Міністерство освіти і науки України

Національний університет водного господарства та природокористування

Кафедра автоматизації, електротехнічних та комп'ютерно-інтегрованих технологій

04-03-361М

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт
з навчальної дисципліни
«Інформаційні системи і технології в електроенергетиці»
для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Електроенергетика, електротехніка та електромеханіка» спеціальності 141 «Електроенергетика, електротехніка та електромеханіка» денної і заочної форм навчання

Рекомендовано науково-методичною радою з якості ННІАКОТ
Протокол № 8 від 19.06.2023 р.

Рівне – 2023

Методичні вказівки до виконання лабораторних робіт з навчальної дисципліни «Інформаційні системи і технології в електроенергетиці» для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Електроенергетика, електротехніка та електромеханіка» спеціальності 141 «Електроенергетика, електротехніка та електромеханіка» денної і заочної форм навчання [Електронне видання] / Наумчук О. М., Реут Д. Т. – Рівне : НУВГП, 2023. – 85 с.

Укладачі: Наумчук О. М., доцент кафедри автоматизації, електротехнічних та комп'ютерно-інтегрованих технологій;
Реут Д. Т., доцент кафедри автоматизації, електротех¬нічних та комп'ютерно-інтегрованих технологій.

Відповідальний за випуск: Древецький В. В., д.т.н., професор, завідувач кафедри автоматизації, електротехнічних та комп'ютерно-інтегрованих технологій.

Керівник освітньої програми «Електроенергетика, електротехніка та електромеханіка»: Василець С. В., д.т.н., професор кафедри автоматизації, електротехнічних та комп'ютерно-інтегрованих технологій.

Зміст

Вступ

Лабораторні роботи з дисципліни «Інформаційні системи і технології в електроенергетиці» призначені для формування та закріплення здобувачами вищої освіти сучасного рівня знань, умінь і навиків при розробці та використанні інформаційних систем і технологій, які застосовуються в енергетиці, електротехніці та електромеханіці. При виконанні лабораторних робіт студенти набудуть практичних навиків розробки та використання інформаційних систем та технологій, що дасть змогу використовувати набуті знання у професійній діяльності

У лабораторних роботах розглянуто особливості застосування інформаційних систем і технологій з використанням операційної системи Windows та Debian.

Лабораторна робота №1
Застосування технології розробки web-сторінок

Мета роботи

Ознайомлення з основними технологіями розробки web-сторінок: HTML, CSS, JavaScript. Навчитися розробляти найпростішої HTML-сторінки.

Теоретичні відомості

HTML (англ. HyperText Markup Language - мова розмітки гіпертекстових документів) — стандартна мова розмітки web-сторінок в Інтернеті. Більшість web-сторінок створюються за допомогою мови HTML (або XHTML). Документ HTML оброблюється браузером та відтворюється на екрані у зрозумілому для людини вигляді. HTML разом із каскадними таблицями стилів та вбудованими скриптами — це три основні технології побудови web-сторінок.

HTML надає розробнику засоби для:

- створення структурованого документу шляхом позначення структурного складу тексту: заголовки, абзаци, списки, таблиці, цитати, смислові блоки, меню та інше;
- отримання інформації з Інтернету через гіперпосилання;
- створення інтерактивних форм;
- включення зображень, звуку, відео, та інших об’єктів до тексту.

HTML-документ – це файл, який має розширенням *.htm (або *.html). Найпростіший HTML-документ матиме наступний вміст:

```
<!DOCTYPE html>        <!-- Тип документу -->
<html lang="uk">              <!-- Мова вмісту -->
  <head>   <!-- Початок заголовку документа -->
   <meta charset="utf-8"> <!--Кодування символів-->
       <title>Назва сторінки</title>
          <!--Тегом title задається назва сторінки-->
 </head>
 <body>   <!--Початок тіла документу-->
  <h1>Заголовок</h1>
  <p>
```

```
    Абзац тексту.
  </p>
 </body>
</html>
```

Для зручності читання введені додаткові відступи, однак у HTML-документі це не обов'язково, крім того більшість web-браузерів ігнорують символи кінця рядка і множинні пробіли. Тому такі відступи можна не використовувати.

Як видно з прикладу, вся інформація про форматування документа зосереджена у фрагментах розташованих між знаками "<" і ">". Такий фрагмент, наприклад, <html> називається міткою (англ. - tag).

Більшість HTML-міток є парними, тобто на кожну відкриваючу мітку виду *<tag>* є закриваюча мітка виду *</tag>* з тією ж назвою, але з додаванням символу "/". Мітки можна вводити? як великими так і малими літерами. Наприклад, мітки *<body>*, <BODY> і <Body> будуть сприйняті браузером однаково.

Багато міток, крім назви можуть мати *атрибути* - елементи, що дають додаткову інформацію про те, як web-браузер повинен обробити поточну мітку. Коментарі в HTML-документі виділяються наступним чином: <!-- коментар -->

При розробці web-сторінок зазвичай розділяється дизайн та вміст (контент) web-сторінки. Дизайн елементів описується за допомогою CSS (Cascading Style Sheets – каскадні таблиці стилів), де задаються шрифти, розміри, кольори блоків тексту, посилань, властивості фону тощо. CSS можуть бути включені безпосередньо в HTML-документ або підключатись до нього з файлу **.css* в заголовку (між тегами <head> та </head>) наступним чином:

```
<style type="text/css">
@import "example.css";
</style>
```

або

```
<link rel="stylesheet" href="example.css" type="text/css">
```

Винесення CSS у окремі файли дозволяє не включати в кожну сторінку опис спільних стилів, а лише посилання на css-файл. Якщо властивість елементу описана одразу декількома

способами, то найбільший пріоритет має задання її безпосередньо у тезі елементу, меншим пріоритетом – стиль, що застосовується за ідентифікатором, заданим в тезі, ще меншим – стиль, що застосовується до класу, псевдоелементу, всіх тегів заданого типу, а найнижчий пріоритет – стиль, заданий у зовнішньому CSS-файлі.

Розробкою стандартів HTML та CSS займається Консорціум WWW (англ. World Wide Web Consortium, W3C, http://www.w3.org/).

Для динамізації web-сторінок у них вставляються різноманітні скрипти, які, як правило, реалізовуються мовою JavaScript. Використаний скрипт виділяється тегами *<script></script>*, наприклад:

```
<script type="text/javascript">
  if (confirm("Текст вікна підтвердження!")) {
        alert("Натиснуто ОК");
      } else {
        alert("Натиснуто Cancel");
      }
</script>
```

Застосовуючи описані вище технології розробки web-сторінок можна розробляти та редагувати інформаційні матеріали. Вказані технології є основними, що застосовуються при розробці web-ресурсів, що дасть змогу набути основні поняття для подальшого застосовування у розробці складних інформаційних систем.

Програма роботи

1. Ознайомитися з особливостями застосування технологій розробки web-сторінок: HTML, CSS, JavaScript, що розглянуто у теоретичних відомостях

2. Відкрити існуючі web-сторінки та переглянути їх код. Знайти в коді основні HTML-теги, JavaScript та посилання на CSS.

3. Розробити просту web-сторінку зі статичним HTML-контентом.

Порядок виконання роботи

1. Запустити браузер (наприклад, Google Chrome, Mozilla Firefox, або інший) відкрити будь який web-сайт, наприклад http://nuwm.edu.ua, обрати в контекному меню «Початковий код сторінки».

2. Знайти у HTML-коді сторінки обов'язкові теги-елементи сторінки: *<!DOCTYPE html>, <head>, </head>, <body>, </body>, </html>*. Знайти тег, що повідомляє про мову вмісту сторінки (*<html lang="uk_UA"...>*), теги підклюення каскадних таблиць стилів (*<link rel="stylesheet"...>*), посилання на JavaScript'и (*<script type="text/javascript"...>*). Відкрити один з CSS-файлів, знайти підключення файлів з шрифтом тексту. Знайти фрагмент сторінки, який буде відображений користувачеві у випадку, якщо web-браузером не підтримується JavaScript (тег <noscript>).

3. Розробити просту web-сторінку на енергетичну тематику, наприклад про організацію (фірму) використовуючи розглянуту у теоретичних відомостях інформацію. У цій сторінці подати інформацію про організацію та додати дані описового характеру.

4. Створити фрагмент у сторінці, що взаємодіє з користувачем за допомогою JavaScript.

5. Результати роботи представити у вигляді звіту, в якому розмістити розроблений інформаційний ресурс у вигляді HTML-коду та у web -браузері.

***Вимоги до оформлення звіту*:**

- Звіт повинен містити:
- титульну сторінку;
- мету роботи;
- програму роботи;
- HTML-код сторінки з виділеними жирним тегами, передбаченими пукнтом 2 порядку виконання роботи;
- скріншот HTML-сторінки з власним прізвищем й ім'ям;
- скріншот та код сторінки, що використовує JavaScript;
- висновок.

Контрольні запитання

1. Що таке HTML?

2. Що дозволяють реалізувати засоби HTML?
3. Який тег виділяє заголовок тексту?
4. Для чого використовується CSS?
5. Як виділяється JavaScript у HTML-коді?
6. Яким тегом задається вміст, який відображається лише за відсутності підтримки JavaScript браузером?

Лабораторна робота №2
Розробка поведінкових програм інтегральних схем з використанням мови VHDL

Мета роботи

Навчитися розробляти програми поведінки і моделювання інтегральних схем за допомогою мови VHDL

Теоретичні відомості

Сучасні інформаційні технології передбачають використання мов програмування для розробки різних програмних продуктів. Для розробки складних електронних пристроїв використовують інтегральні схеми, які потребують розробки їх структури та поведінки. Однією з найбільш поширених мов розробки інтегральних систем та інших елементів електронних систем є мова VHDL (Very high speed integrated circuits Hardware Description Language). Ця мова була розроблена у 1980 році в результаті реалізації в США проекту по створенню надшвидкісних інтегральних схем. У 1987 році Інститутом Інженерів з Електрики та Електроніки (IEEE) вона була визнана стандартною для розробки електронних систем у США. Зараз VHDL є найбільш поширеною у світі мовою такого призначення, її застосовують при розробці багатьох пристроїв та систем.

VHDL— це мова опису апаратних засобів інтегральних схем та багатьох інших елементів. Основне її призначення це використання у якості інструментального засобу проектування програмованих логічних інтегральних схем (ПЛІС) та надвеликих інтегральних схем (НВІС). VHDL базується на принципі нисхідного проектування і є універсальною для проектування виробів цифрової електроніки.

На даний час мова VHDL використовується в якості міжнародного стандарту опису обчислювальних систем будь якого рівня складності (мікросхема, плата, блок, пристрій, ПК, комплекс, тощо). Вона може бути використана на всіх етапах розробки електронних систем: проектування, верифікація, синтез і тестування апаратури.

У мові VHDL важливе значення надається типам даних

оскільки вони дають змогу визначати особливість розробленого елементу чи пристрою. Ця мова використовується для розробки різноманітних варіантів апаратних проектів, тому засоби контролю типів даних мають велике значення. Такий підхід дає розробнику можливість представити сукупність ліній зв'язку (шин) у вигляді масиву бітів або цілого числа.

Кожен тип даних у VHDL має певний набір прийнятих значень, формат збереження та передачі даних і набір дозволених операцій. У цій мові визначена значна кількість різних типів даних, а також використовуються засоби для утворення користуватських типів даних. У VHDL використовується скалярні і агрегатні типи даних. Об'єкт, який віднесений до скалярного типу, розглядається як закінчена одиниця інформації. Агрегатні типи даних являють собою впорядковану сукупність скалярних одиниць, об'єднаних однаковим ім'ям. Крім того, типи даних поділяються на базові та користуватські.

Розроблений на мові VHDL проект являє собою опис явищ у дискретних системах. Такі явища представляються різними категоріями даних: константи, сигнали та змінні. Константи (constant) використовують для визначення сталих значень, які залишаються незмінними у всьому проекті. Сигнали (signal) – це інформація, що передається між елементами проекту і може бути представлена у вигляді вхідних та вихідних даних розроблюваного обладнання. Змінні (variable) – це інформаційна одиниця, яка використовується для опису внутрішніх операцій у елементах проекту, вони можуть синтезуватись у комірках пам'яті та змінюватися у ході виконання операцій.

Як і у будь якій мові програмування у мові VHDL використовуються оператори, які можна розділити на дві групи: послідовні і паралельні. Послідовні оператори (sequential statement) подібні до операторів багатьох мов програмування. Такі оператори обов'язково використовуються для опису послідовних процесів у частинах програм *process*, або в підпрограмах *procedure* чи *function* і виконуються послідовно один за одним у порядку запису. Паралельні оператори (concurrent statement) виконуються при будь яких змінах сигналів, що задані у вигляді вихідних даних. Такі оператори

виконуються не за послідовним принципом, а тоді, коли реалізація інших операторів програми створила умови для їхнього виконання. У мові VHDL також використовується багато інших операторів, про їх особливість та способи використання детально описані у спеціальній літературі.

Розглянемо особливість описів схем на мові VHDL. Опис будь якої схеми або її фрагменту складається з двох частин. Перша, називається сутністю (entity), містить опис зовнішнього інтерфейсу схеми (перелік входів, виходів та інших компонентів). Друга називається архітектурою (architecture) і містить опис, що визначає внутрішню будову та функціонування схеми.

Приклад опису сутності на мові VHDL:

ENTITY <ім'я сутності> IS

<вхідні і вихідні порти>;

[< фізичні та інші параметри>;]

END [[entity] ім'я_сутності]

Щоб написати повну VHDL-програму, потрібно описати всі вхідні і вихідні сигнали і визначити тип кожного з них. Опис сутності починається ключовими словами *ENTITY ... IS*. Він містить ім'я сутності і опис її зовнішніх виводів, що називаються портами. Крім того, опис може включати інші зовнішні параметри, такі як часові і температурні залежності тощо.

Завершується опис ключовим словом *END*, за яким (для зручності сприйняття тексту програми і додаткового контролю коректності блочної структури) бажано вказати реквізити блоку, що завершується. В даному випадку це *entity* та *ім'я сутності*.

Опис архітектури може виконуватись двома способами:

- як опис структури схеми (*structural*), тобто схеми з'єднань її складових елементів – схем нижчого ієрархічного рівня, аж до рівня з'єднання вентилів (*dataflow*);

- як опис поведінки схеми (*behavioral*).

Опис архітектури починається ключовими словами *ARCHITECTURE ... OF ... IS*, між якими вказується ім'я архітектури та ім'я сутності, якій вона відповідає. Далі вказуються декларації та тіло архітектури.

Приклад опису архітектури:

ARCHITECTURE <ім'я архітектури> OF <ім'я сутності> IS

<декларації типів, сигналів, констант та ін.>

BEGIN

<тіло архітектури>

END [[architecture] <ім'я-_архітектури>]

Розглянемо опис схеми на мові VHDL за допомогою прикладу опису роботи принципової схеми D-тригера, яка представленого на рис. 1

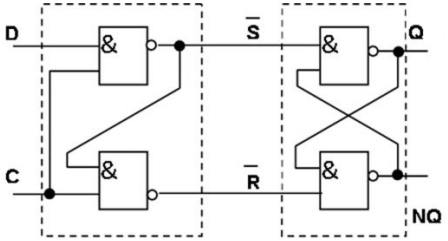


Рис. 1. Принципова схема синхронного D-тригера на логічних елементах І-НІ

Опис сутності D-тригера показаного на рис. 1 на мові VHDL:

```
ENTITY d_flipflop IS
PORT ( D, C: IN bit;
       Q, NQ: OUT bit );
END ENTITY d_flipflop;
```

Цей опис свідчить, що схема на ім'я *d_flipflop* має чотири зовнішні виводи (порти), з них два входи *(IN): D* та *C*; і два виходи *(OUT): Q* та *NQ*. Усі порти належать до типу *bit*, тобто сигнал на їх виводах може набувати значень "0" і "1".

Опису архітектури D-тригера здійснюється через опис його поведінки. Даний D-тригер працює так, що при C = 1, сигнал на виході Q повторює значення сигналу на вході D, а при C = 0, тригер зберігає попередній стан і сигнал на виході Q не змінюється. Сигнал на виході NQ є інверсією сигналу на виході Q.

Поведінку тригера можна описати так:

```
ARCHITECTURE behaviour OF d_flipflop IS
BEGIN
```

```
PROCESS (C,D)
BEGIN
IF  C = '1'  THEN
    Q <= D  AFTER 5 ns;
    NQ <= not D AFTER 5 ns;
END IF;
END PROCESS;
END ARCHITECTURE behaviour;
```

Оператор *PROCESS (C,D)* вказує на те, що робота схеми описується процесом, де зміна сигналів на виходах можлива лише внаслідок зміни вхідних сигналів *C* або *D*. Оператор *IF* визначає, що при *C = '1'*, сигнал *Q* набуває значення *D*, а *NQ* – інверсії *D*. Ключове слово *AFTER* вказує на те, що вказані зміни значень на виходах *Q*, *NQ* відбувається не миттєво, а з затримкою на 5 наносекунд.

Цей приклад, дає змогу уявити особливість використання типів даних, операторів та способів розробки простих проектів інтегральних схем на мові VHDL. Використання цієї мови сприятиме здобуттю практичних навиків у розробці інтегральних схем та опису їх поведінки.

Програма роботи

1. Розглянути особливість використання мови VHDL для розробки та моделювання поведінки інтегральних схем, що описаний в теоретичних відомостях.
2. Навчитися розробляти поведінкові програми інтегральних схем.

Порядок виконання роботи

1. Розглянути приклад розробки поведінкової програми з використанням мови VHDL, що описаний у теоретичних відомостях

2. Проаналізувати поведінку інтегральної схеми за варіантом схеми, яку надасть викладач та розробити програму на основі мови VHDL. При цьому можна використовувати середовище розробки VHDL (за наявності), наприклад *EDAplajground* (https://edaplayground.com/home), Online VHDL Testbench Template Generator (https://vhdl.lapinoo.net/testbench), або будь-

який текстовий редактор.

3. Описати алгоритм функціонування розробленої програми.

4. Результати виконання оформити у вигляді звіту на стандартних аркушах формату А4.

Вимоги до оформлення звіту

Звіт повинен містити:

- титульну сторінку;
- мету роботи;
- програму та порядок виконання роботи;
- виконану програму та її опис з коментарями;
- висновок.

Контрольні запитання

1. Які вимоги використання мов проектування?
2. Що таке VHDL?
3. Які типи даних використовуються у мові VHDL?
4. Як виглядає розроблений на мові VHDL проект?
5. Які оператори використовуються у мові VHDL?
6. Яка особливість описів схем на мові VHDL?
7. Наведіть приклад опису вхідних та вихідних сигналів на мові VHDL?
8. Наведіть приклад опису архітектури на мові VHDL?

Лабораторна робота №3
Розробка та побудова локальних комп'ютерних мереж

Мета роботи

Ознайомитись з будовою і призначенням основних елементів для побудови локальних комп'ютерних мереж типу Ethernet. Навчитися створювати Ethernet мережу, налаштовувати TCP/IP параметри та тестувати з'єднання використовуючи стандартні утиліти ОС Windows

Теоретичні відомості

Ethernet — сімейство базових технологій локальних обчислювальних (комп'ютерних) мереж з комутацією пакетів даних. Один з найбільш поширених методів доступу до передавального середовища в Ethernet - CSMA/CD (множинний доступ з контролем несучої частоти та виявленням колізій). Цей метод використовується у класичній технології Ethernet і дає змогу в певний момент часу здійснювати лише один сеанс передачі даних логічного сегменту мережі. Якщо відбувається два і більше сеанси передачі одночасно, то виникає колізія, яка фіксується ініціатором передачі. При цьому, процес зупиняється, очікується закінчення поточного сеансу передачі і тільки після цього станція-відправник знову намагається повторити передачу.

Компоненти обладнання мереж Ethernet. Карта мережевого інтерфейсу (Network Interface Card - NIC), яку також називають мережевим адаптером або мережевою картою - це пристрій, який здійснює фізичне під'єднання комп'ютера до мережі, тобто забезпечує фізичне сполучення між мережевим кабелем і внутрішньою шиною комп'ютера. У більшості випадків ця карта встановлюється безпосередньо в шину розширення комп'ютера (PCI, ISA, PCI Express та ін.). В окремих випадках карта може бути частиною окремого пристрою, до якого комп'ютер під'єднаний через паралельне або послідовне сполучення.

Карта мережевого інтерфейсу отримує дані від персонального комп'ютера, перетворює їх до відповідного формату і пересилає через кабельну систему до іншої мережевої

карти, яка в свою чергу, приймає дані, переводить їх у форму прийнятну для даного комп'ютера і передає далі на обробку.

Повторювач (repeater) - це пристрій, який отримує електричний або оптичний сигнал з кабелю через відповідний інтерфейс, регенерує його і передає в кабель через інший інтерфейс. Завданням повторювача є пересилання будь-якого вхідного сигналу до всіх інших портів без модифікації і затримки. Також повторювач пересилає сигнали колізій, глушіння, шумів та ін. Повторювачі можуть мати два або більше порти. Прикладом багатопортового повторювача у мережі Ethernet є хаб. Порти повторювачів не повинні бути ідентичними, вони можуть передавати сигнали від входу звичайного дротового кабеля до входу оптоволоконного кабеля в межах однієї і тієїж топології. Наприклад, таким чином можливо переслати сигнал від технології 10Base-T до 10Base-F, але не можливо побудувати повторювач для мереж різних типів, наприклад, для Ethernet і TokenRing. Натомість для здійснення таких сполучень використовують пристрої, які називають *мостами*. Повторювачі відносять до пристроїв 1-го рівня (фізичного рівня) моделі OSI.

Хаби Ethernet. Виготівники мережевого обладнання впровадили пристрої, які забезпечують можливість передавати дані через багато портів мережі Ethernet. Ці пристрої відомі як хаби, які у своєму найпростішому варіанті є звичайними багатопортовими повторювачами. Хаби можна об'єднувати або каскадувати (з використанням ієрархічної схеми), що збільшує кількість портів на окремому сегменті мережі.

Сучасні хаби - це значно складніші пристрої, які мають ряд властивостей, подібних до комутаторів або мостів, що значно покращує їх можливості при адмініструванні мережі. Зокрема, наявність внутрішніх комутаторів у хабі дозволяє розділити його порти на декілька груп. Передавання пакетів даних між цими групами портів може супроводжуватися буферизацією, фільтрацією, контролем правильності пакетів та відкиданням пошкоджених пакетів тощо. Це суттєво покращує експлуатаційні характеристики мережі за рахунок зменшення трафіку в окремих сегментах, більш ефективного використання наявної ширини смуги пропускання. Хаб може мати

адресований порт, що дає змогу застосовувати дистанційне управління хабом з використанням відповідного протоколу високого рівня (наприклад, SNMP).

Середовище передачі даних. Коаксіальний кабель (рис. 1) донедавна був дуже популярний, що пов'язано з його високою перешкодозахищеністю (завдяки металевому обплетенню), а також більшими допустимими відстанями передачі (до кілометра). До цього кабелю складніше під'єднатися з метою несанкціонованого прослуховування мережі, а також він менше продукує зовнішніх електромагнітних випромінювань. Кабельні системи на основі коаксіального кабелю раніше застосовувалась досить широко, однак монтаж і ремонт коаксіального кабелю та його елементів значно складніший від інших видів кабелів, тому такі типи кабелів дедалі рідше використовуються для побудови нових локальних комп'ютерних мереж.

Рис. 1. Коаксіальний кабель

Скручена пара. Існує декілька категорій цього типу кабелю, які позначаються CAT1...CAT7 та визначають ефективний пропускний частотний діапазон. Кабель вищої категорії зазвичай містить більше пар проводів і кожна пара має більше витків на одиницю довжини. Категорії неекранованої скрученої пари визначаються стандартами ANSI/EIA/TIA 568 (Стандарт телекомунікаційних кабельних систем комерційних приміщень) та міжнародним стандартом ISO 11801:

CAT1 використовується тільки для передачі голосу або даних за допомогою модема (смуга частот 0,1 МГц).

CAT2 використовується в мережах Token ring і Arcnet, тип кабеля на 2-і пари провідників, передача даних відбувається на швидкостях до 4 Мбіт/с (смуга частот 1 МГц).

CAT3 використовується для телефонних і локальних мереж 10BASE-T і Token ring, передача даних відбувається на

швидкостях до 10 Мбіт/с або 100 МБіт/с за технологією 100BASE-T4 на відстані не більше 100 метрів, тип кабеля на 4-і пари провідників (смуга частот 16 МГц).

CAT4 використовується в мережах Token ring за технологією 10BASE-T і 100BASE-T4, передача даних відбувається на швидкостях до 16 Мбіт/с по одній парі (смуга частот 20 МГц), кабель складається з 4 скручених пар провідників.

CAT5 використовується при побудові локальних мереж 100BASE-TX і для телефонних ліній, підтримує швидкість передачі даних до 100 Мбіт/с, кабель складається з 4 скручених пар провідників (смуга частот 100 МГц).

CAT5e удосконалена категорія 5, підтримує швидкість передачі даних до 100 Мбіт/с. Кабелі цієї категорії є найбільш розповсюдженими і використовується для побудови локальних комп'ютерних мереж (смуга частот 125 МГц).

CAT6 застосовується в мережах Fast Ethernet та Gigabit Ethernet, складається з 4 пар провідників і здатен передавати дані на швидкості до 1000 Мбіт/с (смуга частот 250 МГц).

CAT6a застосовується в мережах Ethernet, складається з 4 пар провідників і здатний передавати дані на швидкості до 10 Гбіт/с і планується використовувати для обміну даними на швидкості до 40 Гбіт/с (смуга частот 500 МГц).

Порівняльні характеристики найбільш поширених кабельних систем. Зараз найбільшого поширення здобули декілька типів структурованих кабельних систем, які відрізняються характеристиками та особливостями монтажу:

1. Кабельна система на основі неекранованого кабелю типу скручена пара UTP з опором 100 Ом (рис. 2):

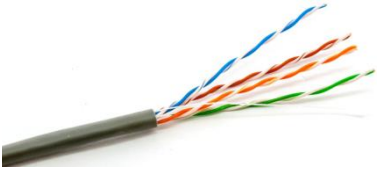


Рис. 2. Неекранований кабель скручена пара UTP- типу

Переваги кабелів UTP:

• багатоваріантність застосування, висока пропускна здатність при невеликій вартості;

• можуть передавати дані зі швидкістю до 100 Мб/с і підтримують сучасні технології передачі (Fast Ethernet, ATM та ін.);

• простота монтажу, невеликий діаметр та вага.

Недоліки кабелів UTP:

• невелика захищеність від механічних пошкоджень;

• чутливість до завад викликаних зовнішніми джерелами електромагнітних полів.

2. Кабельна система на основі екранованого кабелю типу скручена пара STP з опором 150 Ом (рис. 3).

Рис. 3. Екранований кабель скручена пара STP- типу

Переваги кабелів STP:

• забезпечують захист сигналів від впливу зовнішнього середовища та сигналів в інших кабелях;

• зменшують захист від впливу комунікаційних систем зосереджених в одному передавальному середовищі.

Недоліки кабелів STP:

• більша вартість у порівнянні з іншими типами кабелів типу скручена пара;

• необхідно забезпечення заземлення.

3. Кабельні системи на основі волоконно-оптичних кабелів (рис. 4).

Рис. 4. Волоконно-оптичний одномодовий 2-волоконний кабель

Переваги оптоволоконних кабелів:

- велика ширина смуги передавання;
- відносно невелика вартість;
- низьке енергоспоживання;
- нечутливість до електромагнітних завад.

Недоліки оптоволоконних кабелів:

- складність монтажу та необхідність використання спеціальних пристроїв;
- вища вартість кабелю та монтажу у порівнянні з іншими типами.

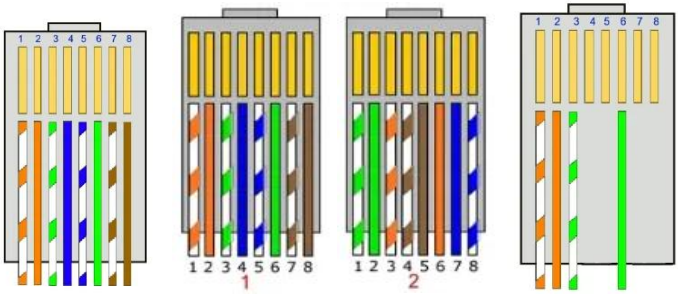
З розглянутих вище кабельних систем найбільшого поширення здобули кабельні системи на основі скрученої пари у яких використовуються роз'єми типу RJ-9, RJ-11, RJ-12, RJ-14, RJ-21, RJ-45, RJ-45S, RJ-50. Серед них найбільше поширеними типами роз'ємів є: RJ-11, RJ-12 та RJ-45. Корпус RJ - конектора, як правило, складається з прозорого пластика, усередині якого кілька ножів-контактів, покритих золотим напиленням. У комп'ютерних мережах (4-жильна скручена пара) за технологіями 10BASE-T, 100BASE-T та 1000BASE-TX зазвичай використовується стандартний конектор RJ-45, що має з'єднання 8P8C (8 Position 8 Contact). Восьмиконтактні модульні з'єднувачі RJ-45 (рис. 5) мають вісім контактів та поділяються на: екрановані і неекрановані, зі вставкою і без, для круглого і плоского, для одножильного і багатожильного кабелю. У новій невикористаній вилці контакти виходять за межі корпуса. У процесі обтискання вони будуть втоплені всередину корпуса, проріжуть ізоляцію проводу і встромляться в жили провідників.

Рис. 5. Зовнішній вигляд восьмиконтактного конектора RJ-45

Для виконання монтажу кабелів типу скручена пара використовується спеціальний обтискний інструмент (рис. 6). Спочатку знімають верхню ізоляцію з кабелю, потім

розплітають та вирівнюють провідники за певною схемою (пряма або перехресна). Пряма схема (рис. 7, а) використовується для підключення пристроїв різних типів, наприклад: ПК - Switch, Switch – Router, або Router - ПК. Перехресна схема (рис. 7, б) використовується для підключення однотипних пристроїв, наприклад: ПК - ПК, Switch - Switch, Router - Router і т. д. Деколи при побудові локальної мережі для швидкості передачі даних до 100 Мб/с та для економії використовують 4-х провідний кабель. Монтаж такого кабелю здійснюється подібно, як і 8-и жильного за виключенням того, що у конекторі задіяні тільки 1, 2, 3 і 6 жили, як показано на рис. 7, в. Далі провідники за відповідно обраною схемою вставляють у конектор так, щоб всі жили розташувалися у своїх напрямних каналах, а зовнішня ізоляція кабелю потрапила під планку затискання конектора. Після цього проводиться обтискання кабелю.

Рис. 6. Обтисний інструмент для кабелю скручена пара



а) б) в)

Рис. 7. Способи розташування провідників у восьмиконтактному з’єднувачі RJ-45: а) пряма схема, б) перехресна схема, в) пряма схема для 4-х жильного кабелю

Діагностика мережі в операційних системах Windows. До складу операційної системи Windows включено ряд комунікаційних утиліт, які дають можливість перевірити працездатність з'єднання з віддаленим вузлом (*ping*), прослідкувати маршрут проходження пакетів до віддаленого вузла (*tracert*) та ін. Для їх запуску достатньо перейти в режим командного рядка *(Пуск → Програми → Система Windows → Командний рядок)* і ввести з клавіатури у відповідь на запрошення назву утиліти з відповідними параметрами. Для зупинки виконання якоїсь команди/утиліти в командному рядку необхідно натиснути комбінацію клавіш Ctrl+C.

При підключенні до мережі Internet кожному пристрою надається власне символьне ім'я, яке надається доменною службою імен — *DNS (Domain Name System).* Вона являє собою розподілену базу даних, в якій підтримується ієрархічна система символьних імен. База даних про відповідність символьних імен і IP-адрес розподілена по DNS-серверах, які розташовані у різних вузлах мережі Internet. Кожного разу, коли у прикладній програмі виникає необхідність перетворити ім'я в IP-адресу відбувається її звернення до сервера DNS.

Операційна система Windows містить набір утиліт, що застосовується для діагностики мережі. Основними завданнями цих утиліт є:

• визначення параметрів і характеристик мережі;

• визначення працездатності мережі;

• у разі неправильного функціонування мережі - локалізація сегмента або сервісу, що викликають несправність.

Головними параметрами мережних підключень є їх канальні і мережні адреси та інші параметри, що впливають на роботу мережного рівня. Кожен комп'ютер в мережі Internet (їх прийнято називати хостами) має адреси двох рівнів: канального і мережного. Канальна адреса хоста визначається технологією, за допомогою якої здійснюється його підключення до Internet. Для ПК, що входять в локальні мережі Ethernet, це так звана MAC- адреса (Media Access Control - управління доступом до середовища) мережного адаптеру, який призначається виробником обладнання і є унікальним. Для існуючих технологій локальних мереж MAC - адреса має 48 - розрядний

формат (6 байт), але зазвичай MAC- адреси представляються в 16- розрядній системі, наприклад, 00-E0-4C-78-23-FD:

• перший біт вказує: для одиночного (0) або групового (1) адресата

призначений кадр;

• наступний біт вказує, чи є MAC- адреса глобально (0) або локально (1) адміністрованою;

• наступні 22 біта є ідентифікатором фірми виробника;

• інші 3 байти призначаються самим виробником.

У якості мережної адреси хосту в мережі Internet використовується IP- адреса (Internet Protocol Address), яка характеризує не окремий комп'ютер або маршрутизатор, а одне мережне з'єднання. У мережі Internet застосовується глобальна унікальність адрес, що забезпечується рекомендаціями спеціального підрозділу InterNIC (Network Information Center). Провайдери послуг Internet отримують діапазони адрес у підрозділів InterNIC, а потім розподіляють їх між своїми абонентами. У разі ізольованої від Internet локальної мережі унікальність мережної адреси потрібна лише в її межах, при цьому IP-адреси повинні вибиратися адміністратором із спеціально зарезервованих для таких мереж блоків «закритих» адрес.

Після розробки мережі, або в процесі її використання необхідно використовувати *системні утиліти мережної діагностики***.** Утиліта *ipconfig* призначена для перевірки правильності конфігурації TCP/IP для операційної системи Windows. Вона виводить значення для поточної конфігурації стека TCP/IP: MAC- і IP- адресу, маску підмережі, адресу шлюзу за замовчуванням, адреси серверів WINS (Windows Internet Naming Service) і DNS, використання DHCP. При усуненні несправностей в мережі TCP/IP слід спочатку перевірити правильність конфігурації за допомогою утиліти *ipconfig*.

Синтаксис утиліти: *ipconfig [/ all] [/ renew [adapter]] [/ release] [adapter]* (тут і далі в квадратних дужках вказані необов'язкові параметри):

• *all* видає весь список параметрів, без цього ключа відображається тільки IP-адреса, маска і шлюз за умовчанням;

• *renew* [*adapter*] оновлює параметри конфігурації DHCP для зазначеного мережного адаптера з ім'ям adapter;

• *release* [*adapter*] звільняє виділену DHCP IP -адресу.

Таким чином, утиліта *ipconfig* (рис. 8) дозволяє з'ясувати, чи ініціалізована конфігурація і чи не дублюються IP- адреси:

• якщо конфігурація ініціалізована, то з'являються IP-адреса, маска, шлюз;

• якщо IP- адреси дублюються, то маска мережі буде 0.0.0.0;

• якщо при використанні DHCP комп'ютер не зміг отримати IP- адресу, то значення буде 0.0.0.0 .

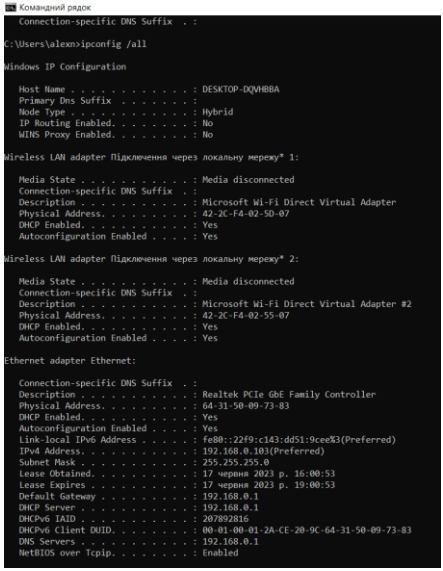


Рис. 8. Відображення встановлених на комп'ютері мережних конфігурацій утилітою *ipconfig*

Утиліта *ping* (Packet Internet Grouper) використовується для перевірки конфігурування TCP/IP та діагностики помилок з'єднання. Вона визначає доступність і функціонування конкретного хоста (вузла) будь якого мережного пристрою, що обмінюється інформацією з іншими мережними пристроями. Команда *ping* перевіряє з'єднання з віддаленим хостом шляхом відправлення до нього *ехо-пакетів* протоколу ICMP (Internet

Control Message Protocol) і прослуховування *ехо-відповідей*. Ping виводить кількість переданих і прийнятих пакетів. Кожен прийнятий пакет перевіряється відповідно з переданим повідомленням. Якщо зв'язок між хостами поганий, то з цих повідомлень визначають скільки пакетів втрачено.

За замовчуванням передаються чотири ехо-пакета довжиною 32 байта, що представляють собою послідовність спеціальних символів. Атрибути утиліти *ping* дозволяють змінити розмір і кількість пакетів, вказати чи слід записувати маршрут, яку тривалість часу встановлювати, чи можна фрагментувати пакети і т.д. При отриманні відповіді визначається, за який час (у мілісекундах) відправлений пакет доходить до віддаленого хоста і повертається назад. Оскільки, значення за замовчуванням для очікування відгуку складає 1 с, то всі значення даного поля будуть менше 1000 мс.

При використанні утиліти *ping* необхідно врахувати те, що затримка, визначена утилітою, викликана не тільки пропускною здатністю каналу передачі даних, а й завантаженістю ПК. Деякі сервери в цілях безпеки можуть не відправляти ехо-відповіді, так як з утиліти *ping* може починатися хакерська атака.

Утиліта *ping* можна використовувати для тестування як з доменним іменем хоста, так і з IP -адресою. Якщо *ping* з IP-адресою виконалася успішно, а з доменним іменем - невдало, це означає, що проблема полягає в розпізнаванні відповідності адреси та імені, а не в мережному з'єднанні.

Синтаксис команди: *ping [-t], [-a], [-n count], [-llength], [-f], [-i ttl], [-v tos], [-r count], [-s count], [-j host-list], [-k host-list], [-w timeout], [destination-list]*

Параметри команди:

• *-t* виконує команду ping до переривання (Ctrl-Break - переглянути статистику і продовжити, Ctrl-C - перервати виконання команди);

• *-a* дозволяє визначити доменне ім'я віддаленого комп'ютера за його IP-адресою;

• *-n count* посилає кількість пакетів Echo, вказане параметром count (за

замовчуванням передається чотири запиту);

• *-l length* посилає пакети довжиною *length* байт (максимальна довжина 8192 байти);

• *-f* посилає пакет з встановленим прапорцем «Не фрагментувати», що забороняє фрагментованість пакета на транзитних маршрутизаторах;

• *-i ttl* встановлює час існування пакета на величину *ttl* (кожен маршрутизатор зменшує *ttl* на одиницю, тобто час існування є лічильником пройдених маршрутизаторів (хопів));

• *-v tos* встановлює значення поля «сервіс», що задає пріоритет обробки пакета;

• *-r count* записує шлях вихідного пакету і пакету, що повертається в полі запису шляху, *count* - від 1 до 9 хостів;

• *-s count* задає максимально можливу кількість переходів з однієї підмережі в іншу (хопів);

• *-j host-list* направляє пакети за допомогою списку хостів, визначеного параметром *host -list*). Максимальна кількість хостів дорівнює 9;

• *-k host-list* направляє пакети через список хостів, визначений у *host-list*, причому зазначені хости не можуть бути розділені проміжними маршрутизаторами (жорстка статична маршрутизація);

• *-w timeout* вказує час очікування timeout відповіді від віддаленого хоста в мілісекундах (за замовчуванням - 1с);

• *-destination-list* вказує віддалений вузол, до якого треба направити пакети ping , може бути ім’ям хоста або IP -адресою ПК.

Найчастіше у форматі команди *ping* використовуються опції *-t* та *-n*. Приклад виконання утиліти *ping* наведено на рис. 9.

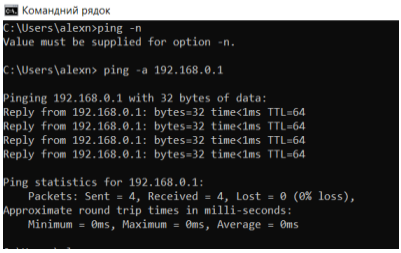


Рис. 9. Приклад використання утиліти *ping -a*

Програма роботи.

1. Ознайомитись з будовою і призначенням основних елементів для побудови локальних мереж типу Ethernet.

2. Навчитися створювати Ethernet мережу, налаштовувати TCP/IP параметри та тестувати з'єднання використовуючи утиліти ОС Windows.

Порядок виконання роботи.

1. Розглянути види кабелів, розташування провідників у восьмиконтактному з'єднувачі RJ-45 (пряма та перехресна схема) для 8-и та 4-и жильного кабелю за теоретичним матеріалом.

2. Виконати обтискання кабелю скручена пара за наступним алгоритмом:

а) зняти зовнішню ізоляцію з кабелю мінімум на 20 мм;

б) розділити провідники та розташувати їх в один рядок сформувавши у послідовності, яка вказана на рис. 7;

в) за допомогою спеціального інструменту (див. рис. 6) підрізаємо провідники на відстань близько 14 мм від краю оболонки кабелю;

г) розташовуємо провідники в роз'ємі RJ-45, спочатку восьмижильний кабель, а потім чотирижильний так, щоб вони увійшли у відповідні канали, а оболонка кабелю заходила в роз'єм приблизно на 6 мм. При цьому, потрібно дотримуватися схеми розташування, яка зазначена на рис. 7;

д) за допомогою спеціального інструменту виконуємо обтискання втискаючи ножі-контакти роз'єму RJ-45 всередину корпусу, тим самим прорізуючи оболонку провідників та створюючи електричний контакт;

е) візуально перевіряємо правильність обтискання та за допомогою спеціального тестера.

3. Підключити обтиснений кабель до мережевого пристрою та візуально визначаємо правильність його роботи.

4. Визначити правильність підключення до мережевого обладнання (комп'ютера чи комутатора/концентратора) шляхом перевірки налаштування TCP/IP параметрів.

5. Перевірити налаштування параметрів мережі. Для налаштувань TCP/IP параметрів у ОС Windows перейдіть в

«*Налаштування мережі та інтернету*», або перейти в меню: «*Панель керування → Мережа й Інтернет → Центр мережевих підключень і спільного доступу → Стан Ethernet → Ethernet - властивості*». Після цього ОС відразу запропонує увійти у відповідне меню. При вдалому підключенні у цій вкладці з'явиться значок підключення до відповідної мережі  , якщо ж підключення невдале, тоді на зображенні буде відображений хрестик і зазначено про помилку  . При цьому, необхідно перевірити правильність з'єднання та правильність прокладання кабелю.

6. Виконуємо ручне налаштування підключення. Для цього виберіть властивості підключення та перейдіть до налаштувань протоколу TCP/IP (рис. 10). Зазвичай у сучасних операційних системах властивості протоколів налаштовуються автоматично. Для ручного налаштування у властивостях протоколу вибираємо отримання IP адреси вручну. Автоматичне отримання можливе при наявності DHCP сервера в мережі (наприклад, це може бути роутер з функцією DHCP). Для задання адреси вручну необхідно її вказати наприклад, 192.168.0.1, а маску підмережі - 255.255.255.0.

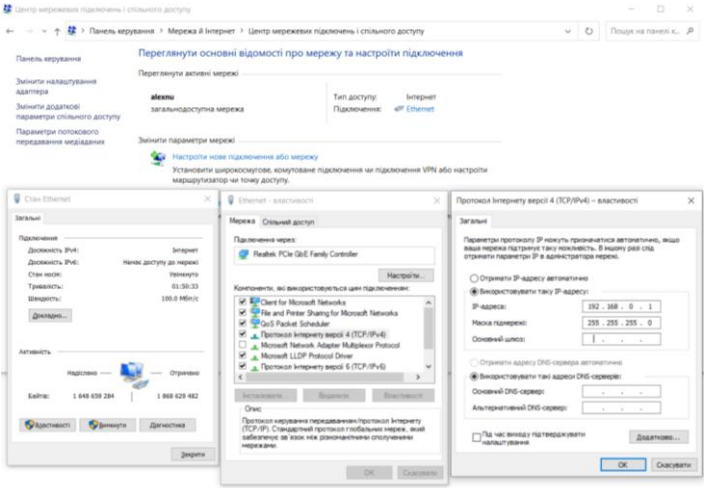


Рис. 10. Приклад послідовності налаштування підключення за протоколом TCP/IP

7. Виконуємо тестування з'єднання ПК з мережею. Для цього використаємо утиліти *Ipconfig* та *Ping* вказавши їх атрибути так, як це описано у теоретичних відомостях.

8. За результатами тестування проаналізувати отримані дані та оцінити ефективність функціонування мережі. При потребі необхідно змінити налаштування параметрів TCP/IP і виконати тестування повторно.

9. Результати виконання оформити у вигляді звіту на стандартних аркушах формату А4.

***Вимоги до оформлення звіту*:**

Звіт повинен містити:

- титульну сторінку;
- мету роботи;
- програму роботи;
- фотографії та опис процесу виконання, який описаний в п. 2, 3;
- скріншоти та опис процесу налаштувань TCP/IP параметрів у ОС Windows (п. 4-7);
- скріншоти та опис процесу тестування з'єднання (п. 8, 9);.
- висновок.

Контрольні запитання

1. Що таке Ethernet?
2. Які ви знаєте види кабелів?
3. З чого складається скручена пара?
4. Які є варіанти обтискання скрученої пари?
5. Які категорії кабелів найбільш розповсюджені?
6. Які утиліти використовуються для тестування мережі в ОС Windows.
7. Що таке ARP?
8. Яку інформацію можна отримати з аналізу IP-адреси? Що таке MAC адреса?
9. Яку функцію в мережі виконують DNS-сервери?
10. Як протестувати з'єднання з віддаленим хостом?
11. Як визначити доменне ім'я хосту з відомою IP-адресою?
12. Як визначити IP-адресу хосту із зазначеним доменним ім'ям?

Лабораторна робота №4
Розробка та використання безпроводових мереж

Мета роботи

Ознайомитись з будовою і призначенням основних складових безпроводової мережі Wi-Fi та навчитися налаштовувати і використовувати Wi-Fi-маршрутизатори.

Теоретичні відомості

Технологія Wi-Fi (IEEE 802.11) - це сімейство технологій безпроводового передавання у радіодіапазоні. Сімейство стандартів IEEE 802.11 визначає фізичний та канальний рівень протоколів передавання, вони відрізняються фізичною реалізацією та швидкістю. Серед існуючих Wi-Fi мереж найпоширенішим є стандарт IEEE 802.11b, він дає змогу передавати дані зі швидкістю 11 Мбіт/с на відстань до декількох кілометрів, використовуючи смугу частот 2.4 Гц. На основі IEEE 802.11b будують безпроводові локальні мережі Wireless LAN (WLAN). Мережа Wi-Fi може працювати в двопунктових сполученнях, однак найчастіше використовують один або декілька пунктів доступу.

Принцип дії мережі Wi-Fi. Кожен Wi-Fi-адаптер (станція) постійно сканує ефір у пошуку сигналів від пунктів (точок) доступу. Сканування буває пасивним та активним. При пасивному скануванні Wi-Fi-адаптер переглядає окремі канали в пошуку найсильнішого сигналу з пунктів доступу. А кожен пункт доступу періодично передає сигнал (кадр) присутності. В цьому кадрі є ідентифікатор пункту та доступні швидкості передавання. При активному скануванні Wi-Fi-адаптер сам ініціює сканування, передаючи кадр вимоги на передавання, а пункти доступу відповідають кадрами присутності і надалі процес відбувається так, як при пасивному скануванні.

Після сканування відбувається процес автентифікації, який ініціює Wi-Fi-адаптер, що передає запит до пункту доступу. Цей пункт перевіряє запит і підтверджує або відхиляє його. Wi-Fi-адаптер після успішної автентифікації обирає пункт доступу, з яким буде працювати та узгоджує швидкість передавання. Пункт доступу відповідає кадром підтвердження у якому наводиться

додаткову інформацію про себе. Після цього, починається сеанс передавання.

Режими аутентифікації у Wi-Fi-мережах:

1. *Відкрита аутентифікація*. Підключення відбувається без пароля, шифрування не використовується, всі дані передаються у відкритому вигляді тому вони можуть бути перехоплені.

2. *Personal (персональна аутентифікація)*. Використовується єдиний пароль для всіх пристроїв у мережі, при цьому кількість пристроїв обмежена (невелика).

3. *Enterprise (відокремлена аутентифікація)*. Використовуються визначені паролі для різних користувачів з використанням сервера аутентифікації (застосування протоколів Radius, LDAP), захищеність користувачів найвища, але необхідно використовувати спеціальне обладнання.

Для розгортання Wi-Fi-мережі використовують безпроводові точки доступу і безпроводові адаптери. Однак у найпростішому випадку для передачі даних через Wi-Fi-з'єднання навіть не потрібно використання точки доступу. Серед режимів функціонування Wi-Fi-мереж широкого застосування отримали: *Infrastructure* і *Ad Hoc*.

У режимі *Ad Hoc*, що також називають *Independent Basic Service Set (IBSS)*, або *Peer to Peer* (точка-точка), вузли мережі безпосередньо взаємодіють один з одним без участі точки доступу. Цей режим потребує мінімального устаткування: кожен Wi-Fi-клієнт такої мережі повинен бути оснащений тільки Wi-Fi-адаптером. При такій конфігурації не потрібно створення мережної інфраструктури. Основними недоліками режиму Ad Hoc є: обмежений діапазон дії мережі і неможливість підключення до зовнішніх мереж. Якщо обидва Wi-Fi-клієнти перебувають у безпосередній близькості, або в межах прямої видимості, то режим Ad Hoc дозволяє об'єднати їх у одну мережу. Такий режим може бути ефективним при передачі даних з одного пристрою на інший. Але, якщо необхідно об'єднати в таку Wi-Fi-мережу пристрої розташовані на більших відстанях, або в різних приміщеннях, то режим Ad Hoc не ефективний, оскільки потужності передавачів і чутливості приймачів для забезпечення стійкого з'єднання буде недостатньо. У такому випадку для організації ефективної Wi-Fi-мережі потрібно

застосовувати стаціонарну точку доступу. Перевагою цього підходу є те, що це дає змогу розширити зону покриття (радіус дії) Wi-Fi-мережі.

Точка доступу в безпроводовій мережі виконує функцію, яка аналогічна до функції комутатора традиційної кабельної мережі і дозволяє поєднувати всіх клієнтів у єдину мережу. Завдання точки доступу - координувати обмін даними між всіма клієнтами мережі і забезпечити всім клієнтам рівноправний доступ до середовища передачі даних.

Режим функціонування безпроводової мережі на базі точки доступу називається - *Infrastructure Mode*. Розрізняють два різновиди режиму Infrastructure Mode: основний *BSS (Basic Service Set)* і розширений *ESS (Extended Service Set)*. У режимі BSS всі вузли мережі зв'язуються між собою тільки через одну точку доступу, що може виконувати також роль моста до зовнішньої мережі. А у розширеному режимі, тобто ESS, використовується інфраструктура з декількох мереж BSS, причому самі точки доступу взаємодіють одна з одною, що дозволяє передавати дані від однієї BSS до іншої. Між собою точки доступу з'єднуються за допомогою кабельної мережі, або радіомостами.

Стандарти безпроводового зв'язку. Існує кілька основних типів безпроводових стандартів: 802.11a, 802.11b і 802.11g та ін. Відповідно до цих стандартів використовуються різні типи устаткування:

- 802.11a – високошвидкісний стандарт WLAN для частоти 5 ГГц. Підтримує швидкість передачі даних 54 Мбіт/с;
- 802.11b – стандарт WLAN для частоти 2,4 ГГц. Підтримує швидкість передачі даних 11 Мбіт/с;
- 802.11g – встановлює додаткову техніку модуляції для частоти 2,4 ГГц. Призначений для забезпечення швидкостей передачі даних до 54 Мбіт/с.

Устаткування безпроводових мереж включає *точки доступу* (Access Point) і *безпроводові адаптери*. Точки доступу виконують роль концентраторів, що забезпечують зв'язок між абонентами та між собою, а також функцію мостів, що здійснюють зв'язок з кабельною локальною мережею та з Інтернетом. Декілька близько розташованих точок доступу

утворюють зону доступу Wi-Fi, в межах якої всі абоненти, які мають безпроводові адаптерами отримують доступ до мережі. Такі зони доступу (*Hotspot*) створюються в місцях масового скупчення людей: в аеропортах, студентських кампусах, бібліотеках, офісах, бізнес-центрах і т. п.

Метод доступу до мережі – CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) забезпечує можливість кожній точці доступу обслуговувати декількох абонентів, але чим більше абонентів до неї під'єднано, тим менш ефективна швидкість передачі для кожного з них. На українському ринку представлені точки доступу і безпроводові маршрутизатори різних виробників: 3Com, Asus, Asante, D-Link, Gigabyte, MSI, Multico, Trendnet, US Robotics, ZyXEL та ін.

Налаштування точки доступу. Найкращий способом налаштування точки доступу за допомогою комп'ютера (зі встановленим мережним адаптером Ethernet), який підключений до мережевого комутатора. Розглянемо приклад налаштування точки доступу D-Link DWL-G700AP. Для цієї точки доступу за замовчуванням встановлена IP адреса 192.168.0.50 з маскою підмережі 255.255.255.0. Для того, щоб приступити до налаштування точки доступу необхідно призначити комп'ютеру статичну IP адресу з тієї ж підмережі, що і для D-Link DWL-G700AP. Для налаштувань параметрів TCP/IP протоколів потрібно перейти в меню: «*Панель керування → Мережа й Інтернет → Мережеві підключення*» і виконати налаштування в спеціальному мережевому інтерфейсі.

Для налаштування підключення точки доступу до мережі потрібно перейти у браузері за адресою 192.168.0.50 ввівши її в полі адресації. *Зауваження*. IP-адреса повинна бути в межах локальної мережі, тобто 192.168.0.X. У вікні авторизації необхідно ввести логін і пароль. За замовчуванням у багатьох точках доступу використовується логін: *admin* та пароль: *admin*, або порожнє поле. Для точного визначення цих параметрів необхідно звернутися до технічної документації на точку доступу. Після успішної авторизації відкривається сторінка з налаштуваннями точки доступу D-Link DWL-G700AP (рис. 1).

Перелік пунктів головного меню точки доступу D-Link DWL-G700AP: *Home* – домашня початкова сторінка, тут проводяться

всі основні налаштування *Wireless* (безпроводової мережі), *LAN* - підключення до локальної мережі, *DHCP* - протокол динамічної конфігурації та *Wizard* – доступ до помічника налаштувань. Основні налаштування безпроводової мережі за допомогою меню *Wireless* приведено на рис. 2.

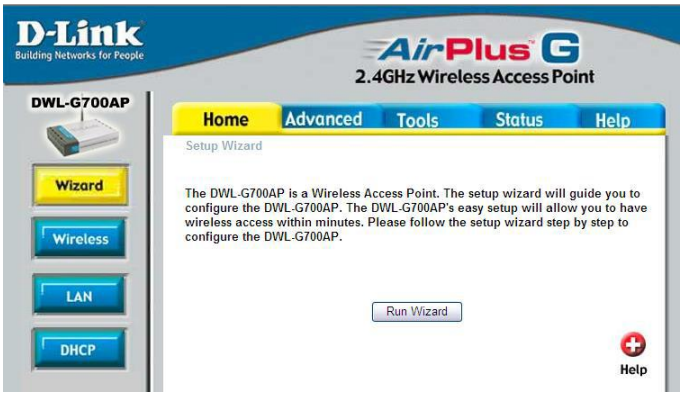


Рис. 1. Меню з налаштуваннями точки доступу D-Link DWL-G700AP

Рис. 2. Основні налаштування безпроводової мережі Wireless

Технології захисту бездротових мереж. *WEP* (Wired Equivalent Privacy) — стандарт захисту, який ґрунтується на методі потокового кодування з використанням алгоритму RC4 (загальний секретний ключ). *WPA* (Wi-Fi Protected Access) - технологія захисту, яка грунтується на протоколі TKIP (Temporal Key Integrity Protocol), що використовує стійкий механізм шифрування. TKIP змінює ключ шифрування для кожного переданого пакета, що ускладнює можливість його підбору. *WPA PSK* (WPA Pre-Shared Key) – це спрощена версія технології WPA, яка може застосовуватися для невеликих безпроводових мереж. У ній, як і в WEP, використовується статичний ключ, але він автоматично змінюється в певних часових інтервалах. *WPA2* є покращеною версією WPA та більш захищеною завдяки заміні TKIP на CCMP (блочне шифрування з кодом автентичності повідомлень). На даний час застосування WPA2 є обов'язковою для всіх сертифікованих Wi-Fi пристроїв. *WPA2 PSK* – це спрощена версія WPA2, яка аналогічна до WPA PSK, але з використанням AES-шифрування.

Налаштування сервісу DHCP. Як було описано вище DHCP – це протокол динамічної конфігурації, який дає змогу отримувати динамічні IP-адреси. На рис. 3 показано варіанти настроювання конфігурації DHCP. В даному випадку ця служба вимкнена і вузли, що будуть підключатися до цієї точки доступу повинні задавати IP адресу у ручному режимі. При ввімкнені DHCP (enabled) пристрої можуть отримувати IP адресу автоматично з вказаного діапазону (Starting IP address – ending IP address). Така IP адреса буде змінюватися через певні проміжки вказаного часу (Lease Time). Більшість сучасних мобільних пристроїв (смартфони, планшети) не мають налаштувань статичної IP-адреси, тому для їх нормального функціонування у цій мережі потрібно ввімкнення служби DHCP.

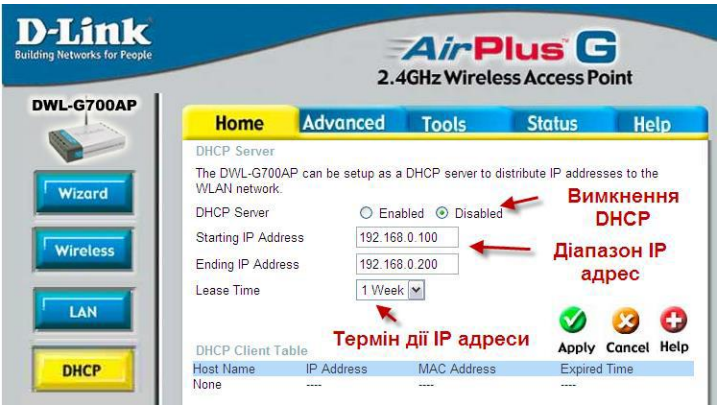


Рис. 3. Настроювання протокола динамічної конфігурації DHCP

Для перегляду різної статистичної інформації: списку підключених вузлів, помилок та інших параметрів використовується меню Status (рис. 4).

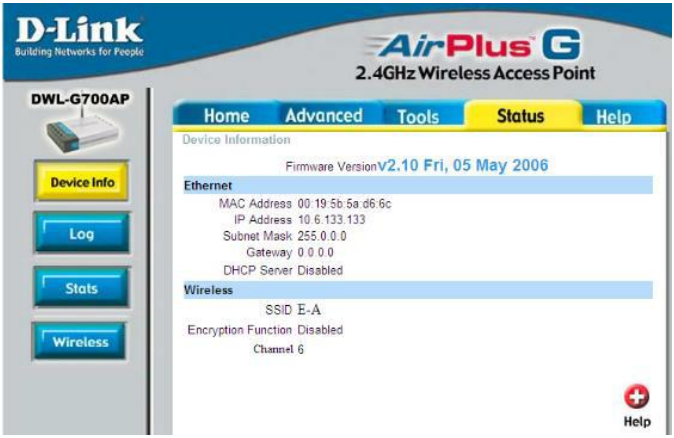


Рис. 4. Вигляд меню Status

Налаштування списку дозволених комп’ютерів за MAC-адресами можна виконати у додаткових налаштуваннях *Advanced* (рис. 5).

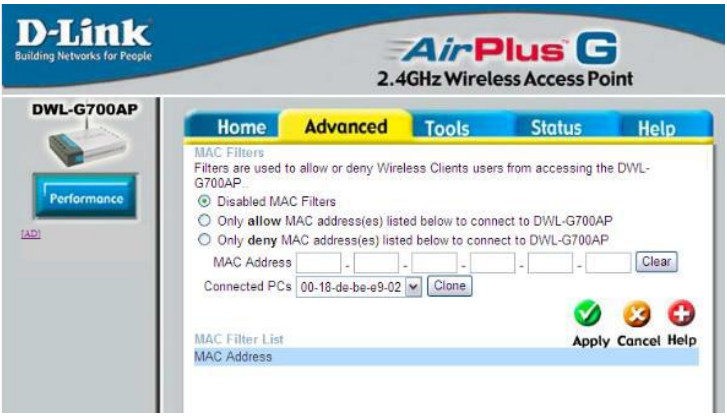


Рис. 5. Меню додаткових налаштуваннях Advanced

Використовуючи *Tools*–меню для налаштувань прав доступу (рис. 6) можна обмежити доступу приховавши назву точки доступу. Це дасть змогу, приховати її видимість для інших, а при необхідності підключення до неї потрібно ввести її ім'я.

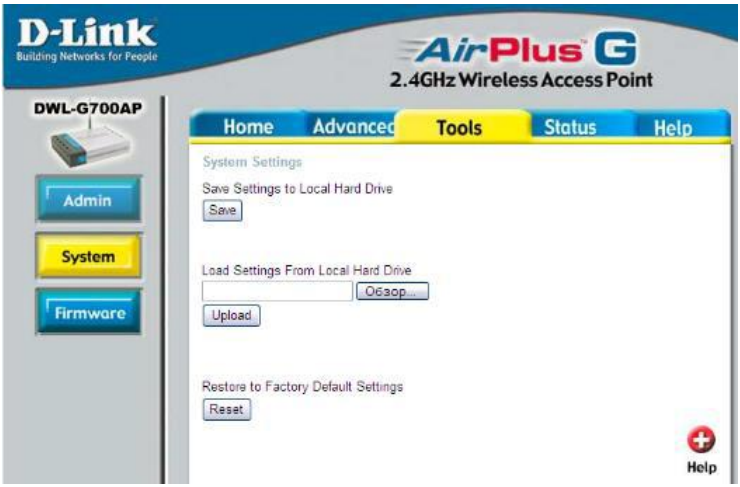


Рис. 6. Меню для налаштувань прав доступу Tools

Налаштування безпроводового Wi-Fi з'єднання на комп'ютері з ОС Windows. Для того щоб переглянути доступні

безпроводові мережі і під'єднатися до Wi-Fi-мережі на локальному комп'ютері з операційною системою Windows 10 необхідно натиснути на відповідне зображення мережевого підключення у правому нижньому куті рядку стану. Після цього з'явиться перелік доступних безпроводових мереж. Для того щоб переглянути властивості безпроводової мережі можна у цьому ж вікні поряд з кнопкою доступних мереж натиснути на вкладку «*Налаштування мережі та інтернету*», або перейти в меню: «*Панель керування → Мережа й Інтернет → Мережеві підключення → Стан Wi-Fi → Властивості безпроводової мережі*» (рис. 7).

Підключення до невидимої мережі. Щоб налаштувати підключення до невидимої мережі (тієї яка не надсилає повідомлення з SSID ідентифікатором) потрібно перейти в пункт «*Стан Wi-Fi → Властивості безпроводової мережі*» та на відповідній вкладці вибрати необхідну мережу і зазначити галочку «*Підключатися,навіть якщо мережа не передає своє ім'я (SSID)*» (рис. 7).

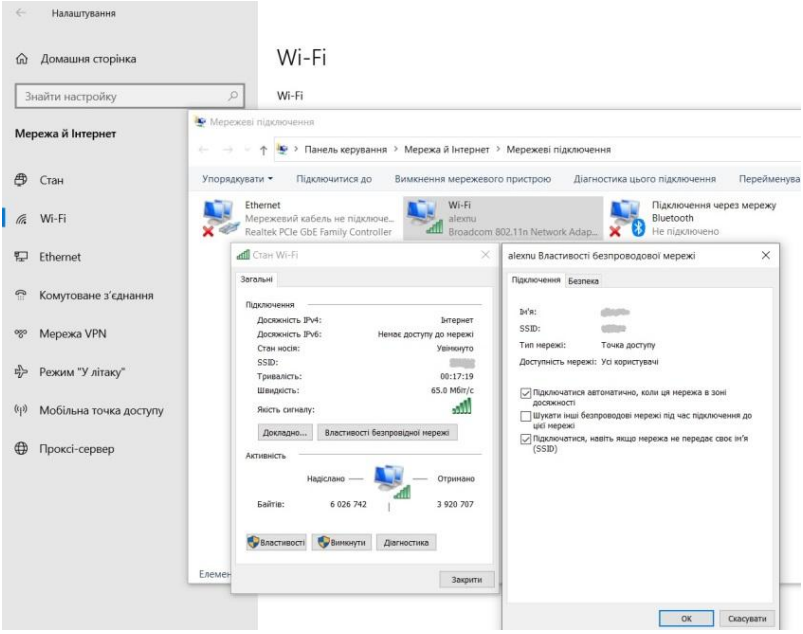


Рис. 7. Настроювання безпроводової мережі

Також можемо здійснити інші налаштування мережі та методи її аутентифікації.

Програма роботи

1. Ознайомитися з будовою і призначенням основних складових безпроводової мережі Wi-Fi.
2. Навчитися використовувати технології захисту безпроводовго Wi-Fi-з'єднання.

Порядок виконання роботи

1. Під'єднати точку доступу (Wi-Fi маршрутизатор) до локального комп'ютера за дпомогою патч-корда RJ45. *Зауваження*. У якості прикладу використано Wi-Fi маршрутизатор TP-Link TL-WR8411N. У випадку використання іншого маршрутизатора опис меню з налаштуваннями можуть відрізнятися.
2. Налаштувати TCP/IP властивості для під'єднання до точки доступу та записати попередні налаштування. Для цього необхідно перейти: *Панель керування* > *Мережа й Інтернет* > *Мережеві підключення* > *Ethernet* > *Властивості Протокол інтернету версії 4 (TCP/IPv4) - властивості.*

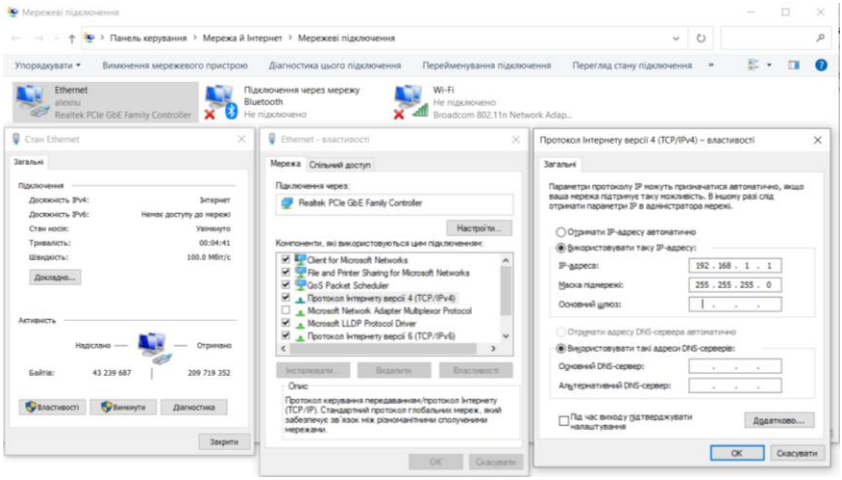


Рис. 8. Налаштування TCP/IP властивості для під'єднання до точки доступу

3. Перевірити зв’язок з точкою доступу за допомогою утиліти *ping* ввівши вказану IP-адресу (рис. 9).

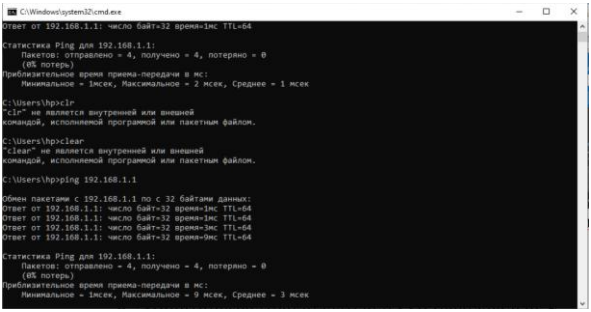


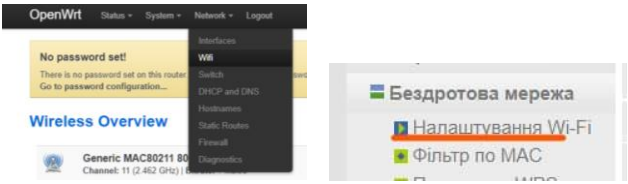
Рис. 9. Перевірити зв’язок з точкою доступу

4. Ознайомитись з наявними налаштуваннями маршрутизатора (меню *Status*):

- DHCP-сервера;
- MAC-адреси підключених пристроїв;
- алгоритм шифрування даних у безпровідній мережі;
- швидкість з’єднання;
- потужність сигналу та ін.

Щоб зайти в налаштування маршрутизатора необхідно в браузері відкрити сторінку за адресом основного шлюзу (див. п.2). У стандартних налаштуваннях часто використовують IPv4 (192.168.0.1), який береться з налаштувань TCP/IP протоколу. В різних маршрутизаторах він може відрізнятися, зазвичай він вказаний в документації до нього, або на самій точці доступу. Також для входу на цю сторінку потрібно мати логін і пароль адміністратора точки доступу, він також вказаний в документації, або на зворотній стороні маршрутизатора.

5. Використовуючи меню *Network* > *Wifi* (рис. 10, а), якщо меню маршрутизатора україномовне, то його вигляд буде, як на рис. 10, б. Після цього змінюємо назву точки доступу (SSID) на довільну.



а) б)

Рис. 10. Зовнішній вигляд меню маршрутизатора з вибором налаштування мережі

6. Змінюємо параметри налаштування мережі натиснувши *edit* (*редаг.*) ввівши назву мережі, у нашому прикладі це - xxx та режими роботи маршрутизатора.

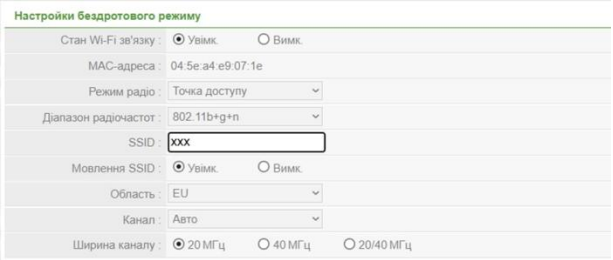


Рис. 11. Зміна назви точки доступу (SSID)

7. Вибираємо тип шифрування даних та задаємо пароль доступу (рис. 12).

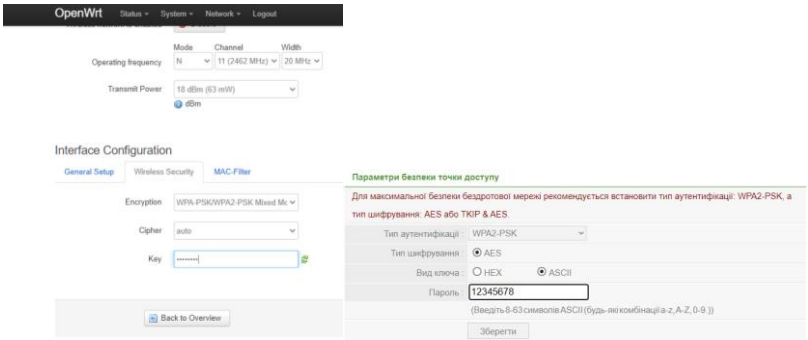


Рис. 12. Приклад зміни типу шифрування та пароля доступу для різних типів маршрутизаторів

8. Виконуємо підключення до налаштованої точки доступу та перевіряємо наявність з'єднання (рис. 13).

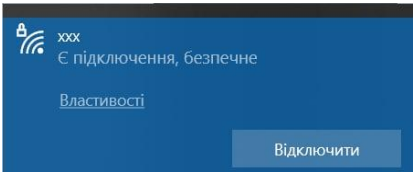


Рис. 13. Перевірка можливості з'єднання з маршрутизатором

9. Під'єднатися до точки доступу з довільного пристрою та провести тестування з'єднання за допомогою утиліт *ping*, *netstat* та ін. (рис. 14).

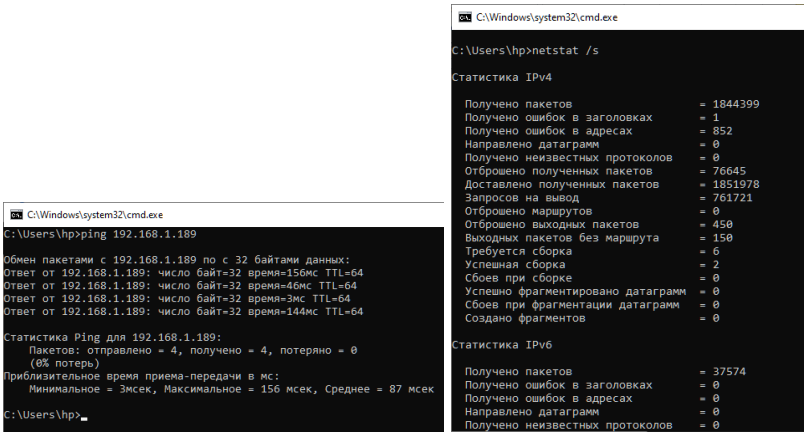


Рис. 14. Перевірка з'єднання з маршрутизатора за допомогою утиліт *ping*, *netstat*

10.Змінити параметри аутентифікації (WPA-PSK, WPA2-PSK та ін.) почергово перевіряючи та доналаштовуючи з'єднання на маршрутизаторі.

11.Знайти під'єднаний/ні пристрій/ої в меню статистики на точці доступу та зафіксувати їх MAC-адреси (у нижчеподаному прикладі доступні тільки 3-и пристрої) (рис. 15).

DHCP Leases

Hostname	IPv4-Address	MAC-Address	Leasetime remaining
Redmi7A-Redmi	192.168.1.189	c8:3d:dc:14:33:1d	11h 56m 13s
?	192.168.1.227	40:2c:f4:02:5d:07	11h 33m 1s
Aleks	192.168.1.221	64:31:50:09:73:83	11h 46m 21s

Рис. 15. MAC-адреси під'єднаних пристроїв

12. Виконати тестування роботи фільтру MAC-адрес. Для цього спочатку заблокуйте доступ до тестованого пристрою вказавши його MAC-адресу, а потім дозвольте підключення тільки цьому пристрою (рис. 16).

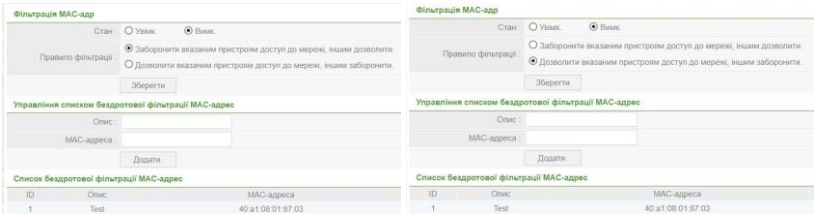


Рис. 16. Налаштування фільтрації за MAC-адресами

13.Виконати приховування точки доступу (SSID) за допомогою встановлення відповідної відмітки (рис. 17) та перевірити результат за допомогою стороннього пристрою.

Рис. 17. Виконання приховування маршрутизатора

14.За допомогою послідовності, яка описана в теоретичних відомостях та показана на рис. 7 виконати під'єднання до прихованої (невидимої) мережі та протестувати це з'єднання.

15.Записати параметри налаштувань точки доступу зробивши резервну копію (*load settings from local hard drive*).

16.Від'єднати точку доступу від локального комп'ютера та повернути налаштування TCP/IP до початкових значень.

17.Результати виконання всіх пунктів оформити у вигляді звіту на стандартних аркушах формату А4.

Вимоги до оформлення звіту:

Звіт повинен містити:

- титульну сторінку;
- мету роботи;
- програму роботи;
- скріншоти виконання всіх пунктів, які описані у порядку виконання роботи;
- описи процесу виконання з поясненнями виконаних дій;
- висновок.

Контрольні запитання

1. Що таке Wi-Fi?
2. Які стандарти Wi-Fi ви знаєте?
3. Яка максимальна швидкість Wi-Fi?
4. Яке обладнання використовується для безпроводового доступу?
5. Де і як здійснюється налаштування безпроводової точки доступу?
6. Які налаштування потрібно задати для підключення на локальному комп'ютері?
7. Що таке SSID?
8. Які стандарти шифрування ви знаєте?
9. Як приховати точку доступу?
10. Що таке DHCP, для чого він використовується?
11. Як дозволити підключення до точки доступу лише певним пристроям?
12. Як повернути налаштування точки доступу в початковий стан?

Лабораторна робота №5
Використання технології віртуалізації, DHCP та DNS-серверів

Мета роботи

Ознайомитись з можливостями віртуалізації VirtualBox. Вивчити способи отримання IP-адреси вузлів та навчитись використовувати DHCP-клієнт та налаштовувати DHCP та DNS-серевер.

Теоретичні відомості

Віртуалізація – надання абстрагованих від апаратної реалізації обчислювальних ресурсів, що забезпечує логічну ізоляцію процесів, виконуваних на одному фізичному ресурсі. Віртуалізація дозволяє запустити декілька операційних систем (ОС) на одному комп'ютері. За допомогою цієї властивості можна убезпечити операційну системи та ПК від помилок та пошкоджень на етапі розробки і впровадження нових операційних систем, програм та функцій, а також для безпечного використання мережевих функцій та налаштувань.

Поширеним способом віртуалізації є бінарна трансляція, що полягає у перехопленні гіпервізором інструкцій віртуалізованої системи та їх заміні на "безпечні" інструкції, що далі виконуються процесором (напр., VMWare Workstation, VirtualBox, QEMU). Гіпервізор – це програма, яка керує фізичними ресурсами обчислювальної машини та розподіляє ці ресурси між декількома різними операційними системами, дозволяючи запускати їх одночасно.

Більшу продуктивність віртуалізовних систем забезпечує паравіртуалізація («спосіб свідомого співробітництва»), гостьові операційні системи підготовлюються для виконання в віртуалізованому середовищі, для чого програмне ядро цих операційних систем дещо модифікується. Операційна система взаємодіє із програмою гіпервізора, який надає їй гостьовий API, замість використання безпосередньо таких ресурсів, як таблиця сторінок пам'яті.

Апаратна віртуалізація – віртуалізація за допомогою спеціальної процесорної архітектури. На відміну від програмної

віртуалізації, можливе використання ізольованих гостьових систем, керованих гіпервізором безпосередньо. Апаратна віртуалізація забезпечує більшу продуктивність у порівнянні з продуктивністю невіртуалізованної машини, що дає віртуалізації можливість практичного використання і широкого застосування. Найбільш поширені технології віртуалізації Intel-VT і AMD-V (Xen, VMWare Workstation, VirtualBox).

DHCP (Dynamic Host Configuration Protocol – протокол динамічної конфігурації вузла) — це протокол прикладного рівня OSI, що дозволяє комп'ютерам автоматично одержувати IP-адресу та інші параметри, необхідні для роботи в мережі. Для цього комп'ютер звертається до спеціального серверу DHCP. Під час запуску системи комп'ютер, який є DHCP-клієнтом, відправляє в мережу запит на отримання IP-адреси. DHCP-сервер відповідає і відправляє повідомлення-відповідь, яка містить IP-адресу і деякі інші конфігураційні параметри. При цьому сервер DHCP може працювати в різних режимах:

1. Динамічний розподіл. Адміністратор присвоює IP-діапазон адрес на сервері DHCP. Кожен клієнтський комп'ютер в мережі повинен запитати IP-адресу від DHCP-сервера, коли мережа ініціалізується за концепцією "оренди". Коли закінчується термін оренди, якщо вона не буде продовжена, DHCP-сервер має право повернути адресу і призначити її на інші комп'ютери.

2. Автоматичне виділення. Сервер DHCP буде постійно призначати вільну IP-адресу з діапазону, встановленого адміністратором, запитуючому комп'ютеру. Основна відмінність з динамічним розподілом в тому, що сервер зберігає записи минулих оренд і намагається привласнити ту ж адресу тому ж комп'ютеру для майбутніх мережних підключень.

3. Статичний розподіл. Сервер DHCP здійснює призначення IP-адрес виключно на основі таблиці MAC-адрес, які зазвичай заповнені вручну адміністратором мережі. Якщо MAC-адреса комп'ютера не зазначена в таблиці, йому не буде призначена мережева адреса.

NAT (від англ. Network Address Translation — «перетворення мережевих адрес») — це механізм у мережах TCP/IP, який дозволяє змінювати IP-адресу у заголовку пакету, що проходить

через пристрій маршрутизації трафіку.

DNS (Domain Name System) — ієрархічна розподілена система перетворення імені хоста (комп'ютера або іншого мережевого пристрою) в IP-адресу.

Нерекурсивна процедура запиту DNS імен:

– DNS-клієнт звертається до кореневого DNS-сервера з вказівкою повного доменного імені;

– DNS-сервер відповідає клієнту, вказуючи адресу наступного DNS-сервера, який виконує обслуговування домену верхнього рівня, заданого в наступній старшій частині імені;

– DNS-клієнт виконує запит наступного DNS-сервера, який його надсилає до DNS-сервера потрібного піддомена і т.д., доти, доки не буде знайдено DNS-сервер, який повністю відповідає запитуваному імені IP-адреси. Сервер дає кінцеву відповідь клієнту.

Рекурсивна процедура:

– DNS-клієнт запитує локальний DNS-сервер, який обслуговує піддомен, якому належить клієнт;

– якщо локальний DNS-сервер відповідь знає, то повертає її клієнту, в протилежному випадку виконує ітеративні запити до кореневого сервера до тих пір, поки не отримає відповідь;

– після отримання відповіді сервер передає її клієнту.

Таким чином, при рекурсивній процедурі клієнт фактично передоручає роботу власному серверу. Для прискорення пошуку IP-адрес DNS-сервери часто застосовують кешування відповідей, які проходять через них.

Програма роботи

1. Створити віртуальну машину та ознайомитись з інтерфейсом VirtualBox.
2. Запустити віртуальну машину та встановити операційну систему Debian.
3. Навчитися використовувати DHCP та DNS-протоколи для захисту доступу до мережі.
4. Налаштувати та запустити DNS та DHCP-сервер у віртуальній машині.

Порядок виконання роботи

1. Запустити віртуальну машину Oracle VM VirtualBox.

Зауваження. Для роботи Oracle VM VirtualBox необхідно, щоб в налаштуванні BIOS було увімкнено апартну віртуалізацію на ПК. Якщо процесор ПК не підтримує апаратну віртуалізацію AMD-V або Intel VT-x необхідно в налаштуваннях віртуальної машини зняти відмітку «*Увімкнути VT-x/AMD-V*» на вкладці *Система > Прискорення*.

2. Скопіювати файл віртуального диска на локальний комп'ютер.

3. Створити нову віртуальну машину (кнопка *Create/Створити* на панелі інструментів). Вибір операційної системи у вікні створення впливає лише на відображувану піктограму створеної віртуальної машини та не обмежує використання у віртуальній машині інших операційних систем, наприклад, Windows (у нашому випадку вказуємо тип Debian).

4. Вказати розмір RAM не менше ніж 512 Мб.

5. Створити новий віртуальний диск, прийнявши параметри за замовчуванням. Підключаємо ISO-образ інсталяційного диску з якого буде встановлюватися операційна система та вказуємо шлях до папки з попередньо завантаженою інсталяцією операційної системи. У нашому випадку це ОС Debian з ядром Linux. Установочний ISO-образ завантажуємо за посиланням https://cdimage.debian.org/debian-cd/current/i386/iso-cd/debian-12.0.0-i386-netinst.iso. У процесі встановлення вказуємо логін і пароль для звичайного користувача та користувача з правами адміністратора, тобто *root*, а також інші параметри операційної системи. Після встановлення ОС Debian вимикаємо віртуальну машину.

6. Для повторного запуску віртуальної машини в меню *Пам'ять* вибираємо встановлену операційну систему та за допомогою опції налаштування (кнопка *Налаштувати*) задаємо тип підключення до мережі — *проміжний адаптер/мережевий міст*.

7. Знову запускаємо віртуальну машину. Входимо в систему з логіном та паролем вказаними при встановленні операційної системи.

8. Виконати команду *sudo ip a* (*sudo* – виконати від імені

іншого користувача, за замовчуванням – суперкористувач *root*, *ip* – налаштування інтерфейсу, ключ *a* – показати всі) та скопіювати результат у звіт. Вимкнути машину.

Зауваження. Якщо вхід в операційну систему здійснено під користувачем *root*, то команду *sudo* можна не застосвувати, а виконати команду *ip a*.

9. Змінити *налаштування мережі* у Oracle VM VirtualBox гостьової системи на NAT. Для цього необхідно пройти за вказаним шляхом (*Машина* > *Налаштувати* > *Мережа* > *Тип підключения* > *NAT*) та виконати зміни.

10. Порівняти отримані IP-адреси операційної системи Debian, яка запущена з віртуальної машини з адресою хост-системи, тобто з ОС Windows за допомогою команди *ipconfig -a*. Ці адреси мають належати до однієї підмережі.

11. Перевірити час передачі пакетів до вузла en.wikipedia.org програмою ping: *ping en.wikipedia.org*.

12. Перевірити час передачі пакетів до будь якого вузла вказавши його IP-адресу. Виконати команду *ping* в ОС Windows та порівняти дані з її виконанням в ОС Debian. Для зупинки виконання команди *ping* використати комбінацію CTRL+C.

13. Встановити *dnsmasq* та файловий менеджер Midnight Commander (MC) за допомогою команди *apt install dnsmasq mc*. Відкрити файл налаштування DHCP та DNS-сервера *dnsmasq* (файл знаходиться в каталозі */etc*, назва файлу: *dnsmasq.conf*). Змінити діапазон видаваних IP-адрес. Прив’язати MAC-адресу одного з хостів до IP-адреси. Для цього скористайтесь файловим менеджером MC та редактором *Nano*. Оскільки редагування файлів налаштувань дозволено лише користувачу *root* (суперкористувачу), запуск файлового менеджера здійснюється командою *sudo mc* (відповідно без *sudo*, якщо ви працюєте під користувачем *root*). Навігація по об’єктам у каталозі здійснюється клавішами керування курсором, перехід у батьківський каталог (на рівень вище) – вибором *.. (дві крапки)*, перегляд текстового файлу – натисканням F3, редагування – F4. При першій спробі редагування з *mc* буде запропоновано обрати текстовий редактор за замовчуванням. Рекомендується обрати *Nano* вибором відповідної цифри та натисканням Enter. Діапазон IP-адрес до видачі задається рядком *dhcp-range=*. За

замовчуванням ці рядки закоментовані (починаються з ‘#’ - символу коментаря та ніяк не інтерпретуються при зчитуванні налаштувань). Знайдіть і розкоментуйте рядок (видаливши символ #), задайте діапазон адрес від 192.168.45.12 до 192.168.45.67 та час оренди 72 години. Аналогічно виконайте налаштування прив’язки IP-адреси до MAC-адреси, яке виконується подібним чином, знайшовши рядок виду: *#dhcp-host=MAC-адреса,IP-адреса*. Щоб зберегти внесені зміни, натисніть *Ctrl+O* та підтвердіть ім’я файлу для збереження натисканням Enter.

14. Вийдіть з текстового редактора натисканням *Ctrl+X* та з файлового менеджера натисканням *F10*. Для застосування нових налаштувань перезапустіть *Dnsmasq* командою *sudo service dnsmasq restart* (відповідно без sudo, якщо ви працюєте під користувачем *root*).

Зауваження. Виконавши пп. 13 і 14 ми налаштували можливість отримання іншими вузлами мережі IP-адрес, зокрема з прив’язкою MAC-адрес, що обмежує доступ інших комп’ютерів з невказаними нами MAC-адресами.

15. Результати виконання оформити у вигляді звіту на стандартних аркушах формату А4.

Вимоги до оформлення звіту:

Звіт повинен містити:

- титульну сторінку;
- мету роботи;
- програму та порядок виконання роботи;
- скріншоти з налаштуваннями створеної віртуальної машини;
- скріншоти з отриманими IP-адресами при використанні: *проміжний адаптер/мережевий міст* та *NAT*;
- скріншот файлу налаштувань *dnsmasq*;
- висновок, що пояснює різницю між отриманими адресами з та без використання NAT.

Контрольні запитання

1. Назвіть поширені технології апаратної віртуалізації.
2. Яка відмінність паравіртуалізації від повної віртуалізації?

3. Яке призначення протоколу DHCP?

4. В чому різниця в роботі DHCP-серверів, налаштованих на статичний та динамічний розподіл адрес?

5. Яке призначення DNS?

6. Для чого використовують DNS-кешування?

Лабораторна робота №6
Резервне копіювання та відновлення даних

Мета роботи

Навчитись використовувати засоби резервного копіювання жорсткого диска та відновлювати дані з резервної копії.

Теоретичні відомості

Сучасні системи резервного копіювання інформації передбачають ефективну стратегію, організаційні рішення і політику збереження даних. Існують різні технології резервного копіювання, які відрізняються витратами коштів і часу:

- *повне резервне копіювання* — вибрані дані будуть скопійовані повністю. Найнадійніший спосіб, але потребує найбільшої кількості ресурсів, місця для зберігання даних і часу копіювання, тому в такому вигляді застосовується рідко, зазвичай комбінується з іншими видами. Дозволяє відновити втрачені дані з нуля швидше за всі інші види копіювання;
- інкрементне копіювання — записуються тільки ті дані, які були змінені з часу минулого копіювання. Для таких копій потрібно значно менше пам'яті, ніж при повному копіюванні і записуються вони значно швидше. При такому підході також необхідно періодично робити повну резервну копію, при будь-якій аварії систему відновлюють з такої копії, а потім накочуються на неї всі наступні інкрементні копії в хронологічному порядку. Важливим елементом інкрементного копіювання є відновлення видалених файлів і проміжних версій, які змінювалися;
- диференціальне резервне копіювання — схоже на інкрементне, тобто копіюються тільки зміни, зроблені з моменту останнього повного копіювання. Його відмінність полягає в тому, що в кожну наступну копію зберігаються зміни з попередньої і додаються нові. Для відновлення після аварії знадобиться тільки повна копія і остання з диференціальних, що значно скорочує час відновлення.

Створення резервної копії (*backup*) даних надає можливість виконати відновлення інформації при втраті оригіналу, з якого було створено резервну копію. При цьому під втратою треба

розуміти настання події, що призвела до зміни даних, після чого вони втратили цінність або були видалені з носія. Приклад: умисне завдання шкоди через видалення важливої для підприємства інформації.

Об'єкти резервного копіювання — це дані або сукупність даних, з яких можна створити резервну копію. Приклади об'єктів: файли або теки, дані прикладних програм, дані операційної системи чи сама ОС, образи віртуальних машин та дисків віртуальних машин, файлові системи тощо.

Рівні резервного копіювання. *Повне резервне копіювання* (Full Backup або L0) — повна копія даних. Рівень, який забезпечує створення повної копії об'єкту резервного копіювання. Цей рівень дозволяє забезпечити максимальну відповідність оригіналу даних його копії.

Диференційне резервне копіювання (Differential Backup або L1) — копіювання змін, що були зроблені після створення останньої повної копії. Створення такої копії потребує більше часу та займає більший об'єм, ніж додаткове копіювання, але дозволяє пришвидшити процес відновлення. Загалом є альтернативою між створенням повної або додаткової копії.

Додаткове резервне копіювання (Incremental Backup або L2) — копіювання змін, що відбулись із часу повного, диференційного або додаткового копіювання. Загалом на додаткове копіювання затрачається менше часу, бо копіюється менше файлів. Однак процес відновлення даних займає більше часу, оскільки повинні спочатку відновлюватися дані останньої повної копії і після цього - всі резервні копії, від яких залежить додаткова копія.

Під час роботи операційної системи деякі файли можуть використовуватись нею та бути недоступними для читання користувачем, тому резервна копія системного розділу ОС повинна виконуватись при неактивній ОС або використовувати технології ОС, що забезпечують актуальність вмісту файлів (Shadow Copy, Snapshot).

Щоб забезпечити можливість відновлення системного розділу у випадку, коли операційна система не може завантажитись з нього, резервну копію створюють за допомогою технологій LiveCD/LiveUSB/PXE. Це дозволяє за критичної

помилки завантаження операційної системи завантажити «рятувальний» LiveCD/LiveUSB/PXE та відновити вміст системного розділу. Лазерний диск, що дозволяє завантажити операційну систему з нього без необхідності встановлення, називають LiveCD. Якщо аналогічну функцію має USB-накопичувач, його називають LiveUSB. PXE (англ. Preboot Execution Environment) — середовище для завантаження комп'ютерів за допомогою мережевої карти без використання жорстких дисків, компакт-дисків чи інших пристроїв, що застосовуються при локальному завантаженні операційної системи. Для організації завантаження системи в PXE використовуються протоколи IP, UDP, DHCP та TFTP. PXE-код, який зазвичай знаходиться в ПЗП мережевої карти, отримує з мережі по протоколу TFTP (отримаючи для цього адресу TFTP-сервера за допомогою DHCP) виконуваний файл, після чого передає йому управління.

Як правило, резервну копію зберігають в файлі-образі диска. Образ диска — файл, що містить у собі повну копію вмісту та структури файлової системи та даних, що містяться на диску. Образ розділу диску бажано розміщувати на іншому фізичному носії, щоб у випадку виходу з ладу диску не був втрачений і образ диску.

Програма роботи

1. Ознайомитися з технологіями резервного копіювання даних з теоретичних відомостей.

2. Завантажити Live-CD з засобом резервного копіювання HDD та виконати резервне копіювання диска.

3. Відформатувати віртуальний диск та відновити дані з резервної копії.

Порядок виконання роботи

1. Скопіювати програми Clonezilla Live та GParted на локальний ПК відповідно з *https://clonezilla.org/downloads.php* та *https://gparted.org/download.php,* при цьому потрібно скопіювати образ у форматі *iso* з необхідними параметрами системи.

2. У віртуальній машині OracleVM VirtualBox створити новий жорсткий диск. Для цього у OracleVM VirtualBox Менеджер

необхідно вибрати пункт: *Налаштування > Пам'ять > Контролер:SATA > Жорсткий диск* натиснувши відповідну кнопку, яка *Додає жорсткий диск* (праворуч від меню *«Контролер: SATA»)*. Після цього під'єднуємо до віртуальної машини ISO-образ GParted та завантажуємося з нього натиснувши відповідну кнопку, яка *Додає привод оптичного диску* (праворуч від меню *«Контролер: IDE»)*. Після завантаження GParted у списку доступних жорстких дисків (у правому верхньому куті менеджера GParted натискаємо кнопку) вибираємо порожній жорсткий диск (як правило цей диск буде відображатися, як «не розподілено»). Виконуємо форматування цього нового жорсткого диску у файловій системі *ext4*. Для цього у меню *Пристрій* вибираємо *Створити таблицю розділів* тип таблиці *msdos*. Під час створення розділу (резервний диск) необхідно задати йому *мітку розділу*, у відповідному вікні, за допомогою якої його можна буде відрізнити при створенні резервної копії (якщо цю опцію не виконати, то відрізнити новий диск від існуючого буде складно). Після цього створюємо новий розділ (клацнувши по диску правою клавішею) і застосовуємо зміни.

3. Перезавантажити віртуальну машину та завантажитись з *Live CD Clonezilla*. Для цього у меню віртуальної машини у пункті: *Налаштування > Пам'ять > Контролер:IDE > Оптичний привід* вказати шлях до завантаженого *iso*-образу *Clonezilla* (див. п. 1). Запускаємо віртуальну машину і обираємо подальші налаштування *Live CD Clonezilla* за замовчуванням. Після виконання аналізу дискового простору ми отримаємо два диска, один з яких має встановлену операційну систему, а інший резервний. Обираємо другий диск на який буде виконуватися резервне копіювання.

Виконуємо резервне копіювання одного жорсткого диска в образ диска на іншому, використовуючи вказівки Clonezilla. Для кращого розуміння інтерфейсу при встановленні вибираємо українську або російську мову. Після запуску вибираємо роботу з дисками або розділами використовуючи образи, так як показано на рис. 1.

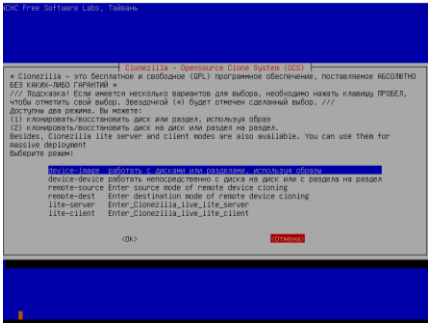


Рис. 1. Вибір у програмі *Clonezilla* режим роботи з дисками або розділами використовуючи образи

Після цього вибираємо команду rerun1 (рис. 2).

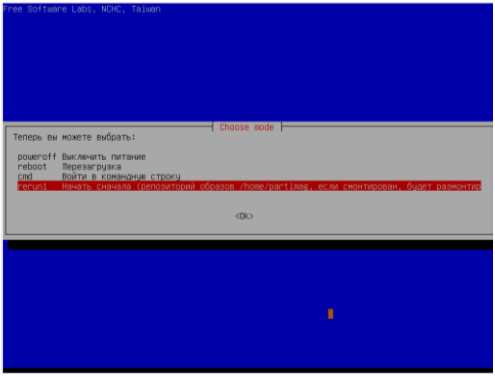


Рис. 2. Вибір команди команду rerun1

Для вибору необхідного диску на який буде встановлена резервна копія необхідно зі запропонованого списку вибрати створений попередньо резервний диск.

Зауваження. Щоб розрізняти новостворений розділ (резервний диск) необхідно задати йому мітку розділу, якщо цього не було зроблено раніше за допомогою програми GParted (п. 2). При цьому, перезапускаємо віртуальну машину і використовуючи програму GParted додаємо мітку розділу у новоствореному диску. Також розрізнити ці диски (зі встановленою ОС Debian та новостворений резервний диск)

можна порівнявши їх розміри, попередньо визначивши розмір диску зі встановленою ОС.

Після цього продовжуємо процес у Clonezilla з попереднього пункту. По завершенні створення резервної копії програма Clonezilla запропонує перевантажитися.

4. Перезавантажити віртуальну машину з ОС Debain. Після цього, затерти нулями завантажувальну область диска командою *sudo dd if=/dev/zero of=/dev/sda bs=1k count=1k* (якщо виконано вхід під користувачем *root*, то виконуємо команду без *sudo*).

Зауваження. Виконуючи попередню дію ми імітуємо пошкодження диску, або операційної системи.

5. Переконатись в неможливості завантаження ОС Debian з даного жорсткого диска, перезапустивши ОС. Завантаження не повинно відбуватися у зв'язку з тим, що ми пошкодили завантажувальну область (повинно з'явитися повідомлення про фатальну помилку).

6. Відновити встановлену ОС Debian з раніше створеної резервної копії за допомогою програми *Clonezilla*, повторно заватаживши з *Live CD Clonezilla* та обравши при розгортанні образу диска в якості цільового диска той, на якому був знищений завантажувальний запис. Переконатись в нормальному завантаженні відновленої ОС.

Для виконання цього процесу запускаємо знову Clonezilla так як в п. 3, проводимо ті ж самі налаштування, проте в кінці обираємо опцію *restoredisk*. Потрібно вказати *відновити образ з диску* вказавши шлях до папки в якій зберігається резервна копія та вибрати режим відновлення, як показано на рис. 3, при цьому має запуститися процес відновлення.

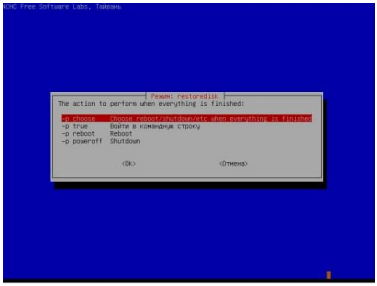


Рис. 3. Вибір режиму відновлення

7. Очікуємо результат копіювання резервної копії Після відновлення перезапускаємо віртуальну машину і переконуємося, що ОС Debian знову запускається у нормальному режимі.

8. Результати виконання оформити у вигляді звіту на стандартних аркушах формату А4.

Вимоги до оформлення звіту:

Звіт повинен містити:

- титульну сторінку;
- мету роботи;
- програму та порядок виконання роботи;
- скріншоти з описами виконання кожного пункту *Порядку виконання роботи.*
- скріншоти та описи: форматування диска в *ext4*, виконання команди *dd* та результату розгортання образу на диск;
- висновок.

Контрольні запитання

1. Назвіть приклади подій, після яких необхідно відновлення даних або всієї системи з резервної копії.
2. Які розрізняють рівні резервного копіювання?
3. Чи завжди можливо виконати повне резервне копіювання системи?
4. В чому різниця між LiveCD та LiveUSB?
5. Яким чином можна завантажити ОС на ПК, який не має можливості підключення зовнішніх носіїв даних?
6. Що таке образ диска та які вимоги ставляться до його зберігання?

Лабораторна робота №7
Особливості використанні протоколів TFTP, FTP, Telnet, SSH

Мета роботи

Ознайомитись з можливостями протоколів FTP, TFTP, Telnet та SSH та навчитись встановлювати та налаштовувати TFTP, FTP, Telnet, SSH-сервери та клієнти, а також розглянути можливості передачі даних з їх використанням.

Теоретичні відомості

Протокол передавання файлів *FTP* (File Transfer Protocol) — це протокол, який використовується для передавання файлів через Інтернет. Протокол FTP зазвичай застосовується для того, щоб зробити файли доступними для завантаження. З його допомогою можна завантажувати web-сторінки в процесі створення чи модернізації web-сайту, виконувати обмін зображеннями та ін.

Протокол передачі файлів *TFTP* (Trivial File Transfer Protocol) в основному використовується для первинного завантаження бездискових робочих станцій. На відміну від FTP протокол TFTP не містить можливостей аутентифікації (хоча можлива фільтрація за IP-адресою). Цей протокол заснований на транспортному протоколі UDP.

Telnet (англ. TErminaL NETwork) — мережевий протокол для реалізації текстового інтерфейсу через мережу. Часто вживається у вигляді програми-клієнта, тому Telnet — це програма з текстовим інтерфейсом, яка дає змогу підключитись до іншого комп'ютера через Інтернет за цим протоколом. Якщо власник або адміністратор надає право підключитися до ПК, то програма *Telnet* дає змогу вводити команди для доступу до програм і служб на віддаленому комп'ютері, ніби так, як би ви працювали безпосередньо за ним. Програму Telnet можна використовувати для доступу до електронної пошти, баз даних і файлів. Як правило, Завдяки простоті програмної реалізації Telnet часто використовується для доступу до вбудовуваних систем, або мережевого обладнання.

Оскільки у протоколі *Telnet* не передбачено використання ні

шифрування, ні перевірки достовірності даних, то він вразливий для будь якого виду атак, до яких вразливий протокол TCP. Тому для доступу до UNIX-подібних операційних систем використовується інший протокол - SSH.

SSH (англ. Secure SHell — «безпечна оболонка») — мережевий протокол рівня застосунків, що дозволяє проводити віддалене управління комп'ютером і здійснювати тунелювання TCP-з'єднань (наприклад, для передачі файлів). SSH схожий за функціональністю з протоколом *Telnet* і *rlogin*, проте шифрує весь потік даних, в тому числі, передані паролі.

Криптографічний захист протоколу SSH не фіксований, він надає вибір різних алгоритмів шифрування. Крім того, цей протокол дозволяє не тільки використовувати безпечний віддалений *shell* (командний інтерпретатор, який використовується в Unix сумісних операційних системах) на ПК, тобто виконувати команди, які подає користувач, або які читаються з файлів, але і тунелювати графічний інтерфейс — X Tunnelling (тільки для Unix-подібних ОС або програм, що використовують графічний інтерфейс X Window System). SSH здатний передавати через безпечний канал будь-який інший мережевий протокол (Port Forwarding), забезпечуючи можливість безпечної передачі не тільки X-інтерфейсу, але і, наприклад, звуку та відео.

Для роботи по протоколу SSH потрібен SSH-сервер і SSH-клієнт. SSH-сервер прослуховує з'єднання від клієнтських ПК і при встановленні зв'язку виконує аутентифікацію, після чого починає обслуговування клієнта. SSH-клієнт використовується для входу на віддалений ПК і виконання команд. Для з'єднання SSH-сервер і SSH-клієнт повинні створити пари відкритих і закритих ключів та обмінятися ними, при цьому також використовується пароль.

Програма роботи

1. Встановити FTP-сервер та перевірити можливість підключення до нього та передачу даних.
2. Встановити TFTP-сервер та перевірити можливість підключення до нього та передачу даних.
3. Встановити Telnet-сервер та перевірити можливість

підключення до нього та передачу даних.

4. Встановити SSH-сервер та перевірити можливість підключення до нього та передачу даних.

Порядок виконання роботи

1. Переконатись, що при підключенні віртуальної машини встановлений тип підключення мережевого адаптера: *проміжний адаптер/мережевий міст*. Запустити ОС Debian у віртуальній машині, видалати *Dnsmasq* командою *sudo apt-get purge dnsmasq*. Занотувати IP-адресу віртуальної машини після виводу даних команди *sudo ip a*.

2. Виконати встановлення сервера *vsftpd* (Very Secure FTP Daemon). Для цього виконуємо команду *sudo apt-get install vsftpd*, а також встановити FTP-клієнт командою *apt install ftp*. Налаштувати сервер, відредагувавши файл конфігурації */etc/vsftpd.conf*. У цьому файлі знаходимо рядок *listen* замінюємо у ньому значення з *NO* на *YES*.

Для цього, подібно, як у попередній лабораторній роботі використовуємо файловий менеджер *MC* запуск якого здійснюється командою *sudo mc* (відповідно без *sudo*, якщо ви працюєте під користувачем *root*). Навігація по об'єктах у каталозі здійснюється клавішами керування курсором, перехід у батьківський каталог (на рівень вище) – вибором *.. (дві крапки)*, перегляд текстового файлу – натисканням *F3*, редагування – *F4* (рис. 1).

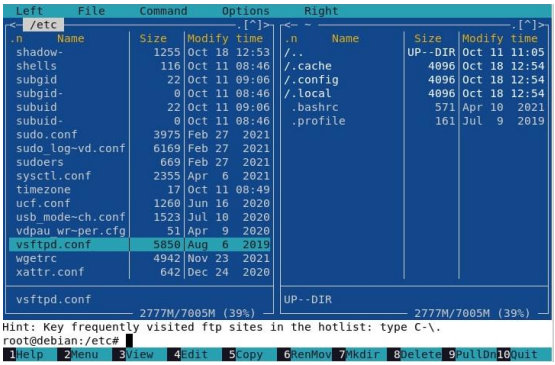


Рис. 1. Вигляд вмісту папки */etc/* з виділеним файлом *vsftpd.conf* у файловому менеджері *MC*

Для редагування файлу *vsftpd.conf* використовуємо текстовий редактор *Nano* (див. попередні лабораторні роботи) (рис. 2).

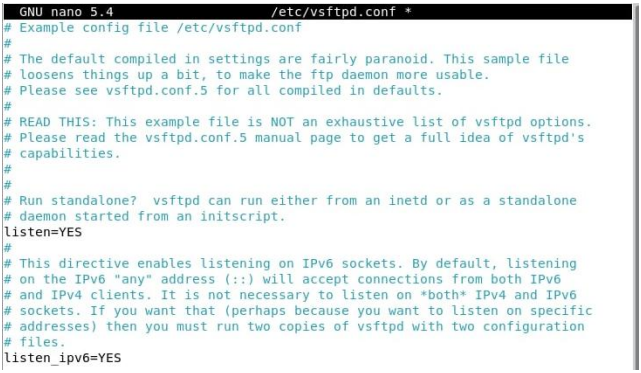


Рис. 2. Вміст файлу *vsftpd.conf* при його редагуванні текстовим редактором *Nano*

Після виконання змін перезапускаємо FTP-сервер командою *sudo service vsftpd restart* (відповідно без *sudo*, якщо ви працюєте під користувачем *root*).

3. Перевірити можливість логіну (входу) на сервер командою *ftp localhost*, ввівши логін та пароль користувача. У випадку, якщо логін і пароль *ftp*-користувача співпадає з логіном і паролем локального користувача (ОС Debian) вхід буде виконано без запиту пароля, у іншому випадку необхідно буде ввести логін і пароль, який, як правило, відповідає локальному (той самий що при вході в Debian). При успішному підключенні буде виведено запрошення вводу команд: *ftp>* (рис. 3).

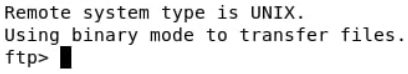


Рис. 3. Вигляд виконання команди при вході на *ftp*-сервер

3. Під'єднатись з хост-системи до віртуальної машини як до віддаленого хосту за допомогою FTP-клієнта (використавши IP-адресу гостьової системи) та створити у будь-якій папці новий текстовий файл на сервері. Скористаємося вбудованим FTP-клієнтом ОС Windows. Для цього у вікні *Провідника* потрібно

перейти в меню «*Комп'ютер*», а потім обрати ярлик: «*Підключити мережевий диск > Підключитися до веб-сайту, де можна зберігати документи та зображень»*. При цьому, використати IP-адресу збережену з п. 1, та виконати всю послідовність дій по підключенню (рис. 4).

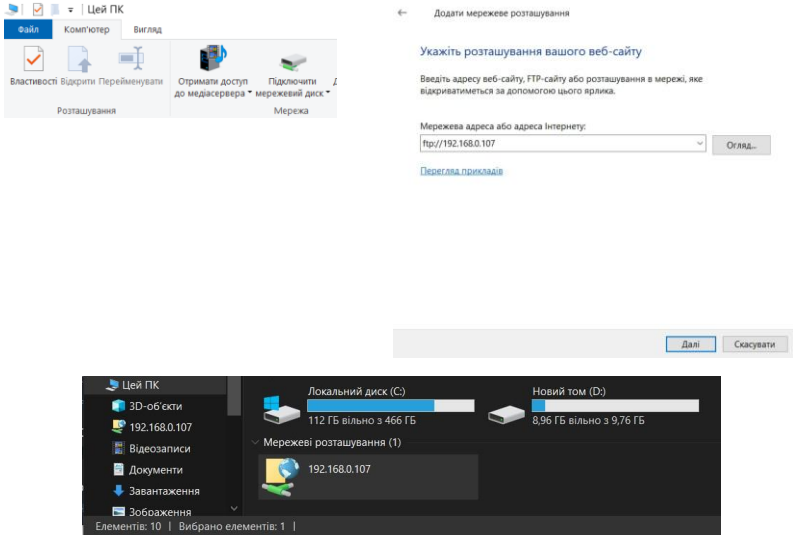


Рис. 4. Процес підключення хост-системи до віртуальної машини

Після цього потрібно повернутись до терміналу гостьової системи (в режимі віртуальної машини у ОС Debian) та вивести вміст зміненої теки командою *ls -la ~*. При цьому, отримаємо результат, який показано на рис. 5 та зберігаємо його у звіт.

```
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls -la ~
output to local-file: /root?
200 EPRT command successful. Consider using EPSV.
150 Here comes the directory listing.
local: /root: Is a directory
226 Directory send OK.
225 No transfer to ABOR.
ftp>
```

Рис. 5. Опис вмісту зміненої теки

5. Встановити TFTP-сервер TFTPD-HPA командою *sudo apt-get install tftpd-hpa*. Змінити налаштування сервера у файлі */etc/default/tftpd-hpa*, вказавши в якості шляху до кореневого каталога файлової системи для клієнтів каталог */srv/tftp* (може бути вказаний за замовченням) (рис. 6). Після цього скопіювати в цей каталог файл *timezone*. Це можна зробити використовуючи програму *MC*, тобто відкрити обидва каталоги у різних вікнах програми *MC* та виконати копіювання так, як показано на рис. 7.

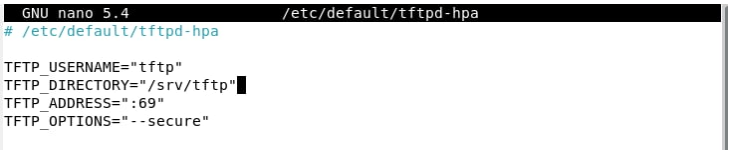


Рис. 6. Параметри налаштування сервера у файлі */etc/default/tftpd-hpa*

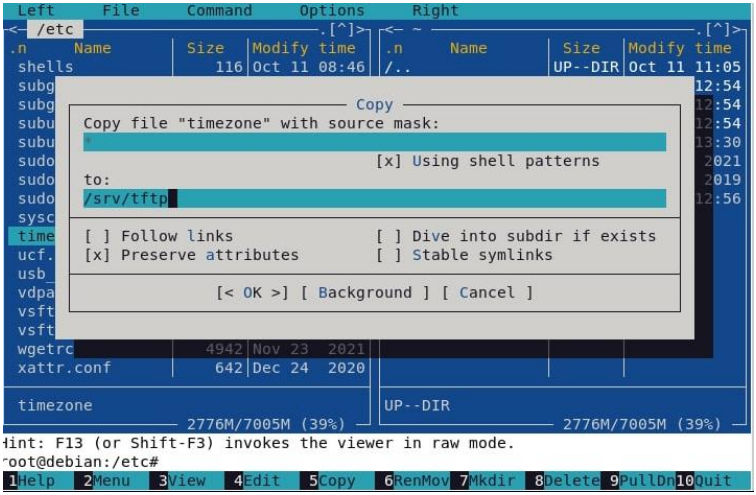


Рис. 7. Процес копіювання файлу *timezone* в каталог */srv/tftp*

Після цього, перезапустити TFTP-сервер використовуючи команду service *tftpd-hpa restart*. Під'єднатись до нього з хост-системи (ОС Debian) за допомогою безкоштовної програми TFTPD32, яку можна завантажити з сайту *https://bitbucket.org/phjounin/tftpd64/downloads/* та скопіювати

файл *timezone* на хост-систему так, як показано на рис. 8. При цьому, файл з вказаного сервера скопіюється на локальний комп'ютер у вказаний каталог. Таким чином, ми встановили TFTP-сервер та перевірили можливість підключення до нього та виконали передачу даних.

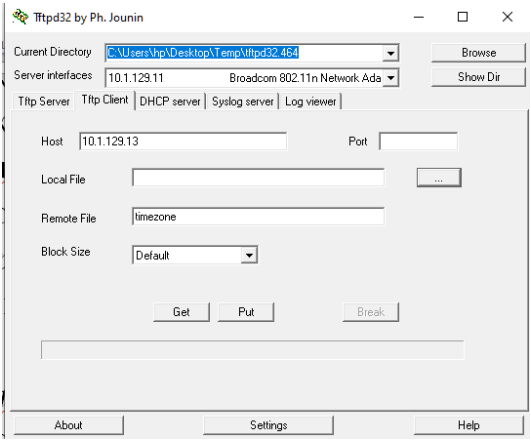


Рис. 8. Копіювання файлу *timezone* на хост-систему

6. Встановити *Telnet*-сервер виконавши команду: *sudo apt-get install telnetd*. Під'єднатись до гостьової системи за протоколом *Telnet* за допомогою програми *PuTTY* (*putty.exe (the SSH and Telnet client itself)*), яку можна запустити з сайту *https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html*. У вікні *PuTTY* виконати команду, наприклад *uname -a*. Зберегти результат у звіт та після виконання всіх дій відключитись від *Telnet*-сервера.

7. Підключитись до гостьової системи за допомогою SSH. Це можна зробити використовуючи ту ж саму програму *PuTTY* вказавши відповідний тип SSH (рис. 9).

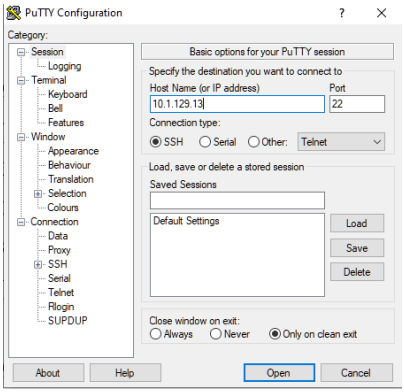


Рис. 9. Робота програми PuTTY в режимі SSH

Після цього отримаємо повідомлення з відображенням ключа для підключення до сервера, яке копіюємо у звіт (рис. 10). Застосування ключа дасть змогу перевірити, що підключення здійснюється справді до потрібного вузла, а не вузла зловмисника з іншим ключем.

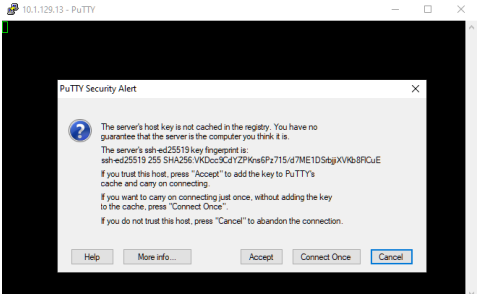


Рис. 10. Відображення ключа з програми PuTTY в режимі SSH

Зауваження. Якщо для входу буде використовуватися логін *root*, то потрібно дозволити його вхід з паролем змінивши у файлі */etc/sshd/sshd_config* рядок з налаштуванням *PermitRootLogin* на значення *yes* видаливши встановлений за замовчуванням параметр. Після цього перезавантажити SSH-сервер командою *service sshd restart*.

Після виконання з'єднання видаліть *TFTP*- та *Telnet*-сервери командою *sudo apt-get purge tftpd-hpa telnetd*.

8. Результати виконання команди з використанням протоколу SSH скопіюйте в звіт. Після завершення вказаних дій виконайте команду *sudo poweroff*.

9. Результати виконання всіх пунктів оформити у вигляді звіту на стандартних аркушах формату А4.

Вимоги до оформлення звіту:

Звіт повинен містити:

- титульну сторінку;
- мету роботи;
- програму роботи;
- скріншоти з пунктів порядку виконання роботи;
- скріншоти файлів усіх налаштувань;
- висновок.

Контрольні запитання

1. Яке призначення й особливості протоколу Telnet?
2. Яке призначення й особливості протоколу SSH?
3. Яке призначення й особливості протоколу FTP?
4. Яке призначення й особливості протоколу TFTP?
5. Як запустити FTP-клієнт у Windows?
6. Як перезапустити сервер VSFTPD?

Лабораторна робота №8
Застосування системи виявлення вторгнень в комп'ютерну систему

Мета роботи

Ознайомитися з існуючими системами виявлення та запобігання вторгнень. Навчитись застосовувати та налаштовувати систему виявлення вторгнень OSSEC.

Теоретичні відомості

Система виявлення атак (вторгнень) — програмний або апаратний засіб, призначений для виявлення фактів несанкціонованого доступу в комп'ютерну систему або мережу або несанкціонованого управління ними через Інтернет. Системи виявлення вторгнень (англ. Intrusion Detection System, IDS) забезпечують додатковий рівень захисту комп'ютерних систем разом з системою запобігання вторгненням (англ. Intrusion Prevention System, IPS).

IDS можуть сповістити про початок атаки на мережу, причому деякі з них здатні виявляти раніш не відомі атаки. IPS не обмежуються лише сповіщенням, але й здійснюють різні заходи, спрямовані на блокування атаки (наприклад, розрив з'єднання або виконання скрипта, який заданого адміністратором). На практиці досить часто програмно-апаратні рішення поєднують у функціональність двох типів систем, тоді їх називають IDPS (IDS і IPS).

Мережеві та системні IDS. Мережеві (Network-based IDS, NIDS) контролюють пакети в мережевому оточенні і виявляють спроби зловмисника проникнути всередину системи або реалізувати атаку «відмова в обслуговуванні». Ці IDS працюють з мережевими потоками даних. Типовий приклад NIDS — система, яка контролює велику кількість TPC-запитів на з'єднання (SYN) з багатьма портами на обраному комп'ютері, виявляючи, таким чином, що хтось намагається здійснити сканування TCP-портів. Мережева IDS може запускатися або на окремому комп'ютері, який контролює свій власний трафік, або на виділеному комп'ютері, прозоро «переглядає» весь потік даних у мережі (концентратор, маршрутизатор). Мережеві IDS

контролюють багато комп'ютерів, тоді як інші IDS контролюють тільки один. Прикладом мережевої IDS є система виявлення вторгнень запобігання атак - Snort, яка, комбінує методи зіставлення по сигнатурам, засоби інспекції мережевих протоколів і механізми для виявлення аномальних дій.

IDS, які встановлюються на хості і виявляють зловмисні дії на ньому, називаються хостовими або системними IDS. Прикладами хостових IDS можуть бути системи контролю цілісності файлів, які перевіряють системні файли з метою визначення, коли в них були внесені зміни. Системи моніторингу реєстраційних файлів (Log-file monitors, LFM), контролюють реєстраційні файли, створювані мережевими сервісами і службами. Прикладом хостової IDS є система OSSEC.

Статичні і динамічні IDS. Статичні засоби роблять «знімки» (snapshot) середовища та здійснюють їх аналіз, розшукуючи вразливе ПЗ, помилки в конфігураціях та ін. Статичні IDS перевіряють версії прикладних програм на наявність відомих вразливостей і слабких паролів, перевіряють вміст спеціальних файлів в директоріях користувачів або перевіряють конфігурацію відкритих мережевих сервісів. Статичні IDS виявляють сліди вторгнення.

Динамічні IDS здійснюють моніторинг у реальному часі всіх дій, що відбуваються в системі, переглядаючи файли аудиту або мережні пакети, що передаються за певний проміжок часу. Динамічні IDS виконують аналіз в реальному часі і дозволяють постійно стежити за безпекою системи.

Програма роботи

1. Ознайомитися з існуючими системами виявлення та запобігання вторгнень з теоретичних відомостей.

2. Встановити систему виявлення вторгень OSSEC, налаштувати та перевірити її роботу.

Порядок виконання роботи

1. Запутити віртуальну машину Oracle VM VirtualBox і у налаштуванні підключення до мережі вказуємо — *проміжний адаптер/мережевий міст*. Оновити індекс доступного в

репозиторії програмного забезпечення командою *sudo apt-get update*. Встановити наявні оновлення командою *sudo apt-get upgrade.*

Зауваження. Якщо вхід в ОС Debian здійснено під користувачем *root*, то команду *sudo* можна не застосвувати.

2. Встановити пакети, необхідні для роботи системи OSSEC: *sudo apt-get install build-essential apache2 libapache2-mod-php apache2-utils libpcre2-dev make zlib1g-dev libpcre2-dev libevent-dev libssl-dev libz-dev libsystemd-dev.* Встановити систему керування версіями *Git* командою *sudo apt-get install git.*

Зауваження. У деяких випадках для встановлення певних пакетів необхідно, у налаштуванні підключення до мережі віртуальної машини, перейти з *проміжний адаптер/мережевий міст* на *NAT*. Для повторного введення тих самих команд можна використовувати на клавіатурі стрілки ↑↓ верх/вниз.

3. Завантажити останні версії системи OSSEC та web-інтерфейсу до неї з Git-репозиторіїв за вказаними адресами https://github.com/ossec/ossec-hids, https://github.com/ossec/ossec-wui командою *git clone адреса_репозиторію.git*. Після цього встановлюємо оновлення командою *sudo ./install.sh.* Щоб виконати інсталяцію потрібно перейти у відповідний каталог *ossec-hids*, командою *cd.* Під час інсталяції краще вибирати англійську мову (*en*) та виконувати всі необхідні дії слідкуючи за процесом (як правило, відповідь на запит системи *y/n* необхідно вибирати *n*). Після виконання інсталяції запускаємо систему OSSEC командою *sudo /var/ossec/bin/ossec-control start* (рис. 1).

```
Cloning into 'ossec-wui'...
remote: Enumerating objects: 205, done.
remote: Total 205 (delta 0), reused 0 (delta 0), pack-reused 205
Receiving objects: 100% (205/205), 217.04 KiB | 1.94 MiB/s, done.
Resolving deltas: 100% (69/69), done.
root@debian:~# cd ossec-hids
root@debian:~/ossec-hids# ./install.sh
```

Рис. 1. Виконання інсталяції системи OSSEC та web-інтерфейсу до неї

4. Встановлюємо web-інтерфейс системи OSSEC. Для цього вміст каталогу *./ossec-wui* (необхідно попередньо знайти місце завантаження каталогу *./ossec-wui*) копіюємо в каталог */var/www*. Це можна зробити за допомогою програми *MC* (як в попередніх

лабораторних роботах) (рис. 2).

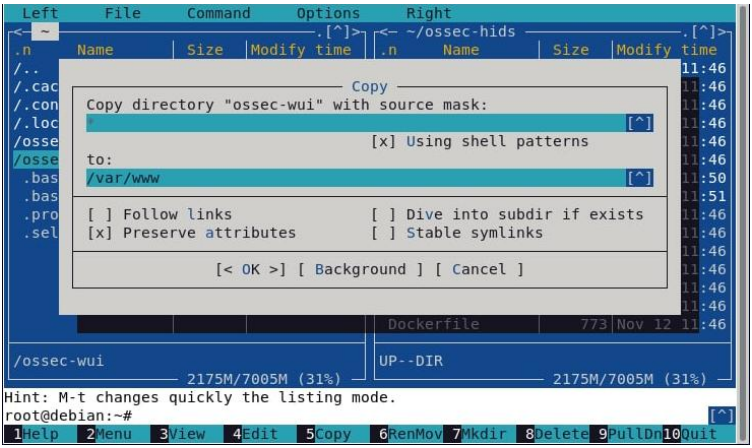


Рис. 2. Копіювання вмісту каталогу *./ossec-wui* в каталог */var/www*

Визначити каталог *./ossec-wui*, як кореневий каталог web-сервера. Для цього у файлі */etc/apache2/sites-enabled/000-default.conf* знайти рядок *DocumentRoot* і встановити властивість *DocumentRoot /var/www/ossec-wui,* зберегти зміни (рис. 3).

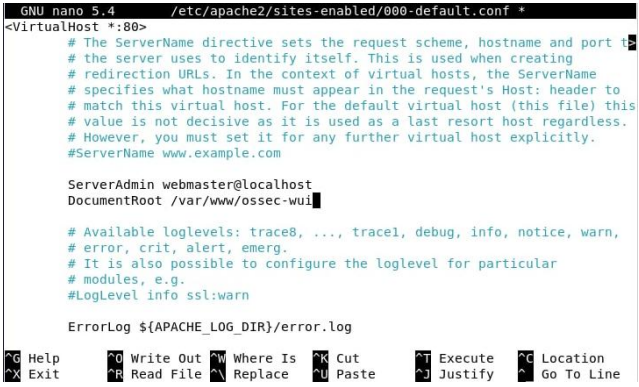


Рис. 3. Надання каталогу *./ossec-wui* властивостей кореневого каталогу web-сервера

Потім потрібно перейти у відповідний каталог *ossec-wui*, командою *cd* і виконати команду *ossec-wui ./setup.sh.* Після цього ввести (створити) логін і пароль, вибрати ім'я сервера *www-data.*

5. Перезапустити web-сервер командою *sudo service apache2 restart* (Рис. 4) і відкрити сторінку у браузері http://IP-адреса_ПК_зі_встановленою_OSSEC/. При цьому, потрібно ввести IP-адресу віртуальної машини, яку можна перевірити по команді *IP a.*

```
bredemar@debian:~$ su -l
Password:
root@debian:~# service apache2 restart
root@debian:~#
```

Рис. 4. Перезапуск web-сервера

Після цього, за допомогою системи виявлення вторгнень ми зможемо проаналізувати всі події, які відбувалися з нашим сервером (рис. 5).

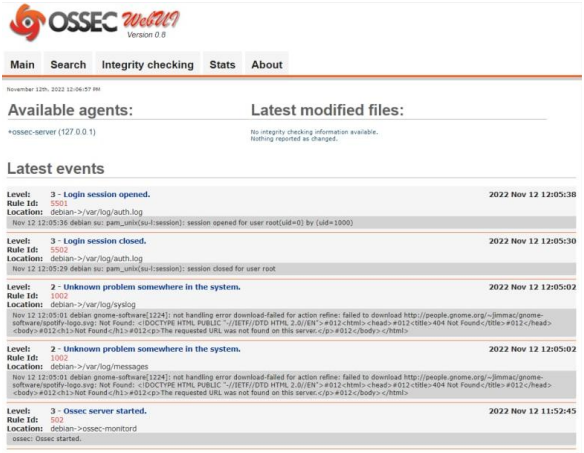


Рис. 5. Зовнішній вигляд системи OSSEC

6. Для перевірки дієвості системи виявлення вторгнень виконати сканування вузла (ПК) зі встановленою системою OSSEC за допомогою *Nmap* (рис. 6), занести в звіт результати.

Рис. 6. Сканування вузла (ПК) зі встановленою системою OSSEC за допомогою *Nmap*

7. Результати виконання всіх пунктів оформити у вигляді звіту на стандартних аркушах формату А4.

Вимоги до оформлення звіту:

Звіт повинен містити:

- титульну сторінку;
- мету роботи;
- програму роботи;
- скріншоти виконання всіх пунктів, які описані у порядку виконання роботи;
- описи процесу виконання з поясненнями виконаних дій;
- скріншоти виконання та результати сканування системи з OSSEC за допомогою Nmap;
- висновок.

Контрольні запитання

1. Що таке IDS?
2. Як класифікують IDS залежно від середовища аналізу?
3. Чим відрізняються динамічні IDS від статичних?
4. Для чого призначений log-file monitor?
5. Чим відрізняється IPS від IDS?

Лабораторна робота №9
Особливості використання елементів розробки web-сайтів

Мета роботи

Навчитись використовувати елементи розробки та використання web-сайтів та організовувати їх захист

Теоретичні відомості

Web-сервер (англ. Web Server) — це сервер, що приймає HTTP-запити від клієнтів, зазвичай веб-браузерів, видає їм HTTP-відповіді, зазвичай разом з HTML-сторінкою, зображенням, файлом, медіа-потоком або іншими даними.

Також Web-сервером називають програмне забезпечення, що виконує функції web-сервера, так і комп'ютер, на якому це програмне забезпечення працює. У даній лабораторній роботі під терміном Web-сервер ми розумітимемо саме програмне забезпечення. Серед найбільш поширеного програмного забезпечення, яке виконує функції web-сервера є Apache HTTP-сервер.

Apache HTTP-сервер — це відкритий веб-сервер для UNIX-подібних, Microsoft Windows, Novell NetWare та інших операційних систем, який розробляється та підтримується спільнотою розробників відкритого програмного забезпечення під керівництвом Apache Software Foundation. Якщо користувач в рядку адреси браузера не вказав шлях до файлу, а лише адресу сайту, за замовчуванням web-сервер надсилає у відповідь файл *index.html*, *index.php* або інший, вказаний у налаштуваннях сервера. Каталог у файловій системі, у якому розміщений цей файл та інші файли і каталоги сайту, повинен бути визначений, як кореневий каталог web-сервера. Браузер користувача не може отримати доступ до файлів, які знаходяться за межами кореневого каталогу сервера (якщо у кореневому каталозі або у підкаталогах немає посилань за його межі).

Для динамічного створення HTML-сторінок у відповідь на запити користувача часто використовується мова PHP – інтерпретована мова програмування, код якої можна вбудовувати безпосередньо в *html-код* web-сторінок. Код PHP в HTML повинен знаходитись між початковим тегом *<?php* та

кінцевим ?> (або між <*script language=ƀhp*> та </*script*>). Дані, необхідні для генерування web-сторінки-відповіді користувачеві, як правило, зберігаються в базах даних (БД). Одією з найпоширеніших систем управління базами даних (СУБД) є MySQL. *MySQL-сервер* виконує обробку SQL-запитів від інших програм, оновлення і керування реляційними БД, створення схеми бази даних і її модифікації, системи контролю за доступом до бази даних.

Система керування вмістом (англ. Content Management System, CMS) — це програмне забезпечення для організації web-сайтів чи інших інформаційних ресурсів в Інтернеті чи в окремих комп'ютерних мережах. Основні функції CMS: надання інструментів для створення вмісту web-сторінок, організація спільної роботи над вмістом, зберігання, контроль версій, дотримання режиму доступу, управління потоком документів, публікація вмісту, представлення інформації у вигляді, зручному для навігації, пошуку.

Велика частина сучасних систем управління вмістом реалізується у вигляді візуального *WYSIWYG редактора* – програми, яка створює html-код зі спеціальної спрощеної розмітки, що дозволяє користувачеві простіше додавати, редагувати й керувати вмістом сайту. Сайти, що використовують CMS, потребують для своєї роботи власне web-сервер, сховище даних (як правило, СУБД) і додаток, який власне реалізовує CMS (написаний на PHP, Perl або інших мовах програмування).

Однією з популярних CMS є *WordPress* — це проста у встановленні та використанні система керування вмістом з відкритим кодом. Сфера її застосування від блогів до складних web-сайтів. Вбудована система тем і плагінів в поєднанні з вдалою архітектурою дозволяє конструювати на основі WordPress практично будь-які web-проекти. Написана на мові програмування PHP з використанням бази даних MySQL. Оскільки WordPress є широковживаною системою, вона також є об'єктом кібератак. Тому при її використанні дуже важливо здійснювати захист web-сайтів.

Одним з популярних елементів захисту розроблених за допомогою WordPress сайтів є плагін (модуль) Wordfence. *Wordfence* здійснює сканування сайту на наявність вірусів,

шкідливих програм, троянських програм, шкідливих посилань і т.п. За допомогою Wordfence можна застосувати двоетапну авторизацію, заблокувати несанкціоновані спроби входу на сайт, додати у чорний список небажані IP-адреси або боти, які навантажують сервер, а також моніторити, з яких IP-адрес заходить в систему адміністратор сайту.

Програма роботи

1. Розглянути особливості використання різних елементів для розробки та використання web-сайтів з теоретичних відомостей.
2. Встановити web-сервер Apache, систему керування базами даних MySQL, інтерпретатор мови PHP, CMS Wordpress.
3. Навчитися використовувати можливості плагіну Wordfence виявляти несанкціоновані зміни структури web-сайтів.

Порядок виконання роботи

1. Запутити віртуальну машину Oracle VM VirtualBox, задавши тип підключення до мережі — *проміжний адаптер/мережевий міст*. Оновити індекс доступного в репозиторії програмного забезпечення командою *sudo apt-get update*. Встановити наявні оновлення командою *sudo apt-get upgrade*
2. Встановити найновіші компоненти, необхідні для роботи CMS Wordpress: web-сервер Apache, систему керування базами даних MySQL, інтерпретатор мови PHP, а також phpMyAdmin та MySQL-клієнт для адміністрування, командою *sudo apt-get install apache2 php8.2 php7.4-mysql mariadb-server mariadb-client.*

Зауваження. У даному прикладі вказано команди для інсталяції *php8.2 php8.2-mysql* з версією 8.2 (https://packages.debian.org/bookworm/php). Але якщо під час інсталяція ОС Debian повідомить про те, що версія вказаних продуктів застаріла, або не існує, то потрібно визначити найновіші, на даний час версії і встановити їх.

3. Завантажити CMS Wordpress командою *wget https://uk.wordpress.org/latest-uk.tar.gz.*
4. Розпакувати архів з CMS Wordpress командою *tar -xzvf latest-uk.tar.gz.*

5. За допомогою MySQL-клієнта під'єднатись до СУБД командою *mysql -u root -p* та виконати команди створення бази даних з ім'ям *wordpressdb* і надання користувачеві *wordpressuser* з паролем *passwordforwp* всіх прав при роботі з усіма таблицями цієї БД. Виконати вихід з MySQL-клієнта командою *exit*.

6. Створити файл конфігурації системи керування вмістом. Для цього відкрити зразок файлу конфігурації командою *nano ./wordpress/wp-config-sample.php* та вказати дані, потрібні для доступу до бази даних. Замість фрагментів тексту *database_name_here, username_here, password_here* вписати вказані раніше ім'я бази даних логін та пароль так, як показано на рис 1. Зберегти змінений файл з ім'ям *wp-config.php* (Ctrl+O для збереження файлу, Y для підтвердження, Ctrl+X для виходу).

Рис. 1. Внесення змін у файл *wp-config-sample.php*

7. Перемістити каталог із сконфігурованою CMS у */var/www* командою *sudo cp -R ./wordpress /var/www* (рис. 2).

```
root@debian:~# nano ./wordpress/wp-config-sample.php
root@debian:~# nano ./wordpress/wp-config-sample.php
root@debian:~# sudo cp -R ./wordpress /var/www
root@debian:~#
```

Рис. 2. Виконання переміщення сконфігурованої CMS

2. У файлі налаштувань web-сервера Apache вказати в якості кореневого каталогу web-сервера шлях до каталогу

/var/www/wordpress: вікрити файл командою *sudo nano /etc/apache2/sites-enabled/000-default.conf* та замінити значення властивості DocumentRoot на */var/www/wordpress*. Зберегти зміни (рис. 3).

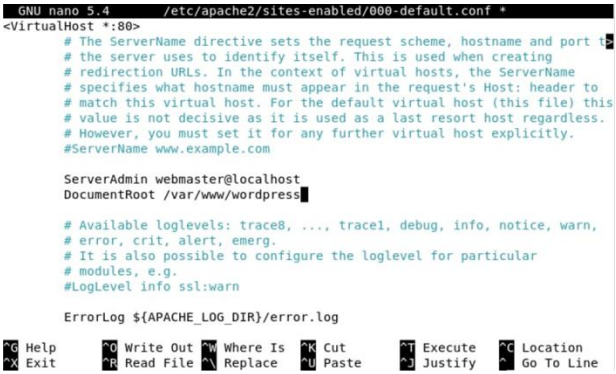


Рис. 3. Внесення змін у файл *000-default.conf*

9. Перезапустити web-сервер командою *sudo service apache2 restart.*

10. Ввести в рядок адреси web-браузера IP-адресу сервера (команда *ip a*), перейти на сторінку встановлення CMS Wordpress та виконати встановлення системи керування вмістом (ввести необхідну інформацію для створення сайту).

11. Після встановлення увійти з вашим логіном та паролем у адмін-панель CMS Wordpress, додати новий запис, останній рядок якого повинен містити ваше прізвище, та натиснути «Опублікувати». Відкрити знову головну сторінку сайту та переконатись, що запис опубліковано (web-сторінку створено).

12. Встановити плагін Wordfence. Для цього у адмін-панелі CMS Wordpress необхідно перейти в розділ плагіни і додати новий плагін Wordpress. Крім того, потрібно змінити налаштування встановленого раніше FTP-сервера вказавши в якості кореневого каталога */var/www/wordpress*, відредагувавши файл конфігурації */etc/vsftpd.conf*.

13. Перевірити можливості плагіну Wordfence виявляти несанкціоновані зміни структури. Для цього необхідно додати у файл *wp-login.php* за допомогою текстового редактора *nano*

рядок *echo 'wordfence Test'* і запустити перевірку у браузері. Плагін Wordfence буде шукати всі відмінності файлів встановленого WordPress від оригінальних і видасть інформацію про те, що відбулися несанкціоновані зміни.

14.Результати виконання всіх пунктів оформити у вигляді звіту на стандартних аркушах формату А4.

Вимоги до оформлення звіту:

Звіт повинен містити:

- титульну сторінку;
- мету роботи;
- програму роботи;
- скріншоти виконання всіх пунктів, які описані у порядку виконання роботи;
- описи процесу виконання з поясненнями виконаних дій;
- висновок.

Контрольні запитання

1. Що таке веб-сервер?
2. Які родини операційних систем підтримуються веб-сервером Apache?
3. Що таке PHP?
4. Як веб-сервер визначає, що в HTML-коді веб-сторінки є фрагмент PHP-коду, який потрібно передати для обробки PHP-інтерпретатору?
5. Що таке MySQL?
6. Які функції систем керування вмістом (контентом)?
7. Які програмні продукти необхідні для встановлення і роботи CMS Wordpress?
8. Що таке пагін Wordfence і як він працює?

Лабораторна робота №10
Аналіз та діагностика комп'ютерних мереж

Мета роботи

Навчитись використовувати засоби аналізу та діагностики комп'ютерних мереж для моніторингу та виявлення проблем при перегляді ARP-таблиць, передачі ехо-запитів, дослідженні шляху до вузлів, скануванні комп'ютерів у мережі.

Теоретичні відомості

ARP (англ. Address Resolution Protocol — протокол визначення адрес) — мережевий протокол, призначений для перетворення IP-адрес (адрес мережевого рівня) в MAC-адреси (адреси канального рівня) в мережах TCP/IP. Він визначений в стандарті RFC 826.

Перетворення адрес виконується шляхом їх пошуку за спеціальною таблицею. Ця таблиця називається ARP-таблицею, зберігається у пам'яті і містить рядки для кожного вузла мережі. В двох стовпчиках містяться IP- та Ethernet-адреси. Якщо потрібно перетворити IP-адресу в Ethernet-адресу, то відбувається пошук запису з відповідною IP-адресою. У ході звичайної роботи мережева програма відправляє прикладне повідомлення, користуючись транспортними послугами TCP. Модуль TCP посилає відповідне транспортне повідомлення через модуль IP. В результаті, складається IP-пакет, який має бути переданий драйверу Ethernet. IP-адреса місця призначення відома прикладній програмі, модулю TCP та IP. Необхідно на її основі знайти Ethernet-адресу місця призначення. Для пошуку відповідної Ethernet-адреси використовується ARP-таблиця.

ping — службова комп'ютерна програма (утиліта), призначена для перевірки з'єднань в мережах на основі TCP/IP. Вона відправляє запити (англ. Echo-Request) протоколу ICMP зазначеному вузлу мережі й фіксує відповіді (англ. Echo-Reply). Час між відправленням запиту й одержанням відповіді (RTT, від англ. Round Trip Time) дозволяє визначати двосторонні затримки у маршруті й частоту втрати пакетів, тобто побічно визначати завантаженість каналів передачі даних і проміжних пристроїв.

Повна відсутність ICMP-відповідей може також означати, що віддалений вузол (або якийсь із проміжних маршрутизаторів) блокує ICMP Echo-Reply або ігнорує ICMP Echo-Request.

Утиліта *ping* є одним з основних діагностичних засобів у мережах TCP/IP і входить у поставку всіх сучасних мережевих операційних систем. Функціональність утиліти *ping* також реалізована в деяких вбудованих операційних системах маршрутизаторів, доступ до результатів виконання *ping* для таких пристроїв за протоколом SNMP визначається відповідними стандартами.

Для отримання шляху проходження пакетів до певного вузла мережі використовують утиліту *tracert/traceroute*. Це службова комп'ютерна програма, яка призначена для визначення маршрутів прямування даних в мережах TCP/IP. *Traceroute* може використовувати різні протоколи передачі даних в залежності від операційної системи пристрою. Такими протоколами можуть бути UDP, TCP, ICMP або GRE. Комп'ютери з встановленою операційною системою Windows використовують ICMP-протокол, операційні системи Linux і маршрутизатори Cisco – протокол UDP. Приклад виконання програми для пошуку шляху до сервера з сайтом НУВГП наведений нижче:

>> tracert nuwm.edu.ua

Tracing route to nuwm.edu.ua [109.87.215.51]

over a maximum of 30 hops:

1 1 ms 1 ms 1 ms FV1 [192.168.2.1]

2 10 ms 1 ms 2 ms ip-37-221-140-1.airbites.net.ua [37.221.140.1]

3 1 ms 2 ms 2 ms c76-rv-dot1q-330.airbites.net.ua [188.230.88.5]

4 6 ms 7 ms 7 ms vl1526.c76-kv-g50.valor.ua [188.230.88.242]

5 6 ms 7 ms 7 ms vl1523.c76-kv-l9.valor.ua [188.230.118.185]

6 7 ms 6 ms 6 ms mirohost-2-ix.giganet.ua [185.1.62.9]

7 15 ms 14 ms 15 ms uck-rivne.ett.ua [78.154.162.38]

*8 * * * Request timed out.*

9 14 ms 13 ms 15 ms nuwm.rv.ua [109.87.215.51]

Для сканування цілих мереж найчастіше використовується утиліта *nmap* (Network Mapper) – це безкоштовне відкрите програмне забезпечення для дослідження та аудиту безпеки мереж і виявлення активних мережевих сервісів. З часу публікації в 1997 р. такий аудит став стандартом в галузі інформаційної безпеки. *Nmap* використовує багато різних методів сканування, таких як UDP, TCP (connect), TCP SYN (напіввідкрите), Reverse-ident, ICMP (ping), FIN, ACK, Xmas tree, SYN- NULL-сканування. *Nmap* також підтримує великий набір додаткових можливостей, а саме:

- визначення операційної системи віддаленого хоста з використанням відбитків стека TCP/IP;
- «невидиме» сканування;
- динамічне обчислення часу затримки і повтор передачі пакетів;
- паралельне сканування;
- визначення неактивних хостів методом паралельного ping-опитування;
- сканування з використанням помилкових хостів;
- визначення наявності пакетних фільтрів;
- пряме (без використання portmapper) RPC-сканування;
- сканування з використанням IP-фрагментації;
- довільне вказання IP-адрес і номерів портів сканованих мереж;
- можливість написання довільних сценаріїв (скриптів) на мові програмування Lua.

Синтаксис застосування утиліти *Namp*:

-sL – створює список працюючих вузлів без сканування портів;

-sP – перевіряє доступність вузла за допомогою ping;

-PN – зчитує всі доступні хости навіть якщо вони не відповідають на ping;

-sS/sT/sA/sW/sM – TCP сканування;

-sU – UDP сканування;

-sN/sF/sX – TCP NULL і FIN сканування;

-p – для вказівки діапазону портів для перевірки;

-sV – детальне дослідження портів для для визначення версій використаних служб;

-O – визначає операційну систему;
-T[0-5] – швидкість сканування;
-D – маскує сканування за допомогою фіктивних IP;
-S – змінює свою IP адресу на вказану;
--spoof-mac – визначити свою MAC адресу.

Програма роботи

1. Розглянути особливості використання засобів аналізу та діагностики комп'ютерних мереж для моніторингу та виявлення їх проблем з теоретичних відомостей

2. Отримати ARP-таблицю хост-системи та віртуальної машини.

3. Надіслати exo-запит та визначити маршрут передачі даних до вузла з хост-системи та віртуальної машини в режимах проміжного адаптера і NAT.

4. Встановити утиліту Nmap та просканувати локальну мережу.

Порядок виконання роботи

1. Запутити віртуальну машину Oracle VM VirtualBox. Оновити індекс доступного в репозиторії програмного забезпечення командою *sudo apt-get update*. Встановити наявні оновлення командою *sudo apt-get upgrade.*

2. Переглянути ARP-таблицю системи командою *sudo arp -a*. Записати результати виводу команди у звіт.

3. Перевірити час передачі пакетів до вузлів мережі, наприклад en.wikipedia.org, або інншi за допомогою утиліти ping: *ping en.wikipedia.org, google.com* та інші. Перевірити час передачі пакетів до вузла вказавши його IP-адресу. Для зупинки виконання команди *ping* використати комбінацію CTRL+C. Виконати команду *ping* в ОС Windows та порівняти дані з її виконанням в ОС Debian.

4. Отримати маршрут до використаних вузлів командою tracert: *sudo traceroute.*

Перевірити параметри команди *tracert* у Windows:

-d - без дозволів у назвах вузлів.

-h максЧисло максимальне число стрибків при пошуку вузла.

-j списокВузлів - вибір маршруту по списку вузлів (тільки для IPv4).

-w таймаут - таймаут кожної відповіді в мілісекундах.

-R - трасування шляху (тільки IPv6).

-S адресаДжерела - використовувана адреса джерела (тільки IPv6).

5. Встановити пакет *nmap*: *sudo apt-get install nmap.*

6. Виконати сканування будь-якого вузла мережі за допомогою утиліти *Nmap* (приклад введення команди: *nmap -sn 192.168.1.1/24*), а також перевірити застосування синтаксису цієї утиліти, що приведений у теоретичних відомостях. Занести результати сканування у звіт.

7. Виконати вищевказані пункти задавши тип підключення до мережі віртуальної машини Oracle VM VirtualBox — *проміжний адаптер/мережевий міст* та *NAT* і порівняти їх значення.

8. Результати виконання всіх пунктів оформити у вигляді звіту на стандартних аркушах формату А4.

Вимоги до оформлення звіту:

Звіт повинен містити:

- титульну сторінку;
- мету роботи;
- програму роботи;
- скріншоти виконання всіх пунктів, які описані у порядку виконання роботи;
- описи процесу виконання з поясненнями виконаних дій;
- висновок.

Контрольні запитання

1. Для чого призначений ARP?
2. Яку інформацію надає програма *ping*?
3. Який результат роботи програми *tracert/traceroute*?
4. Що таке Nmap?
5. Якими можливостями володіє *Nmap*?