

**МІНІСТЕРСТВО ОСВІТИ НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ**

**МЕТОДИЧНІ ВКАЗІВКИ ДО ЛАБОРАТОРНИХ РОБІТ
З КУРСУ
«МУЛЬТИСЕРВІСНІ ТЕХНОЛОГІЇ В КОМП'ЮТЕРНИХ МЕРЕЖАХ»**

Тернопіль
2017

УДК 681.3
М54

Укладачі:

Марценко С.В., кан. техн. наук, доцент,
Поливана У.В., асистент.

Методичні вказівки розглянуто й затверджено на засіданні
кафедри комп'ютерних наук
Тернопільського національного технічного університету імені Івана Пулюя
протокол №8 від 31 січня 2017 р.

Схвалено та рекомендовано до друку методичною комісією
Факультету комп'ютерно-інформаційних систем і програмної інженерії
Тернопільського національного технічного університету імені Івана Пулюя
протокол №6 від 23 лютого 2017 р.

Методичні вказівки до лабораторних робіт з курсу «Мультисервісні технології
в комп'ютерних мережах» / Укладачі : Марценко С.В., Поливана У.В. –
Тернопіль : Тернопільський національний технічний університет імені Івана
Пулюя, 2017 – 20 с.

Відповідальний за випуск: *У.В. Поливана, ст.. викладач.*

© Марценко С.В., Поливана У.В., 2017

© Тернопільський національний технічний
університет імені Івана Пулюя, 2017

ЛАБОРАТОРНА РОБОТА №1

НАЛАШТУВАННЯ СЕРВІСУ IP-ТЕЛЕФОНІЇ ІЗ ВИКОРИСТАННЯМ *ASTERISK*.

Мета роботи: ознайомлення із сервісом IP-телефонії (*Voice over IP*) та налаштування програмного забезпечення *ASTERISK* для її реалізації.

1.1 Теоретичні відомості

IP-телефонія (*Voice over IP – VoIP*) являє собою методику і групу технологій для доставки голосових повідомлень і мультимедійних сесій по інтернет-протоколу (*IP*) мереж, таких як Інтернет.

Термін IP-телефонії або інтернет-телефонії конкретно вказує на надання послуг зв'язку (голос, факс, *SMS*, голосових повідомлень) у публічній мережі Інтернет, а не через телефонну мережу загального користування. Принципи *VoIP* схожі на звичайні телефонні дзвінки із залученням “сигналізації”, налаштування каналів, оцифровки аналогових сигналів і кодування. Замість того щоб передавати дані по мережі з комутацією каналів, відбувається передача по звичній мережі із комутацією пакетів.

Перші провайдери *VoIP*-послуг пропонували бізнес-моделі і технічні рішення, які відображали архітектуру звичної телефонної мережі. Друге покоління постачальників, таких, як *Skype*, побудували закриті мережі приватних баз користувачів, пропонуючи можливість отримувати безкоштовні дзвінки і зручності. Третє покоління провайдерів, таких як *Google Talk*, взяли поняття федеративного *VoIP*, що є відступом від архітектури традиційних мереж. Ці рішення зазвичай дозволяють динамічні взаємозв'язки між користувачами на будь-яких двох доменах в Інтернеті.

VoIP системи використовують контроль сесії і протоколи сигналізації для контролю сигналізації, встановлення викликів. Вони здійснюють передачу аудіопотоків по *IP*-мережам за допомогою спеціальних медіа протоколів, які кодують голос, аудіо, відео. Різні кодеки оптимізують потік даних до середовища передачі та пропускної здатності мережі.

VoIP є на багатьох смартфонах, персональних комп'ютерах та пристроях доступу в Інтернет. Дзвінки та СМС повідомлення можуть бути відправлені через *3G* або *WiFi*.

Asterisk – безкоштовне програмне забезпечення *IP*-телефонії з відкритим вихідним кодом від компанії *Digium*, спочатку розроблене Марком Спенсером. Програма працює на операційних системах *Linux*, *FreeBSD*, *OpenBSD* і *Solaris* та ін. Ім'я проекту походить від назви символу “*” (англ. *asterisk* – “зірочка”).

Asterisk в комплексі з необхідним обладнанням володіє всіма можливостями класичної АТС, підтримує безліч *VoIP*-протоколів і надає багаті функції управління дзвінками, серед них:

- Голосова пошта
- Конференц-зв'язок
- *IVR* (інтерактивне голосове меню)
- Центр обробки дзвінків (переадресація дзвінків у чергу і розподіл їх за абонентам, використовуючи різні алгоритми)
- *Call Detail Record* (детальна запис про виклик)

Для створення додаткової функціональності можна скористатися власною мовою *Asterisk* для написання плану нумерації, написавши модуль на мові *C*, або скориставшись *AGI* – гнучким і універсальним інтерфейсом для інтеграції з зовнішніми системами обробки даних. Модулі, що виконуються через *AGI*, можуть бути написані на будь-якій мові програмування.

Asterisk поширюється на умовах подвійної ліцензії, завдяки якій одночасно з основним кодом, поширюваним по відкритій ліцензії *GNU GPL*, можливе створення закритих модулів, що містять ліцензований код: наприклад, модуль для підтримки кодека *G.729*.

Завдяки вільній ліцензії *Asterisk* активно розвивається і підтримується тисячами людей по всій планеті.

Софтфон (калька з англ. *Software telephone*, програмний телефон) – клас програмного забезпечення для персонального комп'ютера для здійснення телефонних (голосових) або відеодзвінків через Інтернет (у загальному випадку через будь-яку *IP*-мережу) без використання додаткового апаратного

забезпечення за винятком гарнітури, *USB*-телефону, мікрофона і аудіоколонок, вебкамери (у разі відеозв'язку).

Софтфони можуть працювати як за власними спеціалізованим закритим протоколом (*Skype, Mail.ru Агент, ICQ, QIP, Google Talk*), так і по відкритих протоколах *SIP* та *H.323*, що дозволяє їх використовувати в якості абонентського обладнання в системах *IP*-телефонії.

Багато сучасних софтфонів дають користувачам можливість спілкуватися за допомогою миттєвих повідомлень, а також підтримують інші функції, притаманні уніфікованим комунікаціям: голосові і відеоконференції, індикація присутності, відправка *SMS*, тощо.

1.2 Порядок виконання

1. Встановити та налаштувати, використовуючи довільний комп'ютер чи віртуальну машину, сервер на базі однієї із запропонованих операційних систем: *Debian, Fedora, Gentoo, Redhat, SuSE, FreeBSD, NetBSD, OpenBSD, Mac OS X*.

2. Встановити *Asterisk* (літературні джерела з описом процедури встановлення та налаштування наводиться в переліку джерел в кінці тексту лабораторної роботи).

3. Налаштувати *Asterisk* для можливості спілкування двом абонентам через *SIP* протокол (для *SIP*-клієнтів).

4. На двох інших комп'ютерах (можливе використання віртуальної машини), з'єднаних між собою та сервером мережею, встановити софтфон, наприклад, *X-Lite*.

5. Налаштувати встановлені софтфони для роботи із *Asterisk*.

6. Здійснити дзвінок із одного софтфона на інший.

1.3 Порядок захисту

Студент, який виконав завдання лабораторної роботи, подає письмовий звіт про виконання лабораторної роботи і доповідає про результати та дає відповіді на запитання. Виконане завдання оцінюється згідно системи

контролю знань, умінь, навичок, наведеної на відповідній сторінці дистанційного курсу.

1.4 Контрольні запитання

1. Що таке *IP*-телефонія?
2. Для чого призначений *Asterisk*?
3. Що таке *Call Manager*?
4. Що таке *VoIP Gateway*?
5. Що таке софтфон?

ЛАБОРАТОРНА РОБОТА №2

ПРОЕКТУВАННЯ МЕРЕЖІ У СИМУЛЯТОРІ *CISCO PACKETTRACER*

Мета роботи: знайомство із програмним симулятором мережі *Cisco PacketTracer* та його функціональними можливостями.

2.1 Теоретичні відомості

Cisco PacketTracer – це симулятор мережі передачі даних, що випускається фірмою *Cisco Systems*. Дозволяє робити працездатні моделі мережі, налаштовувати (командами *Cisco IOS*) маршрутизатори і комутатори, взаємодіяти між декількома користувачами (через хмару) . Включає в себе серії маршрутизаторів *Cisco* 800, 2600, 2800 і комутаторів 2950 , 2960 , 3650. Крім того є сервери *DHCP*, *HTTP*, *TFTP*, *FTP*, робочі станції, різні модулі до комп'ютерів і маршрутизаторів, пристрої *WiFi*, різні кабелі, тощо.

Успішно дозволяє створювати навіть складні макети мереж, перевіряти на працездатність топології.

2.2 Порядок виконання роботи

1. Встановити *Cisco PacketTracer* на своєму комп'ютері.
2. Розмісти мережеві елементи (*router*, *switch*, робочі станції, тощо) відповідно до рисунку 2.1.
3. Налаштувати мережу (ремарка: мережа складається із двох під мереж *Group_1* і *Group_2*): прописати IP-адреса на мережевому обладнанні, прописати маршрутизацію, тощо. IP-адреса вибрати довільно.
4. Перевірити доступність *Group_2_PC_1* із *Group_1_PC_1* та навпаки використовуючи команду *ping*.
5. Налаштувати і запустити довільний генератор трафіку.

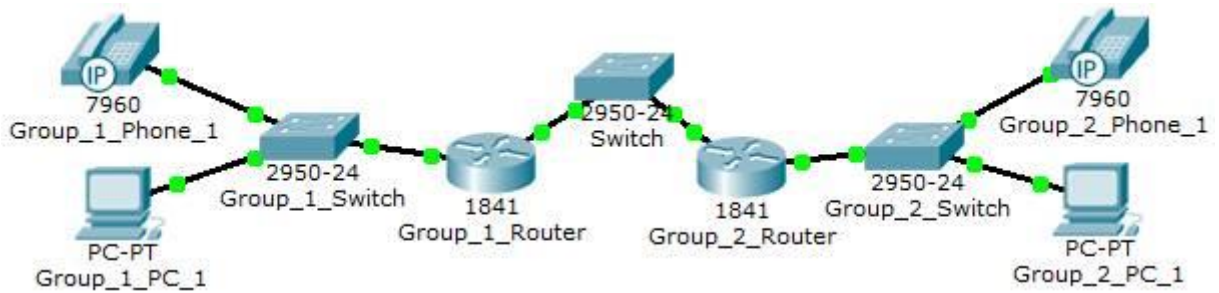


Рисунок 2.1 – Модель мережі

2.3 Порядок захисту

Студент, який виконав завдання лабораторної роботи, подає письмовий звіт про виконання лабораторної роботи і доповідає про результати та дає відповіді на запитання. Виконане завдання оцінюється згідно системи контролю знань, умінь, навичок, наведеної на відповідній сторінці дистанційного курсу.

2.4 Контрольні запитання

1. Для чого призначений *Cisco PacketTracer*.
2. Що таке router.
3. Що таке switch.
4. Що таке маска мережі.
5. Що таке IP-адреса.

ЛАБОРАТОРНА РОБОТА №3

НА НАЛАШТУВАННЯ *QoS* НА МАРШРУТИЗАТОРАХ *CISCO*, ВИКОРИСТОВУЮЧИ *CISCO PACKETTRACER*

Мета роботи: освоїти основи налаштування *QoS* на обладнанні *Cisco*, використовуючи емулятор мережі *Cisco PacketTracer*.

3.1 Теоретичні відомості

QoS (Quality Of Service) – якість обслуговування. Під цим терміном в області комп'ютерних мереж називають імовірність, що мережа зв'язку відповідає заданій угоді про трафік або ж, у ряді випадків, неформальне позначення ймовірності того, що пакет пройде між двома точками мережі.

Для більшості випадків якість зв'язку визначається чотирма параметрами:

- Пропускна смуга (*Bandwidth*), описує номінальну пропускну здатність середовища передачі інформації, визначає ширину каналу. Вимірюється в *bit/s (bps)*, *kbit/s (kbps)*, *Mbit/s (Mbps)*.
- Затримка при передачі пакета (*Delay*), вимірюється в мілісекундах.
- Коливання (тремтіння) затримки при передачі пакетів – джиттер (*Jitter*).
- Втрата пакетів (*Packet Loss*). Визначає кількість пакетів, загублених у мережі під час передачі.

Для простоти розуміння канал зв'язку можна представити у вигляді умовної труби, а пропускну здатність описати як функцію двох параметрів: ширини труби і її довжин.

Коли передача даних зіштовхується із проблемою “вузького місця” для прийому й відправлення пакетів на роутерах звичайно використовується метод *FIFO*: перший прийшов – перший пішов (*First In – First Out*). При інтенсивному трафіку це створює затори, які вирішуються вкрай простим чином: всі пакети що не ввійшли до черги *FIFO* (на вхід або на вихід) ігноруються роутером, і відповідно губляться безповоротно. Розумніший метод – використати

“розумну” чергу, у якій пріоритет у пакетів залежить від типу сервісу – *ToS* (*Type of Service*). Необхідна умова: пакети повинні вже нести мітку типу сервісу для створення “розумної” черги. Звичайні користувачі найчастіше зіштовхуються з терміном *QoS* у домашніх роутерах з підтримкою *QoS*. Наприклад досить логічно дати високий пріоритет *VoIP* і низький – пакетам *FTP*, *SMTP* і клієнта файлообмінної мережі.

Моделі *QoS*:

·Негарантована доставка – *Best Effort Service*. Наявність марки *ToS BestEffort Service* не є механізмом тонкого регулювання, і є ознакою простого збільшення пропускну здатності без якого-небудь виділення окремих класів трафіка й регулювання.

- ·Інтегрований Сервіс – *Integrated Service*. Згідно з *RFC 1633*, модель інтегрованого обслуговування забезпечує наскрізну (*End-to-End*) якість обслуговування, гарантуючи необхідну пропускну здатність. *IntServ* використовує для своїх цілей протокол сигналізації *RSVP*, що забезпечує виконання вимог до всіх проміжних вузлів. У відношенні *IntServ* часто використовується термін – резервування ресурсів (*Resource reservation*).

- ·Диференційоване обслуговування – *Differentiated Service* (*DiffServ*). Описана в *RFC 2474* і *RFC 2475*. Забезпечує *QoS* на основі розподілу ресурсів у ядрі мережі й певних класифікаторів й обмежень на границі мережі, комбінуючих з метою надання необхідних послуг. У цій моделі вводиться поділ трафіка по класах, для кожного з яких визначається свій рівень *QoS*. *DiffServ* складається з керування формуванням трафіка (класифікація пакетів, маркування, керування інтенсивністю) і керування політикою (розподіл ресурсів, політика відкидання пакетів). *DiffServ* є найбільш підходящим прикладом “розумної” пріоритезації трафіка.

Застосування, що вимагають *QoS*:

- ·потоківі мультимедіа-застосування вимагають гарантованої пропускну здатності каналу;

- ·*VoIP* і відеоконференції вимагають невеликих значень джиттера і затримки;

- низка застосувань, наприклад як віддалена хірургія, вимагають гарантованого рівня надійності.

3.2 Порядок виконання роботи

1. Побудувати мережу в *Cisco PacketTracer* зображено на Рис.2.1 (Лабораторна робота №2).
2. Налаштувати мережу.
3. Перевірити коректність налаштувань.
4. Налаштувати *QoS* на роутерах для *VoIP* трафіка, використовуючи *DiffServ*.
5. Перевірити коректність налаштувань *QoS*.

3.3 Порядок захисту

Студент, який виконав завдання лабораторної роботи, подає письмовий звіт про виконання лабораторної роботи і доповідає про результати та дає відповіді на запитання. Виконане завдання оцінюється згідно системи контролю знань, умінь, навичок, наведеної на відповідній сторінці дистанційного курсу.

3.4 Контрольні запитання

1. Що таке *QoS*?
2. Якими параметрами визначається якість зв'язку?
3. Що таке *DiffServ*?
4. Що таке *IntServ*?

ЛАБОРАТОРНА РОБОТА №4

НАЛАШТУВАННЯ MPLS НА МАРШРУТИЗАТОРАХ CISCO, ВИКОРИСТОВУЮЧИ *CISCO PACKETTRACER*

Мета роботи: освоїти основи налаштування *MPLS* на обладнанні *Cisco*, використовуючи емулятор мережі *Cisco PacketTracer*.

4.1 Теоретичні відомості

MPLS – механізм у високопродуктивній телекомунікаційній мережі, який здійснює передачу даних від одного вузла мережі до іншого за допомогою тегів.

MPLS є масштабованим і незалежним від будь-яких протоколів механізмом передачі даних. У мережі, заснованій на *MPLS*, пакетам даних присвоюються мітки. Рішення про подальшу передачу пакету даних іншому вузлу мережі здійснюється тільки на підставі значення присвоєної мітки без необхідності вивчення самого пакету даних. За рахунок цього можливе створення наскрізного віртуального каналу, незалежного від середовища передачі і використовує будь-який протокол передачі даних.

MPLS дозволяє досить легко створювати віртуальні канали між вузлами мережі.

Технологія дозволяє інкапсулювати різні протоколи передачі даних.

Основною перевагою *MPLS* є незалежність від особливостей технологій канального рівня, таких як *ATM*, *Frame Relay*, *SONET/SDH* або *Ethernet*, і відсутності необхідності підтримки декількох мереж другого рівня, необхідних для передачі різного роду трафіку. По виду комутації *MPLS* відноситься до мереж з комутацією пакетів.

Технологія *MPLS* була розроблена для організації єдиного протоколу передачі даних як для програм з комутацією каналів, так і програм з комутацією пакетів (маються на увазі програми з датаграмною передачею

пакетів). *MPLS* може бути використаний для передачі різного виду трафіку, включаючи *IP*-пакети, комірки *ATM*, фрейми *SONET/SDH* і кадри *Ethernet*.

Для вирішення ідентичних завдань раніше були розроблені такі технології, як *Frame Relay* і *ATM*. Багато інженерів вважали, що технологія *ATM* буде замінена іншими протоколами з меншими накладними витратами на передачу даних, що при цьому забезпечать передачу пакетів даних змінної довжини із встановленням з'єднання між вузлами мережі. Технологія *MPLS* розроблялася з урахуванням сильних і слабких сторін *ATM*. У даний час устаткування з підтримкою *MPLS* замінює на ринку обладнання з підтримкою вищезазначених технологій. Ймовірно, що в майбутньому *MPLS* повністю витіснить дані технології.

Зокрема, *MPLS* обходиться без комутації комірок і набору сигнальних протоколів, характерних для *ATM*. При розробці *MPLS* прийшло розуміння того, що на рівні ядра сучасної мережі немає необхідності в комірках *ATM* малого фіксованого розміру, оскільки сучасні оптичні мережі володіють такою великою швидкістю передачі даних, що навіть пакет даних максимальної довжини в 1500 байт відчуває незначну затримку в чергах буферів комутаційного обладнання (необхідність скорочення таких затримок, наприклад, для забезпечення заданої якості голосового трафіку, вплинула на вибір комірок малого розміру, характерних для *ATM*).

У той же час в *MPLS* спробували зберегти механізми оптимізації та управління трафіком і управління окремо від переданого потоку даних, які зробили технології *Frame Relay* і *ATM* привабливими для впровадження у великих мережах передачі даних.

Незважаючи на те, що перехід на *MPLS* дає переваги управління потоками даних (поліпшення надійності і підвищення продуктивності мережі), існує проблема втрати контролю потоків даних, що проходять через мережу *MPLS*, з боку звичайних *IP*-програм.

Технологія *MPLS* заснована на обробці заголовка *MPLS*, який додається до кожного пакету даних. Заголовок *MPLS* може складатися з однієї або кількох

“міток”. Кілька записів (міток) в заголовку *MPLS* називаються стеком тегів. Кожен запис в стеку тегів складається з наступних чотирьох полів:

0																				1																				2																				3																			
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1																																																
+--+																				Label																																																											
										Label										TC S			TTL							Stack																																																	
+--+																				Entry																																																											

Label – значення мітки займає 20 біт;

TC – поле класу трафіку, необхідне для реалізації механізмів якості обслуговування (*QoS*) і явного повідомлення про перевантаження; займає 3 біти;

S – індикатор “дна” стека (*Bottom of stack*); якщо індикатор встановлений в 1, то це означає, що поточна мітка остання в стеку; займає 1 біт;

TTL – необхідне для запобігання петель *MPLS* комутації; займає 8 біт.

У *MPLS* маршрутизаторі пакет з *MPLS*-міткою комутується на наступний порт після пошуку мітки в таблиці комутації замість пошуку в таблиці маршрутизації. При розробці *MPLS* пошук міток і комутація по мітках виконувалися швидше, ніж пошук в таблиці маршрутизації або *RIB* (*Routing Information Base* – інформаційна база маршрутизації), так як комутація може бути виконана безпосередньо на комутаційній фабриці замість центрального процесора.

Маршрутизатор, розташований на вході або виході *MPLS*-мережі називаються *LER* (*Label Edge Router* – граничний маршрутизатор міток). *LER* на вході в *MPLS*-мережу додає мітку *MPLS* до пакету даних, а *LER* на виході з *MPLS*-мережі видаляє мітку *MPLS* з пакету даних. Маршрутизатор, що виконує маршрутизацію пакетів даних, базуючись лише на значенні мітки, називаються *LSR* (*Label Switching Router* – комутуючий мітки маршрутизатор). У деяких випадках пакети даних, які надійшли на порт *LER*, вже можуть містити позначку, тоді новий *LER* додає другу мітку в пакет даних.

Мітки між *LER* і *LSR* розподіляються за допомогою *LDP* (*Label Distribution Protocol* – протокол розподілу міток). Для того щоб отримати повну

картину *MPLS* мережі *LSR* постійно обмінюються мітками та інформацією про кожного сусідньому вузлі використовуючи стандартну процедуру. Віртуальні канали (тунелі), які називаються *LSP* (*Label Switch Path* – шляхи комутації міток), встановлюються провайдерами для вирішення різних завдань, наприклад, для організації *VPN* або для передачі трафіку через *MPLS*-мережу по вказаному тунелю. Багато в чому *LSP* нічим не відрізняється від *PVC* в мережах *ATM* або *Frame Relay* за винятком того, що *LSP* не залежать від особливостей технологій каналного рівня.

При описі віртуальних приватних мереж, заснованих на технології *MPLS*, розташовані на вході або виході мережі *LER* зазвичай називаються *PE*-маршрутизаторами (*Provider Edge* – маршрутизатори на кордоні мережі провайдера), а вузли, що працюють як транзитні маршрутизатори, називаються *P*-маршрутизатори (*Provider* – маршрутизатори провайдера).

Поле значення мітки в *MPLS* заголовку займає 20 біт, таким чином максимально можливе значення мітки 1048576. Наступні номери міток зарезервовані для різних цілей:

Мітка з номером 0 може бути використана тільки як остання мітка в стеку. Наявність позначки з номером 0 позначає що заголовок *MPLS* повинен бути вилучений і подальша маршрутизація пакету повинна ґрунтуватися на значенні заголовка *IPv4*.

- Мітка з номером 1 має особливу назву “*Router Alert Label*” (мітка оповіщення маршрутизатора). Використання даної мітки аналогічно використанню опції “*Router Alert Option*” при передачі в *IP* пакетах. Мітка з номером 1 не може бути використана як остання мітка в стеку.
- Мітка з номером 2 може бути використана тільки як остання мітка в стеку. Наявність позначки з номером 2 означає, що заголовок *MPLS* повинен бути вилучений і подальша маршрутизація пакету повинна ґрунтуватися на значенні заголовка *IPv6*.
- Мітка з номером 3 має особливу назву “*Implicit NULL Label*” (неявна нульова мітка). Дану мітку може привласнювати і розсилати *LSR*, але мітка з номером 3 в дійсності ніколи не може бути використана в стеку

тегів. Якщо *LSR* зустріне дану мітку в стеку тегів, то замість заміни однієї мітки на іншу *LSR* видалить весь стек тегів. Хоча насправді мітка з номером 3 не може з'явитися в стеку тегів, вона повинна бути вказана в *LDP*.

- Мітки з номерами від 4 до 15 зарезервовані.

Існує два стандартних протоколу управління тунелями в *MPLS*-мережі: *LDP* (протокол розподілу міток) і *RSVP-TE*, розширення *RSVP* (протоколу резервування мережевих ресурсів) для оптимізації та управління трафіком. Також існують розширення протоколу *BGP*, здатні керувати віртуальними каналами в *MPLS* мережі.

Заголовок *MPLS* не вказує тип даних, переданих в *MPLS*-тунелі. У разі, якщо виникає необхідність передати два різних типи трафіку між двома маршрутизаторами так, щоб вони по різному оброблялися маршрутизаторами ядра мережі *MPLS*, необхідно встановити два різних *MPLS*-тунеля для кожного типу трафіку.

MPLS як протокол некоректно порівнювати з протоколом *IP*, оскільки *MPLS* працює разом з *IP* і протоколами *IG*. Основні переваги технології *IP/MPLS*: більш висока швидкість просування *IP*-пакетів по мережі за рахунок скорочення часу обробки маршрутної інформації; можливість організації інформаційних потоків в каналах зв'язку. За допомогою міток кожному інформаційному потоку (наприклад, несучому телефонний трафік) може призначатися необхідний клас обслуговування (*CoS*). Потоки з більш високим *CoS* отримують пріоритет перед усіма іншими потоками. Таким чином, за допомогою *MPLS* забезпечується якість обслуговування (*QoS*), притаманне мереж *SDH* і *ATM*; повне відокремлення один від одного віртуальних корпоративних мереж за рахунок створення для кожної з них своєрідних тунелів; прозорий пропуск через ядро *IP/MPLS* трафіку протоколів *Ethernet*, *Frame Relay* або *ATM*, що дозволяє підключати користувачів, що використовують всі ці різноманітні протоколи.

4.2 Порядок виконання роботи

1. Побудувати мережу в *Cisco Packet Tracer* зображено на Рис. 4.1.
2. Налаштувати мережу, як це зображено на Рис. 4.1.
3. Перевірити коректність налаштувань.
4. Налаштувати *Frame-Mode MPLS* для побудованої мережі.
5. Перевірити коректність налаштувань *MPLS*.

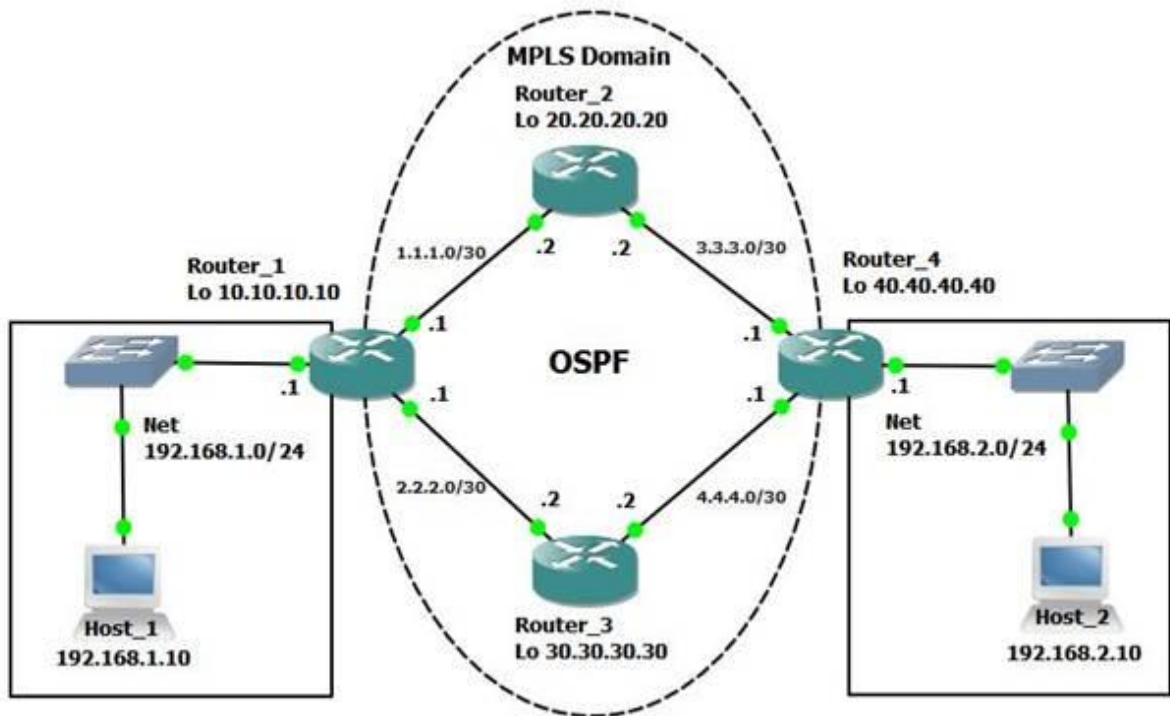


Рисунок 4.1 – Модель мережі

4.3 Порядок захисту

Студент, який виконав завдання лабораторної роботи, подає письмовий звіт про виконання лабораторної роботи і доповідає про результати та дає відповіді на запитання. Виконане завдання оцінюється згідно системи контролю знань, умінь, навичок, наведеної на відповідній сторінці дистанційного курсу.

4.4 Контрольні запитання

1. Що таке *MPLS*?
2. Що таке *Frame-Mode MPLS*?
3. Що таке *OSPF*?

ЗМІСТ

ЛАБОРАТОРНА РОБОТА №1

НАЛАШТУВАННЯ СЕРВІСУ *IP*-ТЕЛЕФОНІЇ ІЗ
ВИКОРИСТАННЯМ *ASTERISK*

3

ЛАБОРАТОРНА РОБОТА №2

ПРОЕКТУВАННЯ МЕРЕЖІ У СИМУЛЯТОРІ *CISCO PACKETTRACER*

7

ЛАБОРАТОРНА РОБОТА №3

НА ЛАШТУВАННЯ *QOS* НА МАРШРУТИЗАТОРАХ *CISCO*,
ВИКОРИСТОВУЮЧИ *CISCO PACKETTRACER*

9

ЛАБОРАТОРНА РОБОТА №4

НАЛАШТУВАННЯ *MPLS* НА МАРШРУТИЗАТОРАХ *CISCO*,
ВИКОРИСТОВУЮЧИ *CISCO PACKETTRACER*

12

Навчально-методична література

Марценко С.В., Поливана У.В.

**МЕТОДИЧНІ ВКАЗІВКИ ДО ЛАБОРАТОРНИХ РОБІТ
З КУРСУ
«МУЛЬТИСЕРВІСНІ ТЕХНОЛОГІЇ В КОМП'ЮТЕРНИХ
МЕРЕЖАХ»**

Комп'ютерне макетування та верстка А.П. Катрич

Формат 60x90/16. Обл. вид. арк. 0,66. Тираж 10 прим. Зам. № 2851.

Тернопільський національний технічний університет імені Івана Пулюя.

46000, м. Тернопіль, вул. Руська 56.

Свідоцтво суб'єкта видавничої справи ДК № 4226 від 08.12.11.