Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ДОПУСТИТИ ДО ЗАХИСТУ:
завідуюча кафедри кібербезпеки
та захисту інформації
________Наталія ЛУКОВА-ЧУЙКО
«14» червня 2022р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

дипломної роботи

бакалавра
(назва освітнього ступеня)

галузь знань 12 Інформаційні технології
(шифр і назва галузі знань)

спеціальність 125 Кібербезпека
(код і назва спеціальності)

освітня програма Кібербезпека
(назва освітньої програми)

на тему: «Особливості тестування на проникнення пристроїв Інтернету речей»

Виконавець: студент IV курсу, групи КБ-42

________________ **Микита ШАРАПОВ**
(підпис) (ім'я прізвище)

	Прізвище, ініціали	Підпис
Керівник	Юрій ЩЕБЛАНІН.	

Нормоконтроль	Сергій ДАКОВ.	

Київ 2022

Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ЗАТВЕРДЖЕНО:
завідуюча кафедри кібербезпеки
та захисту інформації
________Наталія ЛУКОВА-ЧУЙКО
«01» листопада 2021 р.

ЗАВДАННЯ
на виконання дипломної роботи

спеціальності 125 Кібербезпека
(код і назва спеціальності)

освітньої програми Кібербезпека
(назва освітньої програми)

Студентові **КБ-42** **Шарапову Микиті Дмитровичу**
(група) (прізвище ім'я по-батькові)

Тема дипломної роботи Особливості тестування на проникнення пристроїв Інтернету речей

1. ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

Тема дипломної роботи затверджена на засіданні кафедри кібербезпеки та захисту інформації протокол №5 від 29.10.2021 р.

2. ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ
Структури, архітектури, дослідження фахівців, алгоритми тестування на проникнення

3. ЗМІСТ РОЗРАХУНКОВО-ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ
Стандарти інформаційної безпеки, методології тестування на проникнення, технології побудови IoT, протоколи підкючення, операційні системи IoT

4. ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Практична цінність Аналіз існуючих методів тестування на проникнення та усунення недоліків тестування.

5. ДАТА ВИДАЧІ ЗАВДАННЯ

Дата видачі завдання: 01 листопада 2021 року

Завдання видав	(підпис)	Юрій ЩЕБЛАНІН (ім'я, прізвище)
Завдання прийняв до виконання	(підпис)	Микита ШАРАПОВ (ім'я, прізвище)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів робіт	Строки виконання робіт (початок-кінець)	Відмітка про виконання
1	Уточнення постановки задачі	29.10.2021 – 27.01.2022	*виконано*
2	Аналіз літератури	28.01.2022 – 11.02.2022	*виконано*
3	Огляд моделювання загроз	12.02.2022 – 24.02.2022	*виконано*
4	Опис методів моделювання загроз	25.02.2022 – 24.03.2022	*виконано*
5	Збір відомостей щодо Інтернету-речей	25.03.2022 – 07.04.2022	*виконано*
6	Дослідження властивостей розумного дому	08.04.2022 – 05.05.2022	*виконано*
7	Аналіз недоліків тестування	06.05.2022 – 20.05.2022	*виконано*
8	Оформлення пояснювальної записки	21.05.2022 – 04.06.2022	*виконано*
9	Підготовка до захисту дипломної роботи	05.06.2022 – 10.06.2022	*виконано*

Завдання видав	(підпис)	Юрій ЩЕБЛАНІН (ініціали, прізвище)
Завдання прийняв до виконання	(підпис)	Микита ШАРАПОВ (ініціали, прізвище)

Термін подання дипломної роботи до ЕК 06 червня 2022 року

РЕФЕРАТ

Дипломна робота складається зі вступу, трьох розділів, загальних висновків, списку використаних джерел, має 48 сторінки основного тексту, 6 ілюстрації, 3 таблиці. Список використаних джерел містить 61 найменування і займає 6 сторінок.

Мета роботи - удосконалення методу тестування на проникнення пристроїв IoT з урахуванням особливостей даних систем.

Об'єкт дослідження - тестування на проникнення пристроїв Інтернету речей.

Предмет дослідження - удосконалення методу тестування на проникнення пристроїв IoT, яка б давала тестувальнику розуміння основних векторів атак.

Методи дослідження: системний підхід, методи порівняння, індексний метод, структурний аналіз

У роботі проведено аналіз різних наукових досліджень щодо поточного стану ринку IoT та його майбутнього розвитку та тенденцій його змін. Також основних існуючих методів та методологій проведення тестування на проникнення IoT, різних рекомендацій від великих компаній, які займаються IoT

Запропоновано покращення існуючих методів та методологій щодо проведення тестування пристроїв IoT, який покриває всю поверхню атаки та не має відкритих критичних помилок.

Практична частина дипломної роботи полягає в удосконаленняі методів тестування на проникнення пристроїв IoT.

Результати зроблених у дипломній роботі досліджень можуть бути використані у створенні нового покращеного методу і методології з тестування на проникнення пристроїв IoT який покриє всю поверхню атаки і зробить більш надійними такі пристрої і скоротить час для тестування на проникнення.

Ключові слова: пристрої іот, тестування на проникнення, поверхня атаки іот, методології тестування

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

5G	-	5th Generation
6LoWPAN	-	IPv6 over Low-Power Wireless Personal Area Networks
ARM	-	Advanced RISC Machine
API	-	Application Programming Interface
BLE	-	Bluetooth Low Energy (Bluetooth LE)
CVSS	-	Common Vulnerability Scoring System
DDoS	-	Distributed Denial of Service
DNS	-	Domain Name System
GSMA	-	The GSM Association
IDOR	-	Insecure direct object reference
IoT	-	Internet of Things
ISSAF	-	Information System Security Assessment Framework
ISVS	-	IoT Security Verification Standard
JTAG	-	Joint Test Action Group
LoRaWAN	-	Long Range Wide Area Network
LTE	-	Long-Term Evolution 4G LTE
MQTT	-	Message Queue Telemetery Transport
NIST	-	National Institute of Standards and Technology
OSINT	-	Open Source Intelligence
OTA	-	Over-The-Air
OWASP	-	Open Web Application Security Project
PTES	-	Penetration Testing Execution Standard
SMB	-	Server Message Block
SSL	-	Secure Sockets Layer
TLS	-	Transport Layer Security
UART	-	Universal Asynchronous Receiver/Transmitter
Wi-Fi	-	Wireless Fidelity
XSS	-	Cross-Site Scripting

ЗМІСТ

ВСТУП

У сьогоднішньому світі Інтернет-речей не є чимось загадковим, адже пристрої, що відносяться до даної технології оточують більшість населення планети [1]. Такими є як і складні системи на виробництвах та заводах, впроваджені задля підвищення ефективності процесів, так і звичайні фітнес-браслети та розумні телевізори, метою яких є забезпечення повсякденних потреб пересічної людини. Перевагами використання об’єктів IoT є економія часу, спрощення контролю за певними системами, впровадження автоматизації на підприємствах та у агросекторі, розумне управління та багато інших [2].

Безпека IoT все частіше жертвується в користь темпам виробництва нової електроніки, а також не отримує потрібноії уваги від розробників програмного забезпечення та виробників пристроїв і систем через недостатнє фінансування для забезпечення необхідних перевірок та тестування на проникнення у процесі та на фінальних стадіях розробки пристроїв [3]. Проблеми безпеки в пристроях IoT можуть призвести до проблем з конфіденційністю, приватністю та доступності даних, а також до їх повної компрометації. Дивлячись на пристрої, вразливості також можуть спричинити великі фінансові збитки шляхом створення потужних ботнетів або навіть пряму загрозу життю у випадку з біомедичним використанням IoT або автопілот в сучасних машинах. Недосконалі процеси розробки, тестування та імплементації кращих практик безпеки породжують недоліки і проблеми безпеки у складових архітектури IoT та активізують актуальність теми тестування на проникнення пристроїв IoT. Але існуючі методи тестування на проникнення мають низку недоліків у випадку їх застосування до тестування пристроїв IoT: не враховується специфіка поверхні атаки пристроїв IoT, методики у сфері IoT висвітлюють процес тестування лише у контексті відповідності стандартам безпеки, що впливає на гнучкість тестування,не описується необхідний інструментарій та інші особливості тестування IoT. Через це виникає необхідність у вдосконаленні стандартів тестування шляхом розробки методу тестування безпеки пристроїв IoT.

РОЗДІЛ 1
ОСНОВНІ ВІДОМОСТІ ПРО БЕЗПЕКУ В ІНТЕРНЕТІ РЕЧЕЙ ТА ЙОГО ПРИСТРОЯХ

1.1 Аналіз актуальності і стану захищеності IoT

Інтернет речей (IoT) — це мережева концепція, що складається з взаємопов’язаних фізичних пристроїв із вбудованими передавачами та програмним забезпеченням, яке дозволяє передавати дані та обмінюватися в автоматизованому режимі між фізичним світом і комп’ютерними системами за допомогою стандартних протоколів зв’язку. [1].

Платформа Інтернету речей — посередник між фізичними пристроями й центром, інструмент для віддаленого доступу, контролю пристроїв системи та керування ними.

Зазвичай під «річчю» розуміють фізичні пристрої, предмети, механізми, навіть будівлі. Термін був введений Кевіном Ештоном у 1999 році [2] і розвивався за допомогою таких технологій, як Wi-Fi або стандарт 5G [3] для ефективного управління даними та підтримки постійного обміну інформацією. Пристрої IoT є результатом поєднання інформаційних технологій (IT) та операційних технологій (OT).

Багато пристроїв IoT є результатом консолідації, мобільних обчислень, вбудованих систем, великих даних. Пристрої IoT можуть забезпечувати обчислювальну функціональність, надавати можливість зберігання даних та підключатись до мережі для додання функціоналу, якого раніше не було у мережі, підвищувати ефективність, наприклад надавати можливість віддаленого доступ для моніторингу, конфігурації та усунення несправностей.

Дивлячись на розвиток ринку IoT, дослідження IDC припустили, що кількість пристроїв, підключених до Інтернету, включаючи машини, датчики та камери, що складають IoT, складе 75,44 мільярда пристроїв, що генерує 79,4 цетабайта (ZB 10^{21})

даних у 2025 році [4]. Всесвітній економічний форум погоджується з цими оцінками, що буде 75 мільярдів пристроїв, які записують дані про те, як ми живемо, працюємо, рухаємося та працюємо в містах. Однак не кожен дослідник вважає, що показники будуть такими високими, і є ознаки того, що багато компаній починають применшувати передбачувану кількість пристроїв, які будуть в роботі. Наприклад, Statista [5] підраховує, що загальна база підключених до IoT пристроїв у всьому світі складе 16.4 млрд. одиниць до 2025 р., що різко стрибне з 11.57 млрд. одиниць, оцінених на цей рік, що ілюструє наступна діаграма (рисунок. 1.1) [5]. За словами професора Вільяма Вебба, автора книги “*Міф про Інтернет речей*”, буде приблизно 8,5 мільярдів пристроїв [6].

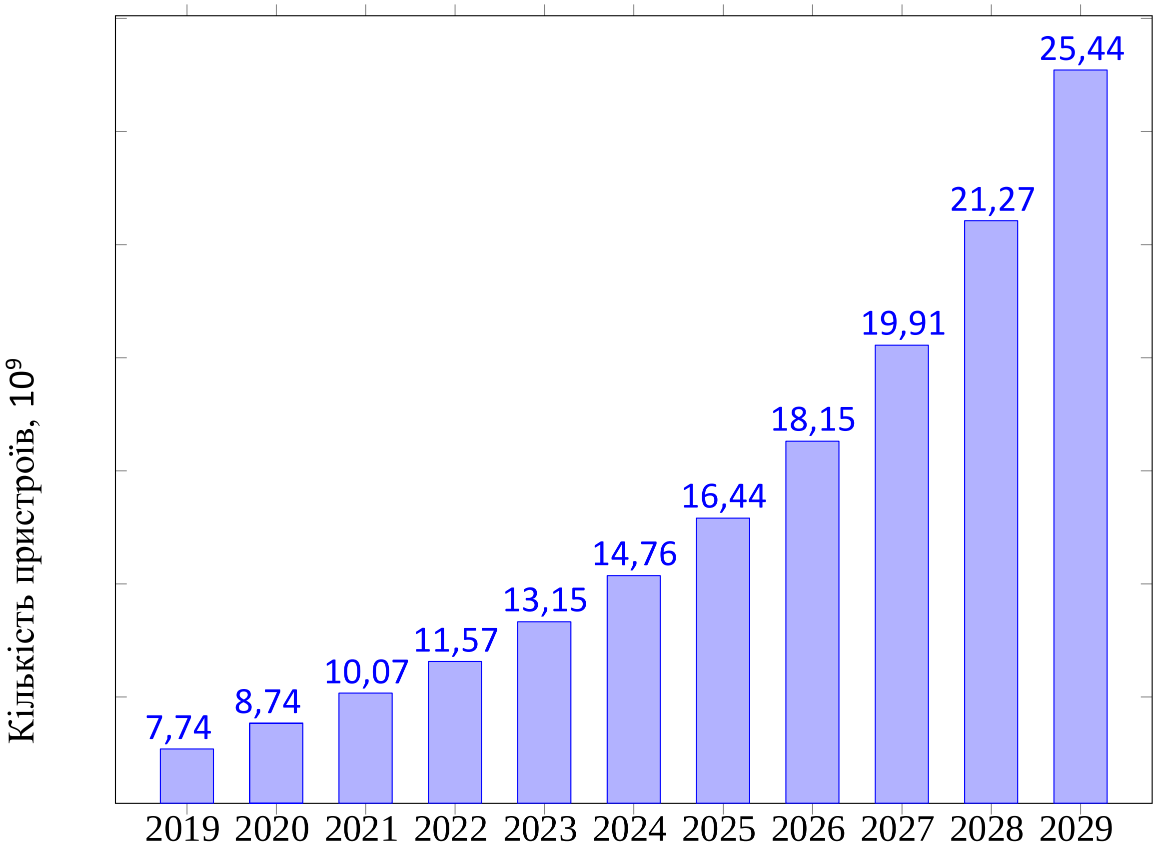

Рисунок 1.1. - Прогноз спеціалістів Statista про кількість пристроїв IoT.

Хоча повна сфера використання Інтернету речей чітко не визначена, вона явно дуже велика. Велика кількість сфер життя мають власні типи пристроїв IoT, наприклад спеціалізоване лікарняне обладнання та інтелектуальні дорожні технологій, і існує велика кількість корпоративних IoT пристроїв. Як приклади приладів Інтернету речей можна навести побутові пристрої: кухонна побутова техніка (мікрохвильові печі, телевізори, кавоварки, холодильники, пральні машини, будильники,),елементи «розумного дому» (термостати, камери домашньої безпеки, замки дверей, лампочки), різноманітні сенсори (датчики освітленості, температури, руху) [7].

Спеціалісти з Foresterr виділяють такі тренди на 2022 рік [8]:

- Збільшення варіантів підключень до мереж.

Перед лідерами ринку представлені дуже широкий спектр технологій, які можно використовувати. Спеціалісти з Forrester очікують, що впровадження технологій, особливо 5G та Wi-Fi, буде знижуватись у 2022 році, оскільки організаціям потрібно зорієнтуватися у наявних технологіях та протоколах.

Низькоорбітні супутники, які також можливо віднести до пристроїв IoT, стають більш актуальними – сьогодні більше 2386 супутників Starlink забезпечують якісній зв'язок. Очікується, що інтерес до супутникових та інших мережевих технологій низького енергоспоживання зросте на 20 відсотків у наступномуроці.

- Виробники приділятимуть більше уваги медичному використанню приладів.

Пандемія COVID-19 ввела значні карантинні обмеження, не дозволяючи людям залишати свої домівки та відвідувати медичні установи у звичайному режимі. Як наслідок, на хронічні та критичні захворювання не приділяли належної уваги, що призвело до поширення носіїв, датчиків моніторингу стану здоров'я та цифрових медичних приладів та відповідного зростання сектору ринку.

- Умови пандемії підштовхують організації до змін у взаємодіях співробітник-роботодавець.

Спричинена пандемією COVID-19 світова економічна криза змушує підприємців полишати дорогу корпоративну нерухомість. Очікується, що в подальшому принаймні 80 відсотків підприємств розроблять комплексні локальні стратегії повернення до роботи, які залучають IoT для підвищення безпеки співробітників та ефективності використання ресурсів.

- Дані про місцезнаходження стануть ключевими для організації сервісів.

Та ж пандемія COVID-19 наголосила на важливості збору даних про геолокацію, для покращення умов для клієнтів і співробітників. За словами спеціалістів з Forrester, до 2022 року бренди повинні використовувати ці дані для формування віртуальних черг або онлайн бронювань. Бренди будуть покладатися на технологічних партнерів, щоб допомогти використовувати дані про місцезнаходження, а також на сторонні джерела даних, яким споживачі довіряють і яким вони контролюють.

Зростання популярності Інтернету речей призвело до появи нових пристроїв, що несе в собі безліч загроз, вразливостей і пов’язаних з ними ризиків.

1.2 Проблеми безпеки в IoT

Проблеми безпеки в IoT стосуються великої кількості пристроїв. Такі проблеми виникають через особливості самої концепції Інтернету речей, несистематичне впровадження найкращих практик і стандартів безпеки на всіх етапах розробки та життєвого циклу пристрою. Виробники та розробники програмного забезпечення пристроїв часто жертвують ними через відсутність належних статей витрат у бюджеті або через брак часу, необхідного для проведення належних тестів безпеки, адже з точки зору бізнесу важливіше швидше випустити пристрій на ринок, щоб зайняти нішу або обійти конкурентів у сфері [9].

З опитування Statista (таблиця.1.2) [10] зрозуміло, що одні з серйозних проблем безпеки IoT сьогодні - це відсутність достатньої кількості кваліфікованого персоналу, що займався би процесами імплементації безпечних практик у процесах розробки пристроїв та проблема захисту данних у IoT. Однак що стосується захисту данних, то воно змінюється на краще, а саме: з’являється все більше “гайдів” від провідних організацій, таких як (NIST [10], ENISA [11], IoTSF [12]), але їх існування не обов’язково гарантує їх впровадження – через відносно молодий вік галузі, в національному законодавстві немає належного регуляторного механізму, а це означає, що широке впровадження стандартів неминуче матиме проблеми з безпекою в міру розробки пристроїв.

Проблеми пристроїв IoT, що можуть впливати на критичні операції:

1) Недостатня кількість персоналу для впровадження безпеки IoT
2) Захист чутливих даних, згенерованих пристроєм IoT
3) Втрата чи викрадення пристроїв IoT
4) Недостатня кількість фреймворків безпеки в середовищі IoT
5) Порушення конфіденційності, пов'язане з даними, згенерованими пристроєм IoT
6) Недостатність ефективного контролю доступу/автентифікації пристроїв
7) Привілейований доступ до пристроїв IoT

Таблиця 1.2

Проблеми безпеки IoT, Statista

Кількість персоналу	25%
Захист даних	25%
Викрадення пристроїв IoT	17%
Недостатня кількість фреймворків	10%
Порушення конфіденційності	7%
Недостатність контролю	5%
Привілейований доступ	11%

Експерти з кібербезпеки та розробники IoT ледве були готові до атаки ботів Mirai у 2016 році, яка навіть привернула увагу федерального уряду США. Ця ескалаційна серія атак об'єднала сотні тисяч недорогих пристроїв для власних цілей атакуючих, отримуючи доступ за допомогою відомих паролів за замовчуванням, таких як «administrator», «password» та «1234» [1311]. Це призвело до DDoS-атаки проти Dyn, постачальника DNS, який є частиною інтернет-інфраструктури багатьох американських гігантів, таких як Amazon, Netflix, Twitter, Starbucks та багато інших. Невдовзі після Mirai, атаки WannaCry і NotPetya нанесли глобального втрату в трильйони доларів, відтак, що вплинули на критичну інфраструктуру та системи IoT, які використовуються у виробництві. WannaCry та NotPetya, були вірусами вимагачами, які були основані на базі експлоїту EternalBlue, який використовує вразливість у реалізації Microsoft протоколу Server Message Block (SMB) [14]. Коли стало відомо, що Mirai було розроблено кількома студентами коледжу, уряди усього світу визнали для себе, що вони повинні вивчити ступінь проблеми безпеки IoT.

З подібних випадків (Mirai, Mozi, Zeroshell [15]) видно, що не завжди у користувачів є розуміння того, яким чином працюють IoT, а також не усвідомлюють того, що пристрої IoT можуть бути вразливими. Це є важливим фактором для того, щоб розуміти, які проблеми безпеки можуть винукти у системах. Розуміння, як IoT впливають на управління ризиками кібербезпеки, важливо, особливо разглядаючи реагування на ризики – прийняття, пом'якшення, уникнення.

Одна з основних тенденцій у безпеці IoT - це, те що питання безпеки у цій сфері не підіймається системно, а лише після дуже великих кібератак. Впровадження стандартів безпеки, а також зменшення ризиків, що виникають у безпеці пристроїв IoT можливо лише через удосконалення існуючих методів тестування на проникнення. Саме так можливо виправити вищезазначені проблеми через те, що тестування на проникнення було створено для ідентифікації існуючих, а не для виявлення нових проблеми з безпекою, що зараз існують у даній сфері та зрозуміти, як саме вразливості будуть використовуватись зловмисниками а також допомогти у пошуку рішень проблем після їх виявлення. Оскільки тестування на проникнення спирається на розуміння архітектури тестованих систем і пристроїв, необхідно провести огляд типової архітектури пристроїв IoT та визначити так звану поверхню атаки для того щоб виявити всі пов'язані загрози, що буде розглянуто далі.

1.3 Архітектура та операційні системи пристроїв Інтернету речей

Екосистеми IoT поєднують пристрої різних типів. Неоднорідність таких елементів (особливо компонентів самих IoT пристроїв) і молодий вік галузі, призвели до суперечливих визначень архітектури екосистеми дослідниками та відсутності уніфікованої архітектури координації пристроїв IoT.

Через це, дослідники називають три основні рівні, а саме: сприйняття, мережевий, прикладний; що притаманні більшості пристроїв IoT [16], але також виділяють ще два рівні в мережевому (транспортний та обробки) та рівень бізнеспроцесів [17] (таблиця 1.3):

1. Рівень сприйняття (perception) або фізичний: Власне фізичні пристрої, наприклад, датчики, виконавчі механізми, розумніречі тощо.

2. Мережевий рівень (network): Інфраструктура зв’язку для пристроїв, серверів та користувачів.

2.1. Транспортний: відповідає за транспортування даних із рівня сприйняття до наступного рівня.

2.2. Рівень обробки: отримує дані з попереднього рівня та обробляє їх за допомогою такої служби, як база даних.

3. Прикладний рівень: програмне забезпечення, яке використовує та керує даними з пристроїв, надаючи послуги кінцевим користувачам.

4. Рівень бізнес-процесів: на ньому вирішуються проблеми бізнес-моделі та керуються процедурами на рівні програми.

Таблиця 1.3

Рівні архітектури пристроїв IoT

Прикладний рівень	
Мережевий рівень	Транспортний рівень
	Рівень обробки
Рівень сприйняття	Рівень бізнес-процесів

Ключові властивості IoT, що є перевагами, але породжують основні проблеми за кібербезпеки, виділяють такі:

1. Гетерогенність (одна з ключових властивостей IoT) – що описує можливість існування і взаємодії широкого спектру платформ, протоколів, типів пристроїв у межах однієї екосистеми, тісно пов’язана з так званою «interoperability», що зумовлює взаємодію різних пристроїв.

2. Підключення - великий спектр можливих варіантів протоколів з’єднання, таких як Wi-Fi, Bluetooth, 4G, 6LoWPAN, 5G, LoRaWAN, та іншими.

Зазвичай, IoT будуються на таких апаратних платформах як ARM, Arduino, Raspberry Pi. Їх об’єднує невелика ціна, невисокий поріг входу у створенні програмне забезпечення для них а також простота побудови систем на цих платформах [18].

Операційні системи, на яких працюють IoT-пристрої, також дуже різні, в залежності від потреб самих пристроїв, їх існує безліч. З прикладів можна найпопулярніші, а саме Ubuntu Core, Android, Riot os, Contiki os.[19]. За даними Statista (таблиця. 1.4), більшість з користувачів віддають свої гроші саме за пристрої з операційною системою Linux або Android [20].

Таблиця 1.4

Операційні системи IoT

Ubuntu Core	34%
Без ОС	18%
FreeRTOS	9%
Windows Embedded	9%
mbed	6%
Contiki	5%
TinyOS	5%
RIOT	3%
Інші	11%

1.4 Поверхня атаки і загрози

Поверхня атаки - це об'єднання способів, які атакуючий використовує щоб проникнути в систему та викрасти дані [21]. Тестування на проникнення пристроїв IoT є досить складною задачею через загальні властивості самої концепції IoT та через велику поверхню атаки, найголовнішою проблемою є те що виробники використоють різні компоненти пристрою та екосистеми. У випадку з пристроями IoT поверхня атаки виглядає наступним чином [22]:

1. Фізичні інтерфейси пристроїв: це незахищені елементи IoT, які можуть бути

використані для підробки пристроїв IoT або скидання базових налаштувань, або отримання прав доступу.

2. Вбудоване програмне забезпечення пристрою: мікро програмне забезпечення може надавати конфіденційні дані, такі як облікові дані, ключі шифрування, ключі автентифікації, версія мікропрограми.

3. Локальне зберігання даних: сховище данних також може бути ціллю атаки, якщо вони не зашифровані, або якщо вони зашифровані, то за допомогою скомпрометованого ключа доступу або якщо немає перевірок цілісності.

4. Механізм оновлення: відсутність шифрування або електронного цифрового підпису, місця для запису програмного забезпечення або відсутні механізми оновлення будуть використовуватися зловмисником для отримання доступу в систему або впровадження шкідливих операцій.

5. Мобільний додаток: зловмисник може використовувати вразливості, пов'язані з мобільним додатком, підключеним до пристрою IoT, як вектори атак (MiBand для Фітнес годиників).

6. Веб-інтерфейс пристрою: він складається з усіх вразливостей мобільного додатку, веб інтерфейса самого пристрою IoT та управління обліковими даними.

Висновки до розділу 1

У розділі розглянуто:

Інтернет речей - це мережева концепція, що складається з взаємопов'язаних фізичних пристроїв із вбудованими передавачами та програмним забезпеченням, яке дозволяє передавати дані та обмінюватися в автоматизованому режимі між фізичним світом і комп'ютерними системами за допомогою стандартних протоколів зв'язку. Спеціалісти з Foresterr виділяють наступні тренди 2022 року в сфері використання IoT:

1) Збільшення варіантів підключень до мереж

2) Виробники електроніки приділятимуть значно більшу увагу медичному використанню пристроїв

3) Умови пандемії підштовхують організації до змін у взаємодіях співробітник-роботодавець

4) Користувацькі дані про місцезнаходження стануть ключем до організації зручних сервісів

З опитування Statista зрозуміло, що однією з проблем безпеки IoT сьогодні, є відсутність достатньої кількості кваліфікованого персоналу, що займався би процесами імплементації безпечних практик у процеси розробки пристроїв IoT, а також проблема стандартизації в сфері безпеки IoT.

До проблем пристроїв IoT, що можуть впливати на критичні операції відносяться:

1) Недостатня кількість персоналу для впровадження безпеки IoT

1) Захист чутливих даних, згенерованих пристроєм IoT

2) Втрата чи викрадення пристроїв IoT

3) Недостатня кількість фреймворків безпеки в середовищі IoT

4) Порушення конфіденційності, пов’язане з даними, згенерованими пристроєм IoT

5) Недостатність ефективного контролю доступу/автентифікації пристроїв

6) Привілейований доступ до пристроїв IoT

Дослідники виокремлюють основні три рівні (сприйняття, мережевий, прикладний), що притаманні більшості пристроїв IoT, але також іноді дослідники виділяють ще два рівні в мережевому, а саме: транспортний та обробки; а також рівень бізнеспроцесів.

РОЗДІЛ 2
ОГЛЯД ІСНУЮЧИХ МЕТОДОЛОГІЙ ТЕСТУВАННЯ НА ПРОНИКНЕННЯ

2.1 Методології тестування на проникнення

Для тестування на проникнення IoT, використовють провідні методологі від лідерів в сегменті безпеки, які вважаються стандартами для тестувальників. Цими методологіямиє:

1)OSSTMM – The Open Source Security Testing Methodology Manual [23].

2)NIST Special Publications 800-115 Technical Guide to Information Security Testing and Assessment [24].

3)PTES — Penetration Testing Execution Standard [25].

4)ISSAF — Information System Security Assessment Framework [26].

2.1.1 Аналіз метод OSSTMM

Методологія - це структурований і формалізований документ, що детально описує етапи тестування на проникнення. Основний напрямок метода спрямований на тестування мереж. В розділах цього метода міститься інформація про:

- Суть тестування, рамки, процеси тестування;
- Аналіз соціальних процесів;
- Тестування безпеки фізичної інфраструктури, безпровідних технологій, даних;
- Рекомендації щодо відповідності тесту державним стандартам;
- Підготовку звіту.

В документі немає інформації про класифікацію вразливостей та додаткового опису до вимог, однак докладно описані основні процедури підготовки до тестування і методи самого процесу тестування. Також методологія має так звану “карту безпеки”, в якій відображаються основні складові, що повинні бути оцінені в процесі тестування (інформаційна безпека, безпека фізичної інфраструктури, безпека безпровідних технологій, безпека Інтернет-технологій та інших). Зазначена інформація, яку атакуючий може отримати в результаті атаки на ті чи інші оцінювані складові [27].

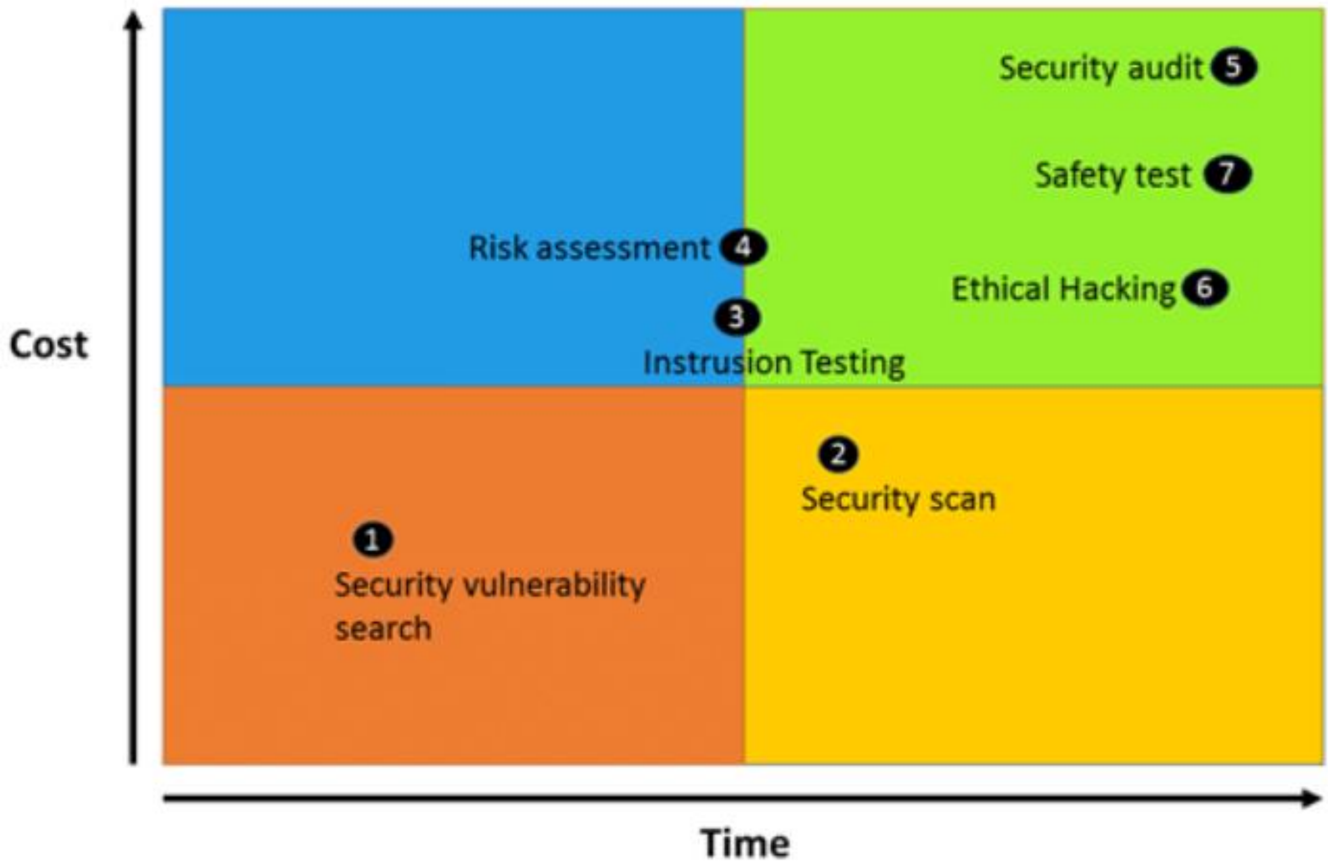

Рисунок 2.1. - Карта безпеки методології OSSTMM.

2.1.2 Аналіз методології NIST SP 800-115

NIST розробив і підтримує цей метод тестування. Він описує загальні етапи тестування на проникнення та технічні проблеми оцінки інформаційної безпеки компанії. Також описує рекомендації щодо аналізу результатів випробувань та розробки заходів щодо зменшення ризиків безпеки.

Виділяє 3 етапи тестування (рисунок 2.2):

1) Планування - описують суть тестування, цілі та задачі для тестування на проникнення;

2) Збір інформації - аналіз вразливостей;

3) Атака – лише якщо збір інформації дав результат то проводится сама атака на пристрій IoT;

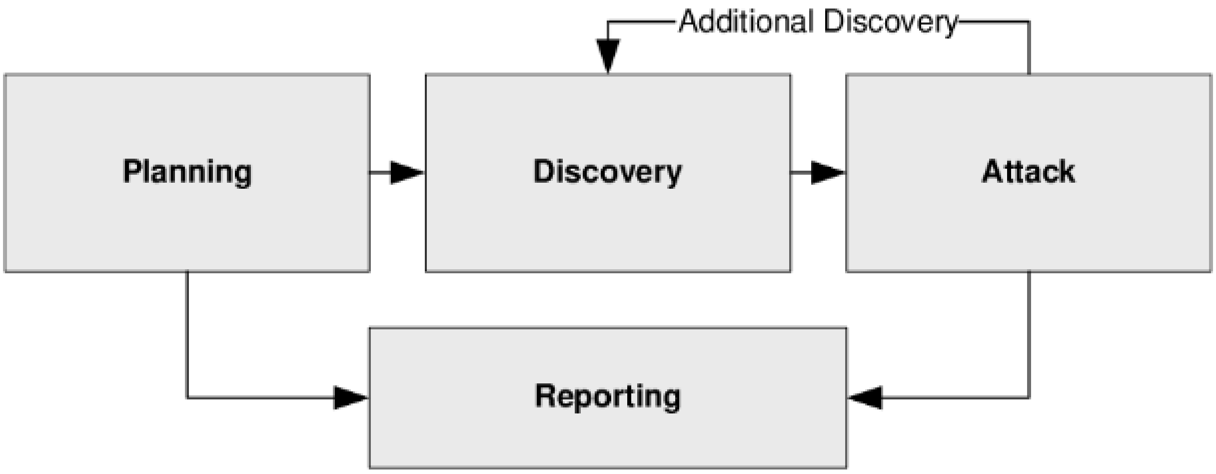

Рисунок 2.2. - З етапи тестування NIST SP 800-115.

Описані етапи аналізу отриманих даних, виявлення причин, що привели до появи вразливостей, розробка рекомендацій по їх знешкодженню та розробки звіту.

За даним документом, тестування на проникнення, окрім стандартних моделей застосувань, можна використовувати так і для визначення:

1) Наскільки захищенна система що до існуючих моделей атак;

2) Додаткових заходів протидії, які могли б послабити загрози.

3) Здатності методів захисту системи до виявлення атак і забезпечення відповідної реакції на них.

Метод розроблен як той, що може використовувати або посилатись на методології і методики інших виробників під час тестування. Також, можно додати, що методологія пояснює як проводити первинний збір інформації та як організовувати процес тесту, включаючи в себе методи оцінки вразливостей, що часто зустрічаються.

2.1.3 Аналіз методології PTES

Стандарт, розроблений для поєднання вимог бізнесу та можливостей постачальника безпеки з масштабованим тестуванням на вторгнення. PTES будується з 7 розділів і додаткового посібника, який описує технічні описи. 7 розділів містять таку інформацію:

1) Початкова взаємодія із замовником - з урахуванням каналів зв'язку, правил взаємодії та контролю, конкретних способів реагування та контролю подій;

2) Збір інформації – охоплює цілі та завдання методу, процес мислення тестувальника, що дозволяє спланувати тест;

3) Моделювання загроз - метод, що допомагає визначити процес тестування на проникнення;

4) Аналіз вразливостей - фокусується на виявленні вразливостей безпеки та методів роботи;

5) Використання вразливостей – цей розділ описує як можно обійти обмеженя безпеки та різних заходів захисту;

6) Після-операційний – Оцінка зараженої системи та пошук варіантів відновлення для подальших атак;

7) Готування звіту — описує основні критерії для написання звітів з тестування на проникнення.

Посібник-гайд описує особливості тестування переважно IT інфраструктури в підприємстві, аналіз безпеки соціальної складової (персонал, соціальні мережі) а також аналіз периметру безпеки. Також цей посібник-гайд покриває вибір інструментів тестування, методів збору інформації.

2.1.4 Аналіз методології ISSAF

Методологію ISSAF розробила група безпеки відкритих інформаційних систем (OSSIG). Згідно з методологією ISSAF, тестування на проникнення складається з 3 етапів:

1) Планування та підготовка тесту;
2) Збір інформації та проведення тестування на проникнення;
3) Написання звіту про проведені тести на проникнення.

Методологія ISSAF дозволяє проводити:

1) оцінку захищеності паролів;

2) оцінку захищеності мережевих пристроїв;

3) оцінку захищеності міжмережевих екранів;

4) оцінку захищеності систем виявлення вторгнень;

5) оцінку захищеності веб – додатків;

6) оцінку захищеності операційних систем;

7) аудит програмного коду;

8) аналіз захищеності баз даних.

2.2 Стандарти оцінки безпеки в IoT

Наразі існує лише один спосіб оцінити стан безпеки IoT: Оцінка безпеки GSMA IoT, розроблена асоціацією GSM [28]. Він ґрунтується на рекомендаціях з безпеки IoT тієї ж асоціації, які включають 85 детальних рекомендацій щодо безпечного проектування, розробки та розгортання послуг IoT. Розробники керівних принципів видавали їх як комплексний набір найкращих практик для полегшення безпечного проектування, розробки та впровадження рішень IoT а також переходу від підходу до вирішення проблем безпеки та оцінки ризиків. GSMA також надає контрольний список безпеки IoT - контрольний список запитань, які задають клієнти під час оцінки безпеки систем IoT, який також базується на рекомендаціях щодо безпеки IoT і зосереджується насамперед на потребах та умовах бізнесу через свою детальну структуру [29]. Містить рекомендації та запитання з наступних розділів:

1) Питання безпеки/конфіденційності IoT;

2) Список та пропозиції сервісних платформ;

3) Контрольний список та рекомендації щодо термінального обладнання.

Водночас опитування пов’язане з потребами бізнесу та враховує особливості розробки обладнання та програмного забезпечення для них.

Крім перерахованих вище технологій, можна згадати міжнародний проект Open

Web Application Security Project (OWASP), який зосереджується на аналізі недоліків програмного забезпечення, посиленні його безпеки та класифікації відомих уразливостей. OWASP розробив проект Top-10, зосереджений на найнебезпечніших атаках на веб-додатки, мобільні додатки, а наступний проект IoT Top-10 розроблявся протягом останніх кількох років [30], висвітлюючи основні проблеми безпеки в Інтернеті. Список топ-10 IoT за 2018 рік (останнє видання) включає такі проблеми безпеки:

1) Слабкі, легко вгадувані або встановлені у коді паролі
2) Небезпечні мережеві сервіси
3) Небезпечні екосистемні інтерфейси
4) Відсутність механізму безпечного оновлення
5) Використання незахищених або застарілих компонентів
6) Недостатній захист конфіденційності
7) Небезпечна передача та зберігання даних
8) Відсутність управління пристроями
9) Небезпечні налаштування за замовчуванням
10) Відсутність фізичного укріплення

Цей список вразливостей призначений переважно для розробників, виробників пристроїв, користувачів та тестувальників, для того щоб мати краще проаналізувати небезпечні прогалини в безпеці. Також в рамках проекту OWASP було створено стандарт IoT Security Verification Standard [31], який виступає аналогом до IoT Top-10. Взагалі, це збірник найпопулярніших рішень при розробці та виробництві IoT пристроїв, що описує модель безпеки IoT, має список основних вимог безпеки до розробки екосистем а також програмних платформ і пристроїв IoT та містить у собі такі розділи:

1) Вимоги до екосистеми IoT;
2) Вимоги до програмного забезпечення в просторі користувача;
3) Вимоги до програмних платформ;
4) Вимоги до протоколів комунікації;
5) Вимоги до апаратних платформ.

2.3 Проблематика тестування на проникнення Інтернету речей

Проблема стандартних методів тестування на вторгнення в секторі оцінки безпеки пристроїв IoT полягає в тому, що вони не охоплюють певні аспекти тестування на проникнення IoT, наприклад, оцінку пристроїв поверхні атаки. Крім того, не всі методології надають інструментів, які можна використовувати для тестування безпеки IoT, і жодна з методологій не враховують можливість оцінки ризику перед його виявленням. Навіть якщо подібні стандарти існують у просторі IoT (GSMA, OWASP), неможливо сказати, чи достатньо цих документів для ефективного тестування на проникнення. OWASP не має вказівок для IoT, а IoT Top-10 орієнтований здебільшого на розробників програмного забезпечення та операційних систем разом зі стандартом GSMA. Існуючі стандарти та методології тестування на проникнення дозволяють перевірити лише те, чи відповідає пристрій або система IoT певним базовим вимогам безпеки, які не покривають повну поверхню атаки, тобто пропускає досить велику кількість загроз, що впливає на гнучкість тестування. Також існуючі методології та стандарти (як загальні, так і в IoT) не пропонують набору інструментів, програмного забезпечення для проведення тестування на проникнення пристроїв Інтернету речей.

Висновки до розділу 2

У даному розділі проведно порівняльний аналіз існуючих методологій тестування на проникнення (OSSTMM, PTES, ISSAF, NIST SP 800-115, GSMA IoT Security Assessment, OWASP), в результаті якого встановлено, що вони:

1) Не торкається всіх етапів тестування на проникнення пристроїв IoT, таких як фізичні тести безпеки, тести безпеки програмного забезпечення, тести безпеки мобільних застосунків;

2) Недосконалий опис а саме непослідовні етапи тестування а також не покривають повноі поверхні атаки;

3) Не описють інструменти, програмні забезпечення які потрібни для тестування IoT пристроїв;

4) Мають недостатню гнучкість.

Ефективність тестування на проникнення пристроїв IoT, особливо коли не має достатньо часу на тестування, залежить в першу чергу від алгоритму дій, якого треба дотримуватись для належного тестування. Тому, тестувальнику потрібен додатковий час на пошук потрібних етапів для поточного тестування на проникнення з існуючих методологій та методичних вказівок, тому що всі вони не струтуровані та стосуються різних пристроїв, що може негативно вплинути на сам процес тестування і результати. Через ці всі проблеми виникає потреба в розробці або удосконалення методу тестування на проникнення, що містив би кращі сторони проаналізованих існуючих стандартів, а також перелік необхіднихпристроїв засобів тестування IoT.

РОЗДІЛ 3
УДОСКОНАЛЕННЯ МЕТОДУ ТЕСТУВАННЯ ПРИСТРОЇВ ІНТЕРНЕТУ РЕЧЕЙ НА ПРОНИКНЕННЯ

Метод – це спосіб пізнання і практичної зміни дійсності, совокупність прийомів, що визначає практичну і пізнавальну діяльність людей. Таким чином, можливо вызначити головну функцію методу, яка полягає у внутрішній організації та регулюванні процесу пізнання або практичної зміни того або іншого об'єкта [32].

Те що вивчається, також ділить методи на дві групи: природознавство і соціально-гуманітарні. У зв'язку з цим виділяють якісні і кількісні методи.

У нашому випадку використовується поняття якісного методу. Вони націлені не на вивчення широкого спектру проявів обкту, а орієнтуються на розкриття причинно-наслдкових явищ.

3.1 Постановка завдання

Для початку потрібно проаналізувати вже існуючі методи тестування на проникнення, поглиблено розібрати всі пункти тестування, які проблеми можуть виникнути в ході тестування, виділити слабкі сторони та вразливості у методі та на основі цього удосконалити метод тестування на проникнення використовуючи більш глибокі інструменти чи програмне забезпечення

3.1.1 Формулювання завдання

Для того щоб розробити ефективний метод тестування на проникнення пристроїв IoT, необхідно:

- Описати основні потреби методу тестування;
- Визначити етапи тестування;
- Удосконалити метод тестування пристроїв IoT на проникнення, що давав би

можливість збільшити площу поверхні атаки та прискорити прийняття критичних рішень у процесі тестування.

3.1.2 Формулювання вимог до методу

Метод тестування пристроїв IoT на проникнення повинен відповідати основним вимогам, а саме:

1) метод повинен враховувати всю поверхню атаки та основні критичні загрози IoT;

2) метод повинен враховувати список з проекту OWASP IoT Top-10 та розглянутих методів тестування на проникнення;

3) метод повинен містити інформацію про можливі загрози та вразливості, що можуть бути знайдені;

4) метод повинен описувати можливі проблеми при тестуванні на проникнення та способи їх рішення;

5) метод повинен описувати які саме інструменти можуть використовуватись при оцінці безпеки IoT та їх використання;

6) метод повинен охоплювати всі основні особливості тестування на проникнення.

3.1.3 Удосконалення методу тестування

Виходячи з вищесказанного, можливо виділити наступні послідовні етапи тестування на проникнення:

1) Підготовка інструментів для тестування;

2) Збір інформації;

3) Тестування фізичної безпеки пристроїв;

4) Тестування безпеки мережевих сервісів та протоколів;

5) Тестування безпеки веб-інтерфейсів та веб-застосунків;

6) Тестування безпеки мобільних застосунків;

7) Оцінка вразливостей.

Щоб оцінити вразливості які були знайдені с етапів вище потрібно використовувати спільну систему оцінювання вразливостей CVSS v3.1 (рисунок. 3.1), що відображає характеристики та суворість вразливостей, як програмних, так і апаратних [33]. CVSS складається з трьох метричних груп:

1) Базової (Base),

2) Часової (Temporal),

3) Екосистемної (Environmental).

Базова група, це група де показник відображає ступінь критичності знайденної вразливості відповідно до її внутрішніх характеристик, які є постійними, і передбачає найважчий вплив на різні системи.

Часова група, вона коригує базову групу, основуючись на загальні фактори, що змінюються з часом, наприклад, чи заражен коду системи експлоїтом чи іншого доказу, що вразливість існує.

Екосистемна група адаптує базову та часову групу до конкретного середовища чи системи. Ця група аналізує наступні фактори, як наявність виправлень у цьому середовищі. Дивлячись на конкретний випадок, достатньо буде оцінки з використанням лише базової групи через те, що не всі фактори інших груп не завжди можуть бути застосовані. Загальна оцінка будується з базового результату і векторної строки, яка стисло текстом відображає складові оцінки.

Основні пунтки CVSS v3.1 складаються з наступних категорій:

1) Вектор атаки: він визначає контекст, можливі вразливості;

2) Складність атаки: виявляє умови атаки, які повинні бути для працездатності експлойту;

3) Рівень необхідних прав доступу: визначає рівень прав доступу, який повинен мати атакуючий до проведення атаки;

4) Взаємодія з користувачем: визначає, можливо атакувати тільки за участі атакуючого, чи потрен користувач;

5) Границі експлуатації: визначає, чи можливе за допомогою застосування вразливості вийти за межі використання;

6) Вплив на тріаду CIA: визначає вплив на конфіденційність, цілісність та доступність.

Дані метрики формують оцінку вразливості від 0 до 10 де 0 – це нульовий рівень вразливості а 10 - критичний рівень вразливості

Таким чином, дана система CVSS дозволяє кількісно, оцінити критичність знайденої вразливості.

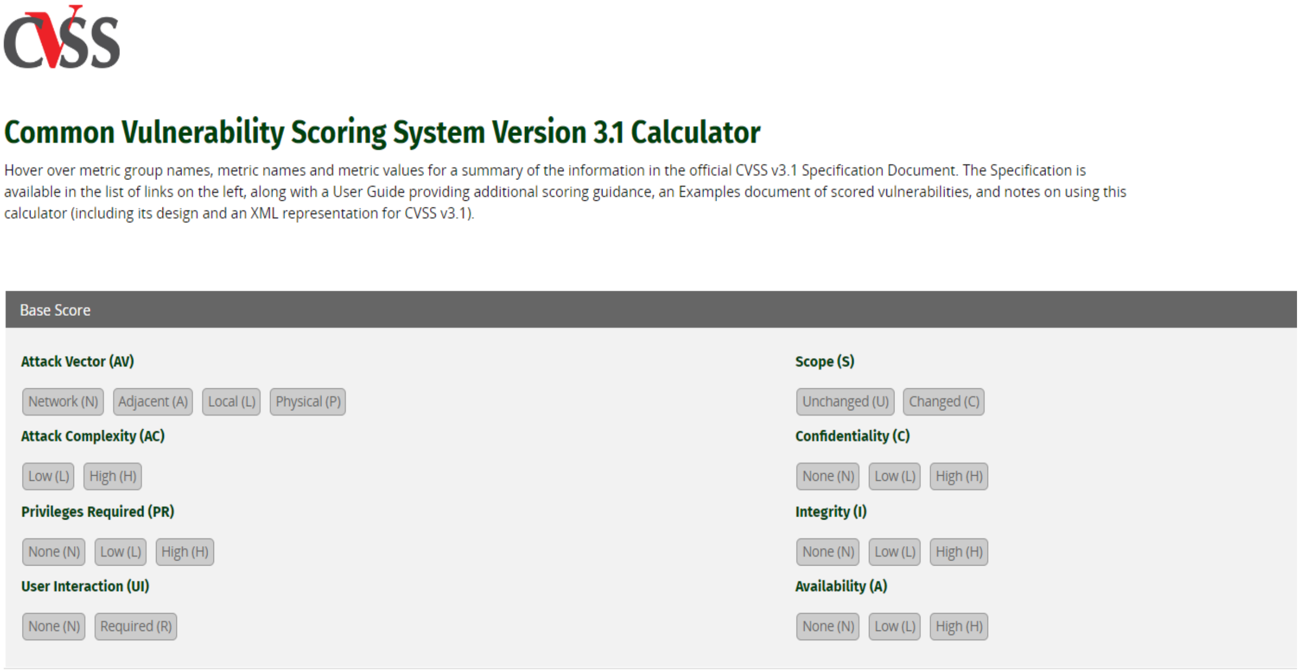

Рисунок 3.1 - Спільна системаоцінювання вразливостей CVSS v3.1.

3.1.4. Підготовка інструментів для тестування

При тестуванні на проникнення пристроїв IoT можливо використовувати велику кількість інструментів, як програмних, так і апаратних. В якості операційних систем для тестування пристроїв IoT використовуються такі дистрибутиви, побудовані на основі Linux:

- OS Kali Linux – поширений інструмент тестувальників на проникнення, підходить для більшості тестів інфраструктури, веб-застосунків, певних мережевих тестів [34];
- Attify OS – аналог Kali Linux, який містить встановлені інструменти,

призначені спеціально для тестів IoT [35];

- Dragon OS – написаний на Debian дистрибутив, що містить встановлені програми для роботи з SDR [36].

На всіх етапах тестування знадобитись різні інструменти, як для аналізу так і для збору інформації. Для програмного забезпечення , що не знаходяться у складі дистрибутивів таких як Kali linux та Attify, наводяться посилання на офіційні джерела для скачування.

1) Фізичний рівень: набір інструментів таких як викрутки, плоскогубці та інші.

2) Також для отримання доступу до портів відлагодження потрібно мати апаратні інструменти:

- JTAG: JTAGulator [38], Attify Badge [39], Shikra [40];
- UART: Attify Badge, Shikra;

Ці адаптери створені для спрощення експлуатації вразливостей на фізичному рівні.

Основні програмні інструменти для взаємодії з апаратним рівнем використовують такі:

- OpenOCD – це програмне забезпечення для програмування та відлагодження мікроконтролерів;
- GDB — відлагоджувач, який дає змогу працювати з майже всіма мовами програмування;

3) Програмне забезпечення: для зворотньої розробки програмного забезпечення зазвичай використовують утиліту binwalk, яка входить до складу Attify та Kali Linux. Рідше використовують Firmware Analysis Toolkit, який схожий до binwalk’а а також функціонал дозволяє емулювати програмне забезпечення пристроїв.

4) Мережеві сервіси і протоколи:

Для тестування безпеки сервісів та протоколів IoT на предмет шифрування та захищеності від основних критичних атаках, тестувальник потребує відповідні фізичні засоби для тестування таких протоколів як ZigBee, BLE, LoRaWAN а також сімейства 802.11:

- ZigBee: XBee [42], Atmel RzRaven [43];

- Bluetooth LE: Ubertooth One [45], ESP-based адаптери;
- LoRaWAN: LoStik [46], CatWAN USB Stick [47].
- 802.11: найпопулярніший це TP-Link TL-WN722N що підтримує режим моніторингу [48].

Перелік апаратних засобів вище — це адаптери або сніфери, які призначені для моніторингу трафіку у мережах, що дозволяють збирати інформацію про пакети в мережі, а також здійснювати певний набір атак.

Для збору такої інформації зазвичай використовують такі утиліти, як:

- tcpdump: збір інформаці про мережеві пакети;
- zbdump: збирає пакети для протоколу ZigBee;
- Wireshark: аналіз мережевих пакетів;
- nmap: виявлення активних та вимкнених сервісів а також підключені пристрої.

Для тестування на стійкість засобами атак грубої сили найпопулярніші це утиліти Hydra та Medusa які встановлені в Kali.

5) Веб-інтерфейси і веб-застосунки

Для тестування на проникнення веб-застосунків використовують наступні проксі як Burp Suite та OWASP ZAP, які дозволяють перехоплювати запити до сервера та детально вивчати їх структуру, вивчати структуру запитів, а також проводити атаки відмови в обслуговуванні (DOS) та багато іншого. Dirbuster та dirb утиліти які також встановлені в Kali та використовують для перебору шляхів до файлів та веб-сторінок, які недоступні з інтерфейсу застосунку. Також можуть використовувати такі утиліти для спеціальних аналізів:

- sqlmap
- sslscan
- nikto
- BeEF [49]
- Hydra, Medusa

6) Мобільні застосунки:

Для аналізу мобільних застосунків під ОС Android можливо використання таких програмних пакетів та утиліт:

- APKtool: утиліта для аналізу складу apk-файлу [51];
- JADx: утиліта для зворотньої розробки Android-застосунків;
- Android Studio: утиліта для відлагодження застосунків [52];
- Android Debug Bridge (adb): відлагодження пристроїв під управлінням Android.

Розробники вказаних програмних пакетів підтримують як операційні системи Linux, так і Windows системи.

При аналізі мобільних застосунків також використовуються проксі Burp Suite, OWASP ZAP для вивчення клієнтсерверної комунікації, а також утиліти MobSF [52] (автоматичний аналізатор) та Frida [53].

3.1.5 Проблеми, з якими може зіштовхнутись тестувальник

Особливість тестування на проникнення IoT може бути досить складною задачею для тестувальника, тому що поверхня атаки достатньо обширна, особливості самого концепту IoT також досить складні через свою унікальність а також широкий спектр пристроїв тапрограмних платформ, що вони використовують.

Найпопулярніші складнощі, які виникають при тестуванні на проникнення пристроїв IoT, виділяють наступні:

1) Тестування фізичної безпеки.

На фізичному рівні існуючі загрози можна подолати, використовуючи основні засоби недостатнього фізичного захисту: доступ до фізичних портів (наприклад, USB), слотів для карт пам’яті, розбирання корпусу пристрою для доступу до материнської плати пристрою, доступ до портів налагодження та флеш-пам’яті. У разі зовнішніх пристроїв вводу/виводу експлуатація часто не є складним процесом, тому в разі розбирання пристрою конструктивні особливості корпусу можуть бути проблематичними. У процесі збору інформації можуть бути знайдені дані про те, як розібрати пристрій, але подальші шанси на успішне зняття корпусу та доступу до материнської плати залежать від навичок тестувальника в ремонті та обслуговуванні портативних електронних пристроїв. Без належного досвіду ви можете пошкодити компоненти корпусу та/або компоненти материнської плати, що може негативно вплинути на подальше проведення тестування на проникнення.

2) Тестування безпеки програмного забезпечення пристроїв.

Отримати програмне забезпечення пристрою для аналізу можна декількома способами, а саме:

- Пошук OSINT;

- Фізична експлуатація;

- Перехоплення заводського оновлення програмного забезпечення OTA (Over The Air).

Складнощі у отриманні програмного забезпечення методами фізичної експлуатації посилаються на попередній пункт, у випадку “оновлення через повітря” треба мати можливість перехоплення безпровідної комунікації за протоколом, яким пристрій комунікує з сервером оновлень. Для цього використовуються методи сніфінгу для збереження перехоплених пакетів, і у випадку, коли трафік зашифрований наприклад протоколом HTTPS, потрібно встановити кореневий сертифікат TLS на пристрої, що може бути неможливо через особливості операційної системи, на якій працює пристрій. Тобто, на цьому етапі можливо оцінити, наскільки безпечно передається програмне забезпечення на пристрій і зробити висновки про відповідні механізми захисту.

Після отримання програмного забезпечення проводиться її аналіз, що може бути ускладнений наявністю шифрування. Перевірку на наявність шифрування дозволяє зробити утиліта binwalk. За її допомогою можна визначити, чи є образ програмного забезпечення скомпресованим і не шифрованим, або ж є зашифрованим, зробивши аналіз файлу. Графік з коливаннями (нерівний) свідчить про те, що програмне забезпечення скомпресовано (рисунок 3.2).

У разі зашифрованого програмного забезпечення графік виглядатиме як рівна пряма.

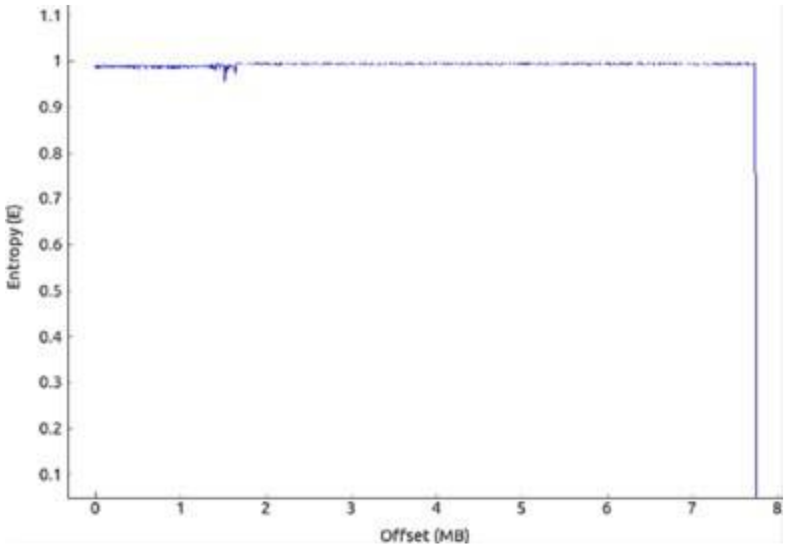

Рисунок 3.2. – Скомпресована програмне забезпечення.

3) Тестування безпеки мережевих сервісів та протоколів.

Через структуру пристрою та опероційній системі IoT можуть виникнути проблеми з тестуванням мережевих служб та протоколів: при використанні утиліти nmap пристрій може перестати працювати належним чином на певний період часу або до наступного завантаження, оскільки nmap надсилає спеціальний формат пакетів на пристрій, який може не обрабляти, а також він може надсилати відносно велику кількість різних «пробивних» пакетів, які деякі пристрої можуть не обробляти належним чином. Невеликі пристрої IoT, такі як медичні пристрої, створені та налаштовані на обробку тільки правильних і добре сформованих пакетів, але не можуть належним чином реагувати на «зловмисні» пакети. Тому тестувальники повинні враховувати часті можливі витрати і необхідність перезавантаження пристрою для відновлення працездатності.

4) Тестування безпеки мобільних застосунків.

Найпоширенішою проблемою, з якою стикаються тестувальники, коли проникають на етап тестування веб-додатка, є обхід так званого SSL, протоколу мобільних операційних систем, що використовує сертифікат серверу у самому застосунок в обхід системного сховища, що ускладнює прослуховування трафіку засобами проксі-серверу. Останнім з дієвих варіантів розв’язання даної проблеми –

використовувати плагін NoPE для проксі Burp Suite, який дозволяє нормально перехоплювати запити і відповіді до сервера без помилок у випадку,якщо пристрій не використовує протокол HTTP [5454]. Цей плагін дозволяють приймати запити з пристрою на створений DNS-сервер та перехоплювати таким чином трафік застосунку.

3.2 Збір інформації

Під час збору інформації та аналізу функціоналу пристрою IoT важливо побачити весь спектр функцій, що він пропонує, тому що описані у попередньому підрозділі пункти можуть не відноситись прямо до певного функціоналу пристрою. Так при тестуванні IoT часто тестувальники використовують підхід «сірого ящику», в якому необхідно знайти як можливо більше інформації про сам пристрій, апаратну складову, функціонал, програмну складову і протоколи, що використовуються, і під час тестування доповнювати отриманні знання з відкритих джерел практичними перевірками. Таким чином зі звичайної наклейки на роутері ми можемо дізнатися логін і пароль, який зазвичай ніхто не змінює, а також додаткову інформацію (рисунок 3.3).

Рисунок 3.3 – Наліпка з інформацією про пристрій.

3.2.1 Фізична безпека

Збирати інформацію під час тестування на проникнення IoT на фізичному рівні, треба розуміти структуру пристрою, визначити його основні функції та вміти розбирати, щоб отримати доступ до материнської плати та пов'язаних інтерфейсів. Для цього треба використовувати засоби OSINT:

1. Google Patents: Патентна база даних часто може дати розуміння загальної функції пристрою; [5555];

2. База даних FCC: ідентифікатори пристроїв FCC дозволяє шукати специфікації пристрою на офіційному веб-сайті Федеральної комісії зі зв'язку (або інших джерелах) для подальшого дослідження функцій і схем, які створюють пристрій. Ідентификатор зазвичай виконаний у вигляді наліпки на днищі пристрою (рисунок. 3.3).

3. Такі наліпки також містять облікові дані або інші дані для входу в інтерфейс керування пристроєм як адмін, і ці теги часто залишаються незмінними навіть після початкового налаштування пристрою, що є однією з основних проблем безпеки пристроїв Інтернету речей.

4. Документація та посібники користувача: найпростіший із способів швидкого вивчення основних функцій пристрою та зробити певні висновки про можливу поверхню атаки і вектори загроз. В тих самих посібниках часто друкують інформацію про початкове налаштування пристрою і містять інформацію про заводські облікові дані до адміністративних інтерфейсів, застосунків тощо. Рідко коли користувачі повністю і коректно завершують початкове налаштування пристроїв, тому такі джерела є корисними для пошуку можливих даних для входу в програмне забезпечення під обліковими даними адміністратора або некоректних налаштувань.

5. Додатково засобами OSINT при тестуванні на проникнення потрібно перевіряти наявність відомих вразливостей пристрою у відкритих базах вразливостей:

1) Exploit-db: база експлоїтів до відомих вразливостей, також містить так звані робочі концепти (Proof-of-Concept) експлоїтів для IoT пристроїв [**Error! Reference source not found.**];

2) National Vulnerabilities Database: база даних вразливостей для програмних забезпечень, яка також містить інформацію і про пристрої IoT [57];

3) Google Hacking Database (GHDB): проект від авторів Exploit-db, що дозволяє виконати спеціально сформований пошуковий запит у Google для знаходження незахищених пристроїв чи відомих вразливостей/проблем безпекидля них [58].

3.2.2 Встановлення програмного забезпечення

Збір інформації щодо версії операційної системи пристрою часто зводиться до перехоплення оновлення пристрою, а також до пошуку образів програмного забезпечення методами OSINT:

1) Офіційні веб-ресурси виробників: часто існує можливість завантажити образ програмного забезпечення з сайту виробника, яка надає можливість вивчити її, не маючи доступу до самого пристрою;

2) Google Hacking Database: у випадку з програмним забезпеченням запити вигляду можуть допомогти знайти програмне забезпечення на інших веб-ресурсах та у випадку, коли виробник не пропонує його до скачування з офіційного сайту. GHDB містить запити до пошукових систем, які чітко вказують, що саме шукати (параметр intext) та де саме шукати (intitle), що збільшує шанси на знаходження потрібного файлу.

3.2.3 Мережеві сервіси та протоколи

Під час збору інформації про мережеві послуги та протоколи потрібно визначити основні протоколи передачи даних, які використовються в пристрої, пошукати інформацію за допомогою відкритого коду, щоб зрозуміти, які інструменти потрібні, які функції тесту, протоколи та потенціал та які проблеми можуть бути виявлені під час перевірки безпеки. Наступним кроком у зборі інформації це використання специфічного протоколу апаратного забезпечення для захоплення частин даних, наприклад, між пристроєм і сервером або між керуючим пристроєм і керованим пристроєм. Аналіз протоколів прикладного рівня, таких як CoAP, MQTT, XMPP, проводиться так само, як і інші протоколи прикладного рівня, наприклад HTTP, DNS, DHCP тобто за допомогою утиліти Wireshark. Для інших протоколів процес майже такий же. Для аналізу перехопленного трафіку необхіден бездротовий адаптер, який може працювати в режимі моніторинга, і утиліта tcpdump або Wireshark, за допомогою яких створюються pcap-файли, які містять інформацію про перехоплені пакети, а далі підлягають аналізу. У випадку з мережевими сервісами використовується утиліта nmap для отримання інформації про доступні зовні сервіси і відкриті пристрою.

3.2.4 Веб-інтерфейси і вебзастосунки

Починається збір інформації з вивчення функціоналу веб-застосунку. Визначається місце введення даних, таких як логін і пароль, наявність прихованих полів та параметрів. Щоб збільшити ефективність збору інформації використовуються утиліти проксування трафіку Burp Suite та OWASP ZAP, які здійснюють пасивне ведення журналу під час ручного перегляду сторінок і активно сканують потрібні нам URL-адреси. На цьому етапі інструмент пошуку DirBuster та dirb які присутні у Kali можна використовувати для пошуку сторінок веб-додатків або іншого прихованого вмісту сторінок. Ці програми перебирають популярні шлях до сторінки із бібліотек. За допомогою цього, наприклад, ви можете виявити приховані URL-адреси на камері відеоспостереження, які дозволяють керувати віддалено наприклад робити знімки без аутентифікації. Також важливо визначити точки входу, куди веб-додатки отримують дані користувача, оскільки більшість вразливостей у веб-додатках виникає через недостатню фільтрацію ненадійного введення користувача.

Також важливим буде зрозуміти, як відбувається аутентифікація користувача, які стоять протоколи безпеки, наприклад чи є захист від перебору паролів на сторінці входу.

Збір інформації про веб-застосунок також можливо зробити автоматично за допомогою утиліт sslscan та nikto. Це автоматичні сканери, які перевіряють безпеку шифрування трафіку та налаштування безпеки сервісів і самого веб-серверу.

3.2.5 Мобільні застосунки

Процес збору інформації при тестуванні мобільних застосунків певним чином схожий на процес з веб-застосунками. Почати треба з вивчення функціоналу застосунку та проаналізувати роботу процесів. За допомогою Burp Suite аналізується клієнт-серверна система: визначаються користувацькі точки входу, перевіряються запити застосунка у різних варіантах користування. Це також треба для пошуку вразливостей типу незашифрованої комунікації, передачі персональних даних у незахищеному виді. Також, потрібно аналізувати комунікації з хмарним API на предмет існування числових та легко вгадуваних параметрів. Програмою sslscan проводиться аналіз підключення до сервера на предмет слабких шифрів, застарілих сертифікатів. Утилітою JADx проводиться декомпіляція Android застосунків для вивчення вихідного коду, у випадку з iOS використовується Clutch2. Для більш детального вивчення безпеки мобільних застосунків потрібно використовувати фреймворк MobSF.

Базуючись на отриманої інформації, тестувальник вже може зрозуміти та зробити оцінку поверхні атаки пристрою та визначити які саме етапи тестування можуть бути релевантні в контексті тестування конкретного пристрою і почати процес самого тестування.

Висновки до розділу 3

В даному розділі було удосконаленно метод тестування на проникнення пристроїв IoT базуючись на сформульовані вимоги та поставлені задачі, а саме: удосконалення методів на основі кращих сторін існуючих методологій і стандартів, дотримання певної структури у розробці, аналізу можливих проблем у процесі а також опису інструментів, що використовуються при тестуванні приладів IoT. У вдосконаленному методі була врахована проблематика тестування, яка була досліджена у попередньому розділі, якими є основні етапи тестування:

1) Підготовка інструментів для тестування;

2) Збір інформації;

3) Тестування фізичної безпеки пристроїв;

4) Тестування безпеки програмного забезпечення ;

5) Тестування безпеки мережевих сервісів та протоколів;

6) Тестування безпеки веб-інтерфейсів та веб-застосунків;

7) Тестування мобільних застосунків;

8) Оцінка вразливостей.

Також було проаналізовано критичні проблеми, з якими може зустрінеться тестувальник, основний набір інструментів, як фізичних так и програмних, які можуть використовуватись при тестуванні на проникнення пристроїв IoT, та описані виділені етапи тестування. Додатково проаналізовано спектр поширених вразливостей.

ВИСНОВКИ

У роботі розглянуто що таке IoT - це мережева концепція, що складається з взаємопов'язаних фізичних пристроїв із вбудованими передавачами та програмним забезпеченням, яке дозволяє передавати дані та обмінюватися в автоматизованому режимі між фізичним світом і комп'ютерними системами за допомогою стандартних протоколів зв'язку.

Проаналізовано дослідження таких видань як Statista та Foresterr. Із цього зрозуміло, що одні з серйозних проблем безпеки IoT сьогодні, це відсутність достатньої кількості кваліфікованого персоналу, що займався би процесами імплементації безпечних практик у процесах розробки пристроїв та проблема стандартизації безпеки у IoT. Через це можно виділити основні проблеми пристроїв IoT, це - недостатня кількість персоналу, захист чутливих даних, втрата чи викрадення пристроїв IoT

Проведено порівняльний аналіз існуючих методологій (OWASP, GSMA IoT Security Assessment, OSSTMM, PTES, ISSAF, NIST SP 800-115), в результаті якого показано, що вони не перекривають всіх аспектів і поверхнь атак. З основного та критичного можна сказати про те, що вони не включають фізичних тестів безпеки, тестів безпеки програмного забезпечення, тестів безпеки мобільних застосунків. Також не описують список інструментів, програмних комплексів для тестування IoT пристроїв і є дуже вузькоспеціалізованими, що не дозволяє застосовувати їх до всіх пристроїв.

Через всі ці проблеми виникає потреба в удосконалені методів тестування на проникнення, що вміщував би кращі сторони існуючих стандартів, містив перелік засобів тестування та інструментів для тестування на проникнення пристроїв IoT.

Також було удосконалено метод тестування на проникнення пристроїв IoT на основі вимог та поставлених задач, а саме: удосконалення методу на основі кращих сторін існуючих методологій і стандартів, дотримання певної стандартизованої структури у розробці, аналізу можливих критичних проблем у процесі тестування, опису необхідних фізичних та програмних інструментів, що застосовуються при тестуванні IoT. У вдосконаленному методі взята до уваги проблематика тестування пристроїв IoT, яку визначали у другому розділі, також наведені відомості про найпоширеніші проблеми, з якими може стикнутись тестувальник.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Considerations for Managing Internet of Things (IoT) Cybersecurity and Privacy Risks.URL: https://doi.org/10.6028/NIST.IR.8228 (дата звернення: 21.01.2022).

2. That ‘Internet of Things’ Thing. URL: https://www.rfidjourn al.com/that-internet-of-things-thing (дата звернення: 22.01.2022).

3. 5G Standard, Release 17. — URL: https://www.3gpp.org/r elease-17 (дата звернення: 22.01.2022).

4. 7 Big Problems With the Internet of Things. URL: https://www.cm swire.com/cms/internet-of-things/7-big-problems-with-the-int ernet-of-things-024571.php (дата звернення: 22.01.2022).

5. Internet of Things - number of connected devices worldwide 2019-2029. URL: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/ (дата звернення: 25.01.2022).

6. The Internet of Things Myth. URL https://www.amazon.com/Internet-Things-Myth-William-Webb-ebook/dp/B086ZZBHYV (дата звернення: 25.01.2022).

7. Langnau L. The state of IoT. URL: https://www.designworldonline.c om/the-state-of-iot/ (дата звернення: 26.01.2022).

8. Predictions 2021: Technology Diversity Drives IoT Growth. URL: https://go.forrester.com/blogs/predictions-2021-technology-di versity-drives-iot-growth/ (дата звернення: 26.01.2022).

9. Top 10 Biggest IoT Security Issues. URL: https://www.intellectsoft.net/blog/biggest-iot-security-issues/ (дата звернення: 29.01.2022).

10. NIST Releases Draft Guidance on Internet of Things Device Cybersecurity. URL: https://www.nist.gov/news-events/news/2020/12/nist-relea ses-draft-guidance-internet-things-device-cybersecurity (дата звернення: 29.01.2022).

11. IoT Security: ENISA Publishes Guidelines on Securing the IoT Supply Chain. URL: https://www.enisa.europa.eu/news/enisa-news/iot-security-enisa-publishes-

guidelines-on-securing-the-iot-supply-chain (дата звернення: 01.02.2022).

12. IoT Security Foundation Publications. URL: https://www.iotsecurityf oundation.org/best-practice-guidelines/ (дата звернення: 01.02.2022).

13. Understanding the Mirai Botnet C. 1093—1110. URL: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis (дата звернення: 20.02.2022).

14. WannaCry Aftershock: Tech. report. URL: https://www.sop hos.com/en-us/medialibrary/PDFs/technical-papers/WannaCry-Aft ershock.pdf (дата звернення: 20.02.2022).

15. Cybersecurity Model Based on Hardening for Secure Internet of Things Implementation. — URL: 10.3390/app11073260 (дата звернення: 20.02.2022).

16. The Internet of Things reference model. URL: http://cdn.iotwf.c om/resources/71/IoT_Reference_Model_White_Paper_June_4_2014.p df (дата звернення: 29.02.2022).

17. Top Hardware Platforms for Internet of Things (IoT). URL: https://circuitdigest.com/article/top-hardware-platforms-for-internet-of-things-iot (дата звернення: 29.02.2022).

18. Top 15 Best IoT Operating System For Your IoT Devices. URL: https://www.ubuntupit.com/best-iot-operating-system-for-your- iot-devices/ (дата звернення: 10.03.2022).

19. Distribution of operating systems used for Internet-of-Things (IoT) devices, as of 2021. URL: https://www.statista.com/statistics/659581/wo rldwide-internet-of-things-survey-operating-systems/ (дата звернення: 11.03.2022).

20. Top Mobile App Development Frameworks in 2021. URL: https://www.clariontech.com/blog/top-mobile-app-development-framewo rks-in-2019 (дата звернення: 11.03.2022).

21. Which Is The Best Platform For Developing A Website. URL: htt ps://www.ikf.co.in/blog/best-platforms-for-website-developme nt-2020 (дата звернення: 11.03.2022).

22. Information Technology Laboratory. Computer security resource center. URL:

https://csrc.nist.gov/glossary/ter m/attack_surface (дата звернення: 11.03.2022).

23. The Open Source Security Testing Methodology Manual 3. URL: https://www.isecom.org/OSSTMM.3.pdf (дата звернення: 11.03.2022).

24. NIST SP 800-115. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pd f (дата звернення: 20.03.2022).

25. Penetration Testing Execution Standard. URL: http://www.pentest-standard.org/index.php/Main_Page (дата звернення: 20.03.2022).

26. OWASP TOP-10. URL: https://owasp.org/www-pdf-archive/OWASP- IoT-Top-10-2018-final.pdf (дата звернення: 20.03.2022).

27. Сравнительный анализ методик оценки межсетевых экранов. URL: http://ojs.itmo.ru/index.php/IMS/article/download/34/35 (дата звернення: 20.03.2022).

28. IoT Security Assessment Standard. URL: https://www.gsm a.com/iot/iot-security-assessment (дата звернення: 01.04.2022).

29. IoT SECURITY GUIDELINES. URL: https://www.gsma.com/iot/wp-content/uploads/2020/03/CLP.13-v2.2-GSMA-IoT-Sec urity-Guidelines-for-Endpoint-Ecosystems.pdf (дата звернення: 02.04.2022).

30. IoT Top-10. URL: https://owasp .org/www - pdf- archive/OWASP-IoT-Top-10-2018-final.pdf (дата звернення: 03.04.2022).

31. IoT Security Verification Standard (ISVS). URL: https://githu b.com/OWASP/IoT-Security-Verification-Standard-ISVS/releases/download/1.0RC/OWASP_ISVS-1.0RC-en.epub (дата звернення: 04.04.2022).

32. Методологія наукових досліджень. URL: https://en.ppt-online.org/87381 (дата звернення: 10.04.2022).

33. Common Vulnerability Scoring System v3.1. URL: https://www.first.org/cvss/v3.1/specification-document (дата звернення: 10.04.2022).

34. The Most Advanced Penetration Testing Distribution. URL: ht tps://www.kali.org (дата звернення: 10.04.2022).

35. Penetration testing distro for security professionals to assess the security of

Internet of Things (IoT) devices. URL: https://www.attify.com/att ifyos (дата звернення: 10.04.2022).

36. DRAGONOS: DEBIAN LINUX WITH PREINSTALLED OPEN SOURCE SDR SOFTWARE. URL: https://www.rtl-sdr.com/dragonos-debian-linux-with-preinstalled-open-source-sdr-software (дата звернення: 18.04.2022).

37. Pro Tech Toolkit. URL: https://www.ifixit.com/Store/Tools/Pro-Tech-Toolkit/IF145-307 (дата звернення: 18.04.2022).

38. JTAGulator 24-Channel Hardware Hacking Tool. URL: https://www.parallax.com/product/jtagulator (дата звернення: 18.04.2022).

39. Attify Badge - UART, JTAG, SPI, I2C. URL: https://www.attif y-store.com/products/attify-badge-uart-jtag-spi-i2c (дата звернення: 18.04.2022).

40. The Shikra. URL: https://int3.cc/products/the-shikra (дата звернення: 18.04.2022).

41. QEMU. URL: https://ru.wikipedia.org/wiki/QEMU (дата звернення: 25.04.2022).

42. Digi XBee. URL: https://www.digi.com/xbee (дата звернення: 25.04.2022).

43. RZRAVEN. URL: https://www.microchip.com/developmen (дата звернення: 25.04.2022).

44. ttools/ProductDetails/PartNO/ATAVRRZRAVEN (дата звернення: 25.04.2022).

45. Ubertooth One. URL: https://greatscottgadgets.co m/ubertoothone (дата звернення: 29.04.2022).

46. LoStik Open source USB LoRa® device. URL: https://www.crowdsupp ly.com/ronoth/lostik (дата звернення: 29.04.2022).

47. CatWAN USB Stick LoRa y LoRaWAN 915Mhz. URL: https://electroniccats.com/sto re/catwan-usb-stick (дата звернення: 29.04.2022).

48. TL-WN722N N150 Wi-Fi USB-адаптер высокого усиления. URL: https://www.tp-link.com/ru/ho me-networking/high-gain-adapter/tl-wn722n (дата звернення: 01.05.2022).

49. Crackle, crack Bluetooth Smart (BLE) encryption. URL: http://l

acklustre.net/projects/crackle (дата звернення: 01.05.2022).

50. The Browser Exploitation Framework. URL: http://beefproje ct.com (дата звернення: 01.05.2022).

51. A tool for reverse engineering Android apk files. URL: https://ibotpeaches.github.io/Apktool (дата звернення: 11.05.2022).

52. ANDROID STUDIO. URL: https://developer.android.com/studio.Clutch. URL: https://github.com/KJCracks/Clutch (дата звернення: 11.05.2022).

53. Mobile-Security-Framework-MobSF. URL: https://github.com/Mo bSF/Mobile-Security-Framework-MobSF (дата звернення: 11.05.2022).

54. Burp-Non-HTTP-Extension. URL: https://github.com/summitt/Burp- Non-HTTP-Extension (дата звернення: 11.05.2022).

55. Google Patents Search Engine. URL: https://patents.google.c om (дата звернення: 11.05.2022).

56. EXPLOIT DATABASE. URL: https://www.exploit-db.com (дата звернення: 19.05.2022).

57. NATIONAL VULNERABILITY DATABASE. URL: https://nvd.nist.gov (дата звернення: 19.05.2022).

58. Google Hacking Database. URL: https://www.exploit-d b.com/google-hacking-database (дата звернення: 19.05.2022).

59. Search Engine for the Internet of Everything. URL: https://www.sho dan.io (дата звернення: 19.05.2022).

60. Common Vulnerability Scoring System Version 3.1 Calculator. URL: https://www.first.org/cvss/calculator/3.1 (дата звернення: 21.05.2022).

61. Яку інформацію вказують на наклейці Wi-Fi роутера і чим вона корисна користувачеві. URL: https://gsminfo.com.ua/38208-yaku-informacziyu-vkazuyut-na-naklejczi-wi-fi-routera-i-chym-vona-korysna-korystuvachevi.html (дата звернення: 21.05.2022).