ПОЛТАВСЬКИЙ ДЕРЖАВНИЙ АГРАРНИЙ УНІВЕРСИТЕТ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ, ПРАВА ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ

Пояснювальна записка

до кваліфікаційної роботи на здобуття ступеня вищої освіти магістр

на тему: **«Модель безпеки системи розумного будинку на основі параметрів контролю доступу»**

Виконав: здобувач вищої освіти
за освітньо-професійною програмою
Інформаційні управляючі системи та технології
спеціальності 126 Інформаційні системи та технології
ступеня вищої освіти магістр
групи 126ІСТмд_21
Грицай І. Ф.
Керівник: Калініченко А.В.
Рецензент: Біловод О.І.

Полтава – 2023 року

ВСТУП

Актуальність теми. Розвиток Інтернету призвів до поширення його супутніх технологій, серед яких важливе місце займає Інтернет речей (IoT) [1]. Термін «Інтернет речей» недавно став широко відомим. Хоча вперше він з'явився в 1999 році під час розробки стандарту для ідентифікації об'єктів за допомогою RFID-міток у логістиці. В сучасний час поняття Інтернету речей використовується для опису сценаріїв, де мережеве з'єднання та обчислювальні можливості розповсюджуються на різні об'єкти, пристрої, датчики та повсякденні речі [2].

Розумний будинок»– це комплекс високотехнологічних пристроїв, спрямованих на забезпечення комфорту та вирішення різноманітних завдань. Сучасні розумні будинки керуються за допомогою сенсорних панелей або через Інтернет зі смартфона або планшету. Спеціальне програмне забезпечення дозволяє регулювати клімат, освітлення, систему водопостачання, а також віддалено спостерігати за подіями в будинку і навіть розважатися без особливих зусиль.

Проблеми, пов'язані з побудовою серверних систем для розумних будівель на базі Інтернету речей, можна розділити на загальні, які характерні для багатьох систем обробки великих обсягів даних, та специфічні, що виникають саме в цій області [3]. До загальних задач відноситься створення та використання ефективних, стійких до відмов, масштабованих та розподілених систем обробки даних. Специфічними для Інтернету речей є проблеми забезпечення надійного управління та моніторингу пристроїв.

Сприяння розвитку Інтернету речей здійснюється завдяки комерційному інтересу та початку роботи над референтними архітектурами для конкретних галузей [4]. Останні десятиліття активно досліджувались питання розвитку методів та моделей оцінювання якості обслуговування, надійності та готовності розумних будинків та систем IoT відомими вченими, такими як В. Харченко, А. Горбенко, К. Тріведі, А. Боярчук. їх дослідження включають аналіз аналітичних моделей хмарних IoT сервісів з використанням спеціалізованих засобів моделювання. Проте питання побудови моделей безпеки системи розумного будинку на основі

параметрів контролю доступу залишається актуальним.

Зв'язок роботи з науковими програмами, темами. Робота відповідає дослідженням в межах науково-дослідної роботи «Розвиток підприємництва: управлінські, економічні, інноваційна та правові аспекти» відповідно до договору №9 від 15.05.2023 р. між ТОВ «ПАФ Гарант» та Полтавським державним аграрним університетом (розділ «Обґрунтування показників оцінювання гарантоздатності розподілених інформаційних систем»).

Метою кваліфікаційної роботи є забезпечення безпеки системи розумного будинку на основі параметрів контролю доступу.

Завданнями кваліфікаційної роботи є:

– аналіз вимог до безпеки систем розумного будинку;

– розробка контекстної моделі контролю доступу для забезпечення безпеки систем розумного будинку;

– моделювання архітектури розумного будинку та його кібербезпеки.

Об'єктом дослідження є процеси аналізу та моделювання кібербезпеки систем розумного будинку.

Предметом дослідження є оцінка кіберзагроз компонентів розумного будинку та побудова моделі дерева атак для розрахунку ймовірності відмови системи при загрозах підсистемі доступу.

Методи дослідження – проведені в роботі дослідження базуються на методах теорії ймовірності, системного і марковського аналізу, систем масового обслуговування, які використовувалися при розробці моделі дерева атак та оцінки кібербезпеки системи.

Інформаційна база кваліфікаційної роботи складається з наукових статей, міжнародних аналітичних видань і звітів, матеріалів наукових конференцій інтернет-ресурсів, що місять інформацію про архітектуру систем розумних будинків, а також даних, отриманих від провідних IT-компаній у сфері Smart House.

Елементи наукової новизни полягають у розроблені та досліджені аналітичної моделі кібербезпеки системи розумного будинку у вигляді дерева атак. Проведено аналіз кіберзагроз і вразливостей системи.

Практична значущість роботи полягає в можливості повторного застосування та модифікації розробленого програмного коду моделі для оцінювання показників готовності і кібербезпеки розумних будівель. Отримані результати можуть бути корисними для ІТ фахівців при моделюванні спеціалізованих інформаційних управляючих систем.

Апробація результатів дослідження відбувалася шляхом оприлюднення доповідей на наукових конференціях, семінарах.

Публікації. За результатами проведеного дослідження опубліковано тези: «Імітаційна модель для оцінювання безпеки системи з віртуалізованою інфраструктурою з врахуванням атак на компоненти», Матеріали XII Міжнар. наук. конференції «Інформаційні технології в енергетиці та агропромисловому комплексі», м. Львів, 04-06 жовтня 2023 р.

Структура та обсяг кваліфікаційної роботи логічно пов'язані з задачами досліджень. Робота містить перелік умовних позначень, вступ, три розділи основної частини, висновки, список використаних джерел, додатки. Загальний обсяг текстової частини дипломної роботи складає 63 сторінки формату А4. Вона містить 21 рисунок і 6 таблиць. В роботі використано 44 науково-технічних джерела.

РОЗДІЛ 1
АНАЛІЗ ВИМОГ ДО БЕЗПЕКИ СИСТЕМ РОЗУМНОГО БУДИНКУ

1.1 Аналіз концепції «розумних» середовища та будинку

Розумне середовище – це об'єднання пристроїв, які спільно використовують свої ресурси [2]. Це може викликати конфлікти між різними пристроями та учасниками, що мають різні цілі та розуміння поточної ситуації, але працюють у спільному інформаційному просторі. Для забезпечення безпеки даних необхідний механізм динамічного контролю доступу до загальних ресурсів. Це призводить до потреби моделі контролю доступу, яка враховує інформацію про стан середовища у формі контексту виконання операцій.

Основна мета «розумного» середовища полягає в зборі та аналізі інформації про навколишнє середовище для створення нових можливостей взаємодії користувачів з оточенням [3]. Також можуть використовуватися механізми адаптації до потреб користувачів. Властивості «розумного» середовища включають:

– пряму взаємодію між пристроями;

– віддалене керування пристроями;

– складний функціонал пристроїв;

– «інтелектуальність» пристроїв;

– різноманітність стандартів мережевої взаємодії.

Автоматизовані системи управління будинками, відомі також як технологія розумного будинку, широко застосовуються для вирішення різних завдань, пов'язаних з експлуатацією будівель. Технологія розумного будинку використовує сучасні системи автоматизації та різноманітні периферійні пристрої з метою забезпечення безпеки, економії ресурсів та поліпшення умов проживання загалом. Однією з важливих особливостей є активна взаємодія різних автоматизованих підсистем для отримання нових можливостей у розпізнаванні та реагуванні на різні ситуації. З фізичної точки зору такі системи представляють собою розвиток

автоматизованих систем управління, що, в свою чергу, є адаптацією автоматизованих систем управління для будівель та споруд.

Розумний будинок – це автоматизована система, яка керує інженерними системами будинку, використовуючи багато різних датчиків. Ця система сприяє безпеці та комфорту людей, які проживають у будинку.

На сьогоднішній день у системі управління електроенергією використовуються різні елементи – реле для домашніх електромереж, датчики температури, руху та звуку. Вони пов'язані з внутрішньою електричною системою будинку або квартири та автоматизують увімкнення/вимкнення світла, опалення, керують роботою певних побутових приладів з урахуванням часових режимів [4].

Одна з переваг розумного будинку полягає в тому, що за допомогою неї можна керувати всіма пристроями, які підключені до мережі, за допомогою одного засобу – мобільного (планшету, смартфону, пульта) та/або стаціонарного (комп'ютера/ноутбука, графічного інтерфейсу системи). Усі датчики і реле також входять до системи та автоматично керують електропостачанням [5]. Крім елементарних побутових приладів, розумний будинок може керувати системами зв'язку, протипожежної сигналізації, охоронної системи, телефонних ліній та багатьох інших, включаючи інженерні комунікації [6].

1.2 Функціональні елементи розумного будинку

Система розумного будинку складається з модулів, кожен з яких відповідає за свою функцію. Компоненти розумного будинку можуть поступово доповнюватися новими модулями для розширення функціоналу системи. Це дозволяє поетапно розвивати систему розумного будинку як цілісну систему. Приклад такої системи наведено на рис. 1.1. У складі системи розумного будинку містяться різноманітні пристрої, зокрема [7]:

– контролер розумного будинку (головний та дискретні модулі вводу-виводу);

- модулі розширення та зв'язку (комутатори, роутери, модулі GPS/GPRS);
- елементи комутації електричних колів (реле, димери, блоки живлення);
- вимірювальні прилади, датчики та сенсори (для руху, температури, освітлення та інші);
- пристрої управління системою (пульти, сенсорні панелі, планшети);
- виконавчі механізми (клапани для води, вентиляції, газу, ролети тощо).

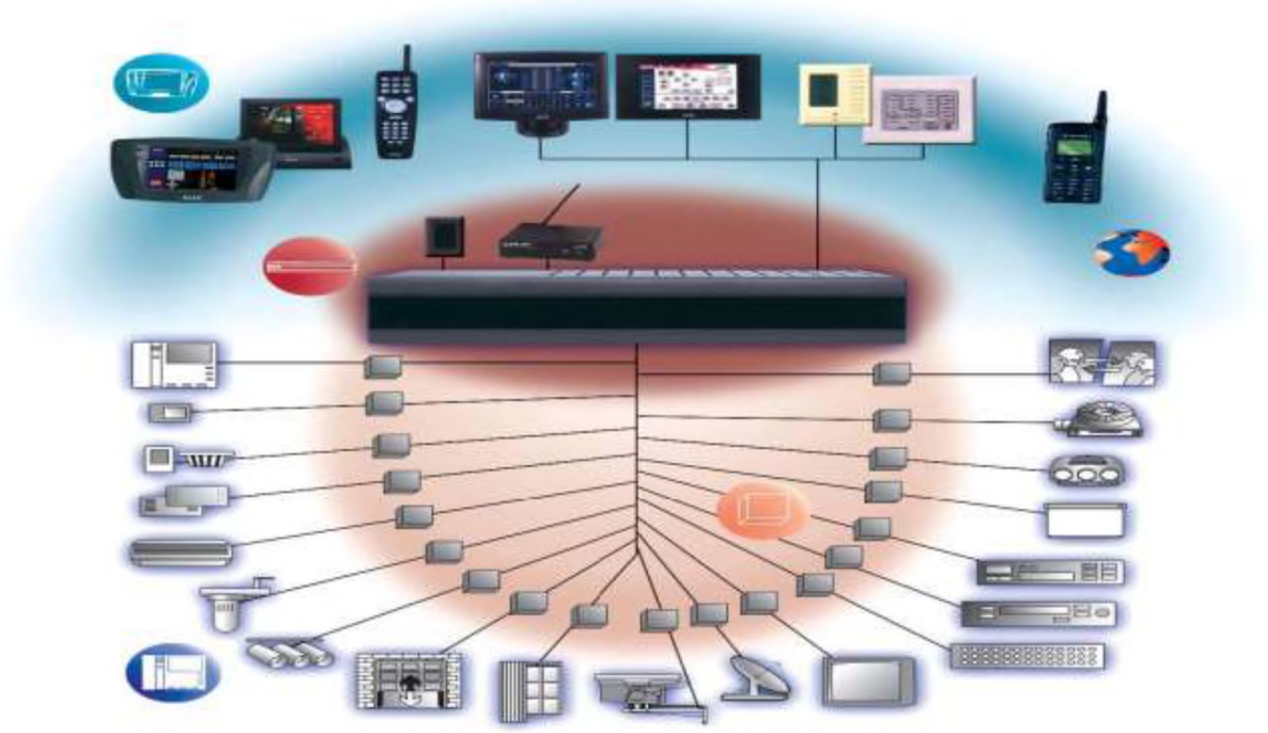

Рисунок 1.1 – Приклад системи розумного будинку

Розглянемо систему розумного будинку, що працює за децентралізованим принципом на основі популярного протоколу European Installation Bus (EIB) та технології KNX (відомої як KNX/EIB). Ця система розумного будинку працює за принципом розподіленого інтелекту, де в кожному виконавчому механізмі вбудований програмований мікрочіп, що керує його функціями та обмінюється даними з іншими пристроями системи [8]. Компоненти децентралізованої системи розумного будинку можна упорядкувати за такими класами:

1. Сенсори (датчики) – «очі» та «вуха» системи, які реєструють зміни контрольованих параметрів, такі як рух, час, освітленість, температура і інші.

Сенсори передають сигнали про зміну значень параметрів через інсталяційну шину.

2. Виконавчі механізми – елементи, які за сигналами від сенсорів чи контролерів здійснюють відповідні дії відповідно до програмованих сценаріїв. Наприклад, димери, що регулюють яскравість лампи за сигналом від датчика освітленості, або реле для відкриття / закриття жалюзі.

3. Контролери – програмні модулі, які приймають сигнали від сенсорів, порівнюють їх та обробляють, а відповідно до заданого сценарію, видають команди виконавчим механізмам.

4. Системні пристрої – компоненти, які забезпечують електроживлення системи та об'єднують сенсори, контролери та виконавчі механізми в єдину систему розумного будинку. Сюди входять блоки живлення, з'єднувальні шини, різноманітні інтерфейси, телефонні та Інтернет-шлюзи тощо.

Загалом, децентралізована система розумного будинку на базі KNX/EIB відкриває можливості для ефективного управління житловим простором, забезпечуючи автоматизацію та зручність у керуванні різними аспектами повсякденного життя в будинку. Використання цієї технології створює сприятливі умови для оптимізації споживання ресурсів та підвищення загального комфорту користувачів.

1.3 Аналіз протоколів для систем розумного будинку

У сфері автоматизації керування будинками застосовуються різноманітні протоколи, проте наразі не існує єдиних стандартів для організації взаємодії пристроїв, які утворюють систему розумного будинку [9, 10]. Застосування локальних мереж для цієї мети вважається не дуже перспективним через їхню складність. Технології, що використовуються в системах розумного будинку, повинні відповідати певним критеріям:

– ефективне використання енергії;

– надійність та безпека передачі даних;

– доступна вартість;

– простота фізичного встановлення.

У багатьох сценаріях використання системи високі швидкості передачі не є обов'язковими. Порівнюючи провідні та бездротові мережі, ключовим аспектом є простота установки мережевих пристроїв. Існують ефективні технології провідних мереж, які відомі як Power Line Communication (PLC) [11]. Вони базуються на тому, що приміщення, де використовується мережа, зазвичай мають електричне освітлення. Таким чином, можливо використовувати провідні та бездротові мережі, застосовуючи технології, такі як X10, INSTEON, HomePlug, Lonworks для зв'язку через електричні мережі, та Bluetooth, Z-Wave та ZigBee для бездротового зв'язку.

ZigBee – це стандарт бездротового зв'язку для систем автоматизації, що включає специфікації мережевих протоколів на верхньому рівні, такому як рівень додатків та мережевий рівень [12, 13]. Нижчі рівні, такі як рівень управління доступом та фізичний рівень, визначаються стандартом IEEE 802.15.4. Рівень програм визначає характеристики пристрою ZigBee, рівень підтримки програм та інтерфейс розробки програм.

Інтерфейс розробки додатків містить опис, що визначає типи даних, дескриптори служб та формати пакетів, що дозволяє швидко створювати прості профілі на основі атрибутів. Об'єкти програм – це програмні модулі, які управляють пристроями ZigBee у кінцевих точках [14].

Підрівень підтримки програм відповідає за передачу даних програмам та профілям пристрою ZigBee та керує їх підключенням у мережі та зберігає відомості про них.

Мережевий рівень відповідає за керування мережевими адресами та маршрутизацію. До його функцій входить запуск мережі, присвоєння адрес, додавання та видалення пристроїв, маршрутизація повідомлень та застосування політики безпеки.

Фізичні рівні визначаються стандартом IEEE 802.15.4 [15]. Рівень керування доступом відповідає за надійний зв'язок пристрою з безпосередніми сусідами та

управління колізіями. Фізичний рівень надає інтерфейс для фізичного передавання даних та складається з двох рівнів у різних частотних діапазонах. Постачальник послуг безпеки забезпечує механізми безпеки для мережевого рівня та рівня додатків, зокрема, шифрування даних. Мережі ZigBee можуть працювати у режимах, що не вимагають шифрування. Основний рівень безпеки не гарантує захисту ключів мережі. Існує механізм лічильників, які постійно зростають, призначений для уникнення атак повторного використання. Але його реалізація може призвести до проблем у функціонуванні мережі, вимагаючи ручного скидання лічильників. Без активації цієї опції, атака повтору стає досить очевидною [16]. Атаки повторного використання та отримання ключа із перехопленого трафіку можуть бути виконані у фреймворку KillerBee, спрямованому на аналіз мереж ZigBee.

Z-Wave – це запатентований набір протоколів, що розробляється та підтримується Z-Wave Alliance [17]. Зараз це один із найбільш перспективних протоколів для використання у сфері створення розумного будинку. Протокол Z-Wave має кілька рівнів, що відповідають моделі OSI [18]:

1. Фізичний рівень.

2. Канальний рівень. Тут контролюється цілісність та адресація пристроїв у прямій видимості. Підтримується багатоадресна та широкомовна розсилка.

3. Мережевий рівень. Специфікації протоколу визначають алгоритм маршрутизації для передачі даних між пристроями, що перебувають поза прямим зв'язком. Всі активні вузли мережі беруть участь у пересиланні пакетів. Шлях пакета визначається перед його відправленням.

4. Транспортний рівень. Тут підтверджується доставка та відправка пакетів в разі втрати. Кожен вузол, який приймає участь у передачі, підтверджує факт отримання повідомлення.

5. Сеансовий рівень. Використовується лише при увімкненому шифруванні для встановлення сеансового ключа.

6. Прикладний рівень. Специфікація Z-Wave визначає алгоритм обробки команд на прикладному рівні. Цей рівень описаний набором класів команд, деякі з яких мають кілька варіантів інтерпретації залежно від класу пристрою.

Ключові елементи автоматизованих систем, такі як замки, зараз використовують шифрування AES-128. Але це шифрування є доповненням до стандарту, і старі пристрої не підтримують його. Крім того, згідно з [19], деякі пристрої мають вразливості в реалізації протоколу обміну ключами, що дає можливість отримати доступ до них за допомогою стандартного ключа – 00000000000000000000000000000000h.

Існують різні рішення для спрощення управління системою розумного будинку на базі протоколу Z-Wave. Одним із таких варіантів, який затверджено Z-Wave Alliance, є Z-Way. Основний інструмент для взаємодії з системою – це вебінтерфейс і відповідне йому API. Однак, вони не забезпечують механізмів аутентифікації та шифрування даних. Таким чином, зловмисник може виконувати будь-які дії з системою розумного будинку після порушення безпеки локальної мережі.

1.4 Аналіз загроз для кібербезпеки системи розумного будинку

Для систем розумного будинку існує ряд потенційних загроз безпеці, які співпадають із загрозами, що властиві багатьом комп'ютерним мережам [20]. У дослідженні [21] розглядаються різноманітні види атак, які спричинили вразливості цих систем. Загрози безпеці для систем розумного будинку можуть включати атаки на проникнення через недостатньо захищені точки доступу до мережі. Підключення таких систем до Інтернету може експонувати їх до ризику використання вразливостей за допомогою шкідливих програм. Це може призвести до несанкціонованого доступу до пристроїв у системі та зловживання їхніми функціями.

Деякі атаки можуть бути спрямовані на перехоплення комунікаційних потоків між пристроями розумного будинку, використовуючи цю інформацію для здійснення несанкціонованого контролю чи стеження за діяльністю мешканців. Крім того, використання слабкого шифрування або відсутність механізмів захисту

можуть створювати ризик для конфіденційності особистих даних, що зберігаються у системі розумного будинку. Дослідження [22, 23] також вказує на можливість атак, спрямованих на самі апаратні засоби системи, використовуючи їх вразливості для незаконного доступу, що може поставити під загрозу не лише дані, а й фізичну безпеку приміщення. Табл. 1.1 містить опис вразливостей систем розумного будинку та можливих наслідків атак на них.

Таблиця 1.1 – Аналіз вразливостей системи розумного будинку

№	Тип атаки	Потенційні загрози	Можливі наслідки
1	Атаки на центральний вузол	Мережа без захисту периметру, підключена до Інтернету	Вихід з ладу системи, витік інформації
2	Вплив вірусів та троянів	Вразливість при підключенні до Інтернету	Виходження з ладу програм та обладнання, порушення конфіденційності
3	Перехоплення інформації	Порушення безпеки радіо або провідних каналів	Потенційне зловживання керуванням системою
4	Несанкціонований доступ адміністратора	Неадекватна аутентифікація	Порушення конфіденційності, цілісності та доступності інформації
5	Неавторизований доступ	Недостатня система автентифікації	Порушення конфіденційності, цілісності та доступності інформації
6	Помилки користувача	Невідповідність захисту	Порушення конфіденційності, можливі системні збої
7	Поломка обладнання	Низька надійність обладнання	Порушення конфіденційності
8	Помилки ПЗ	Нелегальне ПЗ, недостатнє тестування	Порушення конфіденційності, цілісності та доступності інформації

Вивчення цих атак вказує на можливі ризики, які можна зменшити, застосовуючи модель контролю доступу, орієнтовану на контекст.

1.5 Контролер системи розумного будинку

Основний модуль у системі розумного будинку – контролер, який централізує всі пристрої для передачі команд виконавчим механізмам. Ці механізми, у свою чергу, реагують на ці команди, регулюючи роботу різних

пристроїв, наприклад, рух жалюзі. Це головний елемент, який забезпечує керування апаратним обладнанням для розумного будинку [25].

Система розумного будинку відмінно адаптується до різних пристроїв за допомогою спеціальних модулів, адаптерів і перетворювачів інтерфейсу. У комплексі можуть бути включені таймери, датчики, регулятори та вимірювальні пристрої.

Контролер для розумного будинку здатен управляти всією інфраструктурою будинку. На рис. 1.2 зображено централізовану систему, де всі функції керуються центральним контролером.

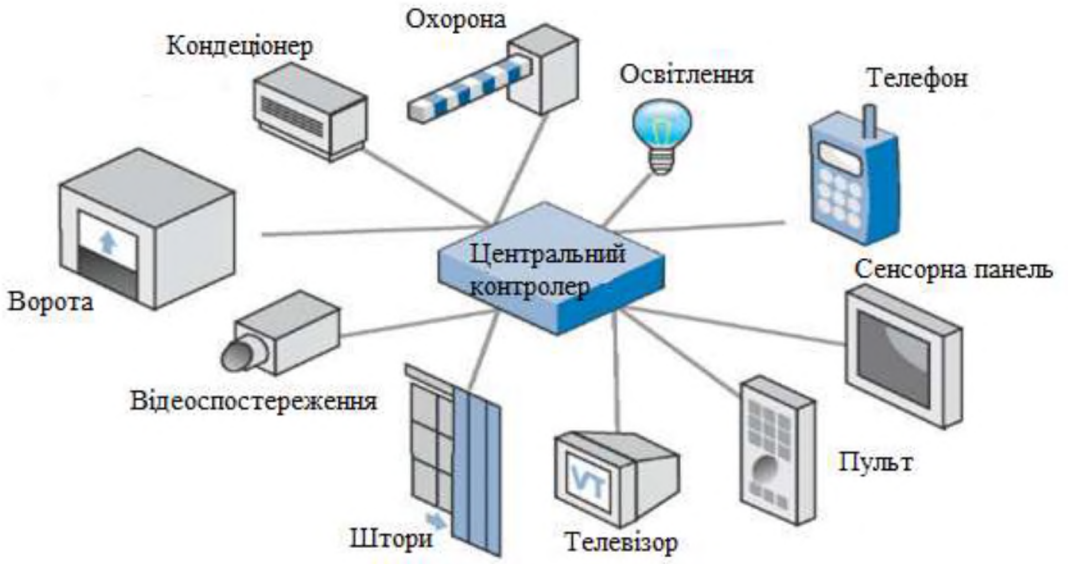

Рисунок 1.2 – Контролер керування системою розумного будинку [26]

Основні функції контролера включають [27]:

- управління виконавчими пристроями після отримання сигналів від зовнішніх датчиків;
- керування кліматом;
- контроль протипожежної безпеки;
- охоронна сигналізація;
- контроль доступу;
- облік споживання енергоресурсів.

Контролер може мати різні варіанти реалізації: він може бути багатофункціональним або має можливість розширення іншими модулями. Функції комутації також можуть передаватися іншим модулям замість обладнання на контролері. Ці контролери підключаються до Інтернету, що дозволяє управляти системою з різних пристроїв, таких як мобільні телефони, планшети або комп'ютери. Вони можуть виступати як комп'ютери з певною кількістю портів.

У контролері передбачається можливість програмування або управління комплексом, а також оснащення кнопками або сенсорною панеллю для управління. Для програмування часто використовують спеціальне програмне забезпечення або навіть окреме середовище розробки.

1.6 Обґрунтування вибору математичного апарату моделювання

Розв'язання задачі кваліфікаційної роботи передбачає створення моделі системи, що вивчається. Моделювання значно скорочує час і витрати на розробку, оскільки дозволяє аналізувати багато варіантів і підвищує ефективність новоствореної системи [28]. Існує різноманіття методів моделювання, кожен з яких відзначається своєю областю застосування. Методи моделювання базуються на теорії подібності, яка ґрунтується на трьох основних теоремах і кількох додаткових положеннях, спираючись на визначальні критерії. Ці критерії встановлюють масштаби, які пов'язують параметри моделі з оригіналом та висвітлюють основні властивості модельованого явища.

Показовим прикладом ефективного використання моделей і методів моделювання є аналіз кібербезпеки та готовності складних систем [28]. Цей аналіз проводять на всіх етапах життєвого циклу, але основна робота зазвичай відбувається на етапі проєктування.

Виділяють наступні методи математичного моделювання при аналізі кібербезпеки і готовності складних систем [29, 30]:

– аналіз видів і наслідків відмов;

– аналіз діагностичного дерева відмов;

– аналіз за допомогою блок-схеми надійності;

– лямбда-метод;

– марковський аналіз;

– ймовірнісно-фізичний аналіз.

Марковські випадкові процеси мають широке практичне застосування в різних областях, включаючи теорію і практику [31]. Ці процеси корисні для оцінки впливу атак на безпеку інформації у випадку, коли такі атаки становлять рідкісні та незалежні події.

Серед методів математичного моделювання для аналізу готовності та доступності системи з урахуванням атак і дефектів використовується марковський аналіз. Цей метод ґрунтується на теорії марковських процесів та використовується для оцінки складних систем і стратегій виявлення дефектів та вразливостей.

Марковський аналіз переважно ґрунтується на індуктивному методі аналізу, що працює від деталей до загального [32]. Він використовується для оцінки функцій складних систем і стратегій виявлення дефектів та вразливостей. Так, при марковському моделюванні, функція готовності буде визначатися за формулою:

$$Pg(t) = \sum P_i(t);\ i : S_i \in S_P;\ S_P \in M\{S_P, S_i\}\ , \tag{1.1}$$

де Pi – ймовірність знаходження системи в працездатних станах (рис. 1.3), причому як непрацездатний розглядаються стан, в якому відбуваються оновлення або заходи діагностики та усунення вразливостей.

Треба відзначити, що припущення, що лежать в основі застосування теорії марковських процесів, можуть не відповідати реальним умовам функціонування, зокрема у виправленні дефектів та вразливостей [33]. Це вимагає додаткових заходів для обґрунтування математичного підґрунтя досліджень.

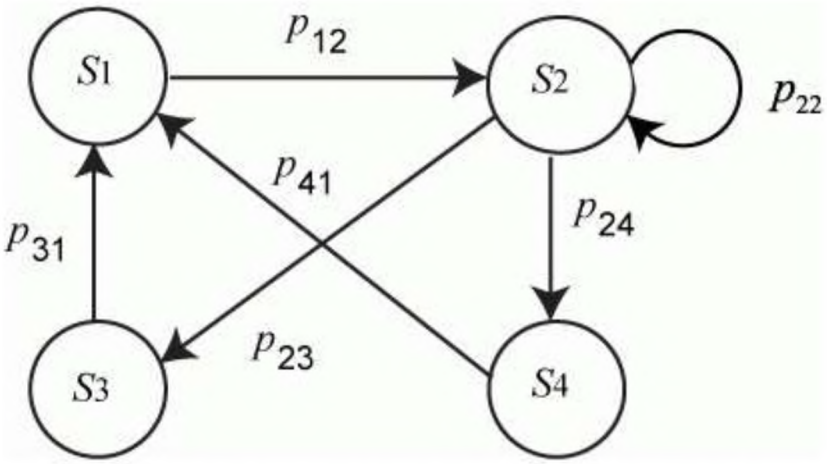

Рисунок 1.3 – Приклад марковського графу станів системи

Методика досліджень формується на основі логіки послідовності вирішення поставлених завдань та відповідності обраному математичному апарату. Такий підхід включає наступні етапи [34]:

– аналіз сформульованих завдань і визначення їх логічного зв'язку, що випливає з загальної мети дослідження та очікуваних результатів;

– визначення послідовності вирішення наукових і практичних завдань;

– обґрунтування та вибір математичного інструментарію для розв'язання завдань, враховуючи їх унікальні особливості;

– вирішення задач, аналіз отриманих результатів на наявність суперечностей, розробка рекомендацій для їх впровадження та подальшого розвитку.

Для проведення досліджень доцільно використовувати методи системного аналізу, векторного аналізу, теорії графів (рис. 1.3), теорії множин, теорії ймовірностей та комбінаторного аналізу.

Дерева атак – це схеми, що показують можливі шляхи атаки на систему [35]. У сфері інформаційних технологій вони використовуються для опису потенційних загроз комп'ютерним системам і методів реалізації цих загроз через атаки. Однак їхнє використання не обмежується лише аналізом інформаційних систем [20]. На рис. 1.4 зображено загальну структуру дерева атак.

Використання дерев атак розширюється на комп'ютерні системи управління. Крім того, дерева атак використовуються для розуміння загроз, що виникають у

фізичних системах. Дерева атак – це багаторівневі схеми, що складаються з кореня, листя та гілок [36]. Аналіз вузлів проводиться знизу вгору. Дочірні вузли – це умови, які мають виконуватися, щоб батьківський вузол також знаходився в активному стані. Коли корінь досягає активного стану, атака вважається успішною. Кожен вузол може бути активним лише через свої прямі нащадки [20].

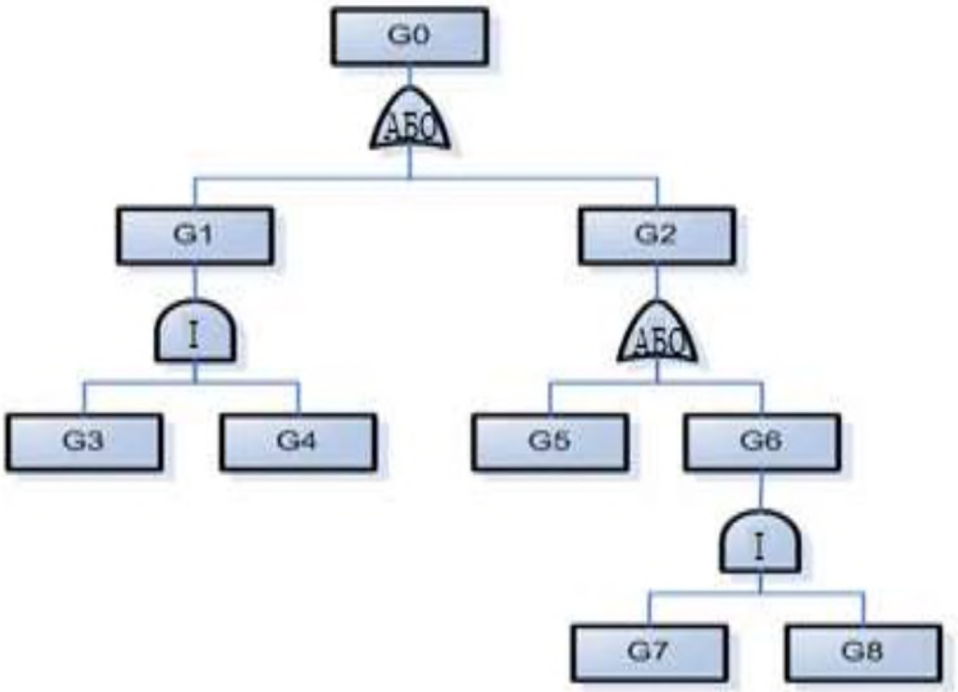

Рисунок 1.4 – Модель системи у вигляді дерева атак

Вузол може також бути дочірнім для іншого вузла; у такому випадку для успішної атаки потрібно кілька кроків. Також важливо враховувати, що атака, яка описана у вузлі, може очікувати, поки одна чи кілька з безлічі атак, що згадуються у дочірніх вузлах, будуть успішно виконані. Дерева атак можуть мати зв'язки типу АБО чи І між нащадками вузла [37].

Дерева атак також асоціюються зі створенням дерева помилок. Методика побудови дерева помилок використовує булеві вирази для встановлення умов, за яких дочірні вузли гарантують виконання батьківських вузлів. Хоча можливо розрахувати ймовірності для вузлів, у реальності точні оцінки ймовірності часто недоступні або обчислення вимагають великих витрат. У випадку динамічної

комп'ютерної безпеки, коли беруть до уваги атаки, випадкові події не є незалежними.

Дерева атак можуть стати дуже складними, особливо при аналізі конкретних атак. Повне дерево атак може включати сотні або тисячі різних шляхів, кожен з яких призводить до успішної атаки. Однак, навіть у такому складному вигляді, ці дерева корисні для виявлення потенційних загроз та способів їх уникнення. Дерева атак можуть служити для формулювання стратегій забезпечення інформаційної безпеки.

Висновки до розділу 1

У першому розділі було проведено аналіз ключових показників якості системи розумного будинку. Було розглянуто історію розвитку цієї технології, концепцію розумного будинку та його складові частини. Розумний будинок представляє собою складну систему високотехнологічних пристроїв, спрямованих на забезпечення певного рівня комфорту.

Також проведений аналіз ринку систем розумного будинку. Цей ринок є дуже перспективним для технологічних компаній, оскільки кожного року збільшується попит на такі системи та їх сервіси. У розділі розглянуто основні функції контролера системи розумного будинку. Це програмне забезпечення дозволяє керувати кліматом, освітленням, водопостачанням та надає можливість віддалено контролювати події в будинку.

На основі цього аналізу сформульована задача дослідження: розробка моделі кібербезпеки системи розумного будинку на основі параметрів контролю доступу у формі дерева атак. Загальне завдання розділено на три часткові задачі, дві з яких розглянуті у наступних розділах.

РОЗДІЛ 2
КОНТЕКСТНА МОДЕЛЬ КОНТРОЛЮ ДОСТУПУ ДЛЯ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ СИСТЕМ РОЗУМНОГО БУДИНКУ

2.1 Формальна та мандатна моделі контролю доступу

Опишемо базову рольову модель [38]. Введемо поняття множини суб'єктів S та об'єктів O доступу. Позначимо їх члени як S_i та O_i

$$s_i \in S, o_i \in O. \quad (2.1)$$

Введемо поняття множини ролей R, що призначаються суб'єктам. Членами цієї множини є елементи r_i:

$$r_i \in R. \quad (2.2)$$

Також, користувачі можуть виконувати певні операції, що задаються множиною операцій P_i:

$$P_i \in P. \quad (2.3)$$

При функціонуванні рольової моделі контролю доступу використовуються такі функції:

1. Функція отримання списку ролей користувача:

$$Roles(s_i) = \{\text{ролі, призначені користувачу } s_i\}.$$

2. Функція отримання списку операцій, доступних для ролі:

$$Ops(r_i) = \{\text{операції, пов‘язані із роллю } r_i\}.$$

3. Функція отримання допустимих для участі операцій над об'єктом:

$$Perms(r_i, o_j) = \{\text{операції над } o_j, \text{ доступні ролі } r_i\}.$$

4. Функція перевірки можливості виконання операції суб'єктом над об'єктом:

$$Execute(s_i, o_j, p_k) = \exists r_m \in Roles(s_i): p_k \in Ops(r_m), p_k \in Perms(r_m, o_j).$$

Перші три функції використовуються для отримання інформації про поточну конфігурацію системи й застосовують модель контролю доступу. Остання функція використовується під час взаємодії суб'єктів доступу із системою. Саме ця функція служить для контролю доступу та визначення можливості суб'єкта виконання запитуваної операції. Модель може бути розширена за допомогою ієрархії ролей

(рис. 2.1). Цей механізм спрощує процедуру додавання нових ролей, базуючись на вже існуючих, і розширює їх повноваження.

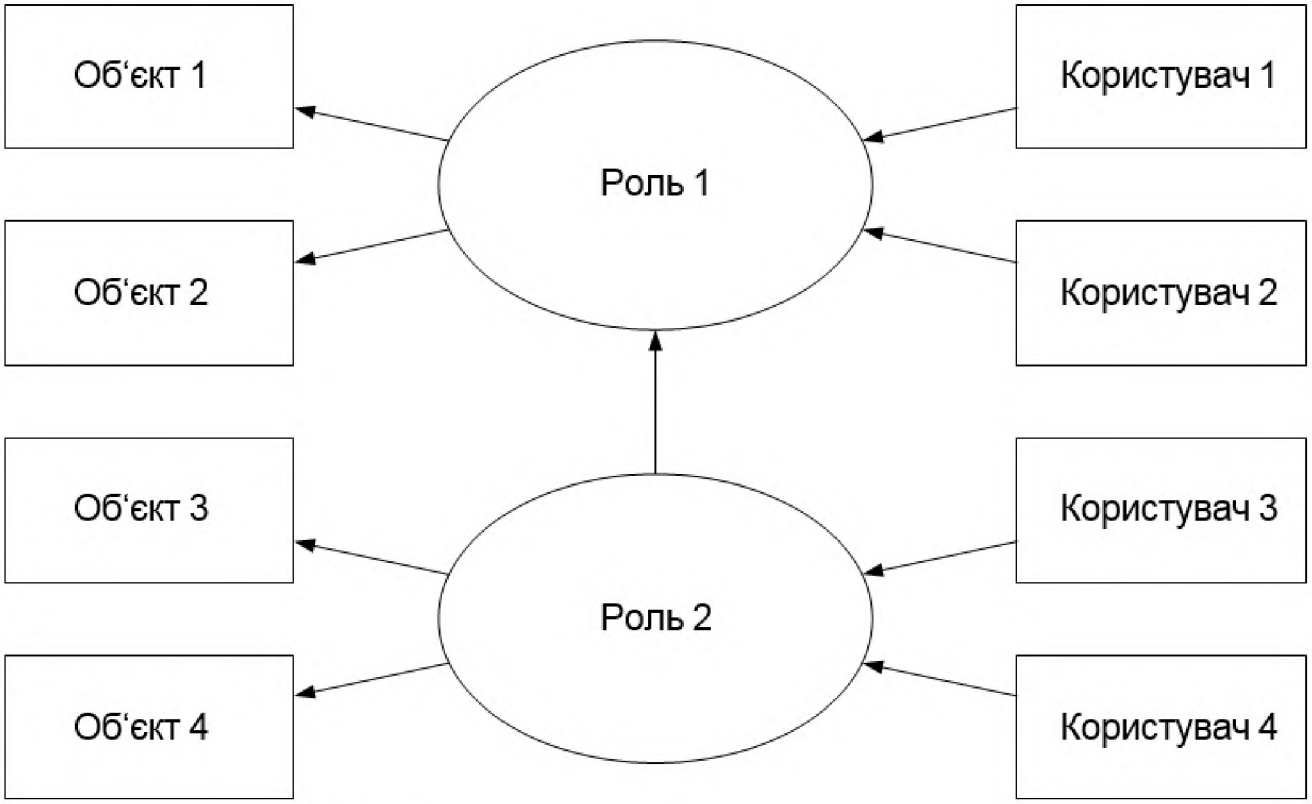

Рисунок 2.1 – Ієрархія у рольовій моделі контролю доступу

На наведеному рисунку 2.1 показано приклад можливостей, що надаються механізмом ієрархії ролей. Так, на основі ролі 1 була створена роль 2. У результаті користувачі 3 і 4 мають доступ до об'єктів 3 і 4, а також до тих об'єктів, до яких мають доступ користувачі з призначеною роллю 1.

Далі були проаналізовані дві мандатні моделі – модель Белла-Лападули та модель Біба. Модель Белла-Лападули призначена для розмежування доступу до інформації, що захищається, і заснована на правилах документообігу, прийнятих у багатьох державних установах [39].

Формальний опис моделі складається з наступних визначень:

– S – множина суб‘єктів доступу,

$$S = \{s_1, s_2, \dots, s_k\};$$

– O – множина об‘єктів доступу, включає в себе множину суб‘єктів,

$$O = \{o_1, o_2, \dots, o_m, s_1, s_2, \dots, s_k\};$$

– R – множина прав доступу: $R = \{read, write\}$;

– L – множина рівнів безпеки, $L = \{l_1, l_2, \dots, l_n\}$;

– $\Lambda = (L, \leq, \cdot, \otimes)$ – решітка рівнів безпеки;

– V – множина станів системи, що складається з упорядкованих пар (F, M), де $F: S \cup O \to L$ – функція, що зіставляє з суб'єктами та об'єктами рівні секретності,

M – матриця прав доступу.

Розглянемо докладніше решітку рівнів безпеки **Λ**. Вона включає множину рівнів безпеки L, оператор часткового несуворого відношення порядку $\leq$, оператори отримання найменшої верхньої і найбільшої нижньої меж $\cdot$ і відповідно $\otimes$. Оператори $\cdot$ и $\otimes$ мають наступний опис:

$$l_1 \cdot l_2 = l \Leftrightarrow l_1, l_2 \leq l \wedge \forall l' \in L: (l' \leq l) \Longrightarrow (l' \leq l_1 \vee l' \leq l_2);$$

$$l_1 \otimes l_2 = l \Leftrightarrow l \leq l_1, l_2 \wedge \forall l' \in L: (l' \leq l_1 \wedge l' \leq l_2) \Longrightarrow (l' \leq l).$$

На рис. 2.2 зображено решітку рівнів безпеки, і відображено допустимі права доступу для суб'єктів різних рівнів.

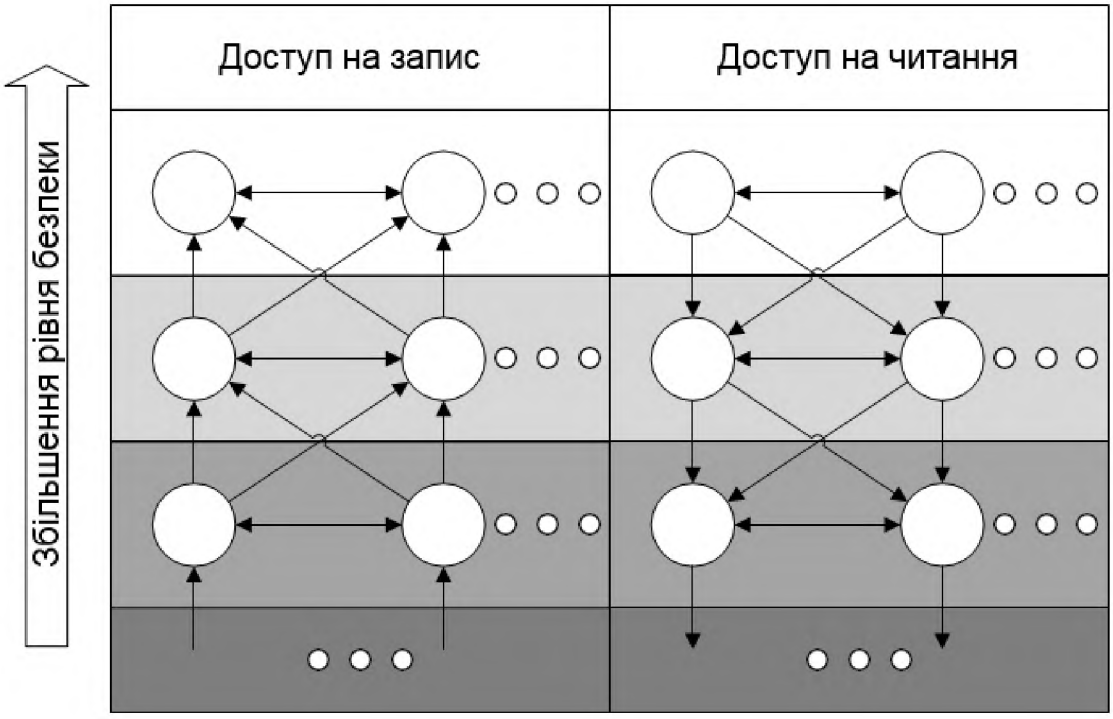

Рисунок 2.2 – Решітка рівнів безпеки

Безпека моделі Белла-Лападули ґрунтується на двох наступних правилах.

Просте правило безпеки полягає у тому, що суб'єкт з рівнем безпеки l_s має доступ $read$ до об'єкту з рівнем безпеки l_o тільки якщо виконується умова:

$$l_o \leq l_s. \tag{2.4}$$

Друге (*) правило безпеки полягає у тому, що суб'єкт з рівнем безпеки l_s має доступ $write$ до об'єкту з рівнем безпеки l_o тільки якщо виконується умова:

$$l_s \leq l_o. \tag{2.5}$$

За виконання цих двох умов стан системи вважається безпечним. Якщо умови виконуються всіма станами системи, то система вважається безпечною.

Модель Біба є модифікацією попередньої моделі і визначається аналогічно. У цій моделі акцент робиться на забезпеченні цілісності даних, а не конфіденційності.

Наведемо короткий опис основних визначень:

– S – множина суб‘єктів, $S = \{s_1, s_2, \dots, s_k\}$;

– O – множина об‘єктів, $O = \{o_1, o_2, \dots, o_m, s_1, s_2, \dots, s_k\}$;

– R – множина прав доступу: $R = \{read, write\}$;

– L – множина рівнів цілісності, $L = \{l_1, l_2, \dots, l_n\}$;

– $\Lambda = (L, \leq, \cdot, \otimes)$ – решітка рівнів цілісності.

У моделі Біба також діють два базові правила.

Просте правило цілісності полягає у тому, що суб'єкт із рівнем цілісності l_s має доступ $read$ до об'єкту з рівнем цілісності l_o тільки якщо виконується умова:

$$l_s \leq l_o. \tag{2.6}$$

Друге (*) правило цілісності полягає у тому, що суб'єкт із рівнем цілісності l_s має доступ $write$ до об'єкту з рівнем цілісності l_o тільки якщо виконується умова:

$$l_o \leq l_s. \tag{2.7}$$

За виконання цих двох умов у всіх станах система вважається безпечною.

2.2 Контекстні моделі контролю доступу

Моделі контекстного контролю доступу відрізняються від тих, що використовують статичну інформацію про систему, що захищається. Вони включають методи використання інформації про стан системи в момент виконання операції, що підлягає контролю [40]. Ця інформація відома як контекст. Контекст визначається як сукупний стан пристроїв системи під час виконання операції. Цей стан описується набором параметрів, які можуть бути зчитані зовнішніми пристроями. Контекст складається з різноманітних елементів, що описують вимірювані параметри системи, наприклад, поточний час, розташування пристрою та активні завдання.

Також контекст може включати історичні дані про окремі параметри. У такому разі необхідно зберігати інформацію про зв'язок параметрів для відстеження їхньої зміни з часом. Історичні дані не суперечать визначенню контексту як стану системи в момент виконання операції, оскільки поточний стан системи є результатом певних подій, що відбулися раніше і відображаються в історії зміни контексту [41].

Однією з важливих задач при розробці контекстних моделей контролю доступу є збір та аналіз контексту. Для ефективного використання контексту для контролю доступу потрібно підготувати отримані дані з сенсорів. Цей процес включає такі етапи:

1. Нормалізація даних: приведення даних з різних пристроїв до єдиної форми для подальшої обробки системою контролю доступу.

2. Фільтрація даних: виділення суттєвих даних для подальшого аналізу.

3. Кореляція даних: співвіднесення різних значень, які залежать одне від одного.

Контроль доступу в контекстних моделях базується на розширених правилах, що враховують контекст. Ці правила дозволяють точніше визначати права доступу, ґрунтуючись на відомих параметрах контексту. Наприклад, мова політик контролю доступу, описана в роботі [42].

Контекстна модель розширює рольову модель, використовуючи контекст для динамічного розподілу ролей. Це досягається за допомогою трьох наборів правил політики контролю доступу:

– TrustValue визначає рівень довіри користувача до елементів контексту на основі їхніх значень.

– Assign_role розподіляє ролі на основі рівнів довіри до окремих елементів контексту.

– Permissions відповідає за відповідності ролей наборам привілеїв.

Політика контролю доступу має бути передбачена наперед, виходячи з умов і вимог до системи. Особливу увагу слід приділяти складанню правил, які визначають рівень довіри користувачеві в залежності від контексту [43]. Процес надання доступу можна представити у вигляді схеми, яку показано на рис. 2.3.

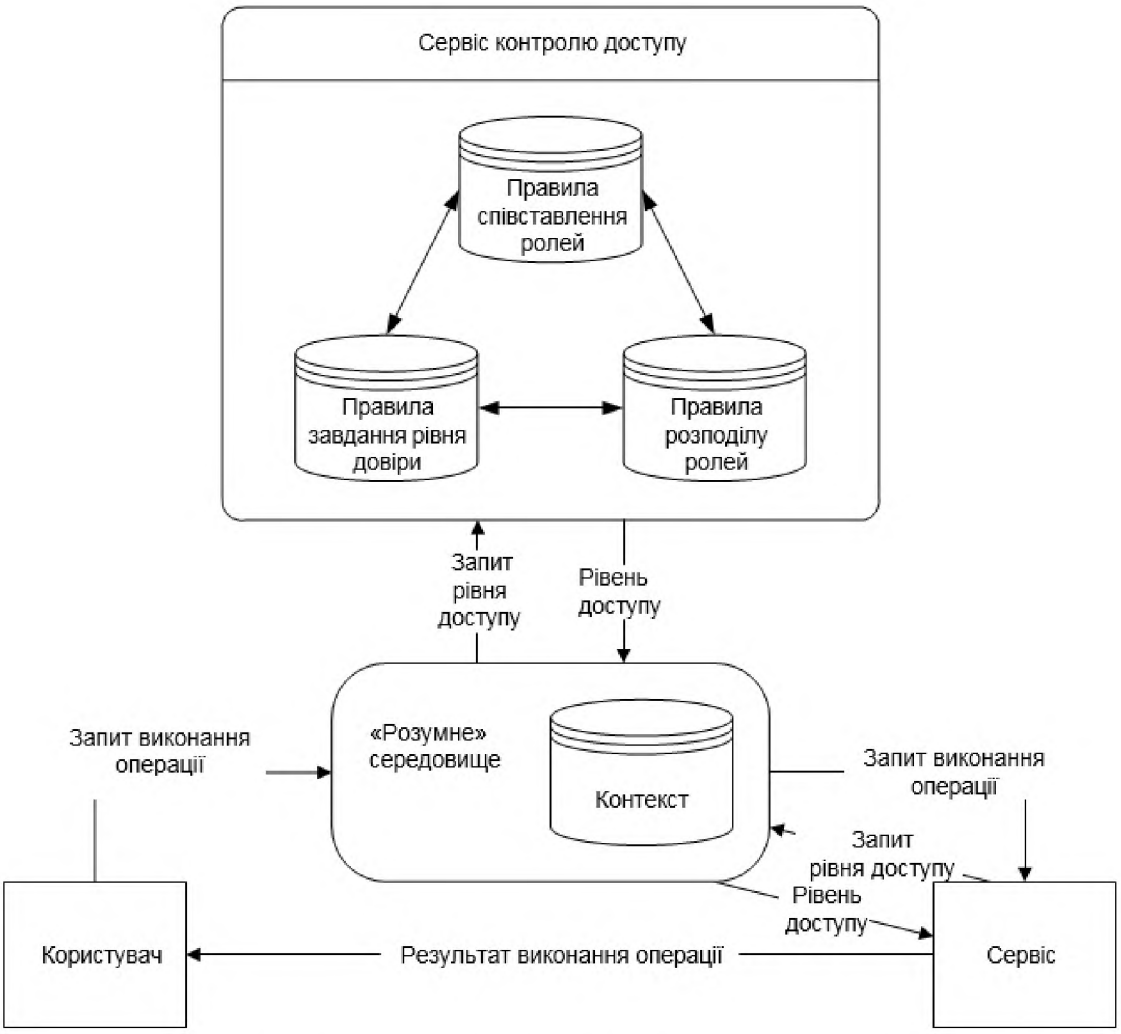

Рисунок 2.3 – Модель контекстного механізму контролю доступу

На рис. 2.3 виділено основні компоненти процесу отримання доступу. Тут показано, як користувач отримує доступ до сервісу в «розумному» середовищі. Користувач спочатку взаємодіє з пристроями, що утворюють це середовище, через спеціальне обладнання, яке об'єднує їх у єдину мережу та збирає контекстову інформацію.

Система включає брокера (сервіс) доступу, який перевіряє права користувача перед виконанням операцій [44]. Коли користувач надсилає запит на виконання операції з певним рівнем доступу, брокер контролю доступу перевіряє права користувача та контекст.

Алгоритм визначення прав доступу має кілька етапів:

1. Перевірка цифрового підпису контексту користувача. Якщо підпис правильний, відбувається перехід до наступного кроку. В іншому випадку доступ відмовляється.

2. Обчислення рівнів довіри користувача на основі елементів контексту.

3. Перевірка можливості присвоєння ролей користувачеві, виходячи з рівнів довіри до контексту.

4. Перевірка прав доступу до запитаного сервісу, враховуючи список ролей користувача.

Якщо доступ наданий, виконується потрібна операція. Результат операції повертається брокеру, а сервіс повідомляє про результат виконання операції.

2.3 Архітектура системи розумного будинку з контекстною моделлю доступу

Наразі найпоширенішим методом створення систем розумного будинку є використання спеціалізованих центральних вузлів. Приклади таких систем представлені у продуктах від компаній Philips і Samsung [12], які ґрунтуються на пристроях, що діють як мости між різними «розумними» пристроями та керуючими програмами через інтерфейси, надані цими мостами.

Хоча мережі, побудовані на технологіях типу ZigBee, можуть використовувати стільникову топологію, зазвичай використання топології «дерево», яка також підтримується ZigBee, є більш поширеним варіантом. У цьому випадку вузли в мережі розділяються на три групи:

1. Центральний вузол: це кореневий пристрій мережі, що забезпечує основні функції спілкування пристроїв та забезпечує безпеку.

2. Проміжні вузли: крім автоматизації розумного будинку, вони розширюють мережу, передаючи повідомлення між іншими вузлами.

3. Кінцеві вузли: ці вузли виконують різні функції для автоматизації розумного будинку.

Рис. 2.4 демонструє приклад такої мережі, де проміжні вузли дозволяють розширювати мережу за допомогою послідовного підключення.

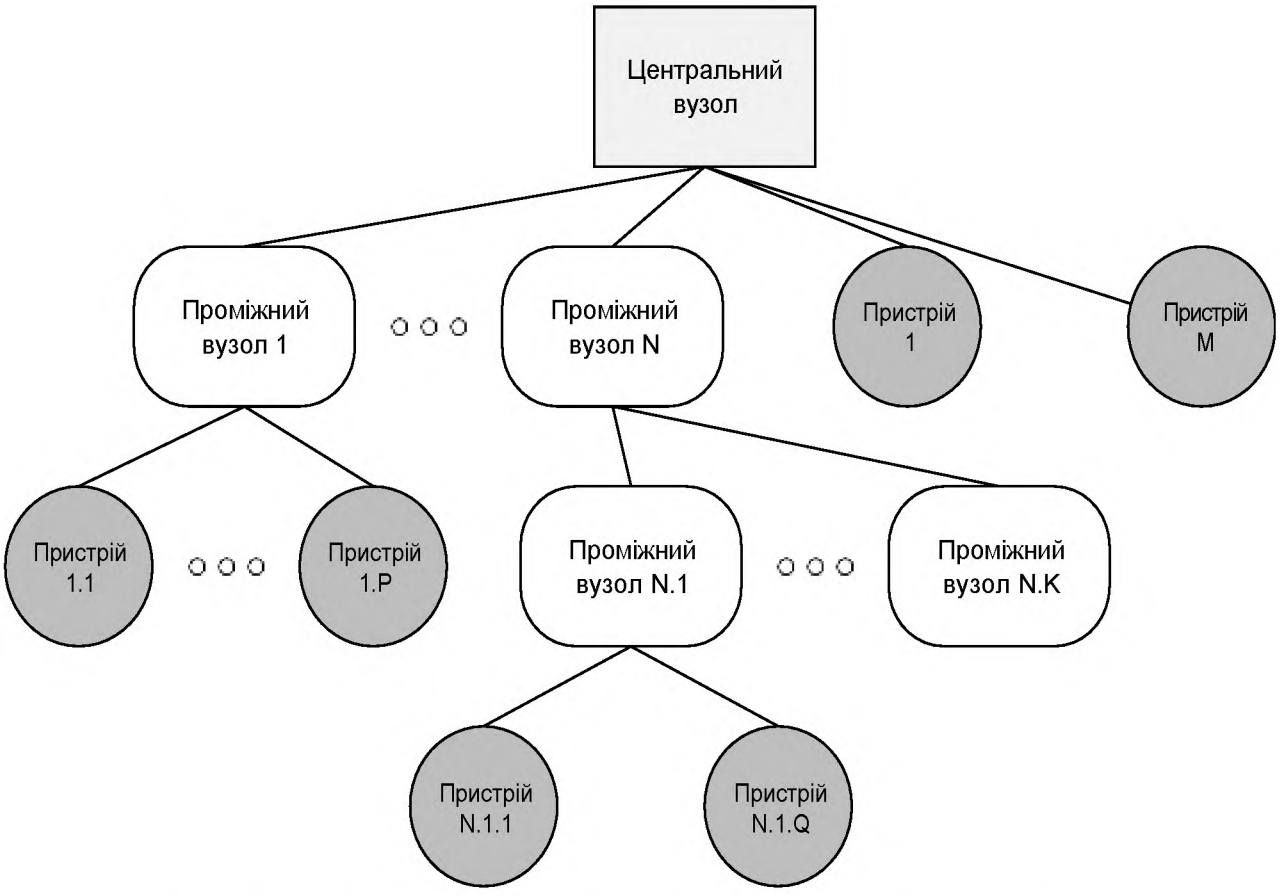

Рисунок 2.4 – Деревоподібна топологія мережі розумного будинку

При використанні такої топології пропонується налагодити шлюз контролю доступу для централізованого керування доступом в системі розумного будинку. Цей метод спрощує процес впровадження контекстного контролю доступу та відповідає підходам, використовуваним у реальних системах розумного будинку.

Рисунок 2.5 – Процес функціонування системи контролю доступу розумного будинку

На рис. 2.5 показаний процес функціонування системи контролю доступу, який розбивається на чотири етапи:

1. Аналіз мережевої взаємодії: пристрої взаємодіють через шлюз контролю доступу.

2. Застосування правил контролю доступу: запити між пристроями фільтруються відповідно до правил контролю доступу.

3. Оновлення контексту: ґрунтується на отриманих запитах та прямому діалозі з пристроями. Журнали значень параметрів та операцій зберігаються як частина контексту, оскільки відображають процес переходу до нового стану.

4. Відновлення правил контролю доступу: на основі оновленого контексту та політики контекстної моделі відбувається оновлення та активація правил контролю доступу.

Для шлюзу контролю доступу розроблена архітектура, представлена рис. 2.6. Компоненти шлюзу об'єднуються у три підсистеми – контролю доступу, зберігання політик і контексту, та взаємодії з мережею.

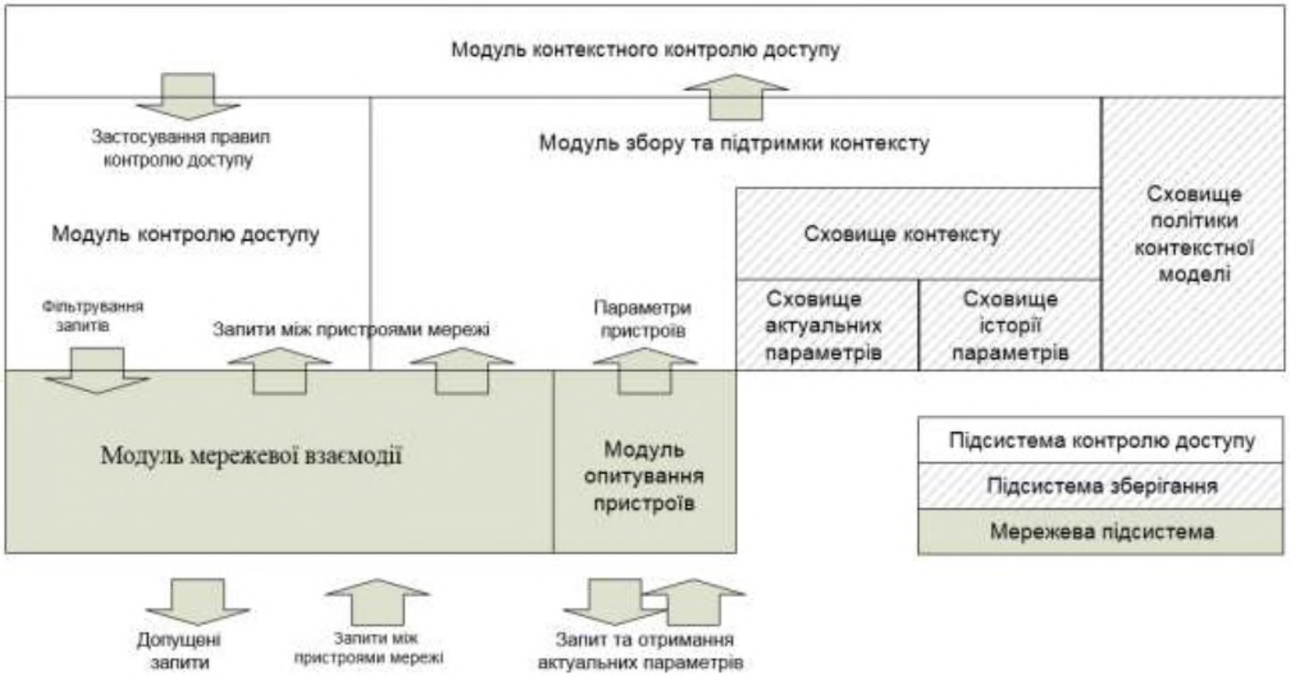

Рисунок 2.6 – Архітектура шлюзу контроля доступу

Далі розглянуто принцип роботи основних компонент системи. Однією з ключових компонент системи, що втілює контекстну модель контролю доступу, є підсистема, відповідальна за зберігання та зібрання контексту. В рамках архітектури системи розумного будинку існує кілька можливостей для виконання цих операцій.

Для розробленої моделі запропоновано декілька методів збору контексту:

– аналіз запитів між пристроями;

– опитування пристроїв.

Рисунок 2.7 – Алгоритм аналізу запитів для оновлення контексту

Перший метод є більш складним у втіленні, але й більш ефективним, оскільки не створює додаткового навантаження на пристрої мережі розумного будинку. Він ґрунтується на тому, що шлюз контролю доступу отримує всі запити між пристроями і відповідає за їхню фільтрацію згідно з правилами контролю доступу.

Отже, шлюз контролю доступу визначає, чи потрібно виконувати запит та отримує результат його виконання. Детальний алгоритм аналізу запитів зображено на рис. 2.7.

Другий метод призначений для отримання параметрів пристроїв, які змінюються без запитів, таких як різноманітні датчики, що вимірюють певні параметри. У відміну від першого методу, який оновлює контекст безперервно під час роботи системи, другий вимагає періодичного опитування пристроїв.

2.4 Метод контролю доступу на основі мандатної моделі Біба

З трьох розглянутих раніше моделей контролю доступу (рольової, мандатної моделі Белла-Лападули, мандатної моделі Біба) для розробленої моделі контролю доступу було обрано модель цілісності Біба. Це вибір здійснено з урахуванням таких вимог:

– модель контролю доступу повинна дозволяти групувати пристрої за різними правами доступу;

– процес зміни прав доступу окремих суб'єктів повинен бути простим для автоматизації;

– початкове конфігурування має бути мінімальним, щоб не переважати політику контекстної моделі.

Перші дві вимоги у певній мірі відповідають всім розглянутим моделям. Проте варто зауважити, що перерозподіл суб'єктів за ролями загалом може бути складнішим завданням, ніж порівняння рівнів доступу. Остання вимога стала вирішальною при виборі між моделями.

З урахуванням класичної мандатної моделі, важливо лише знати підтримуване системою число рівнів доступу та їхні відносини. Рольова модель

потребує визначення ролей, їх повноважень і, у окремих випадках, ієрархії ролей, що створює вимоги до конфігурування системи.

Підхід до рішення завдання забезпечення захисту від скомпрометованих пристроїв у моделі цілісності Біба полягає у накладанні обмежень на окремі пристрої для запобігання виконанню операцій, що можуть впливати на безпеку системи. Це вказує на можливість використання рівнів цілісності пристроїв для реалізації таких заходів.

Контроль доступу у системі базується на мандатній моделі. Пропонована модель є гібридною, розширюючи мандатну модель до контекстної за допомогою додавання механізму динамічного призначення рівнів з урахуванням контексту.

Зручність використання моделі контролю доступу залежить від підходу до визначення політик управління доступом. В запропонованій моделі політики формулюються через дві групи правил:

– Обмеження параметрів системи за умовами.

– Дозволені операції, які суб'єкти можуть виконувати над об'єктами.

Правила, що описують дозволені операції суб'єктів над об'єктами, виглядають наступним чином:

$$Execute(s, o, a), \tag{2.8}$$

де s – суб'єкт, o – об'єкт, a – операція. Операція a входить у множину операцій, допустимих до виконання над об'єктом. Для отримання цієї та іншої, пов'язаної з об'єктом, інформації визначено такі функції:

$Operations(o) = \{$операції, допустимі над об‘єктом $o\}$;

$Description(o, a) = \{$зміна параметрів o після операції $a\}$.

Перша функція слугує для перевірки правил на достовірність. Друга функція використовується для аналізу контексту та виявлення конфліктів, які можуть виникнути при можливому виконанні операцій.

У запропонованій моделі припускається, що правила доступу визначаються

пристроями, які додаються до системи. Це дозволяє уникнути ручної конфігурації системи контролю доступу, що особливо важливо для застосувань, таких як забезпечення безпеки в розумному будинку. Для цього пристрої розділяються на певні класи з визначеними операціями, які вони можуть виконувати. Кожен пристрій потім отримує список прав доступу в термінах класів пристроїв.

При підключенні нового пристрою до системи розумного будинку відбувається оновлення політики контролю доступу. Як показано на рис. 2.8, цей процес полягає у поєднанні списків правил пристроїв системи.

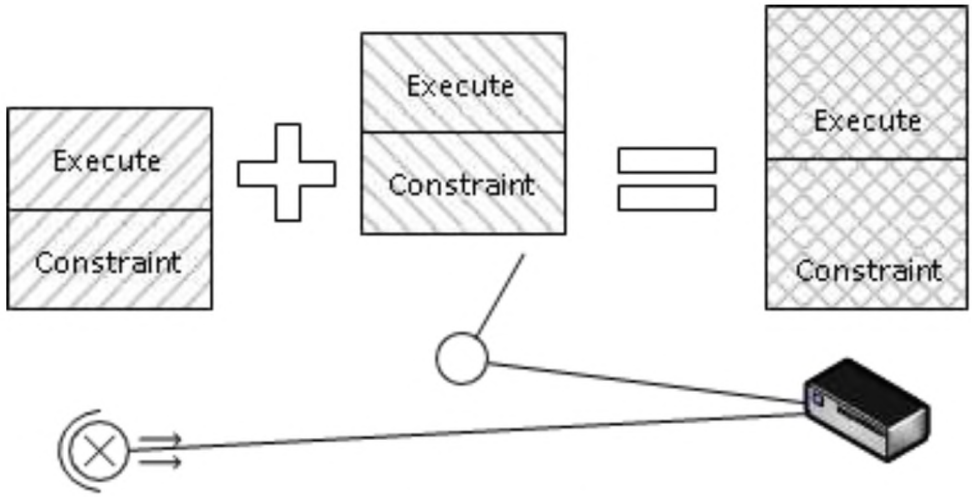

Рисунок 2.8 – Формування правил контролю доступу

Обмеження на параметри можуть бути виражені в одному з наступних варіантів:

$$Constraint(context\_condition, param, value\_expression);$$

$$Constraint(context\_condition, conditional\_expression).$$

Перша форма дозволяє жорстко встановлювати значення, яке має приймати параметр $param$ при виконанні умови $context\_condition$. У цьому форматі правил дозволяється також створювати нові елементи контексту на підставі визначень пристроїв у системі.

Друга форма створює можливість накладання різноманітних обмежень $conditional\_expression$ на елементи контексту. Ці обмеження є частинами опису безпечного стану системи розумного будинку.

2.5 Алгоритми функціонування системи

Для коректної роботи контекстної моделі контролю доступу важливі не лише збір, зберігання та періодичне оновлення контексту. На роботу системи безпеки впливають алгоритми подальшої обробки зібраної інформації та складання правил контролю доступу.

Алгоритм аналізу контексту призначений для пошуку протиріч, що виникають у системі розумного будинку. На вхід алгоритм аналізу контексту отримує політику контролю доступу та актуальний контекст. Основним завданням аналізу контексту розробленої моделі контролю доступу є виявлення протиріч між потенційно можливими операціями у системі розумного будинку із встановленими обмеженнями на контекст. Тому алгоритм аналізу контексту служить визначення списку діючих обмежень $Constraint$ та правил типу $Execute(s, o, a)$, що суперечать чинним обмеженням. Схематично даний процес зображено рис. 2.9.

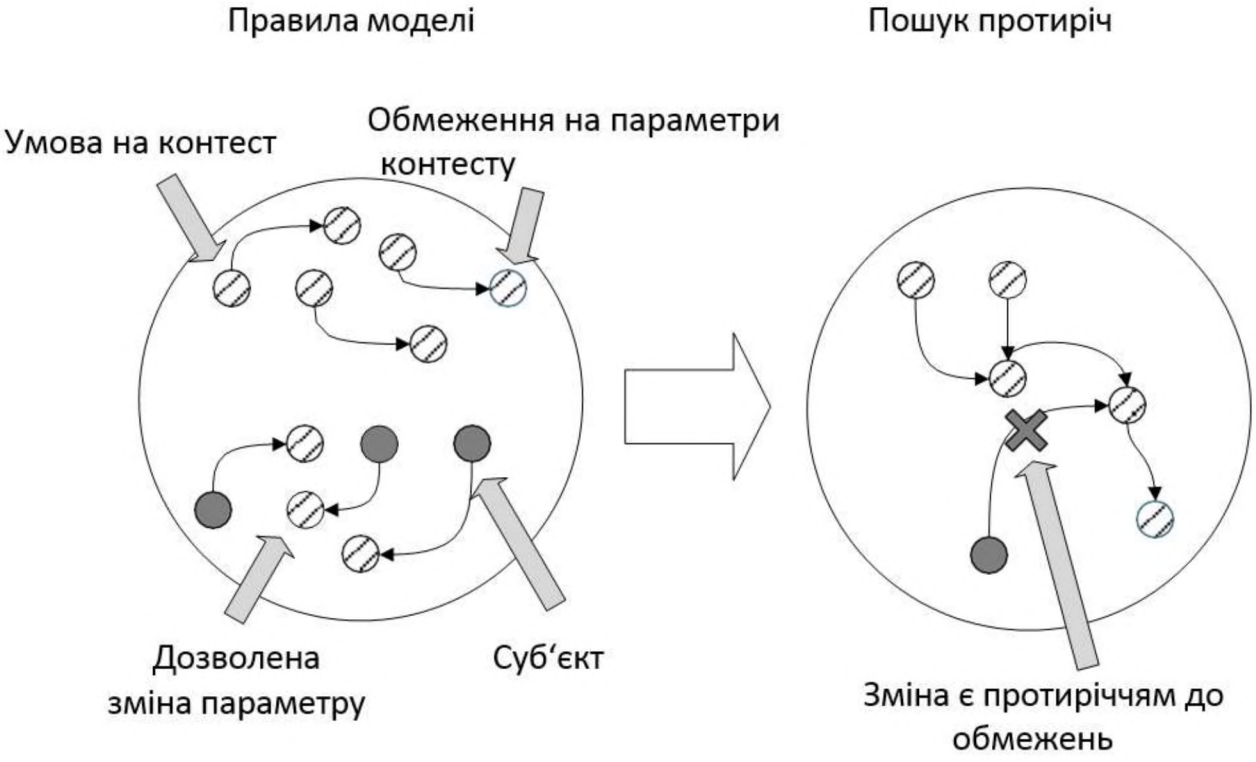

Рисунок 2.9 – Процес аналізу контексту

Важливою особливістю є те, що правила типу $Constraint(condition, parameter, value\_expression)$ дозволяють доповнювати контекст новими параметрами, що обчислюються. Така можливість дозволяє створювати складні ланцюжки правил $Constraint$, які можуть доповнюватися правилами нових пристроїв, що вводяться в систему. Цю особливість необхідно враховувати під час проведення аналізу контексту. Для опису алгоритму з урахуванням цього вводиться поняття задіяного правила $Constraint$.

На початку роботи алгоритму проводиться складання списку правил $Constraint$, що підлягають обробці. Спочатку список складається з усіх правил, що є в системі. Задіяним правилом $Constraint(condition, \dots)$ називається правило зі списку, яке підлягає обробці, умова $condition$ якого виконується у поточному контексті. Таке правило застосовується безпосередньо при залученні. Подальші дії залежать від типу правила.

Якщо це правило має вигляд $Constraint(condition, parameter, value\_expression)$, тоді в контекст додається новий параметр $parameter$ зі значенням $value\_expression$, одержуваним у результаті обчислення. Якщо такий параметр вже присутній у контексті, то нове значення додається до списку можливих значень. Також нове значення додається до списку обмежень на контекст. Суперечності на цьому етапі не визначаються, оскільки вони впливатимуть на призначення рівнів цілісності.

Якщо задіюється правило у вигляді $Constraint(condition, conditional\_expression)$, тоді умова додається до списку обмежень на контекст.

Після задіяння правила воно виключається зі списку правил, що підлягають обробці. Описаний процес повторюється до тих пір, поки наступна ітерація не очікує задіяних правил серед списку правил, які підлягають обробці.

Подальша обробка складеного списку обмежень на контекст полягає у визначенні правил $Execute$, що порушують ці обмеження. Для цього для кожного правила, що використовується в системі у вигляді $Execute(s, o, a)$, виконується перевірка змін параметрів, які викликаються, з використанням функції

$Description(o, a)$. Усі правила, виконання операцій згідно з якими порушує умови, накладені на контекст, додаються до списку правил, що викликають протиріччя. Результатом роботи алгоритму є перелік обмежень на контекст і перелік правил доступу, що суперечать їм.

Рисунок 2.10 – Процес аналізу контексту

Виконання алгоритму призначення рівнів цілісності (рис. 2.10) є заключним етапом роботи розробленої контекстної моделі контролю доступу під час оновлення стану. На вхід алгоритм отримує політику контекстної моделі та перелік

правил доступу, що суперечать обмеженням на контекст. На основі даного списку визначається множина пристроїв, яким необхідно заборонити доступ, і множина пристроїв, доступ до яких потрібно заборонити, щоб усунути протиріччя з контекстом. Зміна правил контролю доступу вибирається на користь такої зміни, що призводить до найменшого впливу на систему як кількість заборонених операцій.

Висновки до розділу 2

У розділі розглянуто особливості моделей системи контролю доступу для розумних будинків. Проаналізовано формальну та мандатні моделі. З трьох моделей (рольової, мандатної моделі Белла-Лападули, мандатної моделі Біба) для розробленої моделі контролю доступу було обрано модель цілісності Біба. Підхід до рішення завдання забезпечення захисту від скомпрометованих пристроїв у моделі цілісності Біба полягає у накладанні обмежень на окремі пристрої для запобігання виконанню операцій, що можуть впливати на безпеку системи. Це вказує на можливість використання рівнів цілісності пристроїв для реалізації таких заходів.

Застосування контексту відкриває нові можливості у сфері контролю доступу. Розглянувши опис моделі контролю доступу, можна виділити декілька переваг контекстних моделей:

– гнучке налаштування політики контролю доступу;

– адаптація системи до оточуючого середовища.

Ці властивості моделі можуть ідеально використовуватися у системах розумного будинку, оскільки вони чудово відповідають особливостям цих систем серед інших засобів автоматизації.

РОЗДІЛ 3
МОДЕЛЬ АРХІТЕКТУРИ РОЗУМНОГО БУДИНКУ ТА ЙОГО КІБЕРБЕЗПЕКИ

3.1 Вхідні обмеження моделі кібербезпеки та її допущення

У системі розумного будинку можна виділити різні підсистеми, такі як електропостачання, освітлення, управління електроприладами, розваги, зв'язок, клімат-контроль, система безпеки та система загального управління. Компоненти архітектури цієї системи різняться залежно від конкретної області їх застосування. При аналізі автоматизації будівель і систем управління [23] можна розділити архітектуру системи на чотири рівні, як показано на рис. 3.1.

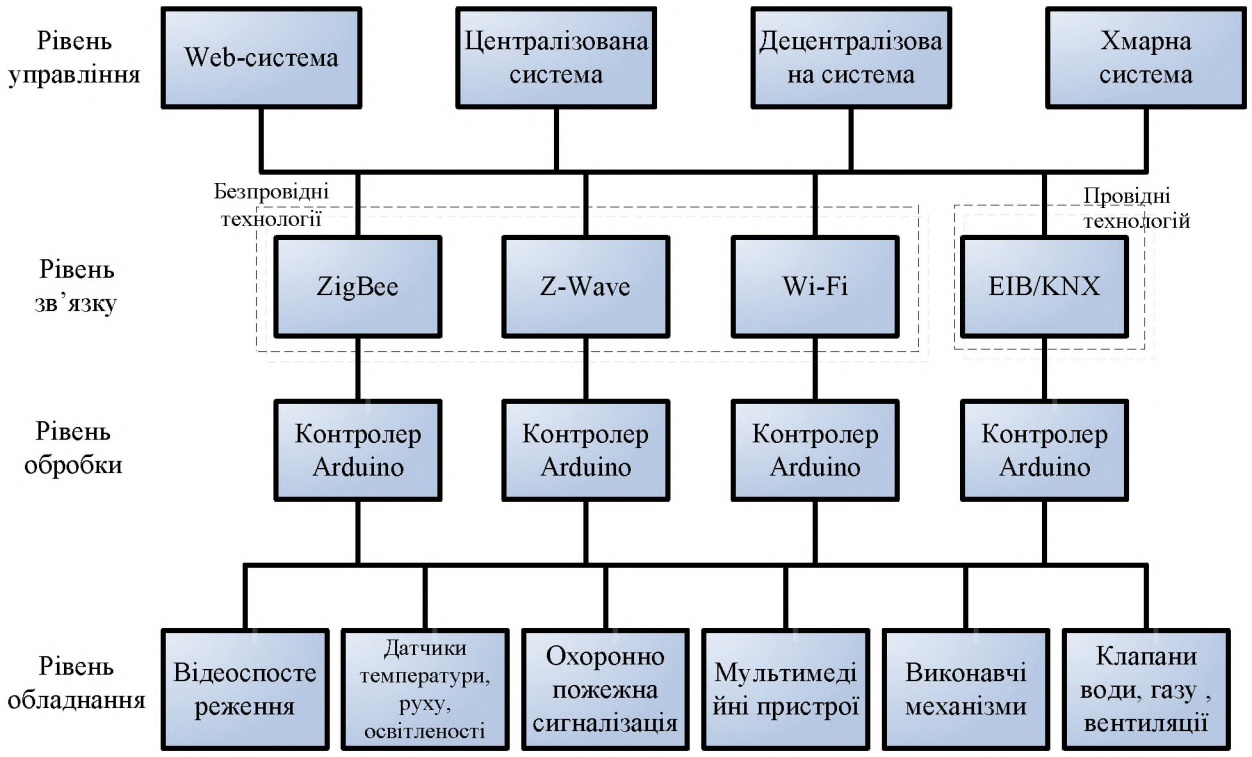

Рисунок 3.1 – Рівні архітектури системи розумного будинку

Перший рівень системи розумний будинок – це управлінський рівень, де відбувається диспетчерське управління, адміністрування та робота з базами даних. Це місце, де сервер обробляє надходження даних від контролерів та різних

пристроїв в будинку. Важливою функцією цього рівня є взаємодія між користувачами та виконавчими системами, що гарантує оптимальну роботу всієї системи. Основна увага при розгляді цього рівня – надійність баз даних та безпека віддаленого доступу. Це важливо для забезпечення безперебійності роботи системи та захисту конфіденційної інформації користувачів.

Другий рівень – рівень зв'язку, що відповідає за обмін інформацією між компонентами системи. У ньому досліджуються різні типи зв'язку, включаючи провідні та безпровідні технології. Провідні зв'язки вважаються надійнішими та безпечнішими, в той час як безпровідні – простіші та доступніші для використання.

Третій рівень – рівень обробки, де контролер збирає та аналізує інформацію від датчиків та контрольно-вимірювального обладнання. Він приймає рішення та видає команди виконавчим механізмам відповідно до програмованих алгоритмів.

Четвертий рівень – це рівень обладнання, де знаходяться датчики, вимірювальні пристрої та виконавчі механізми. Ці компоненти безпосередньо впливають на функціонування системи та її стан.

Для подальшого аналізу розглядаються усі чотири рівні архітектури системи розумного будинку, показані на рис. 3.1.

3.2 Обґрунтування вхідних даних моделі кібербезпеки, що описують атаки на розумний будинок

Оцінка кожної атаки здійснюється з ряду важливих позицій:

- першочерговою є мета, яку намагається досягти зловмисник;
- вид атаки залежить від цілей зловмисника;
- для реалізації атак, використання прикладного рівня семирівневої моделі ISO/OSI є найбільш відповідним через його потенційні можливості та складність;
- багато операційних систем мають певні вразливості, які можуть бути використані для цілеспрямованих атак;
- атаки можуть бути цільовими або розподіленими на декілька пакетів;

– фактор зворотного зв'язку важливий для зловмисників;

– щоб уникнути виявлення або досягти кращої ефективності, зловмисники можуть вибирати різні початкові умови виконання, тип впливу або рівень автоматизації;

– число джерел атаки та кількість з'єднань може різнитися в залежності від мети атаки.

Вектор атаки – це шлях або метод, яким зловмисник може отримати доступ до комп'ютера або мережевого сервера з метою отримання несанкціонованого доступу [10]. Класифікація атак на системи розумного будинку базується на аналізі атак на комп'ютерні системи (рис. 3.2).

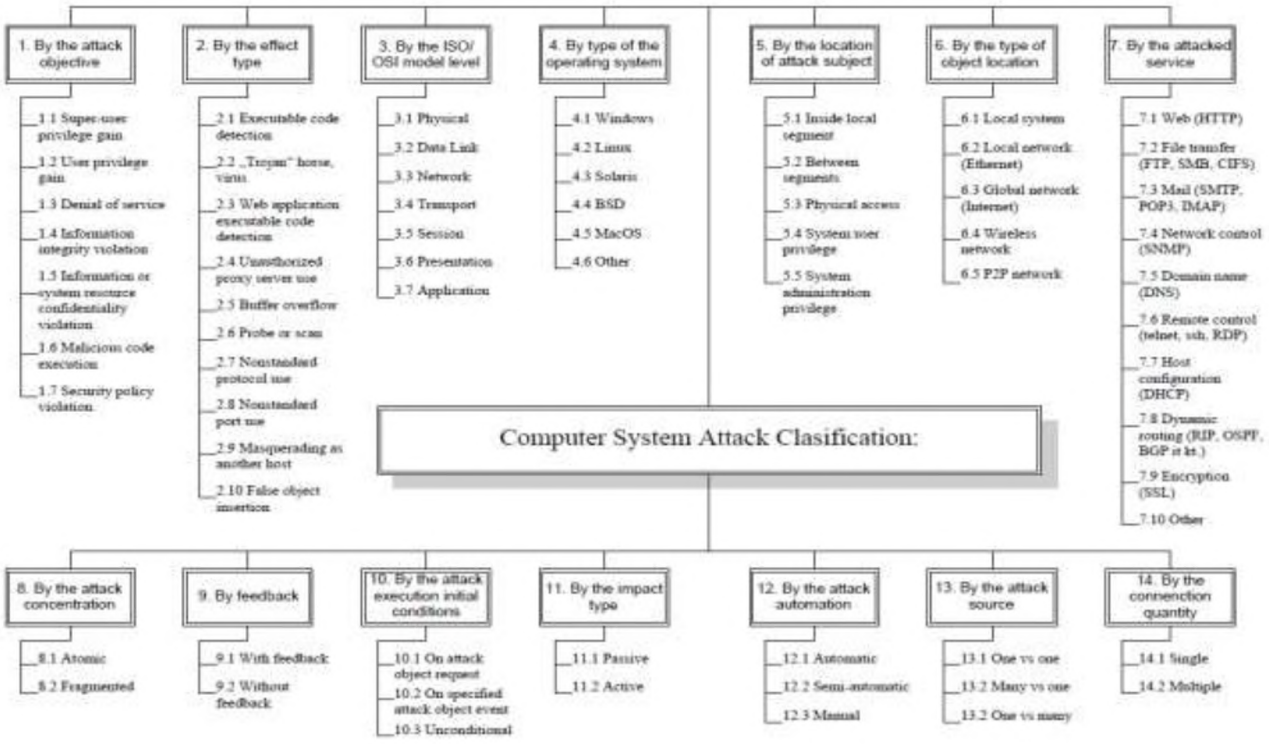

Рисунок 3.2 – Класифікація атак на комп'ютерні системи

В залежності від контексту використання систем автоматизації існують різні категорії осіб, які можуть генерувати атаку, включаючи співробітників (користувачів), фахівців (хакерів), конкурентів, професійних зловмисників та спецслужби. Рівень підготовки атакуючих визначається на трьох рівнях: низькому,

середньому та високому. Організованість атакуючих може бути у вигляді одиночних осіб або груп.

Якщо говорити про складність атак та інструментарій експлуатації вразливостей, вони можуть розподілятися за такими рівнями:

Рівень 1: не потребує особливих інструментів чи знань, може статися випадково;

Рівень 2: мінімальний рівень, використовуються загальні інструменти;

Рівень 3: вимагає технічної підготовки, інструменти можна знайти в Інтернеті;

Рівень 4: вимагає інженерної підготовки, використовуються специфічні інструменти;

Рівень 5: вимагає академічної підготовки, використовуються спеціально розроблені інструменти;

Рівень 6: лабораторне моделювання складних атак.

Ці рівні відображають не лише різний рівень технічних навичок, але й ступінь спрямованості або складності атаки, що може залежати від мети або цілісності самої системи. Наприклад, атаки високого рівня підготовки частіше мають специфічну мету й вимагають вдосконаленої експертизи для їх виконання. Класифікація атак із застосуванням таких категорій спростить аналіз вразливостей та дозволить краще розуміти, які заходи безпеки слід вживати для захисту системи від різноманітних загроз.

3.3 Опис моделі кібербезпеки системи розумного будинку

Основними вимогами користувача вважаються забезпечення надійності системи та захисту кібербезпеки системи та інформації протягом усього періоду експлуатації. Система розумного будинку складається з чотирьох рівнів архітектури, при цьому компоненти на кожному рівні впливають на загальну надійність системи. Відмови на рівні зв'язку та обробки безпосередньо впливають

на готовність системи, оскільки перешкоджають передачі команд та ізолюють виконавчі прилади.

Найбільш доступний для атак є рівень зв'язку, в той час як рівень обробки, хоча менш доступний, може мати вразливості на етапі проєктування та виробництва компонентів. Це безпосередньо впливає на загальну кібербезпеку системи. Крім того, компоненти на інших рівнях управління та обладнання також впливають на готовність системи. Але поодинокі відмови їх компонентів не призводять до відмови системи в цілому.

Архітектура інформаційно-керуючих систем розумних будинків може значно відрізнятися залежно від сфери застосування. На рис. 3.3 наведено дерево атак на верхньому рівні системи розумного будинку.

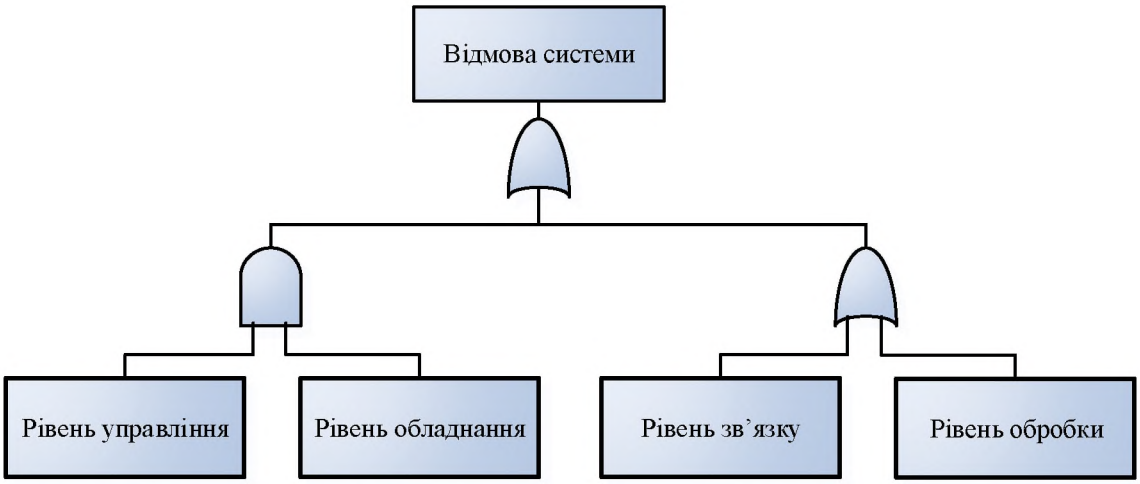

Рисунок 3.3 – Дерево атак верхнього рівня

Оцінка надійності та кібербезпеки системи відбувається за допомогою аналізу дерев атак, який досліджує можливі шляхи відмови системи [24]. Цей метод допомагає розробникам зрозуміти, як система працює у випадку наявності слабких місць у проєкті, які можуть використовуватися зловмисниками. Аналіз дерева атак показує, які компоненти системи потребують підвищення вимог для забезпечення кібербезпеки та надійності протягом усього терміну експлуатації.

Під час використання цього методу система аналізується у контексті експлуатаційного середовища, щоб знайти всі можливі шляхи виникнення відмови.

Для побудови моделі у вигляді дерева атак використовують два типи вентилів (I/AND, АБО/OR). Подія після вентиля «І» відбувається тільки при одночасному прояві змін на вході вентиля, тоді як подія на виході вентиля «АБО» виникає при будь-якій зміні стану компонента.

На рисунку 3.3 представлено дерево верхнього рівня аналізу архітектури системи розумного будинку, що складається з чотирьох рівнів. Дерево атак дозволяє визначити пріоритети кожного рівня у випадку події повноцінної відмови системи. Рівень зв’язку та обробки мають вищий пріоритет та пряме з'єднання через вентиль «АБО» до стану відмови системи. Інші рівні з'єднані через вентиль «І» та не викликають відмови системи, якщо подія відбувається тільки на одному з них.

При аналізі кібербезпеки системи вибирається конкретна подія, така як відмова або атака на компонент системи, і визначаються причини, необхідні для досягнення цієї мети [19]. Аналізуються всі безпосередні, необхідні та достатні причини відмови компонентів системи шляхом покрокового аналізу дерева зверху-вниз, поки не буде досягнуто межі дозволеної моделі, тобто події відмови. Обов'язково потрібно розглянути сценарії кібератак, враховуючи усі можливі цілі атаки на систему та її компоненти на кожному рівні.

3.4 Обчислення результуючих показників

Компоненти архітектури системи розумного будинку в вигляді дерев атак нарощуються поступово знизу-вверх. Модель дерева атак для всієї системи, представлена на рис. 3.4.

Опишемо ймовірність відмов в системи і співвідношення між компонентами, P – ймовірність відмови:

$P_2 = 1-(1- P_6)(1- P_7)(1- P_8)(1- P_9))(1- P_{10}))(1- P_{11})$;

$P_3 = 1-(1- P_{12})(1- P_{13}))(1- P_{14}))(1- P_{15})$;

$P_4 = 1-(1- P_{16})(1- P_{17}))(1- P_{18}))(1- P_{19})$;

$P_5 = 1-(1- P_{20})(1- P_{21})(1- P_{22})$;

$P_x = P_2 \cdot P_3$;

$P_1 = 1-(1- P_x)(1- P_4)(1- P_5)$.

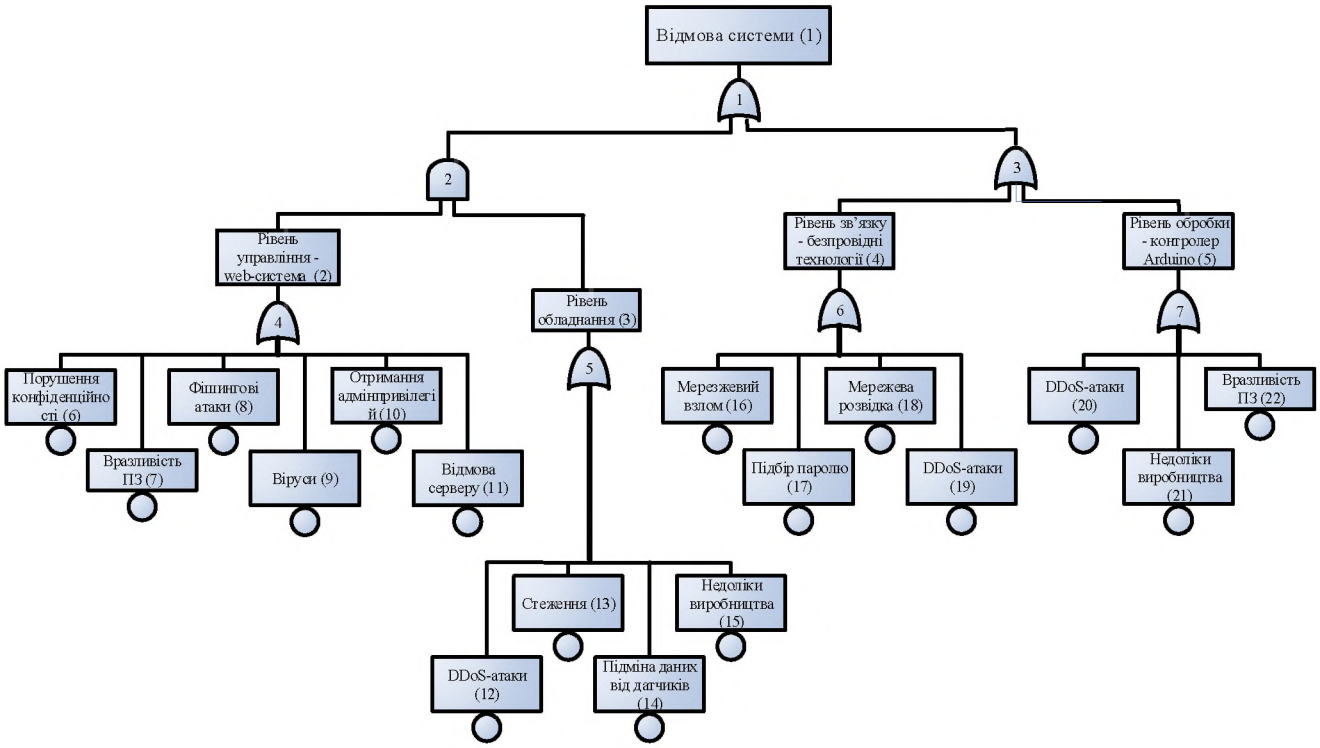

Рисунок 3.4 – Модель дерева атак для оцінки статичних показників кібербезпеки

Таблиця 3.1 – Розрахунок ймовірності відмови системи розумного будинку

Рівень архітектури	№ компоненту	Клас вразливостей компоненту	Імовірність успішної атаки	Імовірність відмови системи в результаті зовнішніх впливів (атак на вразливості) 0,000197016
Управління	1	Порушення конфіденційності (6)	0,0000524	
	2	Вразливість ПЗ (7)	0,0000438	
	3	Фішингові атаки (8)	0,0000184	
	4	Віруси (9)	0,0000218	
	5	Отримання адмінпривілегій (10)	0,0000237	
	6	Відмова серверу (11)	0,0000186	
Обладнання	7	DDoS-атаки (12)	0,0000646	
	8	Стеження (13)	0,0000425	
	9	Підміна даних від датчиків (14)	0,0000471	
	10	Недоліки виробництва (15)	0,0000281	
Зв'язку	11	Мережевий взлом (16)	0,0000195	
	12	Підбір паролю (17)	0,0000312	
	13	Мережева розвідка (18)	0,0000214	
	14	DDoS-атаки (19)	0,0000495	
Обробки	15	Вразливість ПЗ (20)	0,0000227	
	16	Недоліки виробництва (21)	0,0000146	
	17	DDoS-атаки (22)	0,0000381	

У таблиці 3.1 показано розрахунки ймовірності повноцінної відмови системи, а також ймовірності відмови кожного з її компонентів.

3.5 Реалізація систем розумних будинків на базі Arduino

Arduino – це потужний і доступний інструмент для розробки електронних пристроїв, що відрізняється від ПК своєю спрямованістю на взаємодію з навколишнім світом. Ця плата містить мікроконтролер та пропонує відкриту для програмування апаратну платформу, а також спеціальне середовище розробки для взаємодії з різними фізичними пристроями, такими як датчики, перемикачі, двигуни та індикатори.

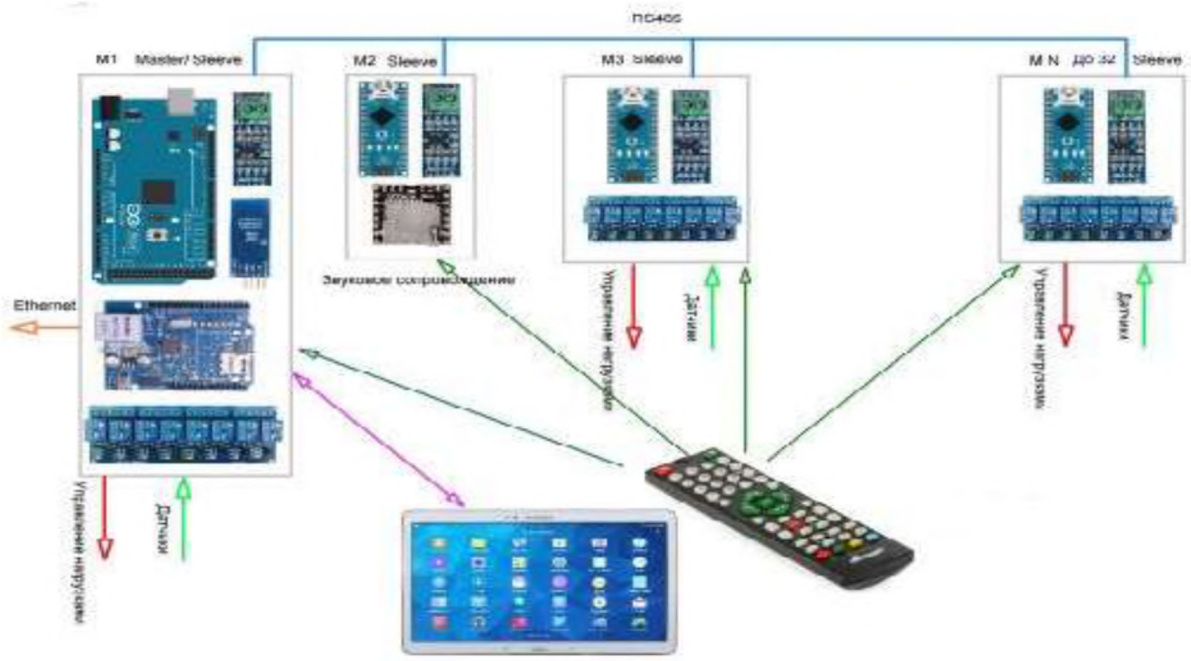

Рисунок 3.5 – Прототип розумного будинку на Arduino

У порівнянні з іншими мікроконтролерами, такими як MIT's Handyboard, Parallax Basic Stamp, Phidgets, Netmedia's BX-24 і подібними, Arduino (рис. 3.5) має свої переваги:

1. Ціна – готові модулі Arduino можна придбати за менше ніж 50 доларів, або зібрати власноруч за дуже доступну ціну.

2. Кросплатформеність – програмне забезпечення Arduino працює на різних операційних системах, таких як Windows, Macintosh OSX і Linux, що відрізняє його від інших систем, орієнтованих лише на Windows.

3. Просте середовище програмування – середовище Arduino просте для початківців, але водночас гнучке для просунутих користувачів. Воно базується на середовищі програмування Processing.

4. Відкрите програмне забезпечення – програмне забезпечення Arduino має відкритий вихідний код, що дозволяє досвідченим програмістам змінювати і розширювати його за допомогою C++ бібліотек.

5. Розширюване відкрите апаратне забезпечення – платформа Arduino заснована на мікроконтролерах Atmel ATmega8 і ATmega168, а схеми модулів опубліковані під ліцензією Creative Commons. Це дозволяє інженерам створювати свої версії пристроїв на основі Arduino і дозволяє збирати дослідні зразки для кращого розуміння їх принципів роботи й економії коштів.

Різноманітність плат Arduino обумовлює їх використання в розумному будинку. Актуальними на даний момент є версії плат Arduino UNO та Arduino MEGA 2560.

Arduino Uno – це пристрій, який базується на мікроконтролері ATmega328. Він має в собі всі необхідні компоненти для зручної роботи з мікроконтролером: 14 цифрових входів/виходів (6 з них можна використовувати як ШІМ-виходи), 6 аналогових входів, кварцовий резонатор на частоті 16 МГц, USB-роз'єм, роз'єм для живлення, роз'єм для внутрішньо-схемного програмування (ICSP) та кнопку скидання. Для початку використання пристрою досить просто підключити його до живлення через AC/DC-адаптер або батарейку або підключити до комп'ютера за допомогою USB-кабелю[6]. На рисунку 3.6 показано зовнішній вигляд плати Arduino Uno.

У ролі перетворювача інтерфейсів USB-UART використовується мікроконтролер ATmega16U2[6]. Arduino Uno може живитися як від USB, так і від зовнішнього джерела живлення, і вибір джерела відбувається автоматично.

Обсяг флеш-пам'яті ATmega328 становить 32 КБ (з яких 0.5 КБ використовує загрузчик). Крім цього, мікроконтролер має 2 КБ оперативної пам'яті SRAM і 1 КБ EEPROM, з якої можна читати або записувати інформацію за допомогою бібліотеки EEPROM [6].

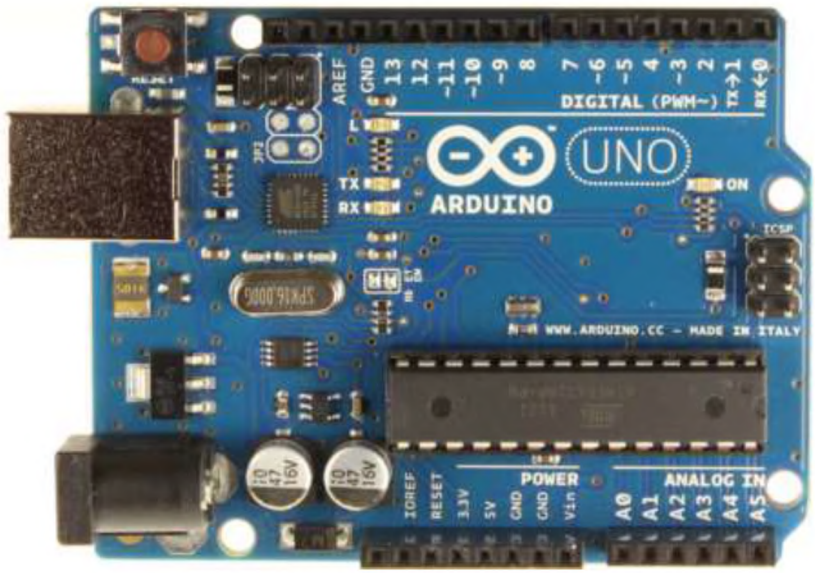

Рисунок 3.6 – Плата Arduino Uno [6]

Arduino Mega 2560 – це пристрій, який базується на мікроконтролері ATmega2560. У його арсеналі – все необхідне для зручної роботи з мікроконтролером: 54 цифрові входи/виходи (із них 15 можна використовувати як ШІМ-виходи), 16 аналогових входів, 4 UART (апаратні приймачі для послідовних інтерфейсів), кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм для живлення, роз'єм ICSP для внутрішньо-схемного програмування та кнопка скидання. Для початку використання пристрою досить просто подати живлення від AC/DC-адаптера або батарейки або підключити його до комп'ютера за допомогою USB-кабелю. Arduino Mega сумісний з більшістю модулів розширення [7]. На рис. 3.7 показано зовнішній вигляд плати Arduino Mega.

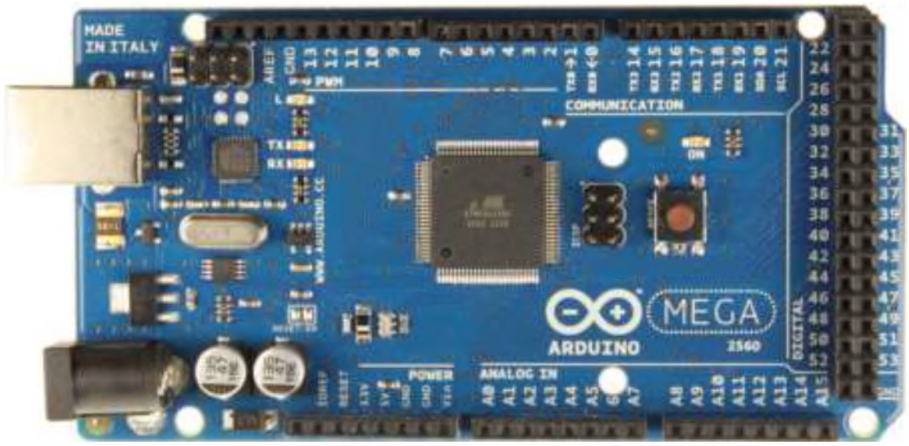

Рисунок 3.7 – Плата Arduino Mega 2560 [7]

У Arduino Mega 2560 для перетворення інтерфейсів USB-UART використовується мікроконтролер ATmega16U2 [7]. Arduino Mega може живитися як від USB, так і від зовнішнього джерела живлення, і вибір джерела відбувається автоматично [7].

Мікроконтролер ATmega2560 має 256 КБ флеш-пам'яті програм (з яких 8 КБ використовує загрузчик), 8 КБ оперативної пам'яті SRAM і 4 КБ EEPROM (для роботи з цією пам'яттю використовується бібліотека EEPROM) [7].

3.6 Техніко-економічне обґрунтування ефективності проєкту

Оцінка економічної ефективності досліджень кваліфікаційної роботи магістра визначає доцільність наукових досліджень та обгрунтовує вибір засобів. Мета роботи полягає у вивченні методів забезпечення безпеки для розумного будинку. Розробка надійної і ефективної системи потребує значних витрат часу, залежних від кваліфікації розробника та комплексності проєкту. Виконання цієї роботи розбивається на кілька етапів, оцінка тривалості яких базується на нормативах часу або експертних оцінках. Для оцінки загального часу науково-дослідної роботи корисно скласти таблицю, а також врахувати витрати часу наукового керівника у межах певного відсотка від сумарних витрат часу інженерів.

Розробку цієї інформаційної системи можна розділити на наступні етапи:

1. Постановка завдання.

2. Збір необхідної інформації та її подальше оброблення.

3. Прийняття рішень щодо оптимального вибору шляху для вирішення поставленої задачі.

4. Аналіз математичної моделі інформаційної системи.

5. Розробка алгоритму програми для цієї інформаційної системи.

6. Налаштування середовища розробки та роботи з вже готовою програмою.

7. Написання програмного коду.

8. Створення та оформлення документації (як електронної, так і паперової).

Для оцінки тривалості виконання окремих завдань часто використовуються нормативи часу або попередній досвід. Наприклад, на деяких підприємствах час на написання однієї операції складає від 0,5 до 1,6 годин, а на п'ять операцій (тривалість зміни) – 8 годин.

У разі їх відсутності звертаються до експертних оцінок по встановленню тривалості кожного етапу (стадії):

– при трьох оцінках:

$$T_{вс}=(t_{min}+4\cdot t_{н.й}+t_{max})/6,$$

– при двох оцінках:

$$T_{вс}=(3\cdot t_{min}+2\cdot t_{max})/5,$$

де $T_{вс}$ – очікуване (середнє) значення тривалості виконання етапу (стадії);

t_{min}, $t_{н.й.}$, t_{max} – відповідно мінімальна, найбільш імовірна і максимальна оцінки тривалості виконання етапу (стадії).

Для визначення загального часу, необхідного науково-дослідній роботі (НДР) або розробці програмного продукту, рекомендується узагальнити витрати часу на кожен окремий етап у таблиці 3.2. Якщо інформація про витрати часу наукового керівника на конкретні етапи обмежена, розумно враховувати їх в межах 5% від сумарного часу, витраченого інженерами на ці самі етапи.

Основуючись на Законі України «Про оплату праці», заробітна плата визначається як «винагорода, обчислена у грошовому виразі, яку виплачує власник або уповноважений ним орган за виконану роботу працівникові». Розмір заробітної плати залежить від складності та умов виконуваної роботи, професійних якостей

працівника, результатів його праці та господарської діяльності підприємства. Заробітна плата складається з основної та додаткової оплати праці.

Таблиця 3.2 – Основні етапи проєкту і час їх виконання

№ з/п	Етап	Середній час виконання етапу, год	
		Інженер	Керівник
1	Постановка задачі	3	10
2	Збір потрібної інформації і наступне її опрацювання	10	5
3	Прийняття рішень щодо вибору оптимального шляху розв'язання поставленої задачі	10	5
4	Аналіз математичної моделі інформаційної системи	15	12
5	Розробка алгоритму програми інформаційної системи	20	10
6	Налаштування середовища розробки і роботи вже готової програми	5	3
7	Написання програми	80	10
8	Написання і оформлення документації (електронної і паперової)	30	20
	Разом	160	60

Основна заробітна плата обчислюється за тарифними ставками, відрядними розцінками або посадовими окладами і не пов'язана з результатами господарської діяльності підприємства. Додаткова заробітна плата включає витрати, що не залежать від фактично відпрацьованого часу. Її розмір залежить від досягнутих та запланованих показників, умов виробництва та кваліфікації працівників. Фонд матеріального стимулювання, утворений за рахунок прибутку, є джерелом додаткової оплати праці.

Розрахунок основної заробітної плати включає пряму заробітну плату і доплати, що становлять у розрахунках 25% – 35% від прямої заробітної плати. При розрахунку заробітної плати важливо враховувати кількість робочих днів у місяці, що у 2023 році складала 21 день середньостатистичного місяця. Згідно з даними Міністерства соціальної політики України, при 40-годинному робочому тижні і 21 робочому дні в місяці, норма в годинах складає 168 годин на місяць.

Розмір місячних окладів керівника та інженерів визначається відповідно до чинних нормативів. Формула розрахунку основної заробітної плати наступна:

$$З_{осн}=Т_с \cdot К_г,$$

де $Т_с$ – тарифна ставка, грн.;

$К_г$ – кількість відпрацьованих годин.

Посадові оклади визначаються згідно з тарифною сіткою через множення окладу за розрядом на відповідний тарифний коефіцієнт. Якщо оклад вказаний з копійками, то числа до 0,5 відкидаються, а від 0,5 і більше – заокруглюються до цілої гривні. У 2023 році розрахунки по окладам виконуються відповідно до Закону України «Про Державний бюджет України на 2023 рік».

Мінімальна зарплата у 2023 році дорівнює прожитковому мінімуму для працездатних осіб, становлячи 3760 гривень щомісяця, а погодинно – 24,84 гривень. Припустимо, що інженер отримує 50 грн. за годину, а керівник – 80 грн. за годину. Оклади для керівника проєкту та інженера складають відповідно 80,0 грн./год. та 50,0 грн./год. Основна заробітна плата розраховується так:

$$З_{осн}=Т_{осн} \cdot К \text{ год.}$$

– Керівник проєкту:

$$З_{осн}=80 \cdot 60=4800 \text{ грн.}$$

– Інженер:

$$З_{осн}=50 \cdot 160=8000 \text{ грн.}$$

Додаткова заробітна плата становить 10–15 % від суми основної заробітної плати і визначається за формулою:

$$З_{дод}=З_{осн} \cdot К_{допл},$$

– Керівник проєкту:

$$З_{дод}=4800 \cdot 0.1=480 \text{ грн.}$$

– Інженер:

$$З_{дод}=8000 \cdot 0.1=800 \text{ грн.},$$

де $К_{допл.}$ – коефіцієнт додаткових виплат працівникам 0,1.

Звідси загальні витрати на оплату праці ($В_{О.П.}$) визначаються за формулою, і становлять:

$$В_{ОП} = З_{ОСН} + З_{ДОД},$$

– Керівник проєкту:

$$В_{ОП} = 4800 + 480 = 5280 \text{ грн.}$$

– Інженер:

$$В_{ОП} = 8000 + 800 = 8800 \text{ грн.}$$

Отже, усього сума складає 14080 грн. Крім того, треба врахувати відрахування на соціальні заходи:

18% – податок на доходи фізичних осіб;

1,5% – військовий збір;

22% – єдиний соціальний внесок.

Разом вони становлять 41,5%. Отже, загальна сума відрахувань на соціальні заходи обчислюється за формулою й складе:

$$Вс.з. = Ф_{ОП} \cdot 0,415$$

$$Вс.з.= 14080 \cdot 0,415 = 5843,20 \text{ грн.},$$

де $Ф_{ОП}$ – фонд оплати праці, грн.

Проведені розрахунки витрат на оплату праці наведені у табл. 3.3.

Таблиця 3.3 – Зведені розрахунки витрат на оплату праці

№ з/п	Категорія працівників	Основна заробітна плата, грн.			Дод. зар. плата, грн.	Нарах. на $Ф_{ОП}$, грн.	Всього витрати на оплату праці, грн.
		Тарифна ставка, грн.	К-сть відпрацьов. год.	Фактично нарах. з/пл., грн.			
1.	Керівник проєкту	80	60	4800	480	2191,20	7471,20
2.	Інженер	50	160	8000	800	3652	12452
Разом				12800	1280	5843,20	19923,20

Затрати на електроенергію одиниці обладнання визначаються за формулою:

$$З_е=W \cdot T \cdot S,$$

де W – необхідна потужність, кВт;

T – кількість годин роботи обладнання;

S – вартість кіловат-години електроенергії.

З 01 березня 2023 року набувають чинності нові тарифи на електроенергію для населення, затверджені Постановою Національної комісії, яка здійснює державне регулювання у сферах енергетики та комунальних послуг від 26.02.2015 року, №220 (зареєстровано в Міністерстві юстиції України 02.03.2015 року за №235/26680).

Згідно з цими новими тарифами, з 01 березня 2023 року населенню (у тому числі тим, хто проживає в житлових будинках і користується кухонними електроплитами), за обсяг, що перевищує 100 кВт·год електроенергії на місяць – 264 копійок за 1 кВт·год.

Якщо потужність комп'ютера становить 300 Вт з підключеним маршрутизатором і працює протягом 220 годин, то:

$$Зе=0,3 \cdot 220 \cdot 2,64 = 174,24 \text{ грн.}$$

Результати розрахунку затрат на матеріали зводяться в табл. 3.4.

Таблиця 3.4 – Визначення величини затрат на матеріали

Найменування матеріальних ресурсів	Норма витрат	Ціна за одиницю грн	Затрати матеріалів грн	Транспортно заготівельні витрати, грн	Загальна сума витрат на матеріали, грн
Папір А4-80	1	100	100	-	100
Ватман	8	10	80	-	80
Коректор	1	20	20	-	20
Друк паперу	120	1	120	-	120
Друк ватману	8	40	320	-	320
Олівець	1	10	10	-	10
Ручка кулькова (чорна)	2	20	40	-	40
Файли	10	1	10	-	10
Разом					700

Використання основних фондів у виробництві відрізняється особливістю – вони потребують відновлення. Це відновлення засобів праці у фізичному вираженні передбачає їх компенсацію у грошовій формі через амортизацію.

Амортизація – це процес перенесення вартості основних фондів на вартість виробленої продукції з метою повного відновлення цих фондів.

Комп'ютери та оргтехніка відносяться до четвертої групи основних фондів.

Для цієї групи річна норма амортизації становить 60% (квартальна – 15%). Для розрахунку амортизаційних відрахувань використовується формула:

$$А = Б_В \cdot Н_А / 100,$$

де А – амортизаційні відрахування за звітний період, грн.

$Б_В$ – балансова вартість комп'ютера, на початок звітного періоду, грн.

$Н_А$ – норма амортизації, %.

$$А = 20000 \cdot 15\% / 100 = 3000 \text{ грн.}$$

Витрати, пов'язані з утриманням виробництва та забезпеченням ефективної роботи керівництва підприємства (компанії) та створенням відповідних умов праці, відносяться до накладних витрат. Розмір накладних витрат може складати 20% від суми основної та додаткової заробітної плати працівників і розраховується за формулою:

$$Н_В = В_{О.П} \cdot 0.2,$$

$$Н_в = 19923,2 \cdot 0,2 = 3984,64 \text{ грн.}$$

Ефективність виробництва – це узагальнене і повне відображення кінцевих результатів використання робочої сили, засобів та предметів праці на підприємстві за певний проміжок часу. Економічна ефективність (Ер) полягає у відношенні результату виробництва до затрачених ресурсів, що представлено формулою:

$$Е_Р = П / Св ,$$

де П – прибуток;

$С_В$ – собівартість.

$$Е_Р = 14839,22 / 26469,60 = 0,56.$$

Поряд із економічною ефективністю розраховують термін окупності капітальних вкладень (Тр) згідно формули:

$$ТР = 1 / Е_Р ,$$

$$ТР = 1 / 0,56 = 1,79 \text{ р.}$$

Про доцільність розробки програми можна сказати при врахуванні наступних критеріїв, які подані в табл. 3.5.

Таблиця 3.5 – Техніко-економічні показники процесу проєктування

№ з/п	Показник	Значення
1.	Собівартість, грн.	26498,60
2.	Плановий прибуток, грн.	14839,22
3.	Ціна, грн	41337,82
4.	Економічна ефективність	0,56
5.	Термін окупності, рік	1,79

Результати розрахунків вказують на те, що розробка матиме оптимальну економічну ефективність на рівні 0,56 і повернеться майже за два роки (1,79 року). Важливо зауважити, що ці розрахунки мають орієнтовний характер і призначені для оцінки приблизної вартості дослідження та розробки даного продукту.

Висновки до розділу 3

У третьому розділі розроблено модель дерева атак для системи розумного будинку, яка дозволяє оцінювати кібербезпеку за статистичними показниками. Проаналізовано чотирирівневу архітектуру цієї системи, де компоненти на всіх рівнях впливають на її загальну надійність. Відмова на рівні зв'язку та обробки має прямий вплив на готовність системи, оскільки це може призвести до ізоляції виконавчих приладів на нижньому рівні через неможливість передачі команд адміністрування.

Компоненти управління та обладнання також впливають на готовність системи, але окремі відмови їх компонент не спричиняють загальної відмови системи.

Аналіз дерева атак показав, які вимоги потрібно підвищити для компонентів системи, щоб забезпечити її кібербезпеку та надійність. Розраховано ймовірність відмови системи розумного будинку внаслідок атак на її компоненти і виявлено, що розроблена система вважається безпечною порівняно з системами високої готовності, маючи показник надійності на рівні 0,999.

ВИСНОВКИ

У роботі була поставлена і вирішена актуальна наукова задача розробки та дослідження моделі кібербезпеки системи розумного будинку на основі параметрів контролю доступу у формі дерева атак.

У ході виконання кваліфікаційної роботи було проведено дослідження стану розвитку галузі в сфері автоматизації будівель, та перспективи їх розвитку в сфері IoT, розглянуто особливості рівнів архітектури та протоколів Інтернету речей, як компонентів розумного будинку. На основі роботи можна сформулювати наступні висновки.

1. Проведено аналіз ключових показників якості системи розумного будинку. Розумний будинок представляє собою складну систему високотехнологічних пристроїв, спрямованих на забезпечення певного рівня комфорту. Також проведений аналіз ринку систем розумного будинку. Цей ринок є дуже перспективним для технологічних компаній, оскільки кожного року збільшується попит на такі системи та їх сервіси. Розглянуто основні функції контролера системи розумного будинку. Це програмне забезпечення дозволяє керувати кліматом, освітленням, водопостачанням та надає можливість віддалено контролювати події в будинку.

2. Розглянуто особливості моделей системи контролю доступу для розумних будинків. Проаналізовано формальну та мандатні моделі. З трьох моделей (рольової, мандатної моделі Белла-Лападули, мандатної моделі Біба) для розробленої моделі контролю доступу було обрано модель цілісності Біба. Підхід до рішення завдання забезпечення захисту від скомпрометованих пристроїв у моделі цілісності Біба полягає у накладанні обмежень на окремі пристрої для запобігання виконанню операцій, що можуть впливати на безпеку системи. Це вказує на можливість використання рівнів цілісності пристроїв для реалізації таких заходів.

3. Застосування контексту відкриває нові можливості у сфері контролю доступу. Розглянувши опис моделі контролю доступу, можна виділити декілька

переваг контекстних моделей:

– гнучке налаштування політики контролю доступу;

– адаптація системи до оточуючого середовища.

Ці властивості моделі можуть ідеально використовуватися у системах розумного будинку, оскільки вони чудово відповідають особливостям цих систем серед інших засобів автоматизації.

4. Розроблено модель дерева атак для системи розумного будинку, яка дозволяє оцінювати кібербезпеку за статистичними показниками. Проаналізовано чотирирівневу архітектуру цієї системи, де компоненти на всіх рівнях впливають на її загальну надійність. Відмова на рівні зв'язку та обробки має прямий вплив на готовність системи, оскільки це може призвести до ізоляції виконавчих приладів на нижньому рівні через неможливість передачі команд адміністрування.

5. Компоненти управління та обладнання також впливають на готовність системи, але окремі відмови їх компонент не спричиняють загальної відмови системи.

6. Аналіз дерева атак показав, які вимоги потрібно підвищити для компонентів системи, щоб забезпечити її кібербезпеку та надійність. Розраховано ймовірність відмови системи розумного будинку внаслідок атак на її компоненти і виявлено, що розроблена система вважається безпечною порівняно з системами високої готовності, маючи показник надійності на рівні 0,999.

Таким чином, поставлені задачі розв'язано у повному обсязі. Напрямком подальших досліджень є покращення показників кібербезпеки системи розумного будинку за рахунок розробки та дослідження марковської моделі системи.