Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

# КВАЛІФІКАЦІЙНА РОБОТА

## Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Модель генерації атак і маркування наборів даних про атаки для систем виявлення вторгнень


Виконав:

студент II курсу, групи КСМм-23-1

Пасічнюк Р. Р.
(прізвище, ініціали)


Спеціальність

123 «Комп’ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма

Комп’ютерні системи та мережі
(повна назва освітньої програми)

Керівник: доц. Ільїна І.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ _____ Коваленко А.А.
(підпис) (прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

Спеціальність 123 «Комп’ютерна інженерія»
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп’ютерні системи та мережі
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
“____” _____ 20___ р.

# ЗАВДАННЯ

## НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Пасічнюку Родіону Руслановичу
(прізвище, ім’я, по батькові)

1. Тема роботи Модель генерації атак і маркування наборів даних про атаки для систем виявлення вторгнень

затверджена наказом по університету від “ 22 ” листопада 2024 р. № 1237 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 20 січня 2025 р.

3. Вхідні дані до роботи системи SCADA, фреймворк, DNP3, MITM.

4. Перелік питань, що потрібно опрацювати у роботі

1) аналіз існуючих підходів;
2) розробка універсального фреймворку;
3) генерація наборів даних;
4) реалізування модулів;
5) емпіричні дослідження.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 15 слайдів

6. Консультанти розділів роботи (заповнюється 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 |
| | | | |
| | | | |

## КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів роботи | Термін виконання етапів роботи | Примітка |
|---|---|---|---|
| 1 | Огляд системи критичної інфраструктури | 26.11.24-30.11.24 | |
| 2 | Аналіз існуючих наборів даних для виявлення | 02.12.24-05.12.24 | |
| | вторгнень | | |
| 3 | Тестування систем виявлення вторгнень | 06.12.24-10.12.24 | |
| 4 | Розробка фреймворку для генерації атак та | 11.12.24-21.12.24 | |
| | маркування даних | | |
| 5 | Проведення експериментів | 23.12.24-03.01.25 | |
| 6 | Оформлення матеріалів кваліфікаційної роботи | 04.01.25-07.01.25 | |
| 7 | Подання кваліфікаційної роботи керівникові | 08.01.25-11.01.25 | |
| 8 | Подання кваліфікаційної роботи на рецензу- | 13.01.25-17.01.25 | |
| | вання | | |
| | | | |

Дата видачі завдання 25 листопада 2024 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Ільїна І.В.
(підпис) (посада, прізвище, ініціали)

# РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 58 с., 5 рис., 1 дод., 28 джерел.

SCADA, КІБЕРАТАКИ, ФРЕЙМВОРК, СИСТЕМА ВИЯВЛЕННЯ ВТОРГНЕНЬ, DNP3, MITM, ІН'ЄКЦІЯ, МАСКУВАННЯ, ФЛУДИНГ.

Метою кваліфікаційної роботи є розробка моделі генерації даних кібератаки на системи SCADA, яка забезпечує модульність, гнучкість та адаптивність для створення реалістичних сценаріїв атак. Це дозволяє покращити розробку систем виявлення вторгнень і підвищити рівень безпеки критичної інфраструктури.

У ході виконання кваліфікаційної роботи було проведено аналіз сучасних загроз та вразливостей SCADA-систем, визначено ключові вимоги до генерації тестових даних для кібератак, а також розроблено архітектуру фреймворку, який включає модулі для ін'єкції, маскування, флудингу, атак типу "людина посередині" та відтворення.

Фреймворк дозволяє виконувати атаки, орієнтовані на специфічні протоколи автоматизації, такі як DNP3, MODBUS та інші. Реалізовані модулі забезпечують можливість перехоплення, модифікації та повторного використання легітимного трафіку для симуляції різних типів атак. У роботі особлива увага приділена можливості розвідки мережі та налаштування критичних точок даних, що дозволяє моделювати як локальні, так і віддалені загрози.

# ABSTRACT

Master’s thesis: 58 pages, 5 figures, 1 appendices, 28 sources.

SCADA, CYBERATTACKS, FRAMEWORK, INTRUSION DETECTION SYSTEM, DNP3, MITM, INJECTION, MASQUERADING, FLOODING.

The major goal of this thesis is to develop a model for generating cyberattack data for SCADA systems that ensures modularity, flexibility, and adaptability to create realistic attack scenarios. This facilitates the improvement of intrusion detection systems and enhances the security of critical infrastructure.

In the course of the qualification work, an analysis of modern threats and vulnerabilities of SCADA systems was carried out, key requirements for generating test data for cyberattacks were identified, and the architecture of a framework was developed. This framework includes modules for injection, masquerading, flooding, man-in-the-middle attacks, and replay attacks.

The framework enables the execution of attacks targeting specific automation protocols, such as DNP3, MODBUS, and others. The implemented modules provide capabilities for intercepting, modifying, and reusing legitimate traffic to simulate various types of attacks. Special attention in the work is paid to network reconnaissance and configuring critical data points, enabling the modeling of both local and remote threats.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ ..... 7
ВСТУП ..... 8
1 ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ ..... 10
1.1 Огляд систем критичної інфраструктури ..... 10
1.2 Виявлення вторгнень ..... 17
2 НАБОРИ ДАНИХ ДЛЯ ВИЯВЛЕННЯ ВТОРГНЕНЬ В SCADA ..... 21
2.1 Набори даних для виявлення вторгнень ..... 21
2.2 Генерація наборів даних ..... 24
2.3 Таксономії атак на SCADA ..... 25
2.4 Моделі атак ..... 27
2.5 Кібератаки ..... 28
3 МОДЕЛЬ ГЕНЕРАЦІЇ НАБОРІВ ДАНИХ АТАК SCADA ..... 31
3.1 Фреймворк генерації атак на SCADA ..... 31
3.2 Компоненти системи ..... 35
ВИСНОВКИ ..... 45
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ ..... 46
ДОДАТОК А Графічний матеріал кваліфікаційної роботи ..... 50

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

CRC – (циклічна перевірка надлишковості) (англ., Cyclic Redundancy Check)

DNP3 – (розподілений мережевий протокол, версія 3) (англ., Distributed Network Protocol 3)

IDS – система виявлення вторгнень (англ., Intrusion Detection System)

IP – інтернет-протокол (англ., Internet Protocol)

MAC – контроль доступу до середовища (англ., Media Access Control)

MITM –атака «людина посередині» (англ., Man-In-The-Middle)

PDU – протокольний блок даних (англ., Protocol Data Unit)

TCP/IP – протокол управління передачею/інтернет-протокол (англ., Transmission Control Protocol/Internet Protocol)

SCADA –системи диспетчерського управління та збору даних (англ., Supervisory Control and Data Acquisition)

# ВСТУП

У сучасних умовах стрімкого розвитку цифрових технологій та автоматизації виробничих процесів кіберзагрози стають однією з найважливіших проблем для забезпечення безпеки критичної інфраструктури. Системи SCADA (Supervisory Control and Data Acquisition), які широко використовуються для моніторингу та управління інженерними мережами, об'єктами енергетики, транспорту, водопостачання та інших галузей, є особливо вразливими до кібератак. Компрометація цих систем може призвести до значних економічних збитків, порушення роботи ключових сервісів та загрози життю й здоров'ю людей.

Ефективна протидія кібератакам потребує розробки надійних систем виявлення вторгнень (IDS), які дозволяють своєчасно ідентифікувати та реагувати на загрози. Проте для створення та тестування таких систем необхідні реалістичні набори даних, що відображають реальні сценарії атак на SCADA-системи. На сьогоднішній день існуючі набори даних часто не враховують специфіку SCADA-протоколів, таких як DNP3 та MODBUS, що значно обмежує їхню придатність для досліджень у сфері кібербезпеки.

Ця кваліфікаційна робота присвячена розробці моделі генерації атак і маркування наборів даних про атаки, яка дозволить створювати реалістичні та адаптивні сценарії для систем виявлення вторгнень. Розроблена модель забезпечує можливість відтворення різних типів атак, таких як ін’єкція, маскування, флудинг, атаки типу "людина посередині" (MITM) та відтворення (replay), що підвищує її універсальність і практичну цінність.

Актуальність роботи обумовлена зростанням кількості та складності кібератак на критичну інфраструктуру, а також необхідністю створення ефективних інструментів для моделювання атак і тестування систем захисту.

Мета роботи полягає в розробці моделі генерації атак та маркованих наборів даних для вдосконалення систем виявлення вторгнень у SCADA-системах.

Результати роботи спрямовані на покращення захисту критичної інфраструктури та підвищення ефективності боротьби з кіберзагрозами в умовах постійного розвитку технологій і зростаючої складності атак.

# 1 ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ

## 1.1 Огляд систем критичної інфраструктури

Загальний опис систем критичної інфраструктури дозволяє нам закласти основу для роботи. Спочатку розглянемо ключове обладнання промислової автоматизації, що використовується для виконання критичних процесів. Далі надається огляд технік автоматизованого управління процесами, які регулюють критичні процеси, що виконуються за допомогою автоматизаційного обладнання. І проаналізуємо комунікаційні технології, які забезпечують зв'язок між взаємопов'язаним обладнанням автоматизації. Кожен із цих аспектів критичної інфраструктури є важливим, оскільки вони використовуються для представлення всієї системи критичної інфраструктури. Представлення цих аспектів дозволяє нам обговорити їх можливе використання, що, своєю чергою, дозволяє розробляти корисні набори даних для кібератак на системи SCADA. Використовуємо передачу електроенергії як приклад, однак представлені концепції можуть бути застосовані й до інших галузей, таких як виробництво та очищення води. Інфраструктура передачі електроенергії містить системи управління та компоненти автоматизації, які забезпечують завершення промислового процесу.

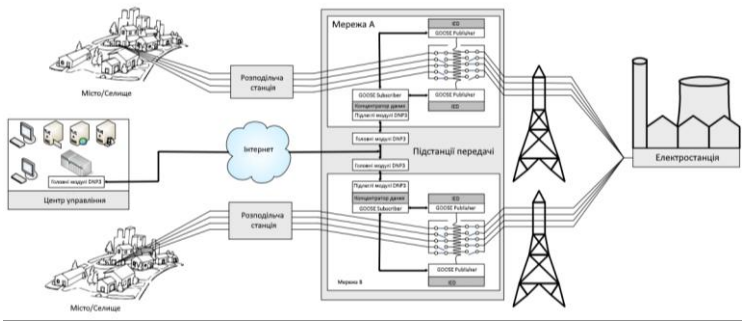


Рисунок 1.1 – Абстрактний вигляд мережі передачі електроенергії

Системи управління можуть бути географічно масштабованими системами або обмеженими виробничою мережею на заводі. Системи управління можуть бути інтегровані в IT-інфраструктури через Ethernet або бездротові мережі, де системи управління можуть централізовано адмініструватися інженерами з управління [1]. На рисунку 1.1 показано приклад мережі критичної інфраструктури передачі електроенергії. Деякі концепції цієї роботи базуватимуться на системах SCADA у мережах передачі, у критичній інфраструктурі.

Типові системи SCADA складаються з компонентів Master і Slave, які взаємодіють із сенсорним обладнанням. Описуючи роль SCADA Master і Slave, їх можна розглядати просто як клієнт і сервер [1]. У більшості випадків SCADA Slave можна розглядати як сервер, оскільки він має здатність збирати інформацію від сенсорних компонентів або виконавчих механізмів і надавати зворотний зв'язок Master. SCADA Master можна розглядати як клієнт через його здатність запитувати інформацію у Slave, що має схожі характеристики з клієнтською програмою [1]. У великих SCADA-мережах може існувати ієрархія Master і Slave, у якій присутні підмайстри (Sub-masters). Підмайстри поєднують характеристики як Master, так і Slave, оскільки вони можуть отримувати запити від Master вищого рівня для отримання інформації від кількох Slave нижчого рівня.

Такі запити виконуються за допомогою комунікаційних протоколів SCADA, таких як MODBUS, S7comm або DNP3, які можуть використовувати інтернет-протоколи або послідовний зв'язок [1]. Комунікація від підстанції потім повертається до центру управління, який зазвичай знаходиться у віддаленому місці.The majority of critical systems rely on the operations of many smaller systems to complete an entire critical process. In this section we identify some technologies used to manage, control, and provide automation to critical infrastructure systems.

На рисунку 1.2 можна побачити таке обладнання, яке використовується в системах управління: інтелектуальні електронні пристрої (IEDs),

програмовані логічні контролери (PLCs), інтерфейси "людина-машина" (HMIs) та робочі станції. Розуміння функціональності обладнання, розглянутого в цьому розділі, є критично важливим, оскільки це дозволяє ідентифікувати різні методи атак.

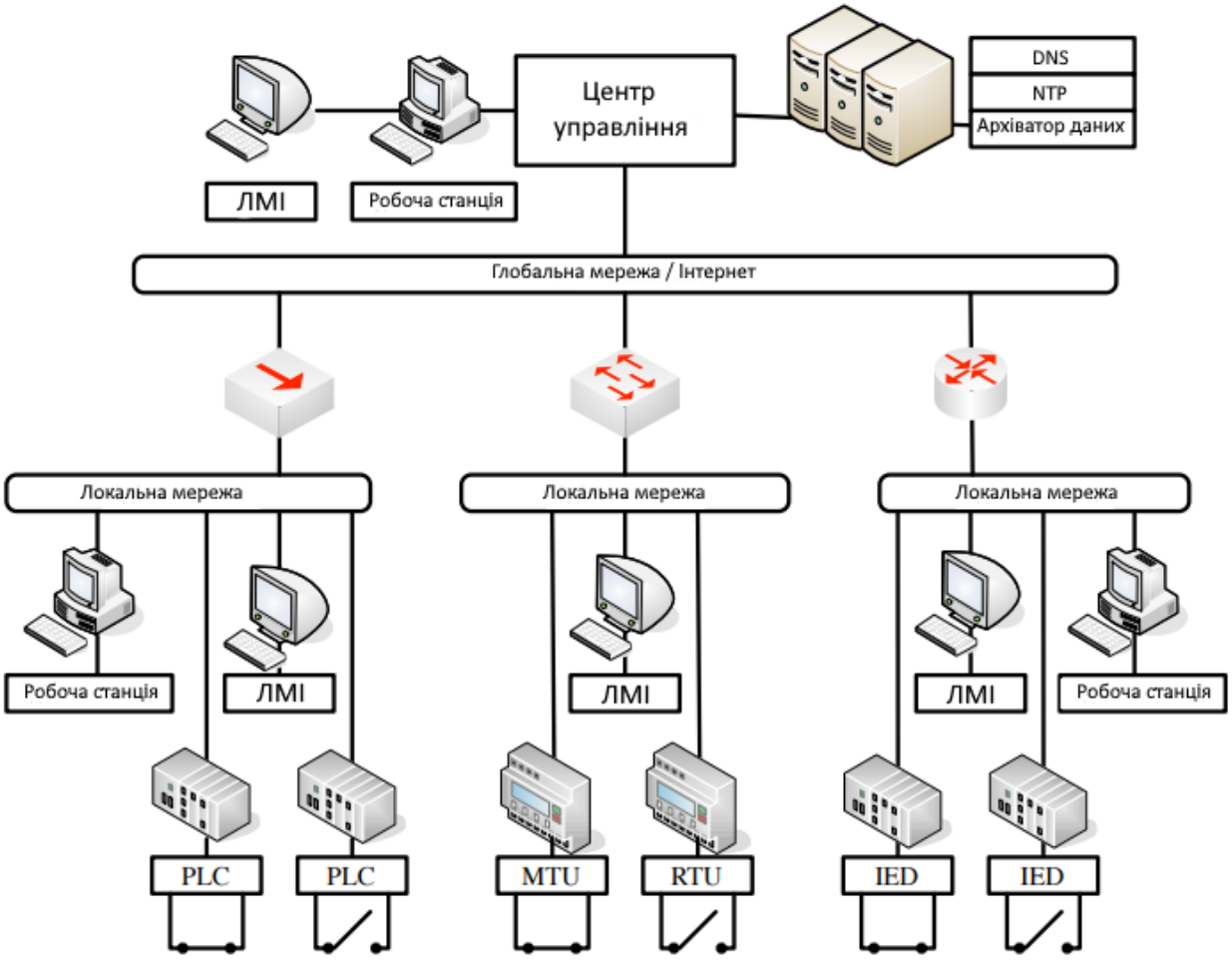


Рисунок 1.2 – Приклад загальної мережі критичної інфраструктури

Для кращого розуміння концепції створення, генерування та маркування наборів даних атак на SCADA, у цьому розділі також надається огляд критичної інфраструктури на базі SCADA: зокрема, інфраструктури критичної передачі електроенергії, SCADA-протоколів, які використовуються для створення набору даних, та огляд пов'язаних робіт, що включають сучасні методи створення, генерування та маркування наборів даних мережевих атак.

Мікропроцесорні контролери впроваджуються в системи управління для забезпечення процесу автоматизації. ПЛК (програмовані логічні контролери) можна знайти на нижчому рівні систем управління, оскільки вони забезпечують операції вводу та виводу для фізичного обладнання, такого як двигуни та виконавчі механізми, для виконання фізичної операції. ПЛК є одним із ключових компонентів автоматизації в системах управління, оскільки цей пристрій здатний забезпечити автоматизаційні можливості для управління та завершення промислового процесу [2].

Інтелектуальні електронні пристрої (IEDs) є автоматизаційними пристроями, які використовуються в мережах передачі електроенергії та здатні виконувати одну або кілька функцій для захисту, вимірювання, запису несправностей та управління [3]. Подібно до ПЛК, основними компонентами IED є блок обробки сигналів, мікропроцесор із пристроями вводу та виводу, а також комунікаційний інтерфейс [3]. IED вважаються одним із більш досконалих видів обладнання завдяки їх використанню для підтримки та управління передачею електроенергії [3]. Оскільки функціонування IED є критично важливим, зловмисники можуть націлюватися на таке обладнання для маніпуляцій або пошкодження критичних процесів .

Такі мікропроцесорні контролери також містять об'єкти пам'яті, такі як регістри та архіви, де зберігається набір значень вводу та виводу. Ці значення містять інформацію про операції системи, набори даних, ідентифікатори пристроїв та стан процесів управління, які контролюються контролером, подібно до концепції бази даних. Ці значення можуть оновлюватися або зчитуватися через комунікаційні протоколи за допомогою програмного забезпечення для управління процесами, реалізованого у вигляді мови програмування або логіки, яка виконується на пристрої.

Інтерфейс "людина-машина" (HMI) може бути додатком з графічним інтерфейсом користувача (GUI), який використовується на настільному комп'ютері, інтерактивному сенсорному екрані або консолі. HMI використовується для відображення даних системи, подачі сигналів тривоги

та демонстрації тенденцій для сенсорного обладнання та виконавчих механізмів. Програми HMI можуть бути веб- або настільними додатками, які використовуються для взаємодії з автоматизованими активами. HMI збирає інформацію та дані, отримані від ПЛК і IED. HMI забезпечує інтерактивний інтерфейс для техніків або інженерів, дозволяючи їм контролювати або взаємодіяти з обладнанням автоматизації. HMI може бути розташований у централізованому місці, наприклад, у будівлі управління на об'єкті або в диспетчерській центрі. Більшість інформації, отриманої через HMI, є критично важливою і використовується для забезпечення коректної роботи критичної системи. Оскільки HMI використовується для управління процесами та інтегрований з активами автоматизації, він є ідеальною ціллю для кібератак.

Робоча станція є типовим настільним комп'ютером, який використовується інженером або техніком на місці для налаштування обладнання автоматизації. Робоча станція використовується для завантаження конфігурацій до ПЛК і IED через спеціалізовані послідовні або Ethernet-з'єднання. Крім того, робоча станція також використовується для розробки та завантаження логіки автоматизації для ПЛК [4]. Основне поточне використання робочої станції полягає у виконанні запитів та управлінні активами автоматизації для аналізу тривог. Оскільки робоча станція використовується для розробки та впровадження критичних процесів, вона може стати ціллю для зловмисників, які намагаються зібрати інформацію, пошкодити або маніпулювати критичними системними процесами [4].

Було розглянуто компоненти автоматизації, які можна знайти у системах критичної інфраструктури, та їхню роль у таких системах. Далі визначаємо деякі загальні методи управління процесами, реалізовані в мережах автоматизації. Важливо ідентифікувати ці широко використовувані методи управління.

Для автоматизації обладнання, такого як ПЛК, потрібна графічна мова програмування, відома як (ladder logic) або код драбини. Логіка драбини забезпечує інженерам і технікам віртуальне представлення логічних схем у вигляді мови програмування, заснованої на графічних електричних схемах та діаграмах логічних релейних компонентів [4]. Сама мова дозволяє ПЛК взаємодіяти з компонентами вводу та виводу, які доступні для управління та моніторингу критичного процесу. Постачальники ПЛК зазвичай пропонують інтегровані середовища розробки (IDE), які забезпечують платформу для розробки логіки драбини для критичного процесу. Деякі з поширених IDE для розробки логіки драбини включають Step7, CX-Programmer і RSLogix. Такі середовища розробки взаємодіють із ПЛК, запитуючи інформацію про можливі значення вводу та виводу. Це дозволяє розробнику створювати процес автоматизації, використовуючи значення вводу та виводу, доступні на ПЛК. Програма, розроблена з використанням логіки драбини, може бути завантажена IED у відповідний ПЛК через Ethernet або послідовне з'єднання. Далі розглянемо огляд трьох загальних технік управління, які можуть бути застосовані до обладнання автоматизації. Ці техніки управління включають замкнутий контур, розімкнутий контур і адаптивне управління.

Управління в замкнутому контурі, також відоме як управління зворотним зв'язком, використовується для регулювання характеристик автоматизованої системи. Це досягається, коли виміряне значення виходу системи може бути відрегульоване до бажаного значення через керуючий вплив. Цей вплив є тим, що впливає на виміряне значення виходу системи. Якщо в системі існує збурення, це також може вплинути на виміряний вихід. Зворотний зв'язок від виміряного виходу коригує керуючий вплив у наступному циклі виконання. Ця корекція дозволяє системі забезпечувати необхідний виміряний вихід для наступного циклу.

Приклад, представлений на рисунку 1.3, зображує систему управління з одним входом і одним виходом (SISO). Ця конкретна система управління забезпечує один керуючий вплив і один виміряний вихід. Елементи, показані

на рисунку 1.3, є необхідними для забезпечення управління зворотним зв'язком. Контролер використовується для визначення налаштування керуючого впливу (установча точка), необхідного для досягнення опорного входу. Це досягається шляхом обчислення значень, наданих керуючим впливом, із поточними та попередніми значеннями помилки управління. Перетворювач (датчик) використовується для підготовки виміряного виходу для порівняння з опорним виходом, який потім обробляється контролем помилки.

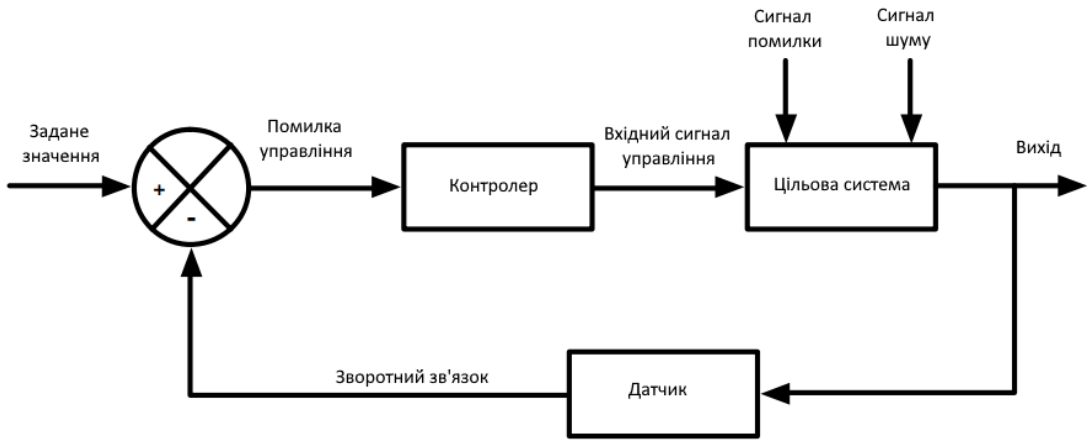


Рисунок 1.3 – Замкнутий цикл управління

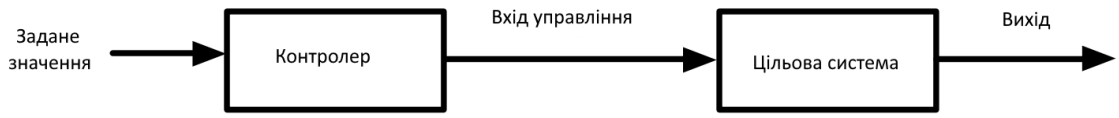


Рисунок 1.4 – Управління в розімкнутому циклі

Система з розімкнутим контуром, зображена на рисунку 1.4, є типом системи безперервного управління, яку також називають системою управління без зворотного зв'язку. Вихід системи з розімкнутим контуром не впливає на керуючі дії, пов'язані з вхідним сигналом, на відміну від системи з замкнутим контуром, описаної раніше. Очікується, що виміряний вихід,

створений цільовою системою, точно відповідатиме вхідному керуючому сигналу [5].

Системи з розімкнутим контуром не можуть вносити корективи у вхідний сигнал управління, на відміну від системи з замкнутим контуром, яка здатна до самокорекції. У таких системах немає зворотного зв'язку, щоб виправляти помилки або адаптуватися до збурень, внесених у цільову систему [5].

Деякі з основних характеристик системи з розімкнутим контуром включають відсутність порівняння між бажаними значеннями та фактичними значеннями. Крім того, немає саморегуляції або контрольної дії на виміряний вихід. Кожен із вхідних сигналів управління передбачає надійний операційний процес. Будь-які зміни або збурення, внесені в систему, теоретично не впливають безпосередньо на її вихід [5].

## 1.2 Виявлення вторгнень

Виявлення вторгнень є концепцією, яка включає передбачення або ідентифікацію аномальної поведінки в системах або мережах. Для цього система виявлення вторгнень (IDS) повинна пасивно моніторити мережний трафік, не втручаючись у повідомлення, що передаються через мережу. Існує два методи виявлення вторгнень: засновані на сигнатурах і засновані на аномаліях.

На сьогодні існує багато технік для впровадження таких систем у мережах зв'язку на основі Ethernet, однак усе ще потрібні значні дослідження для розробки ефективних IDS для систем управління виробничими процесами (ICS) та критичної інфраструктури. Це також потребує від IDS пасивного моніторингу мережевого трафіку без втручання у повідомлення, що передаються мережею.

Два основні методи виявлення вторгнень – це методи, засновані на сигнатурах, і методи, засновані на аномаліях [6]. Наприклад, базове

виявлення вторгнень може включати ситуацію, коли несанкціонований пристрій відправляє повідомлення до Slave PLC, маскуючись під Master. Якщо IP-адреса такого пристрою не зареєстрована в IDS для отримання доступу, IDS позначить будь-яке повідомлення від цього пристрою як вторгнення.

Компоненти, які використовуються в критичній інфраструктурі, розгортаються для підтримки промислових процесів протягом багатьох років, а іноді й десятиліть. Крім того, такі компоненти, як ПЛК (програмовані логічні контролери), є простими, оскільки вони виконують лише базові програмні програми для реалізації процесу управління. Більшість цих компонентів не забезпечують додаткових програмних функцій для безпеки .

Виявлення вторгнень вважається економічно ефективною технікою для виявлення атак на мережі критичної інфраструктури. Використання IDS не потребує дорогої заміни існуючих, довгострокових, застарілих компонентів, розгорнутих у мережах SCADA. Фінансові витрати на ліцензування SCADA та робочу силу, необхідну для заміни такого обладнання, є економічно недоцільними для комунальних і промислових підприємств.

Комунальні підприємства надають перевагу виявленню вторгнень над використанням криптографії. Це пов’язано з витратами часу для обробки критичних повідомлень, труднощами з методами обміну ключами та витратами на обчислені та мережні ресурси при використанні криптографії на ПЛК. Для впровадження таких криптографічних систем критична інфраструктура може потребувати значного простою. Такі простої не є ідеальними для промислових операцій комунальних підприємств.

IDS може бути реалізована як додатковий пристрій, який виявляє аномалії та сигнатури відомих атак.

На сьогодні існує два методи виявлення вторгнень у комунікаційних мережах: засноване на сигнатурах та засноване на аномаліях.

Перший підхід – це виявлення, засноване на сигнатурах, яке передбачає, що IDS зіставляє мережний трафік із відомими шаблонами атак,

які називають сигнатурами атак [7]. Підхід, заснований на сигнатурах, вимагає ретельно розроблених сигнатур вторгнень. Ці сигнатури є набором правил або інструкцій у вигляді програми або конфігурації, які використовуються для ідентифікації шаблонів атак або порушень операційних процедур мережі. Теоретично виявлення на основі сигнатур може забезпечити високу точність виявлення разом із низьким рівнем помилкових тривог.

Другий підхід – це виявлення, засноване на аномаліях. Цей метод вимагає, щоб IDS розуміла нормальну поведінку системи управління. Це можна здійснити шляхом порівняння усього мережевого трафіку системи із заздалегідь визначеним набором подій. Якщо трафік не відповідає цим подіям, спрацьовує тривога. Виявлення, засноване на аномаліях, не потребує сигнатур атак, що є ідеальним підходом, якщо сигнатури для нових атак відсутні.

Системи, засновані на аномаліях, потребують навчання для виявлення або прогнозування будь-якої аномальної чи шкідливої поведінки. Для розробки таких систем використовується підхід, заснований на реалізації механізму машинного навчання. Ця система вимагає видобування характеристик із мережі для визначення прийнятної та шкідливої поведінки. Цей підхід базується на техніці під назвою навчання, яка потребує існування повного набору даних мережного трафіку, що містить аномальну та шкідливу поведінку.

Snort є широко використовуваною системою IDS із відкритим кодом, заснованою на сигнатурах. Ця IDS була розширена та адаптована до закритих програм. Snort містить різні модулі, які дозволяють виявляти відомі атаки на IT-системи. Завдяки успіху Snort у використанні в IT-системах, її також було адаптовано для використання у середовищах систем управління та автоматизації. У рамках проєкту Digital Bond Quickdraw були випущені кілька препроцесорів IDS для широко поширених, стандартних галузевих та власних протоколів SCADA. Ці препроцесори протоколів SCADA

використовуються Snort для інтерпретації даних протоколів SCADA, включаючи MODBUS TCP, Ethernet Industrial Protocol (EtherNet/IP), S7comm та DNP3.

Крім препроцесорів протоколів SCADA, Digital Bond випустила набори правил для кожного підтримуваного протоколу, які використовуються з IDS Snort, а також набір сигнатур для вразливостей SCADA, що відомі для систем управління.

# 2 НАБОРИ ДАНИХ ДЛЯ ВИЯВЛЕННЯ ВТОРГНЕНЬ В SCADA

## 2.1 Набори даних для виявлення вторгнень

Набори даних мережних атак є цінним ресурсом для досліджень у сфері виявлення вторгнень. Це пояснюється тим, що такі набори даних допомагають дослідникам оцінювати та перевіряти нові методи виявлення вторгнень. Набори даних атак містять інформацію про структуру та шаблони однієї або декількох атак.

При розробці наборів даних слід враховувати різні концепції, такі як тип трафіку атак та нормального трафіку, які потрібно генерувати, спосіб виконання атак, а також вимоги до тестових стендів для створення наборів даних.

Набори даних атак на SCADA створювати складно через відсутність реалістичних тестових стендів SCADA-мереж, які б відображали реальні промислові процеси. Крім того, створення таких наборів даних у реальній критичній інфраструктурі є складним завданням через питання конфіденційності та ризик пошкодження компонентів системи управління.

Два з найбільш широко використовуваних наборів даних атак, що застосовуються дослідницькою спільнотою, це DARPA98 і KDDcup99. Ці набори даних містять мережний трафік, що імітує трафік на середньому за розмірами військовому об'єкті ВПС США. Дослідники виявили низку недоліків у підході, використаному для створення та вставки даних атак у набори. Серед основних проблем було виявлено відсутність схожості з реальним мережним трафіком. DARPA-99 і KDD-99 були визнані неточними, оскільки синтетичний трафік атак був доданий до набору даних після його збору. Вставка синтетичного трафіку в існуючий набір даних призводить до відсутності реалістичних відповідей з боку цільових пристроїв, що знижує достовірність методів виявлення, які оцінюються за допомогою цих наборів.

Автори [8] надали керівництво щодо оцінювання та створення того, що вважається "нормальною поведінкою" в корпоративних IT-мережах, що є важливим для виявлення аномалій, оскільки IDS має розуміти, як виглядає нормальна поведінка, щоб виявляти незвичайну. Процес, запропонований Авторами [8], також передбачав створення середовища, яке не вимагає анонімності під час збору даних, що дозволяло набору виглядати більш реалістичною IT-інфраструктурою. Ця анонімність передбачає забезпечення конфіденційності конфігурацій або сервісів, адресації чи розташування сервісів і відкритого тексту, який передається мережею. Для процесу створення атак автори використовували програмний фреймворк Metasploit, що складається з різних модулів для тестування проникнення в мережі. Автори використовували сценарії атак на основі модулів Metasploit, доступних для вразливих конфігурацій у їхньому тестовому середовищі. Також надали набір керівних принципів, які є корисними для створення нормального мережевого трафіку SCADA у наборах даних.

Автори [9] представили набір даних атак, призначений для заміни застарілого набору даних KDDcup99, з адаптацією для бездротових технологій. Застосували методи, подібні до методів представлених в [8], оскільки вони також використовували фреймворк Metasploit для здійснення сучасних атак на мережеві бази даних у тестовому середовищі. Ці сучасні атаки включають відомі експлойти системи, створені спільнотою тестувальників проникнення. Атаки охоплювали різні мережеві сервіси, від атак на SSH, FTP до вебсервісів. У роботах [9] та [8] не було загальних вимог для створення аномального або шкідливого трафіку. Автори зосереджувалися лише на використанні сучасних атак, доступних для певних IT-додатків, програмних сервісів або операційних систем. Крім того, ці роботи не надають методів створення наборів даних атак для SCADA.

Один із найбільш релевантних наборів даних, що містять атаки на SCADA, був представлений Моррісом і Гао [10], які створили чотири набори даних атак для протоколу SCADA MODBUS. Морріс і Гао [10] згенерували

28 атак на тестовому стенді серійної мережі MODBUS. Вони використовували спеціально створені програми на C для реалізації атак через серійний зв'язок. Серійний трафік мережі MODBUS був зафіксований та записаний із використанням методу "bump-in-the-wire", при якому атаки також впроваджувалися в серійну мережу.

Одним із обмежень цього підходу є те, що набір даних базується на серійному зв'язку. Хоча цей набір даних дійсно включає атаки, спрямовані на важливий протокол SCADA, сучасні SCADA-технології переважно використовують Ethernet. Крім того, набір даних не містить фоновий трафік.

Автори [10] класифікували атаки на чотири категорії: розвідка, впровадження відповідей, впровадження команд та відмова в обслуговуванні. Вони надали детальний опис кожної атаки та її реалізації. Атаки, представлені в наборі даних [10], були створені за допомогою програм на C та C++, оскільки існує обмежена кількість програмних інструментів, спеціально спрямованих на серійні мережі MODBUS.

Однак у роботі [10] не було представлено чіткого фреймворку для генерації атак або їх маркування. Представлені вимоги до генерації атак адаптують вимоги до впровадження, представлені Моррісом і Гао [10], що дозволяє створити точний фреймворк для генерації наборів даних атак.

В роботі [11] зібрали різні типи мережевих наборів даних для дослідження ефективності алгоритмів виявлення аномалій на основі n-грам аналізу. Інші автори використовували набори даних мереж ICS, зібрані на двох різних об'єктах водоочистки та розподілу води, у своїх дослідженнях.

Автори [12] зібрали реальні мережеві набори даних ICS, що включають трафік DNP3 між автоматизованими підстанціями SCADA (SA) та трафік IEC 61850 (MMS, GOOSE) всередині внутрішньої мережі SA в енергетичній мережі Південної Кореї.

Автори [13] створили марковані набори даних для нормального та шкідливого трафіку Modbus у середовищі SCADA sandbox. Sandbox є віртуалізованим середовищем, що імітує систему SCADA та розподіл

електроенергії. Їхній набір даних містить лише чотири відомі атаки, реалізовані за допомогою Metasploit.

## 2.2 Генерація наборів даних

Існують різноманітні техніки, які використовуються дослідниками для генерації трафіку SCADA. Ці техніки включають використання спеціально створених програмних інструментів або існуючих програмних реалізацій. Ці інструменти або методи можуть бути розширені не лише для створення легітимного трафіку, але й для генерування шкідливого трафіку. Серед них використовуються клієнт-серверні програми, програмні реалізації відкритого коду, інструменти для фаззингу, TCP replay та бібліотеки маніпуляції з пакетами, такі як Scapy.

Прикладами інструментів для фаззингу є ті, які використовуються для тестування на проникнення. Інструменти для фаззингу тестують коректність реалізації протоколу. Автори [14] провели дослідження реалізацій протоколу MODBUS/TCP, розробивши та оцінивши програму для фаззингу MODBUS TCP (MTF). Цей інструмент використовувався для тестування реалізацій протоколів MODBUS, щоб забезпечити їхню цілісність у мережі SCADA. Автори [14] запропонували вимоги до атак типу "flooding" та "spoofing" для реалізації фаззингового інструменту MODBUS, які, на мою думку, можуть бути адаптовані для інших реалізацій інструментів фаззингу для генерації атак.

Scapy є провідним інструментом для генерації мережевого трафіку, який широко використовується дослідниками для створення трафіку. Автори [15] створили реалізацію GOOSE, протоколу SCADA, щоб моделювати архітектури мереж GOOSE на тестових стендах підстанцій. Згенерований трафік використовувався для аналізу продуктивності та мережевих потоків у тестовому середовищі і зрештою міг бути застосований для тестування реальної інфраструктури.

Автори [16] представили методологію для симуляції трафіку MMS, використовуючи алгоритм генерації трафіку, реалізований за допомогою інструменту маніпуляції пакетами Scapy на Python. Такі реалізації на Python використовуються для стимуляції тестового середовища легітимним мережевим трафіком SCADA. Однак ці методи також застосовуються для спуфінгу протоколів, коли зловмисник імітує характеристики певного комунікаційного протоколу. Одним із обмежень роботи [16] є відсутність робочого стеку протоколу MMS, що обмежує можливість реалізації складних операцій, які можуть бути потрібними в цільовому протоколі для інших складних протоколів SCADA.

На основі пов'язаних робіт було прийнято ідею про створення клієнтів і сервісів, а також використання маніпуляцій із пакетами для генерації даних атак. Ці методи обрано через їхню здатність поєднуватися для створення великих обсягів даних атак. Після визначення основних особливостей наборів даних мережевих атак і методів їх генерації змогли визначити атаки SCADA, які могли б стимулювати тестові стенди.

Далі досліджуємо таксономії атак SCADA, що сприяють створенню наборів даних атак SCADA. Оскільки більшість наборів даних створюється для IT-систем, нам довелося переглянути відповідні таксономії атак SCADA, щоб отримати ідеї для генерації атак.

## 2.3 Таксономії атак на SCADA

З ідей, представлених у таксономіях атак SCADA, можна вивести вимоги для створення наборів даних атак SCADA. Для генерації атак у SCADA-мережах необхідно переглянути методи та інструменти, які використовували інші дослідники, що дозволяє розробити власні вимоги до генерації атак SCADA. Більшість протоколів SCADA проєктувалися без урахування питань безпеки, що дає змогу зловмисникам використовувати

функціональність протоколу під час його застосування в критичній інфраструктурі.

В роботі [17] запропонували таксономію атак протоколу DNP3, зосереджену на різних мережевих рівнях стеку протоколу DNP3. Вони виокремили такі категорії загроз: переривання, перехоплення, модифікація та створення. Кожна з цих категорій ідентифікує різні операції, які може виконати зловмисник для маніпуляцій із протоколом DNP3.

В роботі [18] представили таксономію атак на протокол SCADA MODBUS, яка ідентифікує комбінацію атак, що базуються на MODBUS через TCP/IP. Завдяки доступності стандартних технологій більшість протоколів SCADA, таких як MODBUS, можуть передаватися через TCP/IP або UDP/IP, що дозволяє зловмисникам маніпулювати інтернет-протоколами для впливу на системи SCADA.

В роботі [19] акцентували увагу на проблемах, успадкованих протоколами SCADA від інтернет-протоколів, які є вразливими до традиційних атак, таких як MITM, впровадження (injection) та маскування (masquerading). Інтеграція атак інтернет-протоколів і SCADA дає змогу зловмисникам виконувати різноманітні нелегальні дії у мережах SCADA.

В роботі [20] надали збірку сценаріїв атак SCADA, зосереджених на зондуванні, скануванні, затопленні, автентифікації, обході, спуфінгу, перехопленні, перенаправленні, читанні/копіюванні, завершенні, виконанні, модифікації та видаленні. Ці атаки SCADA можуть бути спрямовані на мережеві сервіси, такі як сервери, клієнти та широкомовлення.

Системи критичної інфраструктури використовують SCADA для забезпечення географічно віддалених операцій із обладнанням критичної інфраструктури. В роботі [21] описуються виклики захисту критичної інфраструктури як "надзвичайно складні" через те, що цілі дизайну SCADA-систем принципово відрізняються від традиційних IT-систем.

Більшість попередніх робіт концентруються на атаках, орієнтованих на традиційні IT-системи, та атаках на ізольовані протоколи автоматизації, але

ці концепції не комбінуються, обмежуючи охоплення можливих атак на критичну інфраструктуру.

В роботі [22] проаналізували технологічні досягнення в архітектурі мереж SCADA. Вони наголосили на високій залежності систем управління від IT, що спричиняє проблеми безпеки, такі як уразливість програмного забезпечення, шкідливі програми, проблеми в операційних процедурах і слабкі місця в дизайні або реалізації протоколів.

В роботі [23] надали огляд питань кібербезпеки та конфіденційності в технологіях "розумних мереж". Вони дійшли висновку, що більшість технологій у "розумних мережах" мають потенційні уразливості через властиві проблеми IT-систем.

## 2.4 Моделі атак

Корисність фреймворків кібератак допомагає дослідникам, розробникам і фахівцям із безпеки створювати й реалізовувати надійні системи, дозволяючи їм підготуватися до кібератак або пом'якшити їх наслідки. При цьому існує небагато моделей атак, які допомагають зрозуміти підходи та наслідки атак. Нижче представлено роботи, пов'язані з моделями атак, які ілюструють ландшафт атак.

В роботі [24] пропонується детальний аналіз дерев атак, що є "формальною методологією для аналізу безпеки систем і підсистем". Дерево атак представляє багатогієрархічну модель, яка відображає цілі зловмисника. Ці цілі, представлені у вигляді вузлів, використовуються для експлуатації вразливостей у цільовій системі. Верхній вузол вважається фінальною метою, коли зловмисник успішно завершує атаку. Кожен дочірній вузол у дереві атак є проміжною метою, яку зловмисник має виконати для просування вище в ієрархії. Вузли можуть виконуватися за допомогою логічних операцій "AND" і "OR". Операція "AND" вимагає виконання всіх цілей для просування, тоді як "OR" вимагає виконання лише однієї цілі. Ця

методологія була застосована до системи SCADA, де було розроблено набір цілей атак для впливу на систему генерації електроенергії. Більшість цілей атак у зосереджувалися на поганій реалізації мережевого обладнання та ненадійних паролях для контролю доступу. Хоча дерево атак дає розуміння впливу на інші системи під час кібератак, ці концепції не розглядаються детально щодо алгоритмів, які використовують зловмисники, або типів компонентів, які необхідно скомпрометувати для проведення атак.

Запропонований фреймворк класифікації кібератак ідентифікує окремі характеристики атак і механізми, за якими вони працюють. Він забезпечить методологію для ідентифікації та пом’якшення практичних атак на критичну інфраструктуру.

STRIDE – це модель загроз, розроблена корпорацією Microsoft, яка допомагає IT-фахівцям класифікувати загрози за категоріями: спуфінг особистості, модифікація даних, неможливість відмови, розкриття інформації, відмова в обслуговуванні та підвищення привілеїв. Ці категорії дозволяють IT-фахівцям формулювати загрози та розглядати можливі вектори атак. Хоча деякі категорії STRIDE можна застосувати до критичної інфраструктури, вони не враховують вплив, такий як фізичне пошкодження обладнання або проблеми безпеки. Наприклад, мета "відмова в обслуговуванні" описується як позбавлення доступу або можливості використання певної послуги, але не уточнює, чи стосується атака IT-систем, протоколів зв'язку, конфігурацій або процесів управління.

Щоб визначити відповідні вимоги для створення маркованих наборів даних атак, необхідно переглянути роботи, пов'язані з атаками, генерацією трафіку SCADA та розробкою наборів даних.

## 2.5 Кібератаки

Далі розглянемо огляд і базову інформацію у сфері кібератак. Буде представлено пов’язані теми та експерименти, які дозволили дослідити методи вивчення та генерації даних про кібератаки для SCADA.

Мережева розвідка не є новою концепцією для традиційних IT-інфраструктур. Однак створення карти існуючих SCADA-мереж для аудиту або виявлення вторгнень є складним завданням. Процес може включати пасивне спостереження за повідомленнями в мережі та їхніми адресами джерела й призначення або активне підключення до діапазону адрес і портів та очікування відповіді від пристроїв, яким призначені адреси в зазначеному діапазоні. Крім того, мережна розвідка SCADA може отримувати стани регістрів пам’яті або деталі конфігурації мережного обладнання автоматизації. Через додатковий рівень адресації, специфічний для протоколу автоматизації, для SCADA потрібен ще один рівень картографування. Деякі техніки мережевої розвідки, які використовуються для мереж на базі Ethernet або Інтернету.

Пасивна мережна розвідка включає пристрій, що спостерігає за мережевим трафіком. Такий пристрій не взаємодіє з трафіком чи цільовими пристроями, що знижує ризик пошкодження обладнання або переривання мережевої комунікації. Нижче наведено короткий огляд двох пасивних методів: дзеркалювання портів та використання мережевого крана.

Дзеркалювання портів — це конфігурація комутатора, яка дозволяє пересилати трафік із вибраного порту на пристрій для аналізу. Аналізатор може бути комп’ютером із високоресурсною мережею, що використовує інструменти аналізу, наприклад Wireshark або TCP dump, для захоплення трафіку.

Мережевий кран – це фізичний мережевий пристрій, який забезпечує пасивний метод моніторингу мережевого трафіку. У критичній інфраструктурі мережеві крани зазвичай є оптичними розгалужувачами, що перенаправляють трафік на аналізатор. Як і дзеркалювання портів, трафік може бути переданий на високоресурсний мережевий інтерфейс для аналізу.

Активна мережна розвідка потребує взаємодії пристрою розвідки з цільовими пристроями. У цьому випадку пристрій створює мережеві транзакції для отримання адресної інформації та можливих конфігурацій.

Одним із найбільш поширених інструментів активної розвідки є Nmap, який використовує IP-пакети для виявлення хостів і відкритих портів у мережі. Іншим інструментом є Masscan, здатний сканувати весь простір публічних IP-адрес за три хвилини.

Нав’язливі методи мережевої розвідки включають такі техніки, як ARP-отруєння, що вводить пристрої в оману, змушуючи їх пересилати трафік через пристрій-зловмисник. Це дозволяє зловмиснику аналізувати весь трафік без потреби додаткового обладнання.

Розвідка SCADA-мереж є складнішою через особливості протоколів автоматизації, таких як DNP3, S7comm, MMS і GOOSE, які мають додаткові схеми адресації. З цієї причини наукова спільнота продовжує розробляти нові методи для розвідки мереж критичної інфраструктури. Наприклад, в роботі [25] описуються алгоритми, що застосовуються для сканування традиційних мереж, але не адаптовані до специфіки SCADA-протоколів. В роботі [26] представили фреймворк для інтернет-сканування, орієнтованого на обладнання SCADA.

MITM-атаки (man-in-the-middle) дозволяють зловмиснику розташувати себе між сторонами, що спілкуються, щоб перехоплювати, модифікувати чи руйнувати повідомлення. Крім того, атаки відтворення (replay attacks) дозволяють повторно відправляти попередньо використані повідомлення для маніпуляцій із цільовою системою. Більшість протоколів автоматизації, таких як DNP3, не мають механізмів аутентифікації, що робить їх вразливими до таких атак.

Дослідники, зокрема [27] із Sandia National Laboratory, створили тестові стенди для демонстрації атак на SCADA-системи, використовуючи вразливості корпоративних мереж. Хоча ці атаки стосувалися загальних проблем IT-безпеки, вони не аналізували безпосередньо слабкі місця протоколів SCADA.

# 3 МОДЕЛЬ ГЕНЕРАЦІЇ НАБОРІВ ДАНИХ АТАК SCADA

В розділі представлено фреймворк для створення даних атак SCADA. Цей фреймворк пропонує методологію для генерації кібератак SCADA з метою створення наборів даних. Згенерований трафік атак буде сприяти розробці систем виявлення вторгнень на основі мережевих даних. Таким чином, фокус фреймворку зосереджено на атаках, які є специфічними для мережних середовищ.

## 3.1 Фреймворк генерації атак на SCADA

Ознайомившись із основами SCADA, оглядами таксономій атак, методами генерації даних та наборами даних атак, боло визначено 10 вимог, які сприятимуть створенню даних про кібератаки SCADA. Ці вимоги є мінімально необхідними для створення інструменту атак, здатного симулювати систему управління за допомогою мережевих атак.

Використовуючи запропоновані 10 вимог, можна розробити модульний інструмент для генерації атак, що буде симулювати реакцію цільового пристрою на атаки. Нижче наведено наші вимоги для генерації даних атак SCADA:

1. здатність розбирати повідомлення протоколів SCADA. Інструмент повинен мати можливість аналізувати повідомлення SCADA-протоколів, що дозволяє отримувати інформацію про їхню структуру та вміст;
2. здатність відтворювати стек протоколів SCADA. Інструмент має імітувати функціонування стеку протоколів SCADA, забезпечуючи точне відтворення операцій протоколу;
3. здатність прослуховувати локальний трафік SCADA-мережі. Має бути можливість пасивно перехоплювати трафік SCADA-мережі для аналізу та виявлення моделей поведінки;

4. здатність впроваджувати аномальні повідомлення протоколу SCADA у мережу. Інструмент повинен бути здатним інжектувати аномальні повідомлення у мережу для перевірки реакції системи на несанкціоновані дії;

5. здатність змінювати дані протоколу в режимі реального часу. Повідомлення протоколу мають бути модифіковані в реальному часі для тестування динамічної реакції цільової системи;

6. надання сервісу майстра протоколу для маскування. Інструмент має підтримувати імітацію майстра протоколу (master service), дозволяючи маскуватися під легітимні джерела;

7. надання сервісу веденого протоколу для маскування. Має бути реалізований сервіс веденого протоколу (slave service), щоб імітувати поведінку підконтрольного пристрою;

8. надання функцій розвідки SCADA-мереж для виявлення додатків SCADA. Інструмент має забезпечувати функції для розвідки SCADA-мереж, виявлення додатків та збору інформації про них;

9. здатність відтворювати попередні повідомлення протоколу SCADA Інструмент повинен мати можливість відтворювати раніше перехоплені повідомлення для тестування реакції системи;

10. здатність перевантажувати сервіс SCADA аномальними повідомленнями. Повинна бути можливість здійснювати атаки затоплення (flooding) аномальними повідомленнями для тестування стійкості сервісів SCADA.

Перша вимога – це здатність розбирати повідомлення протоколів SCADA. Це дозволяє інструменту атак декодувати або витягати компоненти з повідомлення протоколу. Друга вимога – здатність відтворювати стек протоколів SCADA. Це дозволяє інструменту атак дотримуватись правил SCADA-протоколу для реалізації складних атак. Третя вимога – здатність перехоплювати (sniff) дані, що дає змогу інструменту атак захоплювати інформацію з будь-якого з’єднання SCADA. Четверта вимога – здатність впроваджувати аномальні повідомлення в мережний трафік SCADA, що

дозволяє маніпулювати операціями системи на цільових пристроях SCADA. П’ята вимога – це здатність змінювати повідомлення протоколу в реальному часі, що дозволяє виконувати атаки типу "людина посередині" (MITM) у реальному часі. Шоста вимога – це здатність надавати сервіс master протоколу, що дозволяє здійснювати маскування, а сьома вимога – це здатність надавати сервіс Slave протоколу для маскування. Деякі протоколи SCADA мають значні відмінності між операціями master і slave пристроїв; тому інструмент генерації атак повинен враховувати обидва типи. Восьма вимога – здатність виконувати мережну розвідку або зондування пристроїв. Це дозволяє інструменту атак створювати карту цільових пристроїв SCADA. Дев’ята вимога – це здатність відтворювати попередні повідомлення SCADA-протоколу. Це демонструє значущість використання легітимного трафіку у невідповідний час. Десята і остання вимога – це здатність затоплювати SCADA-сервіс повідомленнями для маніпулювання будь-якими автоматизованими функціями цілі.

Далі розглянемо запропонований фреймворк генерації даних про кібератаки SCADA, який відповідає визначеним вимогам. Фреймворк, зображений на рисунку 4.1, складається з чотирьох категорій: базові мережеві модулі, модулі SCADA, модулі атак і розширені модулі атак.

Базові мережеві модулі використовуються для взаємодії фреймворку з мережевим середовищем, таким як Ethernet, бездротова мережа або серійне з’єднання. До базових мережевих модулів належать: сокети, сервер і клієнт.

Модулі SCADA застосовуються для відтворення протоколу SCADA в цільовій системі. Ці модулі забезпечують можливість заміни цільового протоколу SCADA у фреймворку, тому будь-які значні зміни необхідно вносити лише в реалізації модулів SCADA. До модулів SCADA входять: PDU (Protocol Data Unit), спуфер (spoofing), майстер (master) і ведений пристрій (slave).

Кожен із цих модулів розширюється або ініціалізується як екземпляр для створення таких модулів у категорії атак: маскування веденого

(slave masquerading), маскування майстра (master masquerading), ін'єкція (injection) і атака типу "людина посередині" (MITM). Категорія атак дозволяє модулям SCADA розширюватися для атак, заснованих на протоколах SCADA. Для адаптації модулів атак до нового протоколу потрібні лише незначні зміни в реалізації.

Розширені модулі атак використовуються для здійснення конкретних атак на цільові протоколи SCADA. До розширених модулів атак належать: розвідка (reconnaissance), затоплення веденого (slave flooding), модифікація MITM, захоплення MITM (MITM hijacking), відтворення ін'єкцій (injection replay), відтворення майстра (master replay) і затоплення майстра (master flooding).

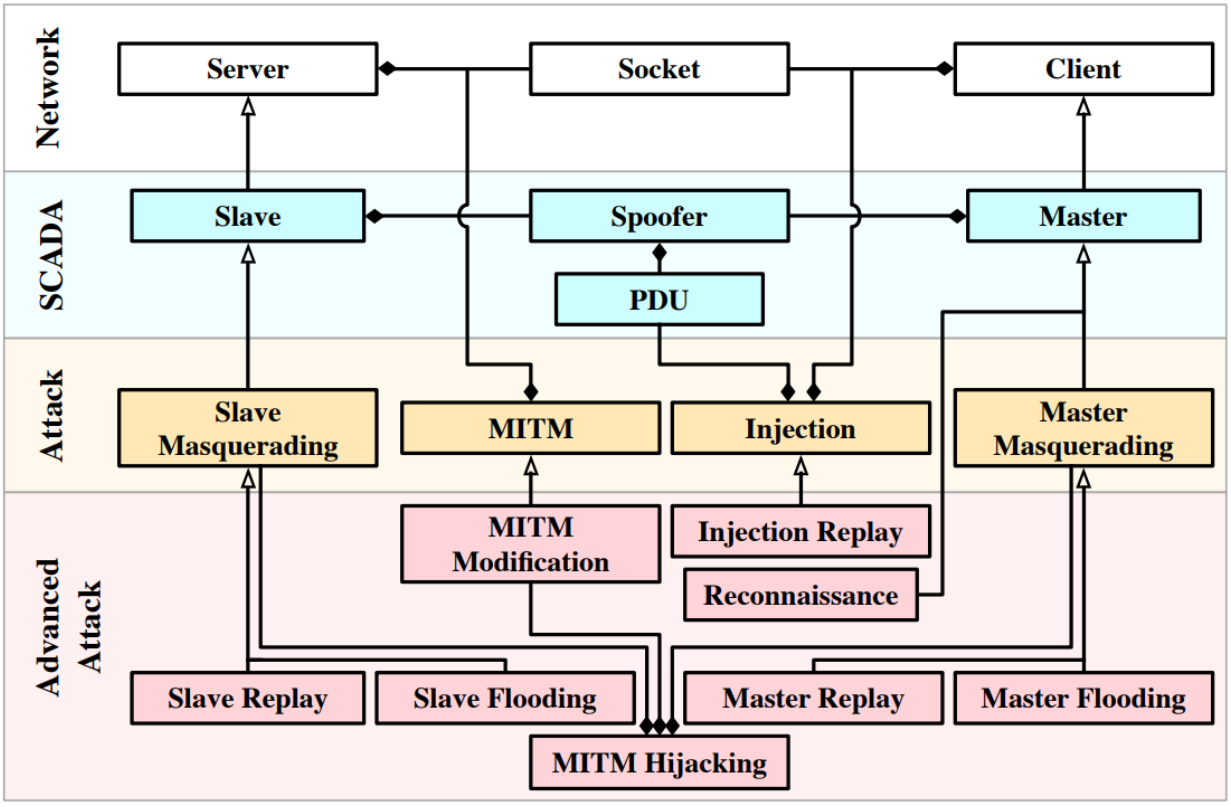


Рисунок 3.1 – UML діаграма класів представленого фреймворку генерації даних про кібератаки SCADA

Далі опишемо детально кожен з компонентів.

## 3.2 Компоненти системи

У цьому розділі представлено кожен модуль фреймворку генерації атак SCADA та демонструємо його відповідність поданим вимогам. Для ілюстрації впливу реалізації алгоритмів, представлених у модулях атак, зосереджуємо увагу на цільовому протоколі SCADA DNP3.

Перша вимога для розробки фреймворку генерації атак SCADA – здатність розбирати повідомлення протоколів SCADA – передбачає розробку структури PDU (Protocol Data Unit) цільового протоколу SCADA, що забезпечує логіку маніпуляцій із мережними пакетами. Модуль PDU дозволяє створювати екземпляри повідомлень протоколу SCADA на основі його специфікацій.

Процес маніпуляції пакетами є критично важливим для роботи всього фреймворку, оскільки він забезпечує можливість модифікувати протокол SCADA, інжектувати повідомлення протоколу SCADA та спуфінг стеку протоколу SCADA. Це дозволяє фреймворку генерації даних бути розширюваним для різних типів атак SCADA.

Модуль PDU знаходиться на рівні SCADA у фреймворку, оскільки кожен цільовий протокол SCADA вимагає власного індивідуального модуля PDU для генерації атак.

Спуфінг – це метод імітації характеристик системи, протоколу або процедури з метою надання неправдивої або оманливої інформації. Модуль спуфера може потребувати набору структур даних, які дозволяють відстежувати та контролювати операції, такі як стани даних цілі, наприклад, регістри пам’яті. Крім того, операції з даними можуть використовуватись під час маскування під існуючий пристрій автоматизації.

Здатність забезпечувати спуфінг є необхідною для виконання вимоги 2. Модуль спуфера у фреймворку дозволяє зловмиснику імітувати логіку мережного стеку цільового протоколу SCADA. Спуфер використовує модуль PDU для відтворення мережного стеку протоколу SCADA. Модуль спуфера

може потребувати базу даних або доступ до файлової системи для зберігання інформації про об’єкти або встановлені точки для відповідного протоколу.

Модуль спуфера знаходиться на рівні SCADA у фреймворку, оскільки кожен цільовий протокол SCADA вимагає власного індивідуального модуля спуфера для генерації атак.

Ін’єкція – це процес, за допомогою якого зловмисник може вставляти повідомлення у вже існуюче з'єднання між двома сторонами, що спілкуються, з метою маніпуляції. Модуль ін’єкції повинен буде перехоплювати існуюче з'єднання між двома цільовими пристроями, щоб відстежувати інформацію про послідовність з'єднання; для цього можна використовувати бібліотеку сокетів для перехоплення трафіку, який потім передається до модуля ін’єкції. Оскільки перехоплення є необхідним методом для ін’єкції, модуль ін’єкції дозволяє виконати вимогу 3.

Деякі протоколи SCADA, які не використовують послідовний порт як частину правил протоколу, інкапсулюються в транспортному протоколі (наприклад, TCP), що може усунути потребу у перехопленні для атак ін’єкції. Модуль ін’єкції використовується для введення серії повідомлень у з'єднання цільової системи, виконуючи таким чином вимогу 4.

Алгоритм для атак ін’єкції описано в лістингу 3.1.

Лістинг 3.1 – Алгоритм для атак ін’єкції

```
1: packet ← {ethernet, ip, tcp}
2: pdu ← PDU()

3: procedure INJECT
4:     sequence ← packet_tcp_seq + len(pdu) + len(mal_pay)
5:     packet_tcp_seq ← sequence
6:     SOCKET.SEND({packet, pdu, mal_pay})

7: procedure UPDATE(p)
8:     packet_ethernet_src ← p_ethernet_src
9:     packet_ethernet_dst ← p_ethernet_dst
10:    packet_ip_src ← p_ip_src
11:    packet_ip_dst ← p_ip_dst
12:    packet_tcp_src ← p_tcp_src
13:    packet_tcp_dst ← p_tcp_dst
```

```
14:     packet_tcp_seq ← p_tcp_seq
15:     packet_tcp_ack ← p_tcp_ack
16:     if PDU ∈ p then
17:         pdu_src ← PPDU_src
18:         pdu_dst ← PPDU_dst
19:         pdu_seq ← (PPDU_seq+1)

20: procedure SNIFF
21:     while running do
22:         p ← SOCKET.RECV
23:         if p ≠ ∅ then
24:             UPDATE(p)
```

Використовуючи екземпляр модуля PDU, модуль ін’єкції може створювати одне або кілька вставних повідомлень PDU і вставляти ці повідомлення в мережу, викликаючи маніпуляцію цільовими пристроями SCADA.

Модуль ін’єкції спрямований на створення транспортного рівня мережі протоколу SCADA, наприклад TCP/IP, для відстеження послідовності та забезпечення коректності. Це продемонстровано у рядках 8-19 алгоритму з використанням TCP/IP для ілюстрації. Ідеально, модуль повинен мати доступ до цільового з’єднання через Ethernet-хаб, дзеркальний порт на комутаторі або прямий мережний інтерфейс на цільовому пристрої.

Модуль ін’єкції оновлює будь-які поля послідовності, які потрібні для PDU протоколу SCADA. Процедура inject, показана в алгоритмі 3.1, активується додатковою функцією, умовою або вводом користувача, що запускає процес ін’єкції. У транспортну послідовність ін’єктованого пакета (packet) додається довжина створеного PDU (pdu) та шкідливого навантаження (mal_pay), що дозволяє пакету бути прийнятим цільовим пристроєм SCADA при ін’єкції, оскільки введені повідомлення синхронізовані зі з’єднанням.

Деякі пропрієтарні протоколи SCADA мають власний транспортний рівень, тому будь-яка реалізація модуля ін’єкції для такого протоколу повинна враховувати його специфічний функціонал. Корисність модуля ін’єкції полягає в його здатності розширюватися та створювати різноманітні атаки ін’єкції проти SCADA-протоколів.

Модуль ін’єкції знаходиться на рівні SCADA у фреймворку, оскільки кожен цільовий SCADA-протокол потребує власного індивідуального модуля ін’єкції для створення атак.

Маскування – це процес, за допомогою якого зловмисник імітує пристрій або сервіс, надаючи або легітимну, або хибну інформацію з метою маніпуляції цільовим клієнтом або сервером. У представленому фреймворку існують два модулі маскування: Master Masquerading, який є розширенням модуля Master, і Slave Masquerading, який є розширенням модуля Slave. Кожен із представлених модулів маскування повинен бути реалізований із логікою для відповіді відповідно до конфігурації цільової системи.

Процес маскування «удає» з себе існуючий сервіс і виконує ті самі контрольні процеси, що й реальний сервіс модуля Master, і Slave. Наприклад, якщо реальний Master налаштований на запит трьох об’єктів даних з інтервалом у 3000 мс, Master маскування також має бути налаштований на запит цих самих об’єктів даних з таким самим часовим циклом. Проте можна внести незначні корективи для створення бажаних станів для атаки.

Наприклад, реалізація маскування Slave може містити екземпляри всіх можливих об’єктів даних, які використовуються для цільового контрольного процесу. Маскування Slave може містити тригери, що дозволяють встановлювати ці об’єкти даних у стани, необхідні для атаки, тим самим маніпулюючи цільовим майстром.

Лістинг 3.2 – Процедура автоматизації маскування для Master пристроїв

```
1: spoofer ← Spoofer()
2: running ← True
3: queue_out ← Queue()

4: procedure AUTOMATION
5:     while running do
6:         request1 ← SPOOFER.REQUEST(message1)
7:         request2 ← SPOOFER.REQUEST(message2)
8:         request3 ← SPOOFER.REQUEST(message3)
9:         sleep(interval)
```

```
10:         queue_out(request1)
11:         queue_out(request2)
12:         queue_out(request3)
```

Лістинг 3.3 – Процедура автоматизації маскування для Slave пристроїв

```
1: spoofer ← Spoofer()
2: trigger ← False

3: procedure AUTOMATION
4:     while running do
5:         if trigger then
6:             unsolicited ← SPOOFER.UNSOLICITED(message)
7:             queue_out(unsolicited)
8:             trigger ← False
```

Модулі Master Masquerading і Slave Masquerading дозволяють фреймворку повністю виконати вимоги 6 і 7.

Розвідка – це процес отримання інформації про цільову систему. Цей процес може включати сканування ARP, TCP sync або, а також збір додаткової інформації про протокол SCADA, його функціональність чи конфігурацію. Модуль розвідки дозволяє отримувати інформацію про ініціалізовані об’єкти даних і значення точок даних, виявляти регістри пам’яті або інші дані, що зберігаються на контролерах автоматизації.

Отримуючи та вивчаючи таку інформацію про процес SCADA, модуль розвідки може ініціалізувати власні об'єкти даних для подальшої взаємодії з ціллю в атаках типу маскування. Покращуючи модулі маскування, можна здійснювати складні атаки на цільову систему SCADA. Завдяки реалізації цієї функціональності модуль розвідки задовольняє вимогу 8: забезпечення функцій виявлення/розвідки мереж SCADA для цільових додатків SCADA у представленому фреймворку.

В алгоритмі 3.4 наведено абстрактний приклад алгоритму виявлення адрес підпорядкованих пристроїв (slave) SCADA.

Лістинг 3.4 – Алгоритм визначення адреси

```
1: src ← 0
2: dst ← 0
3: rsp ← False
4: while ¬rsp ∧ src ≤ MAX_ADDRESS do
5:     while ¬rsp ∧ dst ≤ MAX_ADDRESS do
6:         PDU_src ← src
7:         PDU_dst ← dst
8:         SEND(PDU)
9:         dst ← dst + 1
10:    src ← src + 1
```

Деякі протоколи SCADA можуть використовувати додатковий рівень адресації, наприклад DNP3 і Modbus. Представлений алгоритм виявлення адрес дозволяє знаходити мережу SCADA-протоколу. Створюється PDU SCADA, і адреси джерела (PDUsrc) та призначення (PDUdst) інкрементуються доти, доки цільовий пристрій не надасть відповідь (rsp). Значення адреси джерела та призначення інкрементуються, поки обидва значення не досягнуть максимального значення адреси.

Атака типу відтворення (Replay attack) використовується зловмисником, який отримав легітимне операційне повідомлення, використане цільовою системою, і повторно використовує це повідомлення, щоб маніпулювати або пошкодити цільову систему. Виконуючи такі атаки, зловмисник може порушити або змінити роботу системи, експлуатуючи раніше використаний легітимний трафік для створення шкідливого впливу на систему.

Атаки типу відтворення добре відомі та часто використовувалися для атак на механізми автентифікації. Додавши модуль відтворення (Replay module) до фреймворку, змогли виконати вимогу 9 — здатність повторно відтворювати попередні повідомлення протоколу SCADA.

Наприклад, повторне відтворення трафіку, що містить паролі, може бути використано для примусової автентифікації неавторизованого користувача. У представленому фреймворку описано три типи атак

відтворення: Injection Replay, Slave Replay та Master Replay. Кожен із цих модулів забезпечує метод відтворення раніше зібраних легітимних повідомлень системи управління.

Injection Replay – це розширення модуля Injection, яке налаштовується на використання одного або кількох критичних повідомлень із колекції раніше перехопленого трафіку SCADA. Повідомлення обробляється екземпляром модуля PDU, що дозволяє синхронізувати повідомлення з цільовим з'єднанням. Після активації ін’єкції повторно відтворювані повідомлення вводяться у з'єднання (див. Алгоритм 1).

Master Replay – це розширення модуля Master Masquerading, яке відтворює лише повідомлення майстра з раніше зібраного трафіку. Цей процес здійснюється шляхом фільтрації повідомлень до атаки, а потім їх повторної передачі у послідовності з урахуванням проміжку часу між повідомленнями алгоритм 3.5.

Лістинг 3.5 – Алгоритм SCADA PDU Replay

```
1: replay_pcap ← Queue(pcap)

2: procedure REPLAY
3:     last_out_time ← 0
4:     while replay_pcap ≠ ∅ ∧ running do
5:         msg ← replay_pcap₁
6:         wait_time ← (msgₜ - last_out_time)
7:         sleep(wait_time)
8:         socket.send(msgPDU)
9:         last_out_time ← msgₜ
```

Коли мова йде про цільові протоколи автоматизації, повторно відтворені повідомлення можуть потребувати оновлення деяких їхніх обчислюваних полів або синхронізації з умовами цільового пристрою, наприклад, послідовних номерів, CRC (контрольних сум) та контрольних кодів. Ці завдання повинні оброблятися модулем PDU.

В Алгоритмі 3.6 представлено процес підготовки до атаки відтворення, яка спрямована на підпорядкований пристрій (slave).

Лістинг 3.6 – Основний фільтр повідомлень

```
1: replayList ← {}
2: procedure MasterMessageFilter(p)
3:     if PDU ∈ p then
4:         if p_src ← master_src then
5:             replayList = replayList ∪ p_PDU
6: for p ∈ capture do
7:     MasterMessageFilter(p)
```

Змінна replayList отримує структуру даних списку для збору списку структур пакетів із вхідного процесу. Процедура MasterMessageFilter використовується для фільтрації пакетів p, які спеціально надсилаються підпорядкованому пристрою майстром (master).

Процедуру MasterMessageFilter можна викликати у реальному часі функцією перехоплення трафіку або програмою обробки збереженого трафіку, яка аналізує раніше перехоплені дані мережевої комунікації. Процес MasterMessageFilter перевіряє, чи містить пакет цільовий протокол PDU, після чого визначає напрямок пакета.

Напрямок (до slave або master) у протоколі автоматизації може бути визначений за допомогою бітових прапорців, адресації на канальному рівні протоколу або кодів функцій, які вказують функції підпорядкованих пристроїв. Якщо умова напрямку виконується, пакет p буде доданий до списку replayList як підготовка до атаки відтворення.

Флудинг (Flooding) – це процес відправлення серії повідомлень до цільового сервісу SCADA, що змушує його залишатися в стані, бажаному для зловмисника. У результаті цільова система або втрачає легітимні запити, або повертається в стан, який відповідає намірам зловмисника.

Представлений фреймворк використовує два модулі – Master Flooding і Slave Flooding, щоб задовольнити вимогу 10: здатність затоплювати сервіс SCADA аномальними повідомленнями.

Master Flooding, розширення модуля Master Masquerading, містить набір попередньо визначених або налаштованих повідомлень. Ці повідомлення надсилаються до цільового підпорядкованого пристрою (slave) з максимальною швидкістю, щоб помістити його в бажаний для атаки стан, шляхом перезапису регістрів пам’яті або точок даних.

Аналогічний метод використовується модулем Slave Flooding, розширенням модуля Slave Masquerading, де критичні повідомлення відправляються назад до цільового майстра (master). Це поміщає майстра у критичний стан, що може порушити його функціонування або спричинити збої в роботі системи.

Ці модулі забезпечують можливість створення сценаріїв атак типу флудинг, які можуть вплинути на обидві сторони комунікації SCADA-протоколів (майстрів і підпорядкованих пристроїв).

Атака типу "людина посередині" (Man-In-The-Middle, MITM) — це процес, за допомогою якого зловмисник проникає в існуюче з'єднання та перехоплює повідомлення між сторонами, що спілкуються. Найпоширеніший підхід до MITM-атак у мережах – використання ARP-отруєння. Однак існують і інші підходи до здійснення MITM-атак, включаючи компрометацію мережевих карт або обладнання, такого як маршрутизатори чи комутатори.

Модуль MITM, представлений у фреймворку, використовується для задоволення вимоги 9. Цей модуль дозволяє перехоплювати трафік між цільовими сторонами, обробляти його через модуль PDU, а потім перенаправляти до призначеної сторони.

Модуль MITM має значно ширші можливості, ніж просто читання та пересилання інформації із з'єднання лістинг 3.7.

Лістинг 3.7 – MITM Forwarding

```
1: procedure FORWARDING(packet)
2:     if packet_ether_dst ≡ Attacker_mac then
3:         if packet_ip_src ≡ Master_ip then
4:             packet_ether_dst ← Master_mac
5:             SOCKET.SEND(packet)
6:         else if packet_ip_src ≡ Slave_ip then
7:             packet_ether_dst ← Slave_mac
8:             SOCKET.SEND(packet)
9:         else Drop packet
```

# ВИСНОВКИ

У даній кваліфікаційній роботі була представлена розробка фреймворку для генерації даних про кібератаки SCADA, який враховує сучасні виклики та вимоги до захисту систем критичної інфраструктури. Основна увага приділялася розробці модульного підходу, який дозволяє забезпечити гнучкість та масштабованість у процесі створення тестових даних для виявлення та аналізу атак.

У рамках роботи було виконано:

– аналіз існуючих таксономій атак на протоколи SCADA та їх впливу на критичну інфраструктуру;

– визначено 10 основних вимог до фреймворку генерації атак, які дозволяють створювати реалістичні дані для тестування;

– розробка та імплементація модулів фреймворку, таких як PDU, спуфер, модулі ін’єкції, маскування та MITM, для реалізації атак на мережевому рівні.

Результати роботи дозволяють зробити висновок, що запропонований фреймворк здатний генерувати дані, які можуть бути використані для розробки та тестування систем виявлення вторгнень, а також для глибшого розуміння поведінки SCADA-систем під час кібератак.

Подальший розвиток фреймворку може включати розробку експериментального стенда для перевірка роботи фреймворку на тестовому середовищі DNP3, що продемонструє практичну цінність реалізованого підходу.

# ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Abou el Kalam, Anas. "Securing SCADA and critical industrial systems: From needs to security mechanisms." International Journal of Critical Infrastructure Protection 32 (2021): 100394.

2. Conti, Mauro, Denis Donadel, and Federico Turrin. "A survey on industrial control system testbeds and datasets for security research." IEEE Communications Surveys & Tutorials 23.4 (2021): 2248-2294.

3. Gönen, Turan, Chee-Wooi Ten, and Ali Mehrizi-Sani. Electric power distribution engineering. CRC press, 2024.

4. Knapp, Eric D. Industrial Network Security: Securing critical infrastructure networks for smart grid, SCADA, and other Industrial Control Systems. Elsevier, 2024.

5. Pellegrino, Felice Andrea, et al. "Closed-loop control from data-driven open-loop optimal control trajectories." 2022 European Control Conference (ECC). IEEE, 2022.

6. Рубан, І. В., В. О. Мартовицький, and С. О. Партика. "Класифікація методів виявлення аномалій в інформаційних системах." Системи озброєння і військова техніка 3 (2016): 100-105.

7. Justindhas, Y., and P. Jeyanthi. "Attack detection and prevention in IoT-SCADA networks using NK-classifier." Soft Computing 26.14 (2022): 6811-6823.

8. Yang, Zhen, et al. "A systematic literature review of methods and datasets for anomaly-based network intrusion detection." Computers & Security 116 (2022): 102675.

9. Cordero, Carlos Garcia, et al. "On generating network traffic datasets with synthetic attacks for intrusion detection." ACM Transactions on Privacy and Security (TOPS) 24.2 (2021): 1-39.

10. Thomas Morris and Wei Gao. Industrial control system traffic data sets for intrusion detection research. In Critical Infrastructure Protection VIII, pages

65–78. Springer, 2014.

11. Stabili, Dario, et al. "Daga: Detecting attacks to in-vehicle networks via n-gram analysis." IEEE Transactions on Vehicular Technology 71.11 (2022): 11540-11554.

12. Seokjun Lee, Hyunguk Yoo, Jungtaek Seo, and Taeshik Shon. Packet diversity-based anomaly detection system with ocsvm and representative model. In Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2016 IEEE International Conference on, pages 498–503. IEEE, 2016.

13. Antoine Lemay and José M Fernandez. Providing scada network data sets for intrusion detection research. In CSET@ USENIX Security Symposium, 2016.

14. Lazaridis, George, et al. "Securing Modbus TCP Communications in I4. 0: A Penetration Testing Approach Using OpenPLC and Factory IO." 2023 IEEE Conference on Standards for Communications and Networking (CSCN). IEEE, 2023.

15. Y. Lopes, D. C. Muchaluat-Saade, N. C. Fernandes, and M. Z. Fortes. Geese: A traffic generator for performance and security evaluation of iec 61850 networks. In Proceedings of ISIE 2015, June 2015. doi: 10.1109/ ISIE.2015.7281552.

16. Garitano, I., Siaterlis, C., Genge, B., Uribeetxeberria, R., & Zurutuza, U. (2012, September). A method to construct network traffic models for process control systems. In Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012) (pp. 1-8). IEEE.

17. Samuel East, Jonathan Butts, Mauricio Papa, and Sujeet Shenoi. A Taxonomy of Attacks on the DNP3 Protocol. In Critical Infrastructure Protection III, pages 67–81. Springer, 2009.

18. Peter Huitsing, Rodrigo Chandia, Mauricio Papa, and Sujeet Shenoi. Attack Taxonomies for the Modbus Protocols. International Journal of Critical

Infrastructure Protection, 1(0):37 – 44, 2008. ISSN 1874-5482. doi: http://dx.doi.org/10.1016/j.ijcip.2008.08.003

19. B. Zhu, A. Joseph, and S. Sastry. A Taxonomy of Cyber Attacks on SCADA Systems. In Proceedings of 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing Internet of Things (iThings/CPSCom), pages 380–388, Oct 2011. doi: 10.1109/iThings/CPSCom.2011.34.

20. Z. Drias, A. Serhrouchni, and O. Vogel. Taxonomy of attacks on industrial control protocols. In 2015 ICPE - NTDS, July 2015. doi: 10.1109/NOTERE.2015.7293513.

21. R.E. Johnson. Survey of scada security challenges and potential attack vectors. In Internet Technology and Secured Transactions (ICITST), 2010 International Conference for, pages 1–5, Nov 2010.

22. Cristina Alcaraz, Gerardo Fernandez, and Fernando Carvajal. Security aspects of scada and dcs environments. In Javier Lopez, Roberto Setola, and StephenD. Wolthusen, editors, Critical Infrastructure Protection, volume 7130 of Lecture Notes in Computer Science, pages 120–149. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-28919-4. doi: 10.1007/978-3-642-28920-0_7. URL http://dx.doi.org/10.1007/ 978-3-642-28920-0_7.

23. Jing Liu, Yang Xiao, Shuhui Li, Wei Liang, and C. L. Philip Chen. Cyber security and privacy issues in smart grids. Communications Surveys Tutorials, IEEE, 14(4):981–997, Fourth 2012. ISSN 1553-877X. doi: 10.1109/SURV.2011.122111.00145.

24. Bruce Schneier. Attack trees, December 1999. URL https://www. schneier.com/paper-attacktrees-ddj-ft.html.

25. Donnet, B., & Friedman, T. (2007). Internet topology discovery: a survey. IEEE Communications Surveys & Tutorials, 9(4), 56-69.

26. Myers, David, Ernest Foo, and Kenneth Radke. "Internet-wide Scanning Taxonomy and Framework." Proceedings of the 13th Australasian Information Security Conference (AISC 2015). Vol. 27. 2015.

27. URIAS, Vincent; VAN LEEUWEN, Brian; RICHARDSON, Bryan. Supervisory Command and Data Acquisition (SCADA) system cyber security analysis using a live, virtual, and constructive (LVC) testbed. In: Milcom 2012-2012 ieee military communications conference. IEEE, 2012. p. 1-8.

28. Пасічнюк Р.Р., Ільїна І.В.,МОДЕЛЬ ГЕНЕРАЦІЇ АТАК І МАРКУВАННЯ НАБОРІВ ДАНИХ ПРО АТАКИ ДЛЯ СИСТЕМ ВИЯВЛЕННЯ ВТОРГНЕНЬ // Проблеми інформатизації : XII міжнародна науково-технічна конференція. - 21-22 листопада 2024. –с.105. doi: https://doi.org/10.32620/PI.24.t2