden
B0CK [120]	2000	*Source address* + IGMP	32	Easy deployment	Easy elimination	Egress filtering
Abad [1]	2001	*Header checksum*	16	Deployment in IP, TCP, UDP	Need to know original *TTL*	
Ahsan et al [2]	2002	*DF* field	1		Easy elimination	Traffic normalizer and active warden
Ahsan [3]	2002	*DF* and *Identification* fields	17	Big covert rate	Easy elimination	Traffic normalizer and active warden
Ahsan [3]	2002	*Version*, *IHL* and *Identification* fields	8	Big covert rate	Packet without options	Traffic normalizer and active warden
Cauich et al [20]	2002	*Identification* and *Fragment Offset* fields	29	Big covert rate	Only between neighbouring nodes and not fragmented packet	Traffic normalizer and active warden
Qu et al [95]	2004	*TTL*	1	Hard detection	Without normal initial TTL	Active warden [125]
Zander et al [125]	2006	*TTL*	1	Normal initial TTL	Noisy channel	Active warden [125]
Trabelsi et al [114, 115]	2007	Record route options	320	Big covert rate		
Mazurczyk et al [79]	2009	Predefined number of fragments	1	Big covert rate	Detectable statistical patterns	Statistical tests based on number of fragments, or reassemble in intermediate node [79, 80]
Mazurczyk et al [79]	2009	Modulating the values in *Fragment Offset* field	$N_F - 1$	Big covert rate	Detectable statistical patterns	Statistical tests based on fragments sizes, or reassemble in intermediate node [79, 80]
Mazurczyk et al [79]	2009	Steganogram in fragment payload	$N_F \cdot F_s$	Use of legitimate fragments	Differentiating covert fragments	Reassemble in intermediate node [79, 80]
Mazurczyk et al [79, 80]	2009	Probe messages in PMTUD	up to MTU	Big covert rate	Need to know MTU	Comparing probe messages [79, 80]

Рисунок 2.2 – Приховані канали зберігання в IPv4 [24]

Paper	Year	Method	Advantages	Disadvantages	Defence
Padlipsky et al [90]	1978	Send or not IP packet in interval	Good synchronization	Abnormal shape and regularity	ϵ-similarity test [17] EN and CCE test [41]
Servetto et al [103]	2001	Phantom packets		Need of synchronization	
Ahsan et al [2] Ahsan [3]	2002	Packet sorting		External reordering effects	
Galatenko et al [37]	2005	Packet sorting with ordered dest. addresses	Different error rate with sequence length	External reordering effects	
Shah et al [104]	2006	IPDs	Similar shape and regularity as normal traffic	Needs access to the link between keyboard and computer	EN test [41]
Cabuk [18]	2006	IPDs	Real saved IPDs samples	Abnormal regularity	CCE test [41]
Allix [4]	2007	Sending more or less data than λ	Easy deployment		
Gianvecchio et al [40]	2008	IPDs	Similar shape and regularity as normal traffic	Bad synchronization and easy detection for non iid normal IPDs	Auto-correlation test [127] CCE test [41]
Sellke et al [102]	2009	IPDs	Non-detectable if normal IPDs are iid	Bad synchronization and easy detection for non iid normal IPDs	Auto-correlation test [127]
Mazurczyk et al [79]	2009	Different rates for fragments	No need of synchronization		Reassemble in intermediate node [79, 80]
Mazurczyk et al [79]	2009	Phantom fragments	No need of synchronization	Modified receiver	
Mazurczyk et al [79]	2009	Fragment sorting	No need of synchronization	External reordering effects	
El-Atawy et al [31]	2009	Packet sorting	Error detection and correction		
Zander et al [127]	2011	IPDs	Better synchronisation and stealthiness than [40] and [102]	Reduced robustness against network jitter	

Рисунок 2.3 – Приховані канали таймінгу в IPv4 [24]

Paper	Year	Method	Type	PRBR	Advantages	Disadvantages	Defence
Graf [43]	2003	*Options* in Destination Options header	storage	max field length	Big covert rate	Easy elimination	Traffic normalizer and active warden
Lucena et al [66]	2005	*Traffic class*	storage	8	Big covert rate	Intermediate node can change the field	Stateful Active Warden [66]
Lucena et al [66]	2005	*Flow Label*	storage	20	Big covert rate	Many flow labels between two hosts	Stateful Active Warden [66]
Lucena et al [66]	2005	*Source Address*	storage	128	Big covert rate	Easy elimination	Egress filtering
Lucena et al [66]	2005	*Hop limit*	storage		Easy deployment	Noise channel	Stateful Active Warden [66]
Lucena et al [66]	2005	*Payload Length*	storage	up to $(2^{16} - \|LegP\|)$B	Legitimate packet	Failure in the ICV calculation	
Lucena et al [66]	2005	*Option Data Length* and *Option Data* in the Hop-by-hop options header	storage	up to 2038B	Very high rate	Failure in the ICV calculation	Stateless Active Warden [66]
Lucena et al [66]	2005	*Reserved* in Routing header	storage	4B	Big covert rate	Failure in the ICV calculation	Stateless Active Warden [66]
Lucena et al [66]	2005	fabricating addresses in Routing header	storage	up to 2048B	Very high rate	Many different routing headers between two nodes	Stateful Active Warden [66]
Lucena et al [66]	2005	*Option Data Length* and *Option Data* in the Destination options header	storage	up to 2038B	Very high rate	Failure in the ICV calculation	
Lucena et al [66]	2005	False padding in the Destination options header	storage	up to 256B	Very high rate	Failure in the ICV calculation	Stateful Active Warden [66]
Lucena et al [66]	2005	Two *Reserved* fields in Fragment header	storage	8 + 2	Big covert rate	Easy elimination	Stateless Active Warden [66]
Lucena et al [66]	2005	*Next Header* field in Fragment header	storage	8	Big covert rate	Different Next Header values among fragments of same packet	Stateful Active Warden [66]
Lucena et al [66]	2005	fake fragment	storage	up to 64KB/ fragment	Very high rate	Failure in the ICV calculation	Anomalies of single, unrelated fragments [79, 80]
Lucena et al [66]	2005	*Reserved* in Authentication header	storage	2B	Big covert rate	Easy elimination	Stateless Active Warden [66]
Lucena et al [66]	2005	Fake Authentication header	storage	up to 1022B	Very high rate	Sequence number values do not increase monotonically	Stateful Active Warden [66]
Lucena et al [66]	2005	Fake ESP header	storage	up to 1022B	Very high rate	Sequence number values do not increase monotonically	Stateful Active Warden [66]
Lucena et al [66]	2005	False padding value in ESP header	storage	up to 255B	Very high rate		
Mazurczyk et al [79, 80]	2009	PLPMTUD using RSTEG	storage	up to MTU	Very high rate	Use of normal network retransmission rate	Comparing probe messages [79, 80]

Рисунок 2.4 – Приховані канали в IPv6 [24]

Internet Control Message Protocol (ICMP) — це ще один протокол без встановлення з'єднання на мережевому рівні, який використовується для передачі повідомлень про помилки та іншої інформації між вузлами. Повідомлення ICMP надсилаються інкапсульованими в IP-пакетах. Існує 14 (і 16 застарілих) різних типів повідомлень ICMP, які зазвичай містять

лише перші 4 байти з 8-байтного заголовку ICMP. Приховані канали в ICMP будуть працювати для будь-якого мережевого пристрою, що не фільтрує вміст трафіку ICMP Echo, і є дуже простими у реалізації. Інформація про ці канали зібрана на рисунку 2.5.

Paper/tool	Year	Method	Type	PRBR	Advantages	Disadvantages	Defence
Loki [22, 23]	1996	Payload of ICMP Echo Request and ICMP Echo Reply packets	storage	up to 24B, 56B or more	IP-over-ICMP tunnel and command execution	Easy elimination	Blocking and [106] stateless model
Ahsan [3]	2002	*Reserved* field in ICMP Router Solicitation Message	storage	32	Easy deployment	Easy elimination	Traffic normalizer and active warden
ICMP-Chat [84]	2003	Payload of ICMP Echo Request and ICMP Echo Reply packets	storage	up to 24B, 56B or more	IP-over-ICMP tunnel, encryption	Easy elimination	Blocking and [106] stateless model
Skeeve [128]	2004	Payload of ICMP Echo Request and ICMP Echo Reply packets	storage	up to 24B, 56B or more	IP-over-ICMP tunnel and command execution	Easy elimination	Egress filtering, blocking and [106] stateless model
ICMPTX [113]	2005	Payload of ICMP Echo Request and ICMP Echo Reply packets	storage	up to 24B, 56B or more	IP-over-ICMP tunnel and command execution	Easy elimination and bad implementation	Blocking and [106] stateless model
Ping tunnel [110]	2005	Payload of ICMP Echo Request and ICMP Echo Reply packets	storage	up to 24B, 56B or more	TCP-over-ICMP tunnel, reliable	Easy elimination	Blocking and [106] stateless model
V00d00n3t [86]	2006	Payload of ICMP Echo Request and ICMP Echo Reply packets	storage		IPv6-over-ICMPv6 tunnel and different options		

Рисунок 2.5 – Приховані канали в ICMP [24]

Internet Group Management Protocol (IGMP) — це протокол керування груповою (multicast) передачею даних в мережах, що базуються на протоколі IP. Перший прихований канал для IGMP був запропонований Камраном Ахсаном в 2002 [27]. Карлос Скотт [28] запропонував використовувати 8-бітні та 16-бітні зарезервовані поля у IGMPv3 Membership Report, яким зазвичай встановлюється нульове значення та ігнорується приймачем.

Dynamic Host Configuration Protocol (DHCP) — це ще один протокол мережевого рівня, який дозволяє серверу автоматично призначати IP адресу комп'ютеру з певного діапазону, налаштованого для певної мережі.

Рубен Ріос [29] запропонував і реалізував наступні надійні приховані канали в інструменті HIDE DHCP:

- підміна значення 32-бітного поля XID, яке генерується клієнтом випадковим чином. Це скритно, але пропускна здатність обмежена
- підміна значення поля SECS, для передачі одного біта
- підміна значення останніх 10B поля CHADDR, коли використовується 48-бітна MAC адреса Ethernet
- допис прихованої інформації після NULL в кінцях строк 64-байтового поля SNAME і 128-байтового FILE

Address Resolution Protocol (ARP) — це протокол для визначення MAC адреси за IP адресою, наприклад, адресою Ethernet. Ліпінг Джи [30] пропонував використовувати останні 8 бітів поля Target protocol address для перенесення секретного повідомлення.

2.1.2 Приховані канали транспортного рівня

Transmission Control Protocol (TCP) і User Datagram Protocol (UDP) — це два протоколи, які працюють на транспортному шарі. TCP забезпечує послуги надійного зв'язку зі встановленням з'єднання. У той же час UDP не потребує встановлення з’єднання, хоч і є менш надійним. Приховані канали передачі інформації для цих протоколів можна побачити на рисунках 2.6 і 2.7 відповідно.

Paper/tool	Year	Method	Type	PRBR per segment	Advantages	Disadvantages	Defence
Covert_TCP [98]	1997	*ISN* and *Acknowledge Sequence Number* fields	storage	64	Easy deployment	Differentiation from normal ISNs	SVM [109] and anomaly tests [85]
Ahsan [3]		*Urgent Pointer*					
Hintz [45]	2002	field	storage	16			
Abad [1]	2001	*Header checksum*	storage	16	Deployment in IP, TCP, UDP	Need to know original *TTL*	
Giffin et al [42]	2002	TCP timestamps	timing	1	No need of synchronisation	Some timestamps are skipped	Ratio of used different timestamps and total number of timestamps [45]
NUSHU [99]	2004	*ISN*	storage	32	Passive channel Indistinguishable ISNs	Differentiation from normal ISNs	Anomaly tests [85] and neural networks [117]
Lantra [85]	2005	*ISN*	storage	32			
Chakinala et al [21]	2006	Segment reordering	timing	$log_2 n!$			
Allix [4]	2007	*Reserved*	storage	4	Easy deployment	Easy elimination	Traffic normalizer and active warden
Cloak [69]	2007	*N* packets over *X* TCP flows	timing		Reliable	Restricts the decoder location	
TCP-Script [70]	2008	TCP bursts	timing		Robust against network jitter, packet loss and reordering	Differentiable from normal traffic	Deviation score for the burst size, entropy of burst size, and inter ACK-data delay [70]
CLACK [71]	2009	*Acknowledge Sequence Number*	storage	32	Reliable and resilience to adverse network conditions		
RSTEG [81, 82]	2010	retransmissions	storage	max IP payload length	Big covert rate	Use of normal network retransmission rate	Passive warden [81]
ACKLeaks [72]	2011	TCP ACK packets	storage		Do not modify any packets	Errors from ACK losses	

Рисунок 2.6 – Приховані канали в TCP [24]

Paper/tool	Year	Method	Type	PRBR per segment	Advantages	Disadvantages	Defence
Abad [1]	2001	*Header checksum*	storage	16	Deployment in IP, TCP, UDP	Need to know original *TTL*	
Fisk et al [34]	2002	Presence/absence of *Checksum*	storage	1	Easy deployment	Easy detection	Active warden [34]
Thyer [112]	2008	*Source address, Length* and *Checksum*	storage	up to 6B	Easy deployment	Easy elimination	Active warden and egress filtering [34]

Рисунок 2.7 – Приховані канали в UDP [24]

Secure Sockets Layer (SSL) (версії 1.0, 2.0 і 3.0) і його наступник Transport Layer Security (TLS) (версії 1.0, 1.1 і 1.2) - це протоколи шифрування/розшифровки даних при передачі для захищеного зв'язку між двома сторонами. Гібридний прихований канал, що є поєднанням простого

мережевого прихованого каналу в TCP і прихованого каналу в криптографічному алгоритмі протоколів SSL/TLS, описаний в працях Джибі Абрахама [31, 32].

2.1.3 Приховані канали прикладного рівня

Для створення прихованих каналів передачі інформації можна використовувати також деякі протоколи прикладного рівня стеку TCP/IP, такі як: HTTP, FTP, DNS, RTP, RTCP, SIP, SDP і т.д.

Paper/tool	Year	Method	Type	PRBR	Advantages	Disadvantages	Defence
Reverse WWW Shell tool [118]	1999	HTTP request/response	storage	varies	Easy deployment	Pre-configured master	Analysing HTTP traffic [14, 88]
Corkscrew [89]	2001	HTTP request/response	storage	varies	SSH over HTTP tunnel	Suspicious SSH traffic	Analysing HTTP traffic [14, 88]
Bowyer [15]	2002	URL parameters or body of GET request	storage	varies	Communication through firewalls	Easy detection if data hide in GET body	Analysing HTTP traffic [14, 88]
Dyatlov et al [30]	2003	HTTP request/response	storage	varies	Very high rate	Easy detection if data hide in GET body	Analysing HTTP traffic [14, 88]
Muted Posthorn [10]	2003	Redirection	storage	up to 1024B	Very high rate		Analysing HTTP traffic [14, 88]
Muted Posthorn [10]	2003	*Referer*	storage	up to 1024B	Very high rate		Analysing HTTP traffic [14, 88]
Muted Posthorn [10]	2003	*Set-Cookie*	storage	up to 4096B	Very high rate		Analysing HTTP traffic [14, 88]
Muted Posthorn [10]	2003	HTML elements and Active contents	storage				Analysing HTTP traffic [14, 88]
Alman [5]	2003	CONNECT method	storage		Establishing VPN		
Eßer et al [32]	2005	Delaying or not a response	timing	1			
HTTunnel [63]	2005		storage		TCP and UDP over HTTP tunnel		Analysing HTTP traffic [14, 88]
Kwecka [61]	2006	consequent linear white space characters (CLRF, SP, HT])	storage	up to 8190B	Very high rate	Easy detection	Inline Filtering Agent [61]
Wondjina [119]	2006	HTTP Entity tags	storage				
Wondjina [119]	2006	*ContentMD5* header	storage	128			
Castro [19]	2006	cookies	storage				
Dunkan et al [29]	2010	*Access-Control-Allow-Origin* and *Content-Location* header	storage				
Dunkan et al [29]	2010	*Date* and *Last-Modified*	storage			*Date* can be rewritten	StegoProxy [13]

Рисунок 2.8 – Приховані канали в HTTP [24]

HyperText Transfer Protocol (HTTP) слугує для передачі та відображення вебсторінок (текстових файлів з розміткою HTML) та файлів. Навіть організації з найбільшою кількістю заборон щодо трафіку зазвичай дозволяють HTTP-трафік, тому він є хорошим засобом для створення прихованих каналів (рисунок 2.8).

File Transfer Protocol (FTP) також інколи використовується для прихованих каналів передачі інформації. Було запропоновано два способи створення прихованих каналів в FTP [33]. Перший кодує приховані біти безпосередньо в команди FTP, тому, якщо є N команд, кожна команда буде представляти $\log_2 N$ біти. Другий залежить від кількості FTP NOOP команд, які надсилаються під час простою. Кількість відправлених NOOP команд дорівнює цілому значенню прихованих даних. Перед надсиланням нового значення слід надіслати команду ABOR. Для FTP є нормальним відправляти команди NOOP або ABOR постійно, щоб запобігти входженню контрольних з'єднань в стан простою.

Domain Name System (DNS) дуже добре підходить для створення прихованих каналів для тунелювання інших протоколів, наприклад IP, TCP або UDP. Особливо цікавими є записи NS, CNAME і TXT довжиною до 255B, а також експериментальний NULL-запис довжиною до 65536B (300B-1200B в реалізаціях). Узагальнення прихованих каналів, що використовують DNS, наведено на рисунку 2.9.

Paper/tool	Year	Method	Type	PRBR	Advantages	Disadvantages	Defence
NSTX [100]	2002	TXT records	storage	up to 255B	IPv4-over-DNS	Splits IP packets	Blocking by forbidding non-compliant names in queries [87]
DNSCat [94]	2004	CNAME records	storage	up to 255B	IPv4-over-DNS	Splits IP packets	Limit the DNS bandwidth
OzymanDNS [55]	2004	TXT records	storage	up to 255B	TCP or SSH-over-DNS and supports of EDNS0	Must provide a reliable communication	Blocking rarely used records or extensions [87]
Anonymous [8]	2005	Negative caching	timing	1	No need to maintain or run a DNS server	Choosing good subdomains	
DNS2TCP [25]	2008	TXT records	storage	up to 255B	TCP or SSH-over-DNS	Must provide a reliable communication	Blocking by forbidding non-compliant names in queries [87]
TUNS [87]	2009	CNAME records	storage	up to 255B	IPv4-over-DNS, doesn't split IP packets	Slower for perfect network	Limit the DNS bandwidth
Iodine [60]	2010	Download - NULL records (or TXT, SRV, MX, CNAME, A records), upload - *Domain Name*	storage	Download - up to 1200B, upload - up to 255B	IPv4-over-DNS and supports of EDNS0	Splits IP packets	Blocking rarely used records or extensions [87]

Рисунок 2.9 – Приховані канали в DNS [24]

Ще однією цікавою областю для розгортання прихованих каналів є програми, що працюють в режимі реального часу через IP, як, наприклад, Voice over IP (VoIP), онлайн багатокористувацькі ігри, потокове передавання в прямому ефірі A/V тощо. У цих програмах, як правило, аудіо та/або відео передається за допомогою відокремлених потоків Real-Time Transport Protocol (RTP), за підтримки його супутника RTP Control Protocol (RTCP). Одна частина RTP працює на транспортному рівні над UDP, а друга - на прикладному. З іншого боку, VoIP має фазу сигналізації на початку (перед передачею звуку за допомогою RTP), яка відбувається за допомогою деякого протоколу сигналізації, наприклад, Session Initiation Protocol (SIP). Робота SIP зазвичай супроводжується деяким протоколом для опису мультимедійних сеансів, як, наприклад, Session Description Protocol (SDP). Можливі приховані канали в RTP та RTCP описані на рисунку 2.10.

Paper/tool	Year	Method	Type	PRBR	Advantages	Disadvantages	Defence
Mazurczyk et al [78]	2008	*Padding* field	storage	8		Easy elimination	Traffic normalizer and active warden
Mazurczyk et al [78]	2008	Extension header	storage	varies	Big covert rate	Easy elimination	Traffic normalizer and active warden
Mazurczyk et al [78]	2008	*Sequence Number* field	storage	16	Hard detection	Only for first RTP packet	
Mazurczyk et al [78]	2008	*Timestamp* field	storage	32	Hard detection	Only for first RTP packet	
Mazurczyk et al [78]	2008	*Timestamp* field	timing	1	No need of synchronisation	Some timestamps are skipped	Ratio of used different timestamps and total number of timestamps [45]
Mazurczyk et al [78]	2008	*NTP Timestamp* in RTCP Report blocks	timing	1	No need of synchronisation	Some timestamps are skipped	Ratio of used different timestamps and total number of timestamps [45]
Mazurczyk et al [78]	2008	in RR and SR in RTCP	storage	up to 160	Big covert rate		
Mazurczyk et al [78]	2008	*authentication tag*	storage	80	Hard detection		Stripping this field [78]
LACK [78]	2008	intentionally delayed packets	timing		Receiver discard packets	Do not exceed accepted level of packet loss	Active warden [78]
Bai et al [9]	2008	*interarrival jitter* in RTCP	storage	32	Use of jitter statistics		
Lizhi et al [64]	2012	Send RTP or (RTP, RTCP) packet	storage	32	Robust with many 0s		

Рисунок 2.10 – Приховані канали в RTP та RTCP [24]

Secure Shell (SSH) — криптографічний протокол клієнт-сервера для безпечного обміну даними, віддаленого виконання команд та інших захищених мережевих служб між двома комп'ютерами, які з'єднуються через захищений канал через незахищену мережу. Приховані канали, що використовують SSH, можна побачити на рисунку 2.11.

Paper/tool	Year	Method	Type	PRBR	Advantages	Disadvantages	Defence
Lucena et al [67]	2004	*MAC* field	storage	up to 160	Randomness simulation	Non reliable	Wrong recomputed MAC value
Lucena et al [67]	2004	Beginning of payload	storage	up to 20B	Big covert rate	Different packet length distribution	Packets length analysis
Perkins [67]	2005	*Random padding* field	storage	up to 255B	Very high rate	Not so random values	Statistical tests

Рисунок 2.11 – Приховані канали в SSH [24]

2.2 Стратегії виявлення та знешкодження прихованих каналів

Приховані канали передачі є предметом постійної гонки озброєнь. З одного боку, зловмисники (або навіть звичайні користувачі) прагнуть таємно передавати інформацію. Для цього хакери розробляють нові інструменти і, з метою підвищення складності виявлення каналів, а також ефективності при надсиланні та отриманні повідомлень, покращують їх з часом. З іншого боку, спеціалісти з кібербезпеки постійно вдосконалюють сигнатури IDS і розробляють нові способи виявлення прихованих каналів. Як тільки інструмент прихованої передачі інформації стає популярним, методи виявлення його використання також можуть бути включені в локальні правила IDS або брандмауера.

Стратегії виявлення прихованих каналів полягають у наступному:

- статистичному аналізі трафіку (пошук нетипових обсягів даних невідомого призначення)
- пошуку недоречних значень у полях протоколів (наявність інформації у зарезервованих чи рідко використовуваних полях, невипадкові числа у полях з рандомізованими значеннями)
- пошук впізнаваного вмісту у несподіваних місцях (магічні числа, текст ASCII або інші ознаки значущих даних у полях, які, зазвичай, не містять дані такого типу)

Після того, як прихований канал був знайдений, існують такі варіанти дій [34]:

- усунення каналу
- обмеження пропускної здатності каналу
- аудит каналу
- документування каналу

Найкращим варіантом є усунення можливості використання прихованого каналу, оскільки навіть канали з дуже низькою пропускною здатністю можуть бути успішно використані. Однак спроба видалення всіх

прихованих каналів призводить до дуже низької ефективності роботи системи, так як приховані канали часто можуть бути видалені тільки шляхом заміни автоматизованих процедур [35]. Крім того більшість методів усунення прихованих каналів мають евристичний характер. Тому повне знешкодження усіх прихованих каналів системи вважається неможливим [34, 36].

Якщо канал не може бути знешкоджений, його пропускна здатність повинна бути максимально зменшена. Прийнятний розмір пропускної здатності каналу залежить від обсягу потенційного витоку даних, що є критичним. Наприклад, якщо канал настільки малий, що секретна інформація застаріє до того як достатній для її використання обсяг буде ексфільтровано, то пропускна здатність каналу є прийнятною. Обмеження пропускної здатності каналу часто є проблематичним, оскільки означає уповільнення системних механізмів або введення шуму, які обмежуватимуть продуктивність системи.

Будь-які приховані канали, які не можна знешкодити, підлягають аудиту. Аудит виступає стримуючим фактором для можливих користувачів каналу. В разі неможливості аудиту чи настільки низької пропускної здатності каналу, що ним можна знехтувати, необхідно принаймні задокументувати їх існування (наприклад, у специфікації протоколу), щоб кожен усвідомлював їх існування і потенційну загрозу.

Існує декілька стратегій знешкодження прихованих каналів:

- захист мережі
- нормалізація трафіку
- захист хостів

Захист мережі для протидії прихованим каналам полягає у блокуванні протоколів та портів, що використовуються прихованими каналами. Наприклад, в наш час ICMP блокується багатьма брандмауерами, що робить такі підходи до створення прихованих каналів

як Loki [37] неефективними. Очевидно, що в Інтернеті деякі протоколи не можуть бути заблоковані, оскільки вони є основою його функціонування (наприклад, IP, TCP, DNS), або тому, що їх послуги занадто важливі (наприклад, email, Web). Однак у закритій мережі протоколи, вразливі до використання прихованими каналами, можуть бути заблоковані або замінені версіями з меншою кількістю можливих прихованих каналів чи прихованими каналами зі значно урізаною ефективністю.

Багато прихованих каналів, описаних раніше, можуть бути усунені за допомогою нормалізації заголовків протоколів, та розширень пакетів. Нормалізація трафіку може бути виконана кінцевими хостами або мережними пристроями, такими як брандмауери або проксі-сервери. Приховані канали, що використовують необов’язкові або зарезервовані біти і заповнення можуть бути легко усунені зашумленням або приведенням до нульового значення полів. Невідомі розширення заголовка можуть бути видалені. Деякі таємні канали використовують той факт, що певні поля заголовків не завжди використовуються. Якщо використання цих полів індексується іншими полями заголовків, то цей факт може бути використаний для нормалізації. Наприклад, встановлювати нульове значення IP ID, якщо встановлено біт DF; встановлювати нульовим заначення Urgent Pointer, якщо біт URG не встановлено; встановлювати Fragment Offset нульовим, якщо встановлено біт DF. Крім того, слід переконатися, що чексуми завжди використовуються та є правильними. Ряд інших полів заголовка можна переписати керуючись певними припущеннями. Наприклад, встановлювати біт DF та приводити значення IP ID і Fragment Offset до нуля, якщо розмір пакету менше розміру MTU (за умови відомого MTU); перезаписувати IP ID, якщо нормалізатор може модифікувати усі фрагменти; Time-to-Live і TCP timestamp також можуть бути перезаписані, якщо нормалізатор знаходиться дуже близько до джерела, або біти низького порядку можуть бути рандомізовані [38].

Захист хостів не може усунути прихований мережний канал, але він може запобігти їх експлуатації в деяких сценаріях застосування. Якщо робочі станції захищені від зламу, встановлення троянів і модифікація програмного забезпечення або мережевого стеку є неможливою. Це унеможливлює використання хакерами прихованих каналів передачі інформації в мережних протоколах.

Висновок до розділу 2:

Існує велика множина різних типів прихованих каналів. Усі вони відрізняються за пропускною здатністю каналу, рівнем таємності та сприятливими умовами використання. Часто їх неможливо помітити не знаючи про їх наявність, оскільки зазвичай вони використовують засоби, що не були призначені для такого способу застосування.

Щоб знешкодити приховані канали передачі інформації в мережних протоколах, необхідно спочатку їх визначити або виявити. Визначення полягає у намаганні виявити спільний ресурс на етапі розробки, який потенційно може бути використаний як основа прихованого каналу, в той час як виявлення є спробою знайти активний прихований канал, вивчаючи супутні його роботі аномалії. Існують різні методи виявлення прихованих каналів в мережних протоколах. Після визначення або виявлення прихований канал може бути ліквідований, обмежений, підданий аудиту та/або задокументований. Для знешкодження прихованого каналу може бути впроваджено захист мережі, нормалізація трафіку, захист хостів.

Результати, отримані у ході дослідження типів прихованих каналів, методів їх створення, виявлення та знешкодження, будуть використані у наступному розділі для вибору засобів боротьби із прихованими каналами (зокрема, з DNS тунелем) та розробки власного методу виявлення та знешкодження прихованих каналів.

3 ВИЯВЛЕННЯ ТА ЗНЕШКОДЖЕННЯ ПРИХОВАНИХ КАНАЛІВ ПЕРЕДАЧІ ІНФОРМАЦІЇ У МЕРЕЖНИХ ПРОТОКОЛАХ

У цьому розділі буде наведено різні способи боротьби із відомими прихованими каналами передачі інформації в мережних протоколах. Серед них готові засоби, спрямовані на виявлення та усунення уже відомих каналів; розроблений протягом дослідження метод аналізу трафіку з метою виявлення аномалій, що свідчитимуть про наявність потенційного прихованого каналу; превентивні заходи унеможливлення використання прихованих каналів. А також висновки щодо найкращих умов використання кожного із способів.

3.1 Засоби виявлення та знешкодження відомих прихованих каналів

Як вже було зазначено раніше, як тільки певний метод створення прихованих каналів набуває поширення, спеціалісти намагаються знайти спосіб його виявити та знешкодити. Тому можна з впевненістю сказати, що для усіх більш-менш відомих прихованих каналів створено готові рішення з викриття та усунення, якщо є можливою формалізація і автоматизація певного підходу до цих задач. Виявлення та знешкодження відомого прихованого каналу буде розглянуто на прикладі DNS тунелю, оскільки він є одним з найбільш вивчених. Далі буде розглянуто декілька різних підходів до його викриття, а також описано використання Snort (IPS&IDS) та брандмауера Palo Alto Networks.

3.1.1 Виявлення відомих небезпечних доменів

Цей метод реалізується аналізом DNS запитів від вашої мережі. У разі виявлення відомого підозрілого домену – пакет відкидається.

Серед публічних DNS клієнтів для цього згодяться:

- Cloudfare (безкоштовний)
- OpenDNS (безкоштовний для домашнього використання)
- Infoblox (рішення для бізнесу)
- Cisco Umbrella (рішення для бізнесу)

Деякі брандмауери забезпечують контроль URL і можуть блокувати відомі небезпечні домени. Антивіруси (наприклад, Crowdstrike Falcon Endpoint Protection) також до цього здатні, але зазвичай це потребує створення власних правил відносно відомих підозрілих IP адрес.

3.1.2 Аналіз трафіку

Моніторинг трафіку є важливою складовою безпеки організації. Якщо ви знаєте типову поведінку вашої мережі, стандартні обсяги запитів, їх очікуваний вміст, години підвищеної активності, кореляцію кількості запитів з їх джерелом, то ви зможете помітити різку зміну характеру активності. Така подія буде свідчити про можливий інцидент безпеки, зокрема появу прихованого DNS тунелю. [39]

Для такого методу викриття прихованих DNS каналів будуть корисними брандмауери, засоби для аналізу мережного трафіку (наприклад, Wireshark), а також такі утиліти як [40]:

- dnsHunter – модуль для MercenaryHuntFramework і Mercenary-Linux, що зчитує .pcap файли, відсіює усі запити крім DNS та виконує співставлення геолокації, що є корисним під час аналізу.
- reassemble_dns – утиліта, що зчитує .pcap файли та аналізує DNS повідомлення.

Далі буде детальніше розглянуто роботу брандмауера Palo Alto Networks у контексті викриття прихованого DNS каналу. Нехай у мережі

організації існує певний DNS тунель. Тепер необхідно спробувати заблокувати використання прихованого каналу. Для початку необхідно проаналізувати трафік. Логічно було б припустити наявність різкої зміни поведінки запитів і велику кількість DNS трафіку. На рисунках 3.1 і 3.2 видно, що брандмауер розпізнав надсилання TCP трафіку через DNS.

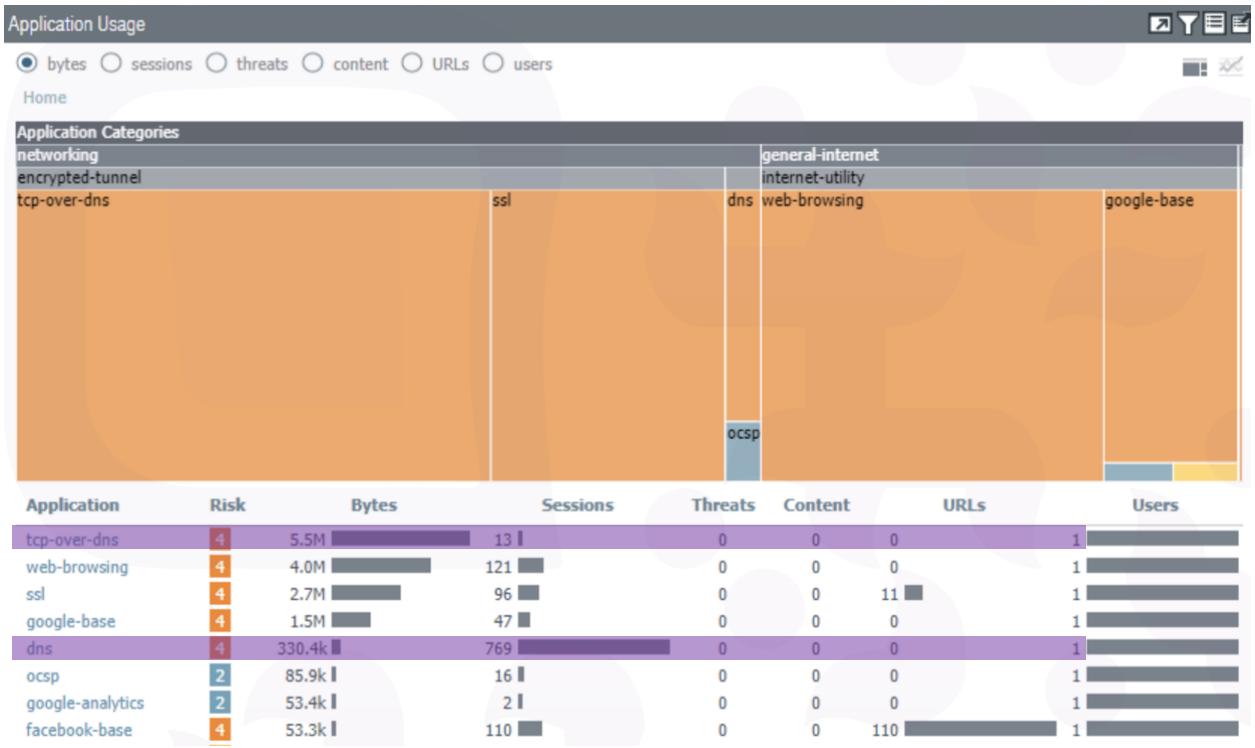

Рисунок 3.1 – Базовий аналіз трафіку застосунків від брандмауера Palo Alto Networks

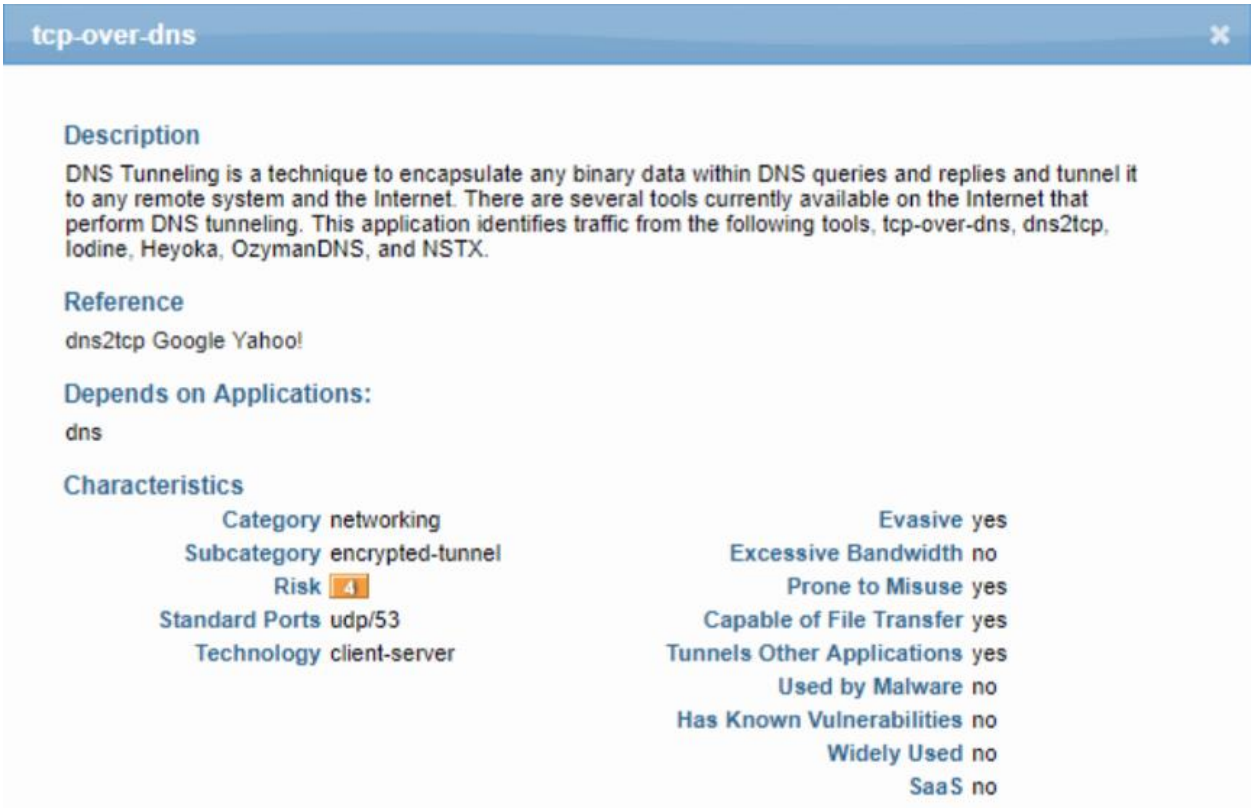

Рисунок 3.2 – Деталі, щодо tcp-over-dns

Схоже, що брандмауер Palo Alto Networks без проблем розпізнає трафік більшості засобів для DNS тунелювання і групує їх в одну категорію. Знаючи це, створити правило, що заблокує їх активність, стає просто. Подальші дії зображені на рисунках 3.3, 3.4, 3.5.

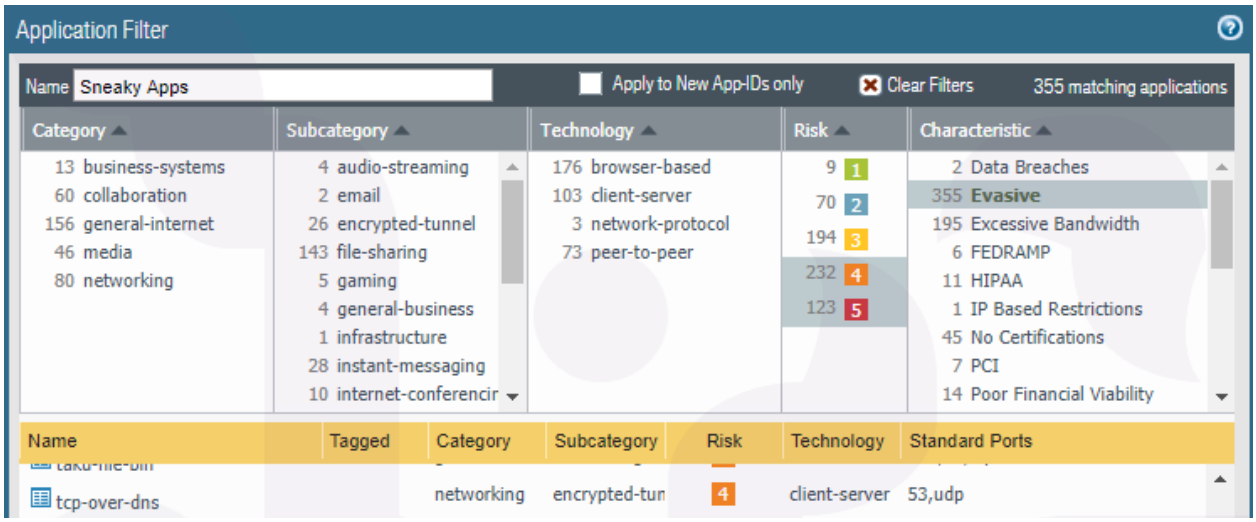

Рисунок 3.3 – Фільтр Evasive застосунків високого рівня ризику

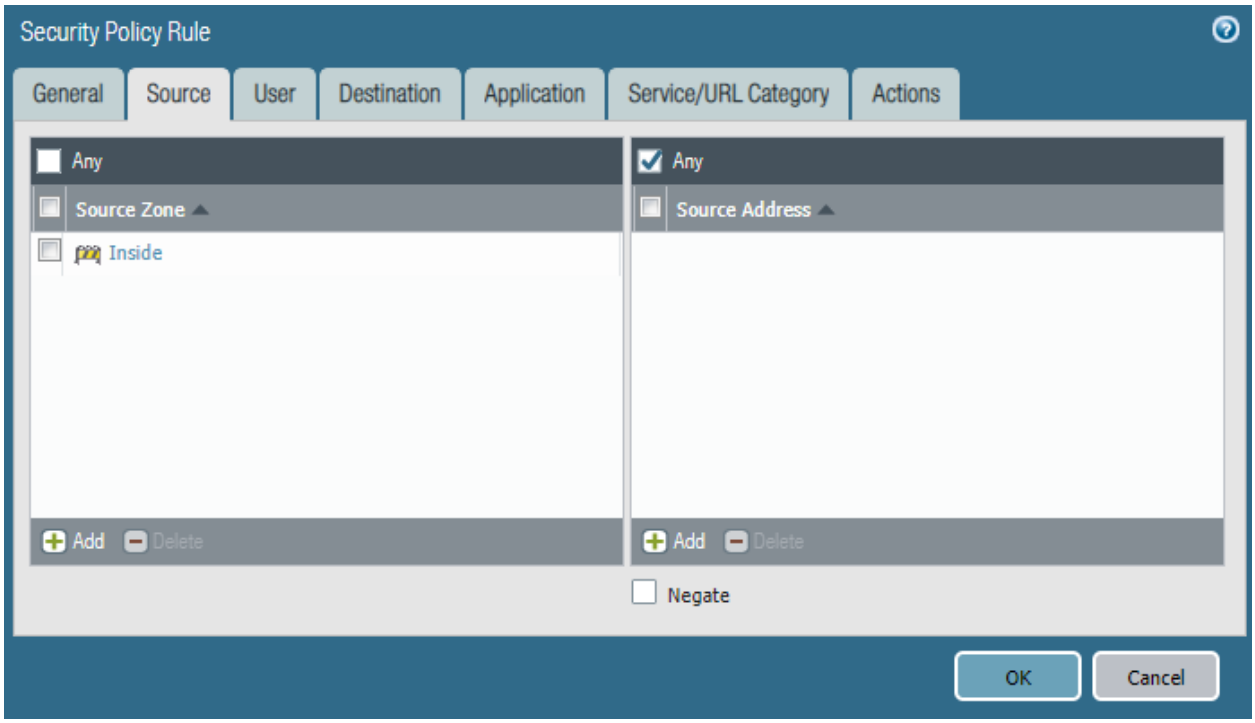

Рисунок 3.4 – Вказуємо, що це правило працює для трафіку, який виходить з нашої мережі

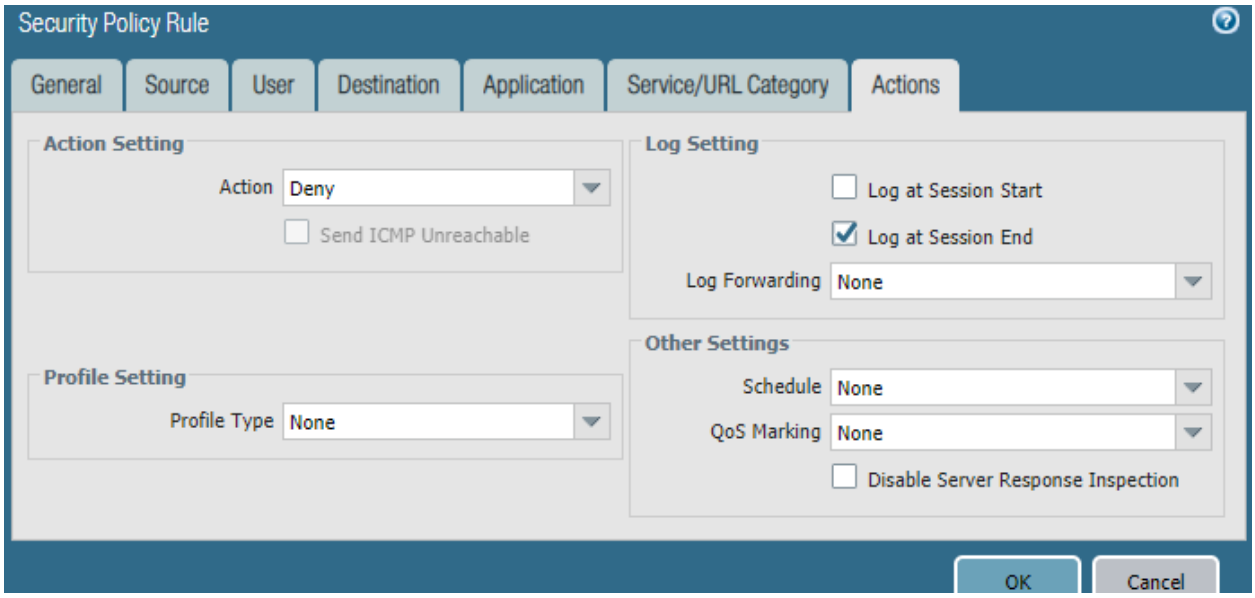

Рисунок 3.5 – Вказуємо, що весь такий трафік необхідно відкидати

3.1.3 Використання Intrusion Prevention System та Intrusion Detection System

Розглянемо таку Intrusion Prevention System (IPS) як Snort. У ньому присутні як специфічні правила для виявлення унікальних пакетів для утиліт DNS тунелювання, так і більш загальні. Ось правило для виявлення трафіку утиліти Iodine:

```
# alert udp $EXTERNAL_NET 53 -> $HOME_NET any (msg : ”APP-DETECT iodine dns tunnelling handshake server ACK”; flow : to_client; byte_test : 1,&,0x80,2; content : ”|00 01 00 01 00|”; depth : 5; offset : 4; content : ”v”; within : 1; distance : 4; content : ”VACK”; within : 200; fast_pattern; metadata : service dns; reference : url,code.kryo.se/iodine/README.html; classtype : policy-violation; sid : 27046; rev : 2;)
```

Схожі правила існують і для DNS тунелів створених за допомогою утиліт OzymanDNS, DNSCat2 [40]. Більш загальні правила зазвичай орієнтуються на довжину URL у DNS запитах, тож їх можна легко обійти скоротивши корисне навантаження у кожному запиті, що звичайно знизить швидкість тунелю.

3.2 Виявлення та знешкодження прихованих каналів на основі аналізу мережного трафіку

Нажаль не усі підходи до виявлення та знищення прихованих каналів можна повністю автоматизувати. Інколи це стається через особливості каналу, але частіше через те, що метод та засоби створення прихованого каналу ще не відомі. Щоб попередити спробу реалізації невідомої загрози

необхідно налаштувати якісний моніторинг системи, що буде сповіщати про будь-які аномалії. У випадку з прихованими каналами передачі інформації в мережних протоколах будуть доречними стратегії виявлення, що розглядалися у попередньому розділі. Далі як proof of concept буде наведено виявлення та знешкодження прихованого каналу передачі інформації в протоколі IP на базі опціонального поля Differentiated Services – DiffServ (колишнього Type of Service – ToS). Спираючись на отримані результати, можна буде вивести загальні правила моніторингу трафіку, що зможуть повідомляти навіть про нові, ще не вивчені приховані канали передачі інформації в мережних протоколах.

Після виявлення аномалій необхідно проаналізувати трафік на предмет наявності прихованого каналу передачі інформації. У разі наявності каналу слід визначити сигнатури виявленого каналу і знешкодити його одним із доступних способів, наприклад, за допомогою нормалізації трафіку.

3.2.1 Завантаження та встановлення необхідного програмного забезпечення

Уданому дослідженні було використано Wireshark для перехоплення пакетів у реальному часі та їх запису до файлу та Elastic Stack для моніторингу та аналізу мережного трафіку. Програмне забезпечення було завантажено з офіційних сайтів розробників:

```
https://www.wireshark.org/#download
https://www.elastic.co/downloads/
```

Wireshark — програма для аналізу мережевих пакетів Ethernet і інших мереж з вільним вихідним кодом.

Встановлення Wireshark (рисунок 3.6) відбувається максимально просто, у декілька кліків, тому наводити весь процес немає сенсу.

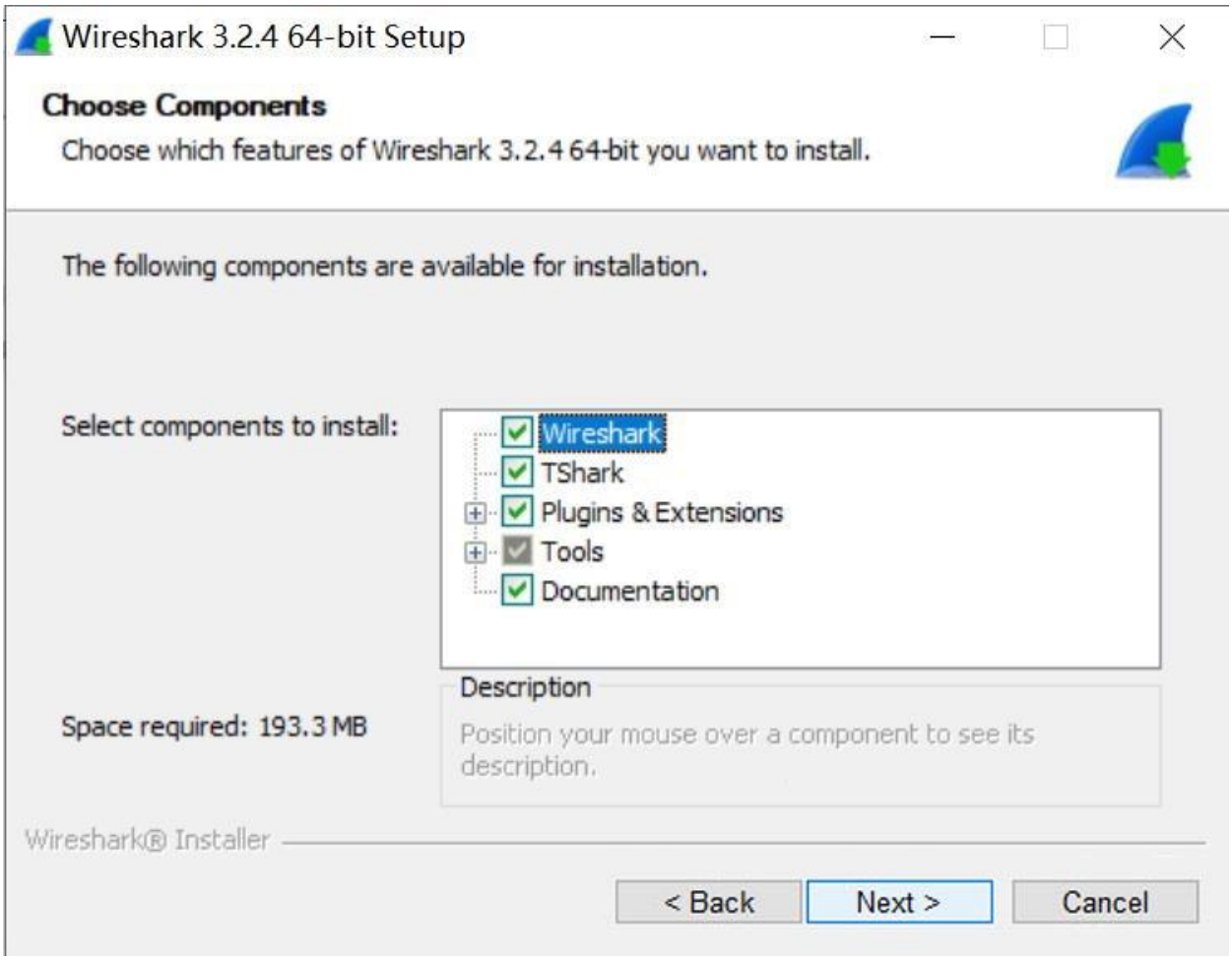

Рисунок 3.6 – Один із етапів встановлення Wireshark

Elastic Stack на даний момент складається з чотирьох компонентів, які можна побачити на рисунку 3.7: Beats, Logstash, Elasticsearch та Kibana.

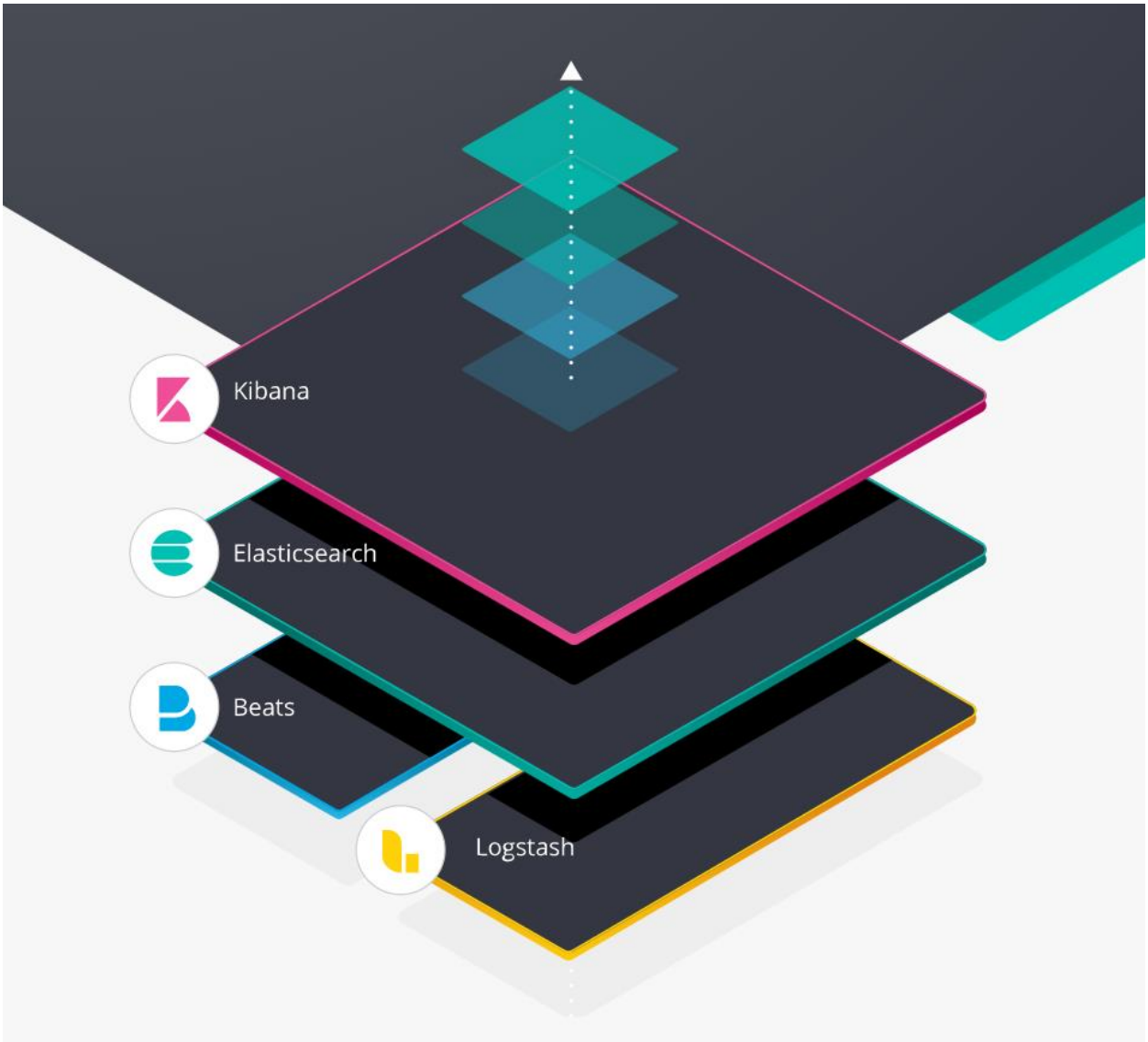

Рисунок 3.7 – Структура Elastic Stack

Beats – це легкі агенти пересилання файлів. Ці агенти встановлюються на сервери і налаштовуються на читання і пересилання вмісту файлу до Logstash. У даному дослідженні використовувався Filebeat.

Logstash - це конвеєр обробки даних на серверній стороні, який збирає дані з багатьох джерел одночасно, займається їх парсингом і трансформуванням та відправляє до Elasticsearch.

Elasticsearch – це пошуково-аналітична система, побудована на Apache Lucene.

Kibana - це інструмент візуалізації, який полегшує пошук даних, що зберігаються в базі даних Elasticsearch, а також надає прості засоби для створення візуалізацій з діаграмами та графіками, дашбордами та багатьма іншими функціями.

Після завантаження всіх чотирьох компонентів з сайту розробника необхідно розпакувати архіви та створити змінні середовища які показано на рисунку 3.8.

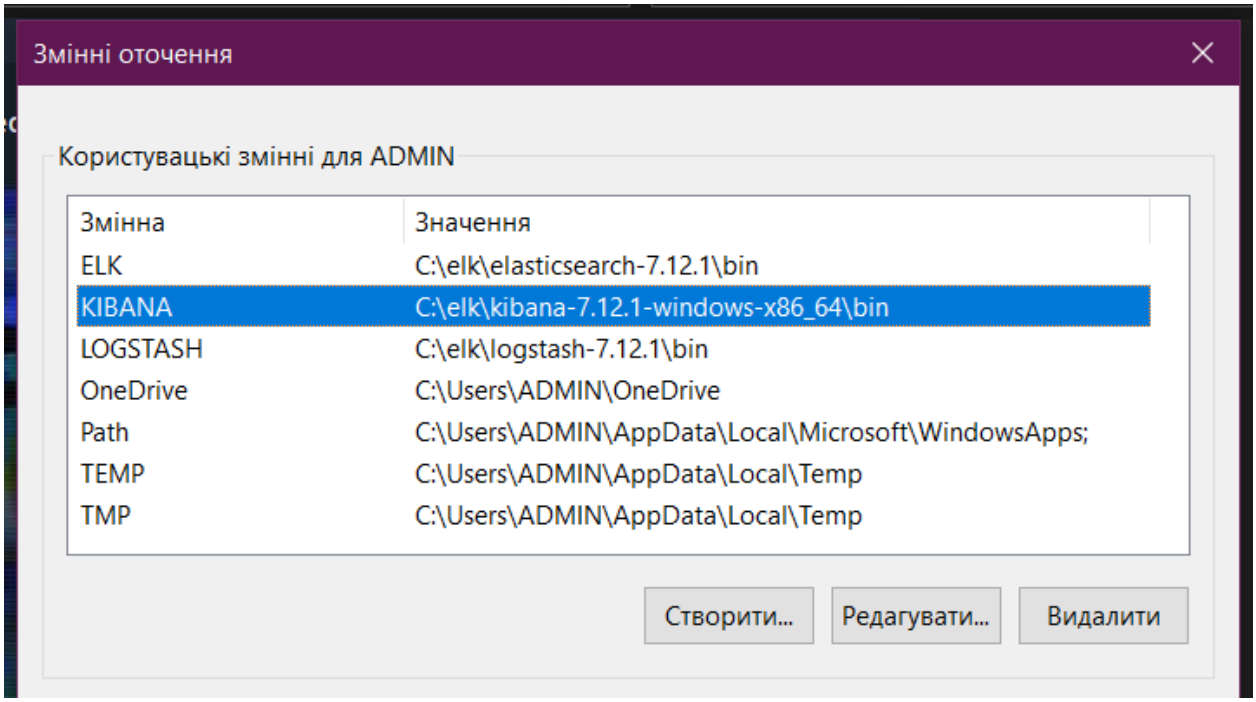

Рисунок 3.8 – Додані змінні середовища

В командній строці необхідно виконати наступну команду для запуску Elasticsearch:

```
elasticsearch.bat -Ecluster.name=mycluster1 -Enode.name=mynode1
```

Результат виконання команди можна побачити на рисунках 3.9, 3.10.

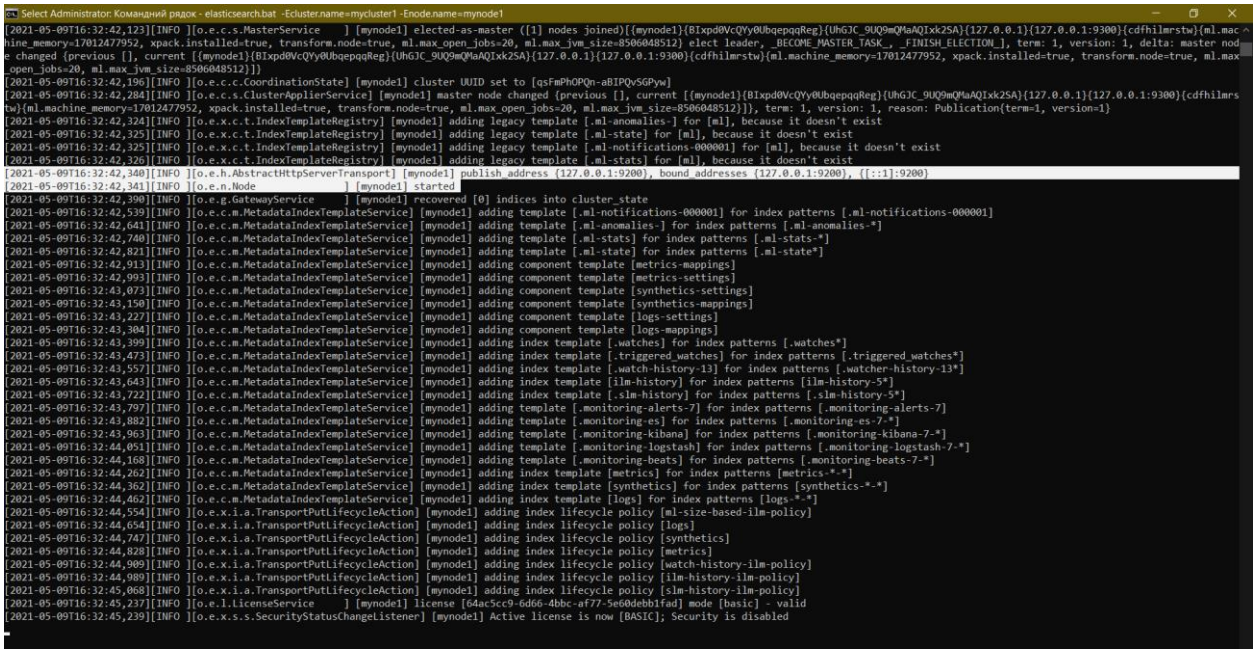

Рисунок 3.9 – Виконання команди для запуску Elasticsearch

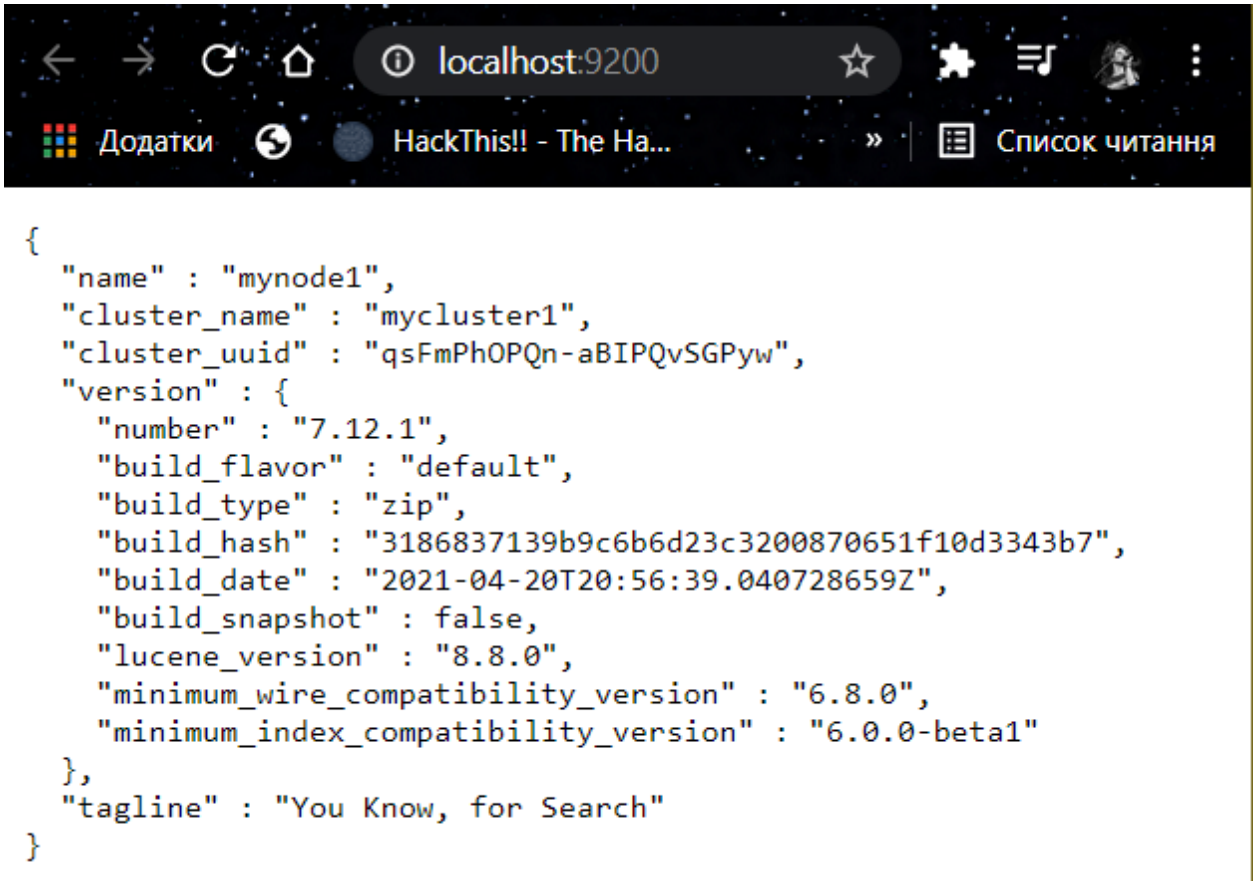

Рисунок 3.10 – Elasticsearch успішно запущено

Далі необхідно відредагувати файл конфігурації Logstash як на рисунку 3.11.

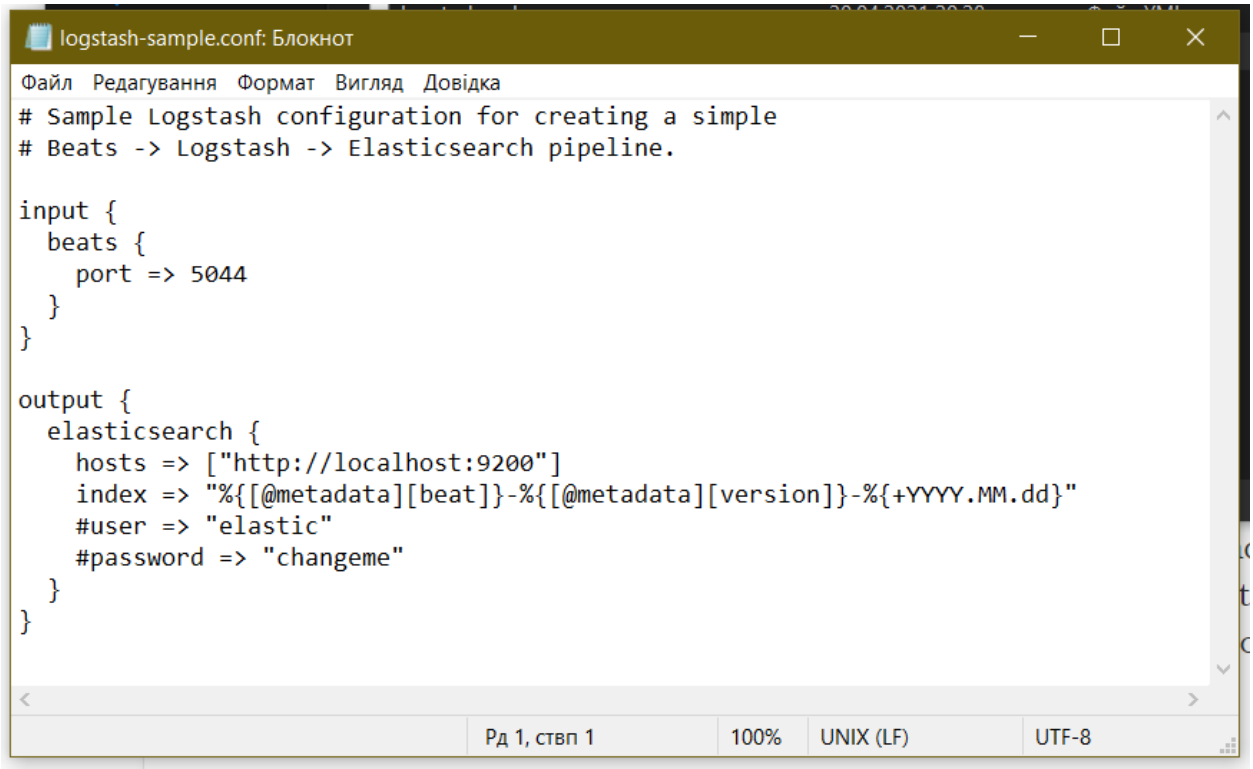

Рисунок 3.11 – Файл конфігурації Logstash

Команда для запуску Logstash:

```
logstash.bat -f c:\elk\logstash-7.12.1\config\logstash-sample.conf
```

Результат виконання вище наведеної команди можна спостерігати на рисунках 3.12 та 3.13.

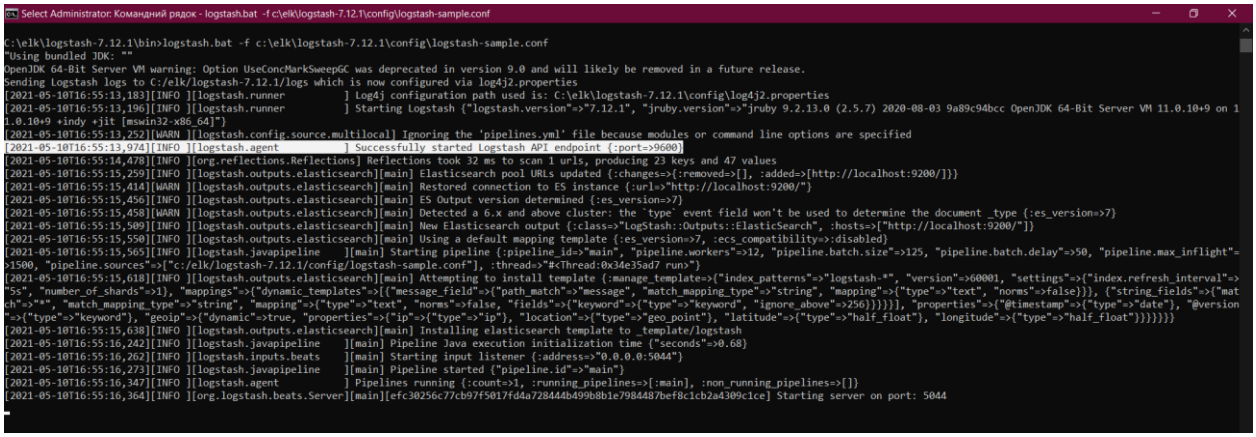

Рисунок 3.12 – Результат виконання команди для запуску Logstash

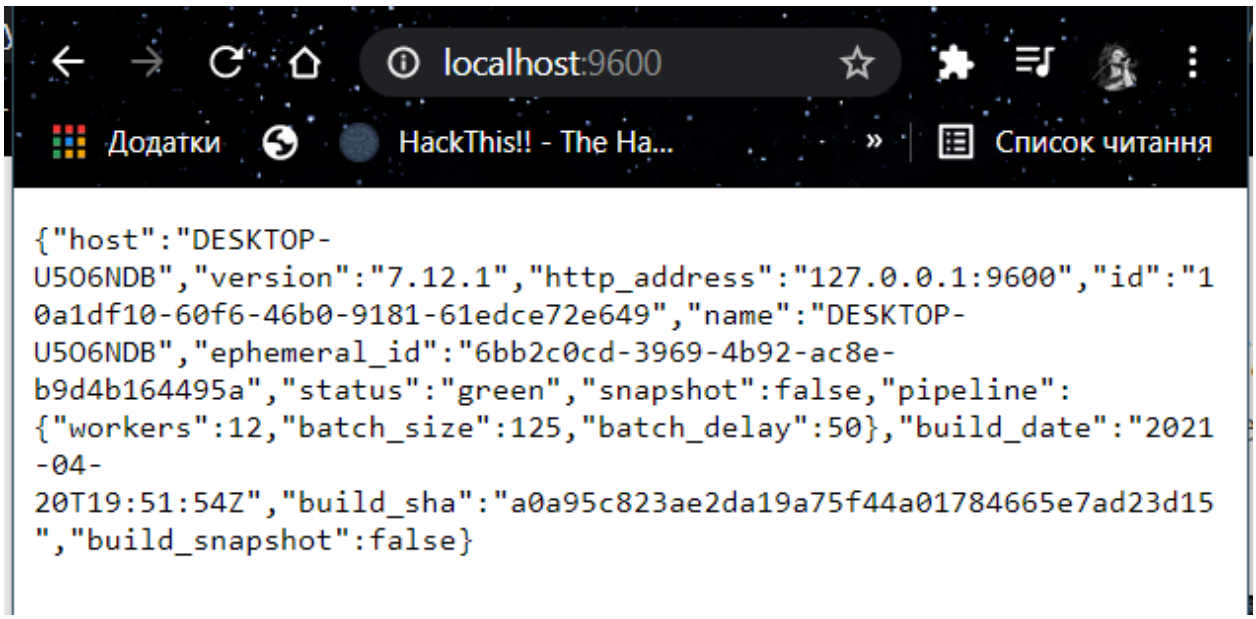

Рисунок 3.13 – Logstash успішно запущено

На рисунках 3.14, 3.15, 3.16 можна побачити конфігурацію для Kibana, результат її запуску в командній строці та у браузері.

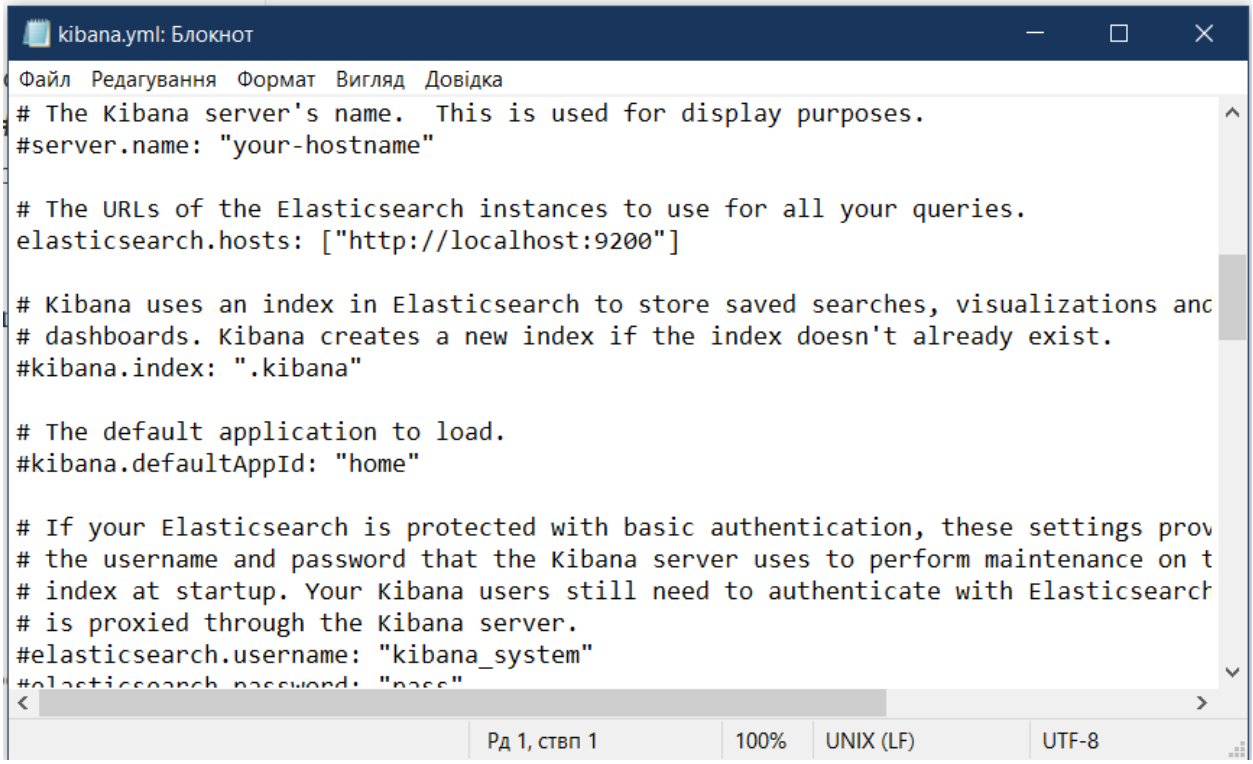

Рисунок 3.14 – Файл конфігурації Kibana

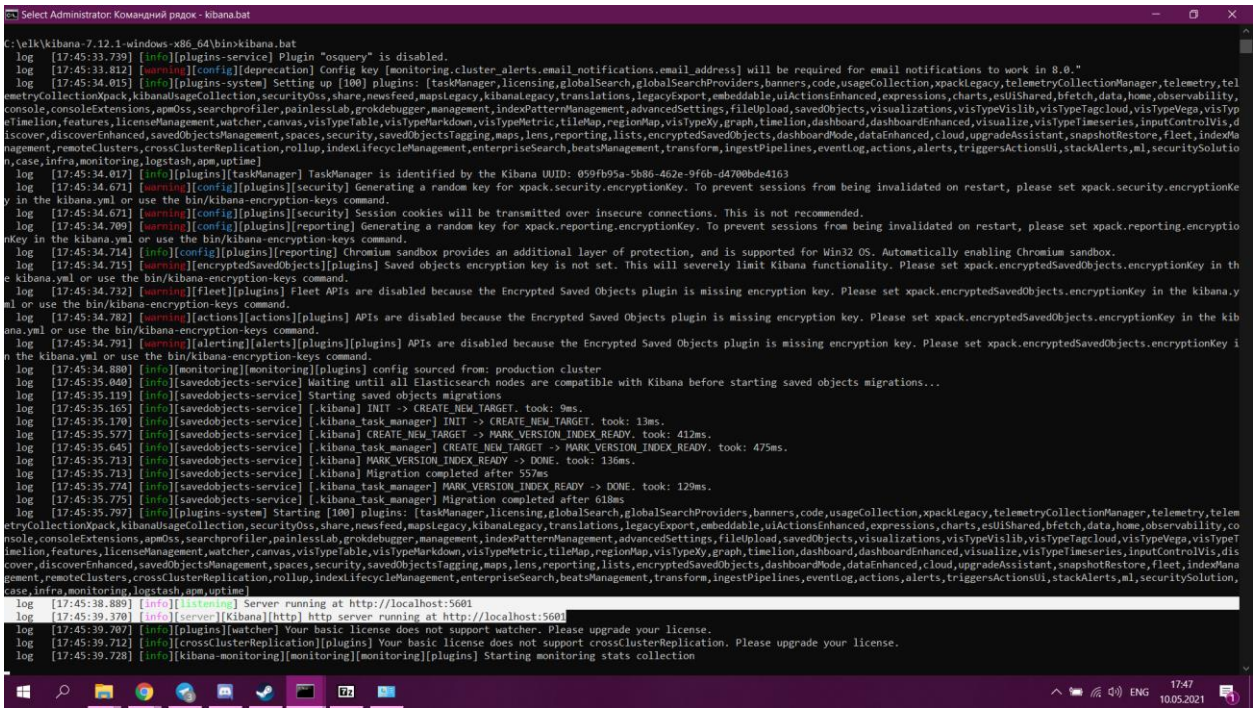

Рисунок 3.15 – Результат виконання команди запуску Kibana в командній строці

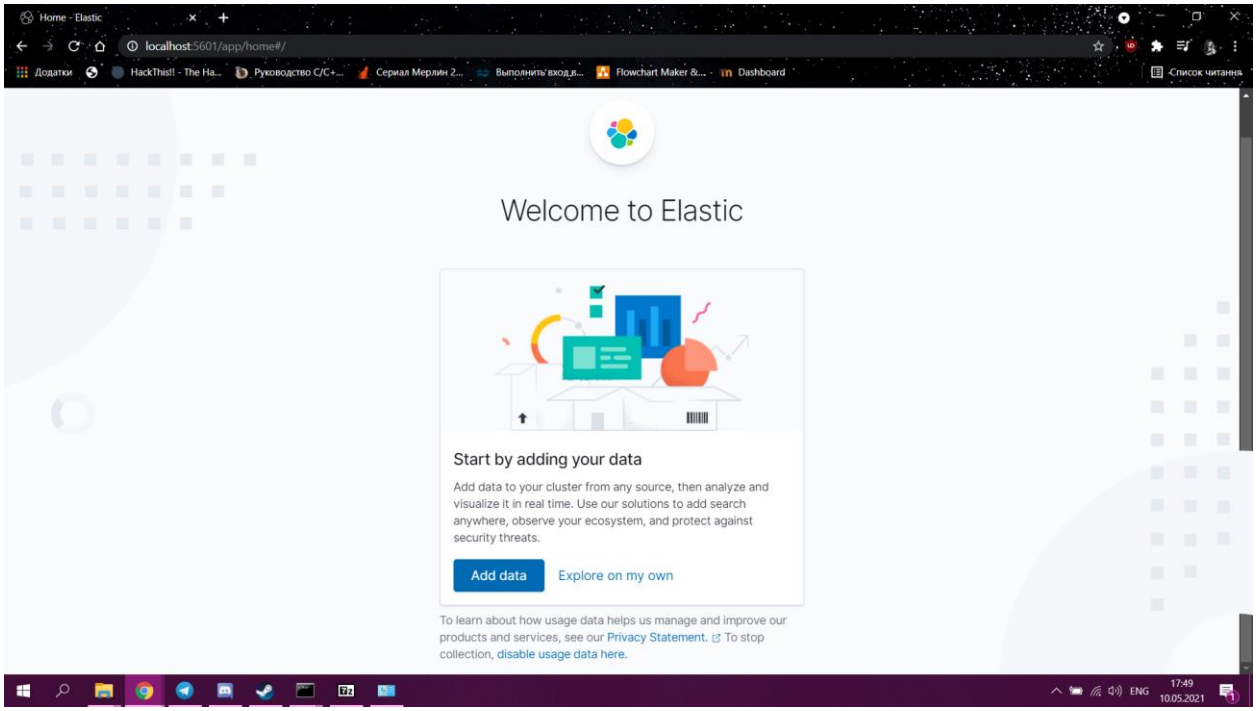

Рисунок 3.16 – Kibana успішно запущена

Встановити сервіс Filebeat можна за допомогою команди:

```
PowerShell.exe -ExecutionPolicy UnRestricted -File .\install-service-filebeat.ps1
```

Виконання команди для встановлення сервісу, конфігураційний файл Filebeat та запуск Filebeat можна побачити на рисунках 3.17, 3.18 і 3.19 відповідно.

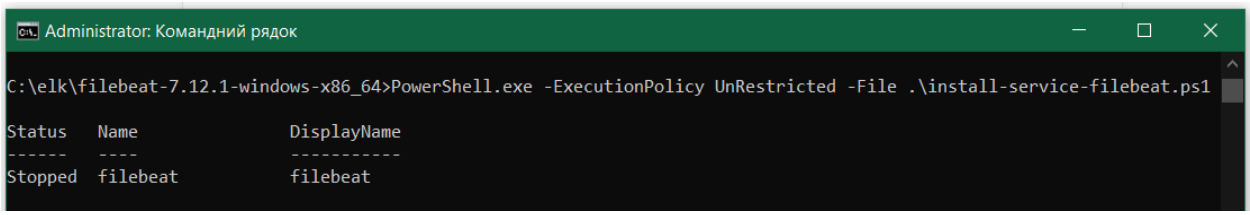

Рисунок 3.17 – Встановлення сервісу Filebeat

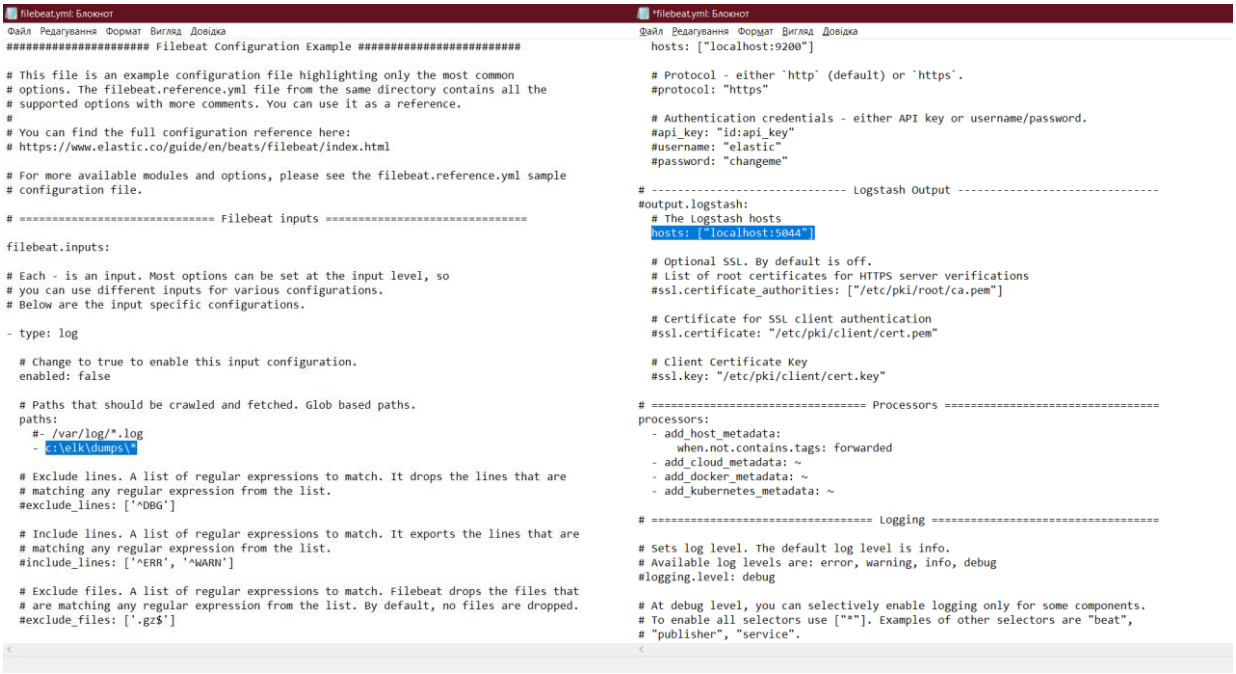

Рисунок 3.18 – Файл конфігурації Filebeat

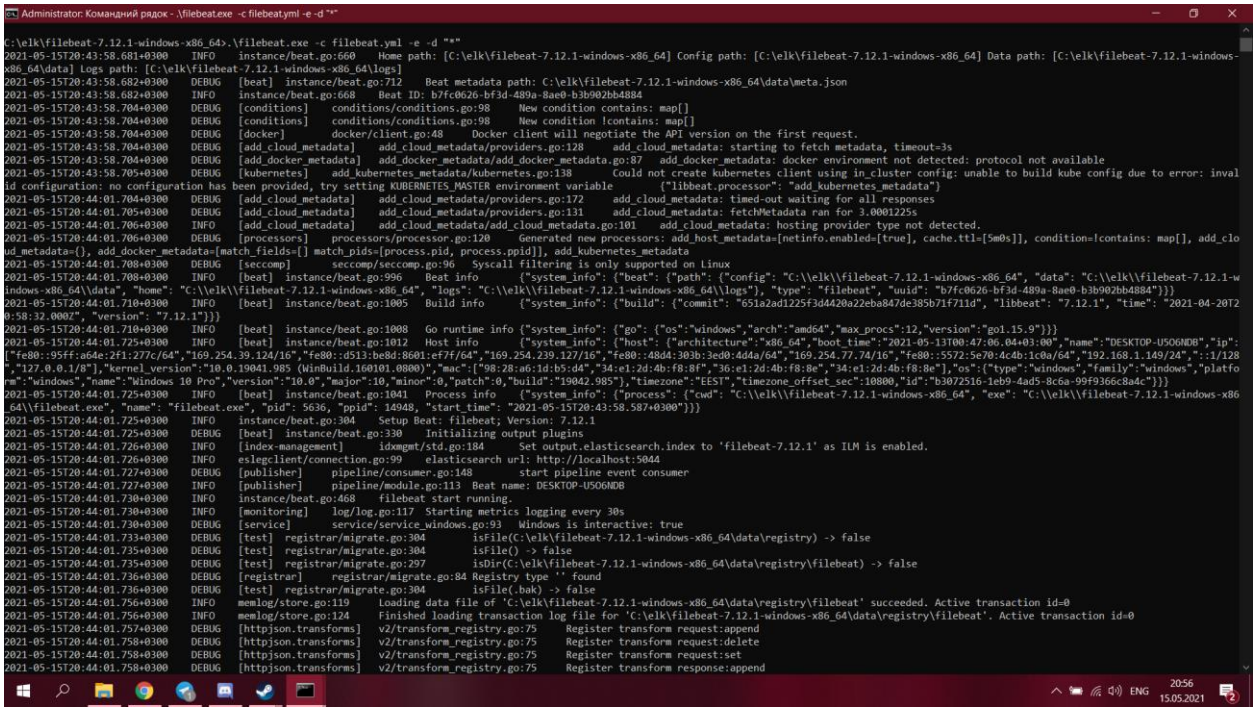

Рисунок 3.19 – Запуск Filebeat

Установка Elastic Stack завершена.

3.2.2 Підготовка Elastic Stack до моніторингу та аналізу мережного трафіку

На додаток до графічного інтерфейсу Wireshark надає утиліту для командного рядка tshark, яку можна використовувати для захоплення трафіку в реальному часі, а також для читання та аналізу дампів трафіку. На виході tshark може видавати звіти та статистику, результати парсингу дампів у різних текстових форматах. Один з форматів виводу, що підтримується tshark з версії 2.2 (випущеної у вересні 2016 року) є формат JSON для Elasticsearch Bulk API.

```
tshark -i Wi-Fi -T ek > C:\elk\dumps\packets.json
```

Наведена вище команда буде виконувати захоплення пакетів на мережевому інтерфейсі Wi-Fi у реальному часі та їх логування у форматі Elasticsearch Bulk API у файл packets.json.

Необроблені дані пакетів містять надзвичайно велику кількість полів. Крім того, вивід tshark -T ek містить всі значення полів як строки, незалежно від того, чи є дані дійсно текстом або числами, що містять часові позначки та IP-адреси. Без правильних типів даних виконання операцій, специфічних для полів певного типу є неможливим (наприклад, з'ясувати середній розмір пакета).

Щоб індексувати числа як числа, позначки часу як позначки часу та запобігати проблемам з індексованими полями, слід явно вказати їх зіставлення. Приклад зіставлення полів та їх типів даних для даного дослідження наведено нижче на рисунку 3.20:

```
PUT _template/packets
{
  "template": "packets-*",
  "mappings": {
    "pcap_file": {
      "dynamic": "false",
      "properties": {
        "timestamp": {
          "type": "date"
        },
        "layers": {
          "properties": {
            "frame": {
              "properties": {
                "frame_frame_len": {
                  "type": "long"
                },
                "frame_frame_protocols": {
                  "type": "keyword"
                }
              }
            },
            "ip": {
              "properties": {
                "ip_ip_src": {
                  "type": "ip"
                },
                "ip_ip_dst": {
                  "type": "ip"
                }

                "ip_ip_proto": {
                  "type": "keyword"
                }

                "ip_ip_dsfield": {
                  "type": "keyword"
                }

              }
            },
            "udp": {
              "properties": {
                "udp_udp_srcport": {
                  "type": "integer"
                },
                "udp_udp_dstport": {
                  "type": "integer"
                }
              }
            }
          }
        }
      }
    }
  }
}
```

Рисунок 3.20 – Зіставлення полів та їх типів даних

Тепер необхідно створити засоби моніторингу та аналізу мережного трафіку – вочер та дашборд. Оскільки поле DiffServ є необов’язковим, його значення зазвичай рівне 0. Вочер спостерігає за кількістю отриманих за останні 5 хвилин пакетів із ненульовим значенням поля DiffServ. У разі перевищення заданого порогового значення кількості таких пакетів буде

сформовано алерт з короткими відомостями про інцидент. У нашому випадку алерт надійде до каналу SOC департаменту у програм Slack. Код вочера наведено в додатку А. На дашборді було розміщено усі необхідні для аналізу отриманого алерту дані.

Описаний вище алгоритм обробки та аналізу мережевого трафіку зображено на рисунку 3.21.

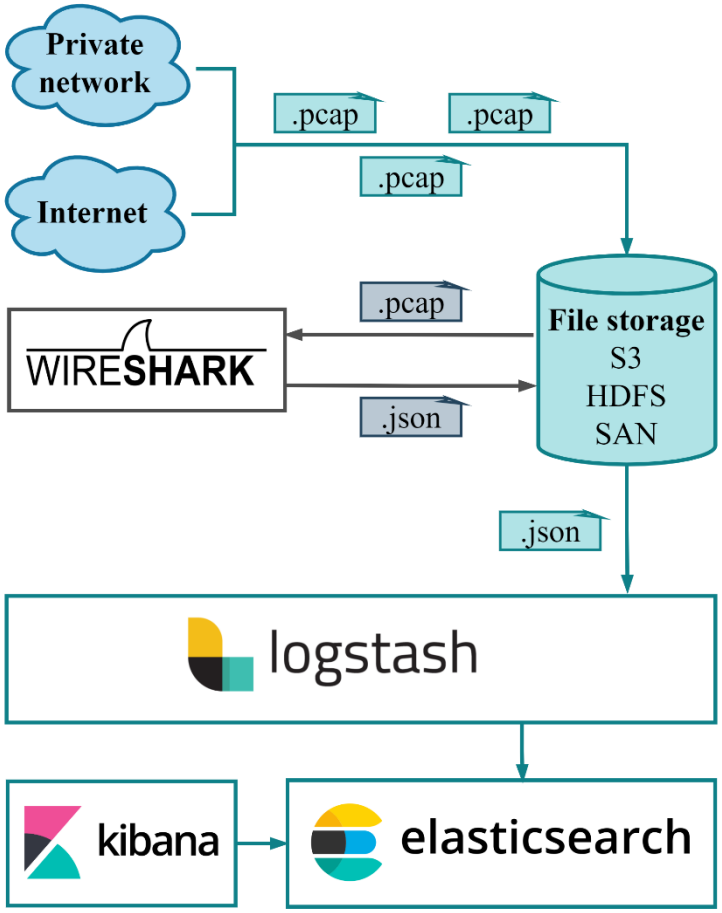

Рисунок 3.21 – Конвеєр аналізу мережевих пакетів за допомогою Wireshark та Elastic Stack

3.2.3 Аналіз отриманого алерту і знешкодження прихованого каналу передачі інформації

Приклад спрацювання алерту зображено на рисунку 3.22 (Синіми прямокутниками приховано інформацію про організацію, що не підлягає розголошенню).

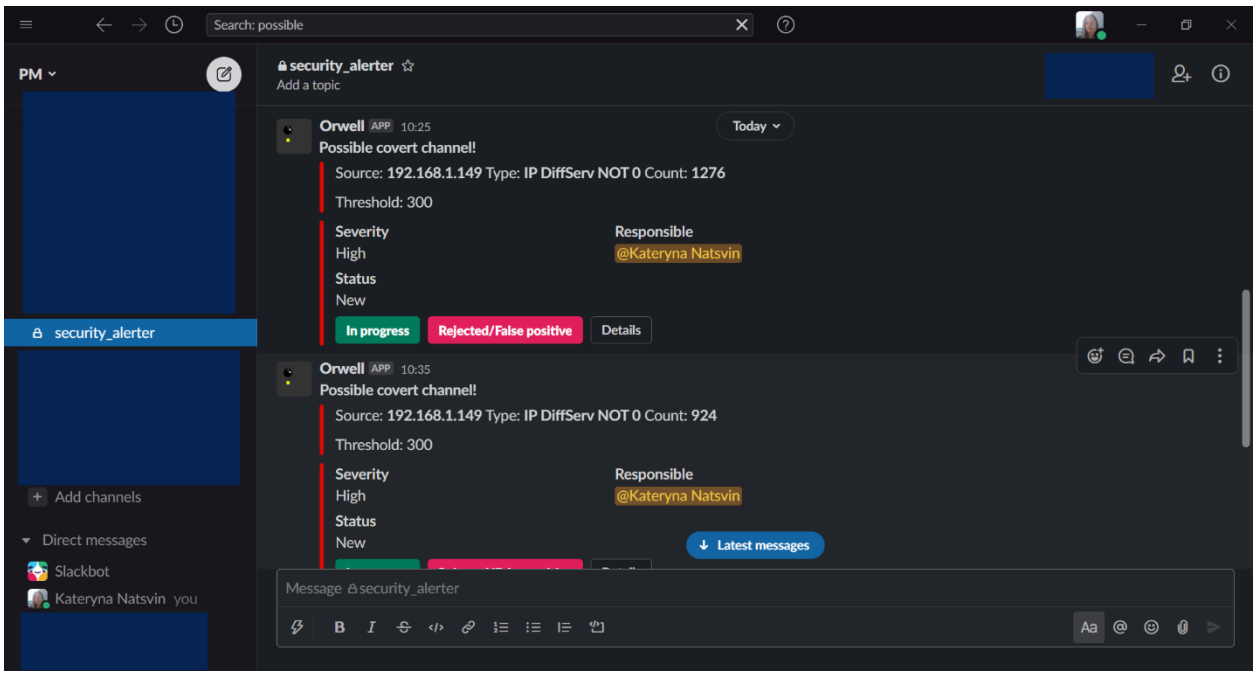

Рисунок 3.22 – Алерт про можливий прихований канал у Slack

Після отримання алерту відповідальний спеціаліст з команди SOC повинен оперативно вивчити надані короткі відомості про інцидент у Slack та приступити до детального аналізу даних за якими був сформований алерт. Перейти до дашборду, який допоможе здійснити аналіз можна за допомогою кнопки Details. Приклад дашборду з даними, що підлягають аналізу, наведено на рисунку 3.23.

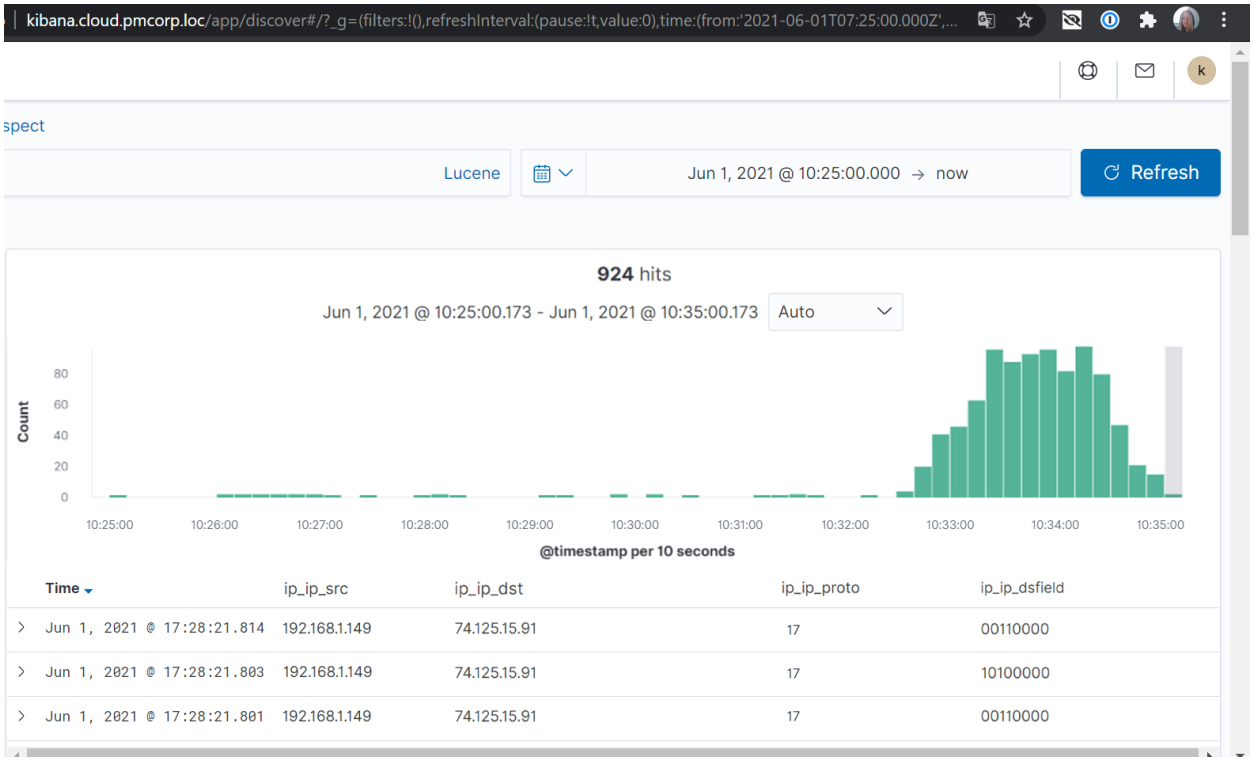

Рисунок 3.23 – Дашборд для аналізу аномального трафіку

У разі якщо, порогове значення кількості аномальних пакетів перевищено незначно або відома причина підвищення їх обсягу, алерт вважатиметься false positive. У разі частого повторення схожих випадків варто провести детальний аналіз сукупності інцидентів на наявність прихованого каналу із низькою пропускною здатністю і за необхідності підвищити порогове значення.

В іншому випадку алерт вважатиметься true positive і потребуватиме швидкого і якісного аналізу для виявлення сигнатури прихованого каналу передачі інформації і його знешкодження шляхом захисту мережі або нормалізації трафіку. Захист мережі здійснюється блокуванням протоколів та портів, що використовуються прихованими каналами, за наявності такої можливості. Нормалізацію трафіку можна реалізувати за допомогою таких інструментів як Mitmproxy, Ettercap та Bettercap.

3.3 Превентивні заходи захисту від прихованих каналів

Через природу і різноманітність прихованих каналів неможливо повністю автоматизувати їх виявлення та знешкодження. Часто простіше попередити появу прихованих каналів, ніж їх виявити. У таких випадках вдаютьсядо захисту хостів. Загальною рекомендацією є використання дієвого антивірусу (наприклад, Crowdstrike Falcon Endpoint Protection). Поведінковий аналіз допоможе знешкодити програмне забезпечення локального клієнта у момент запуску. А заборона відвідування хостами відомих IP з поганою репутацією унеможливить надсилання пакетів злочинцям. Приклади таких правил наведені на рисунках 3.24, 3.25 і 3.26.

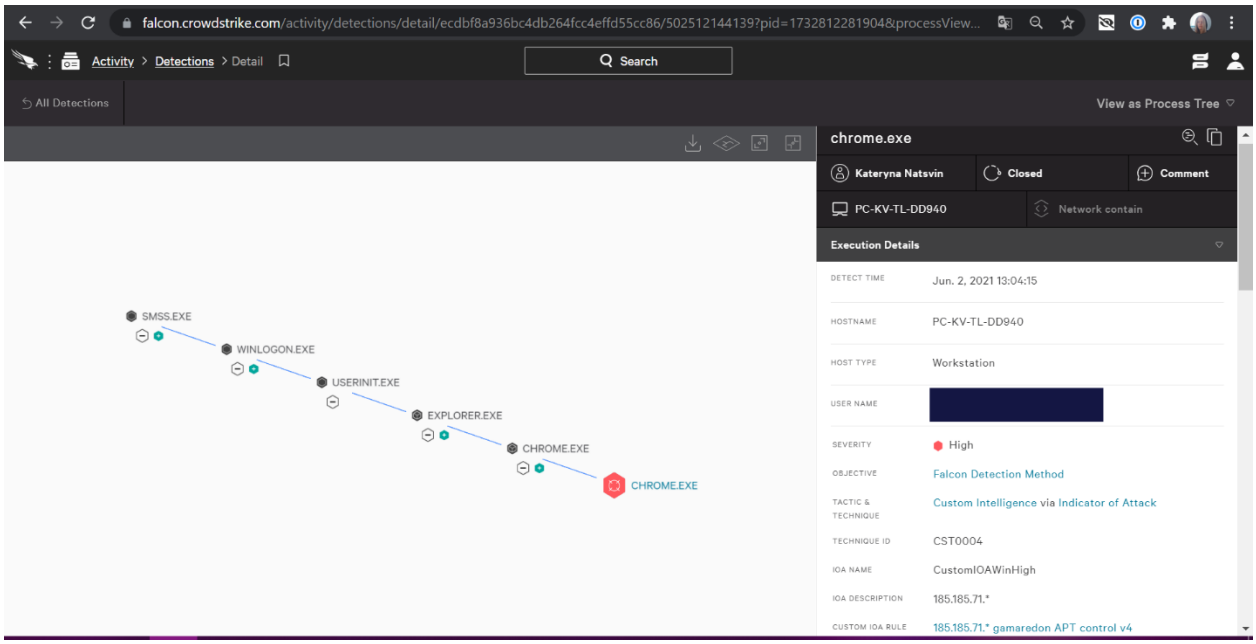

Рисунок 3.24 – Алерт про спробу доступу до проміжку відомих IP з поганою репутацією

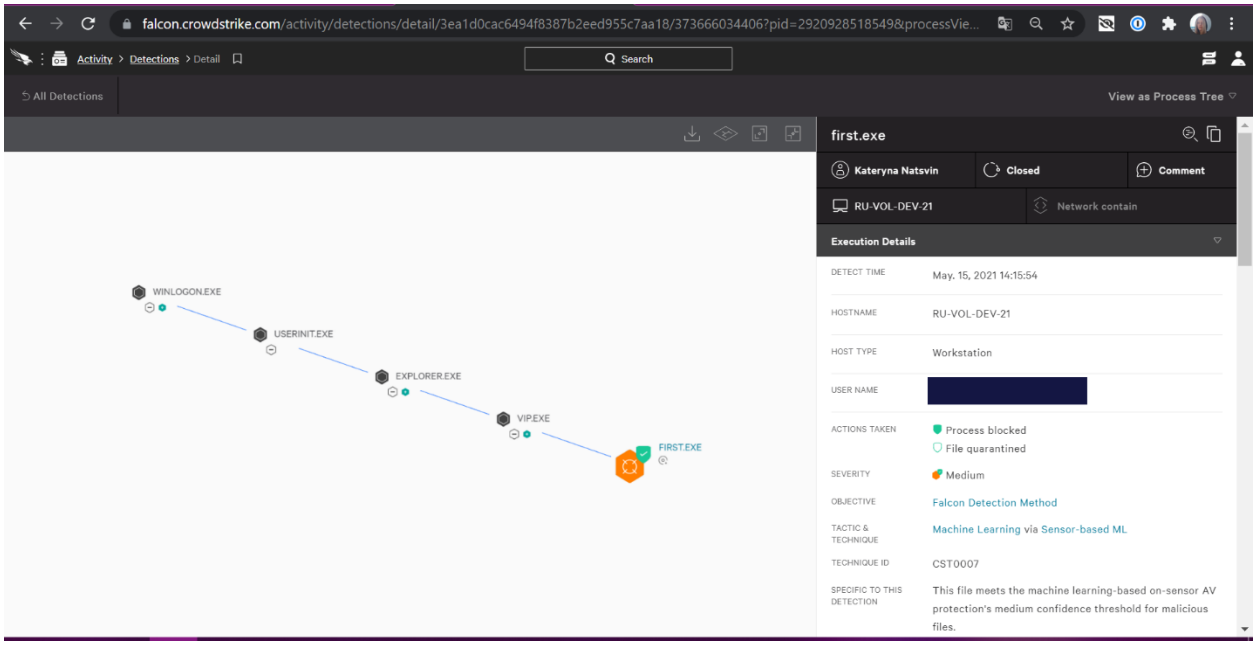

Рисунок 3.25 – Блокування виконуваного файлу на основі аналізу його поведінки за допомогою машинного навчання

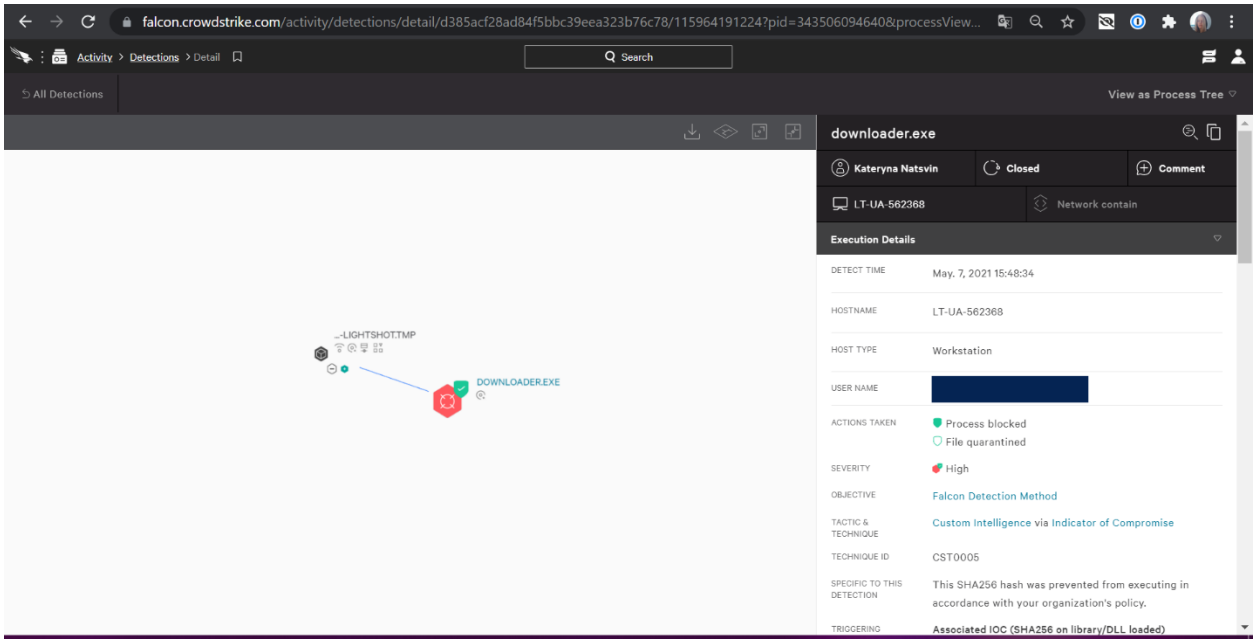

Рисунок 3.26 – Блокування виконуваного файлу на основі відповідності його SHA256 хешу списку заборонених

3.4 Порівняння ефективності наведених способів захисту

Оскільки існують різні підходи до виявлення та знешкодження прихованих каналів передачі інформації в мережних протоколах, варто знати за яких умов вони будуть найбільш ефективні.

Проти прихованих каналів із відомими сигнатурами найбільш ефективними є готові засоби виявлення та знешкодження такі як: публічні DNS клієнти з можливістю перевірки репутації IP адреси, спеціалізовані утиліти, брандмауери, антивіруси та IPS & IDS зі спеціальними правилами.

У випадку, якщо прихований канал створений новим, ще не описаним способом, і його сигнатури ще не відомі, варто застосовувати схожий до описаного раніше метод моніторингу та аналізу трафіку на предмет аномалій. Він дозволить виявити особливості нового прихованого каналу, та швидко знешкодити його за допомогою належного інструмента, наприклад, mitm-proxy.

Превентивні заходи є дієвими як проти вже відомих, так і проти нових прихованих каналів передачі інформації в мережних протоколах.

Комбінація описаних способів дозволить максимально захистити систему від усіх типів прихованих каналів.

Висновок до розділу 3:

Для захисту від відомих прихованих каналів існують спеціальні утиліти та правила/сигнатури для мережних екранів, антивірусів або IPS & IDS. Існують також превентивні заходи боротьби з прихованими каналами. У ході дослідження було розроблено та реалізовано власний метод захисту від прихованих каналів за допомогою моніторингу та аналізу мережного трафіку на предмет наявності аномалій. Для моніторингу трафіку використовувалась комбінація таких інструментів як Wireshark та Elastic Stack. Особливість розробленого методу полягає у можливості детектувати

потенційні приховані канали, які ще не були відомі або досліджені, а тому не мають описаних сигнатур. Тобто за допомогою даного методу можливо виявити нові типи прихованих каналів передачі даних у мережних протоколах.

Кожен спосіб боротьби з прихованими каналами передачі інформації в мережних протоколах має свою сферу застосування. Лише системно вони забезпечують максимальний рівень захищеності системи від прихованих каналів.

ВИСНОВКИ

У роботі розглянуто поняття прихованого каналу передачі інформації у мережному протоколі, причини виникнення та історію розвитку даного явища. Проведено огляд відомих прихованих каналів та аналіз методів їх створення. Приховані канали класифіковано за рівнем стеку TCP/IP та протоколом, на якому вони працюють. Запропоновано стратегії виявлення та знешкодження прихованих каналів.

У практичній частині дослідження проведено огляд методів виявлення та знешкодження відомих прихованих каналів у мережних протоколах. Задля боротьби із прихованими каналами (на прикладі DNS тунелю) застосовано існуючі засоби. Розроблено та реалізовано власний метод виявлення прихованих каналів передачі інформації у мережних протоколах шляхом детектування аномалій мережного трафіку. Метод має значний потенціал виявлення нових, раніше не відомих прихованих каналів завдяки тому, що не фокусується на сигнатурах конкретних каналів, а проводить загальний моніторинг та аналіз мережного трафіку на предмет аномалій. У ході дослідження створено засоби моніторингу та аналізу мережного трафіку – вочер та дашборд. Доведено доцільність використання комбінації Wireshark та Elastic Stack з метою виявлення потенційного прихованого каналу. Розглянуто превентивні заходи боротьби із прихованими каналами.

Отримані результати можуть бути використані для створення комплексного засобу захисту системи від прихованих каналів передачі інформації у мережних протоколах. Розроблений метод може слугувати для виявлення нових, ще не описаних прихованих каналів, для яких не існує сигнатур.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Department of Defence, Department of defence trusted computer system evaluation criteria, Technical Report DOD 5200.28-ST., Supersedes CSC-STD-001-83, 1985
2. Gray, J., Countermeasures and tradeoffs for a class of covert timing channels, at: http://repository.ust.hk/dspace/bitstream/1783.1/25/1/tr94-18.pdf, 1994
3. Gallagher, P. R., A Guide to Understanding Covert Channel Analysis of Trusted Systems, National Computer Security Center, USA, 1993
4. Simmons, G. J., The prisoners problem and the subliminal channel, In: Chaum, D. (ed.) Crypto ’83, Advances in Cryptography, 1984
5. Mazurczyk, W., Szczypiorski, K., Steganography of VoIP Streams, In: Tari, Z. (ed.) On the Move to Meaningful Internet Systems (OTM 2008), Monterrey, Mexico, 2008
6. Lampson, B., A Note on the Confinement Problem, Communications of the ACM, 1973
7. Wolf, M., Covert channels in LAN protocols, In: Berson, T. A., Beth, T. (Eds.) On: Workshop on Local Area Network Security (LANSEC 89), Karlsruhe, FRG, 1989
8. Hintz, A., Covert Channels in TCP and IP Headers, DefCon 10, Las Vegas, Nevada, 2003
9. Rowland, C. H., Covert channels in the TCP/IP protocol suite, DoIS Documents in Information Science, 1997
10. Forte, D. V., SecSyslog: An Approach to Secure Logging Based on Covert Channels, On: First International Workshop of Systematic Approaches to Digital Forensic Engineering (SADFE 2005), Taipei, Taiwan, 2005
11. deGraaf, R., Aycock J., Jacobson Jr., M., Improved Port Knocking with Strong Authentication, On: the 21st Annual Computer Security Applications Conference (ACSAC 2005), Tucson, AZ, 2005

12. Feamster, N., Balazinska, M., Harfst, G., Balakrishnan, H., Karger, D., Infranet: Circumventing Web Censorship and Surveillance, On: the 11th USENIX Security Symposium, San Francisco, CA, 2002
13. Burnett, S., Feamster, N., Vempala, S., Chipping Away at Censorship Firewalls with User-Generated Content, On: the 19th USENIX conference on Security (USENIX Security'10), Washington, DC, 2010
14. Houmansadr, A., Riedl, T., Borisov, N., Singer, E.,I want my voice to be heard: IP over Voice-over-IP for unobservable censorship circumvention, On: the 20th NDSS Symposium 2013, San Diego,USA, 2013
15. Mazurczyk, W., Kotulski, Z., New VoIP Traffic Security Scheme with Digital Watermarking, In: G´orski, J. (ed.) proceedings of: SafeComp 2006, Gdansk, Poland, 2006
16. Mazurczyk, W., Kotulski, Z., New Security and Control Protocol for VoIP Based on Steganography and Digital Watermarking, Annales UMCS Informatica AI, 2006
17. Houmansadr, A., Kiyavash, N., Borisov, N., RAINBOW: A Robust And Invisible Non-Blind Watermark for Network Flows, On: the 16th NDSS Symposium 2009, San Diego, USA, 2009
18. Houmansadr, A., Borisov, N., SWIRL: A scalable watermark to detect correlated network flows, On: the 18th NDSS Symposium 2009, San Diego, USA, 2011
19. Wang, X., Chen, S., Jajodia, S., Tracking anonymous peer-to-peer voip calls on the internet, On: the 2005 ACM Conference on Computer and Communications Security, 2005
20. Wang, X., Reeves, D. S., Robust correlation of encrypted attack traffic through stepping stones by manipulation of inter packet delays, On: the 2003 ACM Conference on Computer and Communications Security, 2003
21. Zander S. Armitage G. Branch P. Covert channels and countermeasures in computer network protocols // IEEE Communications Surveys Tutorials, at: https://www.researchgate.net/publication/220250270, 2007

22. Prendergast P. Covert Channels over Network Traffic: Methods, Metrics, and Mitigations, at: https://www.dst.defence.gov.au/publication/covert-channels-over-network-traffic-methods-metrics-and-mitigations, 2017
23. Epishkina A. Kogos K. Factors Study of countermeasures against covert channels in IP networks, at: https://www.researchgate.net/publication/220250270, 2016
24. Mileva A. Panajotov B. Covert Channels in TCP/IP Protocol Stack, at: https://link.springer.com/article/10.2478/s13537- 014-0205-6, 2013
25. Davidoff S. Ham J. Network forensics : tracking hackers through cyberspace // Pearson Education Inc., at: https://www.oreilly.com/library/view/network-forensics-tracking/9780132565110/, 2012
26. Cabuk S. Network covert channels: Design, analysis, detection, and elimination // PhD Dissertation, Perdue University, at: https://www.researchgate.net/publication/27234356, 2006
27. Ahsan, K., Covert channel analysis and data hiding in TCP/IP, Master thesis, University of Toronto, 2002
28. Scott, C., Network Covert Channels: Review of Current State and Analysis of Viability of the use of X.509 Certificates for Covert Communications, Technical Report RHUL-MA-2008-11, Department of Mathematics, Roal Holloway, University of London, 2008
29. Rios, R., Onieva, J. A., Lopez, J., HIDE DHCP: Covert Communications Through Network Configuration Messages, On: 27th IFIP TC 11 Information Security and Privacy Conference, Greece, 2012
30. Ji, L., Fan, Y., Ma, C., Covert channel for local area network, On: IEEE International Conference of Wireless Communications, Networking and Information Security (WCNIS 2010), 2010
31. Anjan, K., Abraham, J., Behavioral Analysis of Transport Layer Based Hybrid Covert Channel, On: Third International Conference of Recent Trends in Network Security and Applications (CNSA 2010), India, 2010

32. Anjan, K., Abraham, J., Jadhav, M.: Design of Transport Layer Based Hybrid Covert Channel Detection Engine, 2010
33. Zou, X., Li, Q., Sun, S.-H., Niu, X., The Research on Information Hiding Based on Command Sequence of FTP Protocol, In: Khosla, R., Howlett, R. J., Jain, L. C. (Eds.) On: the 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2005), Australia, 2005
34. Jeng A. B., Abrams M. D., On Network Covert ChannelAnalysis, Proc. 3rd Aerospace Computer Security Conf., 1987
35. Proctor N. E., Neumann P. G., Architectural Implicationsof Covert Channels, Proc. 15th National Computer SecurityConf., 1992
36. Moskowitz I. S., Kang M. H., Covert Channels — Hereto Stay?, Proc. 9th Annual Conf. Computer Assurance, 1994
37. daemon9, LOKI2: The Implementation, Phrack Magazine,vol. 7, no. 51, Sept. 1997
38. Wei-Ming H., Reducing Timing Channels with Fuzzy Time, Proc. IEEE Computer Society Symp. Research in Security andPrivacy, 1991
39. Farnham G., Detecting DNS Tunneling, 2013
40. Green A., What is DNS Tunneling? A Detection Guide, 2020

ДОДАТОК А РОЗРОБЛЕНИЙ ВОЧЕР ДЛЯ ELASTIC STACK

```
{
  "trigger": {
    "schedule": {
      "interval": "5m"
    }
  },
  "input": {
    "search": {
      "request": {
        "search_type": "query_then_fetch",
        "indices": [
          "packets-*"
        ],
        "rest_total_hits_as_int": true,
        "body": {
          "size": 100,
          "query": {
            "bool": {
              "filter": [
                {
                  "range": {
                    "@timestamp": {
                      "gte": "now-{{ctx.metadata.query_period}}",
                      "lte": "now"
                    }
                  }
                },
                {
```

```
            "query_string": {
              "query": """ NOT ip_ip_dsfield:"00000000" """,
              "analyze_wildcard": true,
              "default_field": "*"
            }
          }
        ]
      }
    },
    "_source": {
      "excludes": []
    },
    "stored_fields": [
      "*"
    ],
    "script_fields": {},
    "docvalue_fields": [
      "@timestamp",
      "event_data.NewTime",
      "event_data.PreviousTime"
    ]
  }
 }
},
"condition": {
 "compare": {
  "ctx.payload.hits.total": {
   "gt": 0
  }
```

```
    }
  },
  "actions": {
    "orwell": {
      "webhook": {
        "scheme": "http",
        "host": "10.47.0.35",
        "port": 3443,
        "method": "post",
        "path": "/alerts",
        "params": {},
        "headers": {},
        "body": """{"alert.channel":"{{ctx.metadata.channel}}",
"alert.description":"Possible covert channel!", "alert.id":"a0166",
"alert.name":"packets_data", "alert.responsible.subteam":
"{{ctx.metadata.subteam}}", "alert.responsible.subteam_name":
"{{ctx.metadata.subteam_name}}", "alert.requestor.name":"kateryna.natsvin",
"alert.responsible.name":"kateryna.natsvin",
"alert.responsible.slackname":"kateryna.natsvin", "alert.severity":"High",
"alert.source.index":"packets", "alert.source.name":"packets_data",
"alert.source.type":"Watcher", "alert.text":" {{#ctx.payload._value}} Source =
{{ip_ip_src}} Type = IP DiffServ NOT 0 Count = {{count}}\nThreshold =
{{threshold}} \n{{/ctx.payload._value}}\n\n", "alert.title":"Possible covert
channel!", "details.url":"https://siem.pmcorp.loc/app/discover#/view/950c6f60-
9d40-11eb-af98-
95903c521c39?_g=(time:(from:'{{ctx.payload._value.0.starttime}}',mode:abso
lute,to:'{{ctx.payload._value.0.endtime}}'))",
"mitre.attack.platform":"Enterprise", "mitre.attack.tactic":"Command and
Control", "mitre.attack.technique":"Protocol Tunneling", "@metadata":
{"source":"watcher"}}"""
```

```
        }
      }
    },
    "metadata": {
      "subteam": "S01QC63CB4K",
      "channel": "security_alerter",
      "subteam_name": "SOC",
      "threshold": 300,
      "query_period": "5m"
    },
    "transform": {
      "script": {
        "source": "def starttime = Instant.ofEpochMilli(ctx.execution_time.getMillis()-600000); def endtime = ctx.execution_time; def payload = ctx.payload.aggregations.1.buckets.stream().filter(p -> p.doc_count>ctx.metadata.threshold).map(t -> { return ['count': t.doc_count, 'source': t.key, 'user': t.2.buckets.0.key, 'starttime': starttime, 'endtime': endtime]}).collect(Collectors.toList()); return payload;",
        "lang": "painless"
      }
    }
  }
```