

Міністерство освіти і науки України  
Державний університет «Житомирська політехніка»  
Факультет інформаційно-комп'ютерних технологій  
Кафедра комп'ютерної інженерії та кібербезпеки

## **ПОЯСНЮВАЛЬНА ЗАПИСКА**

до випускної кваліфікаційної роботи бакалавра  
на тему: «Проект підсистеми захисту Web-сервера на  
базі ОС Linux»

Виконала студентка 4-го курсу групи КБ-2  
спеціальності 125 «Кібербезпека»

\_\_\_\_\_ Чорнобіль Марина

Доцент кафедри ІПЗ, к.т.н., доцент

\_\_\_\_\_ Лобанчикова Надія

Рецензент ВО професор кафедри КІ та КБ, д.т.н., доцент

\_\_\_\_\_ Воротніков Володимир

Житомир – 2022

Державний університет «Житомирська політехніка»  
Факультет інформаційно-комп'ютерних технологій  
Кафедра комп'ютерної інженерії та кібербезпеки  
Спеціальність 125 «Кібербезпека»  
Освітня програма «Кібербезпека»

ЗАТВЕРДЖУЮ  
Завідувач  
кафедри комп'ютерної  
інженерії та кібербезпеки  
\_\_\_\_\_ Андрій Єфіменко  
1 серпня 2022 р.

### **ЗАВДАННЯ**

на випускн кваліфікаційну роботу бакалавра

Студентка: Чорнобіль Марина Русланівна

Тема роботи: «Проект підсистеми захисту Web-сервера на базі ОС Linux»

Затверджена Наказом університету від 1 серпня 2022 р. № 248/с

Термін здачі студентом закінченої роботи 30 серпня 2022 р.

Вихідні дані роботи (зазначається предмет і об'єкт дослідження):

Об'єктом дослідження є процес проектування підсистеми захисту Web-сервера на базі ОС Linux. Предметом дослідження є технологія проектування підсистем захисту Web-сервера на базі ОС Linux.

Обладнання та програмне забезпечення:

1. Маршрутизатор Cisco 7200;
2. Операційна система GNU/Linux Ubuntu Server;
3. Операційна система GNU/Linux Kali.
4. Мережевий емулятор GNS3.

Консультанти з випускної кваліфікаційної роботи із зазначенням розділів, що їх стосуються

| Розділ | Консультант     | Підпис, дата   |                  |
|--------|-----------------|----------------|------------------|
|        |                 | Завдання видав | Завдання прийняв |
| 1      | Лобанчикова Н.М | 01.08.2022     | 06.03.2022       |
| 2      | Лобанчикова Н.М | 06.08.2022     | 20.04.2022       |
| 3      | Лобанчикова Н.М | 16.08.2022     | 11.05.2022       |

Студентка \_\_\_\_\_ Чорнобіль Марина

Керівник \_\_\_\_\_ Лобанчикова Надія

### Календарний план

| № з/п | Назви етапів випускної кваліфікаційної роботи                                  | Термін виконання етапів роботи  | Примітка |
|-------|--------------------------------------------------------------------------------|---------------------------------|----------|
| 1     | Видача завдання                                                                | 1 серпня 2022                   | Виконано |
| 2     | Розробка плану роботи                                                          | 2 серпня 2022                   | Виконано |
| 3     | Пошук інформації стосовно вразливостей Web-серверів                            | 3 серпня 2022                   | Виконано |
| 4     | Порівняння функціоналу Apache та NGINX                                         | 6 серпня 2022 – 7 серпня 2022   | Виконано |
| 6     | Проектування мережі організації                                                | 8 серпня 2022 – 12 серпня 2022  | Виконано |
| 7     | Проведення аналізу інфраструктури мережі організації                           | 13 серпня 2022 – 15 серпня 2022 | Виконано |
| 8     | Проведення аналізу векторів кібератак та визначення методів для захисту мережі | 16 серпня 2022 – 18 серпня 2022 | Виконано |
| 9     | Налаштування Ubuntu Server та NGINX                                            | 19 серпня 2022 – 24 серпня 2022 | Виконано |
| 11    | Проведення налаштувань безпеки мережі                                          | 25 серпня 2022 – 27 серпня 2022 | Виконано |
| 12    | Тестування системи                                                             | 27 серпня 2022                  | Виконано |
| 13    | Написання пояснювальної записки                                                | 28 серпня 2022 – 29 серпня 2022 | Виконано |
| 14    | Отримання відгуку та рецензій                                                  | 29 серпня 2022                  | Виконано |
| 15    | Захист                                                                         | 30 серпня 2022                  | Виконано |

Студентка \_\_\_\_\_ Чорнобіль Марина

Керівник \_\_\_\_\_ Лобанчикова Надія

## АНОТАЦІЯ

Кваліфікаційна робота складається з проекту підсистеми захисту Web-сервера, його реалізації та пояснювальної записки до нього. Пояснювальна записка містить 63 сторінки та 46 ілюстрацій.

Метою роботи є розробка проекту підсистеми захисту Web-сервера на базі ОС Linux.

В роботі проведено аналіз векторів кібератак та визначено метод для захисту мережі, налаштовано посилення безпеки Ubuntu Server. Налаштовано захист для Web-сервера NGINX. Проаналізовано інфраструктуру віддаленого доступу.

Ключові слова: ВРАЗЛИВІСТЬ, ЗАХИСТ, WEB-СЕРВЕР, UBUNTU SERVER, NGINX, TELNET, SSH, ВЕКТОР КІБЕРАТАК.

|           |      |                  |        |      |                                                              |                |      |         |
|-----------|------|------------------|--------|------|--------------------------------------------------------------|----------------|------|---------|
|           |      |                  |        |      | КІК.КР.Б – 125 – 24 – ПЗ                                     |                |      |         |
| Змн.      | Арк. | № докум.         | Підпис | Дата | Проект підсистеми захисту<br>Web-сервера на базі ОС<br>Linux | Літ.           | Арк. | Аркушів |
| Розроб.   |      | Чорнобіль М.Р.   |        |      |                                                              |                | 5    | 67      |
| Керівник  |      | Лобанчикова Н.М. |        |      |                                                              | ФІКТ, Гр. КБ-2 |      |         |
| Реценз.   |      | Воротніков В.В.  |        |      |                                                              |                |      |         |
| Н. контр. |      | Єфіменко А.А.    |        |      |                                                              |                |      |         |
| Затверд.  |      | Єфіменко А.А.    |        |      |                                                              |                |      |         |

## ANNOTATION

The qualification work consists of a project of the Web server protection subsystem, his implementation and an explanatory note to it. The explanatory note contains 63 pages and 46 illustrations.

The purpose of the work is to develop a project of the Web server protection subsystem based on the Linux OS.

In the work, an analysis of cyber attacks vectors was carried out and a method for network protection was determined, Ubuntu Server security enhancement was configured. Security is configured for the NGINX web server. The remote access infrastructure was analyzed.

Keywords: VULNERABILITY, PROTECTION, WEB SERVER, UBUNTU SERVER, NGINX, TELNET, SSH, CYBER ATTACK VECTOR.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          | 5    |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          |      |

## ЗМІСТ

|                                                                                                       |    |
|-------------------------------------------------------------------------------------------------------|----|
| ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....                                                                        | 7  |
| ВСТУП.....                                                                                            | 8  |
| РОЗДІЛ 1 КЛЮЧОВІ АСПЕКТИ ЗАХИСТУ ТА ПРОТИДІЇ КІБЕРАТАКАМ....                                          | 10 |
| 1.1 Вразливості Web-серверів та типи загроз для Web-серверів.....                                     | 10 |
| 1.2 Теоретичні відомості щодо захисту та протидії кібератакам .....                                   | 12 |
| 1.3 Огляд найпопулярнішого програмного забезпечення для веб-серверів.....                             | 14 |
| 1.4 Огляд функціоналу Web-серверів Apache та NGINX .....                                              | 20 |
| Висновки до розділу 1 .....                                                                           | 27 |
| РОЗДІЛ 2 ПРОЕКТ ПІДСИСТЕМИ ЗАХИСТУ ДЛЯ WEB-СЕРВЕРА НА БАЗІ ОС LINUX .....                             | 28 |
| 2.1 Аналіз IT-інфраструктури.....                                                                     | 28 |
| 2.2 Аналітика інфраструктури віддаленого доступу.....                                                 | 36 |
| 2.3 Проведення аналізу векторів кібератак та визначення методів для захисту мережі підприємства ..... | 39 |
| Висновки до розділу 2 .....                                                                           | 42 |
| РОЗДІЛ 3 НАЛАШТУВАННЯ ТА РЕАЛІЗАЦІЯ ПРОЕКТУ ДЛЯ ЗАХИСТУ WEB-СЕРВЕРА НА БАЗІ ОС LINUX.....             | 43 |
| 3.1 Налаштування та посилення безпеки Ubuntu Server.....                                              | 43 |
| 3.2. Налаштування захисту для сервера NGINX .....                                                     | 51 |
| 3.3. Тестування реалізованих методів безпеки для Web-сервера.....                                     | 57 |
| Висновки до розділу 3 .....                                                                           | 57 |
| ВИСНОВКИ.....                                                                                         | 60 |
| СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ .....                                                                  | 61 |
| ДОДАТКИ.....                                                                                          | 62 |

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ПЗ – програмне забезпечення

БД – база даних

ОС – операційна система

ПБ – програмний брандмауер

PPP – Point-to-Point Protocol

RFC – Request for Components

RBAC – Role Based Access Control

CSP – Content Security Policy

SSL – Secure Sockets Layer

HSTS - HTTP Strict Transport Security

ЦС – Центр Сертифікації

SSH – Secure Shell

RPM – RPM Package Manager

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          | 7    |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          |      |

## ВСТУП

Актуальність теми. У наш час, складно уявити своє життя без використання мереж та Інтернету. Адже саме там є можливість знаходити всю потрібну інформацію, окрім того, він також виконує функцію засобу зв'язку та зберігання даних. Уся інформація в мережі Інтернет існує у вигляді даних на Web-серверах. Для знаходження питання, яке виникає у нас, ми звикли використовувати різноманітні веб-сайти, їх кількість вже переросла мільярд і щодня ця цифра лише збільшується. Веб-сайти звичайно не можуть обійтись без того ж Web-сервера. Без останніх не існувала би ні всесвітня комп'ютерна мережа, ні корпоративні мережі, у звичному вигляді.

За останній час стали частішими випадки кібератак та зламів корпоративних веб-сайтів. Це говорить про їх недостатню захищеність. Для Web-серверів безпека відіграє надважливу роль, а кібербезпека зараз стала буденною проблемою в житті підприємств. Web-сайти піддаються атакам щодня та деякі з них стають фатальними для атакованих компаній. Згідно зі звітом Cybersecurity Ventures за 2021 рік, кожні 11 секунд один бізнес ставав жертвою атаки.

Метою кваліфікаційної роботи є створення проекту підсистеми захисту Web-сервера на базі ОС Linux.

Визначена мета обумовлює наступні завдання:

- Проведення аналізу поняття безпека Web-сервера, його використання при захисті Web-сервера;
- Порівняння найпопулярнішого програмного забезпечення для Web-сервера та використання його для захисту від кібератак;
- Аналіз функціональності ПЗ APACHE та NGINX;
- Аналіз особливостей захисту інфраструктури мережі для роботи Web-сервера;
- Проведення налаштувань для захисту та протидії кібератакам на базі ОС Linux.

**Об'єктом дослідження** є процес розробки захисту Web-сервера з використанням ОС Linux.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          | 8    |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          |      |

**Предметом дослідження** є технологія захисту Web-сервера на базі ОС Linux.

У процесі виконання кваліфікаційної роботи було розроблено проект підсистеми захисту Web-сервера на базі ОС Linux.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          | 9    |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          |      |

# РОЗДІЛ 1 КЛЮЧОВІ АСПЕКТИ ЗАХИСТУ ТА ПРОТИДІЇ КІБЕРАТАКАМ

## 1.1 Вразливості Web-серверів та типи загроз для Web-серверів

В умовах сьогодення вразливості web-серверів постійно змінюються, оскільки кіберзлочинці знаходять нові методи використання прогалин в їх безпеці.

Отож як тільки починається мова про вразливості web-серверів, більша частина людей згадує про «щілини» в їх ПЗ. Це стосується і самих програм-серверів, а саме: Apache, NGINX та інших. Дане програмне забезпечення є доволі складним та об'ємним, тому «щілини» в ньому обов'язково є.

Певно, що ні для кого не є секретом те, що безпека будь-чого залежить від застосування цієї речі. Це ж можемо сказати і про web-сервер. Доволі багато залежить від налаштування ПЗ та й взагалі більшість web-серверів мають великий набір параметрів. Вони стосуються майже всіх аспектів роботи web-серверів. Тому безпека частково залежить й від адміністраторів, котрі обслуговують їх.

Отже, серед поширених вразливостей web-серверів можна виділити наступні: SQL-ін'єкції, міжсайтові сценарії (XSS), DoS атаки, CSRF. Розглянемо кожен з них детальніше:

1. SQL-ін'єкції є однією з перших атак системи безпеки, з її допомогою кіберзлочинці можуть отримати доступ до системи. За допомогою цієї атаки зловмисник може отримати доступ до бази даних web-сервера, підробити особу користувача або ж навіть знищити або змінити дані в БД. SQL-ін'єкції є надвичайно пагубними для організацій, тому що дають можливість злочинцям отримати доступ до інформації клієнтів, такої як номери кредитних карток, паролів та контактної інформації.

За останні декілька років багато відомих витоків інформації сталися саме в результаті атак, що використовували SQL-ін'єкції. Це призвело до погіршення репутацій компаній та штрафів, установлених регулюючими органами. Часом кіберзлочинець спроможний отримувати постійний канал в системі організації. Як результат, з'являється довгострокова уразливість, що може бути непомітною протягом довгого часу.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          | 10   |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          |      |

2. Міжсайтові сценарії (XSS) – це тип ін'єкцій, коли шкідливі сценарії впроваджуються на безпечні та надійні сайти. XSS-атака відбувається, під час використання зловмисником веб-програми, що надсилає шкідливий код. Зазвичай, це має вигляд у формі сценарію зі сторони браузера до іншого кінцевого користувача. Дані атаки користуються успіхом завдяки недолікам, які доволі широко поширені і виникають у місцях, де web-додаток використовує дані, що входять, від користувача в межах вихідних даних, що генеруються, без перевірки чи кодування. За допомогою даного типу атак кіберзлочинець може отримати доступ до будь-яких файлів cookies та іншої особистої інформації, що зберігає web-переглядач і використовує на даному сайті.

3. DoS-атаки, тобто атаки «відмови в обслуговуванні» спрямовані на те, щоб зробити сервер недоступним для користування в цілях, для яких він був створений. За допомогою багатьох способів можливо зробити послугу недоступною для користувача. Це може бути здійснено за допомогою маніпулювання мережевими пакетами, за допомогою програмування чи вразливості обробки ресурсів. Коли служба отримує величезну кількість запитів, вона може зупинитись, якщо було використано вразливість програми чи метод обробки ресурсів службою, котру вона використовує. Даний тип атаки значно погіршує якість обслуговування користувачів. Вони також призводять до затримок відповідей та надмірних втрат, що безумовно впливає на доступність ресурсу.

4. CSRF («Підробка міжсайтового запиту») – це атака, що примушує кінцевого користувача робити небажані дії у web-програмі чи web-сервері, де він наразі пройшов автентифікацію. Для переважної частини сайтів запити браузера автоматично містять облікові дані, що пов'язані з сайтом, такі як файли cookie, IP-адресу, облікові дані домену операційної системи. Зловмисник може примусити жертву за допомогою соціальної інженерії, надіславши електронний лист, тощо. Таким чином він змушує користувача виконати дії, що кіберзлочинець побажав. При успішній атаці жертва може виконати запити на зміну стану, як переказ коштів, зміна електронної адреси. [1]

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 11   |

То які ж поширені недоліки у безпеці web-серверів можна зустріти в просторах Інтернету? Основні вразливості трапляються на рівні ОС та web-сервера, а саме незахищені системи з прогалинами, які може використати злочинець, для отримання доступу під виглядом програми, уникаючи веб-моніторингу і систем журналювання.

Також деякі постачальники розміщують ідентифікатори юзерів та паролі за замовчуванням для web-серверів, які працюють на точках доступу, брандмауерах та навіть системах керування фізичної безпеки. Головною помилкою стає те, що мережеві адміністратори не міняють їх.

Деякі веб-додатки з використанням ідентифікації особи для входу в мережу є також незахищеними в достатній кількості. Адже будь-хто, кому відома схема найменування користувачів у даній мережі, може просто використати інший ідентифікатор користувача і зайти під його іменем в систему. Також подібною проблемою є слабкі мінімальні вимоги до паролю, який завдяки цьому можна легко вгадати чи зламати за короткий проміжок часу.

Необхідно пам'ятати що веб-браузери залишають облікові дані для входу до пам'яті на спільних комп'ютерах. Саме цим і може скористатись зловмисник просто встановивши шістнадцятковий редактор у системі. Таким чином він може здійснити пошук у пам'яті комп'ютера логіна та паролю попереднього користувача.

## **1.2 Теоретичні відомості щодо захисту та протидії кібератакам**

Нині, надзвичайно важливо захищати усі інтернет-ресурси чи додатки від доступу до них зловмисників. Розглядаючи таку задачу, одразу спадають на думку шифрування, брандмауери та використання найновіших програмних впроваджень. Подібні технічні рішення зазвичай стосуються саме інформаційної безпеки як адміністративної, так і грошової.

Можна припустити, що в умовах сьогодення кожен життєздатний бізнес чи установа налічує в собі певний вид web-присутності. Це може бути як сайт для маркетингу, так і електронна пошта чи система керування центром обробки

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          | 12   |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          |      |

даних. Також часто помітною є тенденція, коли багато мережеских адміністраторів та менеджерів із безпеки можуть не знати навіть половини web-серверів у їх мережах. Така погана обізнаність означає відсутність адміністрування, а в результаті до вразливості web-безпеки. Тоді залишається лише питання часу, коли хто-небудь натрапить на них і скористається вразливістю.

Виходячи в вищесказаного, розглянемо стратегію захисту web-сервера від кібератак.

Роботу над безпекою веб-сервера необхідно розпочати з видалення непотрібних служб. Адже ОС і конфігураціям за замовчуванням не вистачає комплексного захисту. Існує багато мережеских служб, які включені в стандартну установку, які навіть можуть не використовуватись під час роботи. Це можуть бути і віддалені служби реєстру, і служби сервера друку та інші. Велика кількість служб запущених в ОС сервера збільшує кількість відкритих портів. Тобто більше дверей в мережі, котрими можуть скористатись зловмисники. Даний етап може не лише зробити сервер безпечніший, а й підвищити його продуктивність.

Також слід створювати окремі середовища для виробництва, тестування чи розробки. Часто розробка і тестування виконуються на робочих web-серверах. Тому інколи можна зустріти сайти або сторінки в мережі Інтернет, що містять в URL-адресі деталі такі, як /new/ чи /test/. Таку тенденцію можна побачити у web-програмах, що знаходяться на ранній стадії розробки. Вони часто містять уразливі місця в безпеці, які можна використати з допомогою онлайн-інструментів, що знаходяться в вільному доступі. Отже, з цього можна зробити висновок, про необхідність зберігання розробки та тестування на серверах, що ізольовані від публічного Інтернету. І не підключаючи їх до важливих даних та баз даних.

Для безпеки web-сервера та мережі необхідно встановити дозволи та привілеї дозволів. Якщо web-сервер було скомпрометовано через ПЗ мережевої служби, то кіберзлочинець може користуватись будь-яким обліковим записом, запущеним мережевою службою, задля виконання завдань. З допомогою простого процесу встановлення мінімальних привілеїв для користувачів сервера

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 13   |

на доступ до файлів web-додатків ті внутрішніх БД може захистити від втрати чи маніпулювання даними.

Завжди необхідно відстежувати журнали сервера. Під час регулярного тестування безпеки web-сервера потрібно зберігати його журнали окремо і також часто перевіряти їх та відстежувати. Дивні записи з журналу можуть розкрити інформацію про успішні атаки та їх спроби, котрі треба обов'язково досліджувати, коли вони виникли.

І безумовно важливим аспектом безпеки є встановлення брандмауера. ПБ можна легко налаштовувати та керувати ними. Вони зможуть захистити web-сервер від несанкціонованих вторгнень та атак.

Наприкінці, потрібно не забувати про регулярне резервне копіювання сервера. Це гарантує, що при компрометуванні захисту безпеки, буде можливо швидко відновити дані. Автоматизація може підвищити ефективність, проте ІТ-працівник має перевірити сервер на наявність проблем, що могли перервати процес. [2]

### **1.3 Огляд найпопулярнішого програмного забезпечення для веб-серверів**

Web-сервер – це різновид сервера, що використовують для розміщення web-сайту та прикладного ПЗ за допомогою мережі Інтернет, та вона відповідає на запити від клієнтів через протокол HTTP (Hypertext Transfer Protocol). У широкому розумінні сервер – це пристрій, що обладнаний спеціальними програмами та підтримується ними, і це дозволяє йому запропонувати послуги для інших пристроїв, які теж користуються цією мережею.

Ціллю web-сервера є отримувати запити клієнтів і відображати вміст web-сайтів у клієнтському браузері через Інтернет.

Web-сервери є основою для веб-хостингу. Адже він завжди має підключення до Інтернету, та кожен підключений сервер має свою унікальну адресу. Дякуючи web-серверам хостинг-провайдери мають можливість керування декількома доменами на одному сервері.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 14   |

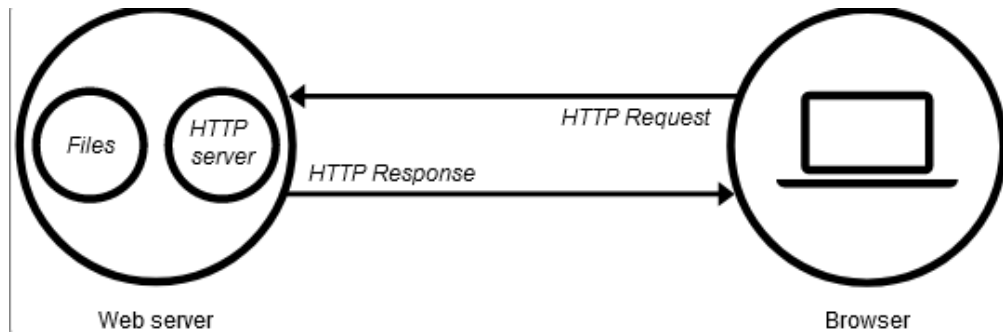


Рисунок 1.1. Принцип роботи Web-сервера

Сьогодні на відкритому ринку можливо знайти різноманітні типи web-серверів, серед яких розробники мають можливість обирати потрібний в залежності від своїх вподобань. Серед найпопулярніших типів web-серверів є:

1. Apache HTTP Web-сервер
2. Web-сервер NGINX
3. Web-сервер інформаційних служб Інтернету (IIS).
4. Web-сервер Lighttpd
5. Web-сервер Sun Java System
6. Jigsaw Server Web-сервер
7. Сервер LiteSpeed Web-сервер

Варто розглянути детально кожен із серверів:

1. Apache HTTP Web Server



Рисунок 1.2. Логотип Web-сервера Apache HTTP

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 15   |

Apache HTTP Web Server був розроблений Apache Software Foundation і є одним із найпопулярніших web-серверів нині. Даний сервер є безкоштовним ПЗ web-сервера з вихідним відкритим кодом, яке випущене відповідно до умов ліцензії Apache2.0. Основною його перевагою над конкурентами є те, що він підтримує майже усі типи ОС, такі як Windows, Linux, Mac OS, Unix, тощо. Це й стало причиною того, що близько 60% машин працюють саме на web-сервері Apache HTTP.

Ще однією причиною для популярності Apache є те, що його доволі легко налаштувати. Це відбувається завдяки модульній структурі й тому, що він є відкритим кодом. Відповідно розробник має можливість додати нові модулі у сервер і вносити будь-які необхідні зміни.

Вважається, що з усіх доступних типів web-серверів HTTP-сервери Apache є набагато стабільнішими та простішими з точки зору функціональності, особливо останні випущені версії, що можуть обробляти набагато більшу кількість клієнтських запитів у порівнянні з іншими версіями.

## 2. NGINX



Рисунок 1.3.Логотип Web-сервера NGINX

NGINX – це тип web-серверів, ПЗ яких також з відкритим кодом. Спершу ПЗ NGINX працювало як web-сервер, але тепер його також використовують як балансир навантаження, потокове передавання медіа та HTTP-кеш, а також як зворотній проксі.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 16   |

Серед переваг цього web-сервера є простота та використання меншої кількості ресурсів, що забезпечує вищий рівень продуктивності. Він також підтримує протоколи IMAP, POP3, SMTP, TCP, UDP.

Протягом останнього часу він стає популярнішим і близько 32% унікальних доменів працюють на сервері NGINX задля розміщення своїх сайтів.

### 3. Internet Information Services Web Server



Рисунок 1.4.Зображення логотипу Web-сервера Microsoft IIS

Internet Information Services Web Server (Web-сервер інформаційних служб Інтернету) є web-сервером від Microsoft, що створений Windows NT. Компанія Майкрософт підтримує даний продукт, і він також працює із кожною з платформ ОС Windows.

Microsoft IIS підтримує протоколи HTTP, HTTP/2, HTTPS, FTP та інші, але не має відкритого коду. Тому процес налаштування є складнішим і модифікації не є такими легкими.

### 4. Lighttpd Web-сервер



Рисунок 1.5. Логотип ПЗ Web-сервера Lighttpd

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 17   |

Lighttpd Web-сервер – це ще один web-сервер, що має відкритий код. Також він є досить гнучким та безпечним, бо він був спеціально розроблений та оптимізований для критичних та високопродуктивний до швидкості середовищ.

Перевагою даного сервера є ефективність, особливо у випадку проблем із завантаженням серверів.

## 5. Sun Java System



Рисунок 1.6. Логотип компанії Sun microsystems

Sun Java System – це web-сервер від компанії Sun, що працює під керуванням Windows, Solaris та HP-UX. Також його підтримують технології JavaServer Pages (JSP), JavaScript на боці сервера і Java Servlets.

Даний web-сервер відзначається надійністю, простотою в використанні та безпекою. Також він був спеціально розробленим для середніх та великих веб-сайтів.

## 6. Jigsaw



Рисунок 1.7. Логотип Web-сервера Jigsaw

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 18   |

Web-сервер Jigsaw є об'єктно-орієнтованим та повнофункціональним web-сервером, що пропонує функції із вдосконаленою архітектурою, що написана на Java. Для цього web-сервера не призначили повноцінний випуск, бо насамперед він був створений як експеримент з новою технологією. Проте більш пізні версії сервера зосередились на розробці кращих функцій.

## 7.LiteSpeed



Рисунок 1.8. Логотип Web-сервера LiteSpeed

Web-сервер LiteSpeed був найпопулярнішим із web-серверів протягом декількох років. Його розробила приватна компанія LiteSpeed Technologies, і він також представлявся як сервер з відкритим кодом.

Web-сервери Litespeed сумісні з функціями серверів Apache, бо в них досить подібні конфігурації. Вони також можуть завантажувати конфігураційні файли Apache і замінити усі функції Apache.[3]

Після короткого огляду популярних web-серверів представлено діаграму, що висвітлює відсоток використання web-серверів розробниками протягом останніх 21 року.

NGINX став теперішнім лідером за багатьма показниками та сильний ріст попиту отримав завдяки збільшенню кількості доменів. Проте Apache web-сервер має найбільшу кількість активних сайтів.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 19   |

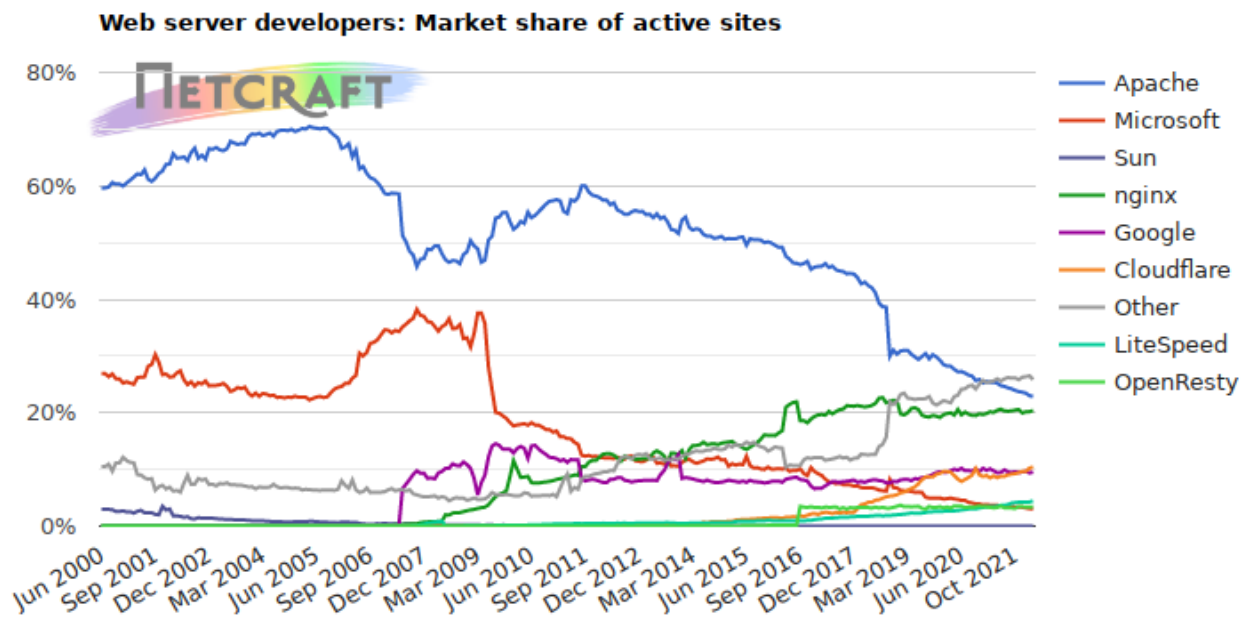


Рисунок 1.9. Рейтинг популярності Web-серверів

Рисунок зображений вище представляє відсотковий рейтинг використання веб-розробниками популярного ПЗ до липня 2021 року включно.

#### 1.4 Огляд функціоналу Web-серверів Apache та NGINX

Нині серед усього різноманіття web-серверів представлених на відкритому ринку можна виділити двох основних конкурентів в web-серверному бізнесі, а саме: Apache та NGINX, на них працюють більше 50% web-сайтів світу.

NGINX представлено задля подолання обмежень продуктивності web-сайтів Apache через збільшення обсягу трафіку даних і ріст кількості користувачів Інтернету.

Архітектура сервера Apache складається з модулів та ядра. Ядро в свою чергу виконує базові серверні можливості, керує паралельністю та приймає підключення. Також модулі виконують різні функції за кожним запитом. Сам web-сервер можливо налаштувати для ввімкнення різних модулів, а саме: керування динамічним вмістом, обробка запитів HTTP чи функції безпеки.

Web-сервер Apache 2 використовує різні модулі для обробки. Основні MPM Apache:

##### 1. Процес MPM;

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          | 20   |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          |      |

2. Робітник MPM;

3. Подія MPM.

Недоліком базової архітектури Apache може стати високе поглинання ресурсів, що спричиняє проблеми з web-сервером (наприклад, зменшення швидкості сервера).

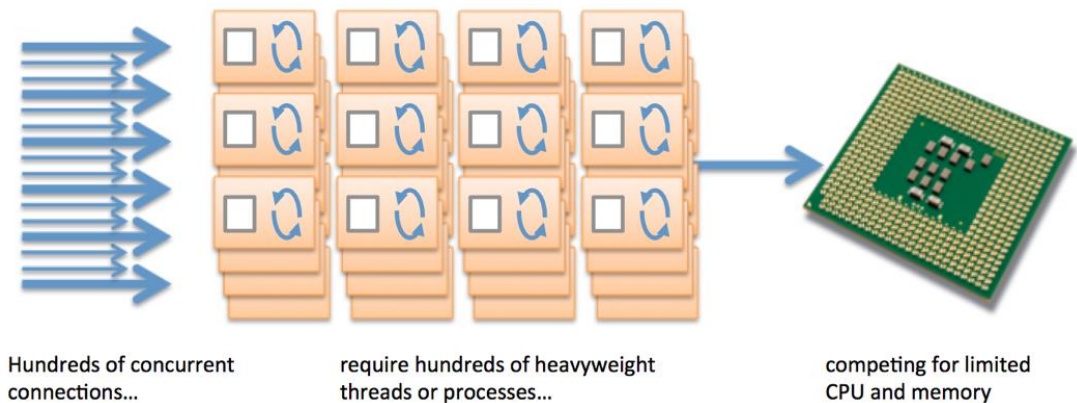


Рисунок 1.10. Зображення алгоритму сповільнення сервера при великій кількості з'єднань

Статичний вміст або файли зберігаються на диску комп'ютера-сервера, а саме CSSфайли, JavaScript файли чи зображення. Apache працює зі статистичним вмістом використовуючи файловий метод.

Обробка динамічного відбувається на самому web-сервері, без будь-яких зовнішніх компонентів. Apache використовує розташування URI (уніфікований ідентифікатор ресурсу), проте він призначений для більш абстрактних ресурсів. Також створюючи або налаштовуючи віртуальний хост Apache користується блоками каталогу в корені документа.

Модульна система Apache дає можливість динамічного завантаження чи вивантаження модулів. Їх також можливо ввімкнути чи вимкнути, аби додавати або видаляти функціональні можливості ті підключати їх до основного сервера.

Ключем успіху Apache стала модель «один сервер, що робить усе». Але через зростання рівня трафіку та збільшення кількості web-сторінок, продуктивність сервера обмежилась, і налаштувати його для роботи з справжнім трафіком стало більш складно.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 21   |

Щодо web-сервера NGINX варто відзначити, що кожен з робочих процесів може обробляти сотні та тисячі HTTP-з'єднань в один час. Архітектура даного сервера обробляє великі навантаження даних більш передбачуваною, якщо розглядати її з точки зору використання оперативної пам'яті, використання центрального процесора та затримки.

NGINX було розроблено, щоб використовувати його для неблокуючого алгоритму обробки з'єднань, який керується подіями. Його процес обробляє тисячі запитів в одному потоці обробки. Ці модулі обробки дають можливість NGINX працювати швидко та з меншою кількістю ресурсів.

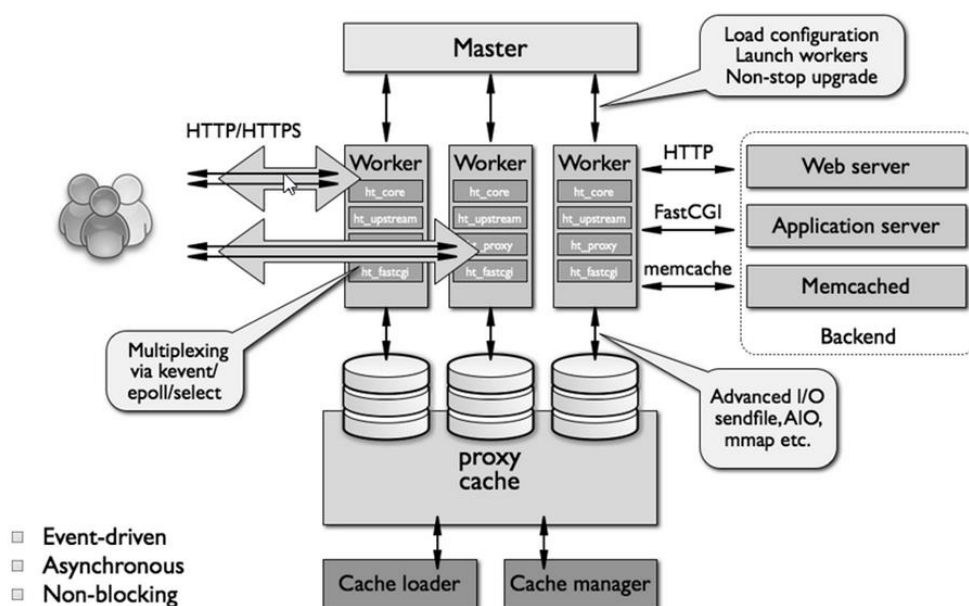


Рисунок 1.11. Архітектура NGINX

Також варто відзначити, що NGINX містить багато функцій та має можливість виконувати різні ролі сервера, а саме:

- Балансування навантаження та HTTP-кеш;
- Інтерфейсний проксі-сервер для Apache та інших web-серверів;
- Зворотній проксі-сервер для протоколів HTTP, HTTPS, SMTP, IMAP, POP3.

Варте уваги і те, що NGINX не надає додаткової конфігурації, а це підвищує продуктивність. Адже заборона налаштування каталогу дає можливість серверу

обслуговувати запити швидше, ніж Apache. Йому не доводиться шукати потрібні файли та інтерпретувати запити, що зробив користувач.

NGINX не дає механізм, що визначає конфігурацію, а для каталогу файлової системи передає URI. Це в свою чергу дає можливість для нього легко працювати і на web-сервері, і на проксі-сервері. NGINX налаштовується та описує, як необхідно реагувати на відмінності у запитах. Також він не перевіряє файлову систему, допоки не буде готовий обслуговувати запит, саме тому він не використовує жодної форми файлів .htaccess.

Для обрання потрібного ПЗ для web-сервера порівняємо їх за різними критеріями:

Таблиця 1.1. Порівняння web-серверів Apache та NGINX

| Критерій для порівняння | Apache                                                                                                                                                                                                                                   | NGINX                                                                                                                                                                                                                                                                                                         |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Базова архітектура      | <ul style="list-style-type: none"> <li>- орієнтований на процес підхід</li> <li>- створення нової теми для кожного з запитів</li> <li>- багатопотоковий підхід</li> <li>- вибір різних підключень та алгоритмів обробки даних</li> </ul> | <ul style="list-style-type: none"> <li>- орієнтований на події підхід</li> <li>- обробка декількох запитів в одному потоці</li> <li>- асинхронна обробка запитів</li> </ul>                                                                                                                                   |
| Продуктивність          | <ul style="list-style-type: none"> <li>- обслуговування статичного вмісту за допомогою методу на основі файлів</li> <li>- обробка динамічного вмісту на сервері</li> </ul>                                                               | <ul style="list-style-type: none"> <li>- можливість паралельно обслуговувати статичний вміст</li> <li>- в 2,5 рази швидша обробка статичного вмісту, ніж Apache</li> <li>- висока ефективність та менші вимоги до системних ресурсів</li> <li>- не обробляє динамічний вміст на сервері, а передає</li> </ul> |

|                                           |                                                                                               |                                                                                  |
|-------------------------------------------|-----------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|
|                                           |                                                                                               | зовнішньому процесору                                                            |
| Підтримка ОС                              | -підтримка всіх Unix-подібних систем (Linux, BSD)<br>-повна підтримка Microsoft Windows       | -підтримка майже всіх Unix-подібних ОС<br>-часткова підтримка Windows            |
| Централізована (розподілена) конфігурація | -дозволяє додаткову когфігурацію на основі кожного з каталогів використовуючи файли .htaccess | -не дозволяє додаткове налаштування<br>-не надає додаткової конфігурації         |
| Інтерпретація запитів                     | -передає розташування файлової системи                                                        | -передає URI для інтерпретування запитів<br>-швидка інтерпретація та відповідь   |
| Функціональні модулі                      | -налічує 60 офіційних динамічно-завантажуваних модулів<br>-модулі можна вмикати чи вимикати   | -основні модулі від сторонніх розробників<br>-модулі не завантажуються динамічно |
| Гнучкість                                 | -підтримує налаштування web-сервера використовуючи динамічні модулі                           | -недостатня гнучкість для підтримки динамічних модулів та завантаження           |
| Безпека                                   | -вище середнього рівень безпеки<br>-підказки конфігурації для обробки DDoS атак               | -високий рівень безпеки<br>-менша кодова база<br>-список порад щодо безпеки      |
| Підтримка                                 | -підтримка користувачів через списки розсилки, Stsck                                          | -підтримка користувачів через списки розсилки, IRC, Stack                        |

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 24   |

|  |               |                   |
|--|---------------|-------------------|
|  | Overflow, IRC | Overflow та форум |
|--|---------------|-------------------|

Зробивши огляд на переваги, обмеження та відмінності обох серверів, було обрано для подальшої роботи Web-сервер NGINX, через його перевагу над Apache за більшістю критеріїв.

Сьогодні більше, ніж половина популярних web-сайтів в мережі Інтернет, наприклад Instagram, Netflix, SoundCloud і навіть Pinterest покладаються на роботу web-сервера NGINX. У той час як кіберзлочинці досить часто обирають web-сервери ціллю для атаки. Вони намагаються віднайти найменшу вразливість для того, аби викрасти конфіденційну інформацію та дані користувачів для подальшого зловживання ними. NGINX також часто ставав свідком подібних атак та виявляв певні уразливості.

Розглянемо декілька найбільш критичних уразливостей сервера NGINX, які були знайдено раніше, і розглянемо те, як можна захистити наш web-сервер:

#### 1. Переповнення буфера купи NGINX SPDY у 2014 році.

У версіях NGINX від 1.3.15 до 1.4.7 та з 1.5.x до 1.5.12 де було реалізовано SPDY була наявна вразливість на основі купи, що дозволяло злочинцю виконувати код за допомогою створеного запиту. За замовчуванням модуль ngx\_http\_spdy\_module не скомпільовано і також в конфігурації параметрів не було –with-debug, за умови використання параметру spdy у директиві listen. Злочинець міг скористатись цим створивши запит для виклику переповнення буфера пам'яті купи після чого виконувався довільний код.

Вирішенням проблеми стало оновлення до останньої версії та використання необхідних патчів, що надає постачальник.

#### 2. Вразливість до підвищення привілеїв у 2016.

Дуже небезпечною стала вразливість сервера до підвищення привілеїв, що призводило до створення нових каталогів журналу з незахищеними дозволами. Вони, відповідно, були використані злочинцями для зміни привілеїв на власний розсуд від web-користувача до root, що скомпроментувало веб-програми, що

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 25   |

розміщені на web-сервері. Дана вразливість впливала на пакети серверів у дистрибутивах Debian та Ubuntu.

Вразливість виникала під час інсталяції NGINX із сховищ за замовчуванням у системах Linux на базі Debian. Оскільки каталог /var/log/NGINX належить www-data, кіберзлочинці замінювали журнал символічним посиланням на інший файл, отримуючи доступ до системи. А потім після перезапуску журнали були замисані до файлу, до якого вело це символічне посилання. Саме так зловмисники і підвищували привілеї до найвищого.

Вирішення даної проблеми було виправлено раніше публікації експлойту в одному з пакетів NGINX на Debian і Ubuntu.

### 3. Віддалена вразливість переповнення цілих чисел у 2017 році.

Дана вразливість виникала через неспроможність NGINX до виконання адекватних граничних перевірок даних, наданих користувачем. За допомогою цього злочинець міг отримати доступ до особистої та конфіденційної інформації чи призвести до збою роботи програми, що в свою чергу призводить до відмови в обслуговуванні.

Такі версії NGINX як 0.5.6-1.13.2 є вразливими до цілочисельного переповнення. Для вирішення проблеми необхідно оновити NGINX до останньої версії. А також отримати додаткову інформацію та рекомендації від постачальника.

### 4. Вразливість контролера у 2020

Кіберзлочинці у версіях контролера сервера до версії 3.2.0, що мали мережевий доступ до API контролера створювали непривілейовані облікові записи користувачів. В свою чергу користувачі створені подібним методом могли завантажити нову ліцензію до системи, проте не могли переглядати чи змінювати її компоненти. Отже, злочинці використовували цю вразливість для створення облікових записів, щоб створити сценарій відмови в обслуговуванні (DoS).

Щоб зменшити ризик виникнення даної вразливості необхідно провести оновлення до останньої версії чи обмежити доступ до мережі для порту API

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 26   |

контролерів NGINX, виконавши налаштування ПЗ брандмауера хоста контролера web-сервера або налаштувати зовнішній пристрій фільтрації пакетів.

#### 5. Вразливість PHP віддаленого виконання коду

Деякі версії PHP 7, що працюють з увімкненим `php-fpm`, вразливі до віддаленого виконання коду. Бездіяльність може призвести до витоку конфіденційної інформації, зміни даних, додаванням нових даних чи навіть атак DoS.

Для пом'якшення даної вразливості необхідно оновити PHP до останньої версії.

#### Висновки до розділу 1

У цьому розділі було проведено аналіз основних вразливостей web-серверів та типів загроз для них. Розглянуто теоретичні відомості щодо захисту та протидії кібератакам.

Також був проведений огляд популярного web-серверів на відкритому ринку та розглянуто саме поняття «веб-сервер».

Оглянуто та проаналізовано два найпопулярніші web-сервери – Apache і NGINX. Також було проведено їх порівняння та обрано Web-сервер для подальшого використання у проекті. Наприкінці було розглянуто критичні уразливості сервера NGINX за останній час.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 27   |

## РОЗДІЛ 2 ПРОЕКТ ПІДСИСТЕМИ ЗАХИСТУ ДЛЯ WEB-СЕРВЕРА НА БАЗІ ОС LINUX

### 2.1 Аналіз IT-інфраструктури

Під час виконання даної кваліфікаційної роботи було розглянуто підсистему невеликої компанії (Додаток А). До її складу входять: два маршрутизатори, 4 комутатори, один сервер, сім робочих станцій. Задля підключення було використано інтерфейси GigabitEthernet і FastEthernet, Ethernet. Важливим моментом є те, що даний сценарій є лише емуляцією за допомогою ПЗ VmwareWorkstation та GNS3, оскільки в розпорядженні не було потрібних обчислювальних потужностей. У таблиці 2.1 наведено топологію підключень однієї підсистеми, котра розглядається.

Таблиця 2.1 – Параметри підключення пристроїв

| Пристрій                    | Інтерфейс | Підключення до пристрою     | Підключення до інтерфейса |
|-----------------------------|-----------|-----------------------------|---------------------------|
| Маршрутизатор Main-Router   | Fa0/0     | Глобальна мережа Internet   | -                         |
|                             | Gig1/0    | Маршрутизатор Server-Router | Fa0/0                     |
|                             | Gig2/0    | Комутатор Admin-SW          | Eth0/0                    |
|                             | Gig3/0    | Комутатор Dep-SW            | Eth0/0                    |
|                             | Gig4/0    | Комутатор Support-SW        | Eth0/0                    |
| Маршрутизатор Server-Router | Fa0/0     | Маршрутизатор Main-Router   | Gig1/0                    |
|                             | Gig0/1    | Комутатор Server-SW         | Eth0/0                    |
| Комутатор Server-SW         | Eth0/0    | Маршрутизатор Server-Router | Gig1/0                    |
|                             | Eth0/1    | Сервер UbuntuServer         | Eth0                      |
| Комутатор Admin-SW          | Eth0/0    | Маршрутизатор Main-Router   | Gig2/0                    |
|                             | Eth0/1    | Робоча станція Admin        | Eth0                      |
| Комутатор Dep-SW            | Eth0/0    | Маршрутизатор Main-Router   | Gig3/0                    |
|                             | Eth0/1    | Робоча станція Dep-WS-1     | Eth0                      |
|                             | Eth0/2    | Робоча станція Dep-WS-2     | Eth0                      |
|                             | Eth0/3    | Робоча станція Dep-WS-3     | Eth0                      |
|                             | Eth1/0    | Робоча станція Dep-WS-4     | Eth0                      |

Продовження таблиці 2.1

| Пристрій                     | Інтерфейс | Підключення до пристрою      | Підключення до інтерфейса |
|------------------------------|-----------|------------------------------|---------------------------|
| Комутатор Support-SW         | Eth0/0    | Маршрутизатор Main-Router    | Gig4/0                    |
|                              | Eth0/1    | Робоча станція Customer-WS   | Eth0                      |
|                              | Eth0/2    | Робоча станція IT-Manager-WS | Eth0                      |
| Сервер UbuntuServer          | Eth0      | Комутатор Server-SW          | Eth0/1                    |
| Робоча станція Admin         | Eth0      | Комутатор Admin-SW           | Eth0/1                    |
| Робоча станція Dep-WS-1      | Eth0      | Комутатор Dep-SW             | Eth0/1                    |
| Робоча станція Dep-WS-2      | Eth0      | Комутатор Dep-SW             | Eth0/2                    |
| Робоча станція Dep-WS-3      | Eth0      | Комутатор Dep-SW             | Eth0/3                    |
| Робоча станція Dep-WS-4      | Eth0      | Комутатор Dep-SW             | Eth1/0                    |
| Робоча станція Customer-WS   | Eth0      | Комутатор Support-SW         | Eth0/1                    |
| Робоча станція IT-Manager-WS | Eth0      | Комутатор Support-SW         | Eth0/2                    |

Таблиця 2.2 – Параметри адресації мережі

| Мережа/<br>Пристрій            | Інтерфейс/Мережний<br>адаптер/Шлюз | IP-адреса     | Маска         | Префікс |
|--------------------------------|------------------------------------|---------------|---------------|---------|
| Мережа А                       | -                                  | 192.168.1.0   | 255.255.255.0 | /24     |
| Мережа В                       | -                                  | 192.168.2.0   | 255.255.255.0 | /24     |
| Мережа С                       | -                                  | 192.168.3.0   | 255.255.255.0 | /24     |
| Мережа D                       | -                                  | 192.168.4.0   | 255.255.255.0 | /24     |
| Мережа Е                       | -                                  | 192.168.5.0   | 255.255.255.0 | /24     |
| Маршрутизатор<br>Main-Router   | Інтерфейс Fa0/0                    | DHCP          |               |         |
|                                | Інтерфейс Gig1/0                   | 192.168.1.1   | 255.255.255.0 | /24     |
|                                | Інтерфейс Gig2/0                   | 192.168.3.1   | 255.255.255.0 | /24     |
|                                | Інтерфейс Gig3/0                   | 192.168.4.1   | 255.255.255.0 | /24     |
|                                | Інтерфейс Gig4/0                   | 192.168.5.1   | 255.255.255.0 | /24     |
| Маршрутизатор<br>Server-Router | Інтерфейс Fa0/0                    | 192.168.1.2   | 255.255.255.0 | /24     |
|                                | Інтерфейс Gig0/1                   | 192.168.2.1   | 255.255.255.0 | /24     |
| Комутатор<br>Server-SW         | Інтерфейс Vlan 1                   | 192.168.2.100 | 255.255.255.0 | /24     |
|                                | Шлюз за замовчуванням              | 192.168.2.1   | -             | -       |

## Продовження таблиці 2.1

|                                 |                       |               |               |     |
|---------------------------------|-----------------------|---------------|---------------|-----|
| Комутатор<br>Admin-SW           | Інтерфейс Vlan 1      | 192.168.4.100 | 255.255.255.0 | /24 |
|                                 | Шлюз за замовчуванням | 192.168.3.1   | -             | -   |
| Комутатор<br>Dep-SW             | Інтерфейс Vlan 1      | 192.168.4.101 | 255.255.255.0 | /24 |
|                                 | Шлюз за замовчуванням | 192.168.4.1   | -             | -   |
| Комутатор<br>Support-SW         | Інтерфейс Vlan 1      | 192.168.4.102 | 255.255.255.0 | /24 |
|                                 | Шлюз за замовчуванням | 192.168.5.1   | -             | -   |
| Сервер<br>UbuntuServer          | Мережний адаптер      | 192.168.4.103 | 255.255.255.0 | /24 |
|                                 | Шлюз за замовчуванням | 192.168.2.1   | -             | -   |
| Робоча станція<br>Admin         | Мережний адаптер      | DHCP          |               |     |
|                                 | Шлюз за замовчуванням |               |               |     |
| Робоча станція<br>Dep-WS-1      | Мережний адаптер      |               |               |     |
|                                 | Шлюз за замовчуванням |               |               |     |
| Робоча станція<br>Dep-WS-2      | Мережний адаптер      |               |               |     |
|                                 | Шлюз за замовчуванням |               |               |     |
| Робоча станція<br>Dep-WS-3      | Мережний адаптер      |               |               |     |
|                                 | Шлюз за замовчуванням |               |               |     |
| Робоча станція<br>Dep-WS-4      | Мережний адаптер      |               |               |     |
|                                 | Шлюз за замовчуванням |               |               |     |
| Робоча станція<br>Customer-WS   | Мережний адаптер      |               |               |     |
|                                 | Шлюз за замовчуванням |               |               |     |
| Робоча станція<br>IT-Manager-WS | Мережний адаптер      |               |               |     |
|                                 | Шлюз за замовчуванням |               |               |     |
| Сервер<br>UbuntuServer          | Мережний адаптер      |               |               |     |
|                                 | Шлюз за замовчуванням |               |               |     |
| Робоча станція<br>Admin         | Мережний адаптер      |               |               |     |
|                                 | Шлюз за замовчуванням |               |               |     |

Налаштування параметрів IP-адресації, протоколу маршрутизації OSPFv2, протоколу мережевого часу NTP та протоколу віддаленого керування SSH відбувалось наступним чином:

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          | 30   |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          |      |

## Налаштування маршрутизатора Main-Router:

```

Main-Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Main-Router(config)#int fa0/0
Main-Router(config-if)#ip address dhcp
Main-Router(config-if)#no sh
Main-Router(config-if)#
*Aug 29 03:47:02.419: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Aug 29 03:47:03.419: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
Main-Router(config-if)#int gig1/
*Aug 29 03:47:13.195: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0 assigned DHCP address 192.168.122.86, mask 255.255.255.0, hostna
tname Main-Router

Main-Router(config-if)#int gig1/0
Main-Router(config-if)#ip address 192.168.1.1 255.255.255.0
Main-Router(config-if)#exit
Main-Router(config)#router ospf 1
Main-Router(config-router)#router-id 1.1.1.1
Main-Router(config-router)#passive-interface Gigabit 1/0
Main-Router(config-router)#network 192.168.1.0 0.0.0.255 area 0
Main-Router(config-router)#exit
Main-Router(config)#ip domain-name chornobil.com
Main-Router(config)#ip dns server
Main-Router(config)#ip ssh version 2
Please create RSA keys (of atleast 768 bits size) to enable SSH v2.
Main-Router(config)#crypto key generate rsa general-key modulus 1024
The name for the keys will be: Main-Router.chornobil.com

% The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

Main-Router(config)#
*Aug 29 03:55:31.903: %SSH-5-ENABLED: SSH 2.0 has been enabled
Main-Router(config)#line vty 0 4
Main-Router(config-line)#logging synchronous
Main-Router(config-line)#login local
Main-Router(config-line)#exec-timeout 20
Main-Router(config-line)#transport input ssh
Main-Router(config-line)#exit
Main-Router(config)#ntp server 0.ua.pool.ntp.org
Main-Router(config)#clock timezone EET +2
Main-Router(config)#
*Aug 29 03:58:53.159: %SYS-6-CLOCKUPDATE: System clock has been updated from 03:58:53 UTC Mon Aug 29 2022 to 05:58:53 EET Mon Aug 29
2022, configured from console by console.

Main-Router(config)#int gig 2/0
Main-Router(config-if)#ip address 192.168.3.1 255.255.255.0
Main-Router(config-if)#no sh
Main-Router(config-if)#ex
*Aug 29 04:01:38.211: %LINK-3-UPDOWN: Interface GigabitEthernet2/0, changed state to up
*Aug 29 04:01:39.211: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet2/0, changed state to up
Main-Router(config-if)#exit
Main-Router(config)#int gig3/0
Main-Router(config-if)#ip address 192.168.4.1 255.255.255.0
Main-Router(config-if)#no sh
Main-Router(config-if)#
*Aug 29 04:02:00.159: %LINK-3-UPDOWN: Interface GigabitEthernet3/0, changed state to up
*Aug 29 04:02:01.159: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet3/0, changed state to up
Main-Router(config-if)#exit
Main-Router(config)#int gig4/0
Main-Router(config-if)#ip address 192.168.5.1 255.255.255.0
Main-Router(config-if)#no sh
Main-Router(config-if)#
*Aug 29 04:02:33.947: %LINK-3-UPDOWN: Interface GigabitEthernet4/0, changed state to up
*Aug 29 04:02:34.947: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet4/0, changed state to up
Main-Router(config-if)#do write

```

Рисунок 2.1. Налаштування конфігурації Main-Router

Налаштування маршрутизатора Main-Router було проведене наступним чином:

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 31   |

```

Server-Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Server-Router(config)#int f0/0
Server-Router(config-if)#ip address 192.168.1.2 255.255.255.0
Server-Router(config-if)#no sh
Server-Router(config-if)#exit
Server-Router(config)#
*Aug 29 04:06:52.535: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Aug 29 04:06:53.535: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
Server-Router(config)#int gig1/0
Server-Router(config-if)#ip address 192.168.2.1 255.255.255.0
Server-Router(config-if)#exit
Server-Router(config)#router ospf 1
Server-Router(config-router)#passive-interface gig1/0
Server-Router(config-router)#network 192.168.2.0 0.0.0.255 area 1
Server-Router(config-router)#exit
Server-Router(config)#ip domain-name chornobil.com
Server-Router(config)#ip dns server
Server-Router(config)#ip host chornobil.com 127.0.1.1
Server-Router(config)#access-list permit 192.168.1.0 0.0.0.255
^
% Invalid input detected at '^' marker.

Server-Router(config)#access-list 22 permit 192.168.1.0 0.0.0.255
Server-Router(config)#access-list 22 permit 192.168.2.0 0.0.0.255
Server-Router(config)#ip ssh version 2
Please create RSA keys (of atleast 768 bits size) to enable SSH v2.
Server-Router(config)#crypto key generate rsa general-key modulus 1024
The name for the keys will be: Server-Router.chornobil.com

% The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

Server-Router(config)#
*Aug 29 04:12:47.975: %SSH-5-ENABLED: SSH 2.0 has been enabled
Server-Router(config)#line vty 0 4
Server-Router(config-line)#access-class 22 in
Server-Router(config-line)#logging synchronous
Server-Router(config-line)#login local
Server-Router(config-line)#exec-timeout 20
Server-Router(config-line)#transport input ssh
Server-Router(config-line)#exit
Server-Router(config)#clock timezone EET+2
% Incomplete command.

Server-Router(config)#clock timezone EET +2
Server-Router(config)#
*Aug 29 04:15:57.563: %SYS-6-CLOCKUPDATE: System clock has been updated from 04:15:57 UTC Mon Aug 29 2022 to 06:15:57 EET Mon Aug 29 2022, configured from console by console.

```

Рисунок 2.2. Налаштування конфігурації Server-Router

```

Main-Router#sh ip int br
Interface IP-Address OK? Method Status Protocol
FastEthernet0/0 192.168.122.86 YES DHCP up up
GigabitEthernet1/0 192.168.1.1 YES manual up up
GigabitEthernet2/0 192.168.3.1 YES manual up up
GigabitEthernet3/0 192.168.4.1 YES manual up up
GigabitEthernet4/0 192.168.5.1 YES manual up up

```

Рисунок 2.3. Результат виконання команди show ip interface brief на маршрутизаторі Main-Router

```

Main-Router#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.168.122.1 to network 0.0.0.0

C    192.168.122.0/24 is directly connected, FastEthernet0/0
C    192.168.4.0/24 is directly connected, GigabitEthernet3/0
C    192.168.5.0/24 is directly connected, GigabitEthernet4/0
C    192.168.3.0/24 is directly connected, GigabitEthernet2/0
S*   0.0.0.0/0 [254/0] via 192.168.122.1

```

Рисунок 2.3. Результат виконання команди show ip route на маршрутизаторі Main-Router

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | KIK.KP.B – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          | 32   |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          |      |

```

Main-Router#sh ip ospf data
Main-Router#sh ip ospf database

      OSPF Router with ID (1.1.1.1) (Process ID 1)

      Router Link States (Area 0)

Link ID      ADV Router    Age      Seq#          Checksum Link count
1.1.1.1      1.1.1.1        1        0x80000002  0x004FEA  0
Main-Router#show ip ospf
Routing Process "ospf 1" with ID 1.1.1.1
Start time: 00:12:45.120, Time elapsed: 00:33:41.992
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Supports area transit capability
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 5000 msec
Minimum hold time between two consecutive SPF's 10000 msec
Maximum wait time between two consecutive SPF's 10000 msec
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msec
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msec
Retransmission pacing timer 66 msec
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0

```

Рисунок 2.4. Результат виконання команд `show ip ospf database` та `show ip ospf neighbor` на маршрутизаторі Main-Router

```

Server-Router#sh ip int br
Interface      IP-Address      OK? Method Status      Protocol
FastEthernet0/0 192.168.1.2     YES manual up          up
GigabitEthernet1/0 192.168.2.1    YES manual up          up
GigabitEthernet2/0 unassigned      YES unset administratively down down
GigabitEthernet3/0 unassigned      YES unset administratively down down

```

Рисунок 2.5. Результат виконання команди `show ip interface brief` на маршрутизаторі Server-Router

```

Server-Router#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, GigabitEthernet1/0

```

Рисунок 2.6. Результат виконання команди `show ip route` на маршрутизаторі Server-Router

```

Server-Router#sh ip ssh
SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits

```

Рисунок 2.7. Результат виконання команди `show ip ssh` на маршрутизаторі Server-Router

Також було проведено налаштування параметрів IP-адресації, протоколу віддаленого керування SSH:

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | KIK.KP.B – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 33   |

```

Server-SW(config)#int vlan 1
Server-SW(config-if)#ip address 192.168.2.100 255.255.255.0
Server-SW(config-if)#ntp broadcast client
Server-SW(config-if)#exit
Server-SW(config)#ip default-gateway 192.1
*Aug 29 04:43:39.636: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
il.com GigabitEthernet1/0 (full duplex).
Server-SW(config)#ip default-gateway 192.168.2.1 255.255.255.0
^
% Invalid input detected at '^' marker.

Server-SW(config)#ip default-gateway 192.168.2.1
Server-SW(config)#ip domain-name chornobil.com
Server-SW(config)#
*Aug 29 04:44:39.630: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
il.com GigabitEthernet1/0 (full duplex).
Server-SW(config)#ip name-server 127.0.1.1
Server-SW(config)#ip ssh version 2
Please create RSA keys to enable SSH (and of atleast 768 bits for SSH v2).
Server-SW(config)#crypto key generate rsa general-key m
*Aug 29 04:45:39.636: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
il.com GigabitEthernet1/0 (full duplex).
Server-SW(config)#crypto key generate rsa general-key modulus 1024
The name for the keys will be: Server-SW.chornobil.com

% The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 0 seconds)

Server-SW(config)#
*Aug 29 04:45:53.988: %SSH-5-ENABLED: SSH 2.0 has been enabled
Server-SW(config)#line vty 0 4
Server-SW(config-line)#logging synch
Server-SW(config-line)#login local
Server-SW(config-line)#exec-timeout 20
Server-SW(config-line)#
*Aug 29 04:46:39.637: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
il.com GigabitEthernet1/0 (full duplex).
Server-SW(config-line)#transport input ssh
Server-SW(config-line)#exit
Server-SW(config)#clock timezone EET +2
Server-SW(config)#do write

```

Рисунок 2.8. Налаштування комутатора Server-SW

```

Admin-SW##conf t
Enter configuration commands, one per line. End with CNTL/Z.
Admin-SW(config)#int vlan 1
Admin-SW(config-if)#ip address
*Aug 29 04:50:56.239: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
.com GigabitEthernet2/0 (full duplex).
Admin-SW(config-if)#ip address 192.168.4.100 255.255.255.0
Admin-SW(config-if)#nt
*Aug 29 04:51:56.235: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
.com GigabitEthernet2/0 (full duplex).
Admin-SW(config-if)#ntp broadcast client
Admin-SW(config-if)#exit
Admin-SW(config)#ip default-gateway 192.168.3.1
Admin-SW(config)#ip domain-name
*Aug 29 04:52:56.235: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
.com GigabitEthernet2/0 (full duplex).
Admin-SW(config)#ip domain-name chornobil.com
Admin-SW(config)#ip name-server 127.0.1.1
Admin-SW(config)#ip ssh version 2
Please create RSA keys to enable SSH (and of atleast 768 bits for SSH v2).
Admin-SW(config)#crypto key generate rsa general-key m
*Aug 29 04:53:56.211: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
.com GigabitEthernet2/0 (full duplex).
Admin-SW(config)#crypto key generate rsa general-key modulus 1024
The name for the keys will be: Admin-SW.chornobil.com

% The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 0 seconds)

Admin-SW(config)#
*Aug 29 04:54:02.243: %SSH-5-ENABLED: SSH 2.0 has been enabled
Admin-SW(config)#line vty 0 4
Admin-SW(config-line)#logging synchronous
Admin-SW(config-line)#login local
Admin-SW(config-line)#exec-timeout 20
Admin-SW(config-line)#transport input ssh
Admin-SW(config-line)#exit
Admin-SW(config)#
*Aug 29 04:54:56.245: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
.com GigabitEthernet2/0 (full duplex).
Admin-SW(config)#clock timezone EET +2
Admin-SW(config)#do write

```

Рисунок 2.9. Налаштування комутатора Admin-SW

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 34   |

```

Dep-SW#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Dep-SW(config)#int vlan 1
Dep-SW(config-if)#ip address 192.168.
*Aug 29 04:58:17.227: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on Ether
.com GigabitEthernet3/0 (full duplex).
Dep-SW(config-if)#ip address 192.168.4.101 255.255.255.0
Dep-SW(config-if)#ntp broadcast client
Dep-SW(config-if)#exit
Dep-SW(config)#ip default-
*Aug 29 04:59:17.241: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on Ether
.com GigabitEthernet3/0 (full duplex).
Dep-SW(config)#ip default-gateway 192.168.4.1
Dep-SW(config)#ip domain-name chornobil.com
Dep-SW(config)#ip name-server 127.0.1.1
Dep-SW(config)#
*Aug 29 05:00:17.237: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on Ether
.com GigabitEthernet3/0 (full duplex).
Dep-SW(config)#ip ssh version 2
Please create RSA keys to enable SSH (and of atleast 768 bits for SSH v2).
Dep-SW(config)#crypto key generate rsa general-key modulus 1024
The name for the keys will be: Dep-SW.chornobil.com

% The key modulus size is 1024 bits
% Generating 1024 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 0 seconds)

Dep-SW(config)#
*Aug 29 05:00:43.699: %SSH-5-ENABLED: SSH 2.0 has been enabled
Dep-SW(config)#line vty 0 4
Dep-SW(config-line)#logging synchronous
Dep-SW(config-line)#login local
Dep-SW(config-line)#exec-time
*Aug 29 05:01:17.238: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on Ether
.com GigabitEthernet3/0 (full duplex).
Dep-SW(config-line)#exec-timeout 20
Dep-SW(config-line)#transport input ssh
^
% Invalid input detected at '^' marker.

Dep-SW(config-line)#transport input ssh
Dep-SW(config-line)#exit
Dep-SW(config)#ntp server 192.168.4.1
Dep-SW(config)#clock timezone EET +2
Dep-SW(config)#DO WRITE

```

Рисунок 2.10. Налаштування комутатора Dep-SW

```

Support-SW#CONF T
Enter configuration commands, one per line. End with CNTL/Z.
Support-SW(config)#INT VLAN 1
Support-SW(config-if)#ip address 192.168.4
*Aug 29 05:03:48.240: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on Ether
.com GigabitEthernet4/0 (full duplex).
Support-SW(config-if)#ip address 192.168.4.102 255.255.255.0
Support-SW(config-if)#ntp broadcast client
Support-SW(config-if)#exit
Support-SW(config)#ip default-gateway 192.168.4.1
Support-SW(config)#ip
*Aug 29 05:04:48.251: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on Ether
.com GigabitEthernet4/0 (full duplex).
Support-SW(config)#ip ssh version 2
Please create RSA keys to enable SSH (and of atleast 768 bits for SSH v2).
Support-SW(config)#crypto key generate rsa general-key modulus 1024
% Please define a domain-name first.
Support-SW(config)#
*Aug 29 05:05:48.231: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
.com GigabitEthernet4/0 (full duplex).
Support-SW(config)#line vty 0 4
Support-SW(config-line)#logging synchronous
Support-SW(config-line)#login local
Support-SW(config-line)#exec-timeou
*Aug 29 05:06:48.232: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on
.com GigabitEthernet4/0 (full duplex).
Support-SW(config-line)#exec-timeout 20
Support-SW(config-line)#transport input ssh
Support-SW(config-line)#exit
Support-SW(config)#ntp server 192.168.4.1
Support-SW(config)#clock timezone EET +2
Support-SW(config)#do write

```

Рисунок 2.11. Налаштування комутатора Support-SW

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 35   |

```

Server-SW#sh ip int br | i Vlan1
Vlan1          192.168.2.100  YES manual administratively down down
Server-SW#show ssh
%No SSHv2 server connections running.
%No SSHv1 server connections running.
Server-SW#sh ntp associations

address          ref clock      st  when  poll reach delay offset disp
* sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured

```

Рисунок 2.12. Результат виконання команд show interface brief, show ip ssh, show ntp associations на комутаторі Server-SW

```

Admin-SW#show ip int br | i Vlan1
Vlan1          192.168.4.100  YES manual administratively down down
Admin-SW#show ntp associations

address          ref clock      st  when  poll reach delay offset disp
* sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
Admin-SW#sh clock
*Aug 29 05:13:56.242: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on Ethe
0 (full duplex).
Admin-SW#sh clock detail
*07:14:00.338 EET Mon Aug 29 2022
Time source is NTP

```

Рисунок 2.13. Результат виконання команд show interface brief, show ip ssh, show ntp associations та show clock detail на комутаторі Admin-SW

```

Dep-SW#sh ip int br | i Vlan1
Vlan1          192.168.4.101  YES manual administratively down down
Dep-SW#show ntp associations

address          ref clock      st  when  poll reach delay offset disp
~192.168.4.1      .INIT.         16  -    512    0 0.000  0.000 15937.
* sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
Dep-SW#sh clock detail
*07:15:54.892 EET Mon Aug 29 2022
Time source is NTP

```

Рисунок 2.14. Результат виконання команд show interface brief, show ip ssh, show ntp associations та show clock detail на комутаторі Dep-SW

```

Support-SW#show ip int br | i Vlan1
Vlan1          192.168.4.102  YES manual administratively down down
Support-SW#sh ntp associations

address          ref clock      st  when  poll reach delay offset disp
~192.168.4.1      .INIT.         16  -    128    0 0.000  0.000 15937.
* sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
Support-SW#sh clock detail
*07:17:32.624 EET Mon Aug 29 2022
Time source is NTP

```

Рисунок 2.15. Результат виконання команд show interface brief, show ip ssh, show ntp associations та show clock detail на комутаторі Support-SW

## 2.2 Аналітика інфраструктури віддаленого доступу

В даному розділі проведено аналіз протоколів віддалено доступу, щоб виявити уразливі місця та можливі варіанти проникнення до підсистеми підприємства.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 36   |

Telnet - це прикладний протокол, який використовується в Інтернеті або локальній мережі для забезпечення двонаправленого інтерактивного текстово-орієнтованого зв'язку за допомогою віртуального термінального підключення. Дані користувача перемежуються в діапазоні з керуючою інформацією Telnet у 8-бітовому байт-орієнтованому з'єднанні даних через протокол керування передачею (TCP).

Telnet надає користувачам двонаправлену інтерактивну текстово-орієнтовану систему зв'язку, яка використовує віртуальне термінальне з'єднання понад 8 байт. Дані користувача перемежуються в смузі з керуючою інформацією Telnet через протокол керування передачею (TCP) . Часто Telnet використовувався на терміналі для віддаленого виконання функцій.

Користувач підключається до сервера за допомогою протоколу Telnet, що означає введення Telnet у командному рядку за таким синтаксисом: telnet hostname порт. Потім користувач виконує команди на сервері, використовуючи певні команди Telnet у підказці Telnet. Щоб завершити сеанс і вийти з системи, користувач завершує команду Telnet за допомогою Telnet.

Telnet можна використовувати для перевірки та усунення несправностей віддалених веб-серверів або поштових серверів, а також для віддаленого доступу до MUD (багатокористувацьких ігор у підземеллях) і довірених внутрішніх мереж.

Важливо не забувати, що Telnet не є безпечним протоколом і також він не зашифрований. За допомогою відстеження з'єднання користувача зломисник може отримати доступ до імені користувача, пароля та іншої приватної інформації, котра вводиться під час сеансу Telnet у відкритому тексті. Використовуючи цю інформацію можна отримати доступ до пристрою користувача.

Існує це один метод безпечного віддаленого входу, а саме протокол SSH (також відомий як Secure Shell) – це метод безпечного віддаленого входу з одного комп'ютера на інший. Він надає кілька альтернативних варіантів надійної автентифікації та захищає безпеку та цілісність зв'язку за допомогою надійного

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 37   |

шифрування. Це безпечна альтернатива незахищеним протоколам входу (таким як telnet , rlogin) і незахищеним методам передачі файлів (таким як FTP ).

Протокол використовується в корпоративних мережах для:

- забезпечення безпечного доступу для користувачів і автоматизованих процесів
- інтерактивна та автоматизована передача файлів
- видача дистанційних команд
- управління мережевою інфраструктурою та іншими критично важливими компонентами системи.

Протокол працює в моделі клієнт-сервер, що означає, що підключення встановлюється SSH-клієнтом, який підключається до SSH-сервера. Клієнт SSH керує процесом встановлення підключення та використовує криптографію з відкритим ключем для перевірки ідентичності сервера SSH. Після етапу налаштування протокол SSH використовує надійне симетричне шифрування та алгоритми хешування для забезпечення конфіденційності та цілісності даних, якими обмінюються клієнт і сервер.

Є декілька варіантів, які можна використовувати для автентифікації користувача. Найпоширенішими з них стали паролі та автентифікація за відкритим ключем.

Метод автентифікації з відкритим ключем в основному використовується для автоматизації, а іноді системними адміністраторами для єдиного входу. Виявилося, що він використовується набагато ширше, ніж ми коли-небудь очікували. Ідея полягає в тому, щоб мати пару криптографічних ключів – відкритий та закритий ключ – і налаштувати відкритий ключ на сервері для авторизації доступу та надання доступу до сервера всім, хто має копію закритого ключа. Ключі, які використовуються для автентифікації, називаються ключами SSH . Автентифікація з відкритим ключем також використовується зі смарт-картками, такими як картки CAC і PIV, які використовуються урядом США.

Основним використанням автентифікації на основі ключів є забезпечення безпечної автоматизації. Автоматичне захищене передавання файлів оболонки

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 38   |

використовується для легкої інтеграції програм, а також для автоматизованих систем і керування конфігурацією.

Після встановлення з'єднання між SSH-клієнтом та сервером дані, що передаються, шифруються відповідно до параметрів, узгоджених під час налаштування. Під час узгодження клієнт та сервер мають узгодити алгоритм симетричного шифрування, який буде використовуватися, та генерують ключ шифрування, який буде використовуватися. Трафік між сторонами, які спілкуються, захищено галузевими стандартними сильними алгоритмами шифрування (такими як AES (Advanced Encryption Standard)), а протокол SSH в той же час містить механізм, що забезпечує цілісність переданих даних за допомогою стандартних алгоритмів хешування (таких як SHA).

Після аналізу конфігурації на пристроях мережі було виявлено використання незахищеного протоколу Telnet. Тому його було відключено, а SSH протокол навпаки ввімкнено та налаштовано.

### **2.3. Проведення аналізу векторів кібератак та визначення методів для захисту мережі підприємства**

Вектор кібератаки – це метод або шлях зломисника може зламати або проникнути в усю мережу/систему. Вектори атак дозволяють хакерам використовувати вразливі місця системи, включно з людським елементом.

Будь-який інцидент безпеки, під час якого неавторизована сторона отримує доступ або викрадає конфіденційні, захищені чи конфіденційні дані, що ставить під загрозу бренд, клієнтів і активи організації. Такі випадки, як DDoS, майнінг біткойнів тощо, також є порушенням безпеки. Порушення даних є найпоширенішими, але не всі інциденти безпеки стосуються крадіжки даних.

Перелік поширених типів векторних кібератак та способів їх уникнення:

#### **1. Скомпрометовані облікові дані**

Ім'я користувача і пароль залишаються найпоширенішим типом облікових даних доступу. Зламані облікові дані описують випадок, якщо облікові дані

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 39   |

користувача, як-от імена користувачів і паролі, доступні неавторизованим особам. Зазвичай це трапляється, коли нічого не підозрюючи користувачі стають жертвами спроб фішингу та вводять свої облікові дані на підроблених веб-сайтах. У випадку втрати, викрадення або розкриття скомпрометовані облікові дані можуть надати зловмиснику доступ до внутрішньої інформації. Хоча моніторинг та аналіз на підприємстві можуть виявити підозрілу активність, ці облікові дані ефективно обходять захист периметра та ускладнюють виявлення. Ризик, котрий створює скомпрометована облікова інформація, залежить від рівня доступу, який він надає. Облікові дані привілейованого доступу, які надають адміністративний доступ до пристроїв і систем, зазвичай становлять більший ризик для підприємства, ніж облікові дані споживача. І не тільки люди мають повноваження. Сервери, мережеві пристрої та інструменти безпеки часто мають паролі, які забезпечують інтеграцію та зв'язок між пристроями.

Щоб цього уникнути потрібно: встановити ефективні політики щодо паролів, не використовувати один і той же пароль для доступу до кількох сервісів, використовувати двофакторну автентифікацію.

#### 1. Слабкі та вкрадені облікові дані

Слабкі паролі і повторне використання паролів роблять розкриття облікових даних шлюзом для початкового доступу зловмисника та його поширення. Останні атаки зловмисного ПЗ, наприклад Mirai, підкреслюють цю загрозу не лише для керованих пристроїв, але й для пристроїв, підключених до IoT.

Програми та протоколи, які надсилають облікові дані для входу через мережу, становлять значну загрозу безпеці. Зловмисник, підключений до мережі, може легко знайти та використати ці облікові дані для бокового переміщення. Наприклад, під час атаки Target зловмисники змогли викрасти облікові дані Active Directory і поширити свою атаку на корпоративну платіжну мережу.

Для уникання цього варто слідкувати за паролями у всій мережі для ідентифікації користувачів з високим ризиком.

#### 2. Зловмисні інсайдери

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 40   |

Зловмисний інсайдер — це працівник, який розкриває приватну інформацію компанії та/або використовує вразливі місця компанії. Зловмисні інсайдери часто є незадоволеними співробітниками. Користувачі, які мають доступ до конфіденційних даних і мереж, можуть завдати значної шкоди через привілейоване неправомірне використання та зловмисні наміри.

Тому для запобігання цього необхідно контролювати доступ до даних та мережі кожного користувача.

### 3. Відсутнє або погане шифрування

Шифрування даних перетворює дані в іншу форму, котру можуть прочитати лише люди, які мають доступ до секретного ключа або пароля. Зашифровані дані зазвичай називають зашифрованим текстом, в той час як незашифровані дані називають відкритим текстом. Метою шифрування даних є захист конфіденційності цифрових даних, що зберігаються в комп'ютерних системах і передаються через Інтернет або інші комп'ютерні мережі. Надійне шифрування має застосовуватися до даних у стані спокою, у русі та, де це можливо, у процесі обробки.

Відсутнє/погане шифрування призводить до того, що конфіденційна інформація, зокрема облікові дані, передається у вигляді відкритого тексту або за допомогою слабких криптографічних шифрів чи протоколів. Це означає, що зловмисник, який перехоплює зберігання, обмін даними або обробку даних, може отримати доступ до конфіденційних даних, використовуючи підходи грубої сили для зламу слабого шифрування.

Для уникнення цього типу вектора атаки потрібно переконатися в тому, що конфіденційні дані зашифровано під час їх зберігання, передачі і обробки.

### 4. Неправильна конфігурація

Неправильна конфігурація — це помилка в конфігурації системи. Наприклад, якщо сторінки налаштування ввімкнено або користувач використовує імена користувачів і стандартні паролі, це може призвести до порушень. Якщо конфігурація сервера додатків не вимкнена, злочинець може визначити приховані

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 41   |

недоліки, і це надасть йому додаткову інформацію. Неправильно налаштовані пристрої і програми є простою точкою доступу для зломисників.

Щоб вберегтися від цього потрібно встановити процедури та системи, що будуть посилювати процес налаштування, а також потрібно використовувати автоматизацію, де це можна зробити. Також важливо проводити моніторинг програм і пристроїв розташованих в мережі.

#### 5. Програми-вимагачі

Програмне забезпечення-вимагач – це форма кібервимагання, за якої користувачі не можуть отримати доступ до своїх даних, доки не буде сплачено викуп . Користувачам показуються інструкції щодо того, як сплатити комісію, щоб отримати ключ дешифрування.

Щоб уникнути атак програм-вимагачів потрібно встановити системи, що захистять всі пристрої мережі від програм-вимагачів. Також не можна встановлювати ПЗ або надавати йому права адміністратора, якщо досконало не відомо, що це за програма і як вона працює.

#### Висновки до розділу 2

В цьому розділі була описана мережа невеликого підприємства. Складено таблицю параметрів підключень мережевий пристроїв та IP-адресації.

Проведено налаштування маршрутизаторів, комутаторів, а також робочих станцій мережі.

Також було проведено аналіз протоколів віддаленого підключення. Зроблено опис основних характеристик та недоліків цих протоколів.

Проведенно аналіз векторів кібератак та визначено методи для захисту мережі підприємства.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 42   |

## РОЗДІЛ 3 НАЛАШТУВАННЯ ТА РЕАЛІЗАЦІЯ ПРОЕКТУ ДЛЯ ЗАХИСТУ WEB-СЕРВЕРА НА БАЗІ ОС LINUX

### 3.1 Налаштування та посилення безпеки Ubuntu Server

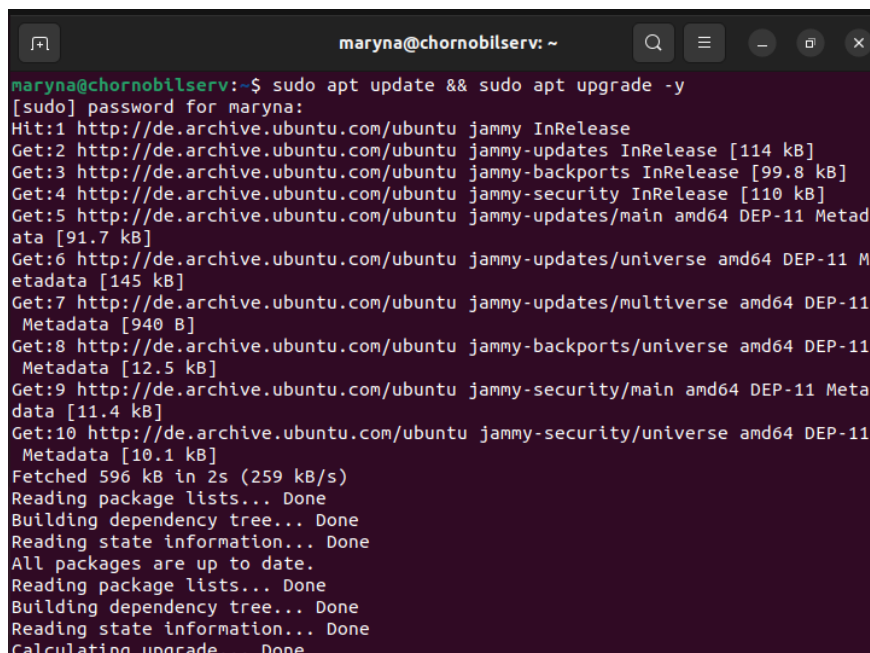
У цьому параграфі було проведено налаштування однієї з ОС Linux, а саме Ubuntu Server, який використовується у даному проекті.

Ключова концепція безпеки полягає в тому, щоб ОС web-сервера була належним чином захищена. Надто часто адміністратори серверів зосереджуються на безпеці на прикладному рівні, наприклад на веб-сервері з динамічним вмістом. Хоча це надзвичайно важливо, не можна забувати про посилення операційної системи сервера, щоб запобігти початковій експлуатації та, як наслідок, атакам після експлуатації, таким як ескалація привілеїв. Захист операційної системи слід запровадити перед розміщенням будь-яких служб, незалежно від того, чи знаходиться система у робочому середовищі чи середовищі розробки.

Отже, далі буде приведено процедуру для посилення безпеки Ubuntu Server.

#### 1. Оновлення сервера.

Перше, що необхідно зробити для захисту сервера, це оновлення локальних сховищ, ОС та встановлених програм:



```
maryna@chornobilserv: ~  
maryna@chornobilserv:~$ sudo apt update && sudo apt upgrade -y  
[sudo] password for maryna:  
Hit:1 http://de.archive.ubuntu.com/ubuntu jammy InRelease  
Get:2 http://de.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]  
Get:3 http://de.archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]  
Get:4 http://de.archive.ubuntu.com/ubuntu jammy-security InRelease [110 kB]  
Get:5 http://de.archive.ubuntu.com/ubuntu jammy-updates/main amd64 DEP-11 Metad  
ata [91.7 kB]  
Get:6 http://de.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 DEP-11 M  
etadata [145 kB]  
Get:7 http://de.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 DEP-11  
Metadata [940 B]  
Get:8 http://de.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 DEP-11  
Metadata [12.5 kB]  
Get:9 http://de.archive.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Meta  
data [11.4 kB]  
Get:10 http://de.archive.ubuntu.com/ubuntu jammy-security/universe amd64 DEP-11  
Metadata [10.1 kB]  
Fetched 596 kB in 2s (259 kB/s)  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
All packages are up to date.  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
Calculating upgrade... Done
```

Рисунок 3.1. Оновлення ОС та встановлених програм на Ubuntu Server

#### 2. Встановлення автоматичної інсталяції оновлень безпеки.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 43   |

Було встановлено пакет `unattended-upgrades`, що дає повний контроль над типом оновлень, котрий хочемо автоматично інсталювати. За допомогою нього встановлено автоматичне оновлення безпеки.

```
maryna@chornobilserv: ~
// Send email to this address for problems or packages upgrades
// If empty or unset then no email is sent, make sure that you
// have a working mail setup on your system. A package that provides
// 'mailx' must be installed. E.g. "user@example.com"
Unattended-Upgrade::Mail "maresidash@gmail.com";

// Set this value to one of:
//   "always", "only-on-error" or "on-change"
// If this is not set, then any legacy MailOnlyOnError (boolean) value
// is used to chose between "only-on-error" and "on-change"
Unattended-Upgrade::MailReport "only-on-error";

// Remove unused automatically installed kernel-related packages
// (kernel images, kernel headers and kernel version locked tools).
Unattended-Upgrade::Remove-Unused-Kernel-Packages "true";

// Do automatic removal of newly unused dependencies after the upgrade
Unattended-Upgrade::Remove-New-Unused-Dependencies "true";

// Do automatic removal of unused packages after the upgrade
// (equivalent to apt-get autoremove)
Unattended-Upgrade::Remove-Unused-Dependencies "false";

// Automatically reboot *WITHOUT CONFIRMATION* if
// the file /var/run/reboot-required is found after the upgrade
Unattended-Upgrade::Automatic-Reboot "false";

// Automatically reboot even if there are users currently logged in
// when Unattended-Upgrade::Automatic-Reboot is set to true
//Unattended-Upgrade::Automatic-Reboot-WithUsers "true";

// If automatic reboot is enabled and needed, reboot at the specific
// time instead of immediately
// Default: "now"
Unattended-Upgrade::Automatic-Reboot-Time "06:00";
```

Рисунок 3.2. Налаштування автоматичних оновлень безпеки та сповіщення на електронну пошту у разі невдачі оновлення, встановлення часу автоматичних оновлень

### 3. Створення нового облікового запису привілейованого користувача.

Для входу на власний сервер не можна входити як `root`, а необхідно створити власний обліковий запис як користувача та надати йому права `sudo`. За допомогою нього і потрібно входити на сервер.

```

maryna@chornobilserv:~$ sudo adduser cyract
Adding user `cyract' ...
Adding new group `cyract' (1001) ...
Adding new user `cyract' (1001) with group `cyract' ...
Creating home directory `/home/cyract' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for cyract
Enter the new value, or press ENTER for the default
  Full Name []: Maryna Chornobil
  Room Number []: 123
  Work Phone []: +380638551237
  Home Phone []: +380635614480
  Other []:
Is the information correct? [Y/n] y
maryna@chornobilserv:~$ sudo usermod -a -G sudo cyract

```

Рисунок 3.3. Створення нового користувача та надання встановлення як root

#### 4. Налаштовано вимоги до складності пароля.

На цьому етапі було перевірено чи ніхто з користувачів не має порожні паролі.

Встановлено вимоги до складності пароля:

- мінімальна довжина пароля 12 символів,
- мінімальну кількість необхідних класів символів для нового пароля, встановлено 4, тобто пароль має налічувати в собі цифри, літери малого та великого регістрів та спеціальні символи,
- відхилення паролів, що містять більше 4 однакових послідовних символів,

#### 5. Встановлено використання ключів SSH для доступу до віддаленого сервера.

#### 6. Заборонено вхід через root в SSH.

#### 7. Змінено порт SSH.

Небезпеку становить те, що за замовчуванням порт SSH – 22, про це знає майже кожен. Тому було розглянуто список портів та знайдено порт, що не використовується іншою програмою.

#### 8. Встановлено брандмауер, ввімкнено його та налаштовано лиш для вказаного дозволу мережевого трафіку.

Оскільки брандауер дозволяє обмежувати вхідний і вихідний трафік на основі набору правил, то поєднавши це з ПЗ для запобігання вторгненням, а саме

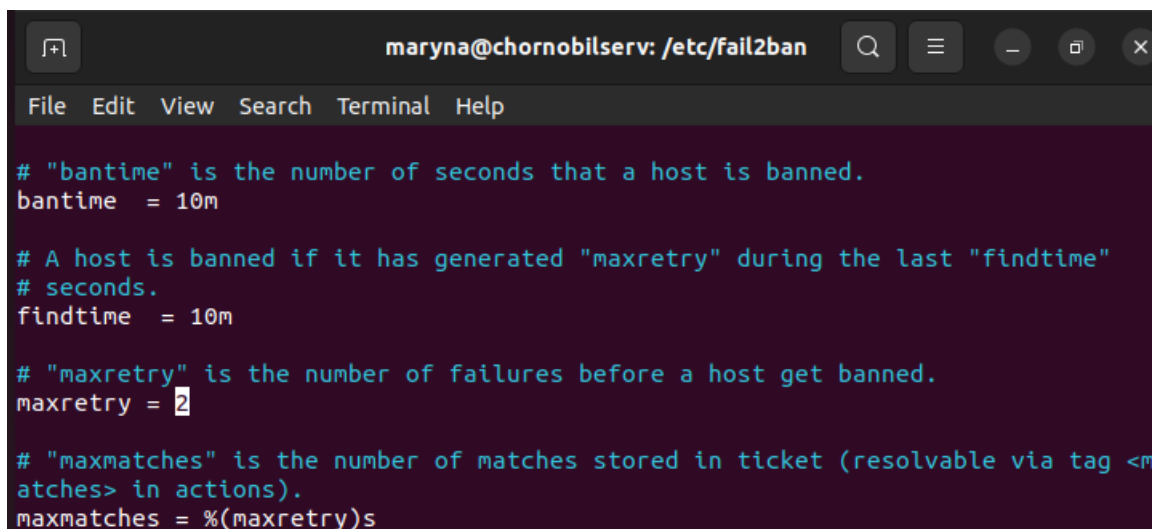
|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 45   |

fail2ban, можливо автоматично заблокувати IP-адресу після попередньо визначеної кількості невдалих спроб входу.

Отже, встановлено та активовано брандмауер ufw. Також встановлено ПЗ для запобігання вторгненням fail2ban.

Fail2ban – це фреймворк ПЗ для запобігання вторгненням, що написаний мовою Python і розроблений для запобігання атакам грубої сили (атаки шляхом перебирання парольних фраз). Він працює шляхом моніторингу файлів журналу для вибраних записів і запущених сценаріїв на їх основі. Найчастіше це використовується для блокування вибраних IP-адрес, що можуть належати хостам, які намагаються порушити безпеку системи. Він може заборонити будь-яку IP-адресу хоста, котрий робить занадто багато спроб входу або виконує будь-яку іншу небажану дію протягом періоду часу, визначеного адміністратором.

За допомогою вищезгаданого додатку встановлено час, на який порушник буде заблокований, максимальну кількість спроб введення паролю:



```
# "bantime" is the number of seconds that a host is banned.
bantime = 10m

# A host is banned if it has generated "maxretry" during the last "findtime"
# seconds.
findtime = 10m

# "maxretry" is the number of failures before a host get banned.
maxretry = 2

# "maxmatches" is the number of matches stored in ticket (resolvable via tag <maxmatches> in actions).
maxmatches = %(maxretry)s
```

Рисунок 3.4. Налаштування параметрів fail2ban

## 9. Видалено непотрібні пакети.

Оскільки немає необхідності запускати FTP-сервер разом із веб-сервером, то такий же принцип стосується деяких інших протоколів, що є потенційно небезпечними.

```

maryna@chornobilserv: ~
E: unable to locate package vsftpd
maryna@chornobilserv:~$ sudo apt-get purge --auto-remove telnetd ftp vsftpd sam
ba nfs-kernel-server nfs-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Package 'nfs-common' is not installed, so not removed
Package 'nfs-kernel-server' is not installed, so not removed
Package 'vsftpd' is not installed, so not removed
Package 'telnetd' is not installed, so not removed
Package 'samba' is not installed, so not removed
The following packages will be REMOVED:
ftp* libcbor0.6* libffi7* libfuse2* ltrace*

```

Рисунок 3.5. Видалення небажаних пакетів

## 10. Створено резервну копію системи Linux.

Надзвичайно важливим кроком є створення резервної копії системи. Адже належне зовнішнє резервне копіювання дозволяє відновити зламаний сервер.

Отже, було інстальовано та налаштовано rsync та rsnapshot пакети.

Rsync — це інструмент командного рядка для копіювання файлів і каталогів між локальною та віддаленою системами, який має бути в наборі інструментів кожного системного адміністратора Linux, а rsnapshot — це утиліта резервного копіювання файлової системи на основі rsync.

## 11. Увімкнено Security-Enhanced Linux

SELinux – функція ядра Linux, що забезпечує механізм підтримки політик безпеки контролю доступу, що забезпечує чудовий захист. SELinux реалізує обов'язковий контроль доступу (MAC). Кожен процес і системний ресурс мають спеціальну мітку безпеки, яка називається *контекстом SELinux*. Контекст SELinux, який іноді називають міткою SELinux, — це ідентифікатор, який абстрагує деталі системного рівня та зосереджується на властивостях безпеки сутності. Це не тільки забезпечує послідовний спосіб посилання на об'єкти в політиці SELinux, але й усуває будь-яку двозначність, яку можна знайти в інших методах ідентифікації; наприклад, файл може мати кілька дійсних імен шляхів у системі, яка використовує зв'язування монтунів.

Перевагами запуску SELinux стали такі функції:

- Усі процеси та файли позначені. Правила політики SELinux визначають, як процеси взаємодіють з файлами, а також як процеси взаємодіють один з

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 47   |

одним. Доступ дозволено, лише якщо існує правило політики SELinux, яке це дозволяє.

- Точний контроль доступу. Виходячи за рамки традиційних дозволів UNIX, якими керує користувач на розсуд і на основі ідентифікаторів користувачів і груп Linux, рішення про доступ до SELinux базуються на всій доступній інформації, такій як користувач SELinux, роль, тип і, за бажанням, рівень безпеки.
- Політику SELinux визначають адміністративно та застосовують у всій системі.
- Покращено пом'якшення атак підвищення привілеїв. Процеси виконуються в доменах і тому відокремлені один від одного. Правила політики SELinux визначають, як процеси отримують доступ до файлів та інших процесів. Якщо процес скомпрометовано, зловмисник має доступ лише до звичайних функцій цього процесу та до файлів, доступ до яких налаштовано для процесу. Наприклад, якщо HTTP-сервер Apache зламано, зловмисник не може використовувати цей процес для читання файлів у домашніх каталогах користувача, якщо не було додано або налаштовано спеціальне правило політики SELinux, яке дозволяє такий доступ.
- SELinux можна використовувати для забезпечення конфіденційності та цілісності даних, а також для захисту процесів від ненадійних вхідних даних.

SELinux – це дуже складна програма для встановлення, тому що якщо її не налаштувати належним чином до першого перезавантаження, це зробить неможливим завантаження всієї операційної системи, тобто все, що окрім екрана початкового завантаження, буде практично недоступним звичайними засобами.

Крім того, Ubuntu вже має складну систему обов'язкового контролю доступу високого рівня, відому як AppArmor, і тому її було вимкнено перед встановленням SELinux, щоб уникнути будь-яких конфліктів.

Також було змінено файл конфігурації утиліти та встановлено наступні налаштування (Рис. 3.6): SELINUX=permissive, SELINUXTYPE=default. Причина

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 48   |

призначення правила дозволу для SELinux полягає в тому, щоб зробити операційну систему доступною, залишаючи SeLinux увімкненим.

```

maryna@chornobil: /etc/selinux
GNU nano 6.2 /etc/selinux/config
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
# enforcing - SELinux security policy is enforced.
# permissive - SELinux prints warnings instead of enforcing.
# disabled - No SELinux policy is loaded.
SELINUX=permissive
# SELINUXTYPE= can take one of these two values:
# default - equivalent to the old strict and targeted policies
# mls      - Multi-Level Security (for military and educational use)
# src      - Custom policy built from source
SELINUXTYPE=default

# SETLOCALDEFS= Check local definition changes
SETLOCALDEFS=0

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^_ Replace    ^U Paste     ^J Justify
  
```

Рисунок 3.6. Редагування конфігураційного файлу SELinux

SELinux – це комплексний модуль, тому він містить багато функцій і опцій. З огляду на це, більшість із цих варіантів можуть бути корисними не всім через їх екзотичний характер. Було встановлено наступні параметри, що одними з основних і найкорисніших параметрів цього модуля.

```

maryna@chornobil:~$ sudo sestatus
[sudo] password for maryna:
SELinux status:                enabled
SELinuxfs mount:              /sys/fs/selinux
SELinux root directory:       /etc/selinux
Loaded policy name:            default
Current mode:                  permissive
Mode from config file:         permissive
Policy MLS status:             enabled
Policy deny_unknown status:    allowed
Memory protection checking:    requested (insecure)
Max kernel policy version:     33
maryna@chornobil:~$
  
```

Рисунок 3.7. Перевірено статус SELinux за допомогою команди sudo sestatus

Важливим налаштуванням стало встановлення глобального рівня доступу для утиліти SELinux. Він визначає як SELinux поводить, коли натрапляє на правило. За замовчуванням SeLinux встановлює режим примусового виконання,

який фактично блокує всі запити, але його можна змінити на дозвіл, який є дещо поблажливим до користувача, оскільки дозволяє доступ, проте реєструє будь-які порушені правила у своєму файлі журналу.

Наступним кроком було увімкнення і вимкнення політик та засобів захисту, що запропоновані, вона є однією з найважливіших опцій у SELinux, оскільки вона дозволяє вмикати та вимикати політики. SELinux має велику кількість попередньо створених політик, які визначають, чи дозволений вказаний запит чи ні. Деякі приклади цього: `allow_ftpd_full_access`, котрий визначає здатність служби FTP входити до локальних користувачів і читати, записувати всі файли в системі, `allow_ssh_keysign`, що дозволяє використовувати ключі під час входу в SSH, `allow_user_mysql_connect`, який дозволяє користувачам підключатися до `mysql`, `httpd_can_sendmail`, який визначає здатність служби HTTP надсилати електронні листи тощо.

За допомогою наступного фрагменту коду було встановлено `polycoreutils-python-utils`, який фактично допомагає в описовому переліку кожної політики, а потім перераховує всі доступні політики для термінал, нарешті, він навчає, як вмикати чи вимикати політику:

```
apt-get install polycoreutils-python-utils
semanage boolean -l
setsebool -P allow_ftpd_full_access ON
```

Наступним було налаштування контролю доступу на основі ролей (RBAC).

RBAC дозволяє адміністраторам перемикатися на спосіб на основі ролей, щоб обмежити дозволи програм. Це означає, що користувачу певної групи користувачів дозволено виконувати певні попередньо визначені дії. Поки користувач є частиною ролі, це нормально. Це те ж саме, що й перехід на `root` під час інсталяції програм у Linux з правами адміністратора.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 50   |

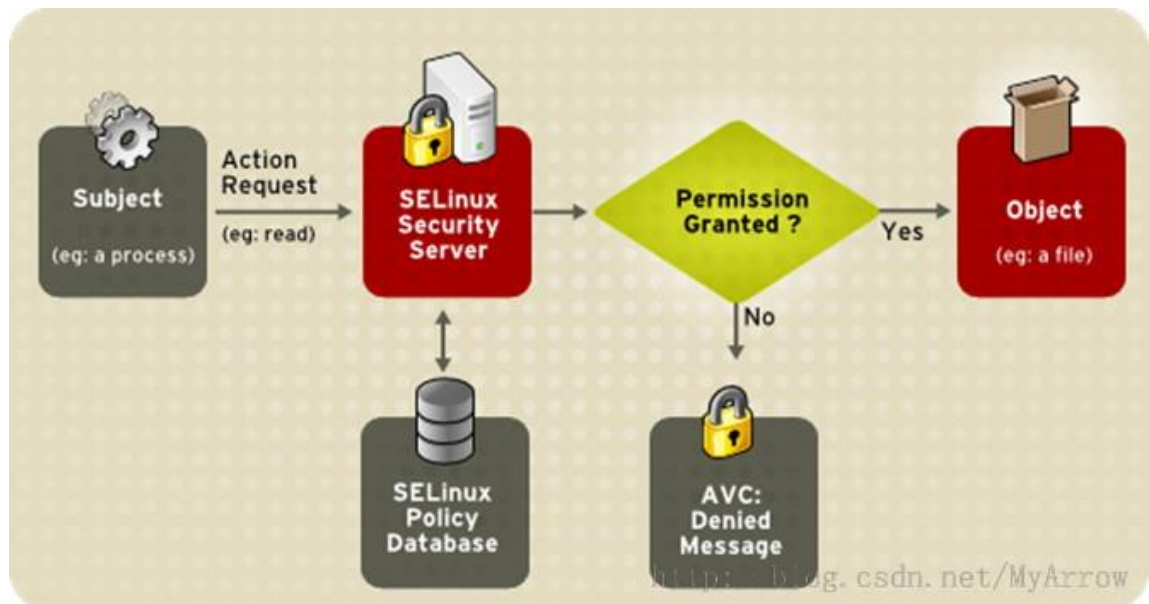


Рисунок 3.8. Реалізація керування доступом на основі ролей у SELinux

### 3.2. Налаштування захисту для сервера NGINX

Першим кроком на даному етапі стала установка сервера NGINX. Його було обрано через його надійність, швидкість та підтримку всіх основних ОС. Після інсталяції серверу було досліджено параметри його конфігурації. Вони містяться у файлі NGINX.conf. Оскільки він є основним файлом для конфігурації, більша частина перевірок безпеки виконана саме за допомогою нього. Також виникла необхідність внести зміни в файли конфігурації віртуального хоста.

Отже, налаштування NGINX зроблено в наступному порядку:

#### 1. Оновлено ПЗ сервера

Основна кодова база NGINX (керування пам'яттю, обробка сокетів, тощо) є дуже безпечною та стабільною, хоча вразливості в самому головному двійковому файлі час від часу з'являються. Через це дуже важливо підтримувати NGINX в актуальному стані. Більшість сучасних дистрибутивів Linux не розміщують останню версію NGINX у своїх списках пакетів за замовчуванням, тому, щоб інстальювати останню версію NGINX через пакет, може знадобитися додати додаткові сховища пакетів у систему.

#### 2. Зібрано Web-сервер з вихідних кодів

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 51   |

Це дозволило запустити останню версію, доступну від команди розробників NGINX, і надасть додаткові конфігурації безпеки.

```
maryna@chornobil: ~
maryna@chornobil:~$ sudo apt-get update && sudo apt-get install automake gcc \l
ibpcre3-dev zlib1g-dev make -y
Hit:1 http://ua.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://ua.archive.ubuntu.com/ubuntu jammy-updates InRelease
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:4 http://ua.archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 110 kB in 1s (143 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
make is already the newest version (4.3-4.1build1).
```

Рисунок 3.9. Оновлення та завантаження необхідних пакетів ОС

Наступним кроком було завантажено вихідний архів і його підпис з офіційного сайту NGINX, а також було отримано ключ підпису розробника. І вже після верифікації розпаковано архів та інстальовано NGINX.

### 3. Вимкнено та видалено деякі небажані модулі веб-сервера

Велика кількість модулів, що були атоматично ввімкнені при інсталяції має бути відключена. Це необхідно для мінімізації ризику можливих атак за допомогою обмеження операцій, що дозволені. Для відключення деяких модулів було перекомпільовано NGINX.

### 4. Вимкнено маркери сервера

Специфікація HTTP рекомендує (але не вимагає), щоб web-сервери ідентифікували себе за допомогою заголовка Server. Історично web-сервери включали інформацію про свою версію як частину цього заголовка. Розкриття запусченої версії NGINX може бути небажаним, особливо в середовищах, чутливих до розкриття інформації. NGINX можна налаштувати таким чином, щоб не відображати свою версію в заголовку сервера, відредагувавши файл NGINX.conf (Рис. 3.10):

```

maryna@chornobil: /etc/nginx
GNU nano 6.2 nginx.conf *
include /etc/nginx/mime.types;
default_type application/octet-stream;
server_tokens off;
##
# SSL Settings
##

```

Рисунок 3.10. Відредагований файл NGINX.conf

## 5. Встановлено контроль ресурсів і обмежень

Для запобігання DoS-атакам, а також атакам переповнення буфера на web-сервер встановлено обмеження для розміру буфера для усіх клієнтів. Цу було реалізовано у файлі NGINX.conf з використанням наступних директив:

- client\_header\_buffer\_size – директива для призначення розміру буфера заголовка для заголовка запиту клієнта;
- client\_body\_buffer\_size – директива для призначення розміру буфера тіла для запиту клієнта;
- client\_max\_body\_size – директива для призначення максимального розміру тіла для запиту клієнта;
- large\_client\_header\_buffers – директива, що вказує максимальну кількість та розмір буферів, що будуть використані для читання великих заголовків запитів клієнта.

```

maryna@chornobil: /etc/nginx
GNU nano 6.2 nginx.conf *
access_log /var/log/nginx/access.log;
error_log /var/log/nginx/error.log;
client_body_buffer_size 1k;
client_header_buffer_size 1k;
client_max_body_size 1k;
large_client_header_buffers 2 1k;

```

Рисунок 3.11. Встановлені обмеження у файлі NGINX.conf

## 6. Вимкнено непотрібні методи HTTP

На сервері було вимкнено методи HTTP, що не будуть у використанні, тому було додано умову в блок розташування у файлі NGINX.conf. За допомогою

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 53   |

неї сервер дозволить лише методи GET, HEAD і POST і відфільтруватиме такі методи, як DELETE і TRACE.

```

maryna@chornobil: /etc/nginx
GNU nano 6.2 nginx.conf *
    large_client_header_buffers 2 1k;
    location / {
        limit_except GET HEAD POST { deny all; }
    }
  
```

Рисунок 3.12. Додавання умови в конфігураційний файл

## 7. Встановлено ModSecurity

NGINX ModSecurity WAF — це брандмауер веб-додатків (WAF), заснований на ModSecurity 3.0, переписаному програмному забезпеченні ModSecurity, яке працює як динамічний модуль для NGINX Plus. NGINX ModSecurity WAF можна використовувати для зупинки широкого спектру атак рівня 7 і реагування на нові загрози за допомогою віртуальних виправлень.

Перед встановленням були завантажені наступні необхідні пакети:

```

maryna@chornobil:/usr/local/src/nginx$ sudo apt-get install bison build-essential
ca-certificates curl dh-autoreconf doxygen flex gawk git iputils-ping libcurl4-gnutls-dev
libexpat1-dev wget zlib1g-dev zlibc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
  
```

Рисунок 3.13. Завантаження пакетів вихідного коду ModSecurity

Наступним етапом стало завантаження та компіляція ModSecurity.

## 8. Налаштовано журнали доступу та журнали помилок

Журнали доступу та помилок NGINX ввімкнено за замовчуванням. Вони знаходяться в файлах logs/error.log та logs/access.log. При необхідності зміни їх розташування використано директивою error\_log у файлі конфігурації сервера. Дану директиву також можна використати для вказання журналів, що записуватимуться. Також змінено директиву access\_log в конфігураційному файлі для вказання розташування, якщо воно відмінне від звичайного для журналів доступу.

## 9. Налаштовано NGINX для включення заголовків безпеки

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 54   |

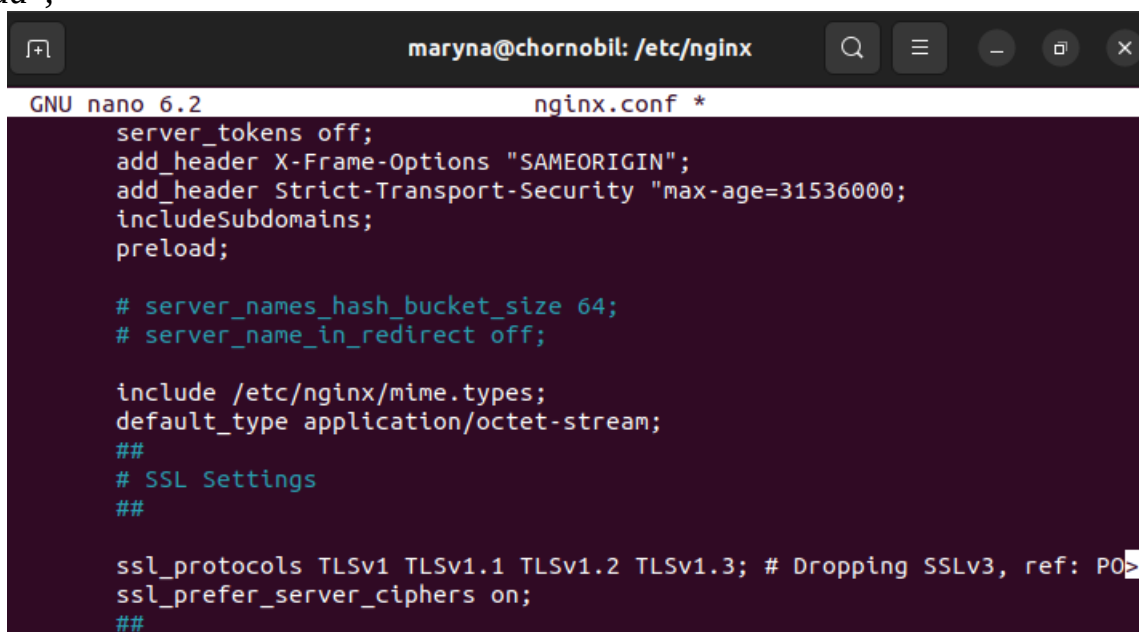
Для посилення безпеки NGINX було додано декілька різних заголовків HTTP. В даному випадку було налаштовано кожен із зазначених.

Для запобігання атакам клікджекінгу було використано заголовок HTTP-відповіді X-Frame-Options. Це дозволяє вказати чи потрібно браузеру відтворювати веб-сторінку в frame чи iframe. Для реалізації цього було додано параметр у файл конфігурації сервера:

```
add_header X-Frame-Options "SAMEORIGIN";
```

HTTP Strict Transport Security (HSTS) — це метод, що використовується веб-сайтами для оголошення того, що доступ до них має здійснюватися лише через безпечне з'єднання (HTTPS). Якщо web-сайт оголошує політику HSTS, браузер повинен відхиляти всі HTTP-з'єднання та забороняти користувачам приймати незахищені сертифікати SSL. Для його додавання також було додано параметр до розділу сервера:

```
add_header Strict-Transport-Security "max-age=31536000;includeSubdomains;preload";
```



```
maryna@chornobil: /etc/nginx
GNU nano 6.2 nginx.conf *
server_tokens off;
add_header X-Frame-Options "SAMEORIGIN";
add_header Strict-Transport-Security "max-age=31536000;
includeSubdomains;
preload;

# server_names_hash_bucket_size 64;
# server_name_in_redirect off;

include /etc/nginx/mime.types;
default_type application/octet-stream;
##
# SSL Settings
##

ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref: PO
ssl_prefer_server_ciphers on;
##
```

Рисунок 3.14. Налаштування X-Frame-Options та HSTS

Політика безпеки вмісту (CSP) захищає ваш веб-сервер від певних типів атак, у тому числі атак міжсайтового сценарію (XSS) і атак із впровадженням даних.

Було реалізовано CSP, з додаванням заголовка Content-Security-Policy (фактичний заголовок має був налаштований відповідно до унікальних вимог):

```
add_header Content-Security-Policy "default-src 'self' http: https: data: blob:
'unsafe-inline'" always;
```

## 10. Створено SSL сертифікат

TLS/SSL працює за допомогою комбінації публічного сертифіката та закритого ключа. Ключ SSL зберігається на сервері в секреті. Він використовується для шифрування вмісту, який надсилається клієнтам. Сертифікат SSL доступний для всіх, хто запитує вміст. Його можна використовувати для дешифрування вмісту, підписаного відповідним ключем SSL.

```
maryna@chornobil: /etc/nginx/certificate
maryna@chornobil:~$ cd /etc/nginx/certificate
maryna@chornobil:/etc/nginx/certificate$ openssl req -new -newkey RSA:4096 -x509 -sha256 -days 365 -nodes -out nginx-certificate.crt -keyout nginx.key
.....+.....
```

Рисунок 3.15. Створення самопідписаного ключа та сертифіката за допомогою OpenSSL

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:UA
State or Province Name (full name) [Some-State]:Zhytomyrska obl.
Locality Name (eg, city) []:Zhytomyr
Organization Name (eg, company) [Internet Widgits Pty Ltd]:ZTU
Organizational Unit Name (eg, section) []:KB-2
Common Name (e.g. server FQDN or YOUR name) []:127.0.1.1
Email Address []:maresidash@gmail.com
```

Рисунок 3.16. Відповіді на запитання про сервер для сертифіката

```
maryna@chornobil:/etc/nginx/certificate$ sudo openssl dhparam -out /etc/nginx/dhparam.pem 4096
Generating DH parameters, 4096 bit long safe prime
.....+.....
.....+.....
.....+.....
.....+.....
.....+.....
```

Рисунок 3.17. Створення групи Деффі-Хеллмана

Також було створено новий фрагмент, що вказує на ключ і сертифікат, конфігурації NGINX у /etc/NGINX/snippets каталозі. У файлі self-signed.conf було встановлено директиву для файлу сертифіката та пов'язаного ключа.

```

maryna@chornobil: /etc/nginx/snippets
GNU nano 6.2 self-signed.conf *
ssl_certificate /etc/ssl/certs/nginx-selfsigned.crt;
ssl_certificate_key /etc/ssl/private/nginx-selfsigned.key;

```

Рисунок 3.18. Встановлення нової директиви у файлі srlf-signed.conf

Наступним кроком стало створення фрагмента конфігурації з параметрами надійного шифрування. В ньому було додано розпізнавач DNS для вихідних запитів.

```

maryna@chornobil: /etc/nginx/snippets
GNU nano 6.2 /etc/nginx/snippets/ssl-params.conf
ssl_protocols TLSv1.3;
ssl_prefer_server_ciphers on;
ssl_dhparam /etc/nginx/dhparam.pem;
ssl_ciphers EECDH+AESGCM:EDH+AESGCM;
ssl_ecdh_curve secp384r1;
ssl_session_timeout 10m;
ssl_session_cache shared:SSL:10m;
ssl_session_tickets off;
ssl_starling on;
ssl_starling_verify on;
resolver 8.8.8.8 8.8.4.4 valid=300s;
resolver_timeout 5s;
#add_header Strict-Transport-Security "max-age=63072000; includeSubDomains; pr
add_header X-Frame-Options DENY;
add_header X-Content-Type-Options nosniff;
add_header X-XSS-Protection "1; mode=block";

```

Рисунок 3.19. Встановлення параметрів надійного шифрування у файлі ssl-params.conf

### 3.3. Тестування реалізованих методів безпеки для Web-сервера

На даному етапі було проведено тестування заходів для протидії кібератакам. Для збору інформації було використано утиліту Nmap.

Nmap дозволяє сканувати вашу мережу та знаходити не лише все, що до неї підключено, але й широкий спектр інформації про те, що підключено, які служби працює на кожному хості тощо. Він дозволяє використовувати велику кількість методів сканування, таких як UDP, TCP-з'єднання (), TCP SYN (напіввідкритий) і FTP. Він також пропонує широкий вибір типів сканування, включаючи проксі (атака відскоку), зворотний ідентифікатор, ICMP (перевірка ping), FIN, розгортка

ACK, Xmas, розгортка SYN, IP-протокол і нульове сканування. Додаткову інформацію див. у розділі Методи сканування сторінки довідки.

Nmap також пропонує низку розширених функцій, таких як виявлення операційної системи за допомогою відбитків пальців TCP/IP, скритного сканування, динамічної затримки та обчислень повторної передачі, паралельного сканування, виявлення непрацюючих хостів за допомогою паралельних пінгів, сканування приманки, виявлення фільтрації портів, прямого сканування RPC (без відображення портів), сканування фрагментації та гнучкої цілі та специфікація портів.

Спершу було встановлено утиліту Nmap. Наступним кроком стало сканування за допомогою імені хоста.

```
maryna@chornobil:~$ nmap -sV 127.0.1.1
Starting Nmap 7.80 ( https://nmap.org ) at 2022-08-29 02:33 EEST
Nmap scan report for chornobil (127.0.1.1)
Host is up (0.00014s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3 (Ubuntu Linux; protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.81 seconds
```

Рисунок 3.20. Результат перевірки відкритих портів сервера

Наступним етапом стала перевірка наявності експлойтів для сервісів та ОС згідно версій.

Отже, першим етапом була перевірка стандартного підключення до маршрутизатора Main-Router по протоколу SSH та Telnet з використанням стандартних портів.

Далі проведено підключення за новими портами SSH а саме 81, з різними користувачами.

### Висновки до розділу 3

У цьому розділі було проведено налаштування та посилення безпеки ОС Ubuntu Server.

Також було проведено встановлення та налаштування заходів безпеки для сервера NGINX.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобиль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 58   |

Кінцевим етапом стало проведення тестування реалізованих методів безпеки.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 59   |

## ВИСНОВКИ

У даній кваліфікаційній роботі було розглянуто ключові аспекти захисту Web-сервера.

В першому розділі було проведено аналіз основних вразливостей web-серверів та типів загроз для них. Розглянуто теоретичні відомості щодо захисту та протидії кібератакам. Також був проведений огляд популярного web-серверів на відкритому ринку та розглянуто саме поняття «веб-сервер». Оглянуто та проаналізовано два найпопулярніші web-сервери – Apache і NGINX. Також було проведено їх порівняння та обрано Web-сервер для подальшого використання у проекті. Наприкінці було розглянуто критичні уразливості сервера NGINX за останній час.

В другому розділі була описана мережа невеликого підприємства. Складено таблицю параметрів підключень мережевий пристроїв та IP-адресації. Проведено налаштування маршрутизаторів, комутаторів, а також робочих станцій мережі. Також було проведено аналіз протоколів віддаленого підключення. Зроблено опис основних характеристик та недоліків цих протоколів. Проведенно аналіз векторів кібератак та визначено методи для захисту мережі підприємства.

У третьому розділі У цьому розділі було проведено налаштування та посилення безпеки ОС Ubuntu Server. Також було проведено встановлення та налаштування заходів безпеки для сервера NGINX. Кінцевим етапом стало проведення тестування реалізованих методів безпеки.

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 60   |

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws. Dafydd Stuttard, Markus Pinto, 2011. 912 с.
2. Hardware & Security. Tobias Scheible, 2022. 548с.
3. Nginx Security: How to harden your server configuration URL:  
<https://www.acunetix.com/blog/web-security-zone/hardening-nginx>
4. Most popular Web Server Application URL:  
<https://digitalintheround.com/what-is-the-most-popular-web-server>
5. How to configure Security Headers in Nginx and Apache URL:  
<https://webdock.io/en/docs/how-guides/security-guides/how-to-configure-security-headers-in-nginx-and-apache>
6. SELinux on Ubuntu Tutorial URL: <https://linuxhint.com/selinux-on-ubuntu-tutorial>
7. What is SSH: Understanding Encryption URL:  
<https://www.hostinger.com/tutorials/ssh-tutorial-how-does-ssh-work>
8. Top 5 most critical NGINX Vulnerabilities URL:  
<https://www.getastra.com/blog/911/top-5-most-critical-nginx-vulnerabilities-found>
9. Cisco 7200 Series Routers Overview URL:  
<https://www.cisco.com/c/en/us/support/routers/7200-series-routers/series.html>
- 10.Ddos attcs prevention with NGINX URL:  
<https://inmediatum.com/en/blog/engineering/ddos-attacks-prevention-nginx/>
- 11.Web Server Vulnerabilities Attacks: How to Protect Your Organization URL: <https://www.techfunnel.com/information-technology/web-server-vulnerabilities-attacks-how-to-protect-your-organization>

# ДОДАТКИ

|      |      |                  |        |      |                          |      |
|------|------|------------------|--------|------|--------------------------|------|
|      |      | Чорнобіль М.Р.   |        |      | КІК.КР.Б – 125 – 24 – ПЗ | Арк. |
|      |      | Лобанчикова Н.М. |        |      |                          |      |
| Змн. | Арк. | № докум.         | Підпис | Дата |                          | 62   |

Проект мережі підприємства

