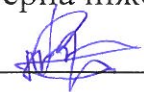


Міністерство освіти і науки України
Державний університет «Житомирська політехніка»
Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки

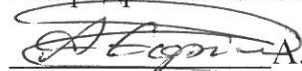
Пояснювальна записка

до випускної кваліфікаційної роботи бакалавра
на тему: «Проект підсистеми “Security Deception” комп'ютерної
мережі організації»

Виконав студент 4-го курсу, групи КІ-3
спеціальності 123 «Комп'ютерна інженерія»

 К. В. Поліщук

Керівник к.т.н., завідувач кафедри КІ та КБ

 А. А. Єфіменко

Рецензент к.т.н., доцент, доцент, КІ та КБ

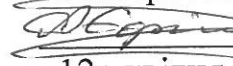
 Ю. М. Россінський

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА КІБЕРБЕЗПЕКИ

СПЕЦІАЛЬНІСТЬ 123 «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ»

ЗАТВЕРДЖУЮ

Завідувач кафедри
комп'ютерної інженерії
та кібербезпеки

 А.А. Єфіменко
«12» квітня 2021р.

ЗАВДАННЯ

на випускню кваліфікаційну роботу

Студентки Поліщук Катерини Вадимівни

Тема роботи: «Проект підсистеми “Security Desception” комп'ютерної мережі організації»

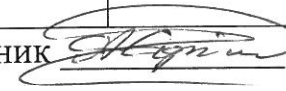
Затверджена Наказом університету від 12 квітня 2021 № 133/С

Термін здачі студентом закінченої роботи 23 червня 2021 р.

Вихідні дані роботи (зазначається предмет і об'єкт дослідження): процес проектування і реалізації підсистеми Security Desception в комп'ютерній мережі організації

Консультанти з випускної кваліфікаційної роботи із зазначенням розділів, що їх стосуються

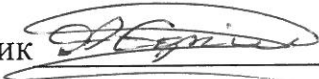
Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1			
2			
3			

Керівник  А.А. Єфіменко

Календарний план

№ з/п	Назва етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постановка задачі. Пошук, огляд та аналіз аналогічних розробок. Формулювання технічного завдання. Опрацювання літературних джерел.	2 березня 2021 – 5 квітня 2021	
2	Проектування структурної схеми	6 квітня 2021 – 12 квітня 2021	
3	Налагодження обладнання мережі	13 квітня 2021 – 26 квітня 2021	
4	Тестування	27 квітня 2021 – 4 травня 2021	
5	Захист	23 червня 2021	

Студент  Поліщук К.В.

Керівник  Єфіменко А.А.

АНОТАЦІЯ

Випускна кваліфікаційна робота бакалавра містить проект комп'ютерної мережі з вбудованим механізмом захисту підсистеми «Security Deception» та пояснювальну записку. Пояснювальна записка до випускної роботи містить 60 сторінок, 41 ілюстрація, 3 таблиці, 3 додатки та 23 літературних джерела.

Метою випускної кваліфікаційної роботи є розробка та реалізація проекту підсистеми Security Deception в комп'ютерній мережі організації.

Об'єктом випускної кваліфікаційної роботи є процес проектування та реалізація підсистеми Security Deception в комп'ютерній мережі організації.

Предметом випускної кваліфікаційної роботи є комп'ютерна мережа з налаштованою підсистемою Security Deception.

В роботі визначено основні завдання на розробку комп'ютерної мережі, проаналізовано аналоги механізму захисту Security Deception. Наведено сценарії роботи комп'ютерної мережі, описана взаємодія підсистеми Security Deception та злоумисника. Також наводяться результати тестування.

КЛЮЧОВІ СЛОВА: IP-АДРЕСА, КОМУТАТОР, МАРШРУТИЗАТОР, MAC-АДРЕСА, DHCP, HONEYPOT, SECURITY DECEPTION, SERVER.

					КІК.КР.Б-123-17-ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Проект підсистеми “Security Deception” комп’ютерної мережі	Літ.	Арк.	Аркуші
Розроб.		Поліщук К.В.		22.06.21				
Керівник		Сфіменко А.А.		22.06.21			4	
Реценз.		Росінський Ю.М.		22.06.21		ФІКТ, Гр. КІ-3		
Н. контр.								
Затверд.		Сфіменко А.А.		22.06.21				

ABSTRACT

Bachelor's graduate work consists of a computer network project with a built-in security mechanism of the "Security Deception" subsystem and an explanatory note.

Documentation for bachelor's graduate work contain 60 pages, 41 illustrations, 3 tables, 3 appendices and 23 references.

The purpose of graduation work is to develop and implement a project of the Security Deception subsystem in the computer network of the organization.

The object of graduation work is the design and implementation of the Security Deception subsystem in the computer network of the organization.

The subject of graduation work is a computer network with a configured Security Deception subsystem.

The main tasks for the development of a computer network are defined in the work the analogues of the Security Deception protection mechanism are analyzed. The scenarios of the computer network are described, the interaction of the Security Deception subsystem and the attacker is described. Test results are also given.

KEYWORDS: IP-ADDRESS, SWITCHBOARD, ROUTER, MAC-ADDRESS, DHCP, HONEYPOT, SECURITY DECEPTION, SERVER.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		5

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

IT – Information Technology – Інформаційні технології

LAN – Local Area Network – Локальна мережа

DMZ – DeMilitarized Zone – Демілітаризована зона

RID – Routing Information Protocol – Протокол маршрутизації інформації

TCP – Transmission Control Protocol – Протокол управління передачею

IP – Internet Protocol Address – Адреса Інтернет-протоколу

BOOTP – Bootstrap Protocol – Протокол Bootstrap

DHCP – Dynamic Host Configuration Protocol – Протокол динамічної
конфігурації хосту

SMB – Server Message Block – Блок повідомлень сервера

SSH – Secure Shell – Захищена оболонка

ПК – A personal computer – Персональний комп'ютер

ОС – Operating System – Операційна система

VLAN – Virtual Local Area Network – Віртуальна локальна мережа

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		6

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1 ПРИНЦИПИ ОРГАНІЗАЦІЇ СИСТЕМИ ТИПУ “SECURITY DECEPTION” В КОМП’ЮТЕРНИХ МЕРЕЖАХ.....	10
1.1. Принципи та технології побудови системи HoneyPot.....	10
1.2. Принципи та технології побудови системи SandBox	15
1.3. Принципи та технології побудови системи Security Deception	16
1.4. Порівняльний аналіз технологій: HoneyPot, SandBox та Security Deception	19
1.5. Постановка технічного завдання	22
Висновки до розділу 1	23
РОЗДІЛ 2. ПРОЕКТУВАННЯ ПІДСИСТЕМИ “SECURITY DECEPTION” КОМП’ЮТЕРНОЇ МЕРЕЖІ ОРГАНІЗАЦІЇ	24
2.1. Проектування фізичної інфраструктури організації.....	24
2.2. Проектування логічної інфраструктури мережі та розробка схеми адресації	28
2.3. Проектування підсистеми Security Deception	30
Висновки до розділу 2	33
РОЗДІЛ 3 ВИБІР ОБЛАДНАННЯ, ПЗ, НАЛАШТУВАННЯ ТА ТЕСТУВАННЯ МЕРЕЖІ ТА ПІДСИСТЕМИ SECURITY DECEPTION	34
3.1. Вибір обладнання та ПЗ для побудови мережі.....	34
3.2 Налаштування роботи мережі та підсистеми Security Deception.....	37
3.3 Тестування роботи мережі та підсистеми Security Deception	42
Висновки до розділу 3	49
ВИСНОВКИ.....	50
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	51
ДОДАТКИ.....	53

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		7

ВСТУП

Зараз комп'ютери та ноутбуки є доступними для всіх, на відміну від перших примірників, які ще і у додачу були громіздкими. Сьогодні кожна людина проводить декілька годин свого дня в інтернеті. Інтернет є відмінним сховищем інформації. Кожна людина може звернутися до його ресурсів і відшукати відповідь на своє питання. Енциклопедії та довідники залишилися в далекому минулому. Тепер по кожному питанню можна знайти відповідь в мережі.

У січні 2021 року було зареєстровано близько 4,6 млрд інтернет користувачів. Інтернет став невід'ємною частиною в житті людей, вони використовують інтернет для запису в лікарні, спілкування з іншими людьми, для оплати платежів, тощо. Вся інформація поступово переходить у цифровий формат, тому потреба покращення заходів безпеки зростає в рази.

Стрімкий технологічний розвиток змушує приділяти пильну увагу особистої безпеки в мережі. Чим сильніше і швидше нові технології проникають в життя людини, тим більшої ваги набуває питання захисту персональних даних. За оцінками експертів, в майбутньому ринок персонального страхування від кіберзагроз досягне багатомільярдного рівня.

Раніше фахівцям з кібербезпеки потрібно було чекати поки зловмисники завершать свою атаку, перш ніж почати аналізувати її та приймати рішення по усуненню наслідків. З появою HoneyPot у 1989 році, все змінилось. З'явилась можливість встановлювати приманки, котрі відволікають увагу від справжньої системи та дають змогу проаналізувати дії зловмисників.

Метою випускної кваліфікаційної роботи є розробка та реалізація проекту підсистеми Security Deception в комп'ютерній мережі організації.

Об'єктом випускної кваліфікаційної роботи є проектування та реалізація підсистеми Security Deception в комп'ютерній мережі організації.

Предметом випускної кваліфікаційної роботи є комп'ютерна мережа з налаштованою підсистемою Security Deception.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		8

В роботі визначено основні завдання на розробку комп'ютерної мережі, проаналізовано аналоги механізму захисту Security Deception. Наведено сценарії роботи комп'ютерної мережі, описана взаємодія підсистеми Security Deception та зловмисника. Також наводяться результати тестування.

Практичне застосування: Сучасна ситуація із захистом периметра мережі все частіше розглядається фахівцями. Відзначається тенденція, при якій зловмисники виявляються через значний час знаходження в мережі. Оскільки його дії несуть в собі загрозу, необхідно вжити заходів, що дозволяють фахівцям з безпеки вивчати зловмисників в сприймається ним як природному середовищі, тобто в імітованому середовищі, де неможливо нанесення реального збитку. Це можливо з впровадженням у мережу підсистеми Security Deception.

Розподіл пасток і приманок доцільно здійснювати по всій мережі і на кінцевих точках, які представляються інформаційними активами організації. Як тільки зловмисник підключається до середовища, видається попередження, що дає організації можливість прийняти рішення – або виключити його з мережі, або вивчити його методи і поведінку в контрольованому середовищі.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		9

РОЗДІЛ 1 ПРИНЦИПИ ОРГАНІЗАЦІЇ СИСТЕМИ ТИПУ “SECURITY DECEPTION” В КОМП’ЮТЕРНИХ МЕРЕЖАХ

1.1. Принципи та технології побудови системи HoneyPot

HoneyPot – ресурс, який представляє собою приманку для зловмисників.

Завдання HoneyPot – піддатися атаці або несанкціонованого дослідження, що згодом дозволить вивчити стратегію зловмисника і визначити перелік засобів, за допомогою яких можуть бути нанесені удари по реально існуючих об’єктах безпеки. Реалізація HoneyPot може являти собою як спеціальний виділений сервер, так і один мережевий сервіс, завдання якого – привернути увагу зловмисників.

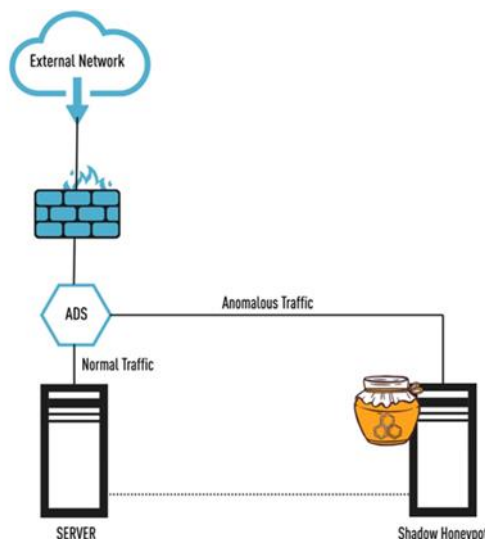


Рисунок 1.1 – Схема розміщення HoneyPot

HoneyPot – ресурс, який без будь-якого впливу на нього нічого не робить. HoneyPot збирає невелику кількість інформації, після аналізу яких ґрунтується статистика методів, якими користуються зловмисники, а також визначається наявність будь-яких нових сучасних рішень, які згодом будуть застосовуватися в боротьбі з ними.

Наприклад, веб-сервер, який не має імені і фактично нікому не відомий, не повинен, відповідно, мати і гостей, що заходять на нього, тому всі особи, які намагаються на нього проникнути, є потенційними хакерами. HoneyPot збирає

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				10
Змн.	Арк.	№ докум.	Підпис	Дата		

інформацію про характер поведінки цих зломисників і про їх способи впливу на сервер. Після чого фахівці в області інформаційної безпеки розробляють стратегії відбиття атак зломисників.

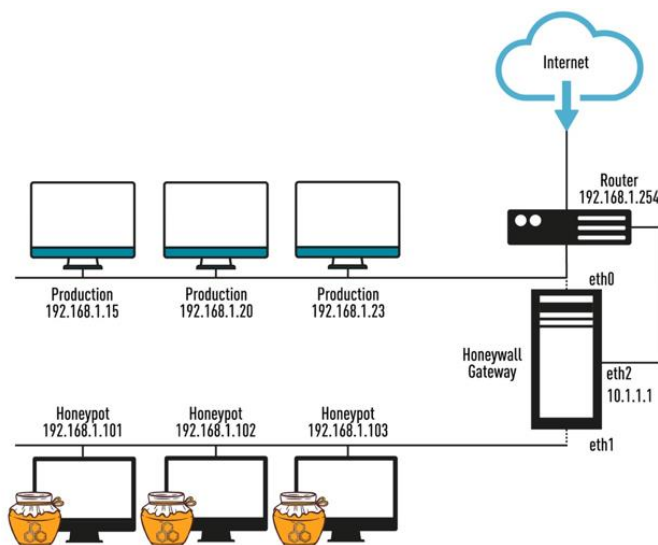


Рисунок 1.2 – Типовий HoneyNet

Якщо об'єднати кілька HoneyPots в мережу, то ми отримаємо вже більш ефективну систему HoneyNet, яка представляє собою емуляцію корпоративної мережі компанії (веб-сервер, файл-сервер і ін. компоненти мережі). Таке рішення дозволяє зрозуміти стратегію дій зломисників і ввести їх в оману. Типовий HoneyNet, як правило, працює паралельно з робочою мережею і абсолютно незалежно від неї. Таку «мережу» можна опублікувати в Інтернет по окремому каналу, під неї також можна виділити окремий діапазон IP-адрес.

Сенс застосування HoneyNet – показати хакеру, що він нібито проник в корпоративну мережу організації, насправді зломисник знаходиться в «ізолюваному середовищі» і під пильним наглядом фахівців ІБ.

Виділяють такі типи HoneyPot:

- HoneyPot створені на виділених серверах;
- програмно-емульовані HoneyPot.

HoneyPot, встановлений на виділеному сервері, дозволяє максимально наблизити його до реального сервера, роль якого він виконує (сервер даних, сервер додатків, проксі-сервер). Емульований HoneyPot швидко відновлюється

при взломі, а також чітко обмежується від основної ОС. Він може бути створений за допомогою віртуальної машини або Honeyd.

Різниця між ними лежить в масштабах мережі. Якщо, наприклад, це мала офісна мережа, то не має особливого сенсу ставити виділений сервер для протоколювання підозрілих подій в мережі. Тут досить буде обмежитися віртуальною системою або навіть одним віртуальним сервісом. У великих організаціях використовуються саме виділені сервери з повністю відтвореними на них мережевими службами. Зазвичай в конфігурації таких служб спеціально допускають помилки, щоб у зловмисника вдався взлом системи. В цьому і полягає основна ідея HoneyPot – заманити зловмисника.

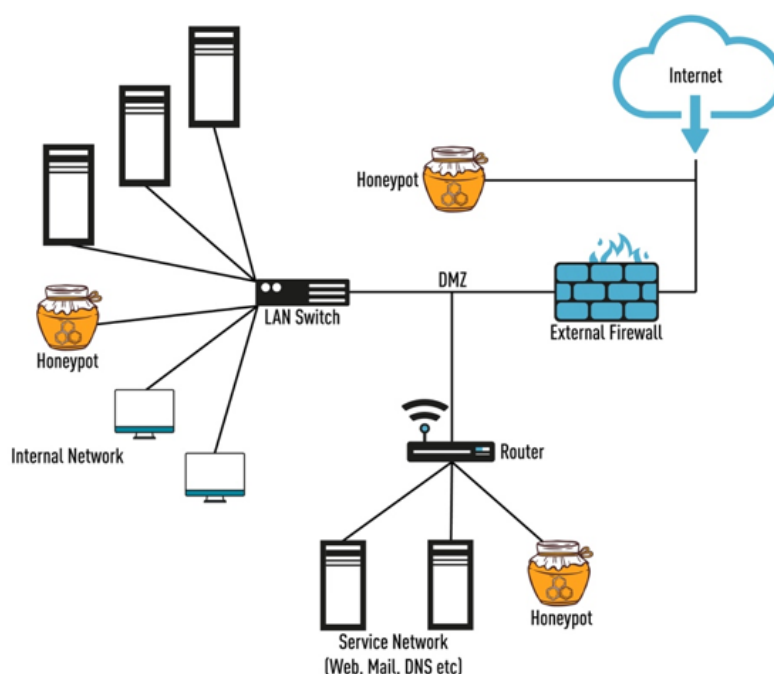


Рисунок 1.3 – Схема розгортання HoneyNet в організації

Технології HoneyPot надають аналітикам деякі переваги:

- збір змістовної інформації;
- невимогливість до системних ресурсів;
- простота установки, конфігурації та експлуатації;
- наочність необхідності використання.

Засоби HoneyPot мають кілька недоліків. HoneyPot не замінюють ніяких механізмів безпеки; вони тільки працюють і розширюють повну архітектуру безпеки.

Головними проблемами HoneyPot є:

- обмежена область бачення;
- можливість розкриття HoneyPot;
- ризик злому HoneyPot і атаки вузлів сторонніх організацій.

Різні варіанти розміщення HoneyPot допоможуть дати повні відомості про тактику нападника. Розміщення HoneyPot всередині локальної мережі дасть уявлення про атаки зсередини мережі, а розміщення на загальнодоступних серверах цієї мережі або в DMZ – про атаки на незахищені мережеві служби, такі як: поштові сервіси, SMB, FTP-сервери і т.д.

HoneyPot може бути встановлений всередині локальної мережі (після firewall) — тобто на комп'ютерах локальної мережі і серверах. Якщо не проводиться віддалене адміністрування мережі, то слід перенаправляти весь вхідний SSH-трафік на HoneyPot. Беручи мережевий трафік, система-пастка повинна протоколювати всі події, причому на низькому рівні. HoneyPot – це не проста програма, яка записує логи сервера. Якщо зловмисник зміг отримати доступ до сервера, стерти всі логи йому не складе труднощів. В ідеалі всі події в системі повинні записуватися на рівні ядра.

У демілітаризованій зоні або DMZ розташовуються загальнодоступні сервери. Наприклад, це може бути веб-сервер або поштовий сервер. Оскільки наявність таких серверів часто привертає увагу спамерів і зломщиків, необхідно забезпечити їх інформаційний захист. Установка HoneyPot на сервери DMZ є одним з рішень цієї проблеми.

Якщо ж немає виділеного веб-сервера, можна емулювати веб-служби за допомогою програмних HoneyPot. Вони дозволяють в точності відтворити неіснуючу насправді веб-сервер і заманити зломщика. Емуляція поштових та

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		13

інших мережевих служб обмежується лише вибором конкретного програмного рішення.

Мережа з двома, трьома і більше HoneyPot за визначенням називається HoneyNet. Вона може бути обмежена від робочої мережі. Керуючий трафік, який потрапляє в HoneyNet, повинен фіксуватися пастками цієї мережі.

Всі відомі HoneyPot можна поділити на 2 класи – відкриті і комерційні.

Таблиця 1.1 – Класи відомих HoneyPot

Відкриті	Комерційні
Deception Toolkit – найперший open-source HoneyPot, датований 1997 роком.	Specter – комерційний низькорівневий HoneyPot для ОС Windows. Емулює 13 різних ОС.
HoneyWeb – емулює різні типи веб-сервісів.	HoneyPot Manager – комерційний HoneyPot. Емулює сервер із встановленою СУБД Oracle.
BackOfficer Friendly – HoneyPot для ОС Windows, може застосовуватися в якості HIPS.	
Honeyd – низькорівневий HoneyPot для Linux, здатний емулювати сотні ОС.	

Також розрізняють такі типи HoneyPot:

- з високою взаємодією;
- з низькою взаємодією;
- клієнтський.

Замість того, щоб імітувати повномасштабну виробничу систему, HoneyPot з високою взаємодією імітують роботу всіх послуг реальної виробничої системи. HoneyPot з низькою взаємодією функціонують як механізми раннього виявлення, використовуючи лише обмежену кількість служб. Клієнтські HoneyPot копіюють важливі частини середовища клієнта, щоб допомогти в більш цілеспрямованих атаках. Фактичні дані про клієнтів не використовуються, натомість клієнтські HoneyPot роблять підроблений хост схожим на законний.

1.2. Принципи та технології побудови системи SandBox

У кібербезпеці SandBox – це ізольоване середовище в мережі, що імітує операційне середовище кінцевого користувача. SandBox використовуються для безпечного виконання підозрілого коду, не ризикуючи нашкодити хост-пристрою або мережі.

Використання SandBox для розширеного виявлення зловмисного програмного забезпечення, забезпечує ще один рівень захисту від нових загроз безпеці – зокрема, шкідливих програм та стелс-атак. А те, що відбувається в SandBox залишається, уникаючи збоїв системи та не даючи програмному уразливості поширюватися.

Середовища SandBox забезпечують активний рівень захисту мережі від нових і вдосконалених стійких загроз (APT). APT – це спеціально розроблені цілеспрямовані атаки, часто спрямовані на компрометацію організацій та викрадення даних. Вони призначені для ухилення від виявлення і часто літають під радаром більш простих методів виявлення.

Тестування SandBox попередньо виявляє шкідливе програмне забезпечення, виконуючи або детонуючи код у безпечному та ізольованому середовищі, щоб спостерігати за поведінкою та активністю цього коду. Традиційні заходи безпеки реагують і базуються на виявленні підписів, що працює шляхом пошуку шаблонів, виявлених у відомих випадках шкідливого програмного забезпечення.

Існує кілька варіантів реалізації SandBox. Розглянемо, що включають основні різновиди її реалізації:

- повна емуляція системи;
- віртуалізація.

При повній емуляції SandBox імітує фізичне обладнання хост-машини, забезпечуючи глибоку видимість поведінки та впливу програми. Віртуалізація використовує SandBox на основі віртуальної машини для вміщення та перевірки підозрілих програм.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		15

1.3. Принципи та технології побудови системи Security Deception

Кібербезпека є важливою складовою життя у мережі. Створення безпечних комп'ютерних систем і додатків є метою діяльності мережесистемних інженерів і програмістів. У зв'язку із складністю і трудомісткістю більшості процесів і методів захисту цифрового обладнання, інформації та комп'ютерних систем від ненавмисного чи несанкціонованого доступу, вразливості комп'ютерних систем становлять значну проблему для їхніх користувачів. Завдяки старанням фахівців з кібербезпеки наша інформація залишається конфіденційною.

Security Deception – це нова категорія захисту кібербезпеки. Продукти даної технології можуть виявляти, аналізувати і захищати від атак нульового дня і розширених атак, часто в режимі реального часу. Вони автоматизовані, точні і дають уявлення про зловмисні дії у внутрішніх мережах, які можуть бути невидимі для інших типів кіберзахисту. Security Deception забезпечує більш активну систему безпеки, намагаючись обдурити зловмисників, виявити їх, а потім перемогти їх, дозволяючи підприємству повернутися до нормальної роботи.

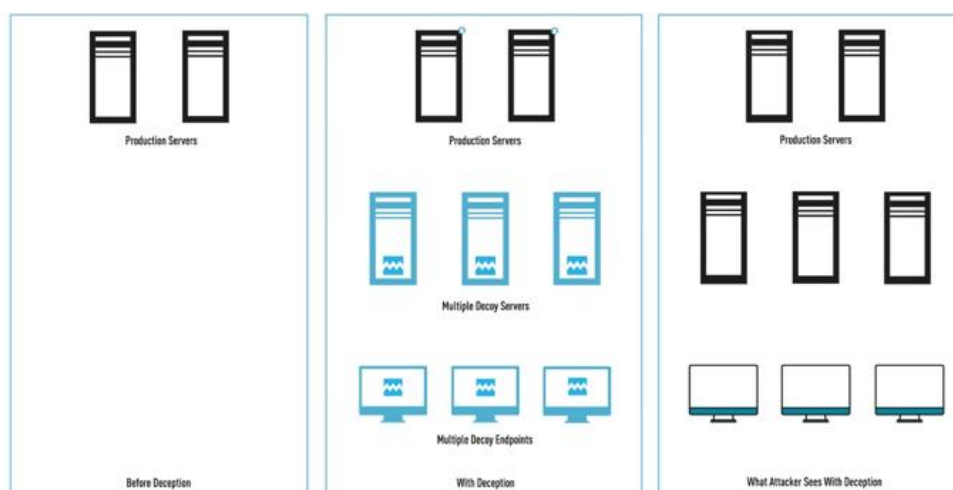


Рисунок 1.4 – Принцип роботи Distributed Deception Platform

Security Deception враховує точку зору зловмисника, метод використання навігації по мережах для ідентифікації та вилучення даних. Він інтегрується з

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		16

існуючими технологіями для забезпечення нової видимості внутрішніх мереж, обміну попередженнями з високою ймовірністю і аналітичними даними про погрози з існуючою інфраструктурою.

Традиційні технології кіберзахисту, такі як міжмережні екрани і захист кінцевих точок, генерують безліч попереджень. На великому підприємстві обсяг попереджень може досягати в деяких випадках мільйонів попереджень в день. Персонал служби безпеки не може легко обробити велику частину активності, але для компрометації всієї мережі досить одного успішного проникнення.

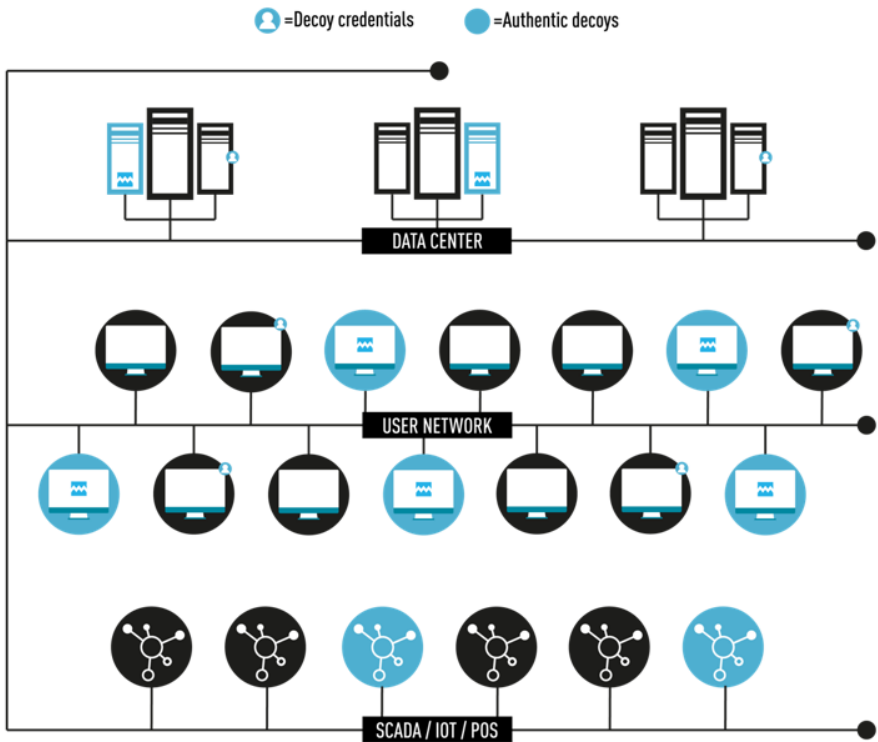


Рисунок 1.5 – Технологія іммітації ІТ-інфраструктури і дезінформації зловмисників

Технологія обману створює попередження, які є кінечним продуктом двійкового процесу. Вірогідність значно зменшена до двох значень: 0% і 100%. Будь-яка сторона, яка намагається ідентифікувати, пінгувати, увійти, переглянути будь-яку пастку (приманку) чи використовує приманку, негайно ідентифікується такою поведінкою. Не варто торкатися цих пасток чи приманок. Це перевага перед евристичним і імовірніснимевристичним

підходами та багатьма тисячами сторонніх попереджень, які генеруються цими методами.

Краща практика показує, що Security Deception, не є окремою стратегією - це додатковий сумісний рівень до існуючого більш глибокого кіберзахисту. Партнерська інтеграція робить його найбільш корисним, а мета полягає в тому, щоб додати захист від самих просунутих і витончених зловмисників, які зможуть успішно проникнути через периметр.

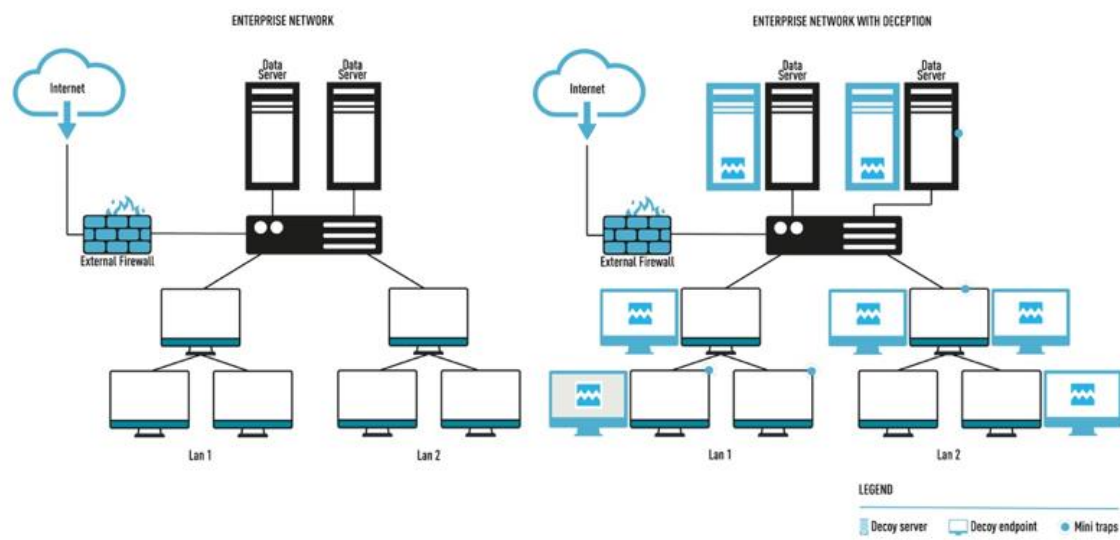


Рисунок 1.6 – Корпоративна мережа до та після впровадження Deception

Однією з найважливіших вимог до успішного впровадження технології обману є те, що вона повинна залишатися невідмінною та свіжою для зловмисника. Якщо зловмисник підозрює, що їх обманюють, вони зроблять все можливе, щоб уникнути пасток та розширити свої зусилля, щоб дістатися до вашого реального майна.

Багато рішень щодо захисту від обману мають в основі машинне навчання та штучний інтелект вбудовані в їхнє ядро. Ці функції не тільки забезпечують підтримку методів обману, але також допомагають зменшити оперативні накладні витрати та вплив на команди безпеки, звільняючи їх від постійного створення нових кампаній обману.

1.4. Порівняльний аналіз технологій: HoneyPot, SandBox та Security Deception

Існує безліч технологій для захисту комп'ютерних мереж організацій. Порівняємо найпопулярніші технології та оберемо для себе найкращий варіант.

Розглянемо такі технології, як:

- SandBox;
- HoneyPots;
- Security Deception.

Security Deception – це мережі і стратегії, які використовуються для попередження кібератак, постійно розвиваються. Security Deception – це тактика кіберзахисту, яка дозволяє дослідникам і фахівцям з інформаційної безпеки спостерігати за поведінкою зловмисників після того, як вони отримали доступ до того, що, на їхню думку, є бізнес-мережею.

Термін «Security Deception» став широко використовуватися у 2017 році, тому може бути складно сказати, чим саме ці рішення відрізняються від інших інструментів, які намагаються обдурити зловмисників, таких як SandBox і HoneyPots.

Подібно цих тактик, Security Deception змушує зловмисників і шкідливі програми розкрити себе, щоб дослідники могли розробити ефективний захист від них, але це більше покладається на автоматизацію та масштабування, а також вимагає менше досвіду для налаштування та управління. Кожна з цих технологій має унікальні вимоги та ідеальні варіанти використання.

SandBox – це система захисту, яка існує майже з самого початку мереж та сторонніх програм. SandBox, введена в 1970-х для тестування програм штучного інтелекту, дозволяє встановлювати та запускати шкідливе програмне забезпечення у закритому середовищі, де дослідники можуть контролювати свої дії для виявлення потенційних ризиків та контрзаходів.

Сьогодні ефективна SandBox найчастіше виконується на виділених віртуальних машинах на віртуальному хості. Це дозволяє безпечно тестувати

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		19

зловмисне програмне забезпечення на різних версіях ОС на машинах, які відокремлені від мережі.

Дослідники безпеки використовують SandBox під час аналізу шкідливого програмного забезпечення, і багато вдосконалених засобів захисту від шкідливих програм використовують його, щоб визначити, чи є підозрілі файли справді шкідливими на основі їх поведінки. Такі типи рішень для боротьби зі шкідливим програмним забезпеченням стають все більш важливими, оскільки значна частина сучасного шкідливого програмного забезпечення затуманена, щоб уникнути антивірусу на основі підписів.

Більшість підприємств не здатні проводити аналіз шкідливого програмного забезпечення на такому ж рівні досвіду, як спеціалізований дослідник або постачальник. Менший бізнес, як правило, отримує найбільшу користь від розгортання SandBox як послуги від постачальника, який вже має реалізації, які можуть автоматизувати весь процес.

HoneyPots і HoneyNets – це навмисно вразливі системи, призначені для привернення уваги зловмисників. HoneyPots – це єдині хости, які спонукають зловмисників намагатися викрасти цінні дані або розширити цільову мережу.

Ідея HoneyNets, які почали циркулювати в 1999 році, полягає в розумінні процесу та стратегії зловмисників. HoneyNets складаються з безлічі HoneyPots, які часто налаштовані на емуляцію фактичної мережі – разом із файловим сервером, веб-сервером тощо – так, щоб зловмисники вважали, що вони успішно проникли в мережу. Натомість вони фактично перебувають у ізольованому середовищі, під мікроскопом.

HoneyPots дозволяють дослідникам спостерігати, як поведуться реальні зловмисники, в той час як SandBox показує лише те, як поводить себе шкідливе програмне забезпечення. Для цієї мети дослідники та аналітики безпеки зазвичай використовують каструлі та медоносні мережі. Дослідники, а також фахівці в галузі ІТ та безпеки, які займаються обороною, можуть використовувати цю інформацію для покращення безпеки своєї організації,

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		20

відзначаючи нові методи атаки та впроваджуючи нові засоби захисту, що відповідають.

HoneyNets також витратить час зловмисників і може змусити їх розчаровано кинути атаку. Це найбільш корисно для урядових організацій та фінансових установ, на які часто спрямовуються хакери.

Таблиця 1.2 – Порівняння технологій захисту

Технологія захисту	Стратегія	Переваги	Недоліки
HoneyPot	Створення сервера/системи, які здаються уразливими	Збір змістовної інформації	Обмежена область бачення
		Невимогливість до системних ресурсів	Можливість розкриття
		Простота установки, конфігурації та експлуатації	Можливість використання в атаках на інші організації
SandBox	Повністю ізолює процес	Низьке споживання системних ресурсів	Неможливість протидії активному зараженню ПК
		Не вимогливі до апаратного забезпечення ПК	Користувач повинен мати знання про принципи функціонування ОС
Security Deception	Повністю автоматизована програма	Швидке і просте впровадження	Необхідна підтримка
		Основа ранньої детекції	Помилкове відчуття безпеки може змусити вас пропустити загрозу

Підвівши підсумки, ми можемо виділити спеціалізацію кожної технології:

- SandBox – відповідає за тестування шкідливого ПЗ;
- HoneyPot – відповідає за принадування зловмисників на окремо виділені пристрої;
- Security Deception – відповідає за розкриття зловмисників, шляхом створення ізольованого середовища на пристроях справжньої мережі.

З інформації викладеної вище, найкращим варіантом для нас є Security Deception, але через новітність технології, нам бракує документації.

Тому для захисту нашої мережі ми обрали підсистему HoneyPot.

1.5. Постановка технічного завдання

Головною метою даної роботи є розробка підсистеми “Security Deception” комп’ютерної мережі організації. Закріплення знань в процесі створення комп’ютерної мережі організації, налаштуванні адресації і в результаті – налаштування технології HoneyPot та пастки для вхідних SSH-з’єднань.

В випускній роботі необхідно розробити структурну схему комп’ютерної мережі організації, наблизити процес віртуальної мережі до реальної та створити її захист. Для реалізації поставленого завдання, основними етапами є:

1. Принципи організації системи типу “Security Deception” в комп’ютерних мережах

1.1 Принципи та технології побудови системи HoneyPot

1.2 Принципи та технології побудови системи SandBox

1.3 Принципи та технології побудови системи Security Deception

1.4 Порівняльний аналіз технологій: HoneyPot, SandBox та Security Deception

2. Проектування підсистеми “Security Deception” комп’ютерної мережі організації

2.1 Проектування фізичної мережної інфраструктури організації

2.2 Проектування логічної інфраструктури мережі та розробка схеми адресації

2.3 Проектування підсистеми “Security Deception”

3. Вибір обладнання, ПЗ, налаштування та тестування мережі та підсистеми “Security Deception”

3.1 Вибір обладнання для побудови мережі

3.2 Базове налаштування та тестування мережних пристроїв та кінцевих вузлів

3.3 Вибір ПЗ для підсистеми Security Deception

3.4 Налаштування та тестування роботи елементів підсистеми Security Deception

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		22

Висновки до розділу 1

Після розгляду теоретичного матеріалу з безпеки було визначено основні системи захисту комп'ютерних мереж. Серед них такі, як: HoneyPot, SandBox та Security Deception. В процесі розгляду було визначено головні принципи та технології побудови цих систем.

Кожна з цих систем застосовує підхід з приналежністю злоумисника чи злочинної програми в ізолюване середовище, де в подальшому аналізують їх поведінку. За допомогою цього дослідження, фахівці в області інформаційної безпеки розробляють підхід стратегії відбиття атак злоумисників в майбутньому.

Також було проведено порівняльний аналіз цих технологій захисту комп'ютерних мереж. В результаті аналізу було виявлено переваги та недоліки кожної системи, які в свою чергу дозволили обрати відповідну технологію захисту мережі для нашого проекту.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		23

РОЗДІЛ 2. ПРОЕКТУВАННЯ ПІДСИСТЕМИ “SECURITY DESCPTION” КОМП’ЮТЕРНОЇ МЕРЕЖІ ОРГАНІЗАЦІЇ

2.1. Проектування фізичної інфраструктури організації

Комп’ютерні мережі – основа спілкування в ІТ. Вони використовуються найрізноманітнішими способами і можуть включати безліч різних типів мереж. Комп’ютерна мережа – це сукупність комп’ютерів, пов’язаних між собою, щоб вони могли обмінюватися інформацією. Найдавніші приклади комп’ютерних мереж – з 1960-х років.

Комп’ютерні мережі використовуються для виконання великої кількості завдань за допомогою обміну інформацією. Деякі речі, для яких використовуються мережі:

- спілкування за допомогою електронної пошти, відео, обміну миттєвими повідомленнями та іншими методами;
- спільні пристрої, такі як принтери, сканери та ксерокс;
- дозволяє користувачам мережі легко отримувати доступ до інформації та підтримувати її.

Розберемо локальні мережі (LAN).

Локальна мережа – це мережа, яка з’єднує комп’ютери в межах обмеженої області. Це може бути в школі, офісі чи навіть вдома.

Мережа локальних мереж складається з кабелів, точок доступу, комутаторів, маршрутизаторів та інших компонентів, що дозволяють пристроям підключатися до внутрішніх серверів, веб-серверів та інших локальних мереж через глобальні мережі. Пристрої можуть використовувати одне підключення до Інтернету, обмінюватися файлами один з одним, друкувати на спільних принтерах, а також мати доступ та навіть керувати один одним. Приклад локальної мережі на рисунку 2.1.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		24

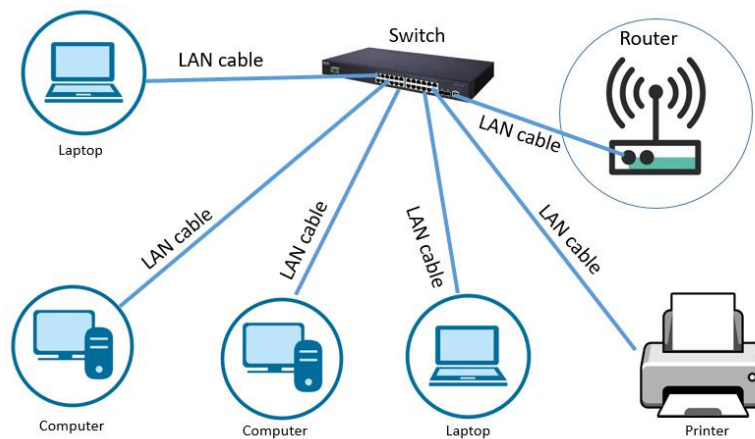


Рисунок 2.1 – Локальна мережа (LAN)

Основною базовою топологією комп'ютерних мереж є топологія типу Зірка. Головною відмінністю даної топології є підключення всіх робочих станцій до одного центрального вузла, в ролі якого, в більшості випадків, виступає комутатор, утворюючи фізичний сегмент мережі. Сегменти зіркоподібної топології можуть функціонувати як окремо, так і в складі складної мережевої топології.

У топології зірки всі вузли опосередковано з'єднуються між собою через один або кілька вимикачів. Комутатор діє як центральна точка, через яку проходять всі комунікації.

На рисунку 2.2 представлена комп'ютерна мережа з топологією зірка.

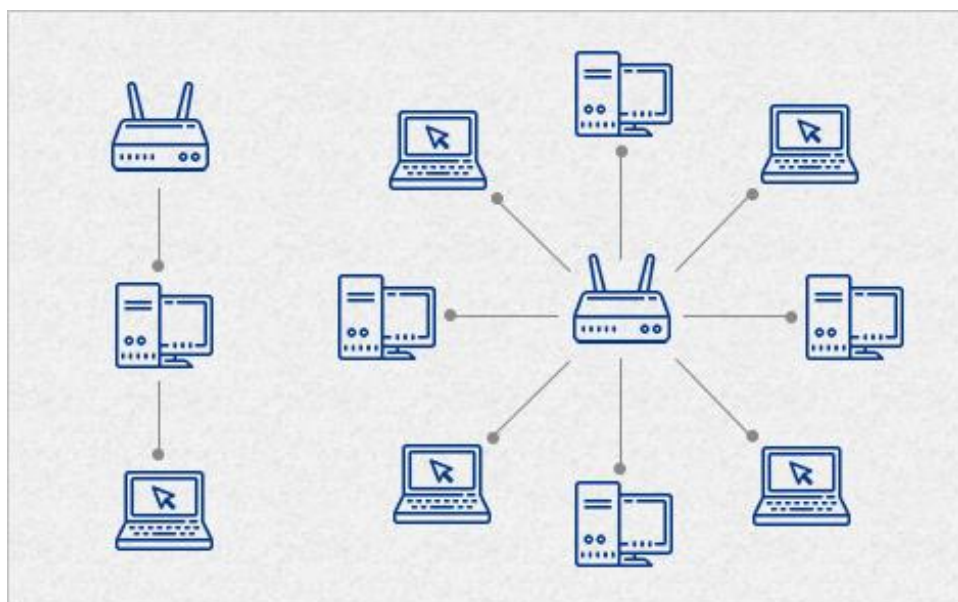


Рисунок 2.2 – Топологія зірки

Топології зірок, як правило, зустрічаються у великих організаціях, таких як навчальні заклади та підприємства, де високі показники є необхідними.

Вони також є в домашніх мережах, особливо в бездротових. У цьому випадку маршрутизатор забезпечує центральне з'єднання для всіх вузлів.

Розглянемо переваги використання топології зірок:

- кожен вузол підключений окремо, отже, вихід з ладу одного вузла або його ланки, (передавальний носій), не впливає на будь-які інші вузли;
- нові вузли можна додати до мережі, просто підключивши їх до комутатора;
- мережі зірок, як правило, мають вищу ефективність, оскільки повідомлення передається лише призначеному одержувачу.

Однак топології зірок мають і свої недоліки:

- вся мережа виходить з ладу, якщо виходить з ладу комутатор, оскільки жоден вузол не може зв'язатися з інтернетом;
- топологія дротових зірок вимагає великої кількості кабелю - у великій мережі це може коштувати дорого.

Для того, щоб не виникало труднощів вручну наділяти кожен комп'ютер своєю ір-адресою, було вирішено налаштувати протокол DHCP в підсистемі "Security Deception" комп'ютерної мережі організації.

DHCP – це протокол, який використовується для автоматичного призначення комп'ютерам параметрів конфігурації TCP/IP. DHCP є стандартом, описаним у RFC 1531. Кожен пристрій у мережі має IP-адресу. Ця IP-адреса може бути призначена статично або динамічно.

Також в даній роботі ми використали мережевий протокол SSH для того аби показати як можна спіймати злочинця у безкінечну петлю.

SSH – це мережевий протокол, який надає користувачам, зокрема системним адміністраторам, безпечний спосіб доступу до комп'ютера через захищену мережу.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		26

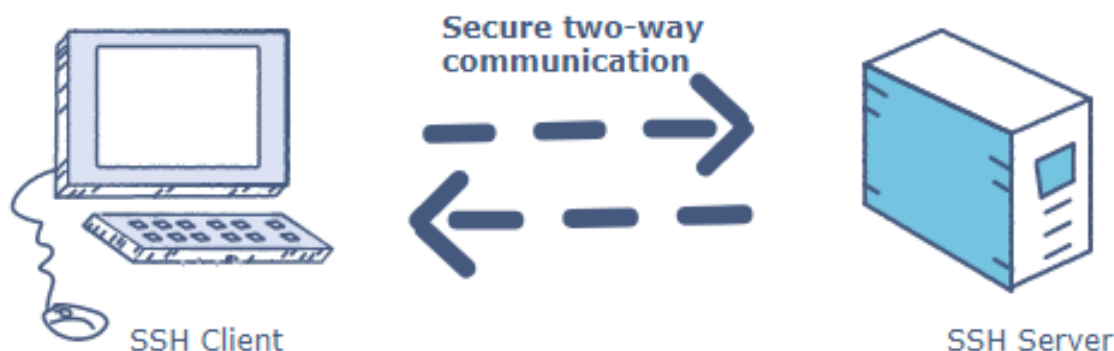


Рисунок 2.3 – Безпечна двустороння комунікація через мережеву архітектуру
Клієнт-Сервер

SSH забезпечує надійну автентифікацію пароля та/або відкритого ключа, за допомогою якого адміністратор системи/мережі може віддалено підключатися до будь-якого комп'ютера/ програми, виконувати команди, а також переміщувати файли за допомогою протоколу передачі файлів SSH.

Ми можемо використовувати протокол SSH у різних сценаріях, таких як:

- забезпечення безпечного доступу для користувачів та автоматизованих процесів;
- інтерактивна та автоматизована передача файлів;
- для видалення віддалених команд;
- управління мережевою інфраструктурою та іншими критично важливими компонентами системи.

Після налаштування протокол SSH використовує алгоритм шифрування та хешування для забезпечення конфіденційності даних, якими обмінюються.

Сервер – це екземпляр комп'ютерної програми, який відповідає на запити, зроблені іншою програмою.

Сервери працюють як частина моделі клієнт-сервер, модель, яка позначає обслуговування даних для клієнтів. Майже весь Інтернет заснований на моделі клієнт-сервер, при цьому мільйони серверів, підключених до інтернету, постійно працюють по всьому світу.

2.2. Проектування логічної інфраструктури мережі та розробка схеми адресації

При проектуванні мережі використана трирівнева ієрархічна модель мережі, в якій використовуються три рівні:

- Рівень ядра (Core layer);
- Рівень розподілу (Distribution layer);
- Рівень доступу (Access layer).

Загальна топологія спроектованої мережі представлена на рисунку 2.3.

Відповідно, рівень ядра складається з одного маршрутизатора.

Рівень розподілу складається зі з'єднаних між собою комутаторами за допомогою витой пари Fast Ethernet для забезпечення високої швидкості на даному рівні.

Рівень доступу представлений комутаторами з під'єднаними до них кінцевими вузлами.

У загальному в мережі використано три комутатори та один маршрутизатор.

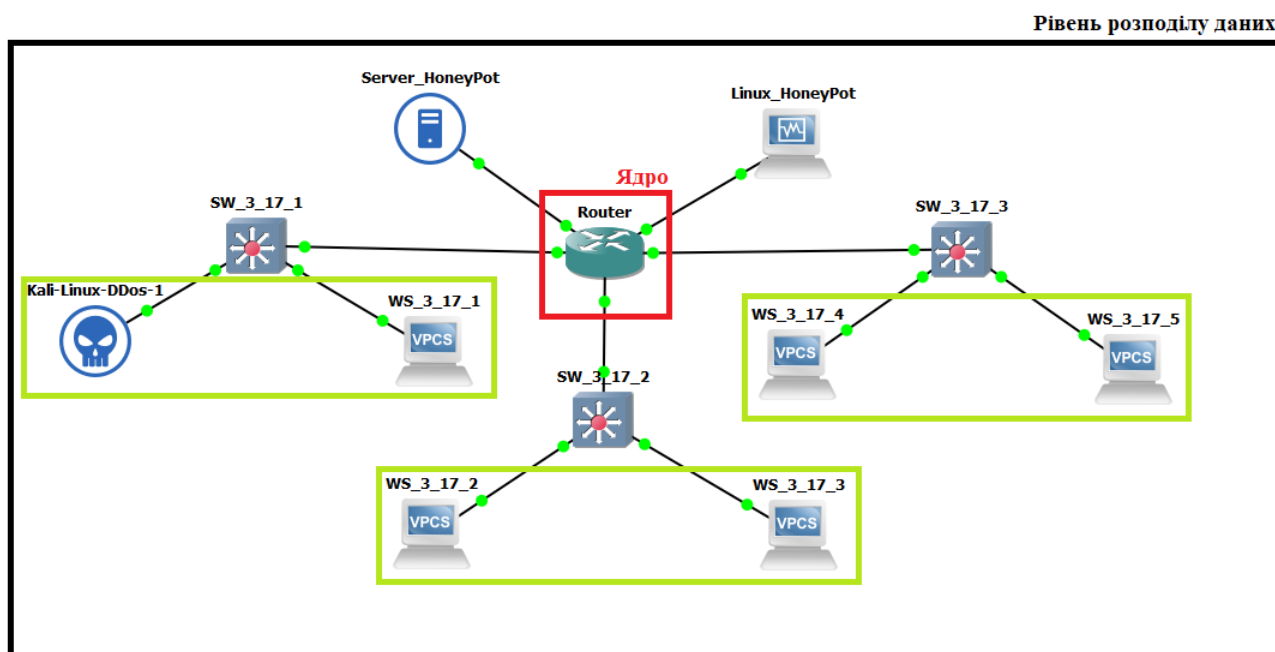


Рисунок 2.4 – Загальна топологія спроектованої мережі

Розглянемо детальніше спроектовану мережу.

Загальна структура мережі наведена у додатку А. Рівень доступу складається з трьох комутаторів, до кожного з яких підключені різні LAN. Комутатор SW_3_17_1 належить до віртуальної мережі LAN1, який має два кінцевих вузла, але розрахований на 253 вузла.

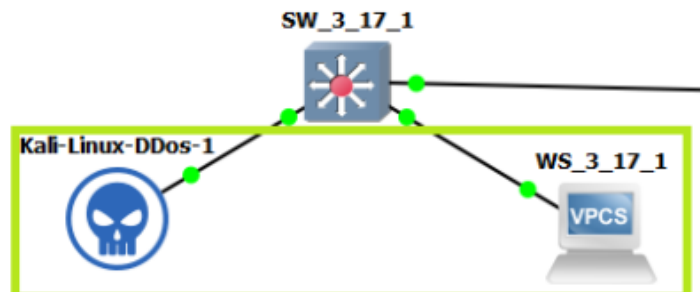


Рисунок 2.5 – До комутатора SW_3_17_1 підключені два кінцевих вузла

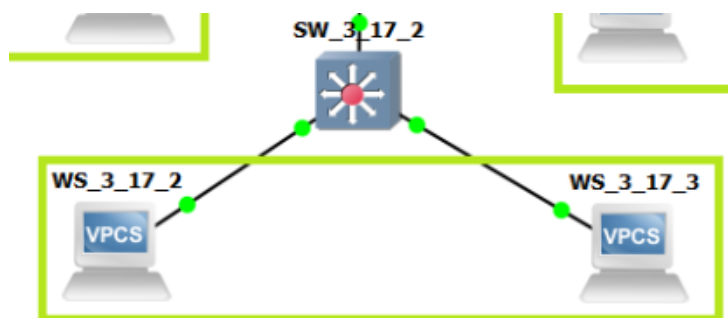


Рисунок 2.6 – До комутатора SW_3_17_2 підключені два кінцевих вузла

Адресація пристроїв мережі на рівні ядра наведена в таблиці 1.3.

Таблиця 2.1 - Адресація пристроїв мережі

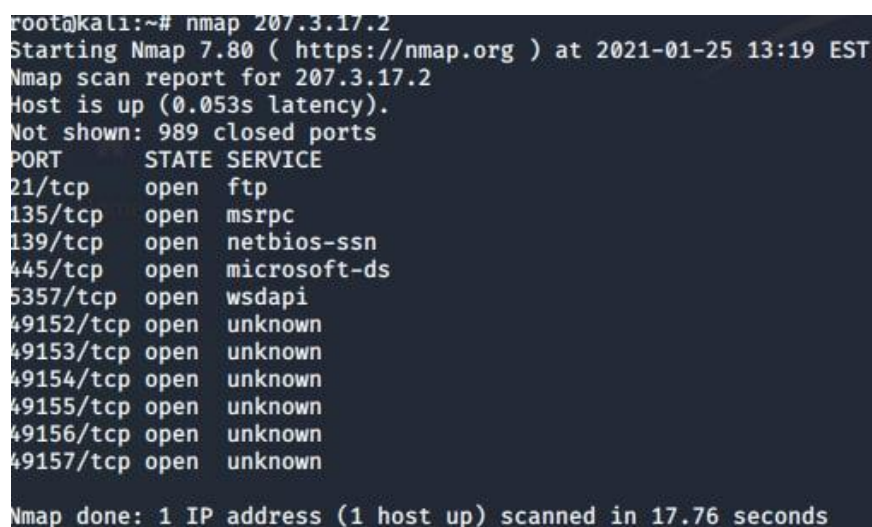
Пристрій мережі	Інтерфейс	IPv4-адреса	Маска мережі	Префікс мережі
Маршрутизатор Router	F0/0	207.3.17.1	255.255.255.0	/24
	F1/1	208.3.17.1		
	F2/0	209.3.17.1		
	F3/0	210.3.17.1		
	F4/0	207.3.17.2		
Комутатор SW_3_17_1	LAN1	208.3.17.2	255.255.255.0	/24
Комутатор SW_3_17_2	LAN2	209.3.17.2		
Комутатор SW_3_17_3	LAN3	210.3.17.2		

2.3. Проектування підсистеми Security Deception

Розглянемо роботу програми HoneyBot.

HoneyBot призначений для надання даних віддаленого датчика та інформації про рух кожному, хто підключається до нього. Але в його програмне забезпечення також вбудована акуратна маленька пастка. Коли зловмисник намагається щось зробити, HoneyBot робить вигляд, що виконує його команду. Моделюючи відповідь, він надсилає попередження своїм власникам. Зловмисник вважає, що він успішно скомпрометував роботу, а насправді все, що він зробив – це попередив команду з кібербезпеки про свою присутність.

На рисунку 2.7 представлений список портів з вимкненим HoneyBot.



```
root@kali:~# nmap 207.3.17.2
Starting Nmap 7.80 ( https://nmap.org ) at 2021-01-25 13:19 EST
Nmap scan report for 207.3.17.2
Host is up (0.053s latency).
Not shown: 989 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
5357/tcp  open  wsddapi
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49157/tcp open  unknown
Nmap done: 1 IP address (1 host up) scanned in 17.76 seconds
```

Рисунок 2.7 – Сканування серверу з вимкненим HoneyBot

Щоб приманка працювала, система повинна здаватися законною. HoneyPot повинен запускати процеси, які, як очікується, працюватиме виробнича система, і міститиме, здавалося б, важливі фіктивні файли. HoneyPot може бути будь-якою системою, яка налаштована з належними можливостями реєстрації. Також непогано розмістити HoneyPot за своїм корпоративним брандмауером – він не тільки забезпечує важливі можливості реєстрації та попередження, а також може заблокувати вихідний трафік, щоб скомпрометований HoneyPot не міг використовуватись для переміщення до інших внутрішніх активів.

Що стосується цілей, то існують два типи HoneyPot: дослідницькі та виробничі. Дослідницькі збирають інформацію про напади і використовуються спеціально для вивчення зловмисної поведінки. Вони збирають інформацію про тенденції зловмисників, штамів шкідливих програм і вразливостей.

З іншого боку, виробничі, орієнтовані на виявлення активних компромісів у вашій внутрішній мережі та обман зловмисника. Збір інформації як і раніше залишається пріоритетом, оскільки HoneyPot дають нам додаткові можливості моніторингу та заповнюють загальні прогалини у виявленні мережесканивань. Виробничі HoneyPot сидять із рештою виробничих серверів і запускають служби, які зазвичай працюють у середовищі. Дослідницькі HoneyPot, як правило, є більш складними і зберігають більше типів даних, ніж виробничі.

На рисунку 2.8 представлений список портів з включеним HoneyBot. Тут ми можемо побачити, що він створив купу підставних портів.

```
9876/tcp open sd
9898/tcp open monkeycom
10000/tcp open snet-sensor-mgmt
10082/tcp open amandaidx
11111/tcp open vce
12174/tcp open unknown
12345/tcp open netbus
13722/tcp open netbackup
20000/tcp open dnp
20005/tcp open btx
20031/tcp open unknown
31337/tcp open Elite
49152/tcp open unknown
49153/tcp open unknown
49154/tcp open unknown
49155/tcp open unknown
49156/tcp open unknown
49157/tcp open unknown
65000/tcp open unknown
```

Рисунок 2.8 – Сканування серверу з включеним HoneyBot

Розглянемо таку ОС, як Kali Linux, в якості атакуючої машини.

Kali Linux - це дистрибутив Linux на базі Debian. Це скрупульозно створена ОС, яка спеціально обслуговує аналітиків мережі та тестерів

проникнення. Наявність безлічі інструментів, попередньо встановлених разом із Kali, перетворює його на етичного хакерського швейцарського ножа.

Раніше відомий як Backtrack, Kali Linux рекламує себе як більш відшліфованого наступника з більш інструментами, орієнтованими на тестування, на відміну від Backtrack, який мав кілька інструментів, які могли б служити одній і тій самій цілі, в свою чергу, роблячи його заваленим непотрібними утилітами. Це робить етичний злом за допомогою Kali Linux спрощеним завданням.

Kali Linux в основному використовується для вдосконаленого тестування на проникнення та аудиту безпеки. Kali містить кілька сотень інструментів, спрямованих на виконання різних завдань інформаційної безпеки, таких як тестування на проникнення та дослідження безпеки.

Переваги використання Kali Linux:

- є вільні у користуванні;
- постачає понад 600 різних інструментів для тестування на проникнення та аналітики безпеки;
- вимоги до апаратного забезпечення є мінімальними.

Найпопулярнішими інструментами для злому є:

- Aircrack-ng – використовується для оцінки безпеки мережі WiFi;
- Nmap – використовується для виявлення мережі та аудиту безпеки;
- Hydra – використовується для злому веб-сканерів, бездротових мереж, розробників пакетів;
- WireShark – використовується для аналізу мережевих протоколів.

Тому Kali Linux є явним фаворитом для атак на HoneyBot.

В даній роботі ми також вирішили використати пастку (tarpit) для вхідних SSH-з'єднань, аби запобігти скануванню IP-адрес. Не секрет, що інтернет – дуже вороже середовище. Як тільки ви піднімаєте сервер, він миттєво піддається масованим атакам і множинним скануванням. На прикладі HoneyPot

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		32

можна оцінити масштаб цього сміттевого трафіку. Фактично, на середньому сервері 99% трафіку може бути шкідливим.

Tarpit – це порт-пастка, який використовується для уповільнення вхідних з'єднань. Якщо стороння система підключається до цього порту, то швидко закрити з'єднання не вийде. Їй доведеться витратити свої системні ресурси і чекати, поки з'єднання не перерветься з таймаут, або вручну розривати його.

Найчастіше тарпіти застосовують для захисту. Техніку вперше розробили для захисту від комп'ютерних черв'яків. А зараз її можна використовувати, щоб зіпсувати життя спамерам і дослідникам, які займаються широким скануванням всіх IP-адрес поспіль.

Одному з сисадмінів набридло спостерігати за цим неподобством і він написав маленьку програмку Endlessh, тарп для SSH, що уповільнює вхідні з'єднання. Програма відкриває порт (за замовчуванням для тестування вказано порт 2222) і прикидається SSH-сервером, а насправді встановлює нескінченне з'єднання з вхідним клієнтом, поки той не здасться. Це може тривати кілька днів або більше, поки клієнт не відвалиться.

Висновки до розділу 2

В даному розділі було проведено проектування фізичної мережної інфраструктури організації. Також було розглянуто такі поняття, як сервер, комп'ютерна мережа, її основні типи та топологія. Описано використовувані протоколи, DHCP та SSH. Проведено проектування логічних зв'язків мережі та розробка схеми адресації. Показано принцип роботи підсистеми Security Deception на прикладі програми HoneyBot. Розглянуто функціонал ОС Kali Linux та обрано його атакуючою машиною. Також було розглянуто порт-пастку SSH tarpit.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		33

РОЗДІЛ 3 ВИБІР ОБЛАДНАННЯ, ПЗ, НАЛАШТУВАННЯ ТА ТЕСТУВАННЯ МЕРЕЖІ ТА ПІДСИСТЕМИ SECURITY DESCPTION

3.1. Вибір обладнання та ПЗ для побудови мережі

Для підсистеми захисту комп'ютерної мережі організації на базі технології Security Descption потрібно обладнання:

- комутатор;
- маршрутизатор;
- сервер.

У даній мережі використовується така модель комутатора, як Cisco SB SF220-48P. Для підприємств, що вимагають високої продуктивності, безпеки та керованості, PoE-комутатор Cisco SB SF220-48P стане відмінним вибором. Він забезпечує необхідний рівень мережевих можливостей.

Завдяки комутатору можна не тільки створити ефективну і надійну мережу для підключення робочої сили, а й скористатися передовими рішеннями для надання послуг передачі даних, голосу і відео в конвергентної інфраструктури.

Девайс швидко і надійно з'єднує співробітників, що працюють в невеликих офісах, з усіма серверами, принтерами, пристроями, які вони використовують.

Завдяки перевірці автентичності пристрої та контролю доступу можна підтримувати цілісність ключової ділової інформації, одночасно надавати зв'язок і підвищувати продуктивність співробітників.



Рисунок 3.1 – Комутатор Cisco SB SF220-48P

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		34

Сьогодні безпечна мережа необхідна для кожного малого бізнесу. Cisco RV340, маршрутизатори безпеки RV345 і RV345P надають бізнес – користувачам високошвидкісного доступу, інтернет – безпеки, захисту брандмауера і політики додатків для максимальної ефективності бізнесу.

Малі підприємства постійно піддаються впливу інтернет-загроз. Гігабітні VPN-маршрутизатори Cisco RV340, RV345 і RV345P Dual WAN з'єднують малі підприємства з Інтернетом і захищають працівників від небажаного контенту і шкідливих веб-сайтів без шкоди для онлайн-досвіду. Маршрутизатор допомагають підвищити продуктивність співробітників і загальну продуктивність мережі, обмеживши інтернет-серфінг відповідними категоріями сайтів і усуваючи небажаний мережевий трафік. Користувачі автоматично захищені від шкідливих або зламаних веб-сайтів, незалежно від категорії сайтів. Кінцеві пристрої і додатки можна ідентифікувати і обробляти відповідно з політиками, які визначаються користувачем, для підвищення продуктивності і оптимального використання мережі.

Завдяки інтуїтивно зрозумілому інтерфейсу RV340, RV345 і RV345P дозволяють запускати в лічені хвилини. Ці маршрутизатори забезпечують надійний зв'язок, який настільки прозора, що ви не будете знати, що вона є.



Рисунок 3.2 – Маршрутизатор Cisco SB RV345P Dual WAN Gigabit

GNS3 – це програмне забезпечення з відкритим кодом, яке імітує складні мережі, наближаючись до можливостей реальних мереж. Все це без спеціального мережевого обладнання, такого як маршрутизатори та комутатори.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				35
Змн.	Арк.	№ докум.	Підпис	Дата		

Це програмне забезпечення надає інтуїтивно зрозумілий графічний користувацький інтерфейс для проектування та налаштування віртуальних мереж, воно працює на традиційному обладнанні ПК і може використовуватися в декількох операційних системах, включаючи Windows, Linux та MacOS.

GNS3 є чудовим альтернативним або доповнюючим інструментом для реальних лабораторій для мережеских інженерів, адміністраторів та людей, які навчаються для отримання сертифікатів, таких як Cisco CCNA, CCNP та CCIE.

VirtualBox – це міжплатформна програма для віртуалізації, яку можна встановити на комп'ютери на базі Intel або AMD під управлінням Windows, Linux, Mac OS, Solaris. Після встановлення цього конкретного програмного забезпечення на рідну операційну систему, що працює на робочому столі / ПК / ноутбуці, ви готові піти і створити стільки екземплярів віртуальних комп'ютерів, які потрібно встановити та запустити. Більшість сучасних комп'ютерів настільки потужні, щоб запускати цілі операційні системи у ваших основних операційних системах, а це означає, що віртуальні машини сьогодні є більш звичними як ніколи. Це машинний емулятор та віртуалізатор.

Зробити це ще простішим - це програмне забезпечення, яке допомагає запустити операційну систему в іншій операційній системі шляхом спільного використання ресурсів хост-комп'ютера, на який вона встановлена. Це можуть бути як однакові, так і зовсім інші операційні системи. Ваш вибір, що б ви не робили з ними. Чим більше оперативної пам'яті, швидкості обробки та зберігання даних у вашій хост-машини, тим багатьма екземплярами різних віртуальних машин ви можете запустити. Як би це просто не було.

Для встановлення HoneyPot на Linux, потрібно встановити програмне забезпечення PenTBox.

PenTBox - це набір для забезпечення безпеки, що містить різні інструменти для оптимізації роботи PenTest. Він запрограмований на Ruby і орієнтований на GNU / Linux, з підтримкою Windows, MacOS та всіх систем, де встановлений Ruby.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		36

3.2 Налаштування роботи мережі та підсистеми Security Deception

Перш за все у даній мережі було налаштовано DHCP на маршрутизаторі, для того, щоб користувачі відразу отримували свою ір – адресу, нам потрібно провести налагодження функціонування DHCP-сервера на маршрутизаторі. Розглянемо сценарій налагодження:

```
Router(config)#ip dhcp pool LAN1
Router(DHCP-config)#network 208.3.17.0 255.255.255.0
Router(DHCP-config)#default-router 208.3.17.1
Router(DHCP-config)#dns-server 208.3.17.1
Router(DHCP-config)#exit
Router(config)#ip dhcp pool LAN2
Router(DHCP-config)#network 209.3.17.0 255.255.255.0
Router(DHCP-config)#default-router 209.3.17.1
Router(DHCP-config)#dns-server 209.3.17.1
Router(DHCP-config)#ex
Router(config)#ip dhcp pool LAN3
Router(DHCP-config)#network 210.3.17.0 255.255.255.0
Router(DHCP-config)#default-router 210.3.17.1 255.255.255.0
Router(DHCP-config)#dns-server 210.3.17.1
Router(DHCP-config)#ex
Router(config)#ip dhcp excluded-address 208.3.17.1 208.3.17.2
Router(config)#ip dhcp excluded-address 209.3.17.1 209.3.17.2
Router(config)#ip dhcp excluded-address 210.3.17.1 210.3.17.2
```

Далі налаштовуємо на маршрутизаторі ір-адреси до комутаторів.

```
Router(config)#int f1/0
Router(config-if)#description LAN_A
Router(config-if)#ip add 208.3.17.1 255.255.255.0
Router(config-if)#no sh
Router(config-if)#ex
Router(config)#int f2/0
Router(config-if)#description LAN_B
Router(config-if)#ip add 209.3.17.1 255.255.255.0
Router(config-if)#no sh
Router(config-if)#ex
Router(config)#int f
Router(config)#int f3/0
Router(config-if)#description LAN_C
Router(config-if)#ip add 210.3.17.1 255.255.255.0
Router(config-if)#no sh
```

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				37
Змн.	Арк.	№ докум.	Підпис	Дата		

Router(config-if)#ex

Налаштовуємо ір-адреси на комутаторах мережі.

SW_3_17_1(config)#int vlan 1

SW_3_17_1(config-if)#ip add 208.3.17.2 255.255.255.0

SW_3_17_1(config-if)#no sh

SW_3_17_1(config-if)#ex

SW_3_17_1(config)#int vlan 1

SW_3_17_1(config-if)#ip default-gateway 208.3.17.1

SW_3_17_1(config)#ip domain name LAN1

SW_3_17_1(config)#no ip domain-lookup

Перейдемо до налаштування підсистеми Security Deception на ОС Linux.

Для цього заходимо в консоль з root правами та встановлюємо PenTBBox (Додаток В). Після встановлення, запускаємо скрипт за допомогою команди ./PenTBBox.rb та обираємо спочатку Network Tools, що ми можемо побачити на рисунку 3.3, а потім HoneyPot.

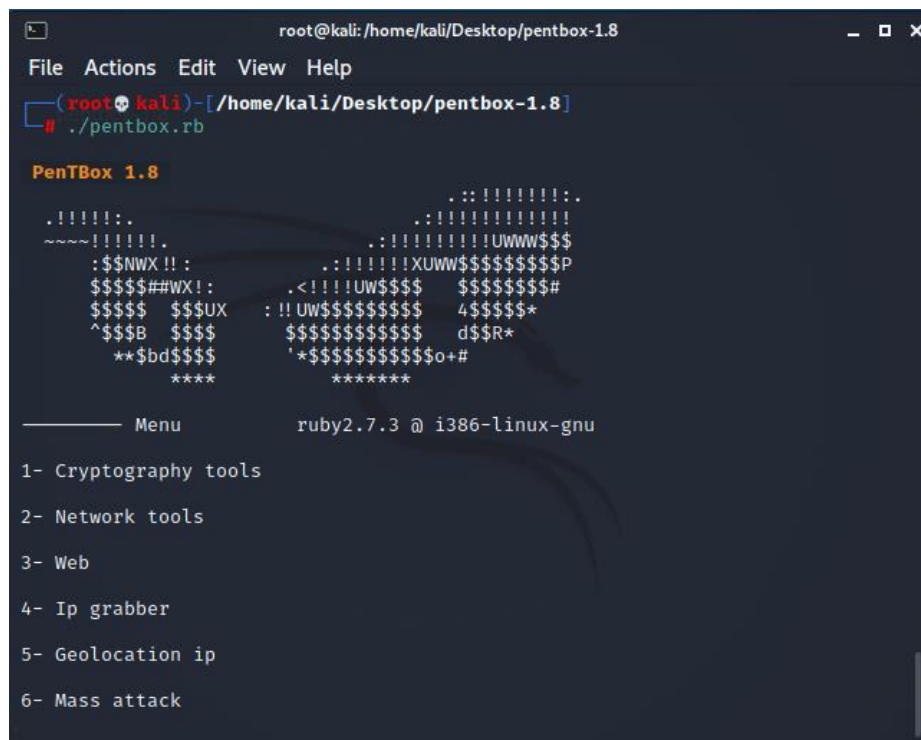


Рисунок 3.3 – Встановлюємо PenTBBox

В наступному меню обираємо пункт 1, для швидкого автоматичного налаштування. Це запустить HoneyPot, який прослуховує 80 порт.

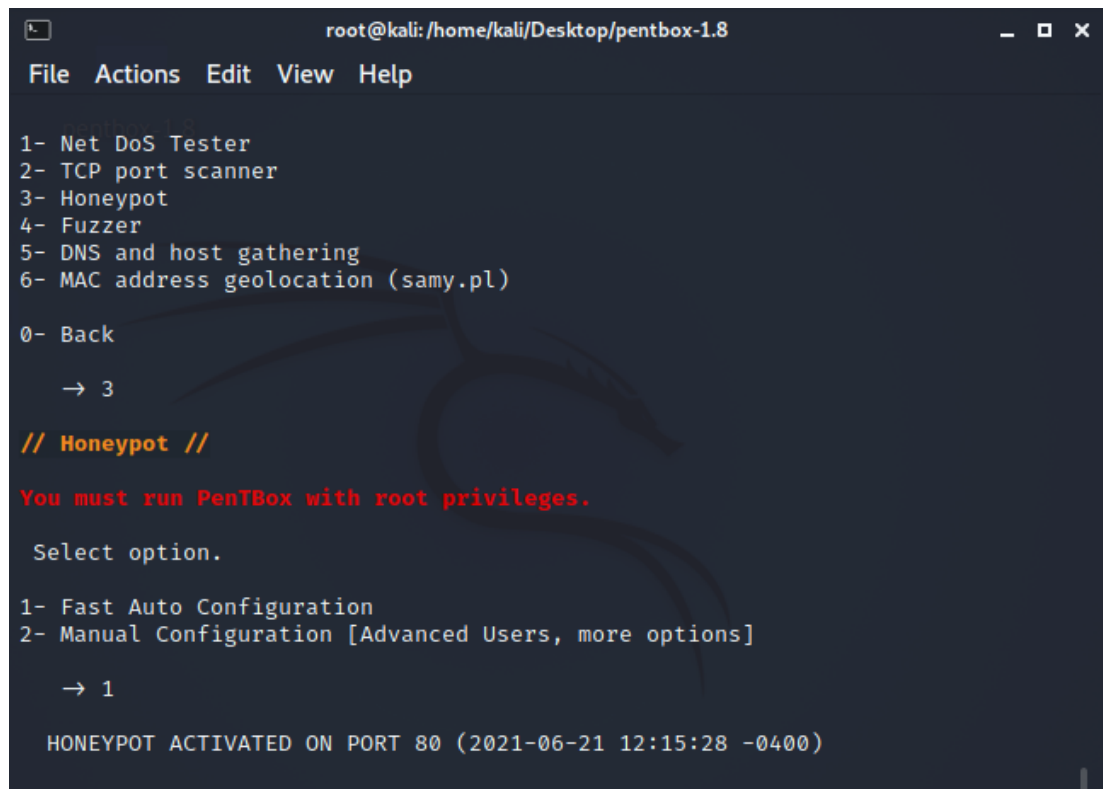


Рисунок 3.4 – Обираємо систему захисту HoneyPot

На сервері встановлюємо HoneyBot та розглянемо його налаштування на рисунку 3.5.

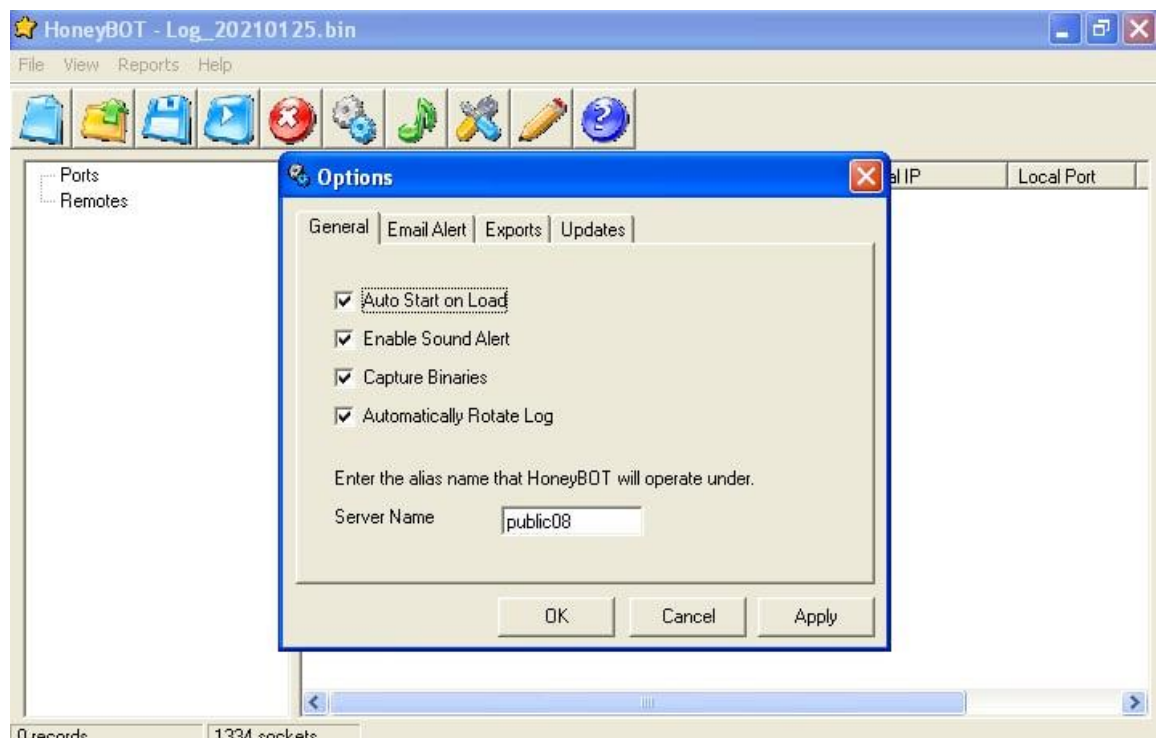


Рисунок 3.6. Налаштування HoneyBot

```

(kali@kali)~$ git clone https://github.com/skeeto/endlesssh.git
Cloning into 'endlesssh'...
remote: Enumerating objects: 346, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 346 (delta 3), reused 9 (delta 2), pack-reused 334
Receiving objects: 100% (346/346), 81.02 KiB | 7.37 MiB/s, done.
Resolving deltas: 100% (208/208), done.

(kali@kali)~$ cd endlesssh/

```

Рисунок 3.7 – Підключення до SSH-серверу

```

(root@kali)~/Desktop$ sudo apt update
Get:1 http://kali.koyanet.lv/kali kali-rolling InRelease [30.5 kB]
Get:2 http://kali.koyanet.lv/kali kali-rolling/main i386 Packages [17.6 MB]
Get:3 http://kali.koyanet.lv/kali kali-rolling/main i386 Contents (deb) [39.3 MB]
Get:4 http://kali.koyanet.lv/kali kali-rolling/contrib i386 Packages [97.6 kB]
Get:5 http://kali.koyanet.lv/kali kali-rolling/contrib i386 Contents (deb) [98.2 kB]
Get:6 http://kali.koyanet.lv/kali kali-rolling/non-free i386 Packages [168 kB]
Get:7 http://kali.koyanet.lv/kali kali-rolling/non-free i386 Contents (deb) [908 kB]
Fetched 58.2 MB in 23s (2,548 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
155 packages can be upgraded. Run 'apt list --upgradable' to see them.

(root@kali)~/Desktop$ sudo apt install make
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
make is already the newest version (4.3-4.1).
make set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 155 not upgraded.

```

Рисунок 3.8 – Завантаження пакетів make та gcc

```

(root@kali)~/endlesssh$ apt install make gcc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gcc is already the newest version (4:10.2.1-1).
gcc set to manually installed.
make is already the newest version (4.3-4.1).
make set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 15 not upgraded.

(root@kali)~/endlesssh$ make
cc -std=c99 -Wall -Wextra -Wno-missing-field-initializers -Os -ggdb3 -o endl
essh endlesssh.c

(root@kali)~/endlesssh$ ./endlesssh 6
[1] 1128

```

Рисунок 3.9 – Виконання команди make та запуск скомпільованого файлу

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				40
Змн.	Арк.	№ докум.	Підпис	Дата		

На рисунку 3.9 ми можемо побачити результат виконання команди `ssh -v -p2222 localhost`, де `-v` вмикає докладний режим, а `-p2222` вказує на порт по якому буде здійснюватися підключення.

```

root@kali: /home/kali/endlessh
File Actions Edit View Help
(root@kali)~[/home/kali/endlessh]
# ssh -p22 localhost 130 x 1
ssh: connect to host localhost port 22: Connection refused

(root@kali)~[/home/kali/endlessh]
# ssh -v -p2222 localhost 255 x 1
OpenSSH_8.4p1 Debian-5, OpenSSL 1.1.1k 25 Mar 2021
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: /etc/ssh/ssh_config line 19: include /etc/ssh/ssh_config.d/*.conf mat
ched no files
debug1: /etc/ssh/ssh_config line 21: Applying options for *
debug1: Connecting to localhost [::1] port 2222.
debug1: Connection established.
debug1: identity file /root/.ssh/id_rsa type -1
debug1: identity file /root/.ssh/id_rsa-cert type -1
debug1: identity file /root/.ssh/id_dsa type -1
debug1: identity file /root/.ssh/id_dsa-cert type -1
debug1: identity file /root/.ssh/id_ecdsa type -1
debug1: identity file /root/.ssh/id_ecdsa-cert type -1
debug1: identity file /root/.ssh/id_ecdsa_sk type -1
debug1: identity file /root/.ssh/id_ecdsa_sk-cert type -1
debug1: identity file /root/.ssh/id_ed25519 type -1
debug1: identity file /root/.ssh/id_ed25519-cert type -1
debug1: identity file /root/.ssh/id_ed25519_sk type -1
debug1: identity file /root/.ssh/id_ed25519_sk-cert type -1
debug1: identity file /root/.ssh/id_xmss type -1
debug1: identity file /root/.ssh/id_xmss-cert type -1

```

Рисунок 3.9 – Підключення до SSH-серверу через 2222 порт

На рисунку 3.10 ми замінюємо порт для підключення на інший.

```

root@kali: /home/kali/endlessh
File Actions Edit View Help
GNU nano 5.4 /etc/ssh/sshd_config *
# $OpenBSD: sshd_config,v 1.103 2018/04/09 20:41:22 tj Exp $

# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 2244
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify

```

Рисунок 3.10 – Порт справжнього SSH з'єднання 2244

3.3 Тестування роботи мережі та підсистеми Security Deception

Для перевірки працездатності нашої мережі потрібно виконати такі кроки:

1. Перевірити налаштування DHCP на маршрутизаторі;
2. Отримати DHCP адреси на різних пристроях (VPCS, Linux , Windows);
3. Перевірити зв'язок між кінцевими пристроями мережі;
4. Під'єднатися з атакуючого Kali Linux до Linux з HoneyPot та перевірити логи в HoneyPot;
5. Спробувати під'єднатися до HoneyPot Linux через протокол SSH, переконатися в тому що тарпїт працює справно та вводить в оману зловмисника;
6. Перевірити працездатність HoneyPot на Windows-сервері, перевірити створення фейкових портів, спробувати під'єднатися до них з атакуючого Kali Linux та перевірити логи.

Для перевірки працездатності протоколу DHCP, скористаємо вбудованою командою на маршрутизаторі. Переглянемо налаштування на рисунку 3.6.

```
Router#show ip dhcp pool

Pool LAN1 :
Utilization mark (high/low)    : 100 / 0
Subnet size (first/next)       : 0 / 0
Total addresses                 : 254
Leased addresses                : 3
Pending event                   : none
1 subnet is currently in the pool :
Current index   IP address range      Leased addresses
208.3.17.7     208.3.17.1 - 208.3.17.254      3

Pool LAN2 :
Utilization mark (high/low)    : 100 / 0
Subnet size (first/next)       : 0 / 0
Total addresses                 : 254
Leased addresses                : 1
Pending event                   : none
1 subnet is currently in the pool :
Current index   IP address range      Leased addresses
209.3.17.4     209.3.17.1 - 209.3.17.254      1
```

Рисунок 3.11 – Виконання команди show ip dhcp pool

Увімкнемо DHCP протокол на усіх кінцевих вузлах. Переглянемо налагодження на рисунках 3.12 – 3.13.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ док.ум.	Підпис	Дата		42

WS_3_17_1

```
WS_3_17_1> dhcp  
DDORRA IP 208.3.17.3/24 GW 208.3.17.1
```

Рисунок 3.12 – Успішна активація DHCP на WS_3_17_1

WS_3_17_2

```
WS_3_17_2> dhcp  
DDORRA IP 209.3.17.3/24 GW 209.3.17.1
```

Рисунок 3.13 – Успішна активація DHCP на WS_3_17_1

```
(root@kali)-[/home/kali/Desktop]  
# dhclient -v eth0  
Internet Systems Consortium DHCP Client 4.4.1  
Copyright 2004-2018 Internet Systems Consortium.  
All rights reserved.  
For info, please visit https://www.isc.org/software/dhcp/  
  
Listening on LPF/eth0/08:00:27:2c:95:14  
Sending on LPF/eth0/08:00:27:2c:95:14  
Sending on Socket/fallback  
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 6  
DHCPOFFER of 208.3.17.6 from 208.3.17.1  
DHCPREQUEST for 208.3.17.6 on eth0 to 255.255.255.255 port 67  
DHCPACK of 208.3.17.6 from 208.3.17.1  
bound to 208.3.17.6 -- renewal in 42871 seconds.
```

Рисунок 3.14 – Успішна активація DHCP на Kali Linux

Після успішного отримання IP-адрес на пристроях мережі, перевіримо їх зв'язок один з одним.

```
(root@kali)-[/home/kali/Desktop]  
# ping 209.3.17.3  
PING 209.3.17.3 (209.3.17.3) 56(84) bytes of data. 255 x  
64 bytes from 209.3.17.3: icmp_seq=1 ttl=63 time=3010 ms  
64 bytes from 209.3.17.3: icmp_seq=2 ttl=63 time=2001 ms  
64 bytes from 209.3.17.3: icmp_seq=3 ttl=63 time=977 ms  
64 bytes from 209.3.17.3: icmp_seq=4 ttl=63 time=20.2 ms  
64 bytes from 209.3.17.3: icmp_seq=5 ttl=63 time=17.3 ms  
64 bytes from 209.3.17.3: icmp_seq=6 ttl=63 time=16.6 ms  
^C  
--- 209.3.17.3 ping statistics ---  
6 packets transmitted, 6 received, 0% packet loss, time 5037ms  
rtt min/avg/max/mdev = 16.640/1007.058/3010.263/1150.111 ms, pipe 3
```

Рисунок 3.15 – Успішно встановлений зв'язок між пристроями на кінцевих вузлах

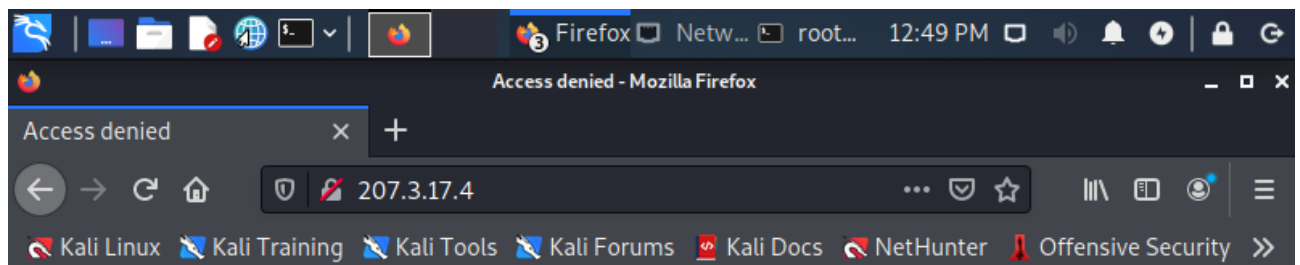
		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				43
Змн.	Арк.	№ докум.	Підпис	Дата		

За допомогою вбудованою команди `show ip dhcp binding`, ми можемо переглянути список всіх IP адрес і зіставлених з ними MAC-адрес, які були видані DHCP сервером. Переглянемо виконання цієї команди на рисунку 3.16.

```
Router#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/      Lease expiration      Type
                Hardware address/
                User name
207.3.17.3      0108.0027.2c95.14    Jun 22 2021 07:09 PM    Automatic
207.3.17.4      0800.272c.9514       Jun 22 2021 07:10 PM    Automatic
208.3.17.4      0108.0027.2c95.14    Jun 22 2021 07:03 PM    Automatic
208.3.17.5      0100.5079.6668.00    Jun 22 2021 07:02 PM    Automatic
208.3.17.6      0800.272c.9514       Jun 22 2021 07:05 PM    Automatic
209.3.17.3      0100.5079.6668.01    Jun 22 2021 07:02 PM    Automatic
209.3.17.4      0100.5079.6668.03    Jun 22 2021 08:52 PM    Automatic
210.3.17.3      0100.5079.6668.04    Jun 22 2021 08:52 PM    Automatic
210.3.17.4      0100.5079.6668.02    Jun 22 2021 08:52 PM    Automatic
```

Рисунок 3.16 – Виконання команди `show ip dhcp binding` на маршрутизаторі

Для перевірки роботи програми HoneyPot на Linux, підключимося до неї з пристрою зломисника та вкажемо у веб-браузері його IP-адресу. Після вводу ми бачимо на рисунку 3.17, що доступ відхилено.



Access denied

IP Address login failed

2021-06-21 12:15:28 -0400

Рисунок 3.17 – Підключення до пристрою з встановленим HoneyPot

Тепер повернемося до нашого HoneyPot, де ми можемо побачити, що була виявлена спроба підключення.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		44

```

root@kali: /home/kali/Desktop/pentbox-1.8
File Actions Edit View Help
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;
q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache

INTRUSION ATTEMPT DETECTED! from 208.3.17.4:40874 (2021-06-21 12:17:59 -040
0)

GET /favicon.ico HTTP/1.1
Host: 207.3.17.4
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:78.0) Gecko/20100101 Firefox/78.
0
Accept: image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Pragma: no-cache
Cache-Control: no-cache

```

Рисунок 3.18 – Спроба підключення зломисника до HoneyPot

```

root@kali: /home/kali/endlssh
File Actions Edit View Help
OpenSSH_8.4p1 Debian-5, OpenSSL 1.1.1k 25 Mar 2021
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: /etc/ssh/ssh_config line 19: include /etc/ssh/ssh_config.d/*.conf mat
ched no files
debug1: /etc/ssh/ssh_config line 21: Applying options for *
debug1: Connecting to localhost [::1] port 2222.
debug1: Connection established.
debug1: identity file /root/.ssh/id_rsa type -1
debug1: identity file /root/.ssh/id_rsa-cert type -1
debug1: identity file /root/.ssh/id_dsa type -1
debug1: identity file /root/.ssh/id_dsa-cert type -1
debug1: identity file /root/.ssh/id_ecdsa type -1
debug1: identity file /root/.ssh/id_ecdsa-cert type -1
debug1: identity file /root/.ssh/id_ecdsa_sk type -1
debug1: identity file /root/.ssh/id_ecdsa_sk-cert type -1
debug1: identity file /root/.ssh/id_ed25519 type -1
debug1: identity file /root/.ssh/id_ed25519-cert type -1
debug1: identity file /root/.ssh/id_ed25519_sk type -1
debug1: identity file /root/.ssh/id_ed25519_sk-cert type -1
debug1: identity file /root/.ssh/id_xmss type -1
debug1: identity file /root/.ssh/id_xmss-cert type -1
debug1: Local version string SSH-2.0-OpenSSH_8.4p1 Debian-5
debug1: kex_exchange_identification: banner line 0: d
debug1: kex_exchange_identification: banner line 1: dMk`PN^utD?
debug1: kex_exchange_identification: banner line 2: po/Lsh7j"85Bnmg% D
debug1: kex_exchange_identification: banner line 3: 1Nj$p{g=%9c

```

Рисунок 3.19 – Після підключення SSH надсилає набір довільних символів

Скануємо сервер з вимкненим HoneyBot на Kali Linux. Розглянемо сканування на рисунку 3.20.

```
root@kali:~# nmap 207.3.17.2
Starting Nmap 7.80 ( https://nmap.org ) at 2021-01-25 13:19 EST
Nmap scan report for 207.3.17.2
Host is up (0.053s latency).
Not shown: 989 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
5357/tcp  open  wsddapi
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49157/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 17.76 seconds
```

Рисунок 3.20 – сканування сервера з вимкненим HoneyBot

Знову скануємо наш сервер з Kali Linux. Розглянемо сканування на рисунках 3.21-3.22.

```
root@kali:~# nmap 207.3.17.2
Starting Nmap 7.80 ( https://nmap.org ) at 2021-01-25 13:18 EST
Nmap scan report for 207.3.17.2
Host is up (0.050s latency).
Not shown: 745 closed ports
PORT      STATE SERVICE
1/tcp     open  tcpmux
3/tcp     open  compressnet
4/tcp     open  unknown
6/tcp     open  unknown
7/tcp     open  echo
9/tcp     open  discard
13/tcp    open  daytime
17/tcp    open  qotd
19/tcp    open  chargen
20/tcp    open  ftp-data
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
24/tcp    open  priv-mail
25/tcp    open  smtp
33/tcp    open  dsp
37/tcp    open  time
42/tcp    open  nameserver
43/tcp    open  whois
49/tcp    open  tacacs
53/tcp    open  domain
70/tcp    open  gopher
79/tcp    open  finger
```

Рисунок 3.21 – Сканування серверу з увімкненим HoneyBot

```

8000/tcp open http-alt
8001/tcp open vcom-tunnel
8008/tcp open http
8010/tcp open xmpp
8080/tcp open http-proxy
8081/tcp open blackice-icecap
8082/tcp open blackice-alerts
8443/tcp open https-alt
8888/tcp open sun-answerbook
9000/tcp open cslistener
9080/tcp open glrpc
9090/tcp open zeus-admin
9100/tcp open jetdirect
9415/tcp open unknown
9535/tcp open man
9876/tcp open sd
9898/tcp open monkeycom
10000/tcp open snet-sensor-mgmt
10082/tcp open amandaidx
11111/tcp open vce
12174/tcp open unknown
12345/tcp open netbus
13722/tcp open netbackup
20000/tcp open dnp
20005/tcp open btx
20031/tcp open unknown
31337/tcp open Elite
49152/tcp open unknown
49153/tcp open unknown
49154/tcp open unknown
49155/tcp open unknown
49156/tcp open unknown
49157/tcp open unknown
65000/tcp open unknown

```

Рисунок 3.22 – Сканування серверу з увімкненим HoneyBot

Атакуюча машина виконує сканування пптар, в результаті якого видно багато фальшивих сервісів. Все це завдяки наявності HoneyPot в системі.

HoneyBot створив фальшивий порт SSH та Telnet. Ініціюємо SSH та Telnet з'єднання через 22-23 порт до нашого сервера.

Розглянемо логи після SSH з'єднання в HoneyBot на рисунку 3.23.

HoneyBOT - Log_20210125.bin								
File View Reports Help								
Ports	Date	Time	Remote IP	Remote Port	Local IP	Local Port	Protocol	Bytes
22	25.01.2021	21:14:02	209.3.17.6	1076	207.3.17.2	22	TCP	28
Remotes	25.01.2021	21:35:52	209.3.17.5	52184	207.3.17.2	22	TCP	32
209.3.17.6								
209.3.17.5								

Рисунок 3.23 – Історія підключення до сервера з Kali Linux та Windows XP

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				47
Змн.	Арк.	№ докум.	Підпис	Дата		

На рисунках 3.24-3.25 ми можемо побачити докладний звіт про підключення, створене зловмисником.

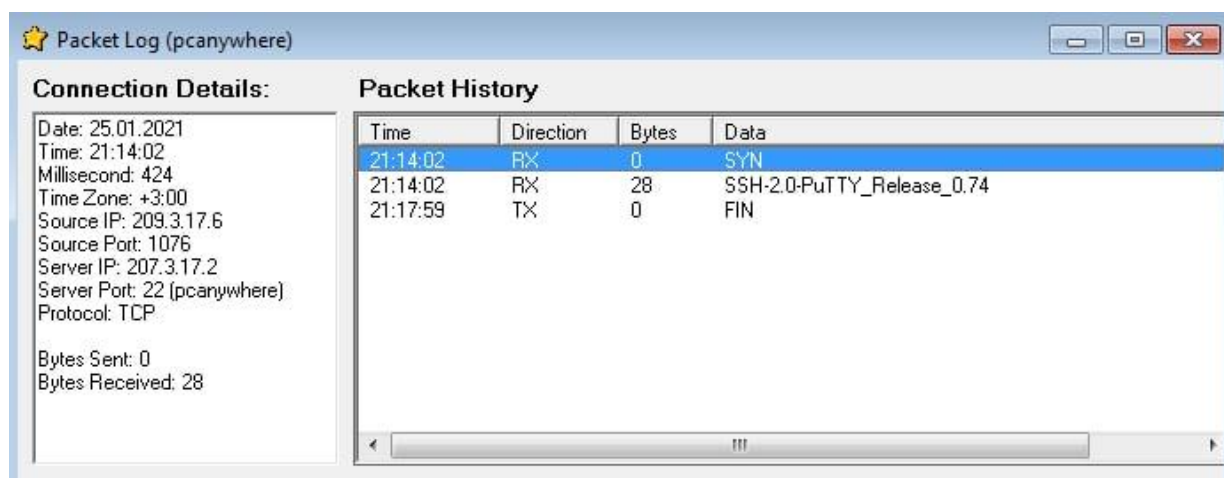


Рисунок 3.24 – Детальна інформація SSH підключення з Windows

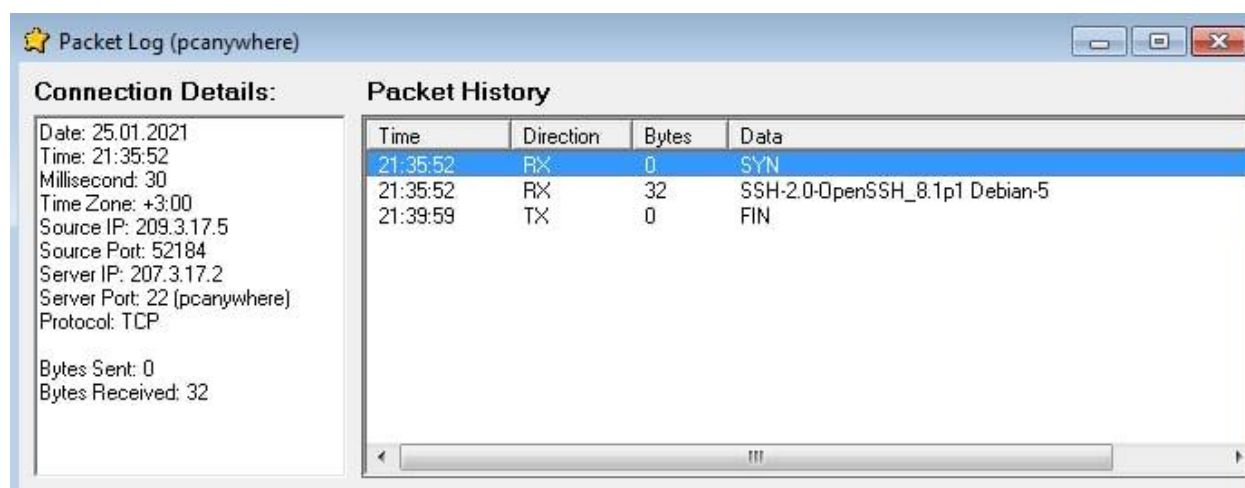


Рисунок 3.25. Детальна інформація SSH підключення з Kali Linux

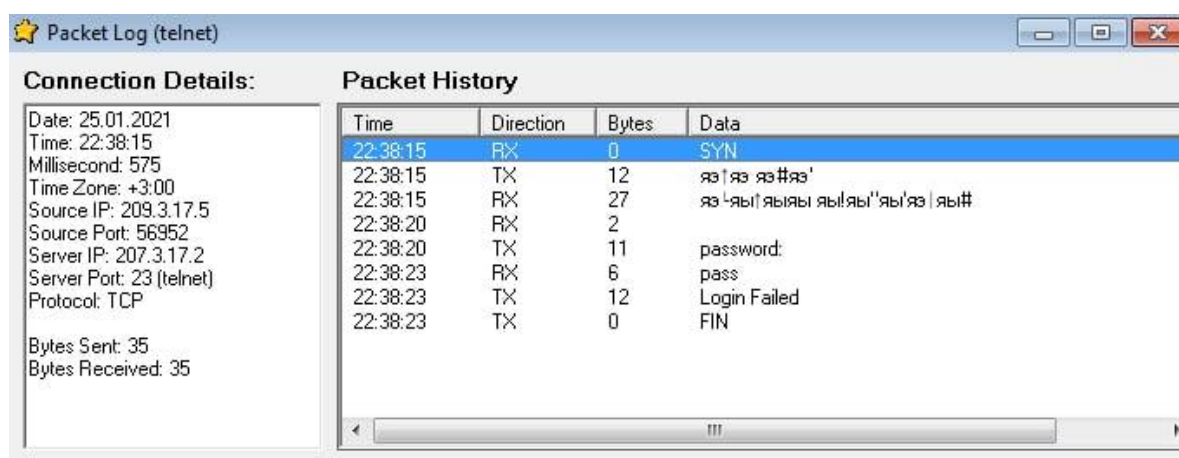


Рисунок 3.26 – Детальна інформація telnet підключення з Kali Linux

З'являється лог з IP-адресою зломисника і портом, до якого він був підключений. Заходимо анонімно на FTP сервер з Kali Linux та перевіряємо логи на HoneyBot.

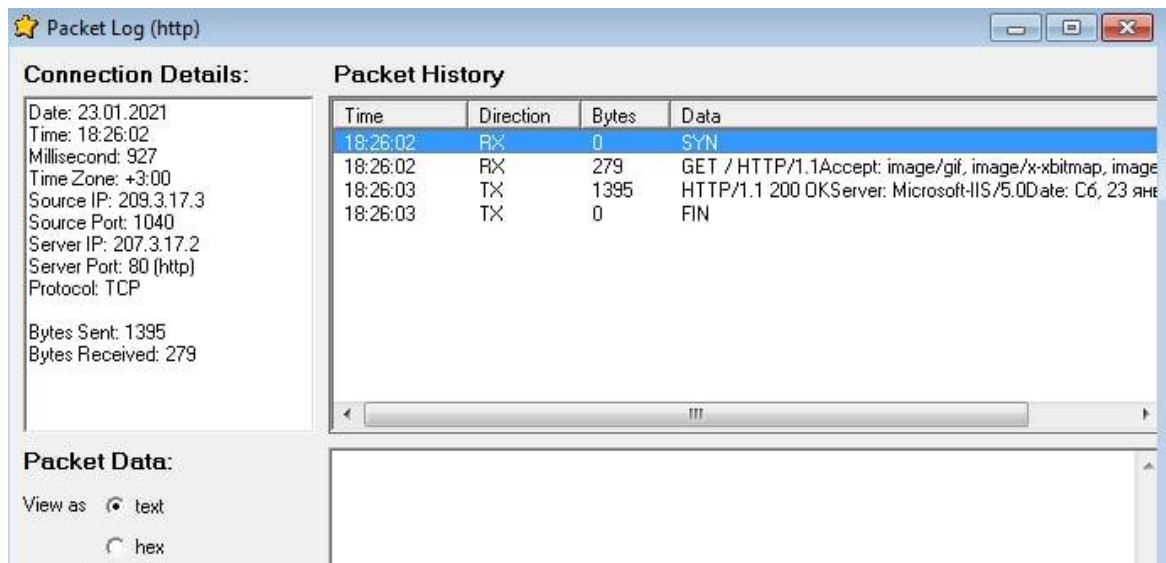


Рисунок 3.27 – Детальна інформація FTP підключення з Kali Linux

Висновки до розділу 3

В даному розділі було проведено обґрунтування вибору обладнання та програмного забезпечення для побудови фізичної мережної інфраструктури організації. Показано налаштування мережі, а саме налаштування протоколу DHCP. Також показано налаштування підсистеми Security Desception, в образі HoneyPot та порт-пастки SSH tarpit. Налаштували FTP сервер. Виконали атаки на сервер та на кінцеві вузли мережі. Таким чином було проведено тестування працездатності мережі та підсистеми Security Desception.

ВИСНОВКИ

У даній кваліфікаційній роботі було досягнуто поставлені задачі через дослідження, проектування та розробку комп'ютерної мережі та підсистеми Security Deception.

Після розгляду теоретичного матеріалу з безпеки було визначено основні системи захисту комп'ютерних мереж. Серед них такі, як: HoneyPot, SandBox та Security Deception. В процесі розгляду було визначено головні принципи та технології побудови цих систем. Кожна з цих систем застосовує підхід з приналежністю зловмисника чи злочинної програми в ізольоване середовище, де в подальшому аналізують їх поведінку. За допомогою цього дослідження, фахівці в області інформаційної безпеки розробляють підхід стратегії відбиття атак зловмисників в майбутньому. Також було проведено порівняльний аналіз цих технологій захисту комп'ютерних мереж. В результаті аналізу було виявлено переваги та недоліки кожної системи, які в свою чергу дозволили обрати відповідну технологію захисту мережі для нашого проекту.

Було проведено проектування фізичної мережної інфраструктури організації. Також було розглянуто такі поняття, як сервер, комп'ютерна мережа, її основні типи та топологія. Описано використовувані протоколи, DHCP та SSH. Проведено проектування логічних зв'язків мережі та розробка схеми адресації. Показано принцип роботи підсистеми Security Deception на прикладі програми HoneyBot. Розглянуто функціонал ОС Kali Linux та обрано його атакуючою машиною. Також було розглянуто порт-пастку SSH tarpit.

Було приведено обґрунтування вибору обладнання та програмного забезпечення для побудови фізичної мережної інфраструктури організації. Показано налаштування мережі, а саме налаштування протоколу DHCP. Також показано налаштування підсистеми Security Deception, в образі HoneyPot та порт-пастки SSH tarpit. Виконали атаки на сервер та на кінцеві вузли мережі. Таким чином було проведено тестування працездатності мережі та підсистеми Security Deception.

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				50
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Дж.Бони. Руководство по CiscoIOS. – Издательство "Русская редакция", 2008. – 784 с
2. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 3-е изд. – СПб.: Питер, 2005. – 958 с.
3. Красов А.В., Масштабируемое HoneyPot-решение для обеспечения безопасности в корпоративных сетях, 2019. – 87с.
4. Олифер В.Г., Олифер Н.А. Сетевые операционные системы. – СПб.: Питер, 2002. – 544 с.
5. Al-Shaer E. Autonomous Cyber Deception: Reasoning, Adaptive Planning, and Evaluation of HoneyThings / E. Al-Shaer, J. Wei, K. Hamlen., 2019. – 239 с.
6. Rowe N. Introduction to Cyberdeception / N. Rowe, J. Rrushi., 2016. – 339 с.
7. Mitnick K. The Art of Deception: Controlling the Human Element of Security / K. Mitnick, W. Simon., 2003. – 577 с.
8. Amoz S. Surviving The Cyber World / Syed Amoz., 2018. – 155 с.
9. Deniz A. Honeypots in Network Security / Akkaya Deniz., 2012. – 64 с.
10. Mohssen M. Honeypots and Routers: Collecting Internet Attacks / M. Mohssen, R. Habib-ur., 2016. – 197 с.
11. Bodmer S. Reverse Deception: Organized Cyber Threat Counter-Exploitation / S. Bodmer, G. Carpenter., 2012. – 352 с.
12. HoneyPot [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/HoneyPot>
13. Подробное руководство по HoneyPot [Електронний ресурс]. Режим доступу до ресурсу: <https://habr.com/ru/company/alexhost/blog/528796/>

		Поліщук К.В.			<i>КІК.КР.Б-123-17-ПЗ</i>	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		51

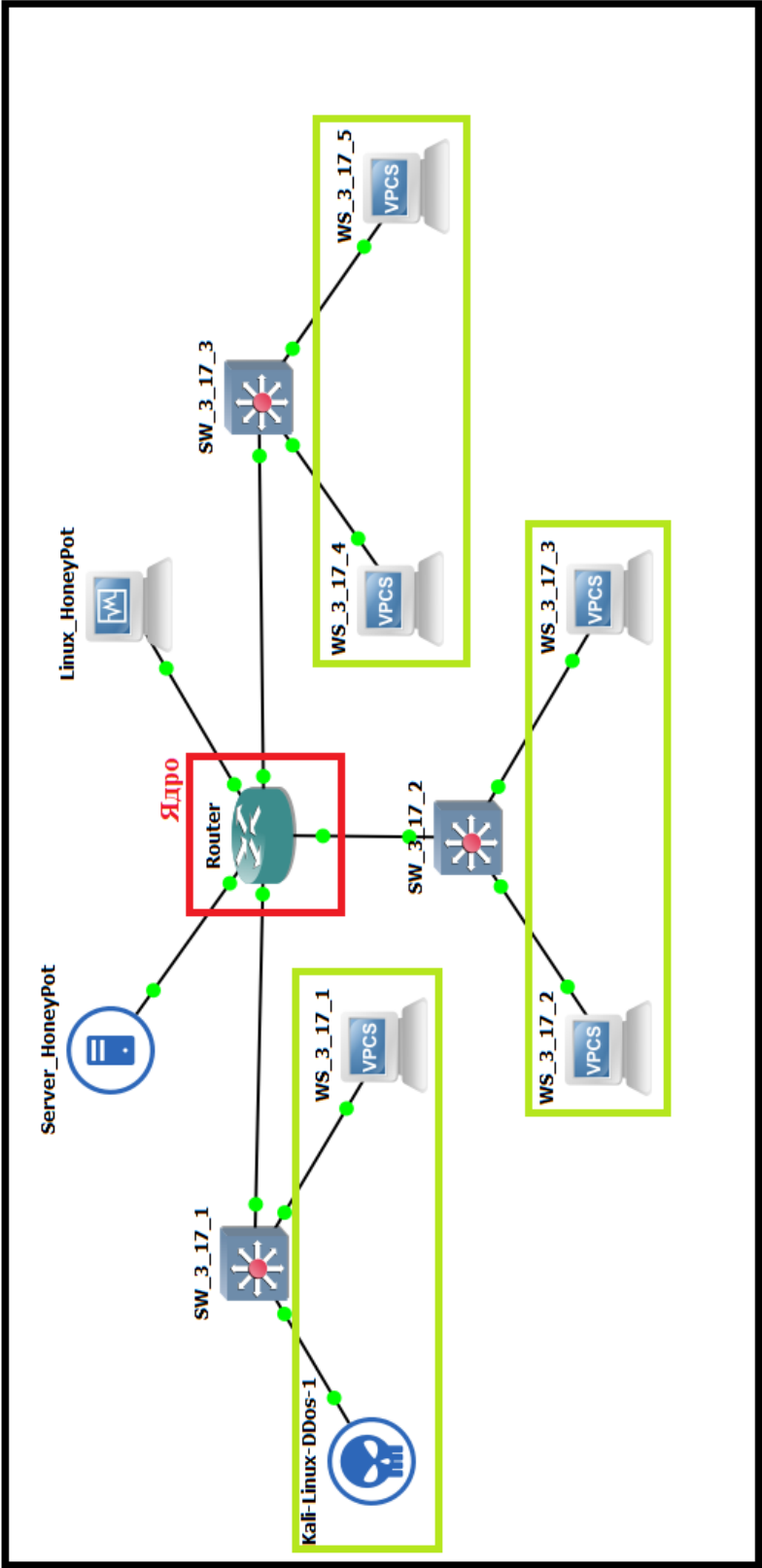
14. Network topologies, protocols and layers [Электронный ресурс] // BBC – Режим доступа до ресурсу: <https://www.bbc.co.uk/bitesize/guides/zr3yb82/revision/1>
15. What is DHCP [Электронный ресурс] – Режим доступа до ресурсу: <https://www.faqforge.com/windows/what-is-DHCP/>
16. Ловушка (тарпит) для входящих SSH-соединений [Электронный ресурс]. – 2019. – Режим доступа до ресурсу: <https://habr.com/ru/company/globalsign/blog/445318/>
17. How to establish a HoneyPot on your network [Электронный ресурс]. – 2021. – Режим доступа до ресурсу: <https://www.comparitech.com/net-admin/how-to-establish-a-HoneyPot-on-your-network/>
18. Смена порта SSH-сервера как мера защиты от брутфорса [Электронный ресурс] – Режим доступа до ресурсу: <https://putty.org.ru/articles/change-default-sshd-port.html>
19. HoneyPot vs Deception на примере Xello [Электронный ресурс]. – 2020. – Режим доступа до ресурсу: <https://habr.com/ru/company/tssolution/blog/522374/>
20. What is dynamic routing? [Электронный ресурс] – Режим доступа до ресурсу: <https://www.educative.io/edpresso/what-is-dynamic-routing>
21. What is GNS3? [Электронный ресурс]. – 2014. – Режим доступа до ресурсу: <http://www.electricmonk.org.uk/2014/02/07/what-is-gns3/>
22. What is SandBox Security? [Электронный ресурс] – Режим доступа до ресурсу: <https://www.forcepoint.com/cyber-edu/SandBox-security>
23. What is Deception Technology? [Электронный ресурс] – Режим доступа до ресурсу: <https://www.forcepoint.com/cyber-edu/deception-technology>

ДОДАТКИ

		Поліщук К.В.			КІК.КР.Б-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		53

Проект мережі

Рівень розподілу даних



Файли конфігурацій мережних пристроїв

Файл конфігурації маршрутизатора R_3_17_1

hostname Router ! boot-start-marker boot-end-marker no aaa new-model ip source-route no ip icmp rate-limit unreachable ip cef !! ip dhcp excluded-address 208.3.17.1 208.3.17.2 ip dhcp excluded-address 209.3.17.1 209.3.17.2 ip dhcp excluded-address 210.3.17.1 210.3.17.2 ip dhcp excluded-address 207.3.17.1 207.3.17.2 ! ip dhcp pool LAN1 network 208.3.17.0 255.255.255.0 default-router 208.3.17.1 dns-server 208.3.17.1 ! ip dhcp pool LAN2 network 209.3.17.0 255.255.255.0 default-router 209.3.17.1 dns-server 209.3.17.1 ! ip dhcp pool LAN3 network 210.3.17.0 255.255.255.0 default-router 210.3.17.1 255.255.255.0 dns-server 210.3.17.1 ! ip dhcp pool LANServer network 207.3.17.0 255.255.255.0 default-router 207.3.17.1 255.255.255.0 dns-server 207.3.17.1 !!	no ip domain lookup no ipv6 cef !! interface FastEthernet0/0 description LANServer ip address 207.3.17.1 255.255.255.0 duplex half ! interface FastEthernet1/0 description LAN_A ip address 208.3.17.1 255.255.255.0 duplex half ! interface FastEthernet2/0 description LAN_B ip address 209.3.17.1 255.255.255.0 duplex half ! interface FastEthernet3/0 description LAN_C ip address 210.3.17.1 255.255.255.0 duplex half ! Router#show ip dhcp binding Bindings from all pools not associated with VRF: IP address Client-ID/ Lease expiration Type Hardware address/ User name 207.3.17.3 0108.0027.2c95.14 Jun 22 2021 10:22 PM Automatic 208.3.17.3 0100.5079.6668.00 Jun 22 2021 11:38 PM Automatic 208.3.17.4 0108.0027.2c95.14 Jun 22 2021 10:22 PM Automatic 209.3.17.3 0100.5079.6668.01 Jun 22 2021 11:38 PM Automatic 209.3.17.4 0100.5079.6668.03 Jun 22 2021 11:38 PM Automatic 210.3.17.3 0100.5079.6668.04 Jun 22 2021 11:38 PM Automatic
---	--

		Поліщук К.В.			KIK.KP.B-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		55

Файл конфігурації комутатора SW_3_17_1

hostname SW_3_17_1 ! boot-start-marker boot-end-marker !! logging discriminator EXCESS severity drops 6 msg-body drops EXCESSCOLL logging buffered 50000 logging console discriminator EXCESS ! no aaa new-model no ip icmp rate-limit unreachable ! ip cef !! no ip domain-lookup ip domain-name LAN1 no ipv6 cef ipv6 multicast rpf use-bgp !! spanning-tree mode pvst spanning-tree extend system-id ! vlan internal allocation policy ascending ! ip tcp synwait-time 5 !! interface Ethernet0/0 duplex auto	interface Ethernet0/1 duplex auto ! interface Ethernet0/2 duplex auto ! interface Ethernet0/3 duplex auto ! interface Vlan1 ip address 208.3.17.2 255.255.255.0 ! ip default-gateway 208.3.17.1 ! no ip http server !! control-plane !! line con 0 exec-timeout 0 0 privilege level 15 logging synchronous line aux 0 exec-timeout 0 0 privilege level 15 logging synchronous line vty 0 4 login ! end
--	---

		Поліщук К.В.			KIK.KP.B-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		56

Файл конфігурації комутатора SW_3_17_2

hostname SW_3_17_2 ! boot-start-marker boot-end-marker !! logging discriminator EXCESS severity drops 6 msg-body drops EXCESSCOLL logging buffered 50000 logging console discriminator EXCESS ! no aaa new-model no ip icmp rate-limit unreachable ! ip cef !! no ip domain-lookup ip domain-name LAN2 no ipv6 cef ipv6 multicast rpf use-bgp !! spanning-tree mode pvst spanning-tree extend system-id !! vlan internal allocation policy ascending ! ip tcp synwait-time 5 !! interface Ethernet0/0 duplex auto	interface Ethernet0/1 duplex auto ! interface Ethernet0/2 duplex auto ! interface Ethernet0/3 duplex auto ! interface Vlan1 ip address 209.3.17.2 255.255.255.0 ! ip default-gateway 209.3.17.1 ! no ip http server !! control-plane !! line con 0 exec-timeout 0 0 privilege level 15 logging synchronous line aux 0 exec-timeout 0 0 privilege level 15 logging synchronous line vty 0 4 login ! end
---	---

		Поліщук К.В.			KIK.KP.B-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		57

Файл конфігурації комутатора SW_3_17_3

hostname SW_3_17_3 ! boot-start-marker boot-end-marker !! logging discriminator EXCESS severity drops 6 msg-body drops EXCESSCOLL logging buffered 50000 logging console discriminator EXCESS ! no aaa new-model no ip icmp rate-limit unreachable ! ip cef !! no ip domain-lookup ip domain-name LAN3 no ipv6 cef ipv6 multicast rpf use-bgp !! spanning-tree mode pvst spanning-tree extend system-id !! vlan internal allocation policy ascending ! ip tcp synwait-time 5 !! interface Ethernet0/0 duplex auto	interface Ethernet0/1 duplex auto ! interface Ethernet0/2 duplex auto ! interface Ethernet0/3 duplex auto ! interface Vlan1 ip address 210.3.17.2 255.255.255.0 ! ip default-gateway 210.3.17.1 ! no ip http server !! control-plane !! line con 0 exec-timeout 0 0 privilege level 15 logging synchronous line aux 0 exec-timeout 0 0 privilege level 15 logging synchronous line vty 0 4 login ! end
---	---

		Поліщук К.В.			KIK.KP.B-123-17-ПЗ	Арк.
		Єфіменко А.А.				
Змн.	Арк.	№ докум.	Підпис	Дата		58

Встановлення PenTBox

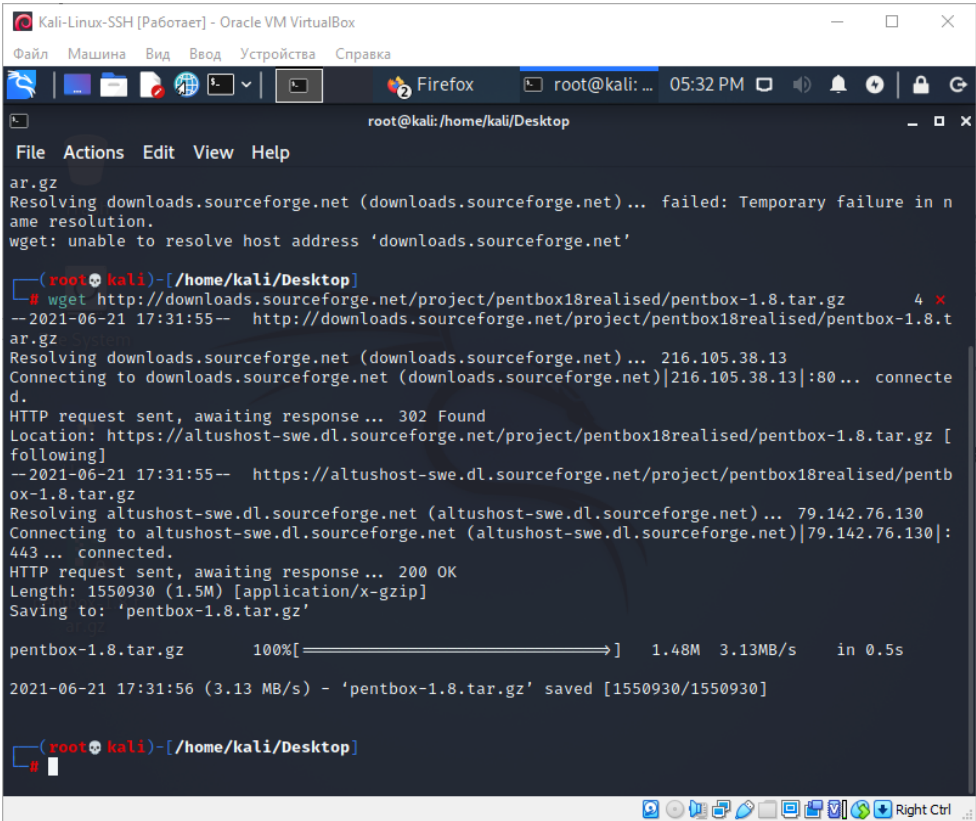


Рисунок В.1 – Завантаження програмного забезпечення PenTBox

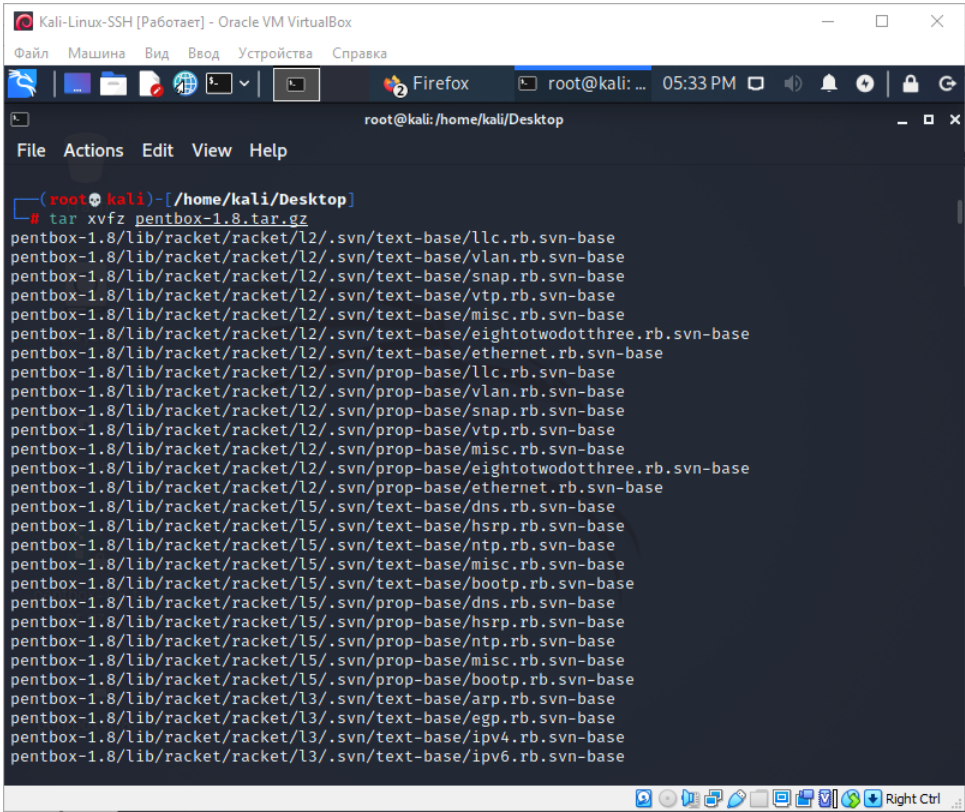


Рисунок В.2 – Розархівування програмного забезпечення PenTBox

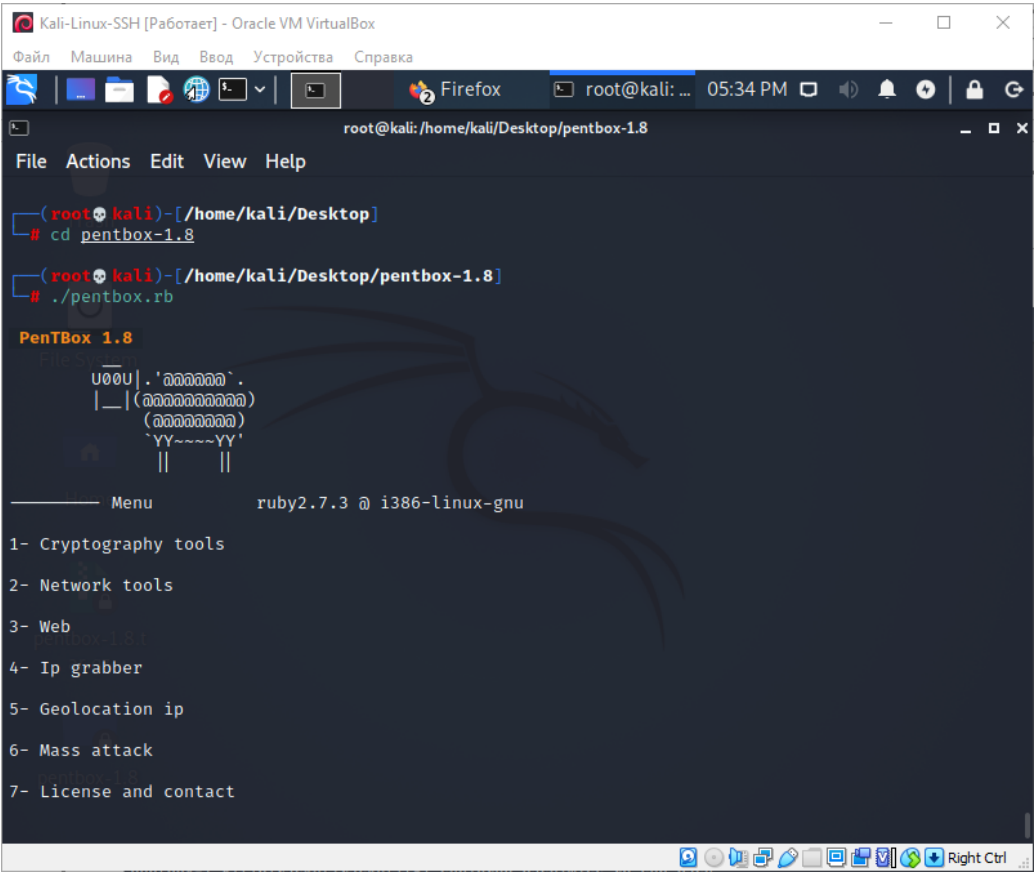


Рисунок В.3 – Запуск програмного забезпечення PenTBox