МАГІСТЕРСЬКА РОБОТА

15.04 – МР. 1859 “С” 2021.11.01.017 ПЗ

МАЛЯРЕНКА ОЛЕКСАНДРА ЮРІЙОВИЧА

2022 р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ПОГОДЖЕНО

Декан факультету
Інформаційних технологій
_____ Глазунова О.Г., д.пед.н, проф.
підпис ПІБ, вчене звання і ступінь

«___»_____2022 р.

ДОПУСКАЄТЬСЯ ДО ЗАХИСТУ

В.о. завідувач кафедри
Комп'ютерних систем, мереж та кібербезпеки
_____ Касаткін Д.Ю., к.п.н., доц.
підпис ПІБ, вчене звання і ступінь

«___»_____2022 р.

МАГІСТЕРСЬКА РОБОТА

На тему: «Дослідження ступеню захищеності обчислювальної мережі_____
Національного університету біоресурсів і природокористування України»__
Спеціальність 123 «Комп'ютерна інженерія»_____
Освітня програма Комп'ютерні системи та мережі_____
Орієнтація освітньої програми _____

Керівник магістерської роботи:_____ / Лахно В.А. /
підпис ПІБ

Виконав:_____ / Маляренко О.Ю. /
підпис ПІБ

КИЇВ-2022

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

«ЗАТВЕРДЖУЮ»
в.о. завідувача кафедри
комп'ютерних систем, мереж і кібербезпеки
_____ / Касаткін Д.Ю., к.п.н., доц. /
підпис ПІБ, вчене звання і ступінь
«__» _____ 20__ р.

З А В Д А Н Н Я

ДО ВИКОНАННЯ МАГІСТЕРСЬКОЇ РОБОТИ СТУДЕНТУ

Маляренко Олександр Юрійович
(прізвище, ім'я, по батькові)

Спеціальність (напрям підготовки): комп'ютерна інженерія
Освітня програма: комп'ютерні системи та мережі
Орієнтація освітньої програми: _____

Тема магістерської роботи: «Дослідження ступеню захищеності обчислювальної мережі Національного університету біоресурсів і природокористування України»

затверджена наказом ректора НУБіП України від "__" _____ р. №
Термін подання завершеної роботи на кафедру _____
Вихідні дані до магістерської роботи емулятор мережі - EVE-NG, операційні системи Linux Ubuntu Desktop 17.10.1, Linux Ubuntu Server 16.04.4, Windows Server S2016 R2

Перелік питань, що підлягають дослідженню:
1. Аналітичний огляд
2. Вимоги до мережі
3. Мережева безпека обчислювальної мережі

Перелік графічного матеріалу (за потреби) _____

Дата видачі завдання "__" _____ 2021 р.

Керівник магістерської роботи _____ Кахно В.А., д.т.н., проф.
(підпис) (прізвище та ініціали)

Завдання прийняв до виконання _____ Маляренко О.Ю.
(підпис) (прізвище та ініціали студента)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз предметної області		Виконано
2	Визначення вимог для мережі		Виконано
3	Дослідження безпеки мережі		Виконано
4	Оцінка ризиків мережі		Виконано
5	Оформлення пояснювальної записки		Виконано
6	Оформлення графічного матеріалу		Виконано

Студент _____ Маляренко О.Ю
(підпис) (ініціали та прізвище)

Керівник проєкту (роботи) _____ Лахно В.А
(підпис) (ініціали та прізвище)

РЕФЕРАТ

Пояснювальна записка: 85 сторінок, 22 рисунка, 6 таблиць, 17 джерел.

КОМП'ЮТЕРНА МЕРЕЖА, ДОСЛІДЖЕННЯ, АНАЛІЗ БЕЗПЕКИ, МЕРЕЖЕВІ АТАКИ, ВИМОГИ ДО МЕРЕЖІ, МЕРЕЖЕВА БЕЗПЕКА, ОЦІНКА РИЗИКІВ, ШКІДЛИВЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, DOS, DDOS, MITM, БРАНДМАУЕР

Предметом дослідження є методи для визначення ступеню захищеності обчислювальної мережі Національного університету біоресурсів і природокористування України.

Об'єкт дослідження – комп'ютерна мережа для Національного університету біоресурсів і природокористування України.

Мета роботи – дослідження ступеню захищеності обчислювальної мережі Національного університету біоресурсів і природокористування України.

Завданнями магістерської роботи є запобігання можливих загроз для комп'ютерної мережі Національного університету біоресурсів і природокористування України.

Перший розділ присвячений аналізу предметної області.

У другому розділі описуються вимоги обчислювальної мережі.

Третій розділ присвячений мережевій безпеці і оцінки ризиків.

Результати досягнуті в процесі роботи – було розроблено та проведено дослідження безпеки комп'ютерної мережі, яке показало, що дана система є повністю працездатною та виконує усі вказані функції, створено звіт оцінки ризиків.

Одержані результати можуть бути використані у усіх підприємствах, адже дані дослідження є універсальними.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

КМ – Комп’ютерна мережа

ОМ – Обчислювальна мережа

ІТ – Інформаційні технології

ПК – Персональний комп’ютера

ВМ – Віртуальна машина

ПЗ – Програмне забезпечення

IP – Міжмережевий протокол, Internet Protocol

WWW – Всесвітня мережа, World Wide Web

IPv4 – Мережевий протокол четвертої версії, Internet Protocol version 4

MAC – Це унікальний ідентифікатор, котрий мають всі активні пристрої , Media Access Control

VLAN – Віртуальна мережа, Virtual Local Area Network

DHCP – Протокол, що відповідає за динамічну видачу IP-адресів, Dynamic Host Configuration Protocol

HSRP – Протокол маршрутизації, при якому створюється група маршрутизаторів з спільним IP-адресом, Hot Standby Router Protocol

VTP – Протокол, що призначений для обміну інформації о VLAN, VLAN Trunking Protocol

DNS – Комп’ютерна система, яка допомагає отримати інформацію про домени, Domain Name System

PAT – Технологія трансляції мережевої адреси, Port Address Translation

NAT – це функція для змінення IP-адреси, коли відбувається передача пакетів, Network Address Translation

PVST – Протокол призначений для роботи з декількома VLAN,

Per-VLAN Spanning Tree

TCP – Основний протокол передачі даних, Transmission Control Protocol

DoS – Тип хакерської атаки, яка полягає в великій кількості зовнішніх запитів, Denial of Service

DDos – Аналогічний тип атаки, як DoS, але відбувається з Великої кількості IP-адресів, Distributed Denial of Service

MITM – Тип хакерської атаки, коли зловмисник прослуховує або змінює зв'язок між двома клієнтами, Man in the middle

ARP – Протокол, який визначає MAC-адресу комп'ютера по IP-адресі, Address Resolution Protocol

HTTP – Протокол прикладного рівня передачі даних

HTTPS – Розширення протоколу HTTP для підтримки шифрування з метою підвищення безпеки

ВСТУП

У сучасному світі комп'ютер став невід'ємною частиною ділового сектору не тільки для професійної діяльності, але й для особистої діяльності. З розвитком технологій з'явилася мережа, і поступово від початкової дротової мережевої технології ми перейшли до бездротової мережевої технології. Тепер, якщо ми замислимося, то зрозуміємо, що мережеві технології впливають на все наше життя.

У світі інформаційних технологій (ІТ) інформація є будівельним матеріалом для ефективної комунікації. Комунікація є засобом, який допомагає нам керувати нашими щоденними професійними та особистими операціями.

В наш час, мабуть, кожна людина використовує комп’ютерні мережі, в деяких випадках, навіть не знаючи цього. Адже завдяки ним, ми отримуємо доступ до гігантської кількості інформації за декілька секунд. Комп'ютерні мережі допомагають користувачам мережі спільно використовувати ресурси та спілкуватися. Чи можете ви уявити собі світ без електронної пошти, онлайнових газет, блогів, чату та інших послуг, що пропонуються Інтернетом?

Підключення наших пристроїв до Інтернету та інших мереж відкриває перед нами світ можливостей, але ці зв'язки також роблять наші пристрої вразливими до пошкоджень, а інформацію - до крадіжки. Вирішити цю проблему допомагає кібернетична та мережева безпека.

Мережева безпека - це будь-які дії організації, спрямовані на запобігання зловмисному використанню або випадковому пошкодженню приватних даних мережі, її користувачів або їхніх пристроїв. Мета мережевої безпеки полягає в тому, щоб мережа працювала і була безпечною для всіх законних користувачів.

Без належного захисту будь-яка мережа є вразливою до зловмисного використання та випадкового пошкодження. Хакери, незадоволені співробітники або погані практики безпеки в організації можуть залишити

вразливими приватні дані, в тому числі комерційну таємницю та особисті дані клієнтів.

Втрата конфіденційних досліджень, наприклад, може потенційно коштувати організації мільйони доларів, позбавляючи її конкурентних переваг, за які вона заплатила. А викрадення хакерами даних клієнтів та їх продаж створює негативний імідж та недовіру до організації.

Більшість поширених атак на мережі спрямовані на отримання доступу до інформації, шляхом шпигунства за комунікаціями та даними користувачів, а не на пошкодження самої мережі. Але зловмисники можуть зробити більше, ніж викрасти дані. Вони можуть пошкодити пристрої користувачів або маніпулювати системами, щоб отримати фізичний доступ до об'єктів. Це залишає майно організації та її членів під загрозою заподіяння шкоди.

Отже, компетентні процедури мережевої безпеки забезпечують захист даних і блокують вразливі системи від зовнішнього втручання. Це дозволяє користувачам мережі залишатися в безпеці і зосередитися на досягненні цілей організації. Більше того, це означає, що клієнти і партнери також можуть впевнено взаємодіяти з організацією.

1 АНАЛІТИЧНИЙ ОГЛЯД

1.1 Дослідження предметної області

У сучасному світі мережева безпека стає все більш складним завданням, оскільки все більше бізнес-додатків переміщуються в приватні і публічні хмари. Більше того, самі додатки зараз мають тенденцію до віртуалізації і розподілені по багатьох місцях, деякі з яких знаходяться поза фізичним контролем команд IT-безпеки. Оскільки кількість атак на компанії постійно зростає, захист мережевого трафіку та інфраструктури має вирішальне значення.

Мережева безпека є ключем до здатності організації надавати продукти та послуги клієнтам і співробітникам. Від інтернет-магазинів до корпоративних додатків і віддалених робочих столів, захист додатків і даних в мережі має важливе значення для розвитку бізнесу, не кажучи вже про захист репутації організації. Крім того, ефективна мережева безпека може підвищити продуктивність мережі, усуваючи простої через успішні атаки.

Безпека охоплює широкий спектр технологій, пристроїв та процесів. Вона відноситься до сукупності правил і конфігурацій, унікально розроблених для захисту комп'ютерних мереж та їх даних. Цілісність, конфіденційність і доступність цих комп'ютерів підтримуються за допомогою мережевої безпеки та програмно-апаратних технологій.

Мережа вважається захищеною лише тоді, коли вона включає три ключові компоненти - конфіденційність, цілісність та доступність. Ця комбінація, що отримала назву "тріада CIA(confidentiality, integrity, availability)", є загальновідомим стандартом, який використовується при створенні політик мережевої безпеки для будь-якої організації[1].

Хоча кожен член вашої організації може докласти зусиль для забезпечення безпеки, мережева безпека стала більш складним завданням в останні роки. Належний захист мереж і підключених до них пристроїв вимагає всебічної мережевої підготовки, глибокого розуміння того, як насправді працюють мережі,

і навичок застосування цих знань на практиці. Дуже важливо, щоб мережі були ретельно і належним чином налаштовані, захищені і контролювалися для повного збереження конфіденційності.

Перш ніж розглядати різні види атак на безпеку і те, як мережева безпека допомагає їх уникнути, необхідно зрозуміти, в чому полягає вразливість мережі. Будь-яка вразливість дає хакерам можливість отримати доступ до інфраструктури, встановити шкідливе програмне забезпечення і навіть викрасти і змінити дані, якщо не знищити або стерти їх.

Вразливість безпеки - це ненавмисна характеристика обчислювального компонента або конфігурації системи, яка збільшує ризик виникнення несприятливої події або збитків внаслідок випадкового впливу, навмисної атаки або конфлікту з новими компонентами системи.

За своїм визначенням, вразливість можна виправити за допомогою виправлення програмного забезпечення, реконфігурації, навчання користувачів, оновлення мікропрограми або заміни обладнання, на відміну від ризику безпеки, який може бути неминучим. З розвитком цифрових систем з'являються нові вразливості.

1.1.1 Вразливості у вихідному коді

Вразливість коду - це термін, пов'язаний з безпекою вашого програмного забезпечення. Це недолік у вашому коді, який створює потенційний ризик порушення безпеки. Цей недолік дозволить хакерам скористатися вашим кодом, приєднавши кінцеву точку для вилучення даних, втрутитися у ваше програмне забезпечення або, що ще гірше, стерти все. Хоча вам здається, що це малоймовірно, дані Contrast Security свідчать, що близько 76% додатків містять щонайменше одну вразливість. 34% містять більше чотирьох вразливостей.

Які ж уразливості може мати код? Хоча список довгий і детальний, згадаємо декілька з тих, які найбільш ймовірно можуть трапитися, а також ті, які завдають найбільшої шкоди. Уразливості, які може мати ваше програмне забезпечення:

– ін'єкція - це уразливість, яка дозволяє зловмисникам "впроваджувати" код в систему за допомогою простих системних викликів. Ці виклики, як правило, виконуються за допомогою зовнішніх програм через команди командного інтерпретатора. Ін'єкції в базу даних або SQL-ін'єкції є найпоширенішими та найнебезпечнішими з усіх. Зазвичай зловмисник знаходить параметр, який проходить через базу даних, використовує цей параметр для передачі шкідливої SQL-команди в якості вмісту. База даних зберігає її і помилково сприймає як код, обманюючи програмне забезпечення для відправки, зміни або видалення бази даних. SQL-ін'єкції настільки небезпечні, що практично не існує веб-додатків, які були б від них захищені, оскільки всі вони працюють на зовнішніх командах. Все, що можна зробити, це написати програму настільки досконало, щоб ін'єкції були більш складними і, отже, не залишали вразливостей;

– міжсайтовий скриптинг (Cross-Site Scripting, XSS) – здійснюється на веб-сайтах. Шкідливий скрипт, як правило, на Javascript і HTML, вводиться як дані на сайт, де він може прикріпитися і викликати проблеми з безпекою. Практично неможливо, щоб браузер користувача виявив такі сценарії, оскільки для нього сценарій надійшов з надійного джерела. Зазвичай це робиться в кодах, які містять конфіденційну інформацію, таку як ваш контактний номер або, що набагато гірше, дані вашої кредитної картки;

– переповнення буферу – це вразливість відбувається, якщо буфер буде переповнений даними або запитами більше, ніж він може обробити. Він переповниться в сусіднє сховище. Це переповнення може призвести до серйозних проблем, таких як збій програмного забезпечення, втрата даних або, що найнебезпечніше, створення точки входу для кібератак. Ця вразливість коду називається переповненням буфера (Buffer Overflow) і залежить від мови програмування. Javascript та Pearl - це дві мови, які уникають таких атак, але мови будівельних блоків, C та C++,

піддаються такому впливу, що вся система може бути скомпрометована. Зловмисники зазвичай роблять це шляхом перезапису блоків коду на шляху виконання в програмному забезпеченні. Дані можуть містити певний скрипт або код, який може спонукати програмне забезпечення до небажаних дій;

– порушена аутентифікація – це вразливість, яка виникає, коли зловмисник використовує різні способи проникнення в чужий акаунт. Це призводить до помилкової авторизації, а потім знову до втрати конфіденційних даних. Хоча деякі з цих проблем не є виною розробників, все ж таки обов'язком кодера стає створення надійного коду, який може вирішити такі проблеми. Код стає вразливим у випадках, коли відсутні багаторазові перевірки або таймаути сеансів. Найпоширенішою вразливістю коду у веб-додатках є ситуація, коли для користувача створюється ідентифікатор сеансу, а хакер якимось чином отримує і використовує перезапис URL-адреси для відтворення цього сеансу. Інший спосіб - хакер може отримати доступ до вашої бази даних паролів, використовуючи інші вразливості безпеки, і якщо вона неправильно хешована, можна змінити кодування та відобразити пароль кожного[2].

1.1.2 Неправильно налаштовані компоненти системи

Неправильні конфігурації є ще однією поширеною помилкою при налаштуванні корпоративних IT-систем. Неправильна конфігурація безпеки виникає, коли параметри безпеки не визначені належним чином в процесі конфігурації або підтримуються і розгортаються з параметрами за замовчуванням. Це може вплинути на будь-який рівень стека додатків, хмари або мережі. Неправильно налаштовані хмари є основною причиною витоку даних, що коштує організаціям мільйони доларів. Неправильна конфігурація може мати місце з різних причин. Сучасні мережеві інфраструктури є складними та постійно змінюються - організації можуть не звертати уваги на важливі

параметри безпеки, такі як мережеве обладнання, яке все ще може мати конфігурації за замовчуванням.

Наприклад, неправильно налаштований сервер бази даних може призвести до того, що дані стануть доступними через звичайний веб-пошук. Якщо ці дані містять облікові дані адміністратора, зловмисник може отримати доступ до подальших даних за межами бази даних або здійснити іншу атаку на сервери компанії.

У разі неправильного налаштування (або відсутності) засобів контролю безпеки на пристроях зберігання даних, величезні обсяги конфіденційних та персональних даних можуть стати доступними широкому загалу через Інтернет. Як правило, не існує способу з'ясувати, хто міг отримати доступ до цієї інформації до того, як вона була захищена.

Навіть якщо організація має захищені конфігурації для своїх кінцевих точок, все одно необхідно регулярно проводити аудит засобів контролю безпеки і конфігурацій, щоб виявити зміну конфігурації. Нове обладнання додається в мережу, системи змінюються і застосовуються виправлення - все це призводить до неправильних конфігурацій.

Розробники можуть створювати мережеві ресурси та правила брандмауера для зручності, а при створенні програмного забезпечення залишати їх незмінними. Іноді адміністратори дозволяють змінювати конфігурацію з метою усунення несправностей або тестування, але вони не повертають її до початкового стану.

Співробітники часто тимчасово відключають антивірус, якщо він перевизначає певні дії (наприклад, запуск інсталяторів), а потім не пам'ятають про його повторне включення. За оцінками, понад 20% кінцевих точок мають застаріле антивірусне програмне забезпечення або антивірус.

Випадки в IT-середовищі, які можуть призвести до неправильної конфігурації системи безпеки:

– облікові записи/паролі за замовчуванням - використання наданих значень за замовчуванням для системних облікових записів та паролів є

поширеною помилкою конфігурації безпеки, яка може дозволити зловмисникам отримати несанкціонований доступ до системи;

– безпечна політика паролів - не впровадження політики паролів може дозволити зловмисникам отримати несанкціонований доступ до системи за допомогою таких методів, як використання списків поширених імен користувачів та паролів для перебору поля імені користувача та/або пароля до успішної автентифікації;

– програмне забезпечення застаріле, а недоліки не виправлені - відсутність оновлення виправлень програмного забезпечення в рамках процесу управління програмним забезпеченням може дозволити зловмисникам використовувати такі методи, як ін'єкція коду для впровадження шкідливого коду, який потім виконується програмою;

– незахищені файли та каталоги - залишення файлів та каталогів незахищеними може дозволити зловмисникам використовувати такі методи, як примусовий перегляд для отримання доступу до файлів з обмеженим доступом або областей в каталозі сервера;

– увімкнені або встановлені невикористовувані функції – не видалення непотрібних функцій, компонентів, документації та зразків робить додаток вразливим до вразливостей, пов'язаних з неправильною конфігурацією, і може дозволити зловмисникам використовувати такі методи, як ін'єкція коду для впровадження шкідливого коду, який потім виконується додатком;

– функції безпеки не підтримуються або не налаштовані належним чином - неможливість належним чином налаштувати та підтримувати функції безпеки робить додаток вразливим до атак на неправильну конфігурацію;

– неопубліковані URL-адреси не блокуються від прийому трафіку від звичайних користувачів - неопубліковані URL-адреси, до яких звертаються ті, хто підтримує додатки, не призначені для прийому трафіку від звичайних користувачів. Якщо не заблокувати ці URL-

адреси, не може становити значний ризик при їх скануванні зловмисниками;

– обхід каталогів - дозволяє зловмиснику отримати доступ до каталогів, файлів і команд, які знаходяться за межами кореневого каталогу. Озброївшись доступом до вихідного коду програми або конфігурації та критичних системних файлів, кіберздочинець може змінити URL-адресу таким чином, щоб програма могла виконати або відобразити вміст довільних файлів на сервері. Будь-який пристрій або додаток, що відкриває інтерфейс на основі HTTP, може бути вразливим до атаки обходу каталогів[3].

1.1.3 Конфігурації довіри

Конфігурації довіри відносяться до дозволів, які ви робите для обміну даними з програмними та апаратними системами. Наприклад, змонтований жорсткий диск може зчитувати конфіденційні дані з комп'ютерного клієнта без необхідності отримання додаткових привілеїв. Між активними каталогами та обліковими записами можуть існувати довірчі відносини, що призводить до безперешкодного потоку даних між джерелами, які постійно не контролюються. Як тільки зловмисник отримує доступ до скомпрометованої системи, він може використати ці вразливості конфігурації довіри, щоб поширити інфекцію з початкової системи та вивести з ладу все ваше IT-середовище.

1.1.4 Слабка практика авторизації

Це стало однією з найпоширеніших причин вразливостей як у користувацьких, так і в корпоративних системах. Користувачі, як правило, дотримуються зручних або комфортних практик облікових даних, надаючи перевагу простоті використання над безпекою. Наприклад, зараз є нормою (незважаючи на рекомендації експертів) зберігати паролі та облікові дані облікових записів у вбудованому менеджері паролів браузера. Потенційно вразливими є слабкі паролі, які використовують поширені алфавітно-цифрові

рядки, а також ті, що повторно використовують персональні дані, такі як ваше ім'я. Найгірші паролі зображені на рис 1.1, всі ці перелічені паролі можна зламати менше чим за 1 секунду. Ці вразливості можна усунути на двох рівнях - через обізнаність користувачів та примусові процеси авторизації, такі як закінчення терміну дії пароля.

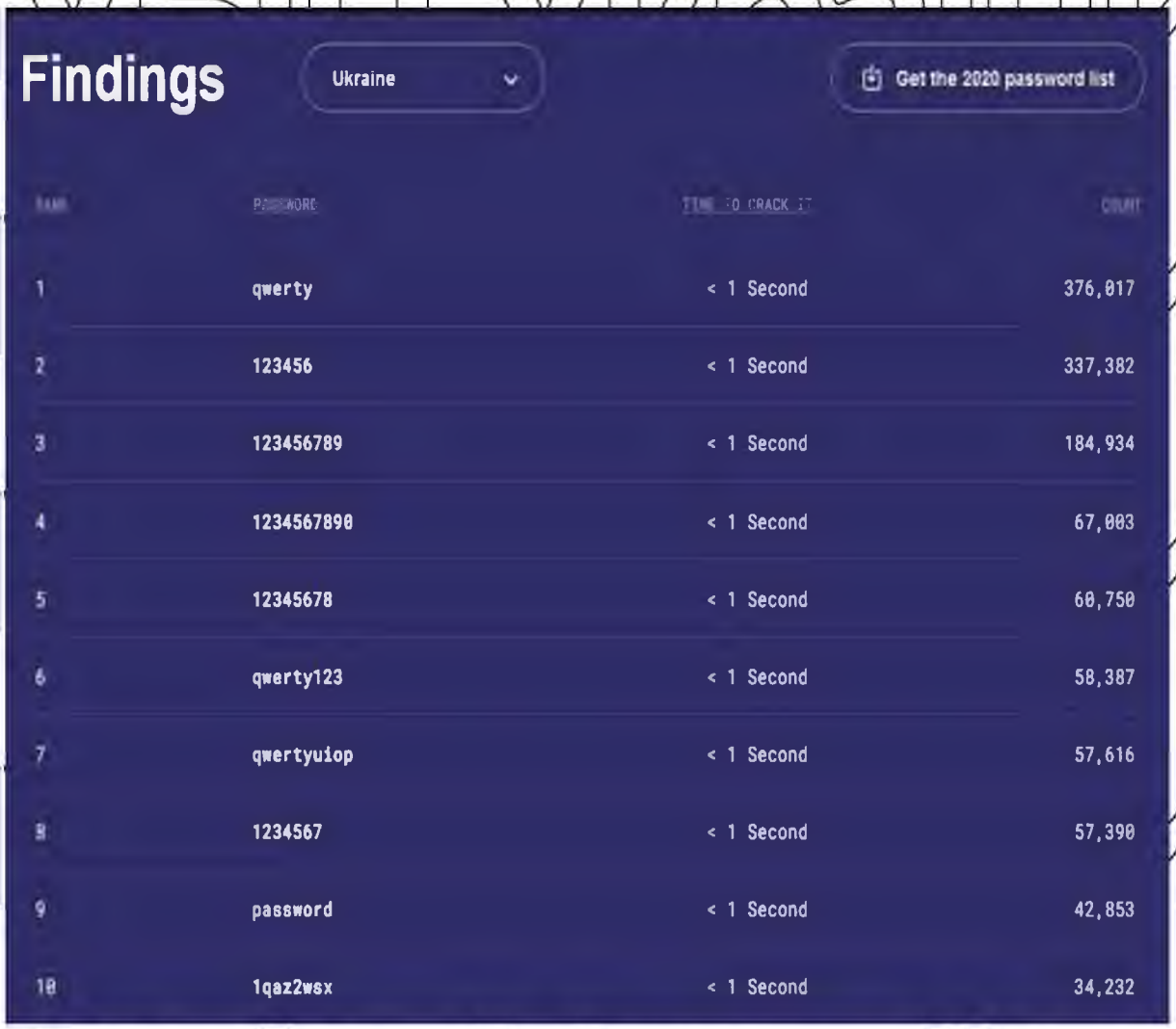

Рисунок 1.1 – Топ-10 популярних і найгірших паролів в Україні

1.1.5 Відсутність належного шифрування

Шифрування підвищує безпеку повідомлення або файлу шляхом шифрування вмісту. Щоб зашифрувати повідомлення, вам потрібен правильний ключ, і вам також потрібен правильний ключ, щоб розшифрувати його. Це найефективніший спосіб приховати спілкування за допомогою закодованої

інформації, де відправник і одержувач мають ключ для розшифровки даних. Ця концепція не надто відрізняється від дітей, які придумують секретні кодові слова та інші непомітні способи спілкування, де тільки вони можуть зрозуміти повідомлення. Шифрування схоже на обмін секретними повідомленнями між сторонами - якщо хтось спробує втрутитися без належних ключів, він не зможе зрозуміти повідомлення.

Важливо розуміти відмінності між симетричним та асиметричним шифруванням і те, як ці технології безпеки працюють у повсякденній безпечній передачі повідомлень.

Симетричне шифрування, що зображене на рис. 1.2 - це широко використовуваний метод шифрування даних, при якому дані шифруються і розшифровуються за допомогою одного секретного криптографічного ключа.

Рисунок 1.2 – Симетричне шифрування

Зокрема, ключ використовується для шифрування відкритого тексту - стану даних до шифрування або після розшифрування - і розшифрування зашифрованого тексту - стану даних після шифрування або до розшифрування.

Симетричне шифрування є одним з найпоширеніших методів шифрування, а також одним з найстаріших, що бере свій початок ще з часів Римської імперії. Відомим історичним прикладом симетричного шифрування в дії є шифр Цезаря, названий на честь не кого іншого, як Юлія Цезаря, який використовував його для шифрування свого військового листування.

Метою симетричного шифрування є захист чутливої, секретної або засекреченої інформації. Воно використовується щодня в багатьох основних галузях промисловості, включаючи оборонну, аерокосмічну, банківську, охорону здоров'я та інші галузі, в яких захист конфіденційних даних людини, бізнесу або організації має першорядне значення.

Симетричне шифрування працює за допомогою потокового або блокового шифру для шифрування та дешифрування даних. Потоковий шифр перетворює відкритий текст у зашифрований по одному байту за раз, а блоковий шифр перетворює цілі одиниці або блоки відкритого тексту з використанням заздалегідь визначеної довжини ключа, наприклад, 128, 192 або 256 біт.

Відправники та одержувачі, які використовують симетричне шифрування для передачі даних один одному, повинні знати секретний ключ, щоб, у випадку відправників, зашифрувати дані, якими вони мають намір поділитися з одержувачами, а у випадку одержувачів, розшифрувати і прочитати зашифровані дані, якими відправники поділилися з ними, а також зашифрувати будь-які необхідні відповіді.

На відміну від симетричного шифрування, яке використовує один і той же секретний ключ для шифрування та дешифрування конфіденційної інформації, асиметричне шифрування, також відоме як криптографія з відкритим ключем або шифрування з відкритим ключем, використовує математично пов'язані пари відкритого та закритого ключів для шифрування та дешифрування конфіденційних даних відправника та одержувача.

Як і при симетричному шифруванні, відкритий текст все одно перетворюється на зашифрований і навпаки під час шифрування та дешифрування відповідно. Основна відмінність полягає в тому, що для асиметричного шифрування даних, що зображено на рис. 1.3, використовуються дві унікальні пари ключів.

Рисунок 1.3 – Асиметричне шифрування

Однією з причин, чому асиметричне шифрування часто вважається більш безпечним, ніж симетричне, є те, що асиметричне шифрування, на відміну від симетричного, не вимагає обміну одним і тим же ключем шифрування-дешифрування між двома або більше сторонами. Так, відбувається обмін відкритими ключами, але користувачі, які обмінюються даними в асиметричній криптосистемі, мають унікальні пари відкритих і закритих ключів, і їх відкриті ключі, оскільки вони використовуються тільки для шифрування, не становлять ризику несанкціонованого розшифрування хакерами, якщо вони стануть відомі, оскільки хакери, за умови збереження закритих ключів у таємниці, не знають закритих ключів користувачів і, відповідно, не можуть розшифрувати зашифровані дані.

Асиметричне шифрування також дозволяє перевіряти автентичність цифрового підпису, на відміну від симетричного шифрування. В основному, це передбачає використання закритих ключів для цифрового підпису повідомлень або файлів, а відповідні їм відкриті ключі використовуються для підтвердження того, що ці повідомлення походять від правильного, перевіреного відправника[4].

З розвитком зовнішніх і ворожих загроз цікаві та інноваційні технології шифрування, такі як пост-квантова криптографія, квантовий розподіл ключів і гомоморфне шифрування, будуть мати вирішальне значення для захисту кібербезпеки.

1.1.6 Внутрішня загроза

Внутрішня загроза - це ризик кібербезпеки, який походить зсередини організації. Зазвичай вона виникає, коли нинішній або колишній працівник, підрядник, постачальник або партнер, який має законні облікові дані користувача, зловживає своїм доступом на шкоду мережам, системам і даним організації. Інсайдерська загроза може бути здійснена навмисно або ненавмисно. Незалежно від намору, кінцевим результатом є порушення конфіденційності, доступності та/або цілісності корпоративних систем і даних.

Інсайдерські загрози є причиною більшості витоків даних. Традиційні стратегії, політики, процедури та системи кібербезпеки часто зосереджені на зовнішніх загрозах, що робить організацію вразливою до атак зсередини. Оскільки інсайдер вже має дійсний дозвіл на доступ до даних і систем, фахівцям з безпеки і програмам складно відрізнити нормальну діяльність від шкідливої.

Зловмисні інсайдери мають явну перевагу над іншими категоріями зловмисників через їх знайомство з корпоративними системами, процесами, процедурами, політиками і користувачами. Вони добре обізнані з версіями

систем та вразливоетями в них. Тому організації повинні протидіяти внутрішнім загрозам щонайменше з такою ж ретельністю, як і зовнішнім загрозам.

1.1.7 Психологічна вразливість

Психологічні вразливості також спричинені людиною, але, на відміну від внутрішніх загроз, вони є ненавмисними, і до них схильна кожна людина. Як люди, ми мотивовані основними психологічними чинниками, такими як прагнення до самозбереження, бажання заощадити/отримати ексклюзивні вигоди та страх перед небезпекою. Хакери зазвичай використовують ці вразливості за допомогою соціальної інженерії. Вони переконують користувачів, що їм необхідно вжити заходів для розблокування вигоди або уникнення несприятливої ситуації. Простим прикладом є психологічна вразливість, яка призводить до того, що багато користувачів натискають на електронні листи, що підробляють акційні знижки, і завантажують шкідливе програмне забезпечення в свої системи[5].

1.1.8 Неадекватна автентифікація

Вразливості автентифікації виникають, коли немає достатньої кількості стримувань і противаг для скидання паролів і облікових даних. Це означає, що хакер може скористатися опцією "забув пароль", яка присутня в кожній системі входу в систему, щоб захопити ваш обліковий запис і знайти чорний хід для здійснення атаки з метою заволодіння обліковим записом. Питання для автентифікації може бути занадто легким для вгадування - наприклад, ваша дата народження, яка є загальнодоступною завдяки соціальним мережам. Або система може не дотримуватися процедур багатофакторної автентифікації, коли компрометація одного пристрою не може вплинути на безпеку облікового запису.

1.1.9 Викриття конфіденційних даних

У міру того, як світ стає все більш цифровим, організації по всьому світу почали збирати все більше і більше персональних даних. Збір та обробка персональних даних допомагають організаціям не тільки краще зрозуміти своїх споживачів та підвищити рівень їхньої задоволеності, але й отримати прибуток. Разом з тим, більшість організацій мають обмежену видимість персональних даних через великий обсяг персональних даних, які вони збирають, та їх розподіл по різнорідним системам. Персональні дані розподілені між великою кількістю платформ і систем, таких як локальні, гібридні та мультихмарні інформаційні активи.

Конфіденційна інформація - це все, що не повинно бути доступним для несанкціонованого доступу, відоме як чутливі дані. Конфіденційні дані можуть включати інформацію, що ідентифікує особу, наприклад, фінансову інформацію або облікові дані для входу в систему. Викриття конфіденційних даних відбувається, коли організація несвідомо розкриває конфіденційні дані або коли інцидент безпеки призводить до випадкового або незаконного знищення, втрати, зміни або несанкціонованого розкриття конфіденційних даних або доступу до них. Така вразливість даних може виникнути внаслідок неналежного захисту бази даних, неправильних конфігурацій при створенні нових екземплярів сховищ даних, неналежного використання систем даних тощо.

Витік конфіденційних даних може бути трьох типів:

- порушення конфіденційності: коли відбувається несанкціоноване або випадкове розголошення або доступ до конфіденційних даних;
- порушення цілісності: коли відбувається несанкціонована або випадкова зміна конфіденційних даних;
- порушення доступності: коли відбувається несанкціонована або випадкова втрата доступу до конфіденційних даних або їх знищення. Це включає як постійну, так і тимчасову втрату конфіденційних даних.

Організації, які збирають конфіденційні дані, несуть відповідальність за їх захист, а невиконання цього обов'язку може призвести до великих штрафів та покарань[6].

1.1.10 Недостатній моніторинг та ведення логів

Майже всі основні інциденти безпеки виникають через використання недостатнього логування, незапланованих стратегій безпеки або недостатнього моніторингу. Компанії, що використовують додатки з недостатніми функціями реєстрації або взагалі без них, ризикують зіткнутися з атаками, на усунення яких може знадобитися так багато часу, що вони можуть завдати значної шкоди всьому технологічному стеку.

Функції логування та моніторингу надають адміністраторам і командам безпеки необроблені дані про трафік, які допомагають виявити потенційні загрози шляхом виявлення незвичайних шаблонів. Ці механізми є основними стовпами безпеки, які формують фундамент надійно керованої системи безпеки.

При відсутності ретельно спланованих механізмів логування організація втрачає аудиторський слід для аналізу безпеки, тим самим дозволяючи векторам атак мати достатньо часу для подальшого проникнення в різні компоненти екосистеми.

Атаки, засновані на недостатньому моніторингу та логуванні вразливостей, зазвичай мають високий рівень поширеності, середній рівень можливостей і низький рівень виявлення. Забезпечення логуванню всіх подій і, як наслідок, моніторингу подій часто вважається першим кроком у виявленні вторгнення.

1.1.11 Вразливості спільного користування

Вразливості спільного користування є неминучою реальністю епохи хмарних технологій. Загальнодоступні хмарні рішення працюють в моделі з декількома орендарями, де спільний набір ресурсів надається в оренду різним

організаціям в різний час, в залежності від масштабу їх потреб в ресурсах. Якщо один з орендарів буде скомпрометований, існує ймовірність того, що атака пошириться на інші організації в хмарі, використовуючи вразливості спільної оренди. Саме тому організації, що мають справу з конфіденційною інформацією, вирішують розподіляти робоче навантаження між державними та приватними орендарями, зберігаючи свої найцінніші дані відокремлено.

1.2 Призначення системи

Комп'ютерна мережа є однією з найважливіших складових в будь-якій організації. Вона відкриває важливі функції для співробітників, такі як:

- спільний доступ до файлів. Ви можете отримати доступ до файлів на інших комп'ютерах мережі. За допомогою мережі ви можете отримати доступ до файлу без необхідності фізично йти до іншого комп'ютера. І ви все контролюєте: в мережі ви можете надавати доступ до того, чим хочете поділитися, і зберігати в таємниці те, що ви хочете зберегти в таємниці;
- потокове передавання даних. Потокове передавання мультимедійних даних - це процес надсилання цифрових мультимедійних даних, таких як фотографії, музика або відео, через мережу на пристрій, який може відтворювати ці мультимедійні дані;
- організація. Існує різноманітне програмне забезпечення для планування, яке дозволяє організовувати зустрічі без постійної перевірки розкладу кожного. Це програмне забезпечення зазвичай включає інші корисні функції, такі як спільні адресні книги та списки справ;
- спільне підключення до Інтернету. Ви можете спільно використовувати широкосмугове підключення до Інтернету - це означає, що вам не

потрібно купувати окремий обліковий запис в Інтернеті для кожного комп'ютера.

– віддалений доступ. Наявність власної мережі забезпечує більшу мобільність при збереженні того ж рівня продуктивності. Завдяки віддаленому доступу користувачі можуть мати доступ до тих самих файлів, даних і повідомлень, навіть коли вони не перебувають в офісі. Цей доступ може бути наданий мобільним портативним пристроям;

– спільне використання принтера. Замість того, щоб купувати принтер для підключення до кожного комп'ютера, ви можете використовувати один принтер і підключити його до мережі. Тоді ним зможуть користуватися всі учасники мережі.

Але головним призначенням є захист даних. Звичайні люди можуть надсилати дані, які можуть бути дуже чутливими, такі як їхні банківські реквізити, ім'я користувача та паролі, особисті документи, дані про покупки в Інтернеті або конфіденційні документи. Дуже важливо зберігати свої дані в надійному місці і комп’ютерна мережа допомагає в цьому і також полегшує компаніям в резервному копіюванню всіх їх даних на віддаленому сервері або інших системах резервного копіювання.

Оскільки захист даних є важливою складовою любої комп’ютерної мережі, тому мережева безпека є одним з найважливіших аспектів, які слід враховувати при роботі через Інтернет, локальну мережу або іншим способом, незалежно від того, наскільки малим або великим є ваша організація. Хоча не існує мережі, яка була б захищена від атак, стабільна та ефективна система мережевої безпеки має важливе значення для захисту даних клієнтів. Хороша система мережевої безпеки допомагає бізнесу знизити ризик стати жертвою крадіжки даних і саботажу.

Пошкодження інтелектуальної власності також є одним з наслідків несправних систем мережевої безпеки. Хакерські атаки надають несанкціонований доступ до інформації компанії або фізичної особи. Прикладом може слугувати злам системи безпеки Citibank, від якого постраждав приблизно

1% його клієнтів у США. Якщо хакер проникає в компанію і викрадає плани, ідеї або креслення, компанія може не мати можливості впроваджувати нові розробки та продукти. Це може зруйнувати бізнес або призвести до його стагнації.

Компанія також може зазнати втрати доходів. Більшість атак на мережу можуть призвести до збою в роботі. Тривалий простій мережі може призвести до втрати доходів, оскільки компанія може бути змушена припинити всі транзакції. Чим довше мережа не працює, тим більше втрачається доходу. Крім відчутної втрати доходів, може постраждати репутація компанії через втрату довіри до неї.

1.3 Огляд існуючих засобів

Віртуалізація комп'ютерних мереж дозволяє налаштовувати і виконувати мережеві експерименти за невеликі кошти і з невеликими зусиллями. Вона дозволяє "створити" декілька віртуальних мережевих пристроїв (повноцінних маршрутизатори, комутатори, хости тощо), які можна легко з'єднати між собою з метою формування мережу на одному комп'ютері. Мережеве обладнання є віртуальним, але має багато характеристик реального, включаючи інтерфейс конфігурації.

Також тестування конфігурацій є загальною потребою як для мережевих адміністраторів, так і для комп'ютерних науковців, які цікавляться мережевими технологіями. Перші можуть скористатися фазою тестування для перевірки того, що певна конфігурація працює так, як очікується, перед її розгортанням, в той час як другі очікується, перш ніж розгортати її, в той час як другі можуть використовувати результати тестування для того, щоб перевірити теоретичні моделі за допомогою практичних експериментів. В ідеалі, тестування має відбуватися в тих самих умовах, в яких конфігурація буде в кінцевому підсумку буде розгорнута конфігурація. Однак, це часто означає введення штучно згенерованого, потенційно шкідливого трафіку в живу мережу, що може завдати

ій шкоди і пошкодження. Ефективною альтернативою тестуванню в реальних умовах є реалізація мережевої конфігурації мережі, що цікавить, в безпечному, ізольованому програмному середовищі, яке максимально наближено відтворює реальну цільову установку.

Емулятор - це програмне або апаратне, або і те, і інше середовище, яке дублює (або емулює) функції однієї комп'ютерної системи (гостя) в іншій комп'ютерній системі (хост-системі), відмінній від першої, таким чином, що поведінка, яка емулюється, максимально нагадує поведінку реальної системи (гостя).

Віртуальна машина - це працююче програмне забезпечення, яке створює рівень абстракції між апаратно-програмною платформою та іншим програмним забезпеченням (можливо, операційною системою). Додатки, що працюють всередині віртуальної машини, взаємодіють з цим рівнем абстракції а не з фізичним обладнанням. Тому віртуальна машина може також реалізовувати віртуальні пристрої (диски, мережеві інтерфейси тощо), які відрізняються від тих, що доступних на платформі, на якій працює емулятор. Існує велика кількість продуктів емуляції, які можна розрізнити на основі прийнятої техніки емуляції, типу пристрою, який вони емулюють, та ліцензії, з якою вони розповсюджуються.

Розглянемо список мережевих симуляторів з відкритим вихідним кодом та мережевих емуляторів, які працюють на Linux або BSD.

1.3.1 Antidote (NRE Labs)

Antidote - це мережевий емулятор, поєднаний з презентаційною платформою, призначений для створення та проведення тренінгів з мережевих технологій. Його користувальницький інтерфейс працює у веб-браузері, включаючи термінали, які студенти використовують для виконання команд на емульованих мережевих пристроях і серверах. Інтерфейс даного емулятора зображено на рис. 1.4.

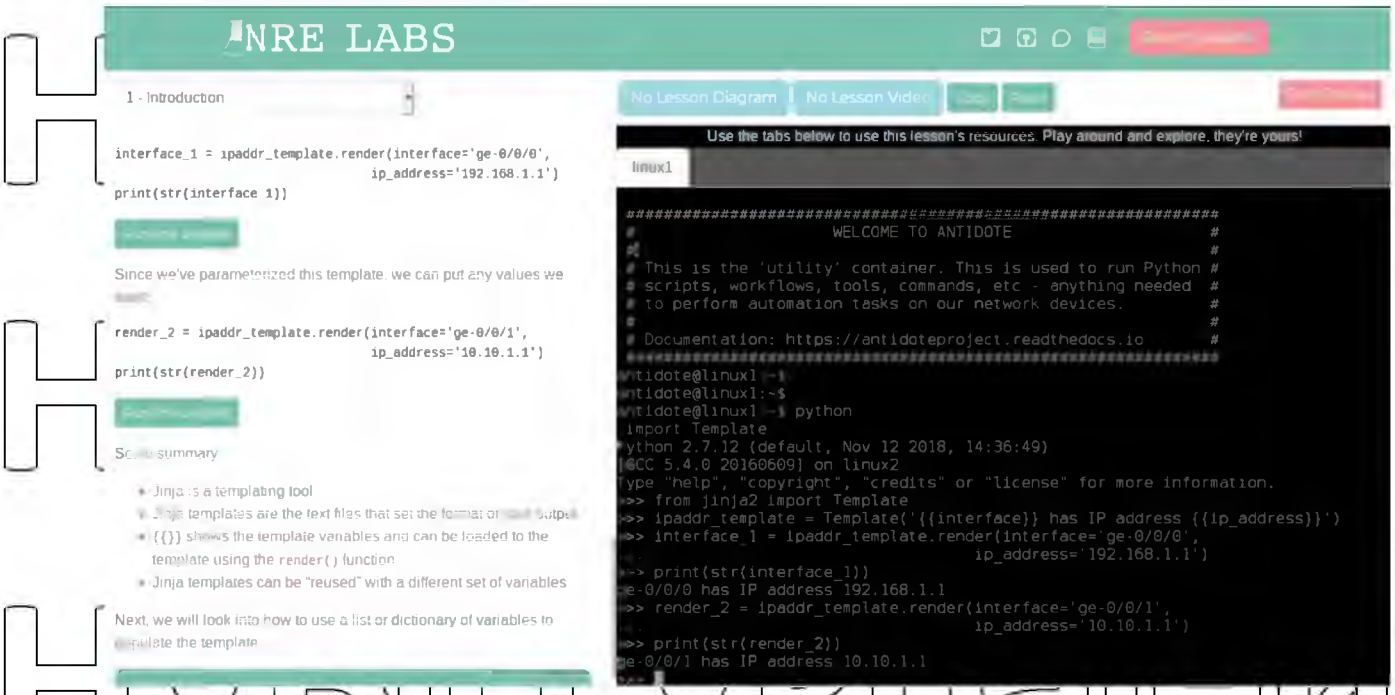

Рисунок 1.4 – Інтерфейс Antidote

1.3.2 Cloonix

Мережевий симулятор Cloonix надає відносно простий у використанні графічний інтерфейс користувача. Cloonix використовує QEMU/KVM для створення віртуальних машин. Cloonix надає широкий вибір готових файлових систем, які можуть бути використані в якості віртуальних машин, а також надає прості інструкції для створення інших кореневих файлових систем віртуальних машин. Cloonix має активну команду розробників, які оновлюють інструмент кожні два-три місяці і дуже швидко реагують на побажання користувачів. Інтерфейс даного емулятора зображено на рис. 1.5.

1.3.3 Containerlab

Containerlab - це мережевий емулятор з відкритим вихідним кодом, який швидко створює мережеві тестові середовища в стилі Devops. Він надає інтерфейс командного рядка для організації та управління мережевими лабораторіями на основі контейнерів. Він запускає контейнери, будує віртуальну проводку між ними для створення лабораторних топологій і управляє життєвим циклом кожної лабораторії.

Containerlab підтримує контейнерні образи маршрутизаторів, доступні від основних постачальників мережевого обладнання. Що ще цікавіше, Containerlab підтримує будь-яку мережеву операційну систему з відкритим вихідним кодом, яка публікується як контейнерний образ. Інтерфейс даного емулятора зображено на рис. 1.6.

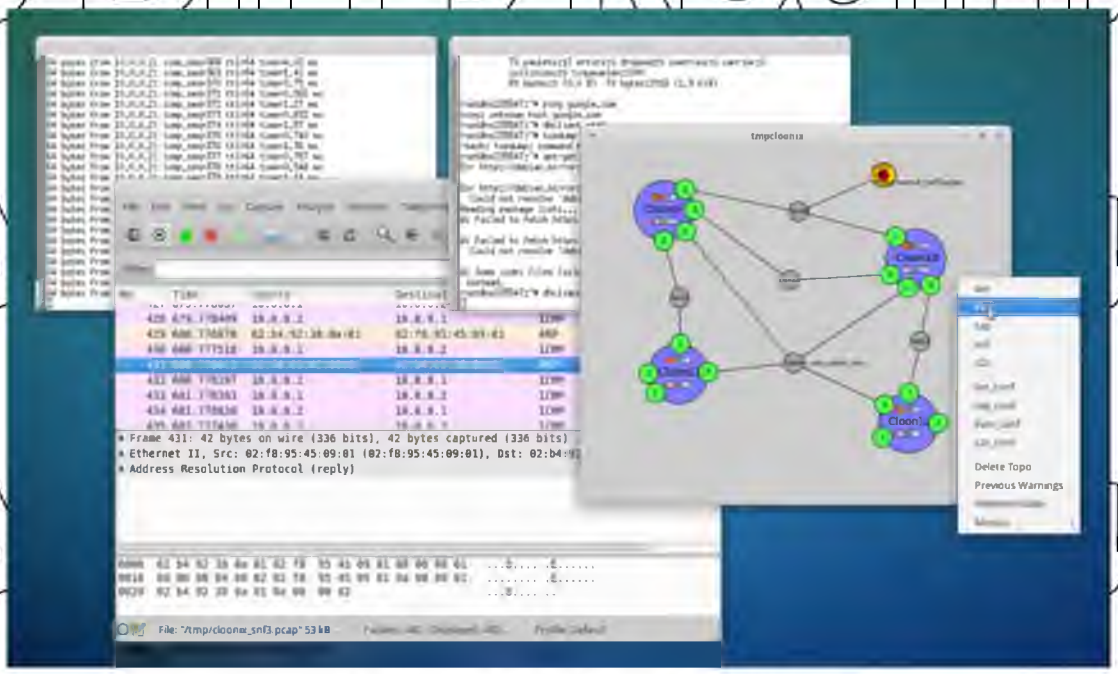

Рисунок 1.5 – Інтерфейс Cloonix

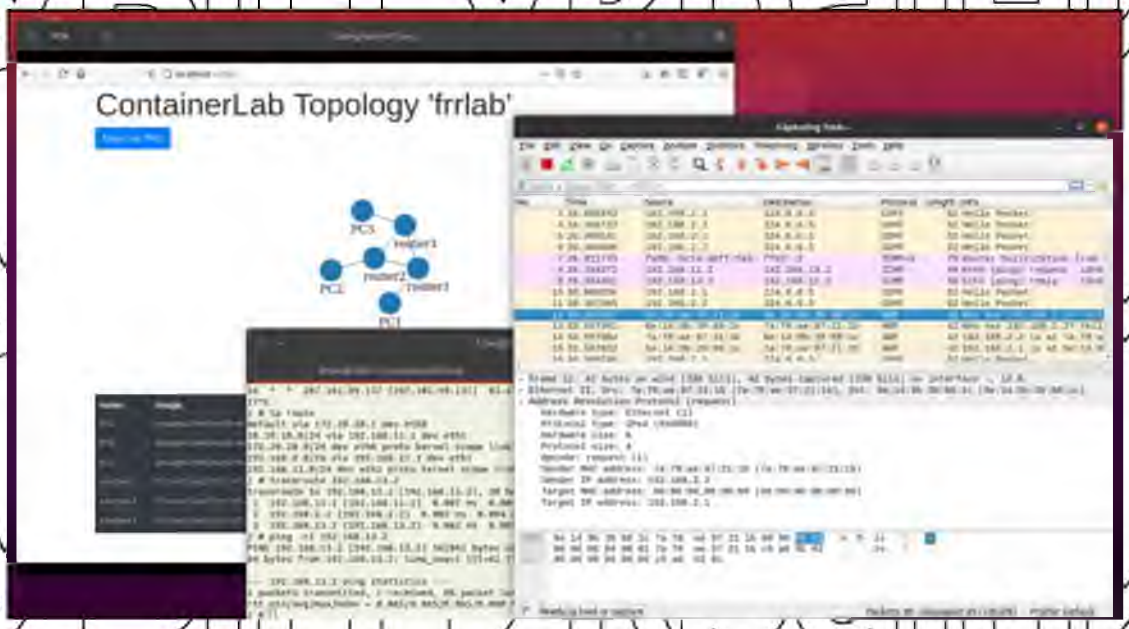

Рисунок 1.6 – Інтерфейс Containerlab

1.3.4 CORE

Емулятор Common Open Research Emulator (CORE) має графічний інтерфейс і використовує функціонал мережевих просторів імен в Linux Containers (LXC) як технологію віртуалізації. Це дозволяє CORE швидко запускати велику кількість віртуальних машин. CORE підтримує симуляцію фіксованих та мобільних мереж.

CORE буде працювати як на Linux, так і на FreeBSD. CORE є форком мережевого симулятора IMUNES і додає деякі нові функціональні можливості в порівнянні з IMUNES. Інтерфейс даного емулятора зображено на рис. 1.7.

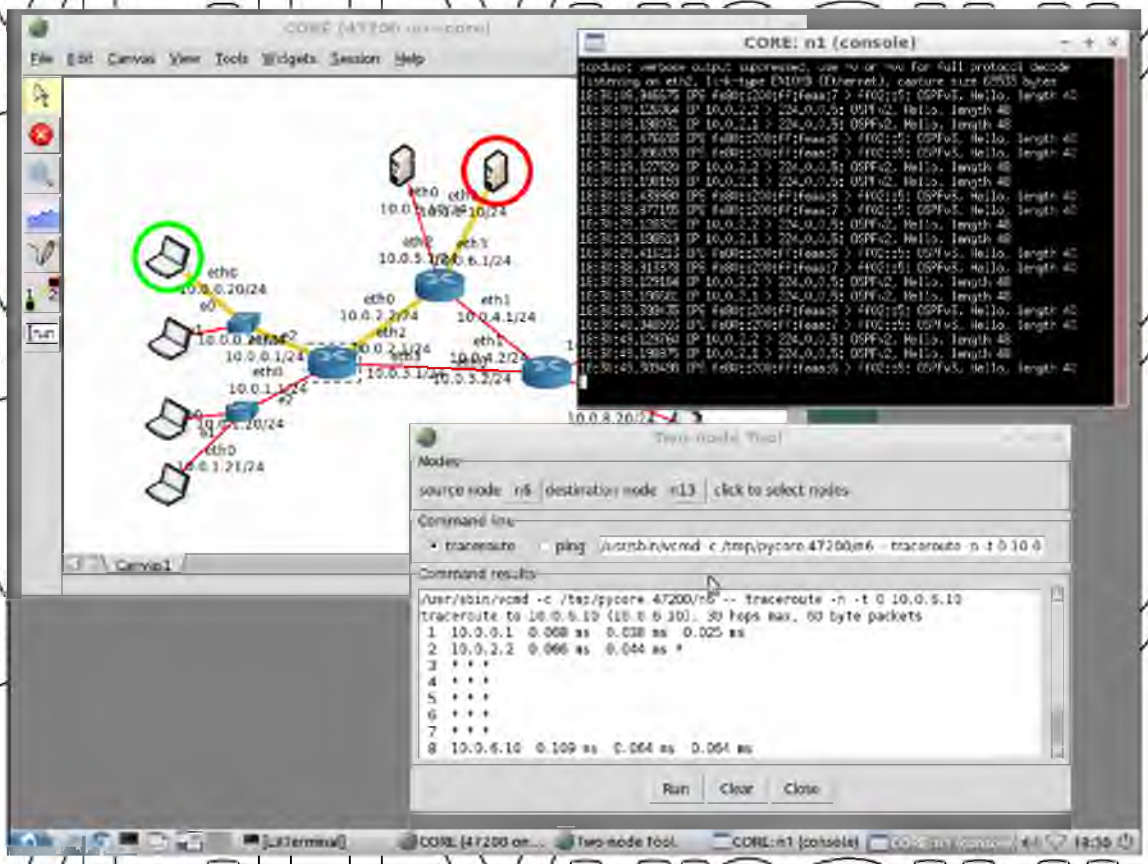

Рисунок 1.7 – Інтерфейс CORE

1.3.5 EVE-NG

EVE-NG - мережевий емулятор, який підтримує віртуалізовані образи комерційних маршрутизаторів (таких як Cisco і NOKIA) і маршрутизаторів з відкритим вихідним кодом. Він використовує Dynamips та IOS-on-Linux для підтримки образів маршрутизаторів та комутаторів Cisco, а також KVM/QEMU

для підтримки всіх інших пристроїв. Він доступний у вигляді образу віртуальної машини, а також може бути встановлений на виділеному сервері під управлінням Ubuntu Linux. Інтерфейс даного емулятора зображено на рис. 1.8.

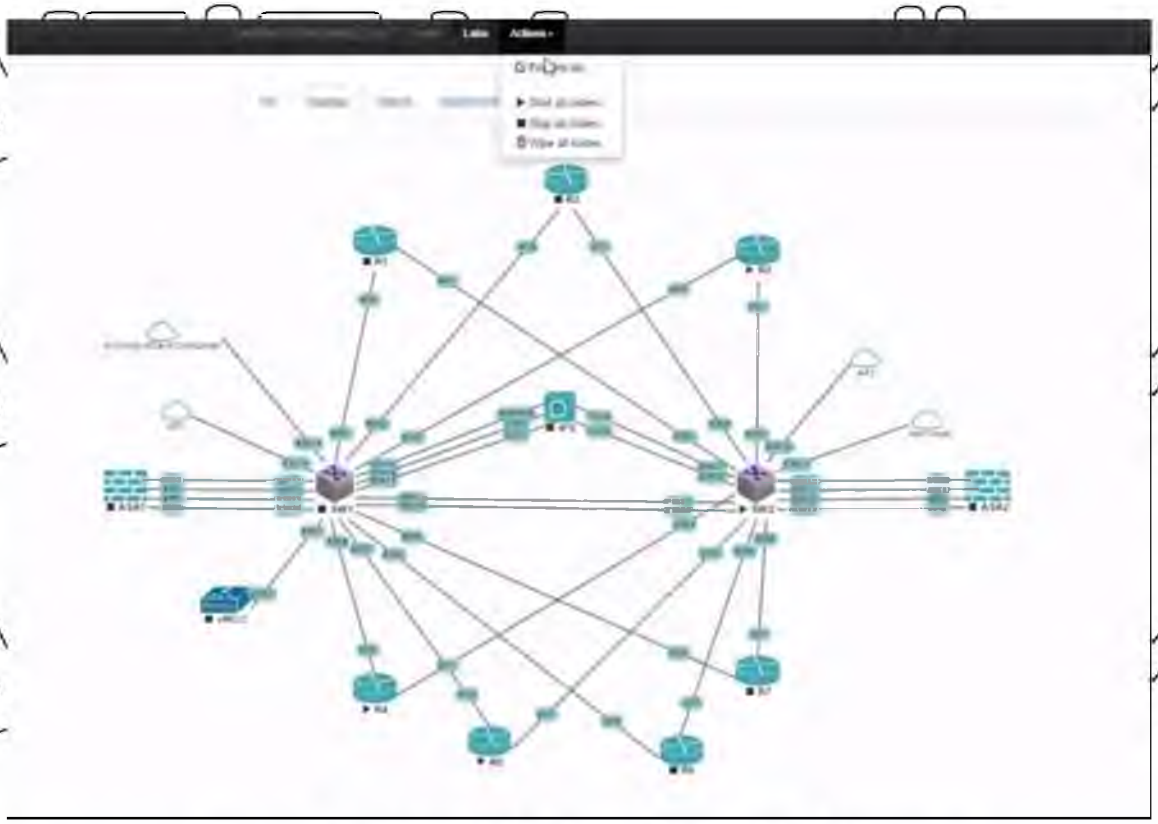

Рисунок 1.8 – Інтерфейс EVE-NG

1.3.6 GNS3

GNS3 - це графічний мережевий симулятор, орієнтований в основному на підтримку програмного забезпечення Cisco та Juniper. GNS3 має велику базу користувачів, що складається в основному з людей, які готуються до іспитів Cisco, і в Інтернеті є багато інформації у вільному доступі про використання GNS3 для моделювання обладнання Cisco.

GNS3 також можна використовувати для моделювання мережі, що складається виключно з віртуальних машин VirtualBox та/або Qemu, на яких працює програмне забезпечення з відкритим вихідним кодом. GNS3 надає безліч готових віртуальних пристроїв з відкритим вихідним кодом, і

користувачі можуть створювати свої власні. Інтерфейс даного емулятора зображено на рис. 1.9.

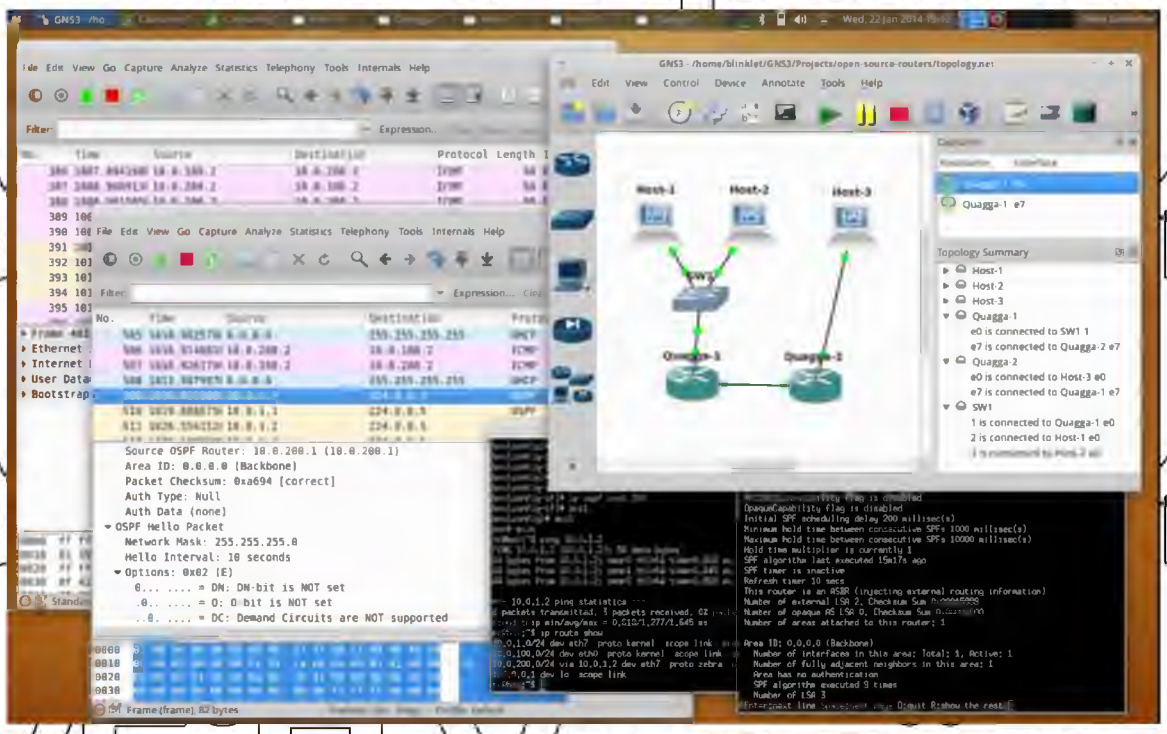

Рисунок 1.9 – Інтерфейс GNS3

1.3.7 IMUNES

Група дослідників із Загребського університету розробила Інтегрований багатопротокольний мережевий емулятор/симулятор (IMUNES) для використання в якості інструменту мережевих досліджень. IMUNES працює під управлінням операційних систем FreeBSD та Linux. Він використовує технологію віртуалізації мережевого стеку на рівні ядра, що надається FreeBSD. Він використовує контейнери Docker і Open vSwitch в Linux.

IMUNES підтримує графічний інтерфейс користувача. Він добре працює і пропонує хорошу продуктивність, навіть при запуску IMUNES у віртуальній машині VirtualBox. Інтерфейс даного емулятора зображено на рис. 1.10.

1.3.8 Kathara

Kathara - це нова версія Netkit, реалізована з використанням сучасних технологій, таких як Docker, та зворотно сумісна з лабораторіями Netkit. На

сайті проекту Netkit є довгий перелік цікавих лабораторних сценаріїв для в дпрацювання з документацією до кожного сценарію. Інтерфейс даного емулятора зображено на рис. 1.11.

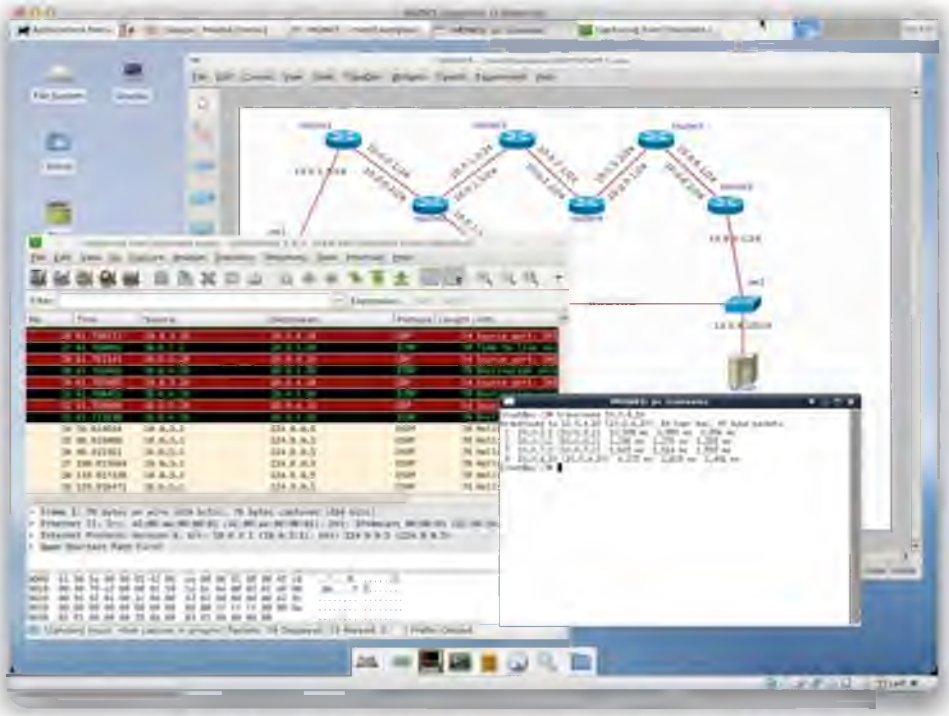

Рисунок 1.10 – Інтерфейс IMUNES

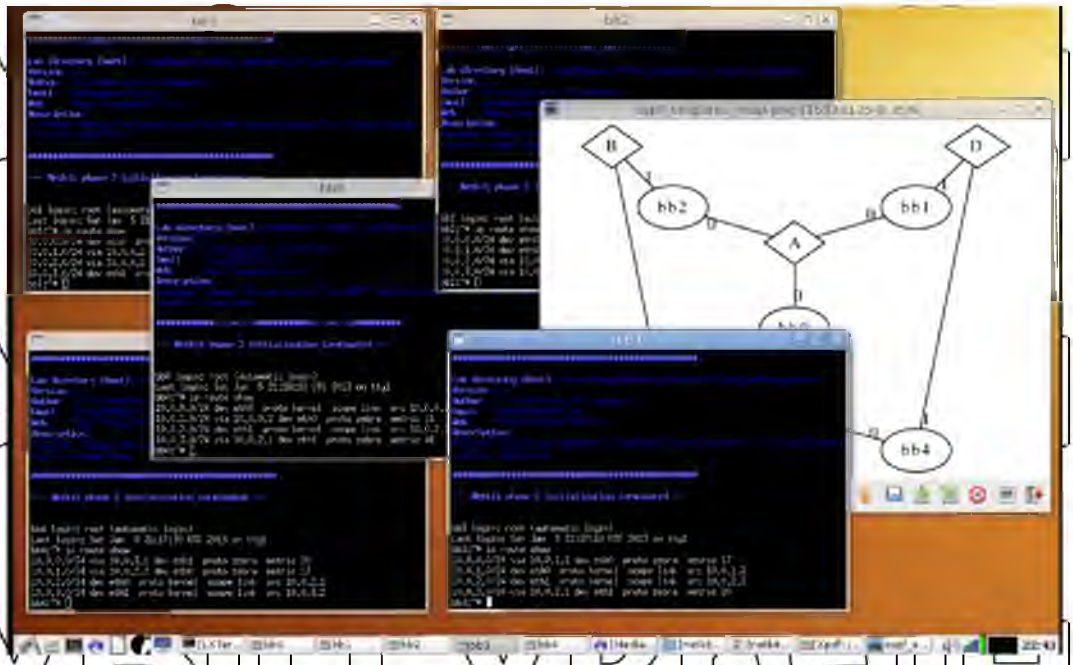

Рисунок 1.11 – Інтерфейс Kathara

1.3.9 Mininet

Mininet призначений для підтримки досліджень в області програмно-визначених мережевих технологій. Він використовує мережев простори імен Linux як технологію віртуалізації для створення віртуальних вузлів. На сайті зазначено, що інструмент може підтримувати тисячі віртуальних вузлів на одній операційній системі. Mininet найбільш корисний для дослідників, які створюють контролери SDN і потребують інструменту для перевірки поведінки і продуктивності контролерів SDN. Знання мови сценаріїв Python дуже корисно при використанні Mininet[7]. Інтерфейс даного емулятора зображено на рис. 1.12.

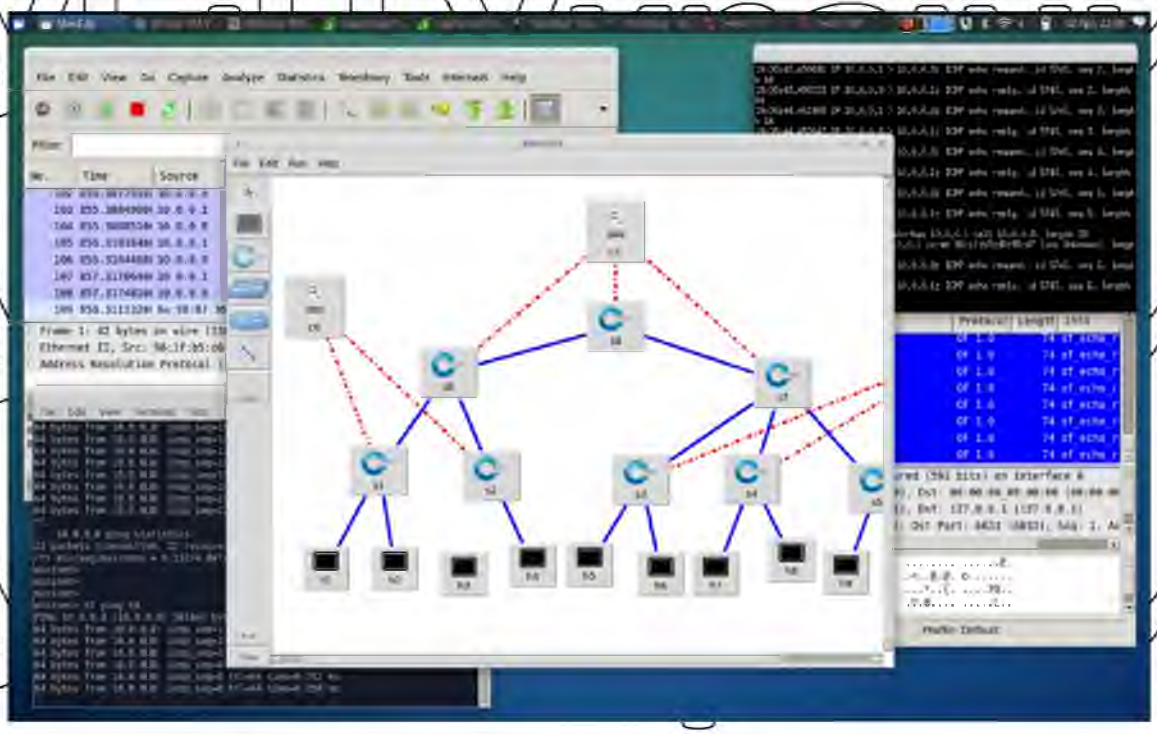

Рисунок 1.12 – Інтерфейс Mininet

2 ВИМОГИ ДО СИСТЕМИ

2.1 Продуктивність (пропускна здатність, затримка передачі)

Як і від будь-якої комп'ютерної системи, від комп'ютерних мереж очікується хороша продуктивність. Це пов'язано з тим, що ефективність обчислень, розподілених по мережі, часто безпосередньо залежить від ефективності, з якою мережа доставляє дані для обчислень. Хоча стара приказка в програмуванні "спочатку зроби правильно, а потім зроби швидко" залишається вірною, в мережах часто необхідно "проектувати для продуктивності". Тому важливо розуміти різні фактори, які впливають на продуктивність мережі.

Продуктивність мережі вимірюється двома основними способами: пропускною здатністю і затримкою. Пропускна здатність мережі визначається кількістю бітів, які можуть бути передані по мережі за певний проміжок часу. Наприклад, мережа може мати пропускну здатність 10 мільйонів біт/с (Мбіт/с), що означає, що вона здатна передавати 10 мільйонів біт щосекунди. Іноді корисно думати про пропускну здатність з точки зору того, скільки часу потрібно для передачі кожного біта даних. Наприклад, у мережі зі швидкістю 10 Мбіт/с передача кожного біта займає 0,1 мікросекунди (мкс).

Пропускна здатність може мати два значення – Bandwidth (ширина смуги пропускання) і Throughput (Пропускна спроможність). Це дещо різні терміни. Перш за все, ширина смуги пропускання (Bandwidth) - це буквально міра ширини смуги частот. Наприклад, застарілі телефонні лінії голосового зв'язку підтримували діапазон частот від 300 до 3300 Гц, про них говорили, що вони мають смугу пропускання 3300 Гц - 300 Гц = 3000 Гц. Якщо слово "пропускна здатність" використовується в ситуації, коли вона вимірюється в герцах, то це, ймовірно, відноситься до діапазону сигналів, які можуть бути прийняті.

Коли ми говоримо про пропускну здатність лінії зв'язку, ми зазвичай маємо на увазі кількість біт в секунду, які можуть бути передані по лінії. Це також іноді називають швидкістю передачі даних. Можна сказати, що пропускна здатність

каналу Ethernet становить 10 Мбіт/с. Однак можна також провести корисну різницю між максимальною швидкістю передачі даних, яка доступна на лінії зв'язку, і кількістю біт в секунду, яку ми можемо фактично передати по лінії зв'язку на практиці. Ми схильні використовувати слово "пропускна здатність" для позначення вимірюваної продуктивності системи. Таким чином, через різні неефективності реалізації, пара вузлів, з'єднаних каналом з пропускною спроможністю 10 Мбіт/с, може досягти пропускної спроможності лише 2 Мбіт/с.

Хоча можна говорити про пропускну здатність мережі в цілому, іноді потрібно бути більш точним, зосередившись, наприклад, на пропускній здатності одного фізичного каналу або логічного каналу між процесами. На фізичному рівні пропускна здатність постійно покращується, і кінця-краю цьому не видно. Інтуїтивно, якщо уявити секунду часу як відстань, яку можна виміряти лінійкою, а пропускну здатність - як кількість бітів, що вміщуються на цій відстані, то можна уявити кожен біт як імпульс певної ширини. Наприклад, кожен біт у каналі зі швидкістю 1 Мбіт/с має ширину 1 мкс, а кожен біт у каналі зі швидкістю 2 Мбіт/с - 0,5 мкс, як показано на рис. 2.1. Чим досконаліша технологія передачі і прийому, тим вужчим може стати кожен біт і, таким чином, тим вища пропускна здатність. Для логічних каналів між процесами на пропускну здатність також впливають інші фактори, включаючи те, скільки разів програмне забезпечення, яке реалізує канал, має обробляти і, можливо, трансформувати кожен біт даних.

Другий показник продуктивності, затримка, відповідає тому, скільки часу потрібно, щоб повідомлення пройшло від одного кінця мережі до іншого. (Як і у випадку з пропускною здатністю, ми можемо зосередитися на затримці одного каналу або наскрізного каналу). Затримка вимірюється строго в термінах часу. Наприклад, трансконтинентальна мережа може мати затримку в 24 мілісекунд (мс); це означає, що повідомленню потрібно 24 мс, щоб пройти шлях від одного узбережжя Північної Америки до іншого. Існує багато ситуацій, в яких важливіше знати, скільки часу потрібно, щоб відправити повідомлення з одного кінця мережі на інший і назад, а не час затримки в один кінець чи бік. Цей показник називається RTT (round-trip time) – час роботи мережі в обидва кінці.

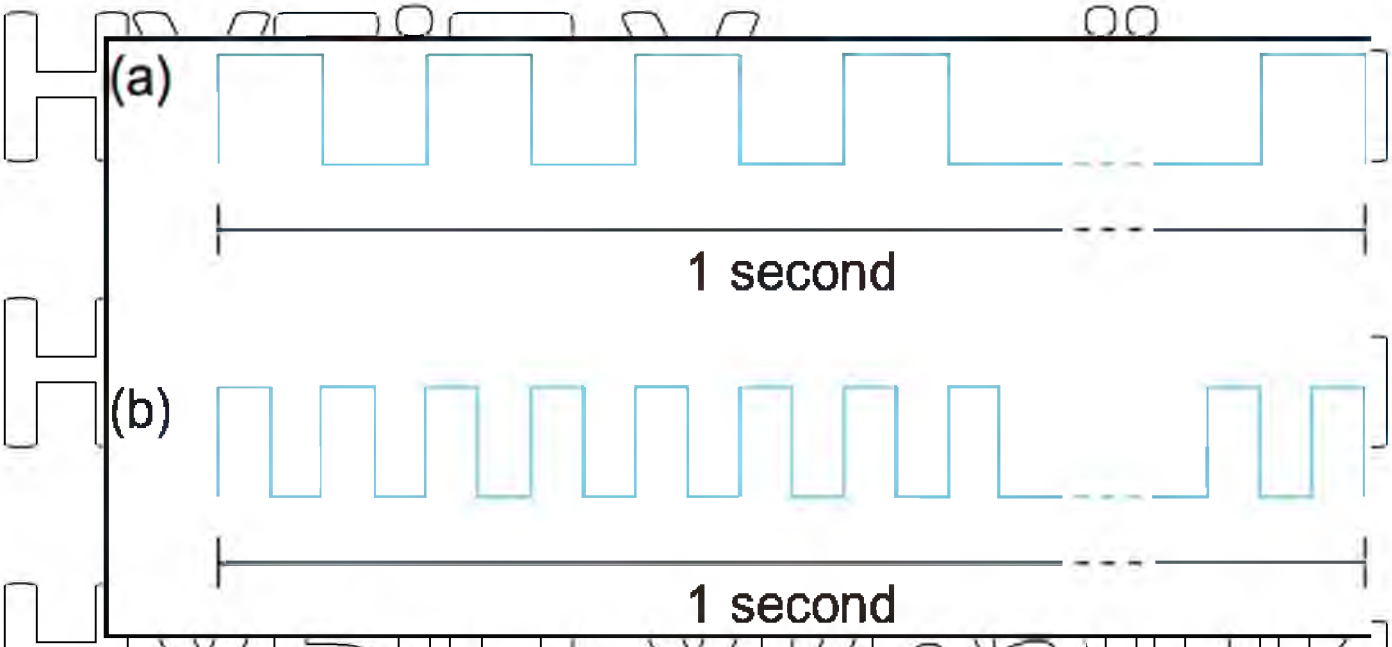

Рисунок 2.1 Біти, що передаються в певній смузі пропускання: (a) біти, що передаються зі швидкістю 1 Мбіт/с (кожен біт має ширину 1 мікросекунду); (b) біти, що передаються зі швидкістю 2 Мбіт/с (кожен біт має ширину 0,5 мікросекунди).

Затримка в КМ має три складові. По-перше, це затримка зі швидкістю поширення світла. Ця затримка виникає тому, що ніщо, включаючи біт на дроті, не може рухатися швидше за швидкість світла. Якщо ви знаєте відстань між двома точками, ви можете розрахувати затримку швидкості світла, хоча ви повинні бути обережними, тому що світло рухається через різні середовища з різною швидкістю: У вакуумі воно рухається зі швидкістю 3,0 × 108 м/с, в мідному кабелі - 2,3 × 108 м/с, а в оптичному волокні - 2,0 × 108 м/с. По-друге, є час, необхідний для передачі одиниці даних. Це функція пропускної здатності мережі та розміру пакета, в якому передаються дані. По-третє, можуть виникати затримки в черзі всередині мережі, оскільки пакетні комутатори, як правило, повинні зберігати пакети протягом деякого часу, перш ніж пересилати їх на вихідний канал. Таким чином, ми можемо визначити загальну затримку:

$$L = P + T + Q, \quad (2.1)$$

де L – затримка (latency),P – поширення (propagation), T – передача (transmit), Q – черга (queue).

В свою чергу поширення визначається:

P = D + S, (2.2)

де P – поширення (propagation), D – довжина дроту, по якому будуть передаватися дані, S – ефективна швидкість світла по цьому дроту.

А формула передачі визначається наступним чином:

T = S + B, (2.3)

де T – передача (transmit), S – це розмір пакета (size), а B – це смуга пропускання, на якій передається пакет (bandwidth).

Якщо повідомлення містить лише один біт і мова йде про один канал зв'язку (а не про цілу мережу), то терміни Transmit і Queue не мають значення, а latency відповідає лише затримці розповсюдження.

Смуга пропускання і затримка в сукупності визначають характеристики продуктивності даного каналу або лінії зв'язку. Їх відносна важливість, однак, залежить від застосування. Для деяких додатків затримка домінує над пропускною здатністю. Наприклад, клієнт, який відправляє 1-байтне повідомлення на сервер і отримує у відповідь 1-байтне повідомлення, обмежений затримкою. Припускаючи, що підготовка відповіді не пов'язана з серйозними обчисленнями, додаток буде працювати набагато інакше на трансконтинентальному каналі з 100 мс RTT, ніж на міжкімнатному каналі з 1 мс RTT. Однак, чи є канал 1 Мбіт/с або 100 Мбіт/с відносно несуттєвим, оскільки в

першому випадку час передачі байта (Transmit) становить 8 мкс, а в другому - Transmit = 0,08 мкс.

На противагу цьому розглянемо програму електронної бібліотеки, яку просять отримати зображення розміром 25 мегабайт (Мб) - чим більша пропускна здатність каналу, тим швидше вона зможе повернути зображення користувачеві. Тут пропускна здатність каналу домінує над продуктивністю. Щоб переконатися в цьому, припустимо, що канал має пропускну здатність 10 Мбіт/с. Передача зображення займе 20 секунд (25 × 106 × 8 біт / (10 × 106 Мбіт/с = 20 секунд), що робить відносно неважливим, чи знаходиться зображення на іншій стороні каналу 1 мс або 100 мс; різниця між часом відгуку 20,001 секунди і часом відгуку 20.1 секунди незначна.

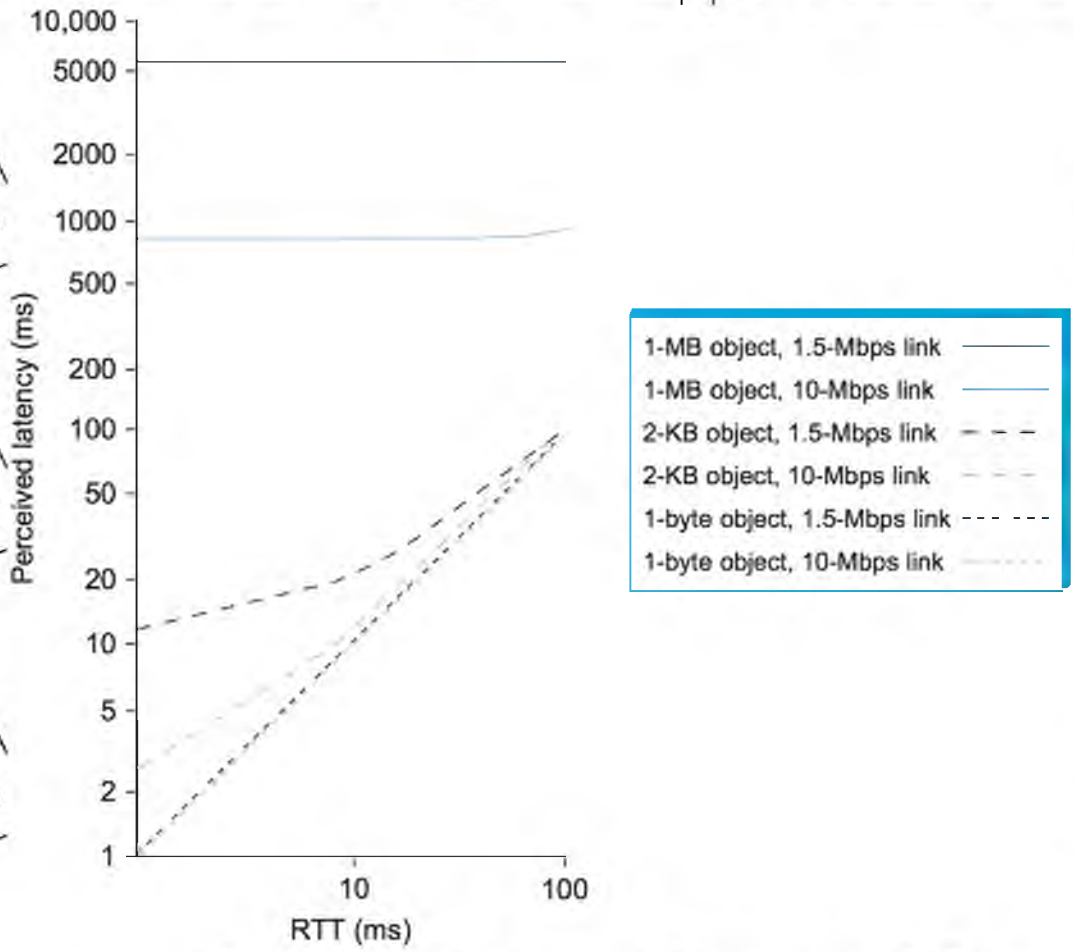

Рис 2.2 Графік переміщення об'єктів різного розміру

Рис 2.2 дає уявлення про те, як затримка або пропускна здатність можуть домінувати над продуктивністю в різних обставинах. Графік показує, скільки часу потрібно для переміщення об'єктів різного розміру (1 байт, 2 КБ, 1 МБ) через мережі з часом очікування від 1 до 100 мс і швидкістю зв'язку 1,5 або 10 Мбіт/с. Використано логарифмічні шкали, щоб показати відносну продуктивність. Для 1-байтового об'єкта (скажімо, натискання клавіші) затримка залишається майже точно рівною RTT, так що не можна відрізнити мережу зі швидкістю 1,5 Мбіт/с від мережі зі швидкістю 10 Мбіт/с. Для об'єкта розміром 2 КБ (наприклад, повідомлення електронної пошти) швидкість з'єднання має велике значення в мережі 1 мс RTT, але незначну різницю в мережі 100 мс RTT. А для об'єкта розміром 1 Мб (наприклад, цифрового зображення) RTT не має ніякого значення - саме швидкість з'єднання домінує над продуктивністю у всьому діапазоні RTT.

2.2 Надійність і безпека

Мережева безпека визначається як процес створення стратегічного оборонного підходу, який захищає дані компанії та її ресурси в мережі. Вона захищає організацію від будь-якої форми потенційної загрози або несанкціонованого доступу. Незалежно від розміру організації, галузі або інфраструктури, рішення мережевої безпеки захищають її від постійно зростаючої загрози кібератак. Розглянемо види атак на мережу.

2.2.1 Шкідливе програмне забезпечення

Шкідливе програмне забезпечення є одним з найшвидших способів поширення шкідливих атак. Воно створюється спеціально для знищення цілі та отримання несанкціонованого доступу до системи. Шкідливе програмне забезпечення здебільшого самовідтворюється, а оскільки воно подорожує Інтернетом, то отримує доступ до всіх комп'ютерів, підключених до мережі. Об'єктом атаки можуть бути і зовнішні пристрої, підключені до мережі. Існує три

основні типи векторів атаки шкідливого програмного забезпечення, які зображені на рис. 2.3.

Рисунок 2.3 Типи векторів атаки шкідливого програмного забезпечення

Вірус - це шкідливе програмне забезпечення, прикріплене до документа або файлу, яке підтримує макроси для виконання свого коду і поширюється від комп'ютера до комп'ютера. Після завантаження вірус перебуває в сплячому режимі до тих пір, поки файл не буде відкрито і не буде використано. Віруси призначені для того, щоб порушити здатність системи працювати. Як наслідок, віруси можуть спричинити значні операційні проблеми та втрату даних.

Хробаки - це шкідливе програмне забезпечення, яке швидко розмножується і поширюється на будь-який пристрій в мережі. На відміну від вірусів, хробакам не потрібні програми-хости для розповсюдження. Хробак заражає пристрій через завантажений файл або мережеве з'єднання, перш ніж він розмножується і поширюється з експоненціальною швидкістю. Як і віруси, хробаки можуть серйозно порушити роботу пристрою та призвести до втрати даних.

Троянські віруси маскуються під корисне програмне забезпечення. Але як тільки користувач завантажує його, троянський вірус може отримати доступ до

конфіденційних даних, а потім змінити, заблокувати або видалити ці дані. Це може бути надзвичайно шкідливим для роботи пристрою. На відміну від звичайних вірусів і хробаків, троянські віруси не призначені для самовідтворення.

Розглянемо, ще деякі види шкідливого програмного забезпечення:

- програмне забезпечення з вимогою викупу - це програмне забезпечення, яке використовує шифрування для відключення доступу жертви до її даних до моменту сплати викупу. Організація-жертва стає частково або повністю нездатною працювати до тих пір, поки не заплатить, але немає ніяких гарантій, що оплата призведе до отримання необхідного ключа дешифрування або що наданий ключ дешифрування буде функціонувати належним чином;
- без файлове шкідливе програмне забезпечення - це різновид резидентного в пам'яті шкідливого програмного забезпечення. Як випливає з терміну, це шкідливе програмне забезпечення, яке працює з пам'яті комп'ютера жертви, а не з файлів на жорсткому диску. Оскільки немає файлів для сканування, його важче виявити, ніж традиційне шкідливе програмне забезпечення. Це також ускладнює криміналістичну експертизу, оскільки шкідливе програмне забезпечення зникає після перезавантаження комп'ютера жертви;
- шпигунське програмне забезпечення - це шкідливе програмне забезпечення, яке таємно працює на комп'ютері та звітує віддаленому користувачеві. Замість того, щоб просто порушувати роботу пристрою, шпигунські програми націлені на конфіденційну інформацію і можуть надавати віддалений доступ зловмисникам. Шпигунське програмне забезпечення часто використовується для крадіжки фінансової або особистої інформації. Особливим типом шпигунських програм є кейлоггер, який записує натискання клавіш, щоб розкрити паролі та особисту інформацію;

- рекламне програмне забезпечення - це шкідливе програмне забезпечення, яке використовується для збору даних про використання Вашого комп'ютера та надання Вам відповідної реклами. Хоча рекламне ПЗ не завжди є небезпечним, в деяких випадках воно може спричинити проблеми для Вашої системи. Рекламне ПЗ може перенаправляти Ваш браузер на небезпечні сайти і навіть може містити троянські програми та шпигунські програми. Крім того, значний рівень рекламного ПЗ може помітно сповільнити роботу системи. Оскільки не всі рекламні програми є шкідливими, важливо мати захист, який постійно і розумно сканує ці програми;

- руткіт - це програмне забезпечення, яке надає зловмисникам віддалений контроль над комп'ютером жертви з повними адміністративними привілеями. Руткіти можуть бути впроваджені в додатки, ядра, гіпервізори або вбудоване програмне забезпечення. Вони поширюються через фішинг, шкідливі вкладення, шкідливі завантаження та скомпрометовані спільні диски. Руткіти також можуть використовуватися для приховування інших шкідливих програм, таких як кейлоггери;

- шифрувальник - це тип шкідливого програмного забезпечення з єдиною метою: стерти дані користувача та унеможливити їх відновлення. Вони використовуються для виведення з ладу комп'ютерних мереж в державних або приватних компаніях різних секторів. Зловмисники також використовують їх для приховування слідів, залишених після вторгнення, що послаблює здатність жертви до реагування[9];

2.2.2 Відмова в обслуговуванні (DoS) та розподілена відмова в обслуговуванні (DDoS)

Атака типу "відмова в обслуговуванні" (Denial-of-Service, DoS) - це атака, метою якої є вимкнення комп'ютера або мережі, що робить їх недоступними для цільових користувачів. DoS-атаки досягають цього шляхом переповнення цілі

трафіком або надсилання їй інформації, яка спричиняє збій у роботі. В обох випадках DoS-атака позбавляє законних користувачів (тобто співробітників, членів або власників облікових записів) послуг або ресурсів, на які вони розраховували.

Жертвами DoS-атак часто стають веб-сервери відомих організацій, таких як банківські, комерційні та медіа-компанії, урядові та торгові організації. Хоча DoS-атаки, як правило, не призводять до крадіжки або втрати значної інформації чи інших активів, вони можуть коштувати жертві значних витрат часу та грошей на їх усунення.

DDos-атака – це та сама DoS атака, але яка відбувається з багатьох джерел. Ефективність DDoS-атак досягається за рахунок використання декількох скомпрометованих комп'ютерних систем як джерел атакуючого трафіку. Експлуатовані машини можуть включати комп'ютери та інші мережеві ресурси, такі як пристрої Інтернету речей.

З високого рівня DDoS-атака схожа на несподівану пробку, яка забиває шосе, перешкоджаючи регулярному руху транспорту до місця призначення.

DDoS-атаки здійснюються з використанням мереж підключених до Інтернету комп'ютерів.

Ці мережі складаються з комп'ютерів та інших пристроїв (наприклад, пристроїв Інтернету речей), які були заражені шкідливим програмним забезпеченням, що дозволяє зловмиснику дистанційно керувати ними. Ці окремі пристрої називаються ботами (або зомбі), а група ботів - ботнетом.

Після створення ботнету зловмисник може керувати атакою, надсилаючи віддалені інструкції кожному боту.

Коли ботнет атакує сервер або мережу жертви, кожен бот надсилає запити на IP-адресу цілі, що потенційно може призвести до перевантаження сервера або мережі, що спричинить відмову в обслуговуванні звичайного трафіку.

Оскільки кожен бот є законним інтернет-пристроєм, відокремити атакуючий трафік від звичайного може бути складно.

Найбільш очевидним симптомом DDoS-атаки є раптова повільна робота або недоступність сайту або сервісу. Але оскільки ряд причин - наприклад, законний сплеск трафіку - може створювати подібні проблеми з продуктивністю, зазвичай потрібне подальше розслідування. Інструменти аналізу трафіку можуть допомогти вам виявити деякі з цих ознак DDoS-атаки:

- підозрілі обсяги трафіку, що надходять з однієї IP-адреси або діапазону IP-адрес;
- потік трафіку від користувачів, які мають спільний поведінковий профіль, наприклад, тип пристрою, геолокацію або версію веб-браузера;
- незрозумілий сплеск запитів до однієї сторінки або кінцевої точки;
- дивні моделі трафіку, такі як сплески в непарні години дня або моделі, які виглядають неприродно (наприклад, сплеск кожні 10 хвилин).

Існують й інші, більш специфічні ознаки DDoS-атаки, які можуть відрізнятися в залежності від типу атаки.

Різні типи DDoS-атак націлені на різні компоненти мережевого з'єднання. Для того, щоб зрозуміти, як працюють різні DDoS-атаки, необхідно знати, як здійснюється мережеве з'єднання.

Мережеве з'єднання в Інтернеті складається з багатьох різних компонентів або "шарів". Як і при будівництві будинку з нуля, кожен шар в моделі має своє призначення.

Модель OSI, показана на рис. 2.4, є концептуальною основою, яка використовується для опису мережевого з'єднання на 7 різних рівнях.

Хоча майже всі DDoS-атаки пов'язані з перевантаженням цільового пристрою або мережі трафіком, атаки можна розділити на три категорії. Зловмисник може використовувати один або декілька різних векторів атаки, або циклічно використовувати вектори атаки у відповідь на контрзаходи, що вживаються ціллю.

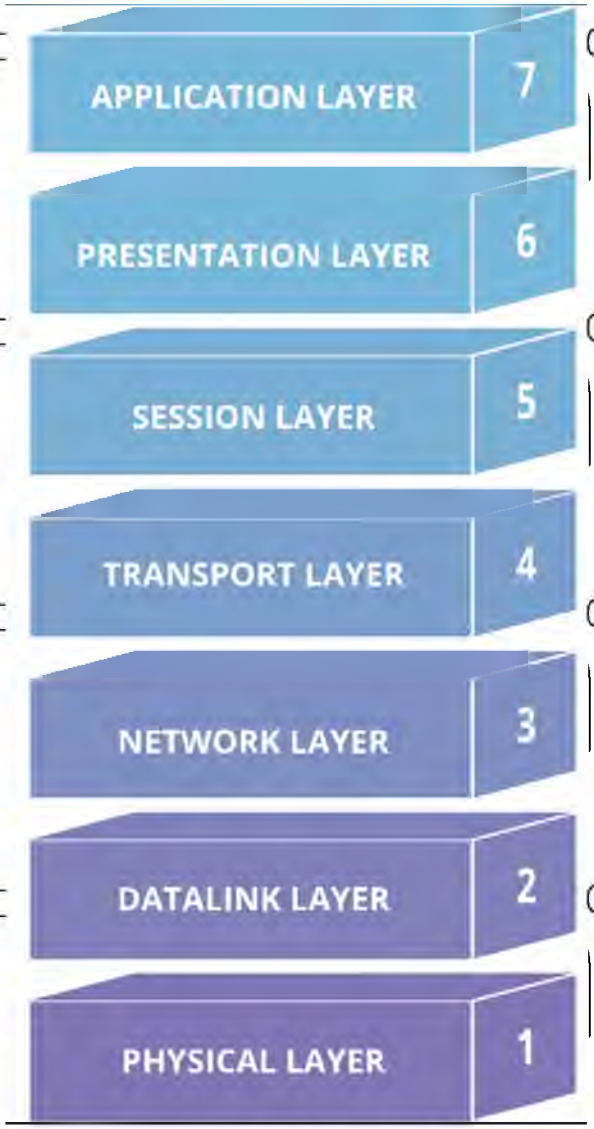

Рисунок 2.4 Модель OSI

Від атак 7-го рівня важко захиститися, оскільки може бути важко відрізнити зловмисний трафік від легітимного, метою цих атак є виснаження ресурсів цілі для створення відмови в обслуговуванні.

Атаки націлені на рівень, де веб-сторінки генеруються на сервері і доставляються у відповідь на HTTP-запити. Один HTTP-запит є дешевим в обчислювальному плані для виконання на стороні клієнта, але може бути дорогим для цільового сервера, оскільки сервер часто завантажує кілька файлів і виконує запити до бази даних для того, щоб створити веб-сторінку.

Протокольні атаки, також відомі як атаки на виснаження стану, викликають збій в роботі сервісу шляхом надмірного споживання ресурсів

сервера та/або ресурсів мережевого обладнання, такого як брандмауери та балансувальники навантаження.

Протокольні атаки використовують слабкі місця в 3 і 4 рівнях стека протоколів, щоб зробити ціль недоступною.

Також існують об'ємні атаки. Ця категорія атак намагається створити перевантаження, споживаючи всю доступну пропускну здатність між ціллю та великою мережею Інтернет. Великі обсяги даних надсилаються на ціль за допомогою ампліфікації або інших засобів створення масового трафіку, таких як запити з бот-мережі[10].

2.2.3 Атака "людина посередині" (man-in-the-middle, MitM)

Атака "людина посередині" (MitM) - це тип кібератаки, під час якої перехоплюється зв'язок між двома сторонами, часто з метою викрадення облікових даних для входу в систему або особистої інформації, шпигунства за жертвами, саботажу зв'язку або пошкодження даних.

Метою атаки є викрадення особистої інформації, такої як облікові дані для входу в систему, реквізити рахунків і номери кредитних карток. Цільовими аудиторіями, як правило, є користувачі фінансових додатків, SaaS-бізнесу, сайтів електронної комерції та інших веб-сайтів, де необхідний вхід в систему.

Інформація, отримана під час атаки, може бути використана для багатьох цілей, включаючи крадіжку особистих даних, несанкціоновані перекази коштів або незаконну зміну пароля.

У широкому сенсі, MITM-атака еквівалентна тому, що листоноша відкриває вашу банківську виписку, записує дані вашого рахунку, а потім запечатує конверт і доставляє його до ваших дверей.

Розглянемо види MITM-атак:

- підслуховування Wi-Fi;
- підміна DNS;
- підміна IP-адрес;
- підміна HTTPS;

- підробка ARP;
- злом електронної пошти;
- злом сесії;
- зняття SSL-шифрування.

2.2.3.1 Підслуховування Wi-Fi

Можливо, ви бачили повідомлення "Це з'єднання не є безпечним", якщо користувалися пристроєм у кафе. Громадський Wi-Fi, як правило, пропонується "як є", без будь-яких обіцянок щодо якості послуг.

За незашифрованими мережами Wi-Fi легко спостерігати. Хоча, це як дебати в громадському місці - будь-хто може приєднатися до них. Ви можете обмежити свій доступ, встановивши на своєму комп'ютері режим "загальнодоступний", який відключає функцію виявлення мережі. Це дозволяє уникнути використання системи іншими користувачами в мережі.

Інший вид атаки Wi-Fi підслуховування відбувається, коли зловмисник встановлює власну точку доступу Wi-Fi "Злий двійник". Зловмисник робить посилання, через мережеву адресу та паролі, ідентичними до реальних. Користувачі ненавмисно або автоматично переходять на "злого двійника", що дозволяє зловмиснику контролювати їхні дії.

2.2.3.2 Підміна DNS

Підміна DNS, також відома як отруєння кешу DNS, передбачає проникнення на DNS-сервер і зміну адресного запису веб-сайту. В результаті користувачі, які намагаються отримати доступ до сайту, перенаправляються за зміненим записом DNS на сайт зловмисника.

Атака йде на отруєння кешу DNS, при якій зловмисник (IP 192.168.3.300) перехоплює канал зв'язку між клієнтом (IP 192.168.1.100) та комп'ютером-сервером, що належить веб-сайту www.estores.com (IP 192.168.2.200).

У цьому сценарії використовується інструмент (наприклад, arpspoof), який вводить клієнта в оману, що IP-адреса сервера - 192.168.3.300. У той же час, сервер змушений думати, що IP-адреса клієнта також 192.168.3.300.

Такий сценарій виглядає наступним чином:

1. зловмисник за допомогою arpspoof видає команду: arpspoof 192.168.1.100 192.168.2.200. Це модифікує MAC-адреси в ARP-таблиці сервера, змушуючи його думати, що комп'ютер зловмисника належить клієнту;
2. зловмисник знову використовує arpspoof і видає команду: arpspoof 192.168.2.200 192.168.1.100, яка повідомляє клієнту, що комп'ютер зловмисника є сервером;
3. зловмисник видає команду Linux: echo 1> /proc/sys/net/ipv4/ip_forward. В результаті IP-пакети, що обмінюються між клієнтом і сервером, переадресовуються на комп'ютер зловмисника;
4. на локальному комп'ютері зловмисника створюється хост-файл 192.168.3.300 estores.com, який прив'язує сайт www.estores.com до його локального IP;
5. зловмисник встановлює веб-сервер на локальному комп'ютері та створює підроблений веб-сайт, схожий на www.estores.com;
6. використовується інструмент (наприклад, dnsspoof) для перенаправлення всіх DNS-запитів на локальний хост-файл зловмисника. В результаті користувачам відображається підроблений веб-сайт, і лише при взаємодії з ним на їх комп'ютери встановлюється шкідливе програмне забезпечення.

2.2.3.3 Підміна IP-адрес

IP-спуфінг - це створення пакетів Інтернет-протоколу (IP), які мають змінену адресу джерела з метою або приховати особу відправника, або видати

себе за іншу комп'ютерну систему, або зробити і те, і інше. Цей метод часто використовується зловмисниками для здійснення DDoS-атак на цільовий пристрій або навколишню інфраструктуру.

Надсилання та отримання IP-пакетів є основним способом зв'язку між комп'ютерами та іншими пристроями, об'єднаними в мережу, і становить основу сучасного Інтернету. Всі IP-пакети містять заголовок, який передує тілу пакета і містять важливу інформацію про маршрутизацію, включаючи адресу джерела. У звичайному пакеті IP-адреса джерела - це адреса відправника пакета. Якщо пакет був підроблений, то адреса джерела буде підроблена.

Підміна IP-адреси аналогічна тому, як зловмисник надсилає посилку комусь із вказаною неправильною зворотною адресою. Якщо особа, яка отримує пакет, хоче зупинити відправника від надсилання пакетів, блокування всіх пакетів з фальшивої адреси не принесе користі, оскільки зворотна адреса легко змінюється. Відповідно, якщо одержувач захоче відповісти на зворотну адресу, його пакет-відповідь піде кудись, а не справжньому відправнику. Можливість підробки адрес пакетів є основною вразливістю, яка використовується багатьма DDoS-атаками.

DDoS-атаки часто використовують підробку з метою перевантажити ціль трафіком, одночасно маскуючи ідентичність зловмисного джерела, запобігаючи зусиллям по усуненню наслідків. Якщо IP-адреса джерела підроблена і постійно рандомізується, блокування зловмисних запитів стає складним завданням. Спуфінг IP-адрес також ускладнює роботу правоохоронних органів і команд кібербезпеки з відстеження виконавця атаки.

Спуфінг також використовується для маскування під інший пристрій, щоб відповіді надсилалися на цільовий пристрій. Об'ємні атаки, такі як NTP-ампліфікація і DNS-ампліфікація, використовують цю вразливість. Можливість модифікації вихідної IP-адреси притаманна дизайну TCP/IP, що робить її постійною проблемою безпеки.

Аналогічно до DDoS-атак, підміна може також здійснюватися з метою маскування під інший пристрій, щоб обійти автентифікацію та отримати доступ до сеансу користувача або "викрасти" його.

2.2.3.4 Підміна HTTPS

Дублювання HTTPS-сторінки наразі неможливе.

Однак, теоретичний підхід до обходу HTTPS був проілюстрований експертами з кібербезпеки. Зловмисник створює авторитетну адресу.

Він використовує літери міжнародних алфавітів, а не стандартні скрипти. Це діє як фішингові електронні листи з незвичайними символами, які ви могли використовувати.

2.2.3.5 Підробка ARP

Протокол дозволу адрес (Address Resolution Protocol, ARP) - це протокол, який дозволяє мережевому зв'язку досягти певного пристрою в мережі. ARP переводить адреси Інтернет-протоколу (IP) в адреси управління доступом до медіа (MAC) і навпаки. Найчастіше пристрої використовують ARP для зв'язку з маршрутизатором або шлюзом, який дозволяє їм підключатися до Інтернету.

Хости підтримують ARP-кеш, таблицю відповідності між IP-адресами та MAC-адресами, і використовують її для підключення до пунктів призначення в мережі. Якщо хост не знає MAC-адресу для певної IP-адреси, він надсилає пакет ARP-запиту, запитуючи інші машини в мережі про відповідну MAC-адресу.

Протокол ARP не був розроблений для забезпечення безпеки, тому він не перевіряє, чи дійсно відповідь на ARP-запит надходить від уповноваженої сторони. Він також дозволяє хостам приймати ARP-відповіді, навіть якщо вони ніколи не відправляли запит. Це слабке місце в протоколі ARP, яке відкриває двері для атак підміни ARP.

ARP працює тільки з 32-бітними IP-адресами в старому стандарті IPv4. Новіший протокол IPv6 використовує інший протокол, Neighbor Discovery Protocol (NDP), який є безпечним і використовує криптографічні ключі для

перевірки ідентичності хостів. Однак, оскільки більша частина Інтернету все ще використовує старий протокол IPv4, ARP залишається в широкому вжитку.

Атака працює наступним чином:

1. зловмисник повинен мати доступ до мережі. Він сканує мережу, щоб визначити IP-адреси щонайменше двох пристроїв - припустимо, це робоча станція та маршрутизатор;
2. зловмисник використовує інструмент підміни, такий як Arpspoof або Driftnet, для відправки підроблених ARP-відповідей;
3. у підроблених відповідях повідомляється, що правильною MAC-адресою для обох IP-адрес, що належать маршрутизатору і робочій станції, є MAC-адреса зловмисника. Це змушує маршрутизатор і робочу станцію підключатися до машини зловмисника, а не один до одного;
4. обидва пристрої оновлюють свої записи ARP-кешу і з цього моменту спілкуються з зловмисником, а не безпосередньо один з одним;
5. тепер зловмисник таємно знаходиться в центрі всіх комунікацій[11].

2.2.3.6 Злом електронної пошти

Зловмисник використовує систему електронної пошти користувача при такому виді кібернетичного вторгнення. Зловмисник також непомітно слостерігає, збираючи дані та підслуховуючи оговорения через електронну пошту. Зловмисники можуть мати шаблон сканування, який шукає цільові ключові слова, такі як "фінансові" або "прихована політика демократії".

Завдяки соціальній інженерії злом електроної пошти працює бездоганно. Щоб імітувати онлайн-друга, зловмисники можуть використовувати відповідні дані з якоїсь викраденої адреси електронної пошти. Spear-фішинг також може бути використаний для того, щоб обманом змусити користувача завантажити шкідливі програми.

2.2.3.7 Злом сесії

Зазвичай ця форма MITM-атаки часто використовується для злому соціальних медіа-платформ. Веб-сторінка містить "файл cookie сеансу браузера" на комп'ютері жертви для більшості платформ соціальних мереж. Якщо людина відходить, цей файл cookie спростовується. Але коли сеанс працює, файл cookie пропонує ідентифікаційні дані, дані про експозицію та дані моніторингу.

Викрадення сеансу відбувається, коли зловмисник викрадає файл cookie конфігурації. Якщо обліковий запис жертви не зламаний шкідливим програмним забезпеченням або зловмисниками додатків, він може виникнути. Це може статися, якщо користувач використовує перехресне вторгнення XSS, коли хакер впроваджує шкідливий скрипт на сайт, який часто відвідують.

2.2.3.8 Зняття SSL-шифрування

HTTP і HTTPS є протоколами прикладного рівня в моделі TCP/IP, як показано на малюнку нижче. HTTPS використовує захищений тунель для передачі та отримання даних, який зазвичай називається SSL/TLS (Secure Socket Layer / Transport Layer Security), і тому до HTTPS додається суфікс "S".

SSL/TLS - це захищений протокол, який використовується для передачі конфіденційної інформації. Цей протокол використовується при обміні конфіденційними даними, такими як, наприклад, банківська інформація та електронна кореспонденція. Безпека протоколу забезпечується шляхом створення зашифрованого з'єднання між двома сторонами (зазвичай клієнтською програмою та сервером). Браузери та веб-сервери регулярно використовують цей протокол, коли потрібне безпечне з'єднання. У більшості сценаріїв при встановленні безпечного з'єднання відбуваються наступні події:

Користувач відправляє незахищений HTTP-запит.

Сервер відповідає по HTTP і перенаправляє користувача на захищений протокол (HTTPS).

Користувач надсилає захищений HTTPS-запит, і починається захищений сеанс.

Цей процес забезпечує розумну гарантію як конфіденційності, так і цілісності. Іншими словами, ми не просто шифруємо повідомлення, які ми надсилаємо, ми гарантуємо, що повідомлення, яке ми отримуємо, не буде змінено по дроту.

Для того, щоб "зняти" SSL, зловмисник втручається в перенаправлення HTTP на захищений протокол HTTPS і перехоплює запит від користувача до сервера. Потім зловмисник продовжить встановлювати HTTPS-з'єднання між собою і сервером, а з користувачем - незахищене HTTP-з'єднання, виступаючи в ролі "моста" між ними.

2.3 Розширюваність і масштабованість

Мережеві команди, великі і малі, залежать від широкого вибору програмного забезпечення, щоб випереджати конкурентів і постійно впроваджувати нові концепції та технології для забезпечення цього. Оскільки інновації заощаджують гроші і скорочують час виходу на ринок, бажано уникати технологічних модернізацій. Додавання нової функції до системи мережевої автоматизації вимагає розширення її функціональності новими рисами, що здійснюється шляхом додавання нового коду. Деякі додатки можуть прийняти цю зміну, а інші - ні.

Розширюваність дозволяє програмній системі дозволяти і приймати розширення своїх можливостей без необхідності значного переписування коду або зміни своєї фундаментальної структури. Завдяки розширюваності ви можете додати нову функцію в існуючу систему так само, як шматочок пазла, який вписується в неї, не порушуючи загальної картини.

Розширюваність показує наскільки легко ваше забезпечення може підтримувати "гачки" для нових функціональних можливостей, інтерфейсів, пристроїв, типів введення тощо. Це також може стосуватися того, наскільки

легко ваше забезпечення може підтримувати нові послуги з найменшими втратами/без втрат для клієнтів.

Масштабованість, як властивість систем, як правило, важко визначити, і в кожному конкретному випадку необхідно визначити конкретні вимоги до масштабованості за тими вимірами, які вважаються важливими. Це дуже важливе питання в електронних системах, базах даних, маршрутизаторах і мережах. Система, продуктивність якої покращується після додавання апаратних засобів, пропорційно до доданої потужності, вважається масштабованою системою.

Алгоритм, дизайн, мережевий протокол, програма або інша система вважається масштабованою, якщо вона є достатньо ефективною та практичною при застосуванні до великих ситуацій (наприклад, великий набір вхідних даних, велика кількість вихідних даних або користувачів, або велика кількість вузлів-учасників у випадку розподіленої системи). Якщо проект або система виходить з ладу при збільшенні кількості, він не масштабується. На практиці, якщо є велика кількість речей (n), які впливають на масштабування, то вимоги до ресурсів (наприклад, алгоритмічна часова складність) повинні зростати менше, ніж n2 при збільшенні n. Прикладом може слугувати пошукова система, яка повинна масштабуватися не тільки за кількістю користувачів, але й за кількістю об'єктів, які вона індексує. Масштабованість - це здатність сайту збільшуватися в розмірах відповідно до попиту.

Масштабованість вимірює, наскільки легко ви можете збільшити або зменшити масштаби ваших операцій. Масштабованість мережі означає, наскільки легко ви можете видалити або додати пропускну здатність мережі. Це стосується того, наскільки легко ви можете збільшити пропускну здатність ваших систем. Це гарантує, що ви зможете йти в ногу зі зростаючими вимогами. Масштабованість мережі важлива для організації з цих причин:

– масштабованість мережі гарантує, що ви зможете задовольнити зростаючі вимоги;

– якщо зростання вашого бізнесу буде випереджати пропускну здатність вашої мережі, то ваші клієнти зіткнуться з перебоями. Через це багато клієнтів перейдуть на інші платформи. Важливо переконатися, що ви підтримуєте високу доступність, оскільки це допоможе вам зберегти своїх клієнтів задоволеними;

– якщо ви стикаєтеся з меншим попитом в міжсезоння, ви можете зменшити масштаб своєї мережі для зменшення витрат на IT.

Простими словами, гнучкість IT дуже важлива для сучасного бізнесу. Якщо ваша мережа є гнучкою, то ви можете легко задовольнити вимоги ваших клієнтів. Це допоможе вам контролювати витрати та уникати перебоїв у наданні послуг.

2.4 Прозорість

Прозорість мережі - це процес надсилання або доступу до даних через мережу таким чином, що інформація не є видимою для користувачів, які спілкуються з локальним або віддаленим хостом, системою, мережею або програмним забезпеченням. Вона може надавати віддалені дані та обчислювальні ресурси локальному користувачеві без надання проміжної мережевої інформації.

Прозорість дозволяє користувачеві отримати доступ до ресурсу (наприклад, прикладної програми або даних), при цьому користувачеві не потрібно знати і, як правило, він не знає, чи знаходиться цей ресурс на локальній машині (тобто на комп'ютері, який користувач використовує в даний момент) або на віддаленій машині (тобто на комп'ютері, розташованому в іншому місці в мережі).

Таким чином, наприклад, коли користувач відкриває каталог або файл, натиснувши на іконку (тобто невелике зображення), що з'являється на екрані дисплея, вміст каталогу може знаходитися на тому ж комп'ютері або на якомусь іншому комп'ютері, розташованому в сусідній кімнаті або на іншому континенті. У деяких випадках на місцезнаходження каталогу або файлу може вказувати його ім'я, але часто це не обов'язково. Аналогічно, коли користувач запускає прикладну програму, програма може працювати на тому ж комп'ютері або на іншому комп'ютері.

Прозорість мережі може бути дуже зручною для користувачів, оскільки вона звільняє їх від необхідності турбуватися про деталі структури мережі та вживати спеціальних заходів для доступу до віддалених даних. Це також може допомогти спростити завдання розробників програм і системних адміністраторів. Вона є головною особливістю Linux та інших Unix-подібних операційних систем. Це стало можливим завдяки використанню протоколу TCP/IP (протокол управління передачею/Інтернет-протокол) та його підтримці, яка вбудована в операційну систему та інше програмне забезпечення.

Для забезпечення прозорості мережі було розроблено низку методів. Наприклад, мережева файлова система (NFS) була створена для того, щоб дозволити користувачам монтувати розділ жорсткого диска (HDD), який існує на віддаленій машині, і використовувати його вміст так, як якщо б він знаходився на локальній машині.

Основною особливістю X Window System є те, що вона дозволяє будь-якій прикладній програмі, яка працює в графічному інтерфейсі користувача (GUI), прозоро працювати як на локальній машині, так і на віддаленій машині. Система X Window System, яка є стандартною для Unix-подібних операційних систем, є повною, крос-платформенною і безкоштовною клієнт-серверною системою для управління графічними інтерфейсами на окремих комп'ютерах і в мережах комп'ютерів[12].

2.5 Керованість

Управління мережею - це сукупність додатків, інструментів і процесів, що використовуються для забезпечення, експлуатації, обслуговування, адміністрування та захисту мережевої інфраструктури. Основна роль управління мережею полягає у забезпеченні ефективного, результативного та швидкого надання мережевих ресурсів користувачам. Воно використовує аналіз несправностей і управління продуктивністю для оптимізації стану мережі.

Навіщо потрібне управління мережею? Мережа об'єднує десятки, сотні і тисячі взаємодіючих компонентів. Ці компоненти іноді виходять з ладу, неправильно конфігуруються, перевантажуються або просто виходять з ладу. Програмне забезпечення для управління мережею підприємства повинно реагувати на ці виклики, використовуючи найбільш підходящі інструменти, необхідні для управління, моніторингу та контролю мережі

Основна мета управління мережею полягає в тому, щоб забезпечити ефективну і безперебійну роботу мережевої інфраструктури. При цьому досягаються наступні цілі:

- мінімізує дорогі перебої в роботі мережі. Збої в роботі мережі коштують дорого. Залежно від розміру організації або характеру порушених процесів, бізнес може зазнати збитків у тисячі або мільйони доларів лише після години простою. Ці втрати - це не лише прямі фінансові наслідки збоїв у роботі мережі - це також вартість зіпсованої репутації, яка змушує клієнтів переглянути свої довгострокові відносини з компанією. Повільні, не реагуючі на запити мережі розчаровують як клієнтів, так і співробітників. Вони ускладнюють реагування персоналу на запити та занепокоєння клієнтів. Клієнти, які занадто часто стикаються з проблемами в роботі мережі, розглядають можливість покинути її;

- підвищення продуктивності. Вивчаючи і контролюючи кожен аспект мережі, мережеве управління виконує кілька завдань одночасно. Завдяки цьому IT-персонал звільняється від повторюваної повсякденної рутини і може зосередитись на більш стратегічних аспектах своєї роботи;
- покращення мережевої безпеки. Ефективна програма управління мережею може виявляти та реагувати на кіберзагрози до того, як вони поширяться та вплинуть на роботу користувачів. Управління мережею забезпечує дотримання стандартів найкращих практик та відповідність нормативним вимогам. Краща мережева безпека підвищує конфіденційність мережі та надає користувачам впевненість у тому, що вони можуть вільно користуватися своїми пристроями;
- цілісний погляд на продуктивність мережі. Ефективне управління мережею забезпечує комплексне уявлення про продуктивність вашої інфраструктури. Ви зможете швидше виявляти, аналізувати та усувати проблеми.

Мережа є основою IT-інфраструктури. Управління цією мережею має вирішальне значення для забезпечення безперебійної роботи організації. Хоча принципи залишаються в основному незмінними, практика управління мережею передачі даних постійно розвивається в тандемі зі змінами технологій[13].

3 ПРОЄКТУВАННЯ, РЕАЛІЗАЦІЯ І АНАЛІЗ БЕЗПЕКИ ОБЧИСЛЮВАЛЬНОЇ МЕРЕЖІ

3.1 Загальний огляд мережі

Розглянемо головну частина мережі, що зображена на рис 3.1. Вона складається з наступних компонентів:

- Linux Ubuntu Server;
- Windows Server;
- комутатор, маршрутизатор, які поєднують сервери;
- граничний маршрутизатор, який підключений до послуг Інтернету ;
- п’ять маршрутизаторів, які поєднані між собою і з граничним маршрутизатором;
- маршрутизатори з додаткових частин мережі (корпусів).

Маршрутизації пакетів даних на маршрутизаторах відбувається завдяки протоколу OSPF.

Протокол OSPF (Open Shortest Path First) належить до сімейства протоколів IP-маршрутизації і є протоколом внутрішнього шлюзу (IGP) мережі Інтернет, який використовується для розповсюдження інформації про IP-маршрутизацію в межах однієї автономної системи (АС) в IP-мережі.

Протокол OSPF є протоколом маршрутизації за станом каналу, що означає, що маршрутизатори обмінюються інформацією про топологію зі своїми найближчими сусідами. Інформація про топологію поширюється по всій АС, так що кожен маршрутизатор в АС має повну картину топології АС. Ця картина потім використовується для розрахунку наскрізних шляхів через АС, зазвичай з використанням варіанту алгоритму Дейкстри. Таким чином, в протоколі маршрутизації за станом зв'язку адреса наступного переходу, на який пересилаються дані, визначається шляхом вибору найкращого наскрізного шляху до кінцевого пункту призначення.

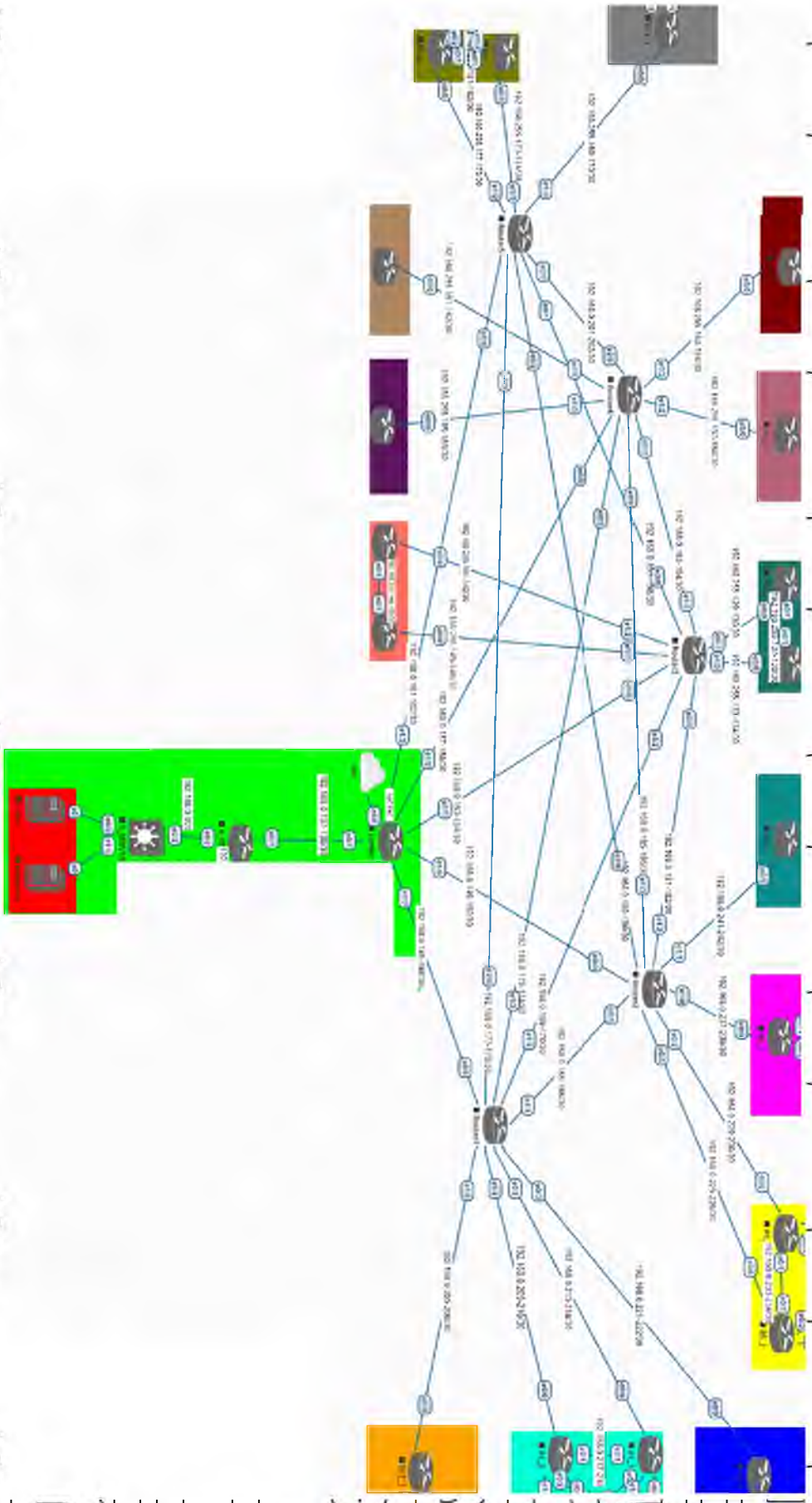

Рисунок 3.1 Головна частина мережі

Основною перевагою протоколу маршрутизації стану каналу, такого як OSPF, є те, що повне знання топології дозволяє маршрутизаторам розраховувати маршрути, які задовольняють певним критеріям. Це може бути корисно для цілей інженерії трафіку, де маршрути можуть бути обмежені для задоволення певних вимог до якості обслуговування. Основним недоліком протоколу маршрутизації за станом зв'язку є те, що він погано масштабується, коли до домену маршрутизації додається більше маршрутизаторів. Збільшення кількості маршрутизаторів збільшує розмір і частоту оновлення топології, а також час, необхідний для розрахунку наскрізних маршрутів. Ця відсутність масштабованості означає, що протокол маршрутизації за станом зв'язку непридатний для маршрутизації в Інтернеті в цілому, що є причиною того, що IGP маршрутизують трафік тільки в межах однієї АС.

Кожен пристрій у мережі на основі протоколу TCP/IP повинен мати унікальну одноадресну IP-адресу для доступу до мережі та її ресурсів. Без DHCP IP-адреси для нових комп'ютерів або комп'ютерів, які переміщуються з однієї підмережі в іншу, необхідно було б налаштовувати вручну.

За допомогою DHCP весь цей процес автоматизований і управляється централізовано. Сервер Linux Ubuntu DHCP підтримує пул IP-адрес і надає в оренду адресу будь-якому клієнту з підтримкою DHCP, коли він запускається в мережі. Оскільки IP-адреси є динамічними (орендованими), а не статичними (постійно призначеними), адреси, які більше не використовуються, автоматично повертаються до пулу для перерозподілу.

На маршрутизаторі, що з’єднаний з провайдером налаштований PAT.

Трансляція адрес портів (Port Address Translation, PAT) - це розширення трансляції мережевих адрес (NAT), яке дозволяє декільком пристроям в локальній мережі (LAN) відповідати одній загальнодоступній IP-адресі. Метою PAT є збереження IP-адрес.

У такому сценарії постачальник послуг Інтернету (ISP) призначає єдину IP-адресу маршрутизатору домашньої мережі. Коли Комп'ютер X входить в Інтернет, маршрутизатор призначає клієнту номер порту, який додається до

внутрішньої IP-адреси. Це, по суті, дає комп'ютеру X унікальну адресу. Якщо комп'ютер Z входить в Інтернет в той же час, маршрутизатор присвоює йому таку ж локальну IP-адресу з іншим номером порту. Хоча обидва комп'ютери мають одну і ту ж публічну IP-адресу і виходять в Інтернет одночасно, маршрутизатор точно знає, якому комп'ютеру відправляти конкретні пакети, тому що кожен комп'ютер має унікальну внутрішню адресу. На рисунку 3.2 зображена схема роботи NAT.

Рисунок 3.2 Схема роботи NAT

Розглянемо додаткові частині мережі, що зображені на рис 3.3. Кожна додаткова частина (корпус) має, як мінімум, один VLAN.

VLAN - це сукупність пристроїв або мережевих вузлів, які взаємодіють один з одним так, ніби вони складають єдину локальну мережу, тоді як насправді вони існують в одному або декількох сегментах локальної мережі. У технічному сенсі сегмент відокремлюється від решти локальної мережі мостом, маршрутизатором або комутатором і зазвичай використовується для певного відділу. Це означає, що коли робоча станція транслює пакети, вони досягають усіх інших робочих станцій у віртуальній локальній мережі, але не виходять за її межі.

Це спрощує багато потенційних ускладнень, викликаних локальними мережами, включаючи надмірний мережевий трафік і колізії. Коли дві робочі

Рисунок 3.3 – Додаткові частини мережі

станції одночасно відправляють пакети даних в локальній мережі, підключеній через концентратор, дані стикаються і не передаються належним чином. Колізія поширюється по всій мережі, що означає, що локальна мережа зайнята і вимагає від користувачів чекати, поки колізія буде повністю передана по всій мережі, перш ніж вона знову стане працездатною - в цей момент вихідні дані повинні бути відправлені повторно.

Віртуальні локальні мережі зменшують частоту колізій та зменшують кількість мережевих ресурсів, що витрачаються даремно, діючи як сегменти локальної мережі. Пакети даних, відправлені з робочої станції в сегменті, передаються мостом або комутатором, який не перенаправляє колізії, а розсилає широкомовні повідомлення всім мережевим пристроям. З цієї причини сегменти називаються "доменами колізій", оскільки вони містять колізії в межах цієї ділянки.

Первісні схеми додаткових частин мережі може змінюватись відносно навантаження. На рис 3.4 зображена схема мережі корпусу з великим навантаженням. А на рис 3.5 зображена схема мережі з мінімальним або середнім навантаженням.

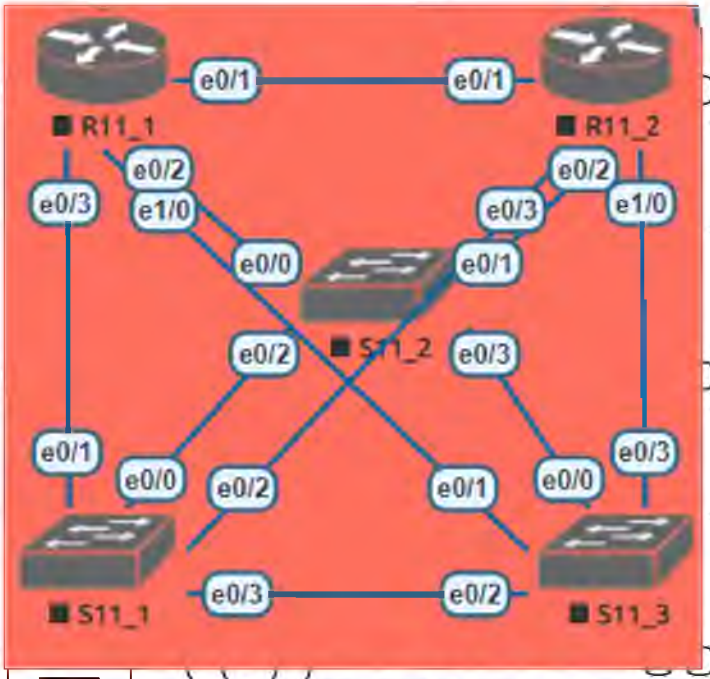

Рисунок 3.4 Схема мережі корпусу з великим навантаженням

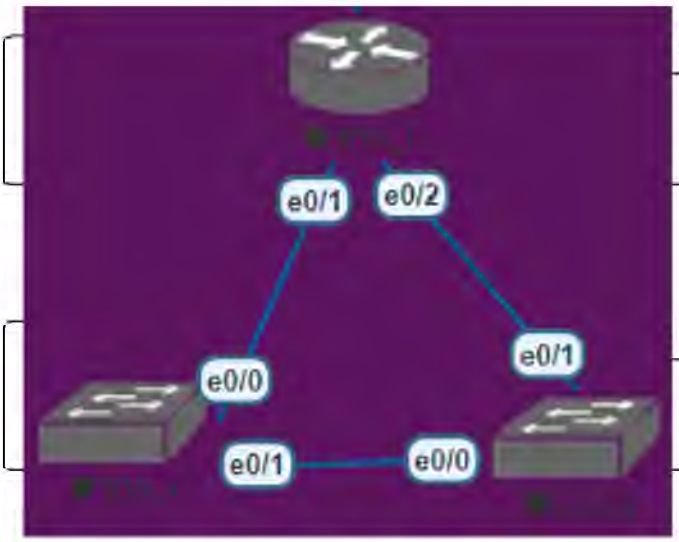

Рисунок 3.5 Схема мережі корпусу з мінімальним або середнім навантаженням

У випадку з великим навантаженням збільшується кількість маршрутизаторів, комутаторів і використовується протокол Rapid PVST, схема якого зображена на рис. 3.6.

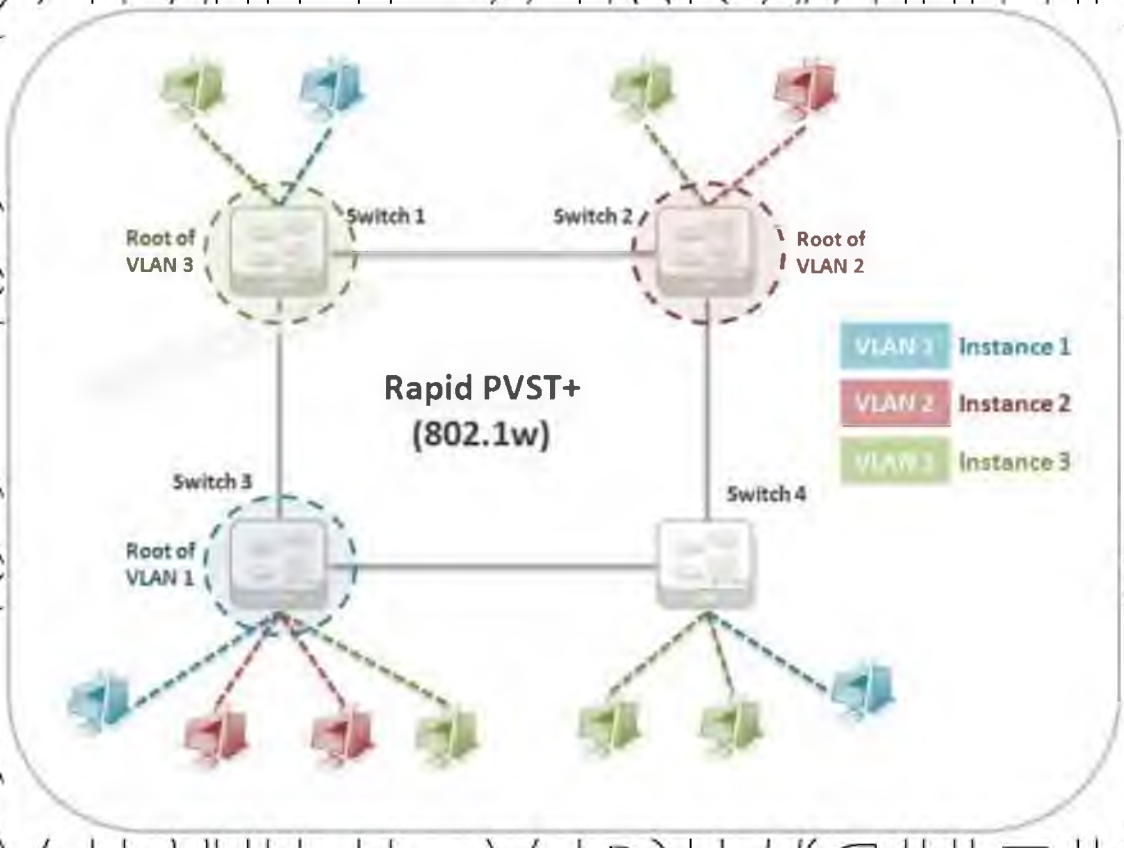

Рисунок 3.6 Схема роботи протоколу Rapid PVST+

3.2 Мережева безпека

Мережева безпека має життєво важливе значення для захисту даних та інформації клієнтів, збереження спільних даних, забезпечення надійного доступу та продуктивності мережі, а також захисту від кіберзагроз. Добре розроблене рішення мережевої безпеки зменшує накладні витрати і захищає організації від дорогих втрат, які виникають внаслідок витоку даних або інших інцидентів безпеки. Забезпечення легітимного доступу до систем, додатків і даних дозволяє здійснювати бізнес-операції та надавати послуги і продукти клієнтам.

Брандмауер (Firewall) - це пристрій мережевої безпеки, який контролює вхідний і вихідний мережевий трафік і дозволяє або блокує пакети даних на основі набору правил безпеки. Його мета - встановити бар'єр між внутрішньою мережею та вхідним трафіком із зовнішніх джерел (наприклад, з Інтернету), щоб заблокувати шкідливий трафік, такий як віруси та хакерські програми.

Брандмауери можуть бути як програмними, так і апаратними, хоча найкраще мати і ті, і інші. Програмний брандмауер - це програма, встановлена на кожному комп'ютері, яка регулює трафік за допомогою номерів портів і додатків, тоді як фізичний брандмауер - це обладнання, встановлене між вашою мережею і шлюзом.

Брандмауери з фільтрацією пакетів, найпоширеніший тип брандмауерів, перевіряють пакети і забороняють їх проходження, якщо вони не відповідають встановленому набору правил безпеки. Цей тип брандмауерів перевіряє IP-адреси джерела та призначення пакета. Якщо пакети відповідають "дозволеним" правилам брандмауера, то вони проходять в мережу.

Брандмауери з фільтрацією пакетів діляться на дві категорії: з відстеженням стану та без збереження стану. Без збереження стану брандмауери досліджують пакети незалежно один від одного і не мають контексту, що робить їх легкою мішенню для хакерів. На відміну від них, брандмауери з відстеженням

стану запам'ятовують інформацію про раніше пройдені пакети і вважаються набагато більш безпечними.

Хоча брандмауери з фільтрацією пакетів можуть бути ефективними, вони в кінцевому підсумку забезпечують дуже базовий захист і можуть бути дуже обмеженими - наприклад, вони не можуть визначити, чи вплине вміст запиту, який надсилається, негативно на додаток, до якого він потрапляє. Якщо зловмисний запит, дозволений з довіреної адреси джерела, призведе, скажімо, до видалення бази даних, брандмауер не матиме можливості дізнатися про це. Брандмауери нового покоління і проксі-брандмауери мають більше можливостей для виявлення таких загроз.

Брандмауери нового покоління (NGFW) поєднують традиційну технологію брандмауерів з додатковою функціональністю, такою як перевірка зашифрованого трафіку, системи запобігання вторгненням, антивірус та інше. Найголовніше - це глибока перевірка пакетів (DPI). У той час як базові брандмауери переглядають тільки заголовки пакетів, глибока перевірка пакетів перевіряє дані в самому пакеті, дозволяючи користувачам більш ефективно ідентифікувати, класифікувати або зупиняти пакети з шкідливими даними.

Проксі-сервери фільтрують мережевий трафік на рівні додатків. На відміну від базових брандмауерів, проксі виступає посередником між двома кінцевими системами. Клієнт повинен відправити запит на брандмауер, де він оцінюється за набором правил безпеки, а потім дозволяється або блокується. Найголовніше, що проксі-брандмауери відстежують трафік для протоколів 7-го рівня, таких як HTTP і FTP, і використовують як перевірку стану, так і глибоку перевірку пакетів для виявлення зловмисного трафіку.

Брандмауери з трансляцією мережевих адрес (NAT) дозволяють декільком пристроям з незалежними мережевими адресами підключатися до Інтернету, використовуючи одну IP-адресу, зберігаючи індивідуальні IP-адреси прихованими. В результаті, зловмисники, які сканують мережу на наявність IP-адрес, не можуть перехопити конкретні деталі, що забезпечує більший захист від

атак. Брандмауери NAT схожі на проксі-брандмауери в тому, що вони діють як посередник між групою комп'ютерів і зовнішнім трафіком.

Брандмауери з багаторівневою перевіркою (SMLI) фільтрують пакети на мережевому, транспортному і прикладному рівнях, порівнюючи їх з відомими довіреними пакетами. Як і брандмауери NGFW, SMLI також перевіряють весь пакет і пропускають його тільки в тому випадку, якщо він пройшов кожен рівень окремо. Ці брандмауери досліджують пакети, щоб визначити стан зв'язку (отже, ім'я), щоб гарантувати, що весь ініційований зв'язок відбувається тільки з довіреними джерелами[14].

Налаштуємо на Ubuntu програмний брандмауер.

UFW, або Uncomplicated Firewall - це інтерфейс до iptables, який спрямований на спрощення процесу налаштування брандмауера. Якщо ви хочете почати захищати свою мережу і не знаєте, який інструмент використовувати, UFW може бути правильним вибором для вас.

1. Потрібно переконатися, що IPv6 увімкнено.

В останніх версіях Ubuntu IPv6 включений за замовчуванням. На практиці це означає, що більшість правил брандмауера, доданих на сервер, будуть включати як IPv4, так і IPv6 версію, причому остання ідентифікується за допомогою v6 у виведенні команди стану UFW. Для цього відкриваємо конфігураційний файл UFW за адресою /etc/default/ufw.

Лістинг 3.1

```
sudo nano /etc/default/ufw
```

І перевіряємо, щоб значення IPV6 було «yes».

2. Налаштування політик за замовчуванням.

Ці правила визначають, як обробляти трафік, який явно не відповідає іншим правилам.

За замовчуванням UFW налаштований на заборону всіх вхідних з'єднань і дозвіл всіх вихідних з'єднань. Це означає, що будь-хто, хто намагається

зв'язатися з вашим сервером, не зможе підключитися, в той час як будь-яка програма всередині сервера зможе зв'язатися із зовнішнім світом. Додаткові правила, що дозволяють певні служби та порти, включені як винятки з цієї загальної політики.

Щоб встановити політику заборони вхідних UFW за замовчуванням, прописуємо наступні команди:

Лістинг 3.2

```
sudo ufw default deny incoming
```

Щоб встановити вихідну політику UFWUFW за замовчуванням, пронисуємо наступні команди:

Лістинг 3.3

```
sudo ufw default allow outgoing
```

3. Дозвіл на підключення по SSH

Якщо включите брандмауер UFW зараз, він буде забороняти всі вхідні з'єднання. Це означає, що потрібно буде створити правила, які явно дозволяють легітимні вхідні з'єднання - SSH або HTTP, наприклад - якщо ви хочете, щоб ваш сервер відповідав на ці типи запитів. Якщо ви використовуєте хмарний сервер, потрібно дозволити вхідні SSH-з'єднання, щоб мати змогу підключатися до свого сервера і керувати ним.

Після встановлення більшість програм, які покладаються на мережеві з'єднання, реєструють профіль програми в UFW, який дозволяє користувачам швидко дозволити або заборонити зовнішній доступ до сервісу. Щоб перевірити, які профілі в даний час зареєстровані в UFW прописуємо наступні команди:

Лістинг 3.4

```
sudo ufw app list
```

Для включення профілю програми OpenSSH прописуємо:

Лістинг 3.5

```
sudo ufw allow OpenSSH
```


Це створить правила брандмауера, які дозволять всі з'єднання на порт 22, який за замовчуванням слухає демон SSH.

Інший спосіб налаштувати UFW на дозвіл вхідних SSH-з'єднань полягає в посиланні на ім'я його служби ssh:

Лістинг 3.5

```
sudo ufw allow ssh
```


UFW знає, які порти і протоколи використовує служба на основі файлу /etc/services.

Як варіант, можна написати еквівалентне правило, вказавши замість профілю програми або імені служби порт. Наприклад, ця команда працює так само, як і попередні приклади:

Лістинг 3.6

```
sudo ufw allow 22
```


4. Увімкнення UFW

Увімкнути брандмауер можна за допомогою:

Лістинг 3.7

```
sudo ufw enable
```


Ми отримаємо попередження про те, що команда може порушити існуючі SSH-з'єднання. Так як ми вже налаштували правило брандмауера, яке

дозволяє SSH-з'єднання, тому можемо продовжувати. Відповідаємо на запит за допомогою клавіш y і натискаємо клавішу ENTER.

Брандмауер тепер активний. За допомогою команди sudo ufw status verbose можна побачити встановлені правила.

5. Дозвіл підключення HTTP, HTTPS, Apache, Nginx, IP-адрес.

На цьому етапі ми повинні дозволити всі інші з'єднання, на які повинен відповідати наш сервер. Зробимо це для HTTP, HTTPS, Apache, Nginx:

Лістинг 3.8

```
sudo ufw allow 80
sudo ufw allow 443
sudo ufw allow 'Apache Full'
sudo ufw allow 'Nginx Full'
```

Також існує кілька інших способів дозволити з'єднання, крім вказівки порту або відомого імені служби. Деякі програми використовують кілька портів замість одного. За допомогою UFW можна вказати діапазони портів:

Лістинг 3.9

```
sudo ufw allow 80
sudo ufw allow 443
sudo ufw allow 'Apache Full'
sudo ufw allow 'Nginx Full'
```

При роботі з UFW можна також вказувати IP-адреси в рамках своїх правил. Наприклад, щоб дозволити з'єднання з певної IP-адреси, наприклад, робочої або домашньої IP-адреси 203.0.113.4, необхідно використовувати параметр from, вказавши потім IP-адресу, яку ви хочете дозволити:

Лістинг 3.9

```
sudo ufw allow from 203.0.113.4
```

Також можна вказати порт, до якого дозволено підключатися IP-адресі, додавши до будь-якого порту номер порту. Наприклад, якщо ви хочете дозволити 203.0.113.4 підключитися до порту 22 (SSH) потрібно в кінці минулої команди to any port 22.

6. Підключення до певного мережевого інтерфейсу

Отже, якщо сервер має загальнодоступний мережевий інтерфейс з назвою eth0, ви можете дозволити HTTP-трафік (порт 80) на нього за допомогою цієї команди:

Лістинг 3.10

```
sudo ufw allow in on eth0 to any port 80
```

7. Заборона підключень

Іноді може знадобитися заборонити певні з'єднання на основі вихідної IP-адреси або підмережі, можливо, тому ваш сервер атакують звідти. Крім того, якщо використовувати політику вхідного трафіку за замовчуванням на дозволену (що не рекомендується), нам потрібно буде створити правила заборони для будь-яких служб або IP-адрес, для яких не хочемо дозволяти з'єднання.

Для написання правил заборони можна скористатися командами, описаними раніше, замінивши allow на deny:

Лістинг 3.11

```
sudo ufw deny from 203.0.113.4
```

Антивірусне програмне забезпечення захищає мережу від різних форм шкідливого програмного забезпечення, включаючи шпигунські програми, програми-вимагачі, трояни, черв'яки та безліч вірусів. Оскільки шкідливе програмне забезпечення може проникати в мережу і залишатися в сплячому режимі протягом тривалого часу, програмне забезпечення може відстежувати

доступ, усунути його, виправити будь-які проблеми, які воно створило, і регулярно перевіряти на наявність аномалій.

Завдяки сегментації мережевий трафік можна класифікувати за різними категоріями, що значно полегшує впровадження політик безпеки. Програмно-визначена сегментація може бути виконана на основі ідентифікації кінцевих точок, на додаток до загальноприйнятих IP-адрес. Таким чином, доступ отримують тільки потрібні люди, а всі підозрілі спроби підключення припиняються

Віддалений доступ VPN забезпечує віддалений і безпечний доступ до мережі компанії окремим хостам або клієнтам, таким як віддалені працівники, мобільні користувачі та користувачі екстранету. На кожному хості, як правило, завантажено клієнтське програмне забезпечення VPN або використовується веб-клієнт. Конфіденційність і цілісність конфіденційної інформації забезпечується багатофакторною аутентифікацією, перевіркою кінцевих точок на відповідність вимогам та шифруванням усіх даних, що передаються.

У звіті Verizon про розслідування витоків даних проаналізовано 41 686 інцидентів безпеки, з яких 2 013 - підтверджені витоки даних. Було виявлено, що ці порушення в основному були спричинені скомпрометованими, повторно використаними або слабкими паролями. Багатофакторна автентифікація (MFA) є важливим інструментом кібербезпеки, який допомагає запобігти таким порушенням. Вона надає пристрою додатковий рівень захисту, надсилаючи одноразовий код для входу в систему.

Резервне копіювання мережі має важливе значення для забезпечення безпеки організації. Хоча існує багато способів резервного копіювання бізнес-інформації, важливо зберігати інформацію та дані таким чином, щоб вони не могли бути втрачені.

Мережеве резервне копіювання - це система, в якій вибрані дані з клієнтів резервного копіювання (одного комп'ютера або мережі комп'ютерів) передаються через мережу (також відому як Інтернет) і надсилаються на ваш сервер резервного копіювання. Цей сервер може перебувати у приватній

власності та управлятися або розміщуватися у хмарному провайдера резервного копіювання – як це часто буває для більшості малих підприємств.

Розширені мережеві системи резервного копіювання можуть також керувати носіями резервних копій, які пов'язані з сервером резервного копіювання через мережу. Цей тип розширених налаштувань особливо корисний для підприємств, які використовують пристрої NAS (мережеві накопичувачі) для спільного доступу до даних.

Система запобігання вторгненням (IPS) сканує мережевий трафік для активного блокування атак. Безпечні пристрої IPS роблять це, корелюючи величезні обсяги розвідданих про глобальні загрози, щоб не тільки блокувати шкідливу активність, але й відстежувати просування підозрілих файлів і шкідливого програмного забезпечення по мережі, щоб запобігти поширенню спалахів і повторному зараженню.

Безпека робочих навантажень захищає робочі навантаження, що переміщуються через різні хмарні та гібридні середовища. Ці розподілені робочі навантаження мають більшу поверхню атаки, яку необхідно захистити, не впливаючи на гнучкість бізнесу.

Поведінкова аналітика виявляє аномальну поведінку в мережі. Інструменти поведінкової аналітики автоматично розпізнають дії, які відхиляються від норми. Тоді ваша команда безпеки може краще визначити індикатори порушення, які становлять потенційну проблему, і швидко усунути загрози.

Шлюзи електронної пошти є вектором загроз номер один для порушення безпеки. Зловмисники використовують особисту інформацію та тактику соціальної інженерії для побудови складних фішингових кампаній з метою обману одержувачів та перенаправлення їх на сайти з шкідливим програмним забезпеченням. Додаток для захисту електронної пошти блокує вхідні атаки та контролює вихідні повідомлення, щоб запобігти втраті конфіденційних даних.

3.3 Оцінка ризиків

Для визначення ступеню захищеності мережі необхідно виконати оцінку ризиків.

Оцінка ризиків в кібербезпеці - це процес виявлення та оцінки ризиків для активів, які можуть постраждати від кібератак. В основному, ви визначаєте як внутрішні, так і зовнішні загрози; оцінюєте їх потенційний вплив на такі речі, як доступність, конфіденційність та цілісність даних; а також оцінюєте витрати, пов'язані з інцидентом кібербезпеки. Маючи цю інформацію, ви можете адаптувати засоби контролю кібербезпеки та захисту даних до фактичного рівня толерантності до ризиків вашої організації.

Щоб розпочати оцінку ризиків ІТ-безпеки, необхідно визначити:

- якими є критичні інформаційно-технологічні активи вашої організації, тобто дані, втрата або витік яких матиме значний вплив на ваші бізнес-операції;
- які ключові бізнес-процеси використовують або потребують цієї інформації;
- які загрози можуть вплинути на здатність цих бізнес-функцій працювати;

Після того, як ви знаєте, що вам потрібно захистити, ви можете приступити до розробки стратегії. Однак, перш обов'язково подумайте, який ризик ви вирішуєте, наскільки він є пріоритетним, і чи підходите ви до нього в найбільш економічно ефективний спосіб.

Регулярне проведення ретельної оцінки ІТ-безпеки допомагає організаціям створити міцний фундамент для досягнення успіху. Зокрема, це дозволяє їм:

- виявити та усунути прогалини в ІТ-безпеці;
- запобігати витоку даних;
- обрати відповідні протоколи та засоби контролю для зменшення ризиків;

– визначати пріоритетність захисту активів, що мають найбільшу цінність та найбільші ризики;
– усунути непотрібні або застарілі заходи контролю;
– Оцінити потенційних партнерів з безпеки;
– встановити, підтримувати і доводити відповідність нормативним вимогам;
– точно прогнозувати майбутні потреби.

Інститут управління ризиками визначає кібернетичний ризик як "будь-який ризик фінансових втрат, збоїв у роботі або шкоди репутації організації внаслідок певного роду збоїв у роботі її інформаційно-технологічних систем". Gartner дає більш загальне визначення: "потенціал для незапланованого, негативного результату бізнесу, пов'язаного з відмовою або неправильним використанням IT".

При оцінці кібер-ризиків важливо деталізувати конкретні фінансові збитки, які вони можуть завдати організації, такі як судові витрати, операційні простої і пов'язана з ними втрата прибутку, а також втрата бізнесу через недовіру клієнтів.

3.3.1 Оцінка ризиків від NIST

Національний інститут стандартів і технології (NIST), що є національним органом з стандартизації в Сполучених Штатах Америки, радить для визначення ймовірності виникнення ризиків враховувати три визначальні фактори:

– мотивація та можливості джерела загрози;
– характер вразливості;
– існування та ефективність поточних засобів контролю.

Для прийняття рішень NIST надає таблицю ймовірностей 3.1.

Для впливу вони розробили ще одну таблицю 3.2, яка допомагає оцінити величину впливу.

Таблиця 3.1 – Ймовірності для прийняття рішень від NIST

Рівень ймовірності	Визначення ймовірності
Високий	Джерело загрози є високо мотивованим та достатньо спроможним, а засоби контролю для запобігання реалізації вразливостей є неефективними.
Середній	Джерело загрози вмотивоване та спроможне, але наявні засоби контролю можуть перешкоджати успішному використанню вразливості.
Низький	Джерело загрози не має достатньої мотивації та спроможності, або наявні засоби контролю не дозволяють або, принаймні, суттєво ускладнюють реалізацію вразливості.

Таблиця 3.2 – Величина впливу від NIST

Масштаб впливу	Визначення впливу
Високий	Реалізація вразливості (1) може призвести до значної втрати значних матеріальних активів або ресурсів; (2) може суттєво порушити, зашкодити або перешкодити місії, репутації або інтересам організації; або (3) може призвести до людських жертв або серйозних травм.
Середній	Реалізація вразливості (1) може призвести до значних втрат матеріальних активів або ресурсів; (2) може порушити, зашкодити або перешкодити місії, репутації або інтересам організації; або (3) може призвести до травмування людей.
Низький	Реалізація вразливості (1) може призвести до втрати певних матеріальних активів чи ресурсів або (2) може суттєво вплинути на місію, репутацію чи інтереси організації.

Потім, на етапі фактичного визначення ризиків, вони розробили матрицю рівнів ризиків, що зображена у вигляді таблиці 3.3 та шкалу ризиків, що зображена у вигляді таблиці 3.4.

Таблиця 3.3 – Матриця рівнів ризиків від NIST

Імовірність загрози	Вплив		
	Низький (10)	Середній (50)	Високий (100)
Високий (1,0)	Низький 10 x 1 = 10	Середній 50 x 1.0 = 50	Високий 100 x 1.0 = 100
Середній (0,5)	Низький 10 x 0.5 = 5	Середній 50 x 0.5 = 25	Високий 100 x 0.5 = 50
Низький (0,1)	Низький 10 x 0.1 = 1	Середній 50 x 0.1 = 5	Високий 100 x 0.1 = 10

Шкала ризиків для цієї матриці виглядає наступним чином:

- високий (>50-100);
- середній (>10-50);
- низький (1-10).

Таблиця 3.4 – Шкала ризиків від NIST

Рівень ризику	Опис ризиків та необхідні дії
Високий	Якщо спостереження або знахідка оцінюється як високий ризик, існує нагальна потреба у вжитті коригувальних заходів. Існуюча система може продовжувати функціонувати, але план коригувальних дій повинен бути впроваджений якнайшвидше.
Середній	Якщо зауваження оцінюється як середній ризик, необхідні коригувальні дії, і повинен бути розроблений план, що включає ці дії протягом розумного періоду часу.

Продовження таблиці 3.4

Низький	Якщо зауваження оцінюється як низький ризик, ООР системи повинен визначити, чи все ще потрібні коригувальні дії, або прийняти рішення про прийняття ризику.

Таблиці та матриця, наведені вище, допомагають визначати ризики, не покладаючись виключно на відчуття[15].

3.3.2 Оцінка ризиків від HIPAA

Крім метода вище, існує багато варіантів визначення ризиків, розглянемо шаблон оцінки ризиків від HIPAA.

Оцінка ризиків HIPAA допомагає організаціям визначити та оцінити загрози безпеці електронної захищеної інформації, включаючи потенціал несанкціонованого розкриття, як того вимагає Правило конфіденційності.

Якщо ваша організація створює, отримує, зберігає або передає захищену інформацію, ви повинні оцінити свої ризики безпеки, щоб переконатися, що ви вжили найкращих можливих заходів для її захисту. Після того, як ви визначите ці ризики, ви повинні впровадити адміністративні, фізичні та технічні засоби захисту, щоб забезпечити відповідність Правилам безпеки HIPAA.

Інструменти аналізу та управління ризиками можуть бути безцінними; вони часто дозволяють захистити конфіденційність, цілісність та доступність вашої електронної інформації більш ефективно та результативно, ніж ви могли б зробити це за допомогою ручних процесів.

Вимоги HIPAA до оцінки ризиків дозволяють адаптувати оцінку до умов та обставин вашої організації, в тому числі:

- розмір, складність та можливості вашої організації;
- технічну інфраструктуру, апаратне забезпечення та можливості безпеки вашої організації;

– ймовірність та критичність потенційних ризиків для захищеної інформації;

– вартість заходів безпеки.

Оцінка ризиків HIPAA буде містити багато специфікацій впровадження, які є детальними інструкціями для задоволення певного стандарту. Деякі з них є обов'язковими, тоді як інші можуть бути виконані.

Обов'язкові специфікації документують політику або процедури, які кожна охоплена організація та її ділові партнери повинні запровадити. Одним із прикладів є аналіз ризиків.

Адресні специфікації не є необов'язковими, але організації мають гнучкість у виборі відповідних процесів або засобів контролю для їх виконання. Наприклад, управління паролями є адресним, оскільки існує багато способів забезпечити доступ до ваших систем тільки довіреним особам. Одним із способів є використання багатофакторної автентифікації.

Ви не можете відмовитися від прийняття специфікації впровадження, грунтуючись лише на вартості[16].

Розглянемо матрицю ризиків, що зображені на таблиці 3.5.

Таблиця 3.5 Матриця рівнів ризиків від HIPAA

		Вплив		
		Низький (0,1)	Середній (0,5)	Високий (1,0)
Ймовірність загрози	Низький (5)	5 X 0.1 = 0.5	5 X 0.5 = 2.5	5 X 1.0 = 1
	Середній (25)	25 X 0.1 = 2.5	25 X 0.5 = 12.5	25 X 1.0 = 25
	Високий (50)	50 X 0.1 = 5	50 X 0.5 = 25	50 X 1.0 = 50

Шкала ризику від HIPAA:

– високий: >25 до 50;

– середній: >5 до 25;

– низький >від 0,5 до 5.

3.3.3 Оцінка ризиків безпеки для мережі університету

3.3.3.1 Виявлення та визначення пріоритетності активів

Активи включають сервери, контактну інформацію клієнтів, конфіденційні документи партнерів, комерційну таємницю тощо. Потрібно пам'ятати, що потрібно працювати з бізнес-користувачами та керівництвом, щоб створити список всіх цінних активів, тому що визначити, що насправді є найбільш цінним для бізнесу технічному спеціалісту важко.

Для кожного активу потрібно зібрати наступну інформацію, залежно від обставин: програмне забезпечення; апаратне забезпечення; дані; інтерфейси; користувачі; допоміжний персонал; місія або мета; критичність; функціональні вимоги; політики безпеки IT; архітектура IT-безпеки; топологія мережі; захист сховищ інформації; інформаційні потоки; технічні засоби контролю безпеки; фізичний захист середовища; екологічна безпека.

Оскільки більшість організацій мають обмежений бюджет на оцінку ризиків, доводиться обмежувати сферу застосування решти кроків критично важливими для місії активами. Відповідно, потрібно визначити стандарт для визначення важливості кожного активу. Загальні критерії включають грошову вартість активу, його правовий статус та важливість для організації. Після того, як стандарт буде затверджений керівництвом і офіційно включений в політику безпеки з оцінки ризиків, його використовуються для класифікації кожного активу як критично важливого, основного або другорядного.

3.3.3.2 Виявлення загроз

Загроза - це все, що може завдати шкоди вашій організації. Хоча хакери та шкідливе програмне забезпечення, ймовірно, першими спадають на думку, існує багато інших типів загроз:

- стихійні лиха. Повені, урагани, землетруси, пожежі та інші стихійні лиха можуть знищити не тільки дані, але й сервери та обладнання. Вирішуючи, де розмістити свої сервери, потрібно думати про ймовірність різних типів стихійних лих;

– відмова обладнання. Ймовірність виходу з ладу обладнання залежить від якості та віку сервера або іншої машини. Для відносно нового, якісного обладнання ймовірність виходу з ладу низька. Але якщо обладнання старе або від "нікому не відомого" постачальника, шанс виходу з ладу значно вищий. Ця загроза повинна бути у списку, незалежно від того, яким бізнесом ви займаєтеся. Люди можуть випадково видалити важливі файли, натиснути на шкідливе посилання в електронному листі або пролити каву на обладнання, на якому розміщені критичні системи;

– зловмисна поведінка. Існує три типи зловмисної поведінки:

– втручання - це коли хтось завдає шкоди вашому бізнесу шляхом видалення даних, організації розподіленої відмови в обслуговуванні (DDOS) проти вашого веб-сайту, фізичної крадіжки комп'ютера або сервера тощо;

– перехоплення - це крадіжка даних;

– видавання себе за іншу особу - це неправомірне використання чужих облікових даних, які часто набуваються за допомогою атак соціальної інженерії або атак грубої сили.

3.3.3.3 Виявлення вразливостей

Вразливість - це слабке місце, яке може дозволити загрозі завдати шкоди організації. Вразливості можна виявити за допомогою аналізу, аудиторських звітів, бази даних вразливостей NIST, даних постачальників, процедур тестування та оцінки інформаційної безпеки (ST&E), тестування на проникнення та автоматизованих інструментів сканування вразливостей.

Крім вразливостей програмного забезпечення існують також фізичні та людські вразливості. Наприклад, розміщення серверної кімнати в підвалі підвищує вразливість до загрози затоплення, а нездатність проінформувати співробітників про небезпеку переходу за посиланнями в електронній пошті

підвищує вразливість до загрози зараження шкідливим програмним забезпеченням.

3.3.3.4 Аналіз елементів управління

Потрібно проаналізувати засоби контролю, які вже існують або знаходяться на стадії планування, щоб мінімізувати або усунути ймовірність того, що загроза скористається вразливістю. Технічні засоби контролю включають шифрування, механізми виявлення вторгнень, а також рішення для ідентифікації та автентифікації. Нетехнічні засоби контролю включають політику безпеки, адміністративні заходи, а також фізичні та екологічні механізми.

Як технічні, так і нетехнічні засоби контролю можуть бути класифіковані як превентивні або детективні. Як випливає з назви, превентивні засоби контролю намагаються передбачити та зупинити атаки. Прикладами є пристрої шифрування та автентифікації. Засоби виявлення використовуються для виявлення загроз, які вже відбулися або знаходяться в процесі. Це є аудиторські сліди та системи виявлення вторгнень.

3.3.3.5 Визначення ймовірності інциденту

Оцінка ймовірності того, що вразливість може бути фактично використана, беручи до уваги тип вразливості, можливості та мотивацію джерела загрози, а також наявність та ефективність ваших засобів контролю, є дуже важливою. Багато організацій використовують категорії "високий", "середній" та "низький" для оцінки ймовірності атаки або іншої несприятливої події.

3.3.3.6 Визначення пріоритетності ризиків інформаційної безпеки

Методи оцінки ризику було розглянуто вище. Для кожної пари загроза/вразливість потрібно визначити рівень ризику для ІТ-системи, виходячи з наступного:

– ймовірність того, що загроза використає вразливість;

– орієнтовна вартість кожної з цих подій;
– достатності існуючих або запланованих засобів контролю безпеки інформаційної системи для усунення або зниження ризику.

3.3.3.7 Рекомендовані заходи контролю

Тепер перейдемо до останнього кроку. Заключним етапом процесу оцінки ризиків є розробка звіту з оцінки ризиків, що представлений у вигляді таблиці 3.6. Для кожної загрози у звіті повинні бути описані відповідні вразливості, активи, що знаходяться під загрозою, вплив на ІТ-інфраструктуру, ймовірність виникнення та рекомендації щодо контролю.

Використовуючи рівень ризику як основу, необхідно визначити дії, необхідні для зменшення ризику. загальні рекомендації для кожного рівня ризику:

– високий - план коригувальних заходів повинен бути розроблений якнайшвидше;
– Середній - план коригувальних заходів повинен бути розроблений протягом розумного періоду часу;
– Низький - команда повинна вирішити, чи приймати ризик, чи впроваджувати коригувальні заходи.

Оцінюючи засоби контролю для зменшення кожного ризику, обов'язково потрібно враховувати. організаційну політику, аналіз витрат і вигоди, операційний вплив. здійсненність. застосовані нормативні акти, загальна ефективність рекомендованих засобів контролю. безпека та надійність[17].

Таблиця 3.6 Звіт з оцінки ризиків

Загроза	Вразливість	Актив	Вплив	Імовірність	Ризик	Рекомендації щодо контролю
Зловмисне людське (втручання) - DDOS-атака. **Висока**	Брандмауер налаштований належним чином та має хороший захист від DDOS. **Низький**	Сайт. **Високий**	Критичні ресурси сайту будуть недоступні. **Критичний**	DDOS було виявлено один раз за рік. **Низький**	Середній потенційний збиток. **Середній**	Моніторинг брандмауера
Збій системи -перегрів в серверній кімнаті. **Середній**	Системам кондиціонування повітря вже чотири роки. **Середній**	Сервери. **Високий**	Всі сервіси будуть недоступні щонайменше 3 години. **Критичний**	Поточна температура в серверній майже в нормі. **Середній**	Високий потенційний збиток. **Високий**	Ретельно слідкувати за температурою і при необхідності купити нове охолодження

Випадкове людське втручання - випадкове видалення файлів. **Високий**	Стихійні лиха – повені. **Низький**				
Дозволи налаштовані належним чином; регулярно створюються резервні копії. **Низький**	Поблизу відсутні водойми. **Низький**				
Файли на файловому ресурсі. **Середній**	Сервери. **Високий**				
Критично важливі дані можуть бути втрачені але майже напевно можуть бути можуть бути відновлені з резервної копії. **Низький**	Всі сервіси будуть недоступні. **Критичний**				
Середній	Паводки не траплялись. **Низький**				
Низький	**Низький**				
резервних копій.	Ніяких дій не потрібно	Продовжувати моніторинг змін дозволів привілейованих користувачів та			

ВИСНОВКИ

У результаті виконання магістерської роботи було досліджено захищеність мережі Національного Університету Біоресурсів і Природокористування України.

У роботі вирішено наступні задачі:

1. Проведено аналіз та опис предметної області. Проаналізовано вразливості безпеки, такі як: вразливості у вихідному коді, неправильно налаштовані компоненти системи, конфігурації довіри, слабка практика авторизації, відсутність надійного шифрування, внутрішня загроза, психологічна вразливість, неадекватна автентифікація, викриття конфіденційних даних, недостатній моніторинг та ведення логів, вразливості спільного користування.

2. Досліджено основні вимоги мережі: продуктивність (пропускна здатність, затримка передачі), надійність і безпека (шкідливе програмне забезпечення, відмова в обслуговуванні (DoS) та розподілена відмова в обслуговуванні (DDoS), атака "людина посередині" (man-in-the-middle, MitM), підслуховування Wi-Fi, підміна DNS, IP-адрес, HTTPS, підробка ARP, злом електронної пошти, сесії, зняття SSL-шифрування), розширюваність і масштабованість, прозорість, керованість.

3. Досліджено мережеву безпеку: брандмауери, сегментація мережі, віддалений доступ VPN, резервне копіювання мережі. Наведено код для налаштування програмного брандмауера.

4. Було зроблено оцінку ризиків: виявлення та визначення пріоритетності активів, виявлення загроз і вразливостей, аналіз елементів управління, визначення ймовірності інциденту та пріоритетності ризиків інформаційної безпеки. Розроблено звіт з оцінки ризиків, що відповідає вимогам сучасного ринку.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. What Is Network Security?. URL: https://www.spiceworks.com/it-security/network-security/articles/what-is-network-security/ (дата звернення: 07.10.2022).

2. What is code vulnerability?. URL: https://www.codegrip.tech/productivity/what-is-code-vulnerability/ (дата звернення: 07.10.2022).

3. Security Misconfiguration: Impact, Examples, and Prevention. URL: https://brightsec.com/blog/security-misconfiguration/ (дата звернення: 07.10.2022).

4. Symmetric vs. Asymmetric Encryption: What's the Difference? – URL: https://www.trentonsystems.com/blog/symmetric-vs-asymmetric-encryption (дата звернення: 08.10.2022).

5. What is a security vulnerability? – URL: https://www.spiceworks.com/it-security/vulnerability-management/articles/what-is-a-security-vulnerability/ (дата звернення: 08.10.2022).

6. What is Sensitive Data Exposure Vulnerability & How to Avoid It? – URL: https://securiti.ai/blog/sensitive-data-exposure/. (дата звернення: 09.10.2022).

7. Open-Source Routing and Network Simulation – URL https://www.brianlinkletter.com/open-source-network-simulators/ (дата звернення: 10.10.2022).

8. Computer Networks: A Systems Approach - Performance – URL https://book.systemsapproach.org/foundation/performance.html (дата звернення: 11.10.2022).

9. What Is Malware? – URL https://www.cisco.com/c/en/us/products/security/advanced-malware-protection/what-is-malware.html#~7-types-of-malware (дата звернення: 12.10.2022).

10. What is a DDoS attack? – URL https://www.cloudflare.com/learning/ddos/what-is-a-ddos-attack/ (дата звернення: 13.10.2022).

11. Man-in-the-middle (MITM) Attacks – URL https://www.javatpoint.com/cyber-security-mitm-attacks (дата звернення: 14.10.2022).

12. Network Transparency Definition – URL http://www.linfo.org/network_transparency.html (дата звернення: 14.10.2022).

13. What is Network Management? – URL https://www.microfocus.com/en-us/what-is/network-management (дата звернення: 15.10.2022).

14. What is a Firewall? – URL https://www.forcepoint.com/cyber-edu/firewall (дата звернення: 15.10.2022).

15. Formula for Calculating Cyber Risk – URL https://stateofsecurity.com/formula-for-calculating-cyber-risk/ (дата звернення: 16.10.2022).

16. HIPAA Risk Assessment Template – URL https://www.netwrix.com/hipaa_risk_assessment_template.html (дата звернення: 17.10.2022).

17. How to Perform IT Risk Assessment – URL https://blog.netwrix.com/2018/01/16/how-to-perform-it-risk-assessment/ (дата звернення: 17.10.2022)