МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
Факультет електроніки та інформаційних технологій

Кафедра комп’ютерних наук

Кваліфікаційна робота магістра

ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНА ТЕХНОЛОГІЯ КЕРУВАННЯ НАВЧАЛЬНИМ ЦЕНТРОМ КІБЕРБЕЗПЕКИ З ФУНКЦІОНАЛЬНІСТЮ СИМУЛЯЦІЙНОГО КІБЕРПОЛІГОНУ

Здобувач освіти гр. ІК.мз-13с Віталій КОВАЛЬ

Науковий керівник,
кандидат фізико-математичних наук,
старший викладач кафедри комп’ютерних наук Анна БАДАЛЯН

Завідувач кафедри
кандидат фізико-математичних наук, доцент Ігор ШЕЛЕХОВ

Суми 2023

Сумський державний університет

Факультет *ЦЗДВН* Кафедра *Комп'ютерних наук*

Спеціальність *122 - Комп'ютерні науки*

Затверджую:
завідувач кафедри _____

"_____" _____ 2022р.

ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТОВІ

Коваля Віталія Вікторовича
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) *ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНА ТЕХНОЛОГІЯ КЕРУВАННЯ НАВЧАЛЬНИМ ЦЕНТРОМ КІБЕРБЕЗПЕКИ З ФУНКЦІОНАЛЬНІСТЮ СИМУЛЯЦІЙНОГО КІБЕРПОЛІГОНУ*

затверджую наказом по інституту від "_ ___" _____ 20 ___ р. № _____

2. Термін здачі студентом закінченого проекту (роботи) _____

3. Вхідні дані до проекту (роботи) _____
_____
_____
_____

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) *1) Аналіз предметної області. 2) Огляд існуючих методів дослідження. 3) Опис методики дослідження бездротових мереж. 4) Практичне дослідження мережі.*

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____
_____

6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання _____

Керівник _____

(підпис)

Завдання прийняв до виконання _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів дипломного проекту (роботи)	Термін виконання проекту (роботи)	Примітка
1	*Огляд сучасного стану проблематики*		
2	*Аналіз стану проблематики*		
3	*Побудова методики дослідження*		
4	*Аналіз мережі*		
	Оформлення документації		

Студент – дипломник _____

(підпис)

Керівник проекту _____

(підпис

РЕФЕРАТ

Записка: 62 стр., 24 рис., 1 додатку, 25 літературних джерел.

Об’єкт дослідження —забезпечення стабільності бездротової мережі навчального Кііберполігону.

Мета роботи — розробка методології та практичне проведення аналізу бездротових мереж з метою забезпечення стабільності роботи мережі для задач Кіберполігону.

Результати — проведено огляд Центрів безпеки ЗВО України та специфіки розбудови Кіберполігонів.

На основі аналізу було сформовано постановку задачі та розглянуті програмні засобі дослідження.

Розроблена методика дослідження бездротових систем з визначенням критеріїв, які забезпечують стабільність роботи мережі.

Проведена практична перевірка стабільності роботи мережі з побудовою карт зон стабільності.

ІНФОРМАЦІЙНА СИСТЕМА, КІБЕРПОЛІГОН,
БЕЗДРОТОВІ МЕРЕЖІ, КІБЕРБЕЗПЕКА, ЗАХИСТ ІНФОРМАЦІЇ.

ЗМІСТ

ВСТУП

Останні роки досить велика увага приділяється захисту інформації у інформаційному просторі. Це питання стає досить актуальним оскільки велика частина нашої діяльності переміщується у цифровий простір і захист інформації та збереження конфіденційності стає все більш важливими питаннями.

Навчання спеціалістів з інформаційних технологій потребує не тільки теоретичне, а і практичне засвоєння та удосконалення знань, що потребує створення навчальних платформ нового типу – кіберполігонів.

Створення платформ дозволяє всебічно та ситуативно проводити закріплення знань та їх практичне відпрацювання, що досить вагомо впливає на якість спеціалістів.

Також такі полігони дозволяють відпрацьовувати штатні та нештатні ситуації підготовлюючи як нових так і діючих спеціалістів до нових загроз сьогодення.

Розбудова будь-якого інформаційно-комунікаційного комплексу потребує ретельного планування та врахування усіх особливостей, оскільки це фундамент на базі якого буде відбуватись створення систем. Неякісне планування може призвести до руйнування у майбутньому всієї системи.

Однією з поширених та активно застосовуємих систем є бездротові системи. Тому і розбудова фундаменту полігону повинна відбуватись з врахуванням у своїй діяльності цього типу.

В рамках даної роботи буде розглянуто питання аналізу бездротових мереж з точки зору створення стабільного покриття.

РОЗДІЛ 1 – АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Огляд проблематики

Швидкий розвиток сучасних технологій призводить до швидкого розвитку да розповсюдження інформаційного простору. У наш час важко представити якусь сферу людської діяльності у якій не залучені сучасні пристрої, гаджети, спеціальне інформаційне обладнання, тощо. Разом з розвитком інформаційних технологій та переходом великої кількості інформації та сфер діяльності збільшуються і загрози цілісності та конфіденційності інформації.

Питання захисту даних, системи, мережі стають все більш актуальними.

Підготовкою спеціалістів по виявленню та захисту систем займаються заклади вищої освіти у рамках 125 спеціальності – «Кібербезпека».

Щоб ефективно захищати інформаційний простір від кіберзагроз сучасності надзвичайно важливо навчати та тренувати фахівців, здатних до кіберзахисту на сучасних прикладах та обладнанні.

Згідно до Стандарту вищої освіти України [1] потрібно готувати спеціалістів, здатних: «..розв’язувати складні спеціалізовані задачі та практичні проблеми з забезпечення кібернетичної безпеки інформаційно-комунікаційних систем».

Під час підготовки фахівців з питань кіберзахисту потрібно баланс між теорією та практикою. Добре розуміння теорії, але без практичного закріплення зводить нанівець підготовку фахівця. Так саме бездумне практикування складових «злому» без розуміння їх природи та принципів роботи дає той самий результат. Студент, під час підготовки, повинен розуміти теорію, логіку концептуальних складових кібербезпеки, але також і вміти практично застосовувати інструментарії та теоретичні знання.

Тому під час навчання потрібно давати студентам практичний досвід з кібербезпеки та надати теоретичні навички, щоб вони могли у майбутньому стати успішними професіоналами.

Для вирішення складової практичного характеру у рамках підготовки фахівців 125 спеціальності «Кібербезпека» на базі кафедри кібербезпеки Сумського державного університету відбувається розбудова навчально-тренувального центру кібезбезпеки.

Однією з ключових особливостей є функціонально симуляційний кіберполігон.

Найчастіще Кіберполігон [2] це спеціально створена та контрольована мережева інфраструктуру. За допомогою якого можна досліджувати прояви різних атак в умовах мінімізації витрат на симуляції безпеки та тестування. Однією з головних особливостей кіберпалігону є контрольованість віртуального середовища. Результати моделювання та тестування продуктивності можна записувати, аналізувати та відтворювати. Це дозволяє запобігти проблемам у реальному житті, що дозволяє збільшити кібернетичний діапазон –підвищенню гнучкості та швидкості реагування рішень ІТ-безпеки. Також Кіберполігон можна застосовувати для онлайн-навчання в кіберпросторі, як фахівців з безпеки локального рівня так і регіонального, міжнародного масштабу для боротьби з загрозами інформаційної безпеки різного рівня та складності.

Кіберполігон інколи називають «віртуальним світом», оскільки він повністю повинен повторювати інфраструктуру об’єкту, дослідження якого проводяться.

Таким чином, кіберполігони в основному призначені для розвитку практичних навичок виявлення та реагування на інциденти інформаційної безпеки за допомогою віртуальних кібернавчань, а також можливо застосовувати для проведення різноманітного тестування апаратного та програмного забезпечення в різних галузях.

Сьогоднішній рівень інформаційних технологій та технічних потужностей дозволяє створити практично ідеальну емуляцію інфраструктури будь-якої складності: від невеликого офісу до національної корпорації або навіть спробувати спроектувати місто. При цьому «емуляцію» можна наповнити моделями реального мережевого обладнання, інструментами будь-яких виробників і розробників. Для моніторингу, контролю. і дослідження захисту модель можна наповнити будь-якими засобами, а також ізолювати в будь-якому стані: від все добре до все досить погано. Зазвичай до моделі додається відповідне віртуальне апаратне та програмне забезпечення для моделювання деяких із найпоширеніших типів атак, які застосовуються для навчання учасників. Сучасні кіберполігони також можна створювати за допомогою хмарних технологій.

Кіберполігони з боку викладачів дозволяють створювати для кожного студента окреме віртуальне лабораторне середовище: воно може включати десятки і навіть тисячі віртуальних мережевих ресурсів.

У цьому середовищі студент отримує можливість виконувати завдання незалежно від однокурсників, зберігати свої результати від заняття до заняття, ставити виконання на «паузу» та додатково готуватися до них під час самостійної роботи.

Кіберполігон це великі можливості як для навчання та і для удосконалення спеціалістів з захисту інформації. У середовищі кіберполігону можна запускати віруси та різноманітні сценарії атак, від яких майбутнім чи нинішнім фахівцям інформаційної безпеки доведеться відбиватися за допомогою наявного набору засобів захисту.

Фахівці можуть спочатку подивитись, прослідкувати, зрозуміти всі етапи «відомих» сценаріїв атак і спробувати відбити їх різними «стандартними» способами. З часом вже запускати довільні випадкові колективні сценарії, набуваючи досвіду і розуміння найбільш ефективних дій і засобів. Такий підхід дозволить не тільки вирости з теоретика в справжнього

експерта, але й підтримувати високий рівень готовності до зустрічі з новими актуальними загрозами.

За рівнем розвитку досягнень та невдач можна спостерігати практично у режимі реального часу через спеціалізовані веб-системи, які спеціально створюють для кіберполігонів.

Після закінчення «бойових дій» кіберполігон можна буде повернути у вихідний стан і він буде готовий до нового запуску.

Ще однією особливістю створення Кіберполігонів є дослідження сучасних тенденцій кібератак, оскільки, наприклад, погіршення критичної інфраструктури [3] через кібератаки може мати значний вплив на національну безпеку, економіку, засоби до існування та безпеку всіх громадян.

Але повна віртуалізація кіберполігонів призводить до ідеалізованості процесів, які у них протікають і тому є необхідність не тільки реалістичності віртуалізацій, а і обладнання Кіберполігонів реальним сучасним обладнанням, що дозволить навчатись реально доторкаючись до обладнання з яким будуть працювати у реальному житті.

Підводячи підсумок можна сказати, що Кіберполігон – це комплекс який складається з сукупності спеціалізованих програмних, програмно-апаратних та апаратних комплексів, які поєднані між собою провідними й безпровідними комунікаційними системами, обов’язково мають локальну обмежену мережу та інтегровані у мережу Інтернет, наявні системи які застосовуються для здійснення моніторингу, впливу на системи управління об’єктами, що становлять інтерес, а також системами захисту власних систем управління від аналогічних дій протиборчої сторони.

Перед початком створення та розбудову Кіберполігону був вивчений досвіт Центрів, лабораторії та навчальні аудиторії з галузі кібербезпеки, які створені і функціонують у багатьох ЗВО України.

1.2. Огляд Центрів безпеки ЗВО України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

На базі закладу розгорнуто дві лабораторії [4]:

- навчально-наукова лабораторія безпеки інформаційно-комунікаційних систем;
- навчально-наукова лабораторія технічної інформаційної безпеки.

Київський національний економічний університет імені Вадима Гетьмана: Навчально-наукова лабораторія «Полігон кібербезпеки»

На базі закладу розгорнуто (рис. 1.1) навчально-наукова лабораторія "Полігон кібербезпеки" [5].

Рис. 1.1. Загальний вигляд лабораторії «Полігон кібербезпеки»

Харківський національний економічний університет імені Семена Кузнеця: «Кіберполігон»

На базі закладу розгорнуто «Кіберполігон» [5].

Кіберполігон (рис. 1.2), який розгорнутий в університету, дозволяє дослідникам, спеціалістам, а також студентам відпрацьовувати тактики відбиття кібератак на об’єкти критичної інфраструктури, проводити симуляції кібератак з одночасним відпрацюванням різних методик кібернападів [5].

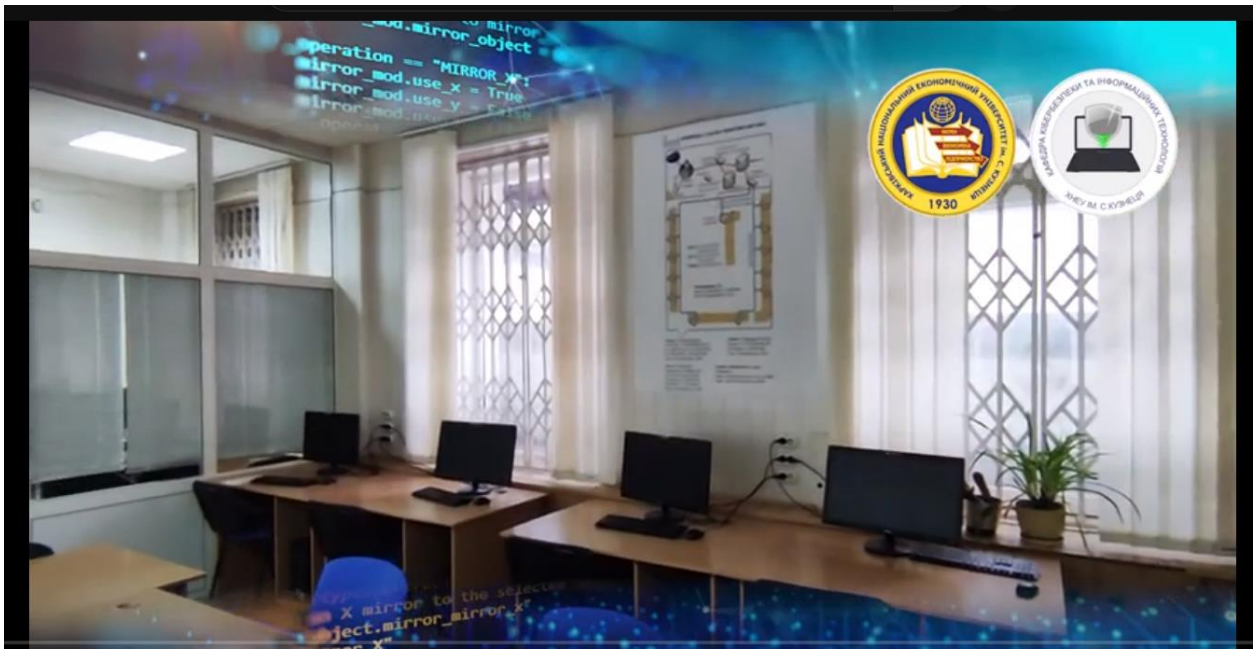

Рис. 1.2. Загальний вигляд «Кіберполігон»

Активно застосовується Кіберполігон при вивчені та практичних робіт з дисциплін:

- веб-безпека,
- безпека серверних систем,
- розширена мережева та хмарна безпека,
- бездротова та мобільна безпека.

Хмельницький національний університет: віртуальна лабораторія «Кіберполігон»

На базі закладу розгорнуто віртуальна лабораторія кіберполігон [6].

Кіберполігон в основному використовується студентами для моделювання реалізації різноманітних мережевих атак та здійснювати їх відбиття, досліджувати та виявляти вразливості програмного забезпечення (рис. 1.3), відновлювати вражену інфраструктуру а також досліджувати механізми недопущення та відновлення після атак [6].

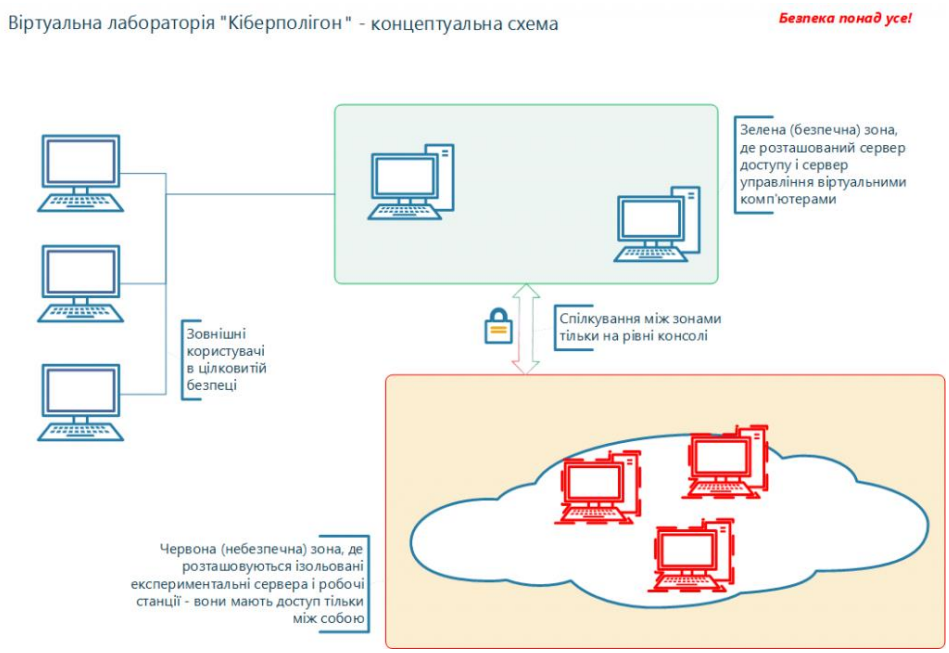

Рис. 1.3. Концептуальна схема віртуального кіберпалігону [6]

Як видно з рис. 1.3. у основі віртуального середовища лежить принцип розбиття віртуального середовища на зони:

- зелена – безпечна,
- червона – небезпечна, де проводяться дослідження.

Розділення взаємодії між зонами виконано за принципом суворого дотримання правила: робота зовнішніх користувачів з «токсичними» середовищами тільки за допомогою консолі.

На рис. 1.4 представлено фізична реалізація полігону. Як видно створені віртуальні сервери та робочі станції поєднані за допомогою ізольованого комутатора без виходу на зовні.

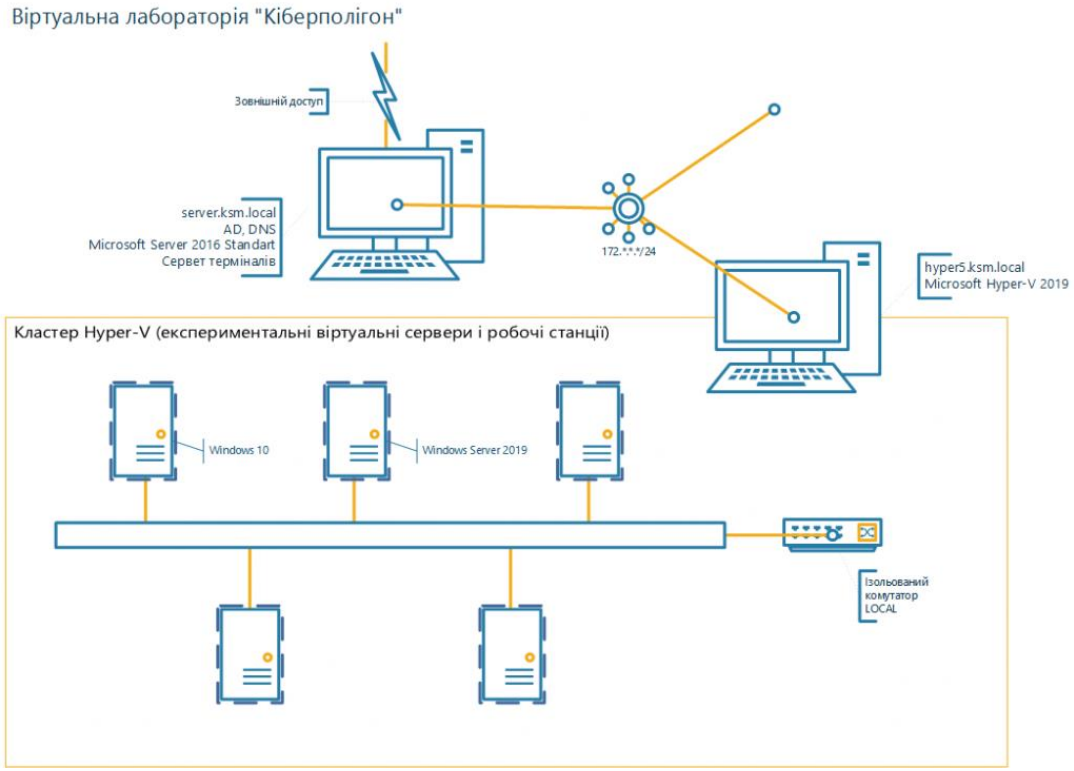

Рис. 1.4. Фізична концепція віртуального кіберпалігону [6]

Військовий інститут телекомунікацій та інформатизації імені Героїв Крут: Навчально-тренувальний комплекс кібербезпеки [7].

На базі закладу, а саме на кафедрі кібербезпеки (рис. 1.5) розгорнуто навчально-тренувальний комплекс кібербезпеки.

Основні завдання які у ньому виконуються:

- виявлення кіберзагроз;
- реагування кіберзагроз;
- протидії кіберзагроз;
- попередження кіберзагроз.

Також на даний час комплекс модернізують під можливість аналізу та розслідування кіберінцидентів

Комплекс налічує 80 автоматизованих робочих місць.

У рамках кіберполігону створений ситуаційний центр, який забезпечує оперативне управління та приймає участь у підготовки та проведенні занять та тренувань.

Доступ до мережі комплексу здійснюється з використанням двофакторної автентифікації.

На межі розділу зовнішнього та внутрішнього простору встановлений потужний брандмауер та антивірусне програмне забезпечення.

Рис. 1.5. Загальний вигляд навчально-тренувальний комплекс кібербезпеки [7]

Інші ВЗО України, такі як, наприклад, Харківський національний університет радіоелектроніки. Державний університет телекомунікацій. Національний авіаційний університет, тощо теж мають різнопланові лабораторно-навчальні комплекси для проведення підготовки спеціалістів.

Однак здебільшого. як це можна було побачити, основний акцент робиться на віртуальні та хмарні технології.

1.3. Огляд проблеми створення симуляційного спеціалізованого Кіберполігону

Швидкий розвиток технологій призводить у свою чергу до збільшення і модифікації технологій. Це у свою чергу призводить до збільшення різних напрямків та спеціалізацій спеціалістів з інформаційної безпеки.

Необхідно відзначити, що сучасний світ спрямовує велику увагу і на самоосвіту і розвиток самостійного придбання нових знань. Сучасні інформаційні цифрові технології надають великі можливості у цьому напрямку. Але вседозволеність у доступі до інформації має і другу негативну сторону проблеми і питанням контролю у цьому напрямку теж повинен займатися державний сегмент України.

Тому створення Кіберполігонів потребує і постійну підтримку у сучасному стані та адаптації під виклики сучасності, особливо підтримки вузконаправлених симуляційних задач.

Однак швидкість реагування на виклики сьогодення з боку державної системи України є не досить швидкі. Так тільки у 2021 році «Державна служба зв'язку та захисту інформації домоглася розширення кількості професій, які стосуються сфери безпеки інформації та кіберзахисту» [8,9].

25 жовтня 2021 року у класифікатор були додані наступні 17 професій:

- Аналітик загроз безпеки.
- Розробник систем захисту інформації.
- Фахівець підтримки інфраструктури кіберзахисту.
- Фахівець криптографічного захисту інформації.
- Фахівець реагування на інциденти кібербезпеки.
- Фахівець з тестування систем захисту інформації.
- Фахівець з технічного захисту інформації.
- Дізнавач сфери кібербезпеки та захисту інформації.
- Інструктор-методист інформаційної безпеки та кібербезпеки.

- Експерт-криміналіст сфери кібербезпеки та захисту інформації.
- Слідчий кіберзлочинів тощо.

Як можна бачити з [10,11] тільки у листопаді 2022 року підготовлені стандарти для нових спеціальностей фахівців і тільки для 6 із 17. Були розроблені та затверджені наступні:

- Розробник систем захисту інформації.
- Адміністратор мереж і систем.
- Фахівець сфери захисту інформації.
- Аналітик з безпеки інформаційно-телекомунікаційних систем.
- Фахівець з питань безпеки.
- Інструктор-методист з інформаційної безпеки та кібербезпеки.

Тобто, потрібно було приблизно рік для розробки та впровадження відповідних стандартів.

У свою чергу це буде потребувати відповідні спеціалізації та вузькоспеціалізовані навчання з боку Кіберполігонів.

Отже створення та розбудова Кіберполігону який буде відповідати вимогам сучасності є досить складним та комплексним завданням і ще на етапі створення концепції потрібно закласти основні принципи його функціонування та задачі, на які він буде направлений.

У рамках даної роботи головний акцент буде направлений на комунікаційні особливості функціонування центру, оскільки це є основою та запорукою «вакуумності», захищеності та надійності.

РОЗДІЛ 2 – ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ

2.1. Постановка задачі

Таким чином, на основі результатів, отриманих у розділі 1 та специфіки галузі досліджуваної роботу була сформована мету проекту – розробити комплексний підхід до аналізу стабільності і надійності бездротових мереж , які будуть застосовуватись при розбудові Кіберполігону.

Основна ціль проекту – сформувати методологію проведення аналізу бездротових мереж.

Для можливості виконання поставленої задачі, вирішено поділити основну задачу на підзадачі:

1. Дослідити існуючи програмні продукти для дослідження бездротових мереж..
2. Виділити основні критерії, які необхідно для стабільності функціонування Кіберполігону..
3. Розробити концепцію методики проведення дослідження бездротових мереж.

2.2. Методи дослідження

Для реалізації поставлених цілей більш детально ознайомимся з існуючими програмними засобами, які дозволяють аналізувати та діагностувати безпровідні мережі .

Ekahau Connect

Досить потужне та дороговартісне програмне забезпечення, яке дозволяє створювати теплові карти Wi-Fi. Використовується для аналізу,

оптимізації та має можливість професійного моделювання бездротової мережі за допомогою візуальних теплових карт Wi-Fi високої чіткості, що дозволяє усунути припущення щодо покриття та продуктивності мережі. Цей додаток (рис. 2.1) для його максимального результату потребує витрат немало часу. Необхідно запустити програму пройтись по території об’єкту, зняти спектрограму і створити теплову карту, яка покаже щільність покриття Wi-Fi. [12]

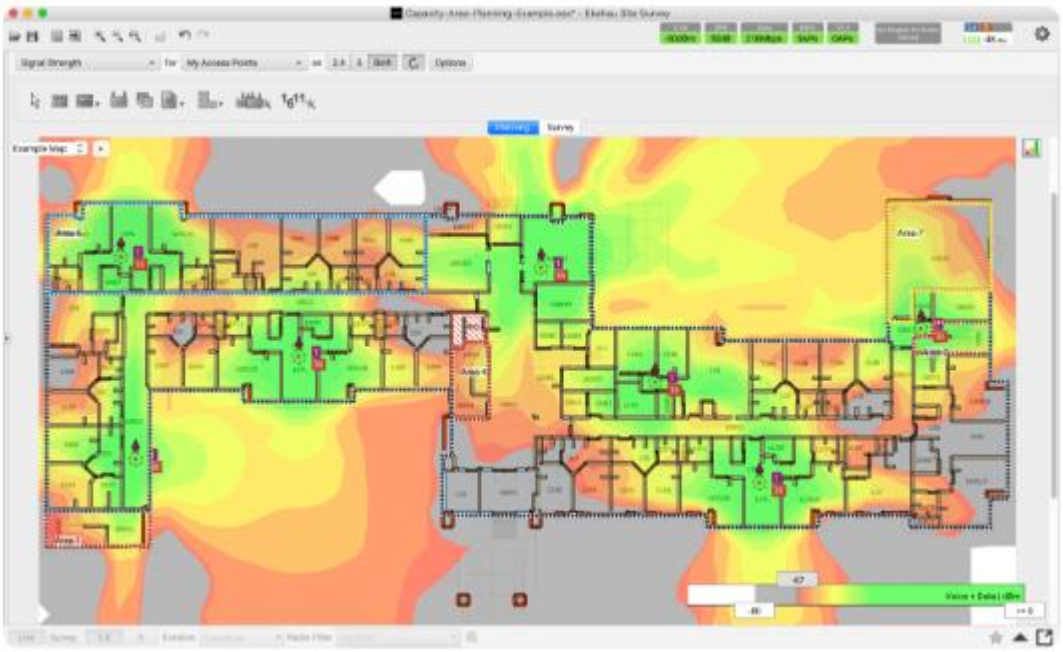

Рис. 2.1. Ekahau HeatMapper

Після проведених досліджень, можна зберігти отримані дані та продовжити дослідження отриманої теплової карти. Додаток особливо корисний у тому випадку, коли тільки лише починає розбудова бездротової мережі, оскільки по-перше можна змоделювати теплову карту опираючись на можливості обладнання, а по-друге є можливість вибрати місце, у якому розміщення роутера є оптимальним. Також система дозволяє, визначати найбільш ймовірні місця встановленого обладнання та його параметри безпеки.

Acrylic Wi-Fi Home

Додаток Acrylic Wi-Fi Home (рис. 2.2) є теж потужним та платним додатком, який дозволяє створювати теплову карту бездротової мережі. Його функціонал дозволяє отримувати дані та аналізувати дані з усіх джерел у межах дії приймача бездротового сигналу.

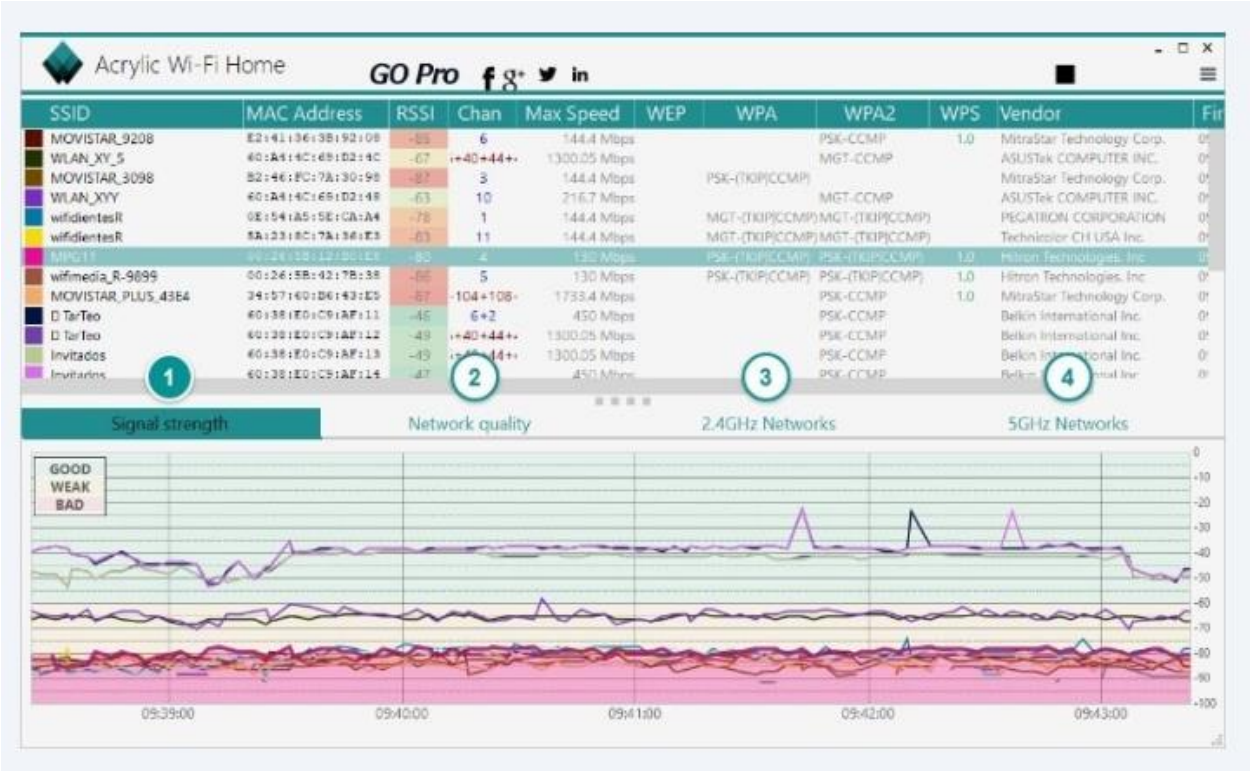

Рис. 2.2 Acrylic Wi-Fi Home

Мінімальний стандартний набір загальних даних, які дозволяє отримати додаток, про бездротові мережі [13]:

- назву мережі (SSID),
- MAC-адресу,
- використовувані канали,
- тип шифрування,
- виробника,
- тип 802.11,
- максимальну швидкість роутера, тощо.

Додаток також дозволяє вимірювати «силу» мережі – RSSI (показник рівня сигналу каналу, що приймається). Дозволяє записувати та будувати графік, як змінюється потужність кожної мережі протягом тривалого часу.

Для отримання повної картини теплової карти необхідно, наприклад, з ноутбуком переміщуватись по приміщенню та спостерігати за зміною потужності сигналу.

Homedale

Додаток Homedale – це Wi-Fi аналізатор (рис.2.3), який працює як виконуваний файл і у одній з комплектацій не потребує інсталяції на ПК [14]. Дозволяє отримувати наступні дані про знайдені джерела бездротового сигналу:

- SSID
- MAC-адреси
- рівень сигналу
- тип шифрування.

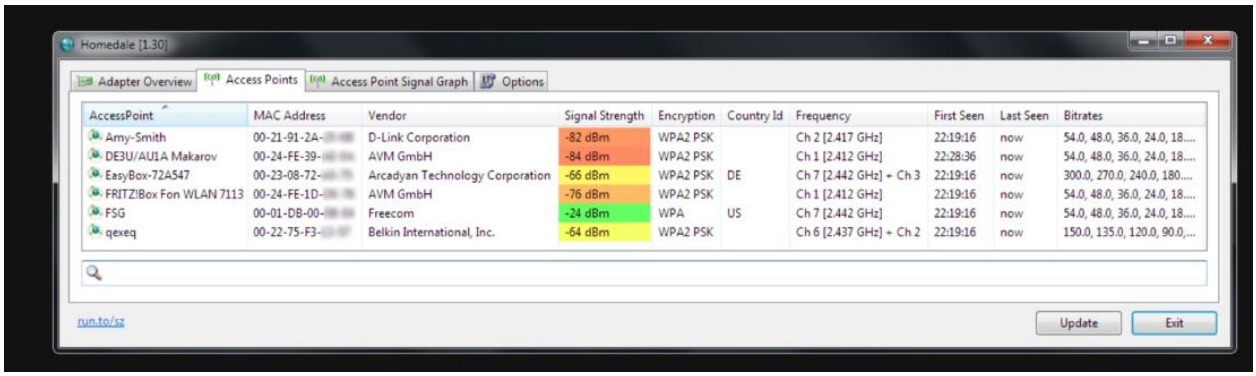

Рис. 2.3 Homedale

Technitium MAC Address Changer

Хоча головне завдання додатку полягає трохи у іншому, але Technitium MAC Address Changer (рис. 2.4) дозволяє отримувати детальну інформацію про доступні мережі та дані про трафік з'єднання. Дозволяє змінювати MAC-адресу комп'ютера для того, щоб перевірити ефективність спрацювання фільтрації роутера по MAC-адресам: блокування підключення сторонніх пристроїв до бездротової мережі [15].

Оскільки у стандартній типізації кожний пристрій, який підключається до мережі, зобов'язаний мати унікальний MAC-адрес. Одним із способів, який дозволяє у мережі налаштувати безпекові можливості, є використання фільтрації та ідентифікації за допомогою MAC-адрес.

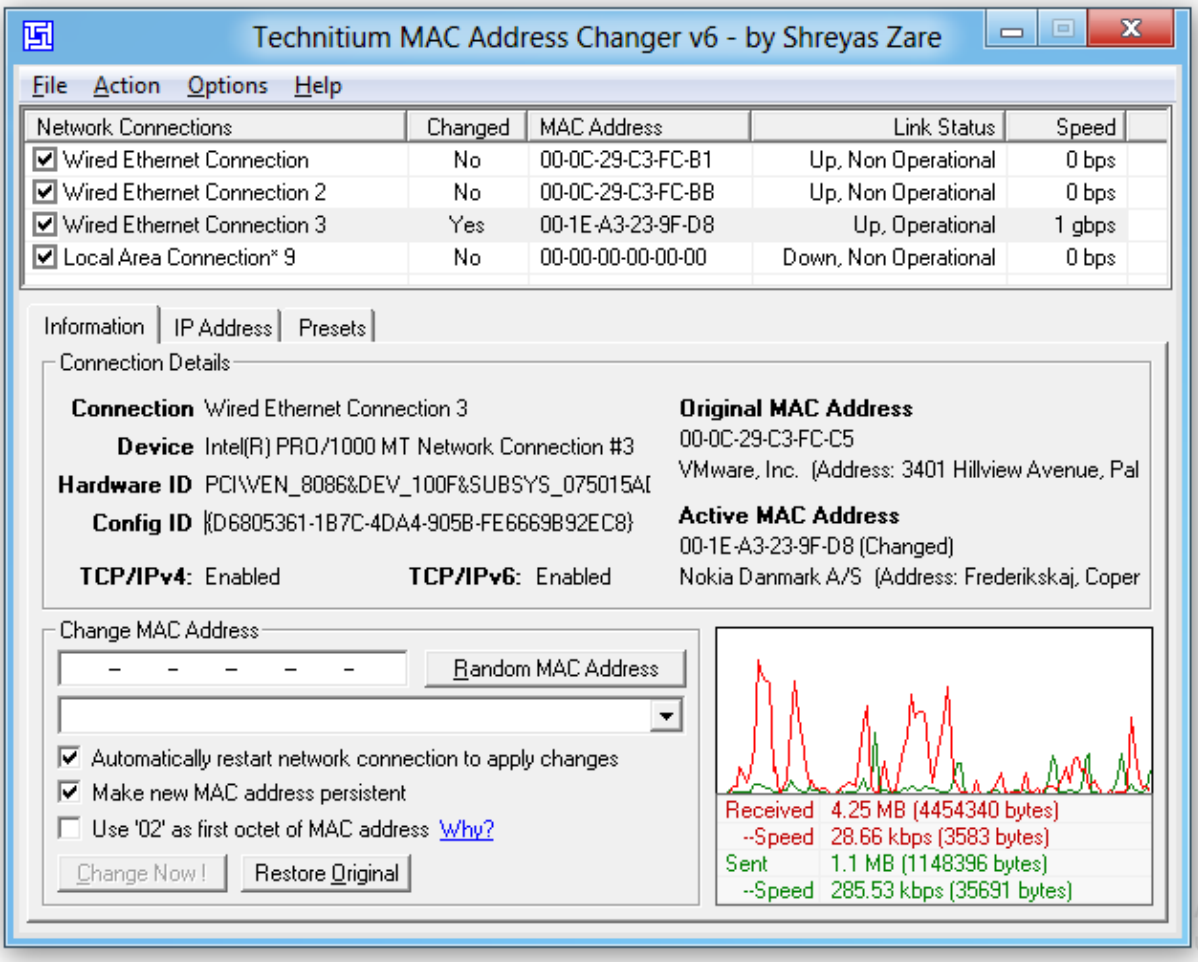

Рис. 2.4 Technitium MAC Address Changer

TamoSoft Throughput Test

На відміну від попередніх програм, основна особливість роботи яких це можливість визначити силу сигналу бездротової мережі, для отримання інформації про фактичну пропускну здатність можна застосовувати додаток TamoSoft Throughput Test (рис.2.5) [16].

Перевірка пропусної здатності базується на TCP і UDP. Адаптований для ОС Windows.

Останню версію додатку вже можна запустити на ОС Mac, однак вимірювання пропускної здатності між комп'ютерами на ОС Windows та ОС Mac досі проблематично.

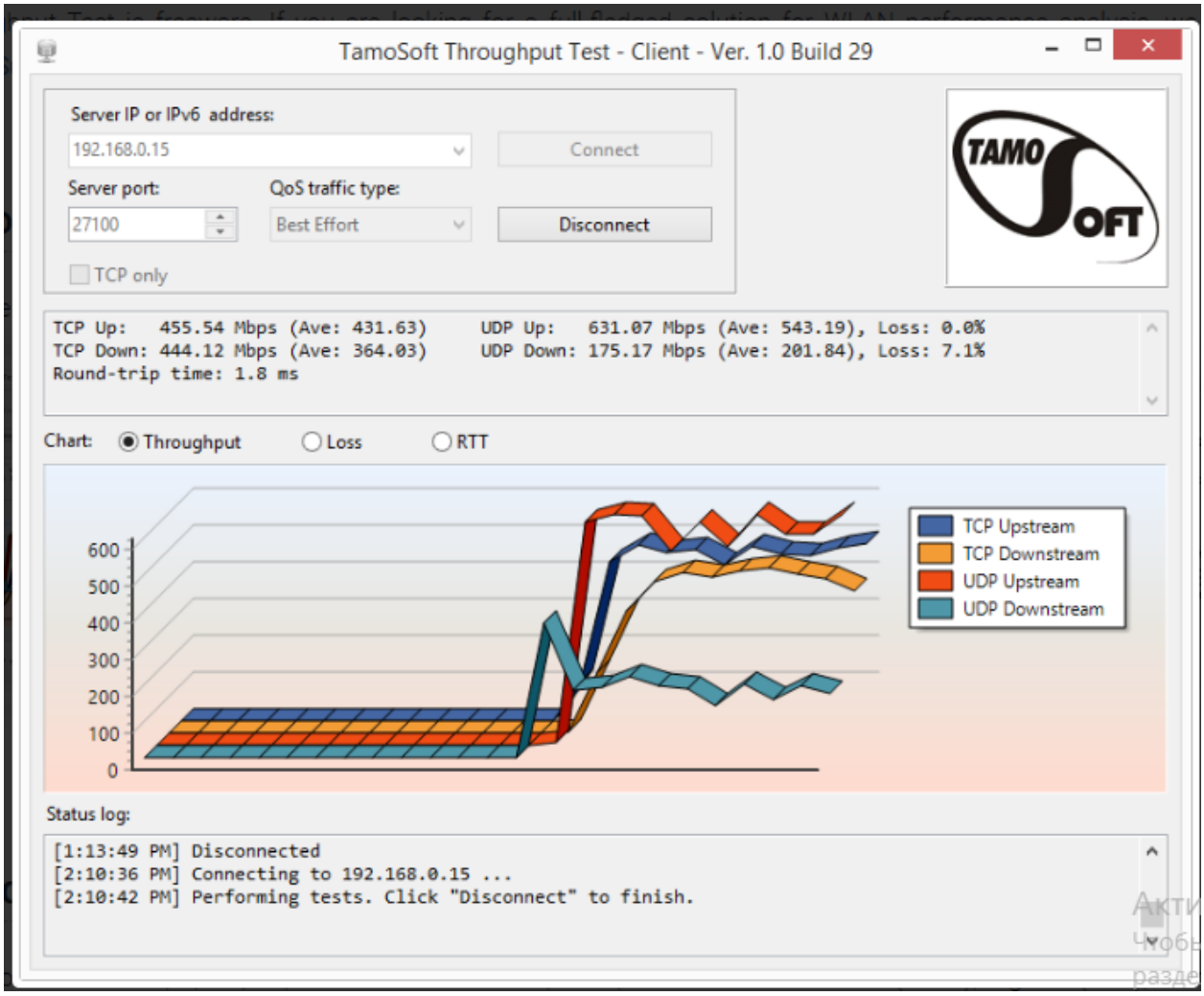

Рис. 2.5 TamoSoft Throughput Test

Ookla Speedtest

Найпростіший, але досить ефективним, методом для вимірювання швидкості є ресурс Ookla Speedtest [17] (рис. 2.6).

До його переваг можна віднести:

Відсутність необхідності завантажувати ПЗ: даний ресурс працює на усіх ОС де встановлений будь-який браузер.

Просте використання – варто перейти на сайт, натиснути «Розпочати тест», і отримати статистику швидкості передачі та прийому даних.

Це досить гарний інструмент для отримання інформації стосовно щодо пропускної здатності інтернет каналу.

Однак потребує наявності постійного Інтернет зв’язку і дозволяє виміряти канал Інтернет з’єднання, а не можливості локальної бездротової мережі.

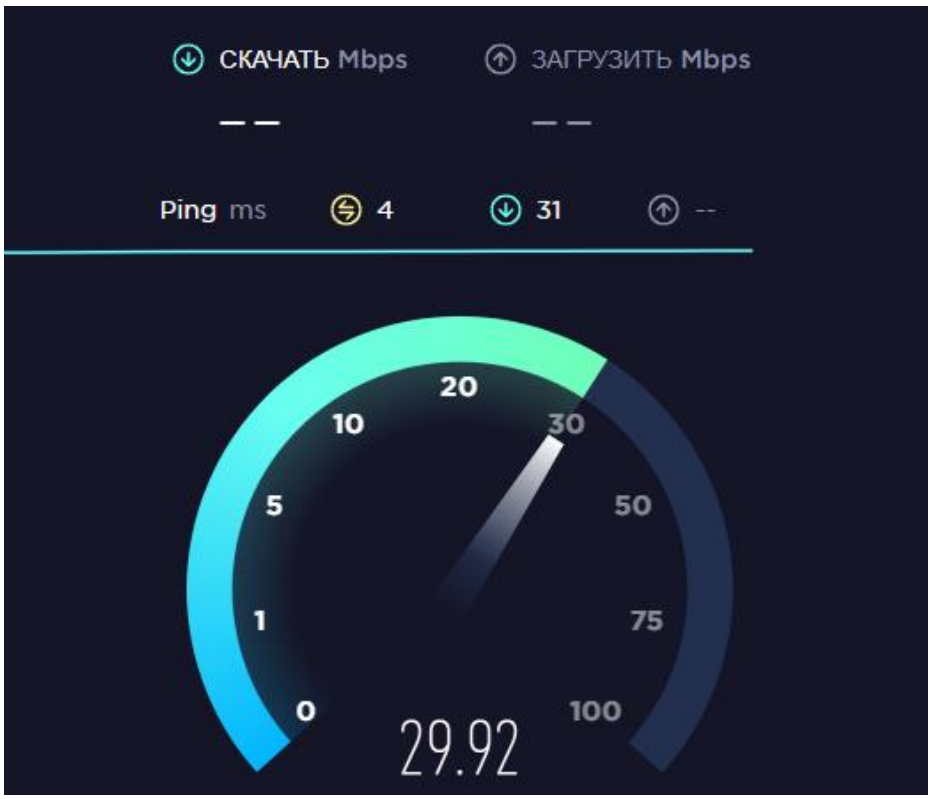

Рис. 2.6 Ookla Speedtest

WirelessNetView

WirelessNetView (рис.2.7) — програма, створена для деталізації бездротової локальної мережі і відображення зібраної інформації. Має досить простим і привабливим графічним інтерфейс [18].

Має можливості: виявити несанкціоновані точки доступу, відображає підключених клієнтів, можна сканувати бездротові мережі на 2,4 ГГц та 5 ГГц, будувати графіки сигналу і їх потужності.

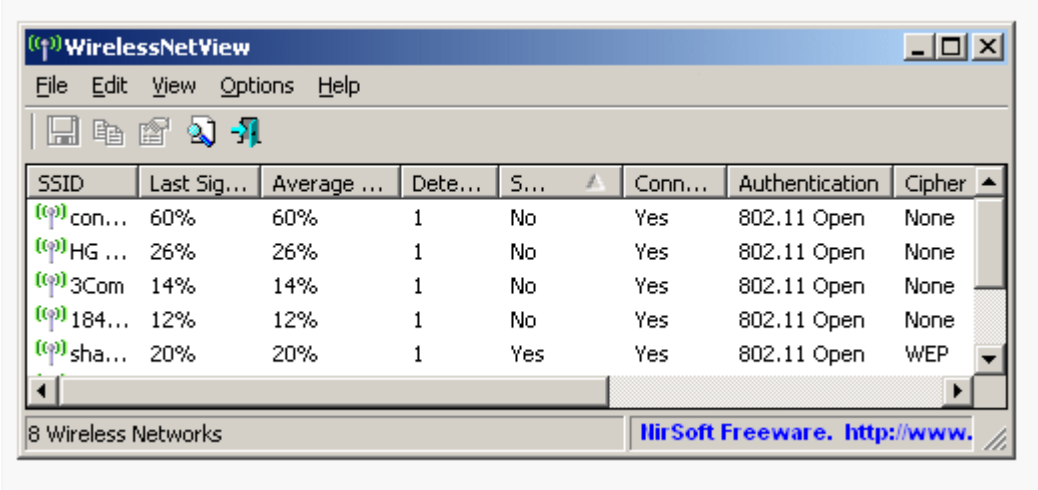

Рис. 2.7 WirelessNetView

Free Wi-Fi Scanner

Free Wi-Fi Scanner (рис. 2.8) являє собою безкоштовний і простий додаток, який дозволяє знаходити інтернет-мережі, визначати рівні сигналу і параметри доступності мережі[19].

Працює з підключеним або базовим бездротовим адаптером. Можна застосовувати фільтрування списку знайдених мереж за параметрами, здійснювати підключення до них і визначення їх стандартів шифрування.

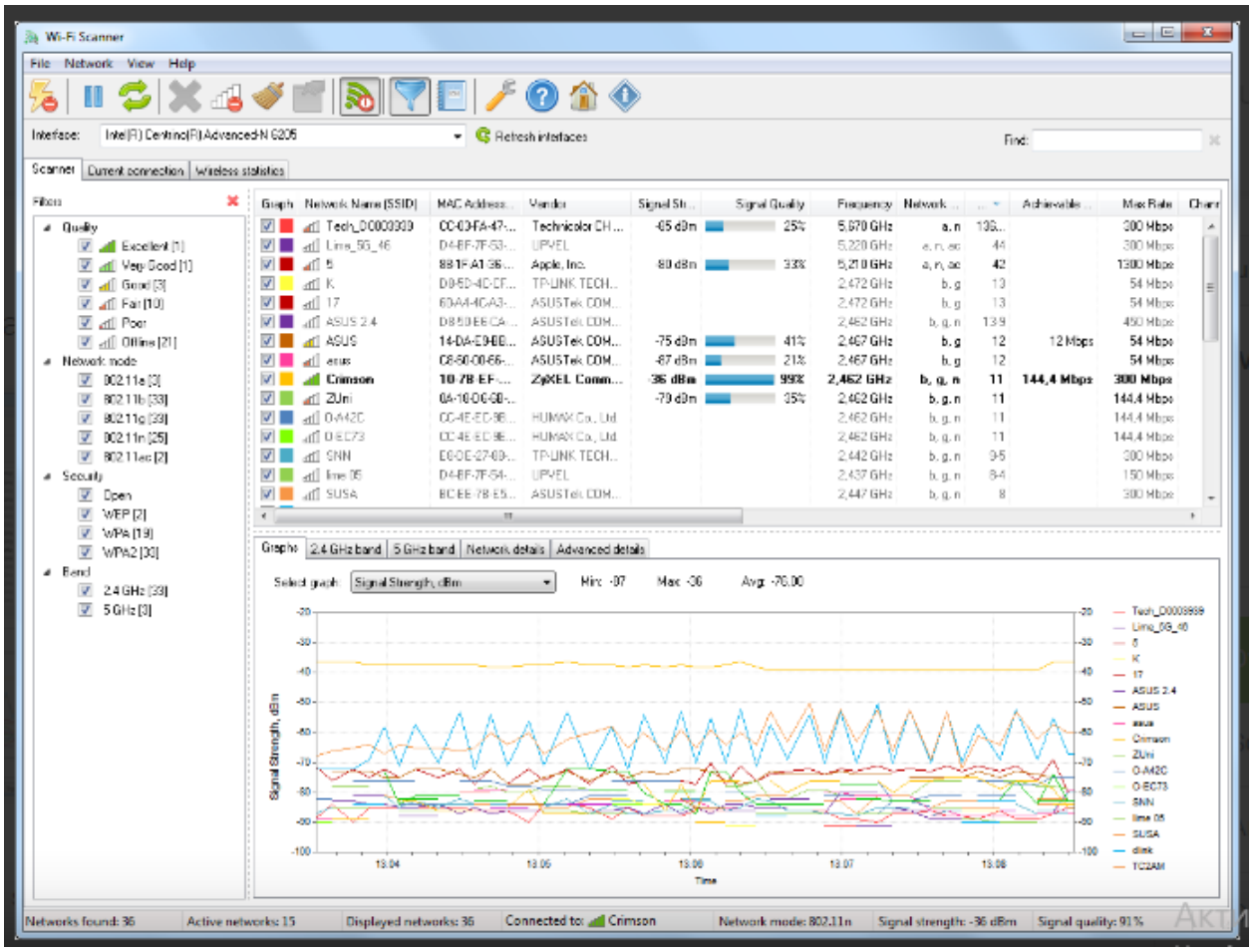

Рис. 2.8 Free Wi-Fi Scanner

LinSSID

LinSSID (рис.2.9) додаток, основна мета якого це моніторинг зайнятості ефіру та аналіз стану завантаження бездротових мереж [20]. Має мінімальну кількість налаштувань для запуску аналізування бездротових мережевих ресурсів, сканування і відображення інформації. Інформацію, яку вдалось отримати показує у вигляді графіків, зокрема, потужність сигналу з адресою, типом безпеки даних (рис. 2.9).

За своїми функціональними характеристиками та зовнішнім виглядом досить сильно нагадує Inssider. Встановлення безкоштовну, використовуючи вихідний код з офіційного сайту компанії.

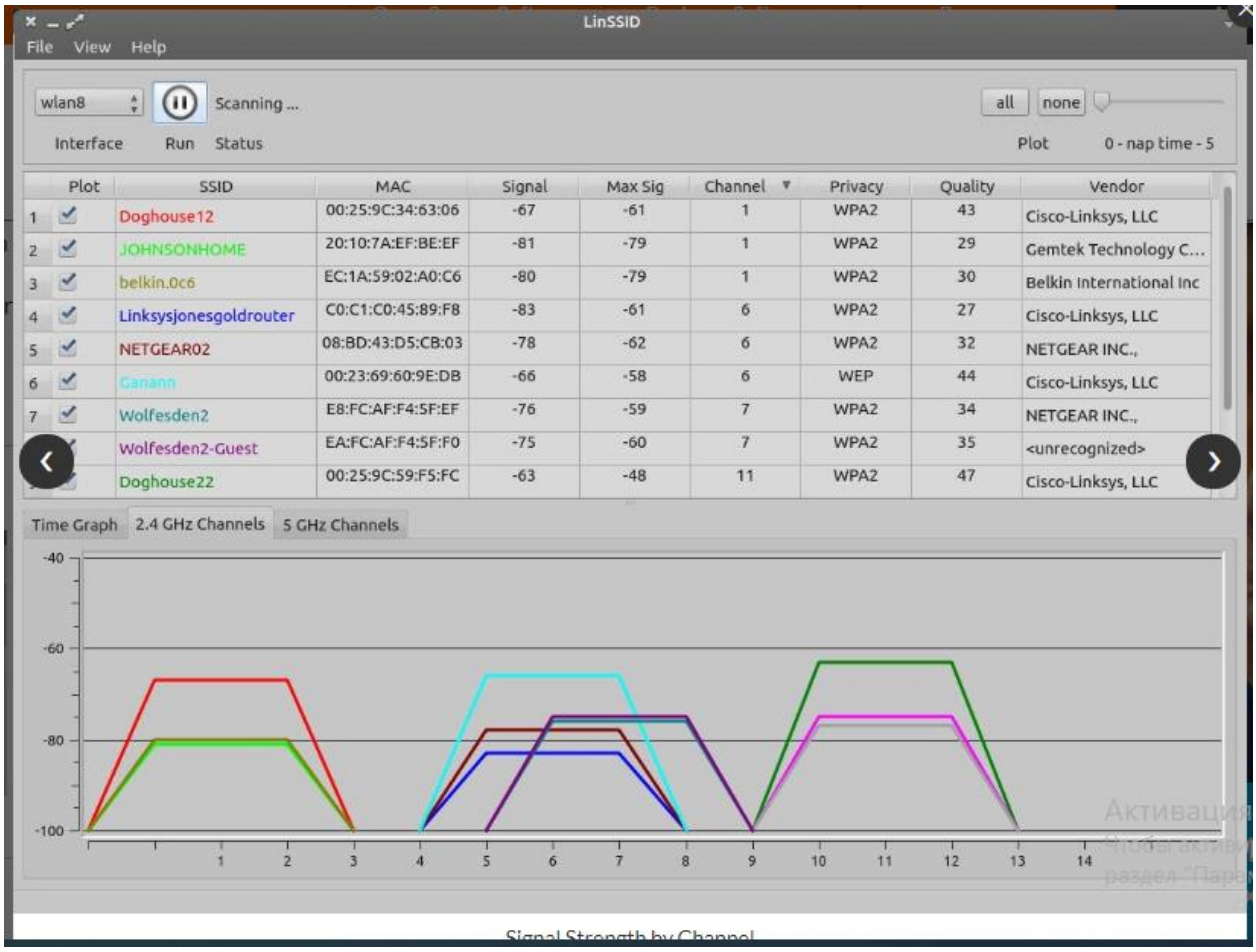

Рис. 2.9 LinSSID

iwScanner

iwScanner (рис. 2.10) додаток сканер безпровідних мереж. Дозволяє отримати інформацію про виявлені бездротові мережі, має можливість встановити швидкість сканування.

Дозволяє запустити сканування вибраної бездротової мережі та будувати графік у реальному часі[21].

iwScanner побудована на консольній утиліті для роботи з бездротовими з'єднаннями iwlist (iwtools), тому для повного функціонування потрібні права адміністратора (root).

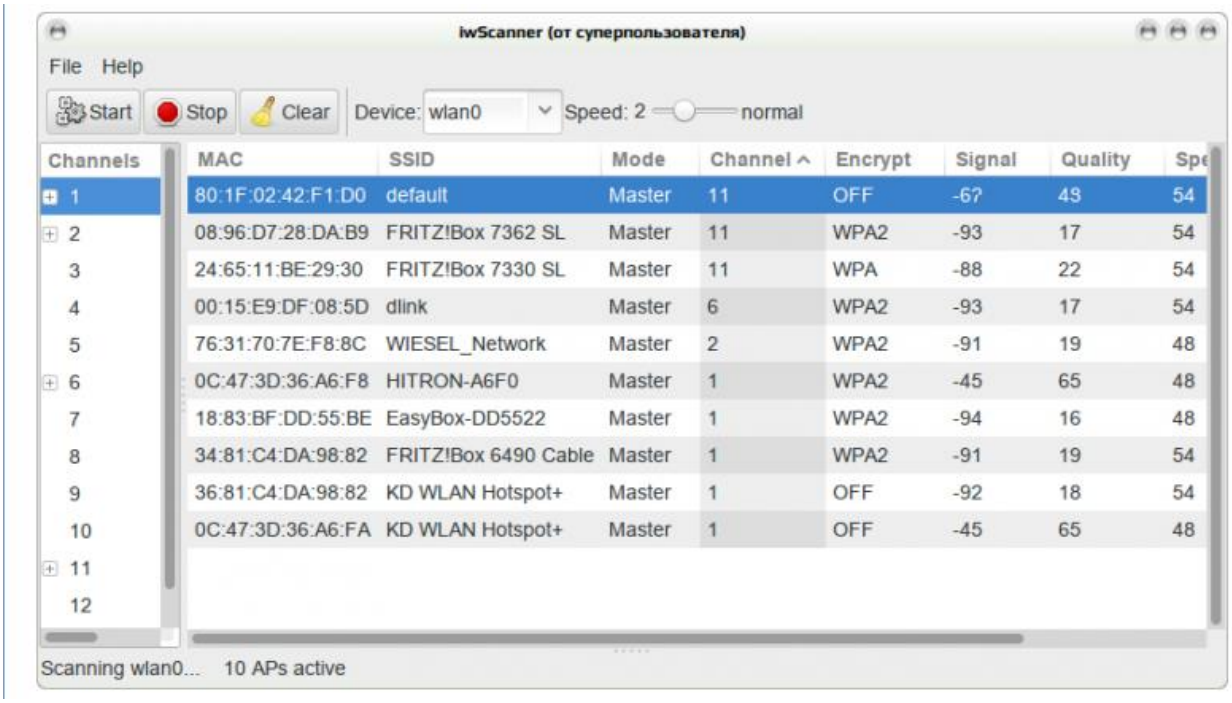

Рис. 2.10 iwScanner

WIFISCANNER

WIFISCANNER (рис.2.11) – це сканер створений для виявлення, виправлення проблем з інтернет-з'єднанням на ОС Mac OS [22]. Він може отримати та відобразити дані про підключені бездротові мережі з коротким описом мережевих типів і оптимальних каналів.

Одна з його переваг полягає у розрахунку та показі рівнів шуму та детальної інформації про сигнальні характеристики.

Програма дозволяє захоплювати пакети, та зберігати їх. Це дозволяє проводити подальший сторонній пакетний аналіз, тобто робити комплексний аналіз бездротових мереж [22]. Може досліджувати канали: 2,4 ГГц, 5 ГГц і 6 ГГц. Кожній точці доступу надавати спеціальну мітку, експорт результатів у файли зі значеннями, створення зведених звітів у форматі HTML, підрахунок кількості каналів, які перекриваються, підтвердження служб розташування для сканування в Mojave.

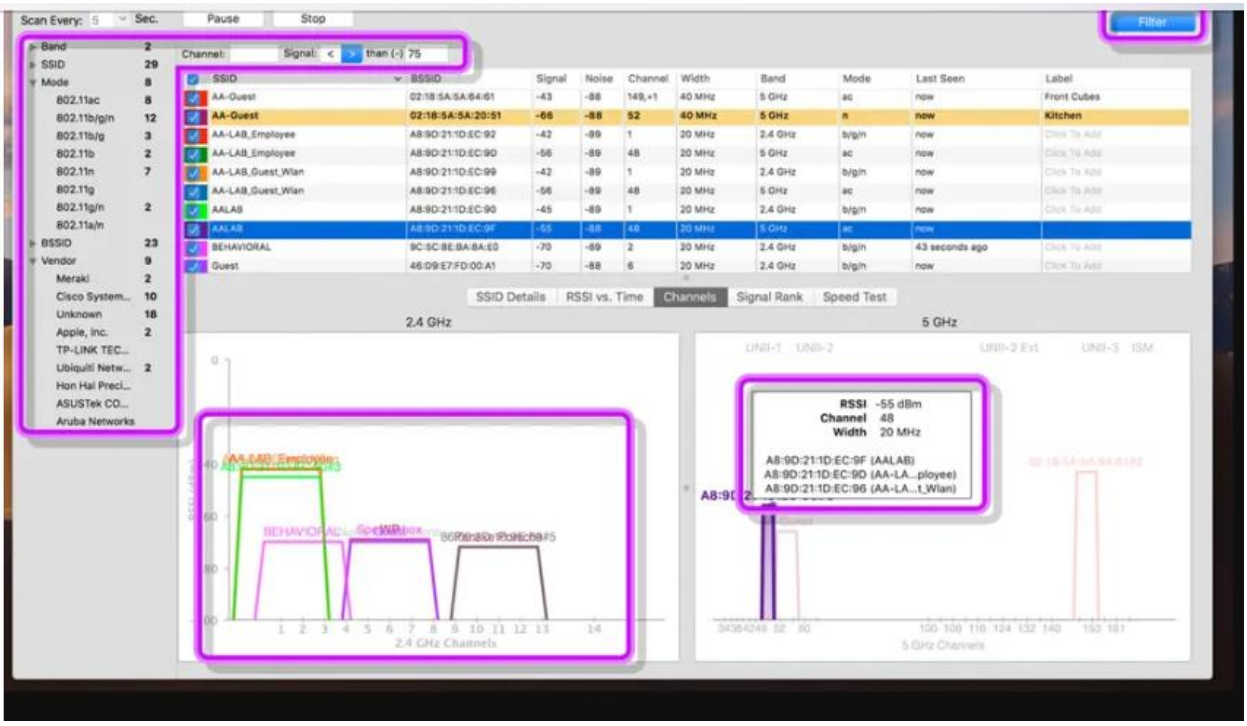

Рис. 2.11 WIFISCANNER

Wifi Analyzer

Wifi Analyzer (рис.2.12) – це додаток, який дозволяє отримувати дані про бездротові системи за допомогою мобільного пристрою.

Основні можливості додатку:

- Показує графіки співвідношення рівня сигналу.
- Показує кількість каналів для кожної мережі, властивості Ad Hoc, шифрування.
- Проводить ранжування каналів за наповненістю.
- Показує інформацію про мережу після підключення (IP, Gateway, DNS, ServerIP, Link Speed, Hidden SSID, Local Mac)
- Дозволяє проводити пошук точок доступу.
- Надає приблизні розрахунки про відстань до точок доступу.

Рис. 2.12 Wifi Analyzer

Усі вище наведені програмні продукти у більшій своїй мірі дозволяють досить добре проводити **статичний** аналіз бездротових мереж.

Як відомо [23] на якість та стабільність робіт бездротових мереж досить сильно впливає велика кількість зовнішніх факторів, найголовніші з них це неправильне налаштування обладнання сусідніх мереж, що призводить до інтерференції та колізій.

Оскільки найчастіше убрати зовнішні завади є досить важко, оскільки фізично Кіберполігон розташовується у кампусі Сумського державного університету та ізолювати його неможливо.

Потрібно провести і динамічні дослідження бездротової мережі Кіберполігону.

РОЗДІЛ – 3 ОПИС МЕТОДИКИ ДОСЛІДЖЕННЯ БЕЗДРОТОВИХ СИСТЕМ

Перед проведенням формування методики аналізу стабільності потрібно виділити основні параметри, якими будемо оперувати і важливі для стабільності функціонування Кіберполігону.

Для стабільного функціонування Кіберполігону потрібно для бездротових систем стабільний зв'язок, а саме:

- пінг,
- швидкість,
- рівень сигналу.

Саме за цими критеріями будемо проводитись статичний та динамічний аналіз бездротових систем.

Узагальнений алгоритм проведення дослідження стабільності бездротової системи наступний:

1. Проведення статичного аналізу.
2. Побудова карти зон статичної стабільності мережі.
3. Проведення динамічного аналізу.
4. Побудова карти динамічних зон стабільності.
5. Порівняння зон стабільності з визначенням зон якісного покриття.

Для проведення статичного аналізу можна вибрати одну з описаних програм у розділі 2. За допомогою програмних засобів відбувається вимірювання рівню сигналу на відповідній території Кіберполігону і по результатам вимірювань будується карта – карта зон стабільності мережі.

Для вимірювання сили сигналу Wi-Fi найчастіше використовується показник рівня сигналу, що приймається: RSSI (received signal strength indicator) – повна потужність сигналу, що приймається приймачем. Даний параметр вимірюється приймачем у дБм.

RSSI може приймати значення у діапазоні 0 – -100 дБм. Чим блище значення до 0 тим сигнал вважається кращий тобто потужнішим.

Для більш-менш якісної роботи прийнято вважати значення RSSI не нижче -65 дБм. При низьких значеннях вже спостерігається зниження якості, це проявляється у: збільшенні швидкості підключення, втратах пакетів, повторній передачі даних, тощо.

Визначимо наступні зони значення відповідності сили сигналу Wi-Fi та його якості:

- зона «A» – відмінні показники сигналу: від -35 до -50 дБм;
- зона «B» – хороші показники сигналу: від -50 до -65 дБм;
- зона «C» – задовільні показники сигналу: від -65 до -75 дБм;
- зона «D» – погані показники сигналу: від -75 до -85 дБм;
- зона «E» – непридатні значення сигналу: від -85 до -100 дБм.

Розбиття на такі числові значення є наближеними. Оскільки значення сигналу ще не є показником якості. Сила сигналу Wi-Fi залежить не тільки від показника RSSI, але й від інших факторів (від завантаженості радіоефіру, від потужності сигналу точки доступу, від перешкод, характеристик мобільного пристрою).

Проводити динамічний аналіз будемо за допомогою програмного додатку iPerf3 [24].

Iperf3 – це кросплатформова консольна клієнт-серверне програмне забезпечення. Програма – генератор TCP та UDP трафіку, що дозволяє тестувати пропускну здатність мережі. З її допомогою можна вимірювати пропускну здатність мережі між будь-яким сервером та клієнтом, проводити навантажувальне тестування каналу зв'язку.

Для завантаження доступні версії утиліти для різних ОС (Windows, MacOS, Ubuntu, Debian, Mint, Fedora, Red Hat, CentOS, OpenSUSE, Arch Linux, FreeBSD).

Для мобільних пристроїв з ОС Android можна скористатися програмою Magic iPerf including iPerf3.

Для виконання тестування програму слід запустити на двох пристроях (це можуть бути як комп'ютери, так і смартфони, планшети).

Один прилад буде виконувати роль сервера, а інший роль клієнта. Між ними відбуватиметься передача даних для вимірювання пропускної здатності з'єднання.

Для отримання якісних та більш точніших даних вимірювання передачі даних по мережі рекомендується:

- залишити при проведенні тестування включеними лише 2 хости, які братимуть участь у тестуванні;
- перед тестуванням визначити локальну IP-адресу пристрою, який буде виступати у ролі сервера;
- перед тестуванням виставити фізичну канальну швидкість на максимальне доступне значення;
- на хостах, які будуть приймати участь у тестуванні, закрити всі запущені користувацькі програми та програми, що передають дані по мережі;
- на час тестування, відключіть на хостах брандмауери та мережеві екрани, або мінімум налаштувати проходження трафіку робочими портами iperf3;
- виконувати тестування кілька разів у одній точці;
- для повної картини виконувати передачу трафіку від клієнта серверу, вимір вихідної швидкості та від сервера до клієнту – вхідна швидкість (Reverse mode).

При вимірюванні каналу зв'язку між пристроями потрібно, щоб проміжних мережевих пристроїв була мінімальна кількість. Так, якщо один хост підключено на швидкості 1 Гбіт/с, а інший на швидкості 100 Мбіт/с, то при тестуванні каналу швидкість не перевищить 100 Мбіт/с.

Для мінімізації втрат та більш якісних результатів досліджень сервер краще використовувати локальний та встановити на ПК, який знаходиться на прямому з’єднанні з джерелом бездротової мережі.

Для запуску у режимі сервер (консольний варіант) є декілька параметрів, які потрібно налаштувати [24]:

- -s, --server: запустити iPerf у режимі сервера;
- -D, --daemon: запустити сервер у фоновому режимі як сервіс Windows.
- -I, --pidfilefile: записувати у файл ID процесів.
- -p, --port <n>: порт сервера для прослуховування сервером і підключення клієнта. Значення порту має бути однаковим як на клієнті, так і на сервері. За замовчуванням номер порту 5201.

Базова консольна команда для запуску сервера буде наступна:

```
iperf3 –s -p 5555
```

Сервер iperf3 запущено, він прослуховує та очікує з'єднання на порту TCP/5555.

З боку клієнта iperf3 виникає ряд питань, оскільки згідно до офіціальної документації [24] додаток має досить велику кількість параметрів налаштування. При виборі їх потрібно оперувати поставленим задачам та цілям дослідження.

Для дефолтового налаштування під задачі перевірки стабільності зв’язку використовуємо наступні параметри:

- -c, --client: запустити iPerf у режимі клієнту;
- -t <sec>: час тестування, у секундах. За замовчуванням 10 секунд;
- -n:<> обсяг трафіку, який необхідно передати при тестуванні;

- -p, --port <n>: порт серверу до якого підключаємось. Значення порту має бути однаковим як на клієнті, так і на сервері. За замовчуванням номер порту 5201;
- -R: зворотний режим (Reverse mode). За замовчуванням сервер приймає дані, а клієнт відправляє.

Отже для тестування вихідної швидкості 30 секунд базова консольна команда буде мати наступний вигляд:

```
iperf3 -c paris.testdebit.info -p 9200 -t 30
```

вхідної швидкості базова консольна команда:

```
iperf3 -c paris.testdebit.info -p 9200 -t 30 -r
```

Для тестування вихідної швидкості по передачі файлу розміром 100Мб базова консольна команда буде мати наступний вигляд:

```
iperf3 -c paris.testdebit.info -p 9200 -n 100M
```

вхідної швидкості базова консольна команда:

```
iperf3 -c paris.testdebit.info -p 9200 -n 100M -r
```

Для перевірки стабільності зв’язку потрібно тестувати мінімум або 20 хвилин або передача трафіку 1Гбайт. Такі значення дозволяють чітко проявити проблеми у бездротовій мережі.

На рис. 3.1. продемонстровано результат роботи вище наведених команд та отримані дані.

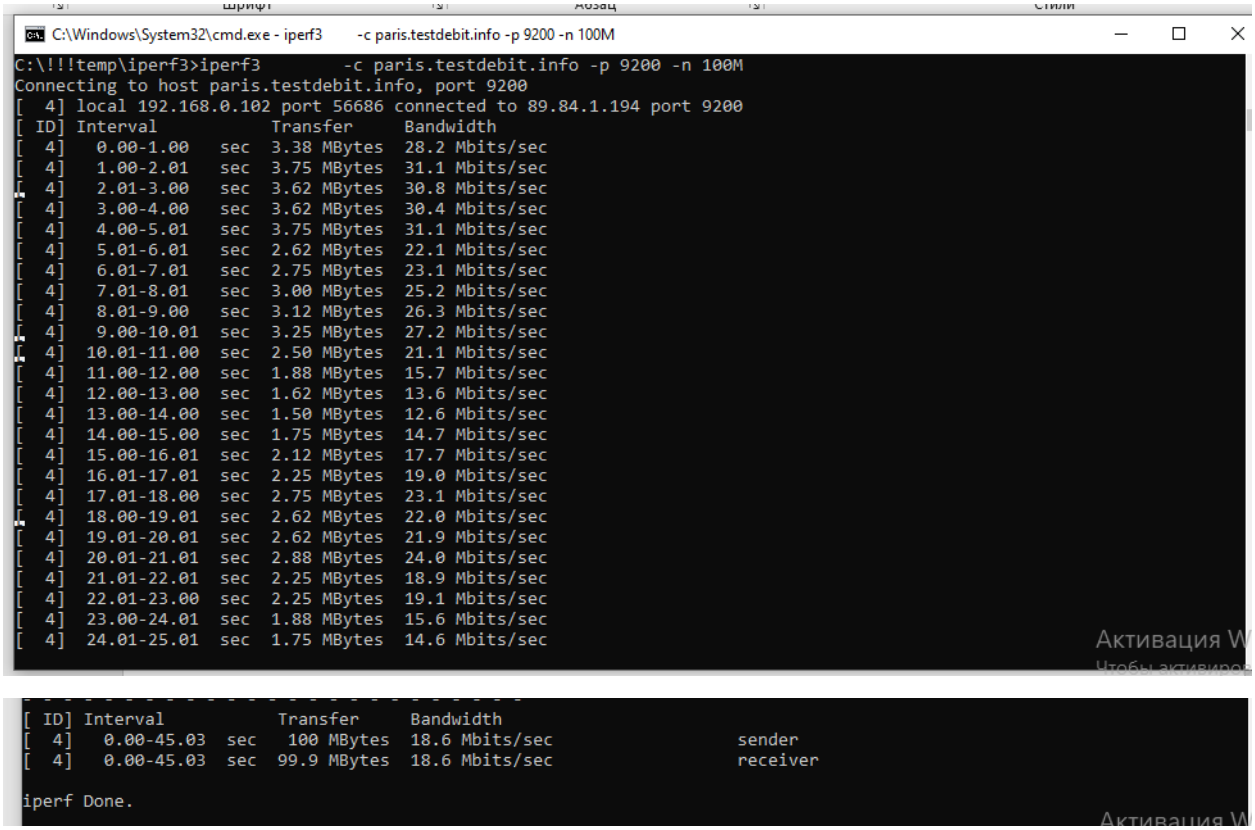

Рис. 3.1 Етапи виконання перевірки

Як можна бачити (рис. 3.1) отримуємо поточну та підсумкову інформацію, яка дозволяє провести аналіз та встановити параметри стабільності зв’язку.

Усі необхідний інструментарій та принципи роботи наведені і можна приступати до практичного застосування та апробації методики.

РОЗДІЛ 4 – ПРАКТИЧНЕ ДОСЛІДЖЕННЯ БЕЗДРОТОВОЇ МЕРЕЖІ КІБЕРПОЛІГОНУ

Опираючись на результати розділу 3 було проведено дослідження стабільності бездротової мережі Кіберполігону.

Оскільки Кіберполігон знаходиться на етапі розбудови то дослідження були проведені для 1 джерела бездротового сигналу, а саме для роутера TP-LINK Archer A64 [25].

4.1. Статичний аналіз

Кафедра кібербезпеки СумДУ розташована у Центральному корпусі кампусу. Вимірювання будемо проводити на 2 поверсі центрального корпусу на території кафедри (рис. 4.1). Правильно б було дослідити всі поверхи та територію за межами для отримання повної картини, але мета роботи дослідження саме території Кіберполігону та кафедри.

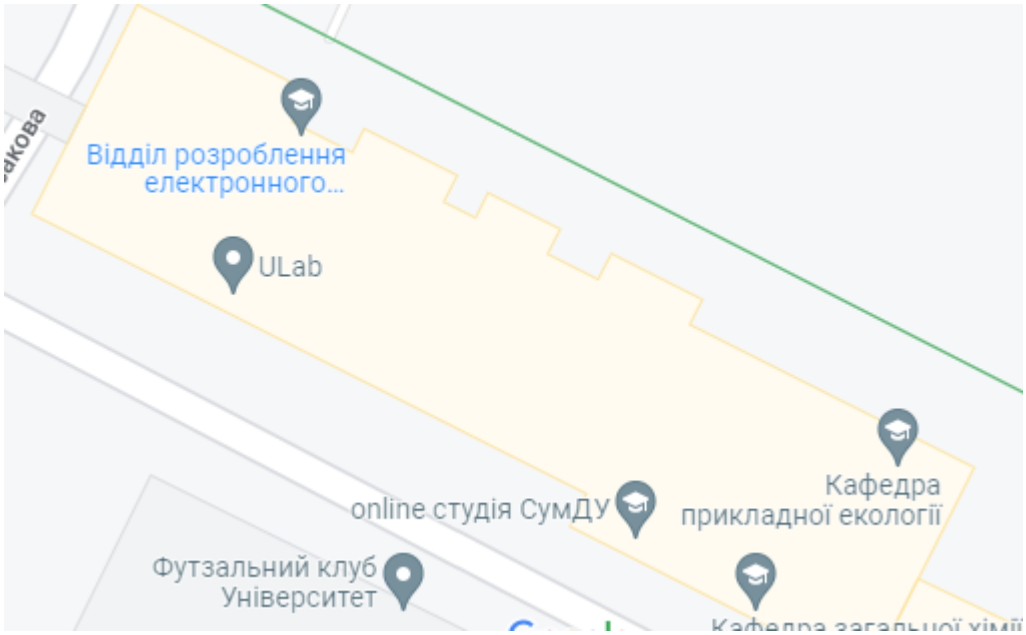

Рис. 4.1. Загальний вигляд Центрального корпусу кампусу СумДУ

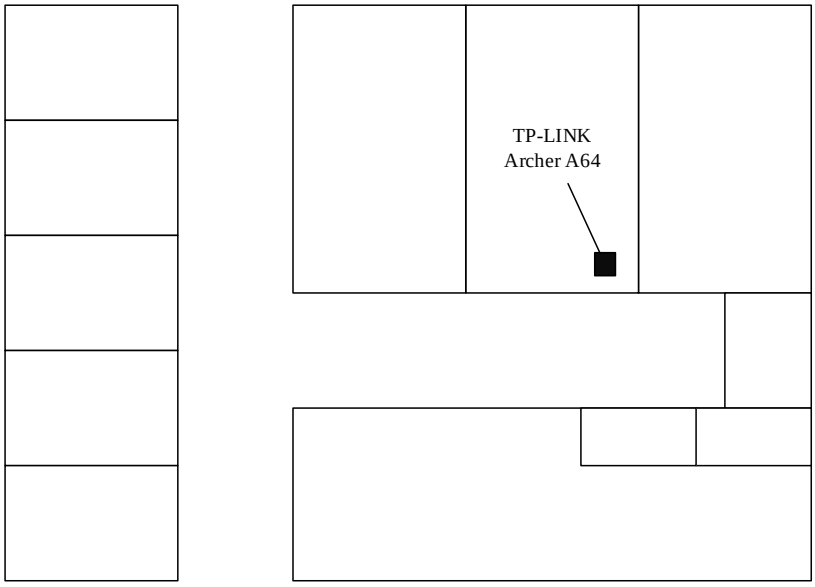

Рис. 4.2. Схематичний план кафедри кібербезпеки

Для вимірювань із розділу 2 виберемо додаток Wifi Analyzer.

Результати вимірювання у кімнаті (рис. 4.2.) безпосередньо біля самого джерела бездротового сигналу представлені на рис. 4.3. Як видно з рисунків діапазон 2,4 GHz досить зашумлений і відсутнє рознесення між різними каналами.

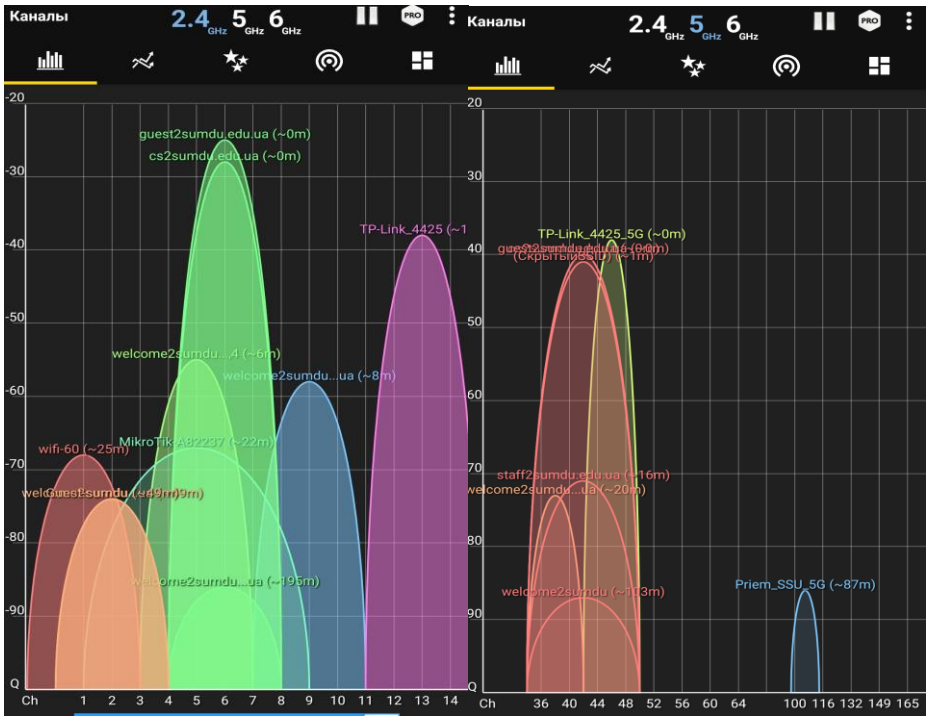

Рис. 4.3. Рівень сигналів 2,4 GHz та 5,0 GHz

Дослідження проводились для діапазону 5,0 GHz оскільки він менше зашумлений і з питань безпеки більш вигідний.

У результаті проведення серії практичних вимірювань була побудована карта зон статичної стабільності мережі. (рис. 4.4)

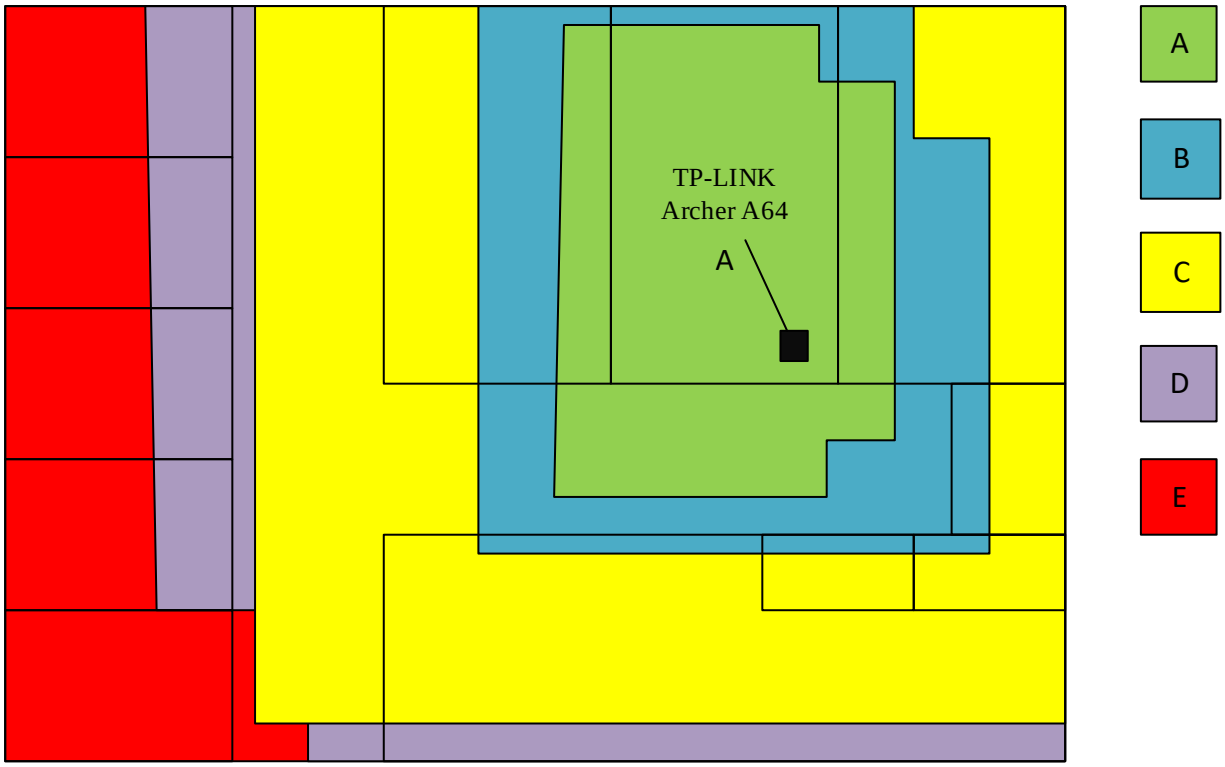

Рис. 4.4. Результат статичного аналізу Кіберполігону

Як видно з рис. 4.4 найкраще (зона «А») покриття у кімнаті з джерелом сигналу.

4.2. Динамічний аналіз

Для проведення динамічного аналізу, відповідно до методики у розділі 3 будемо вимірювати швидкість та пінг у різних точках полігону.

Для побудови зон покриття було введено наступні зони потрібно розділити на зони.

- зона «А» – відмінні показники: від 80% максимальних показників;
- зона «В» – хороші показники: від 60% максимальних показників;

- зона «C» – задовільні показники: від 40% максимальних показників;
- зона «D» – погані показники: від 20% максимальних показників;
- зона «E» – непридатні показники: від 0% максимальних показників.

Для автоматизації був написаний скріпт на PowerShell (Додаток А), який дозволяє запустивши його автоматизувати запуск програмного забезпечення та збір даних. Далі залишалось провести аналіз даних та побудувати зони.

Оскільки тестування динамічних параметрів потребує або 30-60 хвилин тестування однієї точки або передачу файлів розміром 1-5Гбайт застосовувався 1 ноутбук з ОС Windows 10. Результати побудованої динамічної картини зон стабільності представлені на рис. 4.5.

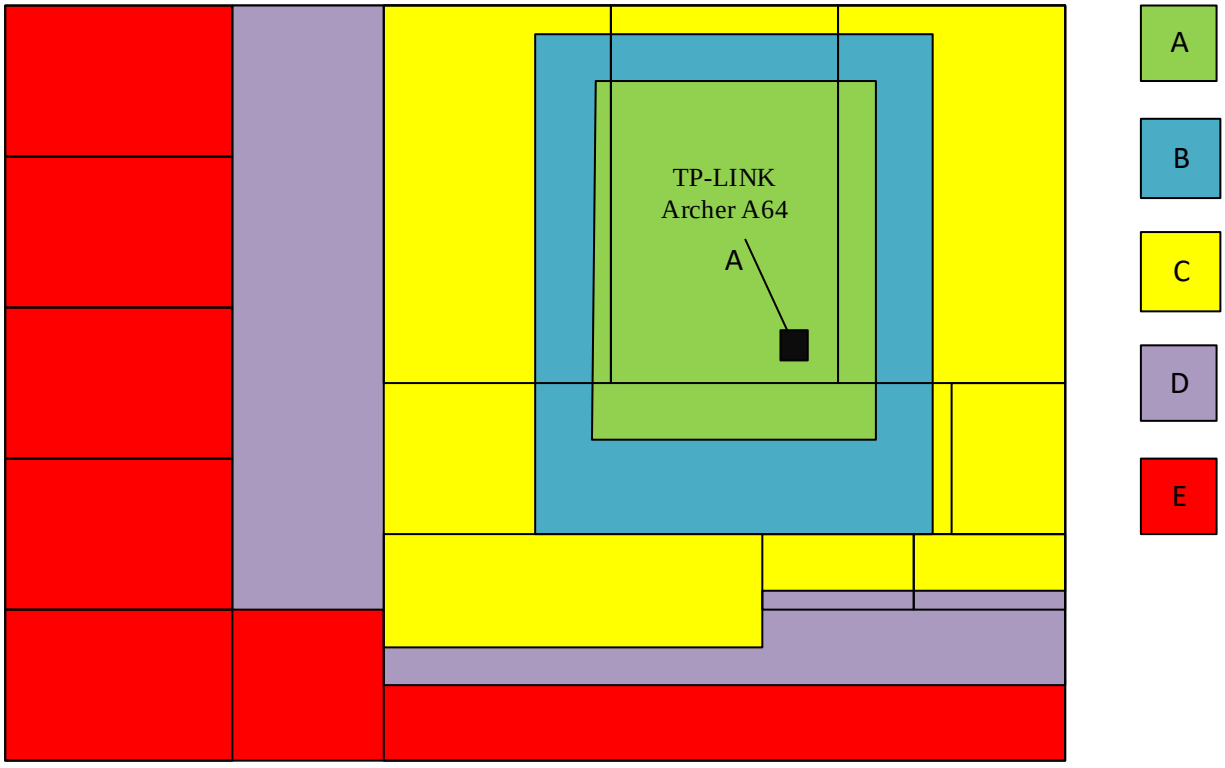

Рис. 4.5. Результат динамічного аналізу Кіберполігону

4.3. Визначення зон якісного покриття

Проводити аналіз з визначення зон якісного покриття, як показала практика, доцільно для зон «А» та «В».

Оскільки саме робота у цих зонах з часом буде відповідати усім вимогам.

Отриманий результат зон стабільності представлений на рис. 4.6.

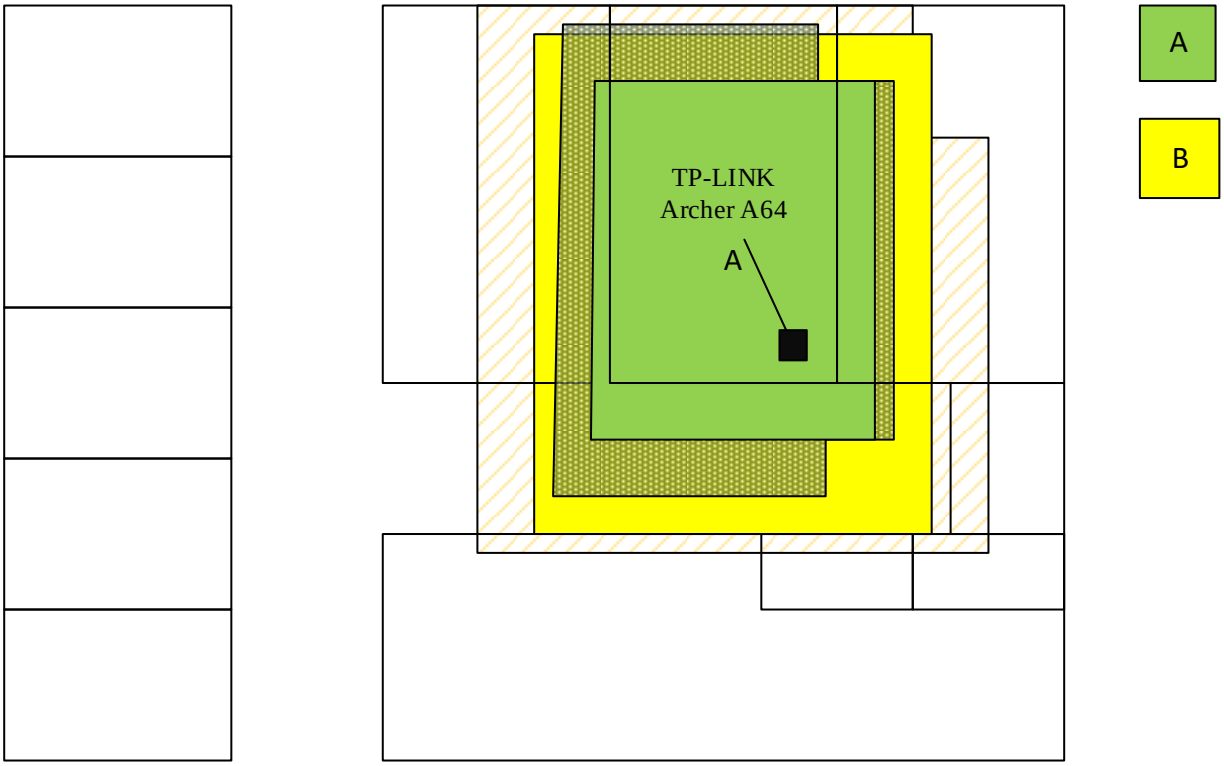

Рис. 4.6. Зони стабільного зв’язку «А» та «В»

З рис. 4.6. видно, що приміщення у якому встановлене джерело бездротового сигналу практично повністю покриває приміщення. (краще встановлювати посередині приміщення). А сигнал за межами приміщення досить сильно втрачає стабільність.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи магістра, було розглянуто питання розбудови навчального центра кібербезпеки з функціональністю симуляційного кіберполігону, а саме особливості побудови бездротових мереж.

Проведено огляд вітчизняних Центрів безпеки ЗВО України та виділені загальні характеристики у розбудові Кіберполігонів.

Розглянуті програмні засобі дослідження стабільності мереж. Виділено необхідність проведення статичного та динамічного аналізу бездротових мереж.

Створена методика дослідження бездротових мереж з визначенням критеріїв, які забезпечують стабільність роботи мережі.

Розроблені скрипти, які дозволяють автоматизувати процес дослідження.

Проведена практична перевірка стабільності роботи бездротової мережі Кіберполігону, побудована карта зон стабільності.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Стандарт вищої освіти за спеціальністю 125 «Кібербезпека» галузі знань 12 «Інформаційні технології» для першого (бакалаврського) рівня - URL:https://mon.gov.ua/storage/app/uploads/public/5bb/626/1a8/5bb6261a84776166409164.pdf (дата звернення: 04.11.2022).

2. Ciuperca E., Stanciu A., Cîrnu C. POSTMODERN EDUCATION AND TECHNOLOGICAL DEVELOPMENT. CYBER RANGE AS A TOOL FOR DEVELOPING CYBER SECURITY SKILLS //INTED2021 Proceedings. – IATED, 2021. – С. 8241-8246.

3. Korniyenko B., Galata L., Ladieva L. Security Estimation of the Simulation Polygon for the Protection of Critical Information Resources //ITS. – 2018. – С. 176-187.

4. Структурні підрозділи ФТІ - Фізико-Технічний Інститут. Фізико-Технічний Інститут. URL: http://ipt.kpi.ua/pidrozdili-ta-kafedri (дата звернення: 04.11.2022).

5. Кіберполігон – Кафедра кібербезпеки та інформаційних технологій. Кафедра кібербезпеки та інформаційних технологій – ХНЕУ ім. С. Кузнеця. URL: http://www.kafcbit.hneu.edu.ua/кіберполігон/ (дата звернення: 04.11.2022).

6. Кіберполігон - Кафедра кібербезпеки. Кафедра кібербезпеки - Хмельницький національний університет. URL: https://kb.khmnu.edu.ua/kiberpoligon/ (дата звернення: 04.11.2022).

7. У ВІТІ розгорнуто кіберполігон. АрміяInform – Інформаційне агентство АрміяInform. URL: https://armyinform.com.ua/2021/12/29/viti-otrymav-kiberpoligon/ (дата звернення: 04.11.2022).

8. До класифікатора внесено 17 додаткових професій у галузі безпеки інформації та кіберзахисту - Юридична Газета. Юридична газета – онлайн версія. URL: https://yur-gazeta.com/golovna/do-klasifikatora-vneseno-17-

dodatkovih-profesiy-u-galuzi-bezpeki-informaciyi-ta-kiberzahistu.html (дата звернення: 30.11.2022).

9. Олександр Потій: до класифікатора внесено 17 додаткових професій у галузі безпеки інформації та кіберзахисту - Державна служба спеціального зв'язку та захисту інформації України. URL: https://cip.gov.ua/ua/news/oleksandr-potii-do-klasifikatora-vneseno-17-dodatkovikh-profesii-u-galuzi-bezpeki-informaciyi-ta-kiberzakhistu (дата звернення: 30.11.2022).

10. Реформування системи підготовки професійних кадрів у сфері кібербезпеки в Україні: затверджені перші шість професійних стандартів - Державна служба спеціального зв'язку та захисту інформації України. URL: https://cip.gov.ua/ua/news/reformuvannya-sistemi-pidgotovki-profesiinikh-kadriv-u-sferi-kiberbezpeki-v-ukrayini-zatverdzheni-pershi-shist-profesiinikh-standartiv (дата звернення: 30.11.2022).

11. Миленко В. В Україні реформують підготовку фахівців у сфері кібербезпеки: шість нових стандартів. novyny.live. URL: https://society.novyny.live/v-ukraine-reformiruiut-podgotovku-spetsialistov-v-sfere-kiberbezopasnosti-shest-novykh-standartov-66202.html (дата звернення: 30.11.2022).

12. Wi-Fi Heatmap Software - Visualize Coverage and Capacity | Ekahau. Ekahau. URL: https://www.ekahau.com/solutions/wi-fi-heatmaps/ (date of access: 05.12.2022).

13. Scanner WiFi gratis | Scanner WiFi para windows | Acrylic Wi-Fi. Acrylic WiFi. URL: https://www.acrylicwifi.com/wifi-scanner/ (date of access: 05.12.2022).

14. Homedale. Wi-Fi 802.11ax (WiFi 6) - Все о новом стандарте передачи данных. URL: https://wifi-ax.com/192-homedale.html (date of access: 05.12.2022).

15. Technitium MAC Address Changer | A Freeware Utility To Spoof MAC Address Instantly. Technitium | Push The Limits. URL: https://technitium.com/tmac/ (date of access: 05.12.2022).

16. TamoSoft Throughput Test - Free WLAN Performance Meter. Wired and Wireless Network Analysis Software by TamoSoft. URL: https://www.tamos.com/products/throughput-test/ (date of access: 05.12.2022).

17. Speedtest от Ookla - Глобальный тест скорости широкополосного доступа. Speedtest.net. URL: https://www.speedtest.net/ru (дата звернення: 05.12.2022).

18. WirelessNetView - Wireless Network Monitoring Software. NirSoft. URL: https://www.nirsoft.net/utils/wireless_network_view.html (date of access: 05.12.2022).

19. Wi-Fi Scanner - Simple and convenient tool for monitoring 802.11a/b/g/n/ac/ax wireless networks. - LizardSystems. Terminal Services Manager, Remote Desktop Audit, Remote Process Explorer, Wi-Fi Scanner, Network Scanner, Find MAC Address, LanSend, Remote Shutdown, Change MAC Address - network monitoring tools by LizardSystems. URL: https://lizardsystems.com/wi-fi-scanner/ (date of access: 05.12.2022).

20. LinSSID. SourceForge. URL: https://sourceforge.net/projects/linssid/ (date of access: 05.12.2022).

21. iwScanner. SuggestUse. URL: https://suse.me/apps/iwscanner/ (date of access: 05.12.2022).

22. WiFi Scanner for Windows and Mac OS 2.4 GHz, 5 GHz, and 6 GHZ / 6E | AccessAgility. WiFi Scanner for Windows and Mac OS 2.4 GHz, 5 GHz, and 6 GHZ / 6E | AccessAgility. URL: https://wifiscanner.com (date of access: 05.12.2022).

23. Сучасні інформаційні технології в кібербезпеці: монографія / А. С. Довбиш, В. К. Ободяк, І. В. Шелехов та ін. ; за ред. В. К. Ободяка, І. В. Шелехова. – Суми : Сумський державний університет, 2021. – 348 с

24. iPerf - iPerf3 and iPerf2 user documentation. iPerf - The TCP, UDP and SCTP network bandwidth measurement tool. URL: https://iperf.fr/iperf-doc.php (date of access: 12.12.2022).

25. AC1200 Wi-Fi маршрутизатор з MU-MIMO. TP-Link Deutschland - Netzwerklösungen für Privat- und Businessanwender. URL: https://www.tp-

link.com/uk-ua/home-networking/wifi-router/archer-a64/ (дата звернення: 14.12.2022).

ДОДАТОК А

```
<#
Description:

#>
############################## INPUT PARAMS ##############################################
<############################## Параметри вхідного масиву для тестування ######
1 рядок тестування вихідного каналу 1800 секунд
2 рядок тестування вхідного каналу  1800 секунд
3 рядок тестування вихідного каналу 1 Гбайт
4 рядок тестування вхідного каналу  1 Гбайт
#################################################################################>

$ArrayIperfCommandStringAdress =(
"iperf3.exe    -c 192.168.0.113 -p 5201 -t 1800",
"iperf3.exe    -c 192.168.0.113 -p 5201 -t 1800   -r",
"iperf3.exe    -c 192.168.0.113 -p 5201 -n 1000M ",
"iperf3.exe    -c 192.168.0.113 -p 5201 -n 1000M -r"
)
<############################## Параметри тестування пінгу  ######>

$pspingCmd = "psping64.exe -4 -n 10s 192.168.0.113 -nobanner"
<############################## Активація параметрів  ######>
$runPspingActiale    = $true
$runSysParamsActiale  = $true
```

```
        ##################################################################
##############################

        <############################ Налаштування поточних
параметрів######>

        $iperfCommandString=@()
        $pathFiles = Split-Path -Parent $PSCommandPath
        $rootPathName  = $pathFiles+"\"
        <############################ Функція опису часу
тестування######>
        function diffTimeName
        {
           param(  [Parameter(Mandatory=$true)] [System.DateTime]$TimeStart,
                 [Parameter(Mandatory=$true)] [System.DateTime]$TimeEnd )

           $TimeDiffName = New-TimeSpan $TimeStart $TimeEnd
           if ($TimeDiffName.Seconds -lt 0)
           {
                  $HrsName = ($TimeDiffName.Hours) + 23
                  $MinsName = ($TimeDiffName.Minutes) + 59
                  $SecsName = ($TimeDiffName.Seconds) + 59
           }
           else
           {
                  $HrsName = $TimeDiffName.Hours
                  $MinsName = $TimeDiffName.Minutes
                  $SecsName = $TimeDiffName.Seconds
           }
```

```
        $DifferenceTime = '{0:00}:{1:00}:{2:00}' -f $HrsName,$MinsName,$SecsName
        write-host -ForegroundColor Yellow  "Start_test_time  : " $TimeStart
        write-host -ForegroundColor Yellow  "End_test_time  : " $TimeEnd
        write-host -ForegroundColor Red "Totals Time: " $DifferenceTime
    }



    function submitIperfName
    {
        param(  [System.String]$pathExecFile,
                [System.String]$logFileName,  [Parameter(Mandatory=$true)] [System.String]$iperfCommandString
                )

        $singleJob=Start-Job -Name "iperf3" -ScriptBlock {
            param( $pathExecFileName, $iperfCommandString, $logFileName)

        $iperftemp = @()
        $ iperftemp =$iperfCommandString.split(' ')
        $variableAttribute1="";  $variableAttribute2="";  $variableAttribute3="";  $variableAttribute4="";  $variableAttribute5="";  $variableAttribute6="";  $variableAttribute7="";  $variableAttribute8="";  $variableAttribute9="";  $variableAttribute10=""; $variableAttribute11=""; $variableAttribute12=""

        for($y=0; $y -lt $ iperftemp.Count; $y++ )
        {
          switch ($y)
          {
```

```
                0 {$variableAttribute1 =$iperftemp [$y]}
                4 {$variableAttribute5 =$iperftemp [$y]}
                2 {$variableAttribute3 =$iperftemp [$y]}
                3 {$variableAttribute4 =$iperftemp [$y]}
                1 {$variableAttribute2 =$iperftemp [$y]}
                5 {$variableAttribute6 =$iperftemp [$y]}
                9 {$variableAttribute10 =$iperftemp [$y]}
                7 {$variableAttribute8 =$iperftemp [$y]}
                8 {$variableAttribute9 =$iperftemp [$y]}
                6 {$variableAttribute7 =$iperftemp [$y]}
               10 {$variableAttribute11 =$iperftemp [$y]}
               11 {$variableAttribute12 =$iperftemp [$y]}
             }
           }
           $timeStart = (Get-Date -format yyy_MM_dd-HH:mm:ss).toString()

           cmd /c $pathExecFile$variableAttribute1 $variableAttribute2
$variableAttribute3 $variableAttribute4 $variableAttribute5 $variableAttribute6
$variableAttribute7 $variableAttribute8 $variableAttribute9 $variableAttribute10
$variableAttribute11 $variableAttribute12

            Start-Sleep -s 2
           # прочитати файл логування
           $f = Get-Content $logFileName
           # додати
           $btemp = "IPERF3: $variableAttribute1 $variableAttribute2
$variableAttribute3 $variableAttribute4 $variableAttribute5 $variableAttribute6
$variableAttribute7 $variableAttribute8 $variableAttribute9 $variableAttribute10
$variableAttribute11 $variableAttribute12 "
           Set-Content $logFileName -value $bName, $fName
```

```
            $aName=Get-NetIPAddress -AddressFamily IPv4 -InterfaceAlias
"Ethernet 2"
            $ipInterfaceName=$a.IPAddress
            $tempComputerName=$env:computername
            $hostName1 ="VMName: "+$tempComputerName+" -IP адреса: "+
$ipInterface + " Startime: " + $timeStart
            $fName = Get-Content $logFileName
            Set-Content $logFileName -value $hostName, $fNameStr

        } -ArgumentList ( $pathExecFile, $iperfCommandString, $logFileName)
        return $singleJob
    }

    function subJobPSPING
    {
       param(  Parameter(Mandatory=$true)] [System.String]$logFileName),

            [[Parameter(Mandatory=$true)] [System.String]$pathExecFile,

        $singleJob=Start-Job -Name "psping" - {
            param ( $pathExecFile, $pspingCmd, $logFileName)

        $temps = @()
        $temps =$pspingCmd.split(' ')
        $variableAttributePsping1="";          $variableAttributePsping2="";
$variableAttributePsping3="";                  $variableAttributePsping4="";
$variableAttributePsping5="";                  $variableAttributePsping6="";
$variableAttributePsping7="";                  $variableAttributePsping8="";
$variableAttributePsping9=""; $variableAttributePsping10=""
```

```
            for($ii=0; $ii -lt $temps.Count; $ii++ )
            {
              switch ($i)
              {
                0 {$variableAttributePsping1 = $temps [$ii]}
                1 {$variableAttributePsping2 =$temps [$ii]}
                2 {$variableAttributePsping3 =$temps [$ii]}
                7 {$variableAttributePsping8 =$temps [$ii]}
                9 {$variableAttributePsping10 =$temps [$ii]}
                5 {$variableAttributePsping6 =$temps [$ii]}
                6 {$variableAttributePsping7 =$temps [$ii]}
                3 {$variableAttributePsping4 =$temps [$ii]}
                8 {$variableAttributePsping9 =$temps [$ii]}
                4 {$variableAttributePsping5 =$temps [$ii]}
              }
            }

            cmd /c $pathExecFile$variableAttributePsping1
$variableAttributePsping2 $variableAttributePsping3 $variableAttributePsping4
$variableAttributePsping5 $variableAttributePsping6 $variableAttributePsping7
$variableAttributePsping8 $variableAttributePsping9 $variableAttributePsping10
>> $logFile
            cmd /c  echo "PSPING64: $pspingCmd" >> $logFileName


            } -ArgumentList ( $pathExecFile, $pspingCmd, $logFile)
            return $singleJob
        }
```

```
function CollectPerfHost
{
 param(
      [string]$hostNameSt,
      [string]$hostIPAddressSt,
      [string]$logFolderSt ,
      [int]$sampleIntervalInt,
      [int]$SysParamsDurationInt
     )

   write-host ">>> hostname          :" $hostNameSt
   write-host ">>> IP Addr           :" $hostIPAddressSt
   write-host ">>> Folder            :" $logFolderSt
   write-host ">>> SamplingTime      :" $sampleIntervalInt
   write-host ">>> Duration          :" $SysParamsDurationInt
   $singleJob=Start-Job -Name "counters" -ScriptBlock {
       param ($hostNameSt, $hostIPAddressSt, $logFolderSt,
$sampleIntervalInt, $SysParamsDurationInt)

       $delimiter = "`t"
       $params = @("\Процесор\% Processor Time",
         "\ Процесор \% User Time",
         "\ Процесор \переривань/с",
         "\ Процесор \% Privileged Time",
         "\ Процесор \% Interrupt Time",
         "\ Процесор \DPCs /c",
         "\ Процесор \% Idle Time",
         "\ Процесор \% DPC Time",
```

```
                "\ОЗП\ Помилки стек/с ",
                "\ ОЗП \Сторінки/с",
                "\ ОЗП \ Доступні байти ",
                "\ ОЗП \Available MBytes",
                "\мережеві параметри \ Всього байт/с ",
                "\ мережеві параметри \ Отримано байтів/с ",
                "\ мережеві параметри \байт відправка/с",
                "\ мережеві параметри\пакетів відправка/с"
                "\ мережеві параметри \пакетів отримання /с",
                "\ мережеві параметри \Packets Outbound Discarded",
                "\ мережеві параметри \Packets Outbound Errors",
                "\ мережеві параметри \Packets Received Discarded",
                "\ мережеві параметри \ Помилки отриманих пакетів ",
                "\ Диск параметри \ Поточна довжина черги диска ",
                "\ Диск параметри \% Disk Time",
                "\ Диск параметри \Avg. Disk Queue Length",
                "\ Диск параметри \Avg. Disk Write Queue Length",
                "\ Диск параметри \Avg. Disk Read Queue Length",
                "\ Диск параметри \обмін/с",
                "\ Диск параметри \запис/с"),
                "\ Диск параметри \ читання/с "

        # створення папки, якщо вона відсутня
           $b=test-path -path $logFolderSt -pathtype container
           if ($b -eq $false)
           {
             try
             {
                New-Item -ItemType Directory -Path $logFolderSt -ErrorAction
SilentlyContinue
```

```
            }
            catch
            {

                $FailedItem = $_.Exception.ItemName
                $ErrorMessage = $_.Exception.Message
        write-host "Проблема при створені папки $logFolderSt. Код помилки $ErrorMessage"
            }
          }

            ## масив параметрів для запису у файл
            $arrayParamName = @()
            foreach($p in $params)
            {
                $temp=$p.substring($p.LastIndexOf("\"))
                $temp=$temp.replace(".","")
                $a=$temp.replace("%","Perc")
                $temp=$a.replace("/","")
                $a=$temp.replace("\","")
                $temp=$a.replace("sec","Sec")
                $temp=$temp.replace(" ","")
                $arrayParamName += @($temp)
            }

$NumSamplesName=[math]::floor($SysParamsDurationInt/$sampleIntervalInt)
            $metrics =Get-Counter -ComputerName $hostIPAddressSt -Counter $params -SampleInterval $sampleIntervalInt -MaxSamples $NumSamplesName

            foreach($metric in $metrics)
```

```
            {
                $obj = $metric.CounterSamples | Select-Object -Property Timestamp, Path, CookedValue;
                #   додаємо дані
                $obj | Add-Member -MemberType NoteProperty -Name Computer -Value $hostIPAddressSt -Force;
                for ($k=0; $k -lt $obj.Count; $k++)
                {
                   $str=$obj[$k].Path
                   $temp = $str.LastIndexOf('\');
                   $rightPart = ($str.Substring($temp + 1)).Split(':')
                   $value12=$obj[$i].CookedValue
        $counterNameStr = $rightPart[0].Trim();


                   $timestamp=$obj[$i].Timestamp
                   $record=$timestamp.ToString("dd-MM-yyyy HH:mm:ss",[System.Globalization.CultureInfo]::InvariantCulture)+$delimiter+$counterNameStr+$delimiter+$value12

                   # $масив параметрів
                   $File=$logFolderSt+"\"+$hostNameSt+"-"+$arrayParamName[$i]+".txt"
        # додаємо отримані дані до файлу
                   out-file -Append -filepath $File  -inputobject $record -encoding ASCII
                }
             }
          } -ArgumentList ($hostNameSt, $hostIPAddressSt, $logFolderSt, $sampleIntervalInt, $SysParamsDuration)
```

```
        return $singleJob
    }
    # функція виконання роботи
    function statusJobsName
    {
      param(                                    [Parameter(Mandatory=$true)]
[System.Collections.ArrayList]$JobsName )

        $str="----- Статус роботи ----------------------------------------"
        write-host $str

        $numRunningJob=0
        ForEach ($jtemp in $JobsName)
        {
         try {
            if ($jtemp.State -eq "Completed")
            {
              $str="-----IDпроцесу:"+[string]$jtemp.Id+"¦"+"Процес_назва:"
+$jtemp.Name +"¦" +"Процес_статус: " +$jtemp.State
              write-host $str
            }
            if ($jtemp.State -eq "Running")
            {
              $numRunningJob++
              $str = "-- ID процесу:"+[string]$jtemp.Id +"¦" +" Процес (назва):"+
$jtemp.Name +"¦" +" Процес (статус): "+ $jtemp.State
              write-host $str
            }
            if ($jtemp[0].State -eq "Failed")
            {
```

```
            $str=$jtemp[0].ChildJobs[0].JobStateInfo.Reason
            write-host $str
        }
        }
        catch {
            $FailedItem = $_.Exception.ItemName
            $ErrorMessage = $_.Exception.Message
            write-host "Код помилки:" $ErrorMessage
            write-host "назва помилки  :" $FailedItem
            write-host "чекаємо 3 секунди"
            Start-Sleep -Seconds 3
            Continue
            }
    }
    $strTemp="------------"
    write-host $strTemp
        $numRunningJobFin = $numRunningJob
    return $numRunningJobFin
}
######Функція видалення процесу ######################
function RemoveJobsName
{
   param(  [System.Collections.ArrayList]$Jobs)

    $str="----- Видалення процесу------------------"
    write-host $str


    ForEach ($jtemp in $Jobs)
    {
```

```
        try {
            if ($jtemp.State -eq "Completed")
            {
              $str="-----ID процесу:"+[string]$jtemp.Id +"¦" +"назва процесу:" +$jtemp.Name +"¦" +"статус процесу: " +$jtemp.State
              write-host -ForegroundColor Cyan $str

              Remove-Job -Id $jtemp.Id
              $str="---ID процесу:"+[string]$jtemp.Id +"¦" +"назва процесу:" +$jtemp.Name +"-> видален"
              write-host -ForegroundColor Cyan $str
            }
            if ($jtemp.State -eq "Running")
            {

              $str = "—ID процесу:"+[string]$jtemp.Id +"¦" +"назва процесу:"+ $jtemp.Name +"¦" +"статус: "+ $jtemp.State
              write-host "статус процесу повинен бути виконано, але все ще виконується "
            }
            if ($jtemp [0].State -eq "Failed")
            {
               $str=$jtemp [0].ChildJobs[0].JobStateInfo.Reason
               write-host $str
            }
            }
            catch {
               $ErrorMessageName = $_.Exception.Message
               $FailedItem = $_.Exception.ItemName
               write-host "Помилка:" $ErrorMessageName
```

```
                write-host "Не вдалось:" $FailedItem
                write-host "чекаємо 3 секунди "
                Start-Sleep -Seconds 3
                Continue
                }
            }
        }
        ######Основна частина скріпта ########################

        foreach ($iperfCommandStringAdress in $arrayIperfCommandStringAdress)
        {
         [System.Collections.ArrayList]$JobsName = @()
         ##### Поточний час ############
         $TimeStartName = Get-Date -format HH_mm_ss
         $timeName=(Get-Date -format yyyy_MM_dd_HH-mm-ss).ToString()

         New-Item -ItemType Directory -Force -Path $rootPathName -Name
$timeName
         write-host ""
         $labelNameFile=$iperfCommandString.Replace("iperf3","").Replace("
","")

         if ( $runPspingActiale)
         {$temp11=$env:computername
           $logFileName = $rootPathName+$timeName + "\" + $temp11+ "_"+
"psping_"+ $labelNameFile + ".txt"
           $pathExecFile = $rootPathName
           write-host -foreground Green "_PSPINGCMD_: "$pspingCmd
           $singleJob=submitJobPSPING $pathExecFile $pspingCmd $logFileName
           $Jobs += @($singleJob)
```

```
        }

        if ($runSysParamsActiale)
        {
          ### Отримання системних параметрів.
          $hostNameSt = $env:computername
          $logFolderName=$rootPath + $time +"\"+ $env:computername+ "_" +
"syscounters"+"\"
      ### IPAddress серверу з iperf3 за дефолтом (192.168.0.113)
          $hostIPAddressSt="192.168.0.113"

      ### час інтервалу за дефолтом 1 секунда
          $sampleIntervalInt = 1
          write-host "HostName:" $hostNameSt "- IPAddress:" $hostIPAddressSt "-
logFolder" $logFolderSt
          $singleJob1__=CollectPerfHost $hostNameSt $hostIPAddressSt
$logFolderName $sampleIntervalInt $SysParamsDurationInt
                $Jobs += @($singleJob1__)
        }
      $logFileName = $rootPath + $time
      $temp11Host=$env:computername
      $logFileName=$logFileName + "\" +$temp11Host + "_"+ "iperf"+ $labelFile
+ ".txt"
      $pathExecFile = $rootPath
      $execFile = $rootPath + "iperf3.exe"
      $cmd=$iperfCommandString -replace('\s', " ")
        $iperfCommandString = $cmd+" --logFileName"+ $logFile
        write-host -foreground Yellow "IperfCommandString:
"$iperfCommandString
```

```
        $singleJob=submitIperfName $pathExecFile $iperfCommandString
$logFileName
        $Jobs += @($singleJob)

        Do {
            $numRunningJobTemp= statusJobs $Jobs
            write-host "номер процесу: "$numRunningJobTemp
            start-sleep -Seconds 5
          } while ($numRunningJobTemp -ge 1)


        RemoveJobs $JobsName

        $TimeEnd = Get-Date -format HH:mm:ss
        diffTimeName $TimeStart $TimeEnd
        write-host ""
      }
```