МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ФАХОВИЙ БІЗНЕС-КОЛЕДЖ
Циклова комісія (кафедра) комп’ютерної інженерії та інформаційних технологій

КВАЛІФІКАЦІЙНА РОБОТА

на тему

ІНТЕРНЕТ РЕЧЕЙ (IOT) ТА ЙОГО ВПЛИВ НА АРХІТЕКТУРУ КОМП'ЮТЕРНИХ МЕРЕЖ

Виконав: студент групи 2К-21
спеціальності 123 комп’ютерна інженерія
Володимир ТИМОШКО

Керівник:
Павло РАТАЙЧУК

Черкаси 2025

ЗМІСТ

ВСТУП

Актуальність обраної теми. Розвиток інформаційних технологій привів нас до різноманітних можливостей з автоматизації побутових, промислових, інфраструктурних потреб. За станом 2025 року в наявності присутня велика кількість пристроїв, які є результатом цього розвитку: роботи-пилососи, розумні обігрівачі, розумні лампи, розумні світлодіодні ленти – і всі вони можуть мати зручне керування через мобільні пристрої. Це дозволяє в темпі сучасного життя витрачати меньше часу на рутинну роботу та застосовувати зекономлений час на заняття більш складною роботою, або просто мати більше часу для відпочинку.

Об’єкт дослідження. Об’єктом дослідження є комп’ютерні мережі.

Предмет дослідження. Предметом дослідження є архітектура інтернету речей, вплив інтернету речей на архітектуру комп’ютерних мереж.

Мета дослідження. Метою роботи є дослідження архітектури інтернету речей, його розвиток, значення в сучасному світі та як ця концепція вплинула на архітектуру комп’ютерних мереж.

Завдання дослідження. Для досягнення поставленої мети були визначені наступні завдання: дослідити концепцію та архітектуру інтернету речей, визначити особливості мереж IoT, їхні вимоги до інфраструктури та безпеки, проаналізувати вплив IoT на традиційні мережі (LAN, WAN, Cloud, Edge Computing), дослідити використання сучасних мережевих технологій для підтримки IoT (5G, LPWAN, SDN, IPv6), розглянути проблеми кібербезпеки IoT-систем та методи їх вирішення, виконати моделювання IoT-мережі та оцінити ефективність її роботи.

РОЗДІЛ І
КОНЦЕПЦІЯ ТА АРХІТЕКТУРА ІНТЕРНЕТУ РЕЧЕЙ

1.1 Визначення та принципи функціонування IoT

Інтернет речей (з англ. Internet of Things, IoT) – це концепція мережі передачі даних, що складається із фізичних пристроїв, які мають засоби і технології для взаємодії між собою та/або навколишнім середовищем.[1]

Технологічно інтернет речей базується на певних принципах, сукупність яких дозволяє відрізнити інтернет речей від інших мереж. Далі йде опис цих принципів та їх особливості. [1][2]

Перший принцип – це наявність засобів ідентифікації пристроїв. Один з таких є RFID – спосіб автоматичної ідентифікації об’єктів, в якому за рахунок радіосигналів зчитуються або записуються дані, що зберігаються в радіовідповідачах або RFID-мітках. Кожна така мітка складається з обчислювального приладу та транспондера (власне сама мітка). Більшість таких міток складаються з двух частин: перша – інтегральна мікросхема для зберігання та обробки інформації, модулювання та демодулювання радіочастотного сигналу і деяких певних інших функцій. Друга частина – антена, що приймає та передає сигнал. Є доволі зручним ідентифікатором за рахунок відсутності необхідності в прямій видимості мітки, може зберігати інформацію до 512 000 байт, має можливість перезапису, зчитування може відбуватися на дистанції до 100 метрів, має підвищену стійкість до несприятливих зовнішніх впливів: механічному, температурному, хімічному, впливу вологи. Може служити більше 10 років. Проте, цю мітко можливо підробити і має схильність до перешкод у вигляді електромагнітних полей. Тобто, не буде підходити в умовах необхідності високого степеню захисту (тільки якщо не буде одним із кроком автентифікації), і погано буде працювати в середовищі з великою кількістю електромагнітних шумів. [3] Наступним засобом ідентифікації є QR-код – оптична мітка, яка зчитується та зберігає в собі інформацію стосовно об’єкта, до якого прив’язана. Характеризується високою швидкістю зчитування, а для збереження достатньо

мати лише поверхню, на яку буде накладено код. [4] Ще існують штрихкоди – графічна інформація, що наноситься на поверхню фізичного об’єкта та дозволяє ідентифікувати його за рахунок технічних засобів. Представляє собою послідовність чорних та білих полос або інших геометричних фігур. За особливостями схожа з QR-кодом, лише формат відображення інформації інший. [5] Також, варто згадати Data Matrix – двухмірний матричний штрихкод, що представляє собою чорно-білі елементи або елементи різного степеню яскравості. На даний момент виділяється хіба-що тим, що є вільним стандартом, проте не має жодного безкоштовного документа, який описав би процес кодування. [6] MAC-адреса – унікальний ідентифікатор, що надається кожній одиниці мережевого обладнання. Вшивається виробником, і пропонується, що в разі появи нового пристрою в мережі, MAC-адреса не буде мати потребу в налаштуванні. Проте, цей ідентифікатор може програмно бути змінений. Деякі веб-інтерфейси дозволяють вільно власноруч змінювати його. [7] IP-адреса – унікальний числовий ідентифікатор приладу комп’ютерної мережі, що працює по протоколу Internet Protocol (IP). Виділяються версії протоколу: IPv4 й IPv6. IPv4 представляє собою 32-бітне число, адрес якого записується у вигляді чотирьох десятичних чисел значенням від 0 до 255, розділеними точками. Є більш простим в структурі та налаштуванні, ніж IPv6. Про IPv6 буде окремо розповідатись далі в роботі. [8]

Другий принцип – наявність засобів вимірювання інформації. Зазвичай, це різні датчики: температурні, вологості, освітлення, тиску, газові, освітлення тощо. Представляють особливу роль в інтернеті речей, адже забезпечують надання даних про середовище, тим самим надаючи інформацію для обробки обчислювальні пристрої мережі. [1]

Наявність засобів передачі даних – під цим мається на увазі застосування конкретних протоколів передачі даних (LPWAN, Wi-Fi, 5G тощо). Конкретно для інтернету речей від засобів передачі даних вимагається енергоефективність, відмовостійкість, ефективність роботи в умовах низької пропускної здатності (до 100 кбіт/с). [1][2]

Варто зауважити важливість всіх перелічених принципів. Відсутність лише другого принципу вже не дасть виділити мережу як мережу IoT, а відсутність першого або третього принципу створить неможливі умови для роботи комп'ютерної мережі загалом.

1.2 Основні компоненти IoT-системи

Для функціонування IoT-системи необхідний певний перечень пристроїв, сукупність та з'єднання яких дозволить назвати мережу інтернетом речей. Це є фізичні пристрої – будь-які об'єкти, що мають компоненти для збору даних або впливу на зовнішнє середовище, та можуть бути підключені в мережу. Також, мережеві технології – засоби зв'язку, які дозволяють з'єднувати фізичні пристрої в одну мережу для взаємодії між ними. Окрім цього, засоби збереження інформації – мережеві сховища (локальний сервер, хмара) або застосування кордонних обчислень (обробка даних відбувається безпосередньо на пристроях або шлюзах, без необхідності передавати дані далі) [9]. Далі йдуть засоби обробки інформації та управління пристроями – мікроконтролери та шлюзи. Іншими словами, обчислювальна частина мережі. Обов'язково програмне забезпечення – програми, які обробляють отримані дані, та подають команди. Саме від програмного налаштування буде визначатись поведінка певних пристроїв мережі при настанні необхідних обставин. І останнє: інтерфейс користувача – засіб керування інтернетом речей кінцевим користувачем (мобільні застосунки, веб-інтерфейси, API для зовнішніх систем). Якби керування мережею було б складним, навряд-чи ця технологія стала би такою популярною. Зручний інтерфейс користувача визначає як легко буде керувати системою, змінювати її налаштування, додавати нові пристрої або навпаки видаляти вже під'єднані.

Якщо описати вище сказане більш коротко, то основними компонентами IoT-системи є власне фізичні пристрої, засоби мережевого зв'язку між фізичними пристроями, сховища збереження даних, засоби обробки та управління мережею, програмне забезпечення, інтерфейс для керування

мережею користувачем. Саме наявність всіх цих компонентів відрізняє звичайну мережу від IoT-мережі. Не можна розглядати мережу як IoT-мережу, якщо буде відсутній хоча б один елемент перелічений раніше. В інакшому випадку це буде інша комп'ютерна мережа. [2]

1.3 Класифікація IoT-пристроїв

Існує різна класифікація IoT-пристроїв, в залежності від того, в якому контексті вони розглядаються. Якщо мова йде про сферу застосування, то виділяються побутові (розумний дім, догляд за людьми літнього віку), пристрої для організацій (в медицині, транспорті), промислові (для виробництва, сільського господарства, продовольства), інфраструктурні (моніторинг навколишнього середовища, керування енергоспоживанням), військові тощо. [1]

За функціональністю виділяють сенсори, актуатори, керуючі пристрої (контролери).

За способом з'єднання дротові та бездротові. [2]

Наприклад, механізм автоматичного відчинення та закривання дверей гаража є, в залежності від масштабів, побутовим пристроєм або пристроєм для організацій, актуатором, що, зазвичай, підключається до мережі дротовим способом з'єднання (керуючись власним досвідом), але може включати й можливість бездротового підключення.

1.4 Протоколи та стандарти зв'язку в IoT (MQTT, CoAP, HTTP/HTTPS, LoRaWAN)

MQTT (англ. Message Queue Telemetry Transport – «Транспортування Телеметрії Черги Повідомлень») – спрощений мережевий протокол, що працює на прикладному рівні стеку протоколів TCP/IP. Використовується для обміну повідомлення між пристроями за принципом видавець-підписник (принцип, при якому відправники повідомлень напряму не прив'язані до кінцевих отримувачів, а інформація з повідомлень ділиться на класи та не має інформації про своїх отримувачів. Інформація до кінцевих отримувачів доходить після фільтрації

повідомлень [10]). Серед можливостей, в порівнянні з іншими протоколами, є більш простий у використанні, адже являє собою програмний блок без зайвої функціональності, що може буде вбудовано в будь-яку систему. Зручний для більшості рішень з датчиками. Легкий у адмініструванні. Не має високого навантаження на канал зв'язку. Добре показує себе в роботі при умовах постійної втрати зв'язку або інших проблем на лінії передачі даних. Може передавати дані в будь-якому форматі. [11]

CoAP (англ. Constrained Application Protocol – «Протокол Обмеженого Застосування») – спеціалізований мережевий протокол прикладного рівня, призначений для використання в інтернет-пристроях з обмеженими ресурсами (низькопотужними). Розроблений для легкого перетворення даних під протокол HTTP для спрощеної інтеграції з Інтернетом, має дуже низькі накладні витрати та виділяється простотою. В IoT-системах застосовується в малопотужних пристроях, які потребують простого та енергоефективного протоколу для передачі даних. Також, наявна багатоадресна розсилка. Цей протокол може працювати на більшості пристроїв, що підтримують UDP або аналог UDP.[12]

HTTP/HTTPS (англ. Hypertext Transfer Protocol (Secure) – «Протокол Передачі Гіпертексту (Захищений)») – мережевий протокол прикладного рівня, призначений для передачі веб-сторінок (HTML-файлів, зображень, застосунків), проте успішно можна передавати й інші файли. Використовується з 1990 року і до сих пір є актуальним. Аналогічними цьому протоколу є FTP т SMTP. Обмін повідомленнями відбувається за звичайною схемою «запрос-відповідь». На відміну від більшості інших протоколів, не зберігає інформацію про свій стан. Особливістю протоколу являється можливість вказати в запиті та відповіді спосіб надання того ж самого ресурсу з різними параметрами: формату, кодуванні, мові і так далі. При доступі даних по FTP або через інші файлові протоколи тип файла визначається по розширенню типу файла, що не завжди зручно, коли HTTP перед тим, як передати самі дані, однозначно визначає яким способом необхідно обробляти отримані дані. [13]

LoRaWAN (англ. Long Range Wide Area Network – «Широкомасштабна Мережа Далекого Діапазону») – бездротовий мережевий протокол, який використовується для побудови IoT-мереж на великій відстані. Цей протокол має низьку потужність, низьку швидкість передачі даних (0,3 кбіт/с – 50 кбіт/с). Може передавати інформацію між пристроями в умовах міста до 4 км. Разом з цим протоколом також визначають LPWA, призначений для бездротового підключення пристроїв, що працюють від акумулятора, до інтернету в регіональних, національних або глобальних мережах і націлений на ключові вимоги інтернету речей, як двонаправлений зв'язок, наскрізна безпека, послуги мобільності та геолокації. Низька потужність, низька швидкість передачі даних і використання IoT вирізняють цей тип мережі від бездротової глобальної мережі, що призначена для підключення користувачів або підприємств та передачі великої кількості даних із великим споживанням енергії. Відомим відкритим аналогом цього протоколу є DASH7. [14]

1.5 Використання хмарних та периферійних обчислень у IoT

Хмарні обчислення – це форма зберігання та обробки даних, при якому всі дані зберігаються та обробляються на віддаленому сервері, до якого, зазвичай, підключаються через інтернет. Дозволяє ефективно перенести обробку з периферії на централізовані сервери, де дані будуть швидко оброблені. Ключовим аспектом хмарних обчислень є можливість віддаленого доступу до даних і ресурсів. Це особливо важливо, якщо інтернетом речей користуються більше, ніж 1 користувач. [15]

Периферійні обчислення – це форма зберігання та обробки даних якомога ближче до джерела їхнього створення або кінцевого користувача. Може поєднуватись із хмарними обчислення для зменшення даних, що необхідно передати хмарі. Застосовуються у невеликих IoT-мережах, де немає необхідності у обробці великих обсягів даних. Також, застосовується в сферах, де питання швидкості обробки інформації є критичним: в медицині, у безпілотних автомобілях і т.д. [16]

Далі представлена порівняльна таблиця 1.5.1 для повноцінного порівнянна хмарних та периферійний обчислень.

Таблиця 1.5.1 – порівняння хмарних та периферійних обчислень

Тип обчислення	Місце, де відбуваються обчислення	Швидкість обробки	Інші особливості
Хмарне	На віддаленому сервері	Повільніша	Ефективне для великих об’ємів даних
Периферійне	Як можна ближче між пов’язаними пристроями	Найшвидша	

РОЗДІЛ ІІ
ВПЛИВ IoT НА АРХІТЕКТУРУ КОМП'ЮТЕРНИХ МЕРЕЖ ТА БЕЗПЕКА IoT

Поява будь-якої нової технології в комп'ютерних мережах може мати значний вплив на всю її інфраструктуру. В цьому розділі буде розглянуто як саме IoT вплинув на комп'ютерні мережі, що нового було внесено, та як постає питання безпеки в мережах IoT.

2.1 Традиційні архітектури комп'ютерних мереж та їх особливості

В комп'ютерних мережах виділяються 2 основних архітектури будування мережевої архітектури: клієнт-серверна та peer-to-peer. В цьому підрозділі будуть розглянуті їх будови та особливості.

Архітектура клієнт-сервер – домінуюча концепція у будуванні розподілених мережних застосунків, що передбачає взаємодію та обмін даними між ними. Складається ця архітектура з наступних компонентів: сервери, клієнти та мережа, яка забезпечує взаємодію між клієнтами та серверами.

Кожний сервер є незалежним один від одного. Клієнти також працюють паралельно і незалежно один від одного. Жорстка прив'язка між клієнтами та серверами відсутня. Сервер здатний одночасно обробляти запити відразу кількох клієнтів, коли клієнти здатні звертатися до різних серверів паралельно. Клієнти можуть знати про доступність до серверу, але не можуть знати про існування інших клієнтів.

Клієнтами можуть бути комп'ютери, мобільні пристрої або інші користувацькі прилади. Сервери зберігають дані, обробляють запити та можуть виконувати інші функції.

Перевагою такою архітектури є централізоване управління, що спрощує адміністрування та контроль за даними. Недоліком є залежність клієнтів від серверів: якщо сервер виходить з ладу, клієнти не можуть отримати доступ до його ресурсів. [17]

Архітектура peer-to-peer (P2P) – це архітектура системи, в основі якої стоїть мережа рівноправних вузлів. Засновується вона на принципу рівноправності учасників між собою. В цій мережі не існує понять клієнтів або серверів – лише рівні вузли, які функціонують одночасно як клієнти та сервери відносно інших вузлів мережі.

Застосовується в середовищах, де необхідна дистрибуція даних без великої централізованої інфраструктури. Перевагою такої архітектури є стійкість, адже вихід з ладу одного або декількох вузлів не зупиняє роботу всієї мережі. Недоліком є складність управління та забезпечення безпеки, адже всі пристрої мають рівний доступ до ресурсів. [18]

2.2 Зміни у мережевій інфраструктурі під впливом IoT

З появою IoT кожен пристрій може стати точкою з'єднання з мережею, що змінює вимоги до передачі даних, енергоспоживання, безпеки та управління. Ось основні зміни, які відбуваються в мережевій інфраструктурі під впливом IoT:

- Масштабованість мережі – IoT вимагає підтримки величезної кількості пристроїв, які мають бути підключеними до мережі (IPv6 стає необхідним для вирішення проблеми нестачі адрес);
- Мережеві топології та типи з'єднань – Поява нових мережевих топологій, зокрема меш-мереж (mesh networks);
- Обробка та зберігання даних – Хмарні та периферійні обчислення;
- Енергоспоживання та ефективність мережі – Використання LPWAN;
- Нова модель управління мережею (SDN) – дозволяє централізовано керувати мережею за допомогою програмного забезпечення;
- Безпека в мережі – шифрування даних, автентифікація пристроїв, а також автоматизовані системи виявлення загроз.

2.3 Використання IPv6 для підтримки IoT

Один з основних недоліків IPv4 — обмежена кількість адрес. IPv6 забезпечує 128-бітові адреси, що дозволяє створити більше ніж 340 унцільйонів

унікальних адрес. Це робить IPv6 ідеальним для IoT, оскільки кожен пристрій може мати свою унікальну IP-адресу.

Серед інших переваг IPv6 виділяють: більш ефективна маршрутизація (без необхідності застосування NAT); вбудований протокол безпеки IPsec; IPv6 має багатоадресну передача, яка забезпечує одночасну доставку даних кільком одержувачам, при цьому не збільшуючи вимоги до пропускної спроможності мережі; в IPv6 використовується технологія SLAAC (автоконфігурація адрес без збереження стану), яка дозволяє пристрою самостійно налаштовувати для себе адресу без додаткових посередників і спеціальних протоколів. [2]

2.4 Роль 5G та LPWAN у розвитку IoT-мереж

5G (з англ. fifth generation – «п’яте покоління») – це п'яте покоління мобільних мереж, яке обіцяє значно вищу швидкість передачі даних, знижену затримку та підтримку величезної кількості пристроїв на одиницю площі. Ці характеристики роблять 5G критично важливим для розвитку IoT, де швидкість, ефективність та надійність з'єднань є вирішальними. [19]

LPWAN (з англ. Low-power Wide-area Network – «енергоефективна мережа ширикого радіусу дії») – це категорія низькошвидкісних, широкодіапазонних мереж, що підтримують великі відстані передачі даних за низьким енергоспоживанням. Ідеально підходять для IoT-пристроїв, які потребують рідкісного обміну даними на великій відстані, і не потребують високої пропускної здатності або миттєвого зворотного зв'язку. Має широкий радіус покриття та низьку вартість використання. [20]

Далі представлена порівняльна таблиця 2.4.1 між 5G та LPWAN.

Таблиця 2.4.1 – порівняльна характеристика технологій 5G та LPWAN

Назва технології	Пропускна здатність	Радіус покриття	Інші особливості
5G	до 10 гбіт/с	15м – 7км	
LPWAN	100 біт/с – 30 кбіт/с	10-15 км	енергоефективне

2.5 Використання SDN (Software-Defined Networking) у IoT

SDN (з англ. software-defined networking «програмно-конфігурована мережа») – мережа передачі даних, в якій рівень управління мережею відділений від пристроїв передачі даних і реалізується програмно, одна з форм віртуалізації обчислювальних ресурсів. Дані з таблиць маршрутизації зберігаються як і на апаратних системах, так і централізовано управляються віддаленою системою, що дозволяє системному адміністратору не змінювати таблиці в кожному мережевому пристрою, а робити все на віддаленій системі.

В IoT-мережах дозволяє значно спростити конфігурування налаштувань для IoT-пристроїв, особливо в мережах з великою кількістю подібних пристроїв.

Подібні мережі ефективно працюють при будуванні інфраструктурних хмарних сервісів, коли по запиту від користувачів необхідно створювати максимально швидкі віртуальні вузли та виділяти віртуальні мережеві ресурси для них, ізольовано від інших користувачів.

Застосовується в умовах надзвичайно великої необхідності в швидкості обробки даних: дозволяє скоротити затримку на проводження мережі за рахунок централізованого керування на програмному контролері та збільшить процент використання ресурсів за допомогою динамічного керування. [21]

2.6 Основні загрози безпеці в IoT-системах

DDoS-атака – заражена мережа комп’ютерів безперервно надсилає величезну кількість засобів до системи. Це призводить до перенавантаження ресурсів системи, збоям в роботі та повного виходу системи з ладу. Також, самі IoT-пристрої можуть примкнути до зараженої мережі, якщо не будуть захищені належним чином.

Експлойт програмного забезпечення – використання лазівок в програмному коді або прошивці пристроїв, що надає їм можливість виконання

небажаних дій, як отримання несанкціонованого доступу до IoT-мережі або виконання шкідливого коду.

MITM-атака (англ. man-in-the-middle – атака посередника) – вставши посередині каналу передачі даних між відправником та отримувачем, зловмисник може отримати цінну конфіденційну інформацію або впливати на процеси керування пристроями.

Фізичне втручання – фізичне підключення до пристроїв для введення злоумисного програмного забезпечення. Таким шляхом зловмисник може безпосередньо підключатись до IoT-пристрою та змінити його налаштування або встановити шкідливе програмне забезпечення. Особливо це стосується пристроїв, що розташовані у публічних та незахищених фізичних середовищах.

Брутфорс атака – метод грубої сили перебору паролів, коли відразу багато варіантів пароля передається на аутентифікацію для визначення правильного паролю. При відсутності необхідного захисту (блокування після декількох невдалих спроб) зловмисник може отримати доступ до системи.

Всі перелічені загрози вимагають впровадження комплексних засобів кібербезпеки.[22]

2.7 Захист IoT-пристроїв від атак

Захист IoT-пристроїв вимагає багаторівневного підходу в зв’язку з тим, що ці пристрої часто є вразливими через обмежені апаратні ресурси, недосконале програмне забезпечення, недостатню безпеку за замовчуванням.

Для захисту IoT-пристроїв використовується достатньо велика кількість методів. Одним з таких є управління поверхнею атаки, інвентаризація та моніторинг усіх пристроїв – необхідно мати карту підключених пристроїв для їх інвентаризації, що дозволить знати точну кількість застосованих пристроїв, ідентифікатори виробників, серійні номери, версії встановленого програмного забезпечення. Іншим є сегментація мережі – у разі успішної кібератаки, корректно сегментована мережа не дозволить захопити контроль над всією мережею та обмежить збитки, сама сегментація це поділ внутрішньої мережі на

кілька окремих підмереж, що, зазвичай, незалежні та ізольовані між собою. Ще одним також є встановлення надійних паролів для IoT – багато пристроїв використовують стандартні або самі по собі прості засоби автентифікації, як встановлення простих чи заводських паролей, які легко підібрати, коли рекомендується після підключення нового IoT-пристрою встановити значно складніший пароль, що має бути стійким для підбору та унікальним. Не варто забувати про захист IoT-пристроїв на фізичному рівні – пристрої, які можуть піддаватись фізичним втручанням, мають бути забезпечені надійним місцем дислокації, щоб до нього не було відкритого доступу; своєчасне оновлення програмного забезпечення – оскільки програмне забезпечення все ще по більшій частині розробляється людиною, в програмному забезпеченні пристроїв можуть приховуватись різні вразливості, коли своєчасне оновлення програмного забезпечення має виправити знайдені вразливості та наділити пристрій більшою надійністю.

Правильна реалізація цих засобів у сукупності значно підвищує стійкість IoT-систем до атак та зменшує ризики пов’язані з використанням незахищених або застарілих пристроїв.[22]

2.8 Використання блокчейн-технологій для безпеки IoT, політики та стандарти безпеки IoT

Технологію блокчейн можна використати для надання IoT-пристроям унікальних цифрових ідентифікаторів, які зберігаються в блокчейні. Їх можна застососувати для аутентифікації пристроїв і забезпечення підключення до мережі лише санкціонованим пристроям. Також, ця технологія запобігає фальсифікації даних, адже люба спроба змінити дані буде негайно виявлена та позначена як шахрайська.

В зв’язку з підвищенням попиту на інтернет речей, з’явились також міжнародні та національні політики, стандарти та регламенти, що регламентують вимоги до захисту інформації в IoT-системах. Політики безпеки регламентують правила взаємодії, автентифікації, шифрування та зберігання

даних в IoT-мережах. Ці політики визначають хото та як може взаємодіяти з пристроями, як забезпечити цілісність інформації а також які дії слід застосовувати у разі порушення безпеки. Серед них є NIST IR 8259, ISO/IEC 30141, ISO/IEC 27402, ETSI EN 303 645.

Узгоджене застосування блокчейн-рішень, продуманих політик та дотримання міжнародних стандартів надає можливість значно підвищити рівень кібербезпеки IoT-середовищ та зменшити ризики, пов’язані з кіберзагрозами.

РОЗДІЛ ІІІ
МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЯ IoT-МЕРЕЖІ

3.1 Методологія дослідження та вибір інструментів моделювання

Методологія моделювання включає послідовні етапи. Першим є формування вимог до мережі – визначення IoT-пристроїв, що будуть застосовуватись, умови роботи та обмеження. Наступним буде проєктування мережі – побудова топології мережі із урахуванням параметрів вказаних в завданні. Далі йде конфігурація пристроїв – програмне налаштування всіх пристроїв мережі для працездатної взаємодії між ними. Після цього буде проведений аналіз ефективності побудованої архітектури IoT-мережі. Це робиться для пошуку слабких або потребуючих більшої уваги частин мережі. Після аналізу побудованої мережі буде проведена оптимізація з урахуванням необхідних її редагувань. Для фіксування змін буде проведене повторне тестування мережі та оцінка впроваджених змін і вплив на продуктивність мережі.

В якості інструмента моделювання IoT-мережі була застосована прикладна програма під операційну систему Windows x64: Cisco Packet Tracer – симуляційне середовище комп’ютерних мереж. Вибір програми обумовлений широким функціоналом, зручним та зрозумілим графічним інтерфейсом, підтримкою великої кількості мережевих пристроїв, включаючи IoT-пристрої, що дозволяє створювати детальні симуляції реальних мереж. Cisco Packet Tracer надає можливість створювати послідовну структуру мережі з використанням маршрутизаторів, комутаторів, мостів, датчиків, актуаторів, побутовими приладами тощо. Є можливість встановлювати зв’язки за допомогою різних типів з’єднання (дротове, бездротове). Налаштування параметрів пристроїв, включаючи IP-адресацію, налаштування апаратних складових пристроїв, що стосуються мереж, політика безпеки тощо. Також, в цій програмі реалізована можливість симулювати роботу мережі в реальному часі, що дає змогу

проаналізувати процеси обміну даними, затримки, навантаження на мережу тощо.

3.2 Створення моделі IoT-мережі та її тестування

По-перше, необхідно визначитися з вимогами мережі. Було прийнято рішення створити на базі IoT-мережі систему розумного будинку. Необхідно, щоб була охоронна система: мають бути датчики на вікнах, вхідних дверях та дверях гаража. У разі несанкціонованого вторгнення має вмикатись сирена.

З урахуванням вище вказаних умов, була створена модель необхідної мережі (рисунок 3.2.1).

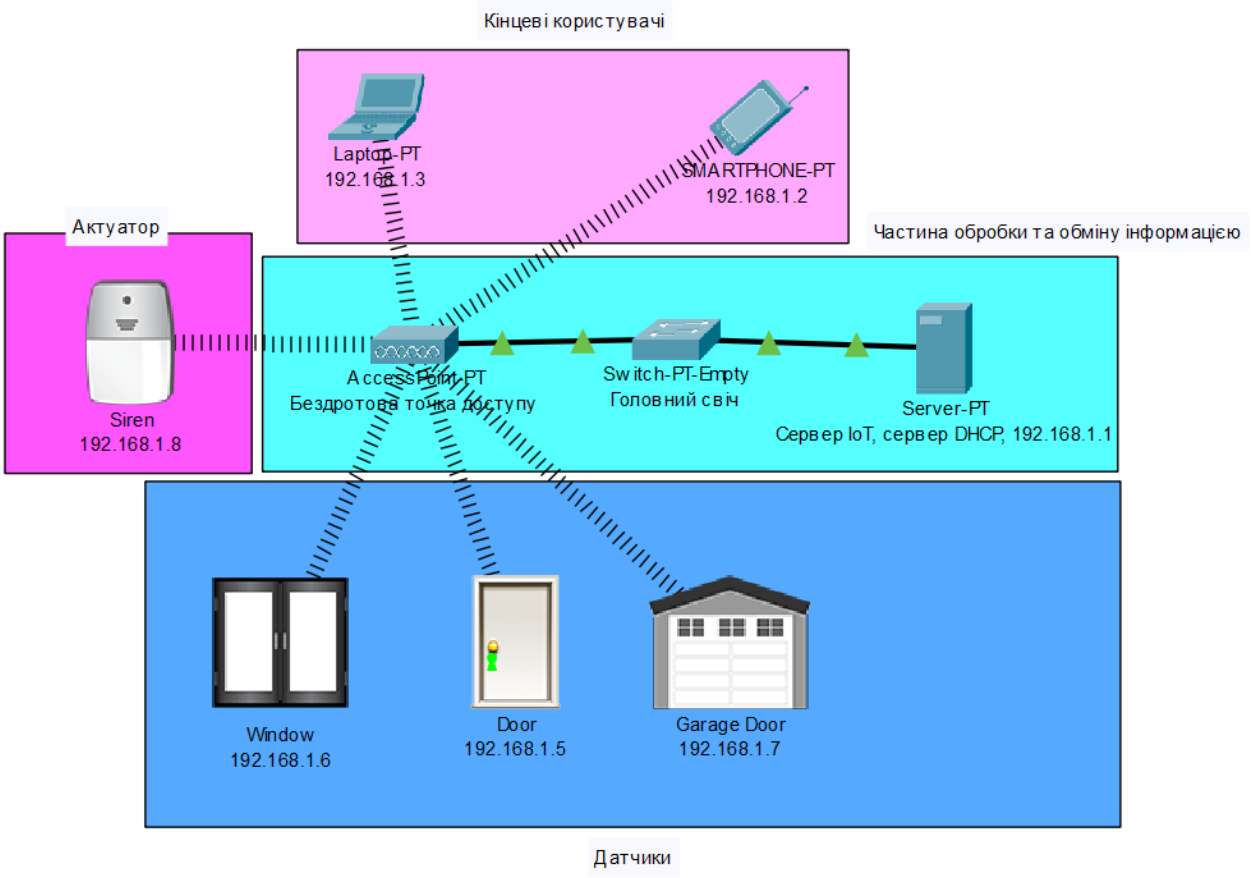

Рисунок 3.2.1 – Початкова модель мережі інтернету речей

В цій мережі є все необхідне для для IoT-мережі. Тут присутні датчики та актуатор, сервер інтернету речей, світч, бездротова точка доступу (дозволяє підключатись пристроям через Wi-Fi), пристрої кінцевого користувача для керування мережею. Апаратно ця модель дотримується всіх описаних раніше

принципів IoT-мережі. Єдине що не видно інтерфейс керування кінцевим користувачем, але він буде представлений далі в ході роботи.

Для простоти налаштування мережі, був також налаштований DHCP сервер для автоматичного надання IP-адреси IoT-пристроям та іншим пристроям мережі. Оскільки IP-адреса сервера це 192.168.1.1, а маска підмережі становить 255.255.255.0, тобто є можливість підключити ще до 254 пристроїв до змодельованої мережі (рисунок 3.2.2, рисунок 3.2.3).

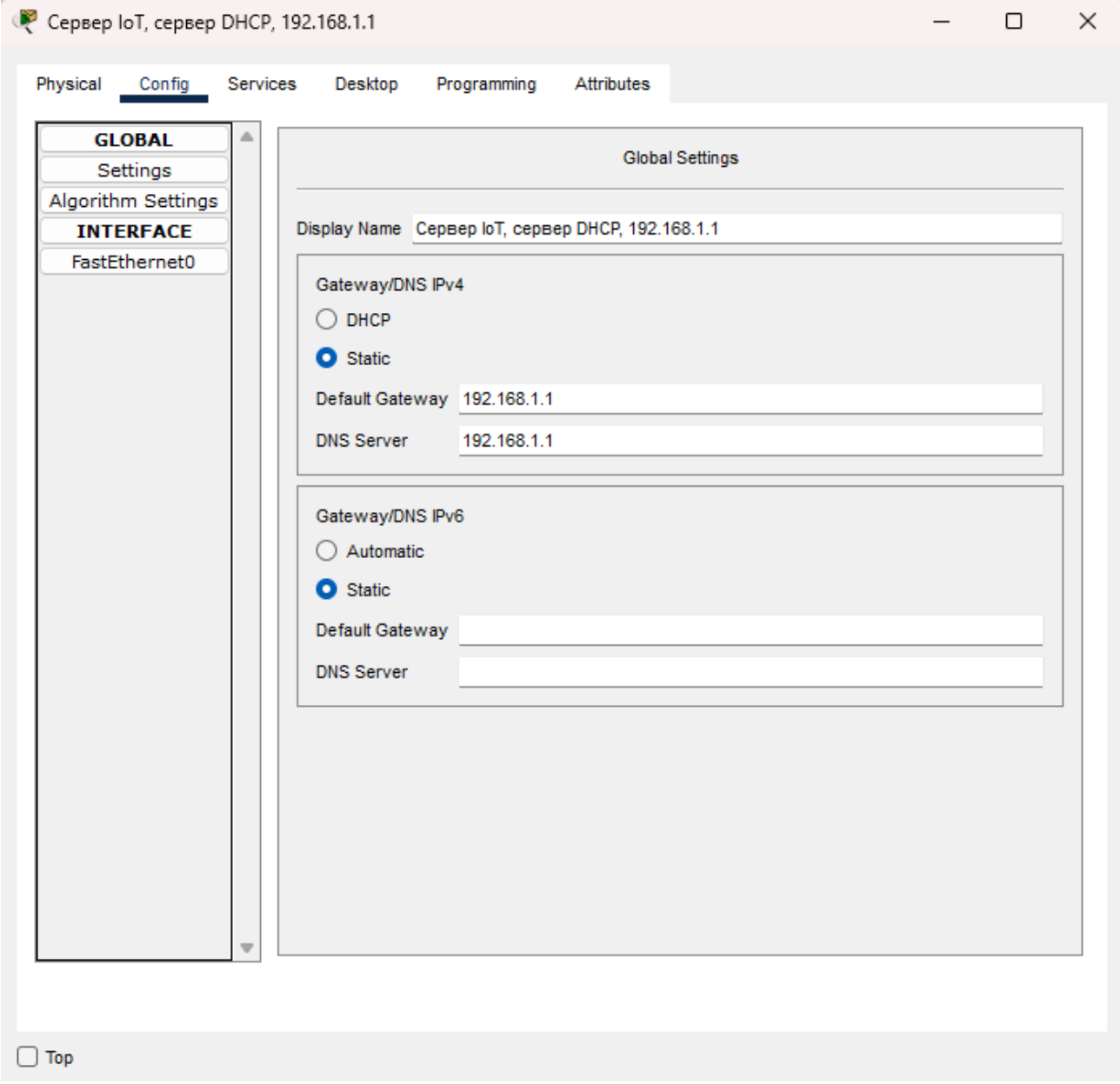

Рисунок 3.2.2 – Меню глобальних налаштувань серверу IoT

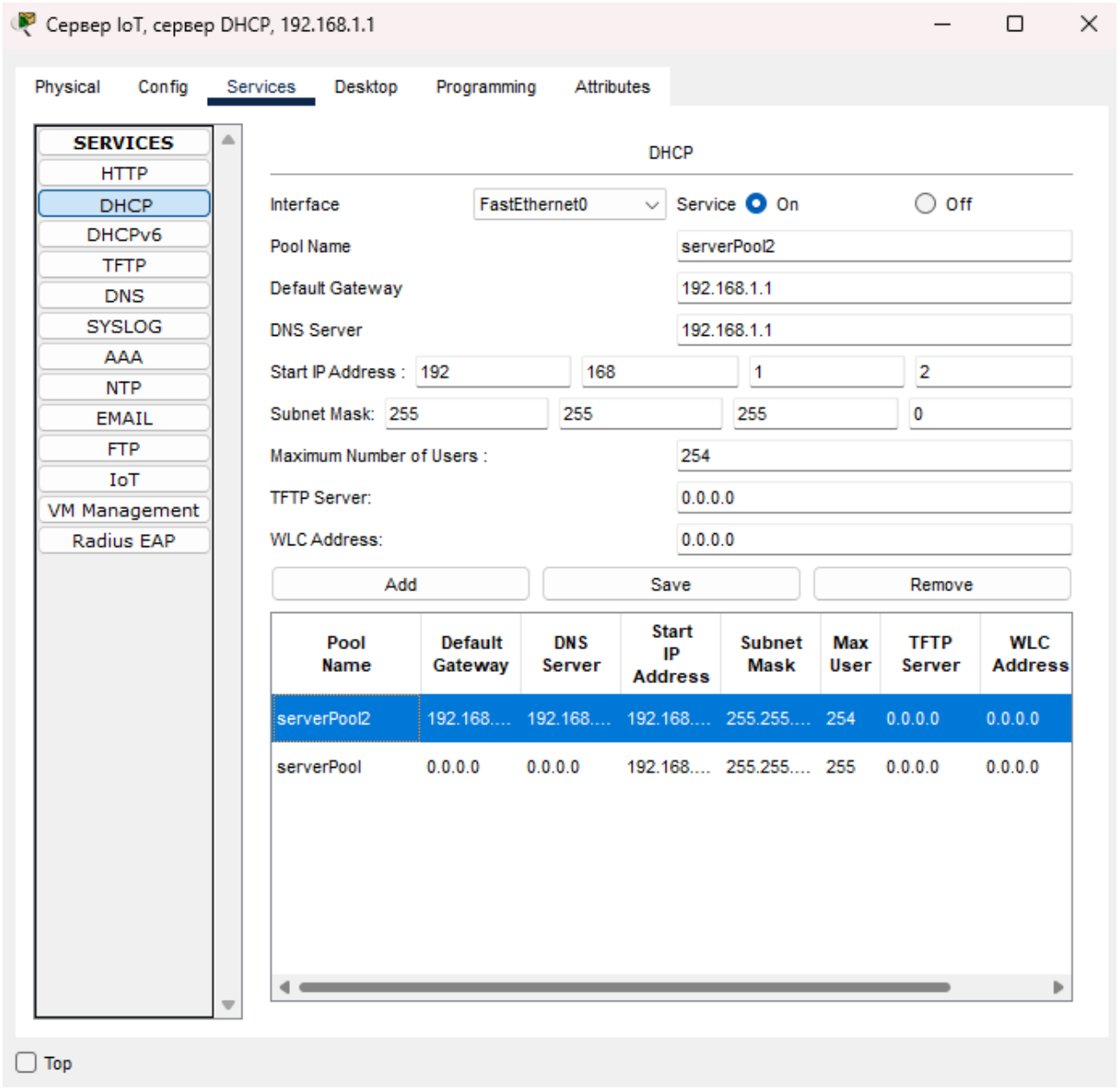

Рисунок 3.2.3 – Налаштування DHCP-серверу

Далі в логічній послідовності йде світч. Він при собі має порти з інтерфейсом Fast Ethernet (рисунок 3.2.4).

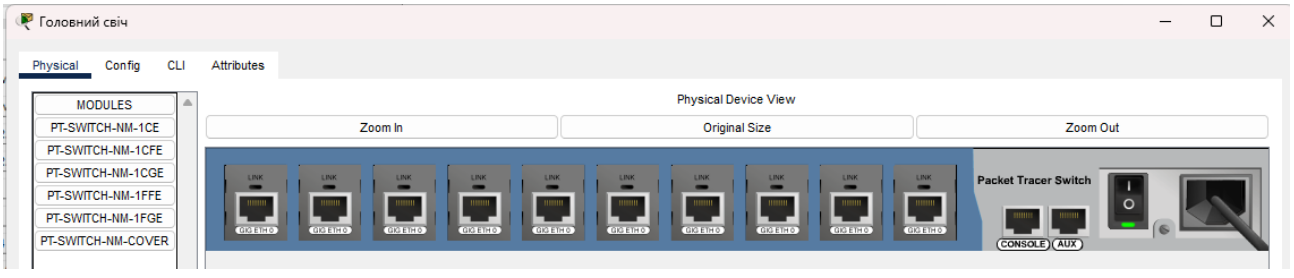

Рисунок 3.2.4 – Відображення портів світча в моделі мережі інтернету речей

Після світча йде бездротова точка доступу. Конкретно в цій моделі він зв’язує всі інші пристрої, що було зроблено для простоти демонстрації. В самій точці доступу можна налаштувати назву мережі, канал передачі, радіус, поставити пароль на мережу (рисунок 3.2.5).

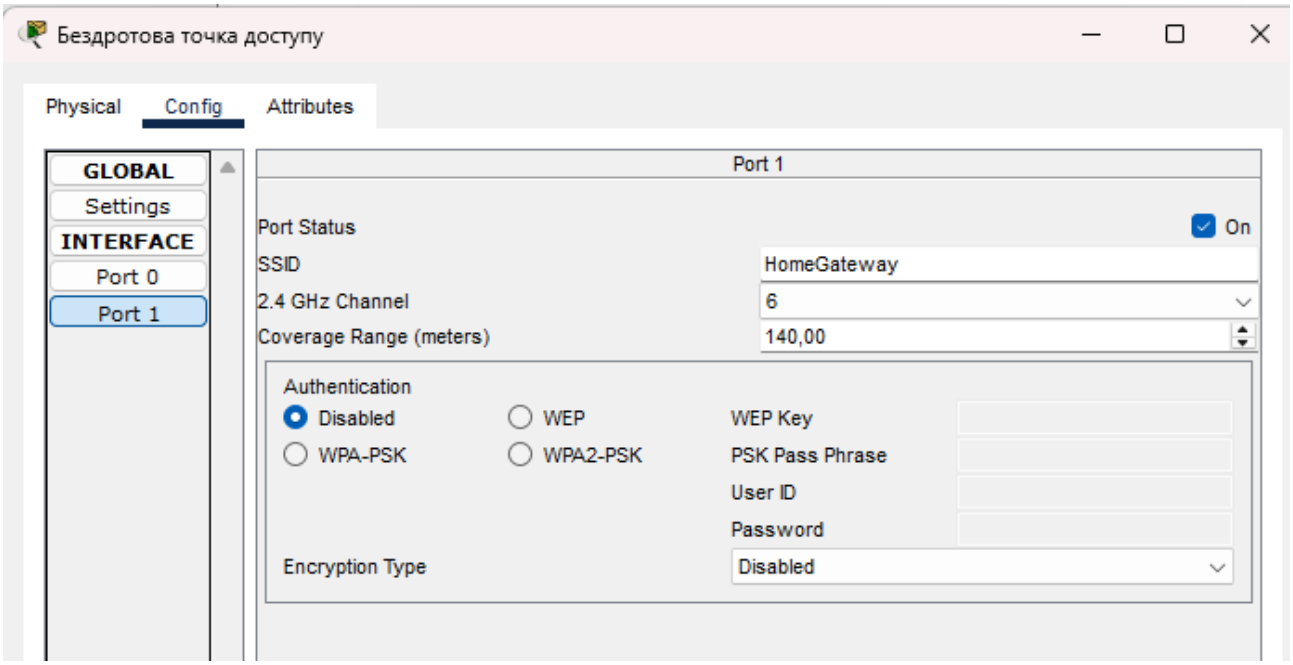

Рисунок 3.2.5 – Відображення налаштувань бездротової точки доступу

Наступними в мережу ввійшли датчики: датчики відкритого вікна, вхідних дверей, гаражних дверей, датчик диму (виділено синім прямокутником на рис. 3.2.1). Отже, для підключення їх до мережі інтернету речей, треба спершу включити цей сервер в налаштуваннях (рисунок 3.2.6). Варто зауважити, що хоча спершу це може здатись непомітним, але запуск мережі інтернету речей також автоматично запускає HTTP-сервер, до якого можна підключитись для програмного функціонального налаштування пристроїв інтернету речей. Тому необхідно за допомогою якогось локального пристрою через HTTP протокол підключитись на адресу серверу (192.168.1.1) (рисунок 3.2.7). Якщо жодного аккаунта не існує, його обов’язково треба зробити – інакше ніяк не вдасться підключити пристрої (рисунок 3.2.8). Як видно на рисунку 3.2.6, один аккаунт вже створений з логіном та паролем admin й admin відповідно.

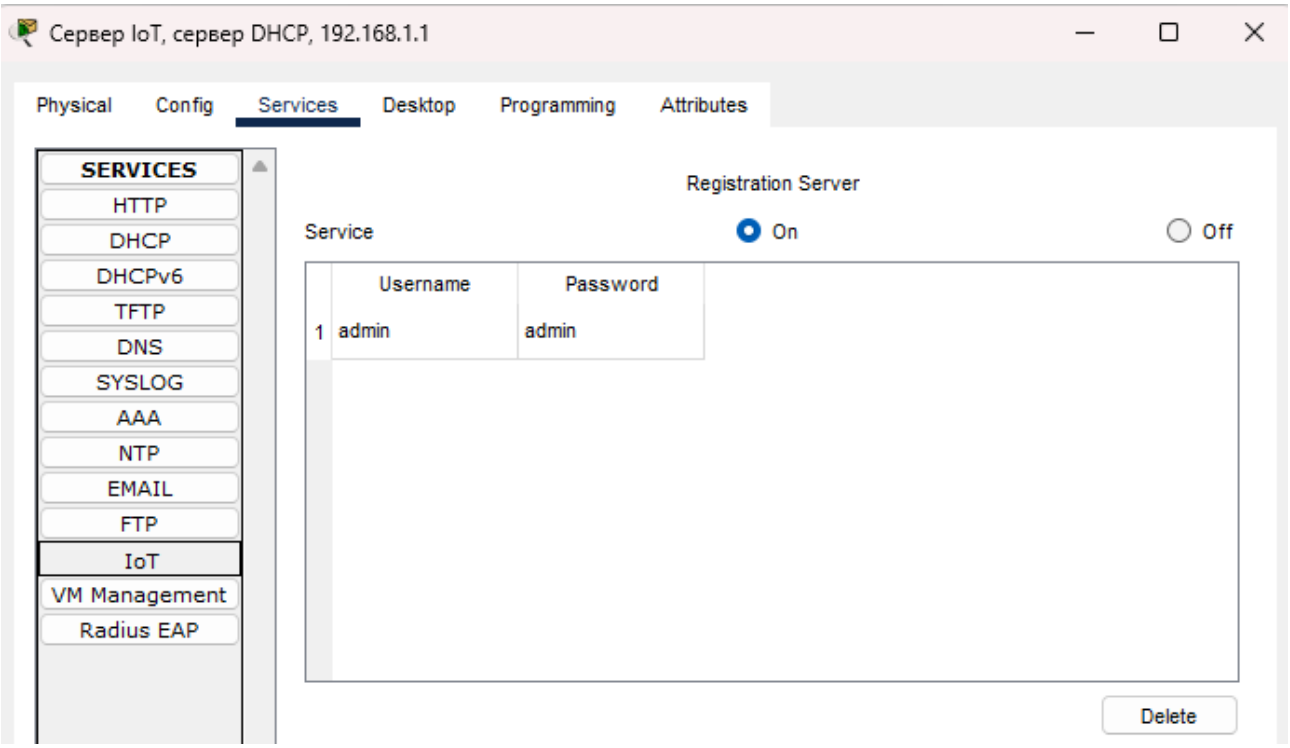

Рис. 3.2.6 – Відображення налаштувань інтернету речей на самому сервері.

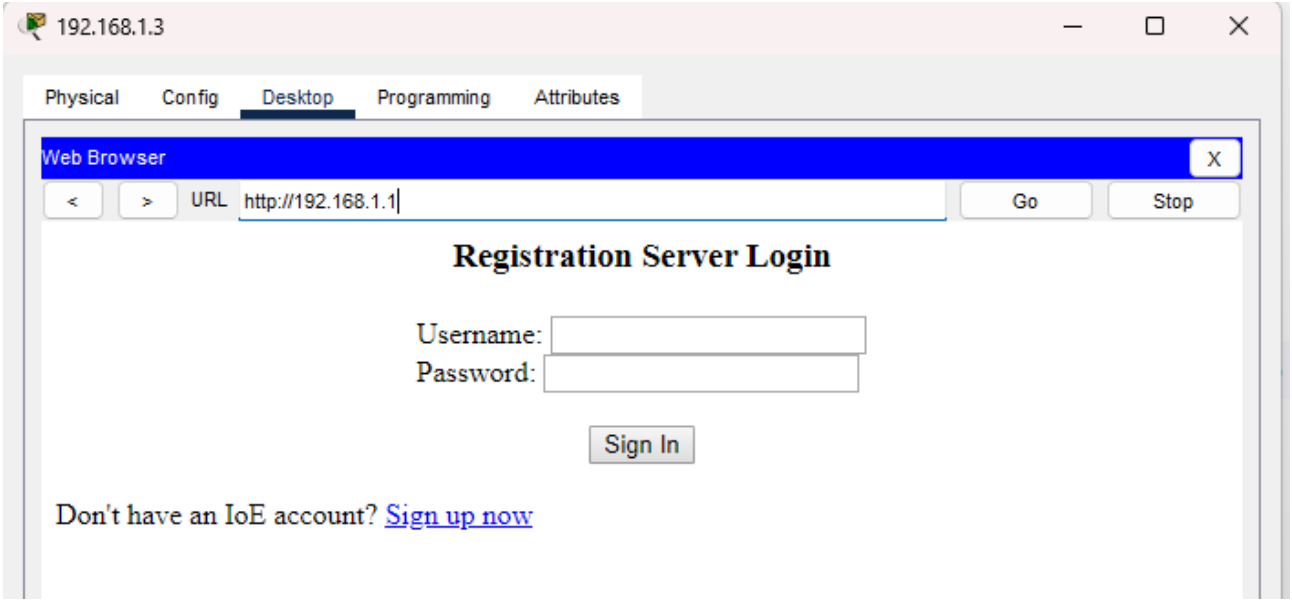

Рисунок 3.2.7 – Відображення зовнішнього вигляду веб-сторінки при підключенні до серверу через протокол HTTP

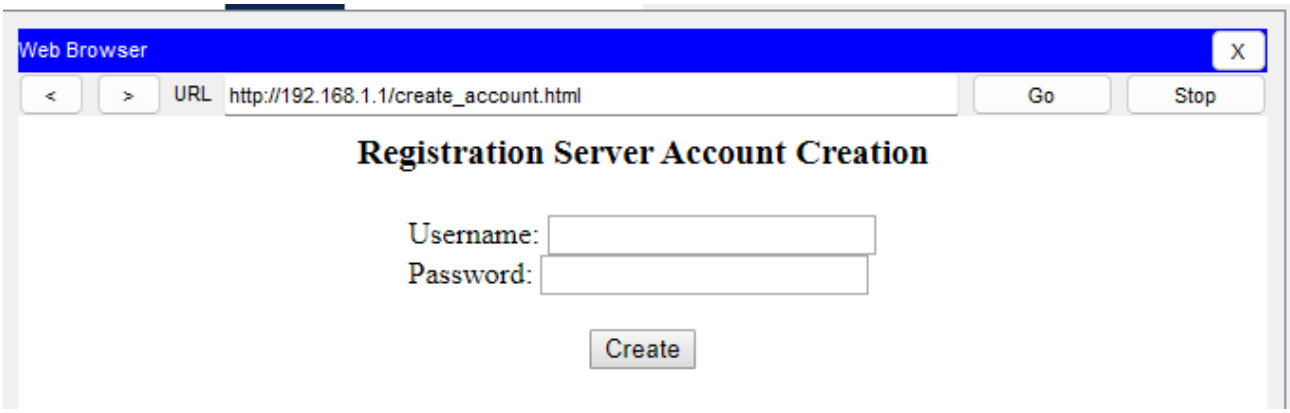

Рисунок 3.2.8 – Меню реєстрації нового аккаунту в мережі IoT

Отже, після створення аккаунту можна нарешті під’єднувати IoT-пристрої. Для їх під’єднання мають співпадати 2 умови: пристрій має бути якимось шляхом під’єднаний до мережі з сервером IoT, пристрою необхідно мати правильні налаштування інтернету речей в глобальних настройках. Конкретно треба вказати адресу серверу, логін та пароль аккаунту. (рисунок 3.2.9)

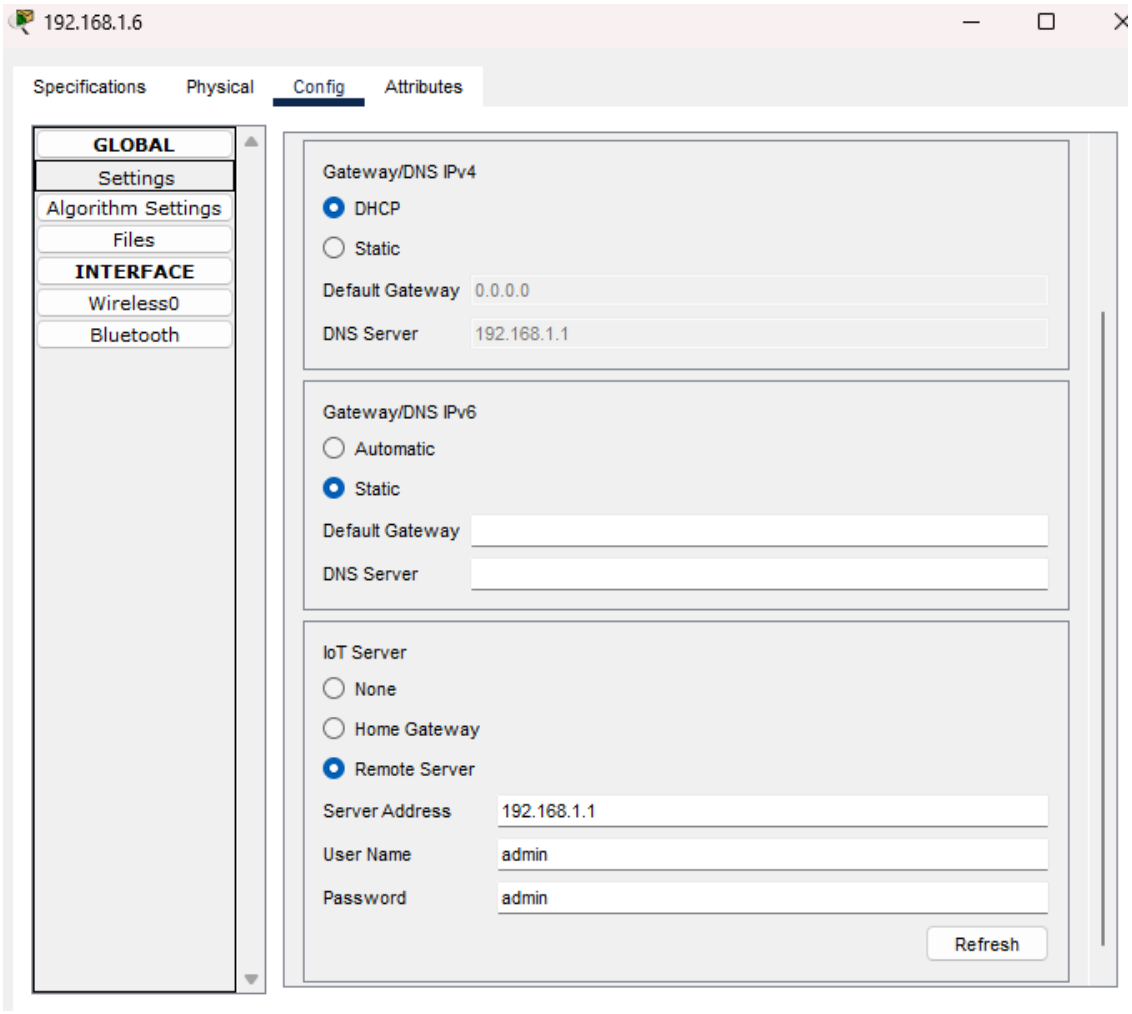

Рисунок 3.2.9 – Меню налаштування інтернету речей на прикладі вікна

Оскільки в нас локальний віддалений сервер, то саме його в налаштуваннях і обираємо, що дає можливість в даних обставинах підключитись пристрою до IoT-мережі. І так має бути скоригований кожний IoT-пристрій мережі.

Залишились лише кінцеві пристрої користувача. Оскільки, як було сказано, HTTP-сервер форсовано запускається у разі запуску IoT-мережі, варто сказати, що робиться це не тільки з ціллю авторизації пристрою. Саме на веб-сторінці й будуть проводитись всі необхідні налаштування поведінки актуаторів при отримання сигналу від датчиків. Значить, щоб налаштувати поведінку IoT-пристрою при отримання даних з датчика, треба ввійти в створений аккаунт, на який було налаштовано IoT-пристрої. При вході в аккаунт на головному вікні буде показано список підключених пристроїв, тип, назва, серійний номер. Для простоти ідентифікації, в якості назви була вказана IP-адреса кожного пристрою (рисунок 3.2.10).

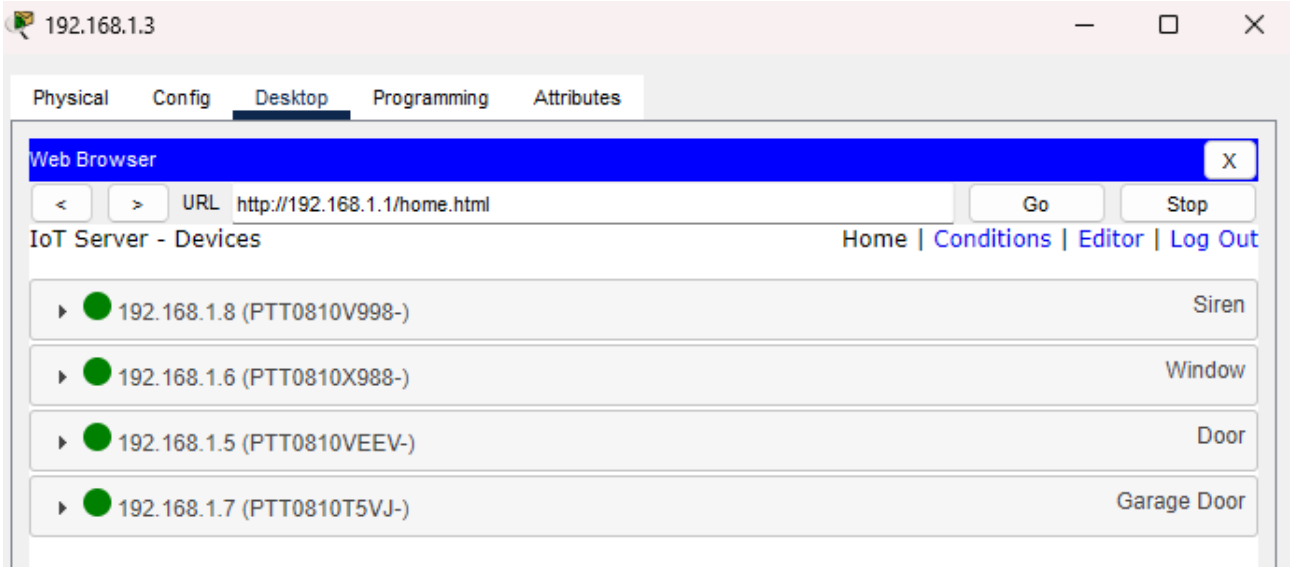

Рисунок 3.2.10 – Інтерфейс головного меню керування IoT-пристроями через пристрій кінцевого користувача

Для того, щоб почати налаштовувати поведінку пристроїв, необхідно перейти на вкладку Conditions (зверху справа на рисунку 3.2.10). В цій вкладці буде список всіх створених алгоритмів дій на сигнал датчика (рисунок 3.2.11).

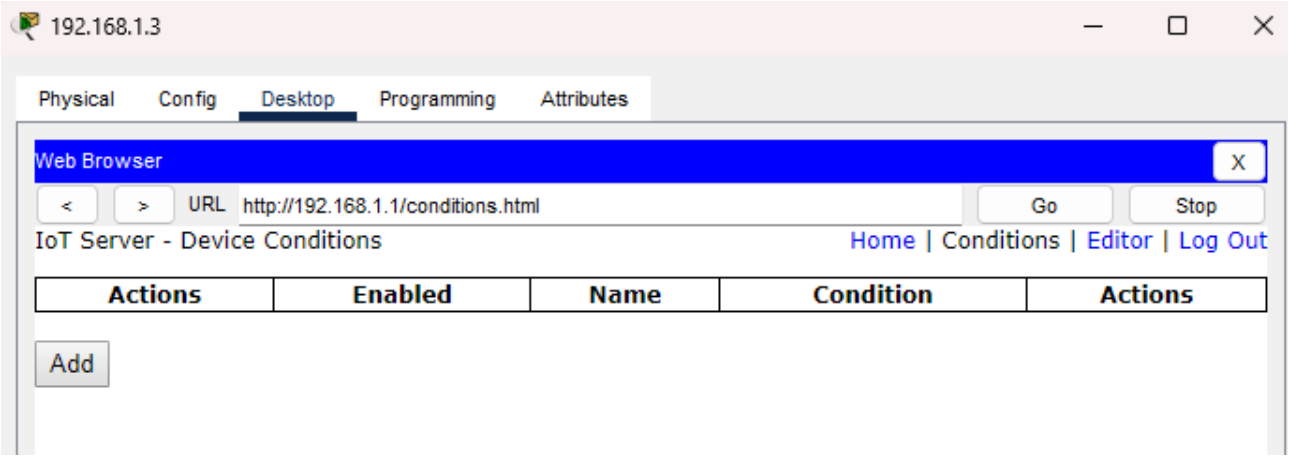

Рисунок 3.2.11 – Інтерфейс керування умовами пристроїв інтернету речей

Конкретно на рисунку 3.2.11 немає жодної умови, відповідно актуатори поки ніяк не задіяні ні при яких обставинах. Згідно до постановки задачі, необхідно, щоб у разі несанкціонованого проникнення має запрацювати сирена. Для прикладу буде представлено процес додавання правил конкретно для вікна. Необхідно натиснути на кнопку Add, після чого з’явиться вікно додавання правила, якому необхідно надати ім’я, умови спрацювання та положення стан кінцевого пристрою у разі задоволення умов (рисунок 3.2.12).

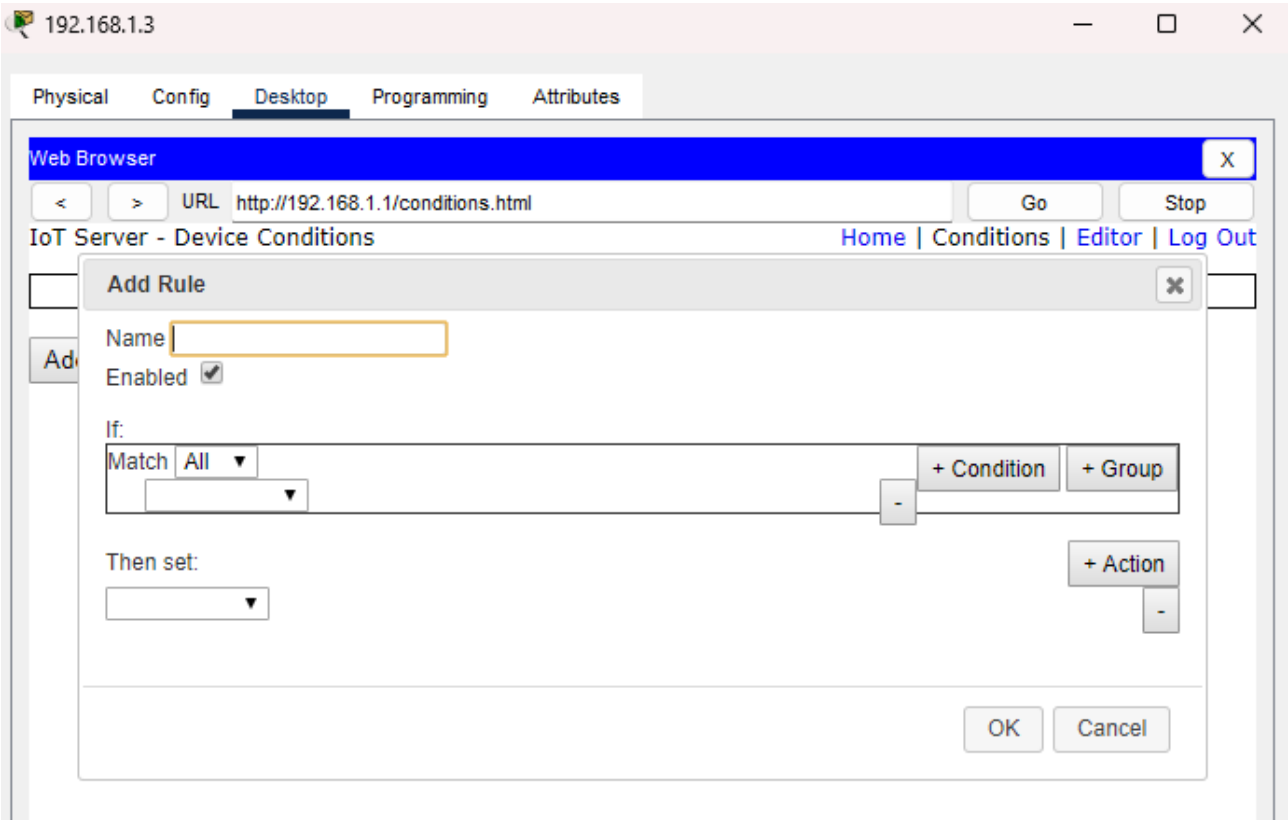

Рисунок 3.2.12 – Вікно додавання правила для IoT-мережі

Правило, що в разі відчиненого вікна має працювати сирена, буде виглядати приблизно як на рисунку 3.2.13.

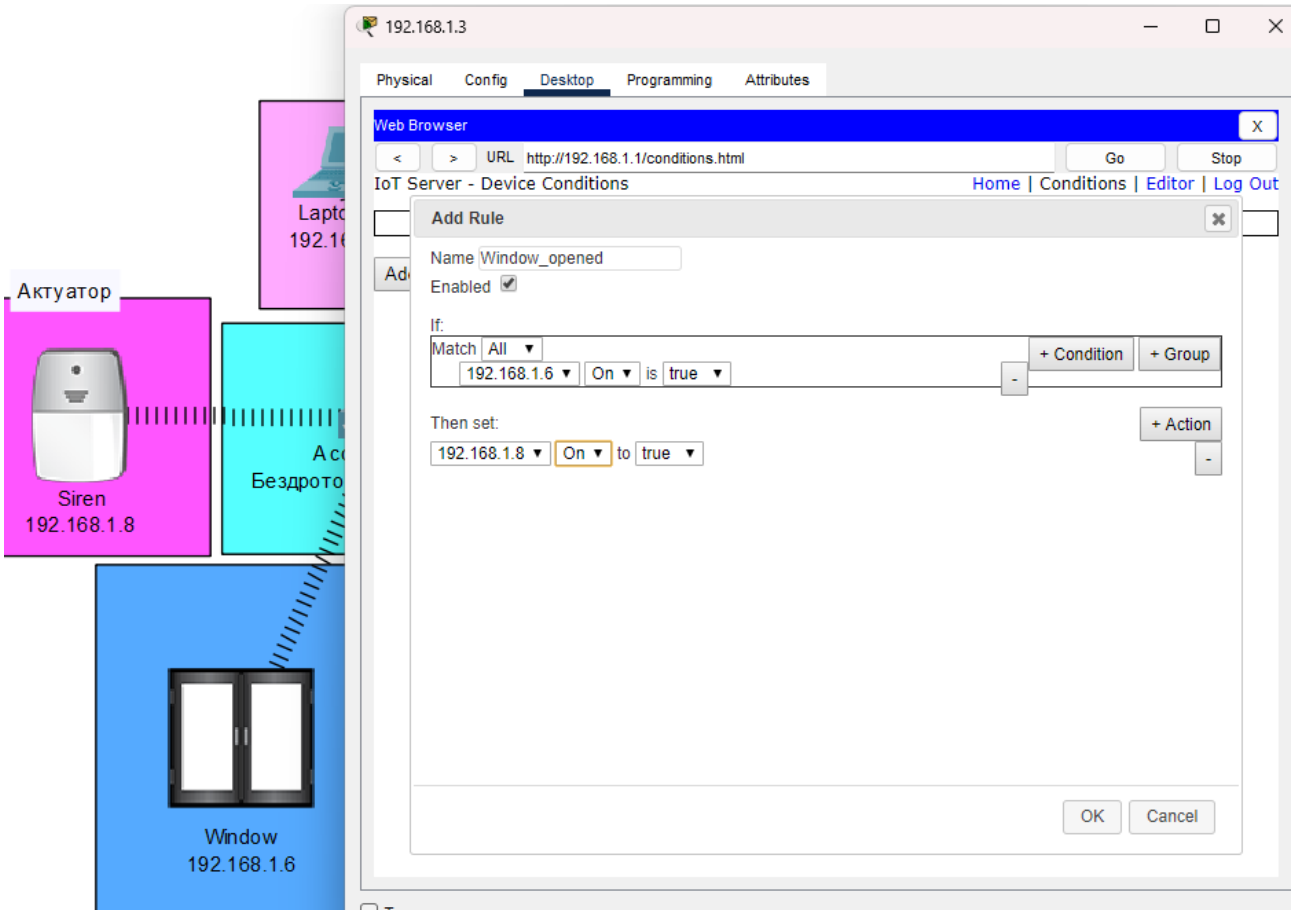

Рисунок 3.2.13 – Відображення умов правила, що сирена буде активуватися при відчиненні вікна

Після цього залишається натиснути на кнопку OK, і правило буде додано в список умов (рисунок 3.2.14).

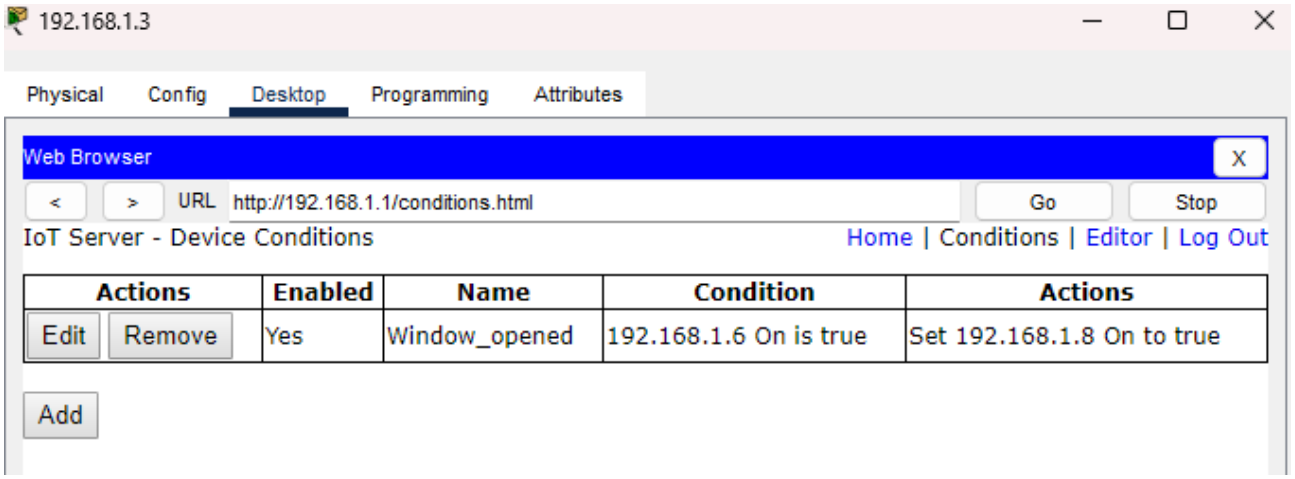

Рисунок 3.2.14 – Список умов після додавання правила

Тепер варто перевірити як працюватиме система. На рисунку 3.2.15 показано фрагмент як виглядає система до відкриття вікна, а на рисунку 3.2.16 після.

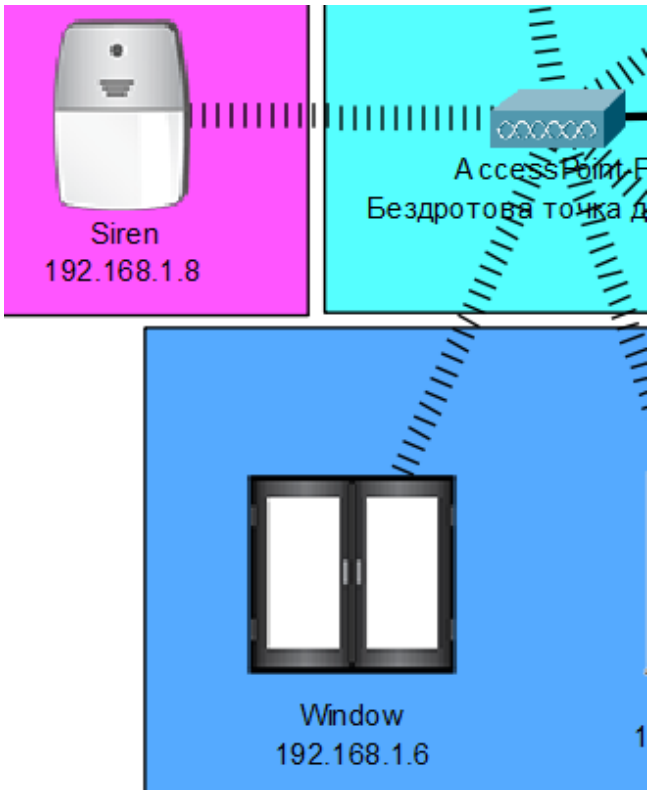

Рисунок 3.2.15 – Фрагмент системи до відкриття вікна

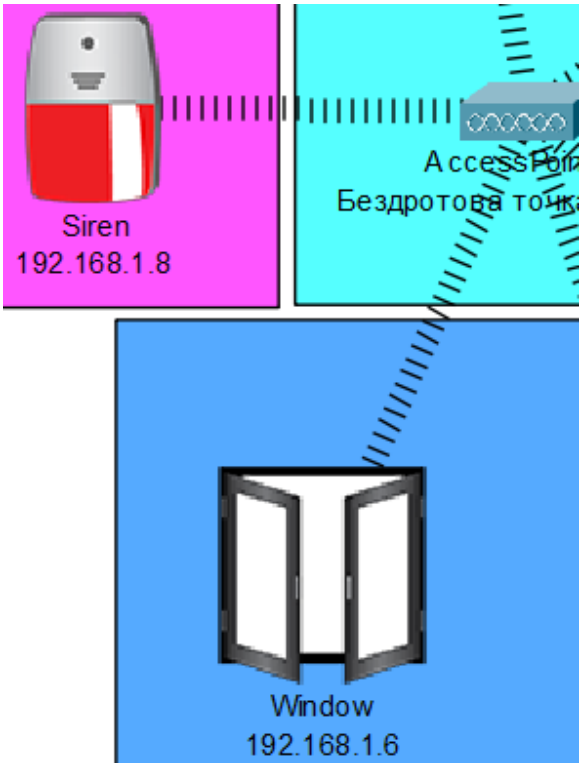

Рисунок 3.2.16 – Фрагмент системи після відкриття вікна

Для повного виконання завдання було додано відповідні правила для вхідних для гаражних дверей. Також, було додано правило автоматичного виключення сирени при умові, що вікно та двері зачинені (рисунок 3.2.17).

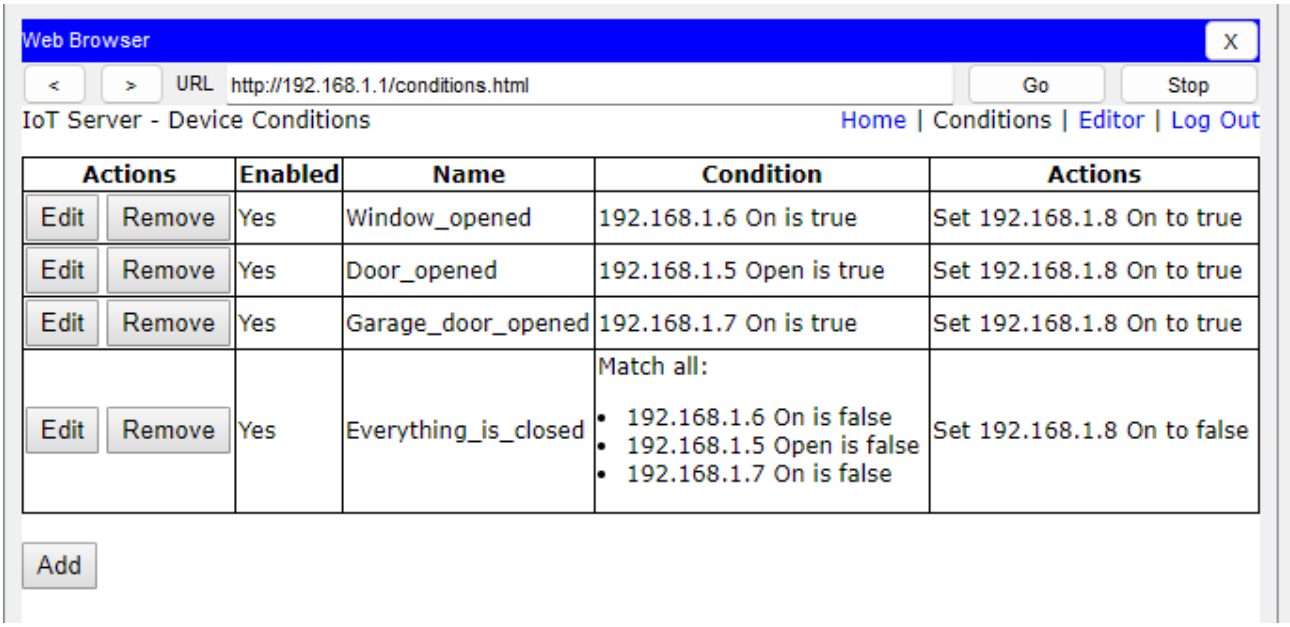

Рисунок 3.2.17 – Кінцевий список правил IoT-мережі

В результаті була отримана дієздатна модель IoT-мережі з можливістю гнучкого масштабування та налаштування.

3.3 Аналіз ефективності архітектури IoT-мережі.

Варто звернути увагу на те, що мережа складається доволі з малої кількістю вузлів. Тобто, вона зайвий раз не перенавантажена мережевими пристроями. Це забезпечує доволі ефективну та оптимальну роботу в рамках швидкості роботи. Та це є і мінусом конкретно цієї моделі. Проблема в тому, що вся мережа тримається на бездротовому підключенні кінцевих клієнтських пристроїв та власне самих IoT-пристроїв. Оскільки створена модель виконує функції саме охоронної системи, де стандарти надійності завжди вище стандартних, цій моделі не вистачає більшого розподілу навантаження між пристроями та більшої надійності.

3.4 Оптимізація роботи IoT-мережі

Беручи до уваги аналіз ефективності архітектури мережі IoT, було прийнято рішення щодо введення засобів розподілу навантаження та підвищення надійності. Першим засобом є додавання світча, до якого будуть підключені датчики і актуатори, а всі інші пристрої мережі будуть підключені до основного світча. Також, додавання додаткової бездротової точки доступу, що має підвищити якість з'єднання у випадку віддалених від покриття пристроїв, й також відокремить датчики і актуатори. Результат проведених робіт можна побачити на рисунку 3.4.1.

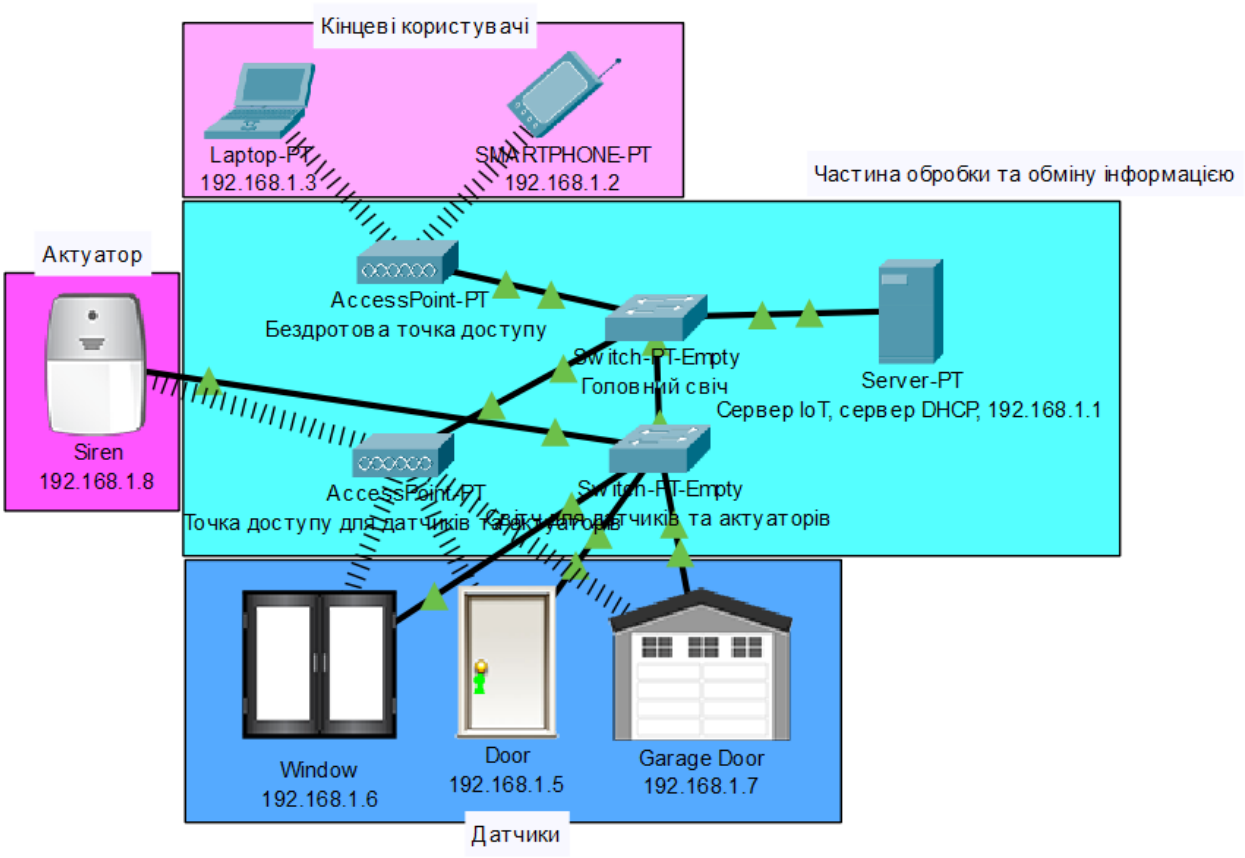

Рисунок 3.4.1 – Результат оптимізації IoT-мережі

3.5 Оцінка впровадження змін та їх вплив на продуктивність мережі

Оцінка проводиться у порівнянні розподілення мережевих клієнтів між мережевим обладнанням та варіаціями типами під'єднання пристроїв.

Як видно, в оптимізованій мережі було додано більше проміжних світчів та самі пристрої під'єднані до мережі дротовим та бездротовим способом. За результатом оптимізації була підвищена надійність та безвідмовність мережі, що

має гарантувати роботу навіть у разі виходу певних вузлів з ладу, питання яке в охоронних системах є критичним та потребуючим необхідної організації мережі для задоволення цих потреб.

ВИСНОВКИ

За результатом роботи було проведено дослідження архітектури інтернету речей, його розвиток, значення в сучасному світі та як ця концепція вплинула на архітектуру комп’ютерних мереж.

В ході виконання завдання було розглянуто концепцію та архітектуру інтернету речей, зокрема основні компоненти IoT-системи, протоколи та стандарти зв’язку, використання хмарних та периферійних обчислень.

Також, було проведено аналіз впливу IoT-мереж на архітектуру комп’ютерних мереж: зміни в мережевій архітектурі, використання IPv6, роль 5G та LPWAN, застосування SDN.

Окрім цього, розглянуто проблеми кібербезпеки IoT та методи захисту від них, включаючи використання блокчейн-технологій, а ще політики й стандарти безпеки IoT.

Для подальшого дослідження було проведено моделювання IoT-мережі, аналіз ефективності її роботи, оптимізація з урахуванням виявлених недоліків, оцінка впроваджених засобів оптимізації мережі.

В кінці була отримана повністю працездатна, гнучка та надійна IoT-мережа, яку можна буде брати за основу при створенні реальних IoT-мереж або моделювання більш складних IoT-систем.

Таким чином, проведена робота дозволила отримати комплексне уявлення про архітектуру інтернету речей, сучасні технології, що застосовуються в цих мережах, практичну основу для подальшої її інтеграції в реальних мережах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Интернет вещей [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/Интернет вещей, вільний. – Дата звернення: 05.06.2025;

2. Інтернет речей [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://uk.wikipedia.org/wiki/Інтернет_речей, вільний. – Дата звернення: 05.06.2025;

3. RFID [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/RFID, вільний. – Дата звернення: 05.06.2025;

4. QR-код [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/QR-код, вільний. – Дата звернення: 05.06.2025;

5. Штриховой код [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/Штриховой_код, вільний. – Дата звернення: 05.06.2025;

6. Data Matrix [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/Data_Matrix, вільний. – Дата звернення: 05.06.2025;

7. MAC-адрес [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/MAC-адрес, вільний. – Дата звернення: 05.06.2025;

8. IP-адрес [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/IP-адрес, вільний. – Дата звернення: 05.06.2025;

9. Кордонні обчислення [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу:

https://uk.wikipedia.org/wiki/Кордонні_обчислення, вільний. – Дата звернення: 05.06.2025;

10. Издатель – подписчик [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/Издатель_—_подписчик, вільний. – Дата звернення: 05.06.2025;

11. MQTT [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://uk.wikipedia.org/wiki/MQTT, вільний. – Дата звернення: 05.06.2025;

12. Constrained Application Protocol [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://en.wikipedia.org/wiki/Constrained_Application_Protocol, вільний. – Дата звернення: 05.06.2025;

13. Constrained Application Protocol [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://uk.wikipedia.org/wiki/HTTP, вільний. – Дата звернення: 05.06.2025;

14. LoRa [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://uk.wikipedia.org/wiki/LoRa, вільний. – Дата звернення: 05.06.2025;

15. Чорнобров К.В. Комп’ютерна інженерія: навч. посіб. / К.В. Чорнобров, І.М. Гаврилюк, В.В. Поліщук. – Луцьк: Луцький НТУ, 2020. – 259 с. – Режим доступу: https://elib.lntu.edu.ua/sites/default/files/elib_upload/123%20посібник%20комп'ютерна%20інженерія/page27.html, вільний. – Дата звернення: 06.06.2025;

16. Піраміда обчислень: хмарні, туманні та периферійні [Електронний ресурс] // GigaCloud. – 2023. – Режим доступу: https://gigacloud.ua/articles/piramida-obchyslen-hmarni-tumanni-peryferijni/, вільний. – Дата звернення: 06.06.2025;

17. Клієнт-серверна архітектура [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу:

https://uk.wikipedia.org/wiki/Клієнт-серверна_архітектура, вільний. – Дата звернення: 06.06.2025;

18. Peer-to-peer [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://uk.wikipedia.org/wiki/Peer-to-peer, вільний. – Дата звернення: 06.06.2025;

19. 5G [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/5G, вільний. – Дата звернення: 06.06.2025;

20. LPWAN [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://ru.wikipedia.org/wiki/LPWAN, вільний. – Дата звернення: 06.06.2025;

21. Програмно-конфігурована мережа [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – 2024. – Режим доступу: https://uk.wikipedia.org/wiki/Програмно-конфігурована_мережа, вільний. – Дата звернення: 06.06.2025;

22. Атаки на IoT: які бувають і як захистити пристрої [Електронний ресурс] // CoreWin. – Режим доступу: https://corewin.ua/blog/attacks-on-iot-how-protect/, вільний. – Дата звернення: 06.06.2025.