НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

«До захисту допущено»
В.о. завідувача кафедри

_____ М.В.Грайворонський
(підпис)
“_____”_____2019 р.

Дипломна робота

на здобуття ступеня бакалавра

з напряму підготовки 6.170101 «Безпека інформаційних і комунікаційних систем»

на тему: «Оцінка захищеності систем SCADA методом моделювання»

Виконав (-ла): студент (-ка) 4 курсу, групи ФБ-52
(шифр групи)

Рішко Катерина Михайлівна
(прізвище, ім’я, по батькові) (підпис)

Керівник Коломицев Михайло Володимирович
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант _____ _____ _____
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали) (підпис)

Рецензент _____ _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ - 2019 року

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

Рівень вищої освіти – перший (бакалаврський)

Напрям підготовки 6.170101 «Безпека інформаційних і комунікаційних систем»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ М.В.Грайворонський
(підпис)
«___»_____2019 р.

ЗАВДАННЯ
на дипломну роботу студенту

Рішко Катерині Михайлівні
(прізвище, ім'я, по батькові)

1. Тема роботи: «Оцінка захищеності систем SCADA методом моделювання»
_____
_____ ,
науковий керівник роботи: Коломицев Михайло Володимирович,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету від «____ » 2019 р. № _____

2. Термін подання студентом роботи 10 червня 2019 р.

3. Вихідні дані до роботи _____
_____
_____

4. Зміст роботи_____
_____
_____
_____

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо) _____
_____
_____
_____

6. Дата видачі завдання _____

РЕФЕРАТ

Робота обсягом 69 сторінок, містить 25 ілюстрацій, 1 таблицю, 19 літературних посилань та 1 додаток.

Мета роботи полягає в оцінці захищеності систем SCADA за допомогою створення моделі реальної спрощеної системи.

Завданням роботи є створення моделі системи, яка допомагає передбачити вразливості та шляхи атаки в новій системі, перед введенням її в експлуатацію, або ж у вже використовуваній системі, коли необхідно ввести в неї зміни, що можуть позначитися на рівні захищеності.

Об’єктом дослідження є сучасні системи SCADA.

Предметом дослідження є захищеність сучасних SCADA-систем.

Методами дослідження є інструменти для оцінки захищеності систем, такі як EAAT разом з CySeMoL.

Результати роботи викладені у вигляді змодельованої спрощеної системи SCADA Siemens а також проілюстровані обчислені за допомогою CySeMoL шляхи атаки на систему.

Результати роботи можуть використовуватись при оцінці захищеності нових SCADA-систем, перед введенням їх в експлуатацію, або ж для оцінки вже використовуваних систем, перед внесенням в них змін, що можуть вплинути на безпеку в цілому.

SCADA-система, оцінка ризиків, кількісний аналіз ризиків, захищеність системи, CySeMoL, EAAT, вектори атаки, система безпеки

ABSTRACT

The work includes 69 pages, 25 figures, 1 table, 19 literary references and 1 appendix.

The purpose of the work is to assess the security of SCADA systems by creating a model of a simplified real system.

The task of the work is to create a system model that helps to predict the vulnerabilities and ways of attack in the new system before it is put into operation, or in the system that is already in use, when it is necessary to introduce changes that can affect the level of security.

The object of the research is modern SCADA systems.

The subject of the research is the protection of modern SCADA systems.

Research methods are tools for evaluating security systems, such as EAAT, along with CySeMoL.

The results of the work are presented in the form of simulated simplified SCADA system Siemens and also illustrated using CySeMoL ways to attack the system.

Work results can be used to assess the security of new SCADA systems before they are put into operation, or to evaluate systems already in use, before making changes to them that could affect overall security.

SCADA system, risk assessment, quantitative risk analysis, system security, CySeMoL, EAAT, attack vectors, security system

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

CAS — Control and Automation Solutions

CVSS — Common Vulnerability Scoring System

CySeMoL — Cyber Security Modelling Language

DMZ — Demilitarized Zone

DoS — Denial-of-service

EAAT — Enterprise Architecture Analysis Tool

FMEA — Failure Mode Effect Analysis

FTA — Fault Tree Analysis

HMI — Human-Machine Interface

ICS — Industrial Control System

IDS — Intrusion Detection System

LAN — Local Area Network

MMI — Man Machine Interface

MTU — Master Terminal Units

OCC — Operation Controls Centers

OCL — Object Constraint Language

OCTAVE — Operationally Critical Threats, Assets And Vulnerability Evaluation

P^2AMF — Predictive, Probabilistic Architecture Modeling Framework

P^2CySeMoL — Predictive, Probabilistic Cyber Security Modelling Language

PLC — Programmable Logic Controller

RTU — Remote Terminal Unit

SCADA — Supervisory Control And Data Acquisition

SMB — Server Message Block

UML — Unified Modeling Language

WAN — Wide Area Network

АСУ ТП — Автоматизована система управління технологічним процесом

ІКС — Інформаційно-комунікаційні системи

ВСТУП

У теперішній час унаслідок глобального поширення комп’ютерних систем у галузі автоматизації промислових процесів усе частіше застосовуються системи збору даних і оперативного диспетчерського управління (SCADA — Supervisory Control And Data Acquisition System). З розвитком систем SCADA збільшилась і кількість випадків атак на них. На жаль, технології, на яких побудовані сучасні SCADA-системи, орієнтовані насамперед на вирішення завдань управління технологічним процесом. Функції безпеки в них або відсутні повністю, або реалізовані за залишковим принципом. Поштовхом до покращення рівня захищеності SCADA-систем стали вразливості та проблеми, що накопичилися в області інформаційної безпеки автоматизованих систем управління технологічним процесом, що не були помічені в процесі проектування, списані на можливі ризики або ж просто проігноровані.

Потреба в терміновому розв’язанні питань безпеки, пов’язаних із системою SCADA, залишається високою. На відміну від звичайних IT-систем, більшість успішних атак на системи SCADA можуть мати серйозні наслідки для економіки країни, її стабільності і, що гірше, безпосередньо вплинути на життя людей. Так, наприклад, у 2015 році сотні тисяч українців залишилися без електроенергії майже на 6 годин через атаку на систему SCADA «Прикарпаттяобленерго», а у 2010 році мережевий комп’ютерний вірус-хробак Stuxnet завдав нищівного удару по ядерній програмі Ірану, через яку було тимчасово припинено роботи зі збагачення урану.

Актуальність роботи зумовлена широким використанням SCADA-систем в автоматизації промислового процесу. Оскільки основними вимогами до таких систем залишаються доступність та прийнятна вартість, їх захищеністю часто нехтують.

Мета роботи полягає в оцінці захищеності систем SCADA за допомогою створення моделі реальної спрощеної системи.

Завданням роботи є створення моделі системи, яка допомагає передбачити вразливості та шляхи атаки в новій системі, перед введенням її в експлуатацію, або ж у вже використовуваній системі, коли необхідно ввести в неї зміни, що можуть позначитися на рівні захищеності.

Об’єктом дослідження є сучасні системи SCADA.

Предметом дослідження є захищеність сучасних SCADA-систем.

Методами дослідження є інструменти для оцінки захищеності систем, такі як EAAT разом з CySeMoL.

Наукова новизна підтверджується тим, що в результаті досліджень була запропонована оцінка захищеності систем методом моделювання саме для систем SCADA. Метод полягає у створенні спрощеної реальної системи SCADA, до якої не вносять компоненти, що не впливають на стан захищеності, безпека якої оцінюється інструментом для моделювання EAAT разом з мовою моделювання CySeMoL.

Практичне застосування в тому, що результати роботи можуть використовуватись для оцінки захищеності нової системи, яку планують вводити в експлуатацію, або для системи, що вже певний час використовується, коли необхідно ввести в неї зміни. Запропонований метод може передбачити наслідки для безпеки різної конфігурації системи або додаткових компонентів.

1 АНАЛІЗ СТРУКТУРИ SCADA-СИСТЕМ

1.1 Призначення SCADA-систем

SCADA (англ. Supervisory Control And Data Acquisition — диспетчерське управління та збір даних) є загальним призначенням для декількох технологій, протоколів і платформ, що використовуються в автоматизованих системах управління технологічним процесом (ICS). Системи SCADA використовуються для автоматизації виробничих ліній, управління електростанціями (ядерними, термоелектричними, вітровими), управління енергосистемами (електроенергією, газом, нафтою, водою), управління очисними спорудами та багатьох інших програм.[1, с.4]

SCADA-системи зручні тим, що дають змогу обробляти інформацію в реальному часі, генерувати звіт про хід технологічного процесу, реагувати на критичні зміни польових даних. Вони відображають інформацію на екрані монітора в зручній для людини формі, що полегшує оператору логічне керування системою.

1.2 Основні компоненти SCADA

SCADA-системи зазвичай складаються з:

- Одного або більше пристроїв інтерфейсу польових даних, зазвичай RTU або PLC, які взаємодіють з пристроями контролю та комутаторами місцевого керування
- Системи зв'язку, яка використовується для передачі даних між пристроями інтерфейсу польових даних і блоками управління та комп'ютерами в центральному хості SCADA. Система може бути радіо, телефоном, кабелем, супутником тощо, або будь-якою їхньою комбінацією.

- Сервера або серверів центрального комп'ютера (іноді їх називають центром SCADA, головним пристроєм або головним терміналом (MTU))
- Набору стандартного та/або спеціального програмного забезпечення (іноді так званим програмним забезпеченням Human Machine Interface (HMI) або програмним забезпеченням Man Machine Interface (MMI)), що використовуються для забезпечення центрального хоста SCADA та оператора терміналу, підтримують систему зв'язку, а також контролюють та дистанційно керують пристроями інтерфейсу польових даних. [1, с.4]

Загальна структура SCADA-системи наведена на Рисунку 1.1. Центри управління операціями (OCC) знаходяться там, де працює система. Вони містять мережі, комп’ютери та бази даних. Весь стан системи, тобто вся інформація про клапани, датчики, перемикачі тощо, зберігається в базі даних SCADA-системи. Віддалений термінальний пристрій (RTU) є представленням керівного приводу або датчика. Оператор може через інтерфейс користувача бачити стан RTU, який зберігається в базах даних. Ця інформація зазвичай відображається на комп’ютерних моніторах і дисплеях великого екрану.

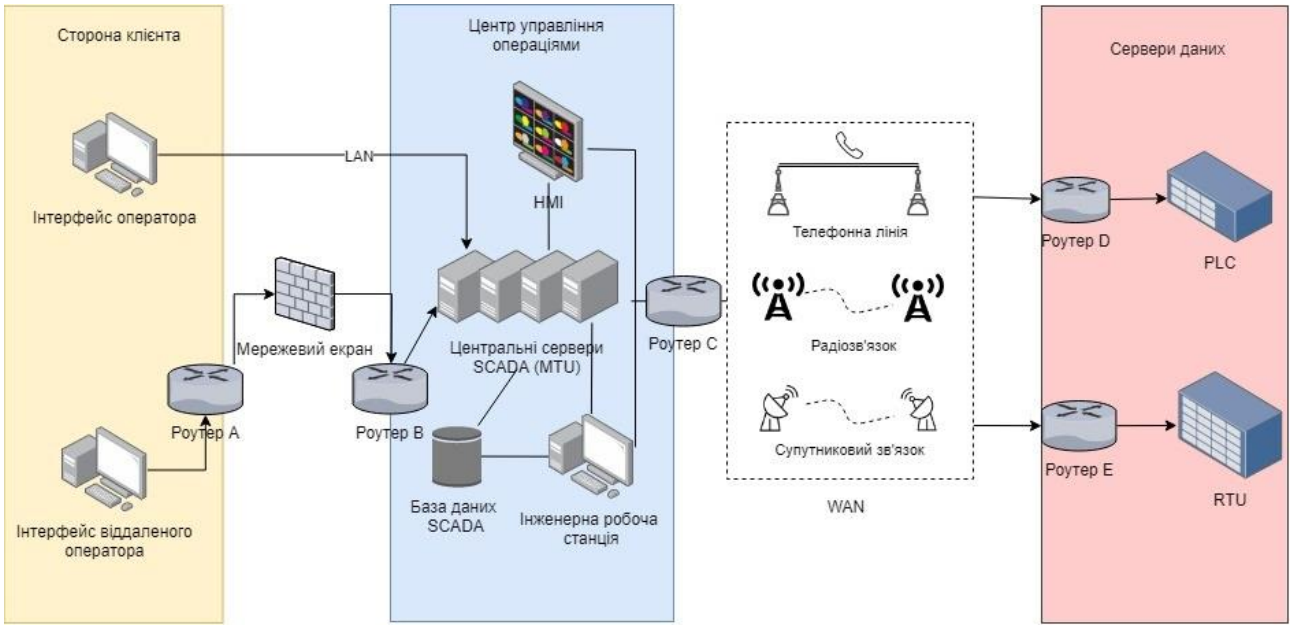

Рисунок 1.1 – Структура SCADA-системи

ОСС віддалено керує клапанами, термостатами, перемикачами та регуляторами. Це означає, що система SCADA є системою дистанційного керування. ОСС отримує звіти через мережу від RTU, які розташовані поблизу цих частин обладнання. У RTU є два завдання. Перший — приймання команд з ОСС і маніпуляція апаратними засобами. Наприклад, відкриття або закриття клапана. Друге завдання полягає в створенні даних, які є частиною збору даних. У системі SCADA також реалізована сигналізація. Ця сигналізація контролює стан системи, порівнюючи дані про систему, що зберігається в базах даних, і в дані, що приходять у режимі реального часу. Наприклад, сигнал може звучати, якщо система виявляє перегрів компонента або якщо виявлено витік. Те, про що попереджає сигнал тривоги, залежить від завдання системи SCADA. Наприклад, система, що контролює виробництво електроенергії, і система, що контролює залізничну систему, не попереджає про те ж саме. Загалом, SCADA-системи керують даними за допомогою ієрархічного методу. Тобто потоки даних з RTU агрегуються. Після цього вони надсилаються на підстанцію, яка з'єднує дані з декількох RTU, після чого ця агрегація надсилається до одного або декількох ОСС, де дані аналізуються.

SCADA-системи в ієрархії програмно-апаратних засобів промислової автоматизації знаходяться на верхньому рівні. Якщо спробувати стисло охарактеризувати основні функції, то можна сказати, що система збирає інформацію про технологічний процес, забезпечує інтерфейс з оператором, зберігає історію процесу і здійснює управління процесом в тому об'ємі, в якому це необхідно.

1.3 Передумови виникнення загроз на системи

У минулому SCADA-системи були обмежені ізольованими середовищами, відносно безпечними від зовнішнього втручання. Ці оригінальні системи були

дуже прості за своїм характером, головним чином тому, що не було задіяно жодної формальної обробки даних або механізмів пам'яті. Проте, ця простота була також головним недоліком, оскільки їх використання для чогось більшого, ніж дрібних і фізично обмежених проектів, було неможливим. Крім того, оскільки реєстрація даних була неможливою, можливості налагодження відмов були дуже обмежені.

З часом, SCADA-системи розвивалися до їх сучасного стану, з використанням розподілених топологій, пристроїв зв’язку з об’єктом (Remote Terminal Unit), програмованих логічних контролерів (Programmable Logic Controllers) і більш розвинених технологій обробки та мережевих технологій.

Системи SCADA досягли значного розвитку в їх можливості комунікації та взаємодії. Хоча оригінальні системи були ізольованими і самодостатніми за природою, вони поступово почали відкриватися до зовнішнього світу, використовуючи мережі передачі даних для власних внутрішніх цілей. Системи почали обмінюватися інформацією із зовнішнім світом або навіть з іншими системами. Поширеним став обмін даних з корпоративною локальною мережею загального призначення (LAN) для обміну інформацією з аудитом ефективності або з додатками управління запасами, підключення до глобальної мережі (WAN) для взаємодії з іншими об’єктами (наприклад, взаємодія двох чи більше електростанцій) або до центру контролю операцій, віддаленого на кілька десятків чи то навіть сотень кілометрів . Такі WAN-з'єднання можуть бути забезпечені за допомогою виділених ліній, віддаленого доступу або самого Інтернету. Крім того, виробники оригінальних приладів часто надає віддалену допомогу, використовуючи такі механізми.

Унаслідок впровадження можливостей обробки даних в системах SCADA, разом з еволюцією вбудованих систем, операційні системи також стали частиною екосистеми SCADA ICS, що розвивалася з часом. Починаючи з пропрієтарних систем, ситуація еволюціонувала до того, що почали

використовувалися похідні Windows або Unix систем разом з операційними системами реального часу, такими як VXWorks або RTLinux.

Ця еволюція принесла значні переваги системам SCADA з точки зору функціональності, раціональності та вартості. Проте вона також тісно пов'язана з деякими з найбільш важливих проблем безпеки. Поступовий перехід до більш відкритих сценаріїв, а також використання інформаційно-комунікаційних технологій та посилення прийняття відкритих, документованих протоколів виявили серйозні слабкі місця безпеки. Крім того, зростаюча тенденція до взаємозв'язку мережі АСУ з організаційною мережевою інфраструктурою Інформаційно-комунікаційних технологій і навіть із зовнішніми мережами (наприклад, для зв'язку з внутрішніми системами компанії або для віддаленого управління зовнішніми підрядниками) створила нову хвилю інцидентів інформаційної безпеки. Фактично спостерігається тенденція до зростання кількості зовнішніх ініційованих атак на системи АСУ у порівнянні з внутрішніми атаками[2, с.252-254]. Як наслідок, стара практика безпеки через неясність (англ. — security through obscurity), основна ідея якої полягає в тому, що зловмисник не знає дечого про налаштування мережі, комп'ютера або програми [3], стала неможливою.

Проте проблема безпеки в системах SCADA вже декілька років ігнорується, і навіть зараз серйозні проблеми зберігаються. Наприклад, небезпечні протоколи, такі як Modbus, все ще широко використовуються у виробничих системах. Більше того, нові функції, такі як можливості автоматичної конфігурації певного обладнання (plug-and-play), тільки погіршилися, оскільки зловмисники виявили, що це цінний ресурс для планування та виконання атак. Попри це, застарілі знання настільки поширені, що деякі менеджери процесів все ще вважають АСУ ізольованими та неявно захищеними, незважаючи на необхідність регулярних оновлень безпеки або процедур виправлення програмного забезпечення. Цим самим вони збільшують ймовірність успішних атак.

1.4 Відмінність між ІКС та АСУ

Хоча такі процедури як регулярне оновлення системи безпеки та внесення виправлень до програмного забезпечення вважаються тривіальними і є частиною регулярного технічного обслуговування в світі ІКС, вони повинні розглядатися по-іншому, коли йдеться про АСУ, головним чином з двох причин:

- Спеціалісти визначили значну кількість компонентів АСУ, які повинні працювати на постійній основі, без перерв, роками без повторної ініціалізації. Зупинка або повторна ініціалізація цих компонентів, навіть за короткий проміжок часу, може спровокувати значне збільшення витрат у промислових процесах, якими вони керують.
- Виробники обладнання повинні надзвичайно ретельно і всесторонньо перевіряти будь-який випуск програмного забезпечення до офіційної сертифікації для використання на платформах АСУ. [2, с.254-256]

Щоб протидіяти загрозам, АСУ асимілювала декілька методів, інструментів і ресурсів безпеки із світу ІКС, таких як мережеві екрани чи системи виявлення вторгнень (IDS). Хоча цей підхід забезпечує економічно ефективний спосіб підвищити рівень безпеки АСУ, він створює додаткові проблеми через відмінності в контексті (у випадку з мережевими екранами, які працюють на основі припущень, що не можуть бути застосованими до середовища АСУ). Через це такі системи потребують розробки спеціальних механізмів захисту, розроблених для відповідності експлуатаційним вимогам таких інфраструктур, але при цьому забезпечуючи адекватний захист.

Створення таких механізмів є однією з головних цілей проекту COCKPITCI. Цей європейський проект зосереджується на підвищенні стійкості та надійності критичних інфраструктур, таких як виробництво енергії та розподілені мережі, шляхом автоматичного виявлення загроз та надання інформації в реальному часі про атаки. [4, с.8]

Незважаючи на очевидні подібності між ІКС та АСУ областями, існує декілька значних відмінностей, коли мова йде про безпеку. Ці відмінності глибоко вкорінені в їхніх специфічних характеристиках. Фундаментальна причина цих відмінностей пов'язана з різним мисленням, з якими ці системи будуються. Це мислення чітко відображається на різних пріоритетах систем АСУ та ІКС. Конфіденційність та безпека мають максимальний пріоритет для мереж ІКС, за якими йдуть цілісність зв'язку та, нарешті, доступність. Для систем SCADA і АСУ в цілому властива інверсія пріоритетів, спричинена їх критичною природою, як зазначено в ISA-99 [5, с.36-37]: доступність по-перше, навіть за рахунок цілісності та конфіденційності.

Ця різниця пріоритетів, проілюстрована на Рисунку 1.2, має реальний вплив на вибір і реалізацію механізмів безпеки. Крім того, він накладає значне навантаження при імпорті механізмів безпеки зі світу ІКС до домену АСУ. Процедури, які тривіальні в світі ІКС, такі як часте виправлення та оновлення системи, можуть стати важкими або навіть неможливими в деяких сценаріях АСУ через ці відмінності. Як приклад можна вказати на неможливість або високу вартість припинення виробництва. Також загальноприйнято, що виробники системи блокують ці оновлення. Простим прикладом є той факт, що оператор критично важливого об'єкта не може встановити оновлення в операційній системі, наприклад Windows, якщо виробник програмного забезпечення SCADA не сертифікує його для оновлення.

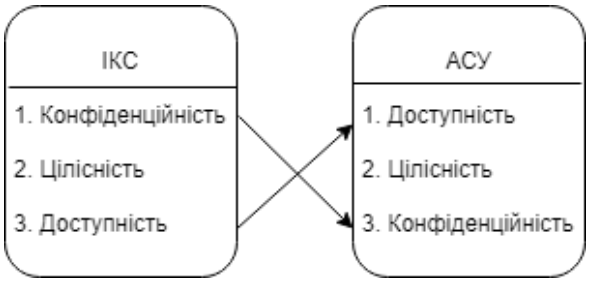

Рисунок 1.2 – Різниця пріоритетів між ІКС та АСУ

Це логічно, оскільки між новою версією Windows і програмним забезпеченням SCADA можуть існувати невідомі та небезпечні конфлікти. Однак процес сертифікації може бути досить повільним, що призводить до відставання на місяці або навіть на роки між випуском виправлень для операційної системи та її прийняттям критичними об'єктами, навіть коли стара операційна система має широко відомі небезпечні вразливості. Оператор критичних об'єктів залишається з дилемою збереження операційних систем, які точно не є безпечними, тобто ставить систему АСУ під загрозу, або втрачає гарантію та підтримку програмного забезпечення SCADA (і ризикує можливими перешкодами) шляхом виправлення операційної системи.

Іншим прикладом таких відмінностей між ІКС та АСУ є протоколи передачі даних SCADA, які відповідають за взаємодію між польовими даними та мережевими пристроями автоматизації, такими як компоненти PLC або RTU, та станції, які контролюють та моніторять їх. Одним із таких протоколів є Modbus, розроблений компанією Modicon у 1979 році, який і досі є одним із найпопулярніших протоколів для програм SCADA, головним чином завдяки своїй простоті та зручності використання. Тим не менш, Modbus страждає від відомих проблем безпеки: відсутність шифрування або будь-яких інших заходів безпеки робить цей протокол вразливим.[6, с.2-9] Незважаючи на ці відомі проблеми, протоколи SCADA, такі як Modbus, мають тривалий термін служби, і вони все ще масово використовуються.

Простіше кажучи, коли мова йде про АСУ, зрілість технологій і платформ оцінюється як неявне визнання вартості та надійності, і навіть розкриття проблем безпеки, пов'язаних з ними, очевидно, не впливає на їх використання та не спонукає до прийняття заходів для їх захисту. Це стало основною причиною багатьох питань безпеки АСУ.

1.5 Загрози SCADA-систем

Як вже зазначалось вище, промислові системи за останні десятиліття почали використовувати стандартні IBM-сумісні персональні комп’ютери, операційні системи реального часу сімейства Windows та Linux, мережеві протоколи стеку TCP/IP, доступ до мережі Інтернет та веб-браузери. Це спричинило появу нових вразливостей та розширення можливостей загроз для таких систем.

Уразливість — це недолік або слабкість у системному проектуванні, програмному забезпеченні, обладнанні або операції, що можуть бути використані зловмисником для порушення політики безпеки. [7, ст. 359]

Загрози — будь-яка обставина або подія, що може негативно вплинути на діяльність компанії (включно з її функціями, іміджем або репутацією), активи компанії або фізичних осіб через несанкціонований доступ, знищення, розкриття, зміну інформації та / або послуг.[8, ст. 135]

Мати уявлення про можливі загрози, а також про вразливі місця, які ці загрози зазвичай використовують, необхідно для вибору (або розробки) оптимальних засобів забезпечення безпеки. Загрози для систем SCADA можна класифікувати в такий спосіб[8, ст. 138-143]:

1. За джерелом загрози:

- внутрішні загрози — виникають, коли хтось надав доступ до системи за допомогою облікового запису на сервері або фізичного доступу до системи. Загроза може бути внутрішньою для організації як наслідок дії співробітника чи збою в процесі організації.

- зовнішні загрози — можуть виникати від осіб або організацій, що працюють поза компанією. Вони не мають авторизованого доступу до комп’ютерних систем або мережі. Найбільш очевидними зовнішніми загрозами для комп’ютерних систем і даних резидентів є природні катастрофи: урагани,

пожежі, повені й землетруси. Зовнішні атаки відбуваються через підключені мережі (дротові та бездротові), фізичне вторгнення тощо.

2. За агентом загрози:

- антропогенні загрози — цей клас включає загрози, спричинені діями людей, таких як інсайдери або хакери та інші.

- екологічні загрози — це загрози, викликані нелюдськими агентами. Вони походять від загроз стихійних лих, таких як землетруси, повені, вогонь, блискавка, вітер або вода, а також від тварин і дикої природи, які можуть завдати серйозної шкоди інформаційним системам. Також цей клас включає інші загрози, такі як заворушення, війни й терористичні напади.

- технологічні загрози — викликані фізичними та хімічними процесами. До них входять використання фізичних засобів для отримання доступу до заборонених ділянок, таких як будівля, серверна або будь-яка інша зона з обмеженим доступом, пошкодження апаратного та програмного забезпечення. До них також включають допоміжне обладнання для підтримки системи, наприклад, джерела живлення

3. За наміром загрози

- навмисні загрози — це навмисне завдання шкоди майну або інформації. До них відносять шпигунство, крадіжку особистих даних, вторгнення в інформаційну систему тощо.

- випадкові загрози — найчастіше викликані помилкою в коді програмного забезпечення, помилкою користувача або оператора, некомпетентності працівників.

В залежності від того, хто виступає в якості агента загрози, навмисні загрози, в свою чергу, також можна класифікувати:

– Зловмисне програмне забезпечення. Промислові системи, як і будь-які інші інформаційні системи, потенційно перебувають під загрозою зі сторони

комп’ютерних вірусів, мережевих хробаків, троянських коней, програм-шпигунів тощо.

– Хакери. Робота хакерів, в основному, направлена на здійснення DoS-атаки, що зупиняє на певний час роботу всієї системи або блокує виконання деяких операцій, на отримання доступу до системи і привілейованого контролю а також на моніторинг та аналіз трафіку.

– Інсайдери. Часто джерелом загроз стають невдоволені внутрішні користувачі, які добре знають архітектуру системи та мають безпосередній доступ до неї. Вони складають одну з основних загроз для SCADA-систем.

– Терористи. Саме цей тип загроз показує кардинальну відмінність між звичайними інформаційними системами та АСУ ТП і становить найбільшу небезпеку. Терористи зацікавлені в атаках на системи, що використовуються для контролю критичної інфраструктури, оскільки, як вже зазначалось, на відміну від звичайних IT-систем, більшість успішних атак на системи SCADA можуть мати серйозні наслідки для економіки країни, її стабільності і, що гірше, безпосередньо вплинути на життя людей.

4. За компонентами інформаційної системи:

- загрози даних
- загрози програмного забезпечення
- загрози апаратних компонентів
- загрози інфраструктури

5. За аспектом інформаційної безпеки, на яку направлена загроза:

- загрози конфіденційності — полягає в тому, що зловмисник несанкціоновано отримує доступ до конфіденційної інформації
- загрози доступності — створення таких умов, при яких доступ до послуги або інформації буде або заблокований на деякий час, що порушить

виконання операцій у системі. Саме загрози доступності вважаються найнебезпечнішими для систем SCADA

- загрози цілісності — пов’язані з імовірністю несанкціонованої модифікації певної інформації, що зберігається в інформаційній системі.

Основна частина загроз доступності — наслідки ненавмисних помилок. До них входять помилки в програмі або неправильно введені дані, що викликали вихід із ладу всієї системи. Зазвичай вони залишають уразливості в системі, якими може скористатися зловмисник. Решта загроз доступності SCADA-систем можна класифікувати за компонентами цих систем, на які націлені загрози:

- відмова користувачів;
- відмова інформаційної системи;
- відмова допоміжної інфраструктури.

У якості загрози відмови користувача виступає людський фактор. Це може бути відсутність відповідної компетенції у фахівця, або взагалі неможливість прийняття рішення через некваліфікованість фахівця.

Джерелами внутрішніх відмов системи є відступ від правил експлуатації, системні помилки, помилки адміністрування системи, відмова програмного забезпечення, відмова апаратного забезпечення, втрата даних і пошкодження апаратури.

Відмова допоміжної інфраструктури полягає в частковому або повному виходу з ладу підсистем (порушення роботи мереж зв’язку, електропостачання, системи охоронно-пожежної безпеки).

Для загроз цілісності SCADA-систем необхідний безпосередній фізичний доступ до системи. Потенційно вразливими є як дані, так і програмне забезпечення. Маючи безпосередній фізичний доступ до системи, зловмисник може без перешкод підмінити дані. Також до загроз цілісності можна віднести втрату інформації при передачі по каналах зв’язку. Часткова втрата пакетів у

мережах телекомунікації АСУ ТП може спричинити отримання хибних результатів і виконання неправильних дій оператором системи. Конфіденційна інформація (паролі, коди доступу_тощо) в АСУ ТП має технічну роль, але порушення її конфіденційності може призвести до несанкціонованого доступу до всієї системи.

1.6 Вразливості SCADA-систем

В АСУ ТП існує велика кількість вразливостей, специфічних для промислових систем, що можуть бути використані зловмисником. До них відносять[9]:

- Людський фактор. Як показують дослідження, спеціалісти, що відповідають за безпеку промислових систем, часто мають або недостатній рівень компетентності, або свідомо нехтують захищеністю таких систем заради забезпечення надійності, підвищення ефективності, доступності, розв’язання технологічних проблем, що виникають в ході експлуатації системи та мінімізацію вартості системи в цілому[8].

- Уразливості операційних систем. Як вже було зазначено, виправлення до операційної системи часто не вносяться операторами свідомо, оскільки вони не можуть вносити корективи без відповідної сертифікації оновлень. Тому часто у роботі використовують версії операційно системи з широко відомими вразливостями, задля забезпечення безперебійної роботи SCADA-систем.

- Слабка автентифікація. Використання поширених паролів є звичайною практикою для промислових систем. Системи багатофакторної автентифікації використовуються досить рідко, а інформація найчастіше передається по мережі у відкритому вигляді.

- Віддалений доступ. Для управління SCADA-системами працівники часто установлюють віддалений доступ по комутативних каналах або за допомогою VPN через мережу Інтернет. Це може привести до серйозних інцидентів інформаційної безпеки.

- Зовнішні мережеві підключення. Хоча для систем SCADA вкрай нерекомендовані зовнішні підключення, у інформаційному звіті CyberX[10] стверджується, що близько 40% АСУ ТП мають пряме підключення до інтернету. Це робить такі системи легкою мішенню для зловмисників.

- Засоби захисту та моніторингу. На відміну від інформаційно-комунікаційних систем, використання IDS і антивірусів не є поширеною практикою для індустріальних систем, оскільки вони можуть негативно впливати на вкрай важливу доступність таких систем.

- Бездротові мережі. В АСУ ТП часто використовуються різні види бездротового зв'язку, включаючи протоколи 802.11, які, як відомо, не надають достатніх можливостей щодо захисту.

- Дистанційні процесори. Певні класи пристроїв для віддаленого керування, які використовуються в промислових системах для контролю технологічних процесів, мають продуктивність, що не завжди дозволяє реалізувати функції безпеки .

- Програмне забезпечення. Програмне забезпечення промислових систем зазвичай не містить достатньої кількості функцій безпеки. Крім того, воно часто містить значну кількість архітектурних недоліків.

- Фізична безпека. Обладнання АСУ ТП може перебувати за межами контрольованої зони. В таких умовах воно не може фізично контролюватися персоналом, і єдиним механізмом фізичного захисту стає використання залізних замків і дверей, що, очевидно, не є серйозною перешкодою для зловмисників.

Основна кількість вразливостей має високий (58%) та середній(39%) ступінь небезпеки. Результати використання цих вразливостей можуть носити різний характер. Успішна атака може спричинити збій в роботі SCADA-системи, пошкодження промислового устаткування, порушення процесу виробництва продукції, зниження її якості, нанесення шкоди здоров'ю людей, флорі та фауні, порушення екологічної безпеки та охорони праці.

Висновки до розділу 1

У цьому розділі було розглянуто архітектуру SCADA-систем, передумови виникнення інцидентів інформаційної безпеки SCADA-систем та їх стан захищеності, загрози та вразливості цих систем.

Системи SCADA збирають дані з датчиків на заводі або інфраструктурному підприємстві й можуть вносити зміни віддалено для оптимізації процесу на основі отриманих даних. Ці системи контролюють низку фізичних параметрів, таких як швидкість конвеєрної стрічки, температура й тиск у резервуарі, або будь-який процес, який можна контролювати без безпосереднього втручання людини.

Вищесказане дає змогу зробити висновок, що через стрімке збільшення вимог до функціональності, надійності та вартості, захищеність таких систем, або не є пріоритетною для власників, або її надзвичайно важко забезпечити через особливості цих систем. Це зумовлює найбільші вразливості безпеки систем SCADA, що полягають у їх первинному дизайні — більшість систем, що використовується у даний час, були розроблені двадцять чи то й більше років тому й не захищені, адже тоді не враховувалась поява корпоративних мереж. Оскільки вони не призначалися для роботи в мережі, більшість систем SCADA, що використовуються у критичній інфраструктурі, не захищені належним чином.

2 ОЦІНКА ЗАХИЩЕНОСТІ СИСТЕМ

2.1 Оцінка рівня безпеки систем SCADA

Існує кілька методів оцінки рівня кібербезпеки архітектури системи і безліч чинників, які необхідно враховувати. Співробітники, відповідальні за IT-безпеку підприємства, часто мають базове розуміння архітектури системи і наслідків, які може мати порушення в системі. Однак, як показують недавні дослідження, їх технічні навички не відповідають їх соціальним і політичним навичкам, які вважаються критично важливими для їх ролі. Отже, не можна очікувати, що ці люди будуть краще розуміти слабкі місця IT-безпеки і їх відповідні залежності. Вирішенням цієї проблеми є використання консультантів, що спеціалізуються в цій області, для проведення оцінки. Однак такий підхід має деякі обмеження:

1) Результати їх роботи дійсні тільки на час їх оцінки.

2) Він дійсний тільки для розглянутих частин архітектури системи.

3) Пряма пропорційна залежність результатів від знань експерта.

Можливе рішення полягає у використанні інструменту, який може виконувати цілісну оцінку кібербезпеки в такий спосіб, щоби його було легко зрозуміти, водночас не потребуючи високих витрат. Такий інструмент був розроблений у Королівському технологічному інституті (швед. Kungliga Tekniska högskolan (KTH), англ. KTH Royal Institute of Technology) у Стокгольмі, Швеція. Цей інструмент, що називається EAAT (англ. — Enterprise Architecture Analysis Tool), використовує P^2CySeMoL (англ. — Predictive, Probabilistic Cyber Security Modeling Language). Проте постає питання, чи здатний інструмент разом з P^2CySeMoL створити оцінку кібербезпеки, яка є одночасно і вірогідною, і точною, якщо порівнювати її з оцінками, зробленими експертами з кібербезпеки. Було вирішено зробити перевірку шляхом створення загальної моделі реальної архітектури системи SCADA, захищеність якої оцінюється як P^2CySeMoL, так і

експертами. Щоб отримати порівняльні результати, метод оцінки P^2CySeMoL та експертної оцінки рівня кібербезпеки буде полягати у використанні тесту Тюрінга.

2.2 Аналіз ризиків

В аналізі ризиків виділяють три основні елементи: оцінка ризиків, управління ризиками та інформування про ризики.

Першим ключовим елементом аналізу ризику є оцінка ризику — процес, за допомогою якого оцінюється ймовірність або частота втрат, а також вимірюється або оцінюється величина збитку (наслідки).

Управління ризиками — це процес, за допомогою якого потенціал (ймовірність або частота) величини та учасників ризику оцінюються, мінімізуються та контролюються.

Інформування про ризик — це процес, за допомогою якого дані про природу ризику (очікувані збитки) і наслідки, підхід до оцінки ризику і варіанти управління ризиками поширюються й обговорюються між особами, які приймають рішення, та іншими зацікавленими сторонами.

Аналіз ризику — це оцінка потенціалу та величини будь-якої втрати і способів управління нею. Якщо є достатня кількість емпіричних даних про такі втрати, тоді ризик може бути безпосередньо оцінений зі статистики фактичних втрат. У більшості випадків дані про втрати невеликі або навіть недоступні, особливо для складних інженерних систем. Отже, аналітик повинен моделювати і прогнозувати ризик. Аналіз ризиків намагається виміряти величину втрат (наслідків), пов’язаних зі складними системами. Як правило, визначають три типи аналізу ризику: кількісний, якісний і їх поєднання. Усі вони широко використовуються залежно від цілей.

2.2.1 Кількісний аналіз ризиків

Кількісний аналіз ризику, або ймовірнісний покликаний оцінити ризик у формі ймовірності або частоти втрати. Невизначеність, пов’язана з оцінкою частоти або ймовірності виникнення небажаних подій і величини втрат (наслідків), характеризується використанням імовірнісних концепцій. Коли докази й дані недостатні, невизначеності, пов’язані з кількісними результатами, відіграють вирішальну роль у використанні результатів. Кількісний аналіз ризику, безсумнівно, є кращим підходом, коли існують адекватні польові дані, дані випробувань та інші докази для оцінки ймовірності або частоти і величини втрат. Використання кількісного аналізу ризику в останні роки неухильно зростає, насамперед завдяки доступності кількісних методів та інструментів, а також нашої здатності робити кількісну оцінку несприятливих подій і сценаріїв у складних системах на основі обмежених даних. Однак використання кількісного аналізу ризику обмежено через його складність, вартість та час виконання.

Прикладами кількісного аналізу ризиків є FTA (англ. — Fault Tree Analysis) та FMEA (англ. — Failure Mode Effect Analysis).

2.2.2 Якісний аналіз ризиків

Цей тип аналізу ризику, мабуть, найбільш широко використовуваний через свою простоту та швидкість у виконанні. У цьому типі потенційна втрата якісно оцінюється з використанням лінгвістичних шкал, таких як «низький», «середній» і «високий». Формується матриця, яка характеризує ризик у вигляді частоти (або ймовірності) втрат у порівнянні з потенційними величинами втрат у якісних масштабах. Потім ця матриця використовується для прийняття рішень в області політики й управління ризиками. Оскільки цей тип аналізу не повинен спиратися

на фактичні дані і вірогідну обробку таких даних, аналіз більш простий у використанні й розумінні. Проте недоліком є те, що такий аналіз носить суб'єктивний характер. Якісний аналіз ризиків — це метод вибору для дуже простих систем, таких як безпека одного продукту, проста фізична безпека і прості процеси.

Прикладами якісного аналізу ризиків є OCTAVE, техніка Delphi, FRAP.

2.2.3 Комбінований аналіз ризиків

Аналіз ризику може використовувати поєднання якісного й кількісного аналізу. Таке поєднання може відбуватися двома способами: частоту або потенційну втрату вимірюють якісно, а величину втрати (наслідки) вимірюють кількісно або навпаки. Крім того, можливий підхід, коли як частота, так і величина втрат вимірюються кількісно, але при розробці політики та прийнятті рішень частина аналізу спирається на якісні методи, наприклад, з використанням якісних заходів політики для кількісних діапазонів втрат. Крім того, кількісні значення ризику можуть бути доповнені додатковою кількісною чи якісною інформацією про ризик для прийняття рішення. Приклади: CRAMM, CORAS.

2.3 Оцінка ризиків

Оцінка ризику — це формальний і систематичний аналіз для ідентифікації або кількісного визначення частот чи ймовірностей і величини втрат для одержувачів через схильності до небезпек (фізичних, хімічних або мікробним агентам) від збоїв, пов'язаних із природними явищами і збоями обладнання, програмного забезпечення та людських систем. Взагалі кажучи, оцінка ризику

зводиться до розгляду трьох основних питань, поставлених Капланом і Гарриком[11].

1. Що може піти не так?

2. Наскільки це ймовірно?

3. Які втрати (наслідки)?

Відповідь на перше питання призводить до виявлення безлічі небажаних (наприклад, аварійних) сценаріїв. Друге питання вимагає оцінки ймовірностей або частот цих сценаріїв, у той час як третій оцінює величину потенційних втрат.

Це визначення підкреслює розробку сценаріїв аварій як невід'ємну частину визначення й оцінки ризиків. Сценарії ризику насправді є однією із найбільш важливих складових оцінки ризику. Розробка сценаріїв ризику починається з набору «вихідних подій» (IE — initiating events), які порушують роботу системи, тобто подій, які змінюють нормальний робочий діапазон або конфігурацію системи. Для кожної IE аналіз виконується шляхом визначення додаткових подій (наприклад, у формі апаратних, програмних або людських помилок), які можуть призвести до небажаних наслідків. Потім визначаються кінцеві ефекти цих сценаріїв (наприклад, характер і величина будь-якої втрати). Імовірність або частота кожного сценарію також визначається з використанням кількісних або якісних методів. Потім оцінюється очікувана величина збитку. Нарешті, безліч таких сценаріїв об'єднуються, щоби створити повну картину ризиків системи. Отже, процес оцінки ризику — це перш за все розробка сценарію, обчислення очікуваних наслідків від кожного можливого сценарію. Оскільки процес оцінки ризиків фокусується на сценаріях, які призводять до небезпечних подій, загальна методологія стає такою, яка дозволяє ідентифікувати всі можливі сценарії, обчислювати їх індивідуальні ймовірності й послідовно описувати наслідки, які випливають із кожного з них.

2.4 Управління ризиками

Оскільки оцінка ризику фокусується на виявленні, кількісному визначенні та характеристиці невизначеності, управління ризиками, по суті, перетворюється в спроби впоратися з такою невизначеністю. Управління ризиками — це практика, що передбачає скоординовані заходи щодо запобігання, контролю та мінімізації збитків, спричинених ризиком, зважування альтернатив та вибору відповідних заходів з урахуванням цінностей ризику, економічних та технологічних обмежень, правових та політичних питань. В управлінні ризиками використовується низка формальних методів та інструментів, включаючи аналіз компромісів, аналіз витрат і вигод, ефективність ризиків, аналіз рішень із декількома атрибутами і прогнозний аналіз збоїв (наприклад, моніторинг стану). Основна увага в управлінні ризиками упродовж життєвого циклу складної системи приділяється:

- Постійному оцінюванню ризиків (що може піти не так?)
- Визначенню, які ризики потребують максимальної уваги
- Використанню стратегії для запобігання, контролю або мінімізації ризиків
- Постійній оцінці ефективності стратегій, їх коригування за потреби

Управління ризиками є найбільш важливою й різноманітної частиною аналізу ризиків. Воно починається з точної оцінки тимчасового (миттєвого або усередненого) ризику. Оскільки конфігурація систем та інші внутрішні та зовнішні чинники також змінюються, це нормально, що значні чинники ризику також змінюються. Управління ризиками включає в себе визначення основних учасників ризику. Зазвичай складні системи показують, що застосовуються правила 80:20 або «принцип Парето». Тобто понад 80 % ризику припадає на частку менше 20 % сценаріїв ризику або елементів складної системи. Управління ризиками включає виявлення цих приблизно 20 % учасників ризику, щоби досягти максимального зниження ризику при обмежених доступних ресурсах.

2.5 Інформування про ризики

Інформування про ризики - це передача або обмін даними, інформацією і знаннями про ризик, результати оцінки ризиків та підходів до управління ризиками між особами, що приймають рішення, аналітиками і іншими зацікавленими сторонами. Інформація може стосуватися форми, ймовірності, частоти, серйозності, прийнятності, керованості або інших аспектів ризику.

2.6 Інструменти для оцінки кібербезпеки

2.6.1 CySeMoL

Фреймворк для моделювання CySeMoL (Cyber Security Modeling Language) був розроблений для забезпечення аналізу системної архітектури щодо рівня кібербезпеки заданої архітектури системи. Для виконання аналізу використовується імовірнісна модель відносин, яка визначає спосіб побудови баєсової мережі з об’єктної моделі. Баєсова мережа включає в себе заздалегідь заданий набір випадкових величин, де відносини між ними встановлюються наперед. Тому баєсові мережі часто не можуть представляти складні і великі системи. Тобто системи, де конфігурація об’єктів або їх кількість варіюється. Це пов’язано з тим, що такі мережі не здатні обробляти концепцію об’єктів. Іншими словами, вони не можуть представляти кілька схожих об’єктів у декількох умовах із загальними принципами. Модель імовірнісних відносин (PRM) розширює баєсову мережу. Вона вводить об’єкти, відносини між ними та їх властивості. Це дозволяє баєсовій мережі обробляти складні і великі системи.

Для проведення аналізу кібербезпеки системи, потрібно моделювати її архітектуру. Це включає визначення об’єктів в архітектурі системи, таких як операційні системи, сервери та персонал. Вони позначаються як активи (Assets)

в CySeMoL. Також необхідно визначити наступні атрибути для кожного об’єкта: можливі способи компрометації активу (AttackSteps) та захист активу (Defence attributes). Приклади активів та їхні атрибути можна побачити на рисунку 2.1. Прикладом може бути ввімкнений мережевий екран в операційній системі, а в AttackStep — DoS. Тут оборона полягає в захисті активу, а AttackStep є можливим способом компрометації активу. Ці Активи з’єднані один з одним, щоби представляти архітектуру системи, яку вони повинні моделювати.

Зловмисник (Attacker) в CySeMoL визначається як професійний тестер на проникнення, який має доступ до загальнодоступних методів та інструментів. Він підключений до активу, який має можливі шляхи компрометації, тобто можливу точку входу. Підключення зловмисника до такого об’єкта означає джерело атаки на систему.

Створення моделі для CySeMoL складається з двох етапів. Першим кроком є визначення якісної структури, тобто які активи повинні бути включені, їх захист і шляхи атаки. Другим кроком буде додавання кількісних даних у цю якісну структуру, які визначають, наскільки ймовірно, що атака буде успішною, з урахуванням зазначених захистів.

Використання літератури, а також матеріалів від експертів у предметній області послужило основою для визначення того, які типи захисту й атаки повинні бути включені в CySeMoL. Залежно від змодельованої архітектури системи, CySeMoL обчислює умовні ймовірності для успіху атаки. Однак важливо визнати, що підсумкові розрахунки слід розглядати як опорні факти, а не жорсткі статистичні дані. Це пов’язано з низкою причин. По-перше, умовні ймовірності, створені CySeMoL, потребують оновлення, оскільки дані про захисників і нападників змінюються з часом. По-друге, CySeMoL більше орієнтований на жорсткі технічні аспекти безпеки, ніж на більш м’які, такі як соціальна інженерія або фізичні атаки, які є досить базовими в системах SCADA. Зловмисник діє суб’єктивно й піддається великим відхиленням. Тому важко

зробити висновок, що розрахунки можна представити для різних типів зловмисників, наприклад, для досвідченого зловмисника та новачка.

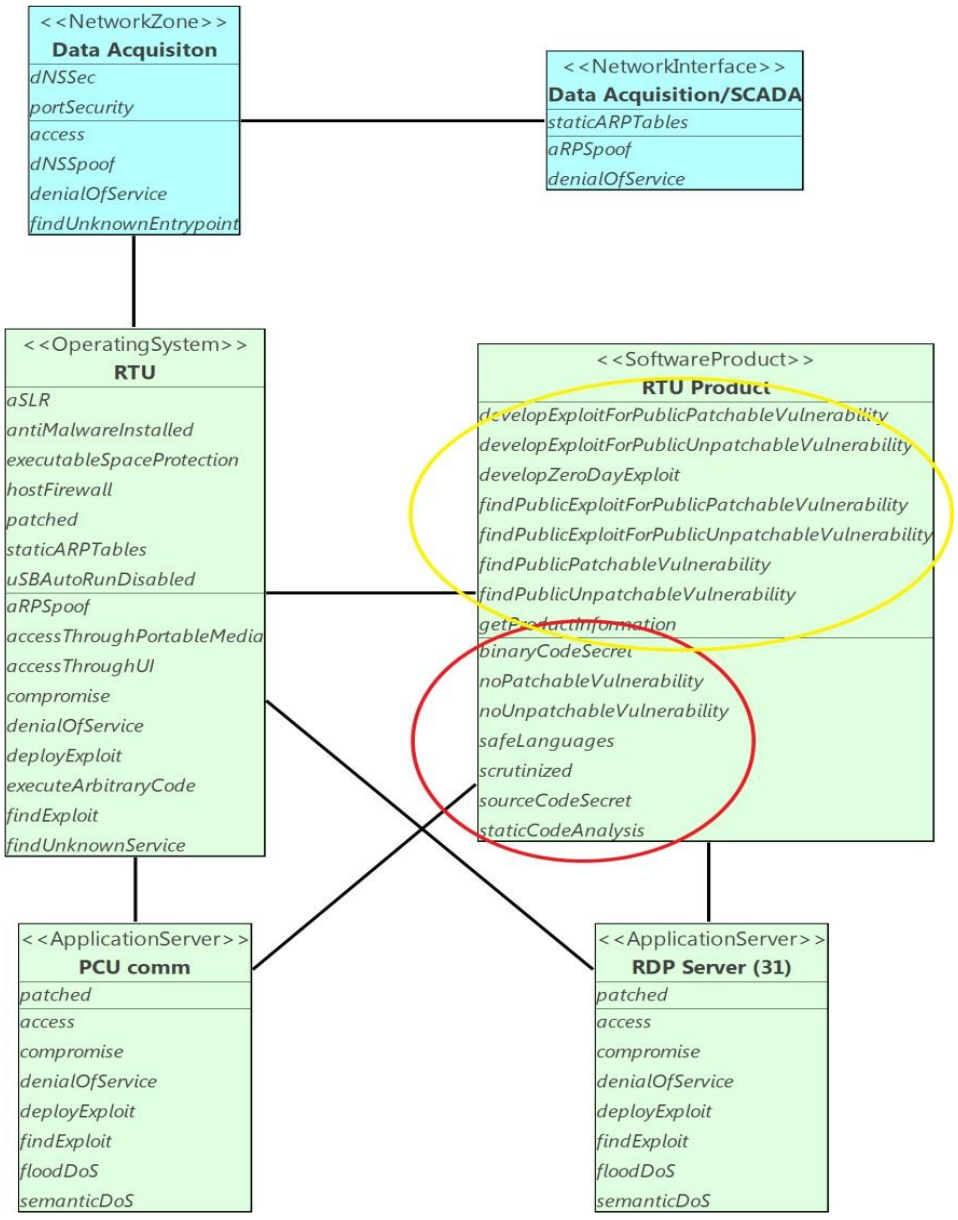

Рисунок 2.1 – Модель CySeMoL. Верхній жовтий овал позначає кроки атак, а нижній червоний – захисні атрибути

2.6.2 P2AMF

P^{2}AMF (Predictive, Probabilistic Architecture Modeling Framework) — це фреймворк, що використовується для аналізу систем програмного забезпечення універсального типу. Сьогодні UML є домінантною концепцією для моделювання програмного забезпечення. В основному всі інструменти проектування, що використовуються для програмної архітектури, підтримують UML-моделювання або ґрунтуються на ньому. Тому для якісного аналізу загальна структура буде виграшною від сумісності з UML. Для аналізу архітектури системи зазвичай використовується OCL, де його призначення варіюється від аналізу продуктивності до аналізу безпеки, а також аналізу впливу. Він також сумісний з UML, що є перевагою. Чого, однак, не вистачає OCL, це здатності вловлювати невизначеність, яка стає все більш важливою, коли йдеться про сучасні програмні системи. P^{2}AMF має здатність, на відміну від UML-OCL, висловлювати цю невизначеність у моделях UML. Невизначеність містить об’єкти, відносини й атрибути в UML і здатність виконувати імовірнісні обчислення. Хоча P^{2}AMF і OCL-UML схожі, вони відрізняються в деяких аспектах. P^{2}AMF використовується для створення підтримки прийняття рішень для макетної системи або реальної системи, у той час як OCL в основному використовується на етапі проектування для вказівки обмежень на майбутні реалізації. Інша відмінність полягає у використанні діаграм об’єктів. Діаграма об’єктів в P^{2}AMF включає в себе важливий аспект, унікальний для P^{2}AMF — на об’єктній діаграмі виконується імовірнісний висновок.

2.6.3 MulVAL

Формування моделі графа атак в MulVAL здійснюється в три етапи. Для початку вивчається набір інформації, що стосується всіх даних, які можуть бути корисними для побудови графа атак, а потім дані збираються за допомогою сканерів вразливостей мережі й результатів, які вони генерують. Інформація, зібрана сканером вразливостей мережі, складається з даних про структуру мережі, підключених хостів і служб. На другому етапі будується графік із використанням зібраної інформації про вразливість разом з інформацією про експлойти, виявлені в різних базах даних. Імовірність отримують із думки експертів, а також з CVSS — загальної системи оцінки вразливостей, яка дає значення складності доступу. Ця інформація аналізується і проводиться розрахунок шляхів атаки і їх успішності. Кожній атаці призначається ймовірність, що ґрунтується на вразливості, з якої вона походить. Такі атаки, як соціальна інженерія або атаки нульового дня, і те, наскільки добре засоби захисту діють для пом’якшення певних загроз, повинні вводитися користувачем вручну. Це зумовлено тим, що сканери вразливостей не можуть надати цю інформацію. Вихідні дані подаються в графічній структурі.

2.6.4 NetSPA

NetSPA працює аналогічно MulVAL в тому сенсі, що збір даних здійснюється за допомогою сканерів мережевих вразливостей. Однак вони по різному ставляться до можливості зловмисника використовувати вразливість. У NetSPA всі виявлені вразливості обробляються так, ніби вони можуть бути використані зловмисником. Відсутня будь-яка кореляція між поведінкою заходів безпеки, які застосовуються на цільовому вузлі, і компетенцією зловмисника.

Також не враховується необхідність конкретної конфігурації системи для використання вразливості. Це може представляти проблему, оскільки сканер вразливостей не може розпізнати, чи може бути вразливість використана. Передбачається, що всі сервери уразливі для NetSPA, оскільки він моделює експлойти нульового дня [12].

2.6.5 TVA

Інструмент TVA схожий на два вищезгадані в даних, які він повинен збирати, і в результатах, які він видає. TVA використовує сканери вразливостей для заповнення своєї мережевої моделі, але замість бази даних вразливостей він використовує базу даних експлойтів, відомих зловмисникам. Щоб описати, коли експлойт може бути застосований і яким буде стан системи після його застосування, експлойт пов’язується з умовами (Jajodia et al., 2005). Symantec DeepSight — це база даних, яка використовується для визначення цих умов.

2.7 Методи експертної оцінки системи

Експертна оцінка — це процес надання системі певної оцінки, яка ґрунтується на думці експертів в області, з метою прийняття подальших рішень щодо системи. Якщо існує складна проблема, для вирішення якої недостатньо емпіричних даних або яку не можна вирішити за допомогою математичної статистики, її долають, використовуючи думку компетентних фахівців. Існує кілька методів експертної перевірки системи.

2.7.1 Тестові випадки

Тестові випадки є одним із домінантних методів для перевірки системи. Вони включають випадки, які були раніше вирішені і які оцінюються з використанням системи й порівнюються з попередніми рішеннями, або нові випадки, які повинні бути вирішені системою й експертом, а згодом порівняні. При використанні тестових випадків виникає проблема, яка полягає припущенні, що експерт завжди правий. Оскільки системне рішення порівнюється з рішенням експертів, були випадки, коли експерт неправильно тлумачив завдання і виробляв гірше рішення, ніж якщо б він інтерпретував його правильно. Коли система порівнювалася з цим рішенням, її результати були набагато кращими. Ще один недолік полягає в тому, що при відсутності доступних тестових випадків і створенні їх вперше для дослідження можуть бути створені такі, що відображають тільки сильні сторони системи.

2.7.2 Тест Тюрінга

У тесті Тюрінга рішення, створені людиною й машиною, оцінюються третьою стороною, щоби визначити, хто і яке рішення створив. Для перевірки системи системні рішення порівнюються з рішенням експерта-людини стороннім експертом. Для об’єктивних результатів, рішення повинні бути замасковані, щоби сторонній оцінювач не міг визначити, ким було створене це рішення. Для багатьох систем тест Тюрінга є найбільш підходящим методом перевірки. Він особливо корисний, коли система повинна бути перевірена кількома експертами, а продуктивність експертів різна, і коли розробнику важко зробити висновки про те, чим системні рішення відрізняються від експертного.

2.7.3 Моделювання

Підключення системи до імітаційної моделі може бути схожим на тестові випадки. Кожне моделювання може розглядатися як тестовий випадок, зміна параметрів для моделювання призведе до різних результатів для кожного прогону. Цей метод дуже ефективний для перевірки простих детермінованих імітаційних моделей. Однак проблема в тому, що моделювання ґрунтується на моделі та не є ідеальним. Тому можуть виникнути проблеми з надійністю і точністю. Іншими словами, модель, яка добре працює в симуляції, може не поводитися так само добре в реальній системі.

2.7.4 Контрольні групи

Багато системи покладаються на взаємодію з людиною для розв’язання проблеми. У цьому випадку тест Тюрінга можна об’єднати з контрольними групами для перевірки. Завдання, які необхідно вирішити, представлені для двох груп: одна з доступом до системи, а інша — без. Частина перевірки тепер така ж, як у тесті Тюрінга, але очікується, що група з доступом до системи повинна перевершити групу без доступу.

При використанні контрольних груп можуть виникнути деякі проблеми. Наприклад, гіпотетично, групи могли б отримати різні результати, хоча жодна не мала доступу до системи. Дослідження дасть неправильний результат, визнаючи одну групу кращу за іншу, хоча вони обирались як свідомо однакові. Або якщо система складна в експлуатації і вимагає навчання, це може бути не вигідно для групи, що використовує її в дослідженні. Тобто група, яка має доступ, не отримає кращий результат, ніж група без доступу, оскільки для отримання вигоди із системи може знадобитися її інтенсивне використання протягом певного періоду часу.

2.7.5 Аналіз чутливості

Аналіз чутливості перевіряє, як реагує система на різні вхідні дані. При його використанні передбачається, що існує один випадок X, у якому відомі проміжні результати, остаточні результати й аргументація. Змінюючи вхідні дані, експерт може визначити, є отримані результати достовірними чи ні. Один із більш простих підходів полягає в тому, щоби знайти випадки, коли зміни у вхідних даних не повинні призводити до будь-яких відмінностей у результаті. Загальна проблема цього методу полягає в тому, що не можна очікувати, що цей метод охопить весь вхідний набір станів.

2.8 Класифікація джерел даних в галузі програмного забезпечення

Збір даних можна розділити на три ступені [13].

1. Перший ступінь — коли дослідник збирає дані в режимі реального часу від безпосередньої взаємодії зі співбесідником.
2. Другий ступінь — це непрямі методи, де збір даних дослідником здійснюється безпосередньо, але без взаємодії з респондентами під час збору даних. Прикладом може бути збір даних, де використовується програмний інструмент, використання якого контролюється через відеозапис.
3. Третій ступінь — це методи, де дані збираються дослідником, аналізуючи вже написані роботи. Наприклад, дослідник збирає дані про систему, посилаючись на її специфікації та інструкції.

Загалом, збір даних першого ступеня завжди є дорожчим, ніж другого або третього. Це пояснюється тим, що він вимагає більше роботи як від вченого, так і від співрозмовника. Однак він також володіє перевагами, оскільки можна контролювати зібрані дані. Інші два ступені не дають стільки свободи контролю

над збором даних. Наприклад, при використанні методу третього ступеня, дослідник обмежується доступною інформацією [13].

2.9 Вибір інструменту для оцінки захищеності системи

Проблема, яку повинні вирішувати всі методи, — це складні графи, які створюються при аналізі систем реалістичних розмірів. Крім того, вони повинні управляти циклами на графах. Інша проблема полягає в отриманні вхідних даних. MulVAR, NetSPA та TVA використовують сканер вразливостей Nessus для збору цих даних. Однак недавній тест на точність показує, що Nessus пропускає більш ніж половину вразливостей при наданні облікових даних для доступу до хостів в мережі і чотири з п'яти вразливостей, коли облікові дані не надаються [14]. Таким чином, автоматичне сканування, на якому засновані ці інструменти, ненадійне, коли необхідно виявляти окремі уразливості. Крім того, в середовищах з чутливими системами, зокрема SCADA-системах, слід уникати сканерів, оскільки вони можуть переривати важливі системні служби [14]. Іншим недоліком наявних інструментів є тип атак, які вони охоплюють. Інструменти розроблені для програмних експлойтів, призначених для служб, що працюють на портах прослуховування машин. Таким чином, вони не здатні моделювати багато типів атак, наприклад злом паролів, соціальну інженерію і DoS-атаки. NETSPA була розширена для включення атак на клієнтів (наприклад, веб-браузери) [15]. Інструменти також не мають можливості передбачати атаки нульового дня, тобто атаки з використанням вразливостей, які невідомі широкому загалу. Звісно, користувач інструменту може ввести гіпотетичні дані в базу даних і виконати аналіз з ними. Проте, потрібна компетентність, щоб визначити, які атаки нульового дня можна очікувати від зловмисника. CySeMoL також моделює атаки і оцінює ті, які може виконати зловмисник. У порівнянні з трьома інструментами, розглянутими вище, CySeMoL аналізує ширший спектр типів

атак і заходів безпеки. Висновок CySeMoL є імовірнісним, як і в MulVAL. CySeMoL оцінює ймовірність того, що різні атаки можуть бути здійснені проти активів в архітектурі системи. Ймовірності, використані в цих розрахунках, були отримані з експериментальних досліджень і досліджень, заснованих на думці експертів по предметної області.

Ґрунтуючись на аналізі ефективності [16], який використовував тест Тюрінга, і враховуючи вищенаведені недоліки інших інструментів, можна зробити висновок, що CySeMoL показав найкращі результати.

У цьому дослідженні [16] були використані п'ять експертів і три новачки в області. Всього було дев'ять учасників, у тому числі CySeMoL. Результати дослідження показані в Таблиці 2.1. Оцінка надавалась по п’ятибальній шкалі. У цьому дослідженні також були два оцінювачі, які аналізували шляхи атаки створені експертами і новачками.

Таблиця 2.1 — Результати експертної оцінки CySeMoL[15]

	Оцінювач1	Оцінювач 2	Середнє	Медіана
Експерт 1	[2,4,3,2,2,2,5,4,3]	[4,4,3,4,4,2,4,4,4]	3.3	4
Експерт 2	[4,4,2,2,4,2,3,2,1]	[4,4,3,3,4,2,2,4,3]	2.8	3
Експерт 3	[2,4,3,4,3,3,3,4,3]	[4,2,4,5,3,4,2,4,3]	3.3	3
Експерт 4	[4,1,4,2,2,3,4,4,4]	[4,2,4,3,3,3,3,4,3]	3.2	3
Експерт 5	[2,2,2,1,1,1,2,2,2]	[2,2,2,2,2,2,2,2,2]	1.8	2
CySeMoL	[2,2,3,1,2,2,3,3,2]	[5,5,4,3,4,2,1,4,2]	2.8	2.5
Новачок 1	[2,4,3,1,2,2,2,3,2]	[2,3,2,2,2,2,2,2,2]	2.2	2
Новачок 2	[1,2,4,1,2,2,2,1,1]	[3,3,3,4,2,2,3,2,2]	2.2	2

З Таблиці 2.1 видно, що CySeMoL виступала краще, ніж всі новачки, краще, ніж один експерт, і зрівнялася з іншим експертом. Троє експертів виступали краще, ніж CySeMoL. Тобто CySeMoL займає позицію 4-5 з 9, разом з Експертом 2.

Середнє значення всіх середніх балів становило 2,7, а медіана – 2,5. Так як CySeMoL мав середнє значення 2,8 і медіану 2,5, то його показники кращі, ніж середнє значення і рівні медіані цього дослідження.

Висновки до розділу 2

У цьому розділі були проаналізовані різноманітні інструменти для оцінки захищеності системи, а також методи експертної оцінки цих інструментів. На основі дослідження[15] було зроблено висновок, що найбільш ефективним для оцінювання захищеності SCADA-системи буде інструмент для системного аналізу EAAT разом з P^2CySeMoL. Вони можуть створити модель архітектури системи й оцінити рівень кібербезпеки — ймовірність того, що зловмисник отримає доступ до захищених ресурсів. Розробники цього інструменту були зацікавлені в оцінці його продуктивності з погляду рівня точності прогнозованих розрахунків, зроблених ним. Для компаній, що пропонують продукти для управління технологічним процесом, де цілісність системи дуже важлива, допомога такого інструменту була б надзвичайно корисною. Це б дозволило їм створювати й перевіряти рівень кібербезпеки, властивий архітектурі системи, до її впровадження й тестування на проникнення. Так недоліки можуть бути легко виявлені і виправлені на етапі проектування системи. Ще одна особливість інструменту полягає в тому, що він дозволяє спрогнозувати можливі наслідки конкретної архітектури системи.

3 МОДЕЛЮВАННЯ СИСТЕМИ БЕЗПЕКИ SCADA-СИСТЕМИ ЗА ДОПОМОГОЮ EAAT ТА CYSEMOL

3.1 Основні елементи CySeMoL

Модель CySeMoL була створена в Королівському технологічному інституті Швеції [17] і далі розвивається Foreseeti[18] в повністю інтегрований інструмент моделювання загроз. Використовуючи CySeMoL для імітації відомих попередніх атак, можна одночасно протестувати модель і знайти області, які можна поліпшити.

CySeMoL використовується для створення моделей системи. Інструмент реалізований у програмному засобі - Enterprise Architecture Analysis Tool (EAAT), що забезпечує зручну взаємодію для моделювання та аналізу. Модель представляється у вигляді графа з вузлами, що відповідають компонентам системи, і ребрами, що показують як ці компоненти зв’язані між собою. Ребра можуть бути різних типів, створюючи різні умовні відносини між вузлами, навіть між однією й тією ж парою.

Кожен вузол у графі CySeMoL має кілька атрибутів, що належать до однієї з двох категорій: «кроки атаки» й «захист». Кожен такий атрибут є вузлом у баєсовій мережі, і їх умовні залежності ґрунтуються на моделі CySeMoL. Наприклад, підграф CySeMoL, показаний на Рисунку 3.1, має три вузли і два ребра. Коли необхідно виконати обчислення, вони перетворюються в баєсову мережу з 27 вузлами й ще більшим числом ребер.

Відносини в мережі засновані як на попередніх дослідженнях, так і на оцінках експертів в області з використанням методу Кука [17]. Об’єднавши кроки атаки й зосередивши увагу на більш конкретних частинах системи, CySeMoL допомагає абстрагуватися від багатьох деталей графа атак, дозволяючи користувачеві зосередитися на компонентах системи, а не на деталях точного методу атаки. Наприклад, існує безліч різних способів, якими сервер може бути скомпрометований, що призводить до того ж результату, але

користувачеві потрібно лише визначити сервер з погляду його операційної системи, іншого програмного забезпечення і відносин із навколишнім світом.

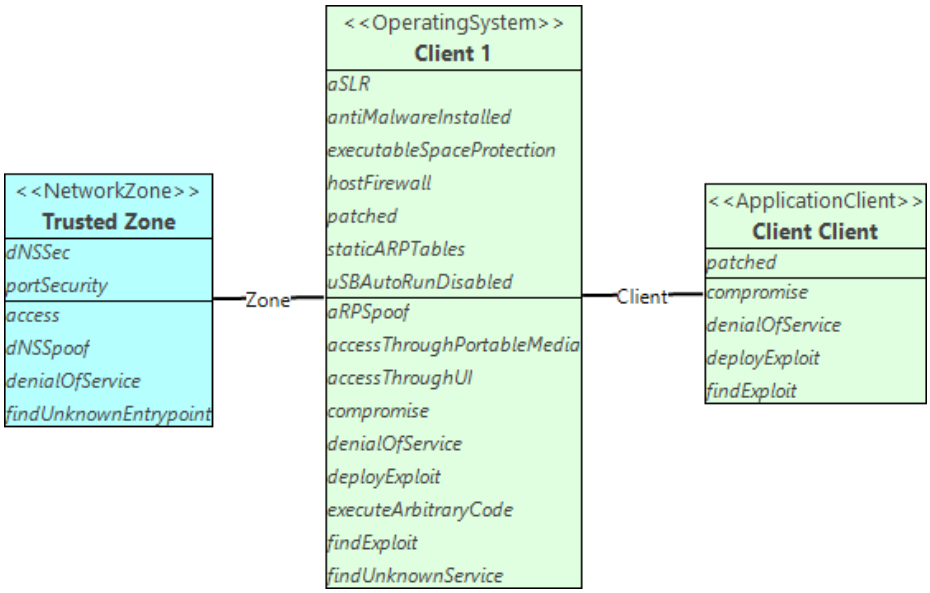

Рисунок 3.1 – Частина моделі CySeMoL

Як зазначено вище, фактичні розрахунки виконуються в баєсовій мережі, виведеної з моделі CySeMoL. Коли необхідно обчислити стан одного вузла, алгоритм повторюється на всіх вузлах, від яких він залежить, й обчислює ті, які забезпечують збереження результатів. P^{2}AMF, з іншого боку, використовує OCL для створення прямого алгоритму, дуже схожого на алгоритм найкоротшого шляху Дейкстри, з додатковим обмеженням логічних вузлів AND. Деякі вузли можуть бути пройдені тільки при виконанні двох або більше умов.

CySeMoL використовує вибірку по методу Монте-Карло з алгоритмом прийняття-відхилення або алгоритмом Метрополіс-Гастингса [17] для створення дійсних станів вузлів. Алгоритм прийняття-відхилення в основному генерує безліч рівномірно розподілених вибірок по всьому простору вибірок, а потім видаляє (відхиляє) такі, що не підходять. Нарешті, коли вузли з доказами були відібрані за допомогою алгоритму P^{2}AMF, CySeMoL обчислює результат. Результатом є граф, на якому «локальні» умовні залежності були перетворені в

ймовірності, що позначають ризик того, що зловмисник досягне успіху в цьому конкретному етапі атаки деяким шляхом через мережу. Також можна зобразити зворотні висновки, щоби показати, які попередні кроки впливали на результати певного кроку атаки. Застосовуючи зворотний висновок із даного етапу атаки назад до зловмисника, можна відстежити найбільш ймовірні шляхи атаки в мережі. Це може допомогти у виявленні проблемних частин мережі, які особливо схильні до атак. Такі частини вимагають додаткової уваги та ресурсів у тому разі, коли система вже введена в експлуатацію, або перепроектування в разі аналізу проекту.

Коли обчислення завершені, результати надаються користувачеві шляхом теплового кодування всіх шаблонів і кроків атаки за шкалою від 0%: зелений - 50%: жовтий - 100%: червоний. Тут відсоток відповідає ймовірності того, що один або кілька зловмисників пройдуть етапи атаки в об'єктній моделі за час, призначений для атаки. Загальна схема теплового забарвлення представлена на Рисунку 3.2[14].

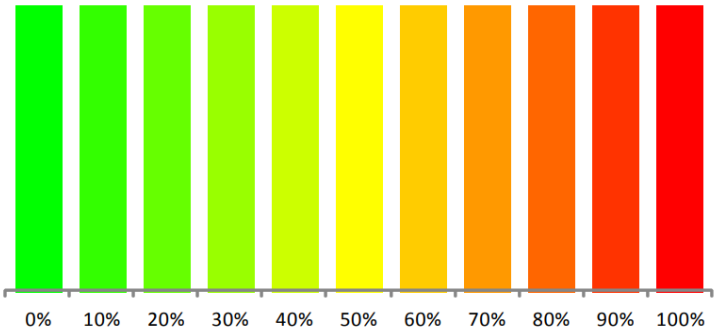

Рисунок 3.2 – Схема теплового забарвлення результатів обчислення CySeMoL (відсотки відповідають ймовірності успіху атаки)[14]

3.2 Моделювання спрощеної системи SCADA Siemens

Оскільки детальний опис всіх компонентів реальної SCADA-системи становить інформацію з обмеженим доступом, було вирішено взяти за основу систему, на яку було здійснено серйозну атаку, оскільки в такому випадку існує багато досліджень та статей на цю тему. Після вивчення атак на SCADA-системи, вибір впав на одну з найвідоміших за весь час атак – Stuxnet, яка вразила і порушила операції в ядерних установах Ірану [19].

Stuxnet - це комп'ютерний черв’як, який був виявлений в 2010 році. У той час він вважався одним з найскладніших шкідливих програм, коли-небудь створених. Зразки черв’яка були ретельно проаналізовані дослідниками [19]. Мета Stuxnet полягала в тому, щоб заразити програмований логічний контролер (PLC) в промислових системах. Зокрема, вважається, що мішенями були системи Siemens SIMATIC WinCC SCADA на ядерних об'єктах Ірану.

Хоча точно невідомо, що саме сталося на цьому конкретному об'єкті і як поширювався черв'як, існує кілька моделей атаки, заснованих на еталонних системах і специфікаціях передового досвіду [19]. На підставі результатів цього дослідження була змодельована мережа, яка може бути схожа на об'єкт і є типовою для мереж SCADA.

Цю опорну систему можна побачити на Рисунку 3.3.[19] На ній показані чотири основні частини мережі, розділені на п'ять мережевих зон. Внизу зображення показано ядро мережі з мережею управління технологічним процесом (Process Control Network) і мережею системи управління (Control System Network). Остання є зоною, що містить фактичний PLC. У верхній частині зображення знаходиться мережа управління підприємством (Enterprise Control Network), типова офісна мережа, з якої виконуються повсякденні операції. Це може бути фізично відокремлена від більш внутрішніх зон і підключена через WAN. Демілітаризована зона (Perimiter Network) дозволяє деяким даними

переміщатися між внутрішньою мережею і мережею управління підприємством (Enterprise Control Network.).

На основі опису топології мережі і потоків даних була створена модель CySeMoL. Незважаючи на те, що мережа досить мала і деталі зведені до мінімуму, отримана модель складається з близько 80 вузлів. Модель Siemens була створена як модель CySeMoL за допомогою EAAT, що дозволяє розбити модель на різні відображення даних (Views), щоб зробити її більш керованою.

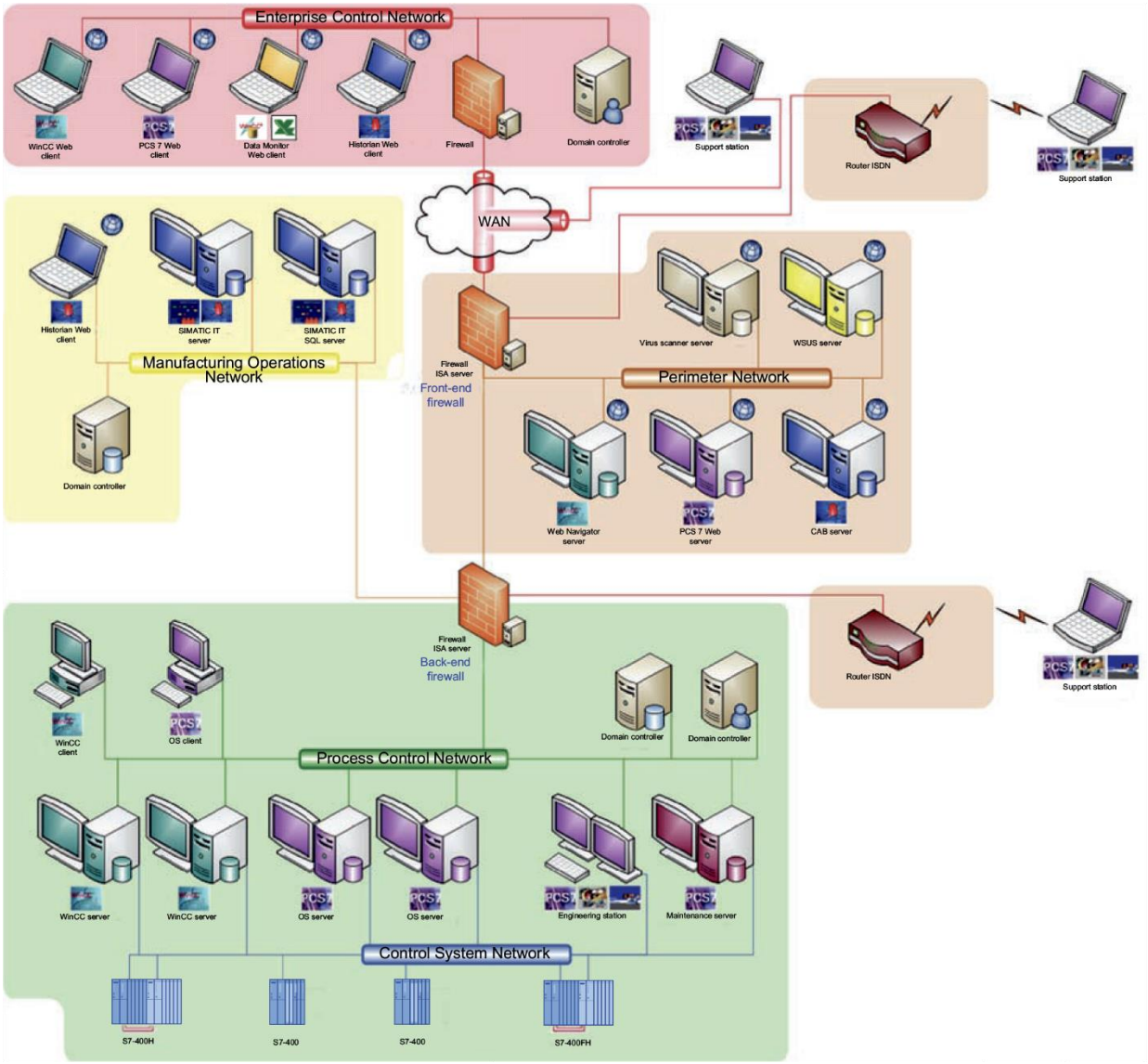

Рисунок 3.3 – Опорна схема SCADA-системи [19]

В Додатку А наведені всі частини моделі Siemens. Вони включені для повноти і кращого розуміння того, як виглядає модель CySeMoL. В цілому передбачається, що мережа застосовує хороші заходи безпеки зі строгими правилами мережевого екрану, а програмне забезпечення регулярно оновлюється. Частина створеної моделі показана на Рисунку 3.4. Це відображення даних моделі CySeMoL показує загальну мережеву топологію системи, виключаючи будь-які робочі станції. Також, як зазначалося вище, CySeMoL дозволяє коригувати конкретизацію кожної частини схеми системи. На Рисунку 3.5 наведена та ж мережева топологія створеної системи зі всіма атрибутами.

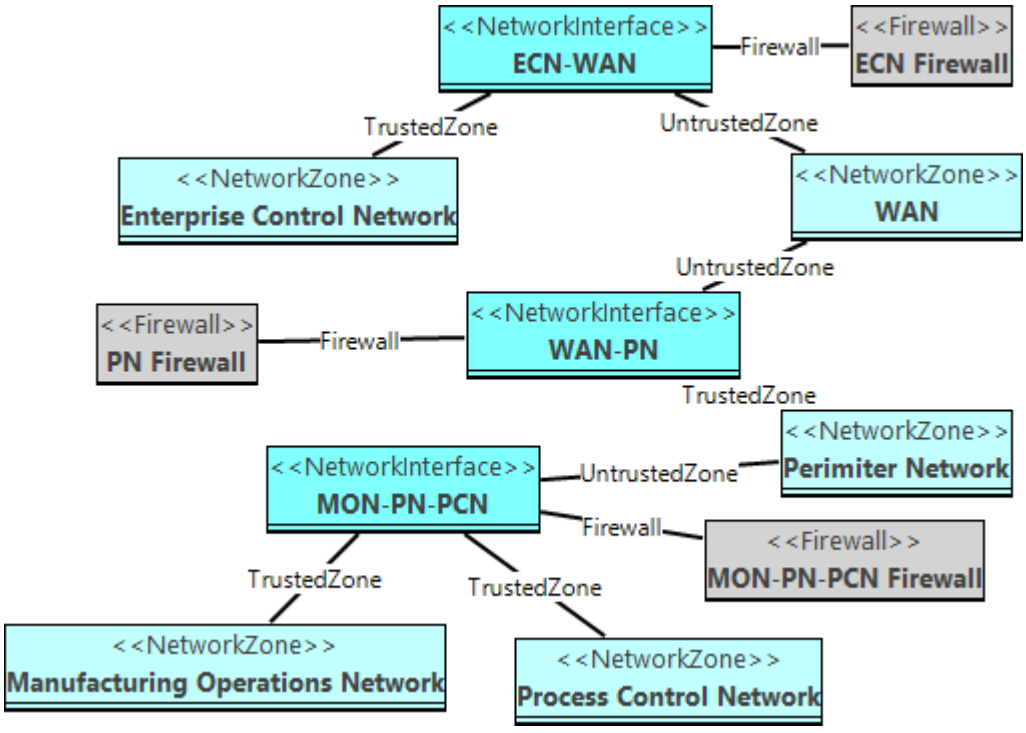

Рисунок 3.4 – Мережеві зони, інтерфейси та мережеві екрани моделі Siemens

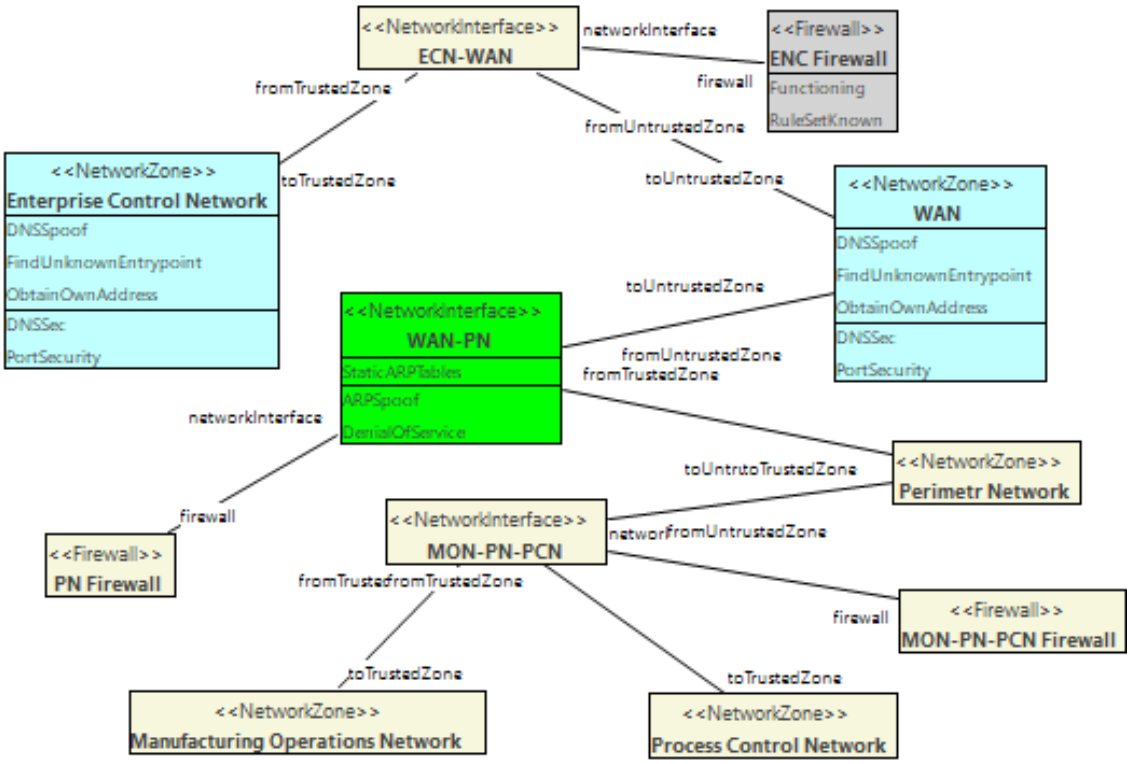

Рисунок 3.5 – Мережева топологія моделі Siemens з розгорнутими атрибутами

3.3 Аналіз змодельованої системи SCADA Siemens

Створена модель аналізувалась на робочій станції Microsoft Windows 10 Pro з процесором Intel Core i7-3632QM, 2201 МГц, з 4 ядрами та 8 логічними процесорами та з 16 Гб оперативної пам’яті. Оскільки модель вийшла досить велика, обчислення результатів зайняли близько 32 годин. Процес оцінювання захищеності наведений на Рисунку 3.6.

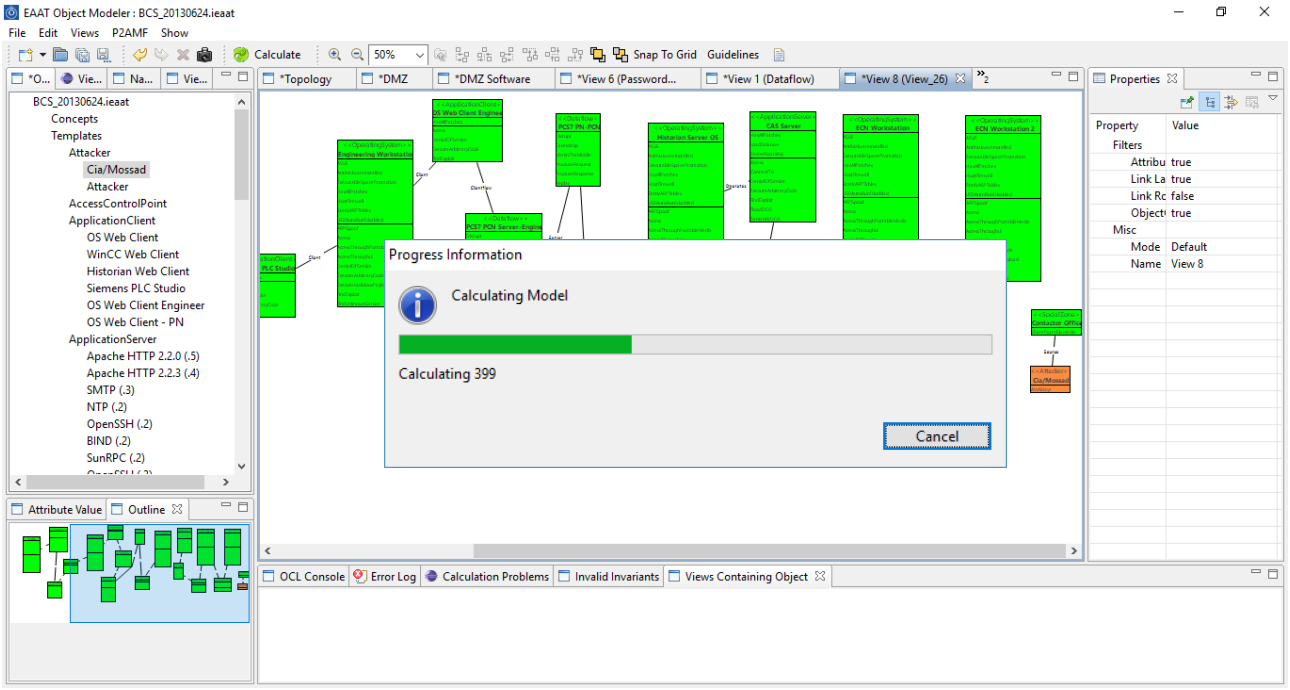

Рисунок 3.6 – Процес оцінювання захищеності системи Siemens

З моделі був розрахований шлях атаки до одного з PLC. CySeMoL досить докладний, і в результаті шлях атаки від атакуючого до PLC містить багато кроків. Крім того, CySeMoL генерує не один шлях атаки, а цілий граф атак, тому можливі кілька варіантів шляхів атаки.

Шлях атаки також зображений на графіку CySeMoL, показаному на Рисунках 3.7, 3.8 і 3.9. Червоні стрілки вказують всі властивості, які впливають на значення вузла, в той час як сині стрілки показують шлях атаки. Зображення громіздкі і їх може бути складно розшифрувати, особливо в вузлах операційної системи. Саме для кращого розуміння нижче наведені кроки атаки в письмовому вигляді.

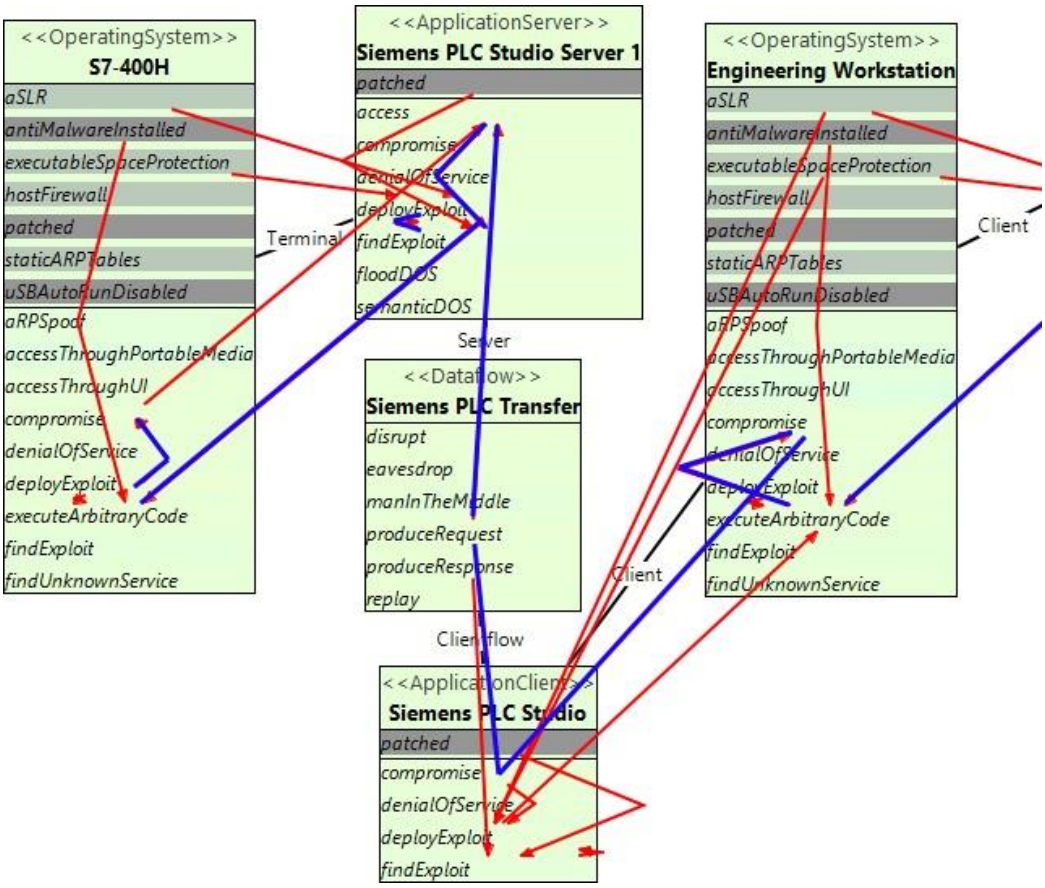

Рисунок 3.7 – Атака на SCADA систему Siemens, обчислена CySeMoL, частина 1

1. Attacker.start, Contractor Office.sharePortableMedia — цей крок атаки стосується можливості того, що підрядники в одній соціальній зоні ділять між собою портативні носії (наприклад, USB-накопичувач).

2. ECN Workstation 2.accessThroughPortableMedia, ECN Workstation 2.executeArbitraryCode, ECN Workstation 2.compromise — крок вказує на можливість отримання доступу до робочої станції ECN Workstation 2 за допомогою переносних носіїв, наприклад, за допомогою експлойтів, що автоматично запускаються, або спеціально створених файлів. Це може дозволити зловмиснику заразити всю мережу управління підприємством. У CySeMoL AccessThroughPortableMedia має значення FALSE, якщо USBAutoRunDisabled має значення TRUE.

3. Enterprise Control Network.access, ECN Workstation.findUnknownService, ECN Workstation.findExploit, ECN Workstation.deployExploit, ECN Workstation.executeArbitraryCode, ECN Workstation.compromise — якщо зловмисник зможе знайти служби, невідомі адміністратору мережі, що працюють на хості, можна атакувати їх для отримання привілеїв на ньому. Оскільки невідомі служби, ймовірно, мають більше вразливостей, вони є серйозними проблемами безпеки. Якщо зловмисник знайде таку вразливість, він може знайти в базі даних і застосувати необхідний експлойт для компрометації робочої станції. З Рисунка А.5 можна побачити, що ENC Workstation — робоча станція Admin, тобто при її компрометації, зловмисник отримає достатні права для доступу до серверу в демілітаризованій зоні.

4. Historian Web Client.compromise, CAS ECN-PN.produceRequest, CAS Server.access, CAS Server.findExploit, CAS Server.deployExploit, Historian Server OS.executeArbitraryCode, Historian Server OS.compromise — використовуючи привілеї скомпрометованого адміністратора, зловмисник встановлює з’єднання з сервером Historian Server OS в демілітаризованій зоні.

5. OS Web Client - PN.compromise, PCS7 PN-PCN.produceRequest, OS Web Server - PCN.access, OS Web Server - PCN.compromise — зловмисник отримує доступ через демілітаризовану зону до серверів мережі керування промисловим процесом.

6. PCS7 PCN Server-Engineer.produceResponse, OS Web Client Engineer.findExploit, OS Web Client Engineer.deployExploit, Engineering Workstation.executeArbitraryCode, Engineering Workstation.compromise — зловмисник інфікує файли проекту PCS7, які інсталюються на інженерну робочу станцію, тим самим компрометуючи її.

7. Siemens PLC Studio.compromise, Siemens PLC Transfer.produceRequest, Siemens PLCStudio Server.access, Siemens PLCStudio Server.findExploit, Siemens PLCStudio Server.deployExploit, S7-400H.executeArbitraryCode, S7-400H.compromise — зловмисник отримує контроль над PLC.

З цих результатів видно, що CySeMoL згоден з тим, що було можливо, що атака сталася, як описано в [19] з огляду на, що мережа виглядала саме так.

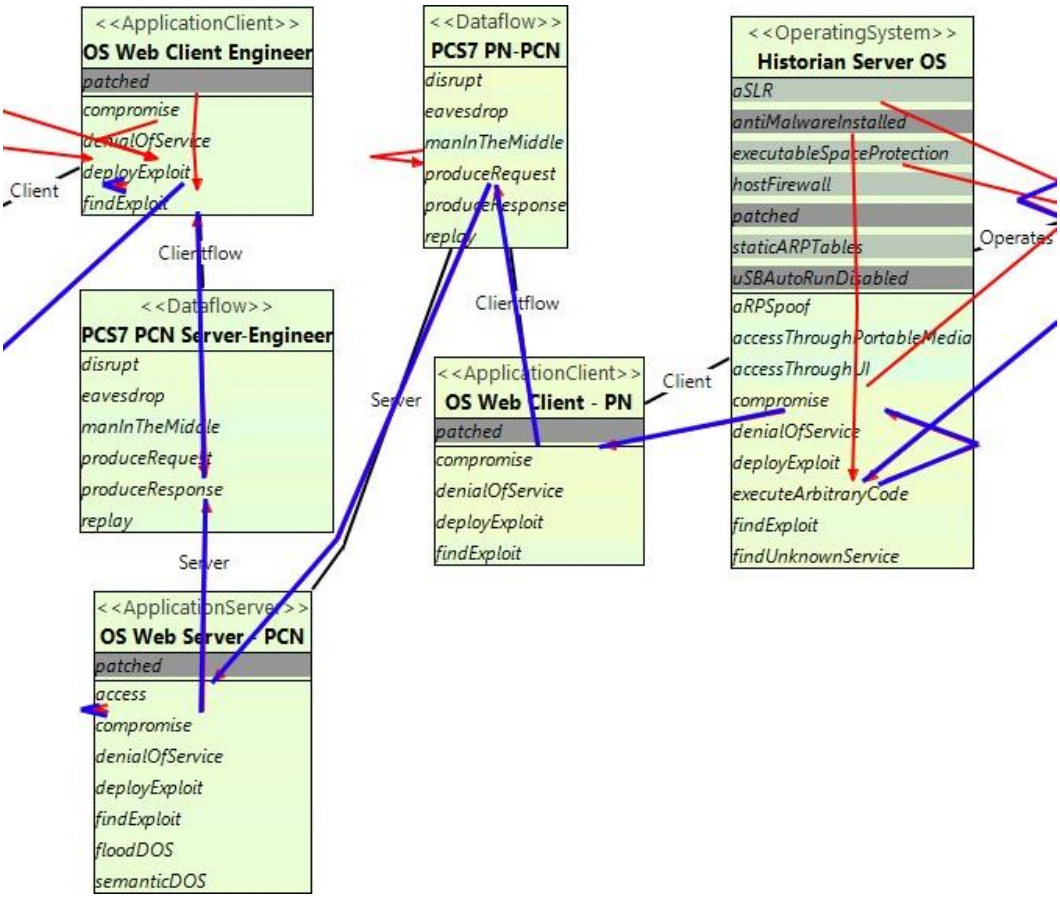

Рисунок 3.8 – Атака на SCADA систему Siemens, обчислена CySeMoL, частина 2

Однак CySeMoL приписує певну позитивну ймовірність кожному з'єднанню в моделі, і тому важко зробити якісні висновки про фактичні ймовірності в цьому випадку. Є два аспекти атаки, які не можуть бути належним чином змодельовані CySeMoL. По-перше, CySeMoL не має поняття привілеїв. У реальній атаці Stuxnet розповсюджувався між хостами в мережі через SMB-ресурси. Само по собі це вимагало тільки прав звичайного користувача, а не кореневого доступу до машини. У моделі CySeMoL, однак, атака моделюється як повна компрометація хоста. По-друге, треба розглядати специфічні для

домену атаки, такі як знищення PLC. CySeMoL передбачає атаки типу злому PLC і не розглядає те, до чого це призведе.

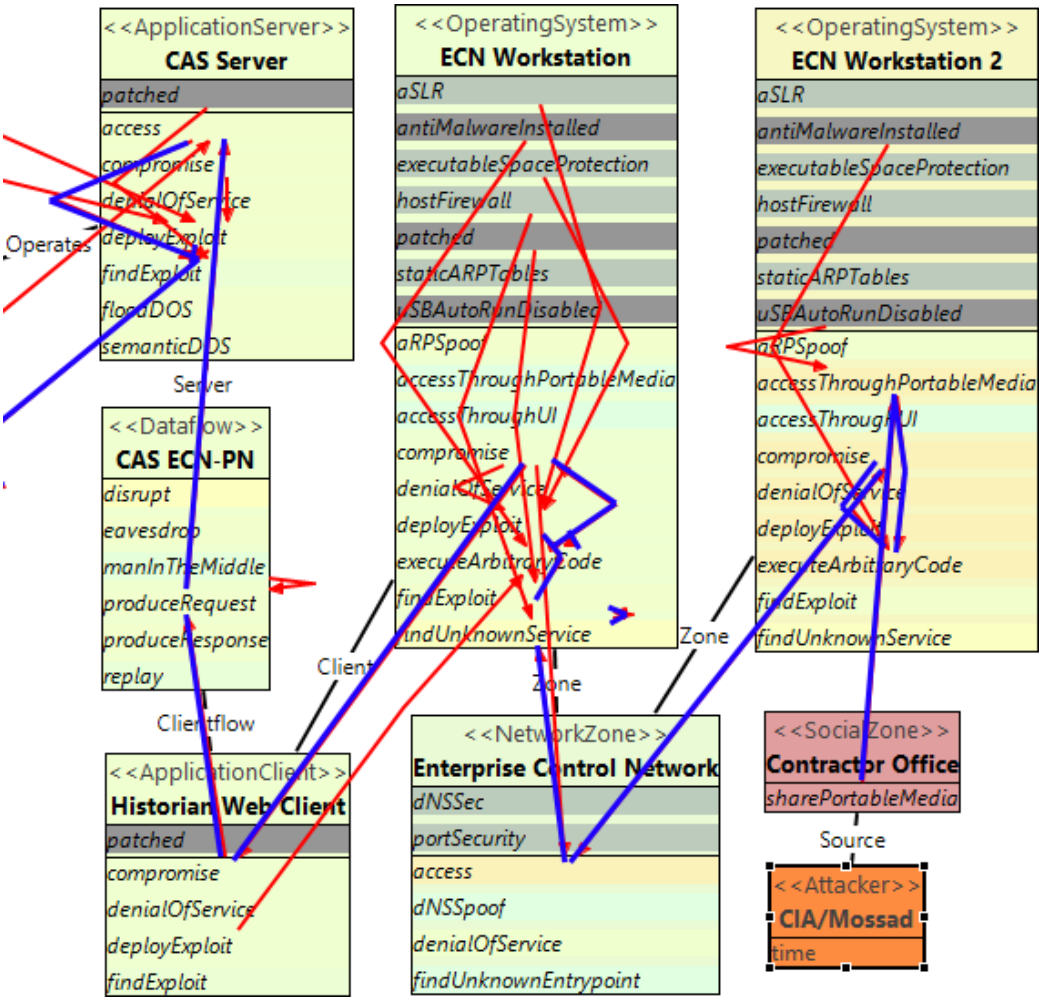

Рисунок 3.9 – Атака Stuxnet, обчислена CySeMoL, частина 3

В даний час один із способів подання рівнів доступу в CySeMoL полягає в тому, щоб мати кілька копій одного і того ж застосунку і підключати до них різні AccessControlPoints. Таким чином, один фізичний застосунок представлено декількома віртуальними, кожен з яких представляє середовище, яке бачить користувач. Приклад цього показаний на Рисунку 3.10. Це можна розширити до рівня операційної системи, створивши дві копії одного і того ж комп'ютера з невеликими змінами в залежності від можливостей користувача. Є, щонайменше, два недоліки у цьому підході. Перш за все, це дублює велику

роботу і робить модель більше. По-друге, він не може належним чином відображати реальні умовні ймовірності між залученими об'єктами. Наприклад, в разі двох призначених для користувача середовищ, CySeMoL розглядатиме це як два окремих комп'ютера, підключених до однієї мережі. Це є проблемою, оскільки ймовірність злому облікового запису адміністратора, якщо ви зламали звичайного користувача, не дорівнює ймовірності злому комп'ютера, якщо ви зламали інший комп'ютер в мережі.

Більш інтуїтивним способом представлення рівнів доступу, без захаращення моделі занадто великою кількістю деталей, може бути введення двох рівнів доступу — звичайних користувачів і адміністраторів, як це зазвичай буває в комп'ютерній системі. Якщо зробити це таким чином, було б достатньо мати лише один екземпляр кожного програмного забезпечення і комп'ютера, але з'єднання між «PasswordAccount» і «AccessControlPoint» можна було б вибрати або «User», або «Admin» замість поточних «Credentials». Це створить невелике додаткове навантаження на моделювання, але потенційно може поліпшити результати. Приклад того, як це може виглядати, показаний на Рисунку 3.11.

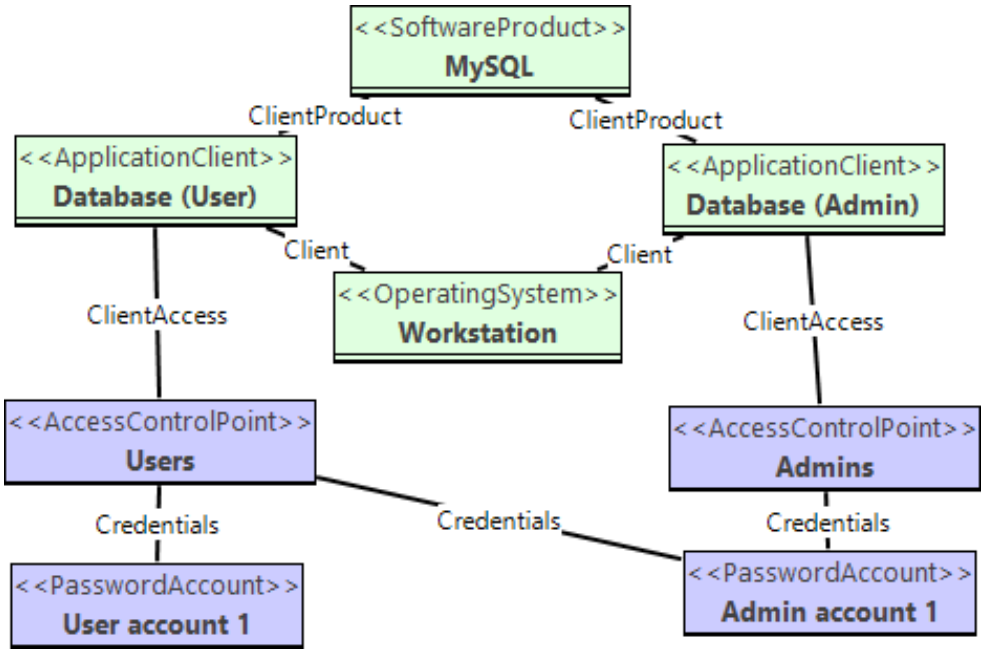

Рисунок 3.10 – Приклад моделювання ACL в CySeMoL

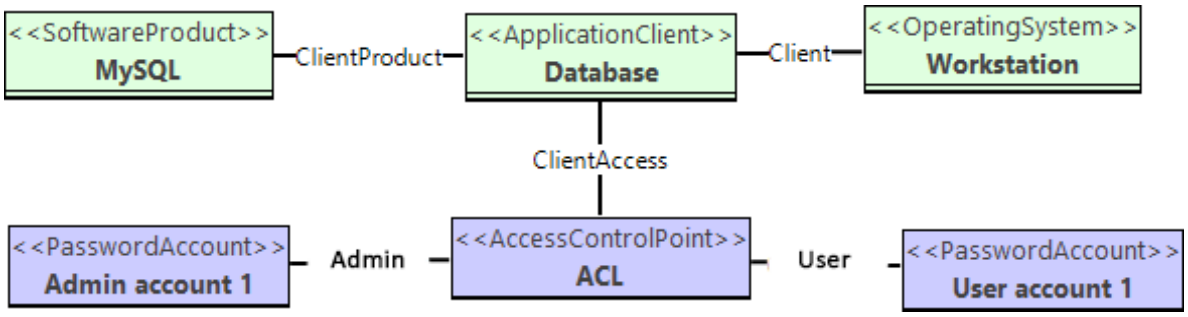

Рисунок 3.11 – Приклад моделювання ACL в CySeMoL

Отже, ґрунтуючись на результатах роботи EAAT разом з CySeMoL, з впевненістю можна стверджувати, що вони обробляють багато аспектів моделювання загроз, попри необхідність удосконалення деяких функцій. Однією із них є досягнення рівноваги в деталях моделі. Надто деталізована модель буде громіздкою і складною для роботи, натомість надто спрощена модель дасть безглузді результати.

Інструменти моделювання загроз, такі, як CySeMoL, можуть виявитися цінним інструментом для системних адміністраторів. Це дослідження показало, що CySeMoL без проблем вдається представити велику частину компонентів систем SCADA, які потенційно впливають на захищеність. Також було обґрунтовано доцільність використання саме CySeMoL, оскільки він не використовує для збору даних мережеві сканери вразливості, які не рекомендовані для систем SCADA.

Висновки до розділу 3

У цьому розділі були показані основи роботи EAAT разом з CySeMoL. Для моделювання обрано іранську SCADA-систему SIMATIC WinCC фірми Siemens, оскільки, через відому атаку Stuxnet, можна отримати дані про реальну структуру системи, які, зазвичай, знаходяться в обмеженому доступі.

Був описаний процес аналізу захищеності та подання результатів. Після моделювання системи, було розглянуто висновки CySeMoL щодо її захищеності. Як і очікувалось, CySeMoL показав хороші результати, передбачивши ймовірність атаки, що за своїм принципом, за деякими даними[19], схожа на атаку Stuxnet.

Отримані результати дозволяють стверджувати, що EAAT разом з CySeMoL є надзвичайно корисним набором інструментів для оцінки системи захисту SCADA-систем.

ВИСНОВКИ

У цій роботі розглянуто структуру, завдання та основні функції систем SCADA, що є частиною АСУ ТП. Детальний аналіз їх архітектури дозволив визначити передумови виникнення загроз для систем та причину їх недостатньої захищеності. Вони полягають у стрімкому збільшенні вимог до функціональності, надійності та вартості, тоді як захищеність таких систем, або не є пріоритетною для власників, або її надзвичайно важко забезпечити через особливості цих систем. Це і зумовлює найбільші вразливості безпеки систем SCADA, що полягають у їх первинному дизайні. Більшість систем, що використовується у даний час, були розроблені двадцять чи то й більше років тому й не захищені, адже тоді не враховувалась поява корпоративних мереж. Оскільки вони не призначалися для роботи в мережі, більшість систем SCADA, що використовуються у критичній інфраструктурі, не захищені належним чином. Також була показана принципіальна відмінність SCADA-системам від інших IT-системам.

Також були проаналізовані різні інструменти для оцінки захищеності систем SCADA. Серед них: CySeMoL, P^{2}AMF, MulVAL, NetSPA та TVA. Ґрунтуючись на дослідженні шведського Королівського технологічного інституту, а також враховуючи недоліки інших інструментів, було обрано найкращий інструмент саме для SCADA-систем — CySeMoL, що реалізований у програмному засобі EAAT. Така комбінація дозволяє без проблем змоделювати систему безпеки SCADA-системи та створити нові окремі компоненти системи, якщо експерту недостатньо стандартних. Результати оцінки захищеності системи подаються у вигляді теплового забарвлення атрибутів, де кожен колір відповідає певному відсотку успішності виконання атаки. Змінюючи атрибути захисту, можна відстежувати їх вплив на загальний стан захищеності системи.

В результаті було створено власну спрощену модель реальної SCADA-системи, на основі системи SIMATIC WinCC компанії Siemens. Результати

оцінки CySeMoL, а саме кроки атаки та їх ймовірність, були проаналізовані. Отримані результати дозволяють стверджувати, що CySeMoL може насправді робити осмислені кількісні висновки щодо захищеності нової системи, яку планують вводити в експлуатацію, або ж окремо оцінювати вплив додаткових компонентів на безпеку SCADA-системи.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Supervisory Control and Data Acquisition (SCADA) System [Електронний ресурс] // Office of the Manager National Communications System. – 2004. – Режим доступу до ресурсу: https://scadahacker.com/library/Documents/ICS_Basics/SCADA%20Basics%20-%20NCS%20TIB%2004-1.pdf.

2. Martti Lehto. Cyber Security: Analytics, Technology and Automation / Martti Lehto, Pekka Neittaanmäki. – Cham, Switzerland: Springer International Publishing AG. – 2015. – 268 с.

3. "Security Through Obscurity" Ain't What They Think It Is [Електронний ресурс] // Jay Beale, Lead Developer, Bastille Linux Project. – 2001. – Режим доступу до ресурсу: https://web.archive.org/web/20070202151534/http:/www.bastille-linux.org/jay/obscurity-revisited.html.

4. D5.4 – CockpitCI System Factory Trials Report [Електронний ресурс] / [R. Pietro, G. Antonio, D. Federico та ін.] // Cybersecurity on SCADA: risk prediction, analysis and reaction tools for Critical Infrastructures. – 2014. – Режим доступу до ресурсу: https://cockpitci.itrust.lu/wp-content/uploads/2015/04/CockpitCI-D5.4-CockpitCI-System-Factory-Trials-Report.pdf.

5. ISA-99.00.01 Security for industrial automation and control systems—part 1: terminology, concepts, and models // American National Standard, Research Triangle Park. – 2007. – 95 с.

6. MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1b3 [Електронний ресурс]. – 2012. – Режим доступу до ресурсу: http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf

7. Sadowsky G. Information Technology Security Handbook / G. Sadowsky, J. Dempsey, A. Greenberg. – Washington: Global Information And Communication Technologies Department, 2003. – 392 с.

8. Guide to Industrial Control Systems (ICS) Security / [K. Stouffer, V. Pillitteri, S. Lightman та ін.]. // NIST Special Publication. – 2015. – №800. – С. 135.

9. Винокурова О. А. БЕЗОПАСНОСТЬ ПРОМЫШЛЕННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ, ВИДЫ УГРОЗ И ОБЩИЕ ПРИНЦИПЫ ЗАЩИТЫ ИНФОРМАЦИИ / О. А. Винокурова, Е. В. Шибарова. // ВЕСТНИК МГУП ИМЕНИ ИВАНА ФЕДОРОВА. – 2016. – С. 4.

10. GLOBAL ICS & IIOT RISK REPORT [Електронний ресурс] // CYBERX. – 2019. – Режим доступу до ресурсу: https://cyberx-labs.com/resources/risk-report-2019/.

11. K. D. Wall. The Kaplan and Garrick Definition of Risk and its Application to Managerial Decision Problems [Електронний ресурс] / K. D. Wall. – 2011. – Режим доступу до ресурсу: https://my.nps.edu/documents/103424423/106950799/DRMI+Working+Paper+2011-3.pdf/bad99104-b54b-4646-9d92-45e16c2f80d8.

12. Jajodia S. Topological analysis of network attack vulnerability / Jajodia S., S. Noel, B. OBerry. – Springer: Springer International Publishing AG, 2005. – 312 с.

13. Runeson, P. Case study research in software engineering. – Hoboken - 2012.

14. A Manual for the Cyber Security Modeling Language [Електронний ресурс] / H.Holm, M. Ekstedt, T. Sommestad, M. Korman. – 2014. – Режим доступу до ресурсу: https://www.kth.se/polopoly_fs/1.588086.1550156425!/cysemol_manual_v2.2_changelog.pdf.

15. CySeMoL: A tool for cyber security analysis of enterprises [Електронний ресурс] / Hannes Holm, Teodor Sommestad. – 2013. – Режим

доступу до ресурсу: http://www.sommestad.com/teodor/Filer/Ekstedt%20et%20al.%20-%202013%20-%20CySeMoL%20A%20tool%20for%20cyber%20security%20analysis%20of%20enterprises.pdf.

16. JOHANSSON D. Empirical test of a tool for cybersecurity vulnerability assessment / DAN JOHANSSON. – 2015. – С. 30–35.

17. Sommestad T. The Cyber Security Modeling Language: A Tool for Assessing the Vulnerability of Enterprise System Architectures [Електронний ресурс] / T. Sommestad, M. Eksted. – 2014.

18. Janulevicius J. Extension of cysemol for cloud computing information security assessment [Електронний ресурс] / J. Janulevicius, N. Goranin. – 2016. – Режим доступу до ресурсу: https://www.researchgate.net/publication/305465666_Extension_of_cysemol_for_cloud_computing_information_security_assessment.

19. Byres E. How Stuxnet Spreads – A Study of Infection Paths in Best Practice Systems [Електронний ресурс] / E. Byres, A. Ginter, J. Langill. – 2011. – Режим доступу до ресурсу: http://www.barr-thorp.com/wp-content/uploads/2011/04/how-stuxnet-spreads.pdf.

ДОДАТОК А

Модель системи Stuxnet в CySeMoL

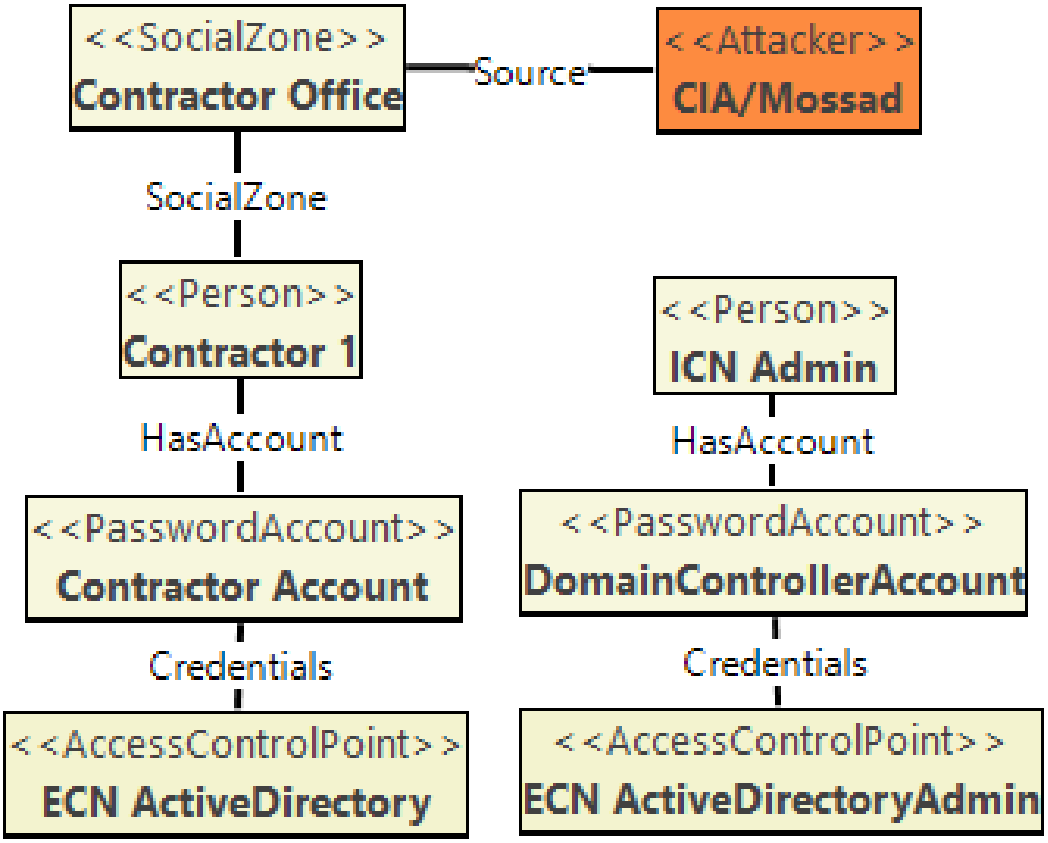

Рисунок А.1 – Працівники та облікові записи в моделі

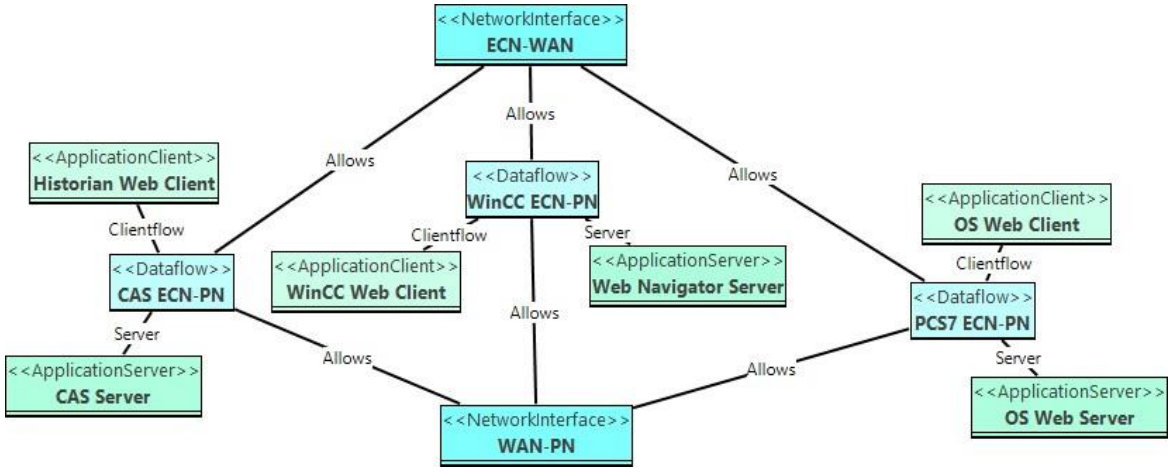

Рисунок А.2 – Потік даних в моделі, частина 1

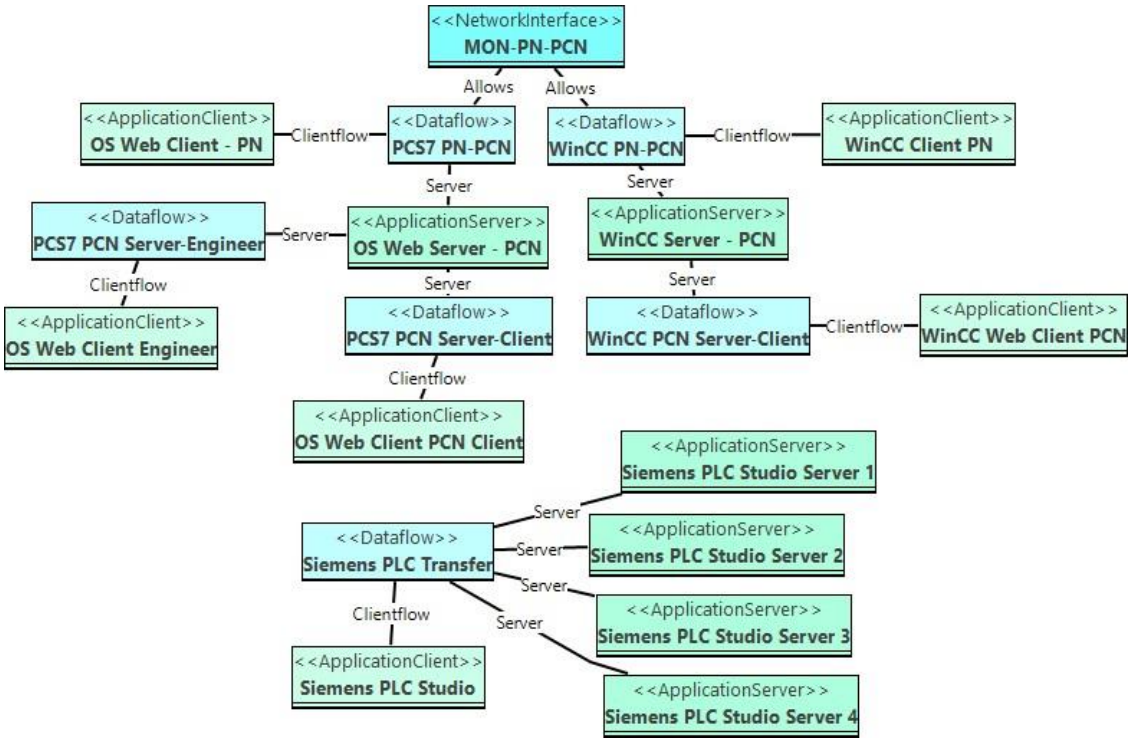

Рисунок А.3 – Потік даних в моделі, частина 2

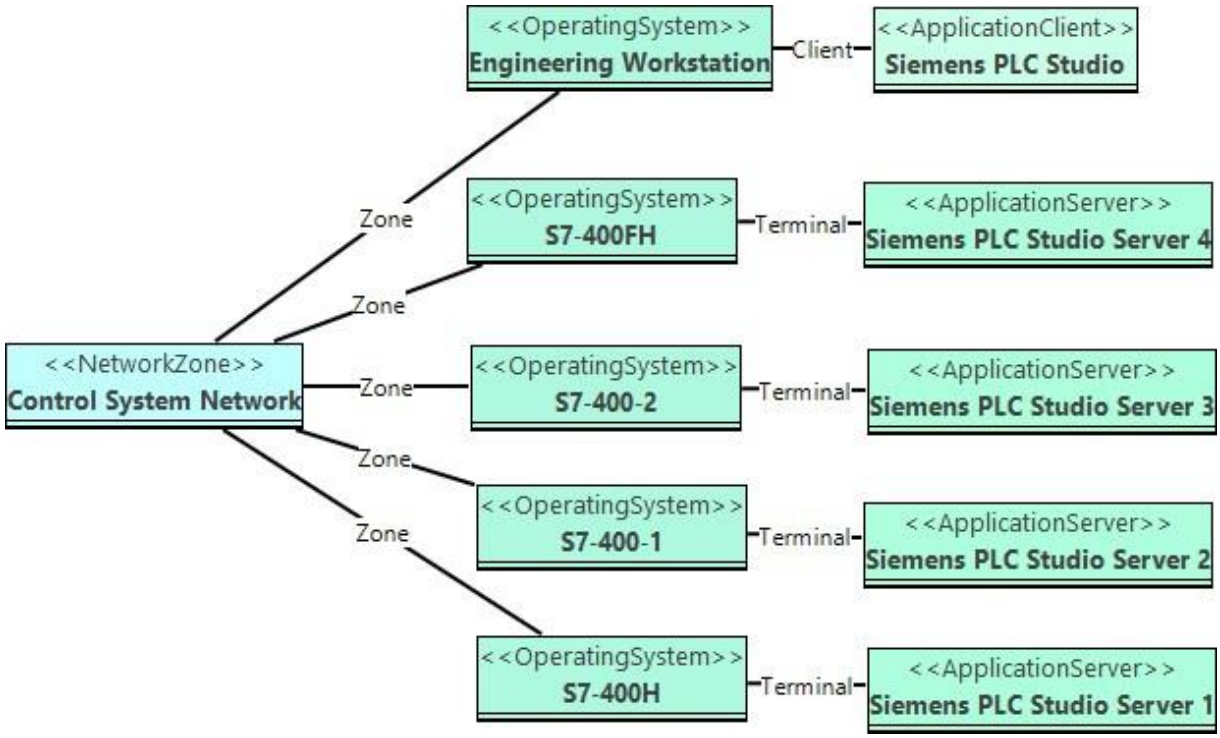

Рисунок А.4 – Мережа систем управління (Control Systems Network) в моделі

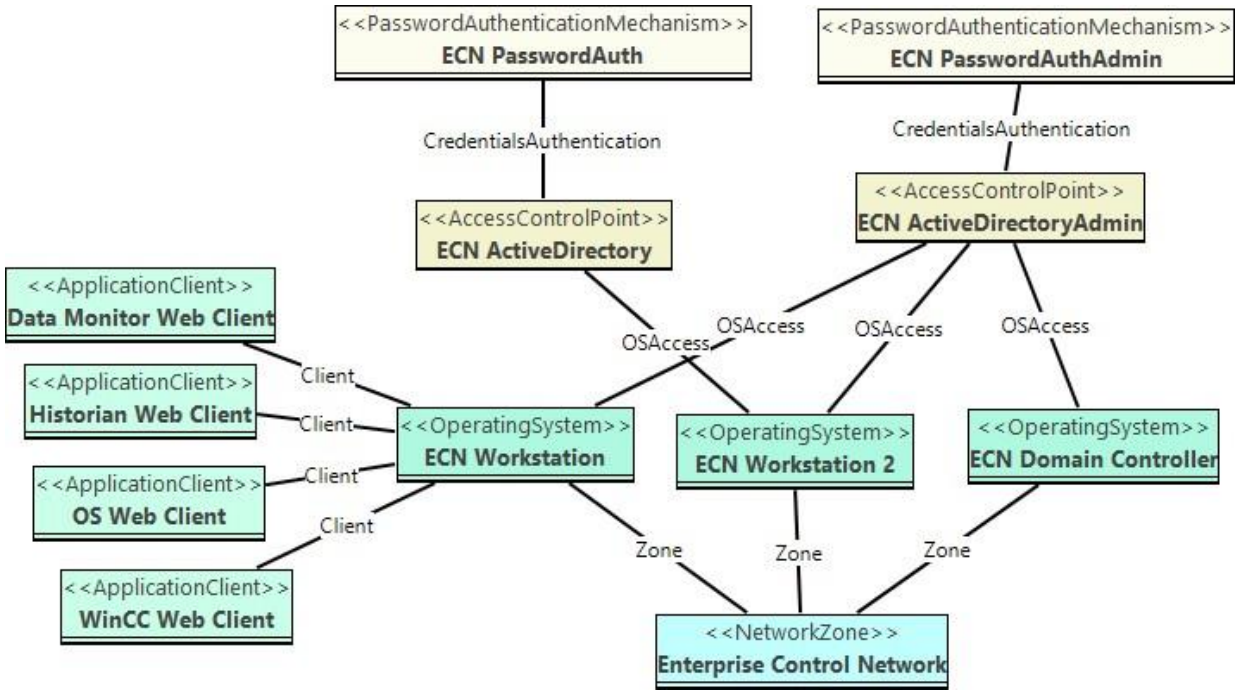

Рисунок А.5 – Мережа керування підприємством (Enterprise Control Network)

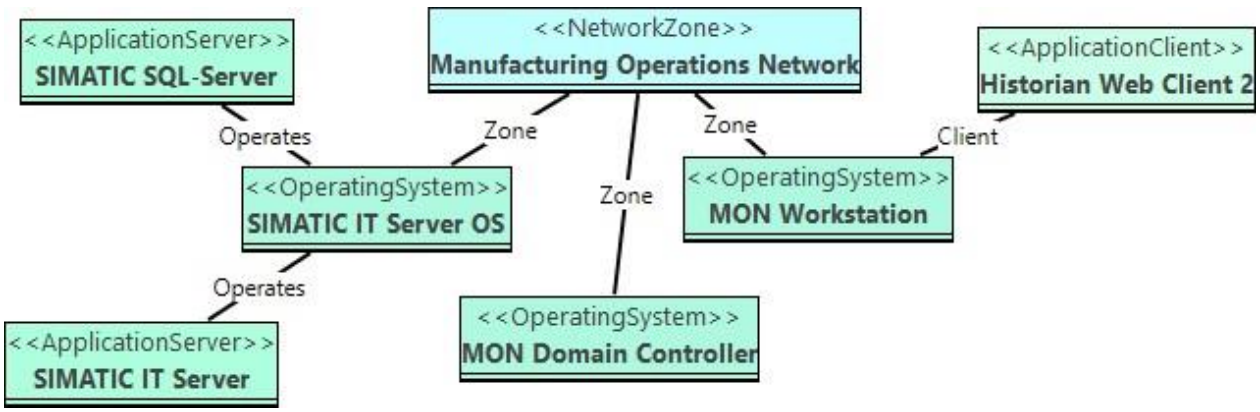

Рисунок А.6 – Мережа виробничих операцій (Manufacturing Operations Network)

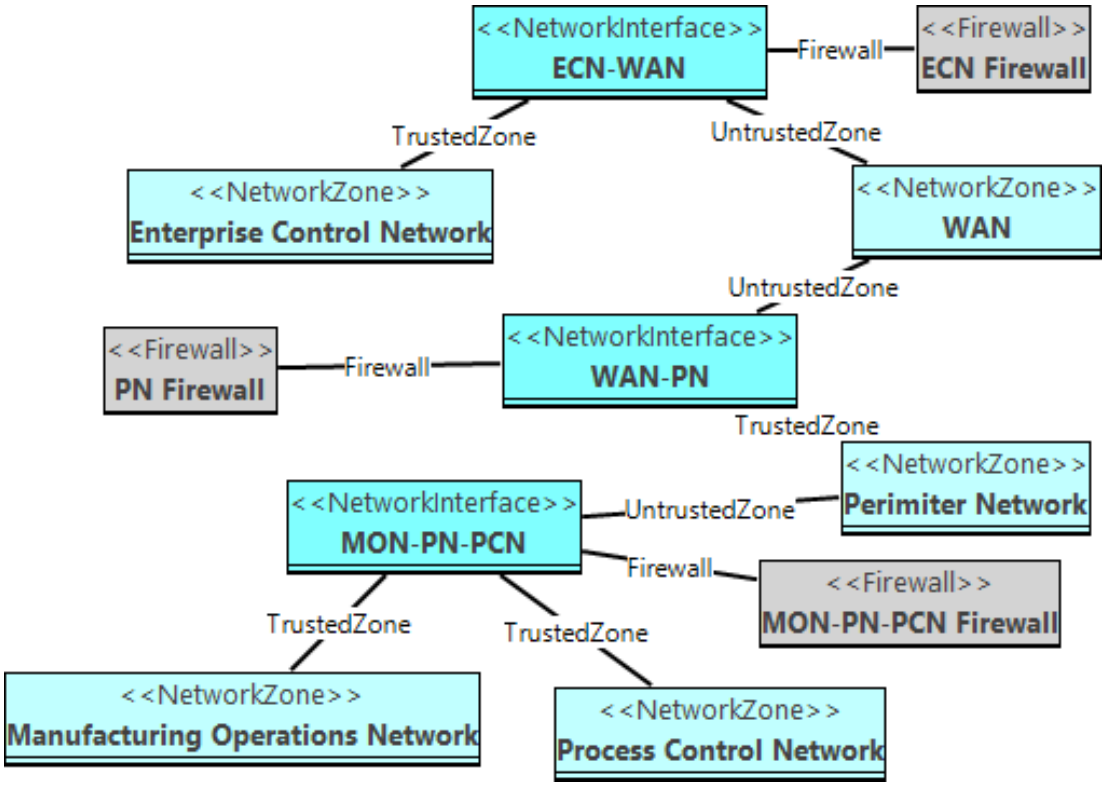

Рисунок А.7 – Топологія мережі моделі Stuxnet

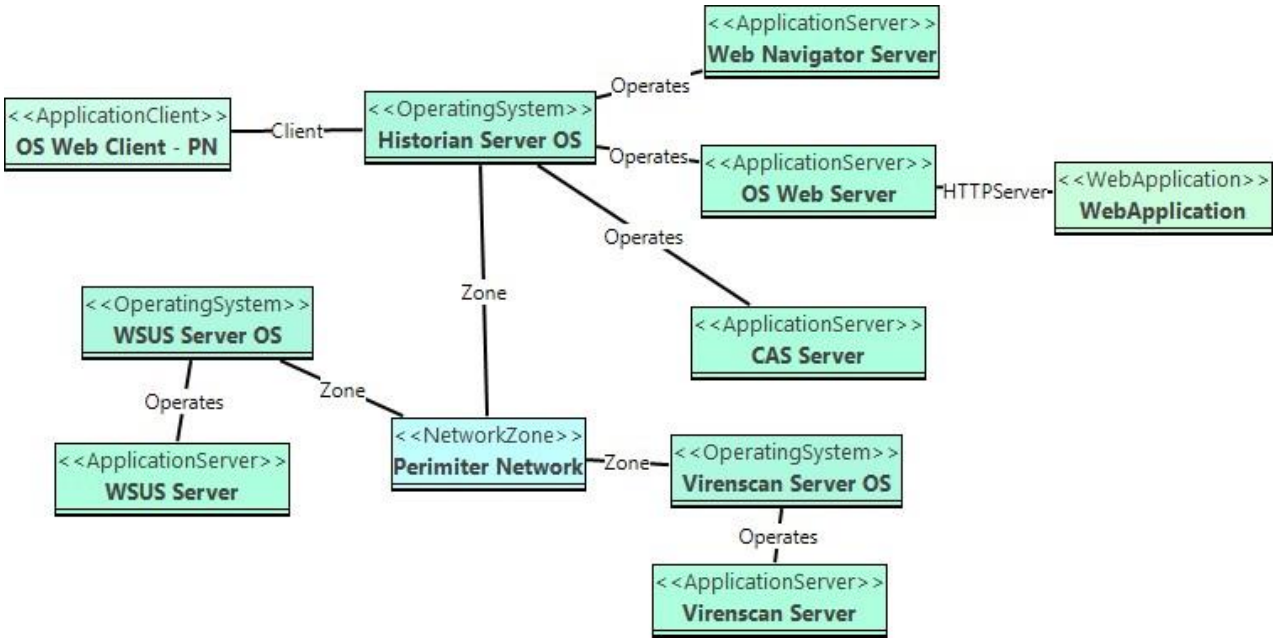

Рисунок А.8 – Демілітраизована зона (Perimeter Network)

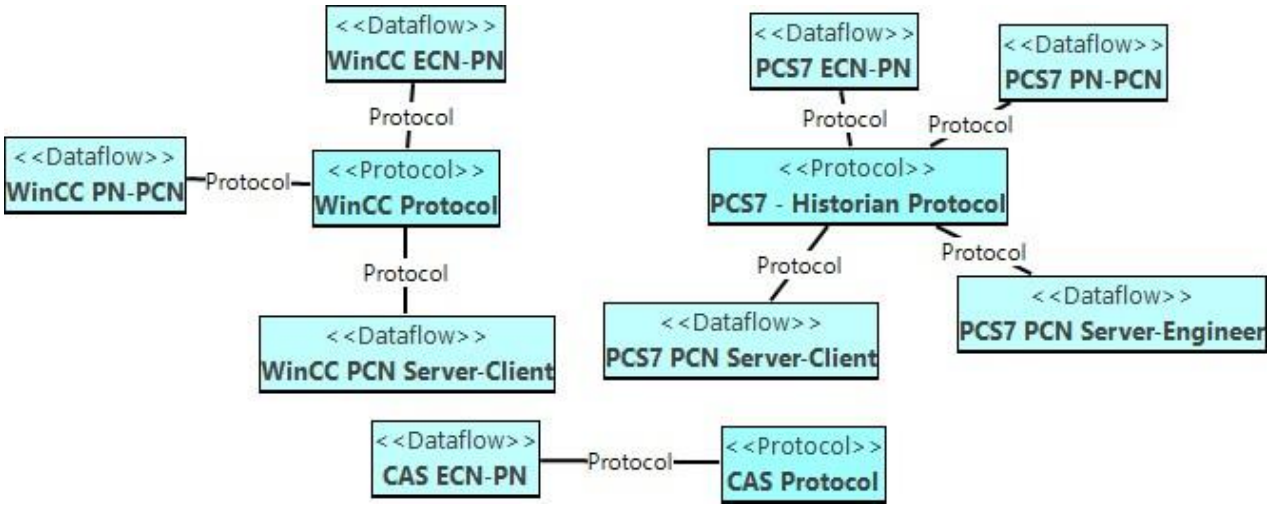

Рисунок А.9 – Протоколи, що використовуються в моделі

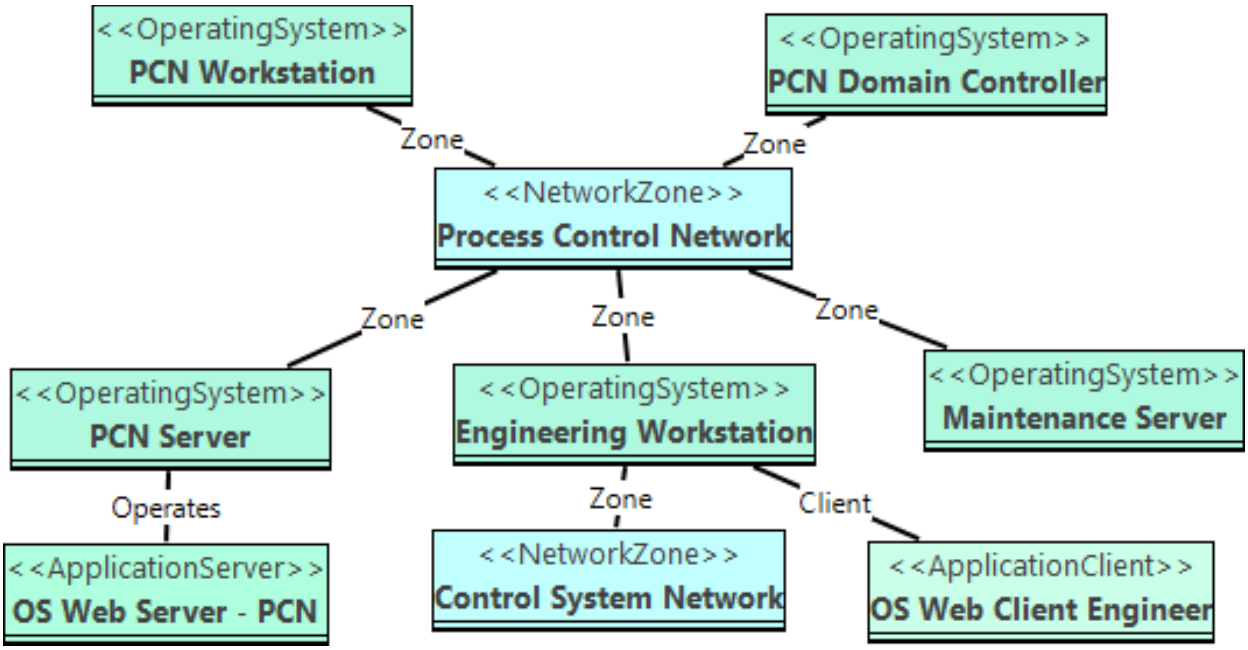

Рисунок А.10 – Мережа керування процесом (Process Control Network)

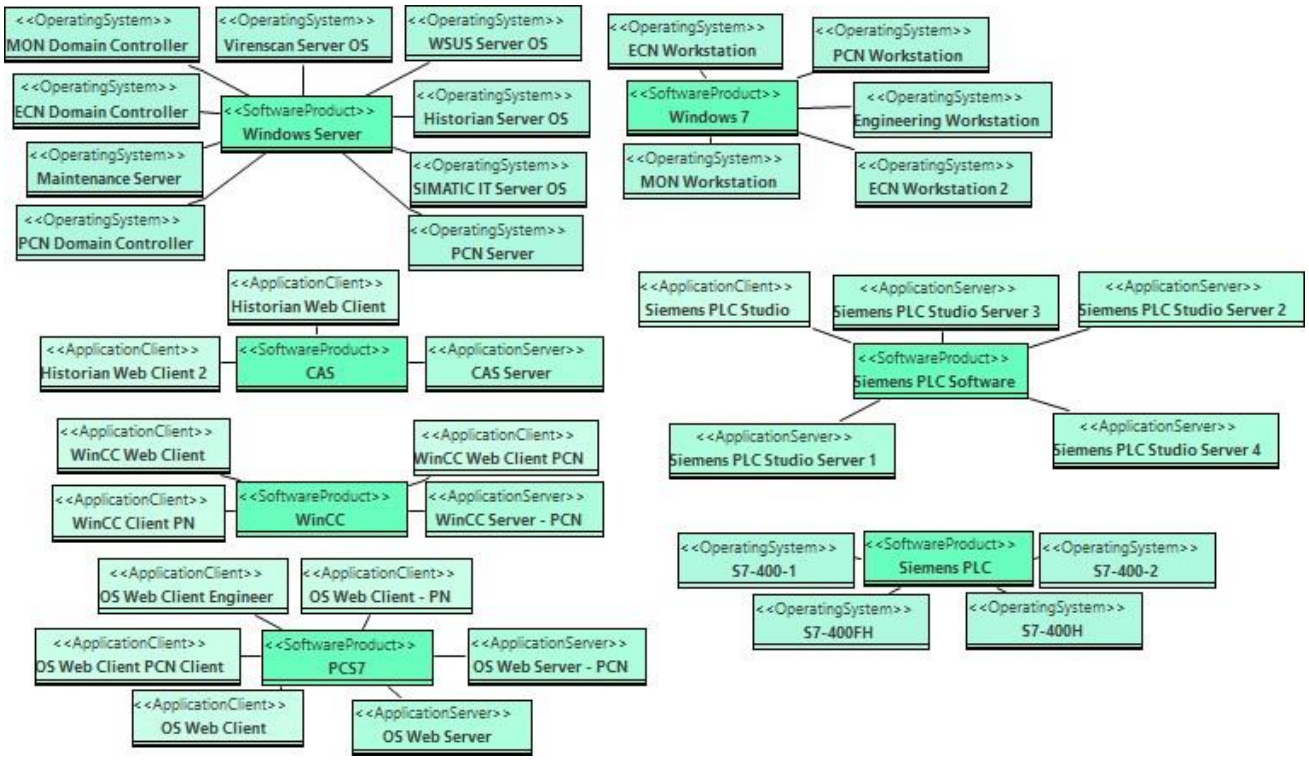

Рисунок А.11 – Програмне забезпечення, що застосовується в моделі