МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

на тему:

«Інформаційно-комунікаційна технологія налаштування корпоративних мультисервісних мереж»

Завідувач випускаючої кафедри — **Довбиш А.С.**

Керівник роботи — **Великодний Д.В.**

Студент гр. ІН.м.-02 — **Мороз Е.В.**

Суми 2021

Сумський державний університет
(назва вузу)

Факультет *ЕлІТ* Кафедра *Комп'ютерних наук*

Спеціальність *«122 - Комп'ютерні науки»*

Затверджую:
зав.кафедрою _____
_____
"_____" _____ 20___р.

ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТОВІ

Морозу Едуарду Вадимовичу
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) *Інформаційна система онлайн замовлень в закладах швидкого харчування*

затверджую наказом по інституту від "____" _____ 20 ___ р. № _____

2. Термін здачі студентом закінченого проекту (роботи)_____

3. Вхідні данні до проекту (роботи) _____
_____
_____
_____

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)
1) Літературний огляд та постановка задачі 2) Моделювання технологій корпоративних сервісів та аналіз їх взаємодії; 3) Створення схеми корпоративної мережі та розробка графічного інтерфейсу 4) Тестування інтерфейсу.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____
_____

6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання _____

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів дипломного проекту (роботи)	Термін виконання проекту (роботи)	Примітка
1.	Аналіз поставленої проблеми предметної області, постановка задачі		
2.	Огляд технологій та протоколів, які плануються використовуватися		
3.	Розробка веб-системи з застосуванням досліджуваних технологій		
4.	Аналіз та тестування отриманих систем		
5.	Оформлення пояснювальної записки до кваліфікаційної магістерської роботи		

Студент – дипломник _____
(підпис)

Керівник проекту _____
(підпис)

РЕФЕРАТ

Записка: 64 стор., 33 рис., 3 додатка, 14 літературних джерел.

Об’єкт дослідження – технології DHCP, DNS, VLAN, VoIP, FTP, Email та протоколи маршрутизації.

Предмет дослідження – особливості роботи та налаштування елементів корпоративної мережі.

Мета роботи — Створення та налаштування мультисервісної мережі у симуляторі Cisco Packet Trace і розробка графічного інтерфейсу.

Методи дослідження – моделювання технологій DHCP, DNS, VLAN, VoIP, FTP, Email у симуляторі Cisco Packet Tracer.

Результати — розроблено веб-орієнтовану інформаційну систему в яку можна ввести дані: ір-адреси та маски для всіх інтерфейсів. Отримані результати можна скопіювати та використовувати в своїх цілях.

DNS, DHCP, VLAN, VoIP, CISCO, RIP

Зміст

ВСТУП

Сучасний світ сьогодні не уявляє своє життя без інтернету, люди використовують його майже для будь-яких цілей: спілкування, бізнес, перегляд розважального контенту та багато інших речей, які тільки можна уявити. Все це працює завдяки налагодженій системі, яку підтримають величезні корпорації та провайдери. Іноді новим швидкозростаючим корпораціям потрібно вийти на абсолютно новий ринок для них, а це потребує певного часу. Велику частку часу забирає налаштування мережі та сервісів, які необхідні для функціювання офісу і інфраструктури в цілому.

Завданням магістерської роботи було обрано створення графічного інтерфейсу для налаштування корпоративних мультисервісних мереж. Проаналізувавши сучасний ринок подібного ПО, нами були знайдені в основному інструкції та рекомендації для конфігурації певних областей мережі, спеціалізованого ПО яке б видавало необхідну конфігурацію знайдено не було.

Саме тому нами була поставлено за мету цієї роботи оглянути популярні та необхідні технології для функціювання корпоративної інфраструктури і прискорення та спрощення процесу налаштування корпоративних структур мережі завдяки комфортному та простому графічному інтерфейсу.

В ході аналізу поставленої задачі було вирішено використовувати такі інструменти як мова розмітки гіпертексту HTML, мережевий емулятор Cisco Packet Tracer для моделювання мереж. Головні задачі цієї роботи спроектувати реальну мережу, дослідити оптимальний метод конфігурування цих мереж і на базі цих досліджень створити графічний інтерфейс для вирішення поставлених задач та який би адаптувався під різні вхідні дані користувача.

В межах цієї роботи необхідно зробити аналіз предметної області, дослідити існуючі технології та методи і вивчити їх слабкі і сильні сторони, щоб удосконалити, або втілити в розробці графічного інтерфейсу.

1. ІНФОРМАЦІЙНИЙ ОГЛЯД СЕРВІСІВ У ЛОКАЛЬНИХ КОМП`ЮТЕРНИХ МЕРЕЖАХ

1.1 VLAN

VLAN (віртуальна локальна мережа) — це внутрішня мережа, яка застосовується для об`єднання групи фізичних пристроїв, які знаходяться в різних локальних мережах розділених власними комутаторами [1]. Локальна мережа (LAN) – це група комп’ютерів і пристроїв, які спільно використовують лінію зв’язку або бездротове з’єднання з сервером в межах однієї географічної області.

VLAN дозволяють мережевим адміністраторам легко розділити одну комутовану мережу, щоб відповідати функціональним вимогам і вимогам безпеки їхніх систем без необхідності прокладати нові кабелі або вносити серйозні зміни в поточну мережеву інфраструктуру. VLAN часто встановлюються великими компаніями для перерозподілу пристроїв для кращого управління трафіком.

VLAN також важливі, оскільки вони можуть допомогти покращити загальну продуктивність мережі, об’єднавши пристрої, які найчастіше спілкуються. VLAN також забезпечують безпеку у великих мережах, дозволяючи вищий ступінь контролю над тим, які пристрої мають доступ один до одного. VLAN, як правило, гнучкі, оскільки вони засновані на логічних з’єднаннях, а не на фізичних.

Один або кілька мережевих комутаторів можуть підтримувати декілька незалежних мереж VLAN, створюючи реалізації підмереж рівня 2 (канал даних). VLAN пов’язана з широкомовним доменом. Зазвичай він складається з одного або кількох мережевих комутаторів.

Існують наступні типи VLAN:

- Протокольний VLAN, який обробляє трафік на основі свого протоколу. Комутатор буде розділяти або переадресовувати трафік на основі протоколу трафіку.

- Статичний VLAN – також називається VLAN на основі портів, потребує адміністратора мережі, щоб призначити порти мережевого комутатора до віртуальної мережі;
- Динамічний VLAN – дає можливість адміністратору мережі просто призначати приналежність до мережі на основі характеристик пристрою, а не розташувати порти комутатора.

Інтерфейси (порти) комутаторів можна призначити одній або кільком VLAN, що дозволяє системам розділятися на логічні групи - на основі того, з яким відділом вони пов'язані - і встановлювати правила щодо того, як системам в окремих групах дозволяється спілкуватися з один одним. Ці групи можуть варіюватися від простих і практичних (комп’ютери в одній віртуальній локальній мережі можуть бачити принтер у цій же мережі, але комп’ютери за межами цієї мережі не можуть), до складних і легальних (наприклад, комп’ютери у відділах банківського обслуговування не можуть взаємодіяти з комп’ютерами в торговому відділі).

Кожна VLAN надає доступ до каналів передачі даних всім хостам, підключеним до портів комутатора, налаштованих на той самий ідентифікатор віртуальної локальної мережі. Тег VLAN — це поле, яке складається з 12 бітів і знаходиться в заголовку Ethernet, яке забезпечує підтримку до 4096 віртуальних локальних мереж на домен комутації. Тегування VLAN стандартизовано в IEEE 802.1Q і найчастіше знають як Dot1Q [2].

Коли фрейм без тегів отримується від підключеного хоста, тег VLAN ID, налаштований на цьому інтерфейсі, додається до заголовка кадру каналу передачі даних, використовуючи формат 802.1Q. Потім кадр 802.1Q пересилається до місця призначення. Кожен комутатор використовує тег, щоб тримати трафік кожної VLAN окремо від інших VLAN, пересилаючи його лише там, де VLAN налаштовано. Магістральні зв’язки між комутаторами обробляють декілька VLAN, використовуючи тег, щоб зберегти їх

відокремленими. Коли кадр досягає порту комутатора призначення, тег VLAN видаляється, перш ніж кадр буде переданий на пристрій призначення [2].

Кілька мереж VLAN можна налаштувати на одному порту за допомогою конфігурації магістралі, в якій кожен кадр, надісланий через порт, позначений ідентифікатором VLAN, як описано вище. Інтерфейс сусіднього пристрою, який може бути на іншому комутаторі або на хості, який підтримує тегування 802.1Q, повинен підтримувати конфігурацію магістрального режиму для передачі та отримання позначених кадрів. Будь-які кадри Ethernet без тегів призначаються до VLAN за замовчуванням, яку можна призначити в конфігурації комутатора.

Коли комутатор з підтримкою VLAN отримує кадр Ethernet без тегів від підключеного хоста, він додає тег VLAN, назначений інтерфейсу входу. Кадр пересилається на порт хоста з MAC-адресом призначення (адреса контролю доступу до медіа). Трансляція, невідома одноадресна та багатоадресна передача (трафік BUM) пересилаються на всі порти VLAN. Якщо раніше невідомий хост відповідає на невідомий одноадресний фрейм, комутатори дізнаються місцезнаходження цього хоста і не заповнюють наступні кадри, адресовані цьому хосту.

Таблиці переадресації комутаторів оновлюються за допомогою двох механізмів. По-перше, старі записи пересилання періодично видаляються з таблиць пересилання, часто за допомогою настроюваного таймера. По-друге, будь-яка зміна топології призводить до скорочення таймера оновлення таблиці пересилання, що ініціює оновлення.

1.2 DHCP

DHCP (Протокол динамічної конфігурації хосту) — це протокол керування мережею, який використовується для динамічного призначення адреси Інтернет-протоколу (IP) будь-якому пристрою або вузлу в мережі, щоб вони могли спілкуватися за допомогою IP. DHCP автоматизує та

централізовано керує цими конфігураціями, замість того, щоб вимагати від адміністратора мережі вручну призначати IP-адреси всім мережевим пристроям. DHCP можна реалізувати в невеликих локальних мережах, а також у великих корпоративних мережах [3].

DHCP призначатиме нові IP-адреси в кожному місці, коли пристрої переміщуються з місця на місце, що означає, що адміністраторам мережі не доведеться вручну налаштовувати кожен пристрій на дійсну IP-адресу або повторно налаштовувати пристрій за допомогою нової IP-адреси, якщо він переміщується на новий розташування в мережі. Версії DHCP доступні для використання в IP версії 4 (IPv4) і IP версії 6 (IPv6). IPv6 став галузевим стандартом у 2017 році — майже через 20 років після того, як його характеристики були вперше опубліковані. Хоча швидкість впровадження IPv6 була повільною, до липня 2019 року понад 29% користувачів Google надсилали запити за допомогою IPv6.

DHCP працює на прикладному рівні стеку протоколу керування передачею/IP (TCP/IP), щоб динамічно призначати IP-адреси клієнтам DHCP та розподіляти інформацію про конфігурацію TCP/IP клієнтам DHCP. Це включає інформацію про маску підмережі, IP-адреси шлюзу за замовчуванням та адреси системи доменних імен (DNS).

DHCP — це протокол клієнт-сервер, у якому сервери керують пулом унікальних IP-адрес, а також інформацією про параметри конфігурації клієнта та призначають адреси з цих пулів адрес. Клієнти з підтримкою DHCP надсилають запит на сервер DHCP щоразу, коли вони підключаються до мережі.

Клієнти, налаштовані за допомогою DHCP, передають запит на сервер DHCP та запитують інформацію про конфігурацію мережі для локальної мережі, до якої вони підключені. Клієнт зазвичай передає запит на цю інформацію відразу після завантаження. DHCP-сервер відповідає на запит клієнта, надаючи інформацію про конфігурацію IP, яку раніше вказав

адміністратор мережі. Це включає конкретну IP-адресу, а також період часу, який також називається орендою, для якого дійсне виділення. Під час оновлення призначення клієнт DHCP запитує ті самі параметри, але сервер DHCP може призначити нову IP-адресу на основі політик, встановлених адміністраторами. Клієнти DHCP також можна налаштувати на інтерфейсі Ethernet.

Сервер DHCP керує записом усіх IP-адрес, які він приділяє вузлам мережі. Якщо вузол переміщено в мережу, сервер ідентифікує його за допомогою своєї адреси керування доступом до медіа (MAC), що запобігає випадковій конфігурації кількох пристроїв з однаковою IP-адресою. Налаштування сервера DHCP також вимагає створення файлу конфігурації, в якому зберігатиметься інформація про мережу для клієнтів.

DHCP не є протоколом маршрутизації і не є безпечним. DHCP обмежується певною локальною мережею, що означає, що один DHCP-сервер на одну локальну мережу є достатнім або два сервери для використання у разі відмови. Більші мережі можуть мати глобальну мережу (WAN), яка містить кілька окремих місць. Залежно від з’єднань між цими точками та кількості клієнтів у кожному місці можна налаштувати декілька серверів DHCP для обробки розподілу адрес. Якщо адміністратори мережі хочуть, щоб DHCP-сервер надавав адресацію кільком підмережам у певній мережі, вони повинні налаштувати послуги ретрансляції DHCP, розташовані на маршрутизаторах, що підключаються, які мають перетинати запити DHCP. Ці агенти передають повідомлення між клієнтами DHCP і серверами, розташованими в різних підмережах [3].

DHCP не має жодного вбудованого механізму, який дозволив би клієнтам і серверам аутентифікувати один одного. Обидва вразливі до обману – один комп’ютер прикидається іншим – і для атаки, коли шахрайські клієнти можуть вичерпати пул IP-адрес сервера DHCP.

1.3 DNS

Система доменних імен (DNS) — це база даних імен, в якій розміщені доменні імена в Інтернеті та переведені в адреси Інтернет-протоколу (IP). Система доменних імен зіставляє ім’я, яке люди використовують для визначення місцезнаходження веб-сайту, з IP-адресою, яку використовує комп’ютер, щоб знайти цей веб-сайт [4].

Наприклад, якщо хтось введе "example.com" у веб-браузер, сервер за лаштунками зіставляє це ім'я з відповідною IP-адресою. IP-адреса за структурою схожа на 203.0.113.72.

Перегляд веб-сторінок та більшість інших видів діяльності в Інтернеті покладаються на DNS для швидкого надання інформації, необхідної для підключення користувачів до віддалених хостів. Відображення DNS розподіляється по всьому Інтернету в ієрархії повноважень. Постачальники доступу та підприємства, а також уряди, університети та інші організації, як правило, мають власні призначені діапазони IP-адрес і призначене доменне ім’я. Вони також зазвичай запускають DNS-сервери для керування зіставленням цих імен з цими адресами. Більшість уніфікованих локаторів ресурсів (URL) побудовано навколо доменного імені веб-сервера, який приймає клієнтські запити [4].

DNS-сервери перетворюють URL-адреси та доменні імена в IP-адреси, які комп’ютери можуть розуміти та використовувати. Вони перетворюють те, що користувач вводить у браузер, у те, що машина може використовувати для пошуку веб-сторінки. Цей процес перекладу та пошуку називається розділенням DNS.

Основний процес вирішення DNS складається з таких кроків:

Користувач вводить веб-адресу або доменне ім’я у браузері.

Браузер надсилає в мережу повідомлення, яке називається рекурсивним запитом DNS, щоб дізнатися, якій IP або мережевій адресі відповідає домен.

Запит надходить до рекурсивного DNS-сервера, який також називають рекурсивним резольвером, і зазвичай ним керує постачальник послуг Інтернету (ISP). Якщо рекурсивний резольвер має адресу, він поверне адресу користувачеві, і веб-сторінка завантажиться.

Якщо рекурсивний DNS-сервер не має відповіді, він запитує ряд інших серверів у такому порядку: кореневі сервери імен DNS, сервери імен домену верхнього рівня (TLD) і авторитетні сервери імен.

Три типи серверів працюють разом і продовжують переспрямовувати, поки не отримають запис DNS, який містить запитувану IP-адресу. Він надсилає цю інформацію на рекурсивний DNS-сервер, і веб-сторінка, яку шукає користувач, завантажується. Кореневі сервери імен DNS і сервери TLD переважно перенаправляють запити і рідко самі надають рішення.

Рекурсивний сервер зберігає або кешує запис A для імені домену, який містить IP-адресу. Наступного разу, коли він отримає запит на це доменне ім'я, він зможе відповісти безпосередньо користувачеві, а не запитувати інші сервери [5].

Якщо запит досягає авторитетного сервера і йому не вдається знайти інформацію, він повертає повідомлення про помилку.

Весь процес запитів до різних серверів займає частку секунди і зазвичай непомітний для користувача.

DNS-сервери відповідають на запитання як всередині, так і за межами власних доменів. Коли сервер отримує запит з-за меж домену на інформацію про ім’я або адресу всередині домену, він надає повноцінну відповідь.

Коли сервер отримує запит від свого домену на ім’я або адресу за межами цього домену, він пересилає запит на інший сервер, зазвичай керований його провайдером.

Ім’я домену зазвичай міститься в URL-адресі. Доменне ім’я складається з кількох частин, які називаються мітками. Ієрархія домену читається справа наліво, кожен розділ позначає підрозділ.

TLD з’являється після крапки в імені домену. Прикладами доменів верхнього рівня є .com, .org та .edu, але є багато інших. Деякі можуть позначати код країни або географічне розташування, наприклад .us для Сполучених Штатів або .ca для Канади.

Кожна мітка з лівого боку TLD позначає інший субдомен домену праворуч. Наприклад, в URL-адресі www. https://www.sumdu.com "sumdu" є субдоменом .com, а "www". є субдоменом sumdu.com.

Може бути до 127 рівнів субдоменів, і кожна мітка може містити до 63 символів. Загальна довжина символів домену може містити до 253 символів. Інші правила включають відсутність початку або закінчення міток дефісами та відсутність повністю числової назви TLD.

Спеціальна група з розробки Інтернету (IETF) вказала правила щодо впровадження доменних імен у Запиті на коментарі (RFC) 1035.

1.4 FTP

FTP (File Transfer Protocol) — це мережевий протокол для передачі файлів між комп’ютерами через з’єднання протоколу керування передачею/протоколу Інтернету (TCP/IP). У пакеті TCP/IP FTP вважається протоколом прикладного рівня [6].

У транзакції FTP комп’ютер кінцевого користувача зазвичай називають локальним хостом. Другим комп'ютером, залученим до FTP, є віддалений хост, який зазвичай є сервером. Обидва комп’ютери мають бути підключені через мережу та належним чином налаштовані для передачі файлів через FTP. Сервери мають бути налаштовані для запуску служб FTP, а на клієнті має бути встановлено програмне забезпечення FTP для доступу до цих служб [6].

Хоча багато передачі файлів можуть здійснюватися за допомогою протоколу передачі гіпертексту (HTTP) — іншого протоколу набору TCP/IP — FTP все ще зазвичай використовується для передачі файлів за лаштунками

для інших програм, наприклад банківських послуг. Він також іноді використовується для завантаження нових програм через веб-браузери.

FTP — це клієнт-серверний протокол, який спирається на два канали зв'язку між клієнтом і сервером: командний канал для керування розмовою та канал даних для передачі вмісту файлу.

Ось як працює типова передача FTP:

Користувачу, як правило, потрібно ввійти на FTP-сервер, хоча деякі сервери роблять доступним частину або весь свій вміст без входу, модель, відома як анонімний FTP.

Клієнт починає розмову з сервером, коли користувач просить завантажити файл.

За допомогою FTP клієнт може завантажувати, завантажувати, видаляти, перейменовувати, переміщувати та копіювати файли на сервері.

FTP-сесії працюють в активному або пасивному режимах:

- Активний режим. Після того, як клієнт ініціює сеанс через запит командного каналу, сервер створює з’єднання даних назад із клієнтом і починає передачу даних.
- Пасивний режим. Сервер використовує командний канал, щоб надіслати клієнту інформацію, необхідну для відкриття каналу даних. Оскільки в пасивному режимі клієнт ініціює всі підключення, він добре працює через брандмауери та шлюзи трансляції мережевих адрес.

Користувачі можуть працювати з FTP через простий інтерфейс командного рядка - з консолі або вікна терміналу в Microsoft Windows, Apple macOS або Linux - або за допомогою спеціального графічного інтерфейсу користувача. Веб-браузери також можуть служити клієнтами FTP [6].

FTP – це стандартний мережевий протокол, який може забезпечити широкі можливості передачі файлів через IP-мережі. Без FTP передачею файлів і даних можна керувати за допомогою інших механізмів, таких як

електронна пошта або веб-сервіс HTTP, але цим іншим параметрам бракує чіткості фокусування, точності та контролю, які забезпечує FTP [7].

FTP використовується для передачі файлів між однією системою та іншою, і він має кілька поширених випадків використання, включаючи наступні:

- Резервне копіювання. FTP може використовуватися службами резервного копіювання або окремими користувачами для резервного копіювання даних з одного місця на захищений сервер резервного копіювання, на якому працюють служби FTP.
- Реплікація. Подібно до резервного копіювання, реплікація передбачає дублювання даних з однієї системи в іншу, але використовує більш комплексний підхід для забезпечення більшої доступності та стійкості. FTP також може бути використаний для полегшення цього.
- Доступ і завантаження даних. FTP також зазвичай використовується для доступу до спільного веб-хостингу та хмарних служб як механізму завантаження даних у віддалену систему.

Існує кілька різних способів налаштувати FTP-сервер і клієнтське програмне забезпечення може здійснювати передачу файлів за допомогою наступних видів FTP:

- Анонімний FTP. Це найпростіша форма FTP. Він забезпечує підтримку передачі даних без шифрування даних або використання імені користувача та пароля. Найчастіше він використовується для завантаження матеріалів, дозволених до необмеженого розповсюдження. Він працює на порту
- FTP, захищений паролем. Це також базова служба FTP, але вона вимагає використання імені користувача та пароля, хоча служба може бути не зашифрованою чи безпечною. Він також працює на порту 21.
- FTP Secure (FTPS). Цей підхід, який іноді називають рівнем безпечних сокетів FTP (FTP-SSL), забезпечує неявну безпеку транспортного

рівня (TLS), щойно встановлюється FTP-з'єднання. Спочатку FTPS використовувався для забезпечення більш безпечної форми передачі даних FTP. Зазвичай за замовчуванням використовується порт 990.

- FTP через явний SSL/TLS (FTPES). Цей підхід забезпечує явну підтримку TLS шляхом оновлення FTP-з'єднання через порт 21 до зашифрованого. Це поширений підхід веб-сервісів і служб обміну файлами для забезпечення безпечної передачі файлів.
- Безпечний FTP (SFTP). Технічно це не протокол FTP, але він функціонує аналогічно. Швидше, SFTP є підмножиною протоколу Secure Shell (SSH), який працює через порт 22. SSH зазвичай використовується системними адміністраторами для віддаленого та безпечного доступу до систем і програм, а SFTP забезпечує механізм у SSH для безпечної передачі файлів [8].

1.5 Email

Електронна пошта (Email) — це обмін повідомленнями, збереженими на комп'ютері, за допомогою телекомунікацій. Повідомлення електронної пошти зазвичай кодуються в американському стандартному коді для обміну інформацією (ASCII). Однак ви також можете надсилати нетекстові файли, такі як графічні зображення та звукові файли, як вкладення, надіслані у двійкових потоках. Електронна пошта була однією з перших дій через Інтернет і досі залишається найпопулярнішим використанням. Великий відсоток загального трафіку через Інтернет становить електронна пошта. Електронна пошта також може обмінюватися між користувачами постачальника послуг онлайн та в мережах, відмінних від Інтернету, як загальнодоступних, так і приватних [9].

Електронну пошту можна розповсюджувати як на списки людей, так і на окремих осіб. Спільним списком розсилки можна керувати за допомогою відбивача електронної пошти. Деякі списки розсилки дозволяють підписатися,

надіславши запит адміністратору списку розсилки. Список розсилки, який адмініструється автоматично, називається сервером списків.

Електронна пошта — це один із протоколів, що входять до набору протоколів протоколу керування транспортом/протоколу Інтернету (TCP/IP). Популярним протоколом для надсилання електронної пошти є Simple Mail Transfer Protocol (SMTP), а популярним протоколом для його отримання є Post Office Protocol 3 (POP3) [10].

Сьогодні термін електронна пошта часто використовується для включення як електронної пошти на основі браузера, такої як Gmail і AOL, так і електронної пошти без браузера, як-от Outlook для Office 365. Однак раніше існувала відмінність, що визначає електронну пошту як програму без браузера. що вимагало виділеного сервера електронної пошти та клієнтів. Перевагами використання електронної пошти без браузера є інтеграція з корпоративними програмними платформами, підвищена безпека та відсутність реклами.

Електронну пошту можна використовувати різними способами, як особисто, так і в організації, а також один на один або серед великої групи людей.

Більшість людей вважають електронну пошту корисним способом спілкування з окремими особами або невеликими групами друзів чи колег. Це дозволяє користувачам легко надсилати та отримувати документи, зображення, посилання та інші файли. Крім того, він надає користувачам гнучкість спілкування з іншими за власним графіком.

Іншим корисним використанням електронної пошти для спілкування один на один або в невеликих групах може бути відправка професійних листів після зустрічей, зустрічей або співбесід або нагадування учасникам про наближення термінів і важливих за часом заходів. Це також дозволяє користувачам швидко й легко нагадувати всім учасникам зустрічі про майбутній захід або повідомляти групу про зміну часу. Цьому сприяє

інтеграція календарів і зустрічей у більшість платформ електронної пошти [10].

Крім того, компанії можуть використовувати електронну пошту для передачі інформації великій кількості співробітників, клієнтів і потенційних клієнтів. Електронна пошта часто використовується для інформаційних бюлетенів, де передплатникам списку розсилки надсилається конкретний рекламований контент від компанії, а також для прямих маркетингових кампаній електронною поштою, де реклама або рекламна акція надсилається цільовій групі клієнтів.

Електронну пошту також можна використовувати, щоб перетворити потенційних клієнтів на клієнтів-платників або перетворити потенційний продаж на завершену покупку. Наприклад, компанія може створити автоматизований електронний лист, який надсилається онлайн-покупцям, які зберігають товари в кошику певний час. Електронний лист може нагадати клієнту, що в його кошику є продукти, і спонукати його завершити покупку до того, як товар закінчиться.

Також поширені електронні листи з проханням залишити відгук після покупки. Вони можуть включати опитування з проханням переглянути якість послуг або нещодавно отриманий продукт.

1.6 Маршрутизація. Основні алгоритми

Мережеві пристрої обмінюються даними між собою за допомогою маршрутизатора, маршрутизатор — це пристрій, який дізнається, які шляхи доступні і на який шлях найкраще перенаправляти трафік. Механізм, за допомогою якого маршрутизатор приймає таке рішення, відомий нам як маршрутизація [11].

Маршрутизація використовується для отримання пакета від одного пристрою та передачі його по мережі іншому пристрої через інші мережі. Якщо в мережі немає маршрутизаторів, маршрутизація не підтримується.

Маршрутизатори спрямовують (перенаправляють) трафік у всі мережі, що становлять об'єднану мережу.

Для маршрутизації пакета маршрутизатор повинен мати таку інформацію:

- Адреса призначення;
- Сусідній маршрутизатор, від якого він може дізнатися про віддалені мережі;
- Доступні шляхи до всіх віддалених мереж;
- Найкращий шлях до кожної віддаленої мережі;
- Методи обслуговування та перевірки інформації про маршрутизацію.

Якщо мережу підключено безпосередньо до маршрутизатора, то він уже знає, як спрямувати пакет до цієї мережі. Якщо мережа не підключена безпосередньо, маршрутизатор повинен дізнатися шляхи доступу до віддаленої мережі за допомогою статичної маршрутизації (введення адміністратором вручну розташування всіх мереж у таблицю маршрутизації) або за допомогою динамічної маршрутизації.

Статична і динамічна маршрутизація – це два методи, які використовуються для визначення того, як відправити пакет до місця призначення.

Статичні маршрути налаштовуються перед будь-яким мережевим зв'язком. З іншого боку, динамічна маршрутизація вимагає від маршрутизаторів обміну інформацією з іншими маршрутизаторами, щоб дізнатися про шляхи через мережу. Статична та динамічна маршрутизація використовуються там, де це необхідно, а деякі мережі використовують обидва.

Адміністратори мережі використовують статичну маршрутизацію або неадаптивну маршрутизацію, щоб визначити маршрут, коли є один маршрут або бажаний маршрут для досягнення трафіком місця призначення. Статична

маршрутизація використовує невеликі таблиці маршрутизації лише з одним записом для кожного місця призначення. Це також вимагає менше часу на обчислення, ніж динамічна маршрутизація, оскільки кожен маршрут попередньо налаштований.

Оскільки статичні маршрути попередньо налаштовані, адміністратори повинні вручну переналаштувати маршрути, щоб вони адаптувалися до змін у мережі, коли вони відбуваються. Статичні маршрути зазвичай використовуються в мережах, де адміністратори не очікують змін.

Динамічна маршрутизація, яку іноді називають адаптивною маршрутизацією, є складнішою, ніж статична, оскільки вона створює більше можливих маршрутів для відправки пакетів через мережу. Динамічні маршрути зазвичай використовуються у великих, плавних мережах, де статичні маршрути були б громіздкими для підтримки та частих змін до таблиці маршрутизації. Оскільки динамічна маршрутизація є складнішою, вона споживає більше пропускної спроможності, ніж статична маршрутизація [12].

Динамічна маршрутизація використовує алгоритми для обчислення кількох можливих маршрутів і визначення найкращого шляху для руху трафіку через мережу. Він використовує два типи складних алгоритмів: протоколи вектора відстані та протоколи стану зв’язку.

Як протоколи вектора відстані, так і протоколи стану зв’язку створюють таблицю маршрутизації в маршрутизаторі, яка включає запис для кожного можливого призначення мережі, групи мереж або певної підмережі. Кожен запис визначає, яке мережеве з’єднання використовувати для відправки отриманого пакета [12].

При використанні протоколу вектора відстані, такого як протокол інформації про маршрутизацію (RIP) або протокол внутрішньої маршрутизації шлюзу (IGRP) — кожен запис таблиці маршрутизації визначає кількість переходів до кожного пункту призначення. Маршрутизатор надсилає свою

таблицю маршрутизації кожному напряму підключеному маршрутизатору і отримує натомість таблиці інших маршрутизаторів. Маршрутизатори, що використовують протоколи вектора відстані, періодично обмінюються своїми таблицями маршрутизації з сусідніми маршрутизаторами.

Протоколи з вектором відстані мають свої переваги і недоліки. Маршрутизатори, які використовують векторні протоколи відстаней, періодично надсилають цілі таблиці маршрутизації, що створює значне навантаження при використанні у великій мережі і може створити загрозу безпеці, якщо мережа стане скомпрометована. Оскільки протоколи вектора відстані визначають маршрути на основі кількості стрибків(hops), вони можуть вибрати повільне посилання замість високошвидкісного каналу, коли кількість стрибків нижча.

Протоколи стану зв’язку, такі як Open Shortest Path First (OSPF) і Intermediate System to Intermediate System (IS-IS), визначають маршрути шляхом обміну пакетом стану зв’язку (LSP) з кожним сусіднім маршрутизатором. Кожен маршрутизатор створює LSP, який містить його попередньо налаштований ідентифікатор разом з інформацією про підключені мережі та підмережі. Потім маршрутизатор надсилає LSP сусіднім маршрутизаторам. Отримані LSP містять додаткову інформацію про шляхи до інших мереж і швидкість передачі даних. Маршрутизатори об’єднують цю інформацію з відомою раніше та зберігають її у своїх таблицях маршрутизації [11].

Як і протоколи з вектором відстані, протоколи стану зв’язку мають свої переваги і недоліки. Однією з переваг протоколів стану зв’язку є те, що вони надсилають оновлення лише тоді, коли в мережі відбуваються зміни, на відміну від протоколів з вектором постійного навантаження, що розміщуються в мережі. Протоколи стану зв’язку також можуть відновлюватися швидше та повторно визначати маршрут у разі несправності зв’язку або маршрутизатора. Але ці протоколи складніші і їх складніше налаштувати та підтримувати.

1.7 VoIP (Voice over Internet Protocol)

VoIP (протокол передачі голосу через Інтернет) — це передача голосового та мультимедійного вмісту через Інтернет-з'єднання. VoIP дозволяє користувачам здійснювати голосові дзвінки з комп’ютера, смартфона, інших мобільних пристроїв, спеціальних VoIP-телефонів і браузерів з підтримкою WebRTC. VoIP – це технологія, корисна як для споживачів, так і для бізнесу, оскільки вона зазвичай включає інші функції, які не можна знайти в звичайних телефонних службах. Ці функції можуть включати запис дзвінків, індивідуальний ідентифікатор абонента або голосову пошту для електронної пошти. Це також корисно для організацій як спосіб уніфікувати комунікації[13].

Процес працює подібно до звичайного телефону, але VoIP використовує підключення до Інтернету замість телекомунікаційних ліній прокладених телефонною компанією. VoIP забезпечується за допомогою групи технологій і методологій, що використовуються для надання голосового зв’язку через Інтернет, включаючи корпоративні локальні мережі або глобальні мережі.

Служба VoIP перетворює голос користувача з аудіо сигналу в цифрові дані, а потім надсилає ці дані через Інтернет. Якщо інший користувач дзвонить зі звичайного телефонного номера, сигнал перетворюється назад у телефонний, перш ніж він досягне цього користувача.

Протокол передачі голосу також може виконувати маршрутизацію вхідних і вихідних дзвінків через існуючі телефонні мережі. Однак деякі служби VoIP можуть працювати лише через комп’ютер або телефон VoIP.

Протокол передачі голосу об’єднує комунікаційні технології в одну єдину систему – це означає, що VoIP може забезпечувати низку аудіо, відео або текстових методів зв’язку. Це може бути особливо корисно для бізнесу, тому командам не потрібно працювати з кількома різними додатками, щоб ефективно спілкуватися один з одним.

Протокол передачі голосу створює цю мережу, дозволяючи користувачам здійснювати дзвінки та проводити веб-конференції за допомогою таких пристроїв, як комп’ютери, смартфони чи інші мобільні пристрої [14].

Деякі загальні особливості можуть включати [14]:

- аудіо дзвінки;
- відео дзвінки;
- голосова пошта;
- миттєві повідомлення;
- командні чати;
- вибрати нового оператора телефонної мережі, не потребуючи нового номера.

Послуги VoIP перетворюють голос користувача з аудіо сигналів у цифрові дані, в яких ці дані потім надсилаються іншому користувачеві - або групі користувачів - через Ethernet або Wi-Fi. Для цього VoIP використовуватиме кодеки.

Кодеки - це або апаратний, або програмний процес, який стискає та розпаковує великі обсяги даних VoIP. Якість голосу може погіршитися під час використання стиснення, але стиснення зменшує вимоги до пропускної здатності. Постачальники обладнання також використовуватимуть власні кодеки.

Процес надсилання даних іншим користувачам включає інкапсуляцію аудіо в пакети даних, передачу пакетів через IP-мережу та де-інкапсуляцію пакетів назад у аудіо на іншому кінці з’єднання.

У корпоративних або приватних мережах якість обслуговування (QoS) зазвичай використовується для визначення пріоритету голосового трафіку над додатками, які не чутливі до затримок, щоб забезпечити прийнятну якість голосу.

Додаткові компоненти типової системи VoIP включають наступне: IP-АТС для керування телефонними номерами користувачів, пристроями, функціями та клієнтами; шлюзи для підключення мереж і забезпечення перемикання збоїв або локальної живучості в разі відключення мережі; і прикордонні контролери сеансів для забезпечення безпеки, керування політикою викликів та мережевих з'єднань.

Система VoIP також може включати бази даних відстеження місцезнаходження для платформ маршрутизації викликів та керування E911 (розширений 911). Це може збирати статистику продуктивності дзвінків для реактивного та про-активного керування якістю голосу.

Виключаючи голосові мережі з комутацією каналів, VoIP зменшує витрати на мережеву інфраструктуру та дає змогу провайдерам надавати голосові послуги через широкосмугові та приватні мережі. Це також повинно дозволити підприємствам керувати єдиною мережею голосу та даних.

VoIP також підтримує стійкість мереж на основі IP, забезпечуючи швидке перемикання збоїв після збоїв і резервного зв’язку між кінцевими точками і мережами.

1.8 Постановка задачі

Провівши аналіз знайденої інформації, задачу цієї роботи можна поставити таким чином: потрібно провести симуляцію реальної мережі, налаштувати досліджені протоколи і технології, оптимізувати їх роботу та створити графічний інтерфейс завдяки веб-орієнтованій системі, яка надає можливість отримувати потрібні конфігураційні налаштування та зручно адаптуватися під різні вхідні дані користувача. Має забезпечувати зручне отримання та перенесення отриманої конфігурації у вигляді набору команд, в реальні пристрої Cisco.

Побудований інтерфейс має бути простим у використанні, направлений на користувачів, які вже мають досвід налаштування подібного обладнання

для мереж, для прискорення процесу налаштування мультисервісної мережі, для майбутнього використання сервісів. Сама веб система не має бути перевантажена різними ускладненими елементами та мати простий і зрозумілий дизайн.

Інтерфейс веб-системи повинен бути у вигляді html документу для більш простішої імплементації у більш складних системах або для перегляду в різних сучасних браузерах. В полях інтерфейсу реалізується можливість заповнити необхідні початкові дані і як результат отримати згенеровану конфігурацію обладнання під задані користувачем мережі, а також мати можливість скопіювати в зручному форматі конфігурацію пристроїв та використовувати її безпосередньо в своїх цілях. Як результат взаємодії с подібним веб-інтерфейсом юзер отримує коректно налаштовану конфігурацію заданої їм мережі.

Постановка задачі:

1. Конфігурація і симуляція мережі і її сервісів у симуляторі Cisco Packet Tracer.
2. Розробка графічного інтерфейсу для налаштування мультисервісної мережі.
3. Тестування розробленої веб-системи в симуляторі Cisco Packet Tracer.

2. ТЕХНОЛОГІЇ РОЗВ'ЯЗКУ ПОСТАВЛЕНОЇ ЗАДАЧІ

2.1 Моделювання мережі з використанням обладнання та технологій Cisco

Cisco Systems, Inc. - є провідною мережевою компанією, найбільш відомою як виробник і постачальник мережевого обладнання. Компанія також надає програмне забезпечення та пропонує супутні послуги. Протягом своєї історії Cisco зосереджувалася на мережевих технологіях на основі Інтернет-протоколу (IP), продуктах маршрутизації, комутації та технологіях для домашньої мережі, IP-телефонії, оптичних мереж, безпеки, мережі зберігання даних та бездротових технологій.

Завдяки ресурсам, які надає компанія, створюється можливість побудови інформаційних мереж, як клієнтського рівня, так і корпоративного, оскільки є можливість симуляції реального обладнання, яке використовується в великих корпораціях і не завжди доступне для рядового користувача, також Cisco має власну операційну систему Cisco IOS, яка допомагає підтримувати мережеві послуги та додатки, дає можливість розгорнути доволі широкий спектр мереж та отримати дуже цінний досвід як у проектуванні подібних мереж, так і загальний досвід проектування мереж, який може поширюватися на інше обладнання та інструменти, також це все дозволяє провести безпечну оптимізацію мереж і перенести її на реальне обладнання. Саме тому Cisco вважають унікальною компанією, яка здатна надати повний спектр послуг та інструментів без необхідності прибігати до допомоги інших компаній

2.2 Конфігурація мережі в симуляторі Cisco Packet Tracer на базі обладнання Cisco

Для того щоб створити веб-систему налаштування корпоративних мультисервісних мереж, нам потрібен прототип мережі, на основі якого вже можна буде визначити оптимальні налаштування та необхідні технології, для цього будемо використовувати Packet Tracer - крос платформний інструмент візуального моделювання, розроблений Cisco Systems, який дозволяє

користувачам створювати мережеві топології та імітувати сучасні комп'ютерні мережі. Цей симулятор дає можливість користувачу симулювати налаштування роутерів та комутаторів Cisco і працювати з ними на прикладах реальних мереж, симулятор дозволяє використовувати командний рядок для мережевих пристроїв та графічні форми для різноманітних налаштувань.

Розпочнемо с побудови основних елементів мережі, створимо мережу, яка складається з двух різних мереж 192.168.10.0/24 і 192.168.20.0/24, які будуть зв`язані між собою за допомогою маршрутизаторів Cisco 2811 та комутаторів Cisco 2950-24.

Поєднаємо всі пристрої в мережі комп`ютери з комутаторами, та комутатори з маршрутизаторами за допомогою перехресного крос-кабелю, самі ж маршрутизатори за допомогою прямого крос-кабелю, як результат отримуємо наступну мережу.

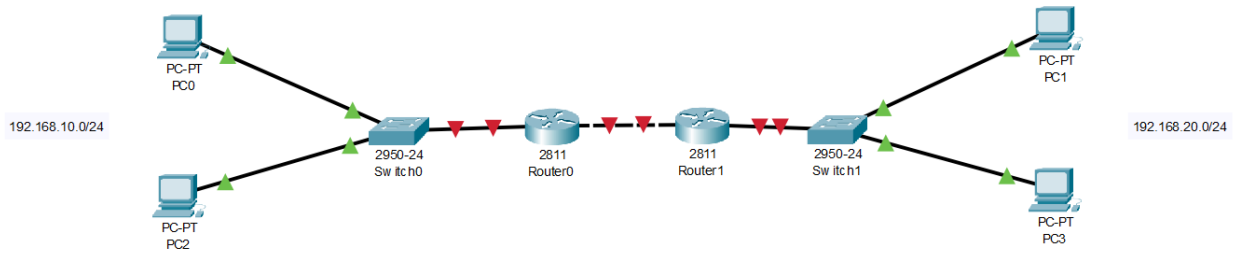


Рисунок 2.1 – Приклад мережі

Як можемо бачити на рисунку, поєднані з маршрутизаторами пристрої неактивні оскільки порти роутерів виключені за замовчуванням. Включимо порти маршрутизаторів за допомогою графічного інтерфейсу або як альтернатива можна використати консольні команди:

configure terminal

interface FastEthernet0/0

no shutdown

do wr – для збереження конфігурації маршрутизатора у внутрішній пам`яті

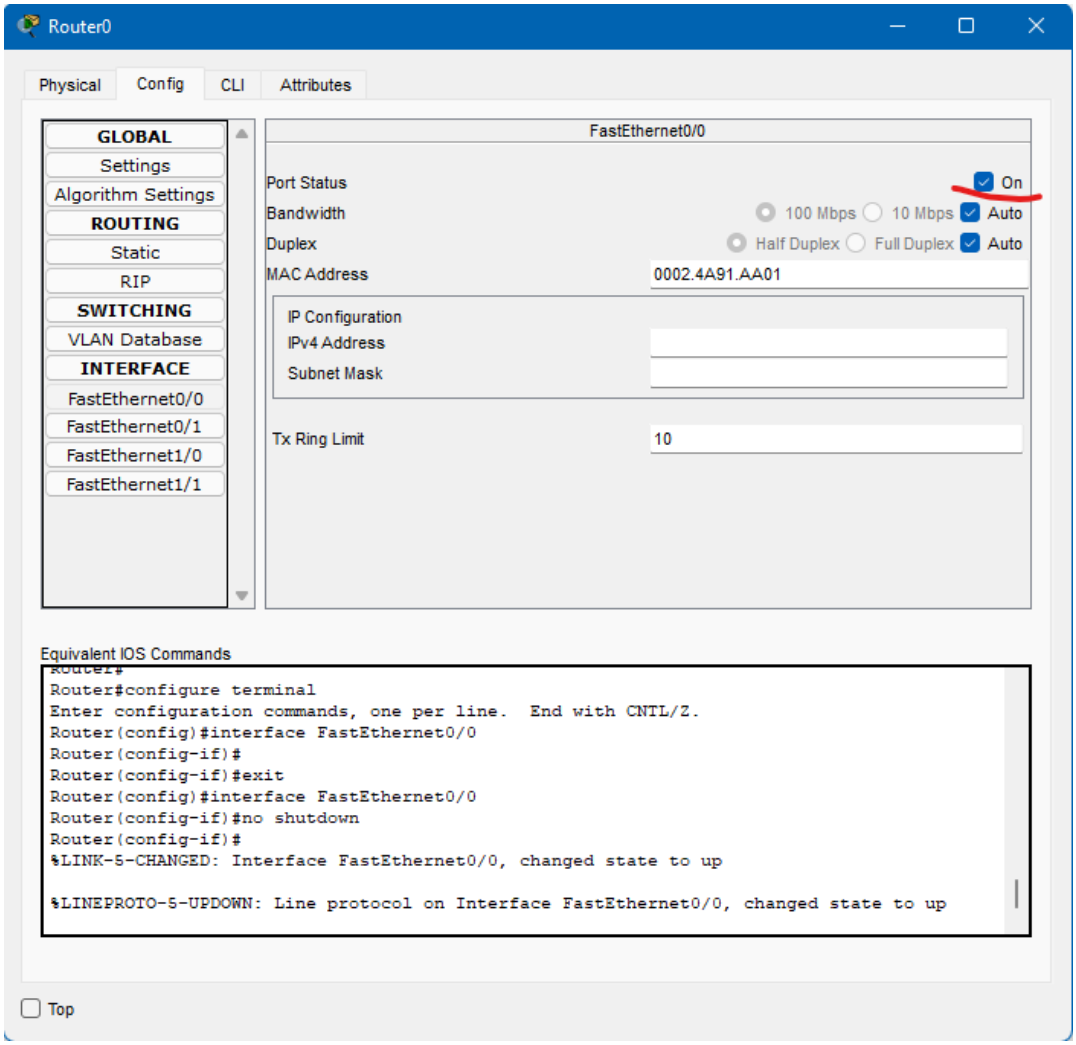


Рисунок 2.2 – Вікно налаштування маршрутизатора

Зробимо подібні операція для всіх портів, які ми будемо використовувати, як результат отримуємо готову для налаштування мережу:

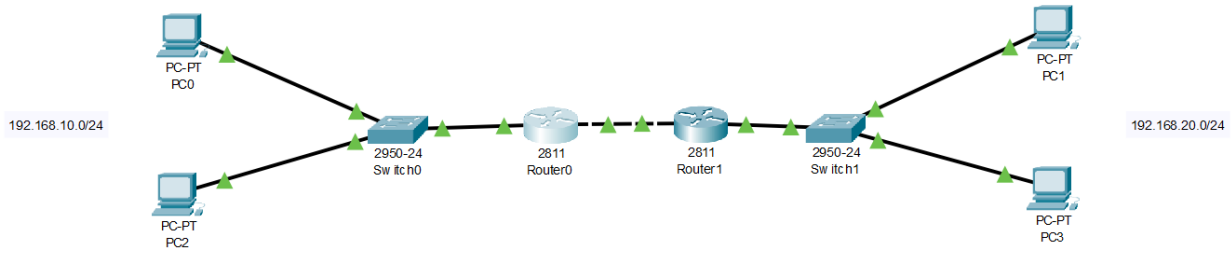


Рисунок 2.3 – Приклад з`єднаної мережі

Розпочнемо з налаштування DHCP для отримання IP-адрес на пристроях, які знаходяться в мережі. Оскільки не кожна корпорація може дозволити собі використовувати виділений DHCP сервер, ми будемо налаштовувати DHCP безпосередньо на роутерах.

Почнемо з роутера Router0, який знаходиться в мережі 192.168.10.0/24. Виділимо роутеру постійну IP-адресу в цій мережі для прогнозованої поведінки в майбутньому, задаємо наступний IP: 192.168.10.1 з маскою мережі 255.255.255.0. Ця операція може бути здійснена за допомогою наступних команд:

```
interface FastEthernet0/0
ip address 192.168.10.1 255.255.255.0
do wr
```

Та виділимо IP для зв`язку маршрутизаторів, які будуть використовувати мережу 20.20.20.0/8, для Router0 виділимо 20.20.20.1 з маскою 255.0.0.0, а для роутера 1 виділимо 20.20.20.2 з маскою 255.0.0.0, для зв`язку між ними будемо використовувати інтерфейс FastEthernet 1/1.

Після призначення IP-адрес маршрутизаторам, перевіримо правильність налаштувань за допомогою ICMP запиту, який будемо відсилати спочатку від Router0 до Router1 і навпаки за допомогою влаштованого симулятора відправки пакетів Packet Tracer.

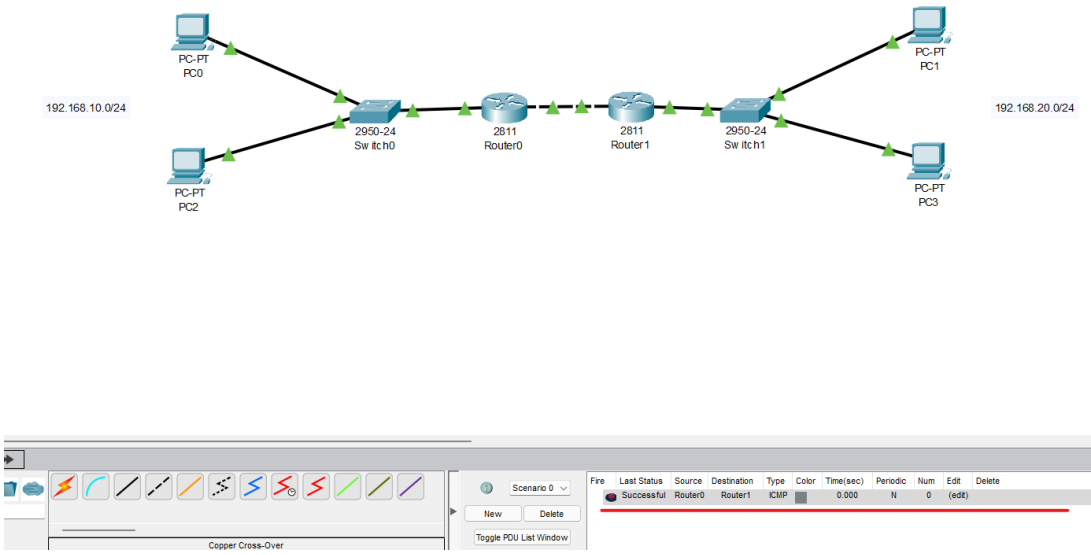


Рисунок 2.4 – Надсилання пакету ICMP від Router0 до Router1

Як можемо бачити перша перевірка успішна, тому переходимо до наступної:

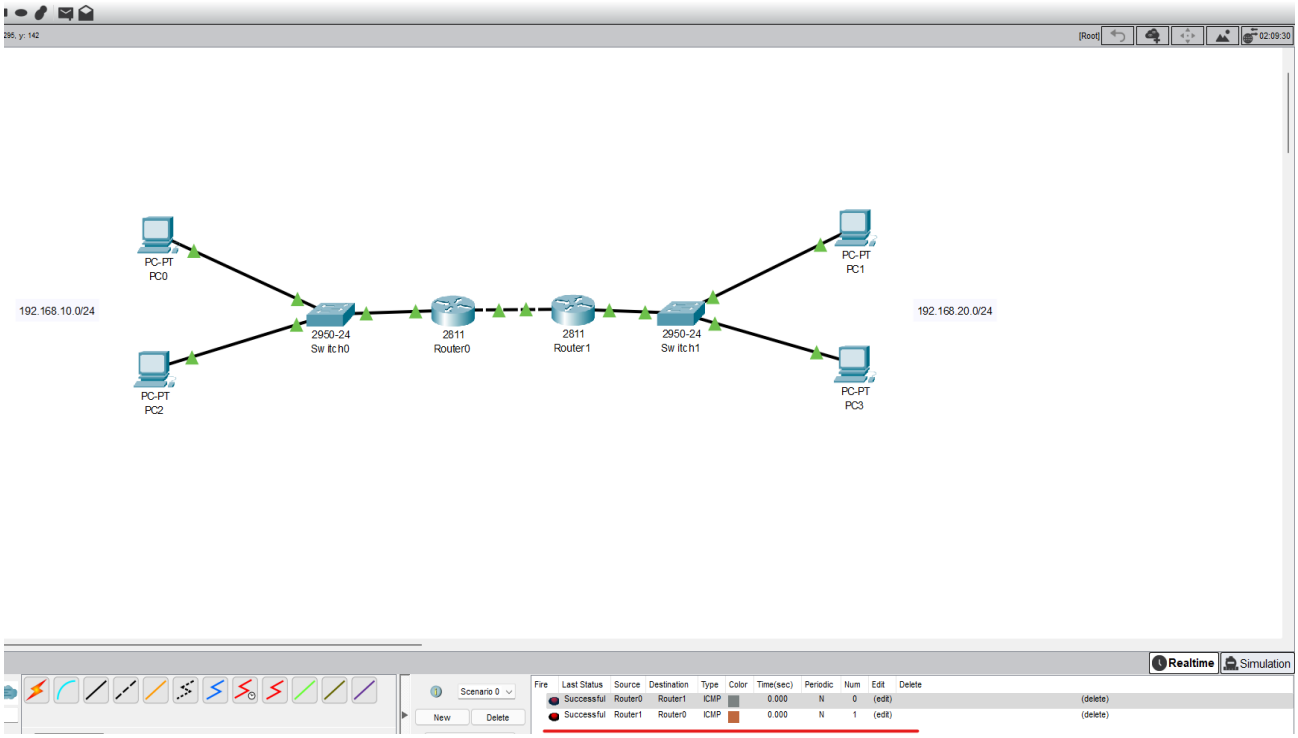


Рисунок 2.5 – Надсилання пакету ICMP від Router1 до Router0

Як можемо бачити обидві перевірки успішні тому можемо переходити до налаштування DHCP. Розпочнемо з Router0, будемо використовувати наступні команди:

```
en
configure t
ip dhcp excluded-address 192.168.10.1
ip dhcp pool R0
default-router 192.168.10.1
network 192.168.10.0 255.255.255.0
dns-server 8.8.8.8
```

За допомогою “ip dhcp excluded-address” виключаємо з пула DHCP IP-адресу роутера, для того щоб в мережі не створювалися конфлікти, “ip dhcp pool” створює DHCP пул на маршрутизаторі, командою “default-router” ми вказуємо IP-адресу роутера, на який буде спрямовуватися трафік в мережі, “dns-server” вказує адресу DNS сервера, який буде видаватися в мережі[15].

Повторимо аналогічні команди, але для Router1.

```
en
conf t
ip dhcp excluded-address 192.168.20.1
ip dhcp pool R1
default-router 192.168.20.1
network 192.168.20.0 255.255.255.0
dns-server 8.8.8.8
```

Після виконаних дій налаштування роутерів можна вважати завершеним. Для перевірки працездатності наших налаштувань, змінимо налаштування комп`ютерів, які представлені в мережі: PC0,PC1,PC2,PC3. Змінимо їх налаштування з статичних на динамічні

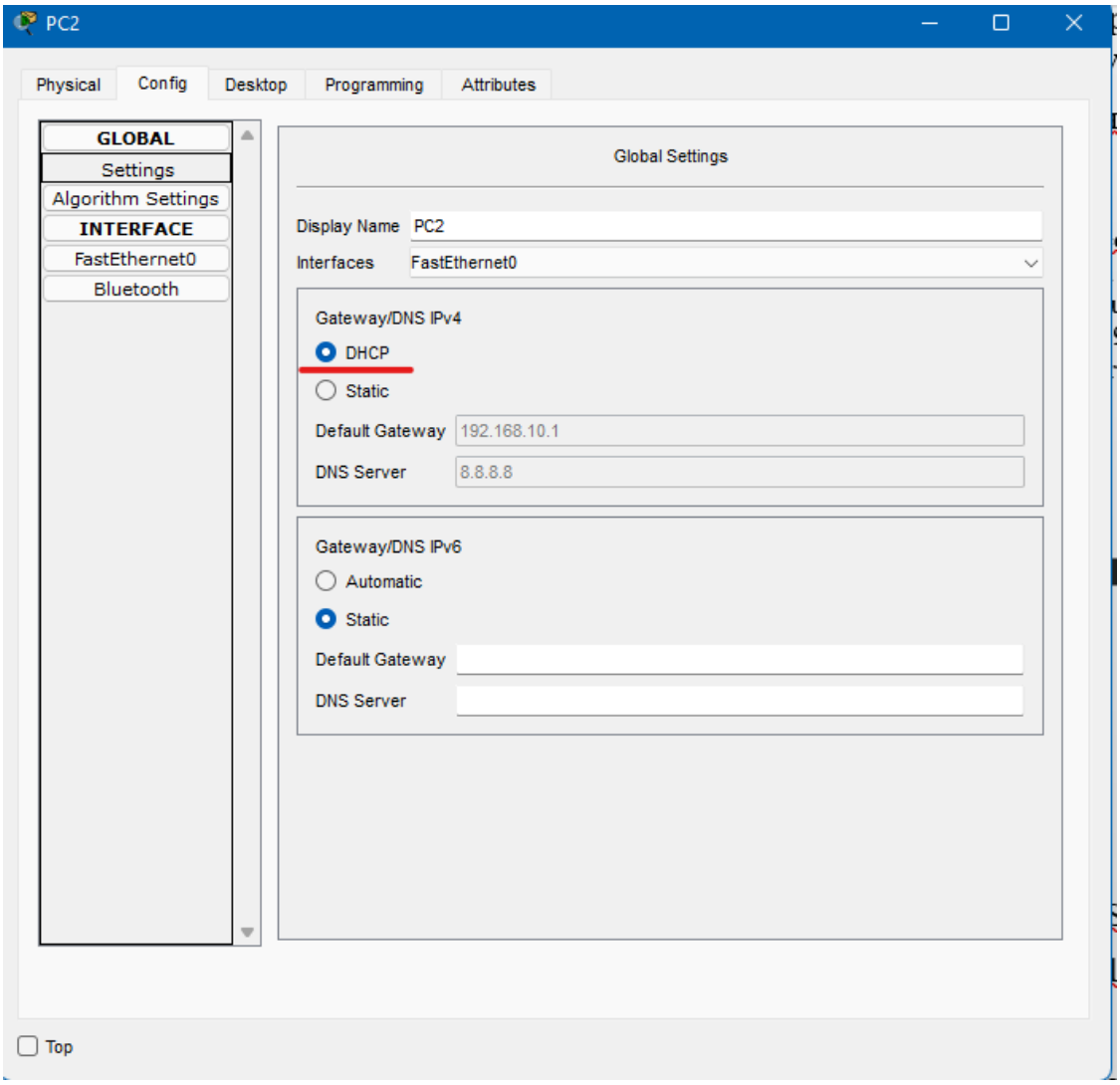


Рисунок 2.6 – Налаштування DHCP на комп`ютері

Після зміни налаштувань, перевіряємо чи отримують комп`ютери налаштування DHCP наводячи курсор на них:

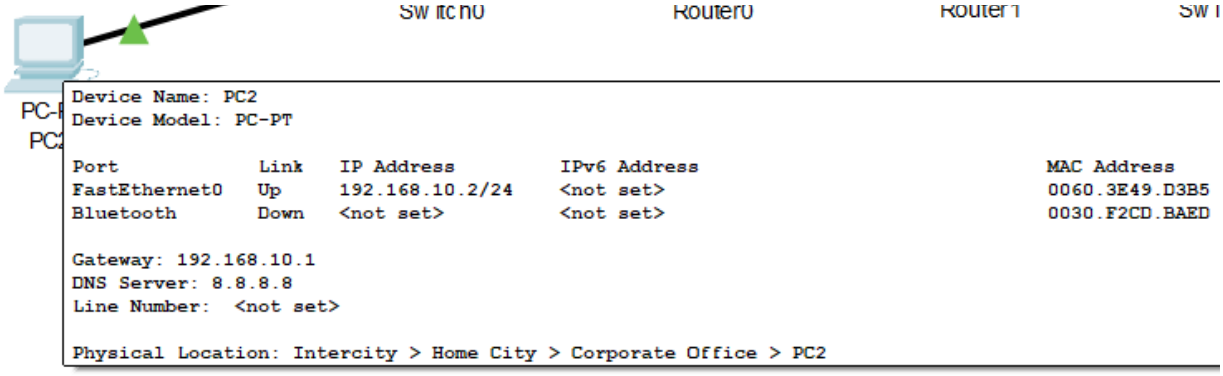


Рисунок 2.7 – Перевірка налаштувань DHCP

На цьому налаштування DHCP можна вважати завершеним.

Перейдемо до налаштування маршрутизації в мережі, оскільки 2 мережі не можуть спілкуватися між собою, оскільки не знають про існування один одного. Для конфігурування маршрутизації будемо використовувати протокол RIP (протокол інформації про маршрутизацію), він дозволить нам розказати роутерам, які мережі вони покривають і передавати цю інформацію іншим роутерам. Налаштуємо RIP спочатку на Router 0, потім на Router 1, за допомогою наступних команд:

```
en
conf t
router rip
version 2
network 192.168.10.0
network 20.20.20.0
do wr
```

За допомогою команди “network” ми вказуємо які мережі покриває роутер, а командою “router rip” ми включаємо протокол RIP на роутері. Використовуючи команду “do show ip route” ми можемо перевірити наші

налаштування на роутерах та побачити, які мережі проток RIP буде передавати.

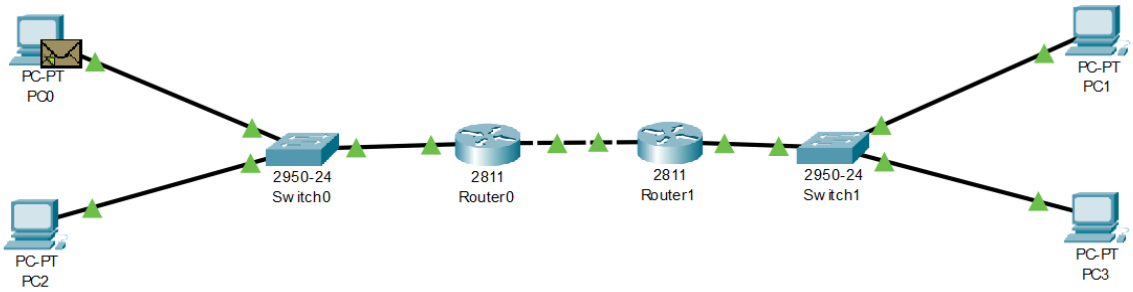
```
Router(config)#do show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       20.0.0.0/8 is directly connected, FastEthernet1/1
L       20.20.20.1/32 is directly connected, FastEthernet1/1
     192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.10.0/24 is directly connected, FastEthernet0/0
L       192.168.10.1/32 is directly connected, FastEthernet0/0
R    192.168.20.0/24 [120/1] via 20.20.20.2, 00:00:04, FastEthernet1/1

Router(config)#
```

Рисунок 2.8 – Перевірка налаштувань протоколу RIP



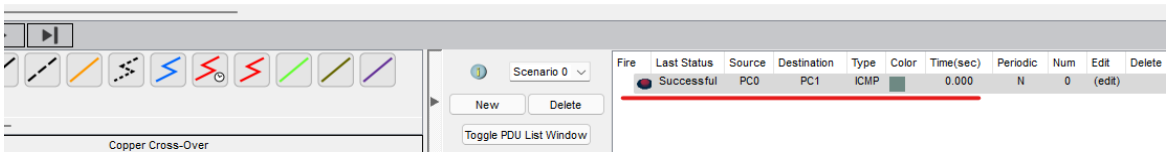


Рисунок 2.9 – Перевірка зв`язку між різними мережами

Як можемо бачити результати успішні отже на цьому налаштування протоколу маршрутизації RIP можна вважати завершеним.

Для максимально приближеного відтворення корпоративної мережі симулюємо наступні сервіси FTP, DNS, Email, Web для цього створимо в нашій мережі 4 сервера та назвемо їх відповідно.

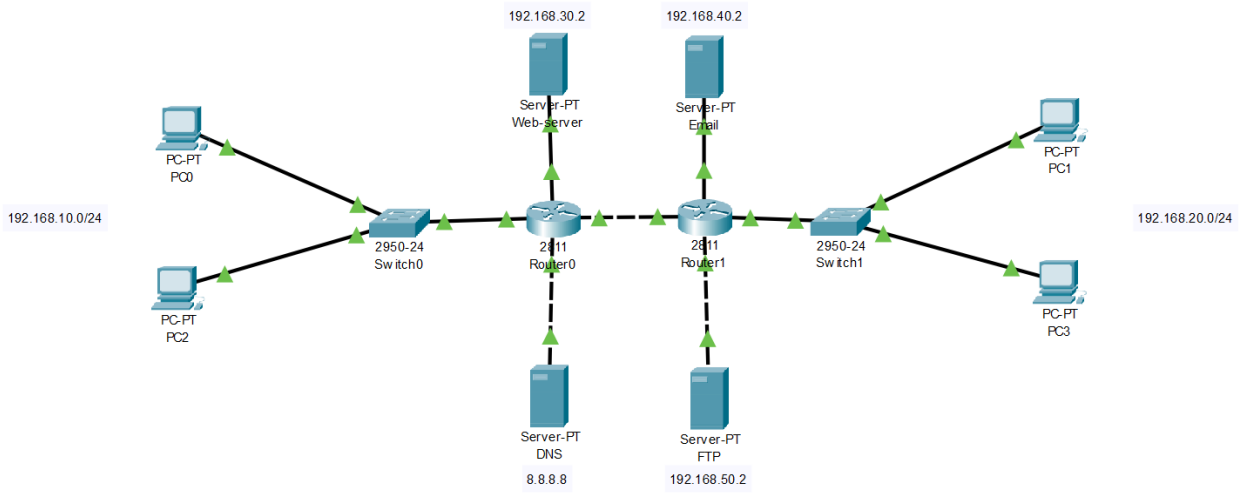


Рисунок 2.10 – Мережа з серверами FTP, DNS, Email, Web

Задаймо наступні IP-адреси створеним серверам:

- FTP – 192.168.50.2;
- DNS – 8.8.8.8;
- Web – 192.168.30.2;
- Email – 192.168.40.2.

Проставимо відповідні IP-адреси роутерам, які пов`язані з серверами, змінивши тільки останню цифру на 1.

Тепер налаштуємо ці сервіси за допомогою графічних форм Cisco Packet Tracer, які дозволяють провести симуляцію роботи цих сервісів, для перевірки як вони працюють в побудованій нами системі.

Розпочнемо з налаштувань DNS, оскільки в DHCP ми вже проставили DNS сервер за замовчуванням, то можемо перейти до безпосередньої конфігурації. Знаходимо в графічній формі DNS – серверу вкладку Services і включаємо тумблер “On”, після цього переходимо до заповнення DNS таблиці де спочатку ми вказуємо ім`я за яким інші користувачі мережі будуть

посилатися в веб-браузері, а потім вказуємо адресу за якою користувач буде переходити на необхідний йому сайт. Для прикладу, якщо користувач буде писати web.com в адресному рядку то він перейде на веб-сайт, який знаходиться за IP-адресом 192.168.30.2, таке ж правило справедливо і для інших сервісів.

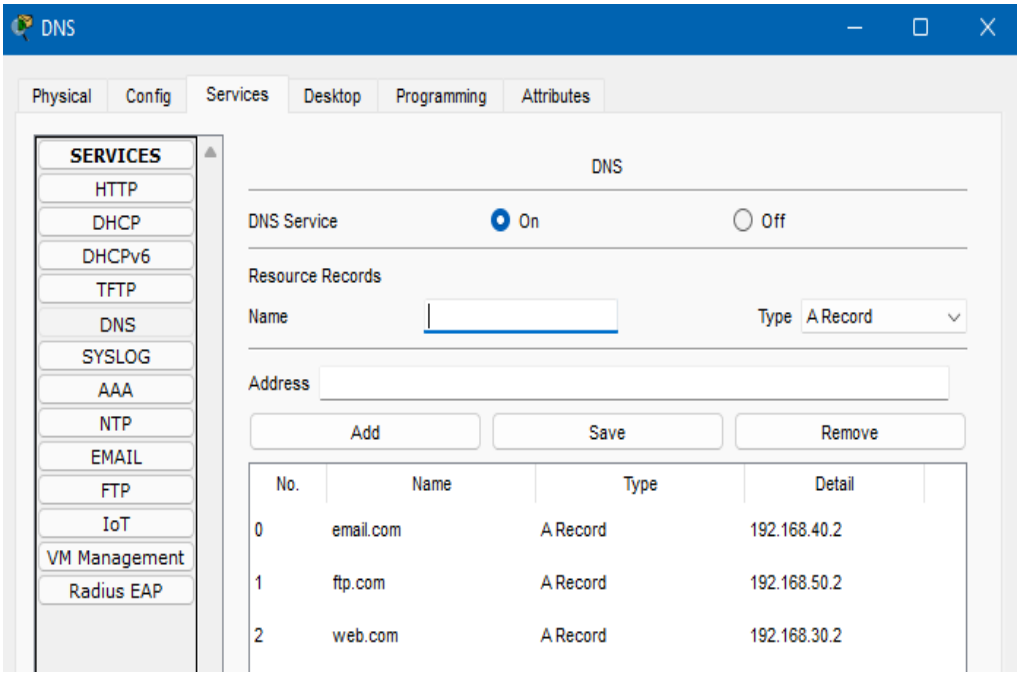


Рисунок 2.11 – Приклад налаштування DNS-серверу

Для перевірки роботи DNS спробуємо перейти з комп`ютера PC0 на сайт web.com за ім`ям, яке збережене в DNS таблиці.

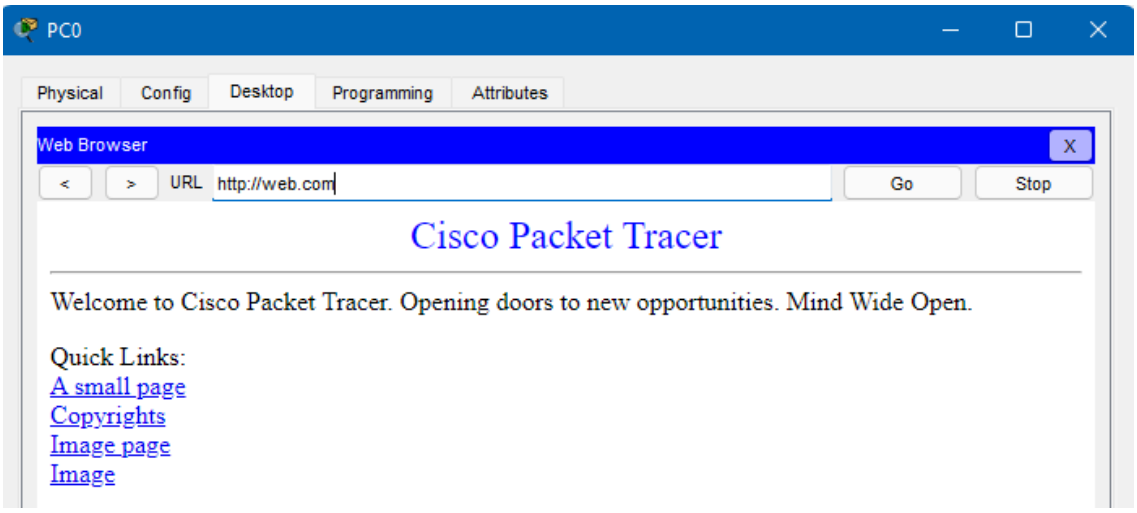


Рисунок 2.12 – Приклад використання DNS-серверу

Як можемо бачити DNS-сервер працює без помилок, тому переходимо до наступних сервісів.

Web-сервер в нашому прикладі зберігає сайти, які видає користувачу при безпосередньому зверненні.

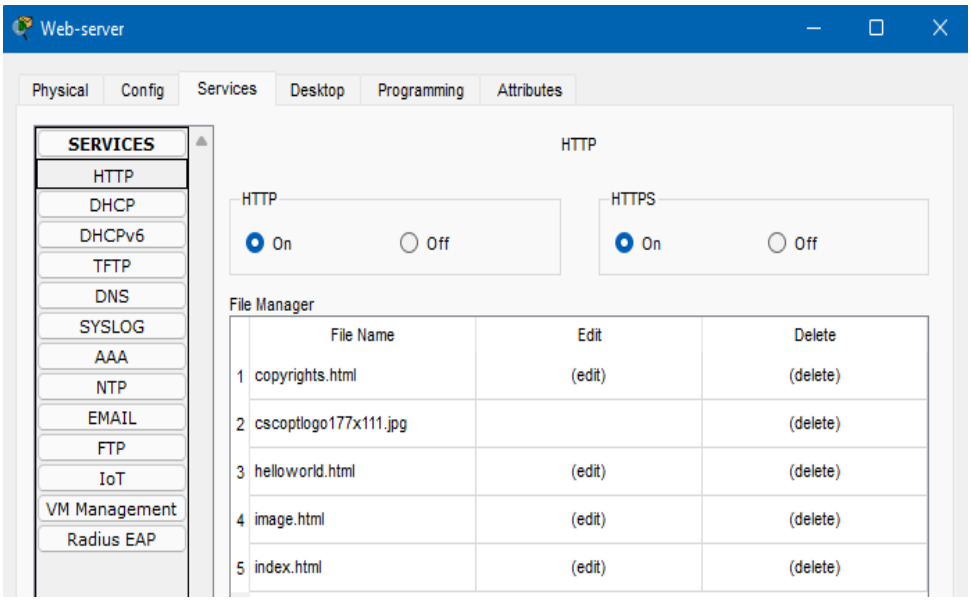


Рисунок 2.13 – Приклад налаштування WEB-серверу

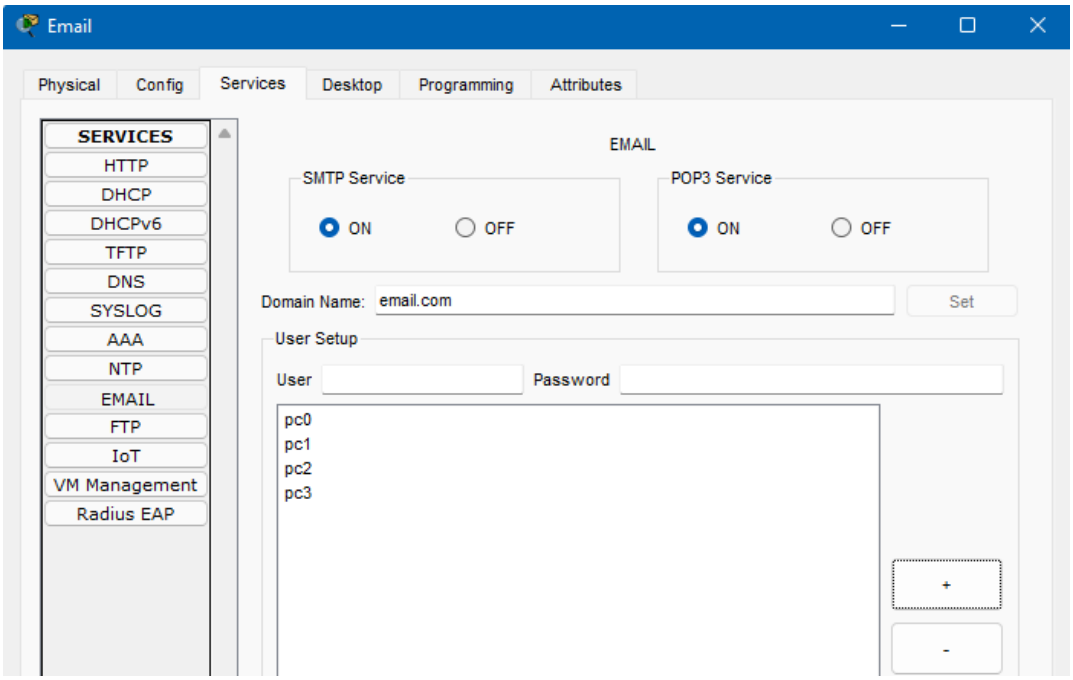


Рисунок 2.14 – Приклад налаштування Email-серверу

Також для роботи потрібно налаштувати на кожному з комп`ютерів особисту електрону скриню, яка буде використовувати Email-сервер, який ми створили.

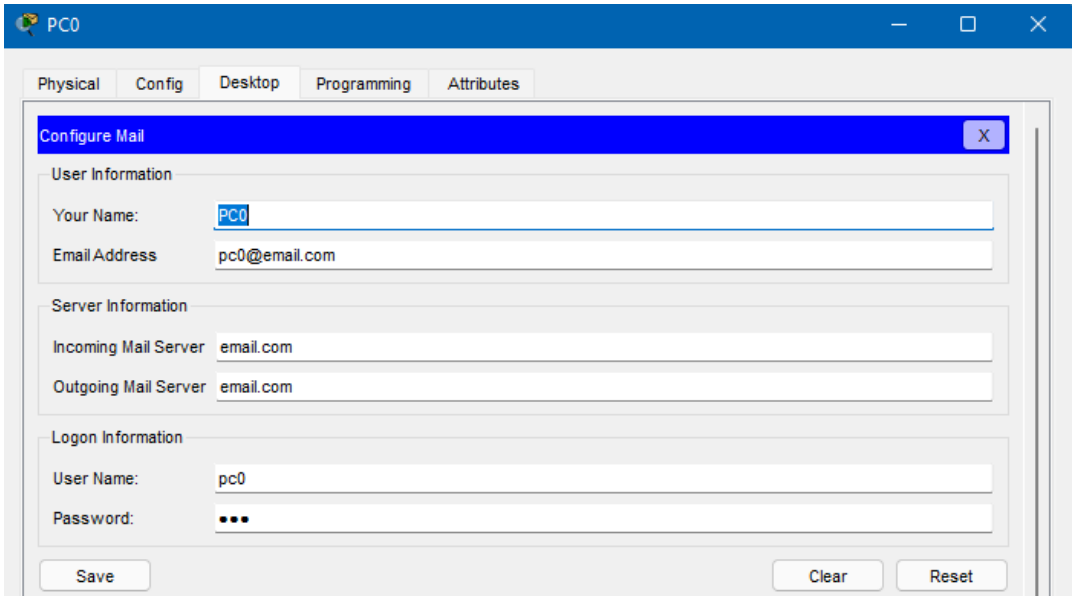


Рисунок 2.15 – Приклад налаштування Email на комп`ютері

Відправимо лист від імені користувача PC0, користувачу PC3.

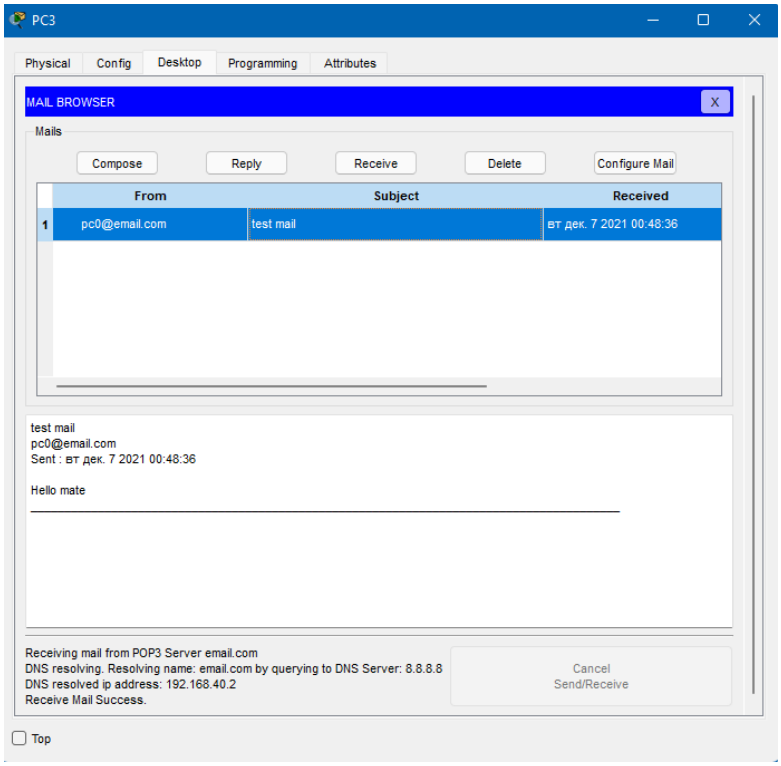


Рисунок 2.16 – Перевірка Email – сервісу.

Як можемо бачити користувач PC3 успішно отримав лист від користувача PC0.

Налаштуємо FTP – сервіс: перейдемо до графічного інтерфейсу FTP – серверу та створимо облікові дані для користувачів мережі з різними рівнями дозволу.

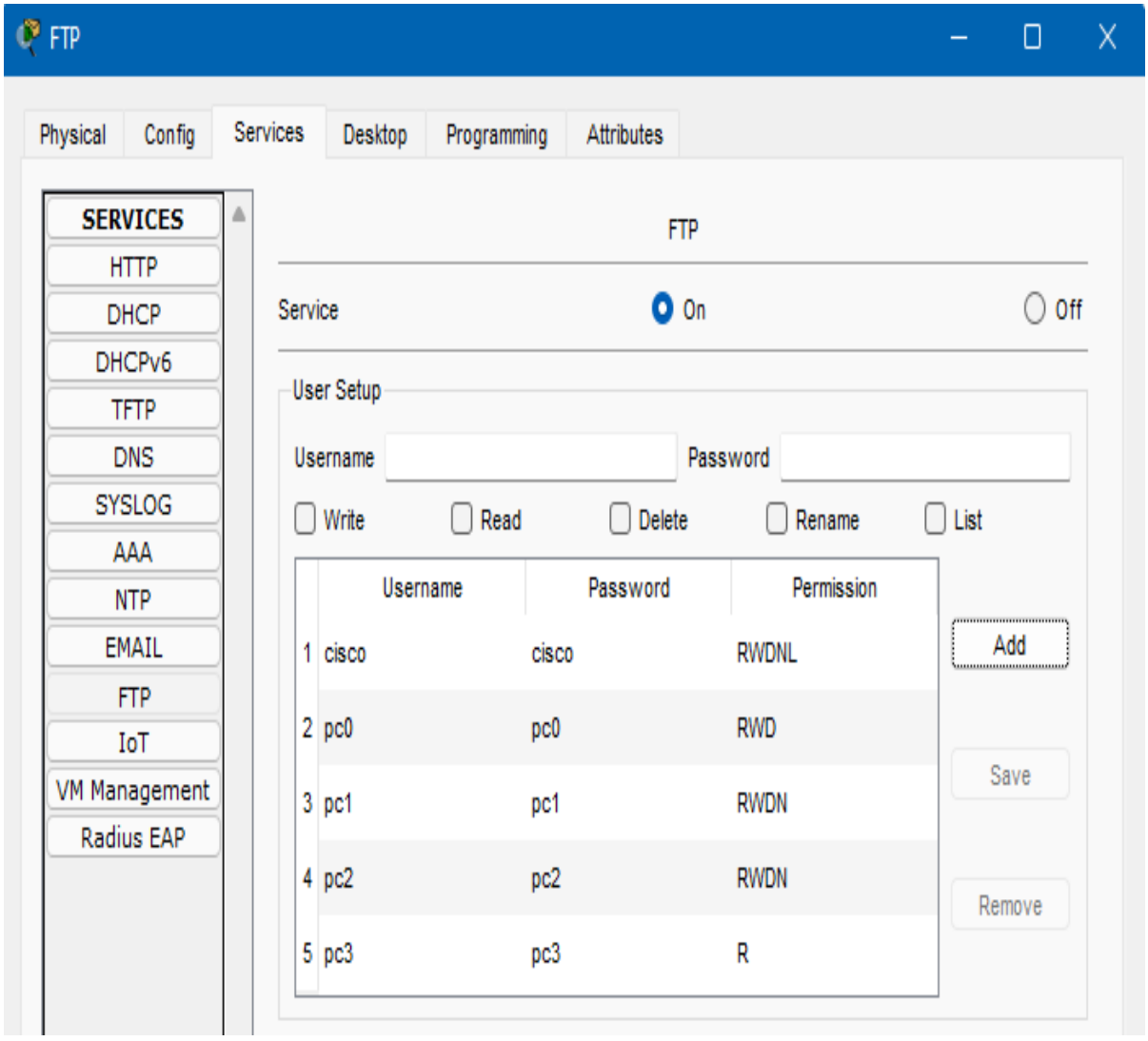


Рисунок 2.17 – Налаштування FTP – серверу

Для перевірки роботи цього сервісу спробуємо покласти заздалегідь створений текстовий файл користувача PC0 на FTP – сервер ftp.com. За допомогою наступних команд виконаних в командному рядку отримуємо результат:

ftp ftp.com

put test.file

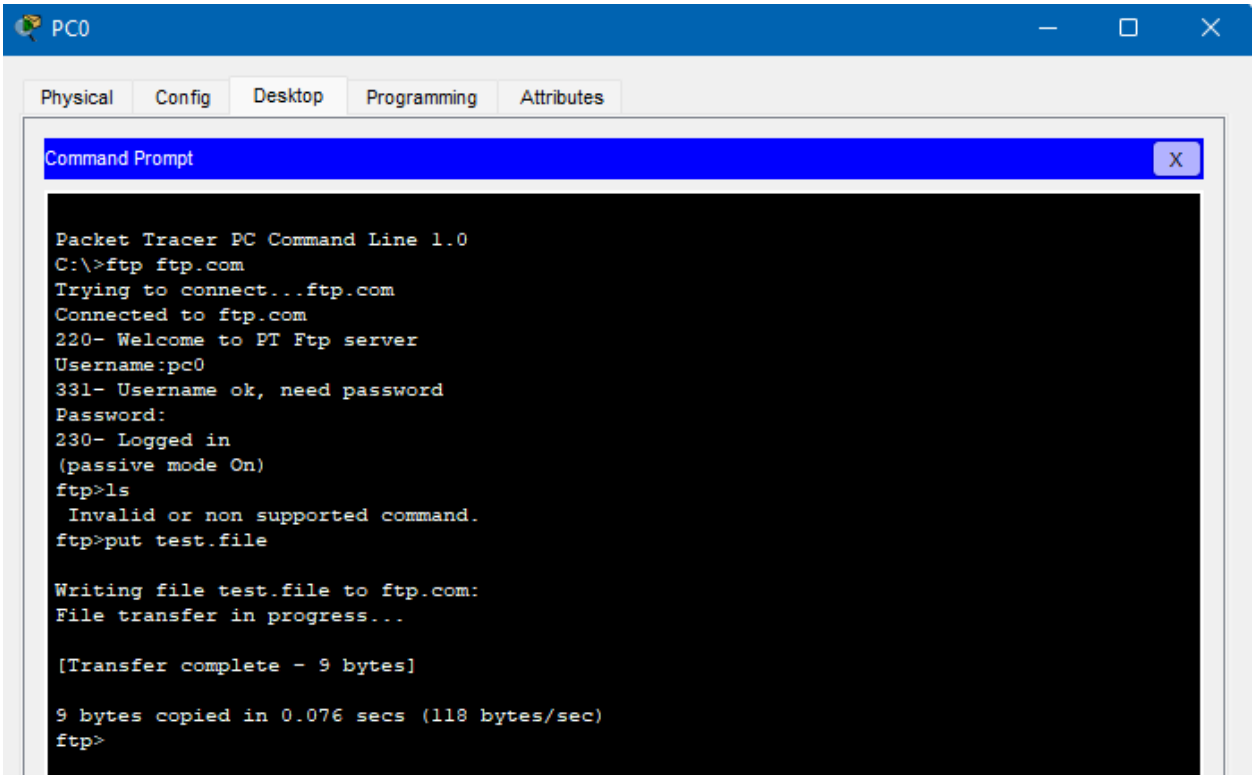


Рисунок 2.18 – Приклад роботи FTP-серверу

Перейдемо до налаштування VLAN в нашій мережі, це дозволить розділити певні мережі від один одного для збільшення безпеки. Для більш наглядної симуляції цієї технології добавимо по персональному комп`ютеру в обидві мережі PC4 і PC5 та почнемо з налаштування комутатора, за допомогою наступних команд:

Switch12>enable

Switch12#configure terminal

Switch12(config)#vlan 11

Створюємо VLAN з номером 11 на комутаторі, за допомогою аналогічних команд створимо VLAN 12. Потім для коректної роботи потрібно призначити інтерфейсам комутатора, які VLAN він буде використовувати, використовуючи наступні команди:

Switch12(config)#interface FastEthernet 0/2

Switch12(config-if)#switchport access vlan 11

Проводимо аналогічні дії для інших портів, які ми плануємо використовувати Fa 0/3, Fa0/4. Після цього нам потрібно буде вказати trunk

порт який буде акумулювати трафік всіх VLAN та надсилати дані роутеру, в нашому випадку це інтерфейс Fa0/1, тому за допомогою команд:

Switch12(config)#interface FastEthernet 0/1

Switch12(config-if)#switchport mode trunk

Ми вказуємо, що порт FastEthernet 0/1 буде працювати в режимі trunk. Для коректної роботи DHCP в нашому випадку ми повинні вказати IP-адресу кожному з VLAN-ів, для VLAN 11 вказуємо ip address 192.168.10.5 255.255.255.0, а для VLAN 12 вказуємо ip address 192.168.12.5 255.255.255.0, для більш явного виділення VLAN ми будемо використовувати мережу 192.168.12.5.

На цьому налаштування комутатора можна вважати завершеним, перейдемо до налаштувань маршрутизатора – розпочнемо з того, що для коректної роботи VLAN нам потрібно буде створити віртуальні порти, на інтерфейсі, який з`єднує маршрутизатор та комутатор, в нашому випадку це інтерфейс FastEthernet 0/0, тому почнемо з того що видалимо IP-адресу з цього інтерфейсу, потім створимо віртуальний порт FastEthernet 0/0.11 та задамо йому IP-адресу з мережі VLAN, яку ми будемо використовувати 192.168.10.1/24:

```
int fa0/0.11
encapsulation dot1Q 11
ip address 192.168.10.1 255.255.255.0
```

Аналогічно робимо і для VLAN 12.

Також для більш гнучкого налаштування мережі створимо access-list, який буде забороняти контактувати комп`ютерам з іншими мережами, які знаходяться ззовні, крім тих що ми дозволимо. Використовуючи команди:

En

Conf term

Access-list 11 permit 192.168.10.11 – цією командою ми створюємо access-list в якому дозволяємо зовнішнім мережам контактувати з комп`ютером за заданою ip адресою

Access-list 11 deny any – цією командою ми блокуємо доступ іншим комп`ютерам у зовнішні мережі

Int fa0/0.11

Ip access-group 11 in - цією є командою ми задаємо, що для цього інтерфейсу буде працювати політика дозволу яку ми створили раніше.

На цьому налаштування VLAN можна вважати завершеним перейдемо до тестування функціоналу. Спробуємо надіслати ICMP пакет від PC 0 до PC 1 який буде знаходитись в іншій мережі:

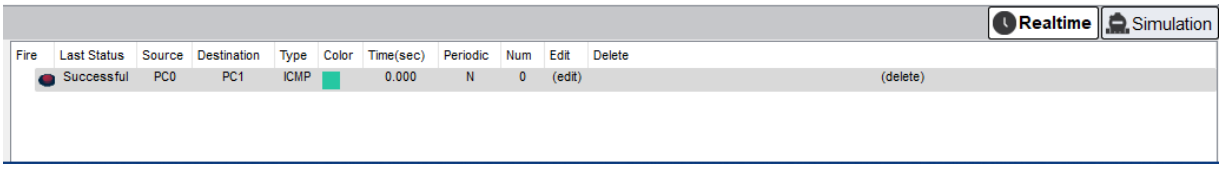


Рисунок 2.19 – Приклад роботи VLAN

Як бачимо пакет надіслано успішно, тому перейдемо до перевірки наших списків допуску, спробуємо відправити ICMP запит від PC 4, який знаходиться в тій же VLAN що і PC0, але за списком допуску він не має доступа до зовнішніх мереж:

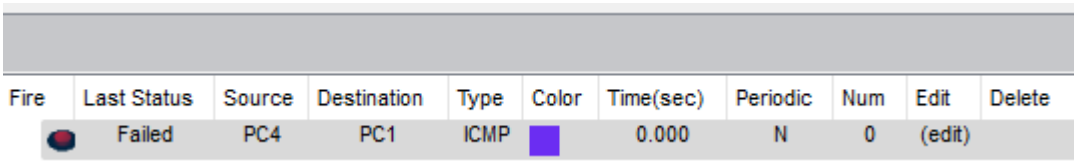


Рисунок 2.20 – Приклад роботи access-list

Як можемо бачити наші налаштування працюють коректно тому, на цьому конфігурування VLAN можна вважати завершеним.

VoIP є дуже зручним інструментом оскільки дозволяє налаштувати власну голосову лінію в офісі і заощаджувати на витратах на Телеком операторах. Перейдемо до конфігурації VoIP, додамо IP-телефони 7960, які присутні у Cisco packet tracer для симуляцій VoIP технологій. Для коректної

працездатності такого сервісу, нам необхідно буде створити окрему VLAN для передачі трафіку VoIP, налаштувати DHCP для того щоб телефони могли отримувати свій номер та мережеві налаштування та налаштуємо комутатор для опрацювання VoIP трафіку. Розпочнемо з налаштування DHCP і VLAN на роутері. Розпочнемо з маршрутизатора: створимо новий DHCP пул, який буде генерувати IP для телефонів, використовуючи наступні команди:

```
ip dhcp pool VOIP
network 192.168.100.0 255.255.255.0
default-router 192.168.100.254
option 150 ip 192.168.100.254
```

Після цього задамо, майбутню VLAN, яку ми будемо використовувати для передачі VoIP трафіку на віртуальному інтерфейсі FastEthernet 0/0.20:

```
interface FastEthernet0/0.20
encapsulation dot1Q 20
ip address 192.168.100.254 255.255.255.0
```

Та також додамо нову мережу до нашого протоколу RIP і створимо електронні номери, які будуть використовувати IP-телефони, наступні вказівки налаштовують сервіс телефонії на роутері:

```
telephony-service
max-ephones 5
max-dn 5
ip source-address 192.168.100.254 port 2000
```

Після активування цього сервісу ми можемо прописувати номера телефонів:

```
ephone-dn 1
 number 54001
```

Переходимо до налаштувань комутатора, який пов`язує пристрої мережі з маршрутизатором. Прописуємо наступні команди:

```
vlan 20
name VOIP
interface range FastEthernet0/2-4
switchport voice vlan 20
```

Після проведеної роботи, спостерігаємо що телефони отримали їх номер та IP

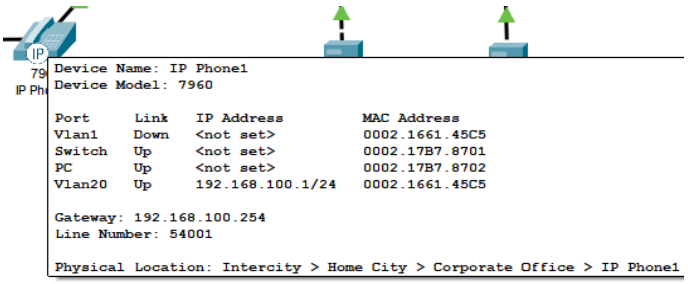


Рисунок 2.21 – Приклад конфігурації IP-телефона

Перевіряємо працездатність телефонів телефонуючи з одного телефона на інший

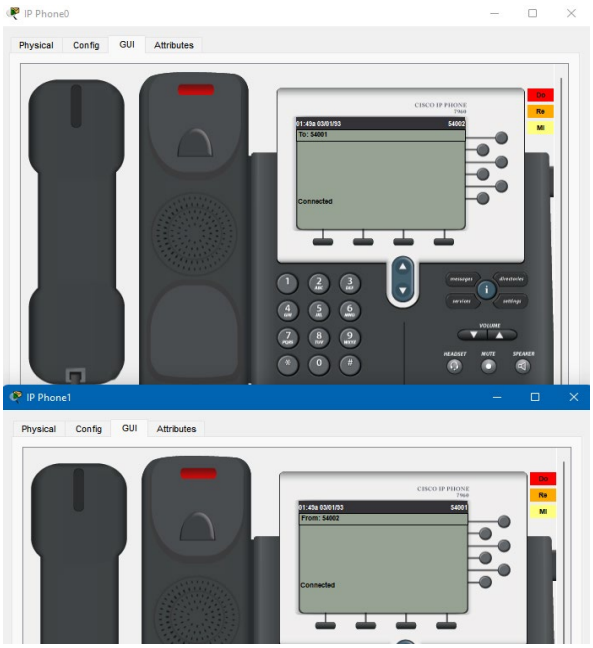


Рисунок 2.22 – Приклад роботи IP-телефона

Після аналогічних дій в іншій мережі, для того щоб вони мали можливість спілкуватися між собою через телефон потрібно вказати роутерам, в яку мережу відправляти запит на з`єднання для певних патернів номера, для цього прописуємо наступне:

```
dial-peer voice 1 voip
 destination-pattern 4300.
 session target ipv4:192.168.101.254
```

Проводимо дзеркальні дії для іншої мережі і перевіряємо правильність наших налаштувань телефонуючи на номер, який знаходиться в іншій мережі:

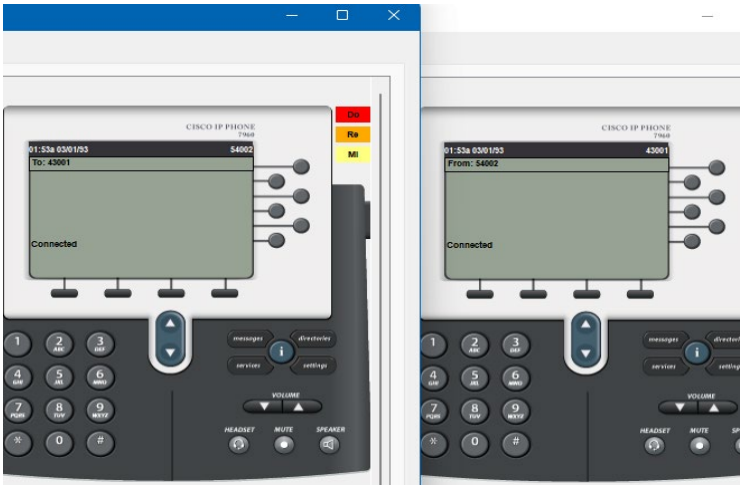


Рисунок 2.23 – Приклад роботи IP-телефона в різних мережах

Як результат отримуємо працюючий сервіс VoIP у двох різних мережах.

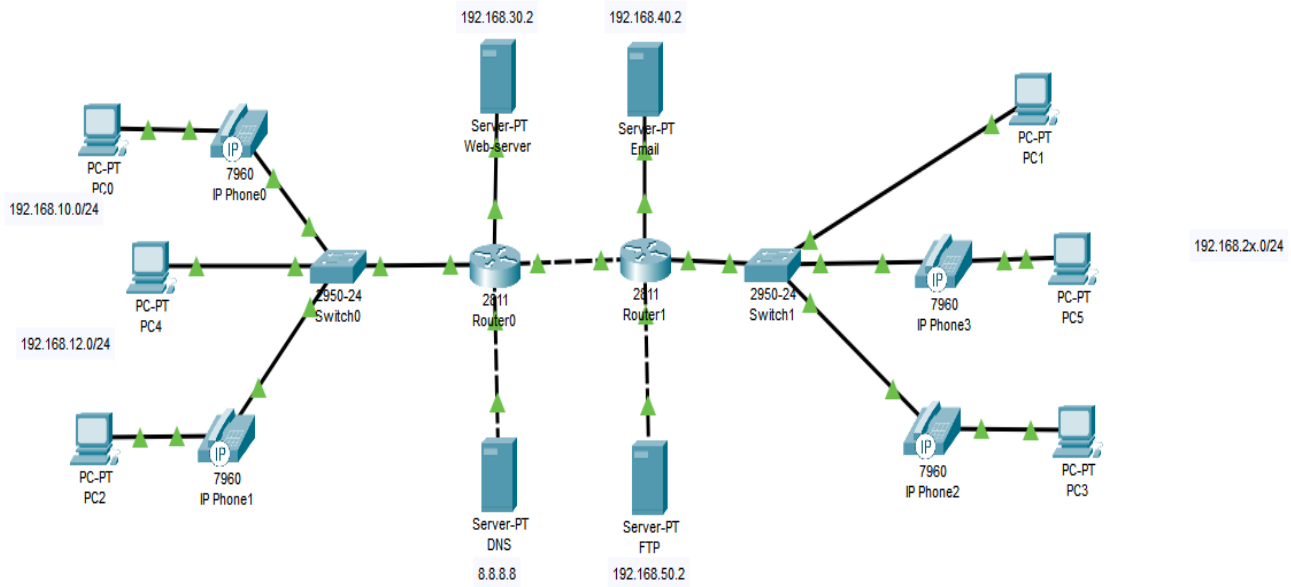


Рисунок 2.24 – Приклад завершеної корпоративної мережі

2.3 Розробка веб-орієнтованої системи використовуючи мову програмування Python

Для реалізації задуманого веб-інтерфейсу, було вирішено використовувати інтерпретовану об'єктно-орієнтовану мову програмування Python. Python став однією з найшвидших і найпопулярніших мов програмування у світі.

Python універсальний, простий у використанні та розробці. Однак у Python є кілька недоліків, які слід враховувати, наприклад, обмеження швидкості. Давайте подивимося на плюси та мінуси Python

Ось деякі найважливіші переваги Python:

- Універсальний, простий у використанні та швидко розвивається, Python зосереджується на читабельності коду. Мова є універсальною, акуратною, легкою у використанні та навчанні, читабельною та добре структурованою.
- Python динамічно типізується, що робить його зручним і швидшим у розробці, забезпечуючи REPL.
- Завдяки гнучкості Python легко проводити дослідницький аналіз даних — в основному шукати голки в копиці сіна, коли ви не впевнені, що це за голка. Python дозволяє використовувати найкраще з різних парадигм програмування. Він об’єктно-орієнтований, але також активно використовує функції функціонального програмування.
- Ви можете безкоштовно завантажити Python і написати код за лічені хвилини. Розробка за допомогою Python не викликає проблем.

Досвідчені програмісти завжди рекомендують використовувати правильні інструменти для проекту. Корисно знати не лише переваги Python, а й його недоліки.

Основні недоліки Python:

- Обмеження швидкості, Python є інтерпретованою мовою, тому ви можете виявити, що вона повільніша, ніж деякі інші популярні мови. Але якщо швидкість не є найважливішим фактором для вашого проекту, то Python вам варіант.
- Проблеми з багатопоточністю, потоки не дуже оптимально використовуються в Python через глобальне блокування інтерпретатора (GIL). GIL — це мьютекс, який дозволяє одночасно виконувати лише один потік. У результаті багатопотокові програми, пов’язані з процесором, можуть працювати повільніше, ніж однопотокові.
- Споживання пам'яті, слід враховувати, що споживання пам’яті Python дуже велике. З цієї причини це може бути не найкращим вибором для завдань із інтенсивною пам’яттю. Це може бути проблематично, коли в ОЗУ активна велика кількість об’єктів.

Для реалізації поставленої задачі Python відповідає усім вимогам

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Розробка графічного інтерфейсу налаштування корпоративних мультисервісних мереж

Мультисервісна мережа була симульована в емуляторі Cisco Packet Tracer, було проведено налаштування таких сервісів, як DHCP, VLAN, VoIP, FTP, Email, DNS і протоколу маршрутизації RIP 2 версії, всі налаштування проводились на симульованому обладнанні Cisco та може бути перенесено на реальне обладнання без будь-яких проблем. Після проведеної роботи були знайдені деякі недоліки цього симулятора: рутинне налаштування різних мережевих пристроїв займає багато часу та потребує пропрієтарних знань, також не всі сервіси налаштовуються повністю через графічні форми. Саме через це розробка веб-орієнтованої інформаційної системи є дуже актуальною задачею на сьогодні для автоматизації процесу сервісів, які потрібні для функціонування мультисервісних мереж.

Проект було реалізовано з використанням мов програмування Python і JavaScript, мови розмітки HTML і каскадних таблиць CSS.

Інтерфейс системи вийшов простим і зрозумілим для рядового користувача, який дозволяє користувача без перепон використовувати цей інтерфейс для отримання налаштувань.

При відкритті програми користувач потрапляє на головну сторінку та спостерігає 6 полів для ведення інформації і схеми на яку можна орієнтуватися, щоб краще розуміти для якого пристрою різні конфігурації будуть призначатися.

Користувач має вибір використовувати стандартні значення для мереж, які будуть конфігуруватися базуючись на них, або ж провести деякі зміни і вибрати IP-адреси які підходять для його задачі і його мережі. Після заповнення всіх полів вводу даних, користувач натискає кнопку “Get Configuration” і користувач отримує потрібні йому налаштування для мережевих пристроїв.

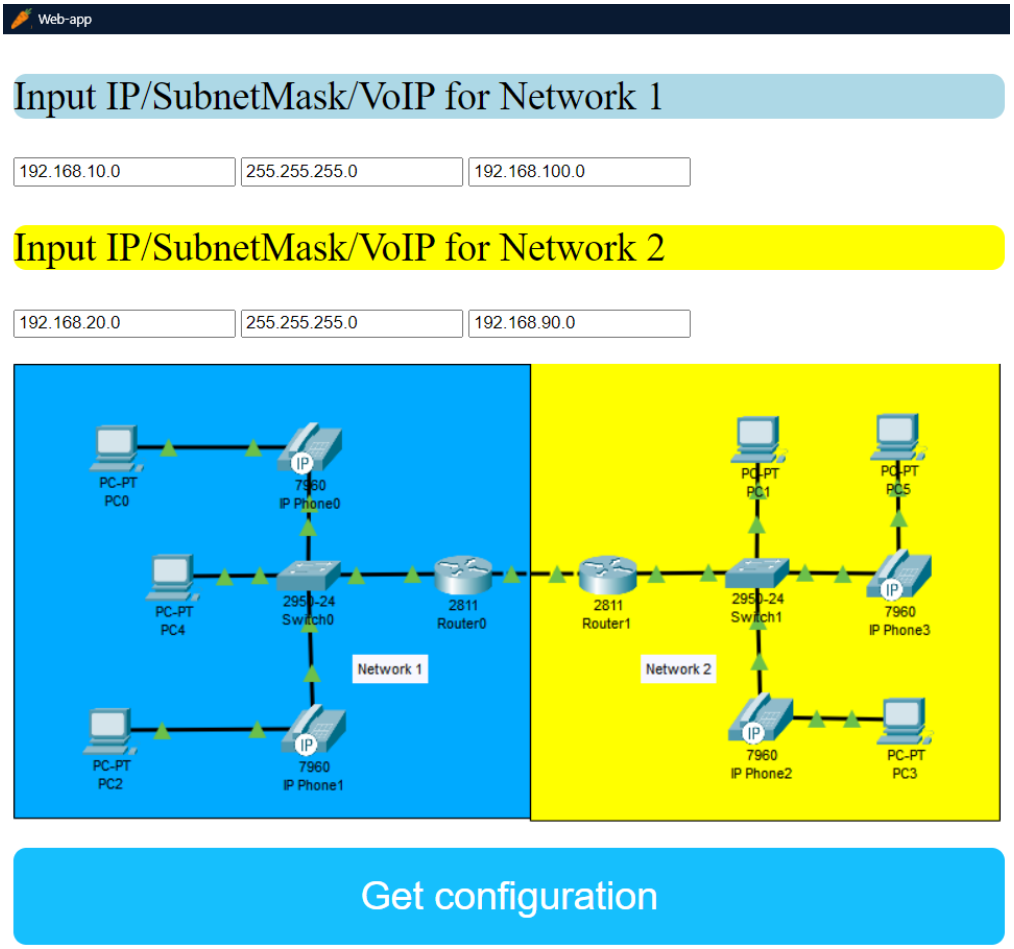


Рисунок 3.1 – Інтерфейс розробленої веб-системи

Router 0

```
enable
conf t
interface FastEthernet0/0
no ip address
duplex auto
speed auto
no sh

interface FastEthernet0/0.11
encapsulation dot1Q 11
ip address 192.168.10.1 255.255.255.0
ip access-group 11 in
no sh

interface FastEthernet0/0.20
encapsulation dot1Q 20
ip address 192.168.100.254 255.255.255.0
interface FastEthernet1/1
ip address 20.20.20.1 255.0.0.0
duplex auto
speed auto
no sh

router rip
version 2
network 20.0.0.0
network 192.168.10.0
network 192.168.100.0

telephony-service
max-ephones 5
max-dn 5
ip source-address 192.168.100.254 port 2000
auto assign 4 to 6
auto assign 1 to 5
ephone-dn 1
number 54001
ephone-dn 2
number 54002
ephone-dn 3
number 54003

ip dhcp pool voip
network 192.168.100.0 255.255.255.0
default-router 192.168.100.254
option 150 ip 192.168.100.254
```

Router 1

```
enable
conf t
interface FastEthernet0/0
no ip address
duplex auto
speed auto
no sh

interface FastEthernet0/0.11
encapsulation dot1Q 11
ip address 192.168.20.1 255.255.255.0
ip access-group 11 in
no sh

interface FastEthernet0/0.20
encapsulation dot1Q 20
ip address 192.168.90.254 255.255.255.0
interface FastEthernet1/1
ip address 20.20.20.1 255.0.0.0
duplex auto
speed auto
no sh

router rip
version 2
network 20.0.0.0
network 192.168.20.0
network 192.168.90.0

telephony-service
max-ephones 5
max-dn 5
ip source-address 192.168.90.254 port 2000
auto assign 4 to 6
auto assign 1 to 5
ephone-dn 1
number 43001
ephone-dn 2
number 43002
ephone-dn 3
number 43003

ip dhcp pool voip
network 192.168.90.0 255.255.255.0
default-router 192.168.90.254
option 150 ip 192.168.90.254
```

Switch 0

```
ip dhcp excluded-address 192.168.10.1 192.168.10.10
ip dhcp pool vlan11
network 192.168.10.0 255.255.255.0
default-router 192.168.10.0
dns-server 8.8.8.8
int range of fa0/2-12
switchport access vlan 11
switchport voice vlan 20
int vlan11
ip address 192.168.10.5 255.255.255.0
```

Switch 1

```
ip dhcp excluded-address 192.168.20.1 192.168.20.10
ip dhcp pool vlan11
network 192.168.20.0 255.255.255.0
default-router 192.168.20.0
dns-server 8.8.8.8
int range of fa0/2-12
switchport access vlan 11
switchport voice vlan 20
int vlan11
ip address 192.168.20.5 255.255.255.0
```

Рисунок 3.2 – Результат виконання системи

Тепер після того, як користувач отримує налаштування мережевих пристроїв він може використовувати їх на власних мережевих пристроях.

3.2 Тестування веб-орієнтованої інформаційної системи в симуляторі Cisco Packet Tracer

Для перевірки розробленої системи протестуємо її в мережевому симуляторі Cisco Packet Tracer.

Будемо перевіряти роботу сервісів DHCP, VLAN та VoIP, бо саме ці сервіси можливо реалізувати безпосередньо на обладнанні Cisco.

Заповнюємо або використовуємо стандартні значення мережі:

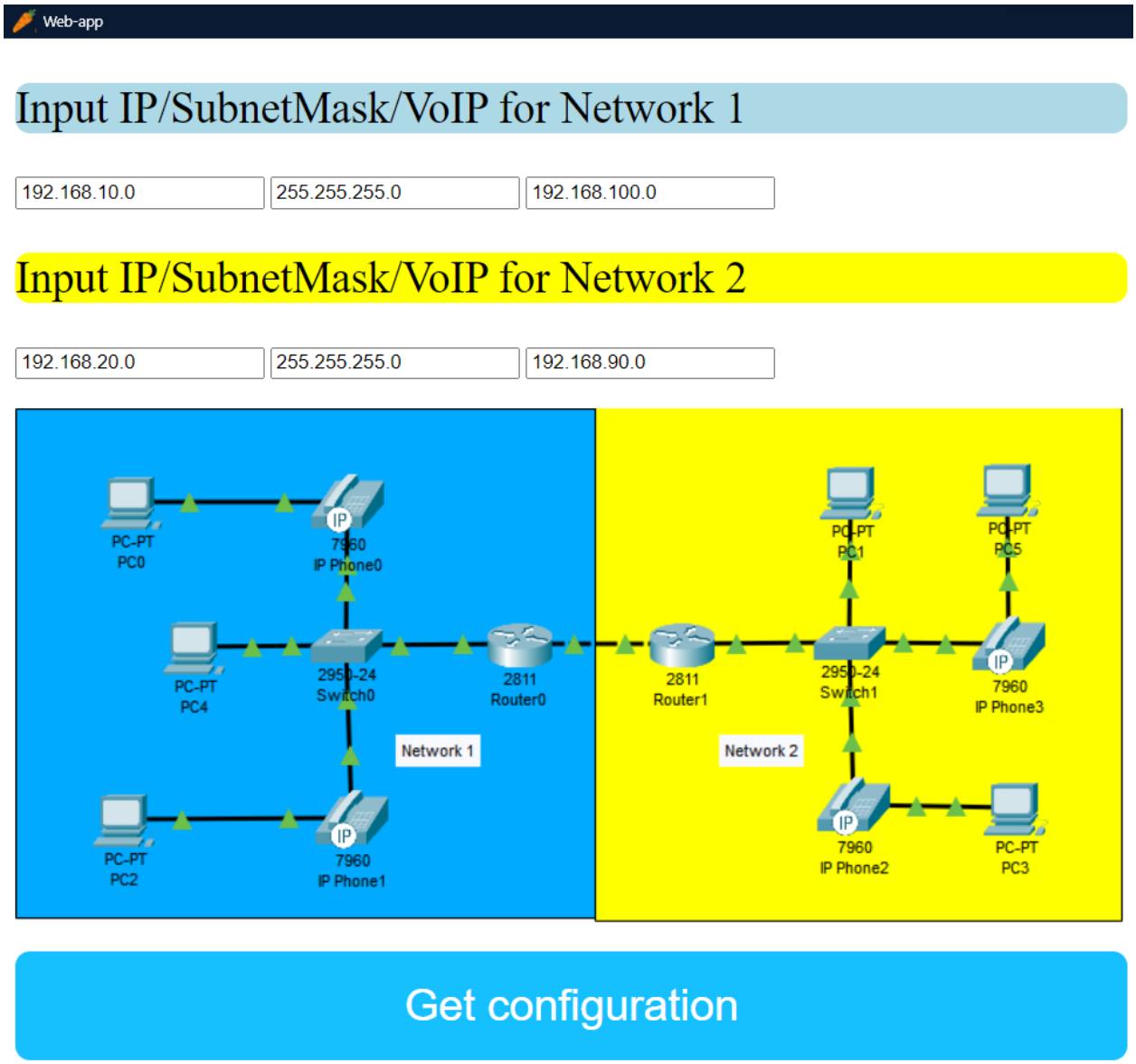


Рисунок 3.3 – Заповнені поля мережі в графічному інтерфейсі

Натискаємо “Get Configuration” та отримуємо згенеровані налаштування для маршрутизаторів та комутаторів, які будуть використовуватись для нашої мережі в Cisco Packet Tracer.

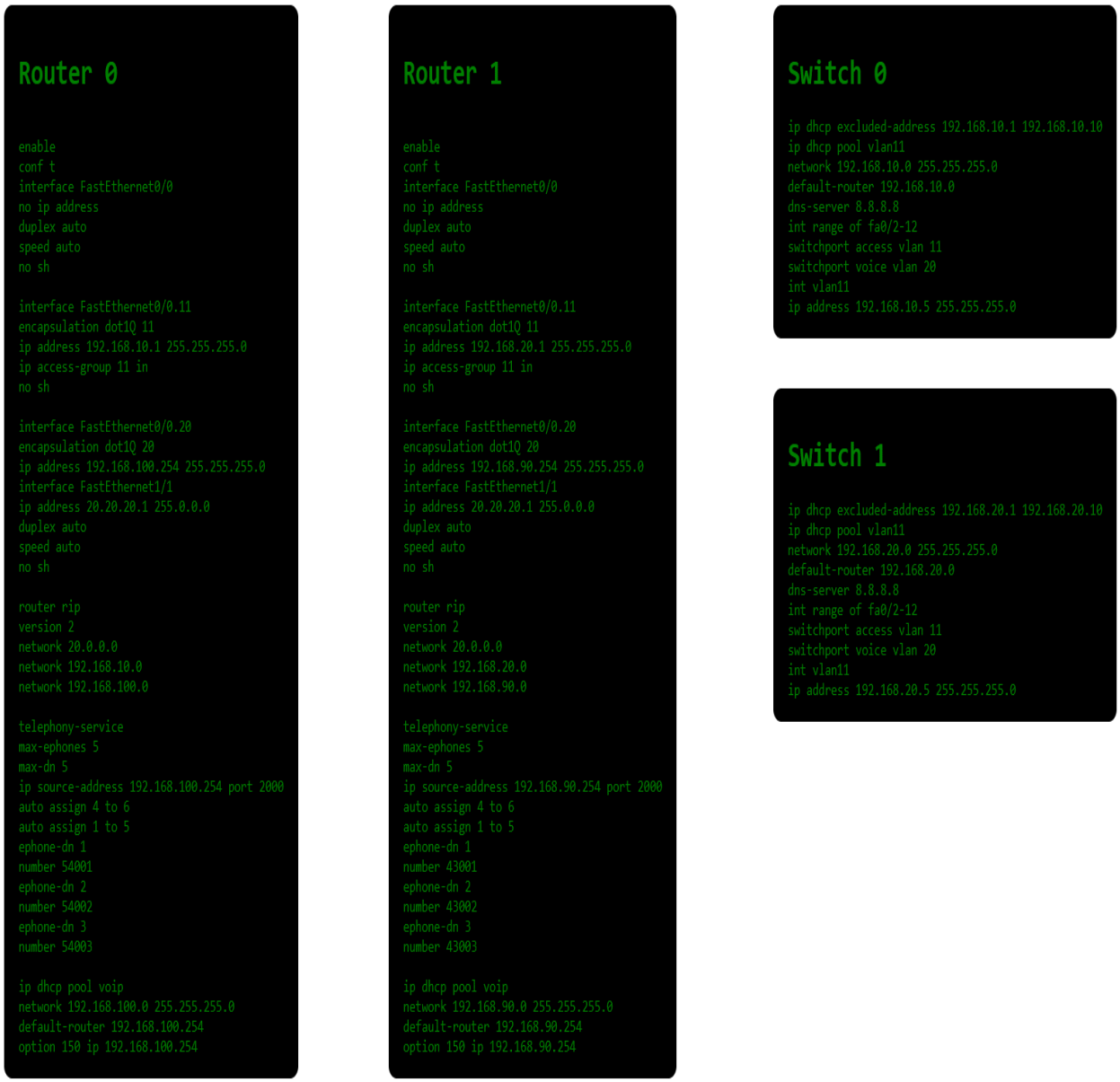


Рисунок 3.4 – Отриманий набір команд для мережевих пристроїв

Після отримання команд переходимо до симулятора Cisco Packet Tracer. Відтворюємо схему, як на головній сторінці системи та переносимо отримані команди на пристрої, розпочнемо з Router0 та перенесемо отримані команди для роутера в командне вікно конфігурування роутера, після вставки результатів бачимо, що інтерфейси які були неактивні, активувалися, та на схемі також з`явився зв`язок між комутатором та маршрутизатором:

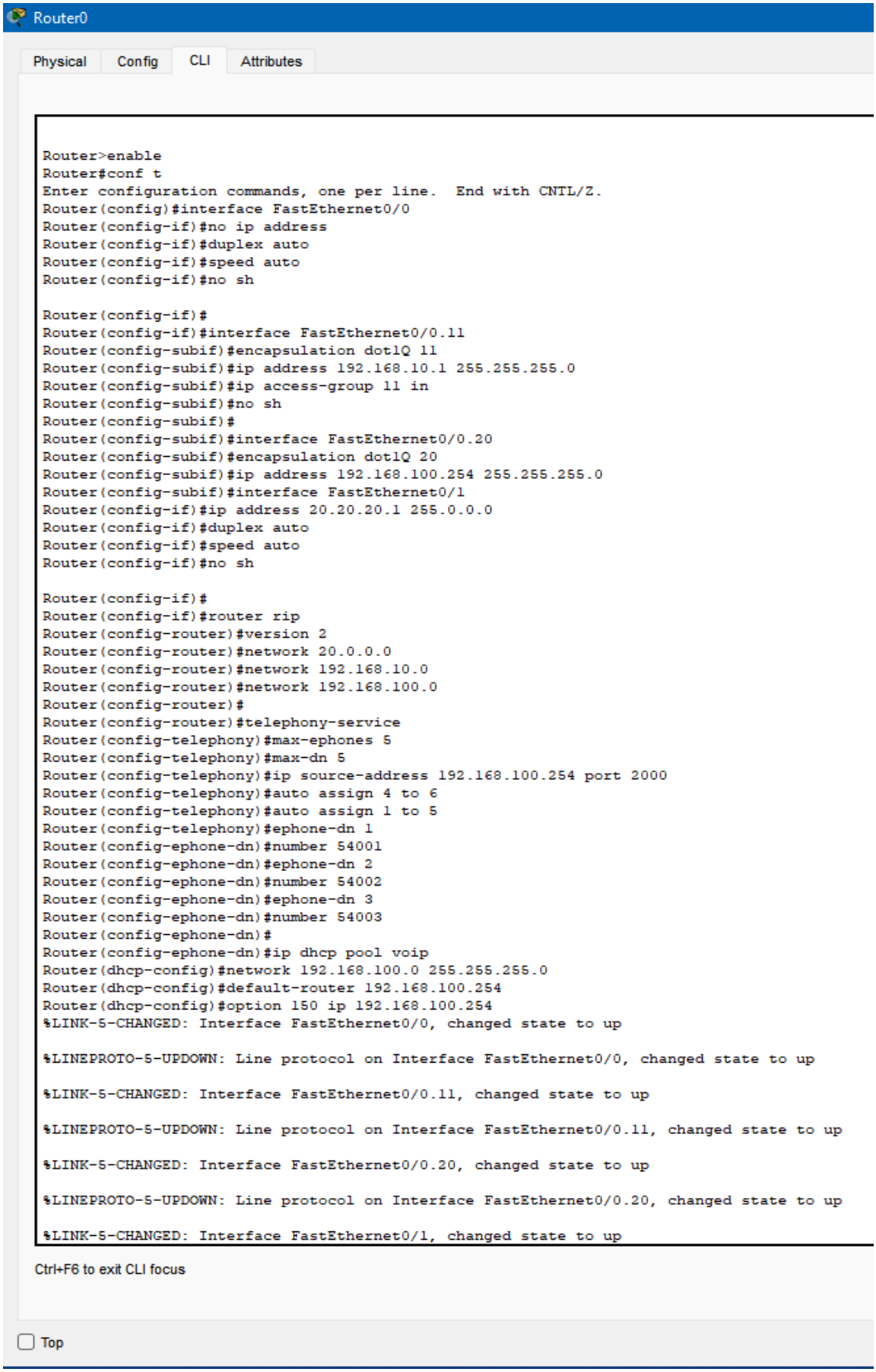


Рисунок 3.5 – Вікно налаштувань першого роутера

Аналогічно налаштовуємо інший маршрутизатор та переходимо до налаштування комутатора експортуємо налаштування до комутатора таким же чином:

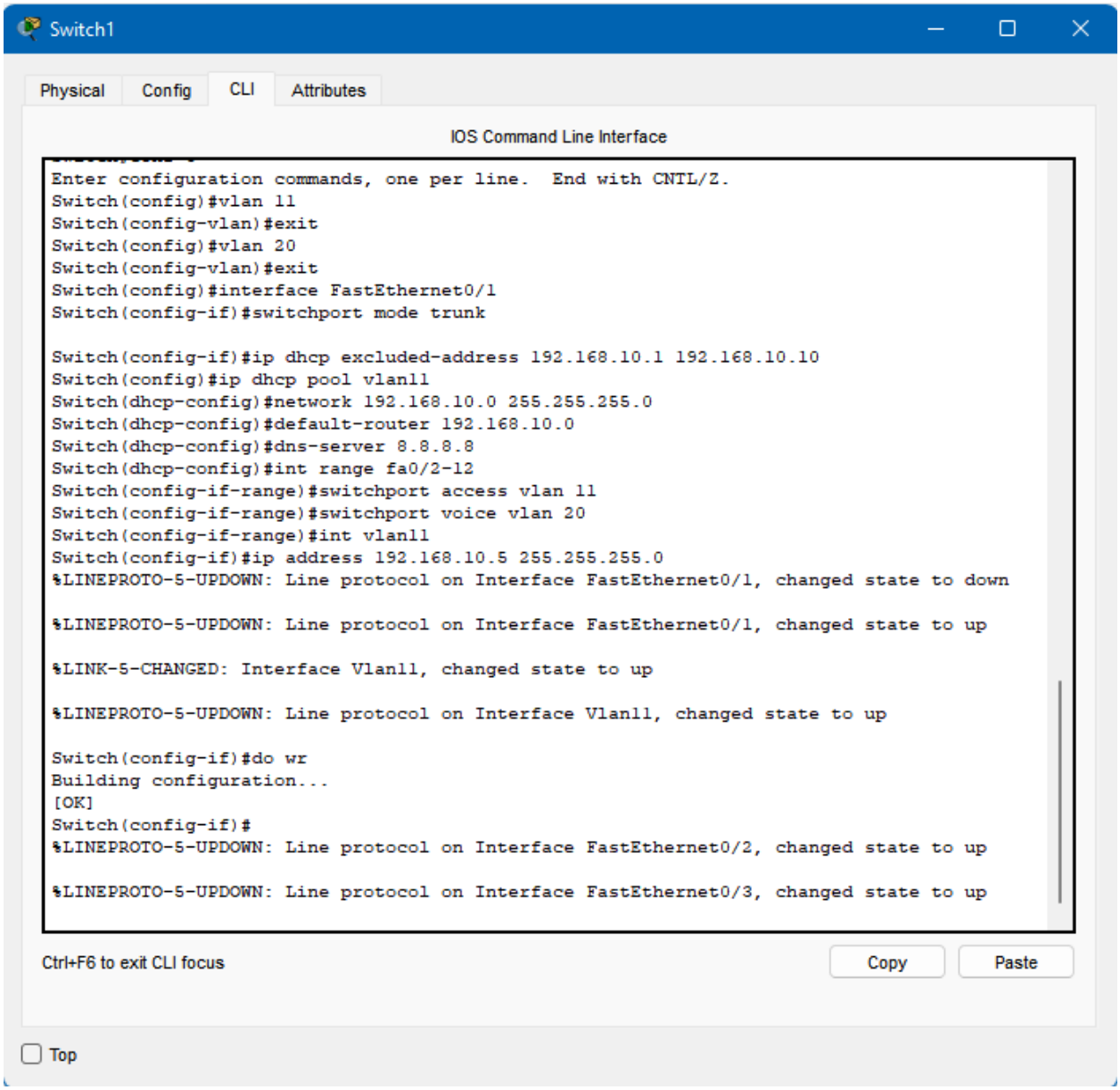


Рисунок 3.6 – Вікно налаштувань комутатора

Проводимо аналогічні дії для іншого комутатора та отримуємо працюючу мережу. Переводимо персональні комп`ютери в DHCP режим, та підключаємо IP-телефони до мережі. Після виконаних дій спостерігаємо, що мережа працює коректно і потрібно провести певні тести, щоб впевнитись в правильності налаштувань. Розпочнемо з перевірки базової роботи мережі, спробуємо відправити команду ping з комп`ютера однієї мережі в іншу:

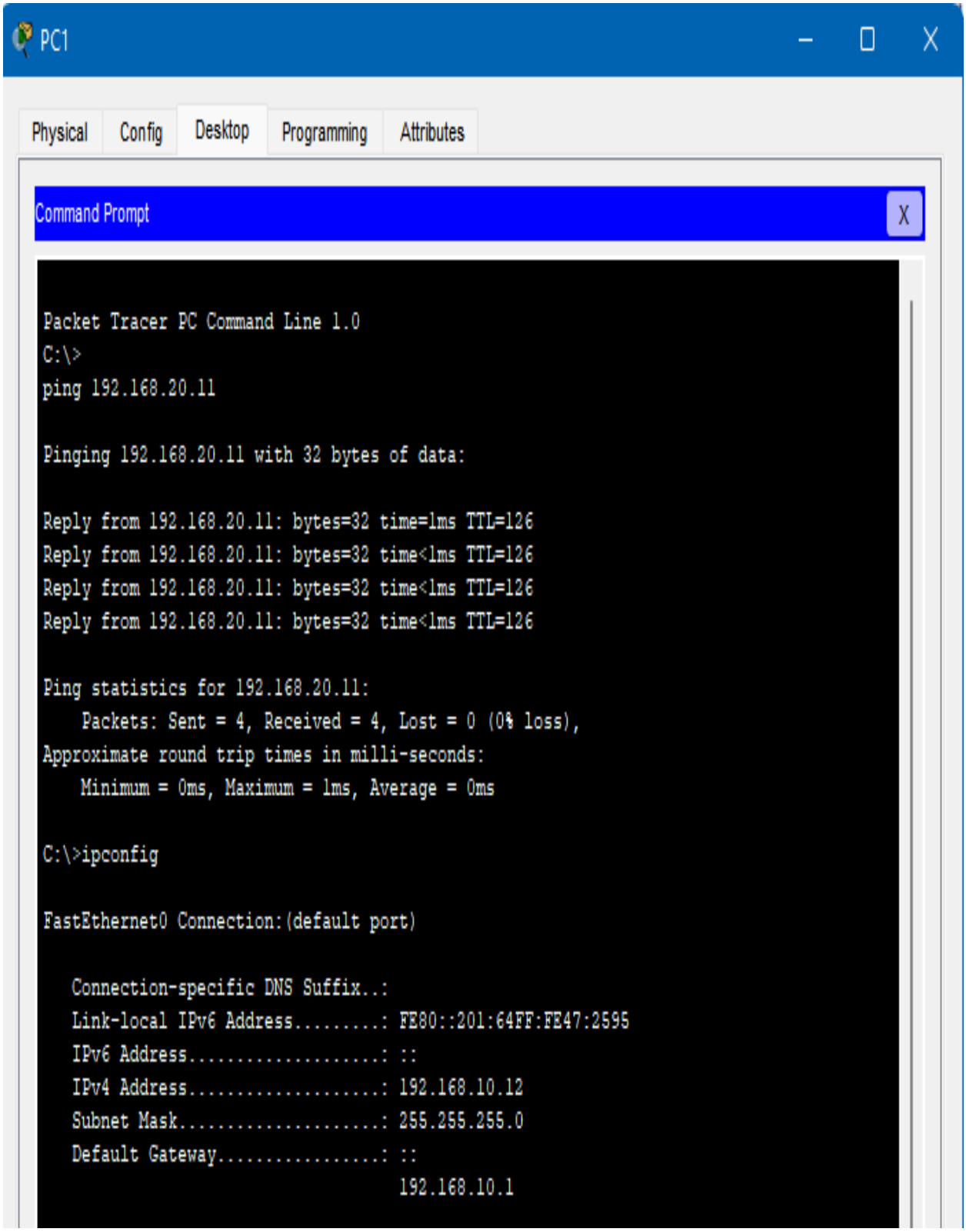


Рисунок 3.7 – Перевірка мережі командою “ping”

Як можемо бачити команда відпрацювала успішно.

Також можемо бачити, що комп`ютер успішно отримав такі параметри як: IPv4 адресу, мережеву маску та напрям по замовчуванню, отже можемо

бачити, що DHCP сервіс працює. Також можемо спостерігати, що VLAN, який був вказаний, був успішно створений та працює:

```
Device Name: Switch1
Device Model: 2950-24
Hostname: Switch

Port              Link   VLAN   IP Address         MAC Address
FastEthernet0/1   Up     --     --                 0030.F283.9B01
FastEthernet0/2   Up     11     --                 0030.F283.9B02
FastEthernet0/3   Up     11     --                 0030.F283.9B03
FastEthernet0/4   Down   11     --                 0030.F283.9B04
FastEthernet0/5   Down   11     --                 0030.F283.9B05
FastEthernet0/6   Down   11     --                 0030.F283.9B06
FastEthernet0/7   Down   11     --                 0030.F283.9B07
FastEthernet0/8   Down   11     --                 0030.F283.9B08
FastEthernet0/9   Down   11     --                 0030.F283.9B09
FastEthernet0/10  Down   11     --                 0030.F283.9B0A
FastEthernet0/11  Down   11     --                 0030.F283.9B0B
FastEthernet0/12  Down   11     --                 0030.F283.9B0C
FastEthernet0/13  Down   1      --                 0030.F283.9B0D
FastEthernet0/14  Down   1      --                 0030.F283.9B0E
FastEthernet0/15  Down   1      --                 0030.F283.9B0F
FastEthernet0/16  Down   1      --                 0030.F283.9B10
FastEthernet0/17  Down   1      --                 0030.F283.9B11
FastEthernet0/18  Down   1      --                 0030.F283.9B12
FastEthernet0/19  Down   1      --                 0030.F283.9B13
FastEthernet0/20  Down   1      --                 0030.F283.9B14
FastEthernet0/21  Down   1      --                 0030.F283.9B15
FastEthernet0/22  Down   1      --                 0030.F283.9B16
FastEthernet0/23  Down   1      --                 0030.F283.9B17
FastEthernet0/24  Down   1      --                 0030.F283.9B18
Vlan1             Down   1      <not set>          0050.0F1E.E100
Vlan11            Up     11     192.168.10.5/24    0050.0F1E.E101

Physical Location: Intercity > Home City > Corporate Office > Main Wi
```

Рисунок 3.8 – Вікно інформації про VLAN

Перейдемо до перевірки VoIP сервісу, спробуємо зателефонувати з телефону однієї мережі на телефон, який знаходиться в іншій:

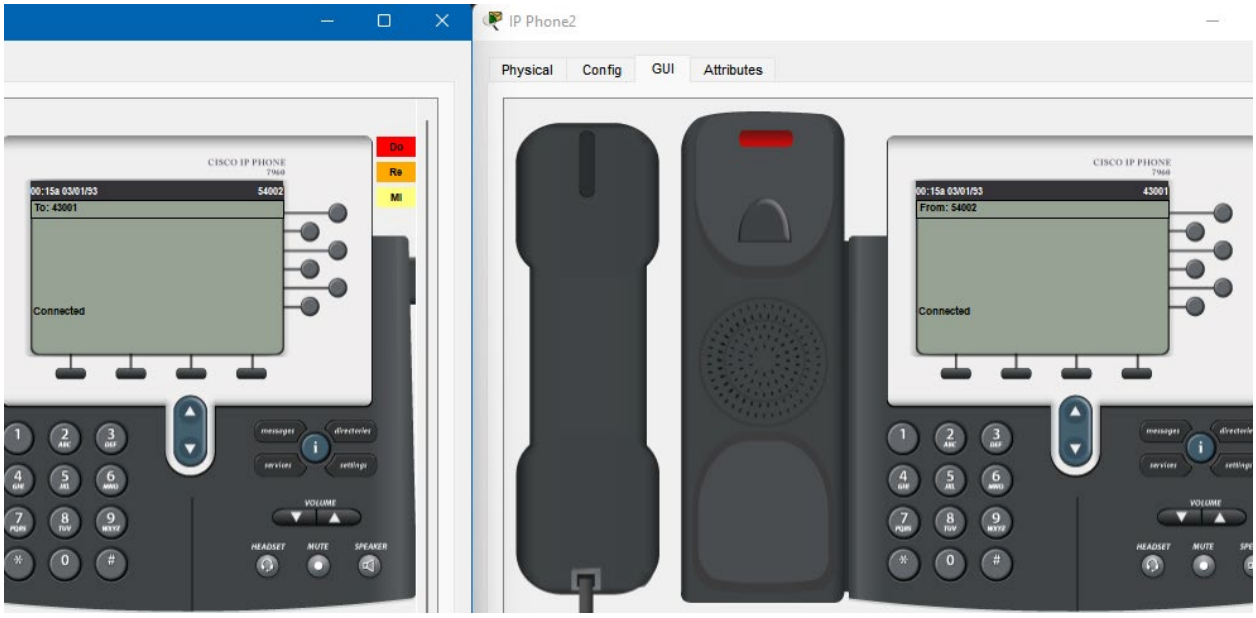


Рисунок 3.9 – Перевірка VoIP телефонії

З отриманих результатів бачимо, що система працює як і було заплановано. За допомогою симулятора Cisco Packet Tracer та обладнання Cisco було створено мультисервісну мережу, яка задовольняє базові потреби корпоративного сегменту та працює на обладнанні Cisco. За допомогою створеної веб-системи було налаштовано такі сервіси як: DHCP, VLAN, VoIP і протокол маршрутизації RIP.

На дані спостереження витрачено всього 10 хвилин, що значно швидше ніж при ручній конфігурації кожного пристрою в мережі.

Провівши тестування програми в умовах наближених до справжніх, провівши аналіз ми прийшли до ствердження, що система значно прискорює налаштування базових сервісів мережі.

ВИСНОВКИ

Під час кваліфікаційної роботи було розглянуто наступні мережеві сервіси: DHCP, DNS, FTP, Email, VLAN, VoIP і протокол маршрутизації RIP та як вони працюють в реальних умовах мережі, а також як вони взаємодіють між собою. Працювавши над цією системою, нами був отриманий досвід роботи з такими симуляторами як Cisco Packet Tracer, GNS3. Сьогодні такі симулятори дозволяють отримати неоціненний досвід, як працювати з недешевим обладнанням безкоштовно та в умовах наближених до реальних, з комутаторами, маршрутизаторами, IP-телефонами. Такі симулятори практичні, але позбавлені від певних автоматизацій рутинних процесів, які б дозволили пришвидшити дослідження мереж та обладнання Cisco, і створення мультисервісних мереж.

На основі проведених досліджень був розроблений графічний інтерфейс для налаштування корпоративних мультисервісних мереж. За допомогою візуального інтерфейсу, користувач отримує можливість отримати конфігурацію мережевих пристроїв, яка дозволяє налаштувати різноманітні сервіси та взаємодії одних з іншими в мережі на обладнанні Cisco, набір отриманих команд в графічному інтерфейсі повністю відповідає документації Cisco та дозволяє застосувати їх до справжнього обладнання Cisco. Для роботи з програмою достатньо її запустити і натиснути кнопку отримання конфігурації, але в той же час, користувач отримує можливість змінити вхідні дані мереж, які будуть використовуватися, тим підстроїти конфігурацію під мережі, які потрібні користувачу.

Отримана веб-система допомагає користувачам з різним рівнем знань однаково використовувати її, для налаштування мультисервісних мереж, де знання базових понять, як працювати з обладнанням Cisco та як працюють мультисервісні мережі, не обов`язкове для використання, тим самим прискорює налаштування мереж та перехід до безпосереднього використання мережі для реальних задач.

СПИСОК ЛІТЕРАТУРИ

[1] X. Sun, Y.-W. E. Sung, S. D. Krothapalli, and S. G. Rao, “A Systematic Approach for Evolving VLAN Designs”.

[2] “What is a VLAN (Virtual LAN)?” https://www.techtarget.com/searchnetworking/definition/virtual-LAN (accessed Oct. 28, 2021).

[3] M. Yaibuates and R. Chaisricharoen, “Implementing of IP address Recovery for DHCP Service,” *International Journal of Applied Engineering Research*, vol. 13, no. 5, pp. 2659–2662, 2018, Accessed: Oct. 28, 2021. [Online]. Available: http://www.ripublication.com

[4] “DNS Privacy Frequently Asked Questions (FAQ)”.

[5] W. B. de Vries, Q. Scheitle, M. Muller, W. Toorop, R. Dolmans, and R. van Rijswijk-Deij, “A First Look at QNAME Minimization in the Domain Name System”, Accessed: Oct. 28, 2021. [Online]. Available: www.niclabs.cl?

[6] “Research on the Design and Implementation of FTP Client Based on Java”, doi: 10.1088/1742-6596/1992/2/022027.

[7] S. Aryza Lubis *et al.*, “Prototype file transfer protocol application for LAN and Wi-Fi communication,” *International Journal of Engineering & Technology*, vol. 7, no. 2, pp. 345–347, 2018, Accessed: Oct. 28, 2021. [Online]. Available: https://www.researchgate.net/publication/326990164

[8] A. O. Hasan, R. Yousif, R. S. Rashid, A. OHasan, A. ZYousif, and R. SRashid, “A Simulation of Wireless Computer Network Salahaddin University Campus of FTP Using OPNET Simulator,” 2017, Accessed: Dec. 09, 2021. [Online]. Available: https://www.researchgate.net/publication/322077251

[9] A. Alghoul, S. al Ajrami, G. al Jarousha, G. Harb, and S. S. Abu-Naser, “Email Classification Using Artificial Neural Network,” *International Journal of Academic Engineering Research*, vol. 2, pp. 8–14, 2018, Accessed: Oct. 28, 2021. [Online]. Available: www.ijeais.org/ijaer

[10] S. Ali and B. Sc, “Performance Evaluation of IMAP and POP3 Protocols Using Optimized Network Engineering Tool (OPNET) MASTER OF ENGINEERING in the Department of Electrical and Computer Engineering”.

[11] A. Ajani *et al.*, “Comparative performance evaluation of open shortest path first, OSPF and routing information protocol, RIP in network link failure and recovery cases UK View project Comparative Performance Evaluation Of Open Shortest Path First, OSPF And Routing Information Protocol, RIP In Network Link Failure and recovery cases,” 2017, doi: 10.1109/NIGERCON.2017.8281901.

[12] M. Ma Gyi, S. San Naing, and P. Ei San, “Performance of Best Route Selection using RIP and OSPF Routing Protocols the Creative Commons Attribution License (CC BY 4.0),” *International Journal of Trend in Scientific Research and Development (IJTSRD) International Journal of Trend in Scientific Research and Development*, no. 5, pp. 1979–1982, 2019, doi: 10.31142/ijtsrd7873.

[13] K. G. Chaudhari, “VOICE OVER INTERNET PROTOCOL (VOIP) IN WIRELESS COMMUNICATION NETWORK-OVERVIEW”, Accessed: Dec. 09, 2021. [Online]. Available: https://ssrn.com/abstract=3729032

[14] J. Packer and W. Reuschel, “VoIP Accessibility: A Usability Study of Voice over Internet Protocol (VoIP) Systems and a Survey of VoIP Users with Vision Loss”.

[15] “IP Addressing: DHCP Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) - Configuring the Cisco IOS DHCP Server [Cisco IOS XE 3SE] - Cisco.” https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipaddr_dhcp/configuration/xe-3se/3850/dhcp-xe-3se-3850-book/config-dhcp-server.html (accessed Dec. 12, 2021).

ДОДАТКИ

Додаток А

```
import eel
import re
eel.init("web")

@eel.expose
def calculate_conf(ip1,ip2,voip1,voip2):
      temp = ip1.replace(" ","").split('.')
      temp2 = ip2.replace(" ","").split('.')
      router1_ip = re.sub('\.\d+$','.1',ip1.replace(" ",""))
      router2_ip = re.sub('\.\d+$','.1',ip2.replace(" ",""))
      voip_r1 = re.sub('\.\d+$','.254',voip1.replace(" ",""))
      voip_r2 = re.sub('\.\d+$','.254',voip2.replace(" ",""))
      vlan1_ip = re.sub('\.\d+$','.5',ip1.replace(" ",""))
      vlan2_ip = re.sub('\.\d+$','.5',ip2.replace(" ",""))
      dhcp_vlan1= re.sub('\.\d+$','.10',ip1.replace(" ",""))
      dhcp_vlan2= re.sub('\.\d+$','.10',ip2.replace(" ",""))
      return  router1_ip  ,router2_ip  ,voip_r1  ,voip_r2,  vlan1_ip,  vlan2_ip,
dhcp_vlan1, dhcp_vlan2
eel.start("main.html", size=(1920,1080))
```

```
<!DOCTYPE html>
<html>
<head>
        <meta charset="UTF-8">
        <title>Web-app</title>
        <script src="eel.js"></script>
        <link rel="icon" type="image/png" href="/favicon.png">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>S
        <link rel="stylesheet" href="main.css">
        <link href="https://fonts.googleapis.com/css2?family=Roboto:wght@100&display=swap"
rel="stylesheet">
        </head>
<body>
        <div style="color: black; ">
        <p style="display: block; border: none; background: lightblue;border-radius: 10px;outline: none;width:
50%;font-size: 30px;"> Input IP/SubnetMask/VoIP for Network 1</p>
        </div>
        <input id="IP1" type="text" placeholder="Enter IP address for network 1" required=""
value="192.168.10.0">
        <input id="mask1" type="text" placeholder="Enter subnet mask for network 1" required=""
value="255.255.255.0">
        <input id="voip_ip1" type="text" placeholder="Enter VoIP IP for network 1" required=""
value="192.168.100.0">
        <br>
        <div style="color: black">
        <p style="display: block; border: none; background: yellow;border-radius: 10px;outline: none;width:
50%;font-size: 30px;"> Input IP/SubnetMask/VoIP for Network 2</p>
        </div>
        <input id="IP2" type="text" placeholder="Enter IP address of network 2" required=""
value="192.168.20.0">
        <input id="mask2" type="text" placeholder="Enter subnet mask for network 2" required=""
value="255.255.255.0">
        <input id="voip_ip2" type="text" placeholder="Enter VoIP IP for network 1" required=""
value="192.168.90.0">
```

```
<div style="margin-top: 20px">
<img src="ip.png">
</div>
<button id="calc">Get configuration</button>
<div class="info"><w id="dem1"></div>
<div class="column">
<div id="Router0" class="tabl">
<h1> Router 0 </h1> <br>
enable<br />
conf t <br />
interface FastEthernet0/0 <br>
no ip address<br>
duplex auto<br>
speed auto<br>
no sh<br>
<br>
interface FastEthernet0/0.11<br>
encapsulation dot1Q 11<br>
ip address <string id="rtip1"></string> <string id="sub1"></string><br>
no sh<br>
<br>
interface FastEthernet0/0.20<br>
encapsulation dot1Q 20<br>
ip address <string id="vlan_voip1"></string> <string id="sub_1"></string><br>
interface FastEthernet0/1 <br>
ip address 20.20.20.1 255.0.0.0 <br>
duplex auto <br>
speed auto <br>
no sh <br>
<br>
router rip<br>
version 2<br>
network 20.0.0.0<br>
network <string id="rip_ip1"></string><br>
network <string id="rip_voip_ip1"></string><br>
<br>
telephony-service <br>
max-ephones 5 <br>
max-dn 5 <br>
ip source-address <string id="voip  1"></string> port 2000 <br>
auto assign 4 to 6 <br>
auto assign 1 to 5 <br>
ephone-dn 1 <br>
 number 54001 <br>
ephone-dn 2 <br>
 number 54002 <br>
ephone-dn 3 <br>
 number 54003 <br>
 <br>
ip dhcp pool voip1 <br>
network <string id="dhcp_voip1"></string> 255.255.255.0 <br>
default-router <string id="src  voip1"></string> <br>
option 150 ip <string id="opt_voip1"></string> <br>
dial-peer voice 1 voip <br>
destination-pattern 4300. <br>
session target ipv4:<string id="target_voip2"></string> <br>
do wr<br>
end <br>
</div>
</div>
<div class="column">
```

```
<div id="Router1" class="tabl1">
<h1> Router 1 </h1> <br>
enable<br />
conf t <br />
interface FastEthernet0/0 <br>
no ip address<br>
duplex auto<br>
speed auto<br>
no sh<br>
<br>
interface FastEthernet0/0.11<br>
encapsulation dot1Q 11<br>
ip address <string id="rtip2"></string> <string id="sub2"></string><br>
no sh<br>
<br>
interface FastEthernet0/0.20<br>
encapsulation dot1Q 20<br>
ip address <string id="vlan  voip2"></string> <string id="sub  2"></string><br>
interface FastEthernet0/1 <br>
ip address 20.20.20.2 255.0.0.0 <br>
duplex auto <br>
speed auto <br>
no sh <br>
<br>
router rip<br>
version 2<br>
network 20.0.0.0<br>
network <string id="rip_ip2"></string><br>
network <string id="rip  voip  ip2"></string><br>
<br>
telephony-service <br>
max-ephones 5 <br>
max-dn 5 <br>
ip source-address <string id="voip_2"></string> port 2000 <br>
auto assign 4 to 6 <br>
auto assign 1 to 5 <br>
ephone-dn 1 <br>
 number 43001 <br>
ephone-dn 2 <br>
 number 43002 <br>
ephone-dn 3 <br>
 number 43003 <br>
 <br>
ip dhcp pool voip2 <br>
network <string id="dhcp_voip2"></string> 255.255.255.0 <br>
default-router <string id="src_voip2"></string> <br>
option 150 ip <string id="opt_voip2"></string> <br>
dial-peer voice 2 voip <br>
destination-pattern 5400. <br>
session target ipv4:<string id="target_voip1"></string> <br>
do wr<br>
end <br>
</div>
</div>
<div class="column">
<div id="Switch1" class="tabl2">
<h1>Switch 0</h1>
en <br>
conf t<br>
vlan 11 <br>
exit<br>
```

```
vlan 20<br>
exit<br>
interface FastEthernet0/1 <br>
switchport mode trunk <br>
ip dhcp excluded-address <string id="switch_ip1"></string> <string id="end_switch_ip1"></string> <br>
ip dhcp pool vlan11 <br>
network <string id="dhcp_switch_ip1"></string> <string id="dhcp_switch_mask1"></string> <br>
default-router <string id="dhcp_def_switch_ip1"></string> <br>
dns-server 8.8.8.8 <br>
int range fa0/2-12 <br>
switchport access vlan 11 <br>
switchport voice vlan 20 <br>
int vlan11 <br>
ip address <string id="vlan_switch_ip1"></string> <string id="vlan_switch_mask1"></string><br>
do wr<br>
</div>
</div>
<div class="column">
<div id="Switch2" class="tabl3">
<h1>Switch 1</h1>
en <br>
conf t<br>
vlan 11 <br>
exit<br>
vlan 20<br>
exit<br>
interface FastEthernet0/1 <br>
switchport mode trunk <br>
ip dhcp excluded-address <string id="switch_ip2"></string> <string id="end_switch_ip2"></string> <br>
ip dhcp pool vlan11 <br>
network <string id="dhcp_switch_ip2"></string> <string id="dhcp_switch_mask2"></string> <br>
default-router <string id="dhcp_def_switch_ip2"></string> <br>
dns-server 8.8.8.8 <br>
int range fa0/2-12 <br>
switchport access vlan 11 <br>
switchport voice vlan 20 <br>
int vlan11 <br>
ip address <string id="vlan_switch_ip2"></string> <string id="vlan_switch_mask2"></string><br>
do wr<br>
</div>
</div>
<script type="text/javascript">
                async function calc_conf() {
                let ip1 = document.getElementById('IP1').value;
                let mask1 = document.getElementById('mask1').value;
                let ip2 = document.getElementById('IP2').value;
                let mask2 = document.getElementById('mask2').value;
                let voip_ip1 = document.getElementById('voip_ip1').value;
                let voip_ip2 = document.getElementById('voip_ip2').value;
                let rip_voip_ip1=voip_ip1;
                let res = null;
                res = await eel.calculate_conf(ip1,ip2,voip_ip1,voip_ip2)();
                document.getElementById('switch_ip1').innerHTML=res[0];
                document.getElementById('dhcp_switch_ip1').innerHTML=ip1;
                document.getElementById('dhcp_def_switch_ip1').innerHTML=res[0];
                document.getElementById('dhcp_switch_mask1').innerHTML=mask1;
                document.getElementById('vlan_switch_ip1').innerHTML=res[4];
                document.getElementById('vlan_switch_mask1').innerHTML=mask1;
                document.getElementById('end_switch_ip1').innerHTML=res[6];
                document.getElementById('switch_ip2').innerHTML=res[1];
                document.getElementById('dhcp_switch_ip2').innerHTML=ip2;
```

```
                document.getElementById('dhcp_def_switch_ip2').innerHTML=res[1];
                document.getElementById('dhcp_switch_mask2').innerHTML=mask2;
                document.getElementById('vlan_switch_ip2').innerHTML=res[5];
                document.getElementById('vlan_switch_mask2').innerHTML=mask2;
                document.getElementById('end_switch_ip2').innerHTML=res[7];
                document.getElementById('rtip1').innerHTML=res[0];
                document.getElementById('sub1').innerHTML=mask1;
                document.getElementById('sub_1').innerHTML=mask1;
                document.getElementById('vlan_voip1').innerHTML=res[2];
                document.getElementById('rip_ip1').innerHTML=ip1;
                document.getElementById('rip_voip_ip1').innerHTML=voip_ip1;
                document.getElementById('voip_1').innerHTML=res[2];
                document.getElementById('dhcp_voip1').innerHTML=voip_ip1;
                document.getElementById('src_voip1').innerHTML=res[2];
                document.getElementById('opt_voip1').innerHTML=res[2];
                document.getElementById('target_voip1').innerHTML=res[2];
                document.getElementById('rtip2').innerHTML=res[1];
                document.getElementById('sub2').innerHTML=mask2;
                document.getElementById('sub_2').innerHTML=mask2;
                document.getElementById('vlan_voip2').innerHTML=res[3];
                document.getElementById('rip_ip2').innerHTML=ip2;
                document.getElementById('rip_voip_ip2').innerHTML=voip_ip2;
                document.getElementById('voip_2').innerHTML=res[3];
                document.getElementById('dhcp_voip2').innerHTML=voip_ip2;
                document.getElementById('src_voip2').innerHTML=res[3];
                document.getElementById('opt_voip2').innerHTML=res[3];
                document.getElementById('target_voip2').innerHTML=res[3];
        }
        jQuery('#calc').on('click', function() {
                calc_conf();
                document.getElementById('Router0').style.display='inline-block';
                document.getElementById('Router1').style.display='inline-block';
                document.getElementById('Switch1').style.display='inline-block';
                document.getElementById('Switch2').style.display='inline-block';
        });
                </script>
    </body>
</html>
```

Додаток В

```
body {
        background: white ;
        color: black;
}
#calc {
        display: block;
        border: none;
        margin-top: 15px;
        background: #16BFFD;
        border-radius: 10px;
        padding: 20px;
        color: white;
        outline: none;
        width:  50%;
        font-size: 30px;
        cursor: pointer;
}
#calc:hover{
        opacity: .8;
}
#Router0{
        display: none;
```

```
    background: #000 none repeat scroll 0 0;

    border-radius: 10px;
    color: green;

    font-family: monospace;
    font-size: 14px;
    margin-bottom: 20px;
    margin-top: 20px;
    padding: 18px;
    text-align: left;

}
#Router1{
        display: none;
                background: #000 none repeat scroll 0 0;
    border-radius: 10px;
    color: green;
    font-family: monospace;
    font-size: 14px;
    margin-bottom: 20px;
    margin-top: 20px;
    padding: 18px;
    text-align: left;
}
        .column {
                width: 480px;
                float: left;
                                        }
#Switch1{
        display: none;
                background: #000 none repeat scroll 0 0;
    border-radius: 10px;
    color: green;
    font-family: monospace;
    font-size: 14px;
    margin-bottom: 20px;
    margin-top: 20px;
    padding: 18px;
    text-align: left;
}
 #Switch2{
        display: none;
                background: #000 none repeat scroll 0 0;
    border-radius: 10px;
    color: green;
    font-family: monospace;
    font-size: 14px;
    margin-bottom: 20px;
    margin-top: 20px;
    padding: 18px;
    text-align: left;
}
```