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

До захисту допущено
Завідувач кафедри
_____ **Дмитро ЛАНДЕ**
(підпис)
«_____»_____2024 р.

Дипломна робота

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Системи, технології та математичні методи кібербезпеки»
спеціальності 125 «Кібербезпека»

на тему: Аналіз безпеки і методи виявлення атак WI-FI

Виконав: здобувач вищої освіти **IV** курсу, групи ФБ-04
(шифр групи)

Мартиненко Денис Олександрович_____
(прізвище, ім'я, по батькові) (підпис)

Керівник: к.т.н. доцент кафедри ІБ Коломицев Михайло Володимирович___
(посада, науковий ступінь, вчене звання, прізвище, ім'я, по батькові) (підпис)

Рецензент: к.ф.-м.н., доцент кафедри ММЗІ Южакова Г.О.
(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 – 2024 року

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

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

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Дмитро ЛАНДЕ
(підпис)
«___»_____2024 р.

ЗАВДАННЯ
на дипломну роботу здобувачу вищої освіти

Мартиненку Денису Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи «Аналіз безпеки і методи виявлення атак WI-FI»,

керівник роботи к.т.н. доцент кафедри ІБ Коломицев Михайло Володимирович,

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «31» травня 2024 р. № 2251-с

2. Термін подання здобувачем вищої освіти роботи «14» червня 2024 р.

3. Вихідні дані до роботи

Навчальний та робочий плани, наукова література та відомості про найпопулярніші типи загроз безпеки Wi-Fi мереж.

4. Зміст роботи

- Літературний аналіз основ безпеки WI-FI мереж та огляд існуючих інструментів моніторингу мережі
- Дослідження найпопулярніших видів атак на WI-FI мережі
- Розробка рекомендацій та програмного коду мовою програмування Python

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо)

6. Дата видачі завдання: «18» вересня 2023 р.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів дипломної роботи	Примітка
1	Отримання завдання	10.04.2024	
2	Вивчення літератури	11.04.2024 – 14.04.2024	
3	Проходження практики	15.04.2024 – 19.05.2024	
4	Написання плану роботи	20.05.2024 – 21.05.2024	
5	Написання першого розділу	22.05.2024 – 26.05.2024	
6	Написання другого розділу	27.05.2024 – 31.05.2024	
7	Написання третього розділу	01.06.2024 – 04.06.2024	
8	Написання четвертого розділу	05.06.2024 – 09.06.2024	
9	Написання п’ятого розділу	10.06.2024 – 13.06.2024	
10	Передзахист дипломної роботи	14.06.2024	
11	Захист дипломної роботи	21.06.2024	

Здобувач вищої освіти _____ Денис, МАРТИНЕНКО
(підпис) (**Власне ім’я, ПРІЗВИЩЕ**)

Керівник роботи _____ Михайло, КОЛОМИЦЕВ
(підпис) (**Власне ім’я, ПРІЗВИЩЕ**)

РЕФЕРАТ

Робота обсягом 71 сторінка включає 29 ілюстрацій, 1 таблицю, 17 джерел літератури та 1 додаток.

Метою роботи є розробка ПЗ для своєчасного виявлення атак, розробка рекомендаційних вказівок та аналіз існуючих підходів та інструментів.

Об’єктом дослідження є бездротові Wi-Fi мережі та їх безпека.

Предметом дослідження є методи захисту інформації в бездротовій мережі.

Основні результати роботи включають розробку методів для аналізу безпеки Wi-Fi мереж та виявлення потенційних атак, а також пропозиції щодо поліпшення захисту мереж. Рекомендації представлено на основі аналізу найбільш поширених атак на Wi-Fi мережі.

Додатково було розроблено програмний код мовою Python для своєчасного виявлення можливих атак на Wi-Fi мережу.

Дане програмне забезпечення показує інтегрований підхід до забезпечення належного рівня безпеки.

Простота використання дозволяє застосовувати програмне рішення широкому колу користувачів.

Інструмент буде корисним як для рядового користувача, так і для системних адміністраторів великих організацій.

Ключові слова: WI-FI, безпека мереж, атаки на WI-FI, аналіз безпеки, методи виявлення атак.

ABSTRACT

The 71-page paper includes 29 illustrations, 1 table, 17 references and 1 appendix.

The purpose of the work is the development of software for the timely detection of attacks, the development of guidelines, and the analysis of existing approaches and tools.

The object of research is wireless Wi-Fi networks and their security.

The subject of research is methods of information security in a wireless network.

The main results of the work include the development of methods for analyzing the security of Wi-Fi networks and identifying potential attacks, as well as suggestions for improving network security. Recommendations are presented based on the analysis of the most common attacks on Wi-Fi networks.

Additionally, a Python program code was developed to detect possible attacks on Wi-Fi networks promptly.

This software demonstrates an integrated approach to ensure an adequate level of security.

The ease of use allows the software solution to be used by a wide range of users.

The tool will be useful for both ordinary users and system administrators of large organizations.

Keywords: WI-FI, network security, WI-FI attacks, security analysis, attack detection methods.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

WEP - Wired Equivalent Privacy

WPA - Wi-Fi Protected Access

RC4 - Rivest Cipher 4

IEEE - Institute of Electrical and Electronics Engineers

TKIP - Temporal Key Integrity Protocol

AES - Advanced Encryption Standard

SAE - Simultaneous Authentication of Equals

KRACK - Key Reinstallation Attacks

OSI - Open Systems Interconnection model

HTTP/HTTPS - Hypertext Transfer Protocol / Hypertext Transfer Protocol Secure

HSTS - HTTP Strict Transport Security

SSH - Secure Shell

IP - Internet Protocol

TCP - Transmission Control Protocol

UDP - User Datagram Protocol

ICMP - Internet Control Message Protocol

VPN - Virtual Private Network

ADSL - Asymmetric Digital Subscriber Line

ARP - Address Resolution Protocol

DNS - Domain Name System

MiTM - Man in The Middle

MAC - Media Access Control

IDS - Intrusion Detection System

IPS - Intrusion Prevention System

WPS - Wi-Fi Protected Setup

DoS/DDoS - Denial of Service / Distributed Denial of Service

ВСТУП

Сучасний світ важко уявити без інтернету та можливостей, які він надає. Станом на 2024 рік, передача понад 60% всього мережевого трафіку відбувається за допомогою Wi-Fi мереж. Згідно зі статистичними даними, залежність від бездротових мереж стрімко зростає, що в свою чергу збільшує ризики, пов’язані з їх безпекою.

Питання захисту даних у бездротових мережах є не тільки значним технічним викликом, але й важливим аспектом забезпечення приватності та корпоративної безпеки. Зловмисники невпинно розробляють нові способи атак, що робить постійне оновлення знань та інструментів захисту критично важливим завданням сьогодення.

Актуальність роботи визначається стрімким зростанням залежності суспільства від бездротових технологій та необхідністю забезпечення надійного захисту інформації в умовах постійно зростаючих кіберзагроз.

Сучасні Wi-Fi мережі використовуються в різноманітних сферах життєдіяльності, від домашнього інтернету до корпоративних мереж, що робить їх захист пріоритетним завданням у сфері кібербезпеки. Розробка та впровадження ефективних методів виявлення та нейтралізації атак на Wi-Fi мережі є актуальною проблемою, вирішення якої забезпечить безпечне використання бездротових технологій.

Мета дослідження полягає у розробці ефективних методів захисту Wi-Fi мереж від сучасних кібератак, аналізі існуючих підходів та інструментів для забезпечення безпеки бездротових мереж, а також у розробці рекомендацій для підвищення їхньої захищеності.

Завданням дослідження є проведення аналізу теоретичних основ безпеки Wi-Fi мереж, включаючи стандарти, методи шифрування та аутентифікації.

Також було оцінено існуючі інструменти для моніторингу та аналізу безпеки Wi-Fi, такі як Wireshark, tcpdump та CommView.

Визначено найпоширеніші типи атак на Wi-Fi мережі та методи їх виявлення.

Розроблено заходи протидії та рекомендації для підвищення рівня безпеки бездротових мереж.

Створено програмний код мовою Python для своєчасного виявлення можливих атак на Wi-Fi мережі.

Об’єктом дослідження є бездротові Wi-Fi мережі та їх безпека.

Предметом дослідження є методи виявлення та нейтралізації атак на Wi-Fi мережі.

Методи дослідження включають аналіз літературних джерел, огляд сучасних інструментів для аналізу безпеки Wi-Fi та розробку і тестування програмного забезпечення для виявлення атак.

Наукова новизна одержаних результатів полягає в розробці програмного забезпечення для своєчасного виявлення потенційних загроз.

Практичне значення одержаних результатів полягає в можливості їх застосування для підвищення безпеки бездротових мереж у різних сферах, зокрема в домашніх умовах, корпоративних мережах та громадських місцях. Запропоновані рекомендації та розроблене програмне забезпечення можуть бути використані для запобігання кібератакам та забезпечення стабільного функціонування Wi-Fi мереж.

У першій частині роботи ми зосереджуємось на теоретичних основах безпеки Wi-Fi, розглядаючи історію розвитку стандартів, основні технології та методи аутентифікації і шифрування.

Другий розділ присвячений аналізу інструментів для моніторингу та аналізу безпеки Wi-Fi, таких як Wireshark, Tcpdump, та CommView.

Третій розділ охоплює дослідження найпоширеніших типів атак на бездротові мережі та методи їх виявлення.

Четверта частина роботи зосереджується на розробці заходів протидії та рекомендаціях щодо підвищення рівня безпеки мереж.

Завершується робота розробкою програмного коду на мові Python для своєчасного виявлення потенційних атак, що демонструє практичне застосування теоретичних знань у реальних умовах.

Це дослідження спрямоване на вирішення актуальних проблем безпеки в сучасних бездротових мережах, виходячи з детального аналізу сучасних тенденцій та загроз, що забезпечує важливий внесок у поліпшення технологій захисту інформації.

1 ТЕОРЕТИЧНІ ОСНОВИ БЕЗПЕКИ WI-FI

Перший розділ дипломної роботи присвячений аналізу теоретичних основ безпеки Wi-Fi. У сучасному цифровому світі, де бездротові технології використовуються повсюдно, забезпечення безпеки Wi-Fi мереж стає вкрай важливим. Зловмисники постійно шукають вразливі точки в захисті бездротових мереж, що може призвести до втрати важливої особистої, корпоративної або державної інформації. Цей розділ має на меті детально описати стандарти, методи шифрування та аутентифікації, які застосовуються у Wi-Fi мережах, а також виділити головні вразливості цих мереж.

Розділ починається з огляду стандартів Wi-Fi, таких як WEP, WPA, WPA2, і WPA3, де кожен стандарт розглядається з точки зору його історії, основних характеристик, а також сильних і слабких сторін у контексті захисту даних. Далі аналізуються методи шифрування та аутентифікації, що використовуються для захисту від несанкціонованого доступу та забезпечення конфіденційності переданих даних. Крім того, в розділі висвітлюються основні вразливості, що зустрічаються в Wi-Fi мережах, з особливою увагою до тих, що найчастіше експлуатуються зловмисниками.

Завершується розділ розглядом сучасних підходів та інноваційних рішень у сфері безпеки Wi-Fi, що відкриває шлях до розробки нових технічних та організаційних заходів для зміцнення захисту бездротових мереж. Аналіз цих теоретичних основ є необхідним для подальшого вивчення інструментів аналізу та методів виявлення атак, описаних у наступних розділах роботи.

1.1 Огляд стандартів безпеки Wi-Fi (WEP, WPA, WPA2, WPA3)

Основою безпеки будь-якої бездротової мережі є стандарти, які визначають методи шифрування та аутентифікації. В історії розвитку Wi-Fi стандартів можна виділити кілька ключових етапів, кожен з яких вніс свої корективи в підвищення безпеки бездротових мереж. Огляд стандартів безпеки Wi-Fi включає чотири основні протоколи: WEP, WPA, WPA2 та WPA3, які було розроблено з метою посилення захисту бездротових мереж від різноманітних загроз.

WEP був першим спробою стандартизації безпеки бездротових мереж під егідою IEEE 802.11 у 1997 році. Він мав на меті забезпечити конфіденційність, аналогічну тій, що пропонують проводові мережі. Основний метод шифрування, який використовувався у WEP — це потоковий шифр RC4, який шифрував дані, використовуючи статичний ключ. Однак, основна слабкість WEP полягала в короткому ініціаційному векторі, що дозволяло зловмисникам легко відновлювати ключ шифрування через аналіз трафіку. Це вразливе місце виходило на передній план особливо при використанні декількох пакетів даних з однаковим вектором.

Розробка WPA була відповіддю на недоліки WEP і була представлена у 2003 році. WPA використовував протокол TKIP (Temporal Key Integrity Protocol), який генерував новий ключ для кожного пакету, використовуючи хеш-функції і переплетення ключів, що істотно підвищило рівень безпеки. TKIP також включав механізми перевірки цілісності даних, які захищали від змін даних під час передачі. Тим не менш, через використання RC4, TKIP все ще був вразливий до деяких специфічних атак, і це підтримувало потребу в розробці більш стійкого рішення.

У 2004 році було введено WPA2, який став новим стандартом для безпеки Wi-Fi мереж. Найбільша відмінність WPA2 від його попередників полягає в застосуванні AES (Advanced Encryption Standard), блочного шифру, що надав високий рівень безпеки і був значно стійкіший до криптографічних атак. WPA2 підтримує два режими: Personal (WPA2-PSK) для домашнього використання та Enterprise (WPA2-EAP) для використання в організаціях, де потрібен більш високий рівень безпеки.

Останній у цій серії, WPA3, був представлений у 2018 році, вносячи подальші покращення до безпеки бездротових мереж. Основною інновацією WPA3 є протокол SAE (Simultaneous Authentication of Equals), який забезпечує кращу захист від атак з вгадування паролів, особливо в відкритих мережах. WPA3 також покращує процеси шифрування та встановлення ключів, що забезпечує додаткову захищеність проти атак типу "людина посередині".

1.2 Методи шифрування та аутентифікації

Методи шифрування та аутентифікації в Wi-Fi мережах зазнали значних змін протягом років, стаючи дедалі складнішими та безпечнішими з введенням кожного нового стандарту безпеки, від WEP до WPA3. У своєму розвитку ці методи відіграли ключову роль у забезпеченні захисту передаваних даних і підтвердженні ідентичності користувачів.

Першим стандартом, WEP, використовувався потоковий шифр RC4, який, хоч і був швидким та ефективним, мав суттєві недоліки через статичну природу ключів і короткі ініціаційні вектори. Це призводило до вразливостей, де зловмисники могли відносно легко перехоплювати та дешифрувати дані.

WEP не мав сильної вбудованої системи аутентифікації, що дозволяло відносно легко використовувати слабкі місця для несанкціонованого доступу до мережі.

WPA приніс суттєві покращення з використанням TKIP, який не тільки включав динамічне ключове шифрування, але й переплетення ключів і алгоритми хешування для перевірки цілісності даних. Однак, TKIP все ще базувався на RC4, що підтримувало деякі вразливості до атак.

Запровадження WPA2 змінило підхід до шифрування, замінивши TKIP на AES, більш безпечний блочний шифр, який став золотим стандартом у шифруванні даних. AES складніший для взлому завдяки використанню довших ключів та складнішій структурі, що підвищило рівень захисту даних у Wi-Fi мережах. WPA2 також включав два режими аутентифікації: PSK для персонального використання та EAP для підприємств, що надавало гнучкість у виборі заходів безпеки відповідно до потреб користувачів.

Найновіший стандарт, WPA3, додатково зміцнив безпеку за допомогою протоколу SAE, який покращує аутентифікацію і робить її стійкішою до атак на вгадування паролів, навіть у відкритих мережах. Це стало значним кроком уперед у захисті приватних даних користувачів та зміцненні довіри до бездротових мереж.

1.3 Основні вразливості Wi-Fi мережі

З розвитком стандартів безпеки Wi-Fi від WEP до WPA3, з'являлися нові технологічні рішення, які сприяли покращенню безпеки, але також відкривали нові шляхи для потенційних атак.

WEP, як початковий стандарт, демонстрував значні вразливості, особливо з його використанням потокового шифру RC4 та статичних ключів шифрування.

Ці особливості дозволяли зловмисникам відносно легко відновлювати ключі шифрування через техніки аналізу трафіку. З появою WPA, було введено TKIP, який забезпечував динамічне оновлення ключів, проте все ще використовував RC4, залишаючи вразливості для специфічних атак, таких як "chop-chop" атака, яка могла ефективно зламати шифрування.

Запровадження WPA2 принесло значний прогрес з використанням AES, який є значно більш стійким до криптографічних атак, ніж його попередники. Проте, навіть цей стандарт показав свої вразливості, особливо з виявленням атаки KRACK, яка використовувала слабкості в протоколі рукостискання для переінсталяції ключів, дозволяючи зловмисникам перехоплювати та дешифрувати трафік.

WPA3, найновіший стандарт, був розроблений для подолання багатьох із цих вразливостей, запроваджуючи поліпшені механізми аутентифікації та шифрування. Однак, як показують дослідження, навіть цей стандарт не є повністю невразливим, з потенційними слабкими місцями у реалізації SAE, які можуть бути експлуатовані при певних умовах.

Висновки до розділу 1

Цей розділ наголошує на важливості розуміння історичного контексту та технічних деталей стандартів безпеки, що дозволяє краще оцінити поточні загрози та потенційні напрямки для подальших досліджень і розвитку в області безпеки бездротових мереж.

Аналіз вразливостей показав, що незалежно від зусиль по зміцненню безпеки, жоден системний механізм не є цілковито невразливим. Атаки, такі як атаки на WEP через повторення ініціаційного вектора, атаки "chop-chop" на WPA, KRACK на WPA2 та потенційні вразливості SAE в WPA3, підкреслюють необхідність неперервного моніторингу, оновлення безпеки та розробки більш стійких технологій.

Важливість комплексного підходу до безпеки, який включає як технічні рішення, так і правила поведінки користувачів, стає ключовим правилом у боротьбі з кіберзагрозами сучасного світу.

Під час аналізу та порівняння методів захисту можна відзначити, що технології обмеження доступу часто не забезпечують достатнього захисту в комп'ютерних мережах. Щодо методів авторизації та шифрування, то ефективний захист можливий лише за умови використання сучасних протоколів та алгоритмів у поєднанні з правильним налаштуванням мережевого обладнання, що забезпечує достатній рівень безпеки.

2 ВИБІР ІНСТРУМЕНТІВ ДЛЯ АНАЛІЗУ БЕЗПЕКИ WI-FI МЕРЕЖІ

Цей розділ має ключове значення для забезпечення ефективного захисту інформації, що передається в бездротових мережах. Враховуючи, що кіберзагрози постійно еволюціонують, важливо мати засоби для глибокого аналізу та виявлення потенційних вразливостей мережі. Саме тому цей розділ зосереджений на детальному огляді інструментів, які застосовуються для моніторингу та захисту Wi-Fi мереж, таких як Wireshark, tcpdump, та CommView.

Аналіз починається з огляду Wireshark, універсального інструменту для захоплення та аналізу пакетів, який дозволяє користувачам детально вивчати мережевий трафік і ідентифікувати потенційні проблеми з безпекою. Wireshark забезпечує глибокий аналіз протоколів та може бути використаний для навчальних цілей, а також для комплексного аудиту мережевої інфраструктури.

Далі розділ переходить до розгляду tcpdump, інструменту командного рядка, який дозволяє здійснювати захоплення пакетів на більш низькому рівні. Tcpdump ідеально підходить для використання в системах без графічного інтерфейсу, і є незамінним інструментом для системних адміністраторів завдяки своїй гнучкості та мінімальному впливу на системні ресурси.

Крім того, аналізується CommView, інструмент, який спеціалізується на моніторингу бездротових мереж. CommView надає додаткові можливості для захисту Wi-Fi мереж, включаючи підтримку зловлення пакетів та аналізу спектру, що робить його корисним для виявлення несанкціонованого доступу та інших загроз безпеки.

Завершується розділ порівнянням цих інструментів, їхніми сильними та слабкими сторонами, а також обговоренням сценаріїв їх застосування. Це допоможе визначити, який інструмент найкраще відповідає потребам завдання. Розуміння того, як ефективно використовувати ці інструменти, є фундаментальним для підвищення рівня безпеки Wi-Fi мереж і захисту від потенційних кіберзагроз.

2.1 Огляд інструменту Wireshark

Wireshark – це широко поширений інструмент для захоплення та аналізу мережного трафіку, який активно використовується як для освітніх цілей, так і для усунення несправностей на комп'ютері або мережі. Wireshark працює практично з усіма протоколами моделі OSI, має зрозумілий для звичайного користувача інтерфейс і зручну систему фільтрації даних. Крім того, програма є кросплатформною і підтримує більшість популярних операційних систем.

Однією з головних можливостей програми є захоплення трафіку мережі. Запустимо програму, після чого нас відразу зустрічає стартове меню, на якому можна побачити доступні для захоплення інтерфейси комп'ютера, посібники від розробників програми та безліч інших цікавих речей.

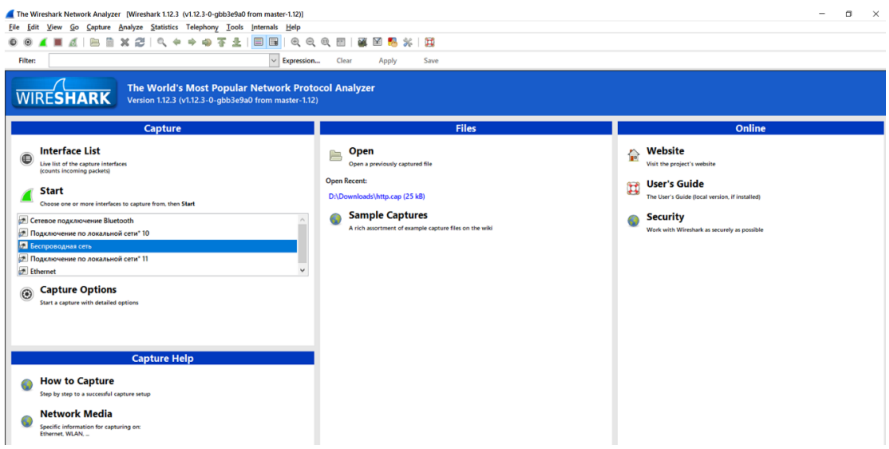


Рисунок 2.1 - Стартове меню

З усього цього нам необхідно звернути увагу на наступну область програми:

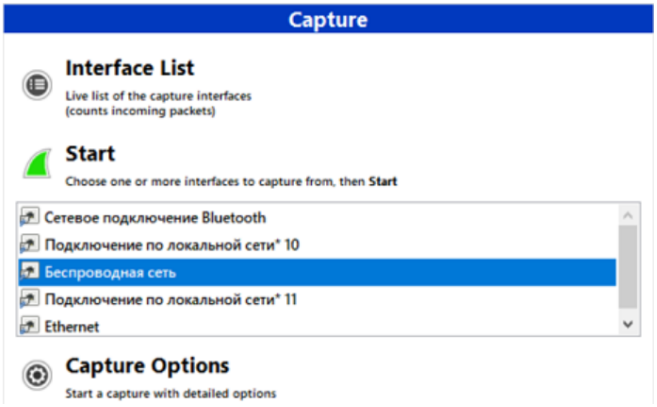


Рисунок 2.2 – Область захоплення трафіку

Тут потрібно вибрати той інтерфейс, через який ми підключені до Інтернету.

Мережевий інтерфейс – це програмне забезпечення, яке взаємодіє з мережним драйвером та з рівнем IP. Він забезпечує доступ для рівню IP до всіх наявних мережних адаптерів, трафік яких ми будемо перехоплювати. Найчастіше у програмі Wireshark можна зустріти мережний інтерфейс бездротової мережі (Wi-Fi) та кабельний (Ethernet).

Тема дипломної роботи напряму зв’язана з Wi-Fi, тому ми виконуємо захоплення "Бездротової мережі", після чого натискаємо "Start" та отримаємо наступний результат:

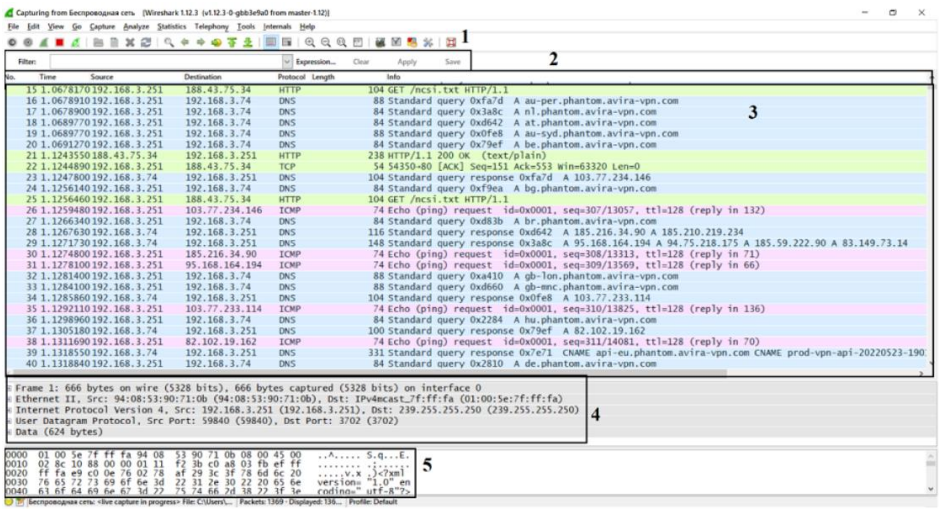


Рисунок 2.3 – Огляд початку процесу захоплення трафіку

Розглянемо докладніше це вікно за пунктами, вказаними на ньому:

1. Панель фільтрів дозволяє знайти необхідну інформацію.
2. Панель найменувань, що поділяє інформацію з пункту 3 на номер, час від початку захоплення трафіку, джерело та адресат, а також протокол, розмір пакета і невелику інформацію про мережевий пакет.
3. Панель пакетів оновлюється в реальному часі. Тут інформація про пакети розділена на стовпці, визначені на панелі найменувань.
4. Панель рівнів, що описує рівні моделі OSI вибраного мережного пакета.
5. Панель метаданих, що представляє дані у шістнадцятковому коді та символах.

Багато програм для передачі інформації використовують протокол HTTP, який дозволяє отримувати різні ресурси з інтернету та відправляти дані назад. Розглянемо один із пакетів, переданих за протоколом HTTP:

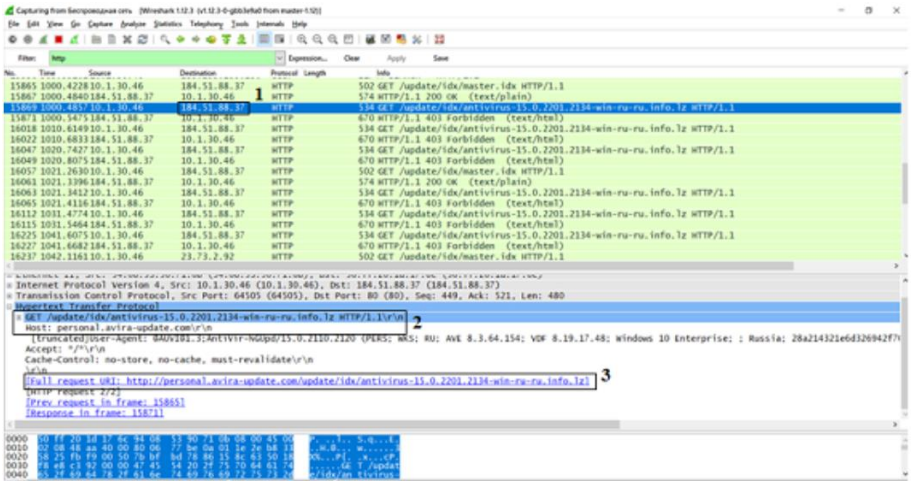


Рисунок 2.4 – Огляд пакету HTTP

У протоколі HTTP для передачі даних використовуються запити GET (призначений для отримання даних) та POST (призначений для надсилання даних).

На малюнку в полі 1 ми бачимо IP-адресу адресата. У полі 2 ми дізнаємося, що сервер антивіруса надіслав запит GET для того, щоб запитати деякі дані про комп'ютер, які необхідні для коректного оновлення програми. В полі 3 ми бачимо те, що цей запит виглядає як URL.

З цього прикладу зрозуміло, що інтерфейс програми «юзер-френдлі», але назріває ще одне питання: Як знайти необхідний пакет для аналізу серед сотень, а інколи й тисяч інших? На це питання Wireshark відповідає своєю можливістю фільтрації пакетів. У спеціальному полі Filter можна ввести необхідні команди або скористатися підказками.

Рисунок 2.5 – Огляд поля Filter

Найчастіше використовується фільтрація за IP-адресами, за номерами порту та протоколами.

Фільтрування за IP-адресою дозволяє нам переглядати всі пакети, що надходять від кого-небудь або ті, що йдуть будь-кому. Наприклад, відберемо всі пакети, що надходять від IP-адреси 10.1.30.46 за допомогою введення у фільтрі «ip.src == x.x.x.x».

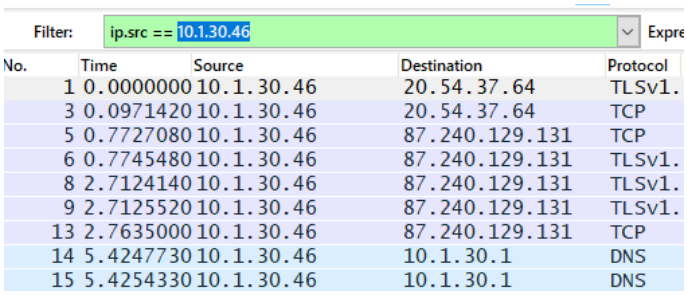


Рисунок 2.6 – Огляд команди "ip.src"

Також можна відфільтрувати трафік мережі IP-адресою одержувача пакетів за допомогою команди «ip.dst == x.x.x.x».

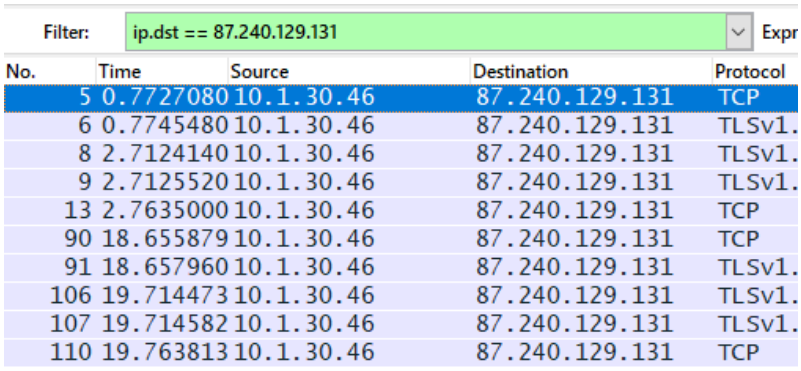


Рисунок 2.7 - Огляд команди "ip.dst"

Крім того, можна побачити пакети незалежно від напрямку трафіку за допомогою «ip.addr == x.x.x.x».

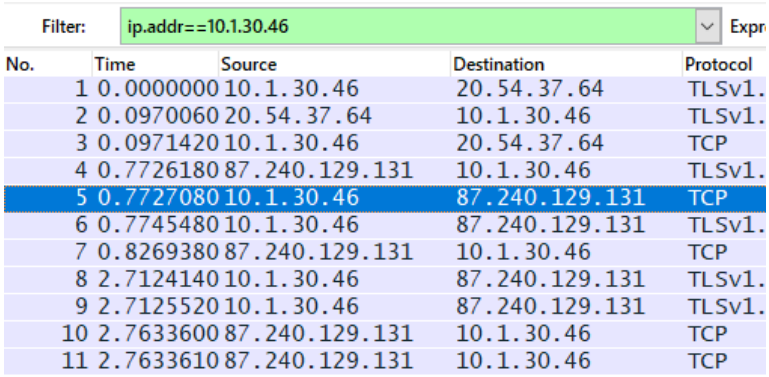


Рисунок 2.8 – Огляд команди "ip.addr"

Для фільтрації за номером порту використовується «.port = x» після назви протоколу. Наприклад, для перегляду TCP-порту 80, який використовується для незашифрованого трафіку HTTP, використовуємо команду «tpc.port == 80».

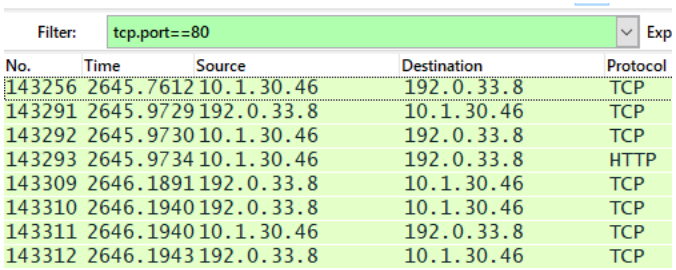


Рисунок 2.9 – Огляд команди "tcp.port"

Для фільтрації трафіку за пакетами протоколів необхідно просто ввести назву протоколу. Фільтри можна комбінувати за допомогою логічних операторів І «and/&&», АБО «or/||» і НЕ "not/!"

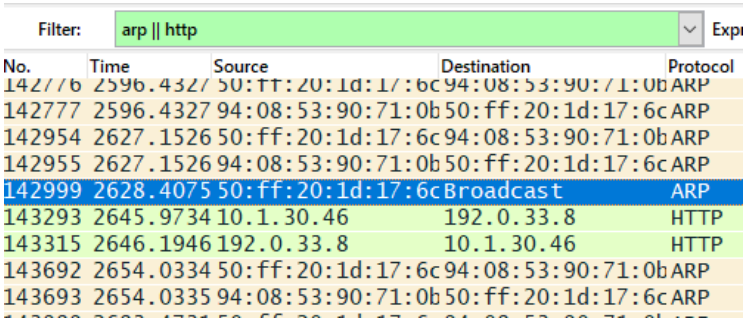


Рисунок 2.10 – Огляд логічних операторів

2.2 Огляд інструменту tcpdump

Tcpdump — це інструмент аналізу пакетів з командного рядка. Подібно до Wireshark, ми можемо використовувати tcpdump для перехоплення та аналізу пакетів, усунення проблем з'єднання та пошуку потенційних проблем безпеки в мережі. Tcpdump — це портативна утиліта командного рядка, яку можна використовувати навіть тоді, коли графічний інтерфейс недоступний, а Wireshark не встановлено.

Tcpdump є одним з найстаріших та найбільш використовуваних інструментів для захоплення мережевих пакетів на платформах Unix та Linux. Цей інструмент командного рядка дозволяє користувачам здійснювати детальний моніторинг всього мережевого трафіку, що проходить через інтерфейс, на якому він запущений. Основною перевагою tcpdump є його легкість та гнучкість, які дозволяють йому працювати на різних пристроях, від серверів до вбудованих систем.

Tcpdump працює, фільтруючи трафік за допомогою потужного синтаксису для визначення, які пакети слід захоплювати. Це включає можливість фільтрувати трафік за IP-адресами, типами протоколів, портами та іншими параметрами. Ця здатність до глибокої фільтрації робить tcpdump надзвичайно корисним для виявлення специфічних видів мережевої активності, які можуть бути ознаками несанкціонованого доступу або інших форм мережевих атак.

Інший аспект, який робить tcpdump цінним інструментом, полягає у його здатності зберігати захоплені пакети у файлі «.pcap», який можна аналізувати за допомогою інших інструментів, таких як Wireshark. Це дозволяє аналітикам безпеки вивчати мережевий трафік більш детально та у свій час, забезпечуючи змогу проводити глибокі аналізи мережевих інцидентів та розробляти відповідні стратегії відповіді на інциденти.

Однак, важливо зазначити, що tcpdump має свої обмеження, особливо коли мова йде про використання у великих або складних мережах. Його текстовий вивід може бути складно інтерпретувати в реальному часі, і він не має графічного інтерфейсу, що може ускладнити роботу менш досвідчених користувачів. Також, він вимагає налаштувань безпеки на використанні, оскільки здатен перехоплювати всі пакети на мережевому інтерфейсі, що може викликати проблеми з конфіденційністю даних.

Після невеликого передслова про інструмент, слід розглянути практичні моменти.

Перш за все, нам потрібно використати прапорець “-D” для переліку інтерфейсів, доступних для перехоплення:

```
└─$ tcpdump -D
1.eth0 [Up, Running, Connected]
2.any (Pseudo-device that captures on all interfaces) [Up, Running]
3.lo [Up, Running, Loopback]
4.docker0 [Up, Disconnected]
5.bluetooth-monitor (Bluetooth Linux Monitor) [Wireless]
6.nflog (Linux netfilter log (NFLOG) interface) [none]
7.nfqueue (Linux netfilter queue (NFQUEUE) interface) [none]
8.dbus-system (D-Bus system bus) [none]
9.dbus-session (D-Bus session bus) [none]
```

Рисунок 2.11 – Перелік мережевих інтерфейсів

Можна використовувати прапорець "-i" для вказівки інтерфейсу, з якого ми хочемо здійснювати захоплення пакетів. Якщо вказати "any" як інтерфейс, буде захоплено дані з усіх активних інтерфейсів. Також можна налаштувати tcpdump на захоплення пакетів з певного інтерфейсу.

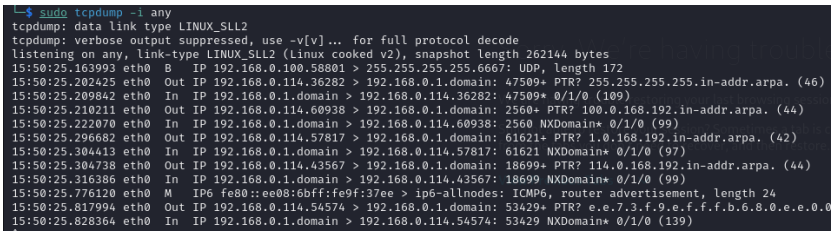


Рисунок 2.12 – Приклад захоплення пакетів

Тепер ми бачимо деталі пакетів у терміналі. Tcpdump продовжить захоплювати пакети, поки ми не перервемо його, натиснувши "Ctrl+C". Для регулювання рівня детальності виводу tcpdump можна використовувати прапорець "-v".

Tcpdump має різноманітні фільтри, які дозволяють захоплювати лише пакети, що відповідають нашим критеріям. Для прикладу ми можемо фільтрувати трафік захоплення на основі протоколу:

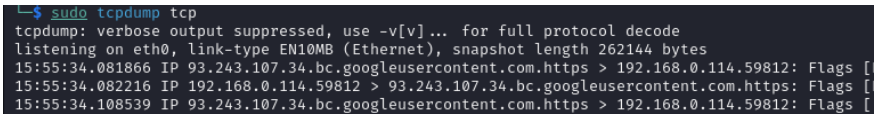


Рисунок 2.13 – Приклад фільтрації трафіку за протоколом TCP

Якщо нас цікавить трафік лише для певного порту, то можна використовувати фільтр "port" для спрямування аналізу. Ця команда захопить увесь трафік, що проходить через порт 80. Якщо ми хочемо бути більш конкретними і захопити трафік, в якому порт 80 є портом відправлення або призначення, можна використовувати наступні команди:

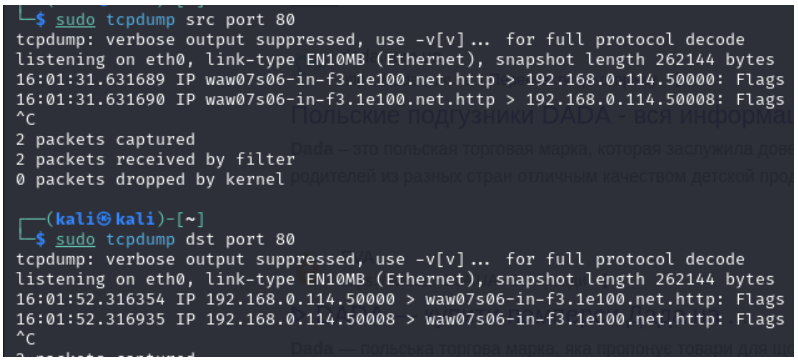


Рисунок 2.14 – Приклад фільтрації трафіку за портом відправлення чи призначення

З іншого боку, якщо нас цікавить трафік лише для певного хоста, то можна використовувати фільтр "host". Фільтр "host" також можна комбінувати з фільтрами "src" (джерело) або "dest" (призначення).

Також, ми можемо комбінувати кілька фільтрів у tcpdump. Для комбінації фільтрів можна використовувати булеві оператори, такі як "and" (і) та "or" (або).

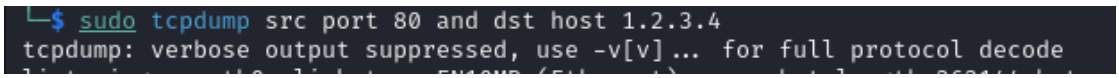


Рисунок 2.15 – Приклад комбінування декількох фільтрів

Результати захоплення пакетів можна зберігати у файлі замість того, щоб виводити їх на екран, використовуючи прапорець "-w". Якщо нам потрібно, щоб tcpdump одночасно зберігав і виводив пакети, слід використовувати прапорець "—print" разом з прапорцем "-w". Збережений файл пізніше можна читати за допомогою прапорця "-r".

Щоб прочитати вміст захоплених пакетів у tcpdump використовується прапорець "-A".

2.3 Огляд інструменту CommView

CommView — це потужний інструмент, спеціалізований на моніторингу та аналізі бездротових мереж, що надає детальний огляд на мережевий трафік і допомагає у виявленні потенційних загроз.

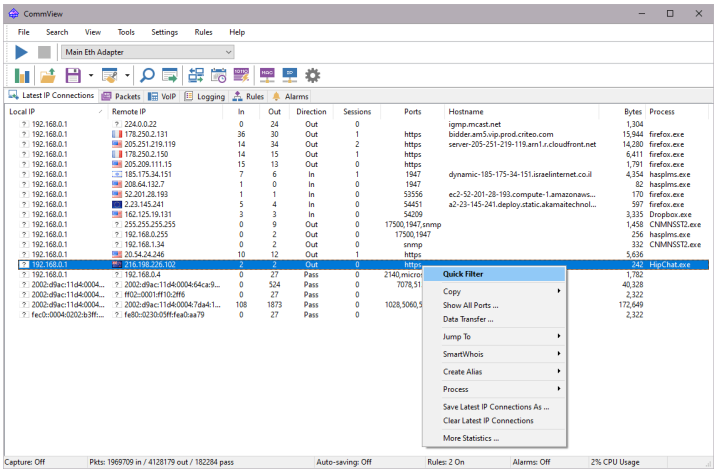


Рисунок 2.16 – Демонстрація інтерфейсу CommView

CommView дозволяє налаштувати гнучкі правила фільтрації пакетів, вибираючи специфічні протоколи для моніторингу і сортування пакетів за різними характеристиками, такими як розмір чи заголовок. Це надзвичайно корисно для аналізу мережевого трафіку та ідентифікації специфічних видів мережевих активностей або загроз.

Підтримка широкого асортименту протоколів, включаючи найбільш розповсюджені застосункові протоколи, а також можливість реконструкції TCP-сесій та UDP-потоків, робить CommView ефективним інструментом для професіоналів. Цей інструмент дозволяє аналізувати трафік на всіх рівнях, від протоколів високого рівня до «сирих» даних пакетів нижчого рівня, як TCP, UDP, ICMP.

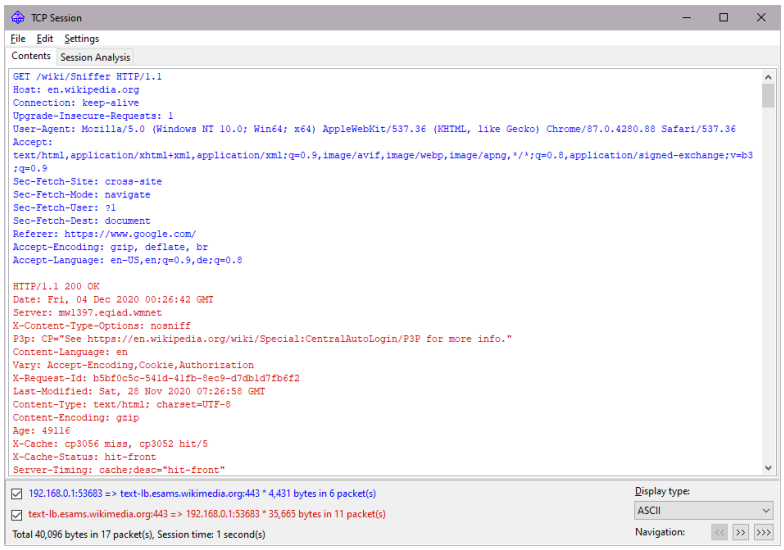


Рисунок 2.17 – Демонстрація реконструкції TCP сесії

CommView також пропонує можливості моніторингу не тільки Wi-Fi трафіку, але й мережевого трафіку через VPN, а також через аналогові, мобільні, ADSL та ISDN модеми, завдяки спеціальному драйверу, що встановлюється в систему. Це розширює сферу застосування інструменту, дозволяючи аналізувати різноманітні види мережевих середовищ.

Крім того, включення генератора пакетів надає користувачам можливість відправляти спеціалізовані пакети на заданий Ethernet-інтерфейс для тестування мережі, а зручний переглядач лог-файлів дозволяє з легкістю відкривати та аналізувати файли журналів.

З усіма цими функціональними можливостями, CommView залишається важливим інструментом для будь-якого мережевого адміністратора або фахівця з безпеки, незважаючи на високу вартість ліцензії. Вона може бути виправданою для професійного використання, але для одноразових перевірок можуть бути розглянуті більш дешеві або безкоштовні альтернативи.

2.4 Порівняння оглянутих інструментів та вибір кращого

У попередніх пунктах було розглянуто ключові особливості таких інструментів, як Wireshark, tcpdump та CommView. За результатами аналізу було створено порівняльну таблицю для візуального представлення їх характеристик і функціональностей.

Таблиця 2.1 – Порівняння особливостей аналізаторів трафіку

Особливість	**Wireshark**	**tcpdump**	**CommView**
Інтерфейс	Графічний користувацький інтерфейс (GUI)	Командний рядок (CLI)	Графічний користувацький інтерфейс (GUI)
Протоколи	Підтримує понад 2000 протоколів	Підтримує багато протоколів, але менше ніж Wireshark	Підтримує багато протоколів, зосереджених на Wi-Fi
Платформа	Windows, Linux, macOS	Більшість Unix-подібних систем, Windows з Cygwin	Тільки Windows
Основне використання	Глибокий аналіз мережевого трафіку	Швидке захоплення пакетів, скриптований аналіз	Спеціалізовано для аналізу Wi-Fi та Ethernet мереж
Функціональність	Великий набір інструментів для аналізу, фільтрації та статистики	Базовий аналіз та фільтрація з командного рядка	Зосереджений на візуалізації та розширеному діагностуванні мережі
Збереження сесій	Може зберігати захоплені сесії для подальшого аналізу	Здатний зберігати дані у форматі, який підходить для подальшого аналізу	Зберігає дані у власних форматах для подальшого аналізу
Спеціальні функції	Детальний декодер пакетів, налаштування плагінів	Ефективне використання ресурсів системи, ідеально підходить для вбудованих систем	Потужні інструменти для зловлення пакетів, оцінка мережі
Ліцензування	Вільне програмне забезпечення (GNU GPL)	Вільне програмне забезпечення (BSD ліцензія)	Комерційне програмне забезпечення

З огляду на аналізовані дані та згадану порівняльну таблицю, для більшості задач аналізу безпеки Wi-Fi мереж вибір падає на Wireshark як найбільш відповідний інструмент. Підставами для такого вибору служать наступні аргументи:

1. Wireshark пропонує зручний та інтуїтивно зрозумілий графічний інтерфейс, що робить його доступнішим для користувачів з різним рівнем досвіду. Це особливо важливо для комплексного аналізу мережевого трафіку, де візуалізація даних може значно спростити ідентифікацію аномалій та забезпечення безпеки.
2. Wireshark підтримує аналіз сотень протоколів, що дозволяє йому ефективно використовуватися в різноманітних мережевих середовищах та забезпечувати глибоке розуміння мережевих процесів.
3. Wireshark надає потужні інструменти для фільтрації та аналізу мережевого трафіку, включаючи можливості детального вивчення кожного пакета. Функціональність фільтрації дозволяє користувачам зосередитись на конкретних аспектах мережевого трафіку, відокремлюючи важливу інформацію від загального потоку даних.
4. Wireshark має велику користувацьку базу та активну спільноту, що забезпечує обмін знаннями та підтримку. Це робить його особливо цінним ресурсом для навчання та вдосконалення навичок у галузі мережевої безпеки.

Зважаючи на ці переваги, Wireshark є оптимальним вибором для широкого спектру завдань, пов'язаних з моніторингом, діагностикою та забезпеченням безпеки мереж. Він забезпечує глибокий інсайт у мережеву активність, що є ключовим для ефективного аналізу та виявлення потенційних загроз у сучасних Wi-Fi мережах.

Висновки до розділу 2

Розділ охоплює аналіз інструментів для моніторингу та аналізу безпеки Wi-Fi мереж, підкреслює важливість вибору відповідного інструменту залежно від конкретних потреб та завдань, які стоять перед фахівцем з мережевої безпеки. Розглянувши такі інструменти як Wireshark, tcpdump, та CommView, можна зробити висновки щодо їхніх сильних та слабких сторін, а також їхньої придатності до різних сценаріїв використання.

Wireshark вирізняється завдяки своїй універсальності та глибокому аналізу даних, що робить його відмінним вибором для більшості завдань моніторингу та діагностики мереж. Його графічний користувацький інтерфейс та розширені можливості фільтрації сприяють легкості використання та глибокому аналізу мережевого трафіку.

Tcpdump є корисним для ситуацій, що вимагають легкості інструменту та його здатності працювати в обмежених умовах, таких як на вбудованих системах або при необхідності швидкої автоматизації моніторингу через командний рядок.

CommView спеціалізується на аналізі Wi-Fi мереж і пропонує специфічні функції, які можуть бути особливо корисними для детального аналізу бездротового трафіку, з великим акцентом на діагностику та управління безпекою в бездротових мережах.

Обираючи між цими інструментами, варто враховувати специфіку своєї мережі, рівень своїх технічних знань та конкретні безпекові потреби. Хоча кожен інструмент має свої переваги, Wireshark є найоптимальнішим варіантом через його багатофункціональність і зручність використання.

3 ОГЛЯД ОСОБЛИВОСТЕЙ ТА СПОСОБІВ ВИЯВЛЕННЯ НАЙПОПУЛЯРНІШИХ АТАК НА WI-FI МЕРЕЖІ

В попередньому розділі основним інструментом для аналізу трафіку з метою виявлення найпопулярніших атак було обрано програмний застосунок Wireshark. Кожній із загроз притаманні особливі характеристики, які при певному наборі знань можна легко ідентифікувати, але існує загальний алгоритм, до якого слід за все звертатися першочергово.

Перший крок у виявленні потенційних загроз Wi-Fi - це захоплення трафіку мережі за допомогою Wireshark. Це дозволяє отримати доступ до всіх передач даних у мережі для подальшого аналізу.

Після захоплення трафіку необхідно використати фільтри для відсіювання непотрібної інформації та концентрації уваги на пакетах, що потенційно можуть містити загрози.

Аналіз заголовків пакетів дозволяє виявити незвичайні або підозрілі активності, такі як незаконне підключення до мережі або атаки переповнення буфера.

Wireshark дозволяє виявляти незашифрований трафік у Wi-Fi мережі, що може свідчити про вразливості в налаштуваннях шифрування мережі.

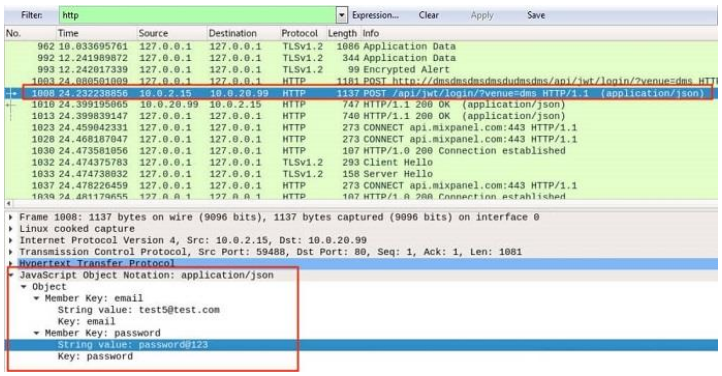


Рисунок 3.1 – Приклад компрометації паролю користувача шляхом аналізу незашифрованого трафіку

Ще однією загрозою для Wi-Fi мереж є атаки "Man-in-the-Middle коли зловмисник отримує доступ до комунікації між двома сторонами. Wireshark може виявити підозрілі зміни у пакетах, що можуть свідчити про такі атаки.

Wireshark дозволяє аналізувати шаблони трафіку та виявляти аномальну активність, таку як великий обсяг даних від одного джерела або незвичайні запити.

Загалом розділ присвячений огляду особливостей та способів виявлення найпопулярніших атак на Wi-Fi мережі. У цьому розділі буде розглянуто найпоширеніші типи атак на Wi-Fi мережі, такі як Man-in-the-Middle (MiTM), ARP spoofing та DNS spoofing.

Детальний аналіз цих атак дозволить зрозуміти їхні принципи роботи, а також виявити слабкі місця у захисті бездротових мереж. Крім того, буде описано методи виявлення цих атак за допомогою існуючих інструментів моніторингу та аналізу мережевого трафіку, зокрема Wireshark.

Особлива увага приділяється практичним аспектам виявлення атак, включаючи використання фільтрів для аналізу трафіку, ідентифікацію підозрілих шаблонів трафіку та аналіз заголовків пакетів. Ці знання дозволять ефективніше виявляти загрози та своєчасно вживати заходів для нейтралізації атак, забезпечуючи таким чином безпеку Wi-Fi мереж.

Таким чином, третій розділ роботи має на меті не лише надати теоретичне розуміння методів атак на Wi-Fi мережі, але й запропонувати практичні рекомендації для їх виявлення та протидії, що є важливим кроком у забезпеченні надійного захисту бездротових мереж у сучасних умовах.

3.1 Атака MiTM

Атака "Man in the Middle" (MitM) — це тип кібератаки, де зловмисник вставляє себе в комунікаційний канал між двома сторонами, які вважають, що вони безпосередньо спілкуються один з одним. Це дозволяє зловмиснику перехоплювати, змінювати або фабрикувати повідомлення між оригінальними сторонами без їхньої згоди або знання.

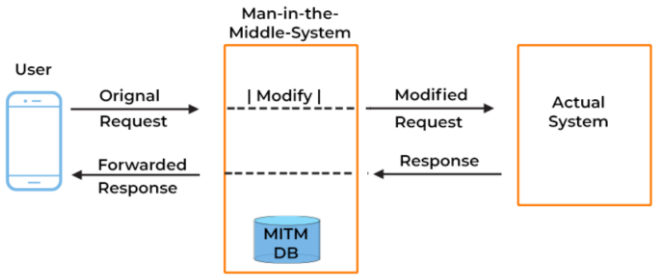


Рисунок 3.2 – Принцип атаки "Man-in-the-Middle"

ARP (Address Resolution Protocol) часто використовується у атаках MitM для отримання контролю над IP-адресами в мережі. Одним із вказівником ARP spoofing є зміна MAC-адреса для статичної IP-адреси, що можна перевірити шляхом спостерігання за ARP-пакетами.

При MitM атаках зловмисник може повторювати або відтворювати автентичні пакети. Використовуючи Wireshark, можна знайти підозрілі повторення пакетів або незвичайні затримки у передачі пакетів.

Якщо зловмисник використовує фальшивий сертифікат для перехоплення зашифрованого трафіку, Wireshark може показати невідповідності в сертифікатах TLS/SSL. Слід переглянути властивості SSL/TLS пакетів, особливо SSL сертифікат handshake повідомлення.

MitM атака може спричинити затримки у мережевому трафіку через обробку пакетів зловмисником. Аналізуючи часові відмітки і затримки між пакетами, можна ідентифікувати підозрілі затримки.

Також варто вивчити трафік, що йде до і від незнайомих або підозрілих IP-адрес і портів, особливо якщо вони не використовуються у звичайній мережевій активності.

3.2 Атака ARP spoofing

Атака ARP spoofing є специфічним видом атаки Man in the Middle, яка здійснюється на рівні локальних мереж (LAN). Цей метод використовується для підробки ARP (Address Resolution Protocol) повідомлень у мережі.

Зловмисник відправляє фальшиві ARP відповіді в мережу з метою асоціювати свою MAC-адресу з IP-адресою іншого пристрою на тій самій локальній мережі, часто шлюзу за замовчуванням. Це дозволяє зловмиснику перехоплювати весь трафік, призначений для цієї IP-адреси.

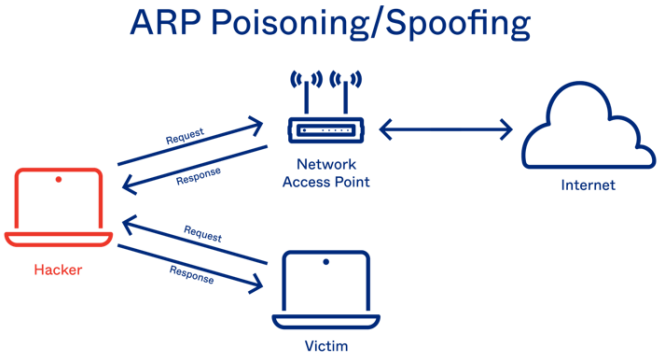


Рисунок 3.3 - Принцип атаки ARP spoofing

Основним методом виявлення ARP spoofing у програмі Wireshark є спостереження за відправленими ARP оголошеннями. Якщо було зафіксовано ARP оголошення, які стверджують, що IP-адреса пристрою має нову MAC-адресу, і є відомості, що цей пристрій не було змінено чи замінено на інший, це може бути знаком ARP spoofing.

Якщо одна MAC-адреса асоційована з кількома IP-адресами або наявні кілька ARP відповідей для однієї IP-адреси з різними MAC-адресами, це може бути індикатором атаки ARP spoofing.

Постійні зміни в ARP таблиці (наприклад, часті зміни MAC-адрес, що асоціюються з однією IP-адресою) можуть бути показником атаки.

Наявність неочікуваних ARP відповідей (unsolicited ARP responses), коли жодного запиту не було зроблено, може бути спробою ARP spoofing.

3.3 Атака DNS spoofing

Атака DNS spoofing, відома також як DNS cache poisoning, є видом кібератаки, при якій зловмисник підробляє відповіді DNS сервера або змінює DNS кеш з метою перенаправлення запитів до фальшивих або шкідливих сайтів. Це може призвести до крадіжки персональних даних, розповсюдження шкідливого програмного забезпечення або перехоплення мережевого трафіку.

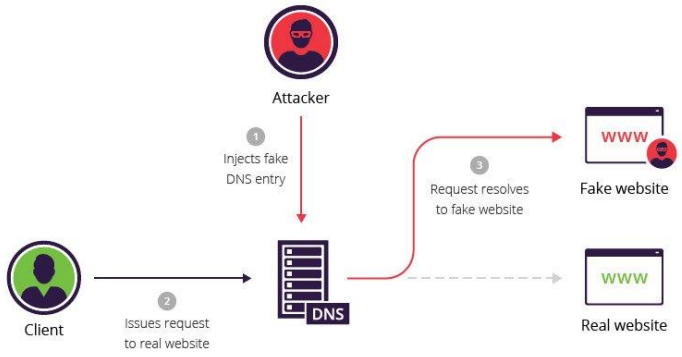


Рисунок 3.4 - Принцип атаки DNS spoofing

Необхідно звернути увагу на несподівані або повторювані відповіді DNS, особливо якщо вони приходять з неочікуваних IP адрес або містять неочікувані зміни в DNS записах.

Також треба звернути увагу на швидкість відповідей. Фальшиві відповіді часто приходять швидше, ніж законні, через відсутність реальної обробки запиту.

Також варто шукати кілька відповідей на один і той же запит, особливо якщо вони містять різні IP адреси. Це може вказувати на наявність зловмисника, який відправляє фальшиві відповіді.

Відповіді DNS, які пов'язують добре відомі домени з незвичайними або підозрілими IP адресами, можуть бути знаком DNS spoofing.

3.4 Демонстрація проведення атаки ARP Spoofing для подальшої реалізації атаки типу “Man-in-the-Middle”

Для більш детального розуміння популярності наведених вище атак в цьому розділі буде розглянуто метод проведення атаки ARP spoofing шляхом «отруєння» ARP таблиці. Це допоможе рядовому користувачу зрозуміти простоту проведення таких атак і, як наслідок, бути більш обережними при використанні Wi-Fi мереж.

Для проведення атаки було розгорнуто дві віртуальні машини. Kali Linux виступає в ролі машини зловмисника, а віртуальна машина з операційною системою Windows 11 – в ролі машини рядового користувача. Також на машині зловмисника було запущено програму Wireshark для подальшого детального огляду атаки.

Першим кроком проведення атаки є визначення IP та MAC адрес обох віртуальних машин, а також роутеру, до якого вони підключені.

Далі ми використовуємо утиліту netdiscover для проведення асоціації IP адрес на машині зловмисника.

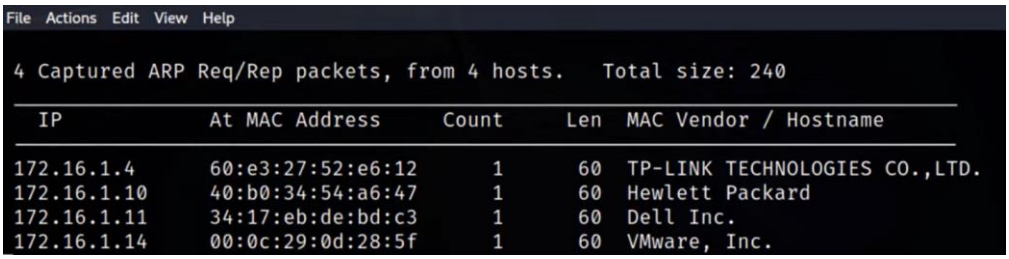


Рисунок 3.5 – Результат роботи утиліти Netdiscover

- 60:e3:27:52:e6:12 – це MAC адреса шлюзу за замовчуванням
- 172.16.1.14 – це ціль зловмисника

Все, що нам залишається надалі – це запустити утиліту arpspoof для проведення атаки:

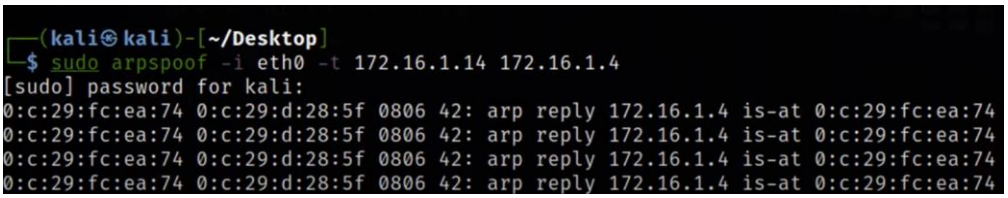


Рисунок 3.6 – Демонстрація проведення атаки ARP Spoofing для машини жертви

Тепер перейдемо на віртуальну машину жертви, та переглянемо ARP таблицю:

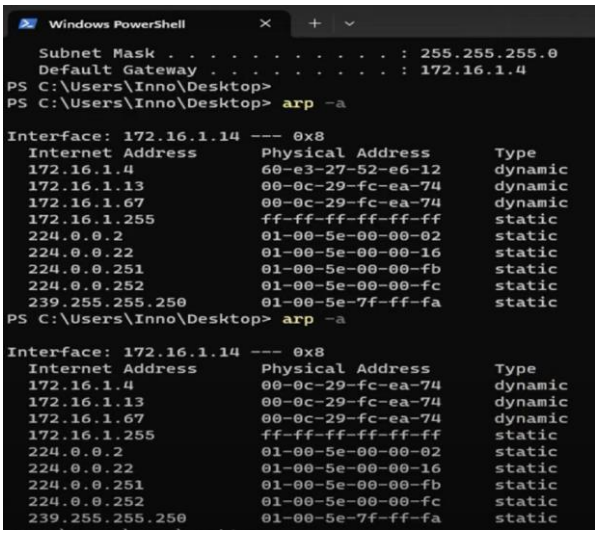


Рисунок 3.7 – Демонстрація «отруєння» ARP таблиці на віртуальній машині жертви

Як бачимо, зараз віртуальна машина вважає, що вона стала роутером, що свідчить про успішне «отруєння» ARP таблиці.

За аналогією, наступним кроком проведення атаки буде отруєння ARP таблиці роутера, для того, щоб він почав вважати машину зловмисника машиною жертви.

Рисунок 3.8 - Демонстрація проведення атаки ARP Spoofing для роутера

На цьому моменті атака завершується, а зловмисник може реалізовувати атаку MiTM.

Відкриємо програму Wireshark, застосуємо фільтр ARP для відсіювання непотрібних захоплених пакетів та проведемо аналіз.

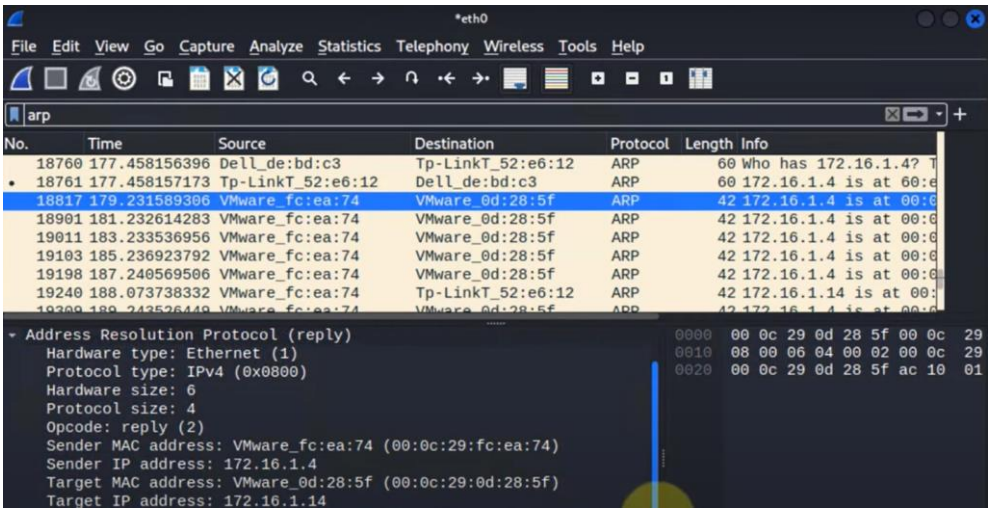


Рисунок 3.9 – Огляд захоплених ARP пакетів в програмі Wireshark

Як бачимо з аналізу пакету ARP, шлюзу за замовчуванням було присвоєно MAC адресу нашої віртуальної машини зловмисника і пакет далі відправився на віртуальну машину Windows 11 жертви, отже MAC адресу вдалося підробити, що свідчить про успішне виконання атаки зловмисником.

Висновки до розділу 3

У третьому розділі було проведено детальний огляд найпоширеніших атак на Wi-Fi мережі та методів їх виявлення. Розглянуті атаки, такі як Man-in-the-Middle (MiTM), ARP spoofing та DNS spoofing, представляють значну загрозу для безпеки бездротових мереж, оскільки дозволяють зловмисникам отримувати доступ до конфіденційних даних, порушувати роботу мережі та завдавати шкоди користувачам.

Аналіз особливостей цих атак показав, що кожна з них має свої специфічні методи реалізації та вимагає різних підходів до виявлення. Визначено, що ефективне виявлення атак можливе завдяки використанню інструментів моніторингу та аналізу мережевого трафіку, таких як Wireshark.

Важливим аспектом ефективного захисту Wi-Fi мереж є знання та розуміння методів атак, а також вміння своєчасно виявляти та нейтралізувати їх. Використання таких методів як аналіз заголовків пакетів, моніторинг шаблонів трафіку та ідентифікація аномалій дозволяє забезпечити більш високий рівень безпеки мережі.

Загалом, розділ підкреслює важливість комплексного підходу до забезпечення безпеки Wi-Fi мереж, що включає як технічні заходи, так і постійний моніторинг та аналіз трафіку. Впровадження описаних методів та інструментів допоможе значно підвищити захищеність бездротових мереж від сучасних кібератак та забезпечити стабільну роботу мережевої інфраструктури.

4 ЗАХОДИ ПРОТИДІЇ ТА РЕКОМЕНДАЦІЇ З ПІДВИЩЕННЯ БЕЗПЕКИ WI-FI

Розділ має на меті дослідити та представити комплексний підхід до зміцнення безпеки бездротових мереж. В умовах зростаючої кіберзагрози та все більш складних атак, захист мережі стає не просто рекомендацією, а обов'язковою вимогою для забезпечення цілісності та конфіденційності корпоративних та приватних даних.

У цьому розділі будуть розглянуті налаштування мережі, які допомагають максимізувати рівень захисту, такі як використання сучасних стандартів шифрування, налаштування безпечних VPN, розгортання мережевих брандмауерів, та імплементація рішень для виявлення та запобігання вторгненням. Особлива увага буде приділена використанню різних програмних та апаратних інструментів, які можуть допомогти у виявленні та нейтралізації потенційних загроз.

Також будуть представлені рекомендації з використання програмного забезпечення та апаратних засобів, які сприяють підвищенню рівня безпеки. Розглянемо найкращі практики для управління Wi-Fi мережами, які включають регулярні оновлення програмного забезпечення, використання складних паролів та ідентифікацію користувачів, що допоможе мінімізувати ризики несанкціонованого доступу.

Цей розділ надає цілісний погляд на стратегії та технології, які є важливими для створення міцного оборонного бар'єру для бездротових мереж, забезпечуючи читачам не тільки теоретичні знання, але й практичні навички для ефективного захисту їхніх мережевих ресурсів.

4.1 Налаштування мережі для максимального захисту

Враховуючи складність сучасних кіберзагроз, ефективне налаштування мережі є вирішальним для захисту конфіденційної інформації та запобігання несанкціонованому доступу.

Основні налаштування мережі, які сприяють максимальному захисту, включають: використання сучасних стандартів шифрування, сегментацію мережі, налаштування брандмауерів та систем виявлення та запобігання вторгненням (IDS/IPS), використання VPN та регулярне оновлення програмного забезпечення.

Слід також зауважити, що більшість сучасних роутерів схильні до зламу спеціалізованими програмами, які використовують вразливості WPS, тому для забезпечення кращого рівня безпеки буде не зайвим прийняття рішення про вимкнення даної функції.

Також гарним методом забезпечення безпеки є налаштування фільтрування підключених пристроїв за MAC-адресою. Такий метод протидії є достатньо ефективним навіть для Open System мереж.

Виконання цих налаштувань та стратегій є фундаментом для забезпечення максимального захисту бездротових мереж, і може істотно знизити ймовірність успішних кібератак, забезпечуючи безпеку та надійність мережевої інфраструктури. Використання цих методик не тільки сприяє захисту даних, але й допомагає підвищити загальну ефективність та продуктивність мережі.

4.1.1 Використання сучасних стандартів шифрування

Сучасні стандарти шифрування, такі як WPA3, пропонують значні переваги для забезпечення безпеки бездротових мереж.

По-перше, WPA3 використовує більш сучасний і безпечний протокол шифрування, що значно ускладнює можливість його злому порівняно зі старішими стандартами, такими як WEP або WPA2. Цей стандарт включає в себе функції, такі як шифрування індивідуальних пакетів даних, що дозволяє забезпечити більш високий рівень захисту для даних користувачів.

По-друге, WPA3 використовує метод шифрування з використанням ключів SAE (Simultaneous Authentication of Equals), що значно зменшує вразливості до атак типу "людина посередині" (Man-In-The-Middle). Ця технологія забезпечує кожному користувачу унікальний ключ, який змінюється з кожним сеансом з'єднання, що робить практично неможливим перехоплення та розшифровку трафіку зловмисниками.

Також, важливим аспектом є підвищена стійкість до спроб атак на основі спроб злому пароля через методи брутфорс (вгадування пароля). WPA3 включає заходи, які затримують спроби підключення після кількох невдалих спроб введення пароля, що ускладнює автоматизовані атаки силовим методом.

Крім того, сучасні стандарти шифрування, такі як WPA3, покращують загальний процес налаштування безпеки. Вони надають можливість використання покращених функцій безпеки без значного збільшення вартості або складності систем, що робить їх доступними для широкого кола користувачів.

4.1.2 Сегментація мережі

Сегментація мережі є ключовим елементом стратегії забезпечення безпеки бездротових мереж і пропонує численні переваги для підвищення загальної безпеки системи. Основною метою сегментації є розділення мережевого трафіку на декілька ізольованих сегментів, кожен з яких містить певний набір ресурсів або служб. Це зменшує ризики, пов’язані зі широкомасштабними атаками всередині мережі, оскільки атака на один сегмент не обов’язково вплине на інші сегменти.

Перша і, можливо, найважливіша перевага сегментації полягає у підвищенні безпеки шляхом обмеження доступу до критично важливих даних та ресурсів. Наприклад, конфіденційні дані можуть бути ізольовані в одному сегменті, в той час як гостьовий доступ до Інтернету надається через інший. Це гарантує, що навіть якщо зловмисники здобудуть доступ до гостьового сегменту, їм буде значно важче дістатися до захищених даних.

Друга перевага сегментації — зменшення загального рівня шуму в мережі та покращення продуктивності. Кожен сегмент має власний обмежений обсяг трафіку, що зменшує затори і покращує швидкість передачі даних всередині кожного сегменту.

Третя перевага включає підвищену гнучкість у впровадженні політик безпеки. Сегментація дозволяє адміністраторам мережі легше налаштовувати і контролювати безпеку, адаптуючи політики до конкретних потреб кожного сегменту. Наприклад, вимоги до безпеки можуть бути суворішими в сегментах, що обробляють чутливі дані, порівняно з менш критичними сегментами.

Загалом, сегментація мережі дозволяє досягнути більшої модульності, безпеки та ефективності управління трафіком, що є вирішальним для забезпечення надійної захищеності сучасних бездротових мереж.

4.1.3 Налаштування брандмауерів та систем виявлення та запобігання вторгненням (IDS/IPS)

Налаштування брандмауерів та систем виявлення та запобігання вторгненням (IDS/IPS) є життєво важливими компонентами захисту бездротових мереж. Брандмауери діють як бар'єр між зовнішнім світом та внутрішніми ресурсами мережі, обмежуючи трафік, який може ввійти або вийти з мережі на основі заздалегідь встановлених правил безпеки. Це дозволяє запобігти несанкціонованому доступу та атакам, забезпечуючи, що лише легітимний трафік може пересуватися через мережу.

Системи виявлення та запобігання вторгненням (IDS/IPS) додатково збільшують рівень безпеки шляхом моніторингу мережевого трафіку на предмет виявлення ознак шкідливої діяльності або відомих атак. IDS функціонує шляхом пасивного спостереження та реєстрації можливих загроз, а IPS активно втручається, блокуючи потенційно шкідливий трафік та здійснюючи дії для запобігання атакам у реальному часі.

Переваги таких систем полягають у можливості забезпечення комплексної захищеності мережі. Завдяки брандмауерам можна контролювати вхідний і вихідний трафік, запобігаючи спробам несанкціонованого доступу. IDS/IPS підвищує цей рівень захисту, виявляючи та нейтралізуючи атаки, перш ніж вони зможуть завдати значної шкоди, що особливо важливо у випадку нульових днів та інших складних кіберзагроз.

Також, використання IDS/IPS може допомогти виявити і зупинити внутрішні загрози, включаючи малварі та інші види зловмисного ПЗ, які можуть бути розповсюджені в мережі. Це забезпечує додатковий рівень безпеки, запобігаючи поширенню шкідливих програм всередині мережі.

Впровадження таких технологій надзвичайно важливе, оскільки воно мінімізуючи перебої в роботі, пов'язані з кібератаками.

4.1.4 Використання VPN

Використання VPN (віртуальних приватних мереж) є однією з найефективніших стратегій для забезпечення безпеки бездротових мереж, особливо у публічних місцях, де безпека Wi-Fi може бути під загрозою. VPN дозволяє користувачам створювати безпечний тунель між їхнім пристроєм та віддаленим сервером VPN, через який проходить весь інтернет-трафік. Це шифрування даних усередині тунелю запобігає можливості перехоплення та аналізу даних зловмисниками, що є особливо важливим у відкритих або слабозахищених мережах.

Перша і основна перевага використання VPN полягає в підвищенні конфіденційності. Всі дані, що передаються через VPN-тунель, зашифровані, що робить майже неможливим їх перехоплення третіми особами. Це означає, що навіть якщо зловмисник зможе втрутитися в мережу, він не зможе розшифрувати вміст даних, які передаються.

Друга перевага полягає в збереженні анонімності. Завдяки VPN, IP-адреса користувача замінюється на IP-адресу VPN-сервера, що ускладнює відстеження онлайн-активності до конкретної особи або пристрою. Це особливо корисно для захисту персональної інформації в мережах, де існує високий ризик спостереження або моніторингу.

Третя перевага використання VPN — доступ до заблокованих ресурсів. VPN може обходити географічні обмеження і цензуру, надаючи доступ до інтернет-ресурсів, які можуть бути обмежені в певних країнах або регіонах. Це робить VPN незамінним інструментом для забезпечення вільного доступу до інформації.

Загалом, використання VPN значно підвищує безпеку, конфіденційність та анонімність в інтернеті, що є дуже важливим фактором.

4.1.5 Оновлення ПЗ

Регулярне оновлення програмного забезпечення є важливою складовою ефективної стратегії безпеки для будь-якої бездротової мережі. Оновлення забезпечують захист від відомих вразливостей, які хакери можуть використовувати для атак. Зокрема, виробники мережевого обладнання та програмного забезпечення регулярно випускають патчі та оновлення, які виправляють помилки та зміцнюють захист від нових загроз. Це означає, що швидке впровадження цих оновлень дозволяє замкнути потенційні "дірки" в безпеці до того, як їх встигнуть використати зловмисники.

Одна з основних переваг регулярного оновлення полягає у захисті від розповсюджених кібератак, таких як віруси, трояни та інші види шкідливого програмного забезпечення, які часто розробляються для використання вразливостей у застарілому програмному забезпеченні. Завдяки вчасному оновленню програмного забезпечення можна ефективно протистояти цим загрозам, забезпечуючи безпеку даних та приватності користувачів.

Крім того, регулярні оновлення допомагають підвищити стабільність системи та знизити ризики непередбачуваних збоїв, що можуть призвести до втрати даних або зниження продуктивності мережі. Часто оновлення включають в себе покращення функціональності та оптимізацію процесів, що підвищує загальну ефективність роботи мережевого обладнання.

Загалом, регулярне оновлення програмного забезпечення забезпечує критично важливий захист від кіберзагроз, підвищує надійність і продуктивність мережевих систем, та сприяє забезпеченню безперебійної роботи і довіри користувачів. Це вкрай важливо для будь-якої організації, яка прагне забезпечити надійний захист своїх цифрових ресурсів.

4.2 Рекомендації з використання програмного забезпечення та апаратних засобів

У сучасному цифровому середовищі важливо не тільки впроваджувати передові практики безпеки, але й ефективно використовувати доступні програмні та апаратні ресурси для забезпечення комплексного захисту.

Використання антивірусного програмного забезпечення, застосування рішень для шифрування та програмного забезпечення для керування доступом є фундаментальними складовими комплексної стратегії безпеки для будь-якої Wi-Fi мережі. Ці інструменти разом забезпечують багаторівневий захист від різних типів загроз, знижуючи потенційний ризик втручань і зловживань.

Антивірусне програмне забезпечення відіграє критичну роль у захисті пристроїв, підключених до мережі, шляхом виявлення, блокування та видалення шкідливих програм. Це допомагає уникнути інфекцій, які можуть спричинити витік даних, пошкодження системи або непомітне шпигунство. Актуалізація антивірусних баз даних забезпечує виявлення та нейтралізацію найновіших загроз, гарантуючи високий рівень безпеки.

Застосування рішень для шифрування є ще одним важливим заходом безпеки. Шифрування даних, що передаються і зберігаються, забезпечує їхню конфіденційність і цілісність. Навіть у випадку перехоплення даних зловмисниками, зашифрована інформація залишається недоступною без відповідного ключа шифрування. Це особливо важливо для конфіденційної корпоративної інформації або особистих даних користувачів.

Програмне забезпечення для керування доступом дозволяє адміністраторам мережі контролювати хто, коли та з яких пристроїв має доступ до мережевих ресурсів.

Це включає можливості аутентифікації, авторизації та аудиту, дозволяючи забезпечити дотримання політик безпеки і запобігти несанкціонованому доступу. Використання політик мінімальних привілеїв та регулярна перевірка прав доступу є важливими для запобігання зловживань та витоку інформації.

Апаратні рекомендації включають застосування апаратних брандмауерів, використання безпечних маршрутизаторів та комутаторів і, звісно ж, регулярне оновлення мережевого обладнання.

Апаратні брандмауери виступають як перша лінія оборони мережі, контролюючи вхідний та вихідний трафік на основі заздалегідь встановлених правил безпеки. Вони забезпечують більш надійний рівень захисту порівняно з програмними брандмауерами, оскільки є спеціалізованими пристроями, оптимізованими для цієї задачі, та менш схильні до впливу шкідливих програм, що можуть вражати хост-системи.

Безпечні маршрутизатори та комутатори оснащені додатковими функціями безпеки, такими як шифрування трафіку, підтримка віртуальних приватних мереж (VPN), а також засоби аутентифікації та авторизації користувачів, які важливі для захисту мережевих ресурсів. Використання такого обладнання дозволяє знизити ризики несанкціонованого доступу та атак, спрямованих на мережеву інфраструктуру.

Регулярне оновлення мережевого обладнання є критично важливим для підтримання безпеки. Виробники обладнання постійно виявляють нові вразливості та випускають оновлення, що усувають ці проблеми. Швидке впровадження цих оновлень забезпечує захист від відомих загроз та скорочує часовий проміжок, протягом якого система може бути вразливою до атак.

Крім того, оновлення часто містять поліпшення функціональності та продуктивності, що також сприяє кращій експлуатації мережевих ресурсів.

Як результат, рекомендації, викладені у цьому розділі, мають на меті надати користувачам практичні поради щодо вибору та використання програмного забезпечення та апаратних засобів, які максимально підвищать безпеку їхніх бездротових мереж. Впровадження цих рекомендацій допоможе створити міцний оборонний бар'єр проти кіберзагроз та забезпечити надійний захист від можливих атак.

4.3 Політики безпеки та кращі практики управління Wi-Fi мережами

В контексті зростаючих загроз у кіберпросторі, правильне управління та впровадження чітких політик безпеки є вирішальним для забезпечення стабільності та захищеності мережевої інфраструктури.

Розробка чітких політик безпеки та плану реагування на інциденти є критично важливими аспектами забезпечення безпеки Wi-Fi мереж. Політики безпеки служать основою для встановлення правил доступу до мережі, визначення процедур аутентифікації та обмежень щодо використання мережевих ресурсів. Вони повинні бути чітко документовані, регулярно оновлюватися і постійно доноситись до відома всіх користувачів мережі, забезпечуючи, знання своїх обов'язків та відповідальності у рамках мережі.

Це допомагає створити контрольоване та захищене середовище, де всі дії користувачів відслідковуються та регулюються згідно з установленими нормами.

З іншого боку, розробка плану реагування на інциденти є необхідною для того, щоб мережа могла ефективно відновитися після порушень безпеки. Чіткий план дій у випадку виявлення інцидентів дозволяє швидко ідентифікувати, оцінити та зреагувати на загрози, мінімізуючи потенційні збитки.

Такий план має включати процедури повідомлення, кроки для ізоляції враженої частини мережі, методи відновлення послуг та стратегії для виправлення та запобігання майбутніх інцидентів. Наявність добре розробленого плану реагування також сприяє підтримці довіри користувачів і партнерів, підкреслюючи відповідальний підхід до управління ризиками та безпекою.

Загалом, ретельно розроблені політики безпеки разом з ефективним планом реагування на інциденти забезпечують міцну основу для захисту Wi-Fi мереж. Вони не лише допомагають уникнути порушень безпеки, але й забезпечують готовність до дій у критичних ситуаціях, знижуючи можливі негативні наслідки для мережі та її користувачів.

Навчання та освіта користувачів відіграє не менш важливу роль у забезпеченні безпеки Wi-Fi мереж. Організація регулярних тренінгів та інформаційних сесій для користувачів дозволяє значно знизити ризики, пов'язані з необережним використанням мережевих ресурсів, та підвищити обізнаність щодо актуальних загроз. Завдяки такому підходу, користувачі стають більш обізнаними в питаннях кібербезпеки, що дозволяє їм виявляти потенційно шкідливі дії та запобігати можливим атакам.

Однією з основних переваг навчання користувачів є зниження людського фактору як причини інцидентів безпеки. Багато кібератак, таких як фішинг або атаки соціальної інженерії, спрямовані на маніпуляцію саме користувачами. Інформованість користувачів про такі методи і знання, як на них реагувати, можуть значно зменшити їх ефективність.

Крім того, регулярні тренінги допомагають популяризувати культуру безпеки на високому рівні, оновлюючи знання користувачів згідно з новими технологіями та загрозами, що постійно змінюються. Це особливо важливо у світі, де методи кібератак постійно еволюціонують.

Основна перевага полягає в тому, що освічені користувачі можуть служити додатковим рівнем захисту, оскільки вони стають здатними не тільки виявляти потенційні загрози, а й активно допомагають у впровадженні політик безпеки та їх дотриманні. Це створює більш безпечне та стійке середовище, де кожен користувач відіграє роль у захисті інформаційних ресурсів.

Останнім критично важливим аспектом розділу є моніторинг та аудит мережі.

Впровадження систем моніторингу та регулярний аудит мережі дозволяють не тільки виявляти, але й оперативно реагувати на несанкціоновані дії. Аналіз логів, моніторинг активності користувачів та перевірка конфігурацій обладнання допомагають виявити потенційні зловживання або аномалії в мережі, що можуть вказувати на кібератаки або технічні неполадки.

Однією з ключових переваг моніторингу мережі є можливість виявлення загроз на ранній стадії, що забезпечує більше часу для реагування та мінімізації потенційних шкідливих наслідків. Регулярний аудит мережі допомагає переконатися, що всі заходи безпеки діють ефективно і що мережеве обладнання не має критичних вразливостей, які можуть бути використані хакерами.

Крім того, систематичний аналіз логів та моніторинг активності користувачів сприяють виявленню внутрішніх загроз, таких як несанкціонований доступ або використання мережевих ресурсів не за призначенням. Це дозволяє адміністраторам мережі вживати відповідних заходів для запобігання зловживань та підтримання високого рівня продуктивності мережі.

Загалом цей розділ підкреслює важливість комплексного підходу до управління безпекою Wi-Fi мереж. Впровадження детально розроблених політик, постійне навчання користувачів, регулярний моніторинг мережі та використання передових технологічних рішень може істотно підвищити рівень безпеки та забезпечити надійний захист від потенційних загроз.

Висновки до розділу 4

Розділ підкреслює важливість комплексного підходу у забезпеченні захисту бездротових мереж. В ньому ми розглянули необхідність адекватних налаштувань мережі, використання передового програмного та апаратного забезпечення, впровадження чітких політик безпеки, а також регулярний моніторинг і аудит. Важливо, що кожен аспект, від налаштувань шифрування до освітніх програм для користувачів, відіграє ключову роль у формуванні надійного захисту мережі.

Налаштування мережі, включно з використанням надійних стандартів шифрування та сегментацією мережі, забезпечують міцний фундамент для безпеки. Програмне забезпечення, яке постійно оновлюється, та апаратне забезпечення, що відповідає сучасним вимогам, допомагають уникнути багатьох загроз. Політики безпеки, що чітко визначають відповідальності та процедури, є критично важливими для підтримання порядку і дисципліни серед користувачів мережі. Регулярний моніторинг та аудит дозволяють не тільки виявляти потенційні вразливості, а й швидко реагувати на інциденти, забезпечуючи оперативне вирішення проблем.

Загалом, успішна стратегія безпеки Wi-Fi вимагає інтеграції технічних рішень із стратегічним плануванням та виконанням.

5 РОЗРОБКА ПРОГРАМНОГО КОДУ МОВОЮ PYTHON ДЛЯ СВОЄЧАСНОГО ВИЯВЛЕННЯ МОЖЛИВИХ АТАК НА WI-FI МЕРЕЖУ

В попередньому розділі з рекомендаціями підкреслено важливість постійного моніторингу мережі для забезпечення комплексного підходу до її захисту.

Цей пункт поставив перед нами серйозний виклик, тому що навіть через невелику Wi-Fi мережу з декількома активними користувачами, проходить надзвичайно велика кількість пакетів за малий проміжок часу.

Помітити підозрілу активність, а тим паче проаналізувати потрібні пакети та помітити особливості, притаманні розглянутим у третьому розділі атакам, стає дуже складною, а інколи й нереальною задачею.

Для підвищення безпеки мережі було розроблено та реалізовано програмний код для аналізу мережевого трафіку з використанням Python та бібліотеки Pyshark.

Метою коду є виявлення різноманітних типів мережевих атак та підозрілої активності, включаючи ARP спуфінг, DNS спуфінг, SSH брутфорс атаки, небезпечні SSH команди, DoS/DDoS атаки, а також різноманітні фішингові спроби.

Бібліотека Pyshark використовується для читання та аналізу pcap файлів, що дозволяє обробляти мережеві пакети та застосовувати різні алгоритми перевірки на зловмисну активність.

Код містить ряд функцій для перевірки специфічних видів атак та іншої підозрілої активності. Також система використовує модуль logging для реєстрації подій, що дозволяє зберігати записи про підозрілу активність та забезпечує можливість аналізу подій після їх виявлення.

Логування ініціюється на самому початку, де налаштовується збереження інформації про виявлені події у файл. Це включає час, тип події та деталізоване повідомлення. Функція log_detection призначена для введення записів в цей журнал, яка активується кожного разу, коли інші функції виявляють щось незвичайне.

Функція check_arp_spoofing виявляє зміни у відповідностях IP-MAC адрес, що є характерним для атак типу ARP spoofing, коли атакувач відправляє підроблені ARP повідомлення в мережу, спрямовуючи трафік на свій пристрій. Код аналізує ARP пакети, перевіряючи зміни в таблиці ARP та виявляючи незапрошені ARP відповіді.

Check_dns_spoofing оцінює швидкість відповідей DNS і відповідність попередніх відповідей для одного запиту. Швидка відповідь чи розбіжності у відповідях можуть вказувати на DNS spoofing, де атакувач відправляє швидку підроблену відповідь, перенаправляючи користувача на шкідливий сайт.

Check_packet_delays фокусується на виявленні затримок між пакетами від однієї IP адреси, що може бути індикатором атак типу Man in the Middle, де атакувач може активно перехоплювати і модифікувати трафік.

Check_unencrypted_traffic і check_ssl_certificates звертають увагу на нешифрований трафік і використання ненадійних SSL/TLS сертифікатів відповідно. Ці аспекти критичні для забезпечення конфіденційності та цілісності даних у мережі.

Check_hsts_headers перевіряє, чи сервери, що використовують HTTP, належним чином імплементують політики безпеки, зокрема HSTS, що змушує браузери користувачів використовувати безпечні HTTPS з'єднання.

Функції check_ssh_bruteforce і check_ssh_commands зосереджені на виявленні атак на SSH сервіси, включаючи брутфорс атаки (послідовні спроби входу) і виконання потенційно шкідливих команд через SSH.

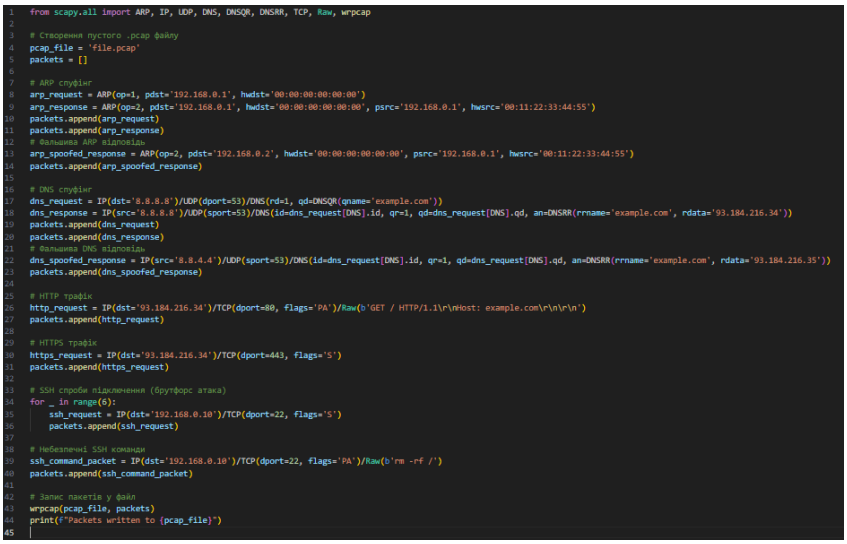
Update_traffic_stats та check_traffic_anomalies відстежують загальний обсяг трафіку для виявлення раптових змін у активності, що можуть вказувати на DoS атаки або інші масштабні спроби порушення нормальної роботи мережі.

Check_dos_attacks та check_phishing_attacks використовують часові мітки та аналіз URL для виявлення атак, спрямованих на перевантаження ресурсів або обман користувачів через фішингові сайти.

Після завершення аналізу всіх пакетів код виконує загальну перевірку аномалій, що виявляються за допомогою порогових значень, та завершує аналіз, закриваючи файл pcap. Така ретельна і багатоаспектна перевірка дозволяє ідентифікувати і логувати широкий спектр мережевих загроз, забезпечуючи високий рівень захисту від потенційних атак.

Цей код ілюструє інтегрований підхід до моніторингу та безпеки мережі, поєднуючи різні техніки та методи для виявлення широкого спектру атак. Кожна функція забезпечує специфічний тип моніторингу і сприяє глибокому аналізу мережевого трафіку, що робить систему гнучкою та ефективною у виявленні зловмисних дій у мережі.

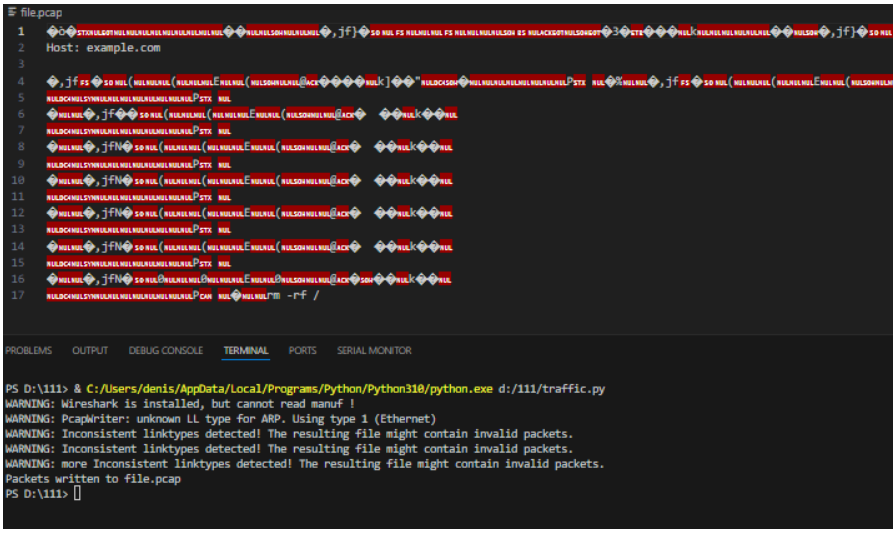
Для демонстрації працездатності програми, з етичних міркувань було вирішено написати ще один Python-скрипт, який застосовує бібліотеку Scapy для створення різноманітних мережевих пакетів, які імітують загальні мережеві атаки, такі як ARP спуфінг, DNS спуфінг, HTTP та HTTPS трафік, а також брутфорс атаки SSH.



```python
from scapy.all import ARP, IP, UDP, DNS, DNSQR, DNSRR, TCP, Raw, wrpcap

# Створення пустого .pcap файлу
pcap_file = 'file.pcap'
packets = []

# ARP спуфінг
arp_request = ARP(op=1, pdst='192.168.0.1', hwdst='00:00:00:00:00:00')
arp_response = ARP(op=2, pdst='192.168.0.1', hwdst='00:00:00:00:00:00', psrc='192.168.0.1', hwsrc='00:11:22:33:44:55')
packets.append(arp_request)
packets.append(arp_response)
# Фальшива ARP відповідь
arp_spoofed_response = ARP(op=2, pdst='192.168.0.2', hwdst='00:00:00:00:00:00', psrc='192.168.0.1', hwsrc='00:11:22:33:44:55')
packets.append(arp_spoofed_response)

# DNS спуфінг
dns_request = IP(dst='8.8.8.8')/UDP(dport=53)/DNS(rd=1, qd=DNSQR(qname='example.com'))
dns_response = IP(src='8.8.8.8')/UDP(sport=53)/DNS(id=dns_request[DNS].id, qr=1, qd=dns_request[DNS].qd, an=DNSRR(rrname='example.com', rdata='93.184.216.34'))
packets.append(dns_request)
packets.append(dns_response)
# Фальшива DNS відповідь
dns_spoofed_response = IP(src='8.8.4.4')/UDP(sport=53)/DNS(id=dns_request[DNS].id, qr=1, qd=dns_request[DNS].qd, an=DNSRR(rrname='example.com', rdata='93.184.216.35'))
packets.append(dns_spoofed_response)

# HTTP трафік
http_request = IP(dst='93.184.216.34')/TCP(dport=80, flags='PA')/Raw(b'GET / HTTP/1.1\r\nHost: example.com\r\n\r\n')
packets.append(http_request)

# HTTPS трафік
https_request = IP(dst='93.184.216.34')/TCP(dport=443, flags='S')
packets.append(https_request)

# SSH спроби підключення (брутфорс атака)
for _ in range(6):
    ssh_request = IP(dst='192.168.0.10')/TCP(dport=22, flags='S')
    packets.append(ssh_request)

# Небезпечні SSH команди
ssh_command_packet = IP(dst='192.168.0.10')/TCP(dport=22, flags='PA')/Raw(b'rm -rf /')
packets.append(ssh_command_packet)

# Запис пакетів у файл
wrpcap(pcap_file, packets)
print(f"Packets written to {pcap_file}")
```

Рисунок 5.1 - Python-скрипт, який застосовує бібліотеку Scapy для створення різноманітних мережевих пакетів

Результатом виконання програми є файл формату .pcap, в якому зімітовано дамп пакетів, які можуть бути виявлені програмою Wireshark при проведенні зловмисником реальних атак.



file.pcap

```
Host: example.com
rm -rf /
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SERIAL MONITOR

```
PS D:\111> & C:/Users/denis/AppData/Local/Programs/Python/Python310/python.exe d:/111/traffic.py
WARNING: Wireshark is installed, but cannot read manuf !
WARNING: PcapWriter: unknown LL type for ARP. Using type 1 (Ethernet)
WARNING: Inconsistent linktypes detected! The resulting file might contain invalid packets.
WARNING: Inconsistent linktypes detected! The resulting file might contain invalid packets.
WARNING: more Inconsistent linktypes detected! The resulting file might contain invalid packets.
Packets written to file.pcap
PS D:\111>
```

Рисунок 5.2 – Результат виконання скрипта, який генерує дамп підозрілого мережевого трафіку

Після створення файлу з підозрілим трафіком можемо запускати наш скрипт-детектор та отримувати результат.

Рисунок 5.3 – Результат обробки дампу пакетів програмним кодом мовою Python для своєчасного виявлення можливих атак

ВИСНОВКИ

Ця робота не лише зміцнює теоретичні основи захисту Wi-Fi мереж, а й вносить практичний вклад у вдосконалення методів аналізу та захисту бездротових мереж.

Використання сучасних аналітичних інструментів та розробка власного програмного забезпечення дозволили виявляти потенційні атаки на ранніх стадіях, що є ключовим аспектом у забезпеченні вищого рівня безпеки в умовах постійної загрози кібератак. Значна увага в дипломній роботі приділялась не тільки технічним аспектам, а й практичному застосуванню розроблених рішень, що робить дослідження релевантним та актуальним.

В ході дослідження було визначено, що багато сучасних атак на Wi-Fi мережі можна ефективно нейтралізувати за допомогою комплексного підходу, що включає застосування сучасних стандартів шифрування, систем автентифікації, а також періодичне оновлення програмного забезпечення.

Розроблені методики і рекомендації можуть бути використані для об'єктивної оцінки вразливостей мережі та формування адекватних заходів щодо їх усунення.

Додаткову цінність роботи представляє порівняльний аналіз інструментів для моніторингу та аналізу мереж, який включав детальні огляди Wireshark, tcpdump та CommView.

Це дало змогу обрати оптимальний інструмент для реалізації захисту Wi-Fi мереж, що є критично важливим для організацій, що залежать від стабільності та безпеки своїх інформаційних систем.

В результаті дослідження було розроблено рекомендації щодо підвищення захищеності Wi-Fi мереж, що можуть бути застосовані в домашніх умовах, корпоративних мережах та громадських місцях.

Практичне впровадження розроблених методик та рекомендацій дозволить зміцнити захист проти сучасних кібератак, забезпечуючи цілісність і конфіденційність важливих даних.

Висновки з цього дослідження відіграють важливу роль у формуванні наукової та технічної бази для подальших розробок у галузі кібербезпеки.

Вони вказують на необхідність подальшого вивчення сучасних технологій захисту даних та розробки нових, більш ефективних методів виявлення та запобігання атакам.

Це дослідження ставить певний маркер у вивченні захисту бездротових мереж, надаючи стимул для наукової спільноти продовжувати роботу в цьому напрямку, розробляючи інноваційні рішення, які зможуть забезпечити надійність та безпеку потоків інформації у всесвітньому масштабі.

Таким чином, дипломна робота не просто розв'язує актуальну проблему захисту Wi-Fi мереж, а й вносить істотний вклад у формування культури безпеки серед користувачів, підвищуючи їх компетентність та відповідальність у використанні інформаційних технологій.

ПЕРЕЛІК ДЖЕРЕЛ ТА ПОСИЛАНЬ

1. Behboodian N. ARP Poisoning Attack: An introduction to attack and mitigations Paperback. — 2012. — Jan. — URL: https://www.amazon.com/ARP-Poisoning-Attack-introduction-mitigations/dp/1468068512.

2. Combs G. Wireshark: Unveiling Network Intricacies. — 1998. — Jan. — URL: https://www.wireshark.org/docs/.

3. Information Security, Privacy and Digital Forensics /S. Patel, N. K. Chaudhary, B. Gohil, S. Iyengar. — 2024. — Jan. — URL: https://link.springer.com/book/10.1007/978-981-99-5091-1.

4. Sonowal G. Phishing and Communication Channels. A Guide to Identifying and Mitigating Phishing Attacks. — 2022. — Jan. — URL: https://link.springer.com/book/10.1007/978-1-4842-7744-7.

5. Rangeforce Materials: Introduction to tcpdump – 2020. – Jun. – URL: https://materials.rangeforce.com/tutorial/2020/06/14/Introduction-to-Tcpdump.

6. Tamos foundation: Commview documentation – 2022. – Dec. – URL: https://www.tamos.com/htmlhelp/commview/index.htm.

7. Wallarm: KRACK Attack – 2024. – Feb. – URL: https://www.wallarm.com/what/what-is-krack-or-key-reinstallation-attack.

8. Spiceworks: MiTM Attack – 2022. – Apr. – URL: https://www.spiceworks.com/it-security/data-security/articles/man-in-the-middle-attack.

9. Imperva learning materials: DNS Spoofing Attack – 2024. – Apr. – URL: https://www.imperva.com/learn/application-security/dns-spoofing.

10. OKTA Foundation: ARP Spoofing Attack – 2022. – Feb. – URL: https://www.okta.com/au/identity-101/arp-poisoning.

11. Stealing the Network by Ryan Russell – 2003. – Dec. – URL: https://www.sciencedirect.com/book/9781931836876/stealing-the-network-how-to-own-the-box.

12. Cryptography and Network Security: Principles and Practice by William Stallings – 2014. – Jan. – URL: https://dl.acm.org/doi/10.5555/2523199.

13. TCP/IP Illustrated, Vol. 1: The Protocols by Richard Stevens – 2009. – May. – URL: https://www.r-5.org/files/books/computers/internals/net/Richard_Stevens-TCP-IP_Illustrated-EN.pdf

14. Applied Network Security Monitoring by Chris Sanders – 2013. – Jun. – URL: https://chrissanders.org/appliednsm.

15. Network Security Essentials: Applications and Standards by William Stallings – 2014. – May. – URL: https://elhacker.info/manuales/Redes/3._Network-security-essentials-4th-edition-william-stallings.pdf

16. CompTIA Security+ Guide to Network Security Fundamentals by Mark Chiampa – 2017. – Aug. – URL: https://www.academia.edu/41663793/Security_Guide_to_Network_Security_Fundamentals_Fifth_Edition.

17. Lecture materials on the subject Digital circuit technology by Volodymyr Stepanenko – 2024. – Feb. – URL: https://www.youtube.com/playlist?list=PLb5IW_0N-8ZWjZhq3S7FFbmHlhgElldz5.

ДОДАТОК А
ПРОГРАМНИЙ КОД МОВОЮ PYTHON ДЛЯ СВОЄЧАСНОГО ВИЯВЛЕННЯ МОЖЛИВИХ АТАК НА WI-FI МЕРЕЖУ

```
import pyshark
import time
import logging
from collections import defaultdict
import datetime
import re

# Налаштування логування
logging.basicConfig(filename='network_analysis.log', level=logging.INFO, format='%(asctime)s - %(levelname)s - %(message)s')

def log_detection(message):
    """Функція для запису повідомлень у лог."""
    logging.info(message)

# Шлях до файла pcap
pcap_file = 'file.pcap'

# Використання pyshark для аналізу pcap файлу
capture = pyshark.FileCapture(pcap_file, keep_packets=False)

# Словники для моніторингу
arp_table = {}  # Таблиця ARP для відстеження IP-MAC відповідностей
arp_activity = {}  # Діяльність ARP для відстеження множинних IP для одного MAC
```

```
packet_delays = {}  # Відстеження затримок пакетів для кожної IP-адреси
ssh_attempts = {}  # Відстеження спроб SSH зламу
dns_queries = {}  # Відстеження DNS запитів і відповідей
dangerous_ssh_commands = ['rm', 'shutdown', 'reboot', 'kill']  # Підозрілі SSH команди
packet_count_per_minute = defaultdict(int)  # Кількість пакетів за хвилину
packet_threshold = 100  # Порогове значення для визначення аномальної активності
dos_activity = defaultdict(list)  # Діяльність DoS для кожної IP-адреси
dos_threshold = 50  # Порогове значення для DoS/DDoS атак за 10 секунд

# Список підозрілих доменів або шаблонів URL для детекції фішингових сайтів
phishing_patterns = [
    re.compile(r'.*login.*', re.IGNORECASE),
    re.compile(r'.*bank.*', re.IGNORECASE),
    re.compile(r'.*update.*', re.IGNORECASE),
    re.compile(r'.*verify.*', re.IGNORECASE),
    re.compile(r'.*secure.*', re.IGNORECASE)
]

# Функції для перевірки атак

def check_arp_spoofing(packet):
    """Перевірка на ARP спуфінг.

    Виявляє зміни у відповідностях IP-MAC, незапрошені ARP відповіді та
    асоціації множинних IP з одним MAC.
    """
```

```
    if 'ARP' in packet:
        src_ip = packet.arp.src_proto_ipv4
        src_mac = packet.arp.src_hw_mac
        if src_ip in arp_table and arp_table[src_ip] != src_mac:
            log_detection(f"ARP spoofing detected: IP {src_ip} has changed from {arp_table[src_ip]} to {src_mac}")
        if packet.arp.opcode == '2' and packet.arp.dst_proto_ipv4 == '0.0.0.0':
            log_detection(f"Gratuitous ARP detected from {src_ip} ({src_mac})")
        if src_mac not in arp_activity:
            arp_activity[src_mac] = set()
        arp_activity[src_mac].add(src_ip)
        if len(arp_activity[src_mac]) > 1:
            log_detection(f"Multiple IPs {arp_activity[src_mac]} associated with MAC {src_mac}")
        arp_table[src_ip] = src_mac


def check_dns_spoofing(packet):
    """Перевірка на DNS спуфінг.

    Виявляє дуже швидкі відповіді та непослідовні DNS відповіді.
    """
    if 'DNS' in packet:
        dns_query_name = packet.dns.qry_name
        response_time = packet.sniff_timestamp
        if dns_query_name:
            if dns_query_name in dns_queries:
                previous_response = dns_queries[dns_query_name]
                if response_time - previous_response['time'] < 0.01:
```

```
                log_detection(f"Possible DNS spoofing: very quick response for
{dns_query_name}")
            if hasattr(packet.dns, 'a'):
                if packet.dns.a != previous_response['response']:
                    log_detection(f"DNS response inconsistency detected for
{dns_query_name}: {packet.dns.a} vs {previous_response['response']}")
        if hasattr(packet.dns, 'a'):
            dns_queries[dns_query_name] = {'time': float(response_time), 'response':
packet.dns.a}


def check_packet_delays(packet):
    """Перевірка на затримки пакетів.

    Виявляє значні затримки між пакетами для визначення можливих MiTM
атак.
    """
    if hasattr(packet, 'ip'):
        src_ip = packet.ip.src
        if src_ip in packet_delays:
            current_time = time.time()
            delay = current_time - packet_delays[src_ip]
            if delay > 1:
                log_detection(f"Possible MitM attack: High delay from {src_ip}
detected")
        packet_delays[src_ip] = time.time()


def check_unencrypted_traffic(packet):
    """Перевірка на нешифрований HTTP трафік.

    Виявляє нешифрований HTTP трафік, що може бути вразливим до атак.
```

```
    """
    if 'http' in packet:
        log_detection(f"Unencrypted HTTP traffic detected from {packet.ip.src} to {packet.ip.dst}")

def check_ssl_certificates(packet):
    """Аналіз SSL/TLS сертифікатів.

    Додана перевірка на використання ненадійних або самопідписаних сертифікатів.
    """
    if 'SSL' in packet or 'TLS' in packet:
        if hasattr(packet.ssl, 'handshake_type'):
            if packet.ssl.handshake_type == '1':  # ClientHello
                if hasattr(packet.ssl, 'handshake_extensions_server_name'):
                    server_name = packet.ssl.handshake_extensions_server_name
                    log_detection(f"SSL/TLS request to {server_name}")
            if packet.ssl.handshake_type == '2':  # ServerHello
                if hasattr(packet.ssl, 'handshake_extensions_server_name'):
                    server_name = packet.ssl.handshake_extensions_server_name
                    if hasattr(packet.ssl, 'handshake_certificates'):
                        certificates = packet.ssl.handshake_certificates
                        if 'self-signed' in certificates or 'expired' in certificates:
                            log_detection(f"Invalid SSL/TLS certificate detected for {server_name}: {certificates}")

def check_hsts_headers(packet):
    """Перевірка на наявність HSTS заголовків.
```

```
    Виявляє наявність HSTS заголовків у HTTP відповідях.
    """
    if 'HTTP' in packet:
        if hasattr(packet.http, 'response_for_uri'):
            if 'strict-transport-security' in packet.http.field_names:
                log_detection(f"HSTS header present in response for
{packet.http.response_for_uri}")


def check_ssh_bruteforce(packet):
    """Перевірка на SSH брутфорс атаки.

    Виявляє підозрілу активність SSH, таку як численні спроби підключення.
    """
    if 'SSH' in packet:
        if 'ssh' in packet and 'protocol' in packet.ssh.field_names:
            if packet.ssh.protocol == "SSHv2":
                src_ip = packet.ip.src
                if src_ip not in ssh_attempts:
                    ssh_attempts[src_ip] = 1
                else:
                    ssh_attempts[src_ip] += 1
                if ssh_attempts[src_ip] > 5:
                    log_detection(f"Possible SSH brute force attack from {src_ip}")


def check_ssh_commands(packet):
    """Перевірка на небезпечні SSH команди.

    Виявляє підозрілі SSH команди, такі як 'rm', 'shutdown', 'reboot' та 'kill'.
    """
```

```
    if 'SSH' in packet and 'data' in packet.ssh.field_names:
        for command in dangerous_ssh_commands:
            if command in packet.ssh.data:
                log_detection(f"Dangerous SSH command detected from {packet.ip.src}: {command}")

# Функція для оновлення статистики трафіку
def update_traffic_stats(packet):
    """Оновлення статистики трафіку.

    Відстежує кількість пакетів за хвилину для виявлення аномальної активності.
    """
    timestamp = datetime.datetime.fromtimestamp(float(packet.sniff_timestamp))
    minute_key = timestamp.strftime("%Y-%m-%d %H:%M")
    packet_count_per_minute[minute_key] += 1

# Функція для перевірки аномальної активності
def check_traffic_anomalies():
    """Перевірка на аномальну активність у трафіку.

    Виявляє аномальну активність на основі порогового значення кількості пакетів за хвилину.
    """
    for minute, count in packet_count_per_minute.items():
        if count > packet_threshold:
            log_detection(f"Traffic anomaly detected: {count} packets in {minute}")

# Функція для перевірки DoS/DDoS атак
```

```
def check_dos_attacks(packet):
    """Перевірка на DoS/DDoS атаки.

    Виявляє підозрілу активність на основі великої кількості запитів з однієї IP-адреси за короткий проміжок часу.
    """
    if hasattr(packet, 'ip'):
        src_ip = packet.ip.src
        current_time = time.time()
        dos_activity[src_ip].append(current_time)
        dos_activity[src_ip] = [timestamp for timestamp in dos_activity[src_ip] if current_time - timestamp < 10]
        if len(dos_activity[src_ip]) > dos_threshold:
            log_detection(f"Possible DoS/DDoS attack detected from {src_ip}: {len(dos_activity[src_ip])} packets in the last 10 seconds")

# Функція для перевірки фішингових атак за допомогою регулярних виразів
def check_phishing_attacks(packet):
    """Перевірка на фішингові атаки за допомогою регулярних виразів.

    Використовує регулярні вирази для перевірки фішингових URL в HTTP-пакетах.
    """
    if 'http' in packet:
        if hasattr(packet.http, 'host') and hasattr(packet.http, 'request_full_uri'):
            host = packet.http.host
            uri = packet.http.request_full_uri
            for pattern in phishing_patterns:
                if pattern.match(host) or pattern.match(uri):
```

```
                log_detection(f"Possible phishing attempt detected: Host - {host}, URI - {uri}")

# Аналіз пакетів з додаванням детекції аномалій та фішингових атак
for packet in capture:
    update_traffic_stats(packet)  # Оновлення статистики трафіку
    check_arp_spoofing(packet)
    check_dns_spoofing(packet)
    check_packet_delays(packet)
    check_unencrypted_traffic(packet)
    check_ssl_certificates(packet)
    check_hsts_headers(packet)
    check_ssh_bruteforce(packet)
    check_ssh_commands(packet)
    check_dos_attacks(packet)
    check_phishing_attacks(packet)

# Перевірка наявності аномалій у трафіку після аналізу всіх пакетів
check_traffic_anomalies()

# Закриття файлу після аналізу
capture.close()
```