Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _Інформаційних радіотехнологій і технічного захисту інформації_____
(повна назва)

Кафедра _____Радіотехнологій інформаційно-комунікаційних систем_____
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти _____перший (бакалаврський)_____

Аналіз методів забезпечення безпеки в IoT системах
(тема)

Виконав:
студент _4_ курсу, групи группа_ITIP-20-1__
_____Климашевський Р.О._____
(прізвище, ініціали)

Спеціальність 126 Інформаційні системи та технології
(код і повна назва спеціальності)

Освітня програма Інформаційні технології інтернета речей
(повна назва освітньої програми)

Керівник _доктор філософії Мерзлікін А.О. _
(посада, прізвище, ініціали)

Допускається до захисту

В.о. зав. кафедри _____ ___Зрудний О.А._____
(підпис) (прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет_Інформаційних радіо технологій і технічного захисту інформації
Кафедра_Радіотехнологій інформаційно-комунікаційних систем_____
Рівень вищої освіти_____перший (бакалавр)_____
Спеціальність_____126 Інформаційні системи та технології_____
(код і повна назва)

Освітня програма Інформаційні технології інтернета речей _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)
«____»_____ 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові_____ Климашевському Руслану Олександровичу_____
(прізвище, ім'я, по батькові)

1. Тема роботи Аналіз методів забезпечення безпеки в IoT системах
_____
_____

затверджена наказом університету від від 27 травня 2024 р. № 500 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 10 червня 2024 р.

3. Вихідні дані до роботи:
літературні джерела та електронні ресурси за темою кваліфікаційної роботи
_____
_____
_____
_____

4. Перелік питань, що потрібно опрацювати в роботі:
перелічити назви всіх розділів роботи від вступу до додатків (див. зміст)
Вступ. 1 ОГЛЯД НАЯВНИХ МЕТОДІВ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ В ІОТ СИСТЕМАХ 2 ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ В СИСТЕМАХ ІНТЕРНЕТУ РЕЧЕЙ 3. МЕТОДИКА З БЕЗПЕКИ В ІОТ.Висновки. Перелік джерел посилання. Додатки

_____
_____
_____

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) _____
Комп'ютерна презентація
_____
_____
_____

_____
_____
_____
_____
_____
_____

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	Ph.D.. Мерзлікін А.О.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз наявних методів безбеки IoT	06.05.24- 15.05.2024	вик.
2	Методи аналізу та забеспечення безпеки IoT	16.05.2024-20.05.2024	вик.
3	Методика з безпеки в IoT	20.05.2024-23.05.2024	вик.
4	Висновки	24.05.2024-30.05.2024	вик.
5	Оформлення пояснювальної записки	01.06.2024-09.06.2024	вик.
6	Представлення роботи на кафедру	10.06.2024	вик.

Дата видачі завдання __06__ ___травня__2024 р.

Студент _____ _____Климашевський Р.О._____
(підпис) (прізвище, ініціали)

Керівник роботи _____ _____Ph.D.. Мерзлікін А.О._____
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи магістра містить 53 сторінки тексту, 7 рисунків, 15 джерел посилання, 2 додатки.

ІНТЕРНЕТ. РЕЧЕЙ. БЕЗПЕКА. АЛГОРИТМ.

Предметом дослідження є безпека IoT пристроїв.

Мета роботи – розробка методики з безпеки в iot.

У наслідок виконаної роботи Огляд та аналіз наявних методів забезпечення безпеки в iot системах. Розроблено програму для перевірки вразливостей у системах Інтернету речей (IoT)

Результати дослідження можуть бути використані для впровадження в IoT системи.

ABSTRACT

The explanatory note of the master's thesis contains 53 hundred pages of text, 7 figures, 15 references, 2 appendices.

INTERNET. THINGS. SECURITY. ALGORITHM.

The subject of research is the security of IoT devices.

The purpose of the study is to develop a methodology for security in IoT.

As a result of the work performed, a review and analysis of existing methods for ensuring security in iot systems. Developed a programme for checking vulnerabilities in Internet of Things (IoT) systems

The results of the study can be used for implementation in IoT systems.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧОК, СИМВОЛІВ, ОДИНИЦЬ І СКОРОЧЕНЬ

IoT – internet of things;

MAC – Media Access Control;

AES - Advanced Encryption Standard;

RSA - Rivest-Shamir-Adleman a;

TLS – Transport Layer Security;

DoS – denial-of-service attack;

ВСТУП

Інтернет речей (IoT) стає все більш поширеним і важливим аспектом сучасних технологій, проникаючи в багато сфер життя, від розумних будинків до промислових систем. Однак зі збільшенням кількості підключених пристроїв і обсягу зібраних даних зростає ризик вразливості та кібератак; забезпечення безпеки в системах IoT стає ключовим викликом для забезпечення конфіденційності, цілісності та доступності даних.

Метою цієї статті є аналіз методів забезпечення безпеки в системах Інтернету речей, щоб виявити існуючі вразливості та запропонувати ефективні рішення для їх усунення. Для досягнення поставленої мети було проведено огляд існуючих методів забезпечення безпеки, виявлено типові вразливості в системах IoT, проаналізовано методи їх виявлення та усунення, а також запропоновано рекомендації щодо підвищення рівня безпеки систем IoT.[1]

Дослідження в галузі безпеки систем IoT мають велике практичне значення, оскільки дозволяють підвищити рівень захисту персональних даних, забезпечити стабільну роботу системи та запобігти негативним наслідкам кібератак. Результати цього дослідження можуть бути використані розробниками та операторами систем IoT для підвищення їх безпеки та надійності.

1 ОГЛЯД НАЯВНИХ МЕТОДІВ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ В ІОТ СИСТЕМАХ

1.1 Аутентифікація та авторизація пристроїв

Автентифікація та авторизація пристроїв є важливими аспектами безпеки в системах Інтернету речей. Автентифікація пристрою - це процес перевірки автентичності пристрою, який гарантує, що пристрій дійсно є тим, за кого себе видає. Авторизація, з іншого боку, визначає права доступу пристрою до ресурсів і функцій системи після успішної автентифікації.

Для забезпечення безпеки систем IoT використовуються різні методи автентифікації та авторизації.

Паролі та приватні ключі: пристрої можуть використовувати паролі та приватні ключі для автентифікації при підключенні до мережі або обміні даними. Важливо забезпечити безпечне зберігання та передачу цієї інформації.

Цифрові сертифікати: цифрові сертифікати використовуються для автентифікації пристрою за допомогою його відкритих і закритих ключів. Це забезпечує вищий рівень безпеки, ніж використання лише паролів.

Біометрична автентифікація: пристрої можна захистити за допомогою біометричної автентифікації, наприклад, сканерів відбитків пальців або розпізнавання обличчя.

Токени доступу: токени доступу можна використовувати для тимчасового надання доступу до ресурсів і функцій на пристрої.

Багатофакторна автентифікація: кілька методів автентифікації (наприклад, пароль + SMS-код) можна комбінувати для підвищення безпеки системи.

Контроль доступу: централізація прав доступу до пристроїв і даних може запобігти несанкціонованому доступу.

Розробники систем IoT повинні враховувати рівень безпеки кожного методу автентифікації та авторизації і вибирати його, виходячи зі специфікацій системи та потреб безпеки.

Автентифікація пристрою. Методи автентифікації включають використання унікальних ідентифікаторів пристроїв, таких як MAC-адреси або серійні номери.[2]

Для більш надійної автентифікації можна використовувати криптографічні методи, такі як асиметричні ключі або цифрові сертифікати.

Важливо також переконатися, що ці ідентифікатори захищені від підробки та перехоплення.

Після успішної автентифікації пристрій повинен отримати дозвіл на виконання певних дій або доступ до певних ресурсів.

Механізм автентифікації повинен враховувати не лише факт автентифікації, але й використання пристрою, права доступу та політику безпеки системи.

Для підвищення безпеки можуть використовуватися методи посилення безпеки, такі як багатофакторна автентифікація, що вимагає декількох типів ідентифікації (наприклад, пароль або SMS) для входу в систему.

Аналіз поведінки пристрою також може бути використаний для виявлення незвичної активності та запобігання несанкціонованому доступу.

Однією з основних проблем безпеки при автентифікації та авторизації пристроїв є підробка ідентифікаційних даних та компрометація ключів безпеки.

Важливо постійно оновлювати та тестувати методи ідентифікації та автентифікації, що використовуються, щоб запобігти вразливостям.

Важливо, щоб методи автентифікації та авторизації пристроїв були простими у використанні та ефективно використовували ресурси.

Також важливо враховувати сумісність з іншими системами та стандартами безпеки.

Автентифікація та авторизація пристроїв є фундаментальними принципами безпеки в системах IoT, а їх ефективна реалізація може допомогти мінімізувати ризик кібератак і забезпечити загальну безпеку системи.

При централізованому підході всі пристрої авторизуються через центральний сервер або хмарну платформу.

При децентралізованому підході кожен пристрій автентифікується та авторизується за допомогою децентралізованого реєстру або технології блокчейн.

Процеси автентифікації включають обмін ключами шифрування, повторну автентифікацію сертифікатів і використання біометрії.

Важливо враховувати можливість атак перехоплення даних під час перевипуску ключів та сертифікатів. Процес автентифікації:

Після успішної автентифікації пристрій має бути авторизований для доступу до певних ресурсів або функцій системи відповідно до встановлених правил і політик.

Авторизація може ґрунтуватися на ролі пристрою або індивідуальних характеристиках пристрою.

Шифрування даних є важливим аспектом безпеки в системах IoT і повинно застосовуватися як при передачі даних через мережу, так і при їх зберіганні на пристрої.

Для забезпечення безпеки рекомендується використовувати надійні алгоритми шифрування та регулярно оновлювати ключі шифрування.

Важливо впровадити механізми контролю доступу, які дозволяють надавати або відкликати доступ до ресурсів залежно від обставин, що змінюються, та вимог безпеки.

Механізми контролю доступу включають рольові моделі, політики доступу та аудит доступу до ресурсів.

Аутентифікація та авторизація пристроїв в системах IoT є складним завданням, що вимагає комплексного підходу до безпеки. Ефективне

використання цих методів може запобігти несанкціонованому доступу та забезпечити загальну надійність системи.

1.2 Шифрування даних

Шифрування даних відіграє важливу роль у забезпеченні безпеки систем IoT. Воно допомагає захистити конфіденційність інформації, що передається між пристроями та процесорами даних; важливо розглянути деталі шифрування в контексті IoT:

Типи шифрування

Симетричне шифрування: для шифрування та дешифрування даних використовується один ключ. Наприклад, AES (Advanced Encryption Standard).[3]

Асиметричне шифрування: використовується пара відкритих і закритих ключів. Відкритий ключ використовується для шифрування, а закритий - для розшифрування. Приклади включають RSA (Rivest-Shamir-Adleman). На рисунку 1.1 показано схему шифрування.

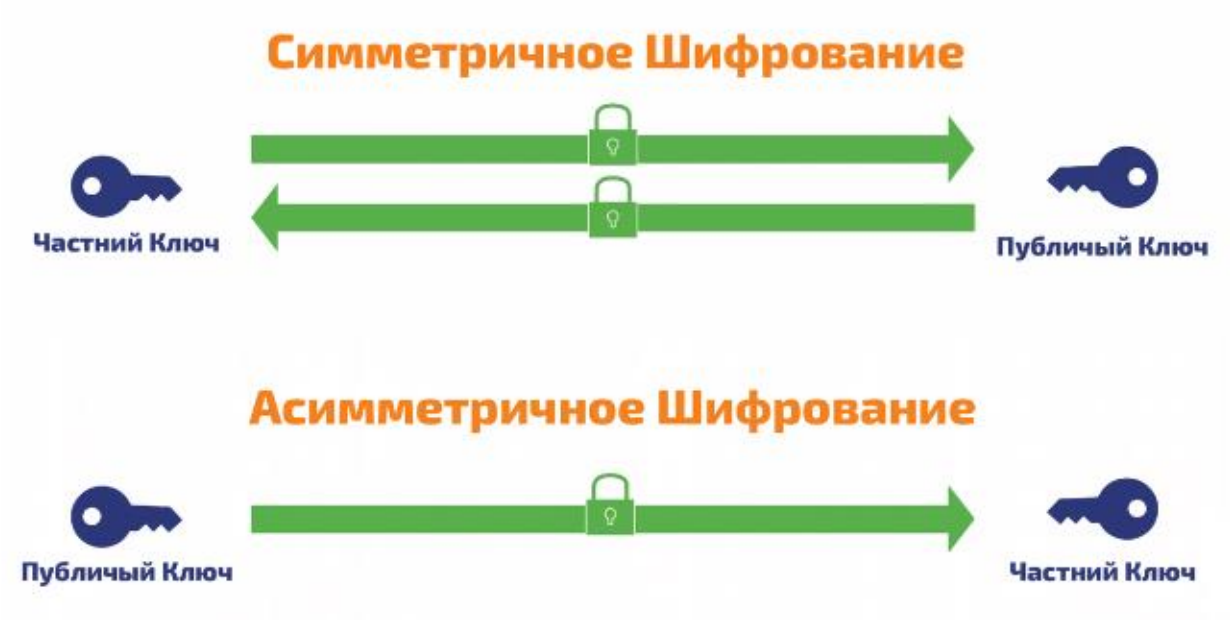

Рисунок 1.1 – Схема шифрування

Використання шифрування в IoT: Шифрування даних під час передачі між пристроями та серверами для захисту даних від несанкціонованого

доступу. Шифрування даних на пристроях захищає інформацію в пам'яті пристрою від витоку під час фізичного доступу. Керування ключами: Безпечне зберігання та спільне використання ключів шифрування має важливе значення для ефективного захисту даних. Використання протоколів шифрування, таких як Transport Layer Security (TLS) для обміну ключами, може допомогти захистити цей процес. Регулярне оновлення ключів: Регулярне оновлення ключів шифрування підвищує безпеку. Це пов'язано з тим, що якщо ключ скомпрометований, дані, зашифровані попереднім ключем, залишаються захищеними. Продуктивність і споживання ресурсів: При виборі алгоритму шифрування слід враховувати його вплив на продуктивність пристрою та споживання ресурсів. Деякі алгоритми шифрування вимагають великих обчислювальних витрат і можуть бути проблематичними для пристроїв з обмеженими ресурсами. Вибір алгоритму шифрування: При виборі алгоритму шифрування для системи IoT слід враховувати його безпеку, продуктивність і підтримку на цільовому пристрої. Деякі алгоритми, такі як AES, пропонують відмінне поєднання безпеки і продуктивності і широко використовуються в системах IoT. Захист від атак: Шифрування даних допомагає захистити інформацію від атак перехоплення, включаючи активні атаки, коли зловмисник намагається змінити дані під час передачі. Крім того, цілісність даних можна перевірити за допомогою схем автентифікації та цифрових підписів. Шифрування на рівні програми: У деяких випадках шифрування може бути реалізоване безпосередньо в додатку, особливо якщо дані обробляються на рівні програми. Це дозволяє гнучкіше керувати процесом шифрування та гарантує захист даних відповідно до конкретних потреб програми. Відповідність стандартам безпеки: При розробці систем IoT важливо дотримуватися стандартів безпеки, таких як ISO/IEC 27001, який визначає вимоги до управління інформаційною безпекою. Крім того, слід також враховувати вимоги до обробки персональних даних, зокрема відповідно до Загального регламенту захисту даних Європейського Союзу (GDPR). Забезпечення конфіденційності та цілісності: Метою шифрування даних в системах IoT є забезпечення конфіденційності (запобігання несанкціонованому доступу до даних) і цілісності (запобігання фальсифікації даних під час

передачі). Управління життєвим циклом ключа: Для ефективного шифрування даних важливо керувати життєвим циклом ключів, включаючи генерацію, зберігання, заміну та знищення ключів. Постійне оновлення ключів та використання різних ключів для різних цілей може зменшити ризик компрометації шифрування. Використання апаратного шифрування: Якщо ваш пристрій підтримує апаратне шифрування, використання апаратного шифрування може підвищити безпеку та продуктивність шифрування. Апаратне шифрування є більш ефективним, ніж програмне, і більш захищеним від певних типів атак. Використання шифрування в конкретних випадках використання IoT: У розумних будинках шифрування даних може захистити конфіденційність інформації про домашнє середовище і поведінку мешканців. У промислових системах шифрування може захистити від несанкціонованого доступу до даних про виробничі процеси. Навчання користувачів і розробників: Для ефективного використання шифрування в системах IoT важливо навчити користувачів і розробників правильному використанню та управлінню ключами шифрування. Це зменшить ймовірність помилок і вразливостей, пов'язаних з неправильним використанням шифрування. Використовуйте сторонні сервіси для шифрування даних: Ви можете зашифрувати свої дані за допомогою стороннього сервісу, наприклад, хмарного сервісу, який надає інструменти для шифрування та управління ключами. Однак при передачі даних до сторонніх систем необхідно враховувати питання безпеки та конфіденційності.

1.3 Моніторинг та аналіз безпеки.

Моніторинг та аналіз безпеки відіграють ключову роль у забезпеченні безпеки систем Інтернету речей. Ці процеси допомагають виявити вразливості, аномальну поведінку і потенційні загрози, що дозволяє швидко реагувати і запобігати інцидентам безпеки. Розглянемо основні аспекти моніторингу та аналізу безпеки в IoT: Збір даних про безпеку. Моніторинг безпеки систем IoT вимагає збору даних про події безпеки, такі як спроби несанкціонованого доступу, зміни конфігурації пристрою тощо. Ці дані можуть надходити з журналів подій пристроїв, систем моніторингу та інших джерел. Аналіз

безпеки: Дані про безпеку можна аналізувати для виявлення аномальної поведінки та потенційних загроз. Аналіз даних може використовувати машинне навчання та аналіз великих даних для виявлення незвичних або підозрілих патернів. Виявлення вразливостей: Моніторинг та аналіз безпеки можуть допомогти виявити вразливості системи, наприклад, недоліки в пристроях або конфігураціях програмного забезпечення. Це дозволяє швидко усунути вразливості та запобігти потенційним атакам. Реагування на інциденти: Моніторинг безпеки дозволяє швидко реагувати на інциденти безпеки, такі як атаки та порушення безпеки. Швидке реагування може мінімізувати шкоду, заподіяну інцидентами, і запобігти подальшій ескалації загроз. Забезпечення відповідності стандартам безпеки Моніторинг і аналіз безпеки можуть допомогти забезпечити відповідність систем стандартам безпеки, таким як GDPR і HIPAA. Це гарантує, що система обробляє дані користувачів відповідно до вимог законодавства та стандартів безпеки. Профілактичні заходи: На основі результатів моніторингу та аналізу безпеки можна вжити превентивних заходів для зниження ризиків і підвищення безпеки. До них відносяться оновлення програмного забезпечення, зміна налаштувань безпеки та вдосконалення політик безпеки. Інтеграція з системами управління загрозами: Інтеграція моніторингу та аналізу безпеки систем IoT з системами управління загрозами може допомогти ефективніше виявляти загрози та реагувати на них. Це дозволить швидше реагувати на нові загрози та атаки, використовуючи інформацію про поточний стан безпеки. Автоматизовані процеси моніторингу та реагування: Автоматизація процесів моніторингу та реагування може підвищити ефективність і швидкість реагування на загрози. Приклади включають автоматичне сповіщення про порушення безпеки та автоматичне застосування виправлень і оновлень безпеки. Моніторинг мережевого трафіку. Одним із способів виявлення аномальної поведінки і загроз є моніторинг мережевого трафіку в мережі IoT. Це може виявити підозрілу активність, таку як сканування портів, атаки на відмову в обслуговуванні та інші типи атак. Реагування на загрози: На основі моніторингу та аналізу безпеки повинна бути розроблена і впроваджена стратегія реагування на загрози. Це може включати блокування доступу до пристроїв і мереж, зміну налаштувань безпеки та інші

контрзаходи. Контроль доступу та автентифікація: Моніторинг безпеки включає контроль доступу та автентифікацію пристроїв і користувачів. Це запобігає несанкціонованому доступу до даних і мережевих ресурсів. Оновлення політики безпеки: На основі моніторингу та аналізу безпеки політики безпеки слід регулярно оновлювати, щоб реагувати на мінливі обставини та загрози. Це може підвищити ефективність захисту і знизити ризики безпеки. Моніторинг та аналіз безпеки в системах IoT відіграють важливу роль у забезпеченні безпеки інформації та її захисту від загроз. Правильно налаштована і використовувана методологія моніторингу та аналізу може допомогти виявити вразливості, аномальну поведінку і потенційні загрози, що дозволить швидко реагувати і запобігати інцидентам безпеки.

1.4 Уразливості IoT систем

Вразливості в системах Інтернету речей (IoT) становлять серйозну загрозу їхній безпеці та можуть мати різноманітні негативні наслідки, включаючи витік даних, порушення конфіденційності та доступності інформації, а також негативний вплив на фізичні об'єкти, що контролюються пристроями IoT. Системи IoT мають такі типові вразливості: на рисунку 1.2 показано схему впливу на вразливі місця в системах IoT.[4]

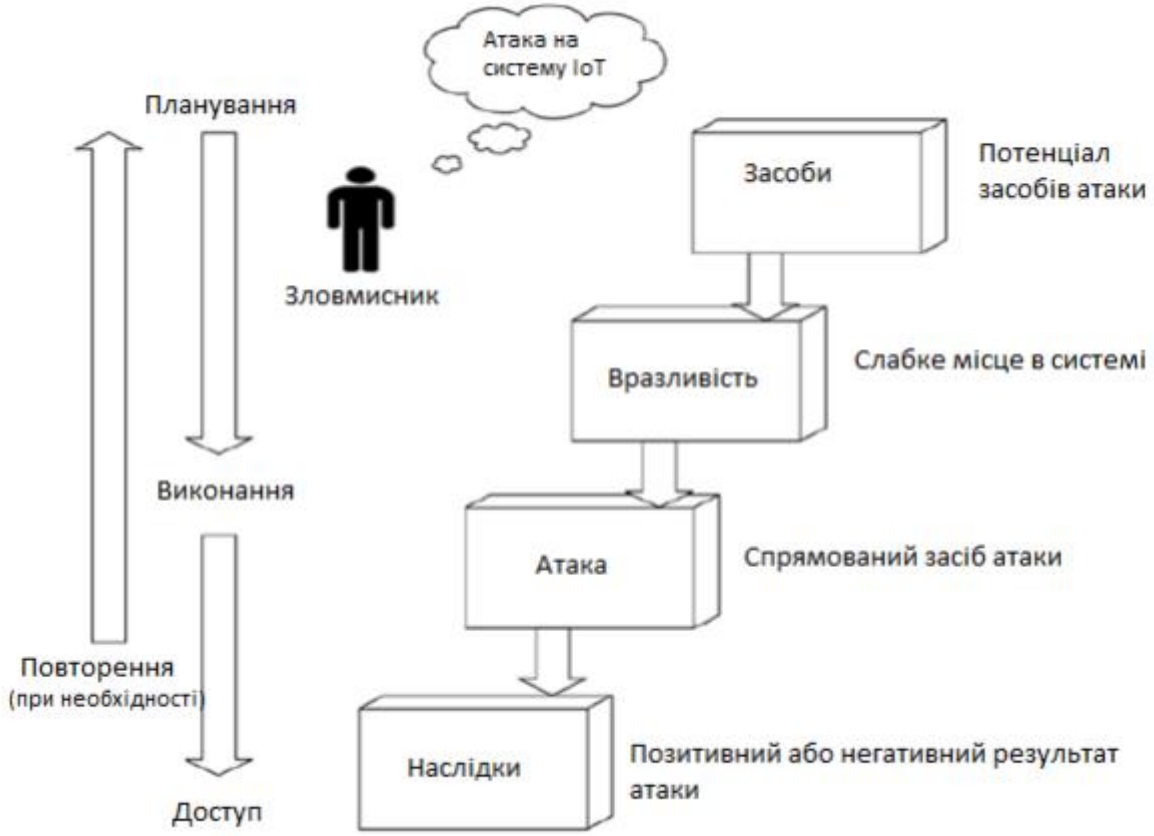

Рисунок 1.2 - Схема впливу на вразливі місця в системах IoT

Недостатня автентифікація та контроль доступу може призвести до несанкціонованого доступу до системи і конфіденційних даних.

Відсутність оновлень програмного забезпечення може призвести до появи невиправлених вразливостей.

Недостатній захист даних може призвести до перехоплення та витоку інформації.

Фізичні атаки можуть призвести до несправності пристроїв або витоку конфіденційної інформації.

DoS-атаки можуть перевантажувати мережеві ресурси і викликати відмову в обслуговуванні.

Зловмисники можуть змінювати конфігурацію пристроїв для атак або отримання несанкціонованого доступу.

Багато пристроїв IoT постачаються з попередньо налаштованими стандартними паролями та логінами за замовчуванням, що робить їх вразливими до атак грубого підбору.[5]

Деякі пристрої IoT можуть бути розташовані в недостатньо захищених місцях, що робить їх вразливими до фізичних атак.

Для забезпечення безпеки систем IoT рекомендується посилити автентифікацію та контроль доступу, регулярно оновлювати програмне забезпечення, шифрувати дані, забезпечити фізичну безпеку пристроїв, захищатися від DoS-атак, використовувати надійні паролі та логіни, контролювати та аудитувати конфігурації, проводити навчання користувачів і періодичний аудит безпеки.

Вразливості в системах Інтернету речей (IoT) становлять серйозну загрозу їхній безпеці та можуть мати різноманітні негативні наслідки, включаючи витік даних, порушення конфіденційності та доступності інформації, а також негативний вплив на фізичні об'єкти, що контролюються пристроями IoT. Системи IoT мають такі типові вразливості: недостатня автентифікація та контроль доступу, відсутність оновлень програмного забезпечення, недостатній захист даних, фізичні загрози, відсутність захисту від відмови в обслуговуванні (DoS), несанкціоновані зміни конфігурації, використання стандартних паролів і логінів, відсутність захисту від фізичного доступу. Для забезпечення безпеки систем IoT рекомендується посилити автентифікацію та контроль доступу, регулярно оновлювати програмне забезпечення, шифрувати дані, забезпечити фізичну безпеку пристроїв, захищатися від DoS-атак, використовувати надійні паролі та логіни, контролювати та аудитувати конфігурації, проводити навчання користувачів і періодичний аудит безпеки.

2. ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ В СИСТЕМАХ ІНТЕРНЕТУ РЕЧЕЙ

2.1 Методи аналізу та забезпечення безпеки в IoT системах

Для забезпечення безпеки систем Інтернету речей (IoT) використовуються різні методи:

Автентифікація та авторизація: автентифікація - це процес перевірки автентичності пристрою або користувача перед наданням доступу до системи. Це може включати використання паролів, біометричних даних або інших засобів ідентифікації.

Автентифікація: процес визначення прав доступу пристрою або користувача до системних ресурсів і функцій після успішної автентифікації.

Шифрування даних забезпечує конфіденційність інформації та робить її недоступною для несанкціонованого доступу. Для цього використовуються різні алгоритми шифрування, такі як AES (Advanced Encryption Standard) і RSA (Rivest-Shamir-Adleman).[6]

Моніторинг безпеки безперервно відстежує активність у мережі та на пристроях для виявлення аномалій і підозрілої активності.

Дані можуть бути проаналізовані для виявлення вразливостей і вжиття заходів для запобігання потенційним атакам.

Регулярні оновлення програмного забезпечення та патчі безпеки допомагають усунути відомі вразливості та запобігти потенційним атакам.

Фізична безпека передбачає захист пристроїв від фізичного доступу зловмисників. Наприклад, пристрої можна розміщувати в захищеному приміщенні або використовувати захисні корпуси.

Захист від атак типу "відмова в обслуговуванні" (DoS) і розподілених атак типу "відмова в обслуговуванні" (DDoS) включає фільтрацію трафіку і використання хмарних ресурсів для протидії атакам.

Захист вбудованих систем від загроз вимагає перевірки та забезпечення захисту вбудованих пристроїв від вразливостей безпеки.

Навчання користувачів основам інформаційної безпеки допоможе їм краще зрозуміти ризики і вжити відповідних заходів для захисту своїх пристроїв і даних.

Дотримання стандартів безпеки та проведення регулярних аудитів безпеки може допомогти виявити та усунути вразливості системи.

2.2 Шифрування даних у системах Інтернету речей

Шифрування даних у системах Інтернету речей (IoT) відіграє важливу роль, оскільки багато пристроїв збирають і передають чутливу інформацію. Наприклад, дані про здоров'я в медичних пристроях або інформація про звички та вподобання в "розумних" будинках.

Застосування шифрування даних в IoT системах допомагає захистити інформацію від несанкціонованого доступу та використання. Основні методи шифрування, що використовуються в IoT, включають в себе:

Симетричне шифрування: У цьому методі один і той самий ключ використовується для шифрування і розшифрування даних. Це ефективний метод, але вимагає безпечного обміну ключем між пристроями.[7]

Асиметричне шифрування: Цей метод використовує пару ключів - відкритий і закритий. Відкритий ключ використовується для шифрування, а закритий - для розшифрування. Цей метод забезпечує безпечний обмін ключами.

Хешування: Хешування використовується для створення унікального "відбитка" даних, який неможливо перетворити назад у вихідні дані. Хешування часто використовується для перевірки цілісності даних.

Симетричне шифрування - це метод шифрування, за якого один і той самий ключ використовується як для шифрування, так і для розшифрування даних. Це означає, що відправник і одержувач повинні мати спільний ключ, який використовується для захисту інформації.

Принцип роботи симетричного шифрування доволі простий: дані перетворюються на незрозумілий для людини вигляд із використанням ключа шифрування, а потім цей самий ключ використовується для перетворення шифротексту назад у вихідні дані. Основна перевага симетричного шифрування - висока швидкість роботи та ефективність.

Однак, симетричне шифрування також має свої недоліки. Один із головних - необхідність безпечного обміну ключами між відправником і одержувачем. Якщо зловмисник отримає доступ до ключа, він зможе розшифрувати всі зашифровані повідомлення.

Для забезпечення безпеки в IoT системах, де використовується симетричне шифрування, важливо приділяти особливу увагу безпечному обміну ключами. Це може включати в себе використання протоколів, таких як Diffie-Hellman key exchange, який дозволяє двом сторонам безпечно обмінятися секретним ключем, навіть якщо канал зв'язку небезпечний. Також важливо регулярно змінювати ключі шифрування і стежити за їх збереженням.

Асиметричне шифрування - це метод шифрування, за якого використовується пара ключів: відкритий і закритий. Відкритий ключ використовується для шифрування даних, а закритий - для їхнього розшифрування. Відкритий ключ може бути загальнодоступним і використовуватися для шифрування повідомлень, тоді як закритий ключ залишається секретним і відомий тільки власнику.

Принцип роботи асиметричного шифрування такий: якщо відправник хоче надіслати повідомлення одержувачу, він запитує в одержувача його відкритий ключ, шифрує повідомлення з використанням цього ключа і надсилає зашифроване повідомлення. Одержувач потім використовує свій закритий ключ для розшифрування повідомлення.[8]

Основна перевага асиметричного шифрування - відсутність необхідності обміну секретним ключем між відправником і одержувачем. Це робить процес обміну даними безпечнішим. Крім того, асиметричне шифрування дає змогу

використовувати цифрові підписи для перевірки автентичності повідомлень та ідентифікації відправника.

Однак, асиметричне шифрування зазвичай повільніше і вимагає більше обчислювальних ресурсів, ніж симетричне шифрування. Тому в практичних додатках часто використовується комбінація обох методів: дані шифруються симетрично з використанням випадкового ключа, який потім сам шифрується асиметрично з використанням відкритого ключа одержувача. Це дає змогу забезпечити високу швидкість і безпеку обміну даними.

Хешування - це процес перетворення вхідних даних довільної довжини у фіксований набір байтів (хеш-значення) з використанням хеш-функції. Хеш-функція має такі властивості:

Унікальність: Унікальність хеш-значення для кожного унікального вхідного значення.

Фіксована довжина: Хеш-значення має фіксовану довжину, незалежно від розміру вхідних даних.

Незворотність: Неможливість відновлення вихідних даних із хеш-значення.

Детермінованість: Для одного і того ж вхідного значення завжди генерується одне і те ж хеш-значення.

Хешування широко використовується для забезпечення безпеки та цілісності даних. Одним з основних застосувань хешування є перевірка цілісності даних. Наприклад, під час передавання файлу можна обчислити хеш-значення файлу до передавання і порівняти його з хеш-значенням файлу після отримання. Якщо вони не збігаються, це може вказувати на помилку в передачі або порушення цілісності файлу.[9]

Також хешування часто використовують для зберігання паролів. Замість зберігання самих паролів у базі даних, система зберігає їхні хеш-значення. Під час автентифікації користувачів, введений пароль хешується і порівнюється з хеш-значенням із бази даних. Це забезпечує безпеку паролів, навіть якщо база даних буде скомпрометована.

Однак важливо пам'ятати, що хешування не є абсолютно надійним. Існують методи атак, як-от колізії (коли два різні вхідні значення дають одне й те саме хеш-значення), які можуть загрожувати цілісності даних. Тому важливо обирати надійні хеш-функції та використовувати додаткові заходи безпеки за необхідності.

2.3 Фізична безпека

Фізична безпека в контексті Інтернету речей (IoT) включає в себе заходи щодо захисту пристроїв від фізичного доступу та несанкціонованого втручання. Важливо забезпечити безпеку пристроїв як на етапі їх виробництва, так і в кінці їх використання. Деякі важливі аспекти фізичної безпеки в IoT включають

Фізичний доступ до пристроїв: якщо фізичний доступ до пристроїв недостатньо обмежений, пристрої IoT можуть піддаватися загрозам. Тільки уповноважені особи повинні мати доступ до пристрою і контролювати доступ.

Фізичний захист пристроїв: пристрої повинні бути захищені від фізичного пошкодження, крадіжки та несанкціонованого доступу. Цього можна досягти шляхом використання захисних корпусів, кріплень або запірних механізмів.

Безпека у виробничому середовищі: важливо забезпечити безпеку пристроїв на всіх етапах виробництва, від проектування до пакування та доставки.

Моніторинг фізичної безпеки: важливо мати системи для моніторингу та повідомлення про будь-який несанкціонований доступ або фізичні атаки на обладнання.

Фізична безпека мережі: на додаток до безпеки самих пристроїв, безпека мережі, через яку пристрої обмінюються даними, також важлива для запобігання фізичним атакам на мережеву інфраструктуру.[10]

Управління конфіденційністю та цілісністю даних: фізична безпека також включає захист конфіденційності та цілісності даних, що зберігаються на

пристроях IoT. Це включає шифрування даних, контроль доступу та аудит діяльності з даними.

Забезпечення безпеки під час утилізації: Коли пристрої Інтернету речей досягають кінця свого життєвого циклу, важливо забезпечити їх безпечну та екологічно чисту утилізацію, щоб запобігти витоку конфіденційних даних і забрудненню навколишнього середовища.

Фізичний захист датчиків і виконавчих механізмів: Для забезпечення надійної роботи датчиків і виконавчих механізмів важливо забезпечити їх захист від впливу навколишнього середовища, такого як вологість, пил і перепади температури.

Фізична безпека розумних будинків і будівель: для забезпечення безпеки розумних будинків і будівель слід враховувати фізичну безпеку пристроїв Інтернету речей, таких як системи контролю доступу, камери відеоспостереження і датчики безпеки.[11]

Навчання користувачів з питань фізичної безпеки: Користувачі пристроїв Інтернету речей повинні бути навчені основам фізичної безпеки, в тому числі тому, як використовувати і зберігати пристрої, а також повинні бути поінформовані про потенційні загрози і способи їх запобігання.

Фізична безпека в промисловості: У промисловому середовищі пристрої Інтернету речей можуть стикатися з більш екстремальними умовами, такими як високі температури, вібрації, хімічні речовини та інші небезпеки. Для забезпечення фізичного захисту слід використовувати спеціалізовані захисні пристрої та корпуси.

Резервне копіювання та відновлення: Щоб забезпечити безпеку даних і пристроїв IoT у разі фізичного пошкодження або катастрофи, важливо мати системи резервного копіювання та відновлення для швидкого відновлення працездатності системи.

Використання безпечних з'єднань: Для забезпечення безпечної передачі даних між пристроями Інтернету речей і хмарними або локальними серверами

слід використовувати безпечні з'єднання, такі як SSL/TLS, які забезпечують шифрування і аутентифікацію.

Фізична безпека в автомобільному секторі В автомобільних системах Інтернету речей важливо забезпечити фізичну безпеку блоків управління і датчиків, щоб запобігти атакам і несанкціонованому доступу, які можуть вплинути на безпеку дорожнього руху.[12]

Проектування безпечних IoT-пристроїв: Виробники пристроїв Інтернету речей повинні враховувати фізичну безпеку при проектуванні і розробці пристроїв, включаючи захист від фізичних атак і забезпечення надійності в різних умовах експлуатації.

Забезпечення фізичної безпеки в IoT вимагає системного підходу і впровадження різних заходів для захисту пристроїв і даних від фізичних загроз і забезпечення надійної і безпечної роботи.

2.4 Захист від атак DoS і DDoS

Атаки типу "відмова в обслуговуванні" (DoS) і "розподілена відмова в обслуговуванні" (DDoS) становлять серйозну загрозу для систем Інтернету речей (IoT), призводячи до відмови в обслуговуванні пристроїв і сервісів, завдаючи шкоди бізнесу та інфраструктурі. Потенціал завдання шкоди бізнесу та інфраструктурі. Для захисту від таких атак необхідно вжити низку запобіжних заходів

Фільтрація трафіку: використання механізмів фільтрації трафіку для відсіювання підозрілого або шкідливого трафіку на ранній стадії. Це знижує навантаження на мережі та пристрої.[13]

Оновлення програмного забезпечення і пристроїв: регулярно оновлюйте програмне забезпечення і прошивку пристроїв, щоб усунути вразливості, які зловмисники можуть використовувати у своїх атаках.

Використання засобів захисту: використовуйте такі засоби захисту, як міжмережеві екрани, системи запобігання вторгненням (IPS) і міжмережеві екрани веб-додатків (WAF), для виявлення і блокування DoS- і DDoS-атак.

Контроль доступу: обмежте доступ до IoT-пристроїв авторизованими користувачами та пристроями. Використовуйте механізми аутентифікації та авторизації для забезпечення безпеки.

Моніторинг і аналіз: моніторинг і аналіз трафіку для швидкого виявлення і реагування на аномальну поведінку. Це дає змогу своєчасно виявляти DoS- і DDoS-атаки.

Розподілені сервіси: розгляньте можливість використання розподілених сервісів і мереж доставки контенту (CDN), щоб розвантажити основні сервери та знизити вразливість до DoS- і DDoS-атак.

Навчання персоналу: навчіть співробітників основ безпеки та захисту від DoS і DDoS, щоб вони могли ефективно реагувати на загрози та запобігати атакам.

Плани аварійного відновлення: розробіть плани аварійного відновлення, які дадуть змогу швидко відновити працездатність системи після успішної DoS- або DDoS-атаки.

Аналіз вразливостей: періодично проводьте аналіз вразливостей системи для виявлення можливих точок входу для атак.[14]

Реагування на інциденти: розробіть процедури реагування на інциденти для швидкого виявлення, аналізу та відновлення після атак.

Захист каналу зв'язку: використовуйте захист каналу зв'язку, такий як шифрування, для захисту від прослуховування та зміни трафіку з боку зловмисників.

Шкідливе ПЗ: перевіряйте пристрої на наявність шкідливого програмного забезпечення, що може бути використане для організації атак.

Бекапи: регулярно створюйте резервні копії даних та конфігурацій для можливості швидкого відновлення після атаки.

Посилення захисту: використовуйте технології, які дозволяють виявляти та захищати від атак, як-от машинне навчання та штучний інтелект.

Ці заходи разом із вищезазначеними допоможуть забезпечити високий рівень захисту для систем Інтернету речей.

Додатково, важливо враховувати специфіку систем Інтернету речей і приділяти увагу наступним аспектам:

Мінімалізація залишкового ризику: Враховуйте, що жодна система захисту не є абсолютною. Плануйте заходи безпеки, щоб мінімізувати залишковий ризик у випадку вразливостей або несподіваних атак.

Безпека від постачальників: Перевіряйте безпеку продуктів і послуг, які ви використовуєте в системі Інтернету речей, і співпрацюйте лише з надійними постачальниками.

Системи моніторингу та реагування: Створіть систему моніторингу, яка виявляє аномальну активність та автоматично реагує на неї, включаючи блокування атак та інші заходи.

Захист інформації: Шифруйте дані, що передаються між пристроями, і зберігайте їх у безпечному місці, щоб уникнути несанкціонованого доступу.

Постійне вдосконалення: Системи захисту Інтернету речей повинні постійно вдосконалюватися, оновлюватися та адаптуватися до нових загроз і вразливостей.

Ці принципи допоможуть забезпечити ефективний захист систем Інтернету речей від різних типів кіберзагроз.

2.5 Захист персональних даних

Захист особистих даних у системах Інтернету речей (IoT) є актуальною проблемою, яка вимагає уваги не лише з технічної, але й з правової точки зору. Законодавство щодо захисту даних може варіюватися в різних країнах, але загальні принципи та вимоги залишаються схожими. Ось деякі ключові правові аспекти захисту в Інтернеті речей:

Законодавство про захист персональних даних: Більшість країн мають законодавство, що регулює збір, обробку та зберігання персональних даних. Це може включати в себе вимоги щодо отримання згоди користувачів на обробку їх даних, а також захист даних від несанкціонованого доступу.

Вимоги до зберігання даних: Деякі країни мають специфічні вимоги щодо зберігання та знищення персональних даних після закінчення їх використання.

Порушення правил захисту даних: За порушення правил захисту даних можуть бути передбачені суворі штрафи та інші санкції. Компанії, які працюють з даними користувачів, повинні бути готові до виконання вимог законодавства.

Міжнародні стандарти та рекомендації: Для полегшення впровадження заходів захисту даних в IoT системах існують міжнародні стандарти та рекомендації, такі як GDPR в Європейському Союзі або CCPA в Каліфорнії, які можуть бути використані як основа для розробки стратегій захисту даних.

Відповідальність за захист даних: Компанії, які збирають та обробляють дані користувачів, несуть відповідальність за їх захист. Це означає, що вони повинні приділяти достатню увагу заходам безпеки та вживати всіх необхідних заходів для запобігання втратам даних або несанкціонованому доступу до них.

Розуміння правових аспектів захисту даних є важливим для розробників, власників бізнесу та користувачів IoT систем, оскільки це дозволяє встановити ефективні стратегії захисту даних та уникнути можливих порушень законодавства.

Права користувачів: Законодавство про захист даних надає користувачам ряд прав, що стосуються їх персональних даних. Ці права можуть включати право на доступ до своїх даних, право на виправлення неправильних даних, право на видалення даних («право бути забутим»), право на обмеження обробки даних та інші. Компанії повинні бути готові виконувати ці права та забезпечувати відповідні механізми для їх здійснення.

Захист даних у сфері медицини та інших чутливих галузях: У галузях, де обробка чутливих даних є необхідною (наприклад, у медичних IoT системах),

законодавство може накладати додаткові вимоги щодо захисту цих даних. Це може включати вимоги до шифрування даних, контролю доступу та інших заходів безпеки.

Конфіденційність та приватність даних: Законодавство про захист даних часто містить вимоги щодо збереження конфіденційності та приватності даних. Це може означати, що компанії повинні забезпечувати захист даних від несанкціонованого доступу та неуповноваженого використання.

Відповідність з місцевим законодавством: З урахуванням того, що законодавство про захист даних може відрізнятися в різних країнах, компанії повинні бути готові до виконання вимог місцевого законодавства, де вони збирають та обробляють дані користувачів.

Повідомлення про порушення безпеки даних: У випадку порушення безпеки даних, компанії можуть бути зобов'язані повідомити про це відповідні органи та користувачів. Це дозволяє швидко реагувати на порушення та зменшує ймовірність серйозних наслідків для користувачів.

У даному контексті можна зробити наступні підсумки:

Загальна оцінка ситуації з правового погляду: У розділі розглянуто основні правові аспекти захисту даних в системах IoT. Визначено, що вони є важливими для розробників, власників бізнесу та користувачів, оскільки дозволяють встановити ефективні стратегії захисту даних та уникнути можливих порушень законодавства.

Необхідність дотримання правових вимог: Зазначено, що дотримання правових вимог щодо захисту даних у системах IoT є ключовим для забезпечення високого рівня довіри користувачів та відповідності законодавству.

Значення міжнародних стандартів та рекомендацій: Підкреслено, що міжнародні стандарти та рекомендації, такі як GDPR в Європейському Союзі або CCPA в Каліфорнії, можуть бути використані як основа для розробки стратегій захисту даних.

Важливість усвідомлення прав користувачів: Зазначено, що усвідомлення прав користувачів щодо їх персональних даних є важливим аспектом захисту даних у системах IoT.

Підкреслення необхідності постійного вдосконалення: Вказано, що у зв'язку з швидким розвитком технологій і змінами в законодавстві компанії повинні постійно вдосконалювати свої підходи до захисту даних.

Ці висновки дають зрозуміти, що захист даних у системах IoT є складним процесом, який потребує уважного вивчення законодавства та впровадження відповідних стратегій безпеки.

3. МЕТОДИКА З БЕЗПЕКИ В IOT

3.1 Потенційні загрози користувачам

Екосистема IoT-технологій являє собою комбінацію різних технологічних зон: зона IoT-пристроїв, мережева зона і хмарна зона. Ці зони можуть бути джерелом цифрових даних. Тобто дані можна збирати з розумного пристрою або датчика з внутрішньої мережі, такого як брандмауер або маршрутизатор, або із зовнішніх мереж (хмара чи додаток. Ці технологічні зони є і об'єктом кримінального інтересу кіберзлочинців.[15]

Під час розслідування комп'ютерних злочинів, пов'язаних із шахрайствами, або комп'ютерних атак, засобами яких так чи інакше були мережеві з'єднання, проводять цифрову експертизу - спеціальне дослідження, що включає аналіз використання мережевих технологій. Залежно від місця зберігання даних у системі IoT експерти у сфері IoT - криміналістики виокремлюють три небезпечні ділянки в ландшафті кіберзагроз: хмара, мережа та пристрій, відповідно виокремлюють хмарну криміналістику, мережеву та криміналістику на рівні пристрою IoT.

Оскільки цінні дані часто зберігаються в хмарі, хмарна інфраструктура є однією з найважливіших цілей для зловмисників. зловмисників. Для проведення традиційної цифрової експертизи експерт-криміналіст спочатку отримує вилучене цифрове обладнання, а потім починає розслідування для вилучення цифрових доказів (цифрових даних, які можна використовувати як доказ вчинення кіберзлочину). вчинення кіберзлочину). Однак якщо дані зберігаються в хмарі, використовується інший сценарій, тому що цифрові докази можуть бути розміщені в хмарних сховищах на різних серверах і їх важко витягти звідти. Крім того, у хмарі обмежений доступ до інфраструктури та інформації про точне місце зберігання даних. Під час розслідування інциденту, що стався в хмарі, постачальник хмарних послуг може запросити

інформацію про ім'я власника даних або місце зберігання відповідних даних. [22]

Слід зазначити, що у хмарних сервісів, які використовують віртуальні машини як сервери, дані можуть зберігатися на цих серверах. Реєстри запису або тимчасові інтернет-файли на серверах можуть бути видалені, якщо вони не синхронізовані з пристроями зберігання, наприклад якщо ці сервери перезапускаються або вимикаються.

Погрози:

1) Умисні дії

Шкідливе ПЗ - програмне забезпечення, призначене для виконання небажаних і несанкціонованих дійв системі без згоди користувача. Це ПЗ може призвести до пошкодження, модифікації або крадіжки інформації. Його небезпека може бути високою.

Експлойт - код, розроблений для використання вразливості з метою отримання доступу до системи. Цю загрозу важко виявити, і в середовищах IoT її небезпека варіюється від високої до критичної, залежно від зачеплених активів

Цільова атака - атака, призначена для конкретної мети, яка проводиться протягом тривалого періоду часу в кілька етапів. Основна мета злочинця - залишатися непоміченим і отримати якомога більше конфіденційних даних, інформації або контролю більше конфіденційних даних, інформації або контролю. Хоча небезпека цієї загрози є середньою, її виявлення - зазвичай дуже складний і тривалий процес.

DDoS-атака - у процесі DDoS-атаки кілька систем атакують одну ціль, щоб навантажити її та призвести до збою. Це можна зробити шляхом створення безлічі з'єднань, переповнення каналу зв'язку або багаторазового повторного відтворення одних і тих самих повідомлень.

Скомпрометований пристрій - Цю загрозу важко виявити, оскільки скомпрометований пристрій важко відрізнити від оригіналу. Ці пристрої

зазвичай мають бекдори і можуть використовуватися для проведення атак на інші системи в навколишньому середовищі.

Втрата конфіденційності - ця загроза небезпечна як втратою конфіденційності користувача, так і впливом стороннього персоналу на елементи мережі.

Модифікація інформації - у цьому разі мета полягає не в пошкодженні пристрою, а в маніпуляції інформацією, щоб викликати хаос або отримати грошовий прибуток

2) Перехоплення інформації

Атака "людина посередині" - активна атака підслуховування, під час якої якій зловмисник передає повідомлення від однієї жертви іншій, щоб змусити їх повірити, що вони розмовляють безпосередньо одна з одною.

Підключення до активної сесії - взяття під контроль активного сеансу зв'язку між двома елементами мережі. Зловмисник може отримати важливу інформацію, зокрема й конфіденційну.

Перехоплення інформації - несанкціоноване перехоплення та модифікація приватної комунікації, наприклад телефонних дзвінків, миттєвих повідомлень, повідомлень електронної пошти.

Мережева розвідка - пасивне отримання внутрішньої інформації про мережу: підключені пристрої, використовуваний протокол, відкриті порти, використовувані служби тощо.

Перехоплення з'єднання - Крадіжка з'єднання для передавання даних, при цьому незаконний хост діє як законний з метою крадіжки, зміни або видалення переданих даних.

3) Вимкнення

Вимкнення живлення - навмисне або випадкове переривання або збій у мережі. Залежно від порушеного сегмента мережі та часу, необхідного для відновлення, небезпека цієї загрози варіюється від високоїдо критичної.

Збій пристрою - збій або вихід з ладу апаратного пристрою.

Збій системи - збій програмних служб або додатків.

Втрата сервісу підтримки - недоступність послуг підтримки, необхідних для правильної роботи інформаційної системи.

4) Технічний бій

Уразливості на програмному рівні - пристрої IoT часто вразливі через слабкі паролі, незмінені паролі, встановлені за замовчуванням, програмних помилок і помилок конфігурації.

Сторонні помилки - помилки в активному елементі мережі, викликані неправильним налаштуванням іншого елемента, який має до нього пряме відношення

5) Катастрофи

Стихійні лиха - повені, сильні вітри, сильні снігопади, зсуви ґрунту та інші стихійні лиха, які можуть пошкодити пристрої фізично.

Аварії в середовищі IoT - аварії в середовищі розгортання IoT-обладнання, що призводять до їхньої непрацездатності.

6) Фізична атака

Модифікація пристрою - модифікація пристрою, внесення змін у пристрій (наприклад, шляхом використання поганої конфігурації портів, використання відкритих конфігурації портів, використання відкритих портів).

Знищення пристрою - псування, крадіжка тощо.

Різні загрози несуть різні потенційні небезпеки, які різняться залежно від сценаріїв використання.

Таблиця 3.1 - Способи захисту від загроз [13].

Способи захисту	Опис
Управління інформаційною безпекою та управління ризиками	Заходи безпеки, що стосуються аналізу ризиків безпеки інформаційної системи, політики, акредитації, показників та аудиту, а також безпеки людських ресурсів
Управління екосистемами	Заходи безпеки щодо картування екосистем і відносин екосистем
Архітектура інформаційної безпеки	Заходи безпеки, що стосуються конфігурації систем, управління активами, поділу систем, фільтрації трафіку та криптографії

Адміністрування інформаційної безпеки	Заходи безпеки щодо адміністративних облікових записів та адміністративних інформаційних систем

Продовження таблиці 3.1

Способи захисту	Опис
Керування ідентифікацією та доступом	Заходи безпеки щодо аутентифікації, ідентифікації та прав доступу
Технічне забезпечення інформаційної безпеки	Заходи безпеки щодо процедур технічного забезпечення IT безпеки та віддаленого доступу
Управління інцидентами комп'ютерної безпеки	Заходи безпеки щодо аналізу та реагування на інциденти безпеки в інформаційній системі, а також звіт про інциденти

3.2 Інформаційна безпека пристроїв IoT з використанням апаратної підтримки

Уразливості в системі Інтернету речей (IoT) призводять до виникнення цілої низки загроз і атак, здатних поставити під загрозу критично важливу інфраструктуру і навіть національну безпеку. Згідно зі щоквартальним звітом McAfee про кіберзагрози, щохвилини з'являється 176 нових загроз. Зовсім недавно було здійснено DDoS-атаку на недорогі IoT-пристрої за допомогою ботнету Mirai, який за чотири місяці заразив понад 2,5 мільйона пристроїв. Очікується, що кількість атак, які використовують уразливості безпеки, зростатиме: до кінця 2020 року більшість із 26 мільярдів IoT-пристроїв не забезпечуватимуть адекватний захист від кібератак; простота веб-інтерфейсу IoT-пристроїв робить їх уразливими для атак із віддалених місць; очікується, що в майбутньому кількість IoT-пристроїв, вразливих для кібератак, зростатиме. (iii) Безпека мережі IoT.

Хоча існують ефективні способи підвищення безпеки мережі пристрою, багато з них не підходять через скромну обчислювальну потужність останнього. Більшість із цих рішень використовують вразливе програмне

забезпечення. Тому важливо враховувати і, за можливості, використовувати апаратну підтримку поряд із програмними засобами захисту IoT-пристроїв для запобігання несподіваним загрозам.

Перш ніж розробляти і впроваджувати рішення щодо захисту від атак на IoT-пристрої, важливо зрозуміти можливості зловмисників і їхні цілі. Фізичний доступ до простих і недорогих пристроїв відкриває можливості для сторонніх атак, таких як апаратні помилки, проникнення троянців і заміна пристроїв на підроблені. Однак у цій статті ми розглянемо тільки програмні та мережеві кібератаки.

Основними цілями кібератак на IoT-пристрої є спотворення вихідних даних (що призводить до несанкціонованої поведінки), порушення поточних процесів (відмова в обслуговуванні) або розкриття конфіденційної інформації, такої як ключі та паролі. Сучасні IoT-пристрої збирають величезну кількість даних про своїх користувачів, що вимагає надійного захисту, якого часто не вистачає.

На рисунку 3.1 показано типи атак, до яких можуть бути схильні пристрої IoT.

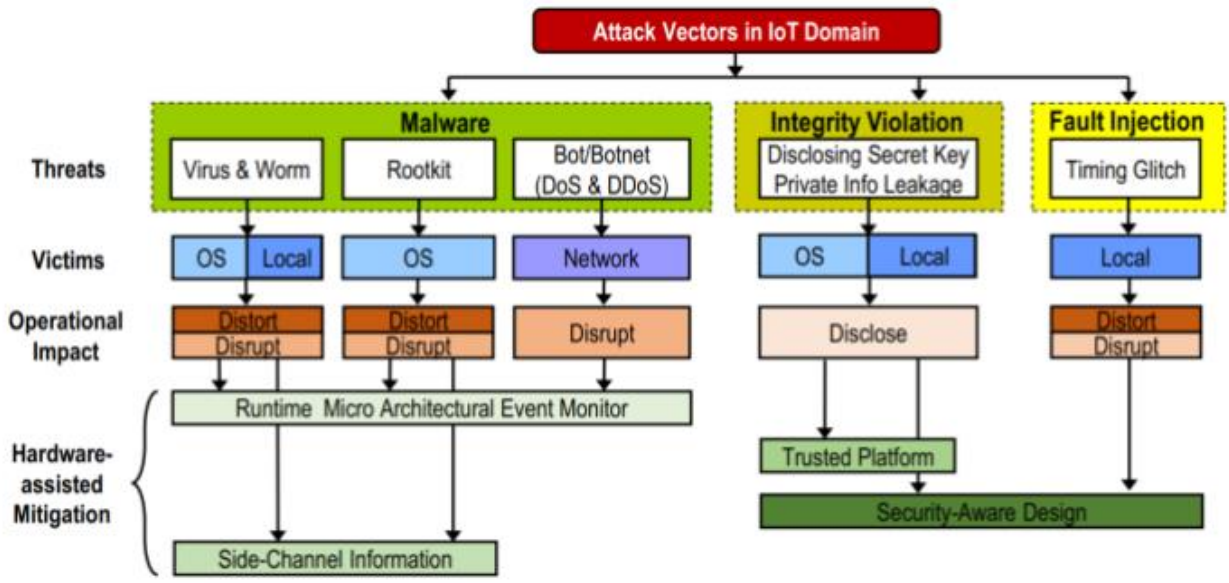

Рисунок 3.1 - Атаки на пристрої IoT та апаратні методи їх запобігання

Сучасні пристрої Інтернету речей вразливі до різних типів шкідливого програмного забезпечення на різних етапах їхньої роботи. Шкідливі програми, такі як віруси, трояни та хробаки, часто націлені на локальні процеси або рівні операційної системи, залежно від складності атаки. Основна мета шкідливого програмного забезпечення - порушити поточну роботу та отримати контроль над пристроями. Руткіти - один з найпоширеніших інструментів, які використовують зловмисники. Руткіти приховують свою присутність, надаючи хакерам постійний привілейований доступ до системи. Пристрої Інтернету речей також можуть стати жертвами атак типу "відмова в обслуговуванні" (DoS) і розподіленої відмови в обслуговуванні (DDoS). Вразливі пристрої можуть діяти як боти, заражаючи інші пристрої та використовуючи мережеві ресурси.

Для зловмисників доступ до приватних ключів, що використовуються для шифрування, або персональних даних, що зберігаються на пристроях IoT, є важливим кроком у компрометації "кореня довіри" системи. Це дозволяє зловмиснику контролювати процес комунікації, викрадати обчислювальні ресурси пристрою і, що найголовніше, отримувати доступ до конфіденційної інформації.

Апаратна безпека є перспективною альтернативою програмним методам захисту. Апаратні системи безпеки використовують спеціальні апаратні модулі і можуть аналізувати мікроархітектуру для виявлення загроз на програмному рівні. Апаратні системи безпеки пропонують широкий спектр рішень для забезпечення безпеки та надійності додатків IoT.

Одним з найважливіших аспектів забезпечення безпечної передачі інформації між пристроями IoT через ненадійні мережі є використання надійних методів управління ключами та обробки даних на апаратному рівні. Для цього широко використовуються модулі довірених платформ (TPM), які зберігають криптографічні ключі та захищають інформацію від несанкціонованого доступу; інші підходи, такі як ARM TrustZone та архітектура Intel Software Guard Extension (SGX), забезпечують безпечне середовище для

критично важливих процесів навіть за наявності потенційно зловмисного програмного забезпечення.

Криптозахищені процесори, як-от AEGIS і Ascend, забезпечують приватну й автентичну обробку даних, захищаючи їх від фізичних атак. Однак вони не забезпечують захист від кіберзагроз, пов'язаних із компрометацією програмного забезпечення. На рисунку 3.2 представлений Модуль TPM від Gigabyte

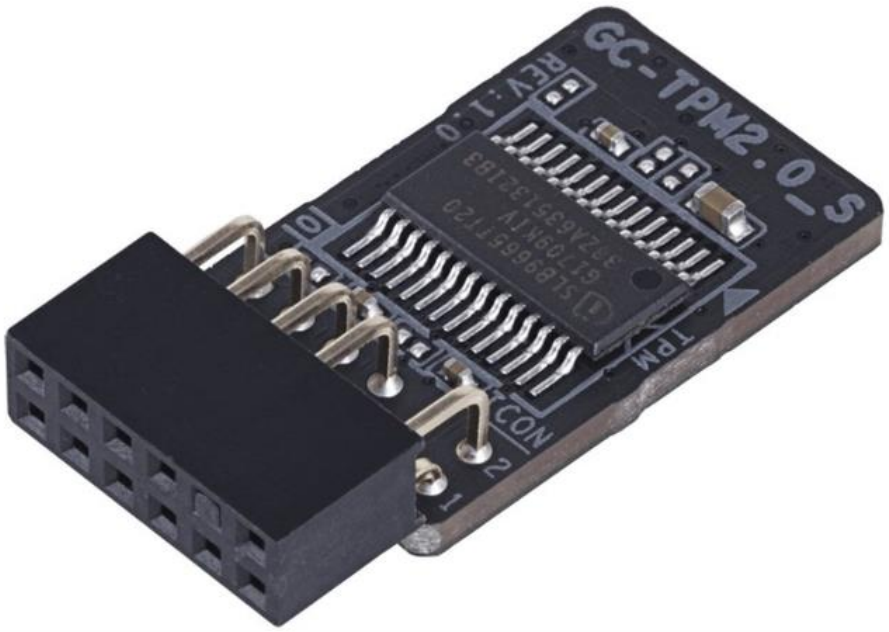

Рисунок 3.2 - Модуль TPM від Gigabyte

Моніторинг подій у мікроархітектурах став важливим інструментом забезпечення безпеки системи, але його ефективність може бути обмежена схожістю між нормальними та шкідливими подіями. Дослідники запропонували використовувати методи машинного навчання для виявлення аномалій з більшою точністю та меншою похибкою. Для цього необхідно вибрати відповідні мікроархітектурні особливості та ефективні методи навчання.

Архітектура системи повинна враховувати, що зміни, внесені в програму зловмисником, зазвичай зберігають її семантику, а деякі підзадачі не можуть

бути принципово змінені. Такий підхід дозволяє використовувати машинне навчання для більш точного виявлення аномалій, таких як шкідливе програмне забезпечення, з меншою кількістю помилкових спрацьовувань.

TPM та інші системи криптографічного захисту забезпечують надійне середовище для чутливих до безпеки додатків, але таке обладнання часто є дорогим, енергоємним і непридатним для легких і недорогих пристроїв Інтернету речей. Крім того, шкідливі програми, такі як віруси, трояни і боти, можуть обходити такі системи і непомітно заражати пристрої, якщо мережа не захищена належним чином. Після зараження шкідливе програмне забезпечення може обійти антивірусне програмне забезпечення на пристрої, що робить його дуже складним для виявлення. У таких випадках можуть допомогти апаратні мікроархітектурні системи моніторингу подій та SIEM (Security Information and Event Management - управління інформацією та подіями безпеки). Вони забезпечують тонку фільтрацію окремих активацій, можуть збирати багатовимірну інформацію і мають швидший час відгуку, ніж програмні антивірусні рішення.

У центрі таких апаратних моніторів знаходиться блок моніторингу продуктивності (Performance Monitoring Unit, PMU), який міститься в сучасних процесорах і SoC. Основне призначення PMU - реєструвати набір мікроархітектурних подій і забезпечувати роботу вбудованого апаратного лічильника продуктивності (HPC), який надає інформацію про продуктивність процесора шляхом підрахунку. Наприклад, один або декілька HPC в PMU можуть підраховувати кількість разів, коли під час виконання програми відбувається заздалегідь визначена подія (дозволена архітектурою), така як промах кешу, що може бути використано для оцінки продуктивності системи, що тестується. PMU архітектури ARM та Intel x 86 можна контролювати за допомогою програмних модулів, таких як інструмент Linux Perf. PMU, які забезпечують зворотний зв'язок в режимі реального часу для діагностики помилок і виявлення вузьких місць в програмному забезпеченні, спочатку були розроблені для моніторингу продуктивності, але вони також добре підходять

для використання в системах SIEM і значно прискорюють обробку інцидентів інформаційної безпеки, допомагаючи виявляти атаки та інші загрози для елементів інфраструктури. Ще одна перевага полягає в тому, що, будучи інтегрованою частиною апаратного забезпечення, PMU працює прозоро для будь-якого програмного забезпечення, що працює на процесорі, і не може бути обманутий зовнішнім шкідливим програмним забезпеченням. Це означає, що сам апаратний монітор не знає про існування процесу. Оскільки шкідливе програмне забезпечення, модифіковані прошивки або руткіти повинні виконувати певні дії, моніторинг подій PMU має потенціал для виявлення такої шкідливої активності.

Розробники з Політехнічної школи Нью-Йоркського університету в Брукліні, штат Нью-Йорк, США, розробили хост-систему під назвою BRAIN (Behaviour-Based Adaptive Intrusion Detection in Networks - адаптивне виявлення вторгнень в мережах на основі поведінки). Запропоновано структуру виявлення DDoS-атак; BRAIN використовує апаратні можливості для моделювання безпечної поведінки та DDoS-атак; для виявлення DDoS-атак BRAIN використовує методи машинного навчання для моделювання поведінки додатків та мережевої статистики. Оскільки кореляція між мережевою статистикою, статистикою додатків і даними HPC є нетривіальною, апаратні події повинні бути обрані з високою точністю. Автори запропонували інтегрований механізм виявлення DDoS (DDoSDE), який відстежує як апаратну, так і мережеву поведінку. Інтерфейс захисту від DDoS (DDoSPI) реагує на виявлені атаки, вносячи IP-адреси до чорного списку (і видаляючи їх, якщо необхідно) на основі динамічних мережевих порогових значень і порогових значень на основі HPC". Це не дозволяє зловмисникам вивчити стандарти та політики безпеки пристрою.

"Однією з головних перешкод при використанні моніторингу мікроархітектурних подій є те, що ті ж самі мікроархітектурні події можуть відбуватися подібним чином (тобто підрахунок частоти і профілів подій) під час поточної роботи, що ускладнює ідентифікацію конкретного програмного

забезпечення як зловмисного. Проблема полягає в тому, що вони можуть не бути чіткою ознакою, щоб позначити певну частину програмного забезпечення як зловмисну. Щоб вирішити цю проблему, дослідники розробили низку методів машинного навчання, які можуть вивчати і розрізняти такі події та ідентифікувати всі типи аномалій з вищою точністю виявлення і меншою кількістю помилок. Існує дві основні вимоги до таких методів

Вибір правильних мікроархітектурних особливостей для збору подій з використанням високопродуктивних обчислень.

Вибір ефективних методів машинного навчання для задач класифікації та регресії.

Щоб задовольнити ці вимоги, необхідно розробити архітектури для аналізу даних про поведінку системи, отриманих від високопродуктивних обчислень. Основні спостереження для побудови такої архітектури полягають у наступному. По-перше, семантика програми не змінюється суттєво, коли зловмисник намагається її реконструювати. По-друге, існують підзадачі, які не можуть бути принципово змінені під час виконання задачі. Виходячи з цих припущень, блоки виявлення аномалій на основі машинного навчання повинні виконувати такі завдання". На рисунку 3.3 представлений алгоритм фаз навчання і спостереження

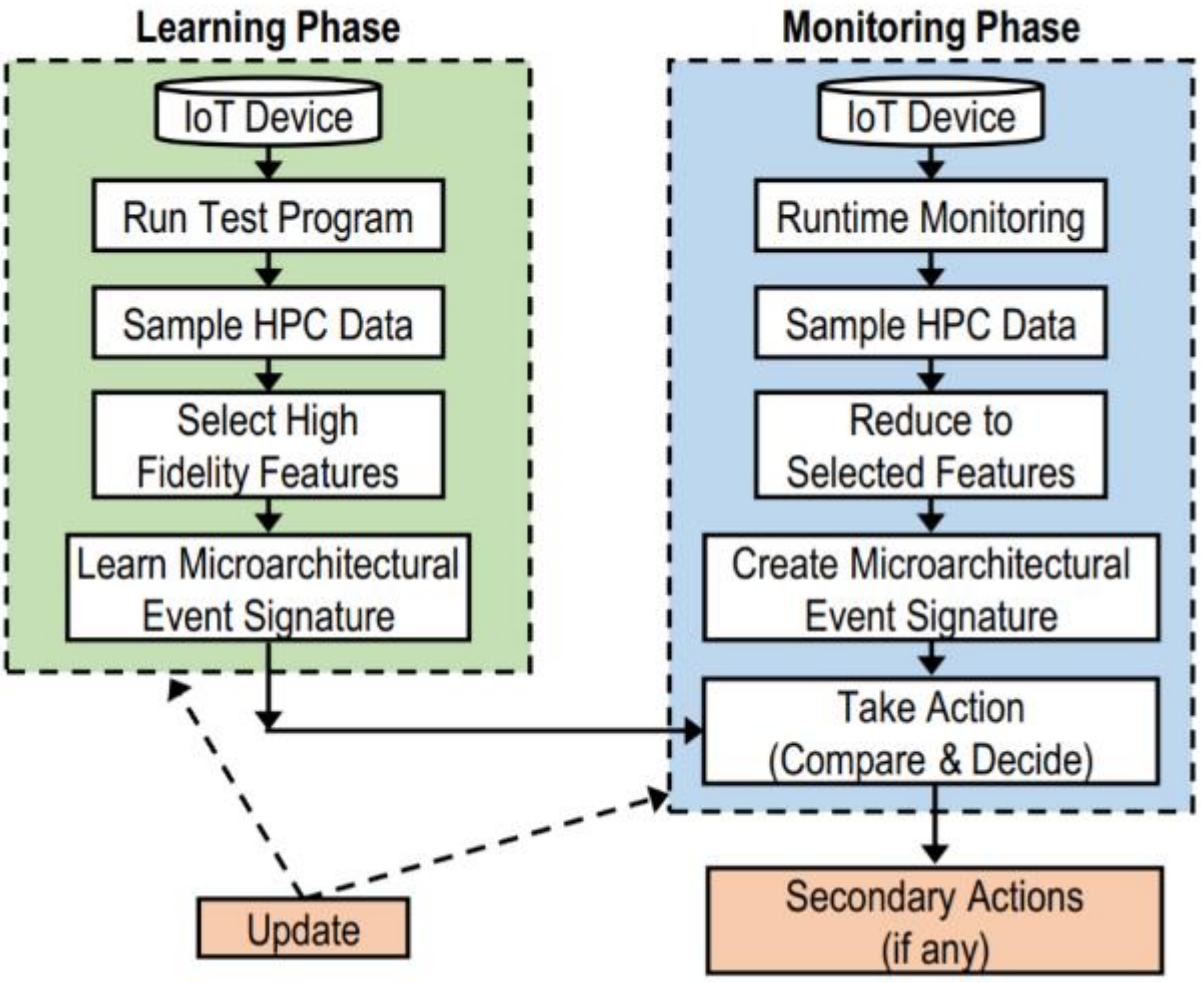

Рисунок 3.3 - Фази навчання і спостереження

Збір даних - на цьому етапі алгоритм вирішує, які дані про мікроархітектурні події слід збирати, і як сенсорний механізм повинен зберігати та обробляти зібрану інформацію.

Аналіз даних - на цьому етапі зловмисна поведінка виявляється за допомогою аналізу даних. Класифікатори машинного навчання використовуються для навчання, тестування та перевірки кореляції між зібраними даними та ненадійною поведінкою.

Прийняття рішень - на цьому етапі вживаються заходи після виявлення загрози. Це може включати в себе надсилання повідомлень користувачам про потенційні загрози, припинення підозрілих транзакцій або вжиття більш критичних заходів, таких як вимкнення всього пристрою для захисту даних і систем.

Модулі виявлення вразливостей регулярно отримують інформацію від цільових модулів, на яких працюють ненадійні програми або шкідливе програмне забезпечення. Архітектура системи повинна дозволяти модулю виявлення працювати з найвищим рівнем привілеїв і незалежно від інших додатків. Крім того, HPC повинен надавати доступ до фізичної пам'яті для зберігання даних та мати ізольовану пам'ять, щоб гарантувати, що модуль виявлення не буде пошкоджений. Обсяг пам'яті, необхідний для зберігання даних машинного навчання, сильно відрізняється для різних типів класифікаторів і вимагає додаткової ємності для зберігання та обчислювальної потужності. Точність використовуваних методів ML і деталізація даних HPC для обраних подій відіграють важливу роль у підвищенні точності та продуктивності всієї системи виявлення.

З метою підвищення ефективності зв'язку HPC+ML був проведений комплексний аналіз з використанням інформації про запущені обчислення HPC. Результати показали, що програмна реалізація різних методів ML на рівні ядра операційної системи працює дуже повільно, обчислюючись мілісекундами, що значно перевищує час виконання шкідливого програмного забезпечення та час вибірки даних на апаратному рівні. Очевидно, що програмних методів класифікації недостатньо для збору даних і виявлення аномалій з високою надійністю. Тому для досягнення низької затримки та високої точності необхідна апаратна реалізація методів машинного навчання. З цією метою експерти з машинного навчання представили рішення, реалізовані на апаратних платформах, таких як Virtex 7, для порівняльного аналізу. Результати показують, що метод OneR є найефективнішим класифікатором доброякісних і шкідливих програм з найвищою точністю і найнижчою обчислювальною потужністю, із загальним показником успішності виявлення близько 81%.

За допомогою дизасемблерів можна відстежувати і реконструювати код, проводити реінжиніринг вихідного коду, здійснювати спільну перевірку апаратного і програмного забезпечення і, що найважливіше, перевіряти цілісність програмного забезпечення, яке працює на пристроях Інтернету речей.

Клас кіберзагроз, відомих як атаки "людина посередині", широко визнаний. Такі атаки використовують інформацію про фізичні процеси, що відбуваються на пристрої, для порушення конфіденційності криптографічної системи, наприклад, шляхом вимірювання часу роботи пристрою, характеристик "напруга-пошкодження", електромагнітного випромінювання тощо. Зібравши достатню кількість статистичних даних, зловмисник може зробити висновок про алгоритм, який використовується в криптосистемі, отримати доступ до секретного ключа або змінити алгоритм після аналізу. Таким чином, зловмисники можуть легко обійти захист і скомпрометувати IoT-пристрої. Але як щодо використання цієї вразливості на благо? Виявити, що IoT-пристрій виконує підозрілі інструкції, так само легко, незалежно від того, чи є він "зараженим", чи ні. Все, що потрібно - це виявити фізичні відхилення в поведінці пристрою. На рисунку 3.4 показана атака за енергоспоживанням на алгоритм RCA.

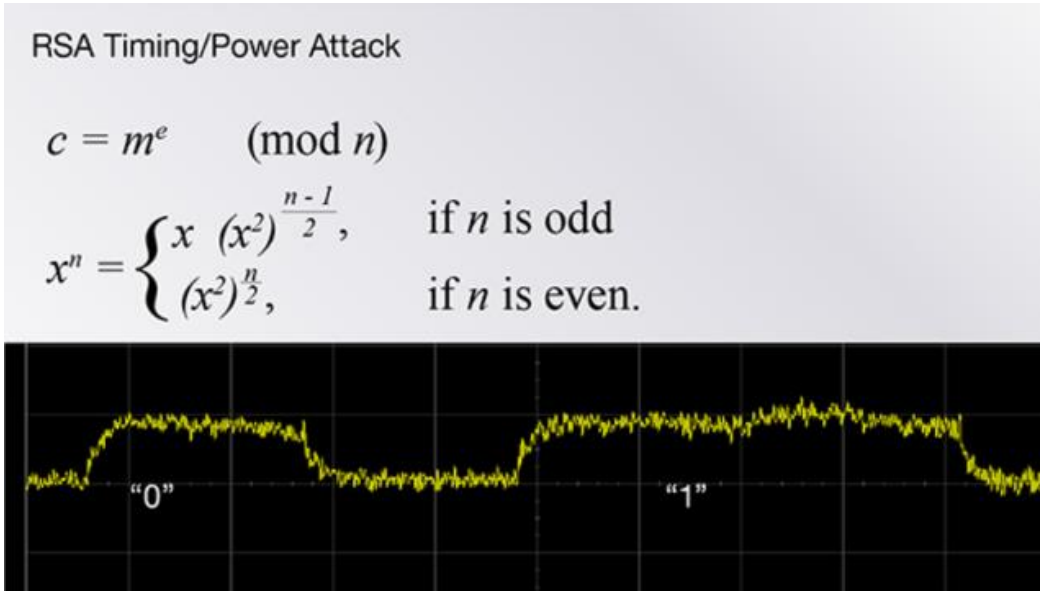

Рисунок 3.4 - Атака за енергоспоживанням на алгоритм RCA

Дослідники показали, що більшість шкідливих програм можна виявити за перериванням живлення через побічні канали. Запропонована ними система відстежує енергоспоживання пристрою і використовує машинне навчання для виявлення можливої аномальної поведінки. Такі методи можуть бути використані для автентифікації та авторизації IoT-пристроїв у ненадійних мережах.

Також було впроваджено систему моніторингу під час виконання, що використовує електромагнітне випромінювання (EMB) як додатковий канал. Ця система використовує керований класифікатор машинного навчання для виявлення аномальної поведінки під час виконання програми, наприклад, вставки шкідливого програмного забезпечення або іншого коду. Цей метод не вимагає ідентифікації характеристик шкідливого програмного забезпечення. Він використовує піки електромагнітного спектра, виміряні під час виконання програми, і порівнює їх із золотими даними, отриманими на етапі навчання. Цей метод може бути придатним для моніторингу безпеки IoT і вбудованих пристроїв, оскільки не вимагає додаткових ресурсів на машині, що моніториться, і не вимагає дротового з'єднання, як при зборі інформації по бічному каналу живлення.

Хоча існують різні методи захисту, такі як TPM, для сучасних пристроїв Інтернету речей, вони мають певні недоліки. Наприклад, таке обладнання вимагає багато енергії і складається з багатьох компонентів, що не підходить для легких пристроїв IoT. Методи виявлення шкідливого програмного забезпечення, засновані на моніторингу продуктивності пристрою, використовують передові методи машинного навчання, але вони забирають багато часу і не завжди ефективні.

Багато методів машинного навчання потребують великих обсягів даних для навчання, що обмежує їхню здатність виявляти нові загрози; системи моніторингу IoT потребують додаткового обладнання і можуть бути скомпрометовані. На цьому тлі розробка апаратних прискорювачів для машинного навчання може підвищити ефективність і безпеку пристроїв IoT.

Незважаючи на те, що існує широкий спектр методів захисту, в тому числі TPM, для підвищення безпеки сучасних пристроїв Інтернету речей, деякі проблеми і обмеження залишаються.

Наприклад, надійне обладнання, таке як TPM, вимагає великої потужності і часто містить багато компонентів, що робить їх непрактичними для легких пристроїв IoT з обмеженими ресурсами. Методи виявлення

шкідливого програмного забезпечення, засновані на моніторингу продуктивності пристрою, використовують передові методи машинного навчання, але можуть бути трудомісткими і неефективними.

Багато методів машинного навчання потребують великих обсягів даних для навчання, що обмежує їхню здатність виявляти нові загрози; системи моніторингу Інтернету речей потребують додаткового обладнання і можуть бути скомпрометовані; а методи машинного навчання не завжди ефективні. На цьому тлі розробка апаратних прискорювачів машинного навчання може підвищити ефективність і безпеку пристроїв Інтернету речей.

3.3 Розробка програми для перевірки вразливостей у системах Інтернету речей (IoT)

Програма, яку ми розробили, призначена для перевірки вразливостей у системах Інтернету речей (IoT) на основі даних про температуру, одержуваних із пристрою. Ось короткий опис цієї програми:

Мета програми: Очікування приходу даних про температуру з пристрою IoT і перевірка їх на вразливості, як-от різка зміна температури або несподівана зміна інтервалу надсилання даних.

Опис роботи програми:

Програма очікує приходу даних кожні 5 секунд.

Якщо дані надходять, їх перевіряють на різку зміну температури. Якщо зміна понад 2 градуси, виводиться повідомлення про злом.

Якщо дані не надходять протягом 5 секунд, програма також виводить повідомлення про злом.

Використання програми: Програму можна використовувати для виявлення потенційних вразливостей у системах IoT, пов'язаних із моніторингом і керуванням температурою, і вжиття заходів для їх виправлення.

Мова програмування: Ми використовували C# для створення цієї програми, але її також можна реалізувати на інших мовах програмування, що підтримують роботу з потоками і мережевими запитами.

Додаткові поліпшення: Для поліпшення програми можна додати функції реєстрації подій, надсилання повідомлень про злом на сервер або електронну пошту, а також більш точну перевірку на вразливості.

Ця програма демонструє базовий підхід до перевірки вразливостей у системах IoT і може бути доповнена і вдосконалена для більш широкого застосування в різних сценаріях.

```
using System;
using System.Threading;

class Program
{
   static void Main()
   {
      Console.WriteLine("Checking for vulnerabilities...");

      double currentTemperature = 0.0;
      bool dataReceived = false;

      while (true)
      {
         Thread.Sleep(5000);


         if (!dataReceived)
         {
            Console.WriteLine("No data received. Possible intrusion!");
            break;
         }

         if (Math.Abs(currentTemperature - 23) > 2)
         {
            Console.WriteLine("Temperature spike detected. Possible intrusion!");
            break;
         }


         dataReceived = false;
      }
```

```
    }
}
```

На рисунку 4.5 показано результати роботи програми.

```
Checking for vulnerabilities...
Received temperature data: 22 degrees Celsius
Received temperature data: 23 degrees Celsius
Received temperature data: 27 degrees Celsius
Temperature spike detected and unusual data sending interval. Possible intrusion!
```

Рисунок 4.5 – Результати роботи програми

Наш алгоритм перевірки вразливостей у системах IoT являє собою простий, але ефективний підхід до виявлення потенційних загроз безпеки. Ось кілька причин, чому його використання може бути корисним:

Простота й ефективність: Наш алгоритм простий у реалізації та не вимагає складних обчислень або налаштування. Він заснований на принципі виявлення різких змін даних, що робить його ефективним для швидкого виявлення можливих загроз безпеки.

Швидка реакція на загрози: Алгоритм працює в реальному часі, що дає змогу виявляти та реагувати на вразливості негайно після їх виникнення. Це дає змогу оперативно вживати заходів щодо захисту системи.

Низьке навантаження на систему: Алгоритм очікує приходу даних кожні 5 секунд, що не створює значного навантаження на систему. Це дає змогу використовувати його навіть на ресурсно-обмежених пристроях IoT.

Універсальність застосування: Наш алгоритм може бути адаптований і використаний у різних сценаріях IoT, де важливо забезпечити безпеку даних і пристроїв.

Попередження про можливий злом: Алгоритм не тільки виявляє вразливості, а й попереджає про небезпеку злому, що дає змогу оперативно вживати заходів щодо захисту системи.

Таким чином, використання нашого алгоритму перевірки вразливостей може допомогти забезпечити безпеку систем IoT завдяки швидкому виявленню та запобіганню можливих загроз.

ВИСНОВКИ

У кваліфікаційній роботі ми зосередили увагу на аналізі та забезпеченні безпеки в системах Інтернету речей (IoT). Розділ 1 роботи присвячений огляду існуючих методів забезпечення безпеки в IoT системах, таких як аутентифікація та авторизація пристроїв, шифрування даних, моніторинг та аналіз безпеки, а також уразливості IoT систем.

У розділі 2 ми детально розглянули методи забезпечення безпеки в системах Інтернету речей, звернувши увагу на методи аналізу та забезпечення безпеки, шифрування даних, фізичну безпеку та захист від атак DoS і DDoS.

Завершальний розділ роботи присвячений методиці забезпечення безпеки в IoT, зокрема інформаційній безпеці пристроїв IoT з використанням апаратної підтримки та розробці програми для перевірки вразливостей у системах Інтернету речей.

Загальною метою роботи було виявлення та аналіз можливих уразливостей в системах IoT та розробка методів їх запобігання та виявлення. В результаті нашої роботи ми розробили програму для перевірки вразливостей у системах Інтернету речей, яка може бути використана для підвищення безпеки та надійності IoT систем.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Neil Chou, Robert Ledesma, Yuka Teraguchi, Dan Boneh, and John C. Mitchell, ―Client-side defence against web-based identity theft,‖ in NDSS. The Internet Society, 2004.

2. Zhao, M., An, B. and Kiekintveld, C., 2016, February. Optimizing personalized email filtering thresholds to mitigate sequential spear phishing attacks. In Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI).

3. G. Tally, R. Thomas, T. V. Vleck, ―Anti-Phishing : Best Practices for Institutions and Consumers‖ Mcafee research technical report, September 2004.

4. Arachchilage, N. A. G. (2015). User-Centred Security Education: A Game Design to Thwart Phishing Attacks. arXiv preprint arXiv:1511.03459.

5. Бойко І.Ф. Застосування методу стохастичних інтегральних зображень для опису відбиття радіолокаційних сигналів від розподілених об’єктів //Вісник НАУ, № 1. – К.: НАУ, 2004. – С. 12 – 17.

6. Wu, M., Miller, R. and Garfinkel, S., 2005. Do Security Toolbars Actually Prevent Phishing Attacks?, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Montreal, Quebec, Canada, 22 - 27April 2006.

7. J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, ―Internet of Things (IoT): A vision, architectural elements, and future directions,‖ Future Generation Computer Systems, vol.29, no.7, pp. 1645–1660, 2013.

8. Методичні рекомендації до підготовки та захисту дипломної роботи (проекту) для студентів галузі знань 1701 «Інформаційна безпека» та спеціальності 125 «Кібербезпека» / Т.В. Бабенко, М.В. Корнєєв, О.В. Кручінін, Д.С. Тимофєєв ; Нац. гірн. ун-т. – Д. : НГУ, 2016. – 44 с.

9. Brian Goetz, Tim Peierls, Joshua Bloch. Java Concurrency in Practice. 2006 432p

10. Білявський Г.О, Бутченко Л.І., Навроцький В.М. Основи екології: Теорія і практикум: Навч. Посібник. Київ, 2002. 352 с.

11. V.A.F. Almeida, A. Bestavros, M. Crovella, and A. Oliveira, "Characterizing Reference Locality in the WWW," Fourth Int. Conf. Parallel Distrib. Inform. Syst. (PDIS), IEEE Comput. Soc, Dec. 1996, Miami Beach, Florida, pp. 92-103.

12. J. M. Almeida, V. A. F. Almeida, and D. Yates, "Measuring the Behavior of a World-Wide Web Server," Proc. Seventh Conf. High Perform. Networking (HPN), IFIP, Apr. 1997, pp. 57-72.

13. . M. Arlitt and C. Williamson, "Web Server Workload Characterization: the Search for Invariants," Proc. 1996 ACM SIGMETRICS Conf. Measurement Comput. Syst., Philadelphia, Pennsylvania, May 1996, pp. 126-137.

14. T. Berners-Lee, R. Cailliau, H. Nielsen, and A. Pecret, "The World Wide Web," Comm. ACM, vol. 37, no. 8, pp. 76-82, Aug. 1994.