МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут автоматики і електротехніки

Кафедра комп'ютерних технологій та інформаційної безпеки

ЗАТВЕРДЖУЮ
Завідувач кафедри комп'ютерних технологій та інформаційної безпеки, к.т.н., доцент
_____ С.М. Нужний
«18» 06 2025 р.

Кваліфікаційна робота
на правах рукопису

КВАЛІФІКАЦІЙНА РОБОТА

РОЗРОБКА НАВЧАЛЬНОГО КОМПЛЕКСУ З НАПРЯМКУ ЗАХИСТУ КОМП'ЮТЕРНИХ МЕРЕЖ

Ступінь вищої освіти: БАКАЛАВР
Галузь знань: 12 «Інформаційні технології»
Спеціальність: 125 «Кібербезпека та захист інформації»
Освітньо-професійна програма: Системи технічного захисту інформації, автоматизація її обробки

Виконав: студент 4 курсу групи 4381
Іванов Дмитро Валерійович

Кваліфікаційна робота містить результати самостійного виконання навчального завдання. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело
_____ Іванов Д.В.

Керівник: доцент кафедри комп'ютерних технологій та інформаційної безпеки к.т.н., доцент **Сірівчук Андрій Сергійович** _____

Миколаїв – 2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут автоматики і електротехніки

ЗАТВЕРДЖУЮ
Гарант освітньої програми, к.т.н., доцент, завідувач кафедри КТІБ
_____С.М. Нужний
«_18_» ____06____ 2025 р.

ЗАВДАННЯ

на кваліфікаційну роботу

Студент: Іванов Дмитро Валерійович

Курс:4

Група: 4381

Ступінь вищої освіти: бакалавр

Галузь знань: 12 Інформаційні технології

Спеціальність: 125 «Кібербезпека та захист інформації»

Освітньо-професійна програма: Системи технічного захисту інформації, автоматизація її обробки

1. Тема роботи: «Розробка навчального комплексу з напрямку захисту комп'ютерних мереж»

затверджена наказом НУК від «_23__» _квітня_____2025 р. № УЧ- 343

2. Вихідні дані до роботи: розгортання симулятору з підтримкою мережевих операційних систем та засобами їх діагностики.
3. Перелік питань, які необхідно розробити: провести аналіз основних питань вивчення інформаційного захисту в комп'ютерних мережах; провести огляд симуляторів комп'ютерних мереж; розгортання та налаштування системи симуляції
4. Терміни виконання завдання:
   термін видачі завдання: «_03_» лютого 2025 р.
   термін подання роботи: «18» червня 2025 р.

№ п/п	Заходи	Дата		Готовність	Відмітка про виконання
		Навч. тиждень	Контрольна дата		
10.	Організаційні збори.	23	03.02.2025	Вибір теми, визначення мети, завдань, об'єкта та предмета дослідження	вик
11.	Організаційні збори	25	20.02.2025	Попередній варіант оглядових розділів, підбір і опрацювання списку використаних джерел	вик
12.	Рубіжний контроль	31	03.04.2025	Попередній варіант повної КР.	вик
13.	Рубіжний контроль	33	20.04.2025	Попередній варіант презентації.	вик
14.	ЄДКІ на рівень «бакалавр»	34	29.04.2025	Здавання ЄДКІ зі спеціальності на ступінь бакалавра	вик
15.	Перевірка на плагіат	42	14.06.2025	1. Електронний варіант кваліфікаційної роботи; 2. Подання КР на перевірку на плагіат.	вик
16.	Кафедральний захист	43	18.06.2025	1. Оформлена КР (в форматі pdf) (передається студентом на кафедрі для внутрішньої рецензії, висновок керівника). У разі отримання позитивної внутрішньої рецензії, КР подається на зовнішню рецензію. 2. Оформлена презентація (в форматі pdf).	вик
17.	Подання документів на захист	44	24.06.2025	1. Подання щодо захисту КР 2. Зовнішня рецензія (окремо від КР). Резюме на КР. 3. Висновок кафедри про КР, допуск до захисту; 4. Електронний варіант та роздрукована презентація в кількості 5 примірників. 5. Електронний варіант КР на диску	вик
18.	Захист КР	44	25.06.2025	1. Приєднання студента до засідання кваліфікаційної комісії (конференції) у встановлений час для проведення процедури дистанційного захисту 2. Процедура захисту КР.	вик

ПРИМІТКА: Завдання додається до виконаної роботи та подається до атестаційної комісії.

Керівник
Доцент кафедри комп'ютерних технологій
та інформаційної безпеки, к.т.н., доцент _____ А.С. Сірівчук

Студент _____ Д.В. Іванов

АННОТАЦІЯ

Іванов Д.В. Розробка навчального комплексу з напряму захисту комп’ютерних мереж. – Кваліфікаційна робота на правах рукопису.

Кваліфікаційна робота на здобуття ступеня вищої освіти «бакалавр» за спеціальністю 125 «Кібербезпека та захист інформації» галузі знань 12 «Інформаційні технології», освітньо-професійна програма «Системи технічного захисту інформації, автоматизація її обробки». – Національний університет кораблебудування імені адмірала Макарова, Миколаїв, 2025.

Пояснювальна записка: 58 с., 41 рис., 20 джерело.

Кваліфікаційна робота присвячена актуальній темі – підвищенню ефективності підготовки фахівців у галузі кібербезпеки шляхом створення навчального комплексу з вивчення захисту комп’ютерних мереж.

Об’єктом кваліфікаційної роботи є підготовка віртуального навчального комплексу дослідження комп’ютерних мереж. Предметом кваліфікаційної роботи є розгортання, встановлення необхідних модулів та попередні налаштування симулятор комп’ютерних мереж.

Метою роботи є створення віртуального середовища для моделювання комп’ютерних мереж.

У роботі проведено аналіз сучасних платформ симуляції мережевих середовищ, обґрунтовано вибір *PNETLab* як базової технології, здійснено розгортання середовища на базі *Proxmox*, реалізовано лабораторні завдання, систему моніторингу ресурсів та управління обліковими записами.

Отримані результати можуть бути використані в навчальному процесі вищих навчальних закладів, при сертифікаційній підготовці, а також у корпоративному навчанні фахівців з кібербезпеки.

Ключові слова: кібербезпека, захист комп’ютерних мереж, навчальний комплекс, *PNETLab*, віртуальна лабораторія, симуляція.

ANNOTATION

Ivanov D.V. Development of an Educational Complex in the Field of Computer Network Security. – Qualification work presented as a manuscript.

Bachelor's qualification work for obtaining the degree of higher education "Bachelor" in specialty 125 "Cybersecurity and Information Protection", field of knowledge 12 "Information Technologies", educational-professional program "Technical Information Security Systems, Automation of Information Processing". – Admiral Makarov National University of Shipbuilding, Mykolaiv, 2025.

Explanatory note: 58 pages, 41 figures, 20 references.

The qualification work is devoted to an urgent topic – improving the effectiveness of training specialists in the field of cybersecurity by developing an educational complex for studying computer network protection.

The object of the qualification work is the preparation of a virtual educational complex for the study of computer networks. The subject of the qualification work is the deployment, installation of the necessary modules and preliminary settings of the computer network simulator.

The purpose of the work is to create a virtual environment for modeling computer networks.

The work analyzes modern platforms for simulating network environments, justifies the choice of PNETLab as the basic technology, deploys the environment based on Proxmox, implements laboratory tasks, a resource monitoring system and account management.

The results obtained can be used in the educational process of higher educational institutions, in certification training, as well as in corporate training of cybersecurity specialists.

Keywords: *cybersecurity, computer network protection, educational complex, PNETLab, virtual laboratory, simulation.*

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ

ВМ - віртуальна машина.

ІТ – інформаційні технології.

ОС - операційна система.

ПК – персональний комп’ютер.

СКМ - симулятор комп’ютерних мереж.

ВСТУП

У сучасному світі, де інформаційні технології стали невід'ємною частиною повсякденного життя, питання захисту комп'ютерних мереж набувають особливої ваги. Зі зростанням обсягу даних, що обробляються, та розвитку технологій, зростають і кіберзагрози, які можуть призвести до значних фінансових, репутаційних та правових наслідків для організацій і окремих користувачів.

Необхідність розробки ефективних методів навчання та підготовки фахівців із кібербезпеки є однією з головних вимог сучасного ринку праці. Зважаючи на те, що кіберзагрози стають все більш складними, фахівці з інформаційної безпеки повинні мати практичні навички для боротьби з ними, що можна забезпечити через інтерактивні навчальні комплекси. Створення таких комплексів дозволяє значно підвищити рівень підготовки студентів та практиків, допомогти їм в освоєнні реальних сценаріїв атак і розробці стратегій їх подолання.

Метою кваліфікаційної роботи є розробка навчального комплексу для підготовки фахівців у сфері захисту комп'ютерних мереж на базі симулятору комп'ютерних мереж (СКМ).

Об'єктом кваліфікаційної роботи є підготовка віртуального навчального комплексу дослідження комп'ютерних мереж.

Предметом кваліфікаційної роботи є розгортання, встановлення необхідних модулів та попередні налаштування СКМ.

Для вирішення поставленої мети вирішено наступні завдання:

– провести огляд основних питань вивчення інформаційного захисту в комп'ютерних мережах;

– провести огляд СКМ;

– провести розгортання СКМ;

– провести початкові налаштування системи;

– побудова навчальної лабораторії для тестування системи.

Результати роботи можуть бути використані як для підготовки студентів у навчальних закладах, так і для професійного розвитку фахівців, які займаються питаннями інформаційної безпеки.

РОЗДІЛ 1. ПРОБЛЕМИ ЗАХИСТУ КОМП’ЮТЕРНИХ МЕРЕЖ

1.1 Поняття комп’ютерної мережі та її роль в сучасному світі

Комп'ютерна мережа – це система взаємозв'язаних комп'ютерів та інших електронних пристроїв, які обмінюються даними між собою через певні канали зв'язку. Основними елементами мережі є комп'ютери, маршрутизатори, комутатори, сервери, клієнтські пристрої, а також програмне забезпечення, що забезпечує їх взаємодію. Мережі можуть бути локальними (*LAN*), глобальними (*WAN*) або віртуальними (*VPN*), що дає змогу підключати пристрої в межах однієї будівлі чи навіть на великі відстані по всьому світу.

Комп’ютерні мережі є основою для обміну інформацією в сучасному світі, з'єднуючи комп'ютери, пристрої та сервери для передачі даних. Мережі дозволяють ефективно передавати інформацію на великі відстані, забезпечуючи комунікацію між користувачами та пристроями в реальному часі. Вони займають центральне місце у розвитку технологій, змінюючи спосіб ведення бізнесу, надання послуг та соціальних взаємодій. Вивчення комп’ютерних мереж є важливим для розуміння функціонування сучасного цифрового світу та забезпечення його безпеки.

Комп'ютерні мережі виконують ключову роль у різних сферах сучасного життя [1]:

– комунікація: мережі дозволяють швидко передавати дані через Інтернет або внутрішні корпоративні мережі. Вони стали основою для електронної пошти, відеоконференцій, чатів і голосових дзвінків, що сприяє глобальній комунікації та співпраці;

– електронна комерція: сучасні торгові платформи, банки, а також онлайн-сервіси працюють через мережі, що дозволяє людям здійснювати покупки, оплату послуг і управління фінансами через Інтернет, що змінює традиційний бізнес;

– медичні технології: телемедицина, дистанційне обстеження пацієнтів і передача медичних даних стали можливими завдяки розвитку комп’ютерних мереж, що дозволяє лікарям надавати якісні послуги навіть на відстані;

– освіта: Враховуючи глобалізацію, багато навчальних закладів почали застосовувати онлайн-платформи для проведення лекцій, семінарів та курсів, що дозволяє здобувати освіту без фізичної присутності в класі.

– мережі як основа для Інтернету речей [2]: комп’ютерні мережі є основною складовою концепції Інтернету речей, де мільйони фізичних пристроїв, таких як датчики, камери, автомобілі та побутова техніка, підключені до Інтернету і можуть обмінюватися даними. Це дозволяє створювати «розумні» міста, «розумні» будинки та автоматизувати багато аспектів повсякденного життя.

1.2 Огляд питань вивчення захисту комп’ютерних мереж

Захист комп'ютерних мереж є важливою складовою забезпечення кібербезпеки, що охоплює не лише питання технологічної реалізації безпеки, але й методи забезпечення конфіденційності, цілісності та доступності даних. У цьому контексті вивчення захисту мереж передбачає глибоке розуміння основних мережевих моделей, протоколів безпеки та інструментів для захисту від загроз.

При вивченні базових навичок надаються знання про такі визначення [3]:

– мережеві пристрої: взаємозв’язок кількох пристроїв, також відомих як хости, які з’єднані кількома шляхами з метою надсилання/отримання даних або медіа. Комп’ютерні мережі також можуть включати кілька пристроїв/носіїв, які допомагають у спілкуванні між двома різними пристроями; вони відомі як мережеві пристрої та включають такі речі, як маршрутизатори, комутатори, концентратори та мости;

– топологія мережі: топологія мережі – це компонування різних пристроїв у мережі. Типи мережевих топологій: шина, зірка, кільце, сітчата-топологія; дерево та гібридна топологія;

– типи архітектури комп'ютерної мережі: комп'ютерна мережа підпадає під такі широкі категорії: клієнт-серверна архітектура – це тип архітектури комп'ютерної мережі, у якій вузли можуть бути серверами або клієнтами. Тут серверний вузол може керувати поведінкою клієнтського вузла; архітектура однорангового зв'язку – в архітектурі *P2P* (одноранговий зв'язок) немає поняття центрального сервера. Кожен пристрій безкоштовний для роботи як клієнта, так і сервера;

– модель OSI: є основою для розуміння функціонування комп'ютерних мереж. Вона складається з 7 рівнів: фізичний, канальний, мережевий, транспортний, сеансовий, представницький і прикладний. Модель OSI використовує абстракції для опису різних функцій в мережах;

– мережеві протоколи: протокол – це набір правил або алгоритмів, які визначають спосіб, яким два об'єкти можуть спілкуватися в мережі, і існує інший протокол, визначений на кожному рівні моделі *OSI*;

– маршрутизація: це процес вибору шляху, по якому дані можуть бути передані від джерела до пункту призначення. Маршрутизація виконується спеціальним пристроєм, відомим як маршрутизатор.

Забезпечення безпеки мережі має вирішальне значення для захисту даних і ресурсів від несанкціонованого доступу та атак. Основні аспекти безпеки мережі включають:

– міжмережеві екрани: пристрої або програмне забезпечення, які відстежують і контролюють вхідний і вихідний мережевий трафік на основі правил безпеки;

– шифрування даних, що передаються: процес кодування даних для запобігання несанкціонованому доступу. Зазвичай використовується в *VPN* , *HTTPS* і безпечній електронній пошті;

– системи виявлення вторгнень (*IDS*) [4]: інструменти, які відстежують мережевий трафік на предмет підозрілої активності та потенційних загроз;

– контроль доступу: механізми, які обмежують доступ до мережевих ресурсів на основі ідентичності та ролі користувача.

1.3 Огляд систем моделювання комп’ютерних мереж

1.3.1 *Cisco Packet Tracer* [5] – це потужний інструмент для моделювання комп'ютерних мереж, розроблений компанією Cisco. Він дозволяє створювати та тестувати мережеві структури в умовах віртуальної симуляції, що корисно для навчання та практики в області мережевих технологій.

Cisco Packet Tracer використовується в основному для навчання та вивчення основ комп'ютерних мереж. Це інструмент для студентів *Cisco Networking Academy*, який дозволяє моделювати та налаштовувати мережі, тестувати протоколи та налаштування віртуальних пристроїв. Його застосовують у курсах для вивчення таких тем, як налаштування маршрутизаторів, комутаторів, бездротових мереж, а також для тестування сценаріїв без потреби в реальному обладнанні.

Основні особливості:

– моделювання пристроїв: створення віртуальних мереж з різними пристроями, такими як маршрутизатори та комутатори;

– підтримка протоколів: підтримує різноманітні протоколи, зокрема *TCP, IP, OSPF, EIGRP*;

– моніторинг та візуалізація: Можливість візуалізації мережі та моніторингу передачі даних;

– безкоштовний доступ: Безкоштовно для студентів і викладачів через *Cisco Networking Academy*.

1.3.2 *ns-3 (Network Simulator 3*)

ns-3 [6] – це потужний симулятор для моделювання комп'ютерних мереж, який використовується для дослідження і розробки нових мережевих технологій.

ns-3 використовується для дослідження та розробки нових мережевих технологій у наукових і дослідницьких проектах. Це потужний симулятор, який дозволяє моделювати та симулювати як фізичні, так і мобільні мережі, що використовуються для тестування різних протоколів, стандартів (*LTE, 5G*), а також для розробки нових концепцій в галузі телекомунікацій і бездротових мереж.

Основні особливості:

– моделювання мережевих сценаріїв: *ns-3* дозволяє створювати та симулювати мережі з різними пристроями, такими як маршрутизатори, комп'ютери, сервери;

– підтримка широкого спектру протоколів: Програма підтримує протоколи *TCP, UDP, IP, OSPF, Wi-Fi, LTE* та інші для тестування реальних умов мереж;

– підтримка реальних сценаріїв: *ns-3* може інтегруватися з реальними мережами, що дозволяє проводити тестування на основі реальних даних;

– моделювання мобільних мереж: Зручний інтерфейс для моделювання мобільних мереж, таких як *5G та Wi-Fi*, для дослідження впливу мобільності на мережеву продуктивність;

– модульність та розширюваність: *ns-3* є модульним, що дозволяє користувачам створювати нові моделі і розширювати функціональність симулятора.

1.3.3 *GNS3 (Graphical Network Simulator-3)*

GNS3 [7] – це потужний графічний симулятор мереж, який дозволяє створювати, тестувати та вивчати складні мережеві топології.

GNS3 застосовується в освіті для навчання мережевих технологій, в сфері інформаційних технологій (IT) для тестування мережевих конфігурацій і оптимізації мереж, а також для віртуалізації мереж. Його використовують для підготовки до сертифікацій (*CCNA, CCNP*), розробки і тестування мережевих рішень та протоколів. Крім того, *GNS3* допомагає мережевим інженерам і адміністраторам створювати лабораторії та тестувати різні налаштування без фізичного обладнання.

Його основні особливості:

– графічний інтерфейс: *GNS3* надає користувачам зручний та інтуїтивно зрозумілий графічний інтерфейс для створення мережевих топологій, що дозволяє легко налаштовувати пристрої та зв'язки між ними;

– імітація реальних пристроїв: *GNS3* дозволяє використовувати реальні образи операційних систем для маршрутизаторів та комутаторів (наприклад, Cisco IOS), що забезпечує точне відтворення роботи пристроїв у віртуальному середовищі;

– моделювання складних мереж: програма підтримує побудову складних мереж з великою кількістю пристроїв, таких як маршрутизатори, комутатори, сервери та інші елементи мережі, з можливістю підключення до реальних мереж;

– підтримка протоколів: *GNS3* підтримує численні мережеві протоколи, зокрема *OSPF, BGP, EIGRP*, а також багато інших, що дозволяє створювати реалістичні мережеві сценарії для тестування різних конфігурацій;

– інтеграція з реальними пристроями та мережами: *GNS3* дозволяє інтегрувати віртуальні мережі з фізичними пристроями, що дає змогу проводити тестування в реальних умовах.

1.3.4 *OPNET (OPNET Modeler)*

OPNET Modeler [8] – це потужний програмний засіб для моделювання, аналізу та оптимізації комп'ютерних мереж і комунікаційних систем. *OPNET* використовується для моделювання, аналізу та оптимізації комп'ютерних мереж. Він дозволяє створювати детальні моделі мереж, тестувати та оцінювати протоколи, проектувати мережі та оцінювати їх ефективність. Цей інструмент застосовується в наукових дослідженнях, проектуванні інфраструктури, а також для оптимізації продуктивності мереж у великих компаніях та бізнес-середовищах. *OPNET* також використовується для підготовки до сертифікацій у галузі мережевих технологій.

Ось основні його особливості:

– моделювання мереж: дозволяє створювати складні моделі мереж, що включають різноманітні компоненти, такі як маршрутизатори, комутатори, хости та сервіси, з високою деталізацією;

– аналіз продуктивності мереж: програма забезпечує глибокий аналіз продуктивності мереж, зокрема часу затримки, пропускної здатності, втрат пакетів та інших ключових параметрів;

– моделювання різних мережевих технологій: підтримує широкий спектр мережевих технологій, таких як *Ethernet, Wi-Fi, LTE, IP*, а також можливість аналізу їх взаємодії в складних сценаріях;

– інтерфейс для розробки додатків: надає можливості для розробки та налаштування власних моделей мережевих пристроїв та протоколів за допомогою вбудованих засобів програмування.

– підтримка великих мережевих топологій: може працювати з великими та складними топологіями, що дає змогу виконувати масштабні симуляції для дослідження великих комунікаційних систем.

1.3.5 *QualNet*

QualNet [9] — це потужний симулятор для моделювання та аналізу продуктивності бездротових і дротових мереж. Він широко використовується для проектування, аналізу та оптимізації мережевих рішень у таких сферах, як військові комунікації, мобільні мережі, а також в академічних дослідженнях і промислових розробках. *QualNet* також застосовується для моделювання складних мережевих систем, тестування нових технологій і протоколів, а також для підготовки до сертифікацій і розробки інноваційних рішень у сфері мережевих технологій.

Ось основні особливості *QualNet*:

– моделювання високопродуктивних мереж: *QualNet* дозволяє моделювати як бездротові, так і дротові мережі, забезпечуючи високу точність моделювання з можливістю симулювання тисяч пристроїв у мережі;

– підтримка складних мережевих протоколів: підтримує широкий спектр мережевих протоколів, таких як *TCP, UDP, IP, OSPF, BGP*, а також специфічні протоколи для бездротових мереж, включаючи 802.11, *LTE, ZigBee*;

– висока масштабованість: Програма дозволяє моделювати великі мережі, включаючи численні вузли та мережеві компоненти, з мінімальними вимогами до ресурсів;

– аналіз продуктивності: забезпечує потужні інструменти для аналізу продуктивності мережі, включаючи час затримки, пропускну здатність, втрату пакетів, що дозволяє оцінити ефективність мережевих топологій;

– інтерфейс для розробки власних моделей: дозволяє користувачам створювати та налаштовувати власні моделі мережевих протоколів і пристроїв, що дає можливість адаптувати симуляції до конкретних вимог.

1.3.6 NetSim

NetSim [10] – це потужний симулятор мереж, який дозволяє створювати, моделювати та тестувати різні типи комп'ютерних мереж. Він використовується для навчання та підготовки до сертифікацій (наприклад, *CCNA, CCNP*), а також для тестування та аналізу мережевих рішень у реальних умовах. *NetSim* застосовується в академічних установах для навчання студентів мережевим технологіям, в *IT*-компаніях для оптимізації мереж і вирішення проблем, а також у дослідженнях нових мережевих технологій і протоколів.

Ось основні його особливості:

– моделювання різноманітних мережевих топологій: дозволяє моделювати різні типи мереж, включаючи дротові та бездротові, а також гібридні топології;

– підтримка реальних протоколів: Програма підтримує численні мережеві протоколи, такі як *IP, TCP, OSPF, EIGRP, BGP*, а також бездротові технології, включаючи *Wi-Fi* та *LTE*;

– інтерактивне навчання: пропонує інтерактивні завдання і лабораторії для навчання мережевим технологіям, що робить його корисним для студентів і професіоналів у сфері мереж;

– аналіз продуктивності мережі: дозволяє проводити глибокий аналіз продуктивності мереж, зокрема моніторинг затримок, пропускної здатності, втрат пакетів та інших важливих параметрів;

– підтримка великої кількості пристроїв: підтримує моделювання великих мереж з численними пристроями, такими як маршрутизатори, комутатори, хости та інші компоненти.

1.3.7 *Mininet*

Mininet [11] – це потужний симулятор мереж, який дозволяє створювати віртуальні мережі та тестувати мережеві протоколи в реальному часі. Він використовується для тестування та розробки *SDN*-протоколів, а також для дослідження нових концепцій у мережах, таких як програмування мережевих пристроїв через *OpenFlow*. *Mininet* є популярним серед дослідників і студентів для навчання та тестування концепцій програмно керованих мереж і вивчення мережевих протоколів. Крім того, він застосовується для тестування і оптимізації мережевих рішень в реальних умовах без необхідності у фізичному обладнанні.

Основні особливості *Mininet*:

– моделювання віртуальних мереж: *Mininet* дозволяє створювати віртуальні мережі з різними пристроями, такими як хости, комутатори, маршрутизатори, що дає змогу створювати реалістичні мережеві топології;

– підтримка *SDN (Software-Defined Networking*): *Mininet* особливо корисний для тестування мережевих технологій, що базуються на *SDN*, таких як *OpenFlow*, завдяки можливості інтеграції з контролерами *SDN*;

– моделювання різних протоколів: *Mininet* підтримує моделювання різних мережевих протоколів, таких як *IP, TCP, UDP, OpenFlow*, що дозволяє проводити тести за різних умов;

– інтерактивний інтерфейс командного рядка: *Mininet* забезпечує зручний командний інтерфейс для управління віртуальними пристроями та мережами, що дозволяє користувачам створювати, налаштовувати та тестувати мережі в реальному часі;

– підтримка масштабованості: *Mininet* дозволяє моделювати масштабовані мережі з великою кількістю пристроїв, що робить його корисним для дослідження великих комунікаційних систем.

1.3.8 *Cisco VIRL (Virtual Internet Routing Lab)*

Cisco VIRL [12] – це потужний симулятор для віртуалізації мережевих пристроїв та розробки мережевих топологій, який використовується для тестування та навчання.

VIRL використовується для навчання мережевих технологій, підготовки до сертифікацій *Cisco* (наприклад, *CCNA, CCNP*), тестування нових мережевих рішень і протоколів. Цей інструмент широко застосовується в навчальних закладах, корпоративних середовищах та для досліджень, де необхідно тестувати складні мережеві топології та конфігурації без використання фізичного обладнання.

Ось основні особливості *Cisco VIRL*:

– моделювання реальних мережевих пристроїв: дозволяє використовувати віртуальні образи реальних пристроїв *Cisco*, таких як маршрутизатори, комутатори, брандмауери, що забезпечує високу точність моделювання;

– підтримка різних протоколів і технологій: підтримує безліч мережевих протоколів і технологій, таких як *OSPF, EIGRP, BGP*, а також *SDN* і *VXLAN* для більш складних сценаріїв;

– інтерфейс для створення мережевих топологій: пропонує зручний графічний інтерфейс для побудови і налаштування складних мережевих топологій, що дозволяє легко інтегрувати різні пристрої в одну модель;

– віртуалізація та масштабованість: програма підтримує віртуалізацію пристроїв, що дозволяє моделювати великі мережі з багатьма пристроями без необхідності в фізичному обладнанні;

– інтеграція з реальними мережами: дозволяє інтегрувати віртуальні мережі з реальними мережами для тестування і оптимізації мережевих сценаріїв.

1.3.9 *EVE-NG Community Edition*

EVE-NG (Emulated Virtual Environment Next Generation) [13] – це потужний інструмент для віртуалізації мереж і моделювання складних топологій, який широко використовується для навчання, тестування та досліджень. *EVE-NG* використовуються для навчання, підготовки до сертифікацій, тестування нових протоколів і конфігурацій, а також для досліджень у галузі мережевих технологій. Цей інструмент популярний серед студентів, професіоналів та мережевих інженерів, оскільки дозволяє створювати реалістичні лабораторії та моделювати великі мережі без потреби у фізичному обладнанні.

Ось основні особливості *EVE-NG Community Edition:*

– моделювання різних типів пристроїв: дозволяє моделювати різноманітні мережеві пристрої, включаючи маршрутизатори, комутатори, брандмауери, сервери та інші компоненти з можливістю використання образів реальних пристроїв, таких як *Cisco, Juniper, Fortinet;*

– підтримка різноманітних протоколів і технологій: підтримує численні мережеві протоколи і технології, включаючи *IP, OSPF, BGP, VXLAN, MPLS*, а також дозволяє моделювати бездротові мережі;

– інтуїтивно зрозумілий графічний інтерфейс: має зручний і доступний графічний інтерфейс, що дозволяє користувачам легко створювати і налаштовувати складні мережеві топології;

– масштабованість і продуктивність: Завдяки можливості використання потужних серверів і віртуалізації *EVE-NG* може моделювати великі та складні мережеві топології з великою кількістю пристроїв, що робить його ідеальним для великих симуляцій;

– підтримка інтеграції з реальними пристроями: дозволяє інтегрувати віртуальні мережі з фізичними пристроями, що дає змогу проводити тестування в реальних умовах;

– підтримує велику кількість мережевих пристроїв від різних виробників, таких як *Cisco, Juniper, Arista*, а також дозволяє інтегрувати інші технології, зокрема *SDN* та *IoT*.

1.3.10 *PNETLab*

PNETLab [14] – це потужний симулятор мереж, який дозволяє створювати та тестувати різноманітні мережеві топології за допомогою віртуальних пристроїв. *PNETLab* використовується для навчання мережевим технологіям, підготовки до сертифікацій (наприклад, *CCNA, CCNP*), тестування нових протоколів і конфігурацій, а також для досліджень в області мережевих рішень. Платформа популярна серед студентів, інженерів і дослідників завдяки можливості створювати віртуальні лабораторії без необхідності в фізичному обладнанні.

Ось основні особливості *PNETLab*:

– моделювання складних мережевих топологій: *PNETLab* дозволяє створювати різні мережеві топології, використовуючи віртуальні образи пристроїв, таких як маршрутизатори, комутатори, брандмауери, сервери, що дає змогу моделювати реалістичні мережеві середовища;

– підтримка різноманітних мережевих протоколів: підтримує такі мережеві протоколи, як *OSPF, BGP, RIP*, а також бездротові технології та інші специфічні протоколи для моделювання реальних мереж;

– інтерфейс командного рядка та графічний інтерфейс: пропонує як графічний інтерфейс, так і можливості управління через командний рядок, що дає користувачам гнучкість у налаштуванні та управлінні мережами;

– масштабованість та інтеграція: дозволяє моделювати великі мережі з кількома пристроями та підтримує інтеграцію з фізичними пристроями, що дає можливість тестувати реальні мережеві конфігурації;

– підтримка віртуалізації: використовує віртуалізацію для створення мережевих середовищ, що дозволяє значно знижувати вартість обладнання та спрощує налаштування мереж;

– програма підтримує різні бренди, такі як *Cisco, Juniper*, а також включає підтримку для програмно визначених мереж (*SDN*) та інших технологій.

РОЗДІЛ 2. РОЗГОРТАННЯ *PNETLAB*

2.1 Загальна характеристика *PNETLab Box*

PNETLab Box є інтегрованим апаратно-програмним засобом, призначеним для створення віртуального навчального або тестового середовища з метою імітації та моделювання мережевої інфраструктури. Він поєднує в собі можливості відкритого емулятора мережі *PNETLab* з компактною апаратною платформою, що забезпечує автономність, мобільність і зручність використання у навчальних, лабораторних, сертифікаційних і дослідницьких процесах [15].

PNETLab – це розширена версія емуляційної системи *EVE-NG* з відкритим кодом, яка дозволяє створювати складні мережеві топології за допомогою візуального веб-інтерфейсу. *PNETLab Box*, відповідно, є готовим пристроєм з попередньо інстальованою та налаштованою системою, що дозволяє користувачам уникнути складностей встановлення та конфігурування середовища [16].

2.1.1 Вимоги до апаратного середовища

Апаратна частина *PNETLab Box* зазвичай реалізована на основі енергоефективного міні-комп’ютера або тонкого клієнта (наприклад, *Intel NUC, HP T640, Dell Wyse*), який має достатні ресурси для запуску віртуалізованих мережевих пристроїв. Типова конфігурація включає багатоядерний процесор із підтримкою технологій апаратної віртуалізації (*Intel VT-x* або *AMD-V*), об’єм оперативної пам’яті від 8 до 64 ГБ, твердотілий накопичувач (*SSD*) обсягом від 256 ГБ, а також один або кілька гігабітних мережевих інтерфейсів [17].

Наявність продуктивного *SSD*-диска та достатньої кількості оперативної пам’яті критично важлива для одночасного запуску кількох віртуальних машин, особливо якщо використовуються повнофункціональні образи операційних систем (наприклад, *Cisco IOSv, FortiGate, pfSense* тощо). *PNETLab Box* також часто комплектується системою пасивного охолодження, що дозволяє безшумну роботу у навчальному класі або лабораторії.

2.1.2 Програмне середовище

Програмна частина *PNETLab Box* заснована на операційній системі *Ubuntu Server* або *Debian* та включає необхідне програмне забезпечення для емуляції та віртуалізації. Основу емуляційної частини становить *QEMU* (для віртуалізації образів мережевих пристроїв), *KVM* (для використання апаратної віртуалізації), а також *Docker*, що забезпечує запуск легких контейнеризованих сервісів [18].

Інтерфейс користувача представлений у вигляді веб-застосунку, який дозволяє створювати топології шляхом перетягування віртуальних пристроїв на робоче поле, з'єднувати їх між собою, а також конфігурувати параметри запуску та взаємодії. Для з'єднання з пристроями використовуються стандартні протоколи *Telnet, SSH, VNC*, що робить середовище універсальним та наближеним до реальних умов [16; 18].

Користувач може імпортувати власні образи (наприклад, *Cisco IOL, ASA, Juniper vSRX, MikroTik CHR*), зберігати проєкти, копіювати їх, інтегрувати із зовнішніми програмами, такими як *Wireshark* (для перехоплення трафіку), *TFTP/FTP*-сервери тощо. Таким чином, *PNETLab Box* є багатофункціональним рішенням, що дозволяє створювати стенди будь-якої складності.

2.1.3 Галузі застосування

PNETLab Box використовується у таких сферах:

– навчальний процес: для викладання комп'ютерних мереж, мережевої безпеки, системного адміністрування, кіберзахисту. Пристрій дозволяє реалізовувати лабораторні роботи без потреби в реальному обладнанні;

– підготовка до сертифікацій: *CCNA, CCNP, Fortinet NSE, CompTIA Security+*, тощо. *PNETLab* дозволяє відтворювати типові практичні завдання з іспитів;

– тестування мережевих рішень: перед розгортанням інфраструктури в продуктивному середовищі мережеві адміністратори можуть змоделювати поведінку протоколів, маршрутизації, *NAT, VPN, VLAN*, тощо;

– кіберзахист і аудит безпеки: створення тестових полігонів для моделювання атак, перевірки правил firewall, *IDS/IPS, honeypot*-структур, тощо [19].

Серед основних переваг *PNETLab Box* можна виділити:

– компактність та автономність – не потребує інсталяції на ПК користувача;

– універсальність – підтримка десятків типів мережевих операційних систем;

– гнучкість і масштабованість – можливість розширення ресурсів або використання в хмарному режимі;

– висока швидкодія – завдяки оптимізації на рівні ОС та гіпервізора;

– можливість роботи без інтернет-з'єднання, що є критичним для безпечного тестування [18; 20].

Таким чином, *PNETLab Box* — це сучасне, ефективне й доступне рішення для побудови віртуальних мережевих лабораторій, яке поєднує гнучкість програмного забезпечення з надійністю і портативністю апаратного забезпечення.

2.2 Встановлення *PNETLab*

У межах цієї роботи було обрано встановити *PNETLab* на сервер кафедри на платформі Proxmox.

Процес встановлення симулятору включає декілька етапів що описані нижче.

1) Зміна формату віртуального образу *PNETLab*. Формат образу .ova не підходить для встановлення на платформи віртуалізації, які засновані на *Linux*, до яких відноситься *Proxmox*, тож формат файлу було змінено за допомогою консолі на формат *vmdk*, потім у формат *qcow2* (рідний формат *Proxmox*).

Процес зміни формату образу диска *PNETLab*:

```
login as: root
root@192.168.131.254's password:
Linux pve 5.4.114-1-pve #1 SMP PVE 5.4.114-1 (Sun, 09 May 2021 17:13:05 +0200)
x                                86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jun  9 11:13:19 2025 from 192.168.131.104

root@pve:~# tar xvf PNET_4.2.10.ova
PNET_4.2.10.ovf
PNET_4.2.10.mf
PNET_4.2.10-disk1.vmdk
root@pve:~# qemu-img convert -f vmdk PNET_4.2.10-disk1.vmdk -O qcow2
PNET_4.2.10                              -disk1.qcow2

root@pve:~# qm importdisk 300 PNET_4.2.10-disk1.qcow2 local-lvm
importing disk 'PNET_4.2.10-disk1.qcow2' to VM 300 ...
  Logical volume "vm-300-disk-0" created.
transferred 0.0 B of 100.0 GiB (0.00%)
...
transferred 100.0 GiB of 100.0 GiB (100.00%)
Successfully imported disk as 'unused0:local-lvm:vm-300-disk-0'
```

2) Запуск веб-інтерфесу *Proxmox* та вхід в систему.

3) Додавання у віртуальну машину *PNETLab* з форматом *qcow2*.

4) Налаштування віртуальної машини:

– основні параметри (*General*) представлені в таблиці 2.1 (рис. 2.1).

Таблиця 2.1 – Основні параметри віртуальної машини

Поле	Значення	Опис
Node	*pve*	назва вузла *Proxmox*, на якому буде створена віртуальна машина
VM ID	*300*	унікальний ідентифікатор віртуальної машини в межах цього вузла
Name	*PNETLAB2025*	назва віртуальної машини
Resource Pool	-	пул ресурсів вузла

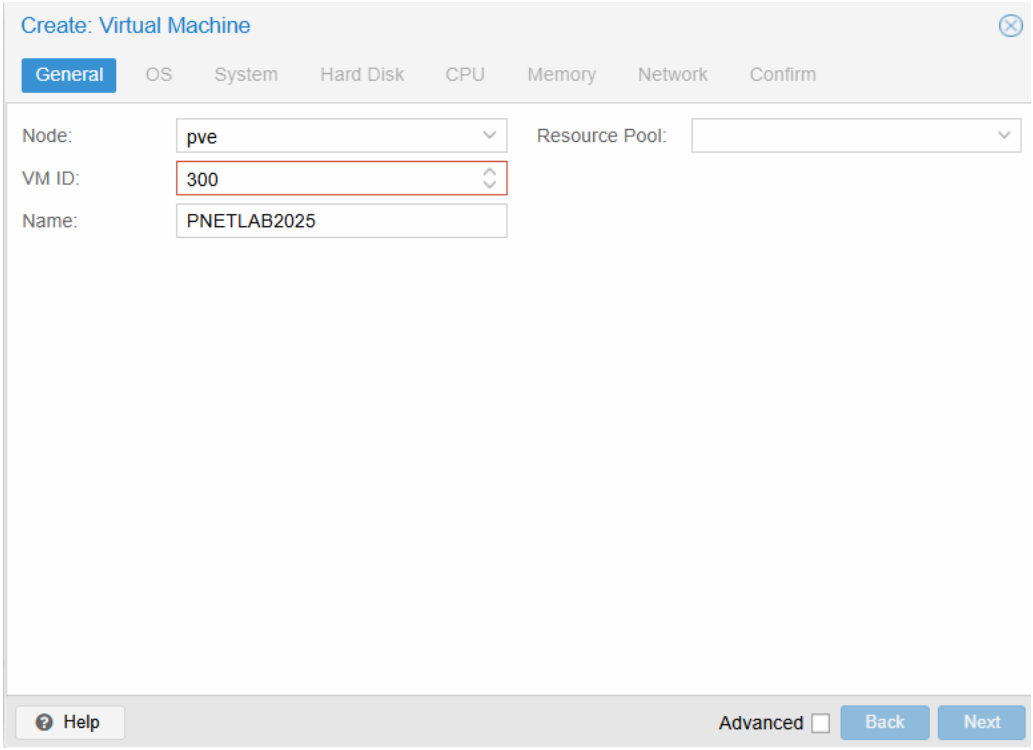


Рисунок 2.1 – Основні налаштування.

– налаштування операційної системи (*OS*) зведено до таблиці 2.2 (рис. 2.2).

Таблиця 2.2 – Параметри операційної системи

Поле	Значення	Опис
Media Source	*Do not use any media*	Жоден *ISO* або фізичний диск не буде використаний
Guest OS Type	*Linux*	Віртуальна машина буде працювати з ОС *Linux*.
Guest OS Version	*5.x - 2.6 Kernel*	Орієнтація на ядро *Linux 2.6–5.x* — оптимізує емуляцію для Linux.

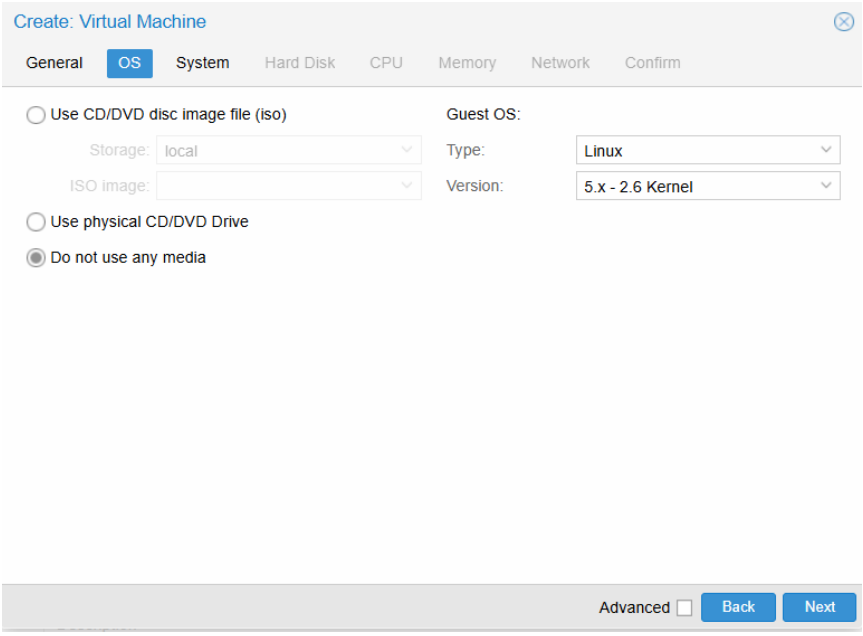


Рисунок 2.2 – Налаштування операційної системи.

– параметри системи (*System*) представлені в таблиці 2.3 (рис. 2.3);

Таблиця 2.3 – Налаштування системи

Параметр	Значення	Пояснення
Graphic card	*Default*	Використовується стандартний відеоадаптер (зазвичай *std* або *vmware*).
SCSI Controller	*VirtIO SCSI*	Це високопродуктивний віртуальний контролер дисків. Рекомендовано для *Linux*-систем.

Продовження таблиці 2.3

Qemu Agent	–	Це агент, який встановлюється в гостьову ОС для покращення інтеграції (можливість отримання *IP*, контроль живлення, тощо). Для *PNETLab* не обов'язково, але рекомендується активувати, якщо ви плануєте встановити *QEMU Guest Agent* у системі.

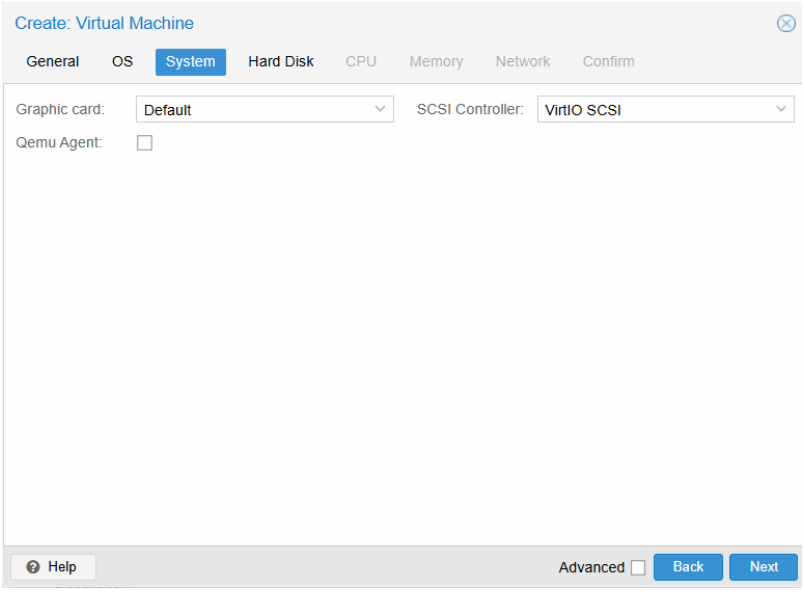


Рисунок 2.3 – Системні налаштування.

– налаштування жорсткого диску (рис. 2.4) (*Hard disk*) представлені в таблиці 2.4.

Таблиця 2.4 – Налаштування жорсткого диску

Параметр	Значення	Пояснення
Bus/Device	*SCSI 0*	Тип інтерфейсу та порядковий номер пристрою. Вказує, що диск буде підключено як *SCSI*-пристрій на позиції 0.
SCSI Controller	*VirtIO SCSI*	Вказує на тип контролера для *SCSI*. *VirtIO SCSI* забезпечує високу продуктивність

Продовження таблиці 2.4

Storage	*local-lvm*	Вказує, що диск буде збережений у *LVM*-томі, який розміщено на локальному сховищі хоста *Proxmox*.
Disk size (GiB)	32	Розмір віртуального диска. 32 гігабайти мінімум для *EVE-NG*.
Cache	*Default*	Кешування вимкнене, тобто диск працює напряму, що зазвичай безпечніше для цілісності даних.
Discard	–	Ввімкнення – дозволяє гостьовій системі повідомляти про звільнені блоки (*TRIM*).

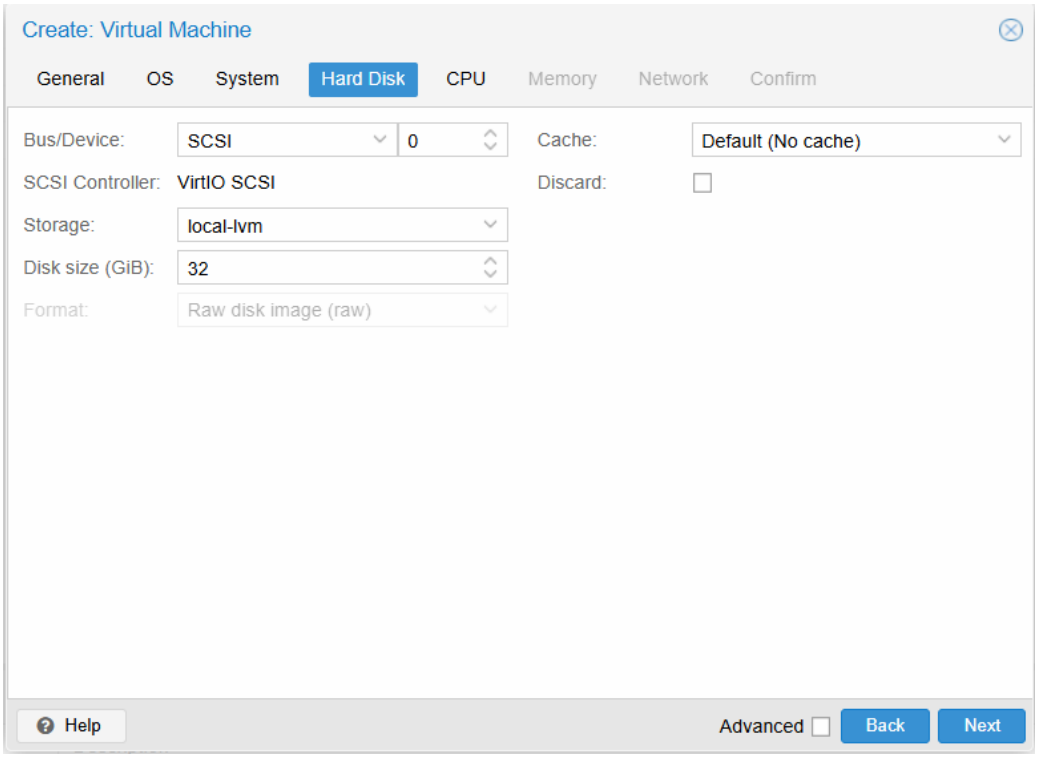


Рисунок 2.4 – Налаштування жорсткого диску.

– налаштування процесору (*CPU*) приведено в таблиці 2.5 (рис. 2.5);

Таблиця 2.5 – Налаштування процесору

Параметр	Значення	Пояснення
Sockets	1	Кількість *CPU*-сокетів, які будуть емулявані для ВМ. Один сокет зазвичай достатній для більшості сценаріїв.
Cores	8	Кількість ядер у межах одного сокета. У цьому випадку ВМ отримає 8 ядер.
Type	*host*	ВМ отримає повний доступ до інструкцій хостового процесора (без емуляції). Це оптимальний варіант для продуктивності, особливо для *EVE-NG*, який інтенсивно використовує *CPU* під час запуску емуляцій.

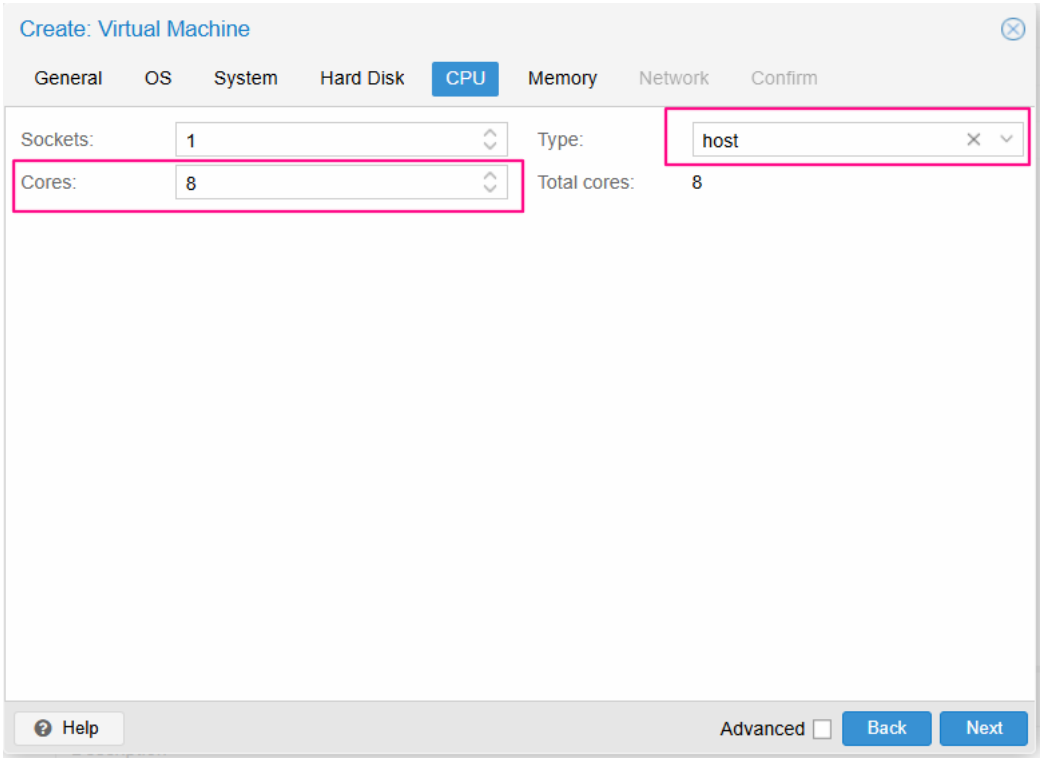


Рисунок 2.5 – Налаштування процесору.

– при налаштуванні об’єму оперативної пам’яті (*Memory*) буде виділено для віртуальної машини — 8192 Міб або 8 ГіБ (гібібайт). (рис.2.6);

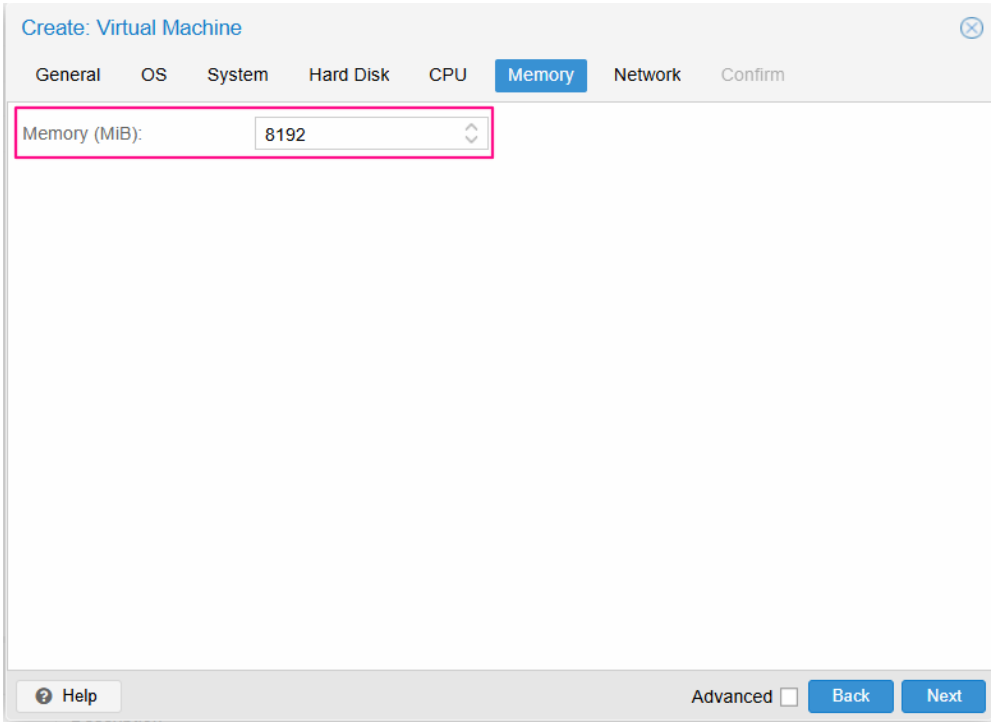


Рисунок 2.6 – Налаштування пам’яті

– параметри мережі ВМ (*Network*) представлено в табл. 2.6 (рис.2.7);

Таблиця 2.6 – Налаштування мережі

Параметр	Значення	Пояснення
Bridge	*vmbr0*	Це основний мережевий міст *Proxmox*, який зазвичай підключений до фізичної мережі хоста. Використовується для надання віртуальній машині доступу до локальної мережі або Інтернету.
Model	*VirtIO (paravirtualized)*	Це найшвидший і найефективніший тип мережевого адаптера для віртуальних машин. Для правильного функціонування *VirtIO* може знадобитися встановлення драйверів всередині ОС гостя (в залежності від ОС).

Продовження таблиці 2.6

VLAN Tag	*no VLAN*	*VLAN* не використовується. Якщо б ВМ мала працювати в ізольованому сегменті мережі — тут вказувався б номер *VLAN*.
MAC address	*auto*	Автоматично генерується *MAC*-адреса для мережевого інтерфейсу.
Firewall	позначено (увімкнено)	Proxmox має власний вбудований фаєрвол, і ця галочка означає, що він буде застосований для цієї віртуальної машини (якщо він активований у налаштуваннях кластера або вузла).

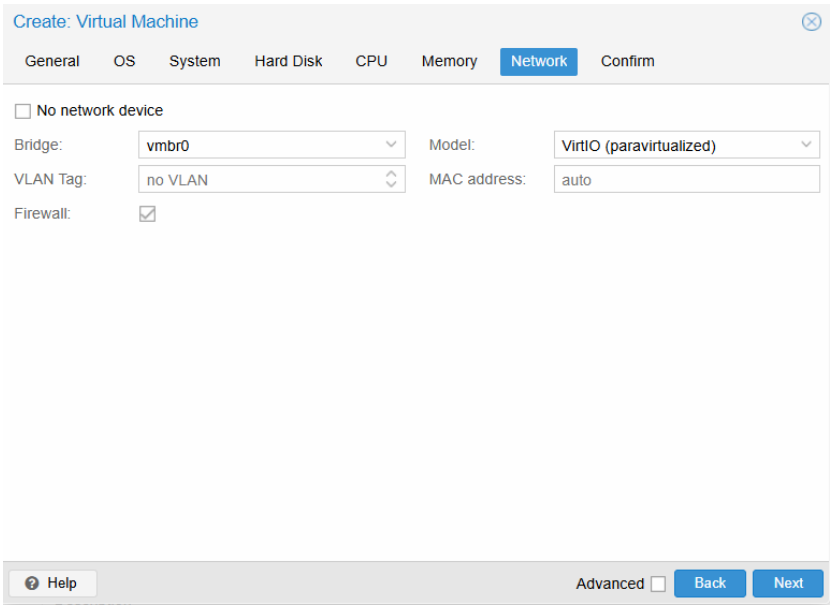


Рисунок 2.7 – Налаштування мережі.

5) Після запуску віртуальної машини відбувається автоматичне завантаження внутрішньої ОС. Після повного запуску на екрані буде відображено *IP*-адресу, за якою доступний веб-інтерфейс (наприклад, 192.168.131.119 (це залежить від того до якої інтернет мережі ви підключені, у цьому випадку до статичної *IP*-адреси)) (рис. 2.8).

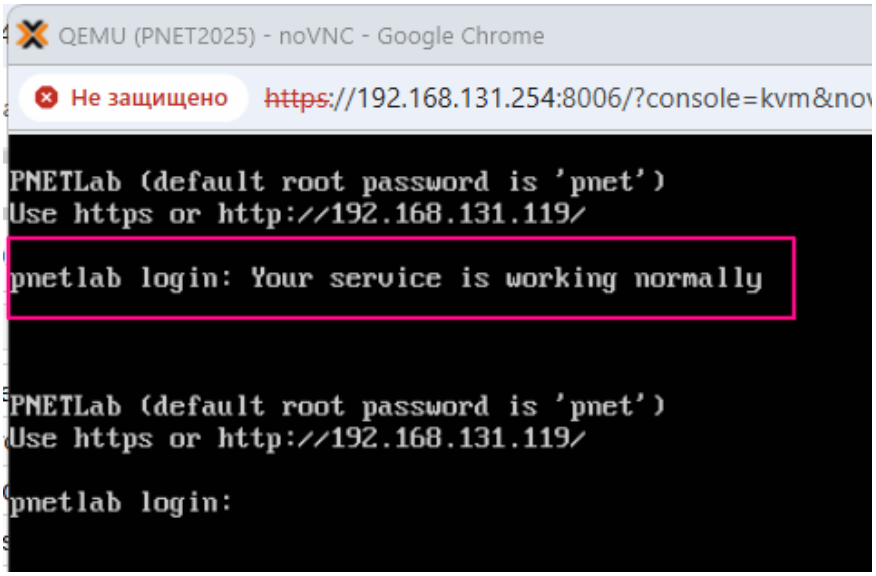


Рисунок 2.8 – Вікно завантаження *PNETLab.*

5) Для того щоб система працювала коректно у консолі віртуальної машини *PNETLab* вводимо функцію *root* для більш точного налаштування та підтрерджуємо пароль (*pnet*).

6) Подальші налаштування системи:

– поле *DNS domain name* залишаємо без змін. У *полі "Use DHCP or Static IP Address for the network adapter on Management Network?"* вибираємо за замовчуванням dhcp, щоб IP-адреса була автоматично призначена від *DHCP*-сервера у мережі (наприклад, від роутера);

– поле *NTP server* залишаємо пустим. *Choose how the VM can connect to the Internet* (обираємо, як ця віртуальна машина буде підключатися до Інтернету), обираємо direct connection (пряме з'єднання з Інтернетом без проксі-серверів). Обираємо цей варіант, якщо у нас звичайне підключення до Інтернету, наприклад, через домашній роутер або офісну мережу без проксі;

7) Після налаштування машини за допомогою *root* прав робота з *PNETLab* доступна через веб-інтерфейс за адресою вказаною в п.4. За замовчуванням ім'я користувача *admin* та пароль *pnet*.

У вікні, що з’явилося після введення *IP*-адреси вибираємо *Offline Mode* (рис 2.9), вводимо дані необхідні для входу (рис. 2.10) і потрапляємо в базовий веб-інтрерфейс програми *PNETLab* (рис. 2.11).

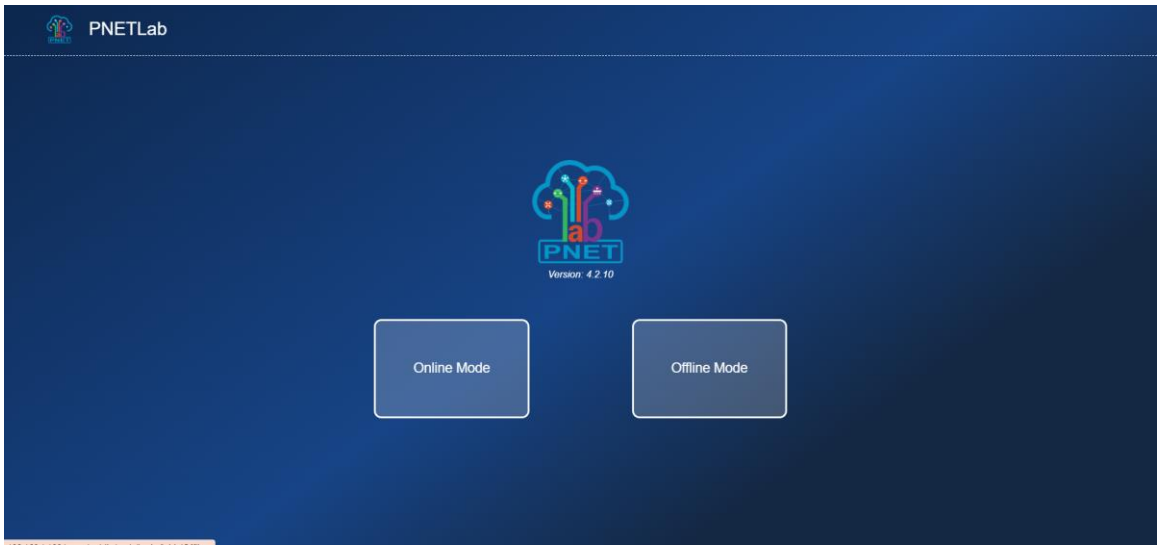


Рисунок 2.9 – Вибір режиму роботи у веб-інтерфейсі.

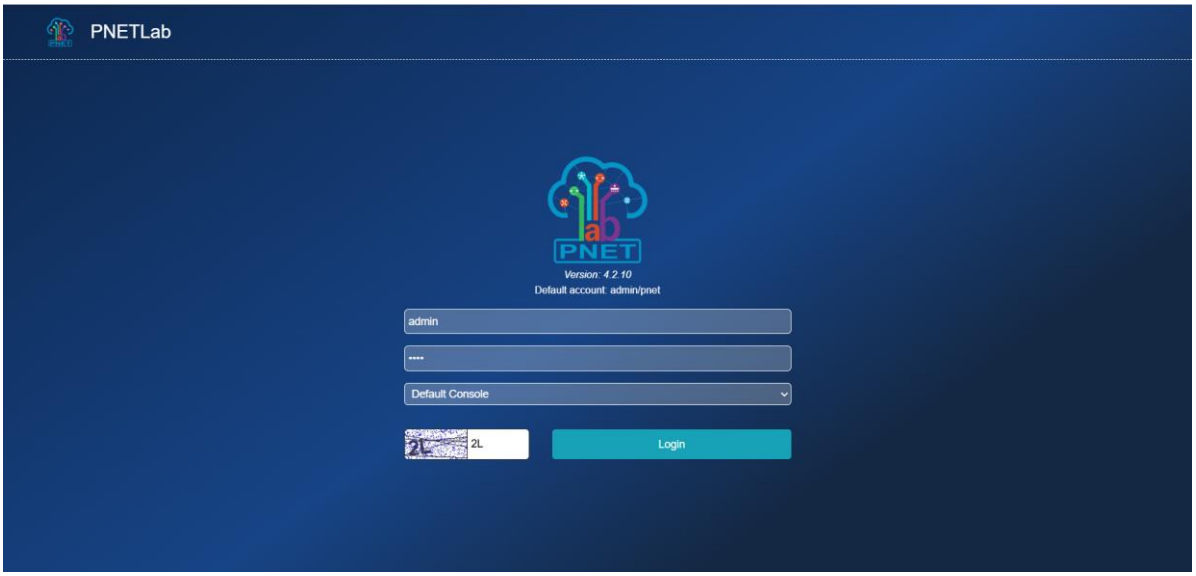


Рисунок 2.10 – Дані для входу у веб-інтерфейс.

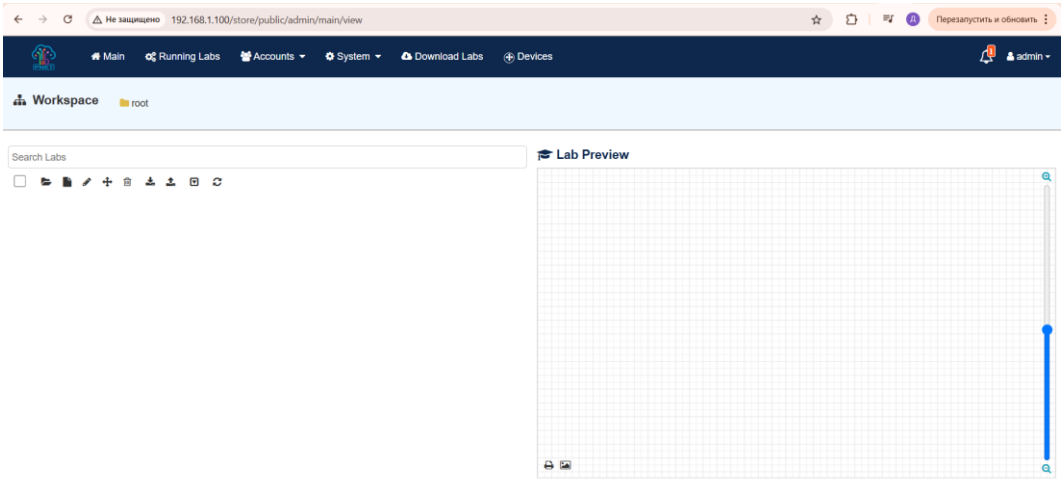
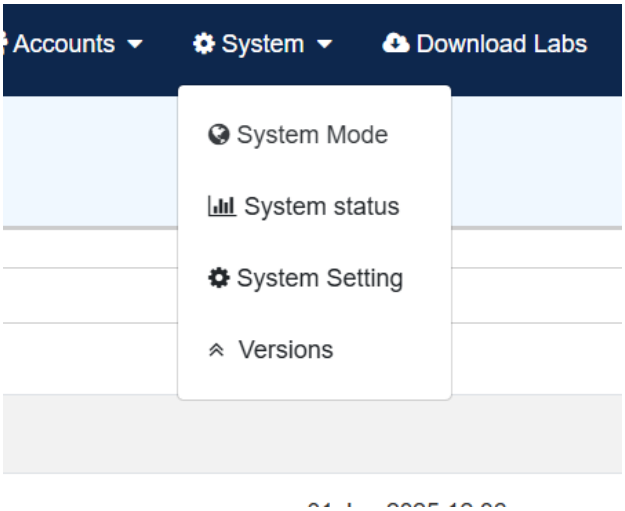
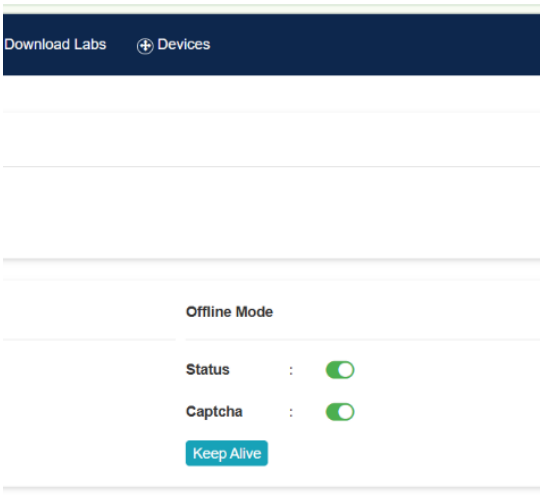


Рисунок 2.11 – базовий веб-інтерфейс *PNETLab*.

Для правильного налаштування роботи *PNETLab* виконуються такі налаштування *offline* доступу (рис. 2.12 а,б).



а)



б)

Рисунок 2.12 - Налаштування offline режиму *PNETLab*

2.3 Огляд інструментів моніторингу системи

Необхідність моніторингу в *PNETLab:*

Під час роботи з *PNETLab* можуть запускатися десятки віртуальних машин, кожна з яких споживає ресурси центрального процесора (*CPU*), оперативної пам'яті (*RAM*), мережі (*NIC*) та підсистеми зберігання даних (диск). Неналежний моніторинг цих компонентів може призвести до зниження продуктивності, збоїв у роботі емуляції, а також до втрати даних або порушення логіки лабораторних досліджень.

Моніторинг дозволяє:

– вчасно виявити навантаження;

– оптимізувати використання ресурсів;

– забезпечити стабільність платформи;

– аналізувати продуктивність пристроїв у середовищі емуляції.

Керування оперативною пам'яттю, процесором, жорстким диском. Починаючи з версії 2.0.8 , *PNETLab* дозволяє керувати процесором, оперативною пам'яттю та жорстким диском на кожному вузлі, у кожному сеансі лабораторії та для кожного користувача. Можливо обмежити обсяг оперативної пам'яті, процесора та жорсткого диска для кожного користувача (рис. 2.13).

У робочій області лабораторії можливо відстежувати інформацію про всі вузли, натиснувши вкладку «Стан системи» . Таблиця статусу автоматично оновлюватиметься через 30 секунд.

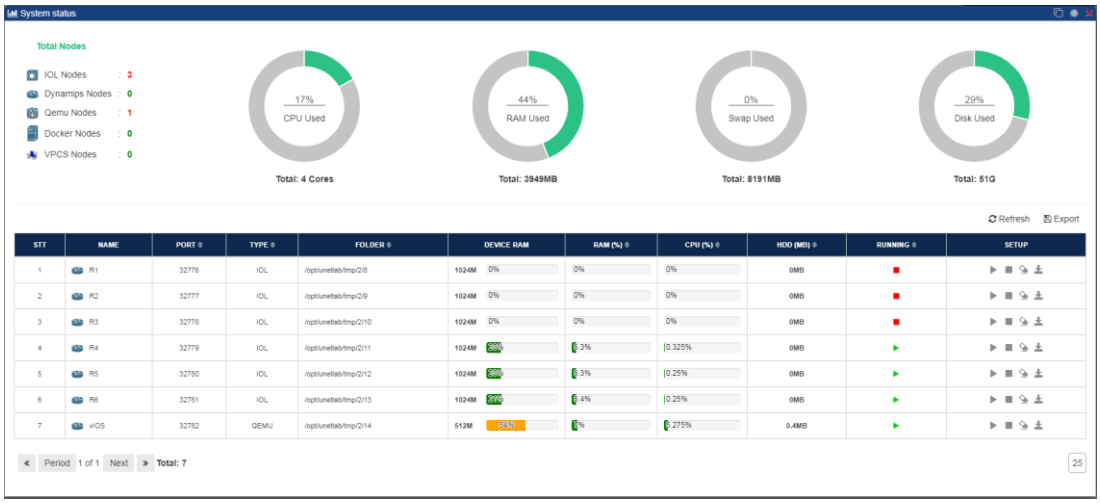


Рисунок 2.13 – Відстеження використання ресурсів від різних компонентів та віртуальних машин *PNETLab*.

Щоб контролювати оперативну пам'ять, процесор, жорсткий диск для кожного користувача, необхідно перейти до розділу Облікові записи > Менеджер користувачів (рис. 2.14).

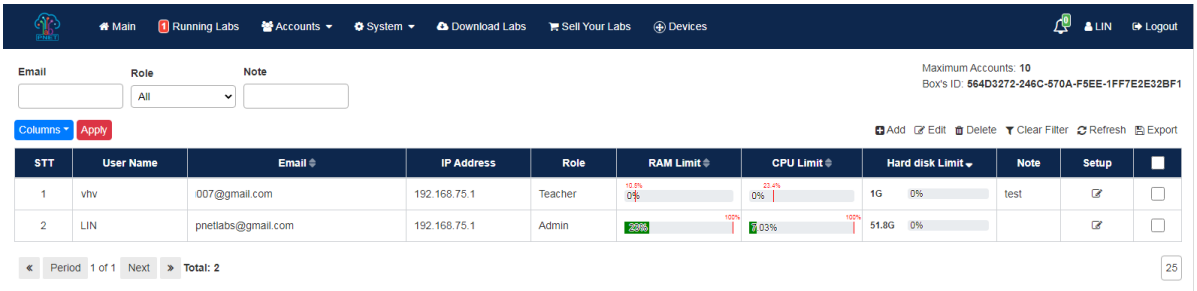


Рисунок 2.14 – Контроль оперативної пам'яті,
процесора, жорсткого диску для кожного користувача.

Можно побачити, скільки відсотків оперативної пам'яті та процесора використовують користувачі (рис. 2.15).

Червоний поріг – це обмеження оперативної пам’яті та процесора. Щоб налаштувати його, необхідно перейти до розділу Облікові записи > Менеджер ролей.

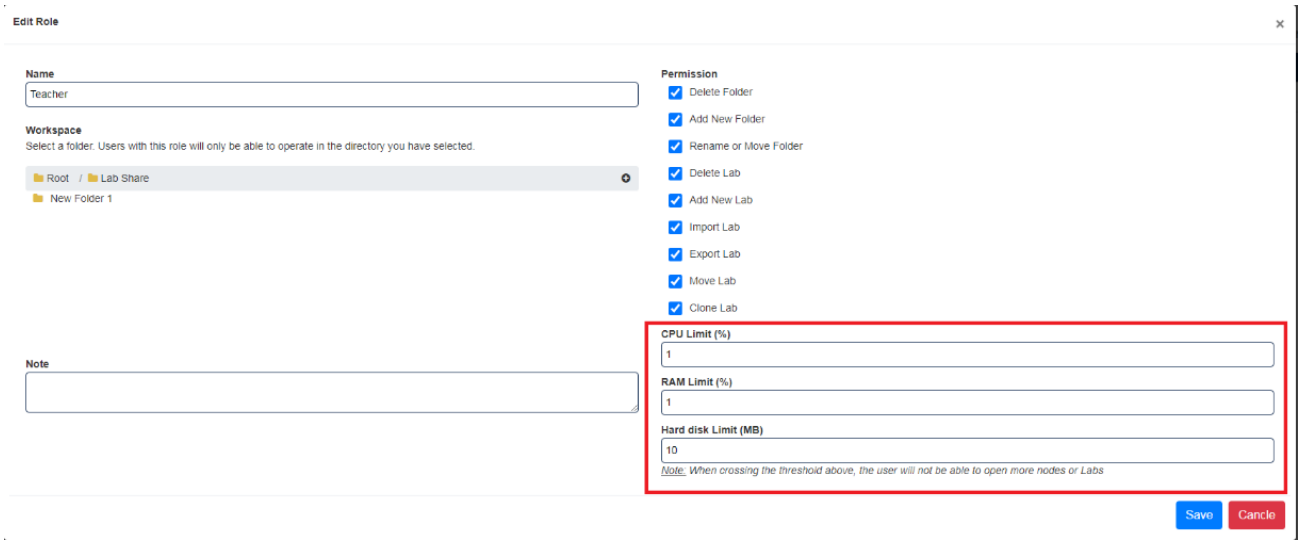


Рисунок 2.15 – Інформація про використання ресурсів користувачами ВМ.

Щоб керувати ресурсами для кожної лабораторії, переходимо до розділу «Запущені лабораторії». Можна розгорнути таблицю, натиснувши кнопку «  » (рис. 2.16).

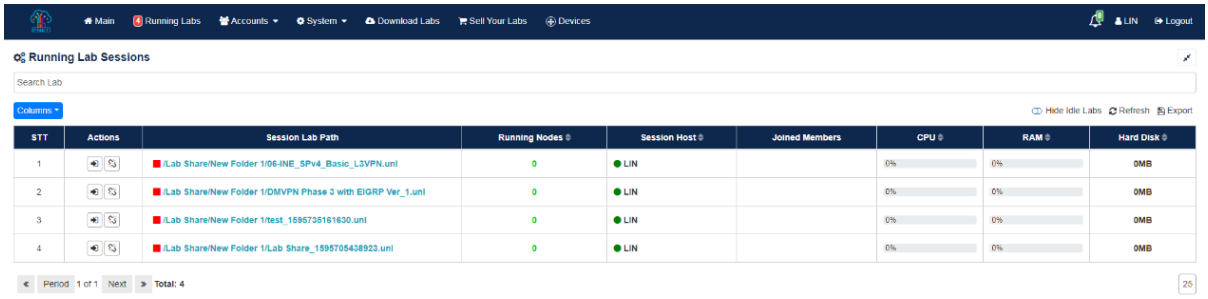


Рисунок 2.16 – Таблиця споживання ресурсів.

Додатковий моніторинг через командний рядок (термінал *Linux*). *PNETLab* зазвичай встановлюється на *Ubuntu/Debian*. Корисні команди представлені в таблиці 2.7.

Таблиця 2.7 – Команди моніторингу в Linux

№	Команда	Призначення
1.	*htop*	Візуальний моніторинг процесора і пам’яті.
2.	*top*	Системний монітор в реальному часі.
3.	*vmstat 1*	Статистика ресурсів (*CPU, RAM, IO*).
4.	*iostat*	Моніторинг навантаження на диск.
5.	*ifstat, iftop*	Статистика мережевого трафіку.
6.	*nmon*	Комплексний моніторинг системи.

Ці утиліти дозволяють отримати детальну інформацію про стан системи в реальному часі та виявити потенційні проблеми у використанні ресурсів.

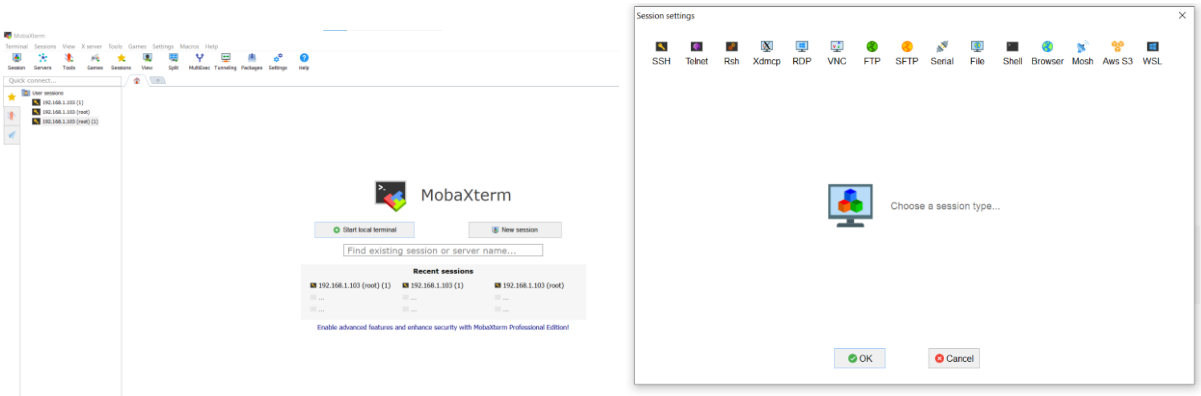
РОЗДІЛ 3. НАЛАШТУВАННЯ *PNETLAB*

3.1 Налаштування системи та додавання образів

Щоб додати віртуальні образи пристроїв знадобиться встановити додатково програму *MobaXterm* з офіційного сайту (рис. 3.1):

Рисунок 3.1 – Вствновлення *MobaXterm Installer Edition.*

Після втсановлення программи необхідно створити сессію з віртуальною машиною *PNETLab* > *Session* > *SSH* (рис.3.2).



а) б)

Рисунок 3.2 – Створення віртуальної сессії.

У вікні Remote host вводимо *IP*-адресу нашої віртуальної машини *PNETLab* з пункту 4 кваліфікаційної роботи. *Specifity username – root* та натискаємо *OK* (рис. 3.3).

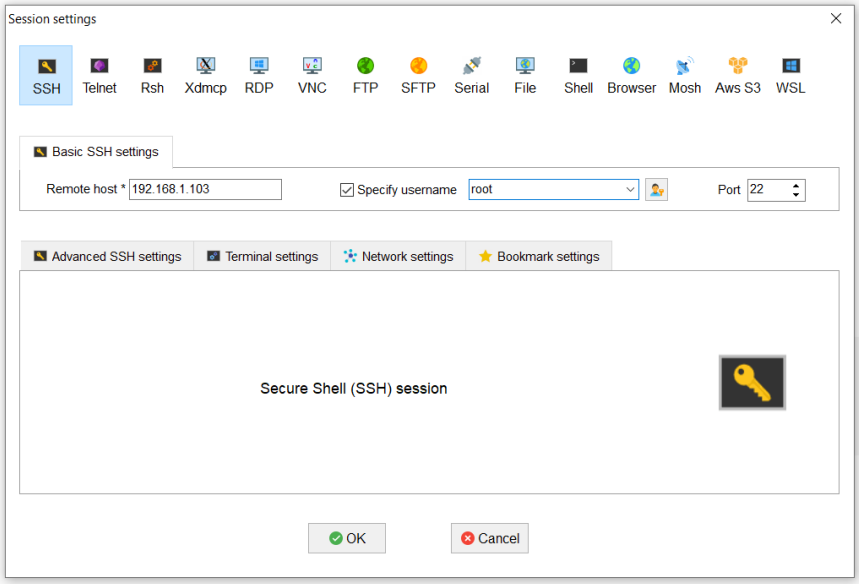


Рисунок 3.3 – Введення даних віртуальної машини.

Після створення віртуальної сесії вводимо пароль від ВМ *PNET* (за замовчуванням *pnet*). Після цього все буде готове для роботи сесіїї між *PNETLab* та *MobaXterm*.

2) На даний момент нам не доступні більшість з віртуальних образів пристроїв для *PNETLab*. Для того щоб їх завантажити у середовище *PNET*, використаємо утіліту *ishare2* (інструмент на основі *CLI*, написаний на *Bash*, для легкого завантаження та керування зображеннями на сервері *PnetLab*).

Для того щоб її встановити перейдемо на веб-посилання *https://github.com/pnetlabrepo/ishare2* , та скопіюємо варіант для встановлення *ishere2 «wget»*.

Піcл цього заходимо в створену сесію в *MobaXterm*, та вставляємо скопійований текст для встановлення *ishere2* та натискаємо *Enter*. Після вибору налаштувань за замувчуванням, *ishare2* буде встновлена.

Для того щоб дізнатись, які віртуальні пристрої доступні для завантаження з утіліти, вводимо команду *«ishare2 search ?»* та натискаємо *Enter*. Після цього на екрані з’явиться близько 1700 пристроїв, які можуть бути додані в *PNETLab* за допомогою *ishare2* (рис. 3.4).

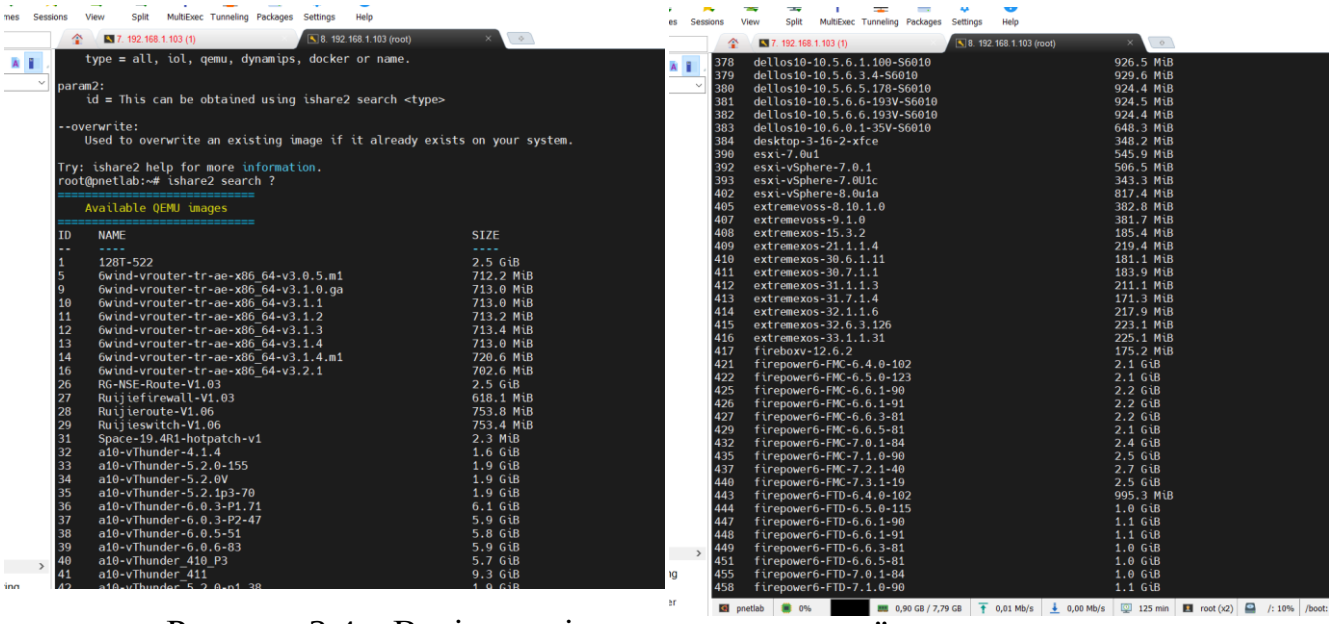


Рисунок 3.4 – Варіанти віртуальних пристроїв для завантаження в *PNETLab.*

Для підвантаження пристрою у *PNET* вводимо команду *«ishare2 pull all, iol, qemu, dynamips, docker or name* номер пристрою у списку ».

Після завантаження повертаємось у веб браузер, там де у нас знаходиться основне середовище *PNETLab*, створюємо новий проект, вводимо для цього назву проекту, за необхідності роздаємо права доступу(рис. 3.5).

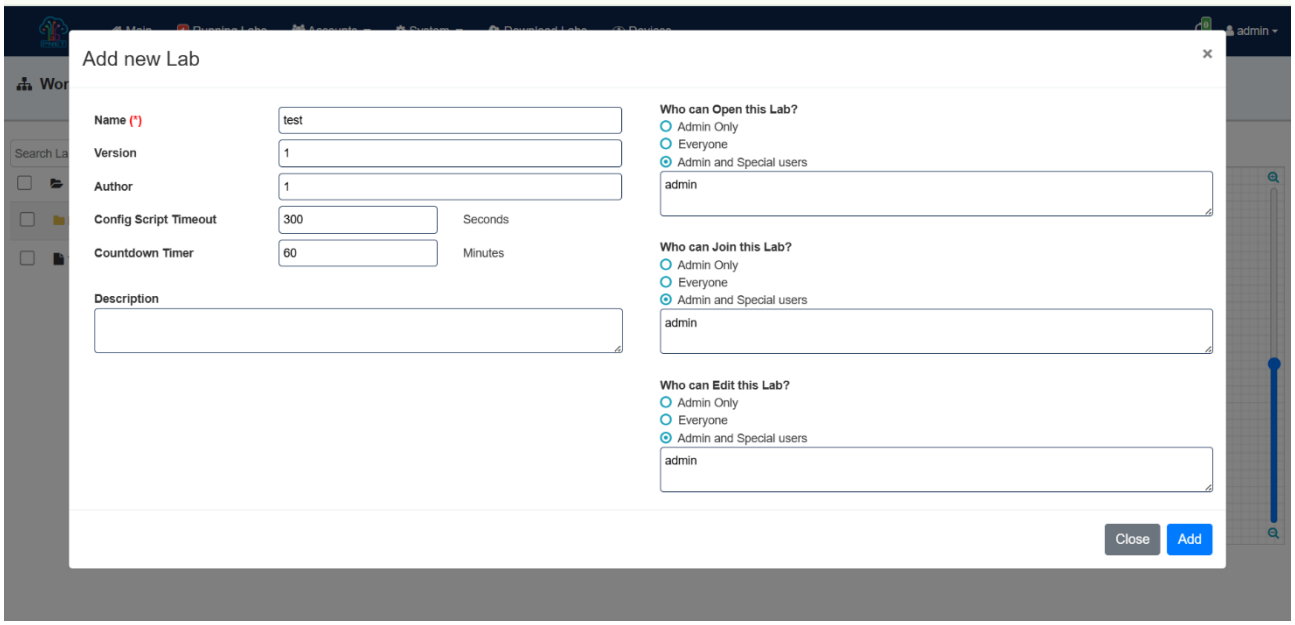
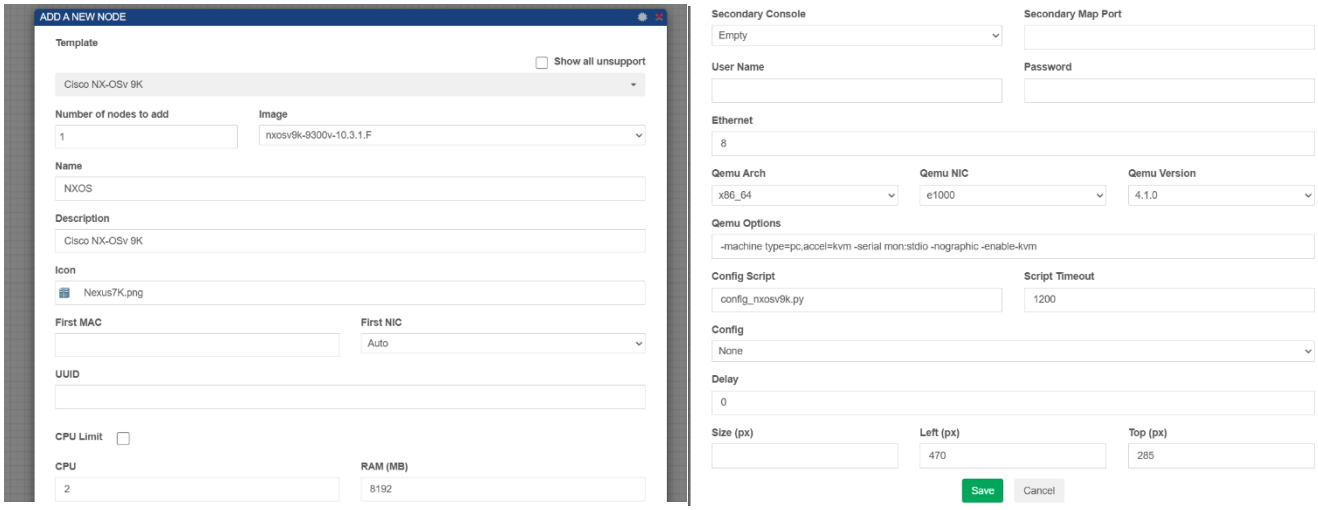


Рисунок 3.5 – Створення нового проекту *PNETLab.*

Після створенно проекту відкриваємо його та у пусте робоче поле натискаємо праву кнопку миші, з запропонованих опцій вибираємо «додати новий об’єкт». Вибираємо підвантажений нами об’єкт. За необхідності визначаємо налаштування для нього (рис. 3.6).



а) б)

Рисунок 3.6 – Налаштування пристрою.

Після натискання *Save*, пристрій, який був попередньо завантажений, з’явится у робочій області *PNETLab* (рис 3.7).

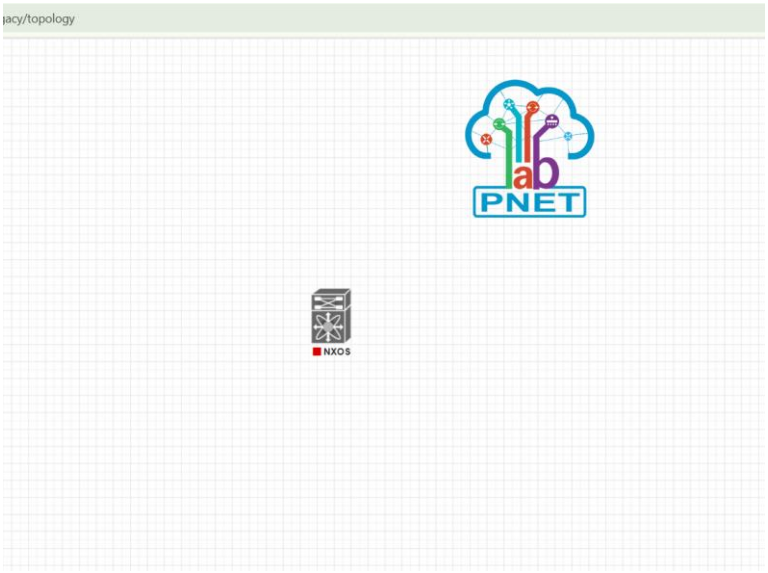
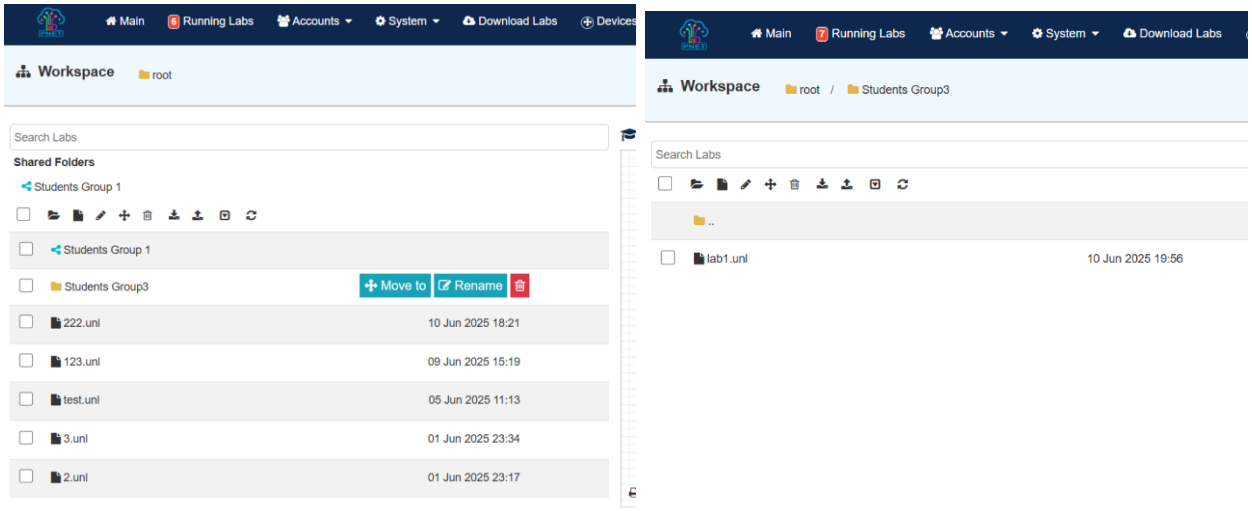


Рисунок 3.7 – Завантажений у проект віртуальний образ пристрою.

3.2 Налаштування користувачів лабораторії

Для додавання студентських облікових записів до *PNETLab* першим кроком буде створення ролей на обліковому записі admin (викладача/власника).

1) Перед почтаком створення нової ролі (групи студентів) необхідно створити папку, до якої вони будуть мати доступ (в даному варіанті Students *Group3*), за необхідності створити лабораторний проект, яким вони зможуть користуватися (рис. 3.8).



а) б)

Рисунок 3.8 – Створення папки для студентів, та лабораторної роботи у ній.

Щоб створити роль: Облікові записи > Керування ролями > Кнопка «Додати».

Вказуємо назву ролі: (*Student Group 3 2025*), робочий простір – це папка, в якій може працювати лише користувач (*Students Group3*), налаштувати дозвіл для ролі. Серед налаштувань дозволу маємо (рис. 3.9):

– видалити папку: Користувачі можуть видалити папку у своєму робочому просторі;

– додати нову папку: Користувачі можуть додавати нову папку до свого робочого простору;

– перейменувати або перемістити папку: Користувачі можуть змінити назву або перетягнути папку на своєму робочому просторі;

– видалити лабораторію: Користувачі можуть видаляти лабораторії у своєму робочому просторі;

– додати нову лабораторію : Користувачі можуть додавати нові лабораторії до свого робочого простору;

– імпорт Лабораторії: Користувачі можуть використовувати Імпорт функції Lab у свою робочу область;

– експорт лабораторії: Користувачі можуть використовувати Експорт функції лабораторії у своєму робочому просторі;

– переміщення лабораторії : Користувачі можуть переміщувати лабораторію на своєму робочому просторі;

– клонувати лабораторію : Користувачі можуть клонувати лабораторію у своєму робочому просторі.

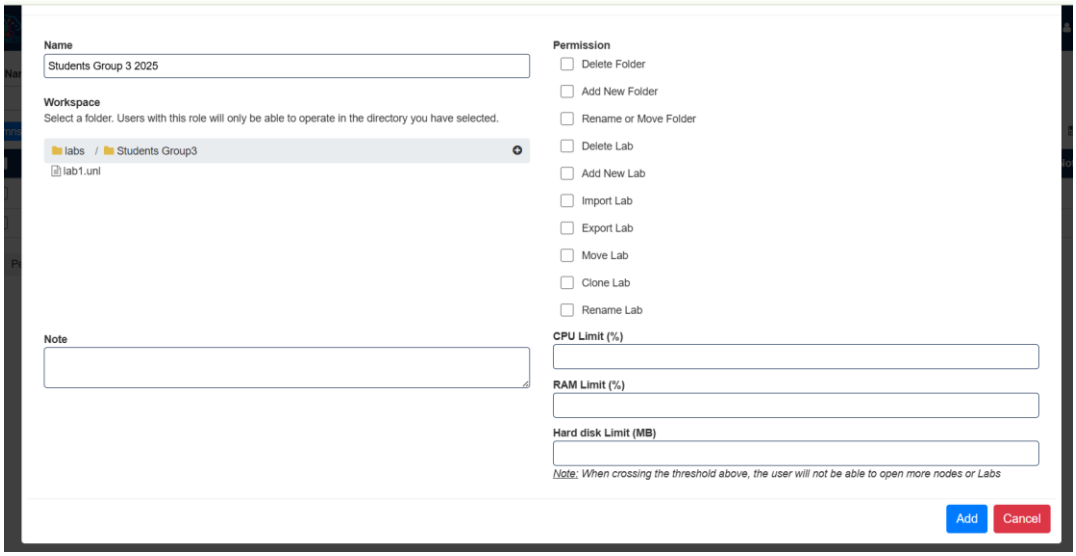


Рисунок 3.9 – Вікно створення нової ролі.

Натиснувши кнопку «Додати », роль збережеться. Тепер можно побачити нову роль у таблиці (рис. 3.10).

	STT	Setup	Role Name	Role Workspace
☐	1	✎	Students Group 3 2025	/Students Group3
☐	2	✎	student	/
☐	3	✎	Group students 1 2015	/Students Group 1

Рисунок 3.10 – Збережена нова роль.

2) Додавання студентських облікових записів:

– щоб додати обліковий запис до своєї скриньки, вибираємо Облікові записи > Керування користувачами > Натисніть кнопку Додати. Потрібно ввести (рис. 3.12): ім’я користувача; пароль; вибрати роль, яка була створена раніше; вибрати вузол, що був створений для цього користувача;

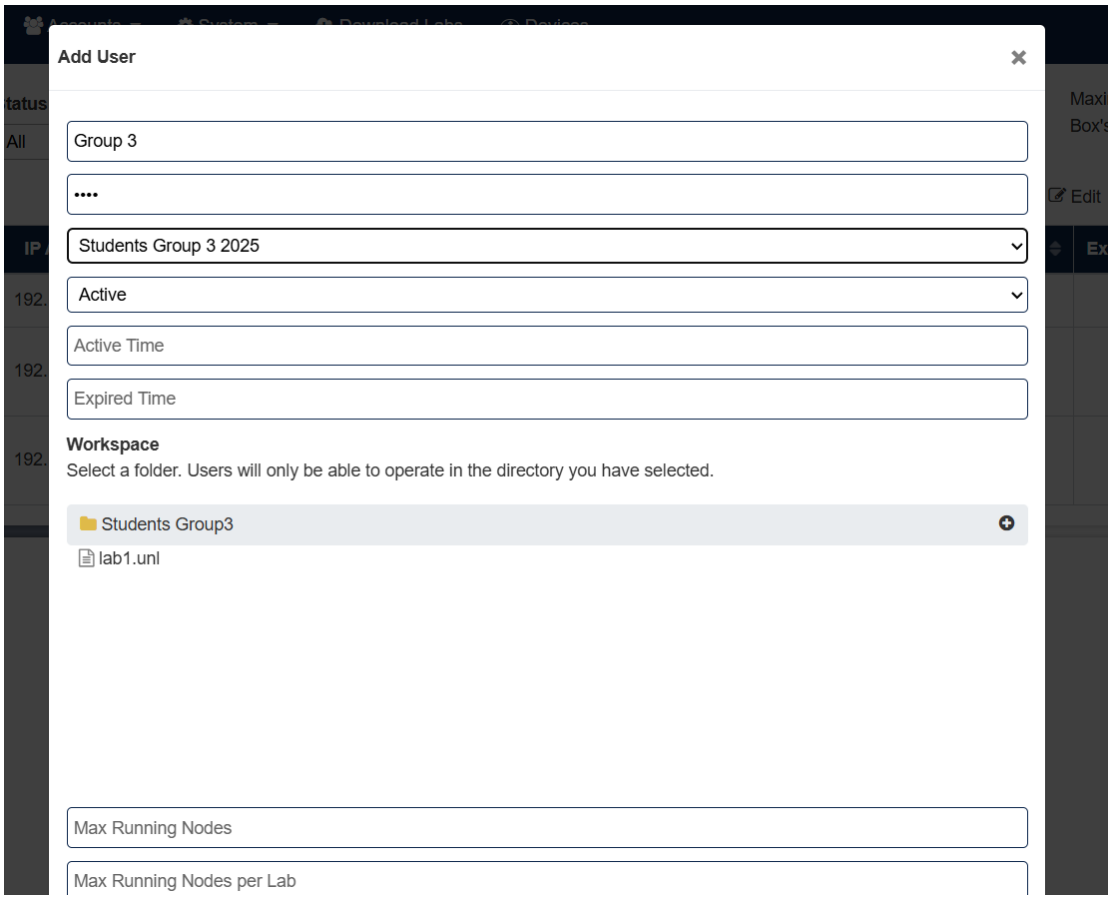


Рисунок 3.12 – Інформація, яку потрібно вказати для створення облікового запису.

– натиснувши кнопку «Додати», можна тепер побачити всіх доданих користувачів у таблиці (рис. 3.14).

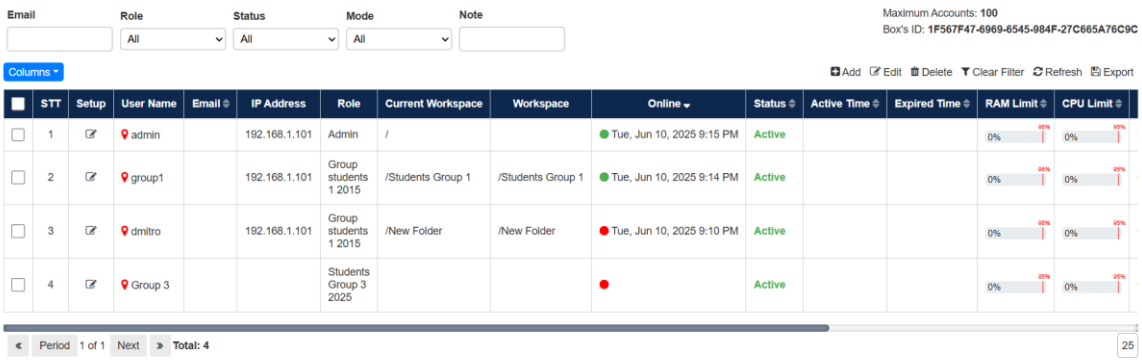


Рисунок 3.14 – Вікно, де можно бачити всі активні облікові записи.

3) Надати спільний доступ до папки (*Students Group3*):

– необхідно вибрати папки, до яких необхідно надати доступ. Змінити дозвіл для користувача в розділі «Спільний доступ до папок» (рис. 3.15).

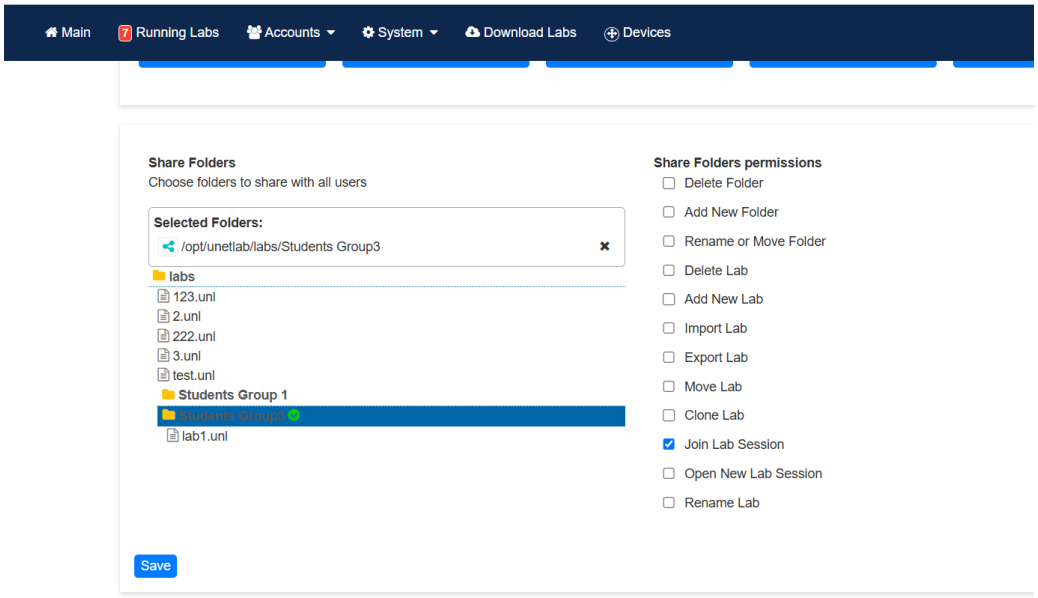


Рисунок 3.15 – Надання доступу до папки студенту (або групі студентів).

Після збереження користувачі бачитимуть папку та матимуть до неї доступ.

4) Налаштування дозволу для лабораторії:

– *PNETLab Box* дозволяє встановити дозволи для кожної лабораторії. Для цього треба вказати, хто може відкривати лабораторію, хто може приєднуватися до лабораторії та хто може редагувати її. Для цього: *Main* > Вибираємо лабораторія для редагування доступу до неї студентів > *Edit* (рис. 3.16);

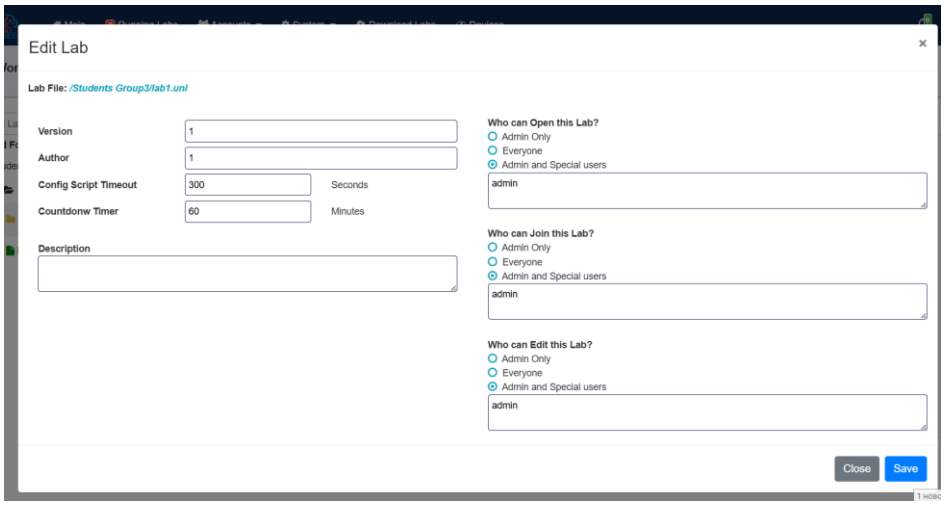


Рисунок 3.16 – Поле для редагування доступу.

– для того, щоб студент мав можливість працювати в лабортаорії, у полях *«Who can Open this Lab?; Who can Join this Lab?; Who can Edit this Lab?»* вибираємо нашого студента (рис. 3.17);

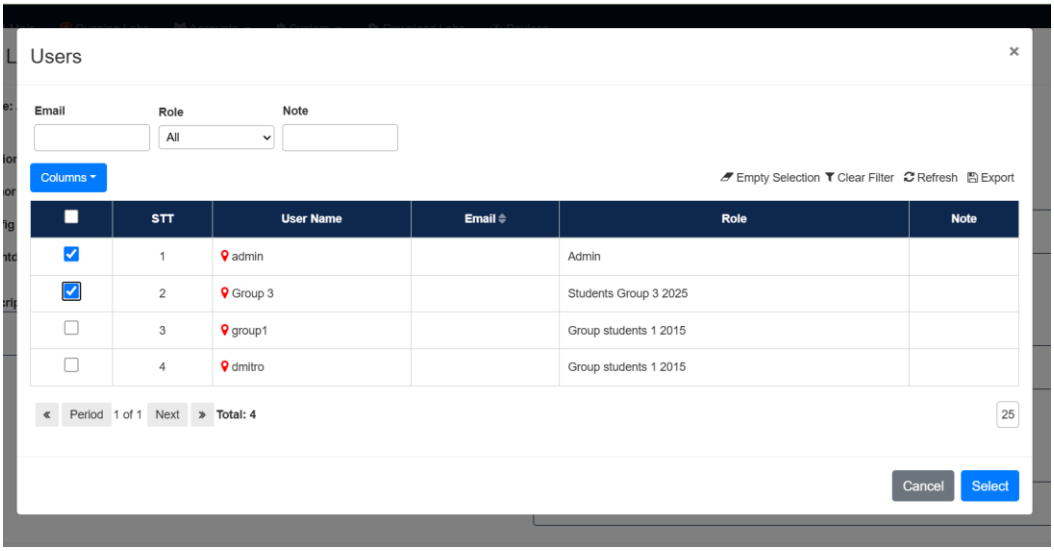


Рисунок 3.17 – Вибір користувача.

– після збереження налаштувань студент зможе в повній мірі користуватися лабораторією (рис. 3.18).

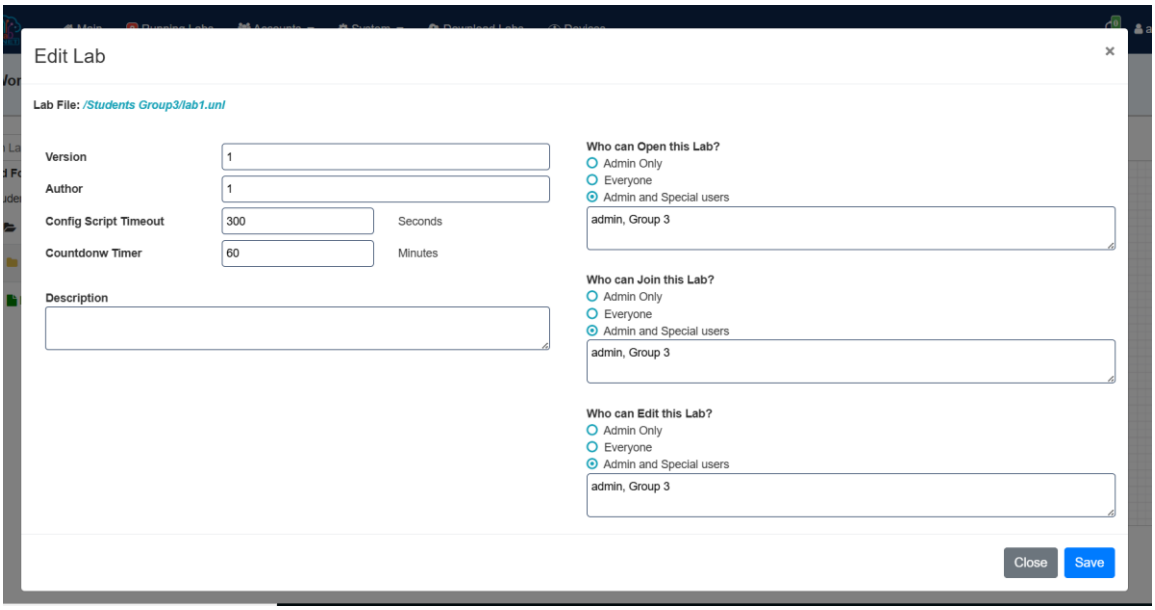


Рисунок 3.18 – Збереження налаштувань доступу.

5) Щоб студент міг підключитися до створеної для нього лабораторної роботи або папки, йому необхідно ввести у веб-інтерфейс активну *IP*-адресу *PNETLab* викладача, заповнити ім’я, пароль, які були створені викладачем.

6) Керування запущеними лабораторіями:

– коли користувач відкриває будь-яку лабораторну роботу, створюється та відображається лабораторний сеанс;

– запуск лабораторної роботи: користувач, який відкрив лабораторну роботу, стане її ведучим (рис. 3.19).

Можно побачити інформацію про організатора сесії, хто приєднується до лабораторії.

Для кожного лабораторного заняття можно виконати певну дію:

– отримати інформацію про лабораторію, натиснувши на стовпець «Шлях до лабораторії сеансу»;

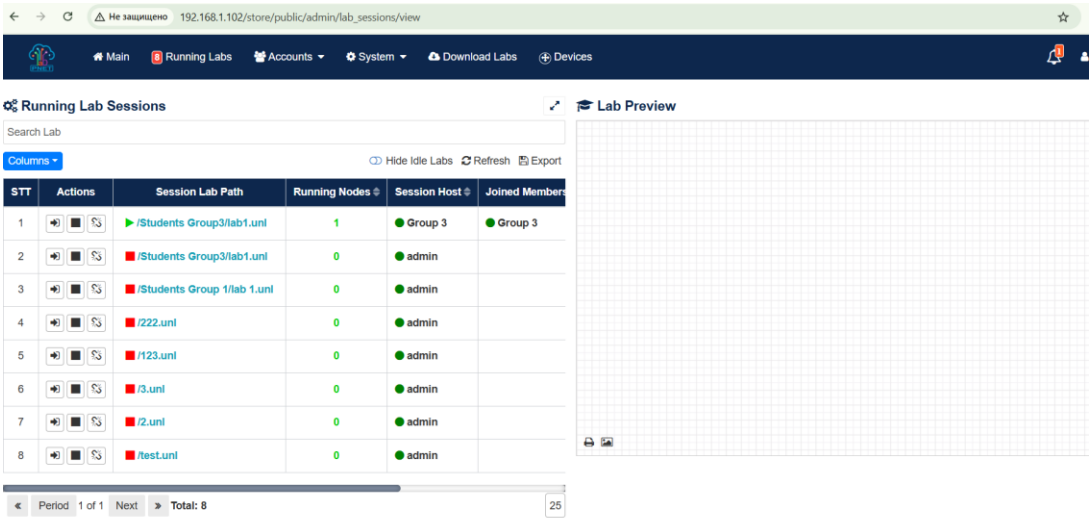


Рисунок 3.19 – Менеджер лпбораторій.

– приєднатися до лабораторії, натиснувши *«Join Lab Session»;*

– знищити лабораторію, натиснувши *«Destroy Lab Session»* (рис. 3.20) : тільки хост сеансу та адміністратор можуть знищити лабораторію, якщо не було налаштовано це для звичайного користувача.

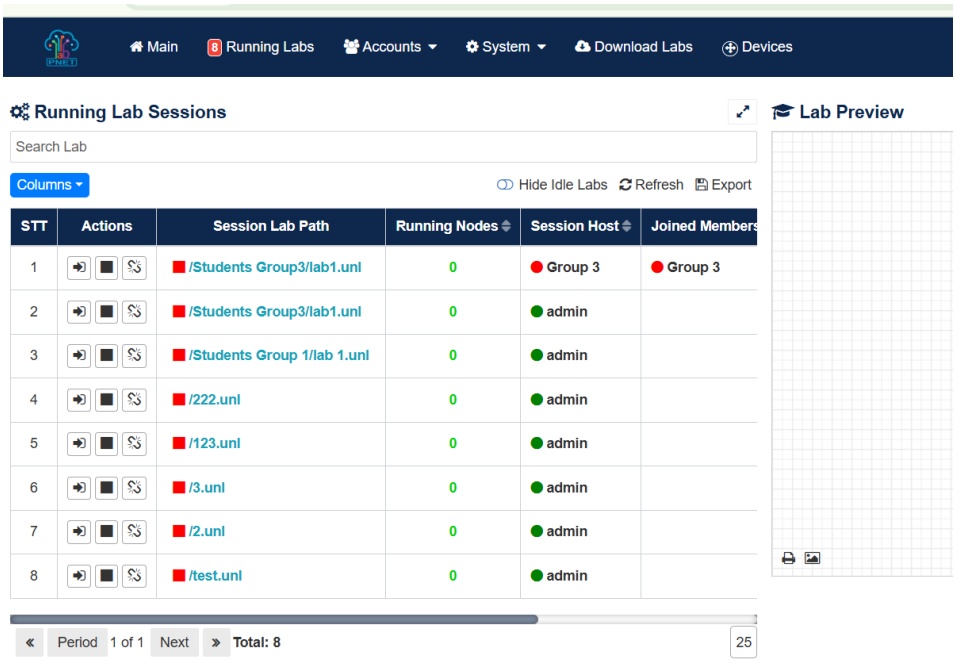


Рисунок 3.20 – Керування лабораторією.

РОЗДІЛ 4. ПЕРЕВІРКА ПРАЦЕЗДАТНОСТІ СИСТЕМИ

4.1 Побудова схеми моделювання

Для перевірки працездатності системи було створено віртуальну лабораторії обміну даними з підтримко тунелю *Ipsec (IP Security)*. Топологія мережі представлена на рисунку 4.1.

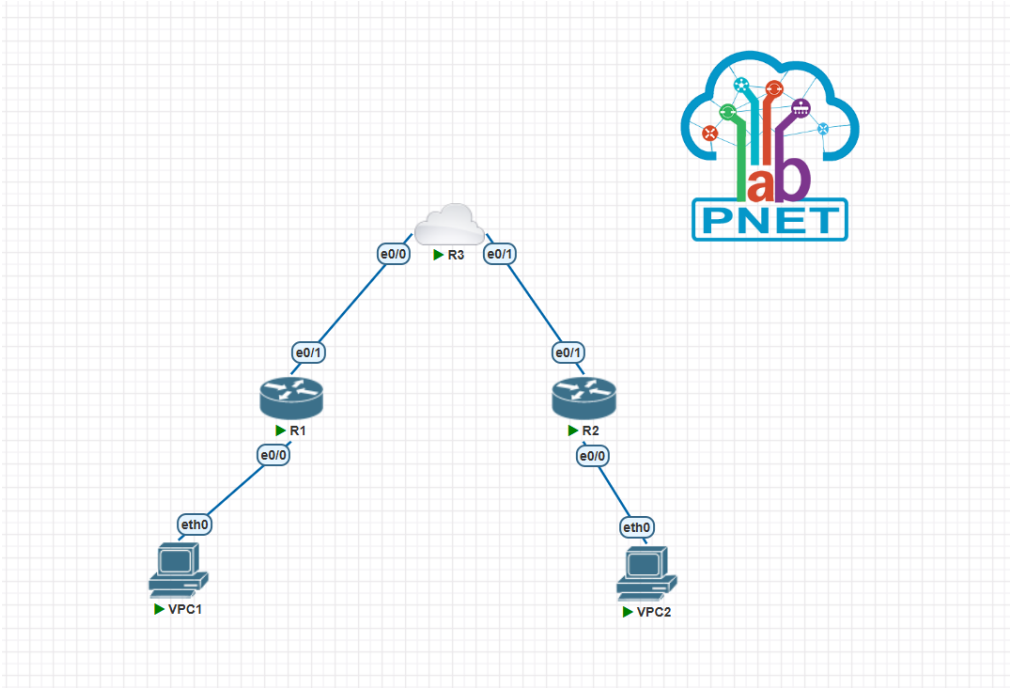


Рисунок 4.1 – Мережева топологія.

Мережева топологія, змодельована в середовищі *PNETLab*. Вона включає три маршрутизатори (*R1, R2, R3*), два віртуальні ПК (*VPC*). Попередні налаштування пристроїв:

1) віртуальний ПК *VPC1*:
   – IP-адреса: 10.10.10.10/24;
   – інтерфейс: *eth0*;
   – підключення до маршрутизатора *R2* через інтерфейс *e0/0*;
2) віртуальний ПК *VPC2*:

– *IP*-адреса: 20.20.20.20/24;

– інтерфейс: *eth0*;

– підключення до маршрутизатора *R3* через інтерфейс *e0/0;*

3) маршрутизатор *R1* компанії *Cisco*:

– *e0/0*: підключення до *VPC* (10.10.10.10);

– *e0/1*: *IP*-адреса 83.144.112.87, підключено до *R3 (e0/0);*

4) маршрутизатор *R2* компанії *Cisco*:

– *e0/1: IP* 76.45.118.56, підключено до *R1*;

– *e0/0*: підключено до *VPC* (праворуч);

5) маршрутизатор *R3* компанії *Cisco*:

– *e0/0: IP* 83.144.112.88, підключено до *R2*;

– *e0/1: IP* 76.45.118.57, підключено до *R3*.

Основна мета — забезпечити зв’язок між двома хостами (*VPC*) через ланцюг маршрутизаторів. Структура може бути використана для:

– налаштування та тестування маршрутизації (статичної або динамічної);

– перевірки доступності мережі (*ping, traceroute*);

– конфігурації *IPsec* як політики безпеки мережі.

4.2 Налаштування обладнання

Для налаштування елементів мережі використовуємо систему віддаленого контролю *Putty*. Для цього у поле *IP* вводимо *IP*-адресу сесії у *PNETLab*, у поле *Port* вводимо порт віртуального елементу та тип з’єднання обираємо *Telnet* (рис. 4.2).

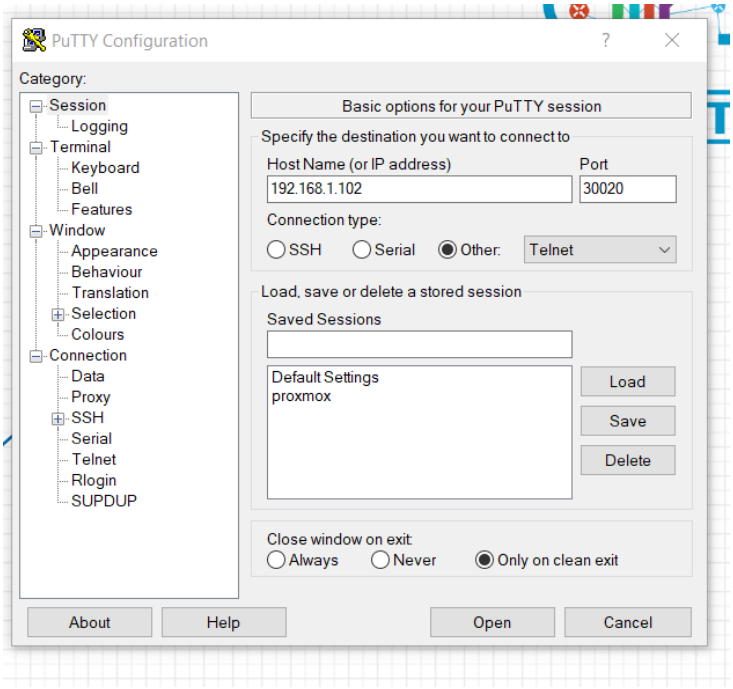


Рисунок 4.2 – Налаштування з’єднання *Putty*.

4.2.1 Налаштування маршрутизатора *R1:*

– вхід в режим налаштування:

enable

conf t

– налаштування з’єднання:

interface ethernet 0/0

ip address 10.10.10.1 255.255.255.0

no shutdown

exit

interface ethernet 0/1

ip address 83.144.112.87 255.255.255.0

no shutdown

exit

– налаштування *ISAKMP* (Фаза 1 *IPsec*) – встановлення «довіри»:

crypto isakmp policy 1
encr 3des
hash md5
authentication pre-share
group 2
lifetime 86400
crypto isakmp key devops address 76.45.118.56
ip access-list extended VPN-TRAFFIC
permit ip 10.10.10.0 0.0.0.255 20.20.20.0 0.0.0.255

– налаштування *IPsec* (Фаза 2) – визначає методи шифрування та автентифікації даних:

crypto ipsec transform-set TS esp-3des esp-md5-hmac
mode tunnel
bash
crypto map CMAP 10 ipsec-isakmp
set peer 76.45.118.56
set transform-set TS
match address VPN-TRAFFIC
interface ethernet 0/1
crypto map CMAP

– налаштування *NAT* з винятком для *VPN*:
ip nat inside source list 100 interface ethernet 0/1 overload
access-list 100 deny ip 10.10.10.0 0.0.0.255 20.20.20.0 0.0.0.255
access-list 100 permit ip 10.10.10.0 0.0.0.255 any

ip route 0.0.0.0 0.0.0.0 83.144.112.88

У результаті маємо функції *R1*:

– підключено локальну мережу *VPC1* (10.10.10.0/24) до інтернету (*R3*);

– встановлено *IPsec VPN* тунель до *R2* (шифрує трафік);

– виключено *VPN*-трафік з *NAT*.

4.2.2 Налаштування маршрутизатора *R2* має аналогічні етапи що й *R1*. Список команд:

enable
conf t
interface ethernet 0/0
ip address 20.20.20.1 255.255.255.0
no shutdown
exit
interface ethernet 0/1
ip address 76.45.118.56 255.255.255.0
no shutdown
exit
crypto isakmp policy 1
encr 3des
hash md5
authentication pre-share
group 2
lifetime 86400
exit
crypto isakmp key devops address 83.144.112.87
ip access-list extended VPN-TRAFFIC
permit ip 20.20.20.0 0.0.0.255 10.10.10.0 0.0.0.255
exit

crypto ipsec transform-set TS esp-3des esp-md5-hmac

mode tunnel

exit

crypto map CMAP 10 ipsec-isakmp

set peer 83.144.112.87

set transform-set TS

match address VPN-TRAFFIC

exit

interface ethernet 0/1

crypto map CMAP

exit

ip nat inside source list 100 interface ethernet 0/1 overload

access-list 100 deny ip 20.20.20.0 0.0.0.255 10.10.10.0 0.0.0.255

access-list 100 permit ip 20.20.20.0 0.0.0.255 any

ip route 0.0.0.0 0.0.0.0 76.45.118.57

У результаті маємо функції *R2*:

– підключено локальну мережу *VPC2* (20.20.20.0/24) до інтернету (*R3*);

– встановлює другим кінцем *IPsec VPN* тунелю;

– виняток *VPN*-трафіку з *NAT* – щоб *VPN* працював.

4.2.3 Виконаємо налаштування для маршрутизатора *R3* як проміжного маршрутизатора:

enable

conf t

interface ethernet 0/1

ip address 76.45.118.57 255.255.255.0

no shutdown

exit

interface ethernet 0/0

ip address 83.144.112.88 255.255.255.0

no shutdown

exit

Маємо функції *R3:*

– проста передача трафіку між *R1* і *R2*;

– не бере участі в шифруванні чи *VPN*;

– служить «інтернетом» між двома маршрутизаторами.

4.3 Перевірка з’єднання мережі

Перевіримо зв’язок між роутерами *R1* та *R3* (рис. 4.3), для цього введемо у консоль *R3* команду *ping* 83.144.112.87.

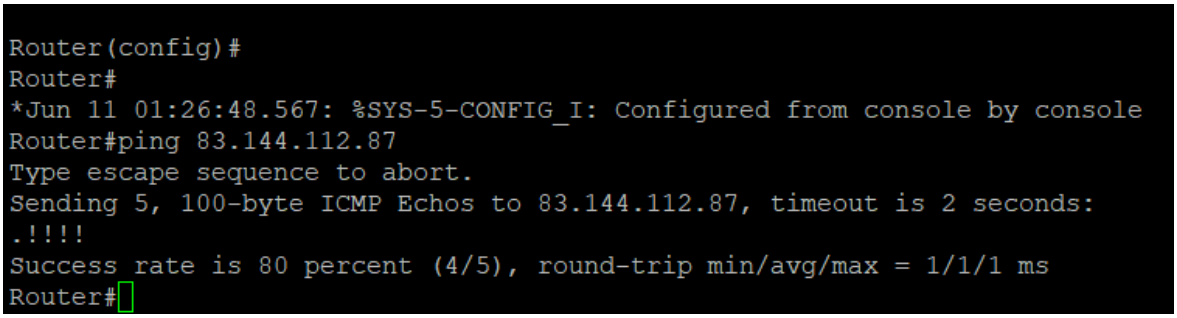


Рисунок 4.3 – Успішна перевірка з’єднання між *R3* та *R1*.

Отримано відповідь:

– відправляється 5 *ICMP*-запитів (ехо-запитів) розміром 100 байтів з затримкою для кожного – 2 секунди;

– 80% успішність зв’язку (4 з 5 запитів отримали відповідь);

– час затримки: мінімум / середній / максимум = 1 мс.

Зв’язок між *R3* та *R2* (рис. 4.4): *ping* 76.45.118.56

Отримано відповідь:

– відправляється 5 *ICMP*-запитів (ехо-запитів) розміром 100 байтів з затримкою для кожного – 2 секунди;

– 80% успішність зв'язку (4 з 5 запитів отримали відповідь);

– час затримки: мінімум / середній / максимум = 1 мс.

```
Router#ping 76.45.118.56
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 76.45.118.56, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
Router#
```

Рисунок 4.4 – Успішна перевірка з'єднання між *R3* та *R2*.

Перевірка з'єднання за допомогою *IPsec* між *VPC1* і *VPC2* через тунель. За допомогою команди ping (з *VPC1* на *VPC2: ping* 20.20.20.20 (рис. 4.5)) .

```
VPCS> ping 20.20.20.20

20.20.20.20 icmp_seq=1 timeout
84 bytes from 20.20.20.20 icmp_seq=2 ttl=62 time=2.170 ms
84 bytes from 20.20.20.20 icmp_seq=3 ttl=62 time=1.029 ms
84 bytes from 20.20.20.20 icmp_seq=4 ttl=62 time=1.077 ms
84 bytes from 20.20.20.20 icmp_seq=5 ttl=62 time=1.051 ms
```

Рисунок 4.5– Перевірка зв'язку між ПК мережі.

Тестування показало наявність безперебійного зв'язку з'єднання між двома комп'ютерами. Таким чином проведення симуляції довело працездатність розгорнутої системи та можливість моделювання комп'ютерних мереж.

ВИСНОВКИ

У даній кваліфікаційній роботі:

– проведено аналіз сучасних програмно-апаратних засобів емуляції комп’ютерних мереж та виділено перспективність використання *PNETLab* в якості симулятору комп’ютерних мереж;

– виконано розгортання програмного середовища *PNETLab* на сервері кафедри, завантажено образи операційних систем мережевого обладнання та проведено попередні налаштування симулятору;

– працездатність системи було доведено створенням та налаштуванням віртуальної лабораторії з симуляції інформаційного обміну з використанням IP-sec.

Результати роботи мають практичну значимість і можуть бути використані для забезпечення лабораторної бази дисциплін, пов’язаних з мережевими технологіями та кібербезпекою.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Cisco Networking Academy. Introduction to Networks v7.0. URL: https://netacad.com

2. Cisco Networking Academy. Switching, Routing, and Wireless Essentials v7.0. URL: https://netacad.com

3. Stallings, W. Network Security Essentials: Applications and Standards. — Pearson, 2013. — 416 p.

4. Bishop, M. Computer Security: Art and Science. — Addison-Wesley, 2003. — 1084 p.

5. Cisco Packet Tracer. URL: https://www.netacad.com/courses/packet-tracer

6. The ns-3 Network Simulator. URL : https://www.nsnam.org/

7. GNS3 — Graphical Network Simulator. URL: https://www.gns3.com

8. OPNET Modeler Product Overview. URL: https://www.riverbed.com/

9. QualNet Network Simulator. URL: https://web.scalable-networks.com/qualnet-network-simulator

10. Boson NetSim for Cisco. URL: https://www.boson.com/netsim-cisco-network-simulator

11. Mininet: An Instant Virtual Network on your Laptop. URL: http://mininet.org/

12. Cisco VIRL (Virtual Internet Routing Lab). URL: https://learningnetworkstore.cisco.com/virl.html

13. EVE-NG Documentation. URL: https://www.eve-ng.net/

14. PNETLab Project Documentation. URL: https://pnetlab.com/docs/

15. Столбовий А. О., Вітюк Б. Б. Організація захисту інформації в комп’ютерних мережах: навч. посіб. – К. : НА СБУ, 2021. – 248 с.

16. *ISO/IEC* 27001:2022. *Information Security, Cybersecurity and Privacy Protection – Information Security Management Systems – Requirements*. – Geneva : ISO, 2022. – 35 p.

17. RFC 2196. Site Security Handbook. URL: https://tools.ietf.org/html/rfc2196

18. Прокопенко Н. А. Основы защиты информации в компьютерных системах и сетях: учеб. пособие. – М. : Градиент, 2018. – 256 с.

19. Колесніков С. О. Адміністрування операційних систем і мережевих служб Linux: навч. посіб. – К. : ДП «Інформ.-аналіт. агентство», 2020. – 295 с.

20. Васильєв В. П. Мережеві технології захисту інформації: навч. посіб. – Харків : ХНУРЕ, 2019. – 234 с.