Міністерство освіти і науки, молоді та спорту України
Національний університет водного господарства
та природокористування
Факультет прикладної математики та комп'ютерно-
інтегрованих систем
Кафедра електротехніки та автоматики

0xx-xx

МЕТОДИЧНІ ВКАЗІВКИ

до виконання контрольної роботи
з дисципліни „ Комп'ютерні системи та
мережі в АСКТП"
для студентів заочної форми навчання, які навчаються за
напрямом підготовки 6.050202 "Автоматизація та
комп'ютерно-інтегровані технології"

Рекомендовано до друку
методичною комісією за
напрямом підготовки 0925
„Автоматизація та комп'ютерно
інтегровані технології"
Протокол № ___ від ____ 2011 р.

Рівне 2011

Методичні вказівки до виконання контрольної роботи з дисципліни “Комп’ютерні системи та мережі в АСКТП” для студентів заочної форми навчання, які навчаються за напрямом підготовки 6.050202 "Автоматизація та комп'ютерно-інтегровані технології". /В.В. Жуковський, - Рівне, НУВГП, 2011. – 32 с.

Упорядник: Жуковський В.В.

Рецензент:

Відповідальний за випуск Б.О.Баховець, професор, завідувач кафедри електротехніки і автоматики.

Зміст

1. ЗАГАЛЬНІ ПОЛОЖЕННЯ

Основний вид навчальної роботи студентів заочної форми навчання - самостійна робота над навчальним матеріалом. При цьому варто орієнтуватися на матеріал тих тем, що зазначені в переліку завдань. Список літератури, що рекомендується, наведений в кінці методичних вказівок.

Згідно з робочим навчальним планом студент повинен виконати домашню контрольну роботу. Виконання контрольної роботи надає можливість підтвердити своє вміння самостійно працювати з різними інформаційними джерелами і на базі цього робити самостійні теоретичні й практичні висновки та пропозиції.

Контрольна робота з дисципліни: „Комп'ютерні системи та мережі в АСКТП" складається з двох теоретичних та одного практичного завдання.

2. ВИМОГИ ДО ОФОРМЛЕННЯ

Контрольна робота повинна бути ретельно відредагованою і чітко віддрукованою (або написаною від руки) на аркушах формату А4.

Вимоги до оформлення курсової роботи за допомогою ПЕОМ:

- шрифт - Times New Roman;
- розмір шрифту - 14 кегель;
- інтервал між рядками - одинарний;
- текст вирівняний по ширині;
- абзац - 5 мм,
- поля: праве, ліве, верхнє - 15 мм, нижнє - 17 мм;
- нумерація сторінок - по центру нижнього поля, наскрізна;
- рисунки, таблиці, формули друкуються по центру, нумерація по правому верхньому полю, основний розмір шрифту - 14 кегель.

Робота повинна також містити зміст, завдання та список використаної літератури. Зразок оформлення титульної сторінки наведено в додатку.

3. ВАРІАНТИ ЗАВДАНЬ

Завдання контрольної роботи включає в себе 2 теоретичних питання та 1 практичне завдання. Варіанти завдань вибираються згідно нижченаведеної таблиці 3.1, де по горизонталі вказано останню цифру залікової книжки, а по вертикалі передостанню. Наприклад, останні дві цифри залікової книжки 89, тоді теоретичними питаннями будуть питання №19 та №38, а практичним завданням - №20.

Варіанти завдань

По горизонталі - остання цифра залікової книжки.

По вертикалі - передостання цифра залікової книжки.

Таблиця 3. 1

	0	1	2	3	4	5	6	7	8	9
0	1, 10, 10	2, 19, 11	3, 17, 12,	4, 16, 13	5, 15, 14	6, 14, 15	7 , 13, 16	8, 12, 12	9, 11, 18	10, 40, 17
1	11, 39, 20	12, 38, 1	13, 37, 2	14, 36, 3	15, 35, 4	16, 34, 5	17, 33, 6	18, 32, 7	19, 31, 8	20, 30, 9
2	21, 29, 10	22, 28, 11	23, 27, 12	24, 37, 13	25, 5, 14	26, 19, 15	27, 18, 16	28, 17, 13	29, 16, 18	30, 15, 17
3	31, 14, 20	32, 13, 1	33, 12, 2	34, 11, 3	35, 40, 4	36, 19, 5	37, 15, 6	38, 16, 7	39, 17, 8	40, 18, 9
4	11, 19, 10	12, 20, 11	13, 21, 12	14, 22, 13	15, 23, 14	16, 24, 15	17, 25, 16	18, 26, 11	19, 27, 18	30, 28, 13
5	35, 05, 20	34, 20, 17	33, 19, 18	32, 18, 14	31, 17, 16	30, 16, 15	29, 15, 14	28, 14, 13	27, 13, 12	26, 12, 11
6	25, 11, 10	24, 40, 9	23, 19, 8	22, 38, 7	21, 37, 6	20, 36, 5	19, 35, 6	18, 34, 5	17, 33, 4	16, 32, 3
7	15, 31, 1	14, 30, 2	13, 29, 3	12, 28, 4	11, 27, 5	10, 26, 6	9, 25, 7	8, 24, 8	7, 23, 9	6, 22, 10
8	5, 21, 11	4, 20, 12	3, 15, 13	2, 14, 14	1, 13, 15	15, 12, 16	16, 11, 10	17, 30, 18	18, 39, 17	19, 38, 20
9	20, 37, 1	21, 1, 2	22, 2, 3	23, 3, 4	24, 5, 5	25, 8, 6	26, 16, 7	27, 19, 8	28, 33, 9	29, 34, 10

Перші дві цифри – номери теоретичних питань.

Третя цифра – номер практичного завдання.

4. ПЕРЕЛІК ТЕОРЕТИЧНИХ ПИТАНЬ

1. Поняття та історія розвитку інформаційних систем.
2. Інтегровані автоматизовані системи управління.
3. Організаційна інтеграція в контексті створення ІАСУ.
4. Функціональна інтеграція в контексті створення ІАСУ.
5. Інформаційна інтеграція в контексті створення ІАСУ.
6. Технічна інтеграція в контексті створення ІАСУ.
7. Визначення промислової мережі та їх градація.
8. Промислові мережі в контексті моделі ISO OSI.
9. Основні робочі характеристики промислових мереж.
10. Електричні шуми, завади та боротьба з ними. Контроль за помилками.
11. Протокол, пакет, його структура.
12. Інкапсуляція даних в контексті мережної взаємодії.
13. Технологія Ethernet: стандарти, принцип роботи, компоненти та обладнання.
14. Технологія Token-Ring: стандарти, принцип роботи, компоненти та обладнання.
15. Технологія FDDI: стандарти, принцип роботи, компоненти та обладнання.
16. Технологія 100VG-AnyLAN: стандарти, принцип роботи, компоненти та обладнання.
17. Технологія ATM: стандарти, принцип роботи, компоненти та обладнання.
18. Технологія ISDN: стандарти, принцип роботи, компоненти та обладнання.
19. Технологія Frame relay: стандарти, принцип роботи, компоненти та обладнання.
20. Технологія DWDM: стандарти, принцип роботи, компоненти та обладнання.
21. Технологія LanDrive: стандарти, принцип роботи, компоненти та обладнання.
22. Технологія Modbus: стандарти, принцип роботи, компоненти та обладнання.
23. Технологія LonWorks: стандарти, принцип роботи, компоненти та обладнання.
24. Технологія CAN: стандарти, принцип роботи, компоненти та обладнання.

25. Технологія Profibus: стандарти, принцип роботи, компоненти та обладнання.

26. Технологія HART: стандарти, принцип роботи, компоненти та обладнання.

27. Технологія AS-Interface: стандарти, принцип роботи, компоненти та обладнання.

28. Технологія EIB: стандарти, принцип роботи, компоненти та обладнання.

29. Технологія ZigBee: стандарти, принцип роботи, компоненти та обладнання.

30. Широкомовні мережі. Множинний доступ.

31. Метод доступу до середовища TDMA.

32. Метод доступу до середовища FDMA.

33. Метод доступу до середовища CDMA.

34. Метод доступу до середовища CSMA/CD.

35. Метод доступу до середовища CSMA/CA.

36. Маркерний метод доступу до середовища.

37. Захист інформації в широкомовних мережах.

38. Порівняльна характеристика технологій промислових мереж

39. Проблеми вибору оптимального протоколу обміну згідно поставленої задачі.

40. Програмування мережної взаємодії.

5. ПРАКТИЧНЕ ЗАВДАННЯ

Практичним завданням є дослідження структури мережного пакета. Пакети можна переглядати за допомогою спеціальних програм: сніферів (Sniffer, від англ. to sniff – нюхати) та спуферів (Spoofer).

Необхідно розглянути і вирішити наступні задачі:

1. Організувати мережну взаємодію згідно свого варіанту завдання.

2. Перехопити необхідні пакети.

3. Проаналізувати пакет канального рівня. Розписати всі інформаційні поля пакетів згідно моделі OSI.

Поняття пакета даних

Інформація в локальних мережах, як правило, передається окремими порціями, блоками, званими в різних джерелах ***пакетами (packets***), ***кадрами (frames) або блоками***. Причому гранична довжина цих пакетів строго обмежена (максимальна довжина). Обмежена довжина пакету і знизу (мінімальна довжина).

Тобто процес інформаційного обміну в мережі є чергуванням пакетів, кожний з яких містить інформацію, що передається від абонента до абонента.

Якби вся необхідна інформація передавалася якимось абонентом відразу, безперервно, без розділення на пакети, то це привело б до монопольного захоплення мережі цим абонентом на досить тривалий час. Вся решта абонентів була вимушена б чекати закінчення передачі всієї інформації, що у ряді випадків могло б тривати десятки секунд і навіть хвилин (наприклад, при копіюванні вмісту цілого жорсткого диска). З тим щоб зрівняти в правах всіх абонентів, а також зробити приблизно однаковою для всіх них величину часу доступу до мережі і інтегральну швидкість передачі інформації, якраз і застосовуються пакети (кадри) *обмеженої довжини*. Це і є основним при комутації пакетів.

Кожний пакет крім корисних даних, які вимагається передати, повинен містити деяку кількість ***службової інформації***. Перш за все, це *адресна інформація*, яка визначає, від кого і кому передається даний пакет (як на поштовому конверті – адреси одержувача і відправника). У різних протоколах різні вимоги до створення пакетів. Умовно це можна порівняти з різними мовами народів світу (протоколами обміну даними), де в кожного народу свої правила побудови речень (створення пакетів).

Нагадаємо, що згідно моделі OSI виділяють 7 рівнів:

Таблиця 5.1

Прикладний	Application layer
Відображення	Presentation layer
Сеансовий	Session layer
Транспортний	Transport layer
Мережевий	Network layer
Канальний	Data link layer
Фізичний	Physical layer

Кожен з цих рівнів визначає свої протоколи (правила обміну).

У мережі має місце фізичне та логічне переміщення даних.

Фізичне переміщення даних починається на верхньому рівні (прикладному) і йде вниз по всіх рівнях моделі. Наприклад: на верхньому рівні було створено інформацію (користувач відкрив браузер і ввів адресу сайту, тобто надіслав запит). Протокол прикладного рівня передає ці дані в певній формі протоколу комунікаційного рівня. На цьому рівні проходить "упаковка" інформації в інформаційний пакет визначеної структури. Цей пакет передається протоколу рівня передачі даних для фізичної пересилки. Потім ці дані переміщуються по мережевому носії у вигляді імпульсів, що відповідають 0 або 1. Цей носій може бути різного виду кабелем, радіоканалом, … Як тільки дані дійшли до комп'ютера-отримувача, вони починають переміщатись знизу догори. На кожному рівні вони обробляються, але виділяється тільки та частинка, яка була запакована на тому ж рівні, що й у комп'ютері-передавачі. В кінці інформація доходить до користувача на прикладному рівні.

Процес вкладання інформації, створеної за деяким протоколом, в секцію даних блока даних іншого протоколу називається **інкапсуляцією**. У випадку наведеному вище це відбувається при вкладанні даних протоколу вищого рівня в протокол нижчого.

Програми аналізатори трафіку

Аналізатор трафіку, або **сніфер** (від англ. to sniff — нюхати) — програма або програмно-апаратний пристрій, призначений для перехоплення і подальшого аналізу, або тільки аналізу мережного трафіку, призначеного для інших вузлів.

Однією з таких програм є програма *The Wireshark Network Analyzer*. Wireshark — програма для аналізу пакетів Ethernet і деяких інших мереж. Має графічний інтерфейс користувача. У червні 2006 року проект був перейменований на Wireshark через проблеми з торговою маркою. Дана програма є безкоштовною і її можна скачати з офіційного сайту: http://www.wireshark.org

Функціональність, яку надає Wireshark, дуже схожа з можливостями програми tcpdump, проте Wireshark має графічний інтерфейс користувача і значно більше можливостей із сортування і фільтрації інформації. Програма дозволяє користувачеві переглядати весь трафік, що проходить по мережі, в режимі реального часу, переводячи мережеву карту в широкомовний режим (англ. promiscuous mode).

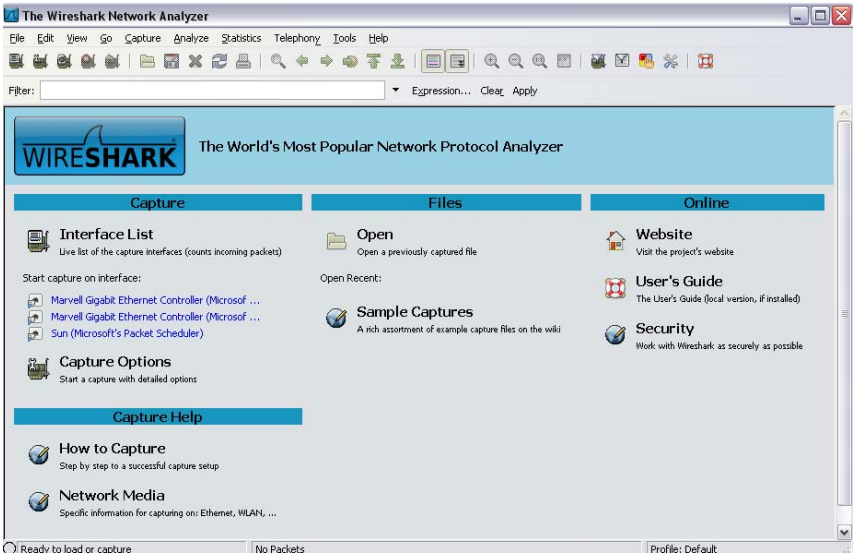

Рисунок 5.1. Вікно програми Wireshark

Програма дозволяє аналізувати багато протоколів різних форматів, починаючи від всім відомих кадрів Ethernet, пакетів TCP/IP, ICMP і закінчуючи протоколами для quake3, TDMA, telnet, daytime. Детальний перелік можна переглянути на сайті http://www.wireshark.org/docs/dfref/ або в довідці.

Потрібно знати, що Wireshark ніяким чином не контролює пакети, не сигналізує про вторгнення чи атаки і не посилає пакетів в мережу. Вона лише дозволяє побачити поточний стан мережевої активності, а подальші дії залежать від користувача.

Головне вікно програми поділене на 4 області:

Capture – вибір інтерфейсів для захоплення пакетів та його налаштування.
Capture Help — довідкова інформація.
Filters — дозволяє відкрити вже збережені налаштування, також містить демонстраційні налаштування для захоплення специфічних пакетів.
Online — довідкова інформація в мережі інтернет.

Робота з програмою починається з вибору інтерфейсу в розділі Interface List:

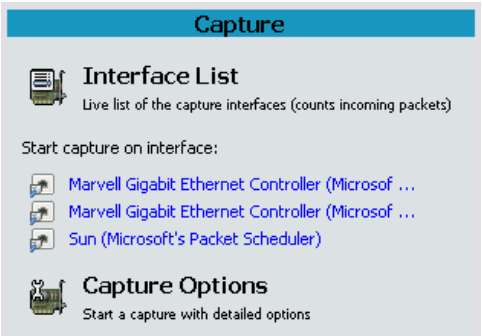

Рисунок 5.2. Діалог вибору мережних інтерфейсів

Він містить всі доступні інтерфейси для захоплення. Це можуть бути як реальні фізичні інтерфейси, представлені мережевими картами, вбудовані в материнську плату, так і віртуальні, створені

різними програмами (Virtual Box, VMWare).

Відразу після того, як буде вибрано інтерфейс почнеться процес захоплення всіх файлів:

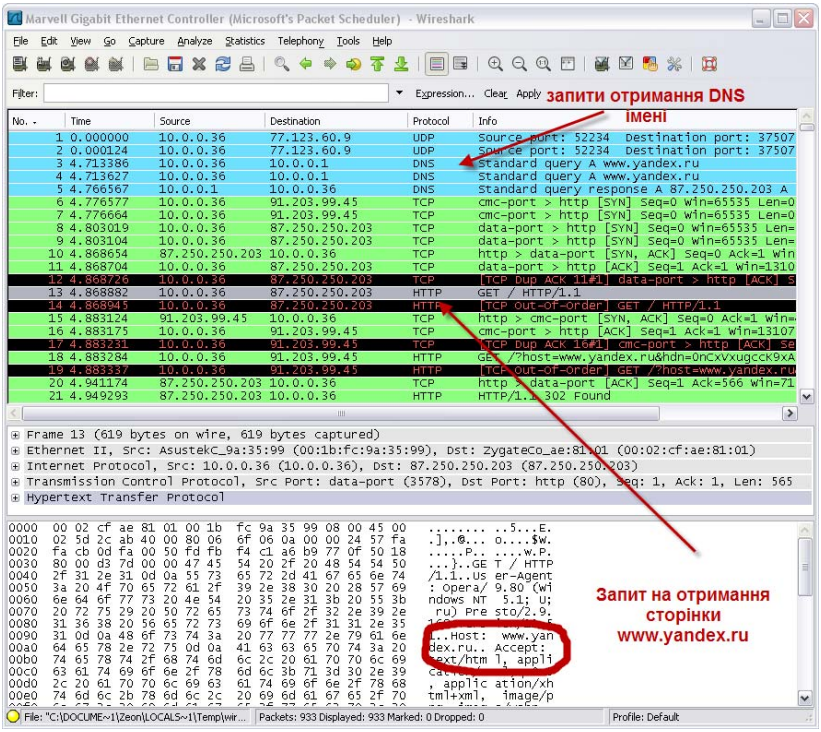

Рисунок 5.3. Процес захоплення пакетів

Піктограма для зупинки процесу має вигляд:

Разом з тим існує декілька варіантів побачити і тим самим зупинити процес захоплення пакетів:

- Використовуючи кнопку "Stop" з діалогу “Capture Info”
- Використовуючи пункт головного меню: “Capture/ Stop”.
- Використовуючи кнопку "Stop" на панелі інструменів.
- Натиснувши гарячі клавіші Ctrl+E.

- Також процес захоплення буде автоматично зупинено при виконанні умови Stop Conditions, наприклад буде досягнуто максимальної к-ті захоплених пакетів.

Якщо вибрати будь-який пакет зі списку, тоді в нижньому вікні програми будуть детально описані протоколи, що використовуються (наприклад TCP, IP, Ethernet, HTTP ітд):

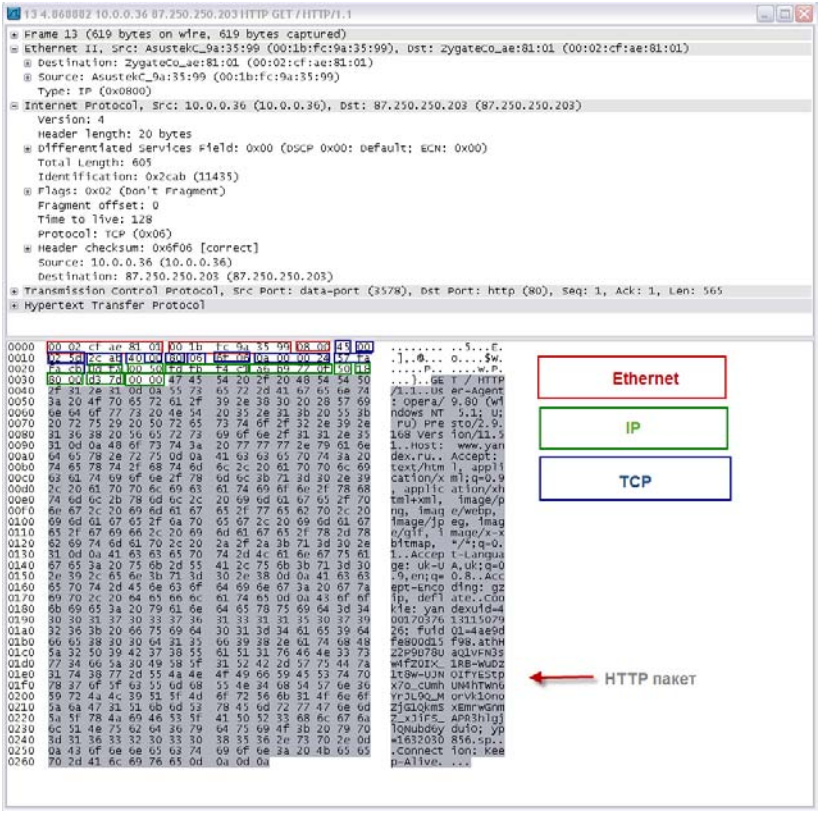

Рисунок 5.4. Деталізація перехопленого пакету

Тобто програма автоматично розпізнає ключові поля загальновідомих протоколів: адреса відправника, адреса призначення, контрольна сума, порт відправника, порт призначення, додаткова інформація.

6. ВАРІАНТИ ПРАКТИЧНИХ ЗАВДАНЬ

Проаналізувати мережну взаємодію з наступним ресурсом:

1. www.ya.ru
2. www.google.com
3. www.gmail.com
4. www.rada.gov.ua
5. http://rutracker.org
6. www.ukr.net
7. www.mail.ru
8. www.facebook.com
9. www.vkontakte.ru
10. www.icq.com
11. www.habrahabr.ru
12. www.wikipedia.org
13. www.ua.fm
14. www.icq.com
15. www.mail.ua
16. http://nuwm.rv.ua
17. www.i.ua
18. www.gismeteo.ua
19. www.rozetka.com.ua
20. www.twitter.com
21. www.liveinternet.ru
22. www.webmoney.ru
23. www.uz.gov.ua
24. http://2ip.ru/
25. http://speedtest.net

Спочатку виконайте команду ping (для перевірки відгуку від сервера), а потім організуйте обмін даними по HTTP протоколу.

Якщо ресурс передбачає прийом даних від користувача (введення логіну і паролю, пошук певних даних ітд) проаналізувати як саме дані передаються на сервер. Знайти ці дані в пакетах, що передаються.

7. ПРИКЛАД ВИКОНАННЯ ПРАКТИЧНОГО ЗАВДАННЯ

Завдання:

Проаналізувати мережну взаємодію з ресурсом www.ya.ru.

Виконання:

Ресурс www.ya.ru належить компанії «Яндекс» і є спрощеною версією ресурсу www.yandex.ua. На спрощеній версії немає реклами та різних інформаційних блоків, а є лише стрічка для пошуку:

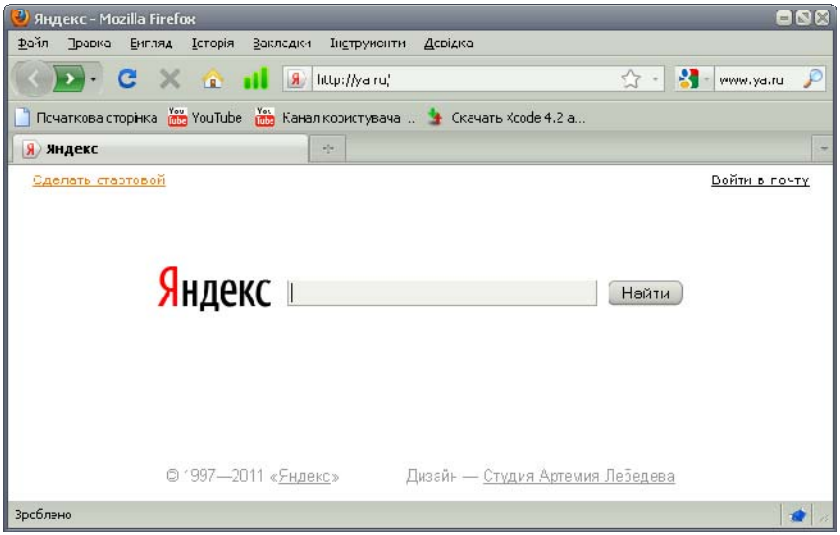

Рисунок 7.1. Ресурс www.ya.ru

Згідно завдання потрібно перевірити мережну взаємодію по ICMP протоколу (команда ping) та по HTTP (перегляд веб сторінок).

Запускаємо сніфер wireshark і виберемо потрібний мережний інтерфейс для захоплення.

Тепер виконаємо команду «ping ya.ru» (для перевірки відгуку від сервера).

Для перевірки відгуку віддаленого комп'ютера запускаємо утиліту ping в командному рядку (Пуск-Все программы-

Стандартные-Коммандная строка):

```
C:\>ping ya.ru

C:\Documents and Settings\Zeon>ping ya.ru

Обмен пакетами с ya.ru [93.158.134.3] по 32 байт:

Ответ от 93.158.134.3: число байт=32 время=69мс TTL=56
Ответ от 93.158.134.3: число байт=32 время=68мс TTL=56
Ответ от 93.158.134.3: число байт=32 время=68мс TTL=56
Ответ от 93.158.134.3: число байт=32 время=67мс TTL=56

Статистика Ping для 93.158.134.3:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0 (0% потерь),
Приблизительное время приема-передачи в мс:
    Минимальное = 175мсек, Максимальное = 198 мсек, Среднее = 186 мсек

C:\>
```

Утиліта *Ping* використовує повідомлення Echo і Echo Reply для визначення, чи потрібна станція досяжна.

Поки процесс захоплення пакетів триває, відкриваємо браузер і в ньому звертаємось до сайту www.ya.ru. Повинні отримати результат, який було зображено вище на рис. 7.2

Тепер зупиняємо захоплення пакетів і дивимось як виглядають перехоплені дані в сніфері. Для цього відфільтруємо лишні пакети. Ми знаємо IP адресу вузла ya.ru - 93.158.134.3. Отже, використаємо фільтр, де вкажемо цю IP адресу. (Для того щоб дізнатися власну IP адресу можна скористатися утилітою ipconfig)

Фільтр виглядатиме так:

```
ip.addr == 93.158.134.3
```

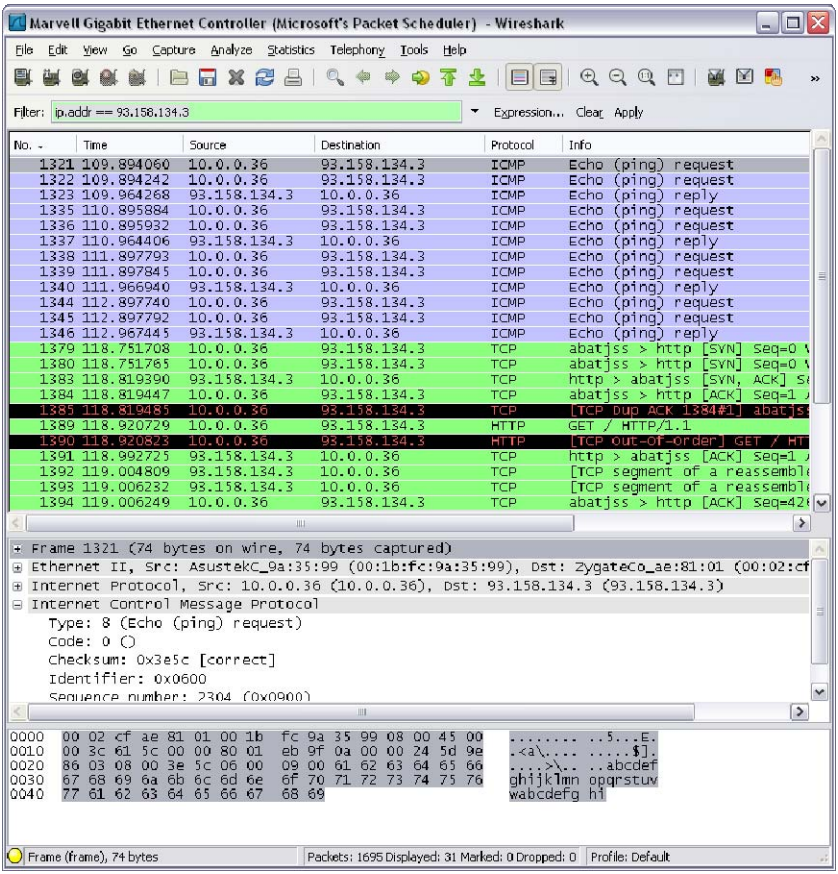

Рисунок 7.2. Фільтрація пакетів до ресурсу ya.ru

Перші пакети, протокол яких ICMP, це пакети утиліти ping. Розглянемо детальніше їхню структуру. Для цього спочатку звернемось до документації, а потім проаналізуємо наші пакети. Виділивши будь-який з пакетів, програма Wireshark надасть інформацію про вміст пакету.

Структура ICMP пакету згідно стандарту

Рисунок 7.3. Структура ICMP пакету

На рисунку 7.3 показано формат ICMP-повідомлення. Перші байти мають однаковий формат для всіх ICMP-повідомлень, але решта залежить від його типу. Значення відповідних полів наведені в таблиці нижче:

Таблиця 7.1

Тип (type)	Тип ICMP-повідомлення; всього існує 15 різних типів;
Код (code)	код ICMP-повідомлення одного типу;
Контрольна сума ICMP (checksum)	16-бітова контрольна сума, що охоплює все ICMP-повідомлення, обчислюється так само як для IP-заголовку; для обчислення контрольної суми поле повинне мати значення 0.
Ідентифікатор (identifier)	Ідентифікатор, який застосовується для узгодження запиту/відповіді; повинен дорівнювати нулю.

Номер послідовності (sequence number)	Номер послідовності, який застосовується для узгодження запити/відповіді; повинен дорівнювати нулю.
Маска адреси (address mask)	32-бітова маска адреси
Дані ICMP (body)	Саме повідомлення, має змінну довжину, його формат залежить від типу ICMP-повідомлення.

Нижче наведені всі *типи* ICMP-повідомлень та їх короткий опис.

Таблиця 7.2

Тип	Зміст
0	ехо (відповідь)
3	призначення недосяжне
4	обірвання джерела
5	переспрямування
8	ехо (запит)
9	оголошення раутера
10	вимога раутера
11	час вичерпаний (TTL=0)
12	проблема із параметром
13	запит часової позначки
14	відповідь про часову позначку
15	запит інформації (застаріле)
16	відповідь про інформацію (застаріле)
17	запит мережевої маски
18	відповідь про мережеву маску

Поле *код* містить код помилки для данограми, про яку повідомляється у даному ICMP-повідомленні. Інтерпретація коду залежить від типу повідомлення:

Таблиця 7.3

Тип	Код	Зміст
3	0	мережа недосяжна
3	1	станція недосяжна
3	2	протокол недосяжний
3	3	порт недосяжний
3	4	необхідна фрагментація і встановлено DF-біт
3	5	невдалий маршрут від джерела
3	6	невідома мережа призначення
3	7	невідомий станція призначення
3	8	станція-джерело ізольована (застаріла)
3	9	доступ до мережі призначення адміністративно заборонений
3	10	доступ до станції-призначення адміністративно заборонений
3	11	для вибраного TOS мережа недосяжна
3	12	для вибраного TOS комп'ютер недосяжний
3	13	зв'язок адміністративно заборонений за допомогою фільтрування:
3	14	порушення приорітету комп'ютера
3	15	роз'єднання на основі приорітету
5	0	переспрямування для мережі
5	1	переспрямування для станції
5	2	переспрямування для мережі на основі TOS
5	3	переспрямування для станції на основі TOS
11	0	TTL рівне 0 під час транзитної передачі

Отже, дивлячись на наший пакет розпишемо поля даних:

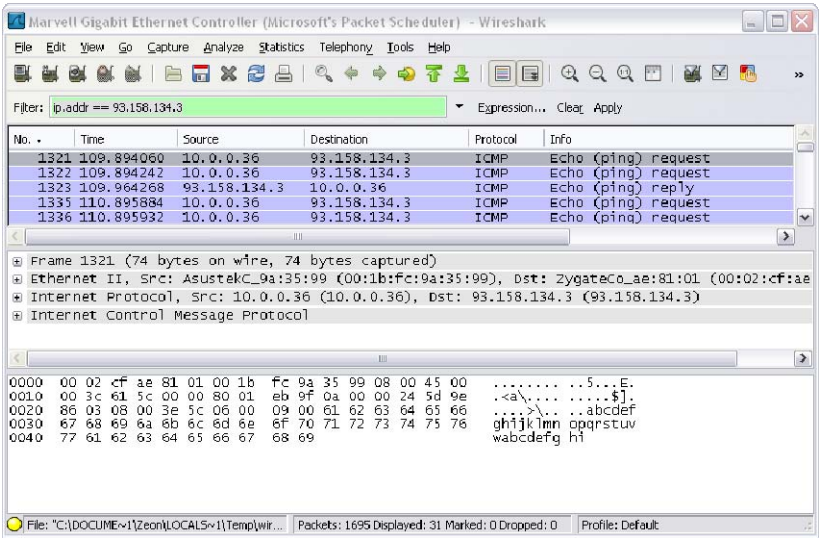

Рисунок 7.4. Обмін ICMP пакетами

Дані Ethernet кадру
00:02:cf:ae:81:01 – MAC адреса призначення
00:1b:fc:9a:35:99 - MAC адреса відправника
0800 – тип Езернет кадру
Дані IP дейтаграми
45 – тип IP дейтаграми
00 3c – довжина IP пакету, 60 байтів
61 5c – ідентифікація; 00 – прапорці; 80 – час життя пакету
01 – тип протоколу, в нашому випадку це ICMP
0xeb9f – контрольна сума
0a 00 00 24 - IP адреса відправника (10.0.0.36)
5d 9e 86 03 - IP адреса призначення (93.158.134.3)
Дані ICMP пакету
08 – Тип 8, дивлячись таблицю це exo (запит)
00 – код 0; 3e 5c – контрольна сума
06 00 – ідентифікатор
09 00 – номер послідовності
І далі 32 байта даних: abcdefghijklmnopqrstuvwabcdefghi

Аналогічно аналізуємо дані, що передаються по HTTP протоколу.

HTTP — протокол прикладного рівня, схожими на нього є FTP і SMTP. Обмін повідомленнями йде за звичайною схемою «запит-відповідь».

Кожен запит/відповідь складається з трьох частин:

- стартовий рядок;
- заголовка;
- тіло повідомлення, що містить дані запиту, запитаний ресурс або опис проблеми, якщо запит не виконано.

Обов'язковим мінімумом запиту є стартовий рядок. Починаючи з HTTP/1.1 обов'язковим став заголовок Host: (щоб розрізнити кілька доменів, які мають одну і ту же IP-адресу).

Запит

Стартові рядки розрізняються для запиту і відповіді. Рядок запиту виглядає так:

‹Метод› ‹URI› HTTP/‹Версія›

де в полі ‹Метод› можливі варіанти:

OPTIONS

Повертає методи HTTP, які підтримуються сервером. Цей метод може служити для визначення можливостей веб-сервера.

GET

Запрошує вміст вказаного ресурсу. Запитаний ресурс може приймати параметри (наприклад, пошукова система може приймати як параметр шуканий рядок). Вони передаються в рядку URI (наприклад:
http://www.example.net/resource?param1=value1&param2=value2).
Згідно зі стандартом HTTP, запити типу GET вважаються ідемпотентними — багатократне повторення одного і того ж запиту GET повинне приводити до однакових результатів (за умови, що сам ресурс не змінився за час між запитами). Це дозволяє кешувати відповіді на запити GET.

HEAD

Цей метод аналогічний методу GET, за винятком того, що у відповіді сервера відсутнє тіло. Це корисно для отримання мета-інформації, заданої в заголовках відповіді, без пересилання всього вмісту. Зокрема, клієнт чи проксі, перевіривши заголовок Last-Modified: (останній час модифікації), таким чином може переконатися, що сторінка на сервері не змінилася від часу попереднього запиту.

POST

Передає призначені для користувача дані (наприклад, з HTML-форми) заданому ресурсу. На відміну від методу GET, метод POST не вважається ідемпотентним, тобто багатократне повторення одних і тих же запитів POST може повертати різні результати

PUT

Завантажує вказаний ресурс на сервер.

DELETE

Видаляє вказаний ресурс.

TRACE

Повертає отриманий запит так, що клієнт може побачити, що проміжні сервери додають або змінюють в запиті.

CONNECT

Для використання разом з проксі-серверами, які можуть динамічно перемикатися в тунельний режим SSL.

В основному використовуються методи GET і POST.

Відповідь

Перший рядок відповіді виглядає так:
HTTP/‹Версія› ‹Код статусу› ‹Опис статусу›

Найтиповіші коди статусів:

- 200 OK — запит виконаний успішно;
- 403 Forbidden — доступ до запитаного ресурсу заборонений;
- 404 Not Found — запитаний ресурс не знайдений.

Отже, якщо успішно було отримано сторінку, то ми можемо здійснити пошук по відповіді «200 OK», а відповідно вище буде запит типу «GET».

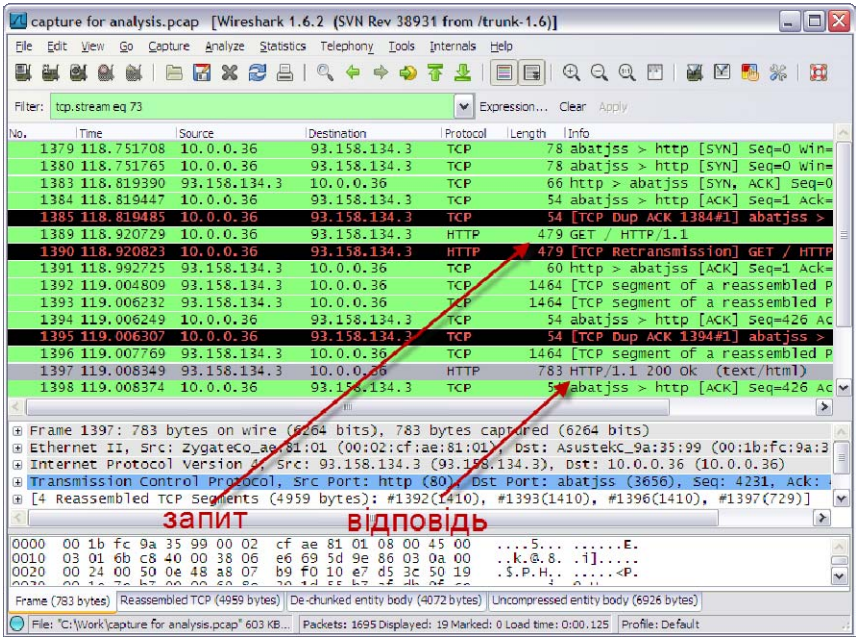

Рисунок 7.5. Обмін HTTP пакетами

Для того щоб переглянути діалог обміну даними між клієнтом і сервером в контекстному меню вибираємо Follow TCP Stream:

Запит:

GET / HTTP/1.1
User-Agent: Opera/9.80 (Windows NT 5.1; U; ru) Presto/2.10.229 Version/11.60
Host: www.ya.ru
Accept: text/html, application/xml;q=0.9, application/xhtml+xml, image/png, image/webp, image/jpeg, image/x-xbitmap, */*;q=0.1
Accept-Language: uk-UA,uk;q=0.9,en;q=0.8
Accept-Encoding: gzip, deflate
Cookie: yandex_gid=213; yp=1324779814.ygu.1; yandexuid=2114
Connection: Keep-Alive

Відповідь:

HTTP/1.1 200 Ok
Server: nginx
Date: Thu, 08 Dec 2011 13:45:20 GMT
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: close
Cache-Control: no-cache,no-store,max-age=0,must-revalidate
Expires: Thu, 08 Dec 2011 13:45:20 GMT
Last-Modified: Thu, 08 Dec 2011 13:45:20 GMT
P3P: policyref="/w3c/p3p.xml", CP="NON DSP ADM DEV PSD IVDo OUR IND STP PHY PRE NAV UNI"
Set-Cookie: yp=; path=/; expires=Mon, 10-Dec-2001 13:45:20 GMT
Set-Cookie: S=; path=/; expires=Mon, 10-Dec-2001 13:45:20 GMT
Set-Cookie: S=; domain=.www.ya.ru; path=/; expires=Mon, 10-Dec-2001 13:45:20 GMT
Set-Cookie: yandex_gid=143; domain=.www.ya.ru; path=/; expires=Sat, 07-Jan-2012 13:45:20 GMT
Set-Cookie: yp=1325943920.ygu.1; domain=.www.ya.ru; path=/; expires=Sun, 05-Dec-2021 13:45:20 GMT
X-XRDS-Location: http://openid.yandex.ru/server_xrds/
Content-Encoding: gzip
І відповідно дані веб сторінки закодовані у форматі gzip

Тепер подивимось як дані інкапсулюються в кадр Ethernet. В порівнянні з рис. 7.4 помітно що додався транспортний рівень моделі OSI (рівень TCP) і замість ICMP в нас присутній прикладний рівень HTTP, що і не дивно, адже тут вже використовувалась інша прикладна програма (веб браузер, а не утиліта ping).

На рис. 7.6 наведено внутрішню структуру HTTP запиту. В розгорнутому вигляді наведено сам запит. Помітно стартовий рядок, заголовок і тіло повідомлення.

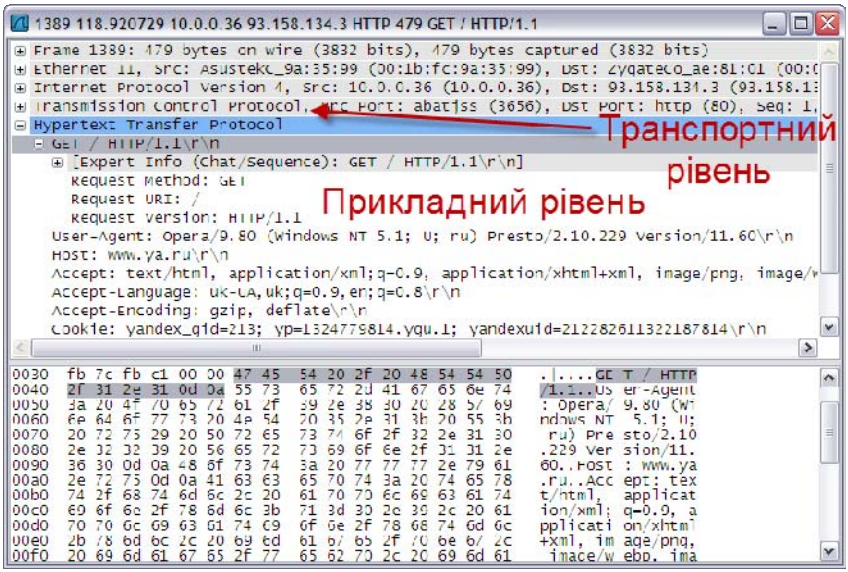

Рисунок 7.6. GET запит

Розглянемо структуру TCP пакету

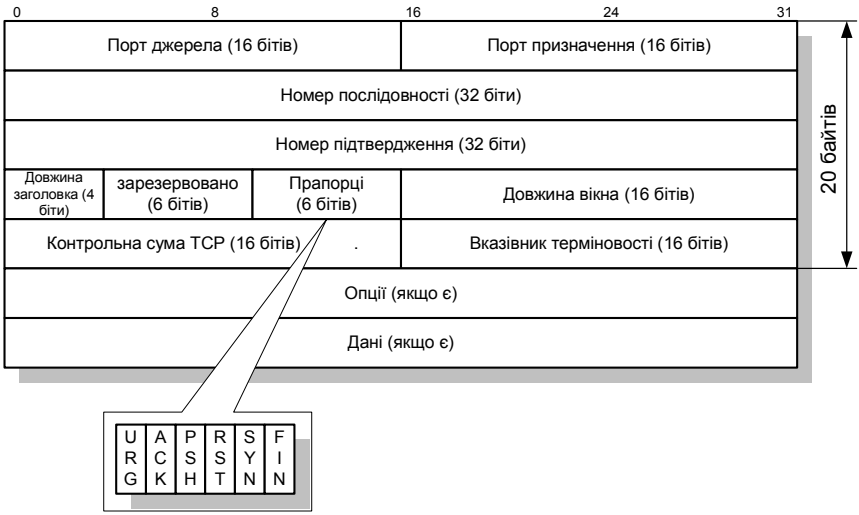

Рисунок 7.8. Структура TCP пакету

Значення полів в TCP заголовку:

Таблиця 7.4

Номер порта джерела	ідентифікатор процесу, що передав пакет;
Номер порта призначення	ідентифікатор процесу, для якого пакет призначений;
Номер послідовності (sequence number)	ідентифікатор першого байта в пакеті з потоку даних від передаючого TCP модуля до приймаючого TCP модуля, тобто кожен байт в потоці має свій номер. Він не обов'язково починається з 0. Оскільки TCP забезпечує дуплексний зв'язок, дані незалежно можуть передаватися в обох напрямках, кожна сторона веде свій незалежний ідентифікатор послідовності. Ідентифікатор послідовності наступного пакету збільшується на число рівне кількості байтів даних присутніх в попередньому пакеті;
Номер підтвердження (acknowledgment number)	номер наступного ідентифікатора послідовності (sequence) який очікується на приймальній стороні, тим самим підтверджує, що всі попередньо передані дані прийнято правильно;
Довжина заголовку (header length - HLEN)	довжина TCP заголовку в 32-х розрядних словах.
Резерв (reserved)	зарезервовано для майбутнього використання;
Коди прапорців (flags)	поле, що займає 6 біт які розглядаються як 6 прапорців, один або більше з яких можуть бути встановлені в 1. • URG - в даному TCP сегменті присутні термінові дані; • ACK - поле *acknowledgment* використано для підтвердження прийнятих даних; • PSH - передати прийняті TCP модулем дані пакету якомога швидше до процесу користувача;

	• RST - ініціатива в розриві або відмова встановлення зв'язку; • SYN - синхронізувати ідентифікатор послідовності, проводиться при встановленні зв'язку; • FIN - джерело пакета повідомляє що воно завершило передавати свою інформацію в даній TCP сесії, хоча не відмовляється від приймання даних від призначення; Коли в пакеті встановлений SYN, або FIN прапорець, то наступний ідентифікатор послідовності додатково збільшується на 1.
Розмір вікна (windows size)	кількість байт, яку може (погоджується) прийняти джерело TCP сегменту. Саме за допомогою *windows size* (збільшуючи чи зменшуючи його значення) проводиться контроль потоку інформації на обох сторонах зв'язку;
Контрольна сума (checksum)	контрольна сума сегменту, що охоплює як TCP заголовок так і дані. На відміну від UDP, в TCP використання контрольної суми завжди є обов'язковим;
Показник терміновості (urgent pointer)	це поле має зміст (аналізується) коли в пакеті встановлено URG прапорець. Число записане тут є позитивним зміщенням, яке треба додати до ідентифікатора послідовності пакету, для того щоб можна було визначити останній байт термінових даних в пакеті;
Опції (options)	додаткові поля заголовку, які розширюють можливості TCP. Вони мають змінну довжину але завжди вирівнюються на 32-х розрядне слово
Дані (data)	дані користувача, кількість байт може бути некратна 32-х розрядним словам.

Дані TCP сегменту:

0e 48 – номер порту джерела
00 50 – номер порту призначення
10 e7 d3 93 – номер послідовності
a8 07 a9 6a – номер підтвердження
50 – довжина заголовка
18 - прапорці
fb 7c – розмір вікна
fb c1 – контрольна сума

8. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

8.1. Базова література

1. Буров Є. Комп'ютерні мережі /За ред. В.Пасічніка. - 2-е вид. оновл. і доп.-Львів: БАК, 2003. - 584с.
2. Новиков Ю.В., Кондратенко СВ. Локальные сети: архитектура, алгоритмы, проектирование. - Москва: ЭКОМ, 2000. - 312с.
3. Пупена О.М., Ельперін І.В., Луцька Н.М., Ладанюк А.П. Промислові мережі та інтеграційні технології в автоматизованих системах. Навчальний посібник. – К.: Ліра-К, 2011. –500с.

8.2. Допоміжна література

1. Камер Дуглас Э. Сети TCP/IP. Т.1 Принципы, протоколы и структура: Пер. с англ. - 4-е изд. - Москва, Санкт-Петербург, Киев: Изд. дом "Вильяме", 2003. -880 с.
4. Хархаліс І.Р., Хархаліс Р.І. Телекомунікації, канали і мережі: Термінологічний словник. - Київ: ІСДО, 1995. - 52с.
5. Блэк Ю. Сети ЭВМ: Протоколы, стандарты, интерфейсы: Пер. с англ /Под ред. В.В.Василькова. - Москва: Мир, 1990. -506с.
6. Халсалл Ф. Передача данных, сети компьютеров и взаимосвязь открытых систем: Пер. с снгл. - Москва: Радио и связь, 1995. - 408с.

9. РЕСУРСИ

9.1. Інтернет ресурси

1. web-сайти періодичних видань - www.chip.ua, www.3dnews.ru, www.ixbt.com, www.cisco.com, www.habrahabr.ru;
2. офіційні web-сайти виробників апаратного та програмного забезпечення -www.intel.ru, www.amd.ru, www.asus.ru, www.nvidia.ru, www.samsung.ru, www.microsoft.ru, www.adobe.ru;
3. електронна бібліотека на сервері кафедри електротехніки і автоматики НУВГП за адресою: e-a\stud_doc\
4. IEEE 802.3™: ETHERNET - http://standards.ieee.org/about/get/802/802.3.html
5. RFC 793, TCP протокол - http://tools.ietf.org/html/rfc793

6. RFC 791, IP протокол - http://tools.ietf.org/html/rfc791
7. ModBus - http://www.modbus-ida.org/specs.php
8. WireShark - http://www.wireshark.org/
9. Code Project - http://www.codeproject.com/

9.2. Бібліотеки

Національного університету водного господарства та природокористування:
м. Рівне, вул. Приходька, 75, 2-й корп. НУВГП
Міжміський код 8(0362). тел. (0362) 22-25-39;

Рівненська обласна універсальна наукова бібліотека:
33000, м. Рівне, пл. Короленка, 6,
Міжміський код 8(0362). Тел./факс: 22-10-63
E-mail: library@libr.rv.ua

Рівненська обласна бібліотека для юнацтва:
33027, м. Рівне, вул. Київська, 18
Міжміський код 8(0362). Тел. 23-02-98
E-mail: molody@ukr.west.net
URL: http://libr.rv.ua

Міська бібліотека міста Рівне:
33028, м. Рівне, вул. Корольова, 2,
Міжміський код 8(0362) Тел. 5-42-86

Центральна бібліотека ЦБС міста Рівне:
33028, м. Рівне, вул. Київська, 44
Міжміський код 8(0362) Тел. 23-35-3
E-mail: library@icc.rv.ua
URL: http://www.library.rv.ua

ДОДАТОК А
ЗРАЗОК ОФОРМЛЕННЯ ТИТУЛЬНОЇ СТОРІНКИ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО ГОСПОДАРСТВА ТА ПРИРОДОКОРИСТУВАННЯ

Факультет прикладної математики та комп'ютерно-інтегрованих систем

Кафедра електротехніки та автоматики

КОНТРОЛЬНА РОБОТА
з дисципліни
„Комп'ютерні системи та мережі в АСКТП"

Виконав (ла):
студент (ка) групи
_____
Залікова книжка
№: _____
Перевірив:
_____

Дата здачі роботи на перевірку:

Рівне - 20__