Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Навчально-науковий
інститут електроенергетики
(інститут)
Факультет інформаційних технологій
(факультет)
Кафедра інформаційних технологій та комп'ютерної інженерії
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА

кваліфікаційної роботи ступеня ___магістра___
(бакалавра, спеціаліста, магістра)

студента _____ Лісунов Ігор Євгенійович_____
(ПІБ)
академічної групи _____123м-22-1_____
(шифр)
спеціальності _____123 «Комп'ютерна інженерія»_____
(код і назва спеціальності)
за освітньо-професійною програмою ___«Комп'ютерна інженерія»_____
(офіційна назва)
на тему «Обґрунтування параметрів комп'ютерної системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT»_____
(назва за наказом ректора)

Керівники	**Прізвище, ініціали**	**Оцінка за шкалою**		**Підпис**
		рейтинговою	**інституційною**	
кваліфікаційної роботи	доц. Шедловський І.А.			
розділів:				
теоретичний розділ	доц. Шедловський І.А.			
синтез системи	доц. Бешта Д.О.			
розроблення програмного забезпечення	ас. Панферова Я.В.			

Рецензент				

Нормоконтролер	проф. Цвіркун Л.І.			

Дніпро
2023

ЗАТВЕРДЖУЮ:

завідувач кафедри
інформаційних технологій
та комп'ютерної інженерії
(повна назва)

_____ __Гнатушенко В.В._
(підпис) (прізвище, ініціали)
«___» _____ 2023 року

ЗАВДАННЯ
на кваліфікаційну роботу
ступеня ____магістр _____
(бакалавра, спеціаліста, магістра)

студенту _____ Лісунов І.Є. **_____ академічної групи** ____123м-22-1____
(прізвище та ініціали) (шифр)

спеціальності _____123 «Комп'ютерна інженерія»_____

за освітньою-професійною програмою _123 «Комп'ютерна інженерія»_____
(офіційна назва)

на тему «Обґрунтування параметрів комп'ютерної системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT»
затверджена наказом ректора НТУ «Дніпровська політехніка» від «___» _____ 2023 р. №____

Розділ	Зміст	Термін виконання
Стан питання та постановка завдання	На основі матеріалів виробничих практик, інших науково-технічних джерел сформулювати наукове завдання, конкретизувати предмет та мету досліджень	09.10.2023
Теоретичний	Обґрунтувати теоретичну базу розв'язання наукового завдання, якому присвячено роботу	25.10.2023
Синтез системи	Розробка комп'ютерної системи	6.11.2023
Розроблення програмного забезпечення	Розробка програмного забезпечення	20.11.2023
Експериментальний розділ	Проведення і обробка результатів експериментів	03.12.2023
Графічна частина	Графічні результати роботи подати у вигляді рисунків, схем, таблиць на 10 арк. формату А4.	03.12.2023

Завдання видано _____ доц. Шедловський І.А._
(підпис керівника) (прізвище, ініціали)

Дата видачі 27 вересня_2023 р.

Дата подання до екзаменаційної комісії 10.12.2023 р._____

Прийнято до виконання _____ Лісунов І.Є._____
(підпис студента) (прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 92 с., 59 рис., 3 табл., 1 дод., 17 джерел.

РОЗУМНИЙ БУДИНОК, ПРОТИПОЖЕЖНИЙ ЗАХИСТ, МЕРЕЖА, ТЕОРІЯ МАСОВОГО ОБСЛУГОВУВАННЯ, ВУЗОЛ

Об'єкт дослідження: комп'ютерна системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT.

Мета: синтез комп'ютерної мережі протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT, визначення параметрів мережевих пристроїв, мають значне перевантаження із-за дії вірусних програм, розгляд можливих ситуацій по втраті працездатності комп’ютерної мережі. Модернізація мережі для підвищення стійкості з інформаційного перевантаження.

Практичний результат: синтез комп'ютерної системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT, що не має виявлених проблем після додаткової модернізації. При аналізі мережі застосовувався науковий підхід з використанням можливостей теорії масового обслуговування для моделювання мережі і вирішення поставлених завдань.

Розроблений практичний підхід до розробки і моделювання комп'ютерних мереж забезпечує ефективність, працездатність і економічну перевагу перед апаратним дослідженням поведінки мережі.

Результати роботи з моделювання комп'ютерної мережі системи протипожежного захисту розумного будинку оформлено у вигляді таблиць, графіків, що пояснюють і демонструють процес розрахунку, моделювання і представлені в пояснювальній записці і в додатку.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАК, ОДИНИЦЬ І ТЕРМІНІВ

ІС – інформаційна система;

КС – комп’ютерна система;

ПК – персональний комп’ютер;

КМ – корпоративна мережа;

ЛМ – локальна мережа;

ВСТУП

Сучасні котеджні комплекси мають широкий спектром загроз, починаючи від злочинності, громадянськими заворушеннями, пожежами та іншими надзвичайними ситуаціями.

Щоб пом'якшити наслідки цих ситуацій, відповідним службам потрібна інформація в режимі реального часу та уявлення про те, що відбувається в їхніх районах та навколо них. Зростає аргумент на користь використання нових технологій для створення більш безпечних компактного проживання. У цьому контексті інтегровані рішення з пожежної безпеки. Ці рішення допомагають зробити міста безпечнішими та захищеними, ефективно запобігати, управляти та реагувати на потенційні сценарії ризику.

Система управління «розумним будинком» призначена для узгодження роботи інженерних систем будинку. Система управління «розумним будинком», оцінюючи стан датчиків, подає команди пультам управління, різним системам «розумного дому», прив'язуючись до реального часу, враховуючі сезонні параметри і т. ін.

Мета роботи і завдання дослідження. Розробка та дослідження роботи комп'ютерної системи протипожежного захисту розумного будинку для котеджному комплексу з віддаленим контролем через сервер MQTT.

Виявлення та усунення проблем з перевантаженням мережевих вузлів, визначення необхідних параметрів, заміна за їх за потреби потрібним мережевим обладнанням. Розробка заходів підвищення стійкості роботи комп’ютерної мережі в умовах підвищеного інформаційного трафіку.

Об’єкт дослідження. Об’єктом дослідження є програмно-технічна реалізація комп'ютерної системи протипожежного захисту розумного будинку.

Предмет і методи дослідження. Предметом дослідження є структура комп'ютерної системи протипожежного захисту розумного будинку, аналіз її

властивостей з метою визначення необхідних граничних параметрів апаратного забезпечення для мережевого обладнання. Дослідження комп'ютерної системи протипожежного захисту розумного будинку здійснювалось за допомогою методів теорії масового обслуговування. Оцінка поведінки комп’ютерної мережі здійснювалась за показниками отриманої її математичної моделі. В магістерській роботі була синтезована і досліджена комп'ютерна система протипожежного захисту розумного будинку. Дослідження проводилось з урахуванням різних показників інформаційного середовища, та пропонувались варіанти вирішення для усунення негативних показників роботи комп’ютерної мережі.

Ідея роботи. Ідея роботи полягала в виявленні і усуненні найбільш вразливих мережевих вузлів комп'ютерної системи протипожежного захисту розумного будинку, які в умовах підвищеного трафіку призведуть до різкого зниження продуктивності роботи мережі.

Практичні результати. Практичні результати полягають в синтезі комп'ютерної системи протипожежного захисту розумного будинку і виявленні можливих її недоліків, пошуку шляхів усунення цих недоліків. Отримані результати ґрунтуються на науковому підході для вирішення поставлених завдань із застосуванням методів моделювання роботи мережі і є гарним методом перевірки працездатності мережі, який не потребує значних капіталовкладень та часу, характерних для апаратного методу дослідження поведінки комп’ютерної мережі.

1 СТАН ПИТАННЯ І ЗАВДАННЯ ДОСЛІДЖЕННЯ

1.1 Загальні відомості

Використання комп’ютерних систем з різноманітними IT-системи дає можливість швидко приймати рішення для різних завдань, спрощувати повсякденну роботу і автоматизувати та вдосконалити структуру інформаційних потоків, в тому числі і системи документообігу. Це також може забезпечити споживачам доступ до унікальних послуг і знизити витрати на виробництво продукції і послуг, в тому числі інформаційних послуг [5].

Комп'ютерні системи можна класифікувати за типами [6]:

- вільне використання даних між різними комп'ютерами;

- при прийомі і передачі даних або інформації відбувається взаємодія користувача;

- інтерактивна інформаційна система – це можливість передачі та обміну даними в діалоговому режимі, тобто це інформаційно-обчислювальна комп’ютерна система;

- управління потоком, наприклад, електронний лист, який містить HTML-файл, викликає обробку документа або оператор електронної пошти отримує електронний документ, вибирає формат, а потім запускає процесор, при цьому весь процес вимагає контролю оператора;

- автоматизація можливості прийому та обробки документів у певному форматі може дозволити використовувати такий процес без участі оператора.

1.1.1 Розумні міста

Численні муніципалітети використовують модель розумного міста у великих містах, щоб покращити якість життя своїх мешканців, ефективно використовувати місцеві ресурси та заощадити операційні витрати. Ця модель включає багато

різнорідних технологій, таких як Cyber-Physical Systems (CPS), Wireless Sensor Networks (WSNs) і Cloud Computing (ClCom).

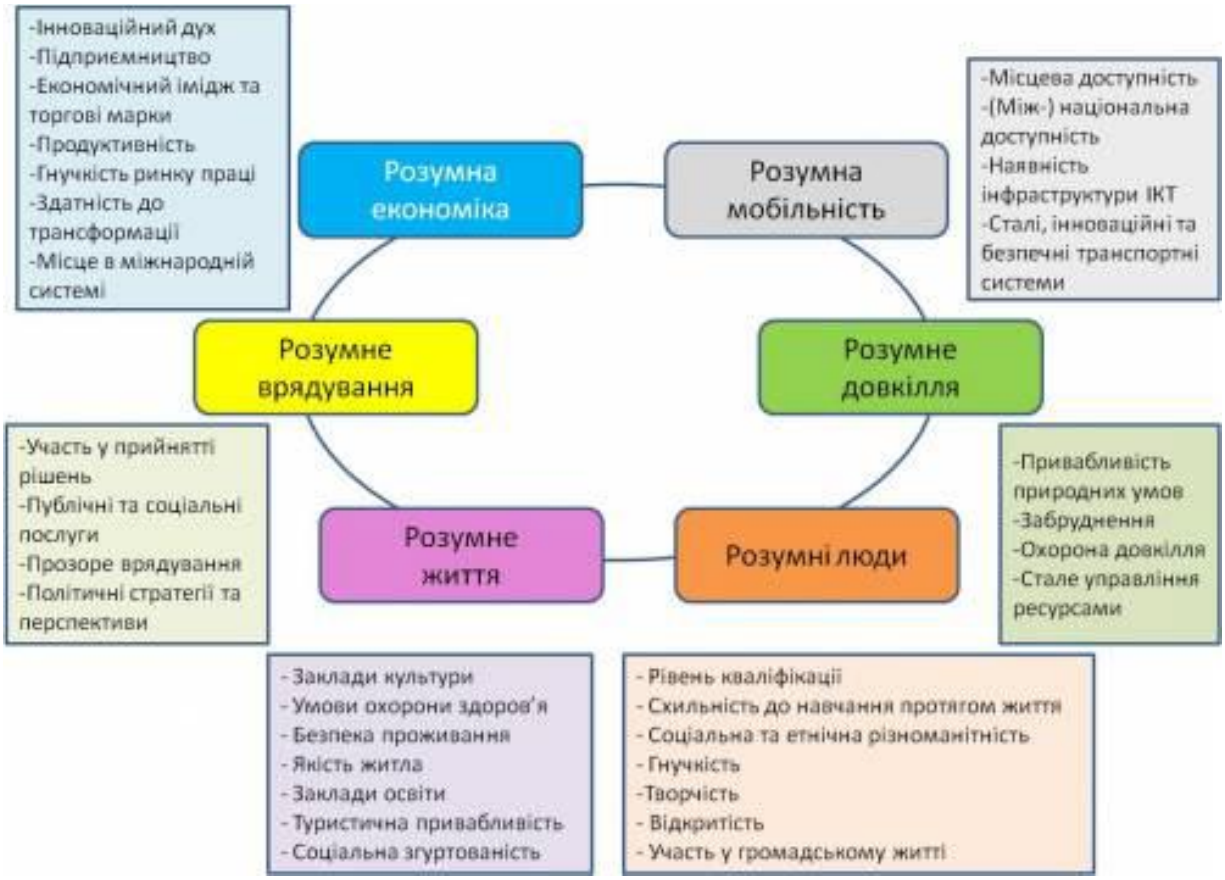


Рисунок 1.1 - Розумне місто

Ефективні мережеві та комунікаційні протоколи необхідні для забезпечення суттєвої гармонізації та контролю багатьох системних механізмів для досягнення цих важливих цілей. У цьому дослідженні визначено мережеві вимоги та характеристики додатків для розумних міст (SCA), а також мережеві протоколи, які можна використовувати для обслуговування різноманітних потоків трафіку даних, необхідних між різними механізмами.

Зараз в крупних містах України триває активне впровадження мережевих проектів кількох систем розумного міста, таких як розумний транспорт, розумна будівля, розумний дім, розумна мережа, розумна пожежна охорона, розумна вода, моніторинг трубопроводів і системи керування та інше.

Місто можна охарактеризувати як «розумне», якщо інвестиції в людський та соціальний капітал, а також в транспорт та сучасну комунікаційну інфраструктуру закладають основи для сталого економічного розвитку та високої якості життя, а природні ресурси раціонально управляються за допомогою спільного управління.

Деякі компанії та міста також використовують «розумні міста» як маркетингову концепцію.

Кожен автоматичний комплекс управління будівлею умовно поділяється на декілька рівнів:

1. Рівень екранів - це те, з чим користувач повинен спілкуватися, людино-машинний інтерфейс. Засобом взаємодії можуть служити планшети, клавіатури, пульти дистанційного керування, кишенькові комп'ютери, мобільні телефони і т.д.;

2. Рівень алгоритмів є серцем системи - це пристрій, в якому вбудовані алгоритми управління працюють безпосередньо - з програмним забезпеченням або їх комбінація, об'єднана в багаторівневу мережу;

3. Рівень комунікації - фізичний і логічний рівень для керованих підсистем, датчиків і двигунів;

4. Рівень управління - єдиний з трьох шарів, з яким взаємодіє користувач. Він може не знати про існування інших. Це засоби прямого зв'язку між користувачем і системою. Інтерфейс може складатися з сенсорних екранів (бездротових і стаціонарних, кольорових і монохромних) різного розміру, кнопкового управління або багатофункціональних пультів дистанційного керування, мобільного телефону, а також персональних і звичайних комп'ютерів, що дозволяють управляти будинком (квартирою) по локальній мережі і віддалено - через інтернет і SMS.

Віддалене управління - це не тільки зручно, але і необхідно. Без нього неможливо регулювати зворотний зв'язок домовласника з будинком.

1.1.2 Комп'ютерної системи «розумного котеджного комплексу»

Сучасні котеджні комплекси мають широкий спектром загроз, починаючи від злочинності, громадянськими заворушеннями, пожежами та іншими надзвичайними ситуаціями.

Щоб пом'якшити наслідки цих ситуацій, відповідним службам потрібна інформація в режимі реального часу та уявлення про те, що відбувається в їхніх районах та навколо них. Зростає аргумент на користь використання нових технологій для створення більш безпечних компактного проживання. У цьому контексті інтегровані рішення з пожежної безпеки. Ці рішення допомагають зробити міста безпечнішими та захищеними, ефективно запобігати, управляти та реагувати на потенційні сценарії ризику.

Беручи до уваги інфраструктуру, задіяну в обмежених місцях проживання, моніторинг та забезпечення пожежної безпеки стає величезною проблемою.

Поява розумних технологій, таких як уніфікований зв'язок та IP-мережі, все частіше сприяє переходу до «безпечнішого міста». В основі цього безпечного міста лежить основний зв'язок між різними зацікавленими сторонами, включаючи правоохоронні органи та державні та охороні установи. З розвитком технологій, що забезпечують більшу сумісність і безперебійний потік інформації через уніфіковані мережі, стало набагато простіше збирати і зіставляти широко доступні дані для скоординованого реагування. Ця структура також гарантує, що спільна платформа та спільний портфель рішень сприятимуть отриманню дієвої інформації та аналітичних даних для прийняття ефективних рішень та швидкого реагування на критичні ситуації.

Кінцевою метою сучасних міст є створення єдиної структури безпеки для забезпечення ефективного реагування на будь-яку серйозну надзвичайну ситуацію. В основі цього лежить інформаційна технологія, яка дозволяє безперешкодно інтегрувати різні окремі компоненти в повне, консолідоване уявлення про інфраструктуру безпеки міста.

Структурна схема комплексу технічних засобів комп'ютерної системи «розумного кодетжного комплексу» складається з трьох рівнів (рис. 1.2):

– нижній рівень – системи розумного дому, таких як охорона, протипожежний захист, клімат контроль та інше;

– середній рівень - контролер, який бере і обробляє інформацію з датчика, а також обладнання мережі передачі даних;

– верхній рівень - АРМ оператора з головним сервером, що знаходиться віддалено та АРМ користувача з «розумного кодетжного комплексу» з локальним сервером.

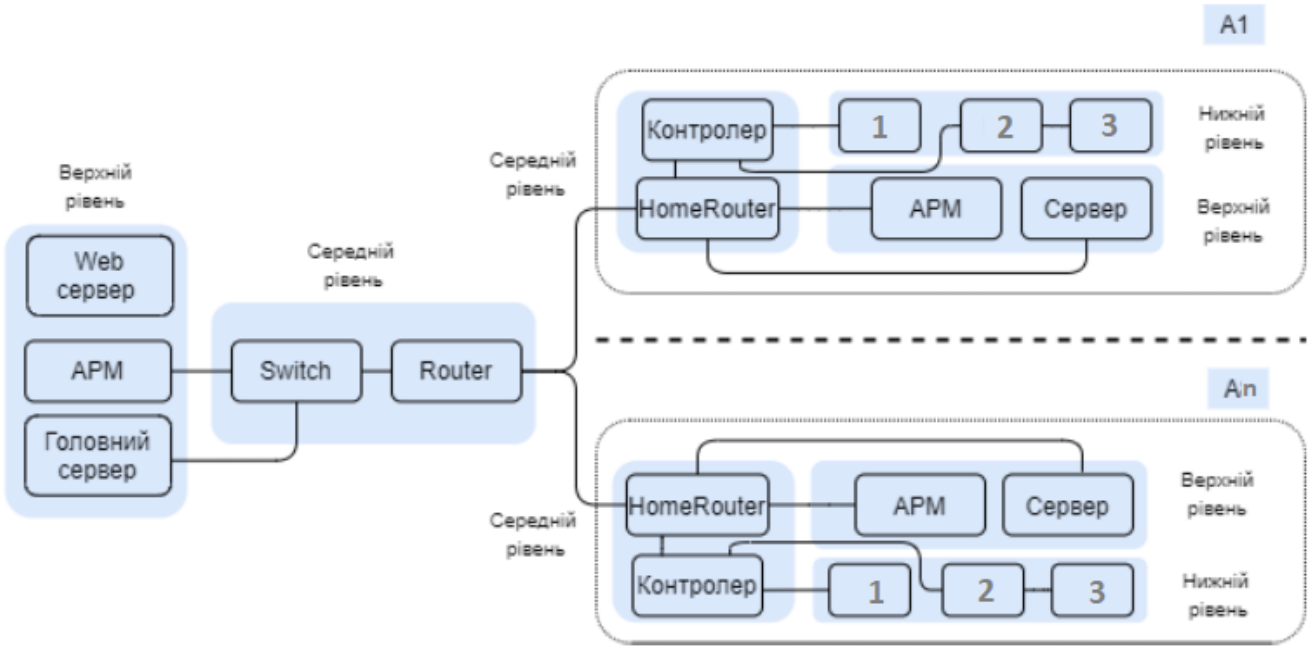


Рисунок 1.2 - Структурна схема «розумного кодетжного комплексу»

Верхній рівень забезпечує збір, інтелектуальну обробку та зберігання отриманої оперативної інформації, сигналізацію про вихід параметрів з зони встановленого діапазону «технологічних параметрів», створення журналу подій та виконую інші корисні функції – автоматичне сповіщення відповідних комунальних служб та інше.

1.1.3 «Розумний будинок»

Технології постійно розвиваються, і вже сьогодні всім відомі реальні можливості, які раніше виглядали як уявлення фантастів, з'являлися тільки в книгах або в кіно. Те, що колись вважалося фантастикою, таке, як автоматичне нагрівання води в чайнику без участі людини, тепер є повсякденною реальністю, доступною завдяки технології "розумного будинку".

Термін "розумний будинок" (Smart Home), походить із заходу і визначає автоматизацію побутових систем, створення зручних систем для спрощення життя людей. Завдяки "розумному будинку" повсякденні обов'язки вже не забирають багато часу, не приносять непорозумінь та втоми власникам будинку [1].

Це поняття часто включає в себе всі аспекти, які значно полегшують проживання та перебування в житлових приміщеннях, будинках і т.д. Зазвичай розуміють, що "розумний будинок" означає автоматизацію чайників, кухонних приладів, освітлення, проекторів, телебачення та інших мультимедійних пристроїв. Проте важливо розуміти, що система "розумного будинку" належить до ширшого спектру, аніж просто автоматизація управління чайником; вона включає в себе керування системами опалення, водопостачання, системами безпеки та відеоспостереженням [1].

Під час створення таких будинків проекти передбачають встановлення спеціальних пристроїв та датчиків, які здатні увімкнути та вимкнути світло, регулювати температуру у кімнатах, контролювати опалення, охоронні системи та, взагалі, полегшити господарям управління будинком.

Система управління «розумним будинком» (рис. 1.3) призначена для узгодження роботи інженерних систем будинку. Система управління «розумним будинком», оцінюючи стан датчиків, подає команди пультам управління, різним системам “розумного дому”, прив'язуючись до реального часу, враховуючі сезонні параметри і т. ін.

Сучасний рівень ІТ-технологій забезпечує власнику управляти параметрами комфорту не тільки з сенсорних панелей та «розумних» вимикачачей, а також використовувати з комп’ютерів планшетів, та мобільних телефонів, дистанційно за допомогою Інтернету з любої точки світу. За допомогою різних систем управління “розумного дому” можна управляти всім устаткуванням, яке функціонує в будинку: світлом, кліматом, охороною, пожежним захистом, звуком, відеоспостереженням, домашнім кінотеатром, різноманітною побутовою електронікою, комп'ютером і та іншим ІТ-обладнанням.

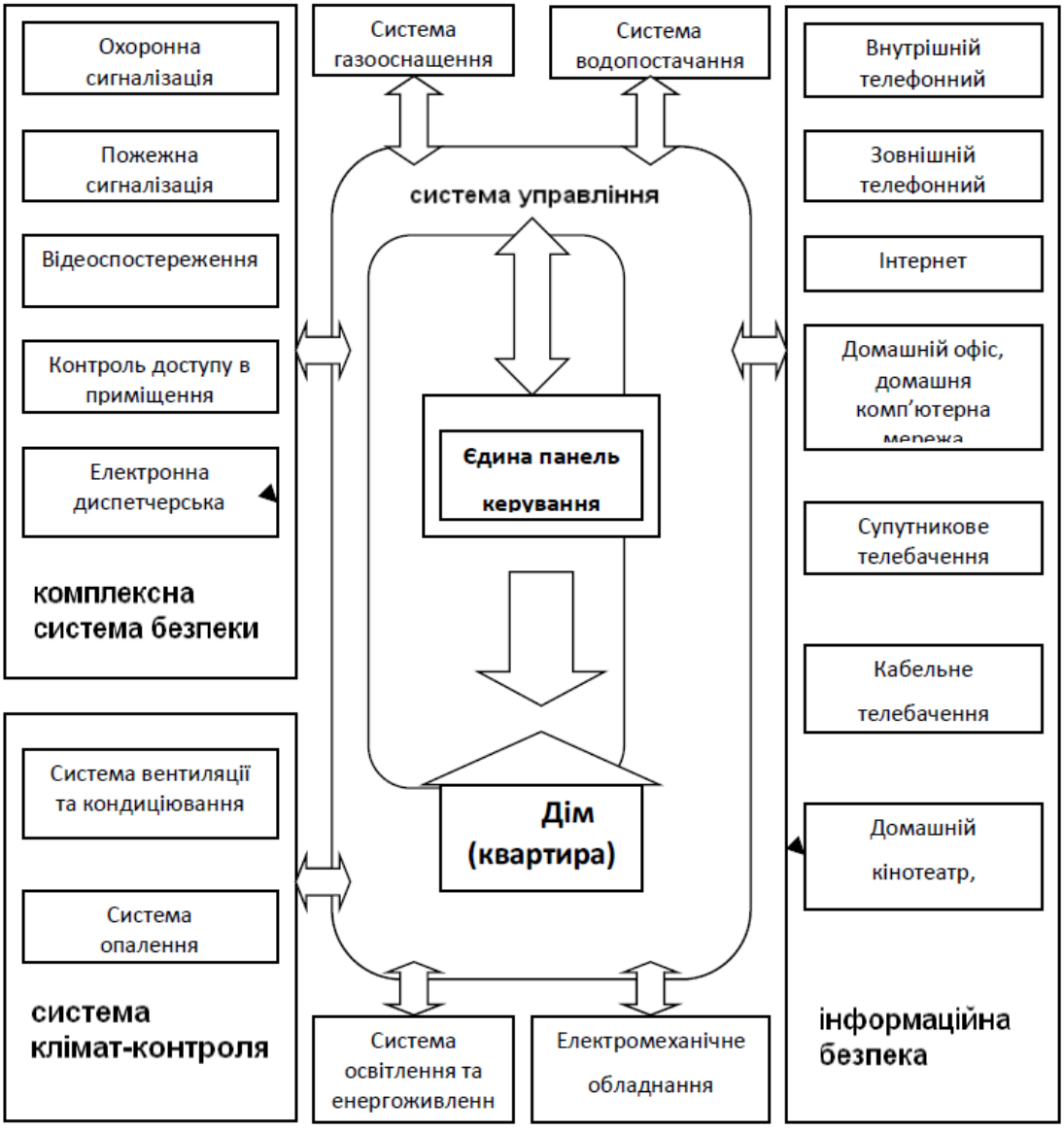


Рисунок 1.3 – Підсистемами керування «розумним будинком»

1.1.4 Комп'ютерна система протипожежного захисту «розумного будинку» з віддаленим керуванням через сервер MQTT

1.1.4.1 Загальна інформація

Комп'ютерна система протипожежного захисту «розумного будинку» в котеджному містечко Concept Riviera (м. Дніпро) з віддаленим керуванням через сервер MQTT.

Рисунок 1.4 – Котеджне містечко Concept Riviera

Новобудова котеджного містечка Concept Riviera проводитися с 2023 р. по 2024 р. і розташоване за адресою: м. Дніпро, пров. Широкий, GPS координати: 48.51474040176241,34.9799100693657, локація Concept Riviera на карті показана на рис. 1.5 [13].

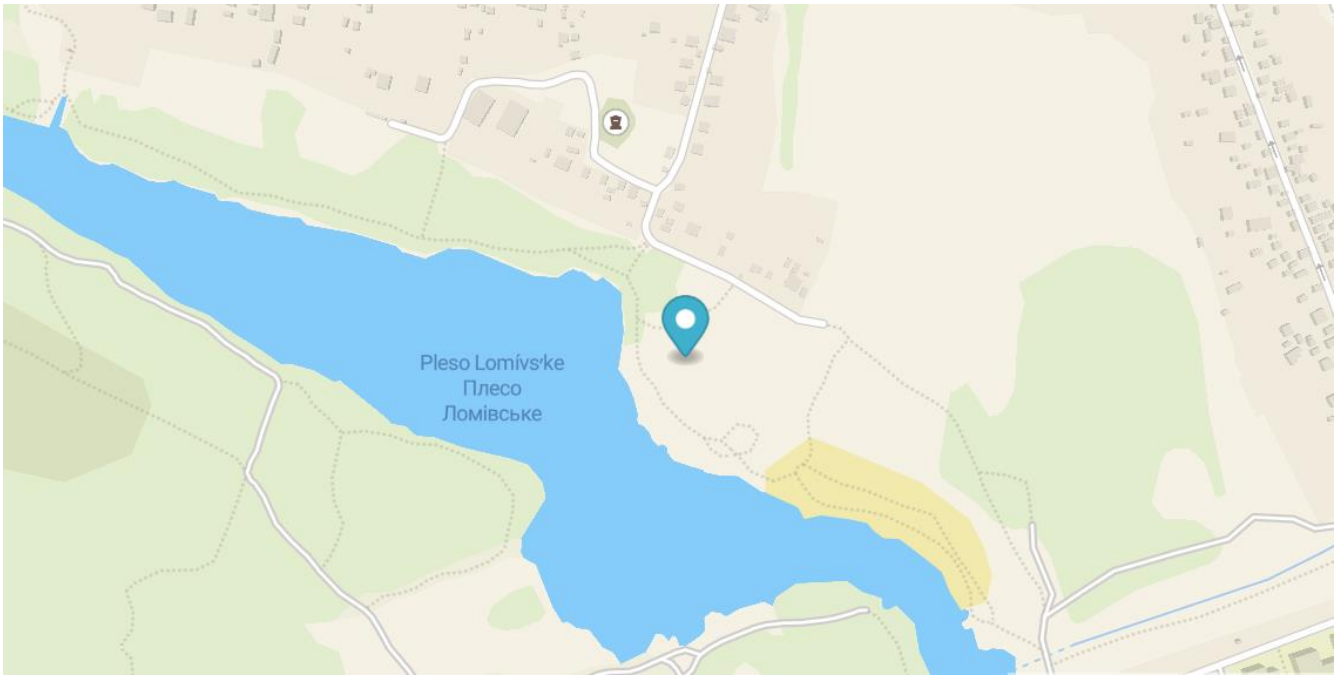


Рисунок 1.5 – Локація котеджного містечка Concept Riviera

Віддаленим керуванням через сервер MQTT - це система, яка призначена для захисту житлових будинків котеджного комплексу від пожежі.

1.1.4.2 Протокол MQTT

MQTT (Message Queuing Telemetry Transport) - це стандартний протокол обміну повідомленнями, розроблений спеціально для використання в додатках IoT.

MQTT був розроблений Енді Стенфорд-Кларком і Арленом Ніппером, які працювали в лабораторії програмного забезпечення IBM в Кембриджі. Протокол став дуже популярним серед Інтернету речей, оскільки він забезпечує модель публікації та підписки, що дозволяє легко створювати зв'язок між різними пристроями або датчиками.

Протокол MQTT заснований на TCP/IP за стандартом OASIS, який дотримується моделі публікації та підписки. Модель публікації та підписки призначена для того, щоб дозволити надсилати повідомлення в будь-якому напрямку між клієнтом і сервером. Це дає можливість пристроям IoT створювати зв'язок один з одним, незалежно від їх географічного розташування. Протокол

MQTT гарантує, що повідомлення доставляються, навіть якщо мережі ненадійні або не відповідають. Він використовує систему підтвердження, яка дозволяє обом сторонам знати, чи правильно були отримані дані чи ні.

Модель публікації та підписки. У цій моделі один пристрій, який називається видавцем, надсилає повідомлення на будь-який інший пристрій, зацікавлений у їх отриманні - це може бути окремий датчик або інший тип комп'ютера, підключеного до Інтернету, наприклад сервер. Ті пристрої, які хочуть отримувати дані від видавця, які називаються передплатниками, надсилають назад підтвердження, якщо отримали їх правильно.

Видавці та клієнти системи не взаємодіють один з одним безпосередньо. Замість цього брокер управляє зв'язком між двома організаціями, фільтруючи всі вхідні повідомлення і розподіляючи їх між потрібними абонентами.

Протокол MQTT представляє ці концепції з брокером MQTT, клієнтами MQTT і темами MQTT (рис. 1.6).

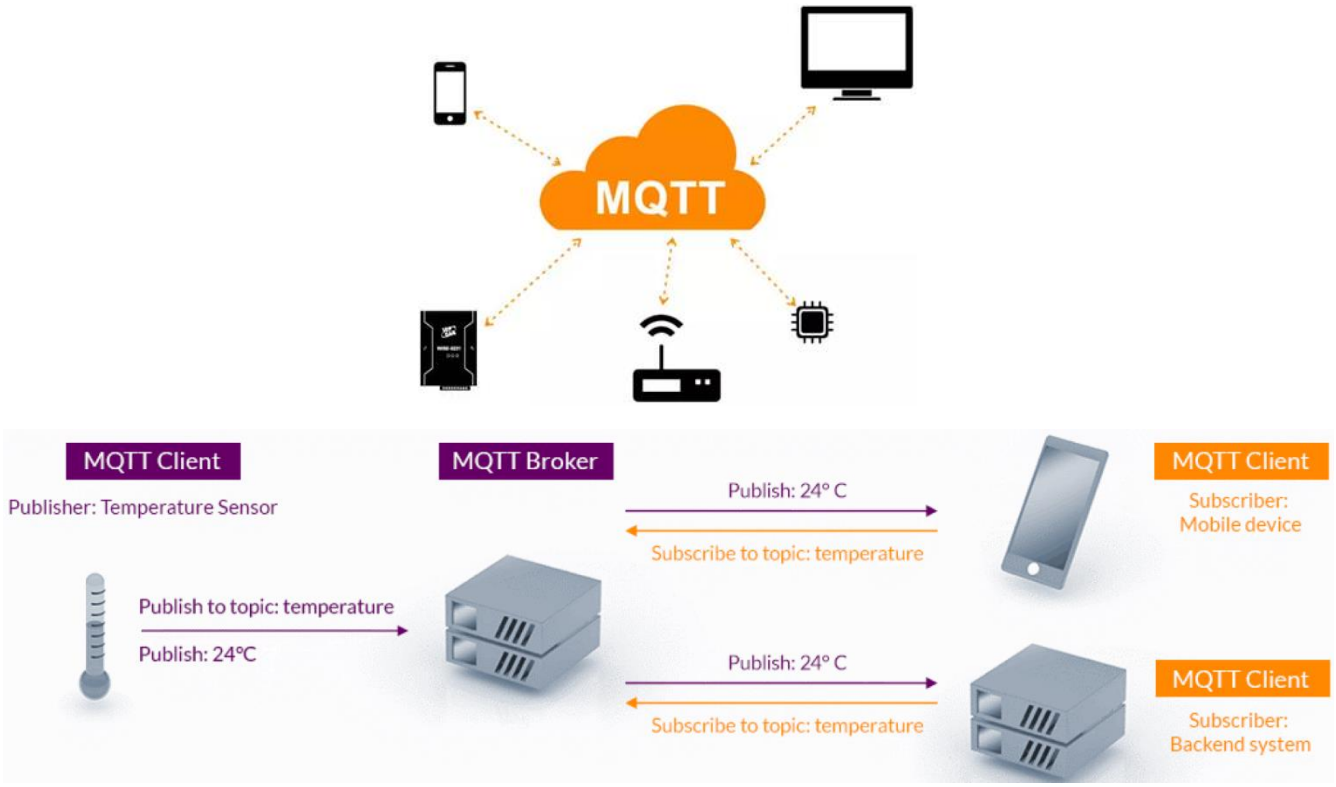


Рисунок 1.6 – Модель публікації та підписки протоколу MQTT

Брокер MQTT. Брокер лежить в основі системи. Він відповідає за отримання всіх повідомлень, їх фільтрацію та надсилання передплатникам, зокрема клієнтам MQTT. Брокер MQTT потенційно може обслуговувати мільйони підключених клієнтів MQTT.

MQTT-клієнт. Клієнт – це, по суті, будь-який пристрій, який може взаємодіяти з брокером для надсилання та отримання повідомлень. Клієнтом може бути крихітний датчик IoT, який доставляє дані через постійні проміжки часу, або розумний додаток на комп'ютері з графічним представленням даних IoT.

Клієнт може підписатися на задану тему в брокера, щоб отримувати відповідні повідомлення цієї теми. Аналогічним чином, клієнт може публікувати повідомлення під задану тематику, які брокер пересилає передплатникам цієї теми.

Тема MQTT. Теми використовуються для реєстрації інтересу до певного типу вхідних повідомлень і, навпаки, для вказівки місця публікації вихідних повідомлень. Тема MQTT може містити кілька рівнів теми, розділених косою рискою (рис. 1.7).

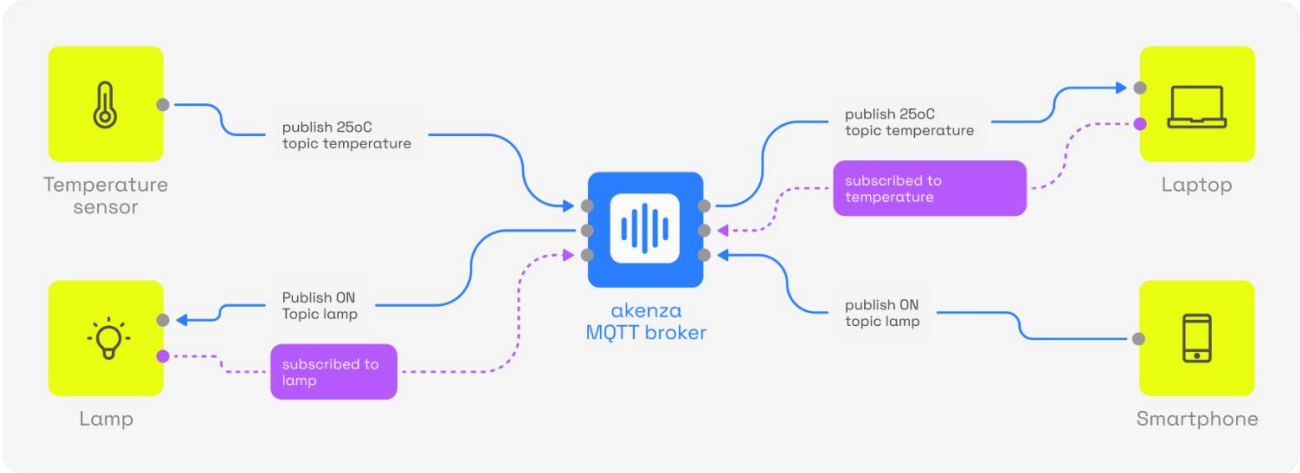


Рисунок 1.7 – Тема протоколу MQTT

В останні роки MQTT став одним з основних протоколів для IoT-рішень. Це пов'язано з декількома факторами.

По-перше, це один із найлегших протоколів, які зараз використовуються в IoT. Це відкритий стандарт, який може бути реалізований на будь-якому обладнанні

або програмному забезпеченні. Клієнтські бібліотеки доступні для всіх основних мов програмування, що дозволяє легко створювати додатки IoT за допомогою MQTT.

Завдяки моделі публікації та підписки, гнучкість, яку пропонує MQTT, дозволяє підтримувати різні типи сценаріїв використання та архітектури проектів IoT. Зазначимо, що видавцям і передплатникам системи навіть не потрібно знати про існування один одного, оскільки всіма з'єднаннями займається брокер.

Протокол дозволяє реалізовувати масштабовані проекти, можливо, з'єднуючи мільйони пристроїв IoT в систему. Двонаправлений зв'язок MQTT дозволяє транслювати повідомлення на великі групи пристроїв.

Нарешті, MQTT підтримує численні механізми автентифікації та безпеки даних, такі як шифрування TLS [17].

1.1.4.3 Апаратна частина комп’ютерної системи

Комп’ютерна система складається з декількох компонентів: Сенсори - це пристрої, які вимірюють різні параметри навколишнього середовища, такі як температура, вологість, концентрація диму та ін. Сенсори передають ці дані на центральний пристрій системи.

Центральний пристрій - це комп'ютер, який обробляє дані з датчиків і приймає рішення про те, чи виникла пожежа. Центральний пристрій також може віддалено керувати іншими пристроями в будинку, такими як пожежні сповіщувачі, системи вентиляції та ін.

Сервер MQTT - це сервер, який забезпечує зв'язок між центральним пристроєм і зовнішнім світом. Сервер MQTT дозволяє користувачам віддалено керувати системою протипожежного захисту.

Система управління «розумний будинок»" використовує передові технології для автоматизації майже всіх інженерних комунікацій, з метою звільнення людини

від непотрібної рутинної роботи. Комп'ютерна система «Розумний будинок» забезпечує [1]:

- повний контроль безпеки, включаючи відеоспостереження та датчики руху для попередження про несанкціонований доступ на територію будинку;

- роботу системи протипожежного захисту;

- автоматичне регулювання електропостачання для забезпечення безперебійної подачі електроенергії та включення резервного живлення за необхідності;

- управління системою освітлення, включаючи аварійне та чергове освітлення;

- підтримку оптимальної температури в будинку для комфортного перебування, враховуючи функціонування систем опалення, кондиціювання та вентиляції;

- оптимізацію системи водопостачання для економії води та встановлення систем антизатоплення;

- автоматизацію роботи побутових приладів, необхідних для прибирання або приготування їжі, а також мультимедійного та телевізійного обладнання;

- контроль якості функціонування та безпеки всіх систем у будинку, а також відстеження стану самої будівлі;

В кваліфікаційні роботі магістра детально будуть розглядатися наступні питання;

1. Синтез системи охоронно-пожежна сигналізація для «розумному будинку», що представляє собою комплекс обладнання, що забезпечує безпеку майна від різних непередбачених ситуацій, таких як протікання та пожежі;

2. Моделювання роботи комп'ютерної системи для охоронно-пожежної сигналізації «розумного котеджного комплексу».

1.2 Галузь застосування комп’ютерної системи

Встановлення системи протипожежного захисту для «розумного дому» дозволяє захистити будівлю від різних непередбачених ситуацій, включаючи:

- протікання в системах забезпечення водою та газом.

- коротке замикання в електромережі.

- займання.

- наслідки аварій різних інженерних систем.

Розумні датчики протипожежної системи реагують на виникнення диму, перебої в роботі електрики тощо. Вони надсилають сигнал головному центру системи, який автоматично вживає комплекс заходів для попередження можливого розвитку нештатної ситуації.

Основні переваги такої системи включають:

при спрацьовуванні датчиків протипожежної системи, активується система, яка блокує електрощит та перекриває газові вентилі.

- протипожежна система може сповіщати не лише осіб, що перебувають у будинку, а й відповідні служби. Також власник будівлі отримує СМС на мобільний телефон, що дозволяє оперативно реагувати у нештатних ситуаціях, керуючи системою зі смартфона.

Усі записи передаються на віддалений сервер через «хмару».

1.2.1 Предмет впровадження апаратної частини для «розумного будинку»

Предмет впровадження апаратної частини - особистий будинок, на прикладі наступної характеристики будинку [2]:

- будинок має два поверхи, включаючи гараж, загальна площа кімнат та приміщень становить 174,6 м^2;

- стіни та несучі конструкції виготовлені з цегли товщиною 400 мм;

- перекриття між поверхами виконане з пустотних залізобетонних плит;

- внутрішні перегородки також виготовлені з цегли, товщина 200 мм;

- стеля - підвісна, загальна висота від підлоги 270 см, висота міжповерхового проміжку 20 см (на першому поверсі) та 50 см (на другому поверсі);

- підлога в кімнатах покрита ламінатом, у кухні, в ванній кімнаті та у санвузлах - кахельною плиткою;

- вікна - металопластикові пакети з подвійним склом;

- двері виготовлені з металу та дерева;

- встановлена автономна система опалення;

- системи електропостачання, водопостачання та каналізації - централізовані.

Таблиця 1.1 - Специфікація приміщень будинку

Ідентифікатор приміщення	Найменування кімнати/приміщення
1	Кабінет на першому поверхі
2	Гостьова кімната
3	Кухня
4	Ванна кімната
5	Санвузол на першому рівні
6	Комора на першому рівні
7	Гараж
8	Коридор на першому рівні
9	Міжповерховий перехід
10	Спальня
11	Дитяча кімната
12	Гостьова спальня
13	Санвузол на другому рівні
14	Комора на другому рівні
15	Кабінет на другому рівні
16	Гостьова спальня
17	Коридор на другому рівні
18	Балкон

Система управління "розумний будинок" включає в себе мережу сенсорів, які контролюють різноманітні параметри, такі як температура, вологість, рівень вуглекислого газу, чадний газ, наявність диму та полум'я.

Сенсорні мережі на кожному з поверхів мають власні локальні сервери, які здійснюють отримання та обробку даних. Центральний сервер обробки даних

відповідає за зберігання отриманої інформації протягом певного часу, її аналіз та прийняття рішень в залежності від ситуацій. Крім того, він забезпечує зв'язок з мережею Інтернет та сповіщення користувачів.

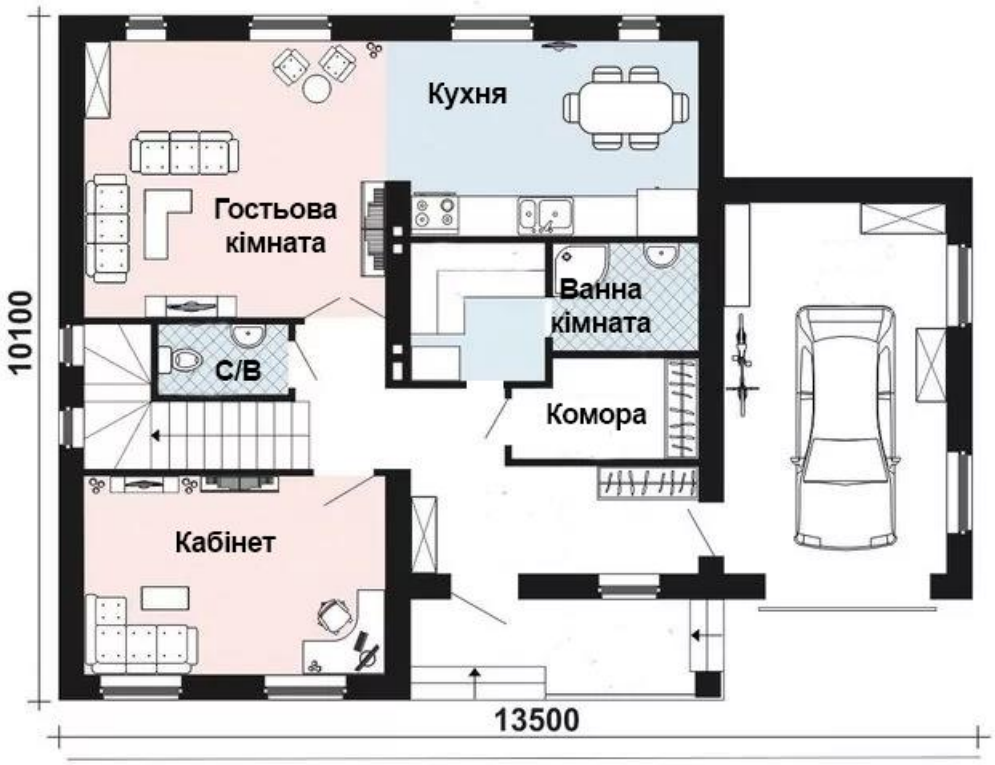


Рисунок 1.8 – Перший поверх будинку

Функціональні особливості комп’ютерної системи Комп'ютерна система - це будь-який пристрій або група взаємопов'язаних чи суміжних пристроїв, один або кілька з яких, діючи згідно з програмою, автоматизовано обробляють дані.

Інформаційна система - це методи, які використовуються для передачі, обробки або зберігання даних з метою досягнення конкретної мети.

Умовно можна класифікувати процеси, які використовують інформаційні системи різного походження [3]:

- введення нової інформації з різних джерел;

- обробка нових даних та їх трансформація в зрозумілий формат;

- оброблені дані передаються користувачам або використовуються іншими системами;

- зворотний зв'язок - це дані, які виправлялися людьми, щоб вхідні дані відповідали потрібним результатам.

Рисунок 1.9 – Другий поверх будинку

На сучасний момент можна виділити властивості сучасної інформаційної системи [4]:

- різні інформаційні системи можна аналізувати, створювати та керувати відповідно до загальних принципів створення систем;

- розвиток інформаційних систем є динамічним;

- при створенні системи слід використовувати системний підхід;

- остаточне рішення будується на вихідних даних.

1.3 Завдання

Метою даної кваліфікаційної роботи є розробка комп’ютерної системи протипожежного захисту для розумного будинку з можливістю віддаленого керування через сервер MQTT. Згідно поставленого завдання, ця комп’ютерна

система має забезпечувати ефективний моніторинг та попередження потенційних пожежних ситуацій, а також вжиття заходів для їх запобігання, у тому числі інформування власників будинку про можливу небезпеку.

Найоптимальнішим рішенням для вирішення цих завдань є використання протоколу MQTT для взаємодії між системами розумного будинку. Для центральної обробки даних вибрано одноплатний комп’ютер Raspberry Pi, який виступає як сервер для зв’язку з мережею Інтернет та забезпечує загальну функціональність системи.

Завданнями при розробці комп’ютерної системи є:

– моделювання "розумного будинку" в Packet Tracer з використанням пристроїв та датчиків протипожежної безпеки;

– розробка правил взаємодії пристроїв протипожежної безпеки з іншими пристроями та довкіллям;

– налаштування працездатності пристроїв пожежної безпеки;

– конфігурація системи моніторингу протипожежного стану через протокол MQTT;

– налаштування системи моніторингу протипожежного стану через веб-сервер;

– моделювання керування пристроями протипожежної безпеки через веб-інтерфейс та MQTT-брокер, з подальшим порівнянням часових відмінностей;

– тестування протипожежного захисту "розумного будинку".

В результаті виконання поставленого завдання буде:

1. Створена комп’ютерна мережа "розумного будинку" із забезпеченим комбінованим типом зв’язку, можливістю конфігурування через веб-інтерфейс як вдома, так і віддалено.

2. Здійснене моделювання роботи комп’ютерної системи для охоронно-пожежної сигналізації «розумного котеджного комплексу».

2 ТЕОРЕТИЧНА ЧАСТИНА

2.1 Моделі масового обслуговування для мультисервісних мереж

В кваліфікаційній роботі магістра визначено завдання з моделювання роботи комп’ютерної мережі системи протипожежного захисту «розумного котеджного комплексу» з віддаленим контролем через сервер MQTT.

Для моделювання роботи мережі слід визначити швидкодію всіх мережевих пристроїв, навантаження на які залежать від задач кінцевих користувачів, пов’язаних з обміном інформацією.

Моделювання роботи мережі здійснити з урахуванням умов наближених до максимально можливих інформаційних потоків в мережі.

Особлива увага приділена питанню наявності шкідливого вірусного програмного продукту, що працює у фоновому режимі і значно уповільнює швидкість роботи мережі перевантажуючи її так як вони виконують певну кількість дій.

2.1.1 Мережі, орієнтовані на з'єднання

Першою системою електронного зв'язку була телеграфія, яка використовувала, наприклад, азбуку Морзе (1836) для передачі інформації. Телеграфна система спиралася на передачу від точки до точки модульованих аналогічних сигналів інтенсивності та ретрансляційних станцій (вузлів) для маршрутизації та повторного посилення сигналів до тих пір, поки пункт призначення не був досягнутий. У мережах телефонії використовувався той же принцип, модулюючи несучий сигнал за інтенсивністю звучання вимовлених слів, аж до переходу на цифрову передачу. Для цифрової передачі голосовий сигнал повинен бути дискретизований, і між зразками одного зв'язку, який тепер називається викликом, можуть передаватися зразки інших викликів. Цей метод називається мультиплексуванням з часовим поділом (TDM). Таким чином, ємність

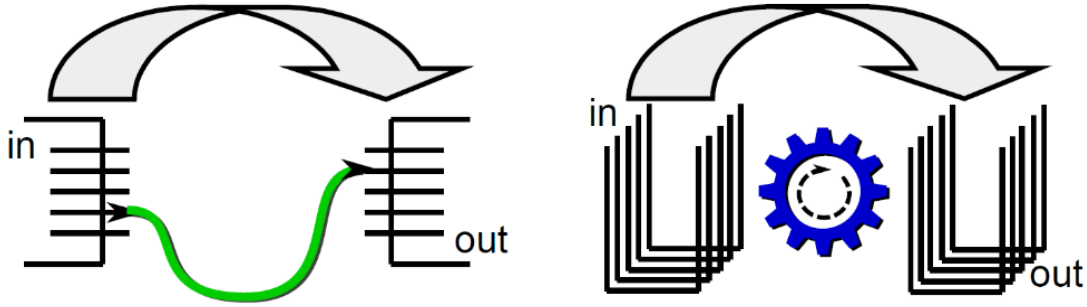
середовища розподіляється на так звані канали передачі. Однак, щоб встановити необхідні наскрізні з'єднання, комутатори тепер також повинні демультиплексувати різні канали, щоб з'єднати їх окремо. Мережі мовлення – це окрема категорія. Вони розподіляють один сигнал на всі приймачі в зоні їх дії, однаково спрацьовуючи сигнал. На відміну від телефонії, радіомовлення переважно використовує мультиплексування з частотним поділом (FDM) для передачі різних каналів по одному і тому ж середовищу. В оптиці ФДМ називається мультиплексуванням з поділом по довжині хвилі (WDM). Подумайте про вогонь, який сяє різними кольорами, і принцип повинен бути чітким.

Сучасні мережі зв'язку підтримують передачу «точка-точка», трансляцію та послуги між ними, так звані багатоадресні послуги, які з'єднують будь-яку групу терміналів. Крім того, середовище змінилося, сьогодні більшість мереж зв'язку покладаються на оптичну передачу на основі кремнеземного волокна, незалежно від технології останньої милі. У межах основних мереж, які іноді називають магістралями, для передачі інформації між фізичними місцями використовується синхронна цифрова ієрархія (SDH) або її нащадок, оптична цифрова ієрархія (OTH). OTH є частиною стандарту оптичної транспортної мережі (OTN), який також інтегрує WDM для розумного використання величезної корисної потужності волокон кремнезему в області Тбіт/с, що занадто багато для одного застосування, а також занадто багато для сучасної цифрової обробки. Для ще більшого збільшення числа мультиплексованих каналів сьогодні використовується мультиплексування з поділом коду (CDM) і комбінації схем мультиплексування для досягнення будь-якої зручної деталізації каналів.

2.1.2 Підключення мереж без підключення

Винахід інтернет-протоколу (ІП, 1974 р.) розділив світ комунікаційних мереж на фізичний і віртуальний зв'язок (рис. 2.1).

IP використовує передачу інформації, будучи з'єднанням без перенесення даних окремими пакетами. Кожен пакет незалежно комутується на кожному проміжному вузлу на основі деякої інформації, наданої самим пакетом. Ключова перевага полягає в тому, що корисне навантаження різних комунікацій, будучи потоком пакетів, кожен з яких створює, автономна діляться наявною ємністю без необхідності заздалегідь уточнювати попит. Така схема розподілу потужностей називається статистичним мультиплексуванням.



а) б)

Рисунок 2.1 - Парадигма комутації ланцюгів і пакетів

а) Комутація каналів - підключіть лінію так, як цього вимагає сигналізація;

б) Комутація пакетів - Пересилання вхідних пакетів один за одним

Парадигму комутації можна порівняти зі звичайною поштовою службою: відсортувати пакети, що надійшли в відділення, в доступні вихідні контейнери на основі інформації, зазначеної на етикетці з адресою пакета. Очевидно, що пакети повинні зберігатися в вузлах, поки обробляється інформація заголовка, мітка. Крім того, якщо в будь-який момент надходить більше пакетів, ніж можна переслати, надлишкові пакети буферизуються. Тому загальна парадигма називається комутацією «зберігай і вперед», що чітко відокремлює її від наскрізної комутації, парадигми, реалізованої традиційними мережами телефонії. Ці двоє такі ж різні, як день і ніч, кішка і собака, телефонія і поштовий зв'язок. Але вони задовольняють один і той же запит: транспортування інформації з одного місця в інше.

2.1.3 Маршрутизація пакетів

Кожен вузол комутації пакетів потребує так званої таблиці маршрутизації, щоб визначити, куди повинен бути переспрямований пакет. Він визначає вихідний порт для кожної адреси призначення. Протокол маршрутизації заповнює та підтримує ці таблиці. Зазвичай протокол маршрутизації реалізує розподіленим способом алгоритм Дейкстри для знаходження дерева найкоротших шляхів від будь-якого вузла до пункту призначення. Таблиці маршрутизації постійно підтримуються для того, щоб реагувати на зміни ресурсів, головним чином на збій або додавання ресурсів. Постійно визначених наскрізних з'єднань не існує. Особливо ні, якщо виконується балансування навантаження по паралельних траєкторіях. Ця функція організації трафіку, наприклад, забезпечується протоколом маршрутизації з відкритим найкоротшим шляхом (OSPF), який покладається на алгоритм Дейкстри k-найкоротших шляхів для пошуку до k паралельних шляхів між будь-якими двома вузлами мережі.

Сам IP не уточнює, яким чином передаються пакети, він припускає, що мережі під ним здатні виконати це завдання. Це додає суттєвої свободи, оскільки дозволяє IP-з'єднанням використовувати різні системи передачі за принципом hop-by-hop. З'єднання «точка-точка» між IP-вузлами може бути реалізовано будь-якою системою передачі. В основній мережі це традиційно комутоване з'єднання, що забезпечується мережею телефонії, але з'являються нові підходи. Використання Ethernet для з'єднання IP-вузлів додає ще один статистично-мультиплексуючий рівень. Однак мережевий рівень Ethernet зазвичай обходить статична конфігурація комутаторів: таблиці комутації та прив'язки портів встановлюються системою управління, а не автономно навчаються на основі трафіку, як це винайшов Ethernet. Такий обмежений Ethernet забезпечує стабільне з'єднання між IP-вузлами. Зауважте, що статичні посилання є обов'язковими для надійної IP-маршрутизації, і тому не можна отримати жодних переваг у продуктивності від додаткової складності, принаймні щодо наскрізної досяжної транспортної продуктивності.

2.1.4 Розподілене керування

IP - це повністю розподілена система зв'язку, яка працює до тих пір, поки два вузли якимось чином з'єднані, подібно до сигнальних пожеж. Він повністю відокремлює передачу «точка-точка» від наскрізного транспортування, як показано на рис. 2.2 У сукупності ці особливості призводять до того, що IP є дуже надійною, широко застосовною, і надзвичайно простий, в основному тому, що немає координації або централізованого управління.

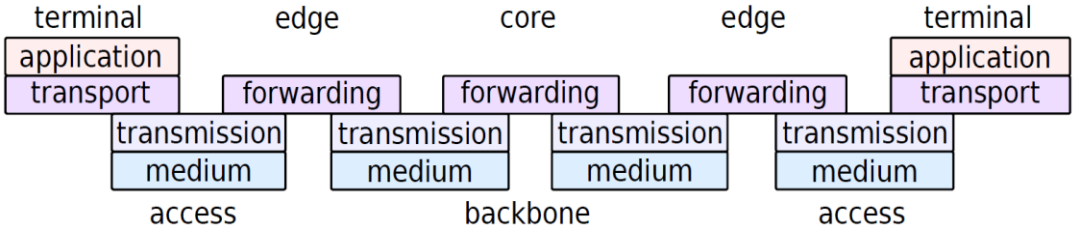


Рисунок 2.2- Віртуальне з'єднання на основі сховища та пересилання

Прості системи майже завжди витісняються менш простими, але вони можуть бути складними в операційному плані. У той час як у мережах телефонії механізм налаштування з'єднання призначає ресурси, які будуть використовуватися виключно для запитуючого з'єднання, IP-мережі призначають ресурси пакет за пакетом. Пакети жадібні, вони хочуть, щоб їх якомога швидше переслали до місця призначення. Вузли не узгоджують свої поточні рішення щодо маршрутизації з нижчестоящими вузлами. Отже, повністю порівнянний з автомобільним трафіком, час від часу виникають затори. Рідка, якщо навантаження невелике, але чим більша частота, тим більше використовуються ресурси. Важливим супутником IP, який піклується про перевантаження, є протокол контролю транспорту (TCP) [RFC675, RFC1122]. Спочатку IP і TCP нероздільні, тому що TCP був розроблений, щоб взяти на себе всі питання надійності мережі: TCP обмежує обсяг трафіку, який джерело може вставити протягом певного часового вікна, і динамічна регулює це обмеження в залежності від успіху попередніх пакетів. Крім того, TCP також гарантує, що кожен пакет буде успішно отриманий шляхом повторної відправки пакетів, які не

підтверджені одержувачем. Це призводить до величезної мінливості транспортних затримок, званої тремтінням. Тому критичні за часом сервіси не люблять TCP. Вони віддають перевагу призначеному для користувача протоколу дейтаграм (UDP) [RFC768], який дуже жадібний, тому що не має механізму підстроювання навантаження, що вставляється, під поточну продуктивність з'єднання. Були запропоновані альтернативні протоколи, які обслуговують критично важливі транспортні послуги більш дружніми до ресурсів, але, здається, рідка впроваджуються або використовуються, наприклад, Datagram Congestion Control Protocol [RFC4340]. Найбільш широко використовуваним сьогодні є Stream Control Transport Protocol [RFC3286, RFC4960], оскільки він адаптований до потокових сервісів.

2.1.5 Надання послуг

Сьогодні майже всі послуги зв'язку реалізовані через IP. Це було викликано повсюдною присутністю IP, що є результатом гнучкості використання будь-якої системи передачі. Також цьому сприяла простота його управління і безперервне збільшення швидкості сучасних процесорів. Об'єднання всіх мереж в одну економить операційні витрати, але при надмірному розтягуванні проста робоча конячка відштовхується. Різні служби викликають різні навантаження, як показано на рис. 2.3, і вимагають різних транспортних якостей.

пакетне транспортування даних додатків

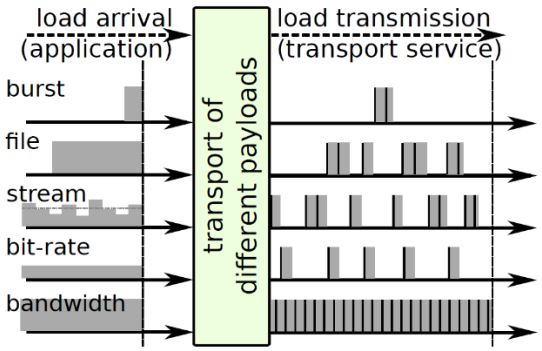


Рисунок 2.3 - Потоки пакетів, які можуть викликати сервіси при передачі по IP

Якщо парадигма комутації мереж цього не підтримує, то або всі з'єднання повинні задовольняти вимоги кожної послуги до якості, або необхідно впровадити додаткові механізми, щоб захистити критично важливі послуги від наслідків перевантажень. Останнє суперечить мережевому нейтралітету, оскільки всі пакети обробляються однаково. Це також порушує нашарування, визначене відомою моделлю взаємозв'язку відкритих систем (OSI) для цифрових систем зв'язку, якщо передача пакета вважається єдиною послугою, що надається мережевим рівнем. Це порушення можна усунути, якщо ми дозволимо параметризувати транспорт, наприклад, додавши мітку, яка посилається на політику, згідно з якою пакет повинен оброблятися на проміжних вузлах мережі, як показано на рис 2.4. Впровадження сервісного рівня дозволяє відокремити послуги від їх технічної реалізації. Цей наступний важливий крок в абстракції мережевої роботи стандартизований як так звана мережева архітектура наступного покоління (NGN). Він постулює, що додаток характеризує послугу, яку він вимагає.

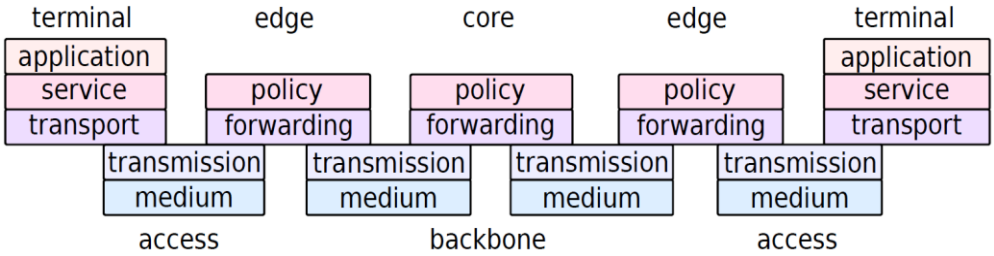


Рисунок 2.4 Віртуальне з'єднання з використанням переадресації на основі політик

Ці послуги та їх особливості складають чітко окреслений прошарок послуг. Для кожної послуги мережа автономна вибирає засоби для її транспортування, надаючи необхідні ресурси та механізми, які складають так званий транспортний шар.

2.1.6 Наскрізна якість обслуговування

Незалежно від того, як визначається та реалізується послуга, продуктивність наскрізного з'єднання залежить від поточного стану ресурсів, які встановлюють

з'єднання. Для з'єднань з комутацією пакетів основною причиною недостатньої продуктивності є тимчасове перевантаження. Повністю непрацююча мережа запропонує ідеальну якість обслуговування. На практиці перевантаження час від часу погіршують продуктивність і при зміні вузлів, якщо мережа добре спроектована і налаштована, що є основними завданнями мережевої інженерії. Однак навіть тимчасові проблеми на шляхах впливають на середню продуктивність, яка визначає, чи є з'єднання корисним для сервісу чи ні. Незважаючи на те, що всі спірні питання та процедури ефективні лише за наявності заторів, вони визначають якість транспортування, яка називається якістю обслуговування (QoS). QoS, що забезпечується ланцюжком досить незалежних мережевих компонентів, які постійно змінюють свій стан, визначити непросто. Рис. 2.5 має намір виразити це математично.

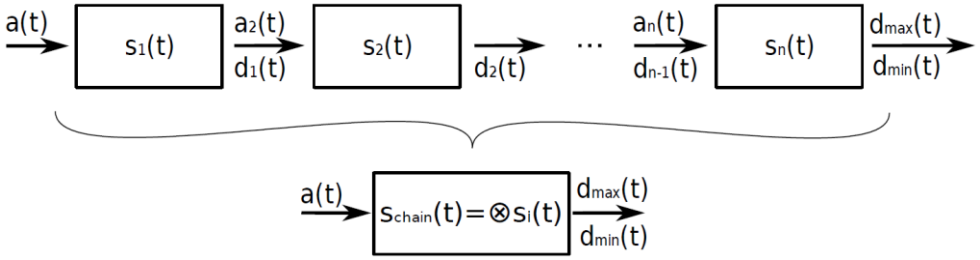


Рисунок 2.5 - Якість з'єднання є каскадом продуктивності компонентів

Прибуття трафіку моделюється функціями прибуття ai(t), відхилення від компонентів функціями відправлення di(t). Зауважимо, що не всі вильоти компонента попереду di−1(t) потребують входу в наступну компоненту, і навпаки, що також трафік по зв'язках, які не є повністю частиною цього ланцюга, може проходити повз задіяні компоненти, додаючи локально трафік ai, j (t). Таким чином, di−1(t) матиме певний вплив на ai(t), але не визначає його повністю: ai(t) = f (di−1(t), ai, j(t)). Вплив компонента на трафік, що проходить його, представлений сервісною функцією si(t), а глобальна сервісна функція з'єднання - schain(t). Зауважимо, що залежність від часу сервісних функцій випливає з їх залежності від навантаження si(t) = f (ai(t)), і що оператор '⊗', який пов'язує

глобальну сервісну функцію з ланцюговими сервісними функціями, не ідентифікує визначену операцію, він є лише заповнювачем.

Методи виведення або апроксимації продуктивності з'єднання schain(t) з продуктивності ланцюгових компонентів si(t) представлені в розділі 4.3.4. А поки що перерахуємо лише основні відносини:

1. Збільшення продуктивності компонента не зменшить QoS з'єднання, що проходить через нього; а підвищення продуктивності всіх компонентів поліпшить QoS всіх з'єднань.

2. Привілеювання пакетів одного з'єднання, ймовірно, погіршує продуктивність усіх з'єднань, які ділять ресурс із привілейованим.

3. Якщо всі з'єднання є однаково привілейованими, вони досягають однакової продуктивності, оскільки без надання пріоритету будь-яким ⇒ середню продуктивність не можна покращити за допомогою привілеїв.

2.2 Мультипротокольна комутація міток

Ґрунтуючись на обмеженнях простої IP-адреси мультипротокольна комутація міток (Multi Protocol Label Switching - MPLS) додає до IP, і окреслює механізми, необхідні для орієнтованого на продуктивність управління навантаженням і диференціації послуг. Сама по собі MPLS не пропонує надання QoS. У попередньому розділі ми розглянули, як продуктивність з'єднання залежить від компонентів, переданих потоком пакетів, що належать службі. Ми відзначили, що продуктивність визначається тимчасовими перевантаженнями, які виникають частіше, якщо ресурси використовуються більше. Зв'язок між ними є прогресивним, і тому середня продуктивність мережі може бути покращена, якщо зменшити пікові навантаження. Це основне завдання дорожньої інженерії. Коротко це означає: помістіть трафік туди, де є ресурси, і ви отримаєте найкращу продуктивність мережі.

Маршрутизація найкоротшого шляху, будучи схемою маршрутизації IP за замовчуванням, погано підтримує балансування навантаження або інші спроби організації трафіку. Як показано на рис. 2.6 потоки трафіку можуть концентруватися на посиланнях, залишаючи інші недостатньо використаними.

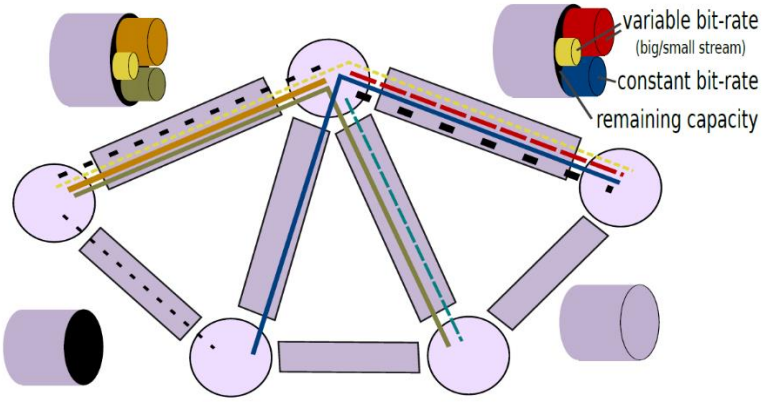


Рисунок 2.6 - IP-мережа з мінімальною маршрутизацією стрибків за замовчуванням

Щоб підвищити продуктивність IP-мережі, ми можемо налаштувати ресурси або зробити ручне переналаштування. Перший підхід дотримується емпіричного правила: подвоїти потужність, коли завантаження досягає критичного рівня, наприклад, 50%. Ця схема коригування реалізує адаптивну односторонню оптимізацію – довгострокову еволюцію мереж, зумовлену попитом. Інший метод намагається перенаправити навантаження, регулюючи вагу зв'язку, що використовується маршрутизацією (мінімальні витрати), як показано на рис. 2.7.

Розрахунок ваги посилань, що забезпечують найкращий розподіл, вимагає хороших знань про потоки трафіку, а не про навантаження на канали, і зазвичай виконується в автономному режимі.

Крім того, після того, як потоки до пункту призначення об'єднані, вони більше не можуть бути розділені на різні шляхи.

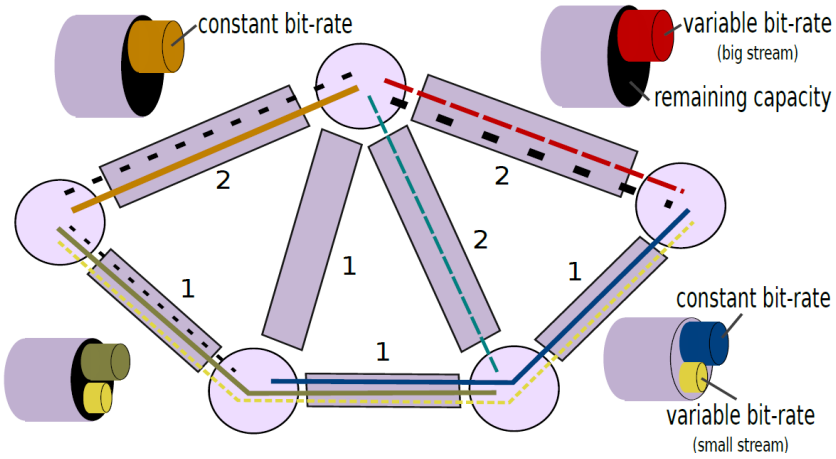


Рисунок 2.7 - IP-мережа з мінімальними витратами маршрутизації

Маршрутизація IP без з'єднання може призначити лише один вихідний порт на адресу призначення. Маршрутизація з відкритим найкоротшим шляхом (OSPF) пропонує в розширенні інженерії дорожнього руху (TE), серед іншого, можливість розподілу навантаження на паралельні ділянки. Зазвичай він виконує це на агрегованому трафіку, і це може призвести до того, що прийом пакетів, що вийшли з ладу, можливо, непридатний для деяких служб.

Обидва мережеві інженерні рішення мають побічні ефекти в масштабах всієї мережі і лише опосередковано досягають мети інженерії трафіку. Зауважимо, що збалансоване використання також може бути досягнуто шляхом маршрутизації трафіку через надмірні об'їзди. Це не може бути рішенням. Ми завжди повинні намагатися мінімізувати пікове використання та загальне використання ресурсів спільно. З точки зору зв'язків, ми хочемо знайти баланс між ефективністю довшого шляху, що складається з низьконавантажених стрибків, і більш короткого шляху, що містить більш високонавантажені переходи.

2.2.1 Потоки даних, орієнтовані на з'єднання

Щоб забезпечити управління навантаженням на основі кожного потоку та перенести перевірені схеми балансування навантаження з комутації ланцюгів, трафік повинен слідувати наскрізна визначеним маршрутам. Це основна функція,

яку забезпечує мультипротокольна комутація міток. Він відокремлює транспортування корисного вантажу від маршрутизації і робить маршрутизацію змінним завданням площини управління.

Назва MPLS складається з двох частин: перемикання міток відноситься до давно відомої схеми заміни міток, яка масштабується реалізує переадресаційні відносини, і мультипротокольний термін, який виражає, що різні протоколи маршрутизації можуть використовуватися незалежно, взаємозаміно і одночасно для визначення маршруту шляху перемикання міток(ЛСП). ЛСП встановлюються і розбиваються за допомогою сигналізації, порівнянної з комутацією ланцюгів. Важливою відмінністю від інших реалізацій заміни міток є: MPLS дозволяє складати мітки, тобто використовувати наявні LSP подібні посилання та тунелювати нові LSP всередині існуючих.

Заміна міток виконується для того, щоб розмір міток залишався маленьким і в той же час дозволяв величезну кількість LSP. Останнє досягається завдяки тому, що мітки можуть вільно використовуватися хміль за хмелем. Перемикання міток виконується на основі кортежу (port, label): для кожного кортежу перемикання надає вихід-кортеж, який визначає вихідний порт і мітку, яка буде використовуватися при наступному переході, як показано на рис. 2.8.

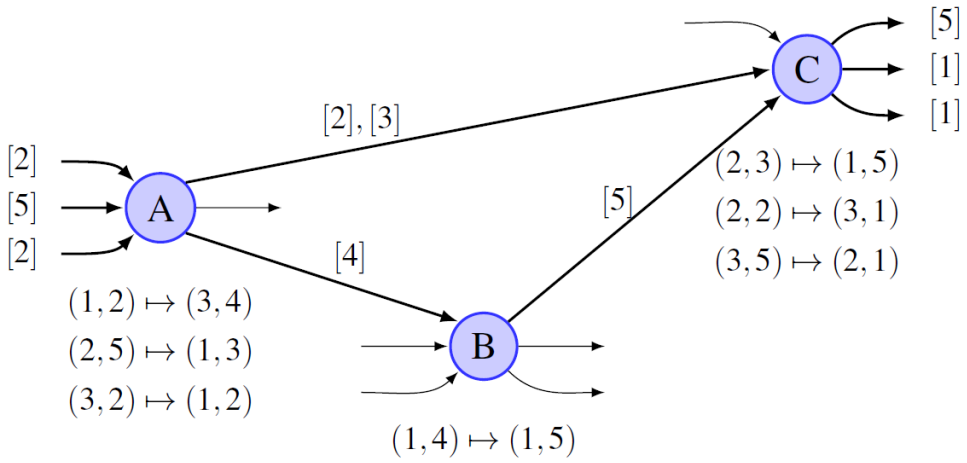


Рисунок 2.8 - Схема заміни міток

Цей механізм працює так само добре, як і мітки для LSP, але він зменшує кількість бітів, необхідних у заголовку пакета, і робить схему масштабованою до будь-якого розміру мережі. Заміна мітки не впливає на продуктивність LSP, і тому ми припускаємо, що це працює і використовується у наступних LSP-ID для ідентифікації різних LSP, а не для завжди мінливих реальних міток.

Для повноти картини: вузли, що виконують перемикання міток, називаються маршрутизатором комутації міток (LSR), а крайові вузли, які призначають навантаження LSP, називаються маршрутизатором з міткою edge (LER) – це найменування використовується в IETF RFC3031, що визначає MPLS [5]. Для послідовності та зручності читання ми дотримуємося загального терміну вузол і припускаємо, що вузли налаштовані на виконання всього, що вимагається на їхньому місці.

2.2.2 Розподіл навантаження на трафік

Управління навантаженням, що означає рекламу очікуваного навантаження за допомогою сигналізації LSP, відстеження навантаження, зайнятого в даний момент, на ресурси та резервування часток потужності для LSP, не є частиною MPLS. Однак для маршрутизації LSP, які відповідають певним вимогам QoS, це важливо. Наприклад, OSPF-TE використовує цю інформацію для маршрутизації LSP. Тому ми припускаємо, що для механізму встановлення та демонтажу з'єднання на основі сигналізації не є проблемою реалізувати рекламу навантаження. Широко використовуваним протоколом сигналізації, що забезпечує необхідні засоби, є, наприклад, RSVP-TE [7]. Таким чином, ми припускаємо, що очікуване навантаження на LSP, особливо його середній обсяг і розподіл, приблизно відомі на кожному вузлу мережі, який проходить LSP, незалежно від використовуваного протоколу сигналізації. Сформувавши цю інформацію, ми розрахуємо потенціал завантаженості на вихідну ланку. Зауважимо, що непризначений IP-трафік або

потрібно добре вгадати, або технічно запобігти впливу на продуктивність LSP, щоб уможливити надійне прогнозування продуктивності.

Там, де це необхідно, ми припускаємо, на додаток до управління навантаженням, таблицю QoS для кожного вузла, яка повідомляє для кожного кортежу введення, політику, яку слід застосувати до пакетів, що передаються у відповідному LSP. Таким чином, ми додаємо диференційований QoS до MPLS. Знову ж таки, оскільки технічно сигналізація не є проблемою, ми припускаємо, що вона доступна відповідно до встановлених варіантів диференціації послуг. Заголовок MPLS вже надає три біти для ідентифікації класу еквівалентності пересилання (FEC), порівнянного з тим, що ми знаємо з асинхронного транспортного режиму (ATM) і рамкового реле (FR), для підтримки диференціації послуг між LSP.

На рис. 2.9 показаний невеликий приклад мережі, який ілюструє, як можна спільно використовувати ресурси зв'язку за допомогою MPLS в поєднанні з управлінням ресурсами. Якщо частки ємності строго зарезервовані для LSP, ми отримуємо віртуальне з'єднання (VC), оскільки наявність необхідної частки потужності (bLSPi) гарантується йому на кожному ресурсі на його шляху.

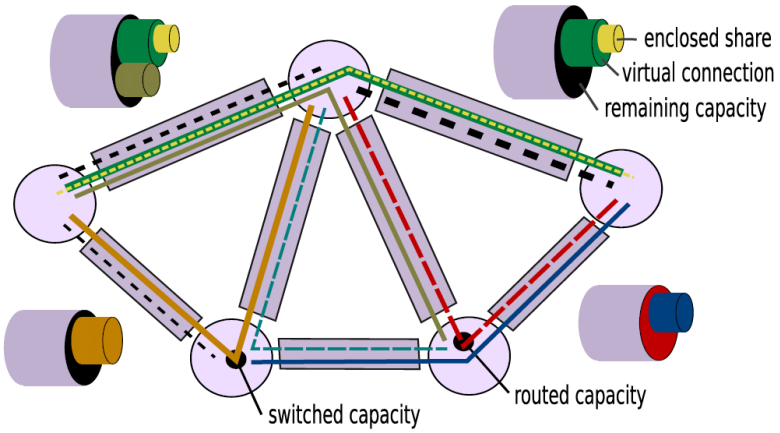


Рисунок 2.9 - Багатопротокольна мережа з комутацією міток (MPLS)

Віртуальним з'єднанням не потрібно конкурувати за ресурси, якщо перезаказ суворо заборонений і жодне джерело не може вставити більше заявленого завантаження bLSPi в будь-який час.

$$\sum_i b_{\mathrm{LSP}_i \exists_{\mathrm{link}}} \leq C_{\mathrm{link}}, \forall_{\mathrm{links}} \tag{2.1}$$

Більш поширеними є LSP, які реалізують віртуальні шляхи. Віртуальний шлях (VP) не має частки потужності, зарезервованої для нього. Пакети, що протікають уздовж VP, поділяють доступні на даний момент ресурси hop за стрибком. Доступ до ресурсів може контролюватися локально за допомогою механізмів спільного використання ресурсів, які можуть реалізовувати різні політики, які називаються поведінкою за переходом (PHB) у номенклатурі Diff-Serv. Відмінність IP-трафіку, не укладеного в LSP, полягає в тому, що всі пакети, призначені віртуальному з'єднанню, орієнтуються вздовж маршруту LSP. Маршрути LSP можуть відхилятися від найкоротшого шляху, яким слідують непризначені IP-пакети, і LSP до того самого пункту призначення можуть тунелювати призначений трафік різними паралельними маршрутами до місця призначення.

2.2.3 Розподіл міток

Цікавим аспектом MPLS є, принаймні, якщо використовується протокол розподілу міток (LDP) [9], що LSP налаштовуються від пункту призначення до джерела. Отже, маршрути LSP можуть бути легко змінені під час експлуатації. Транспорт перемикається на ділянки нового шляху не раніше, ніж буде готова вся ділянка до пункту призначення, як показано на рис. 2.10.

Етапи налаштування LSP:

- тримати вектор шляху для нового з'єднання;
- запустити зворотне налаштування рекурсивної таблиці;
- видалити застарілі записи таблиці маршрутизації.

Етапи налаштування таблиці маршрутизації LDP:

- отримати вихідний кортеж (J, B) і вхідний порт I;
- вибрати мітку A, доступну на порту I;

- встановити запис таблиці маршрутизації (i, a) 1→ (j, b);

- надіслати A на вузол, підключений на порту I.

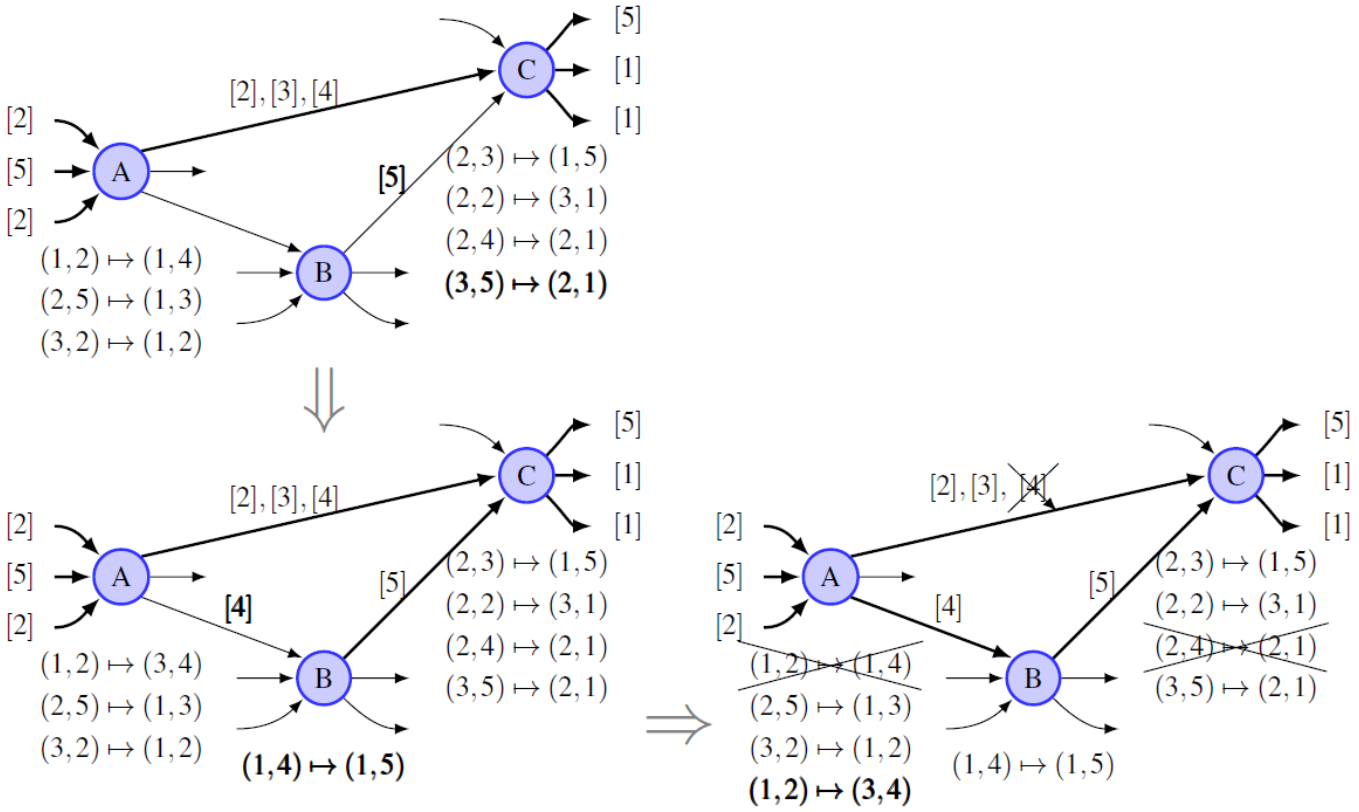


Рисунок 2.10 - Схема розподілу міток

Щоб налаштувати новий шлях або розділ, спочатку потрібно викликати протокол маршрутизації. У разі успіху маршрутизація повертає вектор шляху, який містить упорядкований список вузлів, що однозначно визначає знайдений маршрут. Починаючи з вузла призначення, LDP обходить вузол за вузлом до джерела нового маршруту і створює нові записи таблиці маршрутизації на кожному пройденому вузлу. Зверніть увагу, що таблиці маршрутизації не повинні містити два записи для одного і того ж кортежу. Власне, якщо це станеться, ми дістанемося до джерела і негайно ініціюємо демонтаж заміненого LSP, щоб видалити застарілі записи таблиці.

Зауважте, що декілька записів з однаковим кортежем не є проблемою, вони лише вказують на те, що два LSP зливаються у цьому вузлу. Це має відбуватися під час налаштування заміни, щоб забезпечити плавне перемикання (make-before-

break), і може бути використано для визначення вхідних дерев, які з'єднують багато джерел з одним пунктом призначення (наприклад, спільним висхідним каналом). Робота від пункту призначення до джерела дозволяє відновлювати секційні траєкторії, а безшовність дозволяє динамічна перерозподіляти навантаження (оптимізація в процесі експлуатації). Обидва по суті необхідні для автономних варіантів управління мережею, що самоорганізуються, включаючи стратегії захисту та постійну оптимізацію в динамічному режимі роботи.

2.2.4 MPLS - приклад будь-якої багатопотокової мережі

MPLS забезпечує плановий розподіл навантаження та підтримує різну поведінку для окремих потоків. Обидва представляють потенціал для підвищення наскрізної продуктивності – один у глобальному масштабі, інший локально на вузлових майданчиках. Це відкриває безліч можливостей і викликає деякі невизначеності при його аналізі. Сьогодні MPLS широко використовується і на практиці довів, що може надавати функції, необхідні для ефективної роботи мультисервісних інфраструктур зв'язку. Таким чином, було б марно вивчати точні протоколи, на які спирається MPLS. Проаналізувати всі можливі комбінації (варіанти реалізації) на предмет їх потенціалу QoS-доставки - це окрема історія, але надмірна. Ми вирішили залишити в стороні всі деталі реалізації, і, таким чином, пропустити аналіз протоколів, що використовуються MPLS для реалізації його особливостей, і, відповідно, будь-які більш детальні презентації та обговорення. Зацікавлений читач люб'язно звертається до IETF RFC, що визначають деталі MPLS, розширення тощо, а також до багатьох досліджень з різних аспектів MPLS, опублікованих за останнє десятиліття.

Ми дивимося на продуктивність, яку пропонує MPLS, лише з боку сервісу, і розглядаємо принципову функціональність, а не фактичну реалізацію. З одного боку, це призводить до того, що польові експерименти не покажуть точно таких же результатів. З іншого боку, це рішення робить результати в більш широкому сенсі

корисними, оскільки MPLS називають лише прикладом технології. Функціональні можливості, необхідні для того, щоб наші дослідження застосовувалися, включають:

- перемикання між магазинами та вперед;
- транспортування по заздалегідь прокладеним шляхам;
- об’ява очікуваного навантаження;
- диференційоване по варіантам обробки навантаження у вузлах.

Останнім часом ці принципи були реалізовані аналогічним чином деякими розширеннями Ethernet, і через деякий час може з'явитися революційна схема роботи мережі, яка може реалізувати комутацію ще нижчих рівнів, наприклад, оптичне перемикання імпульсів (OBS) і режим передачі потоку (FTM).

2.3 Cisco-модель комп’ютерної мережі протипожежного захисту «розумного котеджного комплексу»

У бакалаврській кваліфікаційній роботі була розроблена комп’ютерна мережа протипожежного захисту «розумного будинку» з віддаленим контролем через сервер MQTT для «розумного котеджного комплексу». Її структура має дворівневу ієрархічну архітектуру – на верхньому рівні розташоване ядро мережі, яке складається з маршрутизаторів, а на нижньому рівні через комутатори організовано розподіл мережі для кінцевих користувачів – для робочих груп (рис. 2.11).

На верхньому рівні - рівні ядра мережі комп’ютерної системи протипожежного захисту розумного будинку використовуються мережеві маршрутизатори фірми Cisco, а на нижньому рівні реалізація здійснена на мережевих комутаторах теж фірми Cisco.

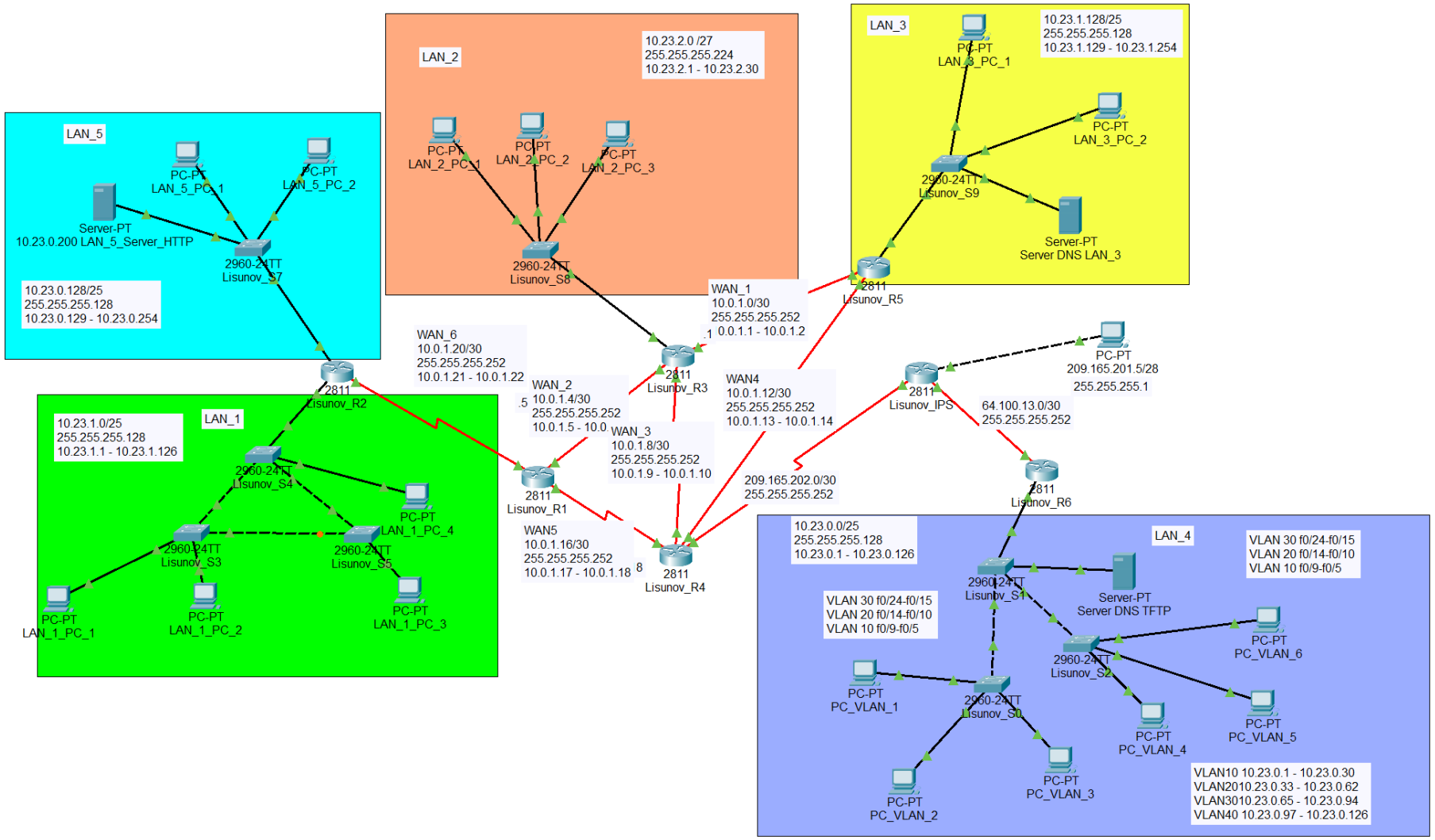


Рисунок 2.11 – Комп’ютерна мережа системи протипожежного захисту «розумного будинку» з віддаленим контролем через сервер MQTT для «розумного котеджного комплексу»

На верхньому рівні мережі комп’ютерної системи протипожежного захисту розташовані сім маршрутизаторів, які мають наступні параметри:

- використано адресний простір 10.23.16.0/22;

- кількість ПК у кожній локальній мережі: LAN1 - 22, LAN2 -12, LAN3 - 28, LAN4 -33, LAN5 - 26;

- VLAN організовано згідно табл. 2.1.

Таблиця 2.1 – VLAN

Номер VLAN	**Ім’я VLAN**	**Примітка**
10	Accounting	Бухгалтерія
20	Resources Department	Відділ кадрів
30	Guest	Гості
40	Devices	Пристрої

Топологія поєднує в собі 5 мереж з кінцевими користувачами, 7 мережевих маршрутизаторів, один зовнішній шлюз з мережевим блоком адрес 10.0.1.0/24. Маршрутизатор та зовнішні шлюзові мережі вимагають по дві IP-адреси.

Всі сегменти середовища IP-підмереж розподілені на п’ять умовних підмереж за допомогою маршрутизаторів з використанням адресація четвертої версії. Завдяки застосованою NAT-технології здійснюється вихід мережі до Internet.

Динамічної маршрутизація організована з використанням OSPF-протоколу, що забезпечує дуже ефективну маршрутизацію для мережевих пристроїв.

Для забезпечення маршрутизації між мережевими пристроями VLAN на комутаторі Lisunov_R6 застосована технологія інкапсуляції 802.1Q.

За допомогою виділеного блоку адрес організована адресація каналів між маршрутизаторами. Використання адресації з протоколом DHCP здійснено для кінцевих мережевих пристроїв VLAN.

Для комп’ютерної мережі системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT використовується топологія «ієрархічної зірки», яка ефективно працює з базовими Ethernet-технологіями.

Робочі групи охоплені рівнем доступу з використанням Fast Ethernet-технології. Fast Ethernet-технології також використовується між маршрутизаторами і комутаторами.

3 СИНТЕЗ СИСТЕМИ

3.1 Технічних вимоги до комп’ютерної системи

3.1.1 Вимоги до системи в цілому

3.1.1.1 Структура і функціонування системи

Система, що встановлена в будинку, повинна виконувати наступні функції:

- збір інформації:

- неперервний збір даних з усіх вбудованих датчиків в будівлі;

- передача цих даних для подальшої обробки та аналізу;

- обробку та аналіз отриманих даних, після отримання всіх даних система проводить їх аналіз, порівнюючи з нормальними показниками;

- збереження отриманих даних;

- зберігання отриманих даних для створення звітів і резервного копіювання на серверах;

- відповідати на будь-які зміни показників відповідно до встановлених стандартів;

- в разі виходу за межі встановлених параметрів або виявлення аномалій має вживати автоматичних заходів безпеки.

- сповіщення користувача або мешканців будинку про виниклу ситуацію.

3.1.1.2 Вимоги до показники призначення

Система повинна забезпечувати безперебійну роботу всіх функцій відповідно до умов технологічного процесу, включаючи:

- захист від несанкціонованого доступу, перешкод та збоїв, це може бути досягнуто за допомогою таких заходів, як шифрування даних, аутентифікація користувачів та використання відмовостійкої архітектури;.

- віддалений доступ до системи з будь-якого місця, де є доступ до Інтернету - це може бути корисно для адміністраторів системи, а також для користувачів, які хочуть управляти системою з дому або з іншого віддаленого місця;

- постійну роботу системи, навіть у разі відключення електроенергії або інших непередбачених ситуацій - це може бути досягнуто за допомогою резервного копіювання даних, використання безперебійних джерел живлення та інших заходів.

- збереження даних в «хмарі», що забезпечує доступ до даних з будь-якого місця та підвищує надійність зберігання.

3.1.1.3 Вимоги до експлуатації

Завдяки повній автоматизації та використанню системи протипожежного захисту в «розумному будинку» лише кваліфіковані інженери можуть обслуговувати систему. Щороку необхідно проводити технічне обслуговування системи, щоб уникнути негативних наслідків.

Лише фахівцям дозволяється виконувати ремонт системи або замінювати компоненти в системах, де не можна використовувати нові компоненти. Датчик вуглекислого газу MH-Z19B, датчик диму MQ-2, датчик температури та вологості DHT21/AM2301A та датчик чадного газу MQ-7 необхідно замінювати кожні вісім років. Заміна інших компонентів має бути завершена протягом десяти років.

3.1.1.4 Вимоги до патентної чистоти

У цій системі необхідно використовувати тільки обладнання або програми з ліцензіями та сертифікатами, що дозволяють використання на території України.

3.1.1.5 Додаткові вимоги

Тип кабелю повинен відповідати проектним розрахункам. Це може бути екранована вита пара або оптоволоконний кабель. Використовуйте кабель UTP категорії 5. Необхідно встановити інформаційні розетки.

У аварійній ситуації вихід з ладу окремих датчиків або датчиків не повинен впливати на роботу системи протипожежного захисту. Крім того, в таких системах потрібне резервне джерело живлення, щоб система могла продовжувати роботу під час відключення або припинення живлення до відновлення основного джерела енергії [9]. При проектуванні системи необхідно враховувати всі можливі варіанти

загроз і визначити, на які дані (адміністрування системи, конфіденційність) спрямовані ці загрози.

Класифікація джерела загроз, спирається на джерела загрози і можна виділити наступні типи загроз:

- антропогенні загрози - це загрози, які створюються людиною, наприклад, злочинцями, хакерами або обслуговуючим персоналом;

техногенні загрози - це загрози, які виникають внаслідок недосконалості технічних засобів, наприклад, засобів зв'язку, комунікаційних мереж, технічних засобів обробки інформації або програмних засобів;

- стихійні загрози - це загрози, які виникають внаслідок природних явищ, наприклад, пожеж, землетрусів, повеней або ураганів.

2. Вразливості безпеки, класифікуються в залежності від причин виникнення вразливості безпеки можна виділити наступні типи вразливостей:

- об'єктивні вразливості - це вразливості, які виникають внаслідок особливостей об'єкта, що захищається, наприклад, технічних засобів, програмного забезпечення або організаційної структури;

- суб'єктивні вразливості - це вразливості, які виникають внаслідок помилок або порушень, наприклад, помилок в програмному забезпеченні або несанкціонованого доступу до інформації;

- випадкові вразливості - це вразливості, які виникають внаслідок збоїв, відмов або пошкоджень технічних засобів;

- для захисту від зовнішніх електромагнітних полів треба використовувати екрановану виту пару, а щоб запобігти перепадам напруги в мережі, на вихідних кабелях необхідно встановити перемикачі;

- для забезпечення нормальних кліматичних умов серверні приміщення повинні вентилюватися, який має бути підключений до заземлення.

3.1.2 Вимоги до функцій які виконує система

Вся система має працювати цілодобово, а крім цього постійно працюють різні служби порятунку. Коли джерело відключено, система повинна переключитися на інше джерело. Усі датчики, сигналізація та вся система повинні

спілкуватися один з одним. Крім того, система також повинна мати можливість розширення. Крім того, система постійно збирає дані, аналізує дані, обробляє дані та зберігає їх на сервері.

3.1.3 Вимоги до видів забезпечення

3.1.3.1 Вимоги до інформаційного забезпечення системи

Система інтелектуального будинку є об'єктом інформаційного характеру, що вразливий перед різноманітними інформаційними загрозами. Захист даних в цій системі не може бути реалізованим єдиним способом, оскільки кожна загроза залежить від конкретного способу розробки системи та використаних технологій.

Більшість існуючих систем вже включають в себе вбудовані компоненти інформаційної безпеки, але є й такі, які досі не використовують ці можливості. При створенні комп'ютерної системи важливо враховувати ключові фактори, такі як:

- можливість шифрування даних та забезпечення захищеного зв'язку між різними датчиками системи;
- здатність до дистанційного контролю та внесення змін;
- висока ефективність захисту віддаленого доступу.

На сьогоднішній день не існує єдиного методу забезпечення інформаційної безпеки в інтелектуальному будинку, оскільки різні системи використовують різноманітні структури захисту.

Найпоширеніші загрози включають атаки на центральний сервер від так званих «хакерів», шкідливі віруси, які можуть завдати шкоди функціональності системи, крадіжку даних через канали зв'язку, несанкціонований доступ до мережі, присутність порушників серед обслуговуючого персоналу, помилки користувача, крадіжку, перебої у електропостачанні, стихійні лиха, поломки апаратних компонентів системи та помилки програмного забезпечення.

3.1.3.2 Вимоги до лінгвістичного забезпечення

Для користувачів система повинна бути тільки українською мовою.

3.1.3.3 Вимоги до системи енергозабезпечення

Надійність, як одна з ключових вимог до систем енергопостачання, визначається кількістю незалежних джерел живлення та структурою електропостачання. З погляду надійності електричного постачання, відповідно до вимог правил влаштування електроустановок та забезпечення безпеки робіт для електротехнічних пристроїв, систему можна розділити на:

- надійність електричного постачання;
- якість електроенергії;
- можливість модифікації та розвитку для розумного будинку;
- ефективне використання ресурсів.

Ці вимоги враховуються на етапі початкового проектування та протягом експлуатації «розумних джерел енергії». Система, яка є складовою енергетичних систем та виконує енергетичні завдання, є як простою, так і складною в аспекті використання та перетворення електричної енергії для технологічних цілей забезпечення електричної енергії. Електроприймачі, як невід'ємні елементи системи, визначають якість роботи цієї системи та її параметри.

При розробці споживачів електричної енергії основні параметри, за якими систематизують, включають надійність електропостачання, режими роботи, потужність, напругу та тип струму.

3.1.3.4 Вимоги до схоронності інформації при аваріях

Аварійні ситуації можуть виникнути з різних причин. З метою збереження інформації в умовах аварійної ситуації систему налаштовують так, щоб дані автоматично зберігалися на «хмарному диску», гарантуючи їхню надійність та невтратність.

3.2 Вимоги до функцій, виконуваних системою

3.2.1 Перелік функцій, задач комплексів

Архітектура системи «розумного дому» охоплює три рівні: нижній, середній та верхній.

На нижньому рівні розташовані різноманітні датчики, такі як температурні та вологості, які відіграють ключову роль в зборі інформації.

Середній рівень включає контролер Arduino Mega, який функціонує як центральний збиральник даних, об'єднуючи і обробляючи інформацію від усіх датчиків.

На верхньому рівні розташована система SCADA, яка виконує роль керування "розумним домом". Інтересною особливістю є те, що розроблена система базується на контролері Arduino Mega, що робить її більш доступною і економічно здешевлює її для споживачів середнього рівня. Це актуально у контексті того, що існуючі SCADA-системи для "розумних домів" зазвичай використовують промислові контролери, що призводить до високих вартостей.

3.2.2 Перелік функціональних підсистем

У складі комп'ютерної системи функціонують різні функціональні системи, які використовують різне програмне забезпечення. Зокрема, це системне, захисне, прикладне та спеціалізоване ПЗ.

До системного програмного забезпечення входять такі компоненти як MQTT 5.0 Broker, операційні системи Windows 10, Android і iOS.

Функції захисту виконує антивірус Avast Antivirus, що відноситься до захисного програмного забезпечення.

Прикладне програмне забезпечення включає в себе веб-браузери, такі як Google Chrome та Opera, і призначене для вирішення різноманітних завдань.

До спеціалізованого програмного забезпечення належить MQTT 5.0 Desktop Client, яке призначене для конкретних функцій та взаємодії з MQTT 5.0 протокол.

3.2.3 Вимоги до регламенту і якості реалізації функцій

Якщо відокремлений об’єкт не зберігає інформацію до моменту, коли користувач або процес отримають необхідні права доступу та поки ці права доступу не будуть анульовані, комп’ютерна система повинна надати послугу повторного використання об’єкта. Впровадження цієї послуги сприяє захисту від атак типу «збір сміття».

Служба обміну конфіденційністю захищає об’єкти від несанкціонованого доступу до інформації, яка міститься в них, під час експорту/імпорту через незахищене середовище.

Точки відновлення – це універсальна послуга, яка дозволяє відновлювати роботу після помилок користувача, програмних або апаратних збоїв, а також забезпечує цілісність баз даних, програм на основі транзакцій тощо.

Реєстрація дозволяє контролювати небезпечну діяльність у комп’ютерній системі. Рівень цієї послуги залежить від цілісності та складності методів аналізу даних журналу та можливостей виявлення загроз.

3.3 Вимоги до видів забезпечення

3.3.1 Інформаційне забезпечення системи

Користувач отримує лише готовий результат свого запиту, а основний процес обробки інформації відбувається через сервер. Сервер даних використовує хмарний сервіс для обробки та зберігання інформації. Найвигіднішим вибором для використання є рішення від компанії Microsoft - Azure IoT.

3.3.2 Передача даних між системними компонентами.

Для передачі даних між компонентами через мережу рекомендується використовувати PHY - LAN PHY і WAN PHY, за стандартом IEEE 802.3a.

3.3.3 Технічне забезпечення системи

Технічні параметри точки доступу включають забезпечення стійкої роботи мережі WiFi. Рекомендована робоча частота 5 GHz сприятиме уникненню перешкод від іншого обладнання, а також зменшить завантаженість мережі від інших пристроїв та мереж.

3.3.4 Вимоги до організаційного забезпечення

Чітке управління правами доступу користувачів до програмного забезпечення та технічних даних повинно включати розподіл прав доступу до

робочого місця, реєстрацію користувачів, які увійшли в систему, та видалення вірусів.

Отримання доступу до даних функціонального блоку має ґрунтуватися на матриці доступу та надавати можливість змінювати дані під час роботи.

3.3.5 Вимоги до складу нормативно-технічної документації системи

Елементи, які мають бути включені:

- технічні креслення, що відповідають установленим нормативам;
- маркувальні позначки для розеток, датчиків та кабелів;
- схема підключення кабельної проводки;
- список з'єднань між кабелями;
- плани розташування обладнання в шафах або стійках;
- програма та методика проведення випробувань.

3.4 Вибір і обґрунтування структурної схеми комплексу технічних засобів комп'ютерної системи

Структура системи розумного будинку визначається функціональними одиницями, які включає система і зв'язками між ними.

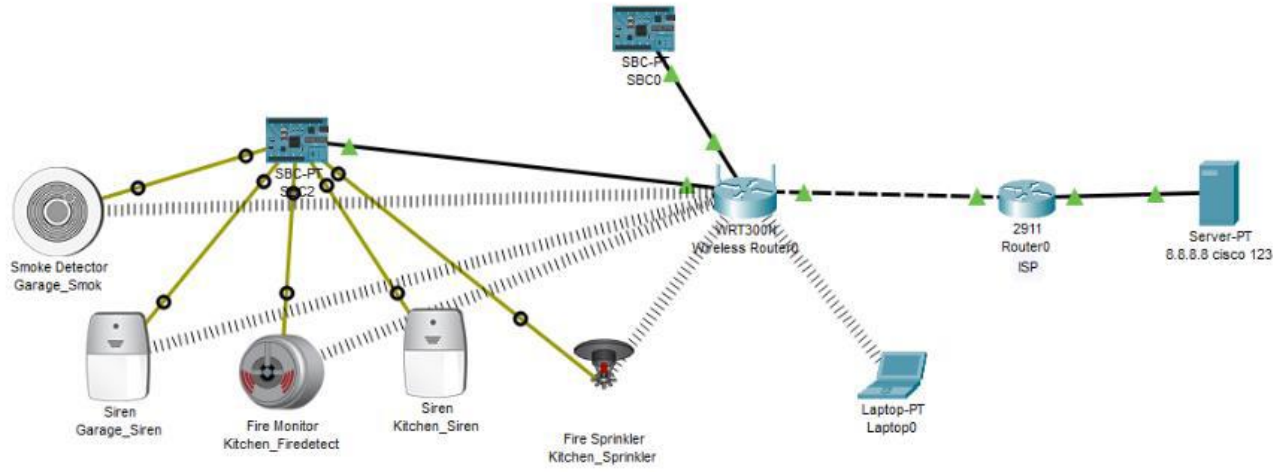


Рисунок 3.1 – Структура схеми системи розумного будинку

Для оптимального функціонування системи, лінійний блок має виступати як первинне джерело енергії, тоді як функціональний блок повинен забезпечити належну роботу цієї основної конструкції.

Доступ до різних ресурсів в домашній мережі повинен бути чітко розмежованим. Для повного доступу до певних даних важливо ввести класифікаційний пароль у систему.

Оскільки можливий доступ до системи через бездротову мережу, важливо приділити увагу захисту інформації в таких випадках.

3.5 Розробка апаратних засобів комп'ютерної системи

3.5.1 Вибір та характеристики пристроїв керування

Преповажна задача, яку необхідно вирішити, полягає у обробці отриманої інформації від датчиків. Для вирішення цього завдання використовується мікроконтролерна платформа Arduino модель Mega 2560 R3. Цей мікроконтролер використовує USB-UART перехідник з додатковим мікроконтролером ATMega16u2, що не вимагає додаткових драйверів і має вищу швидкість, ніж його попередники [12].

Однією з ключових задач при розробці системи розумного будинку є забезпечення зв'язку системи з користувачем та доступу до Інтернету. З цією метою використовується двох-смуговий маршрутизатор TP-Link Archer AX53. За його допомогою забезпечується потужний сигнал завдяки чотирьом фіксованим високопродуктивним антенам. Технологія Beamforming дозволяє концентрувати безпровідний сигнал у напрямку клієнтів для розширення радіусу дії Wi-Fi. Модуль FEM покращує потужність передачі для кращого покриття сигналу [13].

В якості сервера для системи було обрано одноплатний комп'ютер Raspberry Pi 4 Model B. Він оснащений 64-розрядним чотири-ядерним процесором (ARM Cortex-A72) з тактовою частотою 1,5 ГГц, підтримкою двох дисплеїв з роздільною здатністю до 4 K, оперативною пам'яттю до 8 ГБ, бездротовою мережею 2,4/5,0 GHz, Bluetooth 5.0/BLE, True Gigabit Ethernet, USB 3.0 і можливістю PoE. На Raspberry Pi 4 Model B можна встановити повноцінний дистрибутив Linux або спеціалізовану систему розумного будинку, таку як Windows 10 IoT Core з доступом до технології Microsoft Azure [14].

Всі наведені нижче датчики характеризуються високою чутливістю, високою роздільною здатністю та стабільною роботою. Більш того, всі вони ідеально

підходять для використання з платформою Arduino, що сприяє значному прискоренню виконання всіх завдань.

Таблиця 3.1 – Специфікація обладнання

№	Найменування і технічна характеристика	Тип, марка, позначення документа, опитувального о листа	Одиниці виміру	Кількість	Примітки
1	Плата Arduino Mega 2560 R3 мікроконтролер: ATmega16u2; цифрові входи/виходи:54…14 PWM аналогові входи: 16. флеш-пам'ять: 256 KB ОЗП: 8 KB; тактова частота: 16 MHz.	SBC0 MQTT Client1 SBC2 MQTT Client2	од.	2	Основа всієї система.
2	Маршрутизатор TP-Link Archer AX53 Стандарт WI–FI 802.11: 2,4ГГц ax/n/b/g; 5ГГц x/ac/n/a. Порти LAN: Gigabit Ethernet (100/1000). Підтримка протоколів: PPPoE, PPTP, DHCP, DDNS, L2TP,VPN.	Router0 Router1	од.	2	Зв'язок з всією мережою
3	Сервер Raspberry Pi 4 Процесор: BCM 2711 Cortex-A72 Кількість ядер:4. Частота: 1.5 ГГц. Об'єм встановленої пам'яті: 8 ГБ	Server-PT	од.	1	Оброблює всю інформацію
4	Датчик чадного газу MQ-7 навантажувальний опір: 10 К виявлення концентрації газу: 10…1000 ppm; час розігріву: від 60 (напруга підігрівача 5 В) до 90 с (для напруги підігрівача 1,4 В	MQ-7	од.	10	Вимірює шкидливий газ
5	Датчик температури та вологості виробник: AOSONG; DHT21 AM2301A; точність: 0.1 ºC; діапазон вимірювання вологості: 0- 100%; діапазон виміру температури: -40 ~ 80 ºC; точність вимірювання вологості: ± 2% RH; точність вимірювання температури:± 0.5%;	DHT21 AM2301A	од.	10	Слідкує за температурою, при перевищенні температури спрацює
6	Датчик диму MQ- 2 діапазон: 300- 10000 ppm; Rs опір елемента 20 кОм 50ppm толуол;	MQ-2	од.	10	Реагує на дим
7	Датчик полум'я KY-026 хвилі від 760 нм до 1100 нм дальність виявлення вогню1м.	KY-026	од.	10	Реагує на полум'я

4 МОДЕЛЮВАННЯ МЕРЕЖІ РОЗУМНОГО БУДИНКУ

Змодельована мережа «Розумний будинок» у середовищі Cisco Packet Tracer представлена на рис. 4.1. У цьому моделюванні обрано дві зони контролю протипожежної безпеки: гараж та кухня.

Мережа «розумного будинку» включає в себе різноманітні провідні та бездротові пристрої Інтернету речей (IoT) та пристрої мережевої інфраструктури. Бездротовий маршрутизатор Linksys WRT300N виступає як концентратор і маршрутизатор для всіх внутрішніх пристроїв. Усі внутрішні пристрої підключаються до домашньої мережі через Wi-Fi та контролюються за допомогою сервера IoT. Пристрої в будинку можуть з'єднуватися з домашнім шлюзом як через бездротовий, так і через дротовий зв'язок. Домашній шлюз має одну загальнодоступну IP-адресу (180.16.0.2/16), яку надає йому постачальник послуг Інтернету (ISP) (рис. 4.2), і оснащений вбудованим серверним модулем протоколу динамічної конфігурації хоста (DHCP). Цей DHCP розподіляє приватні IP-адреси своїм локальним хостам та пристроям IoT в межах діапазону його пулу IP-адрес (192.168.0.0/24) (рис.4.3).

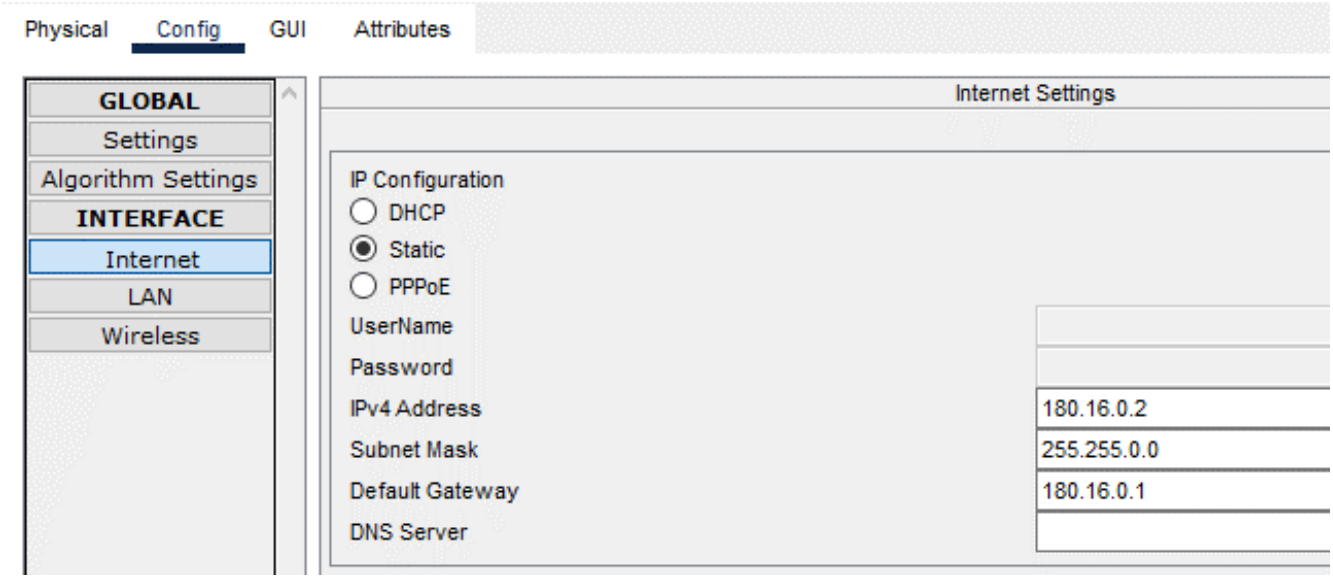


Рисунок 4.1 - Налаштування Internet на домашньому маршрутизаторі

Рисунок 4.2 – Загальний вигляд системи «Розумний будинок»

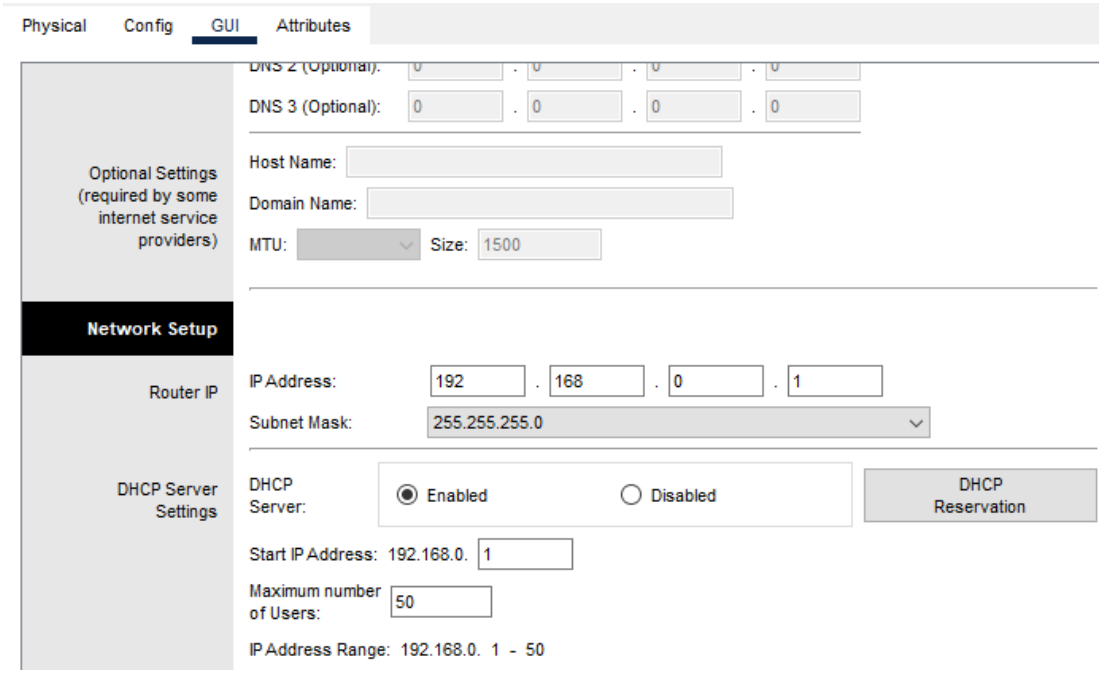


Рисунок 4.3 – Налаштування DHCP бездротового маршрутизатора

Маршрутизатор ISP виконує роль постачальник послуг Інтернету. Мережні налаштування представлено на рис. 4.4.

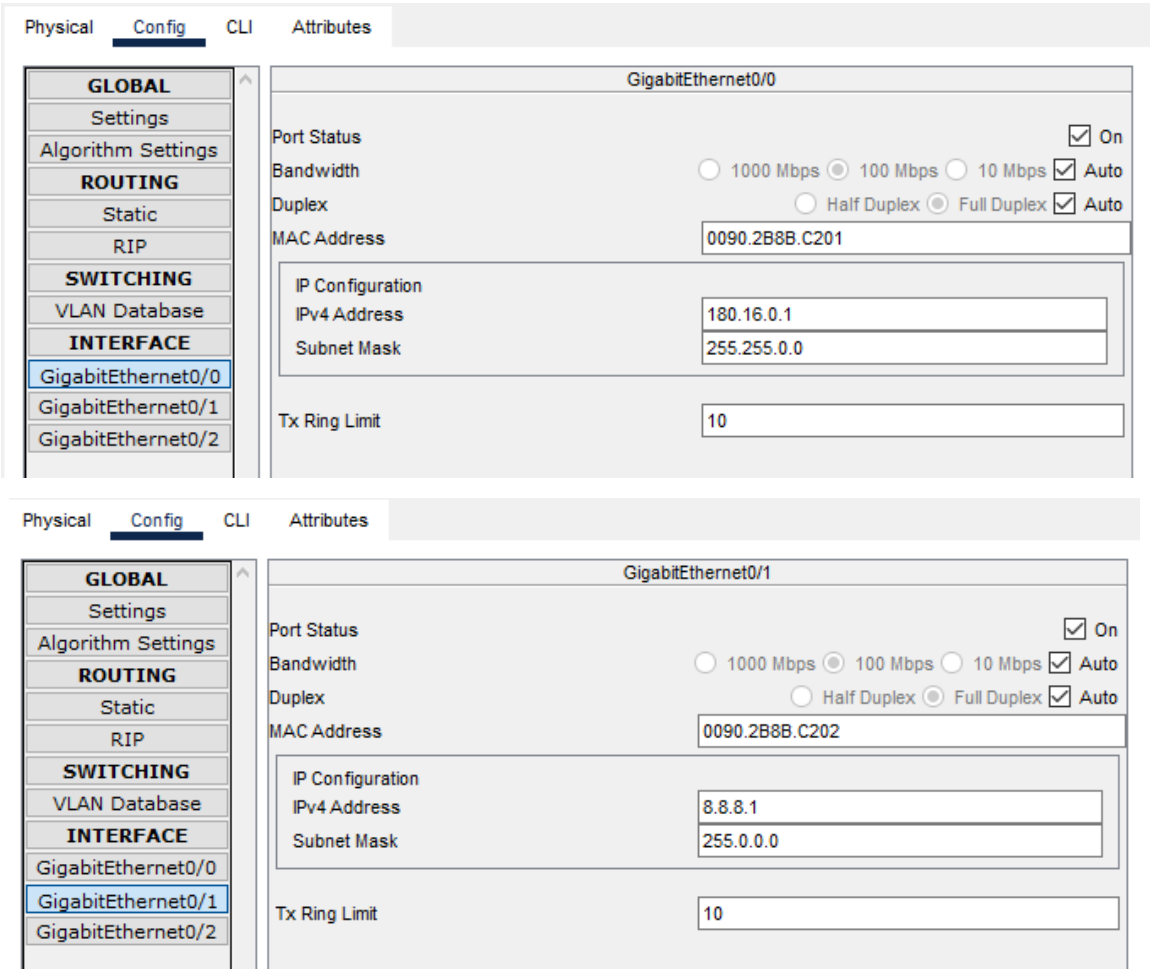


Рисунок 4.4 – Мережні налаштування ISP

Сервер IoT обладнаний веб-інтерфейсом, який дозволяє користувачам дистанційно відстежувати та керувати різними розумними домашніми пристроями через будь-який комп'ютер вдома. Крім того, сервер виконує функцію брокера MQTT. Йому призначена публічна IP-адреса 8.8.8.8/8. На вкладці Service активовано службу IoT, а також створено користувача з ім'ям cisco (рис. 4.5).

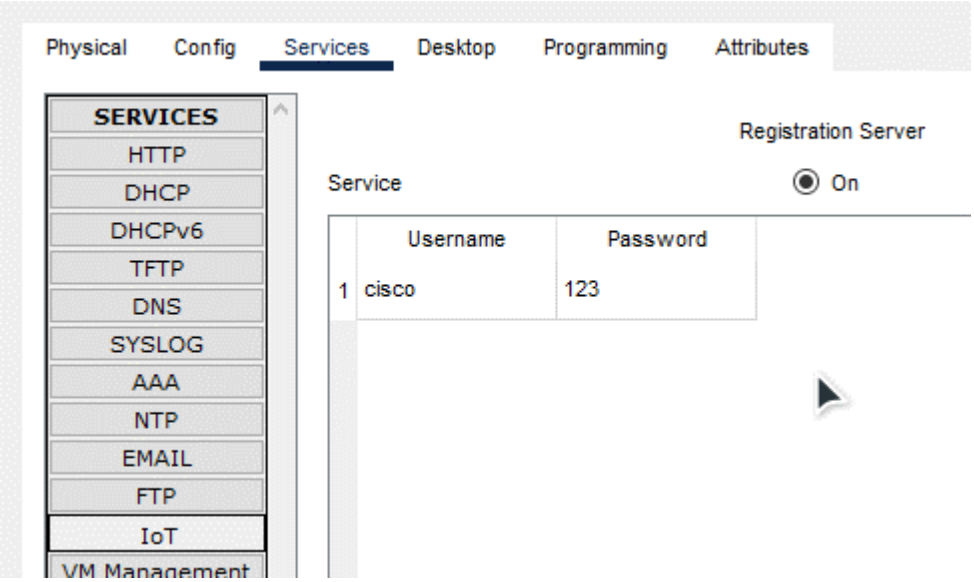


Рисунок 4.5 – Вікна властивостей сервера

Сирени та детектори диму передають тривожні повідомлення про пожежу та інформацію щодо місця розташування джерела пожежі (або загоряння) на контролери SBC. Кабель, який використовується для підключення пристроїв IoT до контролера SBC, отримав назву IoT Custom Cable.

Контролери SBC, розташовані всередині кожної кімнати, відповідають за керування пристроями IoT в мережі та мають зв'язок із кожним пристроєм IoT. SBC, які додаються до «розумного будинку», використовуються для моніторингу рівня диму, виміряного датчиками диму, і вирішення, чи слід активувати сирену, відкривати вікна чи двері. У випадку перевищення порогового рівня монооксиду вуглецю SBC автоматично відкриває вікно, передні двері чи гаражні двері. Ця дія зупиняється лише в разі зниження рівня оксиду вуглецю нижче встановленого порогу.

Контролери SBC також виконують роль клієнта MQTT. У цьому експерименті SBC виступає в якості видавця MQTT, що перевіряє рівень диму, а клієнти IoT виступають як абоненти MQTT, спостерігаючи за станом пристроїв IoT.

У «розумному будинку» є ноутбук, за допомогою якого можна відстежувати стан підключених пристроїв і керувати ними.

4.2 Моделювання протипожежної безпеки «Розумного будинку» з керуванням через web-сервер

Сервер IoT може виступати в якості web-сервера для IoT-пристроїв або брокера MQTT. Розглянемо ситуацію, коли IoT-сервер функціонує як HTTP-сервер. Server IoT надає web-інтерфейс, який дає користувачам можливість віддалено контролювати та управляти різними розумними домашніми пристроями за допомогою будь-якого комп'ютера вдома.

Для підключення пристроїв до бездротової домашньої мережі слід надати кожному пристрою унікальне ім'я та в налаштуваннях мережі вказати отримання мережної адреси через DHCP.

Для підключення пристроїв до IoT web-сервера у розділі IoT Server потрібно вказати його параметри (рис. 4.6).

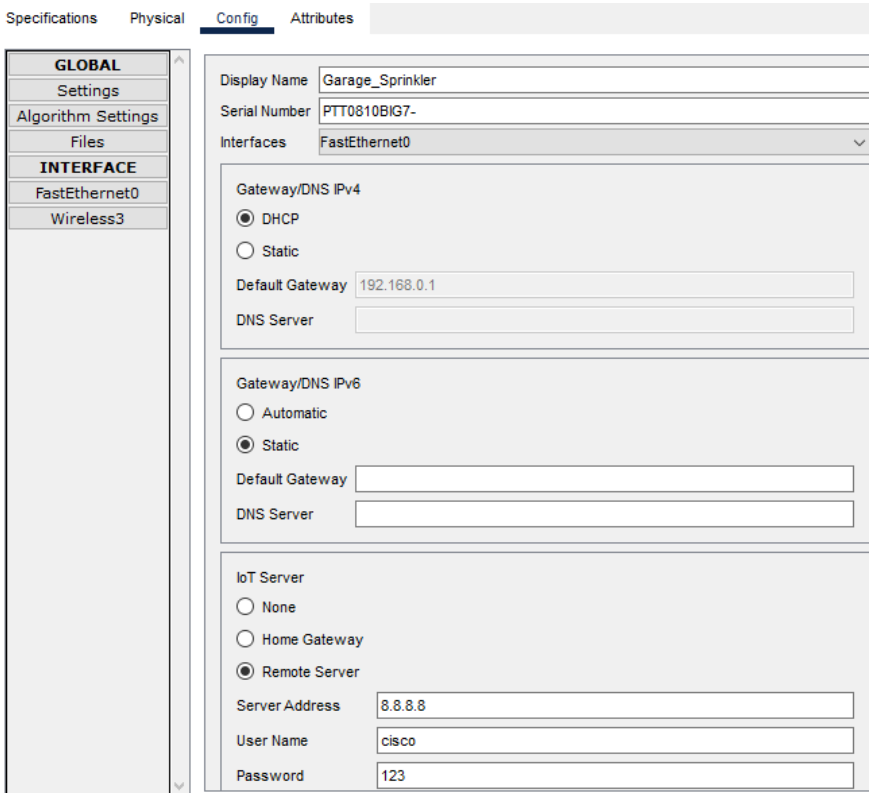


Рисунок 3.6 – Налаштування пристроїв IoT

Зробимо підключення до web-інтерфейсу IoT сервера, для цього зайдемо на ноутбуці у Web Browser і введемо в поле URL IP адресу IoT сервера 8.8.8.8.

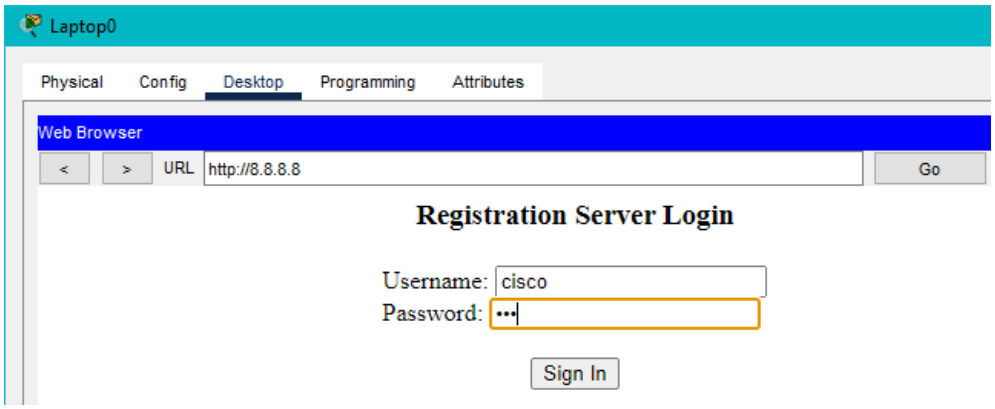


Рисунок 4.7 - Сторінка авторизації для входу на IoT Сервер

Вводимо логін: cisco та пароль: 123 і нам відкривається список усіх підключених IoT пристроїв (рис. 4.7).

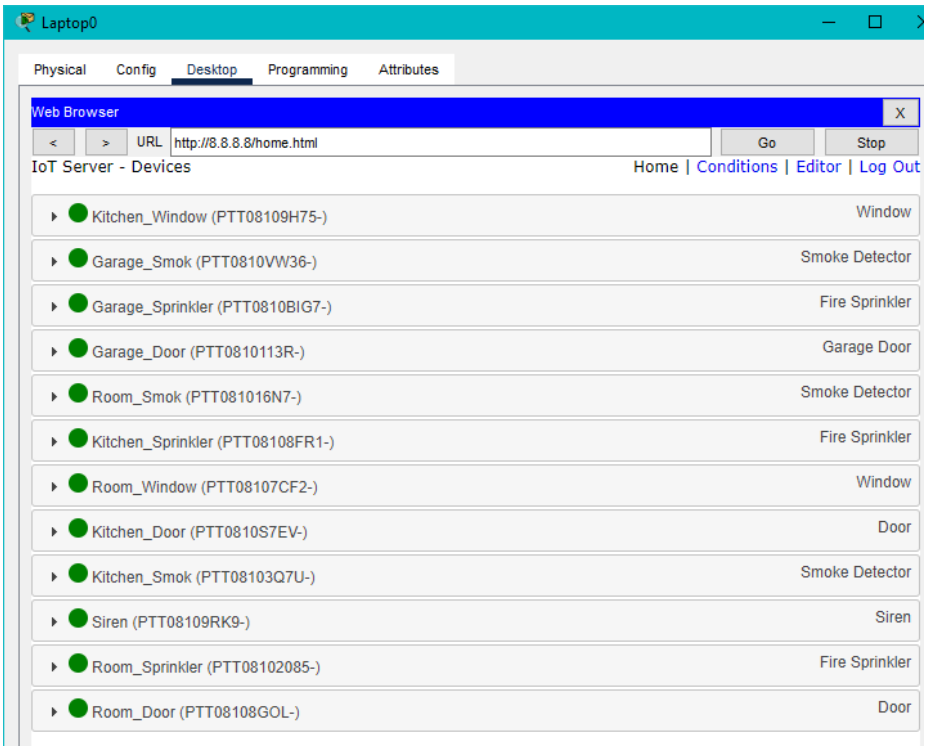
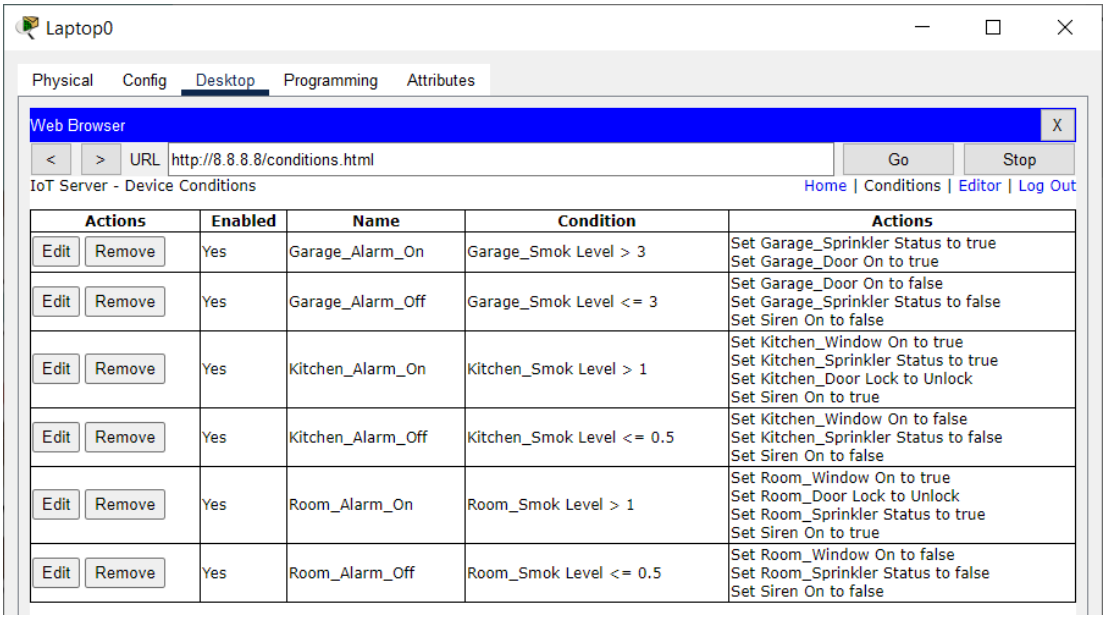


Рисунок 4.8 - Список підключених пристроїв

Розглянемо простий демонстраційний сценарій, пов'язаний із самостійною системою керування кухнею в разі виявлення полум'я. Для виконання цього завдання використовується полум'я як чинник, що впливає на температуру приміщення. Для його реалізації необхідно використовувати IoT-пристрій, а саме датчик полум'я, який передає інформацію на сервер IoT. Сервер IoT неперервно контролювати отримані дані про температуру. Якщо значення перевищує 0,5, сервер генерує команду «Увімкнути» для іншого IoT-пристрою, у даному випадку

- вікна. Вікно відкривається, вмикається стельовий вентилятор та спрацьовує розпилювач води. Відповідні налаштування продемонстровані на рис. 4.9.

Для створення правил включення та виключення для активації пожежного захисту потрібно зайти на сервер IoT через веб-браузер, обрати вкладку «Conditions» і створити необхідні правила відповідно до умов.42



Actions	Enabled	Name	Condition	Actions
Edit Remove	Yes	Garage_Alarm_On	Garage_Smok Level > 3	Set Garage_Sprinkler Status to true Set Garage_Door On to true
Edit Remove	Yes	Garage_Alarm_Off	Garage_Smok Level <= 3	Set Garage_Door On to false Set Garage_Sprinkler Status to false Set Siren On to false
Edit Remove	Yes	Kitchen_Alarm_On	Kitchen_Smok Level > 1	Set Kitchen_Window On to true Set Kitchen_Sprinkler Status to true Set Kitchen_Door Lock to Unlock Set Siren On to true
Edit Remove	Yes	Kitchen_Alarm_Off	Kitchen_Smok Level <= 0.5	Set Kitchen_Window On to false Set Kitchen_Sprinkler Status to false Set Siren On to false
Edit Remove	Yes	Room_Alarm_On	Room_Smok Level > 1	Set Room_Window On to true Set Room_Door Lock to Unlock Set Room_Sprinkler Status to true Set Siren On to true
Edit Remove	Yes	Room_Alarm_Off	Room_Smok Level <= 0.5	Set Room_Window On to false Set Room_Sprinkler Status to false Set Siren On to false

Рисунок 4.9 – Додавання правил у логіку роботи пристроїв

Тестування роботи системи протипожежного захисту розумного будинку (див. рис. 4.10).

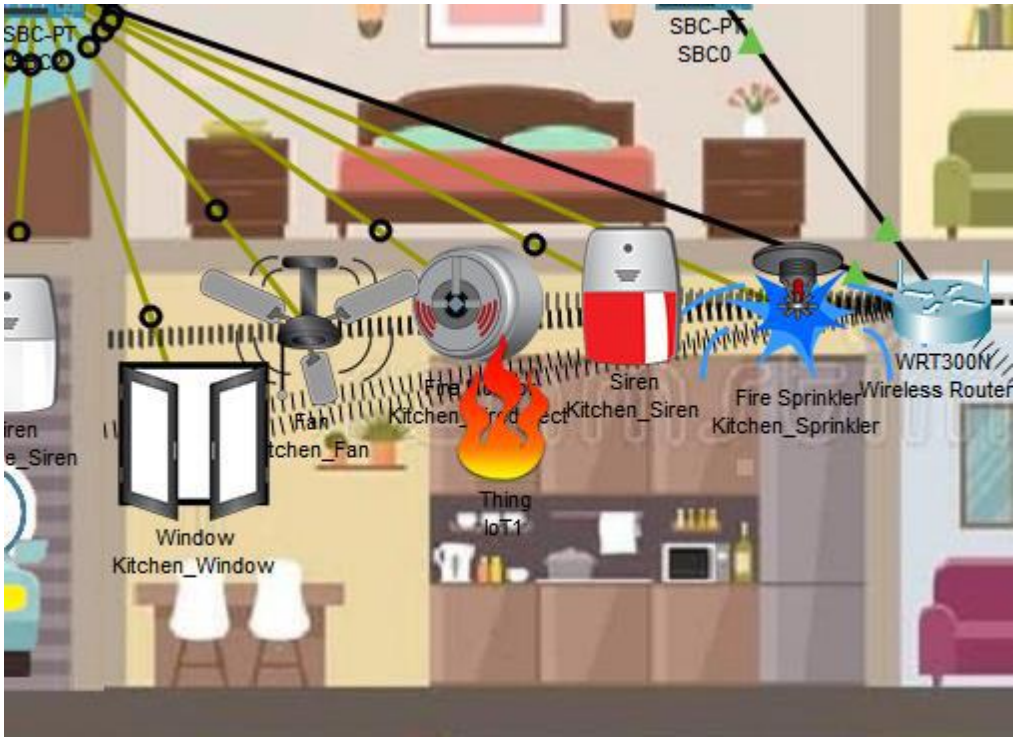


Рисунок 4.10 – Робота системи протипожежного захисту

В режимі симуляції час, що пройшов від отримання веб-сервером сигналу про пожежу та ввімкнення всіх пристроїв 0,845 с (рис. 4.11).

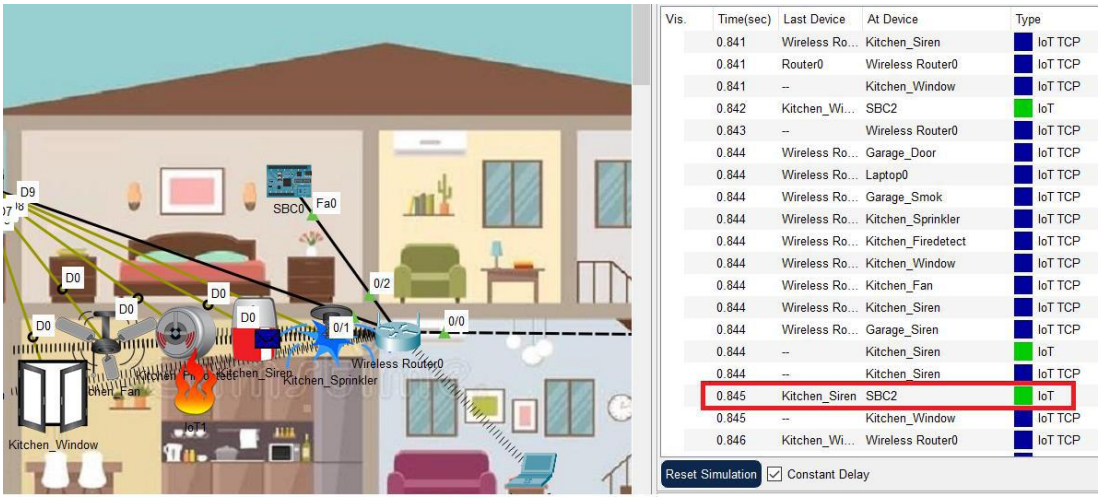


Рисунок 4.11 - Тестування в режимі симуляції

Моделювання в Packet Tracer системи протипожежної безпеки «Розумного будинку» з керуванням через MQTT-сервер

4.3 Моделювання брокера MQTT

Розглянемо випадок, коли IoT-сервер діє як MQTT-брокер. Щоб встановити MQTT Broker на сервері, на вкладці Desktop встановити додаток MQTT Broker (рис. 4.12).

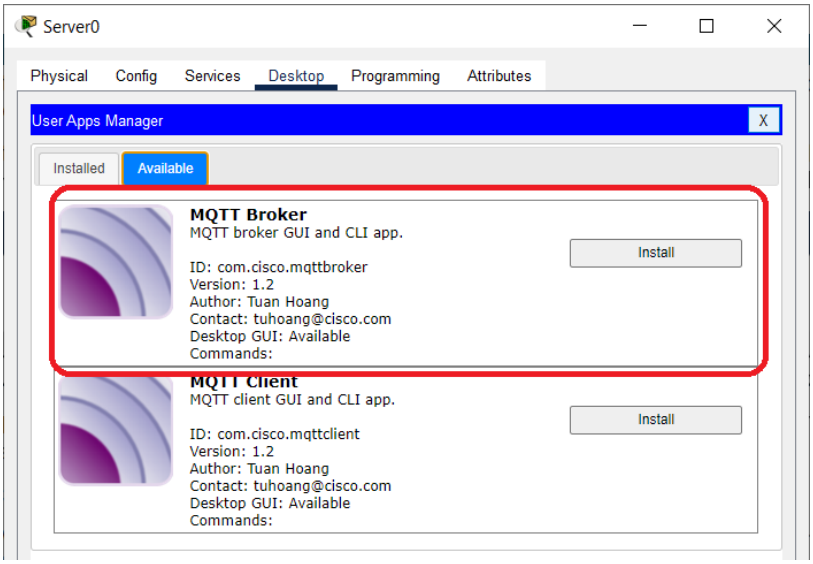


Рисунок 4.12 – Встановлення MQTT Broker

Після встановлення на Desktop з’явиться ярлик брокера MQTT Broker. Після запуску додатку необхідно створити ім’я користувача та пароль (рис. 4.13).

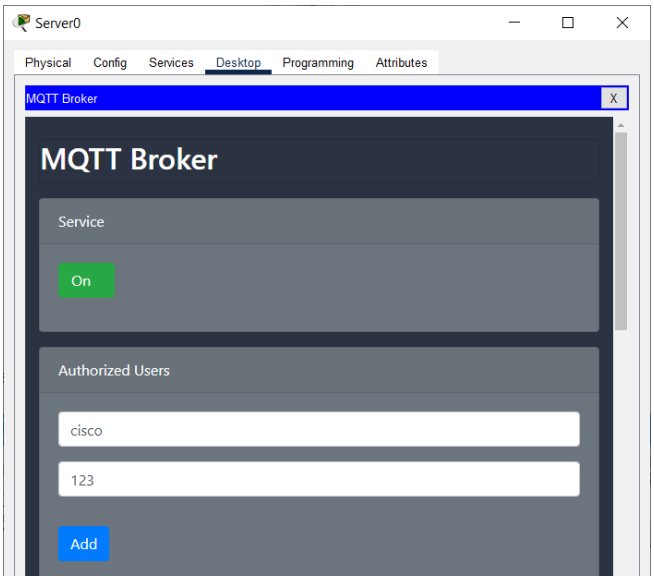


Рисунок 4.13 – Додавання користувача на MQTT Broker

IP-адреса брокера така ж, як і IP-адреса сервера.

4.4 Моделювання клієнта MQTT

Клієнтами MQTT можуть бути будь-які кінцеві пристрої, як от пристрої Інтернету речей, ноутбук або MCU тощо.

Контролери SBC0 виконують роль клієнта MQTT. У цьому експерименті SBC1 діє як видавець MQTT, перевіряючи рівень диму, а клієнти IoT діють як абоненти MQTT для моніторингу стану пристроїв IoT.

На вкладці Dekstop після встановлення MQTT Client необхідно вказати IP-адресу брокера, ім’я користувача та пароль (рис. 4.14).

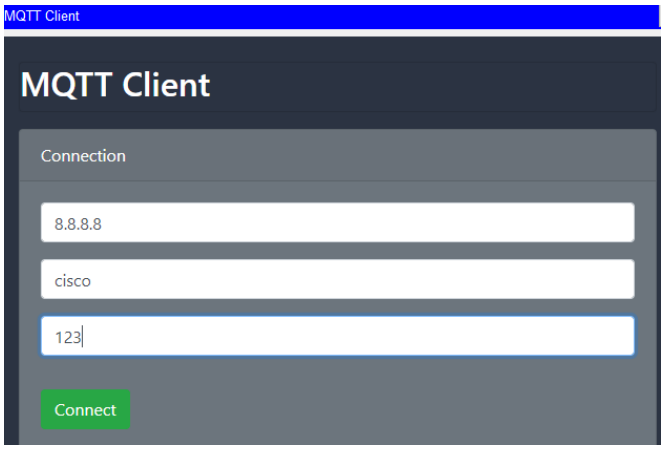


Рисунок 4.14 – Налаштування MQTT Client

Після успішного під’єднання клієнтів на MQTT-брокері відображаються відомості про них (див. рис. 4.15).

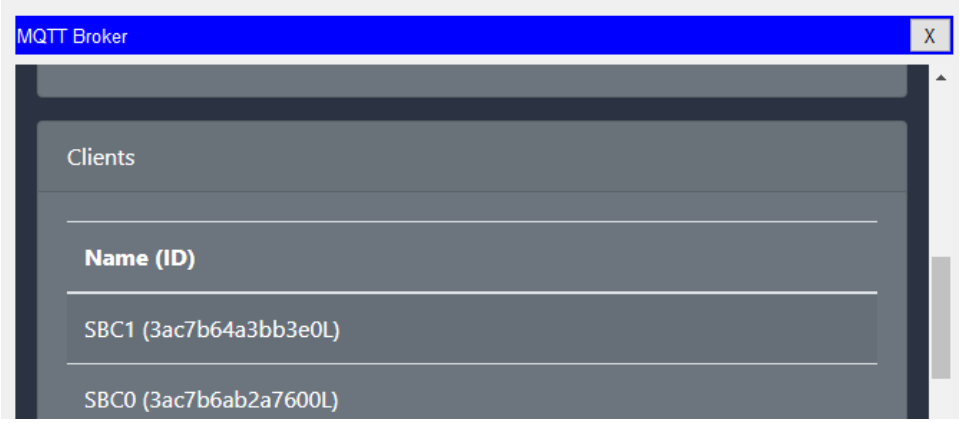


Рисунок 4.15 - Відомості про клієнтів

4.4 Моделювання роботи протипожежної системи по протоколу MQTT

У протоколі MQTT є 2 функції, підписатися і публікувати повідомлення. Підписка (Subscription) – це спосіб змусити програму «прослуховувати» повідомлення на певну тему в «Брокері» а публікація (Publish) – надсилати повідомлення на сервер, щоб інший пристрій міг «прослуховувати» ці повідомлення. Один пристрій завжди повинен бути підписаний на тему, а інший пристрій повинен публікувати повідомлення на цю тему.

В таблиці 4.1 представлені оголошені підписки.

Таблиця 4.1 – Реалізовані підписки

Topic	Payload	Дія
/GARAGE_SMOKE	open	Відкрити двері гаража
/GARAGE_SMOKE	close	Зачинили двері гаража
/GARAGE_DOOR	open	Відкрити двері гаража
/GARAGE_DOOR	close	зачинили двері гаража
/GARAGE_SIREN	on	Включити розпилювач в гаражі
/GARAGE_SIREN	off	Виключити розпилювач в гаражі
/KITCHEN_WINDOW	open	Відкрити вікно на кухні
/KITCHEN_WINDOW	close	Зачинии вікно на кухні
/KITCHEN_SPRINKER on		Включити розпилювач на кухні
/KITCHEN_SPRINKER off		Виключити розпилювач на кухні
/KITCHEN_FAN	on	Включити вентилятор кухні
/KITCHEN_FAN	off	Виключити вентилятор кухні
/KITCHEN_SIREN	on	Включити сирену на кухні
/KITCHEN_SIREN	off	Виключити сирену на на кухні
/KITCHEN_FIRE	no	Включити розпилювач на кухні
/KITCHEN_FIRE	alarm	Виключити розпилювач на кухні
/KITCHEN_ALARM	off	Виключити розпилювач, вентилятор, сирену на кухні
/KITCHEN_ALARM	alarm	Включити розпилювач, вентилятор, сирену на кухні

У цьому експерименті SBC2 діє як видавець MQTT, перевіряючи стан повітря в гаражі та сигналізатор датчику вогню пристрою IoT. SBC0 діє як підписник MQTT за підписками «/GARAGE_SIREN» та «KITCHEN_SIREN», щоб отримувати повідомлення при зміни стану сирен. SBC2 підписан на «/KITCHEN_ALARM», щоб при отриманні команди alarm або off контролер ввімкнув/вимкнув сирену, розпилювач та вентилятор на кухні. На рис. 4.16 показані MQTT брокері відомості про підписки.

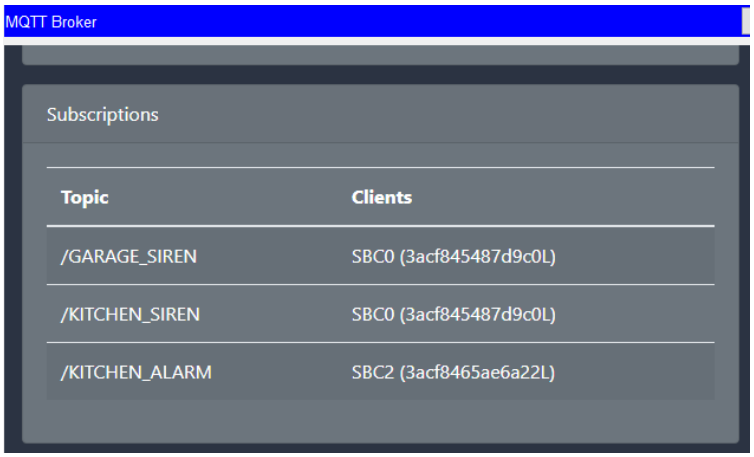


Рисунок 4.16 – Відомості про підписки на MQTT брокері

На рис. 4.17 продемонстровано, як на SBC2 зроблена підписка на топік /KITCHEN_ALARM.

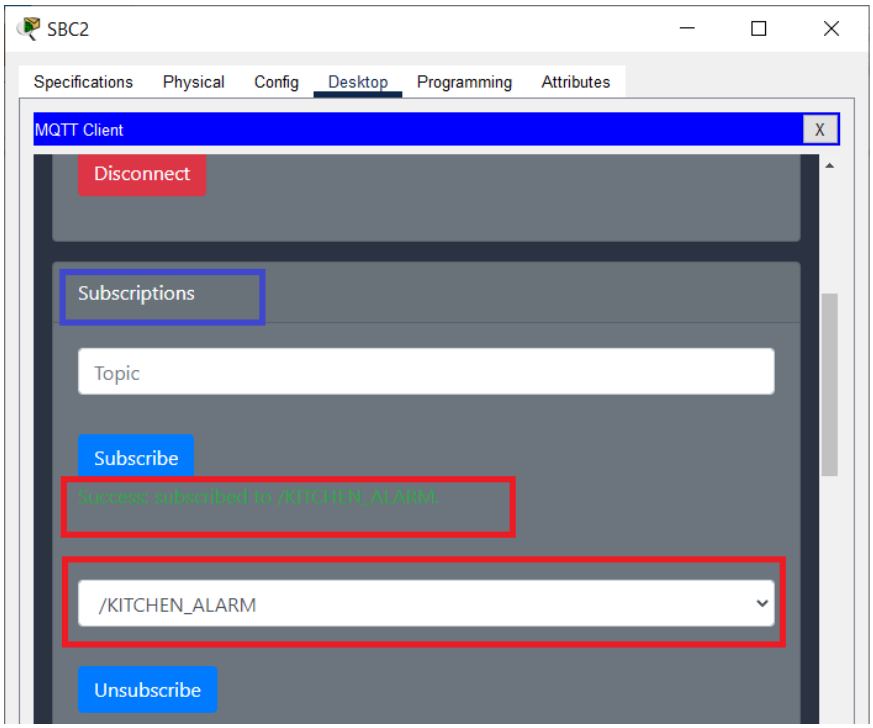


Рисунок 4.17 – Підписка на SBC2

На рис. 4.18 показано, що на SBC0 створена публікація «/KITCHEN_ALARM» з payload alarm, щоб SBC2 контролер ввімкнув сирену, розпилювач та вентилятор на кухні. SBC2 успішно отримав це повідомлення і включає пристрої.

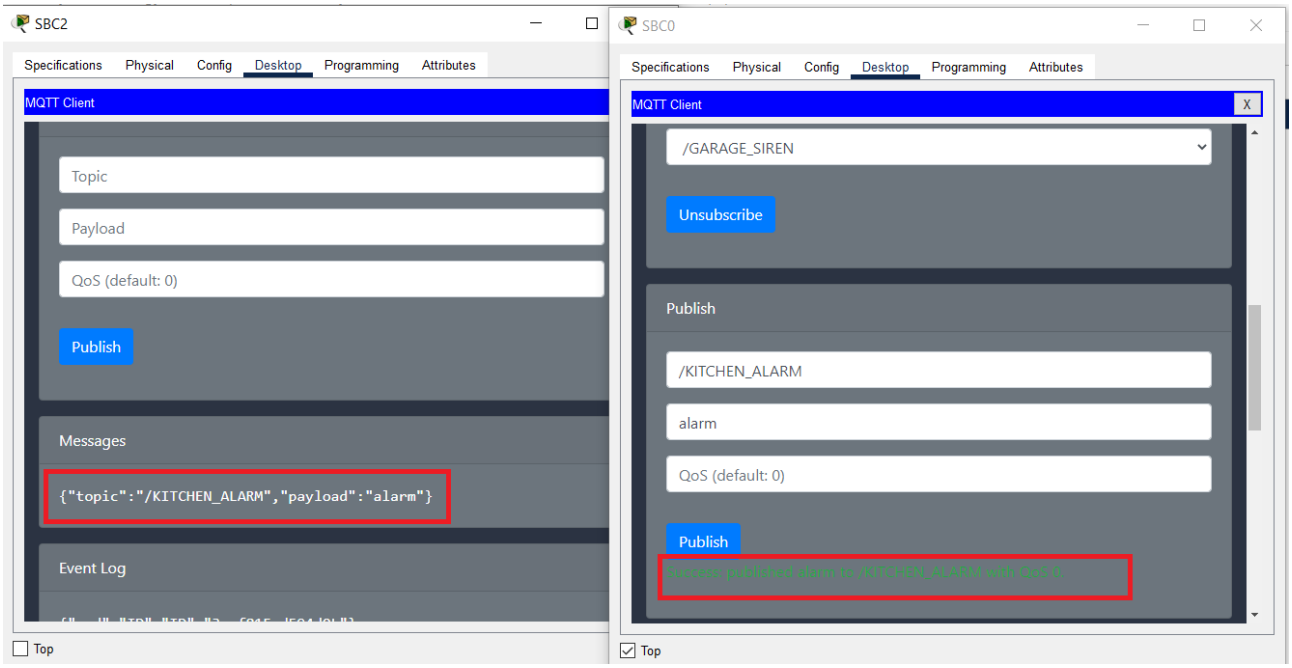


Рисунок 4.18 – Створення публікації /KITCHEN_ALARM/alarm

На рис. 4.19 бачимо час (0,007 с), який пройшов з моменту отримання повідомлення від брокера SBC0 до моменту включення пристроїв.

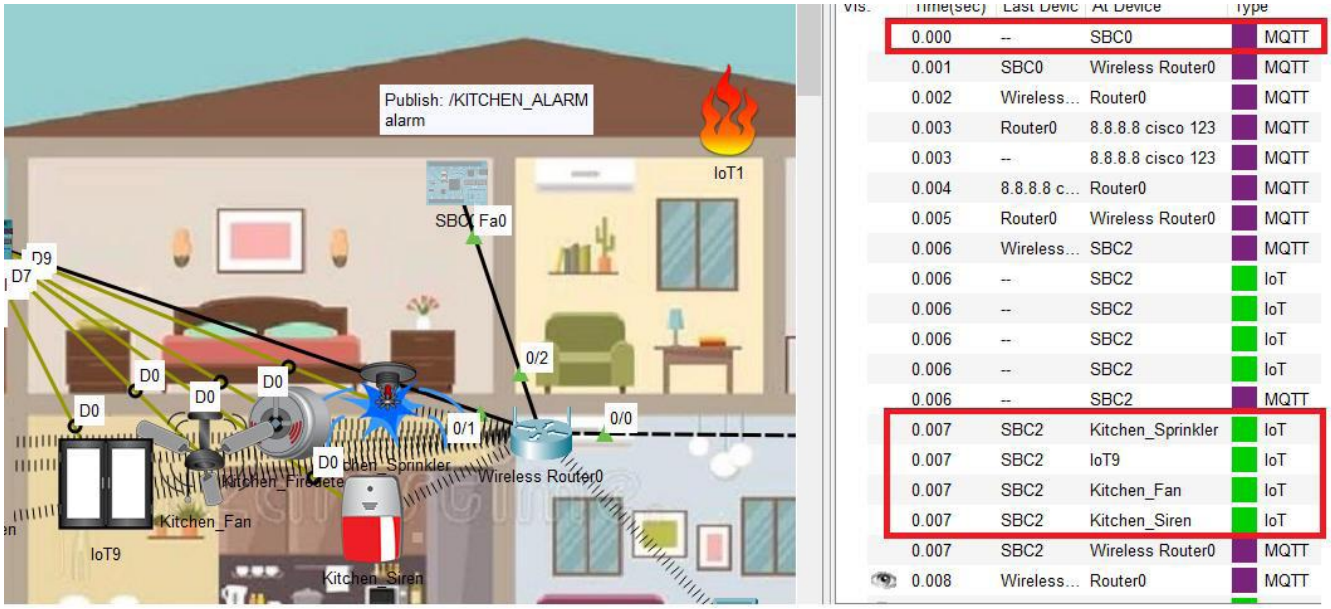


Рисунок 4.19 – Час моделювання експерименту

Після того, як включилась сирена на кухні, SBC0 отримав про це повідомлення (рис. 3.20).

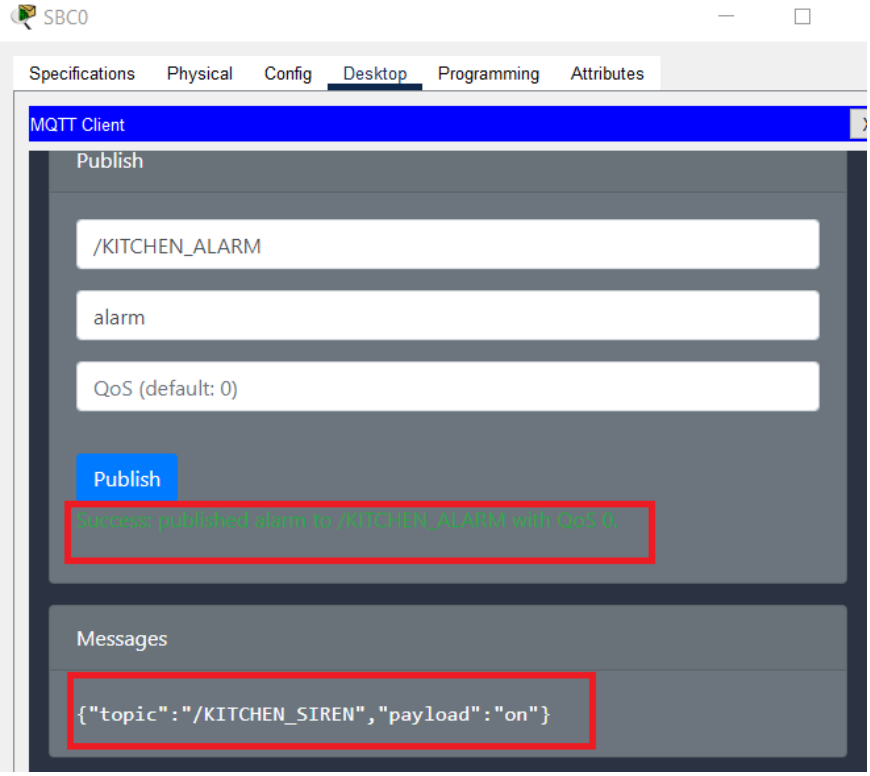


Рисунок 4.20 – Отримання повідомлення про включення сирени на кухні

Проведений експеримент демонструє, що час між публікацією на клієнті MQTT та передачею до контролера 0,007с, в порівнянні з часом 0,845 між web-сервером та контролером. Тож протокол MQTT краще використовувати для швидкої та своєчасної передачі даних між M2M.

5 ЕКСПЕРИМЕНТАЛЬНИЙ РОЗДІЛ

5.1 Математична модель мережі

Мережеві моделі відрізняються від інших більш традиційних динамічних моделей деякими фундаментальними аспектами. По-перше, компоненти системи можуть бути з'єднані нерівномірним та рівномірним навантаженням. Це означає, що в одній мережі деякі компоненти можуть бути дуже добре підключені, а інші – ні. Така неоднорідна зв'язність ускладнює математичний аналіз властивостей системи. У той же час це також дає моделі більше можливостей для більш тісного представлення зв'язків між компонентами системи з реальністю. Можна представити будь-яку топологію мережі (тобто форму мережі), явно вказавши детально, які компоненти до яких інших компонентів підключені і як. Це робить мережеве моделювання обов'язково ресурсомістким. Незалежно від того, чи згенерована мережа за допомогою якогось математичного алгоритму або реконструйована з реальних даних, створена мережева модель буде містити велику кількість детальної інформації про те, як саме з'єднані компоненти. Потрібно ефективно створювати, управляти та маніпулювати цією інформацією.

Кількість компонентів може динамічна збільшуватися або зменшуватися з часом у певних динамічних моделях мережі. Таке зростання (або занепад) топології системи є загальним припущенням, яке зазвичай робиться в генеративних мережевих моделях, які пояснюють самоорганізовані процеси певних топологій мережі. Однак така динамічна зміна числа компонентів в системі здійснює величезний стрибок в порівнянні з іншими більш традиційними моделями динамічних систем, включаючи всі моделі. Це пов'язано з тим, що, розглядаються стани компонентів системи, наявність одного більшого (або меншого) компонента означає, що фазовий простір системи набуває на один більший (або менший) вимір. З точки зору традиційних динамічних систем, зміна розмірів фазового простору системи з часом звучить майже незаконно, але подібні речі трапляються в багатьох реальних складних системах. Мережеві моделі дозволяють природно описати такі божевільні процеси.

Поведінку мережі зв'язку можна змоделювати як потік одиниць трафіку по каналах, з'єднаних вузлами. Моделі мережі пов’язані її з потоком трафіку, подібною до течії. Дискретна модель вузлового зв'язку наголошує на чергуванні пакетів і потоці пакетів від просторової точки до просторової. Модель припускає, що пакети знаходяться в буферах у кожному вузлу та класифікуються за місцем призначення та тривалістю часу, протягом якого вони перебували в буфері.

Магістерській роботі був створений алгоритм виходу пакетів з буфера в кожному вузлу відповідно до їх віку і переходу до наступного вузла по заздалегідь визначеному шляху до місця призначення. Цей алгоритм обчислює швидкість, з якою пакети поширюються до наступної ланки маршруту до місця призначення, припускає джерело пакетів, що походять з вузла, і віднімає пакети, пунктом призначення яких є цей конкретний вузол. Модель континууму, отримана з цієї моделі дискретного потоку, призводить до рівняння безперервності потоку. Рівняння неперервності описує щільність пакетів як функцію часу і простору, так що ми можемо передбачити зміни в глобальних моделях потоків і оптимальних шляхах в мережі. Розв'язки рівнянь потоку в одному вимірі показують, що якщо джерела занадто сильні або потік обмежений, щільність пакетів зростає в найближчому вузлу вище за течією. Коли потужність джерела зменшується, або коли потік відновлюється, буферизовані пакети течуть з пропускною здатністю, поки щільність не зменшиться.

Реалізація функції точного моніторингу за пакетами, пошук ознак зниження продуктивності, загроз безпеці тощо є найскладнішим завданням для моделі мережі. При виникненні критичного стану з мережевим трафіком необхідно вжити спеціальних заходів по забезпеченню стабільної і швидкої роботу мережі.

Модель мережі розроблено за структурною схемою комп'ютерної системи протипожежного захисту «розумного будинку» з віддаленим контролем через сервер MQTT для «розумного котеджного комплексу», її створено в MATCAD за правилами замкнутої системи масового обслуговування (рис. 5.1).

Моделі комп'ютерної системи протипожежного захисту розумного будинку повністю збігається з існуючою структурою мережі фізичного рівня.

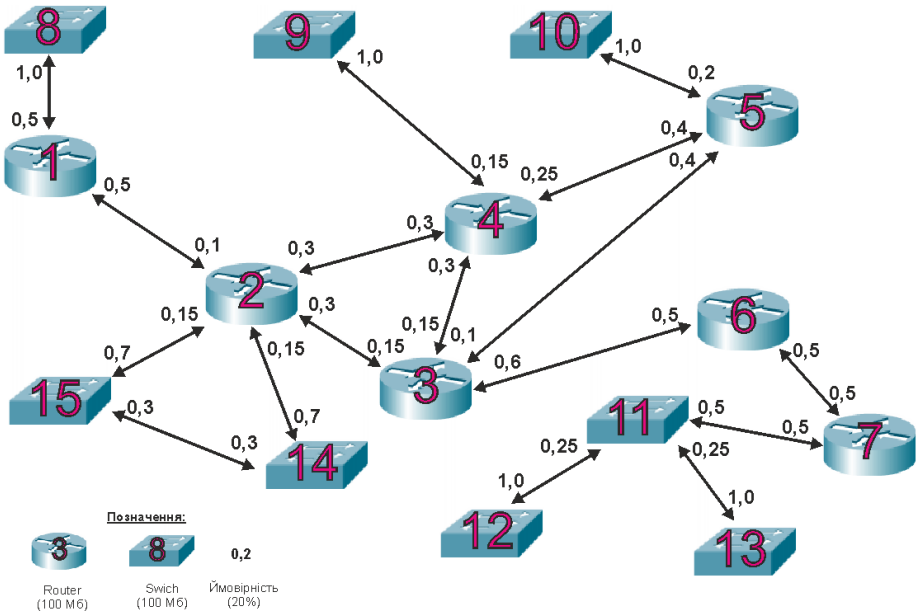


Рисунок 5.1 – Модель комп'ютерної системи протипожежного захисту розумного будинку

Розподіл умовної кількості інформаційних пакетів по каналам мережі обрано за об’єми інформації яка передається, тобто циркулює між джерелом та споживачем.

Згідно зі схемою комп'ютерної системи протипожежного захисту розумного будинку мережі модель містить бюджетні версії роутерів та комутаторів зі швидкістю передачі 100 Мб,

Найбільша ймовірність передачі позначена на схемі моделі – від 0 до 1 (0…100%). Всі вузли моделі визначені відповідно до теорії системи масового обслуговування. Ймовірність передачі інформації самому собі становить 0.

Для вузлів моделі мережі прийнято припущення, що ймовірність зв’язку між собою є однаковою. Також прийнято, що найбільша інтенсивність зв’язку буде через вузол виходу до інтернету.

В структурі моделі комп'ютерної системи протипожежного захисту розумного будинку мережеві вузли з першого по сьомий – є роутерами Cisco зі швидкістю 100 Мб, які мають обслуговувати локальних мережі кінцевих користувачів та виходу в інтернет.

Вузли з восьмого по п’ятнадцятий є мережевими комутаторами рівня доступу кінцевих користувачів. Комутатори також фірми Cisco зі швидкістю передачі інформації 100 МБ.

В моделі мережі зв’язки між роутерами та комутаторами мають вірогідність передачі інформаційних пакетів обрана за реальними показниками та є ключовим моментом для аналізу у розробленого ПЗ в MATCAD.

Ступінь інформаційного навантаження на канал мережевого вузла визначено відповідним номіналом вірогідності, що відображено в маршрутній матриця.

$$Pr := \begin{pmatrix} 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0 & 0.3 & 0.3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.15 & 0.15 \\ 0 & 0.15 & 0 & 0.15 & 0.1 & 0.6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.3 & 0.3 & 0 & 0.25 & 0 & 0 & 0 & 0.15 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.40 & 0.4 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0.7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.3 \\ 0 & 0.7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.3 & 0 \end{pmatrix}$$

Рисунок 5.2 – Маршрутна матриця

Налагодження створеної моделі мережі полягає в завданні швидкодії для кожного вузлу. Час обробки відповідає абстрактним одиницям часу, спираючись на параметри реального мережевого обладнання. Показник 10 застосовано для 100 Мб мережевого обладнання, а показник 1 для 1 Гб мережевого обладнання (рис. 5.3).

$$\tau_i := \begin{pmatrix} 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \end{pmatrix}$$

Рисунок 5.3 – Час обробки повідомлення

5.2 Моделювання роботи мережі

5.2.1 Розрахункова частини

Коефіцієнти ймовірності передачі інформації по каналам розраховуються за нормованим показником за методом Гауса і трансформуються в матрицю, яка представлена на рис. 5.4.

$$e_{\omega} := \begin{pmatrix} 1 \\ 5 \\ 10.222 \\ 4.889 \\ 2.806 \\ 8.178 \\ 4.089 \\ 0.5 \\ 0.733 \\ 0.561 \\ 0 \\ 0 \\ 0 \\ 1.071 \\ 1.071 \end{pmatrix}$$

Рисунок 5.4 – Коефіцієнти ймовірностей

Для обробки інформаційних пакетів в мережевих вузлах моделі задаємо кількість конвеєрів.

Cisco IOS XE - це мережева операційна система для підприємств. Віна працює на таких комутаторах, як Catalyst 9000, маршрутизаторах, таких як ASR 1000, CSR1000v і ISR 1000 і 4000, контролерах бездротової локальної мережі Catalyst 9800, а також на кількох інших пристроях у лінійках продуктів IoT і Cable. Починаючи з випуску IOS XE 16.6, з’явилася підтримка телеметрії на основі моделі, яка надає операторам мереж додаткові можливості для отримання інформації зі своєї мережі.

Традиційно SNMP був дуже успішним для моніторингу корпоративних мереж, але він має обмеження: ненадійний транспорт, неузгоджене кодування між версіями, обмежені параметри фільтрації та отримання даних, а також вплив на ЦП і пам’ять працюючого пристрою під час використання кількох рішень для моніторингу мережі. опитувати пристрій одночасно. Телеметрія на основі моделі усуває багато недоліків застарілих можливостей моніторингу та надає додатковий інтерфейс, у якому телеметрія тепер доступна для публікації.

Так, це функція на основі push. Нам більше не потрібно опитувати пристрій і запитувати робочий стан. Тепер ми лише вирішуємо, які дані нам потрібні, як часто вони потрібні та куди їх надсилати. Після встановлення конфігурації пристрій із задоволенням публікує телеметричні дані стороннім збирачам, вашим інструментам моніторингу, механізмам пошуку та візуалізації великих даних, як-от Splunk і Elastic, або навіть у простому текстовому файлі – це повністю налаштовується, що ви робите з дані. У прикладі нижче ми використовуємо Telegraf + InfluxDB + Grafana для отримання, зберігання та візуалізації даних.

Практичне значення цих функції базується на використанні моніторингу використання ЦП пристрою. Ці данні можна отримати з пристрою Cisco який працює під керуванням оераційної системи IOS XE 16.10.

Моделі YANG є основою телеметрії на основі моделей, які доступні не лише для публікації телеметрії, але й для програмної конфігурації. Ці моделі даних знаходяться в пристрої IOS XE і можуть бути легко завантажені за допомогою таких інструментів, як YANG-Explorer. Усі моделі також публікуються на сторінці YangModels Github , що полегшує доступ до них і їх аналіз.

Інструмент YANG-explorer доступний на сторінці CiscoDevNet Github, де можна завантажити моделі YANG безпосередньо з пристрою IOS XE через NETCONF або інтерфейси RESTCONF і швидко показують, які дані доступні та з якої моделі.

За допомогою YANG можна налаштувати та перевірити підписки на телеметрію з традиційного CLI, а також через програмні інтерфейси NETCONF, RESTCONF і gNMI. Під час використання CLI команди show доступні з набором команд «show telemetry ietf», і налаштовуються аналогічно з командами «telemetry ietf» у режимі конфігурації. При використанні YANG моделі YANG «Cisco-IOS-XE-mdt-cfg.yang» і «Cisco-IOS-XE-mdt-oper.yang» доступні як для конфігурації, так і для робочих наборів даних [11].

В нашому випадку інформація про кількість конвеєрів вказана в матриці. Ці данні обрані відповідно до технічних параметрів діючого мережевого обладнання. Бюджетні мережеві вузли мають один конвеєр для обробки потокової інформації (рис. 5.5).

$m_i :=$

1
1
1
1
1
1
1
1
1
1
1
1
1
1

Рисунок 5.5 – Кількість конвеєрів

ПЗ моделі розраховує матрицю, в якій міститься інформація про можливі ймовірності для станів очікування з обробки інформаційних пакетів. Ця матриця представлена на рис. 5.6.

$T =$

	0	1	2	3	4
0	1	10	100	$1 \cdot 10^3$	$1 \cdot 10^4$
1	1	50	$2.5 \cdot 10^3$	$1.25 \cdot 10^5$	$6.25 \cdot 10^6$
2	1	102.22	$1.045 \cdot 10^4$	$1.068 \cdot 10^6$	$1.092 \cdot 10^8$
3	1	48.89	$2.39 \cdot 10^3$	$1.169 \cdot 10^5$	$5.713 \cdot 10^6$
4	1	28.06	787.364	$2.209 \cdot 10^4$	$6.199 \cdot 10^5$
5	1	81.78	$6.688 \cdot 10^3$	$5.469 \cdot 10^5$	$4.473 \cdot 10^7$
6	1	40.89	$1.672 \cdot 10^3$	$6.837 \cdot 10^4$	$2.796 \cdot 10^6$
7	1	5	25	125	625
8	1	7.33	53.729	393.833	$2.887 \cdot 10^3$
9	1	5.61	31.472	176.558	990.493
10	1	0	0	0	0
11	1	0	0	0	0
12	1	0	0	0	0
13	1	10.71	114.704	$1.228 \cdot 10^3$	$1.316 \cdot 10^4$
14	1	10.71	114.704	$1.228 \cdot 10^3$	$1.316 \cdot 10^4$

Рисунок 5.6 – Ймовірність часу знаходження вузлу у стані очікування обробки

Далі розташовано блок ПЗ, необхідний далі розрахунку середніх значень показників з інтенсивності вхідного потоку (рис. 5.7…рис. 5.9).

$\lambda =$

	0
0	$6.259 \cdot 10^{-3}$
1	0.031
2	0.064
3	0.031
4	0.018
5	0.051
6	0.026
7	$3.129 \cdot 10^{-3}$
8	$4.588 \cdot 10^{-3}$
9	$3.511 \cdot 10^{-3}$
10	0
11	0
12	0
13	$6.703 \cdot 10^{-3}$
14	$6.703 \cdot 10^{-3}$

Рисунок 5.7 – Інтенсивність вхідного потоку

$L =$

	0
0	0.071
1	0.45
2	1.294
3	0.437
4	0.221
5	0.904
6	0.348
7	0.034
8	0.051
9	0.039
10	0
11	0
12	0
13	0.076
14	0.081

Рисунок 5.8 – Кількість пакетів очікування обробки

$t =$

	0
0	11.274
1	14.391
2	20.219
3	14.291
4	12.556
5	17.67
6	13.59
7	10.954
8	11.102
9	10.992
10	0
11	0
12	0
13	11.322
14	12.03

Рисунок 5.9 – Середній час обробки пакету

5.2.2 Моделювання роботи мережі

Моделювання роботи комп'ютерної системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT, створеної в MATCAD, в режимі з запланованого навантаженням має наступні параметри налаштування:

- кількість пакетів: 5;

- час обробки пакету: для всіх вузлів однаковий і становить 10 одиниць, що відповідає 100 Мб обладнанню;

- кількість конвеєрів потокової інформації: 1 одиниця.

Графічний результат моделювання роботи мережі в режимі запланованого інформаційного навантаження продемонстровано на рис. 5.10…5.13.

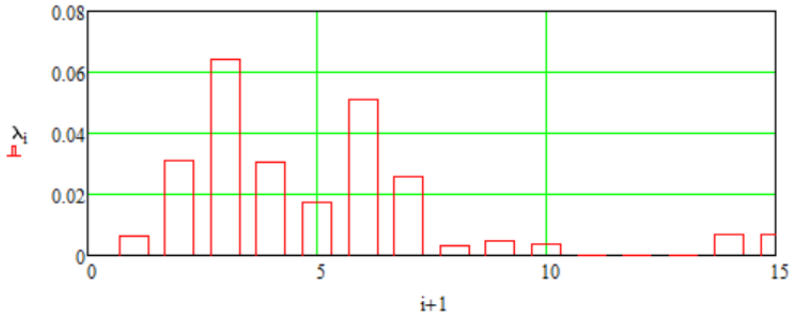


Рисунок 5.10 – Інтенсивність вхідного потоку

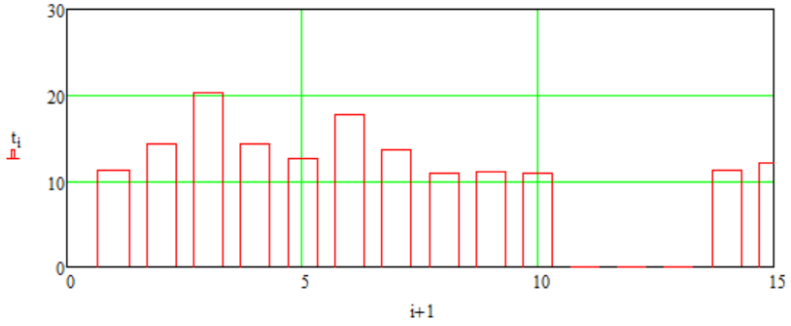


Рисунок 5.11 – Середній час перебування пакета

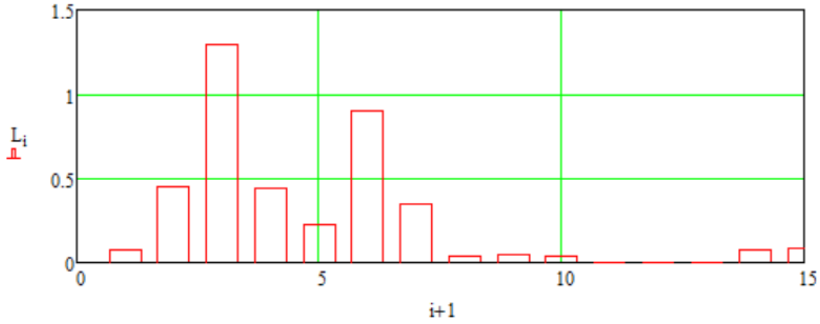


Рисунок 5.12 – Середня кількість пакетів

З цих графіків видно показники роботи мереж для всіх комутаторів та маршрутизаторів комп'ютерної системи протипожежного захисту розумного будинку.

Як гарно видно з рис. 5.13 для стану мережі при довжині повідомлень у 5 пакетів обробка повідомлень є досить швидкою для всіх вузлів зокрема вузлів 3 та 6, ці вузли є проблемним з точки зору інформаційного навантаження на них.

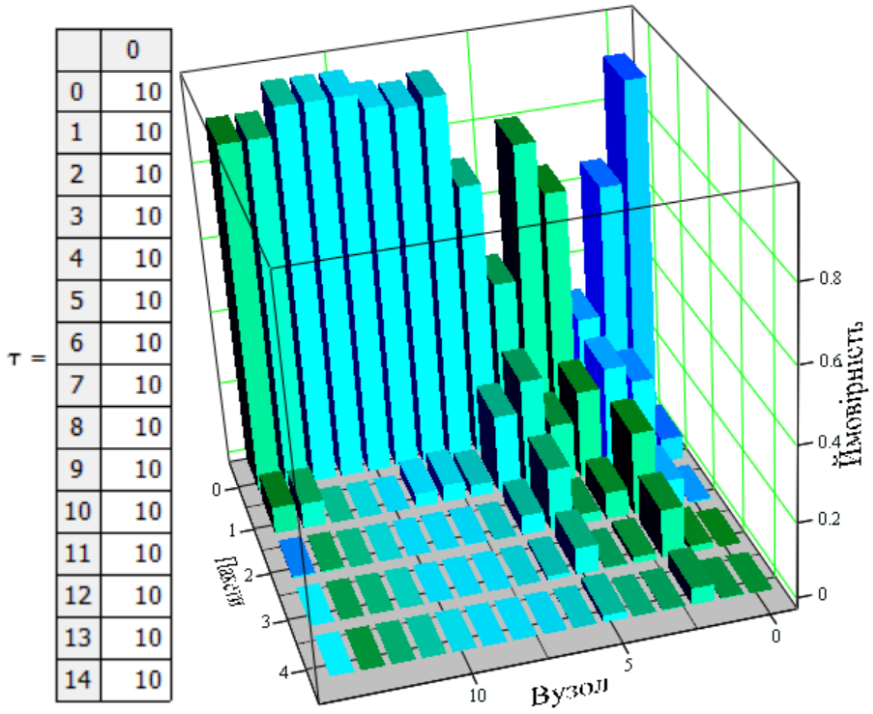


Рисунок 5.13 – Вірогідність черги при номінальному навантаженні (5 пакетів)

Для вирішення цієї проблеми з вузлами 3 та 6, замінимо для них елементу базу - на швидкісне обладнання з пропускною здатністю у 1 Гб (рис. 514). Корекція параметрів вузлів 3 та 6 поліпшила для них картину зі станом інформаційного навантаження на них. Але тепер погіршився стан для вузлів 2, 4, 5 та 7. Тепер поліпшити стан з інформаційним навантаженням на ці вузли. Для цього використовуємо такі замі заходи як і для вузлів 3 та 6. Після проведених маніпуляцій з вузлами 2…7 стан інформаційного навантаження на них поліпшився і зрівнявся з іншими вузлами моделі. Тепер вірогідність черги у два пакети не перевищує значення 0,2 (20%). Ця ситуація стану моделі показана на рис. 5.15.

τ =

	0
0	10
1	10
2	1
3	10
4	10
5	1
6	10
7	10
8	10
9	10
10	10
11	10
12	10
13	10
14	10

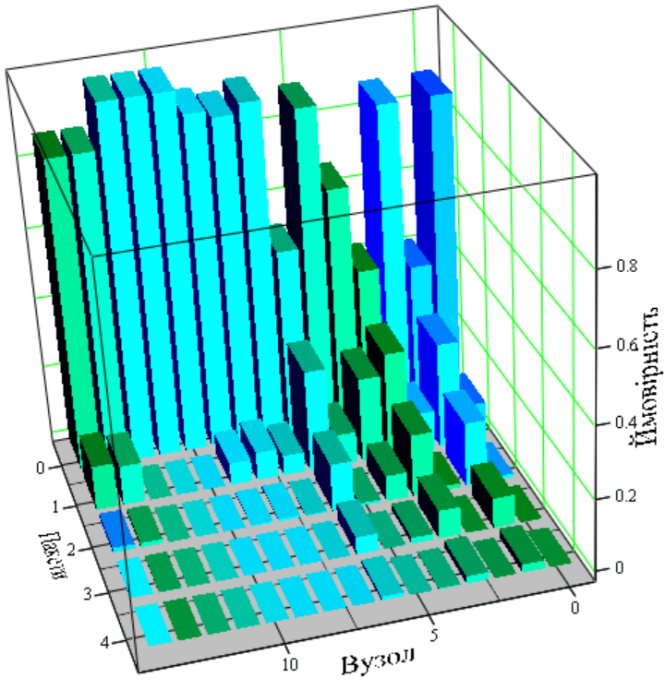


Рисунок 5.14 – Вірогідність черги при номінальному навантаженні (5 пакетів), та поліпшених параметрів для вузлів 3 та 6

τ =

	0
0	10
1	1
2	1
3	1
4	1
5	1
6	1
7	10
8	10
9	10
10	10
11	10
12	10
13	10
14	10

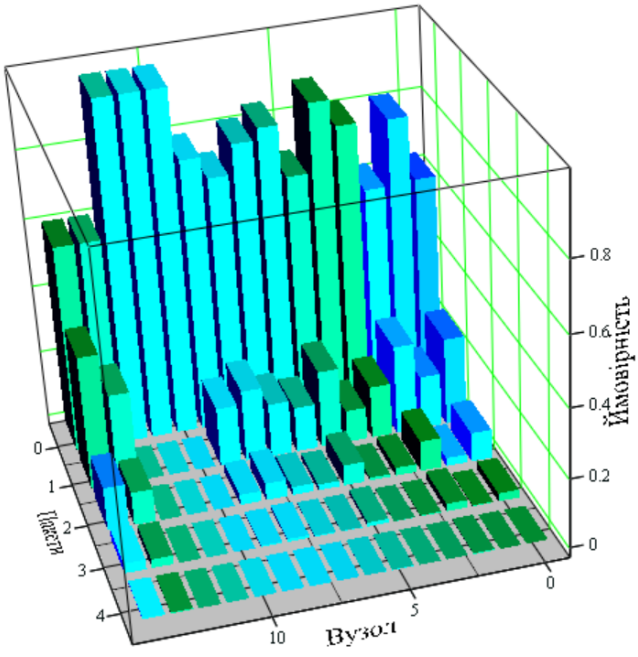


Рисунок 5.15 – Вірогідність черги при номінальному навантаженні (5 пакетів), та поліпшених параметрів для вузлів 2…7

На цьому можна завершувати налаштування мережі, модель знаходиться у гарному стані. Скорегована структурна схема моделі комп'ютерної системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT наведена на рис. 5.16

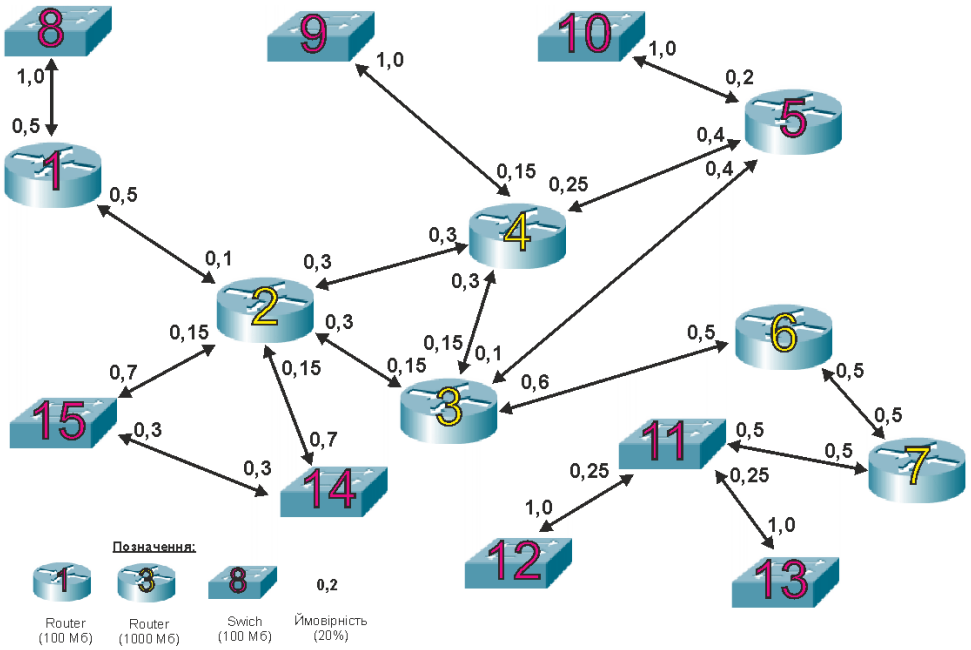


Рисунок 5.16 – Модель комп'ютерної системи протипожежного захисту розумного будинку с поліпшеними параметрами вузлів 2…7

Для любої комп’ютерної мережі важлива мати запас міцності по інформаційному навантаженню, працювати в умовах значно збільшеного обміну інформації. Запас міцності по інформаційному навантаженню забезпечить працездатність комп'ютерів кінцевих користувачів роботу майже у звичайному режимі і вони не відчуватимуть значного збільшення трафіку в мережі.

5.2.3 Запас міцності мережі по інформаційному навантаженню

Тепер промоделюємо роботу комп'ютерної системи протипожежного захисту розумного будинку в умовах різкого збільшення трафіку.

5.2.3.1 Причини збільшення інформаційного трафіку в мережі

Коли мережа перевантажена, швидкість обміну інформації сповільнюється до «повзання», що створює відчуття млявості роботи ПЗ комп’ютера.

Так само, як і трафік у годину пік, у мережах також можуть траплятися аварії Незалежно від того, чи це несправний кабель, неправильно налаштований маршрутизатор або шахрайський пристрій, який перевантажує всю пропускну здатність, аварії можуть швидко призвести до перевантаження мережі.

Перевантаження мережі може статися з різних причин. Ось деякі з найпоширеніших причин:

1. Великий обсяг трафіку. Великий обсяг трафіку може спричинити перевантаження мережі, оскільки мережі мають обмежену пропускну здатність, доступну для передачі даних. Коли обсяг даних, що передаються мережею, перевищує доступну пропускну здатність, це може призвести до перевантаження.

Якщо через мережу намагається протікати занадто багато даних і недостатньо пропускної здатності для їх обробки, мережа сповільниться і стане перевантаженою.

Великий обсяг трафіку може бути викликаний різними факторами, включаючи підвищений попит на програми з інтенсивним об'ємом даних, такі як потокове відео, обмін файлами та хмарні служби. Чим більше пристроїв підключаються до мережі та споживають пропускну здатність, тим більший ризик перевантаження.

2. Застаріле обладнання. По-перше, старе обладнання може бути не в змозі впоратися з більш високими швидкостями та пропускною здатністю, які необхідні для підтримки сучасних додатків і служб. Це може призвести до вузького місця мережі, коли дані сповільнюються, коли вони проходять через застаріле обладнання, що спричиняє перевантаження.

По-друге, застарілому обладнанню може не вистачати функцій, необхідних для визначення пріоритетів та ефективного управління мережевим трафіком. Наприклад, він може не підтримувати політику якості обслуговування (QoS), яка дозволяє мережевим адміністраторам надавати пріоритет певним типам трафіку над іншими на основі їх важливості. Це може призвести до перевантаження, коли пропускна здатність споживається трафіком із нижчим пріоритетом, що спричиняє затримки та зниження продуктивності мережі для критично важливих програм, таких як QoS для VoIP.

По-третє, застаріле обладнання може бути більш сприйнятливим до помилок і збоїв, що може сприяти перевантаженню мережі. Наприклад, комутатор або маршрутизатор із застарілою прошивкою може з більшою ймовірністю скидати пакети або зазнавати інших помилок, що може спричинити перевантаження, змушуючи пристрої повторно передавати втрачені дані.

Нарешті, застарілому обладнанню може не вистачати функцій безпеки, необхідних для захисту від кіберзагроз. У деяких випадках це може призвести до перевантаження, спричиненого атаками типу «відмова в обслуговуванні» (DoS), які переповнюють мережу трафіком і перевантажують застаріле обладнання.

Загалом, застаріле обладнання може бути значним фактором перевантаження мережі. Ось чому важливо регулярно оцінювати та оновлювати мережеве обладнання, щоб переконатися, що воно може відповідати вимогам сучасних програм і служб.

3. Недостатня пропускна здатність. Недостатня пропускна здатність є однією з найпоширеніших причин перевантаження мережі або перевантаження мережі. Якщо мережа має недостатню пропускну здатність, це означає, що доступної пропускної здатності мережі недостатньо для задоволення потреб пристроїв і програм, які її використовують.

Обсяг пропускної здатності, необхідний для конкретної мережі, залежить від безлічі факторів, включаючи кількість підключених пристроїв, типи використовуваних додатків і обсяг даних, що передаються. Якщо потреба в пропускній здатності перевищує доступну пропускну здатність, це може призвести до перевантаження.

Наприклад, якщо кілька користувачів транслюють відео високої чіткості або передають великі файли одночасно, це може швидко споживати доступну пропускну здатність і спричиняти перевантаження. Результатом є низька продуктивність мережі, збільшення затримки та втрата пакетів.

Недостатня пропускна здатність також може спричинити перевантаження у висхідному або низхідному напрямку мережі. Наприклад, якщо компанія має повільне інтернет-з'єднання, вона може бути не в змозі завантажувати або вивантажувати дані достатньо швидко, щоб задовольнити потреби своїх користувачів. Це може спричинити перевантаження мережі, що призведе до затримок і зниження продуктивності.

4. Проблеми з конфігурацією мережі. Проблеми з конфігурацією мережі можуть бути основною причиною перевантаження мережі. Конфігурація мережі

відноситься до параметрів і параметрів, які використовуються для визначення того, як пристрої та програми взаємодіють один з одним через мережу.

Якщо мережеві пристрої налаштовані неправильно, це може спричинити низку проблем, які можуть призвести до перевантаження. Наприклад, якщо маршрутизатор неправильно налаштований і використовує не найефективніший протокол маршрутизації, він може бути не в змозі пересилати пакети досить швидко, що спричиняє перевантаження в мережі.

Аналогічно, якщо мережеві пристрої неправильно налаштовані на використання політик якості обслуговування (QoS), це може призвести до перевантаження, коли трафік із низьким пріоритетом використовує занадто багато пропускної здатності, що спричиняє затримки та зниження продуктивності мережі для критично важливих програм.

Крім того, помилки конфігурації також можуть спричинити проблеми з безпекою мережі, що може сприяти перевантаженню. Наприклад, якщо брандмауер неправильно налаштований і не блокує шкідливий трафік, це може призвести до атак типу «відмова в обслуговуванні» (DoS), які переповнюють мережу трафіком і спричиняють перевантаження.

5 Мережеві атаки. Мережеві атаки можуть бути основною причиною перевантаження мережі, особливо якщо вони пов'язані з розподіленими атаками типу «відмова в обслуговуванні» (DDoS). Під час DDoS-атаки велика кількість скомпрометованих пристроїв, які часто називають «ботнетом», використовується для наповнення цільової мережі трафіком, перевантажуючи мережу та викликаючи перевантаження.

Захист від DDoS-атак може бути особливо складним, оскільки трафік атаки часто здається законним трафіком, що ускладнює для пристроїв мережевої безпеки розрізнення цих двох. Великий обсяг трафіку, що генерується DDoS-атакою, також може насичувати доступну пропускну здатність мережі, що призводить до перевантаження та зниження продуктивності законного трафіку.

Інші типи мережевих атак також можуть сприяти перевантаженню, особливо ті, які передбачають переповнення мережі трафіком. Наприклад, атака Ping флуда передбачає надсилання великої кількості запитів ICMP echo на цільовий пристрій,

що призводить до його перевантаження та втрати відповіді. Аналогічно, атака SYN-флуду передбачає надсилання великої кількості SYN-запитів на цільовий пристрій, споживання його ресурсів і спричинення перевантаження.

6. Непотрібний трафік. Непотрібний трафік - це пакети даних або інформація, які передаються мережею, але не сприяють продуктивній або запланованій роботі мережі. Хоча кожна мережа має певну здатність обробляти дані, надмірний непотрібний трафік може спричинити перевантаження мережі, що може призвести до зниження продуктивності та проблем зі зв'язком.

Ось кілька способів, як непотрібний трафік може спричинити перевантаження мережі:

Споживання пропускної здатності: непотрібний трафік споживає цінну пропускну здатність мережі, яку можна використовувати для передачі основних даних. Коли доступна пропускна здатність зайнята несуттєвим або надмірним трафіком, це залишає менше ємності для важливих даних, що призводить до зниження швидкості передачі та затримок.

Вичерпання ресурсів: мережеві пристрої, такі як маршрутизатори та комутатори, мають обмежену обчислювальну потужність та пам'ять. Коли непотрібний трафік переповнює мережу, ці пристрої повинні обробляти та справлятися з надмірним навантаженням, потенційно вичерпуючи наявні ресурси. Як наслідок, мережевим пристроям може бути важко встигати за попитом, що призводить до перевантаження та проблем із продуктивністю.

Збільшення колізій: У спільних мережевих середовищах, таких як мережі Ethernet, непотрібний трафік може призвести до збільшення кількості колізій. Зіткнення виникають, коли кілька пристроїв намагаються передати дані одночасно, що призводить до втрати та повторної передачі даних. Коли присутній непотрібний трафік, це може посилити рівень зіткнень, спричиняючи перевантаження та знижуючи ефективність мережі.

Переповнення буфера: мережеві пристрої використовують буфери для тимчасового зберігання вхідних пакетів даних, коли приймальна сторона зайнята або не може обробити їх негайно. Однак надмірний непотрібний трафік може заповнити ці буфери, що призведе до переповнення буфера. Коли буфери

переповнюються, нові пакети можуть бути відкинуті або затримані, що спричинить перевантаження та зниження продуктивності.

Неефективність маршрутизації: Непотрібний трафік може вплинути на ефективність протоколів і механізмів маршрутизації. Маршрутизатори використовують різні алгоритми для визначення найкращого шляху передачі даних. Однак, якщо присутній непотрібний трафік, маршрутизатори можуть витрачати час і ресурси на спроби обробити та перенаправити цей трафік, що призведе до неоптимальних рішень щодо маршрутизації. Ця неефективність може призвести до збільшення затримки та перевантаження всієї мережі [13].

5.2.3.2 Робота мережі під дією вірусного ПЗ

Припустимо, що інформаційне навантаження на мережу зросло в двічі під дією вірусного ПЗ і становить 10 інформаційних пакетів. Змоделюємо роботу комп'ютерної системи протипожежного захисту розумного в цих умовах.

Результати моделювання роботи зі збільшеною вдвічі кількістю пакетів приводить до того, що вузли 1, 3, 14 та 15 починають мати більше інформаційне навантаження ніж інші мережі (рис. 5.17).

$\tau =$

	0
0	10
1	1
2	1
3	1
4	1
5	1
6	1
7	10
8	10
9	10
10	10
11	10
12	10
13	10
14	10

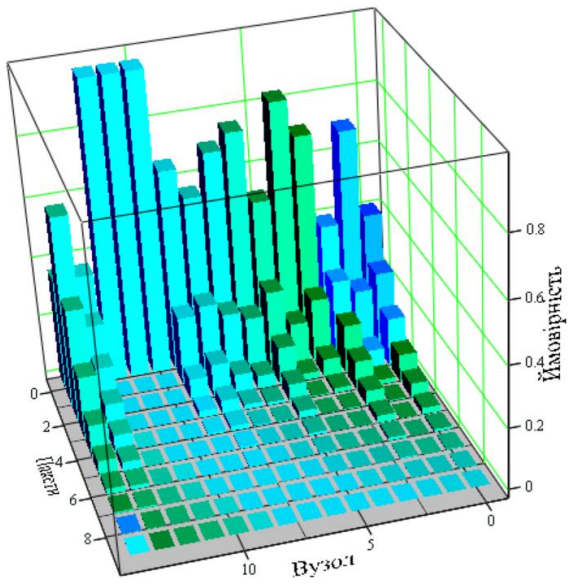


Рисунок 5.17 – Вірогідність черги у вузлах під дією шкідливого ПЗ (10 пакетів)

Вузол 3 вже має максимальну передачі інформації у 1 Гб тому для нього треба застосувати обладнання з більшою кількістю конвеєрів обробки потокової інформації. В вузлах 1, 14 та 15 треба застосувати обладнання зі швидкістю у 1 Гб, замість 100 Мб. Стан мережі для скорегованої мережі показано на рис. 5.18.

τ =

	0
0	1
1	1
2	1
3	1
4	1
5	1
6	1
7	10
8	1
9	1
10	10
11	10
12	10
13	1
14	1

m =

	0
0	1
1	1
2	5
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1

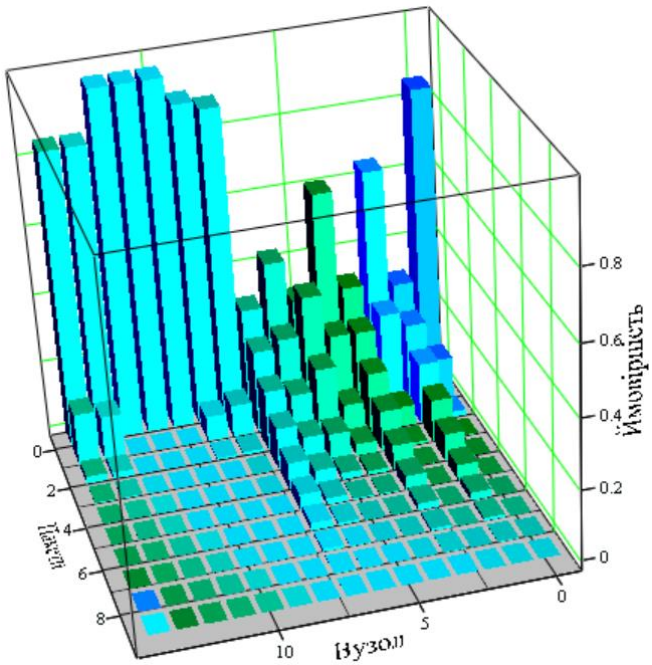


Рисунок 5.18 – Вірогідність черги у вузлах під дією шкідливого ПЗ для скорегованої мережі

Як бачимо для цього випадку налаштувань стан мережі є задовільним, вірогідність черги з двох пакетів не перевищує показник 0,2 (20 %).

Навіть під дією шкідливого вірусного ПЗ, яке значно підвищую інформаційне навантаження на мережу мережа все одно буде зберігати заплановану пропускну здатність.

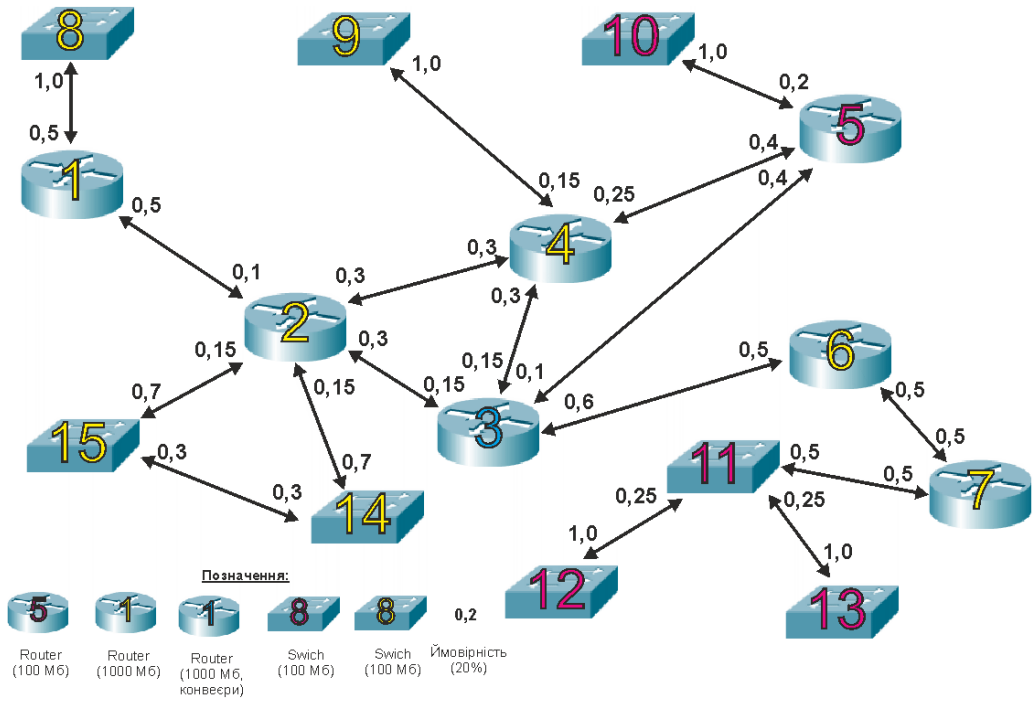


Рисунок 5.19 – Остаточно математична модель комп’ютерної мережі

5.3 Висновки по розділу

Розроблена модель комп'ютерної мережі протипожежного захисту розумного будинку здійснює моделювання інтенсивності навантаження на мережу та дозволяє отримувати час перебування інформаційних пакетів в вузлах. Моделювання роботи моделі мережі з послідовною зміною параметрів мережевих вузлів та порівнянням отриманих параметрів привело до задовільного стану налаштування її показників.

Можна зробити висновок, що навіть при значному підвищенні трафіку в умовах дії вірусного ПЗ, стан мережі буде мати задовільні показники зі швидкості роботи.

ВИСНОВКИ

Магістерська робота є закінченою науковою працею, в якій вирішена науково-практична задача з синтезу програмно-технічної реалізації схеми комп'ютерної мережі протипожежного захисту розумного будинку.

Результати роботи:

1. Функціональні особливості роботи комп’ютерної мережі протипожежного захисту розумного будинку навіть під дію подвійного інформаційного навантаження за умов дією вірусного ПЗ не приводить до втрати працездатності комп’ютерної мережі.

2. Дослідження роботи комп'ютерної мережі протипожежного захисту розумного будинку було проведено з застосуванням методів теорії масового обслуговування. На відміну від дослідження з застосуванням апаратного забезпечення не є фінансово затратним та значно скорочую час дослідження.

3. Моделювання комп'ютерної мережі протипожежного захисту розумного будинку здійснювалося для замкнутої мережевої системи.

4. Математична модель комп'ютерної мережі протипожежного захисту розумного будинку здатна визначати усереднені показники роботи мережі - час перебування і середній показник з кількості інформаційних пакетів. Розроблена модель мережі має безрозмірні показники для мережевих пристроїв, які пропорційно відповідають реальним технічним показникам мережевих пристроїв.

5. При дії вірусного ПЗ ймовірність виникнення черги з двох інформаційних пакетів в мережевих пристроях не перевищує 20 %. Такий результат було застосуванню більш швидкісного мережевого обладнання для проблемних вузлів мережі.

6. Можна рекомендувати налаштування комп'ютерної мережі протипожежного захисту розумного будинку за параметрами скорегованої моделі.

ПЕРЕЛІК ПОСИЛАНЬ

1. Kyas O. To Smart Home: A Step by Step Guide for Smart Homes & Building Automation. – New York: Key Concept Press, 2017. – 337 p.Оглад мережевого обладнання Cisco. Режим доступу: https://dobrovdome.ru/catalog/as-2014

2. Глоба Л.С. Розподілені системи та мережі. Том 1. Підручник. – К.: Політехніка, 2013. – 378 с.

3. Лебедь О.О., Мислінчук В.О., Пастушенко В.Й. Фізичні основи комп'ютерно-інтегрованих інформаційних систем. Навчальний посібник. – Рівне : НУВГП, 2015. – 352 с.

4. Рамський Ю.С., Олексюк В.П., Балик А.В. Адміністрування комп'ютерних мереж і систем: Навч. пос. – Тернопіль: Навчальна книга – Богдан, 2015. – 196 с.

5. Розробка програмного забезпечення комп'ютерних систем. Програмування: навч. посібник / Л.І. Цвіркун, А.А. Євстігнєєва, Я.В. Панферова. – 2-ге вид., випр. – Д.: Національний гірничий університет, 2011. – 222 с.

6. Жуков, І. А. Комп'ютерні мережі та технології : навч. посіб./І. А. Жуков, В. О. Гуменюк, І. Є. Альтман. – К. : НАУ, 2004. – 276 с.

7. Goodwin S. Smart Home Automation with Linux and Raspberry Pi. Second Edition. – Apress Media, 2013. – 328 p

8. Колонтаєвський, Ю. П. Електроніка і мікросхемотехніка [Текст]: підручник для студентів вузів, 2-е вид. / Ю. П. Колонтаєвський, А. Г. Сосков; за ред. докт. техн. наук, проф. А.Г. Соскова. – К.: Каравела, 2009. – 416 с.

9. Голев Д.В., Кононович В.Г., Хомич С.В. Методики оцінки інформаційної захищеності телекомунікацій. Навчальний посібник. – Одеса : ОНАЗ ім. О. С. Попова, 2013. – 217 с.

10. Queueing models for multi-service networkschrome-extension/ Режим доступу: //efaidnbmnnnibpcajpcglclefindmkaj/https://publik.tuwien.ac.at/files/PubDat_238517.pdf

11. Developer Enterprise Streaming Telemetry and You: Getting Started with Model Driven Telemetry. Режим доступу: https://blogs.cisco.com/developer/getting-started-with-model-driven-telemetry

12. How to Avoid The Network Traffic Jam: What is Network Congestion and How to Fix It. Режим доступу: ttps://obkio.com/blog/what-is-network-congestion/

13. Котеджне містечко Concept Riviera, Дніпро. Режим доступу: https://novobudovy.com/kotedzhni-mistechka-dnipropetrovska/concept-riviera

14. Гребенніков В.В. Комплексні системи захисту інформації: проектування, впровадження, супровід. Ужгород: Ужгородський національний університет, 2013. – 161 с.

15. Arduino Mega 2560. Режим доступу: https://doc.arduino.ua/ru/hardware/Mega2560

16. Одноплатний комп’ютер / Raspberry Pi 4 model B. Режим доступу: https://www.tp-link.com/ru/home-networking/wifi-router/archer-ax53/
14.https://york.rv.ua/2020/12/raspberry-pi-4-model-b/

17. Категорія «Розумний дім». Режим доступу: / Що таке «Розумний дім»? http://www.dom-electro.ru/что-такоеумный-дом/

18. What is MQTT and Why it is Important for the Internet of Things Режим доступу: https://blog.akenza.io/what-is-mqtt?utm_term=&utm_campaign=Dynamic+Search+Campaign+-+EU&utm_source=google&utm_medium=cpc&hsa_acc=7184580909&hsa_cam=19248438136&hsa_grp=144426835677&hsa_ad=641301872801&hsa_src=g&hsa_tgt=dsa-19959388920&hsa_kw=&hsa_mt=&hsa_net=adwords&hsa_ver=3&gad_source=1&gclid=CjwKCAiA1fqrBhA1EiwAMU5m_we2tY4kDFK_4CaD2NXYoeo0jdRd9eh2qEmMoVk4h2D28S_xfcOEsxoCj9AQAvD_BwE

ДОДАТОК А

Текст програми

Програмно-технічна реалізація комп'ютерної системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“ДНІПРОВСЬКА ПОЛІТЕХНІКА”

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ПРОГРАМНО-ТЕХНІЧНА РЕАЛІЗАЦІЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ПРОТИПОЖЕЖНОГО ЗАХИСТУ РОЗУМНОГО БУДИНКУ З ВІДДАЛЕНИМ КОНТРОЛЕМ ЧЕРЕЗ СЕРВЕР MQTT

Текст програми

804.02070743.22017-01 12 01

Листів 16

2023

АНОТАЦІЯ

Програмний документ містить ПЗ для реалізації моделі комп'ютерної системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT.

ПЗ реалізовано в середовищі Mathcad 15 під дією операційної системи Windows 10.

ПЗ, розрахунку параметрів комп'ютерної системи протипожежного захисту розумного будинку з віддаленим контролем через сервер MQTT як замкнутої системи масового обслуговування, застосовує метод Бузена для.

ЗМІСТ

1 Перелік використаних змінних, та переклад коментарів

Nn – кількість вузлів;

τ – час обробки;

Pr – матриця ймовірностей.

e – матриця коефіцієнтів.

m – кількість конвеєрів;

N – кількість пакетів;

B – матриця ймовірності черг;

λ –інтенсивність запитів на вході вузла;

L –черга пакетів до вузлу;

t –час перебування пакету в вузу;

Average time spent in the node - середній час перебування інформаційного пакету в мережевому вузлу;

Calculation of function A - розрахунок функції A;

Calculation of the matrix of constants T - розрахунок матриці констант T4

Calculate the constants for the second and subsequent nodes - обчислення константи для другого і наступних мережевих вузлів;

Calculation of the intensity of query processing in network nodes - обчислення інтенсивності обробки запитів в мережевих вузлах;

Calculation of auxiliary coefficients – обчислення допоміжних коефіцієнтів;

Calculation of probabilities of receipt in the last node of applications – j – обчислення ймовірності надходження в останньому мережевому вузлу заявок – j4

Determination of transmission coefficients - розрахунок коефіцієнтів передачі;

Matrix of transfer coefficients - матриця передаточних коефіцієнтів;

Number of nodes in the network - кількість мережевих вузлів.

Transfer matrix - передаточна матриця;

The number of packets circulating in the network - кількість інформаційних пакетів, які циркулюють в мережі;

The number of pipelines in each node - кількість конвеєрів поточної обробки в кожному мережевому вузлі;

The intensity of the input stream - інтенсивність вхідного інформаційного потоку;

The average number of packets per node - середня кількість інформаційних пакетів в мережевому вузлі;

Packet processing time at the node - час обробки інформаційного пакета у мережевому вузлу.

2 Текст програми

Задаємо кількість вузлів мережі 15 (Nn) $Nn := 14$

$i := 0..Nn \qquad j := 0..Nn$

Задаємо час обробки запиту в вузлах мережі (1-1000Мб, 10-100Мб):

$\tau_i :=$

10
10
10
10
10
10
10
10
10
10
10
10
10
10
10

$\tau =$

	0
0	10
1	10
2	10
3	10
4	10
5	10
6	10
7	10
8	10
9	10
10	10
11	10
12	10
13	10
14	10

Розрахунок інтенсивності обробки запитів в вузлах мережі

$\mu_i := \frac{1}{\tau_i}$

$\mu =$

	0
0	0.1
1	0.1
2	0.1
3	0.1
4	0.1
5	0.1
6	0.1
7	0.1
8	0.1
9	0.1
10	0.1
11	0.1
12	0.1
13	0.1
14	0.1

Задаємо матрицю імовірностей передачі (Pr):

$$Pr := \begin{pmatrix} 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.1 & 0 & 0.3 & 0.3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.15 & 0.15 \\ 0 & 0.15 & 0 & 0.15 & 0.1 & 0.6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.3 & 0.3 & 0 & 0.25 & 0 & 0 & 0 & 0.15 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.40 & 0.4 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0.7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.3 \\ 0 & 0.7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.3 & 0 \end{pmatrix}$$

Перевірка правильності заповнення передаточної матриці

$$SumPr_i := \sum_{j=0}^{Nn} Pr_{i,j}$$

$SumPr =$

	0
0	1
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1

Розрахунок коефіцієнтів передачі

$P := Pr^{T}$

$P =$

	0	1	2	3	4	5	6	7	8	9
0	0	0.1	0	0	0	0	0	1	0	0
1	0.5	0	0.15	0.3	0	0	0	0	0	0
2	0	0.3	0	0.3	0.4	0.5	0.5	0	0	0
3	0	0.3	0.15	0	0.4	0	0	0	1	0
4	0	0	0.1	0.25	0	0	0	0	0	1
5	0	0	0.6	0	0	0	0.5	0	0	0
6	0	0	0	0	0	0.5	0	0	0	0
7	0.5	0	0	0	0	0	0	0	0	0
8	0	0	0	0.15	0	0	0	0	0	0
9	0	0	0	0	0.2	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0
13	0	0.15	0	0	0	0	0	0	0	0
14	0	0.15	0	0	0	0	0	0	0	...

Створюємо діаганальну матрицю (D):

$D :=$

$$\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}$$

$P1 := P - D$

P1 =

	0	1	2	3	4	5	6	7	8	9
0	-1	0.1	0	0	0	0	0	1	0	0
1	0.5	-1	0.15	0.3	0	0	0	0	0	0
2	0	0.3	-1	0.3	0.4	0.5	0.5	0	0	0
3	0	0.3	0.15	-1	0.4	0	0	0	1	0
4	0	0	0.1	0.25	-1	0	0	0	0	1
5	0	0	0.6	0	0	-1	0.5	0	0	0
6	0	0	0	0	0	0.5	-1	0	0	0
7	0.5	0	0	0	0	0	0	-1	0	0
8	0	0	0	0.15	0	0	0	0	-1	0
9	0	0	0	0	0.2	0	0	0	0	-1
10	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0
13	0	0.15	0	0	0	0	0	0	0	0
14	0	0.15	0	0	0	0	0	0	0	...

$j := 1 .. Nn$ $i := 0 .. Nn$

$P2_{(j-1),i} := P1_{0,i} + P1_{j,i}$

P2 =

	0	1	2	3	4	5	6	7	8	9
0	-0.5	-0.9	0.15	0.3	0	0	0	1	0	0
1	-1	0.4	-1	0.3	0.4	0.5	0.5	1	0	0
2	-1	0.4	0.15	-1	0.4	0	0	1	1	0
3	-1	0.1	0.1	0.25	-1	0	0	1	0	1
4	-1	0.1	0.6	0	0	-1	0.5	1	0	0
5	-1	0.1	0	0	0	0.5	-1	1	0	0
6	-0.5	0.1	0	0	0	0	0	0	0	0
7	-1	0.1	0	0.15	0	0	0	1	-1	0
8	-1	0.1	0	0	0.2	0	0	1	0	-1
9	-1	0.1	0	0	0	0	0	1	0	0
10	-1	0.1	0	0	0	0	0	1	0	0
11	-1	0.1	0	0	0	0	0	1	0	0
12	-1	0.25	0	0	0	0	0	1	0	0
13	-1	0.25	0	0	0	0	0	1	0	...

$j := 0..Nn - 1 \qquad i := 0..Nn - 1 \qquad PP2_{j,i} := P2_{j,i+1}$

PP2 =

	0	1	2	3	4	5	6	7	8	9
0	-0.9	0.15	0.3	0	0	0	1	0	0	0
1	0.4	-1	0.3	0.4	0.5	0.5	1	0	0	0
2	0.4	0.15	-1	0.4	0	0	1	1	0	0
3	0.1	0.1	0.25	-1	0	0	1	0	1	0
4	0.1	0.6	0	0	-1	0.5	1	0	0	0
5	0.1	0	0	0	0.5	-1	1	0	0	0.5
6	0.1	0	0	0	0	0	0	0	0	0
7	0.1	0	0.15	0	0	0	1	-1	0	0
8	0.1	0	0	0.2	0	0	1	0	-1	0
9	0.1	0	0	0	0	0	1	0	0	-1
10	0.1	0	0	0	0	0	1	0	0	0.25
11	0.1	0	0	0	0	0	1	0	0	0.25
12	0.25	0	0	0	0	0	1	0	0	0
13	0.25	0	0	0	0	0	1	0	0	...

$Q_{j,0} := P2_{j,0}$

Q =

	0
0	-0.5
1	-1
2	-1
3	-1
4	-1
5	-1
6	-0.5
7	-1
8	-1
9	-1
10	-1
11	-1
12	-1
13	-1

$E := lsolve(PP2, Q)$

E =

	0
0	-5
1	-10.222
2	-4.889
3	-2.806
4	-8.178
5	-4.089
6	-0.5
7	-0.733
8	-0.561
9	0
10	0
11	0
12	-1.071
13	-1.071

Створюємо матрицю коефіцієнтів (e):

$$e := \begin{pmatrix} 1 \\ 5 \\ 10.222 \\ 4.889 \\ 2.806 \\ 8.178 \\ 4.089 \\ 0.5 \\ 0.733 \\ 0.561 \\ 0 \\ 0 \\ 0 \\ 1.071 \\ 1.071 \end{pmatrix}$$

Задаємо кількість пакетів які циркулюють в мережі 5 - нормальний режим, 10 - з вірусами (N):

$N := 5$

$i := 0 .. Nn$

$j := 0 .. N - 1$

Задаємо кількість конвеєрів в кожному вузлі (m_i):

$m_i :=$

1
1
1
1
1
1
1
1
1
1
1
1
1
1
1

$m =$

	0
0	1
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1

Розрахунок значень функції A

$$A_{i,j} := \begin{cases} j! & \text{if } m_i \geq N - 1 \\ 1 & \text{if } m_i = 1 \\ j! & \text{if } 1 < m_i < N - 1 \wedge j \leq m_i \\ m_i! \cdot (m_i)^{j - m_i} & \text{if } 1 < m_i < N - 1 \wedge j > m_i \end{cases}$$

A =

	0	1	2	3	4
0	1	1	1	1	1
1	1	1	1	1	1
2	1	1	1	1	1
3	1	1	1	1	1
4	1	1	1	1	1
5	1	1	1	1	1
6	1	1	1	1	1
7	1	1	1	1	1
8	1	1	1	1	1
9	1	1	1	1	1
10	1	1	1	1	1
11	1	1	1	1	1
12	1	1	1	1	1
13	1	1	1	1	1
14	1	1	1	1	1

$$X_i := \frac{e_i}{\mu_i}$$

X =

	0
0	10
1	50
2	102.22
3	48.89
4	28.06
5	81.78
6	40.89
7	5
8	7.33
9	5.61
10	0
11	0
12	0
13	10.71
14	10.71

Обчислення матриці констант T

$$T_{i,j} := \frac{(X_i)^j}{A_{i,j}} \qquad T_{i,0} := 1$$

T =

	0	1	2	3	4
0	1	10	100	$1 \cdot 10^3$	$1 \cdot 10^4$
1	1	50	$2.5 \cdot 10^3$	$1.25 \cdot 10^5$	$6.25 \cdot 10^6$
2	1	102.22	$1.045 \cdot 10^4$	$1.068 \cdot 10^6$	$1.092 \cdot 10^8$
3	1	48.89	$2.39 \cdot 10^3$	$1.169 \cdot 10^5$	$5.713 \cdot 10^6$
4	1	28.06	787.364	$2.209 \cdot 10^4$	$6.199 \cdot 10^5$
5	1	81.78	$6.688 \cdot 10^3$	$5.469 \cdot 10^5$	$4.473 \cdot 10^7$
6	1	40.89	$1.672 \cdot 10^3$	$6.837 \cdot 10^4$	$2.796 \cdot 10^6$
7	1	5	25	125	625
8	1	7.33	53.729	393.833	$2.887 \cdot 10^3$
9	1	5.61	31.472	176.558	990.493
10	1	0	0	0	0
11	1	0	0	0	0
12	1	0	0	0	0
13	1	10.71	114.704	$1.228 \cdot 10^3$	$1.316 \cdot 10^4$
14	1	10.71	114.704	$1.228 \cdot 10^3$	$1.316 \cdot 10^4$

Розраховуєм константи для другого і наступних вузлів

$i := 1 .. Nn \qquad k := 0 .. N - 1$

$$G_{0,j} := T_{0,j}$$

$$G_{i,k} := \sum_{j=0}^{k} \left(T_{i,j} \cdot G_{i-1,k-j} \right)$$

$G =$

	0	1	2	3	4
0	1	10	100	$1 \cdot 10^3$	$1 \cdot 10^4$
1	1	60	$3.1 \cdot 10^3$	$1.56 \cdot 10^5$	$7.81 \cdot 10^6$
2	1	162.22	$1.968 \cdot 10^4$	$2.168 \cdot 10^6$	$2.294 \cdot 10^8$
3	1	211.11	$3 \cdot 10^4$	$3.635 \cdot 10^6$	$4.071 \cdot 10^8$
4	1	239.17	$3.671 \cdot 10^4$	$4.665 \cdot 10^6$	$5.38 \cdot 10^8$
5	1	320.95	$6.296 \cdot 10^4$	$9.814 \cdot 10^6$	$1.341 \cdot 10^9$
6	1	361.84	$7.776 \cdot 10^4$	$1.299 \cdot 10^7$	$1.872 \cdot 10^9$
7	1	366.84	$7.959 \cdot 10^4$	$1.339 \cdot 10^7$	$1.939 \cdot 10^9$
8	1	374.17	$8.233 \cdot 10^4$	$1.399 \cdot 10^7$	$2.041 \cdot 10^9$
9	1	379.78	$8.446 \cdot 10^4$	$1.447 \cdot 10^7$	$2.123 \cdot 10^9$
10	1	379.78	$8.446 \cdot 10^4$	$1.447 \cdot 10^7$	$2.123 \cdot 10^9$
11	1	379.78	$8.446 \cdot 10^4$	$1.447 \cdot 10^7$	$2.123 \cdot 10^9$
12	1	379.78	$8.446 \cdot 10^4$	$1.447 \cdot 10^7$	$2.123 \cdot 10^9$
13	1	390.49	$8.865 \cdot 10^4$	$1.542 \cdot 10^7$	$2.288 \cdot 10^9$
14	1	401.2	$9.294 \cdot 10^4$	$1.641 \cdot 10^7$	$2.464 \cdot 10^9$

$$B_{Nn,j} := \frac{T_{Nn,j}}{G_{Nn,N-1}} \cdot G_{Nn,N-1-j}$$

$$B_{Nn,0} := 1 - B_{Nn,1}$$

$B =$

	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0
8	0	0	0	0	0
9	0	0	0	0	0
10	0	0	0	0	0
11	0	0	0	0	0
12	0	0	0	0	0
13	0	0	0	0	0
14	0.929	0.071	$4.328 \cdot 10^{-3}$	$2.001 \cdot 10^{-4}$	$5.341 \cdot 10^{-6}$

Розрахунок допоміжних коефіцієнтів $Gn_{i,0} := 1$

$Gn =$

	0	1	2	3	4
0	0	401.2	$8.893 \cdot 10^4$	$1.548 \cdot 10^7$	$2.299 \cdot 10^9$
1	1	351.2	$7.288 \cdot 10^4$	$1.177 \cdot 10^7$	$1.643 \cdot 10^9$
2	1	298.98	$5.193 \cdot 10^4$	$6.913 \cdot 10^6$	$7.857 \cdot 10^8$
3	1	352.31	$7.333 \cdot 10^4$	$1.187 \cdot 10^7$	$1.661 \cdot 10^9$
4	1	373.14	$8.169 \cdot 10^4$	$1.381 \cdot 10^7$	$2.003 \cdot 10^9$
5	1	319.42	$6.013 \cdot 10^4$	$8.813 \cdot 10^6$	$1.121 \cdot 10^9$
6	1	360.31	$7.654 \cdot 10^4$	$1.261 \cdot 10^7$	$1.792 \cdot 10^9$
7	1	396.2	$9.094 \cdot 10^4$	$1.595 \cdot 10^7$	$2.381 \cdot 10^9$
8	1	393.87	$9 \cdot 10^4$	$1.573 \cdot 10^7$	$2.343 \cdot 10^9$
9	1	395.59	$9.069 \cdot 10^4$	$1.589 \cdot 10^7$	$2.371 \cdot 10^9$
10	1	401.2	$9.294 \cdot 10^4$	$1.641 \cdot 10^7$	$2.464 \cdot 10^9$
11	1	401.2	$9.294 \cdot 10^4$	$1.641 \cdot 10^7$	$2.464 \cdot 10^9$
12	1	401.2	$9.294 \cdot 10^4$	$1.641 \cdot 10^7$	$2.464 \cdot 10^9$
13	1	390.49	$8.865 \cdot 10^4$	$1.542 \cdot 10^7$	$2.288 \cdot 10^9$
14	1	0	0	0	0

$i := 0 .. Nn - 1 \qquad j := 0 .. N - 1$

$$B_{i,j} := \frac{T_{i,j}}{G_{Nn,N-1}} Gn_{i,N-1-j}$$

$B =$

	0	1	2	3	4
0	0.933	0.063	$3.61 \cdot 10^{-3}$	$1.629 \cdot 10^{-4}$	0
1	0.667	0.239	0.074	0.018	$2.537 \cdot 10^{-3}$
2	0.319	0.287	0.22	0.13	0.044
3	0.674	0.236	0.071	0.017	$2.319 \cdot 10^{-3}$
4	0.813	0.157	0.026	$3.346 \cdot 10^{-3}$	$2.516 \cdot 10^{-4}$
5	0.455	0.293	0.163	0.071	0.018
6	0.728	0.209	0.052	$9.999 \cdot 10^{-3}$	$1.135 \cdot 10^{-3}$
7	0.967	0.032	$9.228 \cdot 10^{-4}$	$2.01 \cdot 10^{-5}$	$2.537 \cdot 10^{-7}$
8	0.951	0.047	$1.963 \cdot 10^{-3}$	$6.297 \cdot 10^{-5}$	$1.172 \cdot 10^{-6}$
9	0.963	0.036	$1.159 \cdot 10^{-3}$	$2.835 \cdot 10^{-5}$	$4.021 \cdot 10^{-7}$
10	1	0	0	0	0
11	1	0	0	0	0
12	1	0	0	0	0
13	0.929	0.067	$4.127 \cdot 10^{-3}$	$1.947 \cdot 10^{-4}$	$5.341 \cdot 10^{-6}$
14	0.929	0.071	$4.328 \cdot 10^{-3}$	$2.001 \cdot 10^{-4}$	$5.341 \cdot 10^{-6}$

$i := 0..Nn \qquad j := 0..N-1$

$$SumB_i := \sum_j B_{i,j}$$

$SumB =$

	0
0	1
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1.005

$$\lambda_i := e_i \cdot \frac{G_{Nn-1,N-2}}{G_{Nn,N-1}}$$

Інтенсивність вхідного потоку

$\lambda =$

	0
0	$6.259 \cdot 10^{-3}$
1	0.031
2	0.064
3	0.031
4	0.018
5	0.051
6	0.026
7	$3.129 \cdot 10^{-3}$
8	$4.588 \cdot 10^{-3}$
9	$3.511 \cdot 10^{-3}$
10	0
11	0
12	0
13	$6.703 \cdot 10^{-3}$
14	$6.703 \cdot 10^{-3}$

$$L_i := \sum_{n=0}^{N-1} \left(n \cdot B_{i,n}\right)$$

Середнє число пакетів в вузлах

$L =$

	0
0	0.071
1	0.45
2	1.294
3	0.437
4	0.221
5	0.904
6	0.348
7	0.034
8	0.051
9	0.039
10	0
11	0
12	0
13	0.076
14	0.081

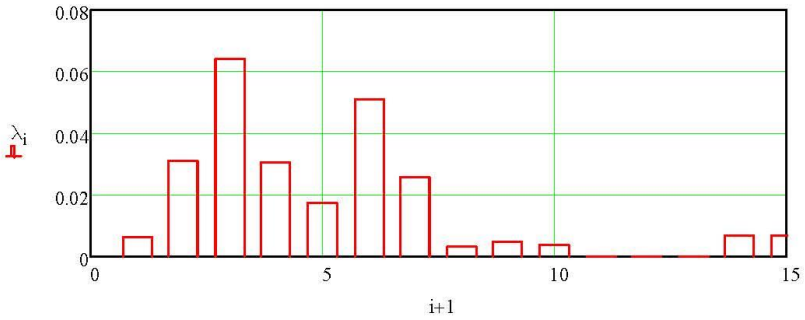
Середній час перебування пакета в вузлі

$$t_i := \frac{L_i}{\lambda_i}$$

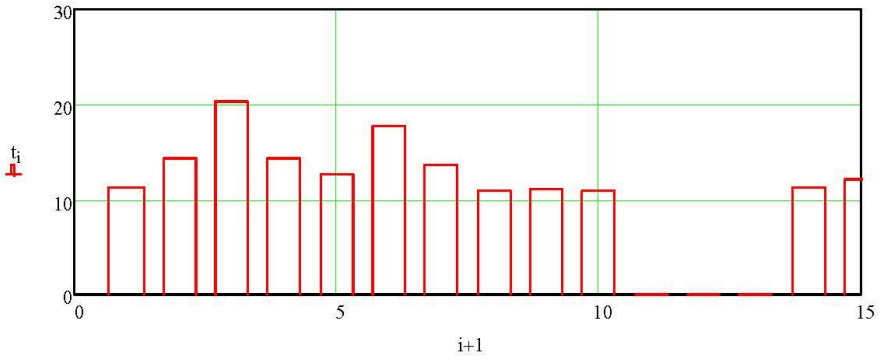
$t =$

	0
0	11.274
1	14.391
2	20.219
3	14.291
4	12.556
5	17.67
6	13.59
7	10.954
8	11.102
9	10.992
10	0
11	0
12	0
13	11.322
14	12.03

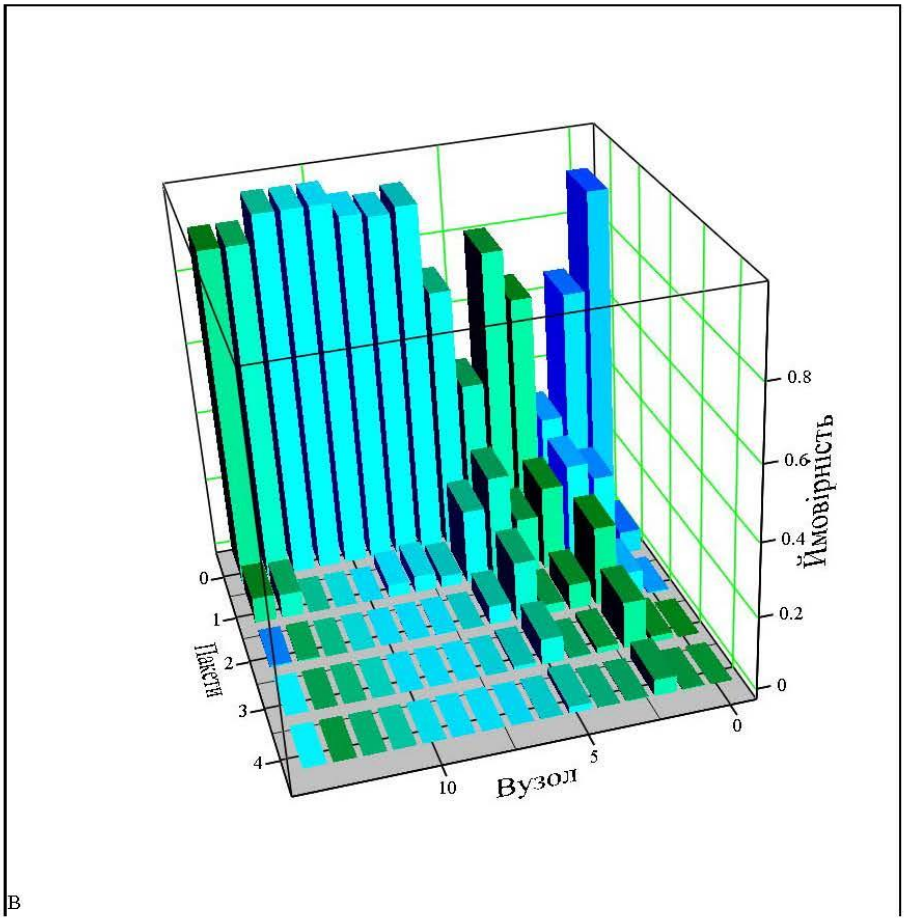
Інтенсивність вхідного потоку



Середній час перебування пакета




Середня кількість пакетів
1.5
1
0.5
0
L_i
0
5
10
15
i+1


0.8
0.6
0.4
0.2
0
Ймовірність
0
1
2
3
4
Пакети
0
5
10
Вузол
B