МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ ДЕРЖАВНИЙ ПОЛІТЕХНІЧНИЙ КОЛЕДЖ

Для спеціальності:
123 Комп'ютерна інженерія

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт
з дисципліни: "Комп'ютерні системи та мережі",

Харків 2018

Методичні вказівки до виконання лабораторних робіт з дисципліни: "Комп'ютерні системи та мережі" для студентів освітньо-професійної програми «Обслуговування комп'ютерних систем і мереж» спеціальності 123 «Комп'ютерна інженерія»

Укладачі: В.О.Величко, М.В.Величко - Харків: ХДПК, 2018, 56 с.

Рецензент: к.т.н, проф., декан МТФ ХНТУБА Ю.В. Журавльов

Розглянуто цикловою комісією інформаційних технологій

Протокол №_____ від _____2018 р.

Голова циклової комісії _____М.М. Бочарніков

Схвалено методичною радою коледжу

Протокол від _____2018 р. №_____

Голова методичної ради _____

ЗМІСТ

Лабораторна робота № 1.
Основні функціональні можливості програмного мережевого емулятора Packet Tracer Cisco Systems.

Мета роботи. Вивчити основні функціональні можливості програмного мережевого емулятора Packet Tracer Cisco Systems.

ХІД РОБОТИ

Короткі теоретичні відомості.

Packet Tracer є інтегрованим середовищем моделювання, візуалізації, спільної роботи і оцінки стану мережі. Packet Tracer допомагає студенту та викладачу створювати мережеві моделі, здійснювати візуалізацію і анімацію передачі інформації в мережі. Як і будь-яке моделювання, Packet Tracer спирається на спрощені моделі мережевих пристроїв і протоколів. Реальні комп'ютерні мережі залишаються еталоном для розуміння поведінки мережі та розвитку навичок для їх побудови. Packet Tracer був створений, щоб допомогти вирішити проблему забезпечення доступу до мережного обладнання для студентів і викладачів.

Найменування	***Опис***
Протоколи	**LAN**: Ethernet (including CSMA/CD*), 802.11 a/b/g/n wireless*, PPPOE **Switching**: VLANs, 802.1q, trunking, VTP, DTP, STP*, RSTP*, multilayer switching*, Etherchannel, LACP, PAgP **TCP/IP**: HTTP, HTTPS, DHCP, DHCPv6, Telnet, SSH, TFTP, DNS, TCP*, UDP, IPv4*, IPv6*, ICMP, ICMPv6, ARP, IPv6 ND, FTP, SMTP, POP3, VOIP(H.323) **Routing**: static, default, RIPv1, RIPv2, EIGRP, single-area OSPF, multi-area OSPF, BGP, inter-VLAN routing, redistribution **Other**: ACLs (standard, extended, and named), CDP, NAT (static, dynamic, inside/outside, and overload), NATv6 **WAN**: HDLC, SLARP, PPP*, and Frame Relay* **Security**: IPsec, GRE, ISAKMP, NTP, AAA, RADIUS, TACACS, SNMP, SSH, SYSLOG, CBAC, Zone-based policy firewall, IPS **QoS**: Layer 2 QoS, Layer 3 Diffserv QoS, FIFO Hardware queues, Priority Queuing, Custom Queuing, Weighted Fair Queuing, MQC, NBAR* *** позначає накладені функціональні обмеження**
Логічний простір	Створення мережевої топології. Пристрої: маршрутизатори, комутатори, сховища (Server, Desktop and Laptop), хаби, мости, бездротові точки доступу, бездротові маршрутизатори і DSL / cable модеми. З'єднання пристроїв здійснюється з використанням мідних, оптоволоконних, коаксіальних кабелів.
Фізичний простір	Підтримує наступні види: ієрархія пристроїв, комутаційні шафи, будівлі, міста. Також підтримується відображення допустимої довжини кабелів в мережі Ethernet, масштабування створених користувачем графіків.
Режим реального часу	Обмін даними відбувається в режимі реального часу. Настроюється конфігурація: DHCP, DNS, HTTP, TFTP, Syslog, AAA, and NTP servers
Режим симуляції	Анімація передачі пакетів Лист подій Є широкий вибір протоколів моделі OSI. Користувач має можливість створення сценарію передачі пакетів.

Логічний простір.

Для того, щоб розташувати пристрій, необхідно вибрати його з меню і перетягнути на головну панель.

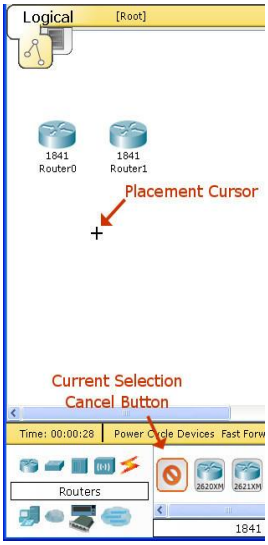


Рис.1 - Вибір пристрою.

Більшість з пристроїв в Packet Tracer мають модулі розширення, необхідні для підключення додаткових портів. Додавання модулів здійснюється в панелі налаштування пристрою. При підключенні нового модуля пристрій повинен бути відключено від електромережі.

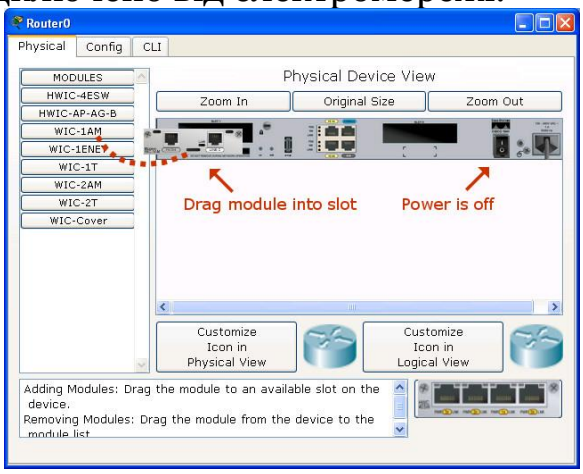


Рис.2 - Додавання модулів розширення.

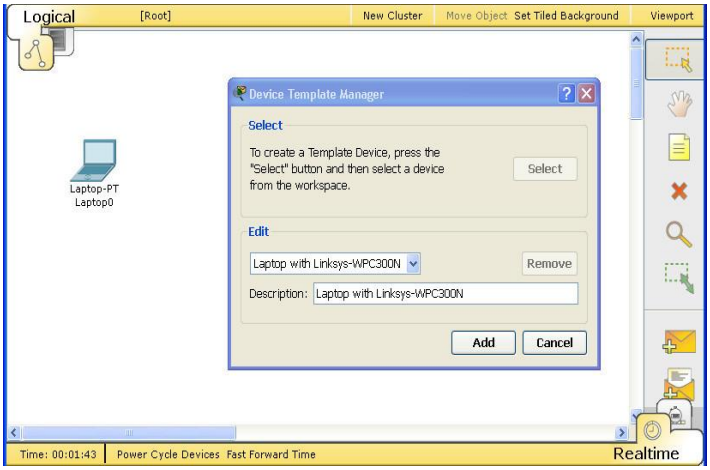


Рис.3 - Створення шаблону пристрою.

Packet Tracer надає можливість створення шаблонів пристроїв. Для створення шаблону необхідно вибрати пристрій, додати необхідні модулі розширення, потім перейти в Custom Devices Dialog. Потім додати опис вибраного пристрою, натиснувши на Select. Додати новий створений користувачем пристрій можна через Custom Made Devices.

Для з'єднання пристроїв між собою необхідно вибрати відповідні кабелі, що розташовані на панелі Connections. Потім потрібно натиснути правою кнопкою миші по одному з пристроїв і вибрати порт підключення. Аналогічні дії виконати для другого пристрою.

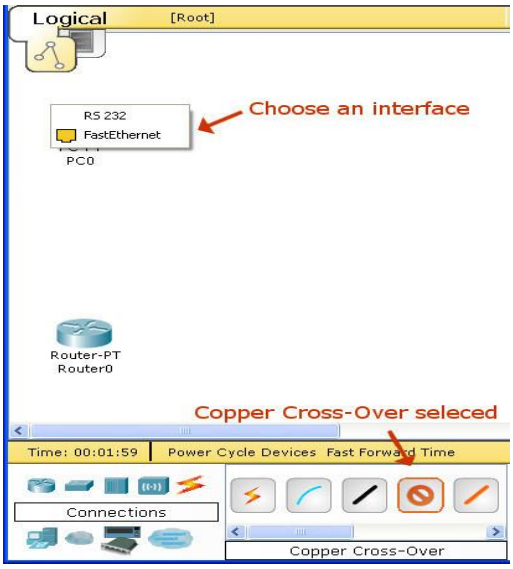


Рис.4 - Створення з'єднань.

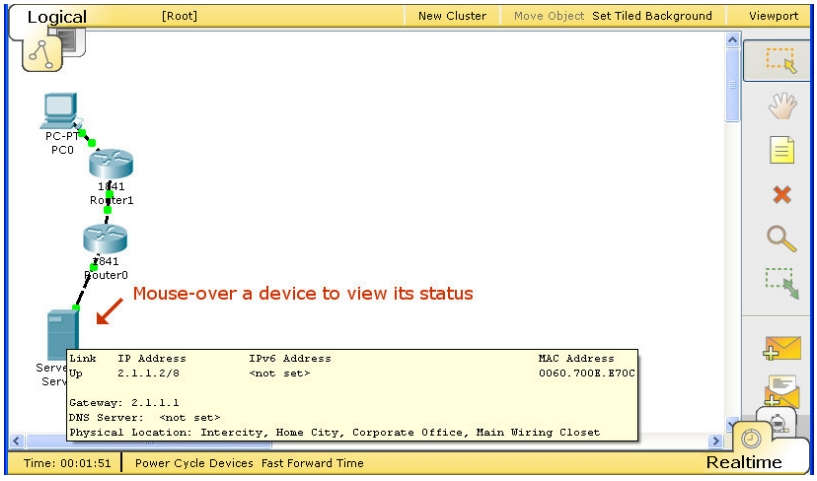


Рис.5 - Функціонування мережі в режимі реального часу.

Режим реального часу.

У режимі реального часу, мережа завжди працює незалежно від дій користувача. Конфігурація мережі здійснюється в реальному часі. При перегляді статистики мережі, вони відображаються в режимі реального часу, як показано на панелі інструментів. На додаток до використання Cisco IOS для настройки і діагностики мережі, ви можете використовувати Add Simple PDU і User Created PDU List для наочної відправки пакета.

Режим симуляції.

У режимі симуляції, ви маєте можливість дивитися свої мережі, працювати в більш повільному темпі, досліджуючи шляхи, по яких пересилаються пакети. При перемиканні в режим моделювання, з'явиться спеціальна панель. Ви можете графічно переглядати поширення пакетів по мережі, натиснувши на кнопку Add Simple PDU. Є можливість контролю швидкості моделювання з використанням кнопки Speed Slider. Також можна переглядати попередні події, натиснувши на кнопку Назад.

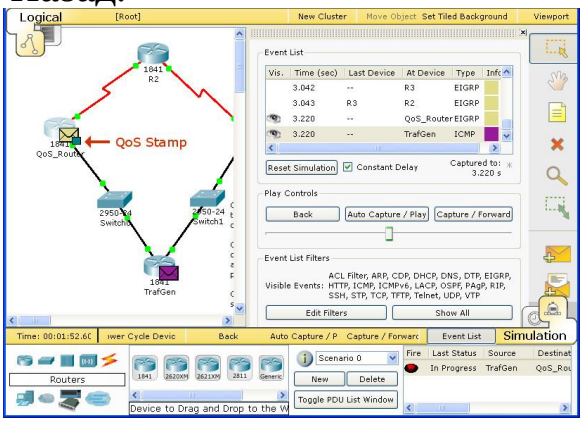


Рис.6 - Режим моделювання.

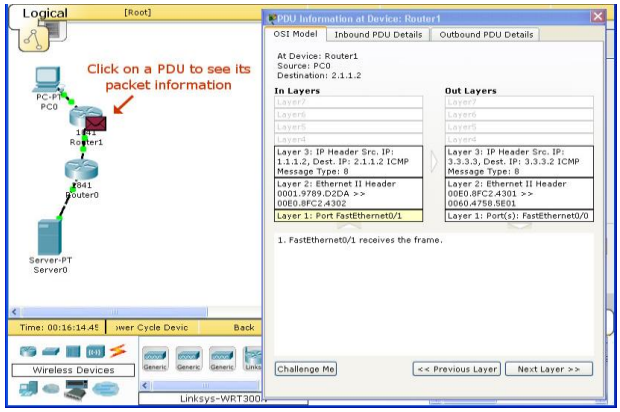


Рис.7 - Інформація про пакет.

Під час моделювання можна клікнути на пакеті, що пересилається і отримати про нього докладну інформацію.

Фізичний простір.

Метою фізичної робочої області є огляд фізичних аспектів логічної топології мережі. Це дає відчуття масштабу і розміщення (як ваша мережа може виглядати в реальному середовищі).

Щоб відобразити фізичний масштаб фізичні робоча область розділена на чотири шари, чотири середовища: міжміський, місто, будинок, і комутаційного вузла. Міжміський є найбільшим з навколишніх середовищ. Він може містити багато міст. Кожне місто може містити безліч будівель. Нарешті, кожна будівля може містити безліч шаф проводки. Розподільна шафа забезпечує вид, який відрізняється від трьох інших видів. Тут ви можете побачити пристрої, які були створені в логічних Workspace; позиціонується в області мережевих технологій стійки і на столах. Три інших шару забезпечують перегляд мініатюр їх макетів. Це за замовчуванням розташовується у фізичної робочої області, але пристрої в шафі можуть бути переміщені в будь-який з шарів.

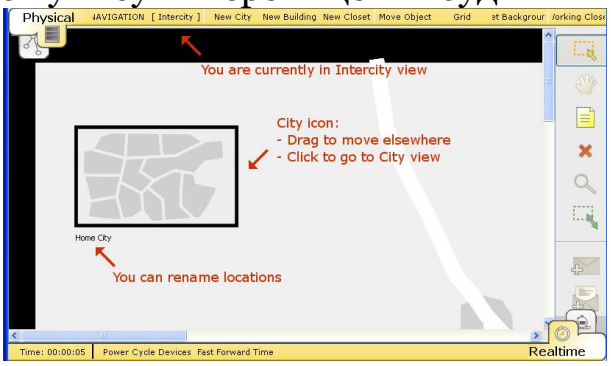


Рис.8 - Міжміський масштаб.

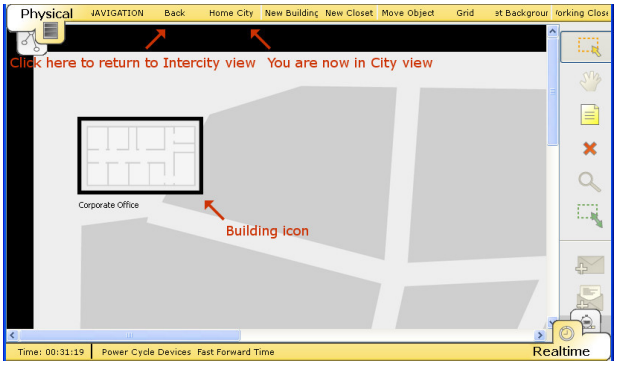


Рис.9 - Масштаб будівлі.

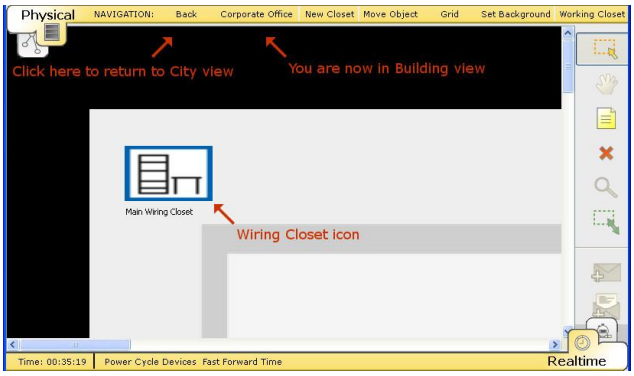


Рис.10 - Комутаційна шафа.

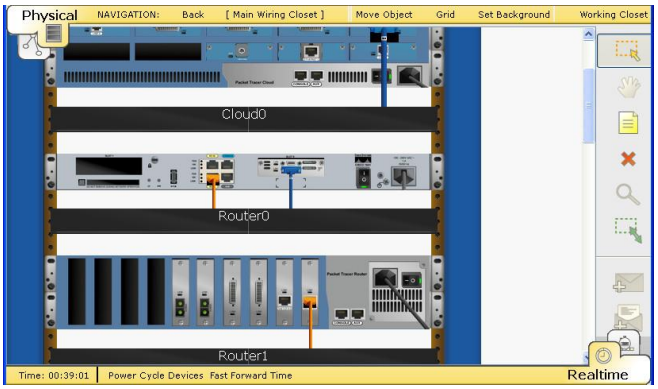


Рис.11 - Пристрої, що розміщені в комутаційної шафі.

Лабораторна робота № 2.
Топологія і побудова мережі в Packet Tracer.

Мета роботи. Ознайомитися з архітектурою стека протоколів TCP/IP з використанням програмного мережевого емулятора Packet Tracer Cisco Systems.

Завдання. Побудувати мережу з використанням маршрутизаторів, комутаторів і кінцевого обладнання.

ХІД РОБОТИ

Короткі теоретичні відомості.

Відкрита система (OSI) - це стандартизований набір протоколів і специфікацій, який гарантує можливість взаємодії обладнання різних виробників. Вона реалізується набором модулів, кожен з яких вирішує просту задачу всередині елемента мережі. Кожен з модулів пов'язаний з одним або декількома іншими модулями. Рішення складного завдання має на увазі певний порядок проходження рішення простих завдань, при якому утворюється багаторівнева ієрархічна структура на рис.12. Це дозволяє будь-яким двом різним системам зв'язуватися незалежно від їх основної архітектури.

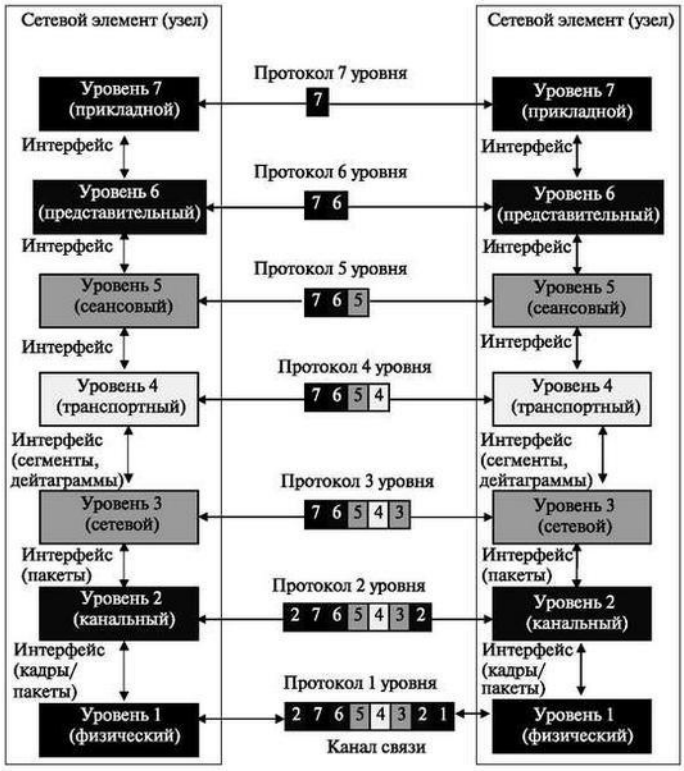


Рис.12 - Модель взаємодії відкритих систем OSI.

Модель OSI складена з семи упорядкованих рівнів: фізичного (рівень 1), ланки передачі даних (рівень 2), мережевого (рівень 3), транспортного (рівень 4), сеансового (рівень 5), уявлення (рівень 6) і прикладного (рівень 7).

Обмін інформацією між модулями відбувається на основі певних правил, які називаються інтерфейсом. Надсилаючи звіт про проблеми модуль верхнього рівня вирішує свою частину завдання, а результат, зрозумілий тільки йому, оформляє у вигляді додаткового поля до вихідного повідомлення (заголовка) і передає змінене повідомлення на до обслуговування в ніжній рівень. Цей процес називається інкапсуляцією.

Стек протоколів Інтернету.

Стек протоколів мережі Інтернет був розроблений до створення моделі OSI. Тому рівні в стеку протоколів Інтернету не відповідають аналогічним рівнями в моделі OSI. Стек протоколів Інтернету складається з п'яти рівнів: фізичного, ланки передачі даних, мережі, транспортного та прикладного. Перші чотири рівні забезпечують фізичні стандарти, мережевий інтерфейс, міжмережева взаємодія і транспортні функції, які відповідають першим чотирьом рівням моделі OSI. Три самих верхніх рівня в моделі OSI представлені в стеку протоколів Інтернету єдиним рівнем, званим прикладним рівнем рис.13.

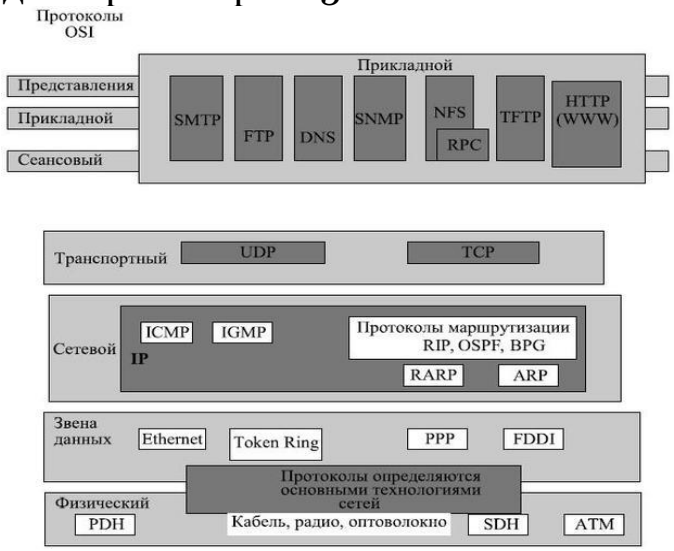


Рис. 13 - Стек протоколів Інтернету в порівнянні з OSI.

Стек базових протоколів Інтернету - ієрархічний, складений з діалогових модулів, кожен з яких забезпечує задані функціональні можливості; але ці модулі не обов'язково взаємозалежні. На відміну від моделі OSI, де визначається строго, які функції належать кожному з її рівнів, рівні набору протоколу TCP/IP містять відносно незалежні протоколи, які можуть бути змішані і узгоджені в залежності від потреб системи. Термін ієрархічний означає, що кожен верхній протокол рівня підтримується відповідно одним або більше протоколами нижнього рівня.

На транспортному рівні стек визначає два протоколи: протокол управління передачею (TCP) і протокол призначених для користувача дейтаграм (UDP). На мережевому рівні - головний протокол міжмережевої взаємодії (IP), хоча на цьому рівні використовуються деякі інші протоколи, про які буде сказано нижче.

Адресація вузлів в IP-мережах.

У мережах TCP/IP прийнято розрізняти адреси мережевих вузлів трьох рівнів:

• фізичний (або локальний) адрес вузла (MAC-адресу мережного адаптера або порту маршрутизатора) - ці адреси призначаються виробниками мережного обладнання;

• IP-адреса вузла (наприклад 192.168.0.1), дані адреси призначаються мережевими адміністраторами або Інтернет-провайдерами;

• символьне ім'я (наприклад, www.microsoft.com) - ці імена також призначаються мережевими адміністраторами компаній або Інтернет-провайдерами.

Розглянемо докладніше IP-адресацію.

Комп'ютери або інші складні мережеві пристрої, підключені до декількох фізичних мереж, мають кілька IP-адрес - по одному на кожен мережевий інтерфейс. Схема адресації дозволяє проводити одиничну, трансляцію і групову адресацію. Таким чином, виділяють 3 типу IP-адрес:

1. Unicast-адреса (одинична адресація конкретному вузлу) - використовується в комунікаціях "один-до-одного".

2. Broadcast-адреса (широкомовна адреса, що відноситься до всіх адрес підмережі)- використовується в комунікаціях "один-до-всіх". У цих адресах поле ідентифікатора пристрою заповнене одиницями. IP-адресація допускає трансляцію передачу, але не гарантує її - ця можливість залежить від конкретної фізичної мережі. Наприклад, в мережах Ethernet широкомовна передача виконується з тією ж ефективністю, що і звичайна передача даних, але є мережі, які взагалі не підтримують такий тип передачі або підтримують з обмеженнями.

3. Multicast-адреса (груповий адрес для многоадресної відправки пакетів) - використовується в комунікаціях "один-до-бататьох". Підтримка групової адресації використовується в багатьох додатках, наприклад, додатках інтерактивних конференцій. Для групової передачі робочі станції і маршрутизатори використовують протокол IGMP, який надає інформацію про приналежність пристроїв певним групам.

Клас мережі	Найменший ідентифікатор мережі	Найбільший ідентифікатор мережі	Кількість мереж
Класс А	1.0.0.0	126.0.0.0	126
Класс В	128.0.0.0	191.255.0.0	16384
Класс С	192.0.0.0	223.255.255.0	2097152

Мережеве обладнання.

Міст (bridge), як і ретранслятор, може з'єднувати сегменти або локальні мережі робочих груп. Однак, на відміну від репітера, міст також служить для розбиття мережі, що допомагає ізолювати трафік або окремі проблеми. Наприклад, якщо трафік одного-двох комп'ютерів або одного відділу "затоплює" мережу пакетами, зменшуючи її продуктивність в цілому, міст ізолює ці комп'ютери або цей відділ. Мости зазвичай вирішують такі завдання. Збільшують розмір мережі. Збільшують максимальну кількість комп'ютерів в мережі.

У середовищі, що поєднує кілька мережевих сегментів з різними протоколами і архітектурою, мости не завжди гарантують швидкий зв'язок між усіма сегментами. Для такої складної мережі необхідний пристрій, який не тільки знає адресу кожного сегмента, а й визначає найкращий маршрут для передачі даних і фільтрує широкомовні повідомлення. Такий пристрій називається маршрутизатором.

Маршрутизатор (routers) працює на мережевому рівні моделі OSI. Це означає, що вони можуть переадресовувати і маршрутизувати пакети через безліч мереж, обмінюючись інформацією (яка залежить від протоколу) між окремими мережами. Маршрутизатори зчитують в пакеті адресну інформацію складної мережі і,

оскільки вони функціонують на більш високому в порівнянні з мостами рівні моделі OSI, мають доступ до додаткових даних. Маршрутизатори можуть виконувати такі функції мостів: фільтрувати і ізолювати трафік; з'єднувати сегменти мережі.

Пристрій Switch - комутатор. Це обладнання відноситься до активного мережевого обладнання, і служить для обробки пакетів в мережі. Свитч (те ж саме, що перемикач, міст, switch, bridge) - пристрій, що служить для поділу мережі на окремі сегменти, які можуть містити хаби і мережеві карти. Світчі є пристроями 2-го рівня, тобто містять в собі порти - пристрої 1-го рівня для роботи з сигналами, але крім того, працюють з вмістом мережевих пакетів - читають поле фізичного адреси призначення (MAC) пакета, що прийшов на один з портів, і в залежності від його значення і таблиці MAC -адрес - порт "ретранслюють" пакет на інший порт (або не ретранслюють).

Нижче на рис.14 приведена спроектована мережа, яка включає в себе наступне обладнання:

- Маршрутизатор;
- Комутатори;
- ПК;
- IP-телефони;
- Сервер.

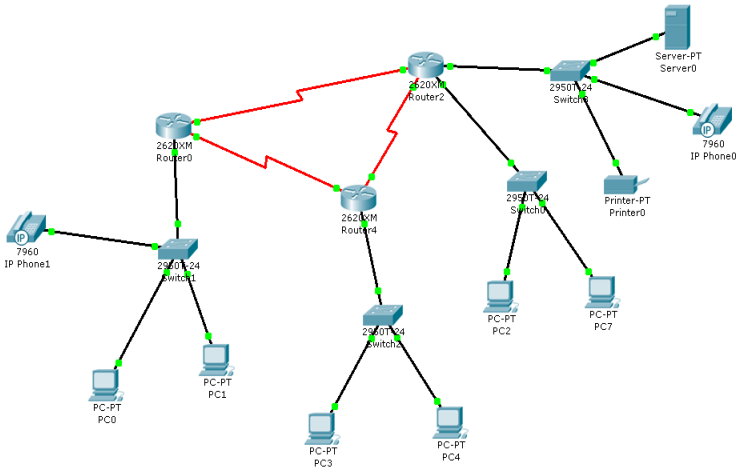


Рис.14 - Результат побудови мережі.

Маршрутизатори з'єднуються між собою за допомогою DCE - кабелю. В даній мережі маршрутизатори використовують RIP - протокол для здійснення передачі даних між різними підмережами (рис. 15).

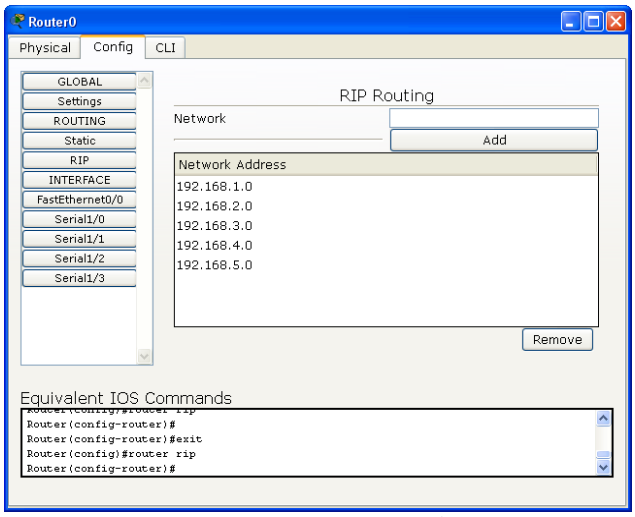


Рис.15 - Список підмереж.

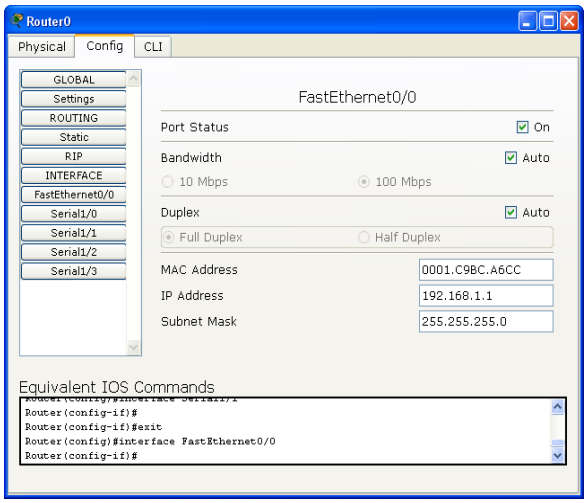
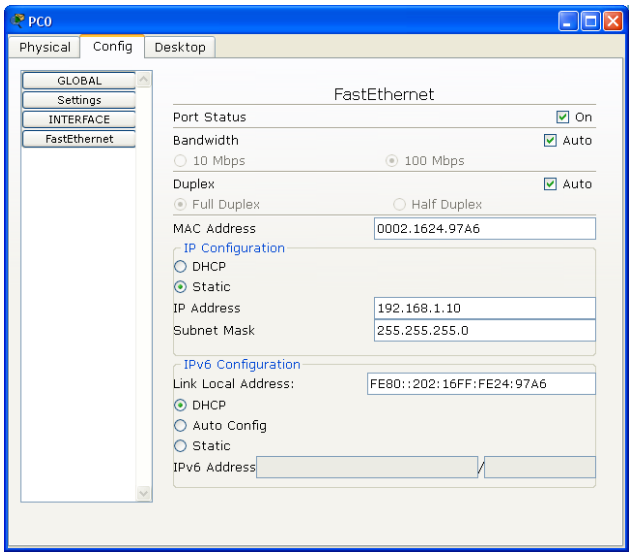


Рис.16 - Призначення IP-адресу. маршрутизатору.



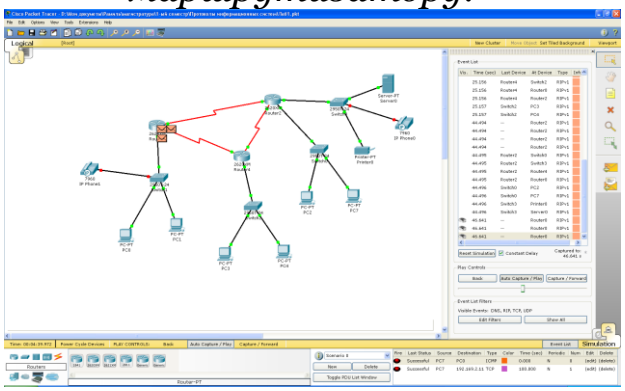


Рис.17 - Призначення IP-адрес ПК.

При цьому, IP-адреси призначаються статично для маршрутизаторів і кінцевого обладнання (рис. 16,17).

Зміст звіту:

1. Короткі теоретичні відомості;
2. Налаштування мережевого обладнання;
3. Змодельована мережа передачі даних із зазначенням параметрів настроювання.

Лабораторна робота №3.
Аналіз протоколів рівня додатки і транспорту.

Мета роботи. Провести аналіз роботи протоколів програм і транспорту з використанням програмного мережевого емулятора Packet Tracer Cisco Systems.

ХІД РОБОТИ

Короткі теоретичні відомості.

Модель TCP/IP.

Сімейство протоколів TCP/IP засноване на чотирирівневої еталонної моделі. Всі протоколи, що входять в сімейство протоколів TCP/IP, розташовані на трьох верхніх рівнях цієї моделі.

Кожен рівень моделі TCP/IP відповідає одному або декількох рівнях еталонної моделі OSI (Open Systems Interconnection - взаємодія відкритих систем), запропонованої ISO - міжнародна організація по стандартам (International Standards Organization).

Типи служб і протоколів, використовуваних на кожному рівні моделі TCP/IP, більш детально описані в наступній таблиці.

Рівень	Опис	Протоколи
Додаток	Визначає прикладні протоколи TCP/IP і інтерфейс програм зі службами транспортного рівня, необхідний для використання мережі.	HTTP, Telnet, FTP, TFTP, SNMP, DNS, SMTP, X Windows, інші прикладні протоколи
Транспортний	Забезпечує управління сеансами зв'язку між комп'ютерами. Визначає рівень служб і стан підключення, використовувані при транспортуванні даних.	TCP, UDP
Інтернет	Упаковує дані в IP-датаграми, що містять інформацію про адреси джерела і приймача, яка використовується для перенаправлення датаграмм від вузла до вузла і по мережах. Виконує маршрутизацію IP-датаграм.	IP, ICMP, ARP, RARP
Мережевого інтерфейсу	Визначає засоби і принципи фізичної передачі даних по мережі, включаючи перетворення бітів даних в електричні або інші сигнали апаратними пристроями, безпосередньо підключеними до середовища передачі, такий як коаксіальний кабель, оптоволокно або кручена пара.	Ethernet, Token Ring, FDDI, X.25, Frame Relay, RS-232, v.35

Протоколи рівня додатків.

SMTP (Simple Mail Transfer Protocol) - простий поштовий протокол. Він підтримує передачу поштових повідомлень електронної пошти через мережу Інтернет. Протокол називається простим, тому що забезпечує передачу інформації користувачам, готовим до негайної доставки. Передача здійснюється в режимі 7-бітових слів. Він вимагає наявності програм переходу від прийнятого в більшості програм формату з 8-розрядними словами до формату з 7-розрядними словами.

Система підтримує:

• посилку одиночних повідомлень одному або кільком одержувачам;

• посилку повідомлень, що включають в себе текст, голосові повідомлення, відео або графічні матеріали.

Протокол передачі файлів (FTP - File Transfer Protocol) використовується для передачі файлів від одного комп'ютера до іншого. Забезпечує перегляд каталогів віддаленого комп'ютера, копіювання, видалення і пересилання файлів. FTP відрізняється від інших протоколів тим, що встановлює два з'єднання між хостами. Одне використовується для передачі інформації, а інше - для управління передачею.

DNS (Domain Name System) - служба доменних імен. Вона здійснює присвоєння унікальних імен всім користувачам і вузлів мережі Інтернет і встановлює логічний зв'язок з їх мережевими адресами. Доменне ім'я представляється ієрархічною структурою, що має кілька рівнів. Типові імена доменів верхнього рівня закріплені таким чином:

.com - комерційні організації;

.gov - урядові установи;

.org - некомерційні організації;

.net - центри підтримки мережі;

.int - міжнародні організації;

.mil - військові структури.

SNMP (Simple Network Management Protocol) - простий протокол управління мережею. Він забезпечує набір фундаментальних дій зі спостереження і обслуговування Інтернету.

Протокол розроблений так, щоб він міг контролювати пристрої, створені різними виробниками і встановлені на різних фізичних мережах. Іншими словами, SNMP звільняє завдання управління від обліку фізичних характеристик керованих пристроїв і від основної технології організації мережі.

Мережева файлова система (NFS - Network File System). Це один з багатьох протоколів (наприклад, протокол RPC - Remote Procedure Call - виклик віддаленої процедури), який дозволяє використання файлів, що містять процедури управління і периферії в іншому комп'ютері.

Тривіальний (найпростіший) протокол передачі файлів TFTP (Trivial File Transfer Protocol). Використовується в простих випадках при початковому завантаженні робочих станцій або завантаженні маршрутизаторів, які не мають зовнішньої пам'яті.

Протокол передачі гіпертексту (HTTP - Hyper Text Transfer Protocol) - транспортний протокол, який застосовується в Інтернеті при обміні документами, представленими на мові опису гіпертекстових документів.

Мова розмітки гіпертексту (HTML - Hyper Text Markup Language). Є одним з головних мов, які використовуються в мережі WWW.

Протоколи рівня транспорту.

Протокол управління передачею TCP (Transmission Control Protocol) є обов'язковим стандартом TCP / IP, який описаний в документі RFC 793 «Transmission Control Protocol (TCP)» і надає надійну службу доставки пакетів, орієнтовану на встановлення з'єднання. Протокол TCP:

• гарантує доставку IP-датаграм;

• виконує розбиття на сегменти і складання великих блоків даних, що відправляються програмами;

• забезпечує доставку сегментів даних в потрібному порядку;

• виконує перевірку цілісності переданих даних за допомогою контрольної суми;

• посилає позитивні підтвердження, якщо дані отримані успішно. Використовуючи виборчі підтвердження, можна також надсилати негативні підтвердження для даних, які були отримані;

• пропонує кращий транспорт для програм, яким потрібна надійна передача даних з встановленням сеансу зв'язку, наприклад для баз даних «клієнт-сервер» і програм електронної пошти.

Як працює TCP.

TCP заснований на зв'язку «точка-точка» між двома вузлами мережі. TCP отримує дані від програм і обробляє їх як потік байтів. Байти групуються в сегменти, яким TCP привласнює послідовні номери, необхідні для правильного складання сегментів на вузлі-приймачі.

Щоб два вузла TCP могли обмінюватися даними, їм потрібно спочатку встановити сеанс зв'язку один з одним. Сеанс TCP ініціалізується за допомогою процесу, званого трьохетапним встановленням зв'язку. У цьому процесі синхронізуються номера послідовності і передається інформація, що управляє, необхідна для встановлення віртуального з'єднання між вузлами.

По завершенні процесу трьохетапного встановлення зв'язку починається пересилання і підтвердження пакетів в послідовному порядку між цими вузлами. Аналогічний процес використовується TCP перед припиненням з'єднання для того, щоб переконатися, що обидва вузла закінчили передачу і прийом даних.

Протокол UDP

Протокол датаграм користувача UDP (User Datagram Protocol) є стандартом TCP / IP, опис в документі RFC 768 «User Datagram Protocol (UDP)». UDP використовується деякими програмами замість TCP для швидкої, простої, но ненадійної передачі даних між вузлами TCP / IP.

UDP забезпечує службу датаграм, що НЕ орієнтована на встановлення з'єднання, що означає, що UDP немає гарантії ні доставку, ні правильність порядку доставки датаграм. Вузол-джерело, якому потрібен надійний зв'язок, повинен використовувати або протокол TCP, або програму, яка сама забезпечує підтвердження и стеже за правильністю порядку датаграм.

UDP	**TCP**
Служба, що не орієнтована на встановлення з'єднання; сеанс зв'язку між вузлами не встановлюється.	Служба, орієнтована на встановлення з'єднання; між вузлами встановлюється сеанс зв'язку.
UDP не гарантує і не підтверджує доставку даних, а також не гарантує порядок їх доставки.	TCP гарантує доставку за допомогою підтверджень і контролю порядку отриманих даних.
Програми, що використовують UDP, відповідальні за забезпечення надійності передачі даних.	Програмами, які використовують TCP, гарантується надійність передачі даних.

UDP - швидкий протокол з невеликими накладними витратами, що підтримує зв'язок «точка-точка» і «точка-багато точок».	TCP повільніше, вимагає великих накладних витрат і підтримує тільки зв'язок «точка-точка».

Нижче на рис.18 приведена спроектована мережа, яка включає в себе наступне обладнання:

- Маршрутизатор;
- ПК;
- Сервер.

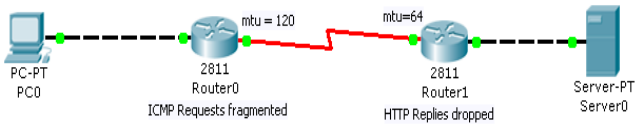


Рис. 18 - Результат побудови мережі.

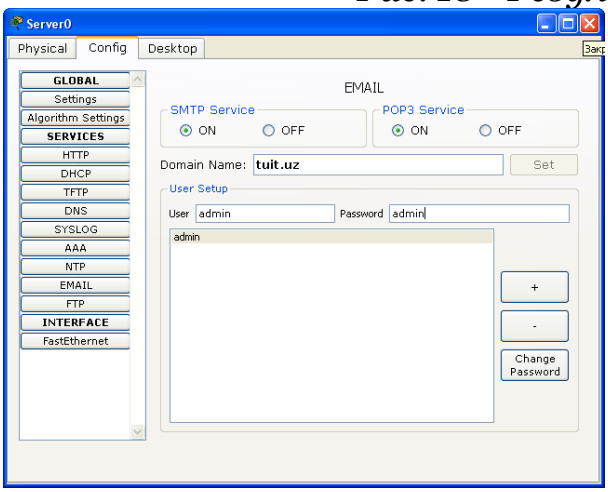


Рис.19 - Настроювання e-mail.

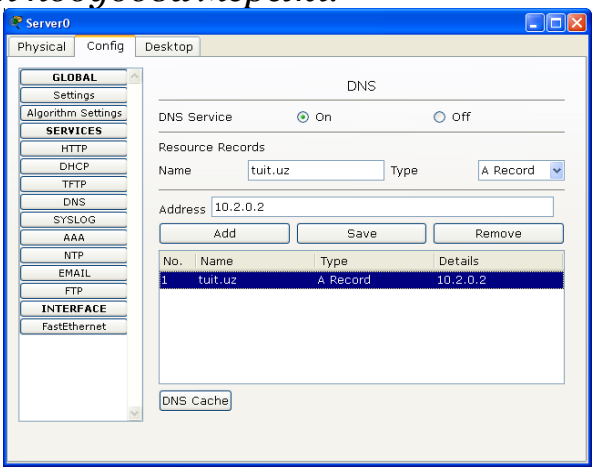


Рис.20 - Настроювання DNS-сервісу.

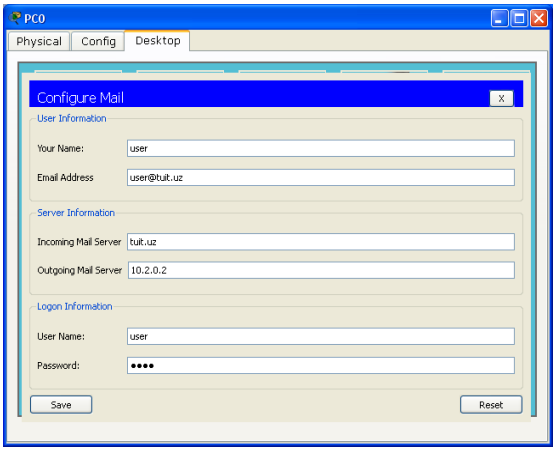


Рис.21 - Настроювання електронної пошти.

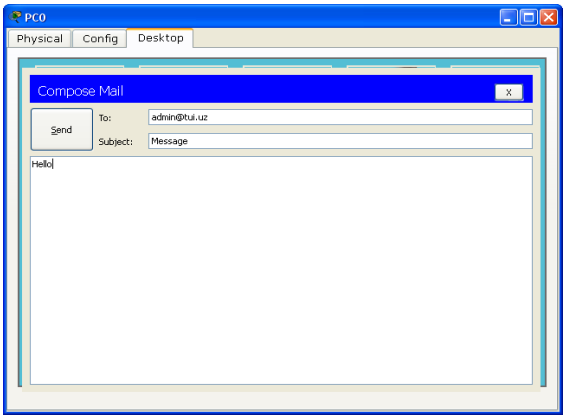


Рис.22 - Передача повідомлення.

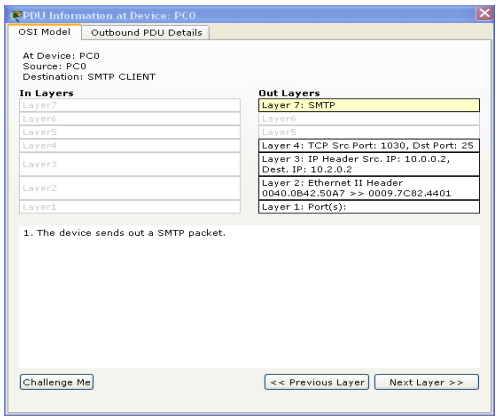


Рис.23 - Передача пакетів SMTP.

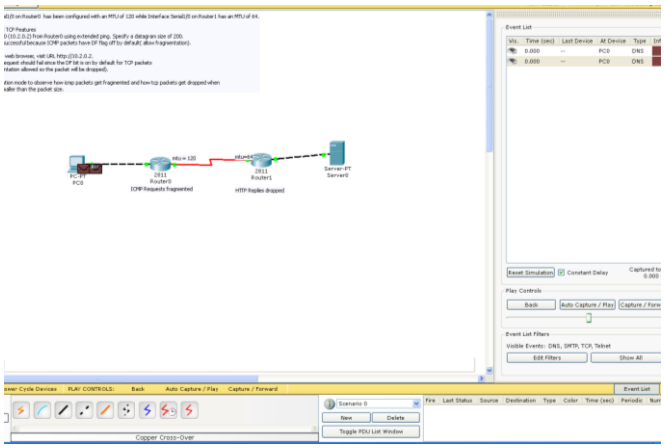


Рис.24 - Робота DNS-сервісу.

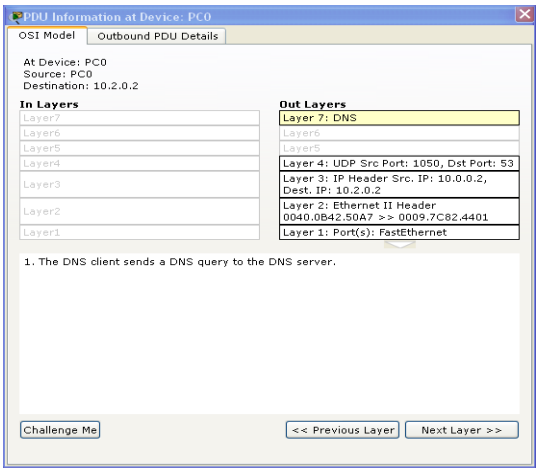


Рис.25 - Виконання DNS-запросу.

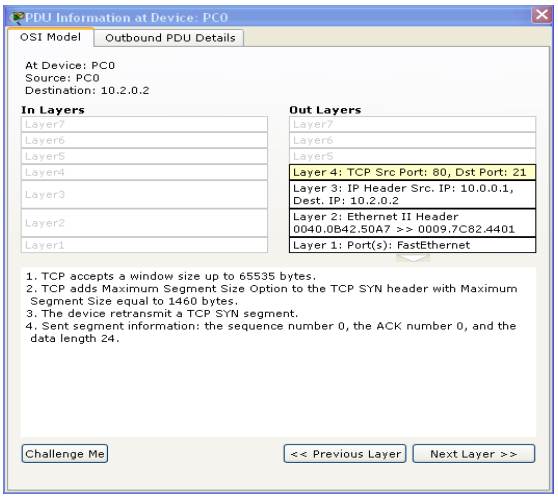


Рис.26 - Передача TCP-пакетів.

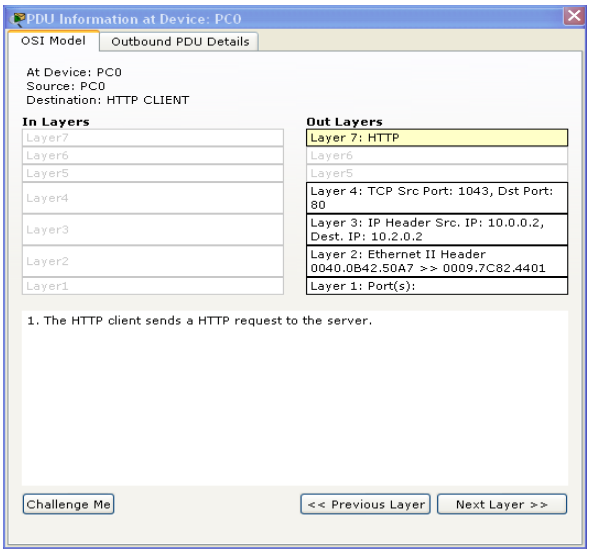


Рис.27 - Виконання HTTP-запросу.

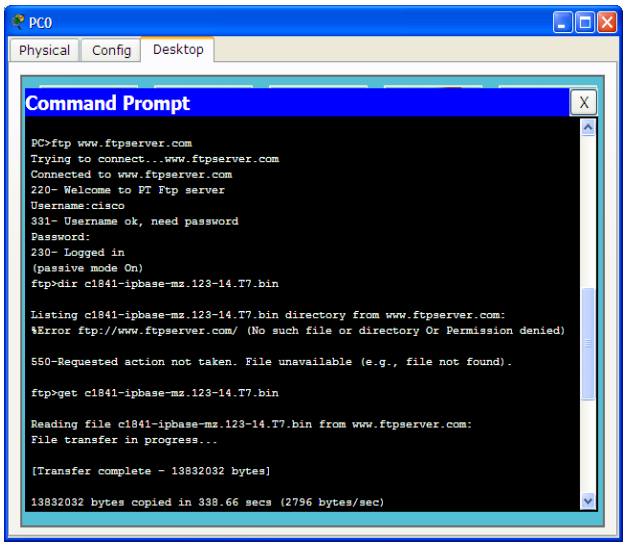


Рис.28 - Робота с FTP-сервером.

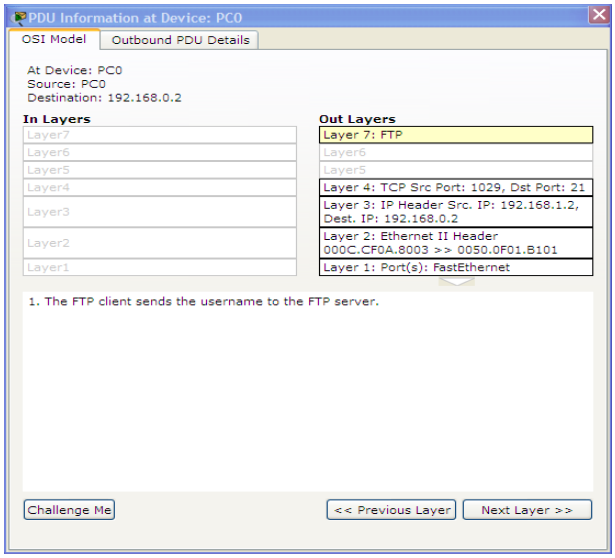


Рис.29 - Авторизація користувача на FTP-сервері.

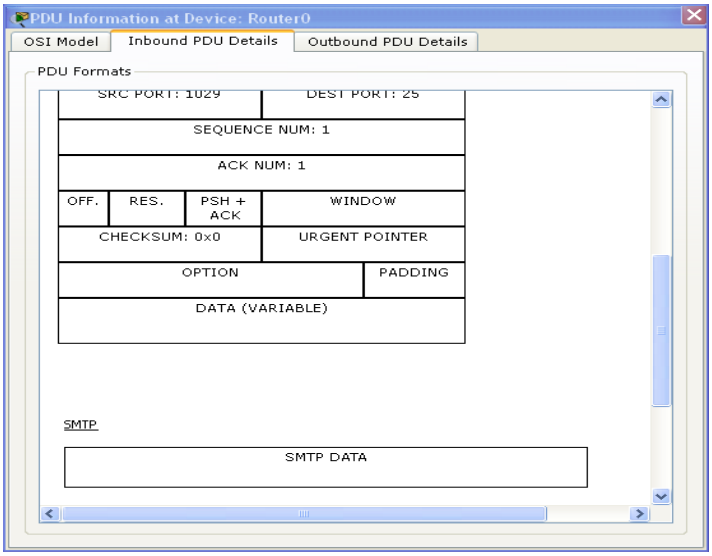


Рис.30 - Структура пакетів.

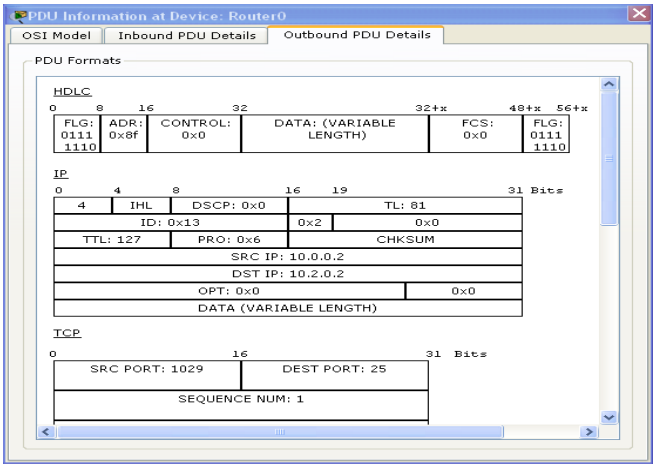


Рис.31 - Структура пакетів.

Зміст звіту:

1. Короткі теоретичні відомості;
2. Аналіз протоколів рівня додатку і транспорту;
3. Порівняння протоколів TCP і UDP;
4. Змодельована мережа передачі даних із зазначенням параметрів настроювання.

Лабораторна робота №4.
E-mail послуги та протоколи.

Мета роботи. Провести аналіз роботи E-mail протоколів і послуг, що надаються з використанням Wireshark.

ХІД РОБОТИ

Короткі теоретичні відомості.

Головною метою протоколу SMTP (Simple Mail Transfer Protocol, RFC-821, -822) служить надійна і ефективна доставка електронних поштових повідомлень. SMTP є досить незалежною субсистемою і вимагає тільки надійного каналу зв'язку. Середовищем для SMTP може служити окрема локальна мережа, система мереж або весь Інтернет.

SMTP базується на наступній моделі комунікацій: у відповідь на запит користувача поштова програма-відправник встановлює двосторонній зв'язок з програмою-приймачем (TCP, порт 25). Одержувачем може бути крайовий або проміжний адресат. SMTP-команди генеруються відправником і надсилаються одержувачу. На кожну команду повинен бути відправлений і отриманий відгук.

Коли канал організований, відправник посилає команду MAIL, ідентифікуючи себе. Якщо одержувач готовий до прийому повідомлення, він посилає позитивне підтвердження. Далі відправник посилає команду RCPT, ідентифікуючи одержувача поштового повідомлення (таких команд можна видати кілька, якщо число одержувачів більше одного). Якщо одержувач може прийняти повідомлення для кінцевого адресата, він видає знову позитивне підтвердження. В іншому випадку він відкидає отримання повідомлення для даного адресата, але не взагалі поштової посилки.

SMTP-відправник і SMTP-одержувач можуть вести діалог з декількома кінцевими користувачами (Рис. 32). Будь-яке поштове повідомлення завершується спеціальною послідовністю символів. Якщо одержувач успішно завершив прийом та обробку поштового повідомлення, він посилає позитивне підтвердження.

SMTP забезпечує передачу поштового повідомлення безпосередньо кінцевого одержувача, коли вони з'єднані безпосередньо. В іншому випадку пересилання може виконуватися через одну або більше проміжних "поштових станцій".

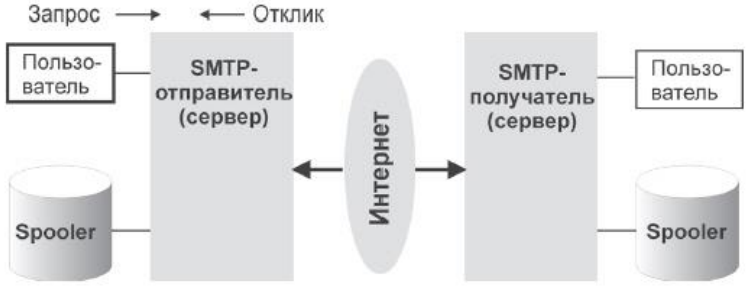


Рис. 32. Схема взаємодії різних частин поштової системи.

Для вирішення поставленого завдання SMTP-сервер повинен знати ім'я кінцевого одержувача і назва поштової скриньки місця призначення. Аргументом команди MAIL є адреса відправника (зворотну адресу). Аргументом команди RCPT служить адреса кінцевого одержувача. Зворотна адреса використовується для посилки повідомлення в разі помилки.

Всі відгуки мають цифрові коди. Команди, відгуки та імена ЕОМ не чутливі до того, малі або прописні символи використані при їх написанні, але це не завжди справедливо при написанні імен та адрес одержувача.

Поштовий протокол SMTP працює тільки з ASCII-символами. Якщо транспортний канал працює з октетами, 7-бітові коди будуть доповнені нульовим восьмим бітом. Саме тут полягала проблема пересилання поштових повідомлень російською мовою (російський алфавіт вимагає 8-бітового подання). Проблема посилюється тим, що для російського алфавіту прийнято 4 кодових уявлення.

Як вже було сказано, процедура відправлення поштового повідомлення починається з посилки команди MAIL, яка має формат:

MAIL <SP> FROM:<reverse-path> <CRLF>,

де <SP> - пробіл, <CRLF> - комбінація кодів повернення каретки і переходу на новий рядок, а <reverse-path> - зворотний шлях (ім'я поштової скриньки відправника). Саме цю адресу використовується, якщо одержувач повідомлення скористається командою reply.

Ця команда повідомляє SMTP-одержувачу, що стартує нова процедура і слід скинути в початковий стан все статусні таблиці, буфери і т.д. Якщо команда пройшла, одержувач реагує відгуком: 250 OK.

Аргумент може містити не тільки адресу поштової скриньки - в загальному випадку він є списком адрес ЕОМ-серверів, через які прийшло дане повідомлення, включаючи, зрозуміло, і адресу поштової скриньки відправника. Першим в списку <reverse-path> адреса ЕОМ-відправника. Після проходження команди MAIL посилається команда RCPT:

RCPT <SP> TO:<forward-path> <CRLF>

Ця команда вказує адресу кінцевого одержувача (<forward-path>). При благополучному проходженні команди одержувач посилає код відклик 250 OK, і запам'ятовує отриманий адресу. Якщо одержувач невідомий, SMTP-сервер пошле відгук 550 Failure reply. Команда RCPT може повторюватися скільки завгодно раз, якщо адресат не один.

Аргумент може містити не тільки адресу поштової скриньки, а й маршрутний список ЕОМ по дорозі до нього. Першим у цьому списку має стояти ім'я ЕОМ, що отримала цю команду. По завершенні цього етапу надсилається власне повідомлення:

DATA <CRLF>

При правильному прийомі цього повідомлення SMTP-сервер реагує посилкою відгуку 354 Intermediate reply (проміжний відгук), і розглядає всі наступні рядки в якості поштового тексту. При отриманні коду кінця тексту відправляється відгук: 250 OK.

Ознакою кінця поштового повідомлення є точка в самому початку рядка, за якою слідує <CRLF>.

У деяких випадках адреса місця призначення може містити помилку, але одержувач знає правильну адресу. Тоді можливі два варіанти відгуку:

1. 251 User not local; will forward to <forward-path>

Це означає, що одержувач бере на себе відповідальність за доставку повідомлення. Таке трапляється, коли адресат, наприклад, мігрував в іншу субмережу в межах зони дії даного поштового сервера.

2. 551 User not local; please try <forward-path>

Одержувач знає правильну адресу і пропонує відправнику переадресувати повідомлення за адресою <forward-path>.

SMTP має команди для перевірки коректності імені адресата (VRFY) і розширення списку адрес (EXPN). Обидві команди в якості аргументів використовують рядки символів (в деяких реалізаціях ці дві команди за своєю функцією ідентичні). Для команди VRFY параметром є ім'я користувача, а відгук може містити його повне ім'я та адресу його поштової скриньки.

Реакція на команду VRFY залежить від аргументу. Так якщо серед клієнтів поштового сервера є два користувачі з ім'ям Ivanov, відгуком на команду "VRFY Ivanov" буде "553 User ambiguous". У загальному випадку команда VRFY Ivanov може отримати в якості відгуків:

250 Vasja Ivanov Ivanov@cl.itep.ru
чи:
251 User not local; will forward to Ivanov@cl.itep.ru
чи:
550 String does not match anything (данная строка ничему не соответствует).
чи:
551 User not local; please try Vasja@ns.itep.ru
чи:
VRFY Chtozachertovchina
553 User ambiguous (несуществующее имя)

У разі роздрукування списку адрес відгук займає кілька рядків, наприклад:
EXPN Example-People
250-Juri Semenov Semenov@ns.itep.ru
250-Alexey Sher Sher@suncom.itep.ru
250-Andrey Bobyshev Bobyshev@ns.itep.ru
250-Igor Gursky Gursky@ns.itep.ru

У деяких системах аргументом команди EXPN може бути ім'я файлу, що містить список поштових адрес.

Команда Send And MaiL (SAML) передбачає доставку повідомлення на екран терміналу адресата і занесення в його поштову скриньку. Для відкриття і закриття комунікаційного каналу використовуються команди:

HELO <SP> <domain> <CRLF>, де <domain> — ім'я запитувача домену.
QUIT <CRLF>

Вираз <forward-path> може бути маршрутом, що має вигляд "@ ONE, @ TWO: VANJA @ THREE", де ONE, TWO і THREE - імена ЕОМ. Це підкреслює відмінність між адресою та маршрутом. Концептуально елементи з <forward-path> переносяться в <reverse-path> при пересиланні повідомлень від одного SMTP-сервера до іншого.

Якщо SMTP-сервер виявить, що доставка повідомлення за адресою неможлива, тоді він формує повідомлення про "недоставлення листа", використовуючи <reverse-path>. Слід також пам'ятати, що і прямий, і зворотний адреси-маршрути, взагалі кажучи, можуть не мати нічого спільного з текстом заголовка поштового повідомлення.

Якщо ви або ваша програма не вказали зворотної адреси, не слід думати, що це завадить роботі поштової програми і вона не буде знати, куди посилати відгуки. Практично всі поштові програми дозволяють довільно модифікувати поле <reverse-path>. Це може бути зручно, якщо ви збираєтеся у відрядження, але ця можливість широко використовується і спамерами.

Поле <reverse-path> застосовується поштовою програмою, коли ви відповідаєте на отримане повідомлення за допомогою утиліти Reply. Таким чином, ваш обурена відповідь спамеру може прийти, наприклад, до вас самому.

Слід пам'ятати, що зворотня IP-адреса (адреса відправника) вказана в кожному пакеті, що посилається адресату!

Протокол POP3 (Post Office Protocol).

Протокол обміну поштової інформацією POP3 (RFC-1939) призначений для розбору пошти з поштових скриньок користувачів на їх робочі місця за допомогою програм-клієнтів. Якщо по протоколу SMTP користувачі відправляють кореспонденцію через Інтернет, то по протоколу POP3 користувачі отримують кореспонденцію зі своїх поштових скриньок на поштовому сервері в локальні файли.

Отже, POP3 (Post Office Protocol version 3). Номер його TCP-порту - 110. Основна відмінність POP3 від інших Інтернет-протоколів верхнього рівня полягає в тому, що в ньому відсутня широкий спектр кодів помилок: у відповідь на будь-яку команду він посилає рядки, що починаються з "+ ОК" або "-ERR ", які сигналізують відповідно про успішне або невдале виконання команди.

Набір основних команд протоколу також досить простий:

USER name.

Це перше, що посилає клієнт після того, як він вважав рядок-вітання + ОК POP3 served ready. Аргумент name вказує ім'я користувача на даному поштовому сервері, для якого потрібно отримати доступ до поштової скриньки.

Приклад:

USER paaa

PASS password.

Після того як користувач вказав своє ім'я, він повинен вказати пароль до своєї поштової скриньки.

Приклад:

PASS doom

STAT.

Якщо користувач існує і правильно ввів свій пароль, він може подивитися пошту. Команда STAT повідомляє поточний стан ящика. Формат відповіді - "+ ОК n m", де n - кількість повідомлень, m - кількість байт. Приклад (в поштовій скриньці знаходяться 11 листів загальним обсягом 1594 байт):

C: STAT

S: +OK 11 1594

LIST n.

Для оцінки розміру конкретного листа існує команда LIST. Формат виведення такої ж, як у STAT. Приклад (третій лист має розмір 512 байт):

C: LIST 3

S: +OK 3 512

RETR n.

Для отримання листа з сервера використовується команда RETR n, де n - номер листа. Приклад (отримуємо четвертий лист):

C: RETR 4
S: +OK 124 octets
S: Здесь
S: идет
S: текст
S: письма
S:.

DELE n.

Після того як листи успішно отримані і збережені локально, їх можна видалити з поштового сервера. Для цього служить команда DELE n .. Приклад (видаляємо перший лист):

З: DELE 1
S: + OK message 1 deleted

NOOP

(Не використовує жодних аргументів). При реалізації цієї команди сервер не робить нічого, лише посилає позитивний відгук.

RSET

(Не використовує жодних аргументів) Якщо будь-які повідомлення помічені як видалені, сервер POP3 видаляє цю позначку і повертає позитивний відгук. наприклад:

К: RSET
С: +OK maildrop has 2 messages (320 octets)

TOP msg n,

де msg - номер повідомлення, а n - число рядків (застосовується тільки в режимі TRANSACTION).

При позитивному відгуку на команду TOP сервер посилає заголовки повідомлень і слідом за ними n рядків їх тексту. Якщо n більше числа рядків в повідомленні, надсилається все повідомлення.

UIDL [msg],

де msg - номер повідомлення є опціонним (Unique-ID Listing).

APOP name digest,

де name - ідентифікатор поштової скриньки, a digest - дайджест повідомлення - MD5 (RFC-1828). Команда використовується тільки на стадії авторизації.

Зазвичай будь-яка сесія починається з обміну USER / PASS. Але так як в деяких випадках підключення до сервера POP3 може здійснюватися досить часто, зростає ризик перехоплення пароля. Альтернативним методом авторизації є використання команди APOP. Сервер, який підтримує застосування команди APOP, додає тимчасову мітку в свою стартову повідомлення. Синтаксис тимчасової мітки відповідає формату ідентифікаторів повідомлень, описаного в [RFC822], і повинен бути унікальним для всіх заголовків повідомлень.

Для завершення сеансу використовується команда QUIT.

Протокол IMAP (Internet Message Access Protocol).

IMAP надає користувачеві великі можливості для роботи з поштовими скриньками, що знаходяться на центральному сервері. Поштова програма, яка використовує цей протокол, отримує доступ до сховища кореспонденції на сервері

так, як ніби ця кореспонденція розташована на комп'ютері одержувача. Електронними листами можна маніпулювати з комп'ютера користувача (клієнта) без постійної пересилання з сервера і назад файлів з повним змістом листів.

Для завершення сеансу використовується команда QUIT. Люба процедура починається з команди клієнта. Люба команда клієнта починається з префіксу-ідентифікатора (звичайно короткий буквено-цифровий рядок, наприклад A0001, A0002 та ін.), Що називається міткою (тег). Для кожної команди клієнт генерує свою мітку. Мається два випадки, коли рядок, посланий клієнтом, не представляє собою закінчену команду. В першому - аргумент команди укомплектований кодом, що визначає число октетів у рядку. В другому - аргументи команди вимагають відповіді від сервера. В обох варіантах сервер надсилає запит на продовження команди, якщо він готов. Такий відклик сервера починається з символу "+".

Дані, передані сервером клієнту, а також статусні відгуки, які не вказують на завершення виконання команди, мають префікс "*" і називаються невідомими відгуками.

Існує три види відкликання завершення сервера: добре (показує успішне виконання), ні (відзначає невдала) або погане (вказує на протокольні помилки, наприклад, не визнається команда або зафіксована синтаксична помилка).

Команди клієнта.

Нижче описані команди IMAP 4.1. Команди розглядаються з урахуванням стану, в якому вони допустимі.

Наступні команди можуть використовуватися в будь-якому стані: CAPABILITY, NOOP та LOGOUT.

CAPABILITY Команда

Аргументи	відсутні	
Відповіді	необхідний немаркований відгук CAPABILITY	
Результат	OK	успішного завершення команди
	BAD	команда невідомо або невірний аргумент

Команда CAPABILITY запрошує перелік можливостей, підтримуваних сервером.

Команда NOOP

Аргументи	відсутні	
Відповіді	ніякої спеціальної відповіді на цю команду не потрібно	
Результат	OK	команда успішно завершена
	BAD	команда невідомо або неверен аргумент.

Команда NOOP нічого не робить і завжди успішно завершується.

Команда LOGOUT

Аргументи	відсутні	
Відповіді	необхідний немаркований відгук BYE	
Результат	OK	преривання сесії завершено
	BAD	невідома команда або невірний аргумент

Команда LOGOUT повідомляє серверу про те, що клієнт перебуває в режимі з'єднання. Сервер повинен послати немаркований відклик BYE, перш ніж відправити маркований відгук OK, після чого завершити розрив з'єднання.

Команди клієнта в стані без аутентифікації.

В стані без автентифікації команди AUTHENTICATE або LOGIN організовує аутентифікацію та переводять систему в стан з аутентифікацією. Об аутентифікації в IMAP можна прочитати в документе RFC-1731. Команда AUTHENTICATE надає загальний механізм для цілої низки аутентифікаційних методів, серед яких команда LOGIN використовується для традиційного введення імені та пароля в текстовому вигляді.

Команда AUTHENTICATE

Аргументи	ім'я механізму аутентифікації	
Відповіді	може бути запрошена додаткова інформація	
Результат	OK	Аутентифікація завершена, здійснено перехід в стан аутентифікації виконано
	NO	Ошибка аутентифікації: непідтримуваний механізм аутентифікації, параметри аутентифікації відхилені
	BAD	Невідома команда або неправильний аргумент, механізм аутентифікації перервано

Команда AUTHENTICATE вказує сервер на механізм аутентифікації, як це описано в [IMAP-AUTH]. Якщо сервер підтримує запрошений механізм автентифікації, він виконує обмін згідно з аутентифікаційним протоколом і ідентифікує клієнта. Він також може узгодити додатковий механізм захисту для подальших протоколів взаємодії. Якщо запрошений механізм аутентифікації не підтримується, сервер повинен відмовити команду AUTHENTICATE шляхом посилки маркированого відклика HI.

Команда LOGIN

Аргументи	ім'я користувача, пароль	
Відповіді	команда не вимагає жодної спеціальної відповіді	
Результат	OK	вхід завершено, система в стані з аутентифікацією
	NO	Логін не пройшов: ім'я користувача або пароль відхилені
	BAD	команда невідомо або неверний аргумент

Команда LOGIN ідентифікує клієнт-сервер і передає пароль користувача відкритим текстом.

Команди клієнта в стані "аутентифікація здійснюється".

В стані "аутентифікацію здійснено" дозволені команди маніпуляції поштовими скриньками как об'єктами-атомами. Команди SELECT та EXAMINE реалізують вибір поштової скриньки та перехід в стан "вибрано".

Додано наступні команди: SELECT, EXAMINE, CREATE, DELETE, RENAME, SUBSCRIBE, UNSUBSCRIBE, LIST, LSUB, STATUS і APPEND. В додаток до стандартних команд (CAPABILITY, NOOP і LOGOUT), в стані "аутентифікацію виконано" допустимі наступні команди: SELECT, EXAMINE, CREATE, DELETE, RENAME, SUBSCRIBE, UNSUBSCRIBE, LIST, LSUB, STATUS і APPEND.

Команда SELECT

Аргументи	назва почтової скриньки
Відповіді	Необхідні немарковані відклики: Флаги, EXISTS, RECENT

	опціонні немарковані відгуки OK: UNSEEN, PERMANENTFLAGS	
Результат	OK	процедура вибору закінчена, система знаходиться в стані вибору
	NO	вибір невдалий: немає такого ящика, доступ до поштового ящика неможливий
	BAD	команда невідома або невірен аргумент

Команда SELECT здійснює вибір поштового ящика, так, щоб забезпечити доступ до повідомлень, що знаходяться там. Перед тим як прислати клієнту OK, сервер повинен надіслати клієнту наступні немарковані дані:

- Флаги - флаги, визначені для поштового ящика
- <n> EXISTS - число повідомлень у поштовому ящику
- <n> RECENT - кількість повідомлень з набором флагов \ Recent
- OK [UIDVALIDITY <n>] - унікальний ідентифікатор правильності

Команда EXAMINE

Аргументи	назва почтової скриньки	
Відповіді	Необхідні немарковані відклики: Флаги, EXISTS, RECENT	
	опціоні немарковані відгуки OK: UNSEEN, PERMANENTFLAGS	
Результат	OK	Переглянути закон, система в стані вибору
	NO	Перегляд не прошел, система в состоянии аутентификация выполнена; нет такого почтового ящика; доступ до поштового ящика неможливий
	BAD	Команда невідома або невірен аргумент

Команда EXAMINE ідентична команді SELECT і дає той же результат, проте, вибраний поштовий ящик ідентифікується "тільки для читання". Ніякі зміни постійного стану поштового ящика в цьому випадку не допускаються. Текст відміченого відповіді OK на команду EXAMINE повинен починатися з коду відповіді [READONLY].

Команда CREATE

Аргументи	назва почтової скриньки	
Відповіді	На цю команду не посилається ніяких специфічних відкликів	
Результат	OK	Команда виконана
	NO	команда не виконана: поштовий ящик з таким ім'ям не може бути створено
	BAD	команда невідомо або неверен аргумент

Команда CREATE створює поштовий ящик із заданим ім'ям. Відклик OK надсилається в разі, коли новий поштовий ящик із вказаним ім'ям створено. Створення INBOX або поштового ящика з ім'ям існуючого поштового ящика є помилкою. Люба помилка при спробі створення поштового ящика викликає позначений відклик NO.

Команда DELETE

Аргументи	назва почтової скриньки	
Відповіді	Команда не вимагає ніяких відкликів	
Результат	OK	команда завершена

	NO	помилка при виконанні команди: не вдається видалити ящик із цим іменем
	BAD	команда невідомо або неверен аргумент

Команда DELETE видаляє поштовий ящик із зазначеним ім'ям. При цьому присилається маркований відклик OK тільки в тому випадку, коли ящик знищено. Помилкою вважається спроба видалити INBOX або ящик с неіснуючим ім'ям.

Команда UNSUBSCRIBE

Аргументи	назва почтової скриньки	
Відповіді	Ця команда не вимагає ніяких специфічних відкликів	
Результат	OK	ліквідація підписів пройшла успішно
	NO	ліквідація підписки не пройшла: це неможливо для даного імені
	BAD	команда невідомо або неверен аргумент

Команда UNSUBSCRIBE видаляє специфічний поштовий ящик із списку активних або підписних поштових ящиків даного сервера, як це визначається командою LIST. Ця команда повертає маркований відгук OK тільки в тому випадку, якщо ліквідація підписки пройшла успішно.

Команда LIST

Аргументи	ім'я	
	назва поштового ящика може містити символи підмени (підстановка)	
Відповіді	немарковані відгуки СПИСОК	
Результат	OK	команда LIST виконана
	NO	команда не пройшла: не можливо виконати LIST для даної моделі або імені
	BAD	команда невідома або невірен аргумент

Команда LIST повертає субнабір імен з повного набору, доступного клієнта. Приймається нуль або більше немаркованих відкликів LIST, що містять атрибути імен, ієрархічні розділювачі та імена.

ВИКОНАННЯ

Доменне им'я серверу: mail.ru

Відповідь сервера на запит HELO:

250-smtp8.mail.ru Привіт sisko24e37e878 [83.69.130.8]

E-mail client	**E-mail server**
Mail from: довільно	250 Ok
Rcpt to: довільно	250 Accepted
Data	354 Enter message, ending with "." On a line by itself

В результаті відправлення повідомлення реальний розмір файлу склал 354 КВ. Розмір файлу, отриманого повідомленням адресатом, виявився рівним 488 КВ. Таким чином, реальний розмір файлу складав 73% від розміру доставки повідомлення, що залишилася інформація представляє собою додаткові дані, отримані в результаті виконання кодування для надійного доставки даних. На

сьогоднішній день більшість поштових служб дозволяють передавати файли розміром до 20-25 МБ.

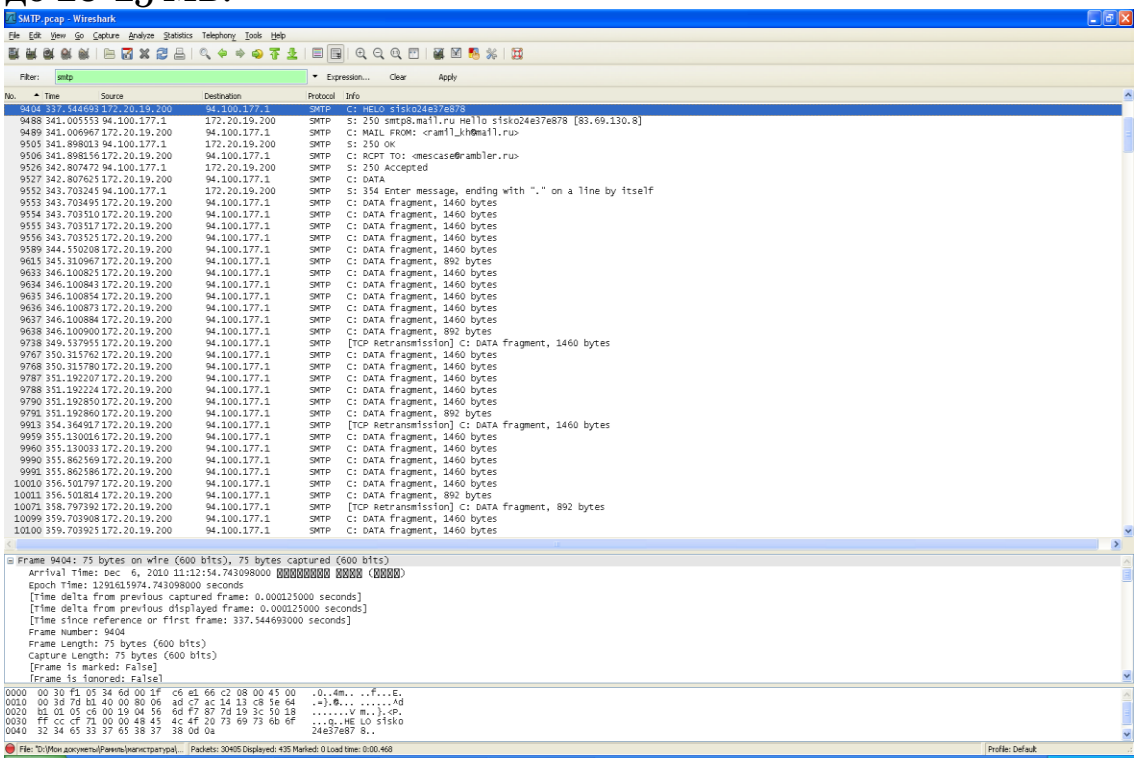


Рис.33 – Початок відправлення.

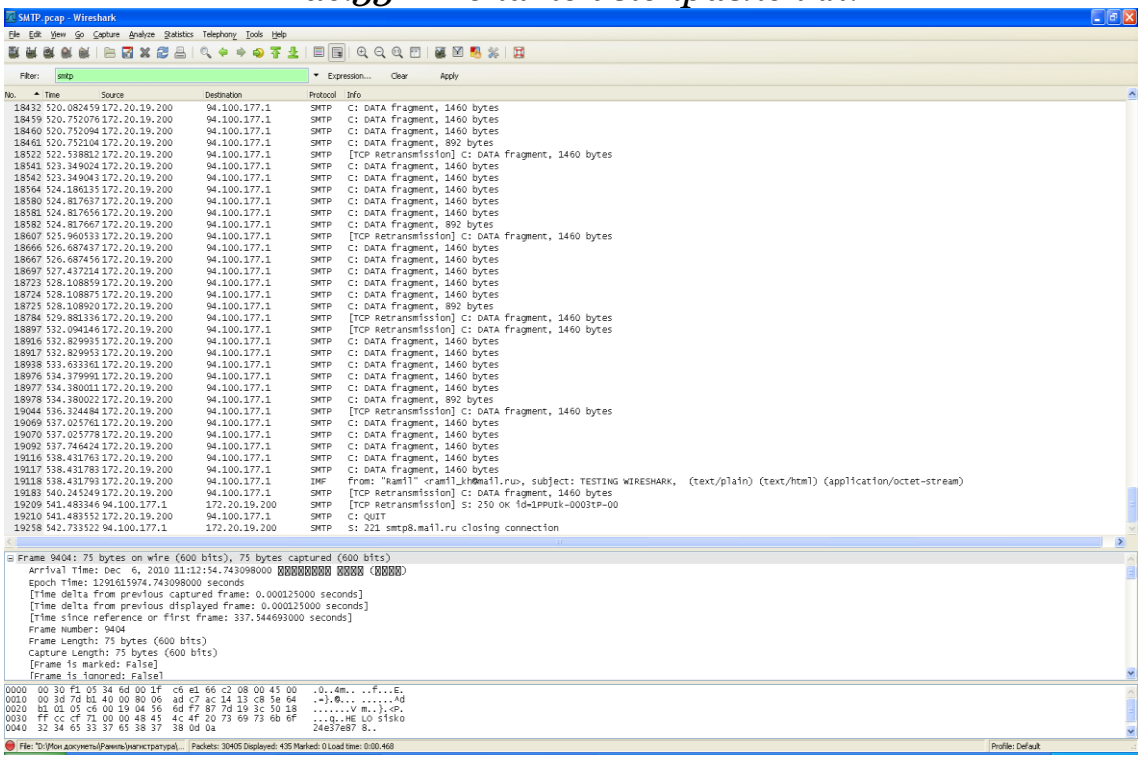


Рис.34 - Закінчення відправлення.

Зміст звіту:

1. Стислі теоретичні відомості;
2. Аналіз протоколів електронної пошти та послуг;
3. Протоколы SMTP, POP3, IMAP 4.1;
4. Результати відправки SMTP - пакетів.

Лабораторна робота №5.
Протоколи транспортного рівня TCP/IP, TCP та UDP.

Мета роботи. Провести аналіз роботи протоколів транспортного рівня TCP / IP (TCP, UDP) з використанням Netstat.

ХІД РОБОТИ

Теоретичні відомості.

Netstat - відображає активні підключення TCP, порти, прослухованими комп'ютером, статистики Ethernet, таблиці IP маршрутизації, протоколи IPv4 (для протоколів IP, ICMP, TCP і UDP) і IPv6 (для протоколів IPv6, ICMPv6, TCP через IPv6 та UDP через IPv6). Запущена без параметрів, команда nbtstat відібражає підключення TCP.

Синтаксис:

netstat [-Aan] [-f семейство_адресів] [-I інтерфейс] [-p ім'я_протоколу] [система] [core]

netstat [-n] [-s] [-i | -r] [-f семейство_адресів] [-I інтерфейс] [-p ім'я_протокола] [система] [core]

netstat [-n] [-I інтерфейс] інтервал [система] [core]

Перша форма команди показує список активних сокетів (sockets) для кожного протоколу. Друга форма вибирає одну з декількох інших мережевих структур даних. Третя форма показує динамічну статистику пересилання пакетів по сконфігурованим мережевим інтерфейсам; аргумент інтервал задає, скільки секунд збирається інформація між послідовними показами.

Опції

-a	Показувати стан усіх гнізд; звичайні сокети, що використовуются серверними процесами, які не показуються.
-A	Показувати адреси будь-якіх керуючих блоків протоколу, пов'язаних із сокетами; використовується для налагодження.
-i	Показувати стан автоматично сконфігурованих (автоконфігурірованих) інтерфейсів. Інтерфейси, статично зконфігуровані в системі, але НЕ знайдені під час завантаження, що не показуються.
-n	Показувати мережеві адреси як числа. netstat звичайно показує адреси як символи. Цей варіант можна використовуват з будь-якім форматом показу.
-r	Показати таблиці маршрутизації. При вікорістанні з опцією -s, показує статистику маршрутизації.
-s	Показати статистичну інформацію по протоколам. При використанні з опцією -r, показує статистику маршрутизації.
-f семейство_адресів	Обмежити показ статистики або адрес керуючих блоків тільки зазначеним семейством_адресов, в якості якого можна вказувати: inet Для сімейства адрес AF_INET, або unix Для сімейства адрес AF_UNIX.
-I інтерфейс	Виділити інформацію про зазначений інтерфейсі в

	окремий стовпець; за замовчуванням (для третьої форми команди) використовується інтерфейс з найбільшим обсягом переданої інформації з моменту останньої перезавантаження системи. Як інтерфейс можна вказувати будь-який з інтерфейсів, перерахованих у файлі конфігурації системи, наприклад, emd1 або loo.
-p ім'я_протоколу	Обмежити показ статистики або адрес керуючих блоків тільки протоколом до зазначеної іменем_протокола, наприклад, tcp.

Активні сокети.

Для кожного активного сокета показується протокол, розмір черг прийому і отримання (в байтах), локальна і віддалена адреса, а також внутрішній стан протоколу. Символьний формат, що часто використовується для показу адрес сокетов, - це або:

імя_хоста.порт

якщо ім'я хоста вказано, або:

мережа.порт

якщо адреса сокета задає мережу, але не конкретний хост. Імена хостів і мереж беруться з відповідних записів в файлі / etc / hosts або / etc / networks.

Якщо ім'я мережі або хоста для адреси невідомо (або якщо вказана опція -n), адреса показується числами. Чи не зазначені або «узагальнені» адреси та порти показуються як «*».

Сокети TCP

Для сокетів TCP припустимі наступні значення стану:

CLOSED	Закрито. Сокет не використовується.
LISTEN	Чекає вхідних з'єднань.
SYN_SENT	Активно починає з'єднуватися.
SYN_RECEIVED	Йде початкова синхронізація з'єднання.
ESTABLISHED	З'єднання встановлено.
CLOSE_WAIT	Дистанційна сторона відключилася; очікування закриття сокета.
FIN_WAIT_1	Сокет закритий; відключення з'єднання.
CLOSING	Сокет закритий, потім віддалена сторона відключилася; очікування підтвердження.
LAST_ACK	Дистанційна сторона відключилася, потім сокет закритий; очікування підтвердження.
FIN_WAIT_2	Сокет закритий; очікування відключення віддаленої сторони.
TIME_WAIT	Сокет закритий, але очікує пакети, ще знаходяться в мережі для обробки

Показ таблиці маршрутизації.

Таблиця маршрутизації показує всі наявні маршрути (routes) і статус кожного з них. Кожен маршрут складається з цільового хоста або мережі і шлюзу (gateway), який використовується для пересилання пакетів. Стовпець flags (прапори) показує

статус маршруту (U, якщо він включений), веде маршрут на шлюз (G), чи був маршрут створений динамічно за допомогою перенаправлення (D) і використовується чи адреса індивідуального хоста (H) замість адреси мережі. Наприклад, інтерфейс закільцьовування (loopback transport provider), loo, завжди має прапор H.

Прямі маршрути створюються для кожного інтерфейсу, підключеного до локального хосту; поле gateway (шлюз) для таких записів показує адресу вихідного інтерфейсу.

Стовпець refcnt показує поточну кількість активних використань для маршруту. Протоколи, орієнтовані на з'єднання, зазвичай використовують в ході з'єднання один маршрут, тоді як протоколи без з'єднання отримують маршрут для кожної посилки одному і тому ж адресату.

Стовпець use показує кількість пакетів, посланих по маршруту.

Стовпець interface показує мережевий інтерфейс, використовуваний маршрутом.

Сумарна статистика передачі даних.

Коли заданий аргумент інтервалу, netstat показує таблицю сумарної статистичної інформації про переданих пакетах, помилки і колізіях. Перша показується як рядок даних, а також кожна наступний 24-й рядок містить сумарну статистичну інформацію з моменту останньої перезавантаження системи. Кожний наступний рядок показує дані, накопичені за черговий вказаний в командному рядку інтервал з моменту останнього показу.

ВИКОНАННЯ.

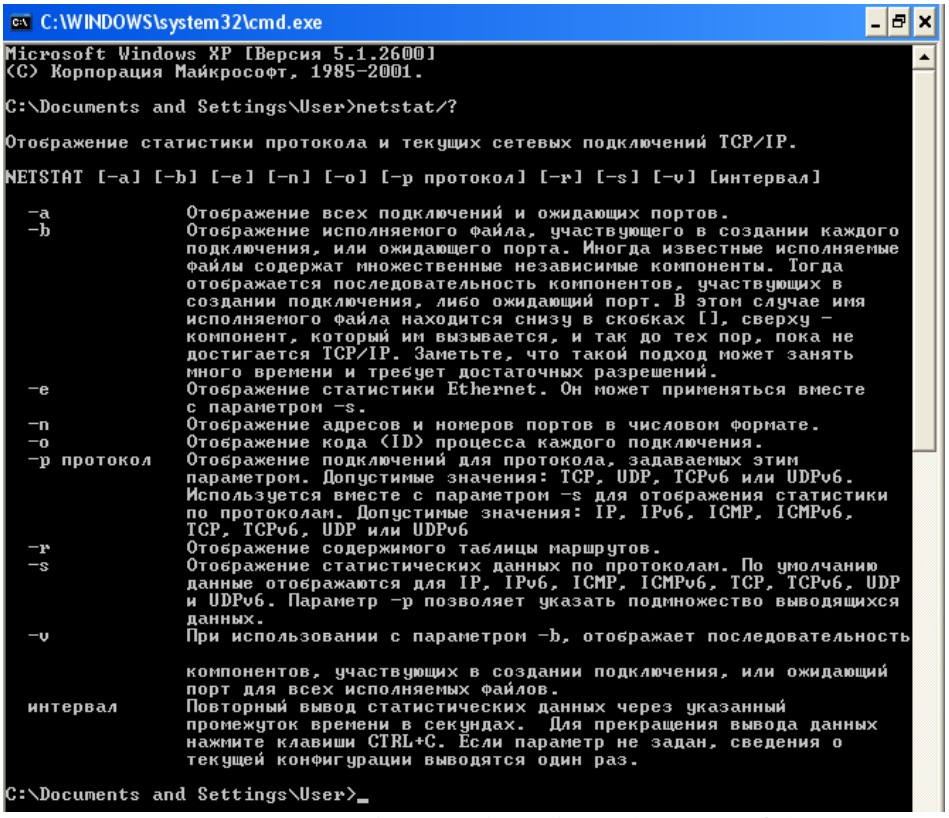


Рис.35 - Виведення довідки (netstat/?).

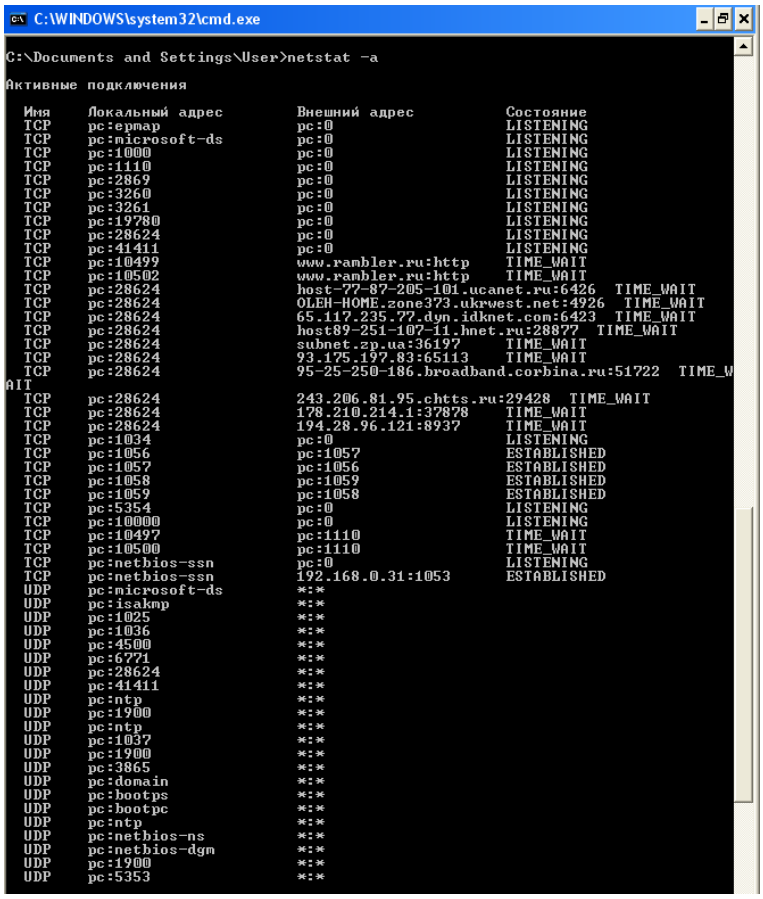


Рис. 36 - Використання Netstat для перегляду наявних з'єднань (netstat -a).

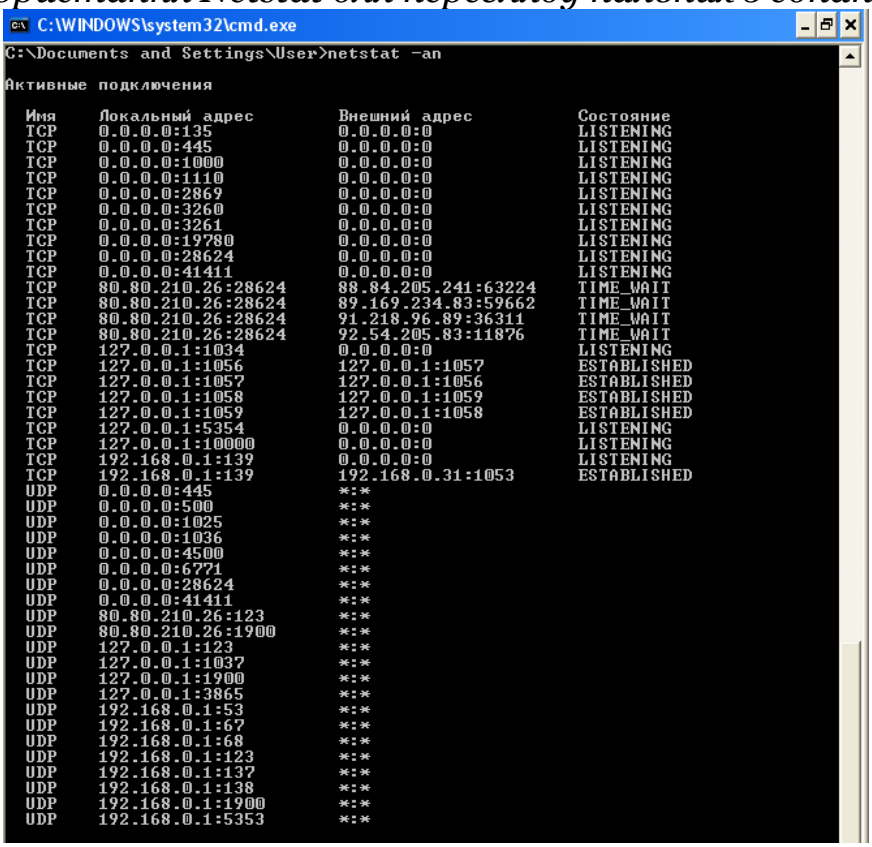


Рис.37 - Вивід з'єднань в числовому форматі (netstat -an).

В процесі виконання команд Netstat комп'ютер клієнта мав вихід в Інтернет. При цьому виконувався запит на сайт www.rambler.ru через браузер Mozilla Firefox. Через певні інтервали часу комп'ютер знаходився в режимі очікування для

встановлення з'єднання (стан TIME_WAIT). Після успішного виконання з'єднання на екран виводилося стан ESTABLISHED.

З'єднання	**Ім'я**	**Локальна адреса**	**Зовнішня адреса**	**Стан**
1.	TCP	pc: 10499	www.rambler.ru: http	TIME_WAIT
	TCP	80.80.210.26:28624	88.84.205.241: 63224	TIME_WAIT
2.	TCP	pc: 10502	www.rambler.ru: http	TIME_WAIT
	TCP	80.80.210.26:28624	89.169.234.83: 59662	TIME_WAIT
3.	TCP	pc: 28624	Host-77-81-205-101.ucanet.ru: 6426	TIME_WAIT
	TCP	80.80.210.26:28624	91.218.96.89: 36311	TIME_WAIT
4.	UDP	pc: ntp	*:*	
	UDP	80.80.210.26:123	*:*	
5.	UDP	pc:1037	*:*	
	UDP	80.80.210.26:1900	*:*	
6.	UDP	pc: 1900	*:*	
	UDP	127.0.0.1:123	*:*	

Зміст звіту:

1. Короткі теоретичні відомості;
2. Основні команди утиліти Netstat;
3. Основні стани, що виводяться утилітою Netstat;
4. Результати аналізу виконання запитів до мережі Інтернет.

Лабораторна робота №6.
Використання Wireshark для аналізу Protocol Data Units.

Мета роботи. Провести аналіз Protocol Data Units з використанням Wireshark.

ХІД РОБОТИ

Короткі теоретичні відомості.

ICMP – протокол.

Протокол управління повідомленнями Інтернету (ICMP - Internet Control Message Protocol) - супутник протоколу IP. ICMP - протокол мережевого рівня. Однак він сам не передає безпосередньо повідомлення даних канального рівня. Ці повідомлення першими інкапсулюються в дейтаграми IP перед переходом до більш низького рівня (рис.38).

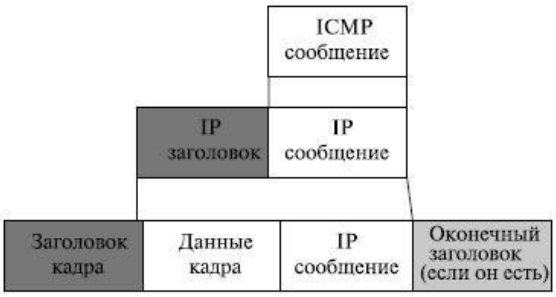


Рис. 38 - Інкапсуляція повідомлення ICMP.

Значення поля протоколу в дейтаграми IP - це "1", воно вказує, що дані IP - це ICMP-повідомлення.

Типи повідомлень ICMP.

ICMP-повідомлення розділені на дві широкі категорії: звіт про помилку повідомлення і запит, як це показано на рис.2.

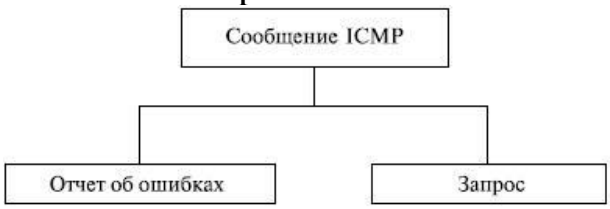


Рис. 39 - Повідомлення ICMP.

Повідомлення про помилку переносить дані про проблеми, що виникають при обміні повідомленнями, з якими маршрутизатор або хост (пункт призначення) можуть зіткнутися, коли вони обробляють пакет IP.

Повідомлення запиту допомагають хосту або мережевому менеджеру отримати задану інформацію від маршрутизатора або іншого хоста. Наприклад, вузли можуть виявити їх сусідів. Також хости можуть виявити і дізнатися про маршрутизаторах на їх мережі, і маршрутизатори можуть допомогти вузлу переадресовувати його повідомлення. В таблиці перераховується ICMP-повідомлення в кожній категорії.

ICMP повідомлення		
Категорія	**Тип**	**Повідомлення**
	3	кінцевий пункт не досяжний
	4	придушення джерела
	11	час минув
	12	проблеми параметрів
	5	перепризначення
	8 или 0	відлуння запит і відповідь
	13 или 14	мітка часу запит і відповідь
	17 или 18	маска адреси запит і відповідь
	10 или 9	маршрутизатор затребування і повідомлення

Формат повідомлення ICMP.

ICMP-повідомлення має 8-байтовий заголовок і розділ даних змінного розміру. Хоча загальний формат заголовка різний для кожного типу повідомлення, перші 4 байта - загальні для всіх. Як показує рис. 40., перше поле, ICMP, визначає тип повідомлення. Поле коду визначає підставу для конкретного типу повідомлення. Останнє загальне поле - це поле контрольної суми. Інша частина заголовка задана для кожного типу повідомлення.

Рис. 40 - Основний формат ICMP.

Розділ даних в повідомленнях про помилки доставляє інформацію для знаходження початкового пакету, який містить помилку. У повідомленнях запиту розділ даних доставляє додаткову інформацію, засновану на типі запиту.

Повідомлення помилки ICMP.

Один з головних обов'язків ICMP полягає в тому, щоб сповістити про помилки. Хоча технології передачі сьогодні надають для передачі все більш і більш достовірні середовища, помилки все ще існують і повинні бути оброблені. IP, як обговорювалося раніше, є ненадійним протоколом. ICMP був частково призначений для компенсування цього недоліку. Однак ICMP не виправляє помилки, він просто сповіщає про них. Виправлення помилки залишають протоколам високого рівня. Повідомлення про помилки завжди посилають первісного джерела, тому що єдина інформація, доступна в дейтаграми про маршрут - адреси IP пункту призначення і джерело. ICMP використовує вихідний адресу IP, щоб послати повідомлення про помилки джерела дейтаграми. Обробляються п'ять типів помилок (рис. 41):

-пункт призначення недосяжний;

-придушення джерела;

-час минув;

-проблеми параметра;

-перепризначення.

Рис. 41 - Типи повідомлень звіту про помилки.

Всі повідомлення про помилки містять розділ даних (рис. 42), який включає заголовок IP початкової дейтаграми плюс перші 8 байт даних в цій дейтаграми. Початковий заголовок дейтаграми додається, щоб дати первісного джерела, який отримує повідомлення про помилки, інформацію безпосередньо про саму дейтаграми. Включені 8 байт даних, тому що, згідно з форматам UDP- і TCP-протоколів, перші 8 байт забезпечують інформацію про номери порту (UDP і TCP) і порядковий номер (TCP). Ця інформація необхідна, щоб джерело могло повідомити протоколам (TCP або UDP) про помилку. ICMP формує пакет даних про помилку, який потім інкапсулюється в дейтаграмму IP (див. (Рис. 42).

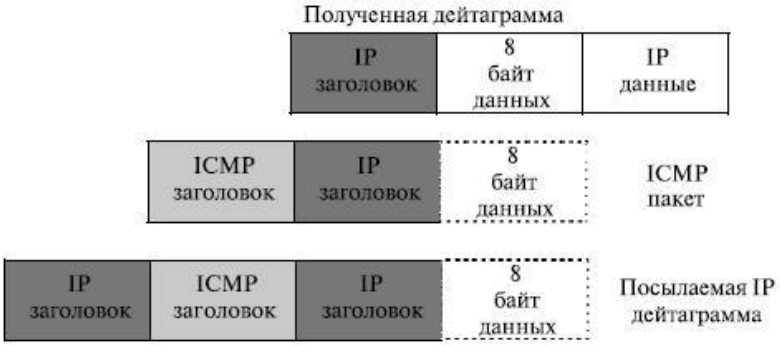


Рис. 42- Зміст поля даних для повідомлення про помилки.

FTP – протокол.

FTP являє найпростіший спосіб обміну файлами між віддаленими комп'ютерами і використовується для завантаження або відвантаження файлів на інший комп'ютер. Це стандартний протокол інтернету, оптимізований для передачі даних через мережі TCP / IP. Для надійної передачі даних FTP встановлює з'єднання клієнт / сервер з використанням двох портів TCP на комп'ютері-клієнті і сервері. Перший порт називається FTP Control і забезпечує початок сеансу і контроль помилок; другий порт називається FTP Data і забезпечує безпосередню передачу даних між клієнтом і сервером. Служба FTP використовує добре відомий порт 21 для операцій контролю і порт 20 для передачі даних. Клієнтські порти TCP присвоюються динамічно при створенні сеансу.

"Добре відомі" номери портів зарезервовані організацією ICANN (Internet Corporation for Assigned Names and Numbers) для використання кінцевими точками програми, які встановлюють зв'язок через протоколи TCP або UDP. Кожен тип програми має свій власний і, отже, "добре відомий" номер порту. Коли додаток на одному клієнті встановлює з'єднання з іншим клієнтом, воно вказує на конкретне застосування за допомогою номера порту. Відомі порти лежать в діапазоні від 0 до 1023, зареєстровані порти - з 1024 по 49151, а решту порти, аж до 65535, використовуються в якості динамічних або приватних портів. Ви можете змінити номер порту, який використовується будь-якою службою IIS, включаючи SMTP і

NNTP. Це робиться з міркувань безпеки в невеликій приватній мережі; проте більшість хакерів використовують сканери портів для знаходження відкритих портів.

Служба FTP дозволяє користувачам завантажувати з сайту файли, наприклад, керівництва по використанню товарів, співробітникам відділу продажів працювати із замовленнями або звітами, перебуваючи у від'їзді або в дорозі. Однак за ці можливості можна заплатити чималу ціну. Протокол FTP здійснює передачу всіх даних, включаючи імена користувачів і паролі, через мережу у відкритому вигляді, що відкриває доступ до локальної файлової системи будь-якому користувачеві в мережі, якщо не забезпечена належна захист (право на локальний вхід потрібно кожному користувачеві, підключається до FTP-сервера).

Протокол не шифрується, при аутентифікації передаються логін і пароль відкритим текстом. У разі побудови мережі з використанням хаба зловмисник за допомогою пасивного сніфферу може перехоплювати логіни і паролі знаходяться в тому ж сегменті мережі користувачів FTP, або, при наявності спеціального ПЗ, отримувати передані по FTP файли без авторизації. При побудові мережі на світчах завдання зловмисника ускладнюється, але злом все одно можливий (MAC-спуфинг, переповнення таблиці адрес). Щоб запобігти перехоплення трафіку, необхідно використовувати протокол шифрування даних SSL, який підтримується багатьма сучасними FTP-серверами і деякими FTP-клієнтами.

Процес нешифрованої авторизації проходить в кілька етапів (символи \ r \ n означають новий рядок):

Установка TCP-з'єднання з сервером (зазвичай на 21 порт)

Посилка команди USER логін \ r \ n

Посилка команди PASS пароль \ r \ n

Якщо до сервера дозволений анонімний доступ (як правило, лише для завантаження даних з сервера), то в якості логіна використовується ключове слово «anonymous» або «ftp», а в якості пароля - адреса електронної пошти:

USER anonymous \ r \ n

PASS someone @ email \ r \ n

Після успішної авторизації можна посилати на сервер інші команди.

Основні команди FTP:

ABOR - Перервати передачу файлу

CDUP - Змінити директорію на вищу.

CWD - Змінити директорію.

DELE - Видалити файл (DELE filename).

EPSV - Увійти в розширений пасивний режим. Застосовується замість PASV.

HELP - Виводить список команд прийнятих сервером.

LIST - Повертає список файлів директорії. Список передається через з'єднання даних (20 порт).

MDTM - Повертає час модифікації файлу.

MKD - Створити директорію.

MLST - Повертає список файлів директорії в більш короткому форматі ніж LIST. Список передається через з'єднання даних (20 порт).

NOOP - Порожня операція

PASV - Увійти в пасивний режим. Сервер поверне адресу і порт до якого потрібно підключитися щоб забрати дані. Передача почнеться при введенні наступних команд RETR, LIST і тд.

PORT - Увійти в активний режим. Наприклад PORT 12,34,45,56,78,89. На відміну від пасивного режиму для передачі даних сервер сам підключається до клієнта.

PWD - Повертає поточну директорію.

QUIT - Відключитися

REIN - Реініціалізіровать підключення

RETR — Скачати файл. Перед RETR повинна бути команда PASV або PORT.

RMD - Видалити директорію

RNFR і RNTO - Перейменувати файл. RNFR - що перейменовувати, RNTO - будь-що.

SIZE - Повертає розмір файлу

STOR - Завантажити файл. Перед STOR повинна бути команда PASV або PORT.

SYST - Повертає тип системи (UNIX, WIN, ...)

TYPE - Встановити тип передачі файлу (Бінарний, текстовий)

USER - Ім'я користувача для входу на сервер

ВИКОНАННЯ

Виконання команди ping 192.168.251.01 с локального комп'ютера, що має IP-адресу 192.168.N.M, призводить до обміну пакетами за протоколом ICMP.

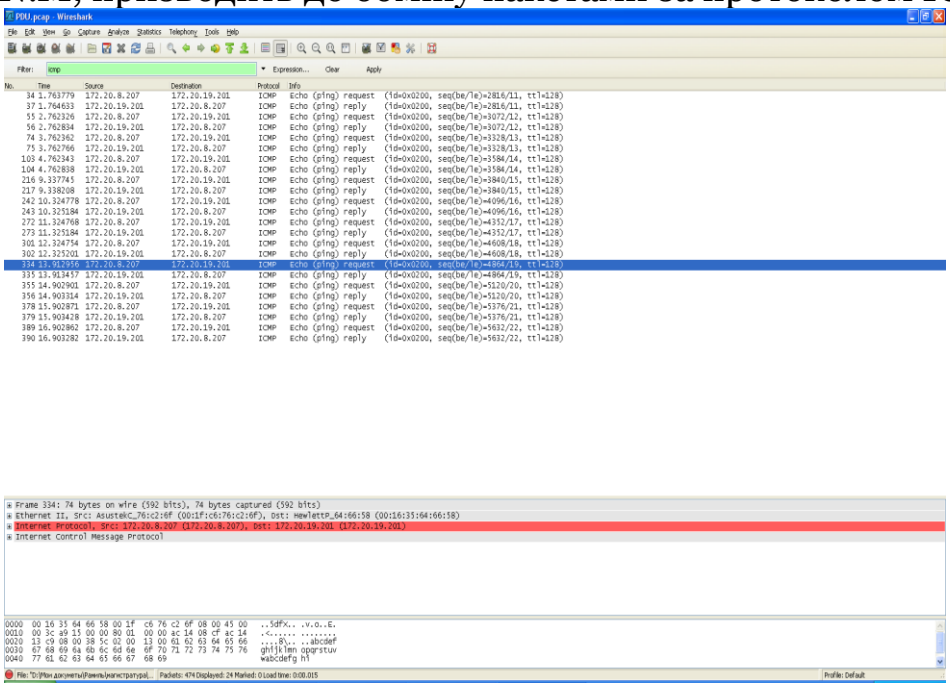


Рис.43 - Список переданих пакетів ICMP.

На рис.43 наведено приклад виконання завантаження файлів з FTP-сервера з використанням команди get.

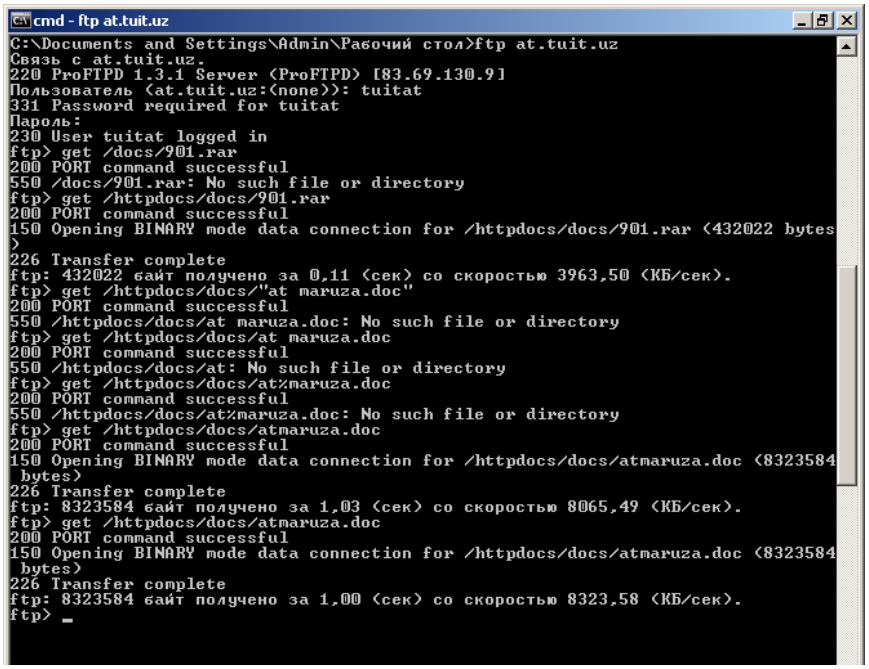


Рис.44 - Виконання запитів до FTP-сервера через командний рядок Windows.

На рис.45 показаний обмін пакетами при виконанні аутентифікації користувача на FTP-сервері. Логін і пароль передаються у відкритому вигляді, тобто не мають ніякої криптографічного захисту.

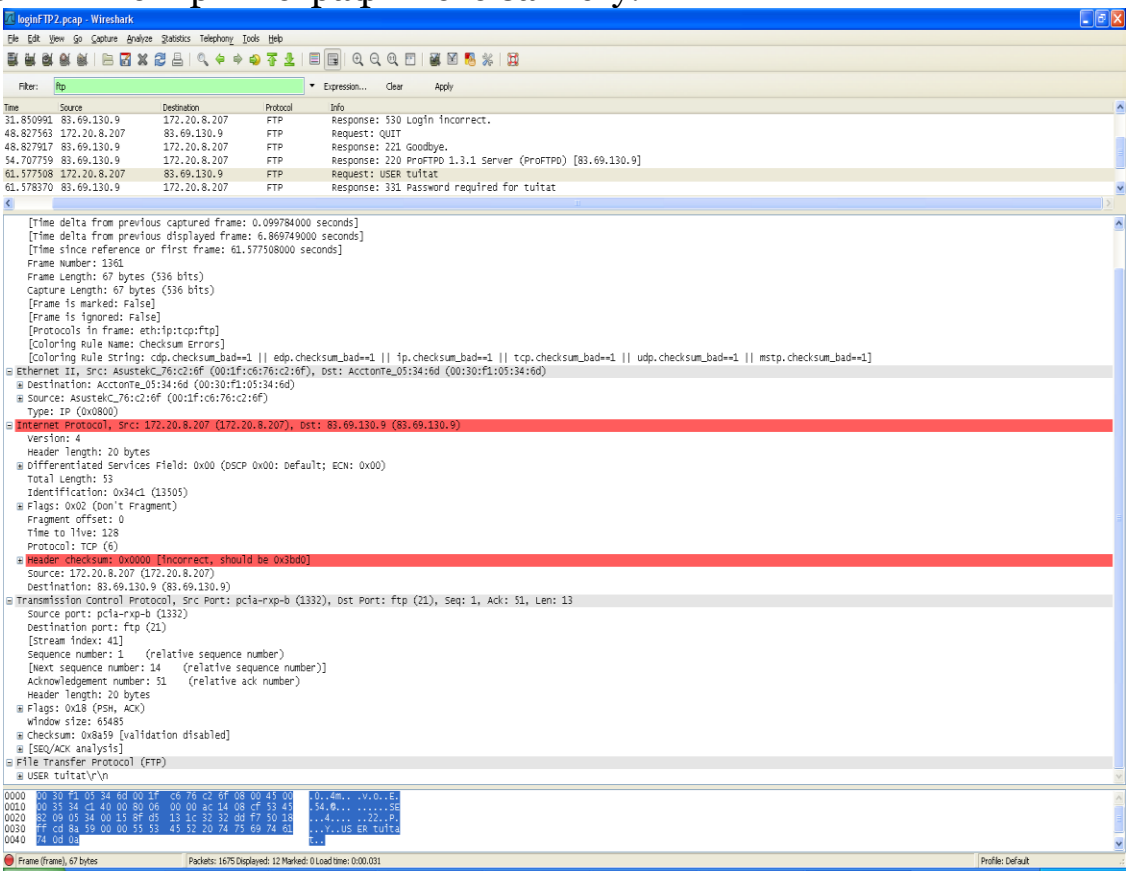


Рис.45 - Переданные FTP-пакеты при аутентификации

На рис. 46. показана передача пакетів при виконанні команди get виконання завантаження файлу atmaruza.doc з FTP-сервера.

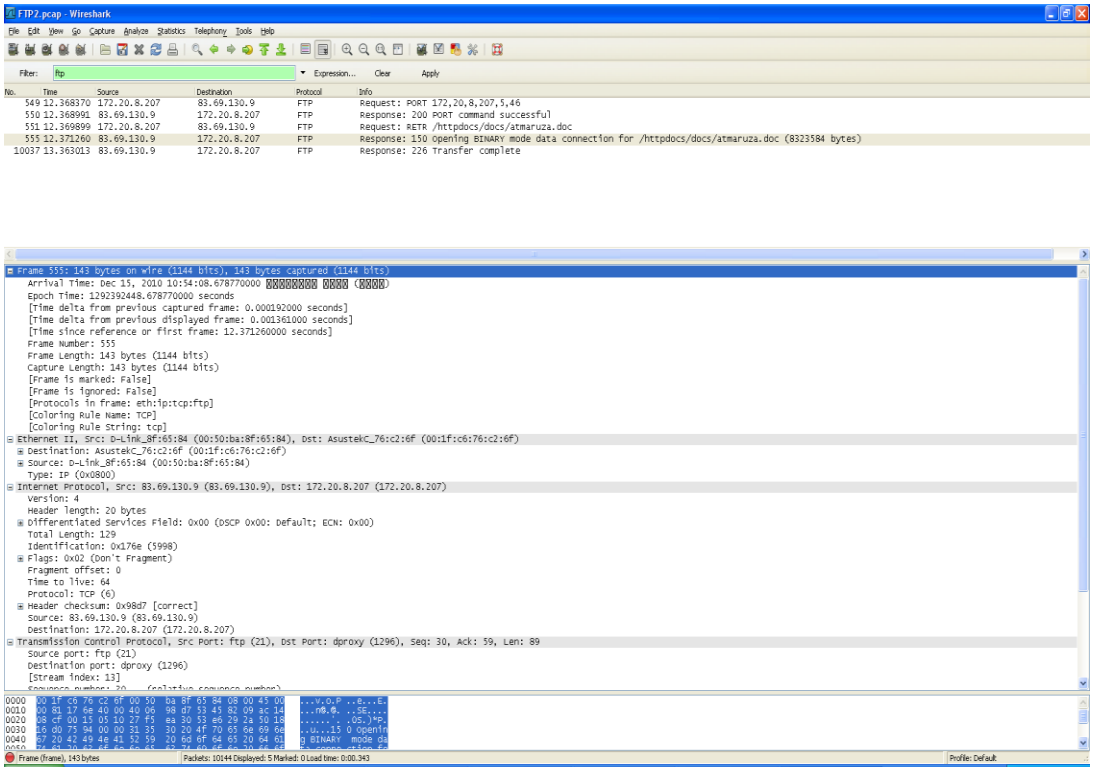


Рис.46 - Передані пакети при виконанні команди get.

На рис.47. показаний обмін пакетами інформації FTP-DATA, що завантажується з FTP-сервера.

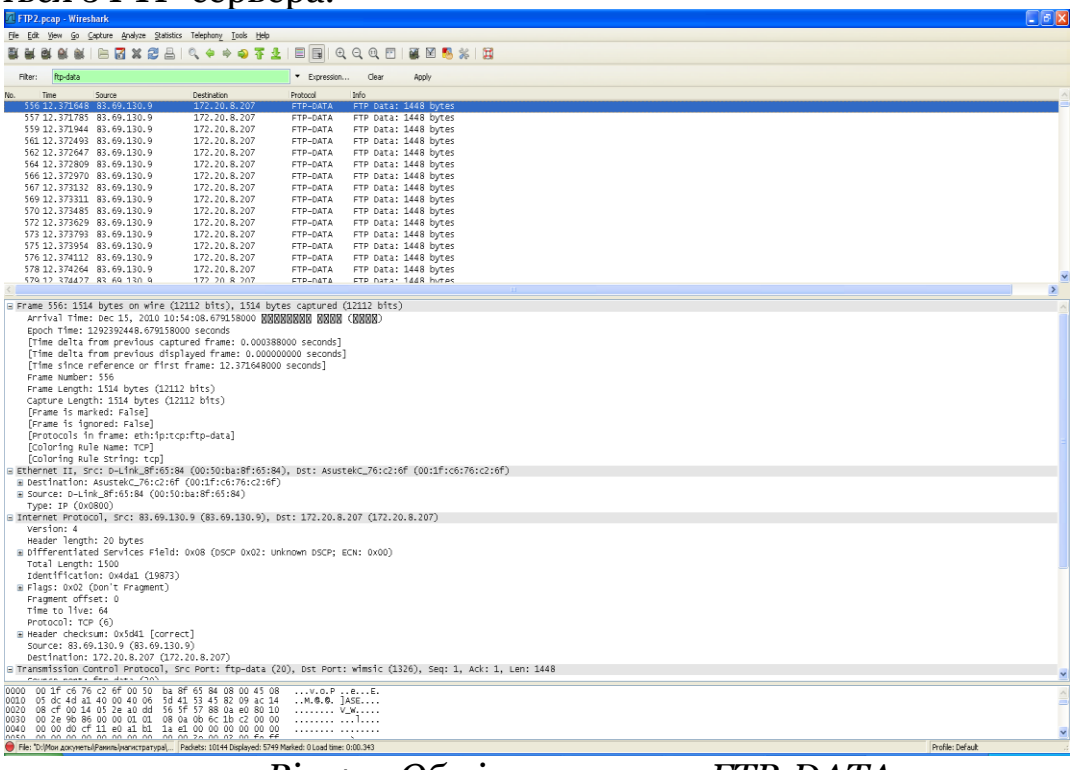


Рис.47 - Обмін пакетами FTP-DATA.

Зміст звіту:

1. Короткий теоретичні відомості;
2. Інформація про обмін ping повідомленнями між комп'ютерами в локальній мережі;
3. Інформація про аутентифікації і скачування даних з FTP-сервера.

Лабораторна робота №7.
Аналіз Gateway обладнання.

Мета роботи. Провести аналіз роботи Gateway обладнання (адреси шлюзів, конфігурація мережевої інформації)

ХІД РОБОТИ

Короткі теоретичні відомості.

Мережевий шлюз (англ. Gateway) - апаратний маршрутизатор або програмне забезпечення для сполучення комп'ютерних мереж, що використовують різні протоколи мережі (наприклад, локальної і глобальної).

Мережевий шлюз конвертує протоколи одного типу фізичного середовища в протоколи іншої фізичної середовища (мережі). Наприклад, при з'єднанні локального комп'ютера з мережею Інтернет ви використовуєте мережевий шлюз.

Роутери (маршрутизатори) є одним із прикладів апаратних мережевих шлюзів.

Мережеві шлюзи працюють на всіх відомих операційних системах. Основне завдання мережевого шлюзу - конвертувати протокол між мережами. Роутер сам по собі приймає, проводить і відправляє пакети тільки серед мереж, що використовують однакові протоколи. Мережевий шлюз може з одного боку взяти пакет, сформатував під один протокол (наприклад Apple Talk) і конвертувати в пакет іншого протоколу (наприклад TCP / IP) перед відправкою в інший сегмент мережі. Мережеві шлюзи можуть бути апаратним рішенням, програмним забезпеченням або тим і іншим разом, але зазвичай це програмне забезпечення, встановлене на роутер або комп'ютер. Мережевий шлюз повинен розуміти всі протоколи, які використовуються роутером. Зазвичай мережеві шлюзи працюють повільніше, ніж мережеві мости, комутатори і звичайні роутери. Мережевий шлюз - це точка мережі, яка служить виходом в іншу мережу. В мережі Інтернет вузлом або кінцевою точкою може бути або мережевий шлюз, або хост. Інтернет-користувачі і комп'ютери, які доставляють веб-сторінки користувачам - це хости, а вузли між різними мережами - це мережеві шлюзи. Наприклад, сервер, який контролює трафік між локальною мережею компанії і мережею Інтернет - це мережевий шлюз. У великих мережах сервер, що працює як мережевий шлюз, зазвичай інтегрований з проксі-сервером і фаєрволом. Мережевий шлюз часто об'єднаний з роутером, який керує розподілом і конвертацією пакетів в мережі.

Мережевий шлюз може бути спеціальним апаратним роутером або програмним забезпеченням, встановленим на звичайний сервер або персональний комп'ютер. Більшість комп'ютерних операційних систем використовує терміни, описані вище. Комп'ютери під Windows зазвичай використовують вбудований майстер підключення до мережі, який за вказаними параметрами сам встановлює з'єднання з локальною або глобальною мережею. Такі системи можуть також використовувати DHCP-протокол. Dynamic Host Configuration Protocol (DHCP) - це протокол, який зазвичай використовується мережевим обладнанням щоб отримати різні дані, необхідні клієнту для роботи з протоколом IP. З використанням цього протоколу додавання нових пристроїв і мереж стає простим і практично автоматичним.

ВИКОНАННЯ

Виконання команди «netstat -r» для отримання інформації про маршрути. IP-адреса шлюзу 172.20.19.160 (192.168.251.04), MAC - адреса мережевого адаптера шлюзу AsustekC_e1: 66: b8 (00: 1f: c6: e1: 66: b8). IP-адреса клієнта 172.20.19.160(192.168.251.6), MAC-адресу мережевого адаптера клієнта AcconTe_05: 34: 6d (00: 30: f1: 05: 34: 6d).

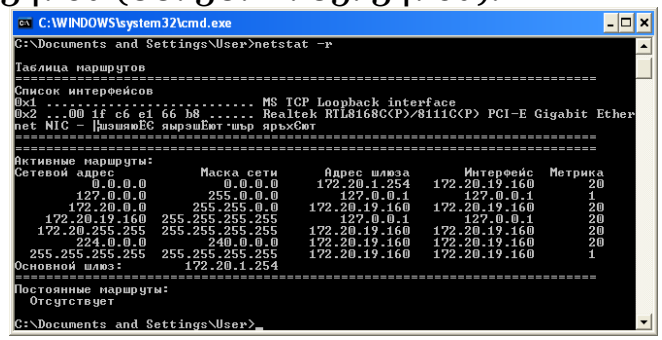


Рис.48 - Отримання інформації про шлюзи.

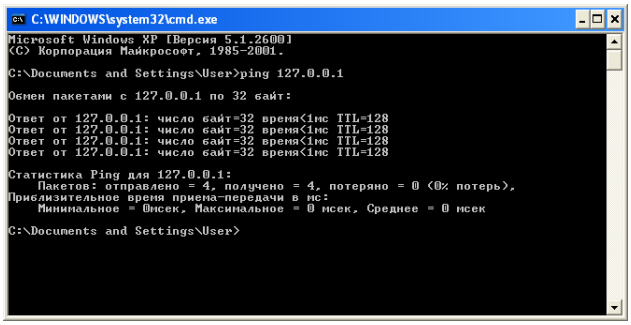


Рис.49 - Виконання ping для localhost (127.0.0.1)

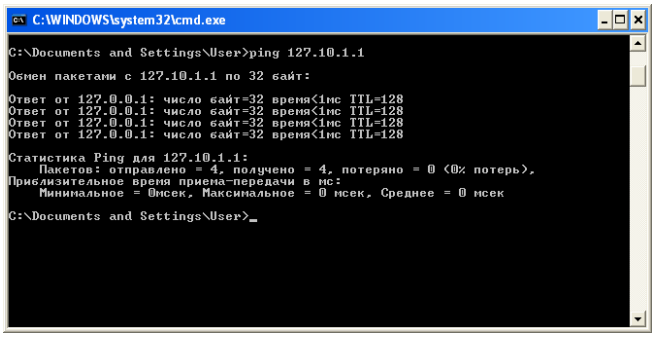


Рис.50 - Виконання команди ping 127.10.1.1

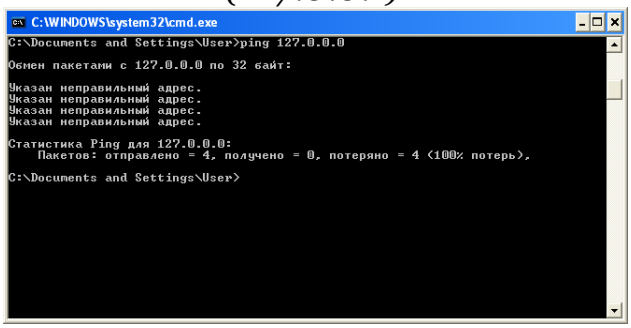


Рис. 51 - Виконання команди ping для підмережі 127.0.0.0

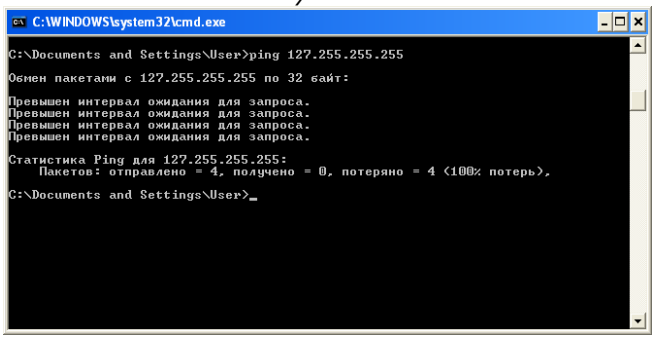


Рис.52 - Виконання команди ping 127.255.255.255

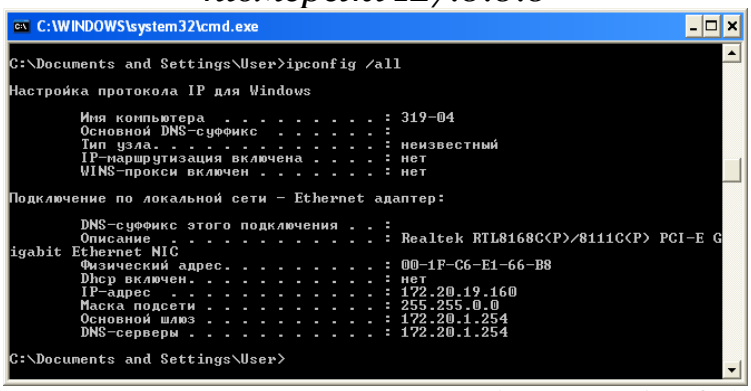


Рис.53 - Выполнение команды ipconfig /all

Таблиця. Інформація про локальний комп'ютер

Найменування	**Адреса**
IP-адреса	172.20.19.160
Маска підмережі	255.255.0.0
Основний шлюз	172.20.1.254
DNS сервер	172.20.1.254

Зміст звіту:

1. Короткий теоретичні відомості;
2. Налагодження та перевірка Gateway.

Лабораторна робота №8.
Налаштування комутаторів.

Мета роботи. Провести настройку мережевого комутатора з використанням Cisco Packet Tracer.

ХІД РОБОТИ

Короткі теоретичні відомості.

Комутуючі концентратори (Switched Hubs) або, як їх ще називають, комутатори (Switches), перемикачі та свічі, можуть розглядатися, як найпростіший і дуже швидкий міст. Вони дозволяють розділити єдину мережу на кілька сегментів для збільшення допустимого розміру мережі або з метою зниження навантаження (трафіка) в окремих частинах мережі.

Як уже зазначалося, на відміну від мостів, комутуючі концентратори не приймають пакети які приходять, а тільки переправляють з однієї частини мережі в іншу ті пакети, які цього потребують. Вони в реальному темпі надходження бітів пакета розпізнають адресу приймача пакета і приймають рішення про те, чи треба цей пакет переправляти, і, якщо треба, то кому. Ніякої обробки пакетів не проводиться, хоча і контролюється їх заголовок. Комутатори практично не сповільнюють обміну по мережі. Але вони не можуть перетворювати формат пакетів і протоколів обміну по мережі. Оскільки комутатори працюють з інформацією, що знаходиться всередині кадру, часто говорять, що вони ретранслюють кадри, а не пакети, як репітерні концентратори.

Колізії комутатор ретранслюються, що вигідно відрізняє його від більш простого репітерного концентратора. Можна сказати, що комутатори виробляють більш глибокий поділ мережі, ніж концентратори. Вони поділяють на частини зону колізій (Collision Domain) мережі, тобто область мережі, на яку поширюються колізії. Логічна структура комутатора досить проста (рис. 54).

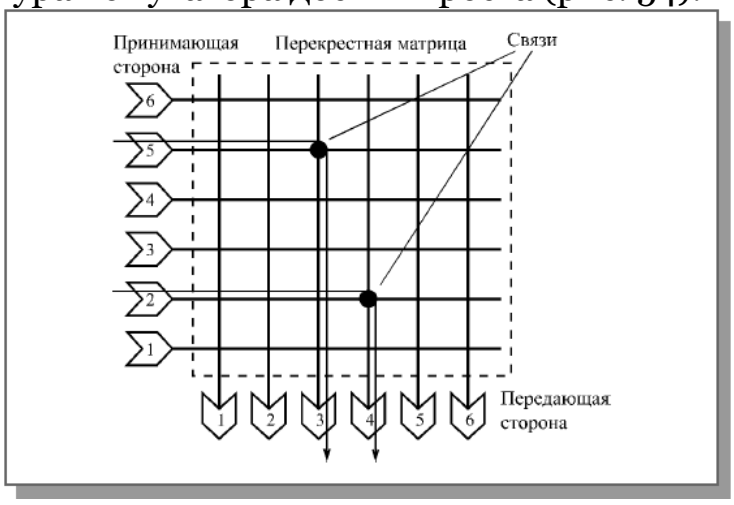


Рис. 54 - Логічна схема комутатора.

Вона включає в себе так звану перехресну (комутаційну) матрицю (Crossbar Matrix), у всіх точках перетину якої можуть встановлюватися зв'язки на час передачі пакета. В результаті пакет, що надходить з будь-якого сегмента, може бути переданий в будь-який інший сегмент. У разі широковещательного пакета, адресованого всім абонентам, він передається в усі сегменти одночасно, крім того сегмента, за яким він прийшов (рис. 55).

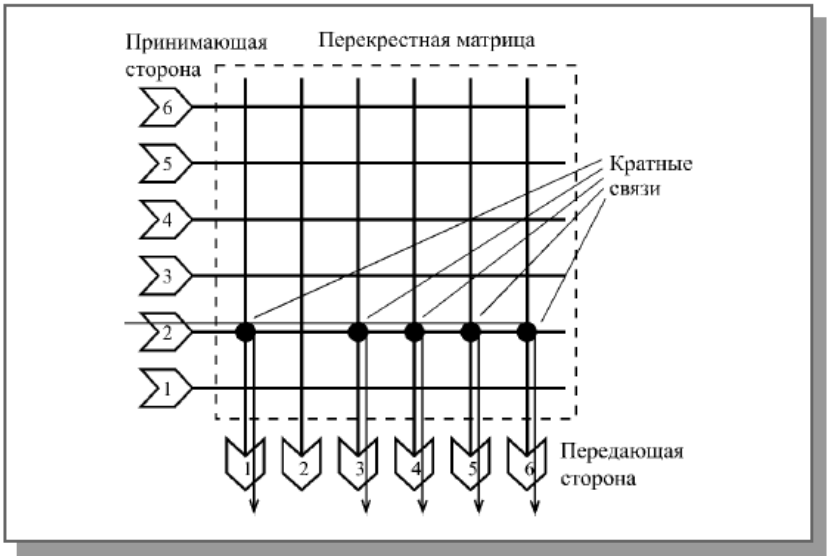


Рис. 55 - Ретрансляція широковещательного пакета

Крім перехресної матриці комутатор включає в себе пам'ять, в якій він формує таблицю MAC-адрес всіх комп'ютерів, підключених до кожного з його портів. Ця таблиця створюється на етапі ініціалізації мережі і потім періодично оновлюється для обліку змін конфігурації мережі. Саме на підставі аналізу цієї таблиці робиться висновок про те, які зв'язки треба замикати, куди відправляти пакет, що прийшов. Комутатор читає MAC-адреси відправника і одержувача в пакеті, що прийшов і передає пакет в той сегмент, в який він адресований. Якщо пакет адресований абоненту з того ж сегмента, до якого належить відправник, то він не ретранслюється взагалі. Широкомовний пакет не передається в той сегмент, до якого приєднаний абонент відправник пакета. Адреса відправника пакета заноситься в таблицю адрес (якщо його там ще немає).

Комутатори випускаються на різну кількість портів. Найчастіше зустрічаються комутатори з 6, 8, 12, 16 і 24 портами. Слід зазначити, що мости, як правило, рідко підтримують більше 4 портів. Розрізняються комутатори з допустимою кількістю адрес на один порт. Цей показник визначає граничну складність підключаються до порту сегментів (кількість комп'ютерів в кожному сегменті). Деякі комутатори дозволяють розбивати порти на групи, що працюють незалежно один від одного, тобто один комутатор може працювати як два або три.

Так само, як і концентратори, комутатори випускаються трьох видів залежно від складності, можливості нарощування кількості портів і вартості:

- комутатори з фіксованою кількістю портів (зазвичай до 30);
- модульні комутатори (з числом портів до 100);
- стекові комутатори.

Комутатори характеризуються двома показниками продуктивності:

Максимальна швидкість ретрансляції пакетів виміряється при передачі пакетів з одного порту в інший, коли всі інші порти відключені.

Сукупна швидкість ретрансляції пакетів виміряється при активній роботі всіх наявних портів. Сукупна швидкість більше максимальної, але максимальна швидкість, як правило, не може бути забезпечена на всіх портах одночасно, хоча комутатори і здатні одночасно обробляти кілька пакетів (на відміну від моста).

Головне правило, якого треба дотримуватися при розбитті мережі на частини (сегменти) за допомогою комутатора, називається "правило 80/20". Тільки при

його виконанні комутатор працює ефективно. Згідно з цим правилом, необхідно, щоб не менше 80 відсотків усіх передач відбувалося в межах однієї частини (одного сегмента) мережі. І тільки 20 відсотків всіх передач повинно відбуватися між різними частинами (сегментами) мережі, проходити через комутатор. На практиці це зазвичай зводиться до того, щоб сервер і активно працюють з ним робочі станції (клієнти) розташовувалися на одному сегменті. Це ж правило 80/20 може бути застосовано і до мостів.

Існує два класи комутаторів, що відрізняються рівнем інтелекту і способами комутації:

-комутатори з наскрізним вирізанням (Cut-Through);

-комутатори з накопиченням і ретрансляцією (Store-and-Forward, SAF).

ВИКОНАННЯ

Побудована мережева топологія (Рис.56) містить наступне мережеве обладнання:

-маршрутизатор;

-комутатор;

-ПК;

-Прямий кабель;

-Перехресний кабель.

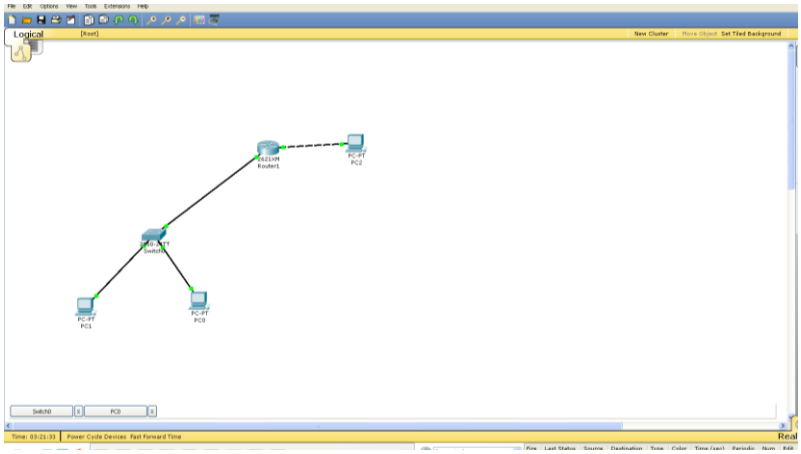


Рис.56 - Топологія мережі.

Табл. Інформація про налаштування обладнання.

Device	Subnet	IP address	Mask	Gateway
PC0	130.1.0.0	130.1.1.2	255.255.0.0	130.1.1.1
PC1	130.1.0.0	130.1.1.4	255.255.0.0	130.1.1.1
PC2	80.0.0.0	80.1.1.2	255.0.0.0	80.1.1.1
Router1 Fa 0/0	130.1.0.0	130.1.1.1	255.255.0.0	-
Router1 Fa 0/1	80.0.0.0	80.1.1.1	255.0.0.0	-
Switch0	130.1.0.0	130.1.1.5	255.255.0.0	130.1.1.1

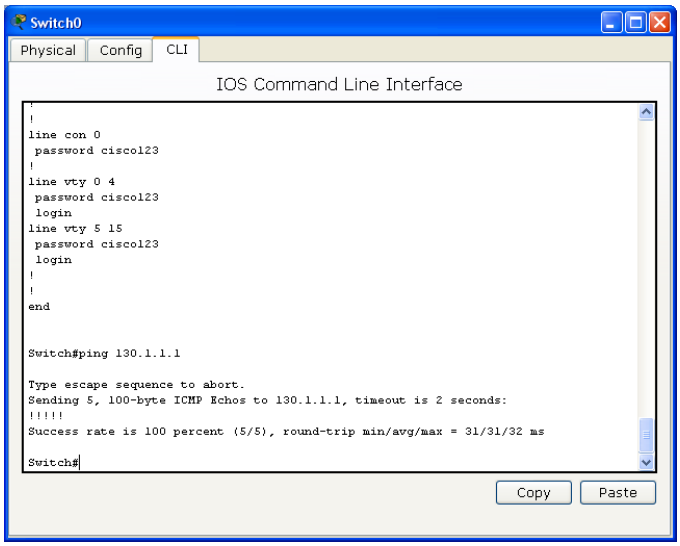


Рис.57 - Виконання ехо-запиту з комутатора на маршрутизатор.

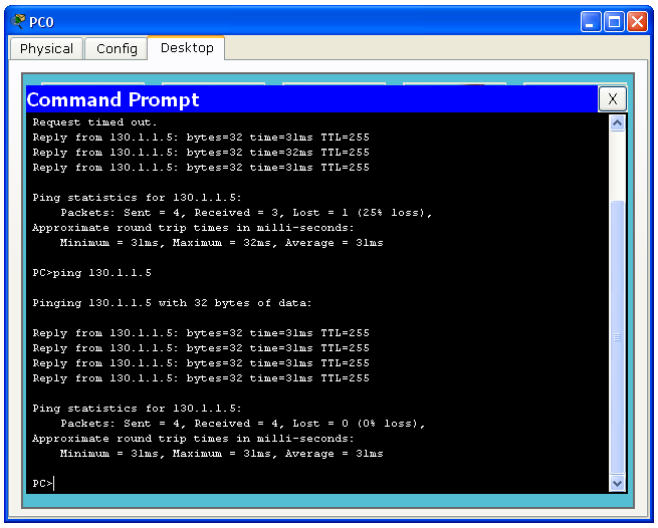


Рис.58 - Виконання ехо-запиту з хоста 130.1.1.2 до комутатора.

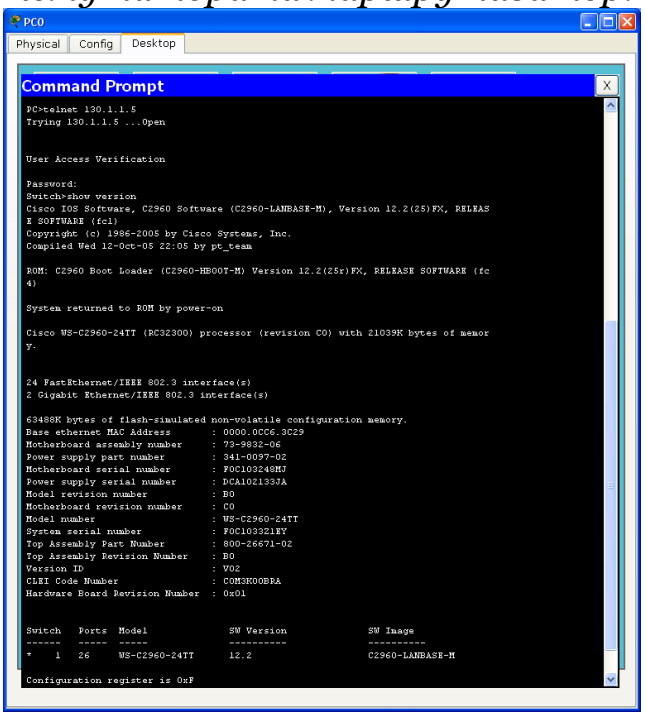


Рис.59 - Виконання telnet з'єднання.

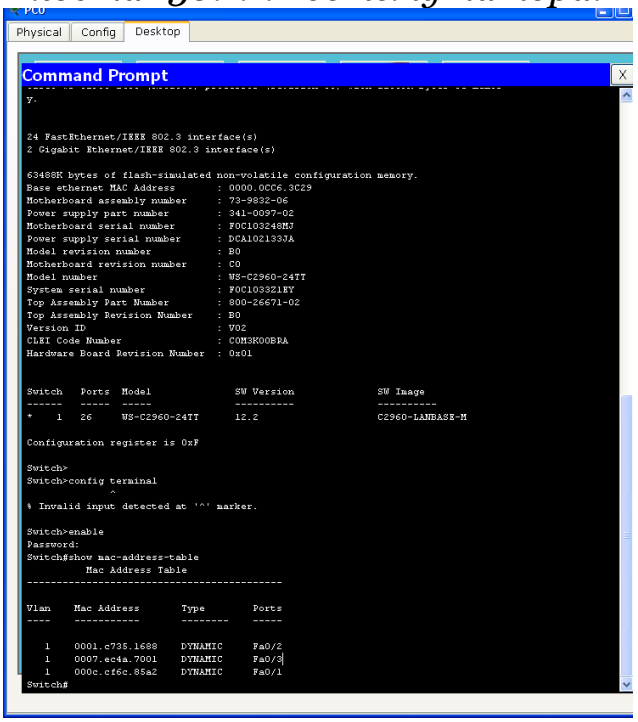


Рис.60 - Виведення списку мас-адрес

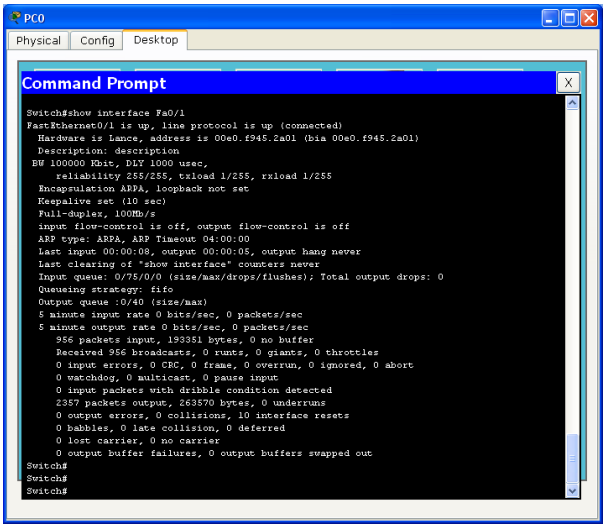


Рис.61 - Перегляд інформації про інтерфейс Fa0/1.

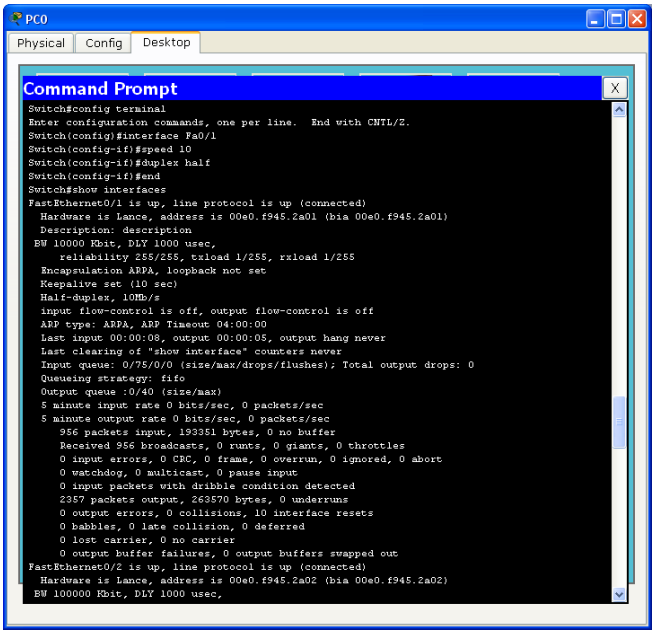


Рис.62. Налаштування полудуплексного режиму (10 Mb/s).

Зміст звіту:

1. Короткі теоретичні відомості;
2. Побудована мережева топологія;
3. Інформація про налаштування мережевого комутатора;
4. Встановлення telnet-з'єднання;
5. Інформація про налаштування швидкості передачі даних.

Лабораторна робота №9.
Конфігурація підмережі та маршрутизатору

Мета роботи: вивчити вимоги до адресного простору підмережі; конфігурувати інтерфейси Serial і Fast Ethernet; протестувати і перевірити конфігурацію.

ХІД РОБОТИ

Короткі теоретичні відомості

Адреси та маски.

Існує два варіанти протоколу IP - IPv4 та IPv6, що відрізняються розрядністю мережевих адрес і можливістю призначення гнучкого пріоритету пакета. В одній і тій же мережі обидва протоколи можуть співіснувати, але поки таких мереж, де підтримувався б протокол IPv6, досить мало. Більш того, мережі, з підтримкою і адмініструванням яких нам доведеться зіткнутися, швидше за все, створені не сьогодні і, отже, побудовані на основі "старого" варіанту IP, IPv4. В існуючій літературі, якщо треба підкреслити, що мова йде про IPv6, вказують, що це саме цей варіант IP, а якщо пишуть про IPv4, вживають термін "IP".

Тут і далі в цій роботі мова піде про IPv4, але багато хто з викладених принципів будуть справедливі і тоді, коли всі перейдуть на IPv6.

Ми будемо називати Мережевим інтерфейсом фізичний або віртуальний (тобто мається на увазі або програмно емульованого) пристрій, який здатний виконувати функції прийому пакетів даних від інших подібних пристроїв і передачі їм пакетів даних. Характерним прикладом мережевого інтерфейсу є мережеві адаптери (мережеві карти) і модеми. Кожен мережевий інтерфейс, який здійснює прийом і передачу пакетів по протоколу IP, повинен мати унікальний мережеву адресу. Під унікальним тут розуміється адреса, яка в межах даної IP-мережі не належить жодному іншому мережевому інтерфейсу. Один інтерфейс може мати кілька IP-адрес, але один і той же IP-адреса не може належати різним мережних інтерфейсів.

У протоколі IP визначено, що IP-адреса складається з чотирьох байт і записується у вигляді чотирьох десяткових чисел, відокремлених один від одного крапками. Кожне число відповідає значенню одного з цих байтів. IP-адреси об'єднані в блоки, які називаються мережами. У цьому значенні слово "мережа" вживається рідше, ніж в більш звичному значенні "сукупність комп'ютерів та інших мережевих пристроїв". Блоки адрес (мережі) класифіковані по класам мереж, які відрізняються один від одного особливостями маршрутизації.

Розбиття всієї множини адрес на блоки послідовних адрес (наприклад, 212.133.5.0-212.133.5.255) обумовлено тим, що в кожній точці мережі повинно бути відомо, в якому напрямку слід відправити пакет, адресований мережному інтерфейсу з якою адресою. Проміжні маршрутизатори, які об'єднують великі мережі і є своєрідними "вокзалами" для пакетів, які прямують з одного регіону планети в інший, не повинні зберігати записи про місцезнаходження кожного мережевого інтерфейсу і про маршрут проходження до нього. Їм досить знати, припустимо, що пакет, наступного за адресою 212.133.5.13, слід відправляти туди ж, куди і інші пакети з вищевказаного діапазону адрес. Це саме можна сказати і до маршрутизаторів, що об'єднує мережу середнього офісу з двома-трьома

діапазонами адрес, але весь тягар навантаження легше уявити на прикладі більш завантажених систем.

Ми можемо уподібнити пакет транзитної посилці, яка випливає з одного міста в інший на різних поїздах з декількома "пересадками". На кожному з проміжних вокзалів працівники пошти знають, в який бік слід відправити посилку, коли вона проходить повз них, але їх зовсім не турбує, куди її відправлять на наступному вокзалі: там про це подбають співробітники наступного поштового відділення.

Щоб полегшити роботу маршрутизаторів із запам'ятовування діапазонів адрес, були придумані маски мереж, що визначають, яку частину IP-адреси займає номер мережі, а яку - номер комп'ютера в цій мережі. Фактично, номер комп'ютера потрібен тільки тому маршрутизатора, до якого безпосередньо підключена локальна мережа, де знаходиться комп'ютер - адресат пакета. На відміну від номера комп'ютера, номер мережі використовується всіма проміжними маршрутизаторами, які передають пакет один одному від місця відправки до місця призначення.

Розглянемо, як використовується маска мережі, на прикладі. Уявімо собі мережу, що складається з трьох сегментів. У кожному з них - по 40 комп'ютерів. Ми вже знаємо, що для того, щоб забезпечити унікальними адресами кожен з них, нам знадобиться 120 адрес, тобто менше, ніж є в мережі класу C. Стало бути, ми можемо призначити для кожної підмережі свій діапазон адрес, при цьому досить використовувати адреси тільки з однієї мережі класу C. Виберемо три діапазону так, щоб в кожному з них було не менше 40 доступних адрес : 192.168.0.0-192.168.0.63, 192.168.0.64-192.168.0.127 і 192.168.0.128-192.168.0.191. Адреси мереж в цілому і широкомовні адреси включені в діапазони.

Комп'ютерам, які знаходяться в одному сегменті, ми присвоюємо адреси тільки з одного діапазону. Тепер наше завдання - пояснити маршрутизатору, через які мережеві інтерфейси слід передавати пакети в кожен із сегментів, і саме тут нам допоможе маска мережі.

У нас є абсолютно незмінна частина адрес в нашій мережі - 192.168.0. Ці три байта адреси однакові для всіх комп'ютерів нашої мережі. Видно, що адреси в різних діапазонах відрізняються значенням останнього байта. Відзначимо, що адреси першого діапазону в двох старших бітах цього байта мають нулі (дійсно, двійкове подання чисел до 63 включно дає значення від 00000000 до 00111111, старші два біти виділені жирним шрифтом). У другому діапазоні в згаданих бітах міститься 01: значення від 64 до 127 представляються в двійковому вигляді числами від 01000000 до 01111111. Аналогічно, третій діапазон дає нам двійкові числа від 10000000 до 10111111. Виділені біти різняться між діапазонами, але однакові в межах діапазону. Значення цих двох бітів (їх може бути більше, це залежить від числа діапазонів - підмереж, на які розбита мережа) прийнято називати номером підмережі; перший діапазон називають нульовою підмережею, другий - першої підмережею і т.д., дивлячись за значенням цих бітів.

Щоб повідомити маршрутизатора, що до його першого мережного адаптера приєднана нульова підмережа, до другого - перша і до третього - друга, ми повинні всього лише вказати маску мережі на кожному з його мережевих інтерфейсів.

Так як ми вже домовилися вважати номером мережі в IP-адресу значення тих бітів, які в масці мають двійкове значення "1", то маска мережі, розділеної на чотири підмережі, матиме значення 255.255.255.192.

Чому значення останнього байта маски - 192? Тому, що 11000000 двоичное дає саме десяткове 192. Чому ми ділимо мережу на чотири підмережі, хоча мова йшла про три діапазонах? Справа в тому, що мережа можна розділити тільки на таку кількість підмереж, яке кратно ступеня двійки. Тому замість трьох підмереж доводиться брати найближче більше їх кількість. Тепер залишається призначити кожному інтерфейсу маршрутизатора адреси в діапазоні тієї підмережі, яка приєднана до цього інтерфейсу, і насолодитися його чіткою роботою - з пересилання пакетів між інтерфейсами.

Класи мереж.

Найбільшою IP-мережею в світі є глобальна мережа Internet. Її адресний простір поділений на діапазони адрес, які називаються "мережами". Мережі розділені на класи. Адреси мережевих інтерфейсів комп'ютерів в мережі, як правило, відносяться до класів A, B або C.

Поділ мереж на класи визначається в RFC 950 і ряді інших RFC2). Крім мереж A, B, C є й інші мережі, що представляють собою менші за розміром блоки мереж, які використовуються для різних службових потреб.

Мережі класів від A до C відрізняються значенням першого байта IP-адреси. Значення першого байта мережі класу A знаходиться в діапазоні від 1 до 126, класу B - від 128 до 191, класу C - від 192 до 223. Класи мереж, в адресах яких перший байт має значення від 224 до 254, іменуються від D до F . Вони мають службове призначення, і їх адреси не використовуються звичайними мережевими інтерфейсами.

Передбачається, що в мережах класу A номер мережі займає перший байт, а решта три - це номер комп'ютера (точніше, номер мережевого інтерфейсу). Таким чином, в мережі класу A може бути до 16777214 інтерфейсів. В мережі класу B номер мережі займає два байти адреси, максимальне число інтерфейсів в такій мережі - 65534. В мережі класу C номер мережі займає три байта, номер комп'ютера - один, максимальне число інтерфейсів в такій мережі - 254.

Зарезервовані мережеві адреси.

Треба мати на увазі, що один комп'ютер може мати кілька мережевих інтерфейсів, а кожен інтерфейс може мати кілька адрес.

Деякі адреси в кожній мережі є зарезервованими і не можуть використовуватися для адресації будь-якого інтерфейсу.

Так, IP-адреса, в якому поле номера комп'ютера заповнене двійковими нулями, використовується в якості номера даної мережі в цілому. Наприклад, адреса 131.45.0.0 позначає цілу мережу класу B. IP-адреса, в якому поле номера комп'ютера заповнене двійковими одиницями, є широкомовною адресою мережі (broadcast address) і застосовується для одночасної розсилки пакета всіх комп'ютерів даної мережі. Отримавши пакет з адресою одержувача 131.45.255.255, кожен комп'ютер мережі 131.45.0.0 (клас B) сприйме цей пакет як призначений йому. Ці зарезервовані адреси використовуються, наприклад, в цілях управління маршрутизацією.

Існують і інші зарезервовані адреси. Адреса 127.0.0.1 завжди вказує на локальний внутрішній інтерфейс системи. "127.0.0.1" позначає для системи те ж саме, що для людини - слово "я". Цей локальний внутрішній інтерфейс потрібно для того, щоб одна програма (клієнт) могла звернутися до іншої програми (сервера), що працює на тому ж комп'ютері, стандартним чином. Наприклад,

можна звернутися з броузера на вашому комп'ютері до веб-сервера на вашому ж комп'ютері. Локальний інтерфейс зазвичай називається lo або loo (від слова loopback - "петля").

Адреса 0.0.0.0 використовується для позначення маршруту за замовчуванням (основного шлюзу), так як ця адреса означає "все мережі".

Основний шлюз.

Він зарезервований для вказівки "кожного одержувача". Якщо пакет не вдається співвіднести з конкретною рядком таблиці маршрутизації, він відправляється в шлюз згідно вказує на нього рядку таблиці маршрутизації. Наприклад, таблиця маршрутизації нашого хоста виглядає так:# netstat -rn

Routing Table: IPv4

Destination Gateway Flags Ref Use Interface

192.168.5.0 192.168.5.33 U 1 2 elxl0

224.0.0.0 192.168.5.33 U 1 0 elxl0

default 192.168.5.1 UG 1 0

127.0.0.1 127.0.0.1 UH 61 1013 loo

Адреса нашого комп'ютера в цій мережі - 192.168.5.33. Припустимо, нам треба відправити пакет за адресою 192.168.5.30. Для цього наша система подивиться в таблицю маршрутизації і виявить там маршрут 192.168.5.0, для якого вказано шлюз (gateway) 192.168.5.33 - наш власний інтерфейс. Стало бути, комп'ютер 192.168.5.30 знаходиться в безпосередньо приєднаної до нас мережі і йому треба послати пакет безпосередньо. Ситуація зміниться, якщо адресатом буде, скажімо, комп'ютер 192.168.10.1. Тоді в таблиці маршрутизації з самого початку не знайдеться відповідного маршруту - адже там немає окремого рядка для мережі 192.168.10.1, вірно? Тоді пакет буде відправлений в основний шлюз, "шлюз", той, що у висновку netstat позначений словом default. Адреса призначення пакетів "в усі інші мережі", тих самих, які відправляються в шлюз, в таблиці маршрутизації в ядрі позначається як 0.0.0.0.

Пакети, призначені для відправки "всім іншим", направляються в основний шлюз. Основний шлюз (default gateway) - це таке місце, куди будь-який комп'ютер мережі відправляє пакет, якщо не знає, в який бік його краще відправити. Дія такого шлюзу подібна до дії листоноші. Якщо Ви хочете зробити сюрприз дівчині, яка живе з вами в одному під'їзді, ви можете покласти ніжне лист їй прямо в поштову скриньку. Якщо ж адресат живе в іншому місті, ви покладете конверт в іншу поштову скриньку, той, з якого листоноша виймає пошту для відправки. Основний шлюз має мережевий інтерфейс, який працює таким "поштовою скринькою" для пакетів, які адресовані з локальної мережі зовні.

ВИКОНАННЯ

На рис. 63 приведена мережева топологія побудована з використанням наступного мережевого обладнання:

персональних комп'ютерів;

комутатора;

маршрутизатора.

Для даної підмережі треба було створено 3 підмережі, а саме:192.168.1.0/24;

192.168.2.0/24;

192.168.3.0/24.

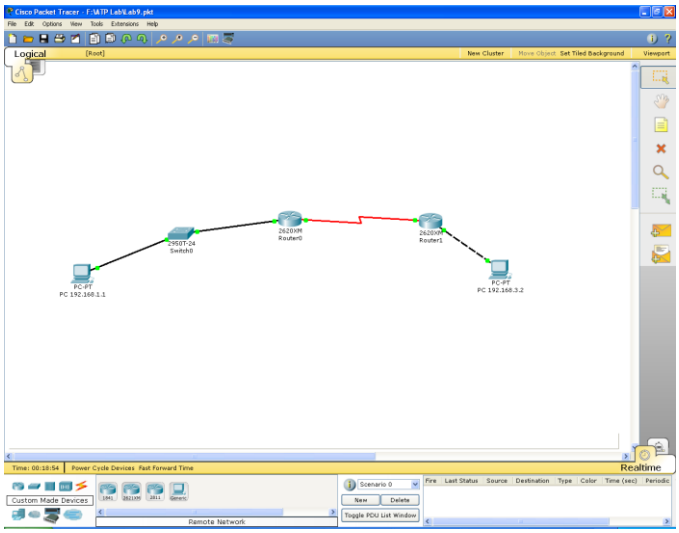


Рис.63 - Топологія мережі

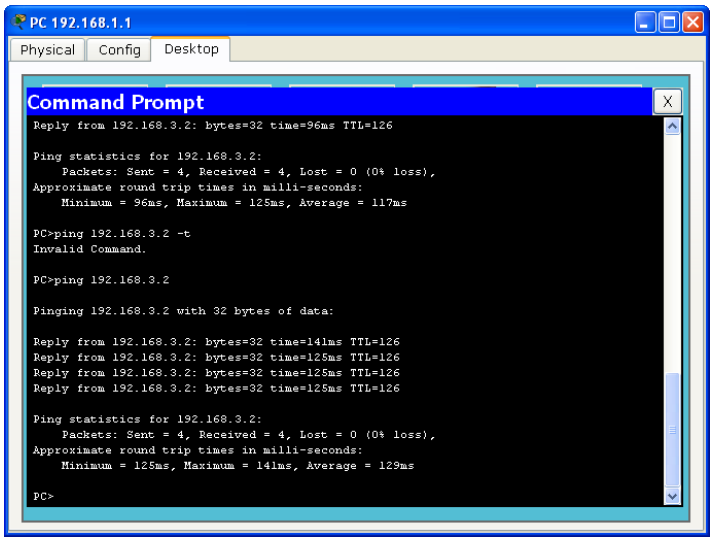


Рис.64 - Виконання команди ping 192.168.3.2

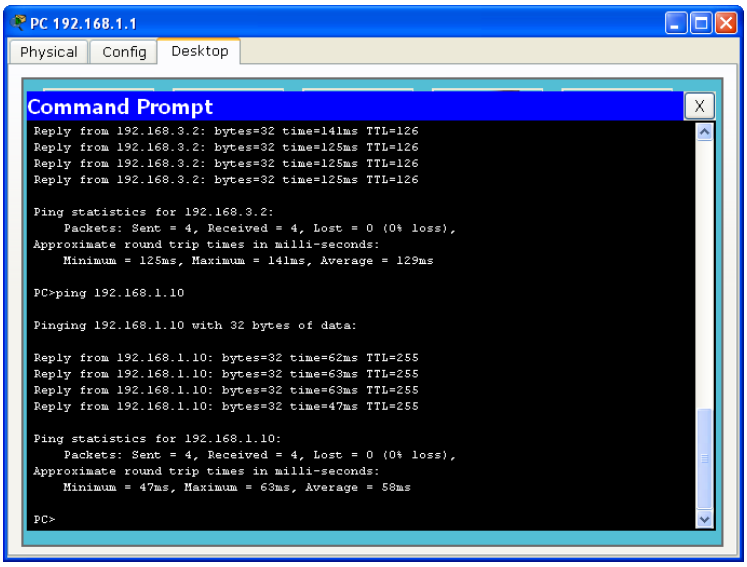


Рис.65 - Виконання команди ping 192.168.1.10

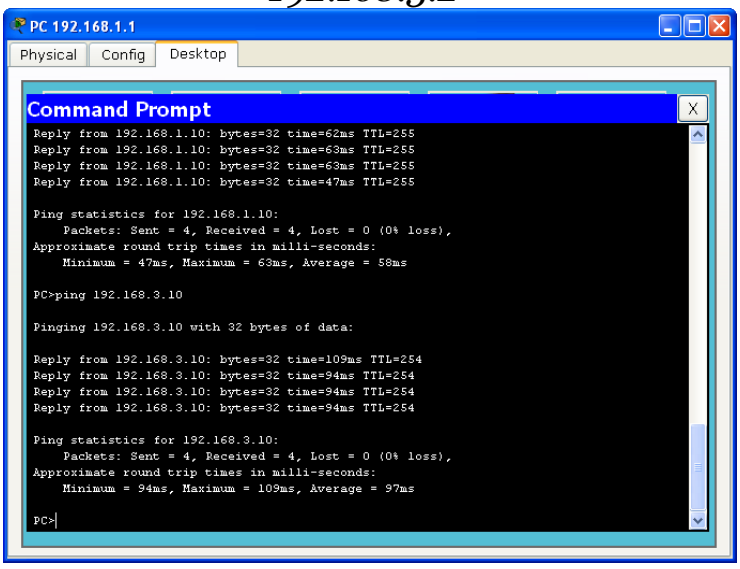


Рис. 66 - Виконання команди ping 192.168.3.10

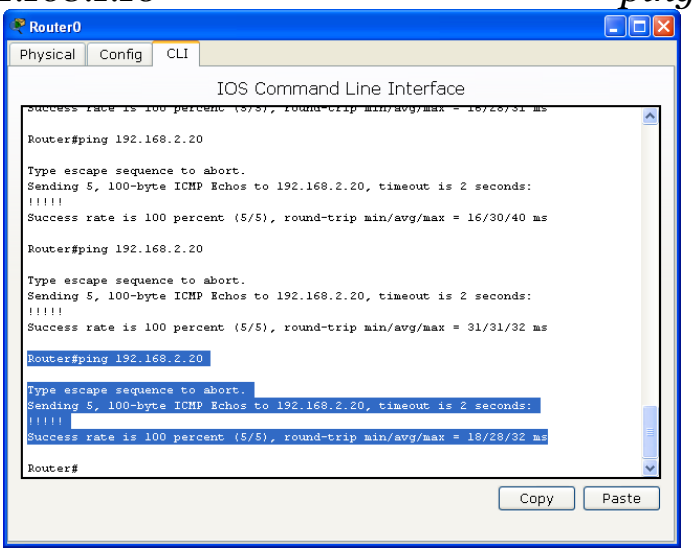


Рис. 67 - Виконання команди «ping 192.168.2.10»(Router0) з Router1

Зміст звіту:

1. Короткі теоретичні відомості;
2. Інформація сконфігурованої підмережі маршрутизаторів;
3. Налаштування адресного простору маршрутизатора.

ЛІТЕРАТУРА

1. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 2-е изд. / В.Г. Олифер , Н.А. Олифер –СПб. Питер, 2004. – 864 с.: ил.
2. Пятибратов А.П. и др. Вычислительные системы, сети и телекоммуникации. М. 1998.
3. Денисова А., Вихарев И., Белов А., Наумов Г. Интернет. Самоучитель. 2-е изд. –Питер. 2004.
4. Глушаков С.В., Ломотько Д.В., Сурядный А.С. Работа в сети Internet/ 2-е изд., доп. и перераб./ Худож. - оформитель А.С. Юхтман. – Харьков: Фолио, 2003. - 399 с. – (Учебный курс).
5. В. Холмогоров Компьютерная сеть своими руками. Самоучитель. СПб.: Питер. 2004. - 171 с.
6. Аппаратные средства локальных сетей. Энциклопедия / М.Гук, - СПб.: Питер, 2004. – 573 с.: ил.

ПОЗНАЧКИ