Міністерство освіти і науки України
Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп’ютерних систем та мереж

Пояснювальна записка
до кваліфікаційної роботи бакалавра
за спеціальністю 123 «Комп’ютерна інженерія»

на тему: СИСТЕМА ВІРТУАЛІЗАЦІЇ PROXMOX VE ДЛЯ КЕРУВАННЯ СЕРВЕРОМ

Проектував	_____	Р. О. Вороненко
Керівник роботи	_____	Ю. О. Кумченко
Нормоконтроль	_____	Д. І. Кузнєцов
Завідувач кафедри	_____	А. І. Купін

Кривий Ріг

2025

Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

Ступінь вищої освіти бакалавр
Спеціальність 123 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ
Завідувач кафедри, голова циклової комісії

_____ А. І. Купін

"____" _____ 20__ року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

_____
(прізвище, ім'я, по батькові)

1. Тема роботи _____
_____
_____
керівник роботи _____,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від "__"____ 20__року №__

2. Строк подання студентом роботи _____
3. Вихідні дані до роботи _____
_____
_____
_____
_____
_____
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)_____
_____
_____
_____
_____
_____
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)_____
_____
_____
_____

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання_____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Строк виконання етапів роботи	Примітка

Студент _____ _____
(підпис) (прізвище та ініціали)

Керівник роботи _____ _____
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка: 75 сторінок, 50 рисунків, 4 таблиць, 0 додаток, 7 використаних джерел.

Об’єкт аналізу – система віртуалізації Proxmox VE для керуванням сервером

Проект складається з трьох розділів.

Перший розділ присвячений аналізу технологій віртуалізації та особливості використання Proxmox VE. Огляду інших систем вірталізації та порівнянню з Proxmox VE. Розгляду актуальності впровадження Proxmox VE в Україні.

У другому розділі розкриті питання вибору апаратної частини серверної мережі на підприємстві, а також наведено перелік програмного забезпечення.

Третій розділ присвячений моделюванню, а також аналізу недоліків. Вкінці розділу подано рекомендації щодо покращення роботи

PROXMOX VE, СЕРВЕР, KVM, LXC, OS, КОМП’ЮТЕРНА МЕРЕЖА, VIRTUALBOX, VMWARE VSPHERE, РОБОЧА СТАНЦІЯ, СТРУКТУРОВАНА КАБЕЛЬНА СИСТЕМА, XENSERVER, HYPER-V.

Explanatory note: 75 pages, 50 figures, 4 table, 0 appendix, 7 sources used.

The object of analysis is the Proxmox VE virtualization system for server management.

The project consists of three parts.

The first section is devoted to the analysis of virtualization technologies and the features of using Proxmox VE. An overview of other virtualization systems and a comparison with Proxmox VE. Consideration of the relevance of implementing Proxmox VE in Ukraine.

The second section discusses the issues of choosing the hardware part of the server network in the enterprise, and also provides a list of software.

The third section is devoted to modeling, as well as analysis of shortcomings. At the end of the section, recommendations are given for improving the work

PROXMOX VE, SERVER, KVM, LXC, OS, COMPUTER NETWORK, VIRTUALBOX, VMWARE VSPHERE, WORKSTATION, STRUCTURED CABLE SYSTEM, XENSERVER, HYPER-V.

ЗМІСТ

					КНУ.РБ.123.25.04.3		
Змн	Арк.	№ Документа	Підпис	Дата			
Розробив		Вороненко			ЗМІСТ	Літера	Аркуш / Аркушів
Перевірив		Кумченко					
Н.контроль		Кузнєцов				КІ-21	
Затвердив		Купін					

ПЕРЕЛІК СКОРОЧЕНЬ

Proxmox VE – Proxmox Virtual Environment
VM – Virtual Machine (віртуальна машина)
KVM – Kernel-based Virtual Machine
LXC – Linux Containers
CPU – Central Processing Unit (центральний процесор)
RAM – Random Access Memory (оперативна пам’ять)
HDD – Hard Disk Drive (жорсткий диск)
SSD – Solid State Drive (твердотільний накопичувач)
LAN – Local Area Network (локальна мережа)
OS – Operating System (операційна система)
LM – Live Migration
MTTR — Mean Time to Recovery
ОП – Операційна система

ВСТУП

Сучасні інформаційні технології вимагають ефективних рішень для управління серверною інфраструктурою. Віртуалізація є одним із ключових підходів, що дозволяє оптимізувати використання апаратних ресурсів, підвищити рівень безпеки та забезпечити гнучкість масштабування. Одним із потужних інструментів для реалізації таких рішень є система віртуалізації Proxmox Virtual Environment (Proxmox VE), яка поєднує технології контейнерної (LXC) та повної апаратної (KVM) віртуалізації.

Proxmox VE базується на Debian Linux і є системою з відкритим вихідним кодом, що забезпечує високий рівень налаштовуваності та адаптивності. Її використання дозволяє розгортати віртуальні середовища без додаткових витрат на ліцензійне програмне забезпечення, що є важливим чинником для організацій з обмеженим бюджетом. Крім того, система має розвинені механізми централізованого управління, резервного копіювання та відновлення, що сприяє підвищенню надійності серверної інфраструктури.

Використання Proxmox VE в Україні є актуальним завдяки її економічній ефективності, незалежності від комерційних постачальників та високому рівню кібербезпеки. Це робить її привабливим рішенням для державних установ, бізнесу та військових організацій.

У межах цієї роботи ми проаналізуємо можливості використання Proxmox VE для керування серверною інфраструктурою. Розглянемо основні компоненти системи, її переваги та недоліки у порівнянні з іншими рішеннями, а також виконаємо оцінку продуктивності та масштабованості в різних сценаріях експлуатації. Отримані результати сприятимуть ефективному впровадженню Proxmox VE у різних сферах, зокрема в державному та комерційному секторі.

1. АНАЛІЗ ТЕХНОЛОГІЙ ВІРТУАЛІЗАЦІЇ ТА ОСОБЛИВОСТІ ВИКОРИСТАННЯ PROXMOX VE

1.1 Сучасні тенденції віртуалізації серверної інфраструктури

1.1.1 Історичний розвиток та еволюція технологій віртуалізації

Історія віртуалізації починається ще в 60-х роках минулого сторіччя.[1] Треба відмітити, що у першоджерел стояли такі компанії як International Business Machines (IBM), General Electric (GE), Bell Labs. Перші кроки у той час зробила саме компанія IBM, яка працювала над можливістю одночасного виконання різних задач декількома програмами. Раніше для того, щоб виконати дві задачі під час одного процесу, було необхідно їх запустити у пакетному режимі. Але це не турбувало користувачів до тих пір, поки комп'ютери були у доступі тільки для вузького кола науковців.

Поступово необхідність у виконанні різноманітних задач зростала. Також збільшувалась кількість робочих систем, випущених безпосередньо в IBM. Все це вимагало робити кроки у пошуку нового інструменту, здатного повністю замінити усі попередні розробки. Почалися розробки мейнфрейма S/360, однак цей продукт у подальшому так і не використовували. Хоча планувалось, що саме S/360 зможе працювати над декількома процесами одночасно у формі пакетного режиму.
Однак влітку 1963 року відбулося відкриття проекту MAC на базі Массачусетського технологічного інституту (MIT). Цікаво, що його мета на самому початку була зовсім іншою, що відображала розшифровка абревіатури – «Математика та обчислення». Але згодом під MAC почали розуміти «Комп'ютер множинного доступу».

Однією з задач, яку повинні були вирішити під час роботи над проектом, було дослідження операційних систем, теорії обчислень та штучного інтелекту. Для цього знадобилось таке обладнання для комп'ютера, за допомогою якого з'явилася б можливість одночасної роботи декільком користувачам. Запит на обладнання був надісланий різним компаніям, у тому числі й IBM.

Як з'ясувалось, система S/360 не підходила, тому був створений новий мейнфрейм CP-40. У подальшому, його використання було обмежено лише науковцями, але саме ця розробка стала основою для першого комерційного мейнфрейма CP-67 з підтримкою віртуалізації. Також була розроблена операційна система CP/CMS. З її допомогою з'явилась можливість на основі модуля CP запустити віртуальну машину на ОС CMS, яку могли застосовувати вже користувачі. В подальшому у 1972 році IBM створила операційне середовище System 370 та операційну систему VM/ESA під назвою System 390.[1]

Це був початок розвитку довгого шляху віртуалізації.[1] Розглянемо його ключові моменти:

1988 рік – поява емулятора ПЗ Soft PC від Insignia Solution. Завдяки цьому програмному забезпеченню з'явилась можливість використовувати застосунки DOS на інших операційних системах.

1990 рік – компанія Sun Microsystems презентувала проект Stealth, який у подальшому трансформувався у відому сучасним програмістам мову програмування Java.

1997 рік – компанія Apple випустила софт Virtual PC, яка працювала за принципом Soft PC. Це дозволило сумістити Windows та Mac.

1998 рік – заснування компанії VMware. На сьогодні вона є найбільш відомою у сфері віртуалізації.

1999 рік – Vmware розробила програмне забезпечення VMware Workstation. У цьому ж році з'явилась VMware Virtual Platform.

2001 рік – вихід на ринок гіпервізора першого типу ESX Server та другого типу ESX Server від компанії VMware. Це були перші більш сучасні рішення для серверної віртуалізації.

2003 рік – вихід на ринок компанії Microsoft, яка придбала Connectix з програмним забезпеченням Virtual PC.

2003 рік – вихід на ринок VMmanager від компанії ISPsystem. Це спеціальна панель управління, завдяки якій стало можливо кожному використовувати технології віртуалізації.[1]

1.1.2 Основні види віртуалізації: апаратна, програмна, контейнерна

Програмна віртуалізація:

Вся віртуалізація проходить на рівні ядра операційної системи. В результаті кожна з окремих віртуальних машин працюють як самостійний сервер. Перевага програмної віртуалізації полягає в тому, що будь-які процеси можуть працювати на високій швидкості.[2]

Апаратна віртуалізація:

Подібна віртуалізація здійснюється на основі процесорної архітектури. Апаратна віртуалізація передбачає поділ процесора на мониторну і гостьову частини (root і non-root режими). При такій віртуалізації за допомогою гіпервізора можливо пряме управління гостьовими системами, які використовуються ізольовано один від одного.

Контейнерна віртуалізація:

Контейнерна віртуалізація пов'язана з ізоляцією на рівні процесів ОС, доступна тільки в Linux і працює завдяки механізмам контейнеризації namespaces і cgroups. Таке рішення зазвичай використовується на рівні різних сервісів, які є частиною програми. Серед відомих прикладів: Docker і OpenVZ.[2]

1.1.3 Переваги та виклики впровадження віртуалізації у сучасних IT-системах

Оптимізація ресурсів і скорочення витрат.[3]

Оскільки компанії не потрібно купувати фізичне обладнання, обслуговувати його чи заміняти у разі виходу з ладу, вона може зекономити кошти та витрати їх на розвиток нових проєктів тощо. Також це дає можливість вивільнити IT-команду для цікавіших завдань, спрямованих на розвиток бізнесу.

Незалежність від апаратного забезпечення:

Можливість розгортання віртуальних машин на будь-якому обладнанні не має обмежень. ВМ стандартизовані й універсальні. Віртуалізація дає змогу об’єднувати різні сервіси без необхідності купувати сервери чи ПК для них. Користувач може паралельно запускати віртуальні машини та ОС. Наприклад, є можливість працювати на одному комп’ютері з Windows і одночасно запустити Linux, але в іншому вікні.

Відмовостійкість та катастрофостійкість:

Для встановлення необхідного ПЗ зазвичай потрібні драйвери. Коли ж щось стається з обладнанням, на якому воно розміщене, потрібно його перенести на таке ж саме. А це час. Якщо якийсь форс-мажор трапиться з ВМ, то гіпервізор перерозподілить її ресурсів на іншу, і додатки продовжать роботу. Так бізнес зможе уникнути простою.

Зручність та мобільність:

Віртуалізація надає більше свободи у роботі. Працівники зможуть підключатись до необхідних програм у будь-який час, із будь-якого пристрою та із будь-якого куточка світу, де є інтернет.[3]

1.2 Огляд існуючих систем віртуалізації

1.2.1 Програмні платформи для віртуалізації: VMware VSphere, Hyper-V, XenServer, KVM

1. vSphere

vSphere, платформа серверної віртуалізації від VMware, є набором продуктів, який включає не тільки віртуалізацію, а й рівні управління та інтерфейсів.[4]

Вона надає низку ключових компонентів, включаючи інфраструктурні сервіси (vCompute, vStorage та vNetwork), сервіси програм, vCenter Server, vSphere Client тощо.[4]

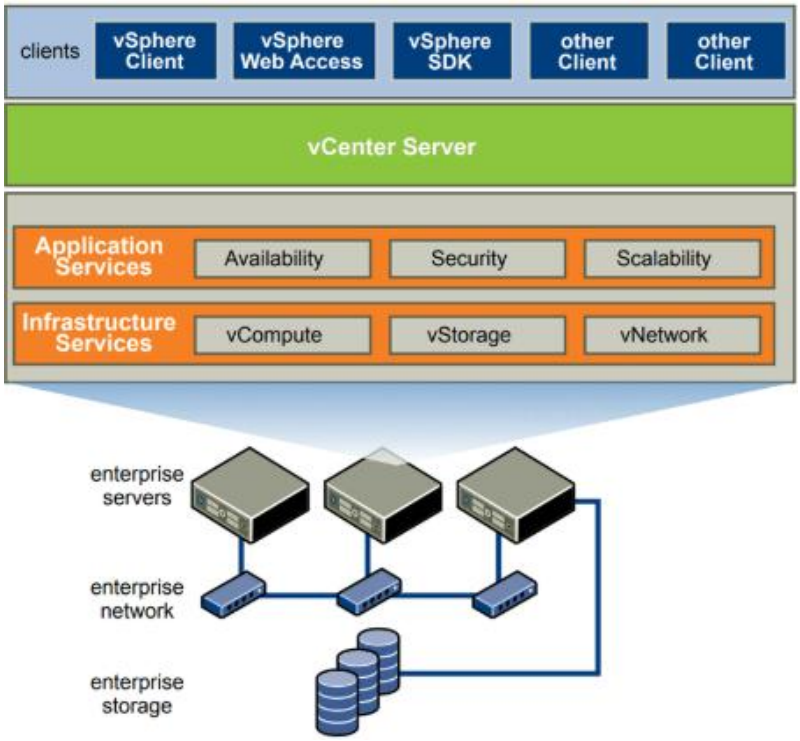


Рисунок 1.1 VMware VSphere

Функціонал VMware vSphere:

1) **vCenter Server:** інструмент централізованого керування, який використовується для налаштування, надання та керування віртуальними IT-середовищами.[4]
2) **vSphere Clien:** vSphere 6.7 має фінальну версію vSphere Web Client на основі Flash. Нові робочі процеси в оновленому випуску vSphere Client включають vSphere Update Manager, бібліотеку контенту, vSAN, політики зберігання, профілі хостів, схему топології VMware vSphere Distributed Switch та ліцензування.
3) **vSphere SDK:** надає інтерфейси сторонніх рішень для доступу до vSphere.
4) **VM File System:** кластерна файлова система для VM.
5) **Virtual SMP:** дозволяє одній віртуальній машині одночасно використовувати кілька фізичних процесорів.
6) **vMotion:** активна міграція із цілісністю транзакцій.
7) **Storage vMotion:** забезпечує міграцію файлів віртуальних машин з одного місця до іншого без переривання обслуговування.
8) **Висока доступність:** у разі збою одного сервера віртуальна машина переміщається на інший сервер із резервною ємністю для забезпечення безперервності бізнес-процесів.
   Планувальник розподілених ресурсів (DRS): автоматично призначає та балансує обчислення за апаратними ресурсами, доступними для віртуальних машин.
9) **Відмовостійкість:** створює копію основної віртуальної машини, щоб забезпечити її постійну доступність.[4]

10) **Розподілений комутатор (VDS):** охоплює кілька хостів ESXi та дозволяє значно скоротити обсяг робіт з обслуговування мережі.[4]

Hyper-V.

Запущений у 2008 році Microsoft Hyper-V допомагає у розширенні чи створенні приватного хмарного середовища. Він сприяє ефективному використанню обладнання, покращує безперервність бізнес-процесів, а також підвищує ефективність розробки та тестування.

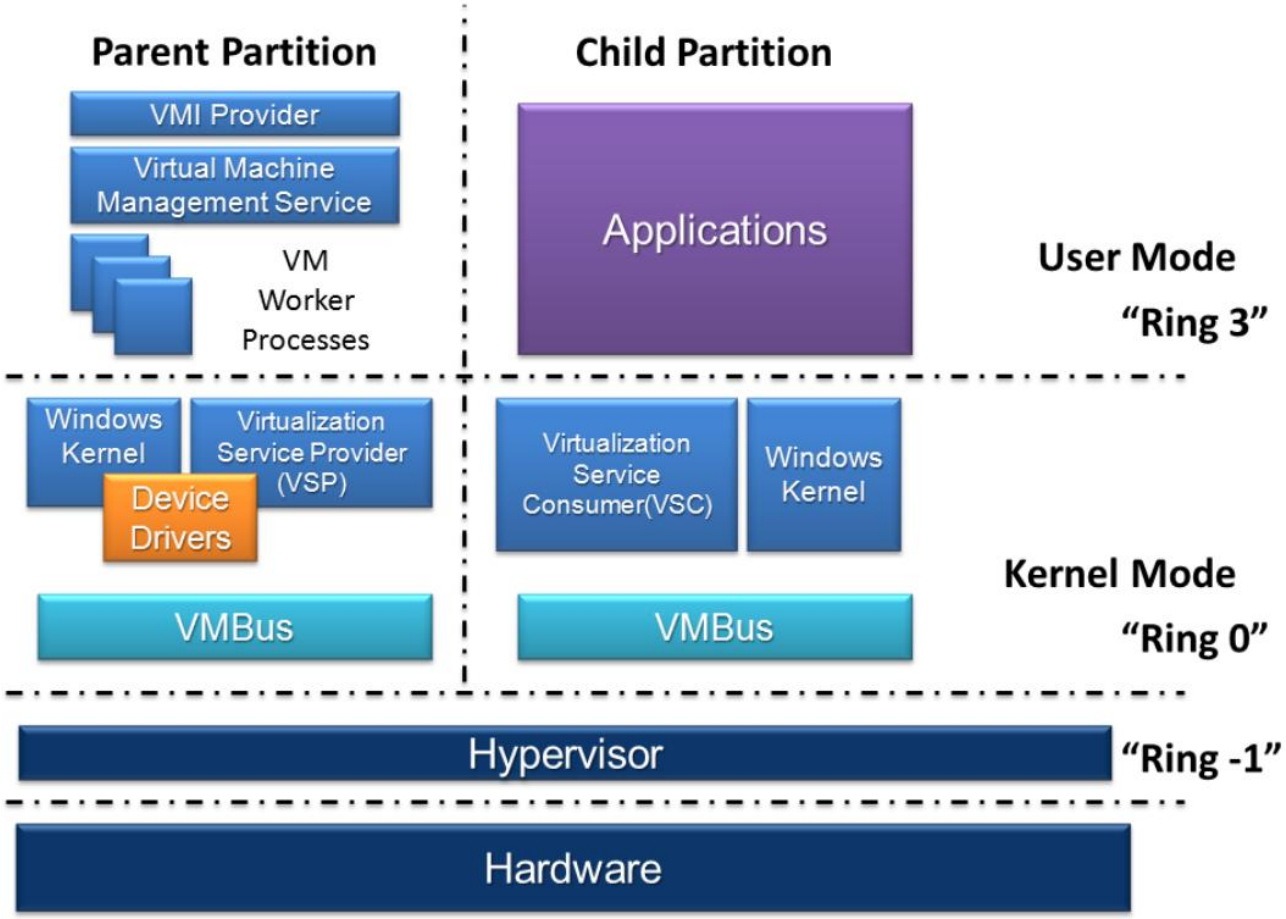


Рисунок 1.2 Hyper-V

Функціонал Microsoft Hyper-V для Windows Server 2019:

- Підтримка постійної пам'яті.
- Оновлення екранованих VM.
- Прості двовузлові кластери.
- Дедуплікація ReFS.
- Оптимізація локальних дискових просторів (Storage Spaces Direct)
- Центр адміністрування Windows.
- Зашифровані підмережі.[4]

XenServer.[4]

Заснований на Xen Project Hypervisor, XenServer є платформою віртуалізації серверів із відкритим вихідним кодом для платформ без операційної системи. Він складається з функцій корпоративного рівня, які допомагають підприємствам легко справлятися з робочими навантаженнями, комбінованими ОС та конфігураціями мережі.

XenServer забезпечує покращену віртуалізовану графіку з NIVIDA та Intel та дозволяє запускати кілька комп'ютерних операційних систем на одному обладнанні.

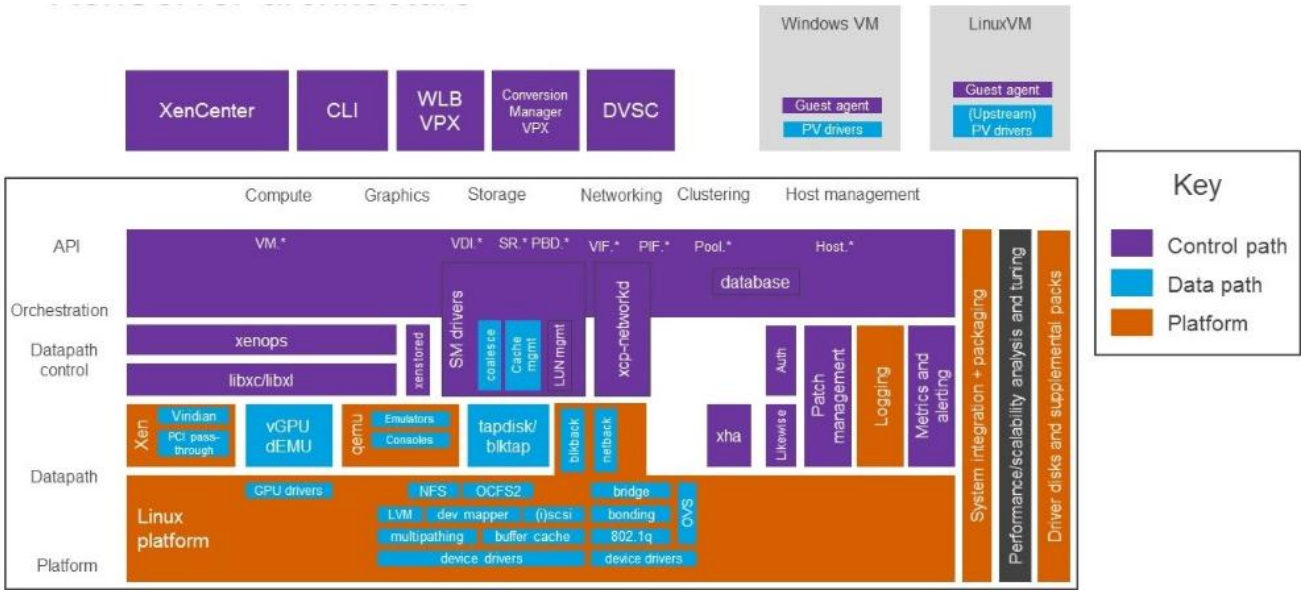


Рисунок 1.3 XenServer

Функціонал Citrix XenServer:

- Відновлення вузла
- Захист хоста від збоїв
- Мультисерверне керування
- Управління динамічною пам'яттю
- Інтеграція Active Directory
- Адміністрація та контроль на основі ролей (RBAC)
- Пули змішаних ресурсів із маскуванням ЦП
- Контролер розподіленого віртуального комутатора
- Вбудоване на згадку кешування операцій читання
- Жива міграція віртуальних машин та сховище XenMotion

KVM

KVM (Kernel-based Virtual Machine), що входить до складу Red Hat Virtualization Suite, є комплексним рішенням для інфраструктури віртуалізації. KVM перетворює ядро Linux на гіпервізор. Він був уведений в основну гілку ядра Linux з версії ядра 2.6.20.[4]

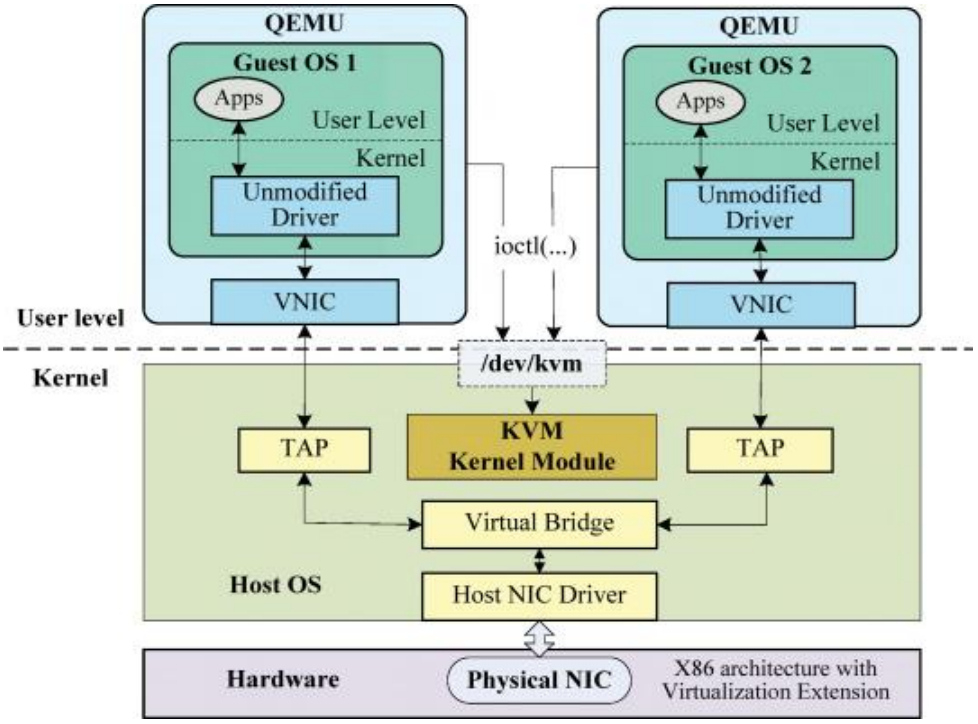


Рисунок 1.4 KVM (Kernel-based Virtual Machine)

Функціонал Red Hat[4] KVM:

- Підтримка контейнерів
- Масштабованість
- Overcommit ресурсів
- Disk I/O throttling
- Гаряча заміна віртуальних ресурсів
- Недороге рішення для віртуалізації
- Red Hat Enterprise Virtualization програмування та API
- Жива міграція та міграція сховища
- Призначення будь-яких PCI пристроїв віртуальним машинам
- Інтеграція Red Hat Satellite
- Підтримка відновлення після збою (Disaster Recovery)

Таблиця 1.1 Зведення по Hyper-V, vSphere, XenServer та KVM

Feature	Windows Hyper-V 2019	vSphere 6.7	XenServer 7.6	KVM
RAM/Host	24TB	12 TB	5TB	12TB
RAM/VM	12 TB for generation 2; 1 TB for generation 1	6 TB	1.5TB	6 TB
CPUs/VM	240 for generation 2; 64 for generation 1;	128	32	240
VM Disk	64 TB for VHDX format; 2040 GB for VHD format	62TB	2TB	10TB
VM Live Migration	Yes	Yes	Yes	Yes
VM Replication supports	Yes	Yes	Yes	Yes
Overcommit resources	No	Yes	No	Yes
Disk I/O Throttling	Yes	Yes	Yes	Yes
Hot plug of virtual resources	Yes	Yes	Yes	Yes

1.2.2 Порівняння технологій KVM та LXC у контексті продуктивності та безпеки

KVM (віртуальна машина на основі ядра) і LXC (контейнери Linux)[5] — це дві популярні технології віртуалізації, які використовуються для створення ізольованих середовищ і керування ними на одному фізичному хості. Ось основні відмінності між KVM і LXC:

1. **Тип віртуалізації:** KVM — це технологія віртуалізації на апаратному рівні, яка використовує гіпервізор для створення та керування віртуальними машинами (ВМ) із власними операційними системами та ядрами. З іншого боку, LXC — це метод віртуалізації на рівні ОС, який використовує функції контейнеризації Linux для запуску кількох ізольованих просторів користувачів на спільному ядрі.
2. **Рівень ізоляції:** KVM забезпечує надійну ізоляцію між віртуальними машинами, оскільки кожна віртуальна машина працює зі своїми власними віртуалізованими апаратними компонентами та працює як незалежна система. LXC пропонує нижчий рівень ізоляції, оскільки контейнери спільно використовують ядро хоста, що потенційно призводить до меншої ізоляції між програмами.[5]

3. **Використання ресурсів:** KVM споживає більше ресурсів, оскільки емулює повні апаратні компоненти та потребує окремих ядер для кожної віртуальної машини, що призводить до збільшення накладних ресурсів.[5] LXC є більш ресурсоефективним, оскільки контейнери спільно використовують ядро хоста та використовують менше ресурсів, що призводить до швидшого часу запуску та ефективного використання пам'яті.
4. **Продуктивність:** KVM забезпечує хорошу продуктивність і підходить для роботи з різними операційними системами та робочими навантаженнями. LXC забезпечує чудову продуктивність для додатків на базі Linux, оскільки дозволяє уникнути накладних витрат на емуляцію апаратних компонентів і роботу кількох ядер.
5. **Портативність:** KVM забезпечує кращу портативність завдяки своїй здатності запускати різні операційні системи та програми в ізольованих віртуальних машинах. Контейнери LXC більше підходять для програм, розроблених для роботи в середовищах Linux, що обмежує їх переносимість на інші операційні системи.
6. **Варіанти використання:** KVM підходить для сценаріїв, які вимагають повної ізоляції, сумісності з різними операційними системами та необхідності запускати застарілі програми. LXC ідеально підходить для легкої контейнеризації програм Linux, мікросервісів і хмарних середовищ, підкреслюючи ефективність і швидке розгортання.
7. **Екосистема та інструменти:** KVM підтримується різними інструментами та платформами керування, включаючи libvirt та інструменти, інтегровані в популярні платформи віртуалізації. LXC отримує переваги від багатої екосистеми інструментів керування контейнерами, як-от Docker, Kubernetes і фреймворків оркестровки контейнерів.
8. **Безпека:** KVM забезпечує кращий захист завдяки повній ізоляції між віртуальними машинами. Безпека LXC залежить від спільного ядра, що потенційно може викликати занепокоєння щодо безпеки під час розміщення ненадійних програм.
9. **Накладні витрати:** KVM створює більші накладні витрати через необхідність емулювати апаратні компоненти та запускати окремі ядра для кожної віртуальної машини. LXC має менші накладні витрати, оскільки він використовує ядро хоста та потребує менше ресурсів для контейнеризації.[5]

1.3 Система віртуалізації Proxmox VE: основні характеристики

1.3.1 Архітектура та ключові компоненти Proxmox VE

Системні вимоги:[6]

Для виробництва серверів необхідне високоякісне серверне обладнання. Proxmox VE підтримує кластеризацію, це означає, що кількома інсталяціями Proxmox VE можна централізовано керувати завдяки інтегрованій функціональності кластера. Proxmox VE може використовувати локальне сховище (DAS), SAN, NAS, а також спільне та розподілене сховище (Ceph).

Рекомендоване обладнання:

- Intel 64 або AMD64 із прапорцем процесора Intel VT/AMD-V.
- Пам'ять, мінімум 2 ГБ для ОС і служб Proxmox VE. Плюс виділена пам'ять для гостей. Для Ceph або ZFS потрібна додаткова пам’ять, приблизно 1 ГБ пам’яті на кожен використаний ТБ пам’яті.
- Швидке та резервне зберігання, найкращі результати з SSD-дисками.
- Зберігання ОС: апаратний RAID із кеш-пам’яттю запису, захищеним від батарей («BBU») або без RAID із кеш-пам’яттю ZFS і SSD.
- Сховище віртуальної машини: для локального зберігання використовуйте апаратний RAID з кеш-пам’яттю запису, що живиться від батареї (BBU) або без RAID для ZFS. Ані ZFS, ані Ceph несумісні з апаратним контролером RAID. Також можливе спільне та розподілене зберігання.
- Також підтримуються резервні мережеві карти Гбіт, додаткові мережеві карти залежно від бажаної технології зберігання та налаштування кластера – 10 Гбіт і вище.
- Для проходження PCI(e) потрібен ЦП із прапором ЦП VT-d/AMD-d.

Мінімальне обладнання (лише для тестування):

- ЦП: 64 біт (Intel 64 або AMD64)
- Процесор/материнська плата з підтримкою Intel VT/AMD-V (для повної підтримки віртуалізації KVM)
- Мінімум 1 Гб оперативної пам'яті
- Жорсткий диск
- Один мережевий адаптер

Тестування з віртуалізацією робочого столу:

Proxmox VE можна встановити як гостьову систему на всіх поширених рішеннях для віртуалізації робочого столу, якщо вони підтримують вкладену віртуалізацію.

Підтримувані веб-браузери для доступу до веб-інтерфейсу:

Для використання веб-інтерфейсу вам потрібен сучасний браузер, який включає:[6]

- Firefox,[6] випуск поточного року або останній випуск розширеної підтримки;
- Chrome, випуск з поточного року
- Microsoft наразі підтримує версію Edge
- Safari, випуск поточного року

1.3.2 Підтримка KVM і LXC

Proxmox VE підтримує як віртуалізацію на базі KVM (повна емуляція апаратного забезпечення), так і LXC контейнери (легковагі віртуальні машини, засновані на ядрі Linux). Веб-інтерфейс: Proxmox VE має зручний веб-інтерфейс, який дає змогу вам керувати всіма аспектами вашого віртуального середовища.

Таким чином, KVM забезпечує надійну ізоляцію та сумісність, що робить його придатним для різноманітних робочих навантажень, тоді як LXC вирізняється ефективністю та швидкістю для програм на базі Linux.[6]

1.3.3 Proxmox VE: Віртуалізація корпоративного рівня

Proxmox VE розроблений для центрів обробки даних і підприємств, яким потрібні надійні, масштабовані та ефективні віртуальні середовища.[7] Створений для широкомасштабного розгортання, він забезпечує інтегровану платформу для легкого керування віртуальними машинами та контейнерами.

Основні функції корпоративного рівня:

- Віртуалізація KVM і LXC : Proxmox VE підтримує як віртуальні машини на основі ядра (KVM), так і контейнери Linux (LXC), що дозволяє поєднувати повну та спрощену віртуалізацію.
- Управління кластерами : Proxmox дозволяє створювати кластери для централізованого керування кількома серверами, підтримуючи масштабованість і розподіл робочого навантаження між центрами обробки даних.
- Висока доступність (HA) : за допомогою HA віртуальні машини автоматично перезапускаються на інших вузлах кластера у разі збою обладнання, забезпечуючи мінімальний час простою.
- Жива міграція : віртуальні машини можна переміщувати між фізичними серверами без переривання операцій, що робить технічне обслуговування безперешкодним.
- Інтеграція зі сховищем : Proxmox підтримує широкий спектр систем зберігання, включаючи локальне сховище, NFS, iSCSI, Ceph і ZFS.
- Резервне копіювання та відновлення : інтегровані рішення для резервного копіювання, включаючи резервне копіювання на основі знімків, дозволяють легко відновлювати дані у разі збою системи.[7]

1.4 Порівняльний аналіз Proxmox VE з іншою системою VirtualBox

Таблиця 1.2 Порівняння функцій: Proxmox проти VirtualBox[6]

Особливість	Proxmox VE	VirtualBox
Цільова аудиторія	Підприємства, ЦОД	Індивідуальні користувачі, розробники
Тип віртуалізації	KVM (повна віртуалізація), LXC (контейнерізація)	Повна віртуалізація
Підтримка кластерів	Має централізоване керування вузлів	немає
Висока доступність (HA)	Вбудовано	немає
Жива міграція	Підтримується	немає
Параметри зберігання	Локальний, NFS, iSCSI, Ceph, ZFS	Локальне сховище, спільні папки
Рішення для резервного копіювання	Інтегрована система резервного копіювання та знімків	Підтримка знімків
Кросплатформна підтримка	Linux (як хост), підтримує кілька гостьових ОС	Windows, macOS, Linux (як хост, так і гостьова ОС)
Інтерфейс користувача	Веб-консоль керування	Графічний інтерфейс користувача (GUI)
Випадок використання	Масштабна віртуалізація серверів	Віртуалізація робочого столу, розробка, тестування
Вартість	Безкоштовний і з відкритим кодом (корпоративна підтримка необов’язкова)	Безкоштовний і з відкритим кодом

1.4.1 Порівняльний аналіз продуктивності у реальному часі

Тести реального світу дають нам чіткішу картину того, як Proxmox VE та VirtualBox працюють за різних сценаріїв.[6] Різноманітні тести висвітлили наступне:

- **Витрати на ЦП і пам’ять** : у завданнях, що інтенсивно використовують ЦП, і в управлінні пам’яттю Proxmox демонструє явні переваги завдяки використанню KVM, який забезпечує апаратно-прискорену віртуалізацію. VirtualBox, як правило, має більші накладні витрати, особливо при одночасному запуску кількох віртуальних машин.
- **Продуктивність вводу-виводу** : у тестах, зосереджених на дисковому вводі-виводі, Proxmox значно перевершує VirtualBox, особливо при використанні передових систем зберігання, таких як ZFS або Ceph. VirtualBox, з іншого боку, може відчувати сповільнення в додатках, що інтенсивно займаються вводом-виводом, через більш обмежені параметри серверної системи зберігання даних.
- **Продуктивність мережі** : Proxmox забезпечує кращу продуктивність мережі, особливо в середовищах з кількома віртуальними машинами, які спілкуються через віртуальну мережу. Незважаючи на те, що VirtualBox функціонує, він має більш обмежені можливості віртуальної мережі та може мати більшу затримку під час інтенсивних мережевих операцій.
- **Масштабованість** : Proxmox VE блищить у великомасштабних розгортаннях, ефективно керуючи десятками віртуальних машин або контейнерів на кількох серверах. Через свою орієнтовану на робочий стіл природу VirtualBox має проблеми з масштабованістю, особливо під час роботи з ресурсомісткими програмами або великою кількістю віртуальних машин.

1.4.2 Керування та обслуговування

Після встановлення і Proxmox, і VirtualBox пропонують різні можливості керування та обслуговування, обслуговуючи відповідні бази користувачів.[7]
Керування Proxmox VE:

- **Веб-інтерфейс керування** : Proxmox пропонує потужну консоль керування на основі браузера, де адміністратори можуть контролювати всі аспекти віртуальних машин, контейнерів, сховища та мереж.
- **Централізоване керування кластерами** : для великих середовищ керування кластерами Proxmox дозволяє адміністраторам контролювати декілька серверів з однієї інформаційної панелі, включаючи оперативну міграцію, високу доступність і моніторинг ресурсів.
- **Автоматизація та створення сценаріїв** : Proxmox підтримує інструменти командного рядка та API, що робить його ідеальним для адміністраторів, яким потрібно автоматизувати завдання або інтегрувати Proxmox у більшу IT-інфраструктуру.
- **Обслуговування** : Proxmox надає вбудовані рішення для резервного копіювання, керування знімками та інструменти моніторингу для забезпечення безперебійної роботи. Системні оновлення та виправлення безпеки можна застосовувати через веб-інтерфейс, мінімізуючи час простою.[7]

Управління VirtualBox[7]:

- **Графічний інтерфейс користувача** : графічний інтерфейс VirtualBox інтуїтивно зрозумілий, що дозволяє користувачам легко керувати віртуальними машинами. Основні завдання, такі як запуск, призупинення або видалення віртуальних машин, можна виконати лише кількома клацаннями миші.
- **Невеликі вимоги до обслуговування** : Оскільки VirtualBox розроблено для використання на робочому столі, він потребує мінімального обслуговування. Оновлення обробляються автоматично, а резервне копіювання можна виконувати за допомогою інструментів сторонніх розробників або створення знімка вручну.
- **Налаштування на рівні користувача** : Хоча у нього відсутні розширені інструменти автоматизації Proxmox, VirtualBox дозволяє користувачам налаштовувати віртуальні машини за допомогою таких функцій, як мережеві мости, підтримка USB і спільні папки, і все це без потреби в значних технічних навичках.[7]

1.4.3 Аналіз витрат і вигод

Розглядаючи, яка платформа забезпечує найкращу цінність, це значною мірою залежить від вашого випадку використання, розміру інфраструктури та вимог до підтримки.

Ціннісна пропозиція Proxmox VE:

- **Безкоштовно для основного використання** : здатність Proxmox пропонувати такі функції корпоративного рівня, як кластеризація, висока доступність і оперативна міграція без ліцензійних зборів, робить його високорентабельним рішенням для компаній, які хочуть керувати великими віртуальними інфраструктурами.
- **Гнучкість підписки** : Proxmox пропонує гнучку модель підписки, яка дозволяє компаніям платити лише за той рівень підтримки, який їм потрібен. Ця гнучкість може бути корисною для невеликих організацій або стартапів, які можуть не потребувати постійної підтримки, але все одно потребують корпоративних функцій.
- **Довгострокова економія** : організації, яким потрібен повний контроль над своєю інфраструктурою без повторних платежів за кожну віртуальну машину, можуть вважати Proxmox більш фінансово стабільним вибором з часом, особливо для великомасштабного розгортання.

Ціннісна пропозиція VirtualBox:

- **Повністю безкоштовно для особистого користування** : VirtualBox — це найбільш економічно ефективне рішення для особистих проектів, розробників і невеликих команд, яким не потрібні широкі функції чи підтримка на рівні підприємства. Це особливо цінно для людей, які використовують віртуальні машини на настільних комп’ютерах і ноутбуках.

- **Низька вартість для корпоративних функцій** : навіть якщо додати вартість Oracle Extension Pack, VirtualBox залишається доступним варіантом для підприємств, яким потрібні деякі розширені функції, але не потрібен повний пакет підтримки на рівні підприємства.
- **Просте ліцензування** : безкоштовна доступність для особистого використання з додатковим комерційним ліцензуванням для компаній робить VirtualBox простим і гнучким вибором для різних користувачів.

1.4.4 Безпека

Безпека є головною проблемою при виборі платформи віртуалізації, оскільки вразливі місця можуть піддавати ризикам віртуальні машини, контейнери та навіть хост-систему.[7] Proxmox VE і VirtualBox пропонують надійні функції безпеки, але вони підходять для різних середовищ і випадків використання. У цьому розділі оцінюються функції безпеки кожної платформи, порівнюються їхні заходи безпеки та досліджується, як вони обробляють уразливості та виправлення.

Функції безпеки в Proxmox VE.

Proxmox VE розроблено для використання на корпоративному рівні, що означає, що він містить розширені функції безпеки, спрямовані на захист великомасштабних розгортань. Зосереджуючись на багатокористувацьких середовищах і критичних інфраструктурах, Proxmox наголошує на контролі доступу на основі ролей (RBAC), безпеці мережі та безпечному управлінні.

Основні функції безпеки в Proxmox VE:

- **Контроль доступу на основі ролей (RBAC)** : Proxmox дозволяє адміністраторам призначати певні ролі та дозволи користувачам, забезпечуючи обмеження доступу до критичних функцій і віртуальних машин відповідно до посадових обов’язків. Це допомагає запобігти несанкціонованому доступу та підтримує контроль над середовищем.
- **Двофакторна автентифікація (2FA)** : для додаткової безпеки Proxmox підтримує двофакторну автентифікацію для входу користувачів. Адміністратори можуть увімкнути це для всіх користувачів, щоб лише авторизований персонал мав доступ до інтерфейсу керування.
- **Зашифровані резервні копії** : Proxmox пропонує можливість шифрувати резервні копії, гарантуючи, що критично важливі дані залишаються захищеними навіть під час зберігання або передачі.
- **Брандмауер та ізоляція мережі** : Proxmox містить інтегровані правила брандмауера, які можна застосовувати до окремих віртуальних машин або контейнерів. Ця функція дозволяє сегментувати мережу, зменшуючи поверхню атаки за рахунок обмеження впливу віртуальних машин на несанкціонований мережевий трафік.[7]

- **Захищений веб-інтерфейс** : консоль керування Proxmox працює через HTTPS, гарантуючи, що зв'язок між адміністратором і сервером зашифрований.[7] Доступ SSH для розширеного керування також захищено за допомогою автентифікації з відкритим ключем.
- **Автоматичні оновлення безпеки** : Proxmox надає регулярні виправлення безпеки через свої репозиторії, особливо для платних передплатників, які використовують корпоративне сховище. Це допомагає організаціям залишатися захищеними від відомих вразливостей.

Функції безпеки Proxmox роблять його особливо придатним для компаній, які керують конфіденційними даними, кількома користувачами та великою віртуальною інфраструктурою.

Заходи безпеки VirtualBox.

VirtualBox в основному розроблений для настільного комп'ютера та особистого використання, але він все ще містить кілька функцій безпеки, спрямованих на захист окремих віртуальних машин і хост-системи. Незважаючи на те, що він не пропонує параметрів безпеки корпоративного рівня Proxmox, VirtualBox зосереджується на забезпеченні цілісності віртуалізованих середовищ у більш легкий і зручний спосіб.

Основні функції безпеки у VirtualBox:

- **Ізольовані віртуальні середовища** : VirtualBox ізолює гостьові віртуальні машини від операційної системи хоста, зменшуючи ризик шкідливого програмного забезпечення або вразливостей у віртуальній машині, що впливає на хост.
- **Безпека гостьових доповнень** : VirtualBox надає гостьові додатки , які покращують інтеграцію віртуальних машин із хост-системою. Ці інструменти включають такі функції, як обмін буфером обміну та доступ до файлів, але їх можна налаштувати для обмеження рівня інтеграції, що зменшує ризик атак між середовищами.
- **Шифрування диска** : за допомогою пакета розширень VirtualBox користувачі можуть шифрувати жорсткий диск віртуальної машини, забезпечуючи безпеку конфіденційних даних, що зберігаються у віртуальній машині.
- **Контроль USB-пристроїв** : VirtualBox дозволяє адміністраторам контролювати доступ до USB-пристроїв для віртуальних машин, обмежуючи можливість потенційно шкідливих пристроїв взаємодіяти з гостьовою ОС.
- **Мережева безпека** : VirtualBox включає кілька мережевих режимів, таких як Host-Only і Internal Networking , які обмежують доступ віртуальної машини до зовнішніх мереж, забезпечуючи додаткові рівні ізоляції мережі.
- **Безпечне завантаження та UEFI** : VirtualBox підтримує безпечне завантаження UEFI, дозволяючи більш безпечно ініціалізувати гостьові віртуальні машини, особливо під час роботи сучасних операційних систем, які використовують безпечне завантаження як механізм захисту.[7]

Заходи безпеки VirtualBox ідеально підходять для користувачів, яким потрібно запускати кілька ізольованих віртуальних середовищ на своїх персональних комп'ютерах або машинах розробки з обмеженим ризиком для хост-системи.

1.4.5 Вразливості та виправлення: критичний погляд

Протягом багатьох років і Proxmox VE, і VirtualBox зазнавали вразливостей системи безпеки, але те, як кожна платформа усуває ці вразливості та надає виправлення, відіграє важливу роль у підтримці довіри користувачів і довгостроковій безпеці.

Вразливості Proxmox VE та виправлення:

- **Регулярні оновлення** : Proxmox активно випускає оновлення для усунення вразливостей. Для користувачів, які підписалися на корпоративне сховище, виправлення доставляються часто, гарантуючи безпеку робочого середовища.
- **Виправлення спільноти** : відкритий вихідний код Proxmox означає, що спільнота бере активну участь у виявленні вразливостей і звітуванні про них. Отримавши повідомлення, виправлення часто розгортаються швидко, а команда Proxmox надає оновлення через свої загальнодоступні сховища.
- **Реагування на критичні вразливості** : Proxmox відомий тим, що швидко усуває критичні вразливості, особливо в корпоративних середовищах, де простої або порушення безпеки можуть мати значні фінансові наслідки.

Вразливості та виправлення VirtualBox:

- **Цикл виправлення Oracle** : як частина екосистеми Oracle, VirtualBox отримує переваги від комплексного процесу виправлення Oracle. Уразливості регулярно усуваються, Oracle щоквартально випускає виправлення безпеки. Однак користувачі безкоштовної версії повинні активно відстежувати та застосовувати оновлення вручну, оскільки для некомерційних версій немає механізму автоматичного оновлення.
- **Внески з відкритим кодом** : як і Proxmox, спільнота з відкритим кодом сприяє виявленню та виправленню помилок і проблем безпеки. Ці спільні зусилля гарантують, що вразливі місця зазвичай усуваються швидко.
- **Уразливості пакета розширень** : Оскільки пакет розширень VirtualBox додає кілька розширених функцій, у минулому він був центром уразливостей. Організації, які використовують ці функції, повинні ретельно встановлювати виправлення, щоб забезпечити постійну безпеку.

Порівняння реакції безпеки:

- **Proxmox фокусується на підприємстві** : Proxmox приділяє значну увагу безпеці на корпоративному рівні з автоматичними оновленнями та надійним механізмом виправлення, доступним для платних передплатників. Для критично важливих середовищ цей підхід забезпечує душевний спокій.

- **VirtualBox Desktop Focus** : VirtualBox покладається на стандартний графік виправлення Oracle, який у деяких випадках може бути повільнішим, особливо для некорпоративних користувачів. Однак для персональних середовищ або середовищ розробки механізму виправлення зазвичай достатньо.

1.5 Актуальність впровадження Proxmox VE в Україні

Впровадження Proxmox VE в Україні сприяє зниженню витрат, підвищенню безпеки та гнучкості, що робить цю систему актуальною для бізнесу, держсектору та військової сфери.

1. **Економічна ефективність та імпортозаміщення**
   - Українські компанії та держустанови прагнуть зменшити залежність від дорогих комерційних рішень (VMware, Microsoft Hyper-V).
   - Proxmox VE є безкоштовною системою з відкритим кодом, що дозволяє значно скоротити витрати на ліцензії та обслуговування.
   - **Приклад**: Державні організації можуть мігрувати на Proxmox VE, зменшуючи витрати на ліцензії VMware, кошти яких можна спрямувати на інші потреби.
2. **Кібербезпека та контроль над даними**
   - Через війну та зростаючі кібератаки критично важливо контролювати IT-інфраструктуру та мінімізувати ризики.
   - Proxmox VE дозволяє локально розміщувати сервери без залежності від західних комерційних рішень та ризику санкцій або обмежень.
   - **Приклад**: Банки та держсектор можуть уникнути потенційних проблем із санкціями, розміщуючи критичні дані на власних серверах.
3. **Гнучкість та масштабованість**
   - Підприємства можуть швидко розгортати нові віртуальні сервери без додаткових витрат на обладнання.
   - Proxmox VE підтримує кластеризацію, що дає змогу створювати масштабовані системи для великих компаній.
   - **Приклад**: IT-компанії, які займаються хмарними рішеннями, можуть використовувати Proxmox VE для гнучкого керування навантаженнями без великих капіталовкладень.
4. **Підтримка open-source спільноти та адаптація під локальні потреби**
   - Українські IT-фахівці можуть змінювати та вдосконалювати систему, додаючи локалізовані функції та підтримку української мови.
   - **Приклад**: Університети та наукові установи можуть використовувати Proxmox VE для навчальних цілей та створення власних серверних інфраструктур.
5. **Використання у військовій та оборонній сфері**
   - Військові об’єкти та оборонні підприємства потребують надійної серверної інфраструктури без ризику витоку даних через комерційні рішення.

- **Приклад**: Використання Proxmox VE у польових дата-центрах для керування безпілотниками, системами зв’язку чи аналітики даних.

Висновок за розділом

У даному розділі було було наведено аналіз технологій віртуалізації та особливостей використання Proxmox VE для керування серверною інфраструктурою. Наведені сучасні тенденції віртуалізації. Порівняно різні системи керування віртуальними машинами та розглянуто особливості архітектури Proxmox VE.

Proxmox VE є потужним інструментом, який поєднує можливості контейнерної (LXC) та апаратної (KVM) віртуалізації, забезпечуючи високу гнучкість і ефективність використання ресурсів. Визначено ключові переваги цієї системи, зокрема її відкритий код, централізоване управління, вбудовані механізми резервного копіювання та можливість створення кластерів.

Проведено порівняльний аналіз Proxmox VE з іншими системами віртуалізації, що дозволило виділити її конкурентні переваги у контексті продуктивності, масштабованості та економічної ефективності. Особливу увагу приділено актуальності впровадження цієї технології в Україні, зокрема у сфері державного управління, бізнесу та військових структур.

Використання Proxmox VE є доцільним для організацій, які прагнуть оптимізувати витрати на IT-інфраструктуру, підвищити рівень безпеки та забезпечити стабільну роботу серверного середовища. Подальші дослідження можуть бути спрямовані на практичне тестування продуктивності цієї платформи у різних сценаріях експлуатації.

2. АПАРАТНА ТА ПРОГРАМНА ЧАСТИНИ СЕРВЕРНОЇ МЕРЕЖІ PROXMOX VE

2.1 Економічне обґрунтування вибору Proxmox VE

Таблиця 2.1. Порівняння витрат на різні системи віртуалізації

Параметр	VMware vSphere	Microsoft Hyper-V	Proxmox VE
Вартість ліцензії, $	5000	3000	0
Вартість підтримки, $/рік	800	500	0
Вартість підтримки, $/рік	платні	частково безкоштовні	безкоштовні
Середня витрата електроенергії, $/рік	1500	1500	1500
Загальна вартість за 3 роки, $	9900	6000	4500

Таблиця 2.2. Розрахунок окупності (ROI) впровадження Proxmox VE

Показник	Значення
Початкові інвестиції, $	3000
Зекономлені витрати на ліцензії за 3 роки, $	5000
Чистий прибуток (економія) за 3 роки, $	2000
ROI (%)	66,7%

Формула ROI:

$$ROI = \frac{\text{Початкові інвестиції}}{\text{Чистий прибуток}} \times 100\% \qquad (2.1)$$

2.2 Вибір серверного обладнання та програмного забезпечення під Proxmox

Підтримка обладнання:

CPU: Intel 64-bit, AMD64 (з апаратною підтримкою віртуалізації VT-x/AMD-V).

RAM: мінімум 2 ГБ для базового запуску (рекомендовано від 8 ГБ і більше для серверних задач).

Сховище: локальні диски (HDD/SSD), мережеві сховища (NFS, iSCSI, Ceph).

Мережа: Ethernet 1 Gbit/s, рекомендується 10 Gbit/s для великих кластерів.

Причини вибору саме Proxmox VE 8.4 ISO:

1) Остання стабільна версія (реліз квітень 2024 року) із повною підтримкою актуальних оновлень безпеки та нових функцій.

2) Безкоштовна ліцензія на основні можливості + можливість купити платну підтримку за бажанням.

3) Інтеграція KVM і LXC для одночасної роботи повноцінних віртуальних машин і легких контейнерів.

4) Підтримка кластеризації — дає змогу об'єднувати кілька фізичних серверів для високої доступності сервісів.

5) Підтримка резервного копіювання, міграції без простою, живого переміщення віртуальних машин.

6) Веб-інтерфейс керування дозволяє керувати усією інфраструктурою через браузер без встановлення додаткового ПЗ.

Для побудови кластерної системи віртуалізації на базі Proxmox VE 8.4 обрано три пристрої (ноутбуки), що відповідають мінімальним та рекомендованим вимогам для роботи в тестовому середовищі.

2.2.1 Основний вузол кластера — Acer Aspire 7

Технічні характеристики:

- Процесор: AMD Ryzen 5 5500U (6 ядер / 12 потоків, базова частота 2.1 ГГц, підвищується до 4.0 ГГц).
- Оперативна пам'ять (ОЗУ): 24 ГБ DDR4.
- Накопичувач: SSD NVMe, 512 ГБ.
- Відеокарта: NVIDIA GeForce RTX 3050 Ti 4 ГБ GDDR6.
- Підтримка віртуалізації: AMD-V (апаратна підтримка віртуалізації ввімкнена у BIOS).

Обґрунтування вибору:

Acer Aspire 7 має високопродуктивний багатоядерний процесор, що забезпечує плавну роботу декількох віртуальних машин одночасно. Великий об’єм оперативної пам’яті дозволяє запускати одночасно до 5–7 віртуальних машин середнього розміру або більше контейнерів LXC. SSD-накопичувач забезпечує високу швидкість доступу до даних і прискорене завантаження віртуальних середовищ.

Функції в кластері:

- Основний вузол керування кластером Proxmox VE.
- Розміщення найбільш ресурсомістких віртуальних машин (наприклад, бази даних або додатків ERP).
- Виконання централізованого моніторингу та адміністрування кластеру.

2.2.2 Продуктивний вузол для балансування навантаження — Asus TUF Gaming F17

Технічні характеристики:

- Процесор: Intel Core i5-11400 (6 ядер / 12 потоків, базова частота 2.6 ГГц, Boost до 4.4 ГГц).
- Оперативна пам'ять (ОЗУ): 16 ГБ DDR4.
- Накопичувач: SSD NVMe, 900 ГБ.
- Відеокарта: NVIDIA GeForce RTX 3050 Ti 4 ГБ GDDR6.
- Підтримка віртуалізації: Intel VT-x, VT-d (віртуалізація ввімкнена у BIOS).

Обґрунтування вибору:

Asus TUF Gaming F17 має сучасний процесор Intel одинадцятого покоління із високою тактовою частотою, що дозволяє ефективно обслуговувати віртуальні машини із підвищеними вимогами до обчислювальних ресурсів. Обсяг пам'яті у 16 ГБ є достатнім для одночасної роботи 3–5 ВМ або великої кількості контейнерів. Ємкий SSD дає можливість зберігати велику кількість даних і робити резервні копії локально.

Функції в кластері:

- Балансування навантаження між основним вузлом та другим вузлом.
- Підтримка високої доступності віртуальних машин.
- Резервне розміщення важливих сервісів при відмові основного вузла.

2.2.3 Легкий вузол для контейнерних рішень — Acer Aspire 3

Технічні характеристики:

- Процесор: Intel Core i3 (покоління 7).
- Оперативна пам'ять (ОЗУ): 4 ГБ DDR4.
- Накопичувач: SSD 500 ГБ.
- Відеоадаптер: інтегрована графіка Intel UHD.
- Підтримка віртуалізації: Intel VT-x.

Обґрунтування вибору:

Acer Aspire 3 має базовий набір характеристик, який обмежує можливості повноцінної віртуалізації на основі KVM через недостатній обсяг оперативної пам'яті. Проте для контейнерної віртуалізації LXC цей пристрій є цілком придатним, оскільки контейнери споживають менше ресурсів і можуть працювати на спільному ядрі операційної системи.

Функції в кластері:

- Розгортання легких сервісів у контейнерах (наприклад, моніторинг, syslog-сервери, базові веб-додатки).
- Створення локальних резервних копій контейнерів і даних.
- Використання як додаткового вузла в кластері для підтримки гнучкості архітектури.

Загальна побудова кластера:

- Node 1 (Acer Aspire 7) — керування кластером, головні ВМ.
- Node 2 (Asus TUF Gaming F17) — балансування і резервування навантаження.
- Node 3 (Acer Aspire 3) — легкі сервіси в контейнерах.

2.2.4 Детально про Proxmox VE версії 8.4

Випущено 9 квітня 2025 р.[8]:

- На основі Debian Bookworm (12.10)
- Останнє ядро 6.8.12-9 як нове стабільне ядро за умовчанням
- Новіше ядро 6.14 як опція
- QEMU 9.2.0
- LXC: 6.0.0
- ZFS: 2.2.7 (з патчами сумісності для ядра 6.14)
- Ceph Squid 19.2.1
- Риф Цеф 18.2.4
- Термін експлуатації Ceph Quincy 17.2.8 закінчився, користувачам рекомендується оновити

Основні моменти:

- Жива міграція з посередницькими пристроями, такими як NVIDIA vGPU.

Це дозволяє мігрувати запущені гостьові віртуальні машини, які використовують посередницькі пристрої.

Потрібна підтримка обладнання та драйверів.

Наразі лише графічні процесори NVIDIA підтримують оперативну міграцію.

- Підтримка зовнішніх постачальників резервного копіювання.

Proxmox VE тепер надає API, який дозволяє розробникам писати плагіни провайдера резервного копіювання.

Плагін постачальника резервних копій може реалізувати функції резервного копіювання та відновлення для зовнішніх рішень резервного копіювання.

Таким чином, зовнішні рішення для резервного копіювання можуть бути повністю інтегровані зі стеком резервного копіювання Proxmox VE та графічним інтерфейсом користувача.

- Спільне використання каталогів хоста з гостьовими віртуальними машинами за допомогою virtiofs.

Virtiofs дозволяє ділитися каталогами між хостом Proxmox VE і віртуальними машинами без накладних витрат на мережеву файлову систему.
Сучасні гостьові системи Linux підтримують virtiofs із коробки.
Гості Windows повинні інсталювати додаткове програмне забезпечення.

- Останнє ядро Linux 6.14 доступне як опційне ядро.
- Ceph Squid 19.2.1 доступний як стабільний варіант.
- Плавне оновлення з Proxmox VE 7.4

Огляд журналу змін:

Покращення веб-інтерфейсу (GUI):

- Дозволити встановити банер згоди, який користувачі повинні підтвердити перед входом (проблема 5463) .[8]

Це може знадобитися з міркувань відповідності та вже підтримується Proxmox Backup Server.[8]

Банер підтримує Markdown і його можна налаштувати в Datacenter → Options → Consent Text.

- Сортуйте вміст зберігання, як-от файли ISO, відповідно до мови браузера та числового сортування (проблема 6138).

Це призводить до більш природного порядку сортування щодо великих літер і чисел у назвах файлів.

- Завантаження шаблонів ISO, CT або пристроїв OVA з URL-адрес тепер використовує налаштований проксі-сервер як для з'єднань HTTP, так і для HTTPS (проблема 3716).
- Мережу міграції в параметрах центру обробки даних тепер можна ввести за допомогою спеціального CIDR (проблема 6142).
- У списках завдань тепер відображається додатковий стовпець дій, який відкриває деталі завдання для кращої видимості.
- У діалогових вікнах підтвердження тепер згадується ім'я гостя, щоб було зрозуміліше, для якого гостя буде застосовано дію (проблема 3181).
- Краще узгоджуйте перевірки привілеїв для додавання пристроїв PCI, USB або VirtIO RNG у веб-інтерфейсі з фактичними перевірками привілеїв у серверній частині.
- Дозволити завантажувати та завантажувати образи дисків у сховища з типом вмісту «Імпорт» для підготовки до імпорту образів дисків у віртуальні машини в майбутньому (проблема 2424).
- Вирішено проблему, через яку натискання зовнішнього посилання на GUI відображало екран входу, навіть якщо поточний сеанс все ще був дійсним.
- Виправлено проблему, через яку редактор зіставлення PCI попередньо вибирав неправильний перемикач.
- Виправлено проблему безпеки, яка дозволяла XSS через певні відповіді гостьового агента QEMU (PSA-2024-00016-1).
- Результати HTML-кодування API перед відтворенням як додатковий захист від XSS (PSA-2025-00002-1).
- Різні менші вдосконалення графічного інтерфейсу.
- Оновіть xterm.js до версії 5.5.0.
- Вирішено проблему, через яку консоль xterm.js мала правильний розмір під час з'єднань із високою затримкою (проблема 6223).
- Оновіть noVNC до версії 1.6.0.
- Виправлено деякі випадки, коли рядки не були позначені як перекладні.
- Виправлено деякі випадки, коли рядки, які можна перекладати, були розділені, що робило потенційно корисний контекст недоступним для перекладачів.
- Покращені переклади, серед іншого:[8]
    1) Болгарська
    2) Французька

3) Німецький
4) Італійська
5) Японський
6) Спрощена китайська
7) Іспанська
8) Традиційна китайська
9) Українська

Віртуальні машини (KVM/QEMU):[8]

- Нова версія QEMU 9.2.0
- Жива міграція з посередницькими пристроями, такими як NVIDIA vGPU (проблема 5175).

Це дозволяє мігрувати запущені гостьові віртуальні машини, які використовують посередницькі пристрої.

Його можна ввімкнути на рівні зіставлення PCI, позначивши пристрій як здатний до живої міграції.

Потрібна підтримка обладнання та драйверів.

Наразі лише графічні процесори NVIDIA підтримують оперативну міграцію.

- Спільне використання каталогів хоста з гостьовими віртуальними машинами за допомогою virtiofs (проблема 1027).

Virtiofs дозволяє обмінюватися файлами між хостом і гостем без залучення мережі.

Підтримка драйверів у гостьовій системі є обов'язковою та реалізована «з коробки» сучасними гостьовими системами Linux із ядром 5.4 і вище.

Гості Windows повинні інсталювати додаткове програмне забезпечення для використання virtiofs.

Віртуальні машини, які використовують virtiof, не можна перенести в реальному часі. Миттєві знімки з оперативною пам'яттю та сплячим режимом неможливі.

- Початкова підтримка AMD SEV-SNP (Secure Nested Paging).

У підтримуваних налаштуваннях SEV-SNP може додатково посилити ізоляцію між хостом і гостем.

Оскільки диски EFI не підтримуються під час використання SEV-SNP, надрукуйте попередження про те, що диски EFI ігноруються.

- Вимкнути стани живлення S3/S4 за замовчуванням.

Відомо, що ці стани живлення спричиняють проблеми в деяких налаштуваннях, наприклад із пропуском vGPU до гостьових систем Windows.

Нова версія машини Proxmox VE 9.2+pve1вимикає стани живлення S3/S4.

Нові віртуальні машини Windows закріплюються на цій версії машини.

Існуючі віртуальні машини Windows уже закріплені на попередній версії комп'ютера, яка підтримує стан живлення S3/S4.[8]

Віртуальні машини з машинною версією latestвимкнуть стан живлення S3/S4 під час наступного запуску.[8]

- Поясніть обробку застарілих версій машини QEMU.

Починаючи з QEMU 10.1, машинні версії буде видалено з попереднього QEMU через шість років.

Proxmox VE підтримуватиме машинні версії приблизно двох попередніх основних випусків Proxmox VE.

Тепер це задокументовано, і QEMU попереджатиме про майбутнє припинення підтримки під час запуску віртуальної машини.

- Майстер створення віртуальної машини тепер розміщує додаткові ISO, як-от для драйверів Windows VirtIO, після ISO встановлення в порядку завантаження (проблема 6116).
- Виправлено проблеми з безпекою, що стосуються форматів зображень, які дозволять зловмисникам з достатньою привілейованістю читати або читати й записувати довільні файли хосту.
- Дозволити налаштувати ціль розгортання для автоматичного розподілу пам’яті (проблема 2413).

Раніше алгоритм розширення націлювався на 80% доступної пам’яті хоста, що може бути занадто мало для налаштувань із великим об’ємом пам’яті.

Ціль розгортання тепер можна налаштувати для кожного вузла.

- Вимкнення віртуальної машини з увімкненим гостьовим агентом QEMU тепер повертатиметься до вимкнення ACPI, якщо гостьовий агент QEMU не активний.

Раніше, якщо гостьовий QEMU був увімкнений, але не активний, спроба завершення роботи не мала ефекту.

- Вимагати, щоб format параметр диска віртуальної машини, якщо встановлено, відповідав формату, який повідомляє рівень зберігання, якщо диском керує Proxmox VE.
- Удосконалення пристроїв VirtIO RNG, які забезпечують ентропію віртуальним машинам:
    1) Дозволити користувачам, які не є root-правами, VM.Config.HWT налаштовувати /dev/urandom та використовувати /dev/random як джерело ентропії.
    2) Дозволити некореневим користувачам із додатковими Mapping.Use привілеями на /mapping/hwrng шляху ACL налаштовувати апаратний генератор випадкових чисел.
    3) Видаліть застаріле попередження про ентропійне голодування під час використання /dev/random як джерела ентропії.
- Дозволити офлайн-міграцію, якщо наявні підключені пристрої.

Раніше зіставлені пристрої неправильно позначалися як локальні ресурси.

- Знімки з оперативною пам’яттю тепер записують стан віртуальної машини у виділений потік вводу-виводу та таким чином зменшують навантаження на основний потік QEMU.[8]

Це може покращити продуктивність і уникнути зависань у гостьовій системі, якщо стан віртуальної машини записується в ненадійне мережеве сховище.[8]

Це також дозволяє уникнути взаємоблокувань у разі перезавантаження гостьової системи під час створення знімка з оперативною пам'яттю.

- Збільште максимальний час очікування для запуску віртуальної машини з кількістю підключених віртуальних мережевих карт (проблема 3588).
- Клонування віртуальної машини зі станом TPM тепер завжди повністю клонує стан TPM.
- Вирішено проблему, через яку виділення нового диска EFI або тома стану TPM для шаблону не перетворювало ці томи на базові томи.
- Вирішіть проблему, яка порушувала живий імпорт із пристроїв OVA з дисками.
- Запобігайте відновленню шаблону, оскільки шаблони не повинні бути запущені повністю.

Шаблони частково запускаються в prelaunchстані під час резервного копіювання.

- Запобігання переміщенню або клонуванню дисків віртуальної машини до сховищ без типу вмісту «Образ диска» (проблема 5284).
- Видалення стану TPM або диска EFI із запущеної віртуальної машини тепер буде зареєстровано як очікувану зміну, оскільки їх від'єднання від запущеної віртуальної машини не підтримується.
- Виправлено проблему, через яку не вдалося створити резервну копію шаблону, якщо віртуальний мережевий адаптер від'єднано (проблема 6007).
- Виправлено деякі помилкові попередження, наприклад попередження, надруковане qm import disk (проблема 5980).
- Усуньте проблему, через яку процеси QEMU для гостьових систем Linux споживали більше ЦП після оновлення до QEMU 9.2.
- Уточніть повідомлення про помилки, якщо клонування не вдається.
- Скасувати патч ядра, який запобігав проходженню iGPU на платформах Intel Skylake.
- Резервне портування виправлення ядра, яке виправляє регресію продуктивності KVM на платформах Intel Emerald Rapids.

Контейнери (LXC):

- Дозволити встановлення режиму виявлення змін для одноразового резервного копіювання контейнера на Proxmox Backup Server (проблема 5936).
- Майстер створення LXC тепер пояснює, що вкладення ніколи не вмикається для привілейованих контейнерів, знімаючи відповідний прапорець.
- Тепер кінцева точка API /nodes/{node}/lxc/{vmid}/interfaces повертає всі налаштовані IP-адреси контейнера (проблема 5339).

Раніше поверталася лише перша адреса.[8]

- Вирішено проблему, через яку відкривання консолі контейнера на іншому вузлі кластера запитувало, чи слід довіряти ключу хоста SSH вузла.[8]
- Ігноруйте конфліктні параметри монтування для точок монтування лише для читання (проблема 5907).

Це вирішує проблему, через яку конфліктні параметри призводять до збою монтування.

Загальні покращення для віртуальних гостей:

- Удосконалення віддаленої міграції:
  1) Дозволити віддалену міграцію гостьових систем із дисками у спільному сховищі, наприклад RBD або iSCSI.
  2) Виправлено проблему, через яку дистанційне переміщення контейнера не вдавалося, якщо nesting було явно встановлено значення 0.
  3) Вирішено проблему, через яку переміщення диска в автономному режимі зазнавало збою в певних ситуаціях, якщо ціль має обмеження пропускної здатності (проблема 6130).

Покращене керування для кластерів Proxmox VE:

- Покращення pveproxy та pvedaemon:
  1) Збільште максимально дозволений розмір запитів POST до 512 КіБ, щоб уникнути проблем із великими конфігураціями (проблема 6230).

Раніше максимально допустимий розмір становив 64 КіБ, який можна було перевищити, наприклад, створивши великі відображення PCI.

  2) Обробники API тепер повертають помилки у відповіді JSON.

Таким чином, клієнтські бібліотеки можуть витягти помилку без аналізу фрази причини стану HTTP.

  3) За потреби поверніть код статусу HTTP 500 «Внутрішня помилка сервера» замість 501 «Не реалізовано».
  4) У разі помилок додайте повідомлення про помилку до тіла відповіді HTTP, якщо не встановлено інший явний вміст.
  5) Уникайте надлишкових відключень, якщо клієнт, що підключається, не надсилає дані (проблема 4816).

Це має зменшити кількість detected empty handle попереджень у журналах.

  6) Надсилайте сповіщення про закриття TLS перед закриттям з'єднання, щоб покращити сумісність із певними клієнтами HTTPS.
  7) Журнали pveproxy тепер можуть додатково згадувати IP-адресу підключення, зчитану із заголовка, що може бути корисним у середовищах, де доступ до Proxmox VE здійснюється через проксі (проблема 5699).

- Покращення системи сповіщень:
  1) Дозволити заміну шаблонів, які використовуються для сповіщень, які надсилаються як звичайний текст, так і HTML (проблема 6143) .[8]

2) Оптимізуйте шаблони сповіщень, готуючи шаблони, які можна замінити користувачем.[8]
3) Уточніть описи режимів відповідності сповіщень (проблема 6088).
4) Виправлено помилку, яка сталася під час створення або оновлення цілі сповіщення.
5) HTTP-запити до цілей webhook і gotify тепер встановлюють Content-Length заголовок.

- Плагін InfluxDB: Виправлено проблему, через яку зібрані дані були б неповними, якщо визначено гості з одним тегом із числовим значенням.
- Виправлено проблему, через яку /cluster/metrics/export поверталися неправильні дані для iowait метрики.
- Удосконалити специфікацію схем відповідей API, щоб спростити взаємодію з API Proxmox VE.
- Оновлення corosync версії 3.1.9 з додатковими патчами посилення.
- Pvereport тепер також відображає WWID підключених дисків для легшого усунення несправностей багатошляхових конфігурацій.

Резервне копіювання/відновлення:

- Підтримка зовнішніх постачальників резервного копіювання.

Стек зберігання даних Proxmox VE тепер надає API для розробки плагінів провайдера резервного копіювання для зовнішніх рішень резервного копіювання.

Плагіни постачальників резервних копій можуть використовувати розширені функції, наприклад відстеження брудних растрових зображень для швидкого інкрементального резервного копіювання.

Постачальники рішень резервного копіювання тепер можуть розробити спеціальний плагін резервного копіювання для інтеграції свого рішення резервного копіювання з Proxmox VE.

- Виправлено умову конкуренції, яка могла перешкоджати належному поширенню помилок під час резервного копіювання контейнера на Proxmox Backup Server.
- Покращення режимів виявлення змін для резервних копій контейнерів на Proxmox Backup Server, представлених у Proxmox VE 8.3:
  1) Виправлено проблему, через яку розмір файлу не враховувався для порівняння метаданих, що могло спричинити помилку наступних відновлення.
- Підвищення надійності резервних копій, що знімаються:
  1) Запишіть ім’я очищуваного образу та очищайте залишки очищуваних зображень у відповідних випадках, наприклад під час наступного резервного копіювання або міграції (проблема 5440).
  2) Покращте обробку помилок для очищення та уникніть застряглих гостьових операцій вводу-виводу.
- Резервне копіювання шаблону віртуальної машини в архіви VMA тепер використовує той самий підхід до резервного копіювання, що й резервне копіювання на Proxmox Backup Server.[8]

- Відновлення файлів із Proxmox Backup Server: перейдіть до blockdev параметрів під час підготовки дисків для віртуальної машини для відновлення файлів.[8]

Крім того, Виправлено короткочасну регресію під час використання просторів імен або шифрування через цю зміну.

Зберігання:

- Плагін iSCSI: зменшіть обсяг перевірок з'єднання через TCP ping, щоб уникнути повторних помилкових попереджень на цільовій стороні (проблема 957).
- Покращення плагіна ESXi:
  1) Уникайте помилок DBUS щодо максимальної кількості правил відповідності на з'єднання (проблема 5876).
  2) Переконайтеся, що образи дисків VMDK правильно визначені.
- Імпорт OVA/OVF: вирішіть проблему безпеки, яка дозволила б зловмиснику з достатніми привілейованими правами отримати копію довільного файлу хосту, імпортувавши спеціально створений пристрій OVA (PSA-2024-00013-1).
- Плагін Btrfs: вирішено проблему, через яку гостьова міграція зазнавала збою, якщо диски мали кілька знімків у Btrfs (проблема 3873).
- Виправлено періодичну проблему, через яку використання ISO на Btrfs завершувалося помилкою.
- Виправлено проблеми з кешуванням у RBD і плагінах зберігання iSCSI для режиму користувача (проблема 6085).
- Виправлено проблему, через яку завдання реплікації не видалялося, якщо його було вимкнено.
- Термін експлуатації Ceph Quincy 17.2 закінчився, користувачам рекомендується оновити принаймні до Ceph Reef .
- Додайте до графічного інтерфейсу необов'язковий стовпець для програми пулу.
- Вирішено проблему, через яку час від часу редагування пулу за допомогою графічного інтерфейсу могло встановлювати різні значення для size та min_size.

Контроль доступу:

- Покращення сфери OpenID Connect:
  1) Додайте підтримку груп (проблема 4411).

Нове groups-claim налаштування визначає претензію OpenID, яка використовується для визначення груп користувача на стороні Proxmox VE.

За бажанням можна автоматично створювати групи, які не існують на стороні Proxmox VE.

  2) Дозволити вимкнути запити до кінцевої точки UserInfo, оскільки деякі постачальники посвідчень не підтримують це (проблема 4234).
- Виправлено помилкове попередження під час використання областей без урахування регістру, таких як Active Directory.[8]

- Поясніть, що зміни пароля для області PAM впливають лише на локальний вузол.[8]

Брандмауер і програмно визначена мережа:

- Додайте перевірку сертифіката TLS для зовнішніх плагінів IPAM/DNS (PSA-2025-00006-1).

Інтеграція плагіна IPAM/DNS (наразі в технічній попередній версії) стека SDN не мала перевірки сертифіката TLS, що потенційно дозволяло атаки MITM.

- Заборонити призначати одній зоні кілька діапазонів DHCP, які перекриваються. Кілька серверних модулів IPAM їх не підтримують.
- Переробити та вдосконалити плагін Netbox IPAM для належної синхронізації видалення та оновлення адрес і мереж (проблема 5496).

Рефакторинг коду для послідовної обробки помилок, про які повідомляє Netbox.

Видаліть підмережі в Netbox після їх видалення в Proxmox VE.

Правильно повернути адресу після її збереження та отримання з Netbox. Це вирішує проблему, яка перешкоджала запуску гостьових систем (проблема 5496).

- Виправлено обробку додавання імен DNS у плагіні PowerDNS для налаштувань із двома стеками та зон.

Відсутність явного надсилання типу запису для додавання (A/AAAA) спричинило помилку в API PowerDNS, у результаті чого записи не було додано взагалі.

- Виправлено виявлення незавершених змін для логічних параметрів конфігурації.

Налаштування, що оцінюється як, false завжди відображалося як зміна, що очікує на розгляд.

- Додайте log_level_format параметр до API брандмауера та коду аналізу (проблема 5925).
- Пропустіть створення правил у прямому ланцюжку для груп безпеки, прив’язаних до інтерфейсу, замість того, щоб викликати помилку.

Правила прямого ланцюжка діють на кількох інтерфейсах, і прив’язувати їх до інтерфейсу не має сенсу.

- Адаптуйте нову proxmox-firewall реалізацію, щоб краще узгодити її параметри та стандартні параметри з pve-firewall.
    1) Виправлено поведінку опції nf_conntrack_allow_invalid в новій proxmox-firewall реалізації.
    2) Розглядайте параметри відсутності брандмауера для інтерфейсу гостьової мережі як false(проблема 6176).
    3) Виправлення правил брандмауера для протоколу ICMP (проблема 6108).
- Переконайтеся, що frr службу ввімкнено під час (повторного) запуску.
- Оновіть frr набір протоколів маршрутизації з версії 8.5.2 до версії 10.2.1.
- frr: додайте можливість dummy розглядати інтерфейси як loopback інтерфейси в OpenFabric.[8]

Покращене керування вузлами Proxmox VE:[8]

- Було виявлено та виправлено декілька вразливостей у GRUB, які можна було використати для обходу SecureBoot (PSA-2025-00005-1).

Документація для SecureBoot тепер містить інструкції щодо запобігання використанню вразливих компонентів для завантаження за допомогою політики відкликання.

- Додайте pve-nvidia-vgpu-helperінструмент для спрощення налаштування драйвера NVIDIA v GPU.
- Резервне портування виправлення ядра, яке дозволяє уникнути зниження продуктивності процесорів Raptor Lake із останнім мікрокодом (проблема 6065).
- Резервне портування виправлення ядра, яке виправляє збої мережі Open vSwitch, які з низькою ймовірністю можуть виникнути під час виходу з ovs-tcpdump.
- Оновіть плагіни DNS, надані acme.sh , до попередньої версії 3.1.0.
- Виправлено помилкове попередження про зламану трубу під час запиту статусу DKMS у pve7to8.
- Вирішено проблему, через яку параметри автоматичної активації для груп томів і логічних томів LVM не враховувалися під час завантаження.

Установка ISO:

- Збільште мінімальну довжину пароля root з 5 до 8 символів для всіх інсталяторів.

Ця зміна виконана відповідно до поточних рекомендацій NIST .

- Надрукуйте більше видимої для користувача інформації про причини невдачі автоматичного встановлення.
- Дозволити встановлювати рівні RAID без урахування регістру у файлі відповідей для автоматичного інсталятора.
- Заборонити автоматичному інсталятору друкувати повідомлення про хід виконання, поки не було прогресу.
- Правильно підтвердьте перевагу користувача щодо перезавантаження у разі помилки під час автоматичного встановлення (проблема 5984).
- Дозволити використовувати двійкові виконувані файли (на додаток до сценаріїв оболонки) як виконуваний файл першого завантаження для автоматичного інсталятора.
- Дозволити властивостям у файлі відповіді автоматичного інсталятора бути в snake_caseабо kebab-case.

Цей kebab-case варіант є кращим для більшої сумісності з іншими форматами конфігураційних файлів Proxmox.

Варіант snake_case буде поступово застарілим і вилученим у наступних випусках основних версій.

- Під час підготовки ISO автоматичного інсталятора перевірте локаль і параметри першого завантаження, замість того, щоб зірвати інсталяцію через неправильні налаштування.[8]

- Запобігання друку некритичних повідомлень журналу ядра, які малювали поверх інтерфейсу інсталятора TUI.[8]
- Зберігайте конфігурацію мережі, визначену за допомогою DHCP у графічному інсталяторі, навіть якщо не натиснути Nextкнопку спочатку (проблема 2502).
- Додайте опцію для отримання повного доменного імені (FQDN) із сервера DHCP за допомогою автоматичного інсталятора (проблема 5811).
- Тепер ISO надсилає та встановлює FRRouting, але тримає службу вимкненою.
- Покращте обробку помилок, якщо в мережі не налаштовано сервер DHCP або не отримано оренду DHCP.

Інсталятор графічного інтерфейсу попередньо вибере перший знайдений інтерфейс, якщо мережу не було налаштовано з DHCP.

Інсталятор повернеться до більш розумних значень для адреси інтерфейсу, адреси шлюзу та DNS-сервера, якщо мережу не було налаштовано з DHCP.

- Додано опцію вимкнення комп'ютера після успішного встановлення за допомогою автоматичного інсталятора (проблема 5880).
- Покращте налаштування максимального розміру ZFS ARC для систем з обмеженим обсягом пам'яті.

У цих системах максимальний розмір ZFS ARC обмежується таким чином, що в системі завжди залишається щонайменше 1 ГіБ пам'яті.

- Переконайтеся, що максимальний розмір ZFS ARC також установлено для додаткових сховищ, навіть якщо коренева файлова система не є ZFS (проблема 6285).
- Змусити інсталяції Btrfs використовувати proxmox-boot-tool для керування системними розділами EFI (проблема 5433).
- Змусьте GRUB встановити завантажувач безпосередньо на диск, щоб переконатися, що система все ще завантажується, навіть якщо змінні EFI були пошкоджені.
- Виправлено помилку в параметрах жорсткого диска інсталятора графічного інтерфейсу, через яку ext4 і xfs показують неправильні параметри після повернення з вкладки додаткових параметрів Btrfs.

Помітні зміни:

- Починаючи з NVIDIA vGPU Software 18, Proxmox VE є офіційно підтримуваною платформою.

Відомі проблеми та критичні зміни:

Для завантаження PXE на віртуальній машині з OVMF потрібен VirtIO RNG.

З міркувань безпеки прошивка OVMF тепер вимикає завантаження PXE для гостей без генератора випадкових чисел.

Якщо ви хочете використовувати завантаження PXE у віртуальних машинах OVMF, переконайтеся, що ви додали пристрій VirtIO RNG. Це дозволено для root і користувачів з VM.Config.HWTypeпривілеями.

OSD, розгорнуті на Ceph Squid, виходять з ладу:[8]

Зараз Ceph Squid має проблему, через яку нещодавно створені екранні меню виходять з ладу . Здається, це впливає зокрема на пули EC і лише на OSD, щойно створені за допомогою Ceph 19.2 Squid.

Ми вже опублікували виправлену версію Ceph (19.2.1-pve3), яка вирішує цю проблему для щойно створених OSD, змінюючи неправильне налаштування за замовчуванням. Рекомендовано оновити кластер Squid.

Крім того, ви також можете вручну змінити проблемне bluestore_elastic_shared_blobs 0 налаштування за допомогою такої команди: ceph config set osd bluestore_elastic_shared_blobs 0.

Якщо ви розгорнули *нові* OSD за допомогою версії Ceph Squid *до* 19.2.1-pve3, тобто будь-якої версії, включно з 19.2.0-pve1 і 19.2.1-pve2, вам слід знищити та відтворити кожне OSD після оновлення до 19.2.1-pve3 або пізнішої версії або зміни цього параметра вручну, як описано вище. Ви можете робити це по одному, чекаючи, поки кластер відновиться до здорового стану між ними.

«Завантажити з URL-адреси» тепер використовує налаштований проксі-сервер для з’єднань HTTPS

«Завантажити з URL-адреси» дозволяє завантажувати ISO, шаблони контейнерів і пристрої OVA безпосередньо з хосту Proxmox VE.

Якщо проксі-сервер було налаштовано в параметрах центру обробки даних у Proxmox 8.3 і нижче, «Завантажити з URL-адреси» використовував проксі лише для з’єднань HTTP, але не для з’єднань HTTPS. Це порушило функцію «Завантажити з URL-адреси» через HTTPS у налаштуваннях, де хост повинен отримати доступ до сховища файлів через проксі-сервер.

Тепер Proxmox VE також використовує налаштований проксі для з’єднань HTTPS. Це може порушити налаштування, які ґрунтувалися на попередній поведінці. Наприклад, налаштування, які мають налаштований проксі та завантажують файли через HTTPS безпосередньо з внутрішнього сховища, і де проксі не дозволяє доступ до внутрішнього сховища. Для таких налаштувань потрібно буде завантажувати файли вручну без проксі.

Плагіни SDN IPAM/DNS (технічний перегляд): може знадобитися оновлення конфігурації, щоб уникнути помилок перевірки сертифікації TLS

Користувачі зовнішніх плагінів IPAM або DNS можуть бачити помилки перевірки сертифіката TLS під час спроби зв’язатися із зовнішніми службами IPAM або DNS після оновлення до Proxmox 8.4.

Причина полягає в тому, що HTTPS-підключення до зовнішніх служб IPAM або DNS не перевіряли сертифікати TLS у Proxmox VE 8.3 і нижче (PSA-2025-00006-1), але тепер перевіряють сертифікати TLS, що може завершитися помилкою, якщо сертифікат не підписаний ЦС, якому довіряє вузол Proxmox VE. Щоб виправити це, або відредагуйте конфігурацію плагіна та надайте відбиток SHA-256 сертифіката зовнішньої служби, або переконайтеся, що вузол Proxmox VE довіряє відповідному ЦС.[8]

Плагін SDN Netbox IPAM (технічний попередній перегляд): може знадобитися повторно створити діапазони DHCP

У цьому випуску виправлено помилку, через яку діапазони IP-адрес у Netbox не створювалися належним чином у простій зоні з підтримкою DHCP. Щоб виправити існуючі підмережі з налаштованими діапазонами DHCP і Netbox як IPAM, у вас є два варіанти:

- Вручну створіть діапазони IP-адрес у Netbox:
  1) Перейдіть до IPAM → IP Ranges і додайте діапазони DHCP, які ви налаштували для своєї підмережі вручну
- Нехай Proxmox VE відтворить IP-діапазони:
  1) Перейдіть до SDN → VNets
  2) Відредагуйте підмережу та видаліть усі діапазони DHCP, а потім натисніть OK
  3) Відредагуйте підмережу, повторно створіть діапазони DHCP, а потім натисніть OK
  4) Застосуйте конфігурацію SDN

Діапазони IP тепер мають відображатися в Netbox, а запуск віртуальних машин/контейнерів має працювати.[8]

2.3 Структурна схема побудови кластера Proxmox VE 8.4 та мережева інфраструктура

2.3.1 Схема з'єднання пристроїв у кластер Proxmox VE

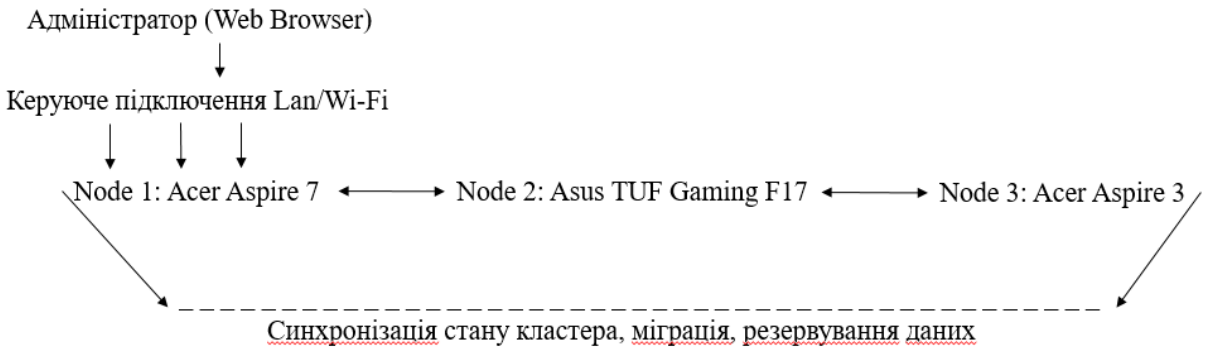


Рисунок 2.1 – Структурна схема кластера на базі трьох вузлів

2.3.2 Пояснення побудови кластера

1. Основні компоненти мережі:

- Комутатор / маршрутизатор: забезпечує фізичне з'єднання всіх вузлів у єдину локальну мережу (LAN).
- Адміністратор: взаємодіє з вузлами через веб-інтерфейс Proxmox VE, використовуючи захищений канал HTTPS.

2. Організація мережі:

- Усі вузли мають статичні IP-адреси для забезпечення стабільної роботи кластера і можливості безперервної синхронізації.

- Рекомендується використовувати провідне Ethernet-з’єднання для вузлів Node 1 і Node 2 для мінімізації затримок, особливо при міграції віртуальних машин.
- Node 3 (Acer Aspire 3) може залишитися на Wi-Fi, якщо немає технічної можливості підключити кабель.

3. Мережеві вимоги:

- Мінімальна швидкість мережі між вузлами має бути не менше 1 Гбіт/с для стабільної роботи кластерної синхронізації і живої міграції ВМ.
- Усі вузли мають бути в одній підмережі, наприклад:
  1) Node 1 – 192.168.1.101
  2) Node 2 – 192.168.1.102
  3) Node 3 – 192.168.1.103
  4) Адміністратор – 192.168.1.100

2.3.3 Обґрунтування вибору топології

Переваги локальної кластерної архітектури:

- Висока доступність (HA): у разі відмови одного вузла інші автоматично беруть на себе його навантаження.
- Легка масштабованість: додавання нового вузла до кластера можливе без простою основних служб.
- Економічність: не потребує окремого дата-центру або дорогого обладнання для тестових і навчальних цілей.
- Безпека даних: завдяки внутрішньому з'єднанню вузлів у межах локальної мережі ризик витоку даних мінімізований.

Ethernet або провідне з'єднання краще Wi-Fi:

- Ethernet-мережа має менші затримки (latency) і стабільнішу пропускну здатність, що критично важливо для живої міграції віртуальних машин і синхронізації станів кластеру.
- Wi-Fi з’єднання допускається лише для малонавантажених вузлів або при відсутності можливості проводового підключення.

2.3.4 Подальші можливості розвитку

Після базового розгортання кластера на трьох вузлах можна розширити інфраструктуру шляхом:

- Додаванням нових пристроїв до кластеру Proxmox VE.
- Встановлення розподіленої файлової системи (наприклад, Ceph) для організації високонадійного сховища.
- Встановлення додаткових мережевих інтерфейсів для розділення службової мережі і мережі користувачів.

2.4 Методика розгортання та налаштування Proxmox VE 8.4

Алгоритм розгортання кластера Proxmox VE:

Крок 1. Підготовка обладнання

- Перевірити технічні характеристики кожного ноутбука (підтримка віртуалізації AMD-V/Intel VT-x).
- Оновити BIOS до останньої версії, увімкнути опції віртуалізації (AMD-V/VT-x, VT-d).
- Підключити всі ноутбуки до спільної локальної мережі через комутатор або роутер.
- Видати статичні IP-адреси кожному ноутбуку в межах однієї підмережі.

Крок 2. Інсталяція Proxmox VE 8.4

- Завантажити образ Proxmox VE 8.4 ISO з офіційного сайту.
- Створити завантажувальну флешку за допомогою утиліти (наприклад, Rufus).
- Завантажити кожен ноутбук із флешки та встановити Proxmox VE, обираючи:
  1) Мову: English (або Ukrainian, якщо є підтримка).
  2) Мережевий інтерфейс: налаштувати правильну IP-адресу.
  3) Розділення диска: використовувати весь доступний SSD-диск.
  4) Створити початкового користувача root із надійним паролем.

Крок 3. Базова конфігурація вузлів

- Після встановлення підключитись до кожного вузла через веб-інтерфейс:

  https://<IP-адреса вузла>:8006
- Виконати оновлення системи через веб-інтерфейс або командою в shell:

  apt update && apt full-upgrade -y

Крок 4. Створення кластеру

На головному вузлі (Acer Aspire 7):

- Зайти в меню Datacenter → Cluster.
- Натиснути Create Cluster.
- Вказати ім'я кластеру (наприклад, *ProxmoxLab*).
- Зберегти ключ кластера (Cluster Join Information).

Крок 5. Приєднання інших вузлів до кластеру

На Asus TUF Gaming F17 та Acer Aspire 3:

- Перейти до меню Datacenter → Cluster.
- Натиснути Join Cluster.
- Ввести IP-адресу головного вузла та токен для приєднання.
- Підтвердити з'єднання та дочекатись синхронізації.

Крок 6. Налаштування сховищ і ресурсів

- Налаштувати спільні сховища для всіх вузлів (за потреби: NFS, Ceph, локальні диски).

- Налаштувати резервні копії для ВМ і контейнерів через Datacenter → Backup.

Крок 7. Створення віртуальних машин і контейнерів

- Завантажити шаблони ОС (наприклад, Ubuntu Server, Debian).
- Створити тестові віртуальні машини (KVM) або контейнери (LXC).
- Перевірити можливість живої міграції ВМ між вузлами.

Крок 8. Налаштування автоматичного резервування (HA)
- Увімкнути високодоступні групи (HA Group) для критичних віртуальних машин.
- Налаштувати автоматичний перезапуск ВМ у разі виходу вузла з ладу.

Крок 9. Моніторинг та підтримка кластеру
- Налаштувати моніторинг стану вузлів і віртуальних машин через вбудовані інструменти Proxmox.
- Налаштувати сповіщення про збої через email або сторонні сервіси (наприклад, Zabbix, Grafana).

2.5 Інформаційна модель системи віртуалізації на базі Proxmox VE

Основні компоненти інформаційної моделі:

1. Користувачі системи:
   - Адміністратори (повний доступ через Web-GUI та консоль).
   - Технічний персонал (обмежений доступ до моніторингу, резервних копій).
   - Користувачі сервісів (доступ тільки до розгорнутих віртуальних середовищ).
2. Інтерфейси доступу:
   - Proxmox Web GUI (порт 8006, через HTTPS).
   - SSH-консоль для адміністрування вузлів.
   - API Proxmox VE для інтеграції з іншими системами або автоматизації задач.
3. Фізичні вузли (ноутбуки):
   - Acer Aspire 7 (головний вузол).
   - Asus TUF Gaming F17 (балансування навантаження).
   - Acer Aspire 3 (контейнерні сервіси).
4. Мережеві компоненти:
   - Локальна мережа (LAN) для внутрішньої взаємодії вузлів.
   - Інтернет-доступ для оновлення пакетів та шаблонів ОС.
5. Сховища даних:
   - Локальні SSD на кожному вузлі для розміщення віртуальних машин і контейнерів.
   - Спільні каталоги для резервних копій.

6. Віртуальні об'єкти системи:
   - KVM-віртуальні машини (гостьові операційні системи Windows, Linux).
   - LXC-контейнери (легкі служби: веб-сервери, бази даних, системи моніторингу).

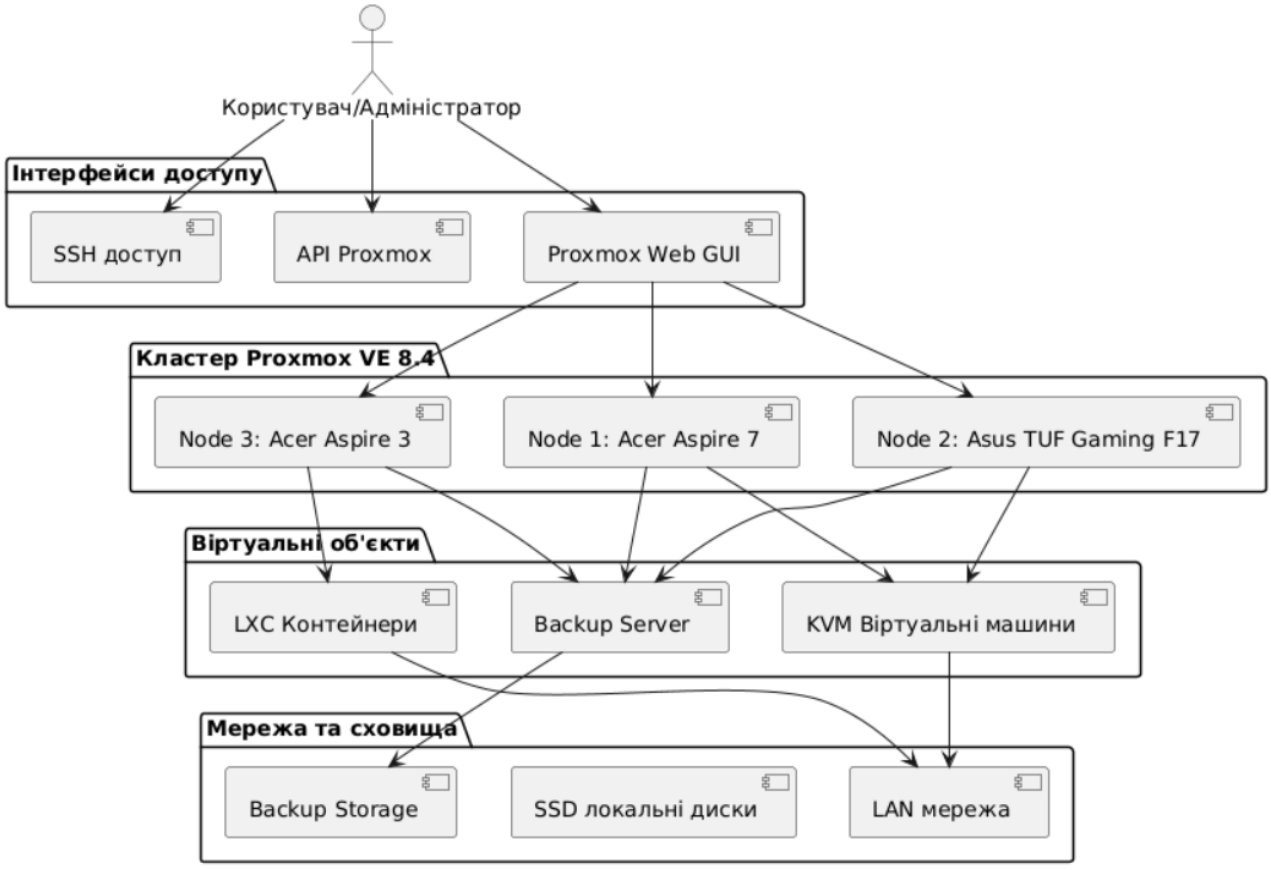


Рисунок 2.2 – Схема інформаційної моделі

Пояснення інформаційної моделі системи:

Інформаційна модель системи віртуалізації базується на розгортанні кластерної архітектури з трьох фізичних пристроїв (ноутбуків), об'єднаних через спільну локальну мережу та керованих централізованим програмним забезпеченням Proxmox VE 8.4 ISO.

Структура учасників системи складається з трьох основних рівнів:

- Рівень користувачів (адміністратори, технічний персонал, кінцеві користувачі сервісів).
- Рівень інтерфейсів управління (веб-інтерфейс Proxmox, SSH-доступ, API-запити).
- Рівень фізичних та віртуальних об'єктів (серверні вузли, віртуальні машини, контейнери, системи резервного копіювання).

1. Користувачі та їх ролі
   - Адміністратор: має повний контроль над системою через веб-інтерфейс Proxmox VE та консольний доступ SSH.
     1. Створює, налаштовує та адмініструє віртуальні машини і контейнери.
     2. Контролює стан кластеру, керує резервними копіями, оновленням системи.

- Технічний персонал: має обмежений доступ для моніторингу системи, перегляду стану віртуальних середовищ, створення резервних копій.
- Користувачі сервісів: взаємодіють тільки з прикладними віртуальними машинами (наприклад, веб-серверами, базами даних), не маючи доступу до системного рівня Proxmox.

2. Інтерфейси взаємодії
- Proxmox Web GUI: веб-браузерний інтерфейс управління кластером через порт 8006 (HTTPS).
- SSH доступ: пряме підключення до кожного фізичного вузла для виконання адміністративних команд.
- API Proxmox: використовується для автоматизації завдань, скриптів і інтеграції з іншими системами моніторингу (наприклад, Zabbix, Grafana).

3. Фізична інфраструктура
Наявні пристрої:
Node 1: Acer Aspire 7
- Головний сервер кластеру.
- Використовується для розміщення критичних віртуальних машин (сервера баз даних, ERP-системи).
- Найпотужніший за процесором (AMD Ryzen 5 5500U) та обсягом оперативної пам'яті (24 ГБ).

Node 2: Asus TUF Gaming F17
- Продуктивний сервер для балансування навантаження і розміщення сервісних ВМ.
- Intel Core i5-11400 забезпечує високу продуктивність, а 16 ГБ оперативної пам’яті достатньо для масштабованих завдань.

Node 3: Acer Aspire 3
- Легкий вузол, орієнтований на роботу з контейнерами LXC завдяки невеликій кількості оперативної пам'яті (4 ГБ).
- Використовується для запуску допоміжних служб (наприклад, моніторинг, сервіс логування).

4. Віртуалізовані об’єкти системи
KVM Віртуальні машини (ВМ):
- Повноцінні гостьові ОС на основі Linux або Windows.
- Запускаються переважно на Acer Aspire 7 і Asus TUF Gaming F17.
- Застосовуються для критичних бізнес-сервісів, які потребують ізоляції та безпеки.

LXC Контейнери:
- Легкі віртуалізовані середовища без повного ядра ОС.
- Переважно розгортаються на Acer Aspire 3.
- Підходять для запуску мікросервісів, веб-додатків, невеликих баз даних.

5. Мережеве середовище
   - Усі вузли об'єднані через локальну Ethernet-мережу за допомогою роутера або комутатора.
   - Кожен вузол має статичну IP-адресу у спільній підмережі (192.168.1.x).
   - Забезпечується синхронізація даних між вузлами, підтримка міграції віртуальних машин без зупинки (Live Migration) та висока доступність (HA).

6. Сховища даних
   - Локальні SSD диски на кожному вузлі використовуються для зберігання дисків віртуальних машин і контейнерів.
   - Backup Storage:
     1. Організоване централізоване зберігання резервних копій.
     2. Планується використання окремого розділу на Asus TUF Gaming F17 для зберігання бекапів.

7. Програмне забезпечення
   - Proxmox VE 8.4 ISO:
     1. Найновіша стабільна версія на момент розгортання.
     2. Включає в себе сучасне ядро Linux, оновлені засоби роботи з віртуалізацією (KVM, LXC), інтеграцію із системами резервного копіювання та масштабованими сховищами (Ceph, ZFS).
     3. Безкоштовне рішення із можливістю комерційної підтримки.

8. Основні принципи роботи системи
   - Користувачі через веб-браузер або API отримують доступ до інтерфейсу управління.
   - Адміністратор створює віртуальні машини і контейнери на обраному вузлі кластера.
   - У разі перевантаження одного вузла віртуальні машини можуть автоматично або вручну мігрувати на інші вузли (Live Migration).
   - Резервне копіювання віртуальних середовищ здійснюється автоматично відповідно до встановленого графіка.
   - У разі збою одного вузла кластер забезпечує автоматичне перенесення навантаження на інші вузли, забезпечуючи безперервність надання послуг (HA - High Availability).

2.6 Критерії оцінки ефективності впровадженої системи

Успішне впровадження системи віртуалізації на базі Proxmox VE 8.4 потребує чіткої оцінки її ефективності. Без належного аналізу неможливо встановити, чи досягнуто очікуваних результатів щодо продуктивності, надійності, масштабованості та економічної доцільності. Тому у цьому розділі визначено ключові критерії оцінки впровадженої інфраструктури.

2.6.1 Критерії надійності системи

Надійність є критичним фактором для будь-якої серверної інфраструктури, особливо коли йдеться про бізнес-процеси або надання безперервних сервісів.
Основні показники надійності:

- Коефіцієнт доступності системи (Availability, SLA):

$$SLA = (1 - \frac{T_{downtime}}{T_{total}}) \times 100\% \quad (2.2)$$

де $T_{downtime}$ — загальний час простою, а T_{total} — загальний час експлуатації.

- Середній час відновлення після відмови (MTTR — Mean Time to Recovery): час, необхідний для відновлення системи після збою. Чим менше цей показник, тим надійніша система.
- Підтримка високої доступності (HA) у кластері: автоматичне перенесення ВМ на доступні вузли у випадку аварії вузла.

Пояснення:

Оцінка надійності дає змогу переконатися, що сервіси продовжують працювати навіть у разі часткової відмови обладнання, мінімізуючи ризики для бізнесу або організації.

2.6.2 Критерії ефективності використання ресурсів

Одним із завдань впровадження віртуалізації є максимально ефективне використання наявних апаратних ресурсів.
Основні показники:

- Коефіцієнт використання CPU (% завантаження процесора).
- Коефіцієнт використання оперативної пам'яті (% зайнятості ОЗУ).
- Коефіцієнт використання сховища (обсяг зайнятого простору на SSD).

Пояснення:

Моніторинг і аналіз використання ресурсів дозволяють уникнути надмірного завантаження вузлів або, навпаки, простоїв ресурсів. Це важливо для планування майбутнього масштабування або оптимізації кількості віртуальних машин.

2.6.3 Критерії масштабованості системи

Масштабованість — здатність системи адаптуватися до зростання навантаження без втрати продуктивності.
Основні показники:

- Можливість додавання нових вузлів у кластер без зупинки роботи.
- Здатність динамічного створення нових віртуальних машин або контейнерів без необхідності модернізації фізичного обладнання.
- Збереження продуктивності при збільшенні кількості віртуальних сервісів.

Пояснення:

Висока масштабованість дозволяє адаптувати систему до зростання організації, зменшуючи початкові витрати та відкладаючи необхідність великої модернізації.

2.6.4 Критерії продуктивності

Оцінка продуктивності дає змогу перевірити, чи відповідає система очікуваним навантаженням.

Основні показники:

- Час завантаження віртуальної машини або контейнера.
- Час відгуку віртуалізованих сервісів (наприклад, бази даних або веб-сервера).
- Пропускна здатність мережі між вузлами кластера (особливо при живій міграції ВМ).

Пояснення:

Висока продуктивність є критичною для забезпечення якості роботи користувачів і бізнес-додатків, розгорнутих у віртуальних середовищах.

2.6.5 Критерії безпеки системи

Безпека даних і захист віртуального середовища — один із найважливіших аспектів віртуалізації.

Основні показники:

- Наявність механізмів аутентифікації і авторизації користувачів (підтримка двофакторної автентифікації).
- Ізоляція віртуальних машин і контейнерів між собою.
- Механізми резервного копіювання і відновлення даних (Backup & Restore).
- Підтримка оновлень безпеки та патчів для Proxmox VE 8.4.

Пояснення:

Комплексний підхід до безпеки дозволяє мінімізувати ризики втрати даних, компрометації віртуальних середовищ або впровадження шкідливого ПЗ.

2.6.6 Критерії економічної доцільності

Оцінка витрат і економії після впровадження системи.

Основні показники:

- Скорочення витрат на придбання фізичного обладнання.
- Зниження експлуатаційних витрат (електроенергія, обслуговування, ліцензії).
- Окупність інвестицій у систему (ROI):

$$ROI = \frac{\text{Чистий прибуток (або економія)}}{\text{Інвестиції}} \times 100\% \quad (2.3)$$

Пояснення:

Економічна ефективність є важливим аргументом на користь віртуалізації, особливо для невеликих компаній або установ із обмеженим бюджетом.

Висновок за розділом

У межах даного розділу було здійснено комплексний аналіз апаратної та програмної частини системи віртуалізації на базі Proxmox VE 8.4. Проведено підбір обладнання для побудови тестового кластеру із трьох вузлів, кожен із яких має специфічні характеристики і відповідає вимогам для забезпечення стійкої та ефективної роботи віртуалізованого середовища.

Основним вузлом обрано Acer Aspire 7, який завдяки сучасному багатоядерному процесору, великому обсягу оперативної пам'яті та швидкому SSD забезпечує розміщення критичних віртуальних машин і керування кластером.

Вузол Asus TUF Gaming F17 виконує функції балансування навантаження та резервного розміщення ВМ, забезпечуючи високу доступність системи. Третій вузол, Acer Aspire 3, оптимально використовується для контейнерних служб із мінімальними вимогами до ресурсів.

На рівні програмного забезпечення було обґрунтовано вибір платформи Proxmox VE 8.4, яка поєднує передові технології гіпервізора KVM та контейнерної віртуалізації LXC.

Proxmox VE забезпечує централізоване керування, підтримку кластеризації, живу міграцію віртуальних машин без простою, резервне копіювання та високий рівень безпеки завдяки відкритому коду та постійним оновленням.

Також розроблено структурну, інформаційну та функціональну моделі системи, які відображають логіку роботи кластеру, взаємодію вузлів між собою та взаємодію користувачів із системою через веб-інтерфейси та API.

Запропоноване рішення забезпечує:

- оптимальне використання ресурсів наявного обладнання;
- можливість масштабування системи у майбутньому;
- високу доступність та стійкість до збоїв;
- зменшення витрат на інфраструктуру завдяки використанню безкоштовного програмного забезпечення;
- гнучкість і мобільність адміністрування через сучасні інтерфейси керування.

Отже, апаратна та програмна частини створеної серверної мережі повністю відповідають завданням побудови ефективної, безпечної та економічно доцільної інфраструктури віртуалізації на базі Proxmox VE 8.4.

3. НАЛАШТУВАННЯ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Для реалізації нашого проєкту підготуємо обладнання та програмне забезпечення.

Технічне обладнання:

- USB флеш-накопичувач SanDisk Ultra USB 3.0. Обсягом 256 GB.[9]
- USB флеш-накопичувач Kingston DataTraveler Exodia. Обсягом 128 GB.[10]
- USB флеш-накопичувач SanDisk SDCZ74. Обсягом 64 GB.[11]
- USB флеш-накопичувач TOSHIBA U401. Обсягом 8GB.[12]
- Ноутбук Acer Aspire 7 A715-42G.[13]
- Ноутбук Asus TUF Gaming F17.[14]
- Ультрабук hp spectre x360 14 2-in-1 laptop.[15]
- Оптичний термінал Huawei EchoLife EG8012H5.[16]
- Патч корд Cablexpert CAT6 UTP Сірий (PP6U-1M) .[17]
- Патч корд Cablexpert CAT5e UTP 1.5 м Синій (PP12-1.5M/B) .[18]

Програмне забепечення:

- Proxmox VE 8.4 ISO Installer.[19]
- Ubuntu Server.[20]

3.1 Встановлення Proxmox VE

Завантажуємо ISO файл з офіційного сайту Proxmo$x$[7] на USB флеш-накопичувач TOSHIBA U401. Після завантаження ISO, вимикаємо пк та вмикаємо знову що зайти у BIOS. У BIOS ми вимикаємо безпечне завантаження (може запросити ввод або створення паролю адміністратора. За потребою вводимо/створюємо пароль адміністратора). До завантаження додаємо у «пріоритет завантаження» наш флеш-накопичувач з завантаженим ISO образом. Запускаємо наш ISO та бачимо:

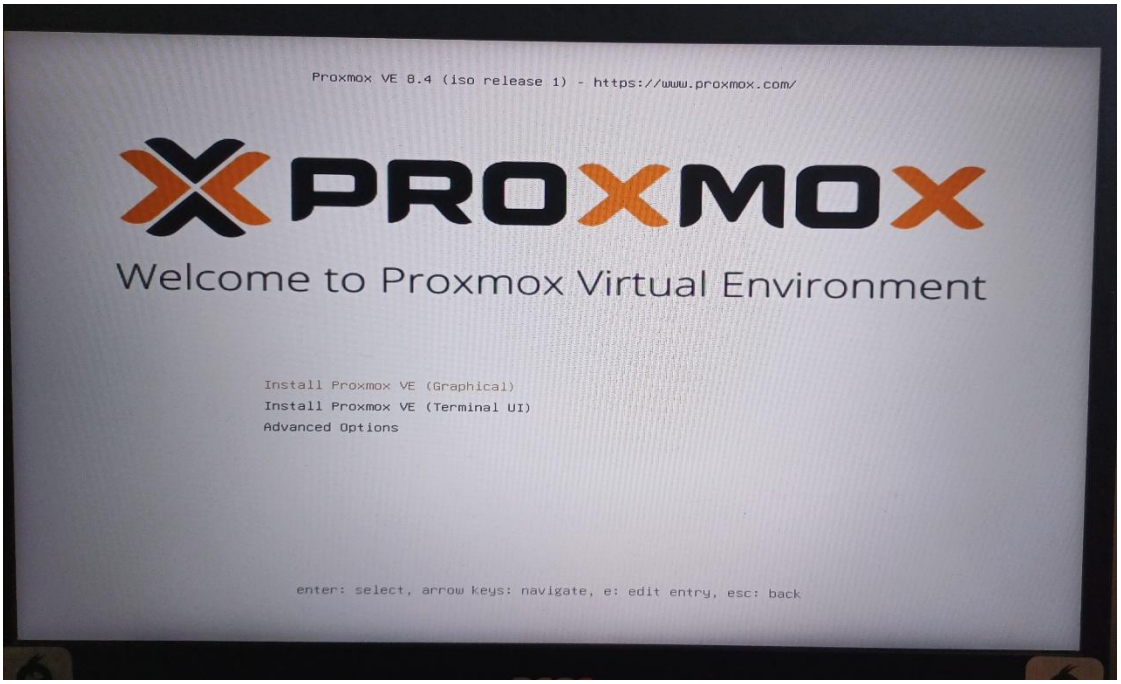


Рисунок 3.1(а) – Інсталятор Proxmox Virtual Environment

Обираємо Install Proxmox VE (Graphical) та розпочинається процес запуску інсталятора:

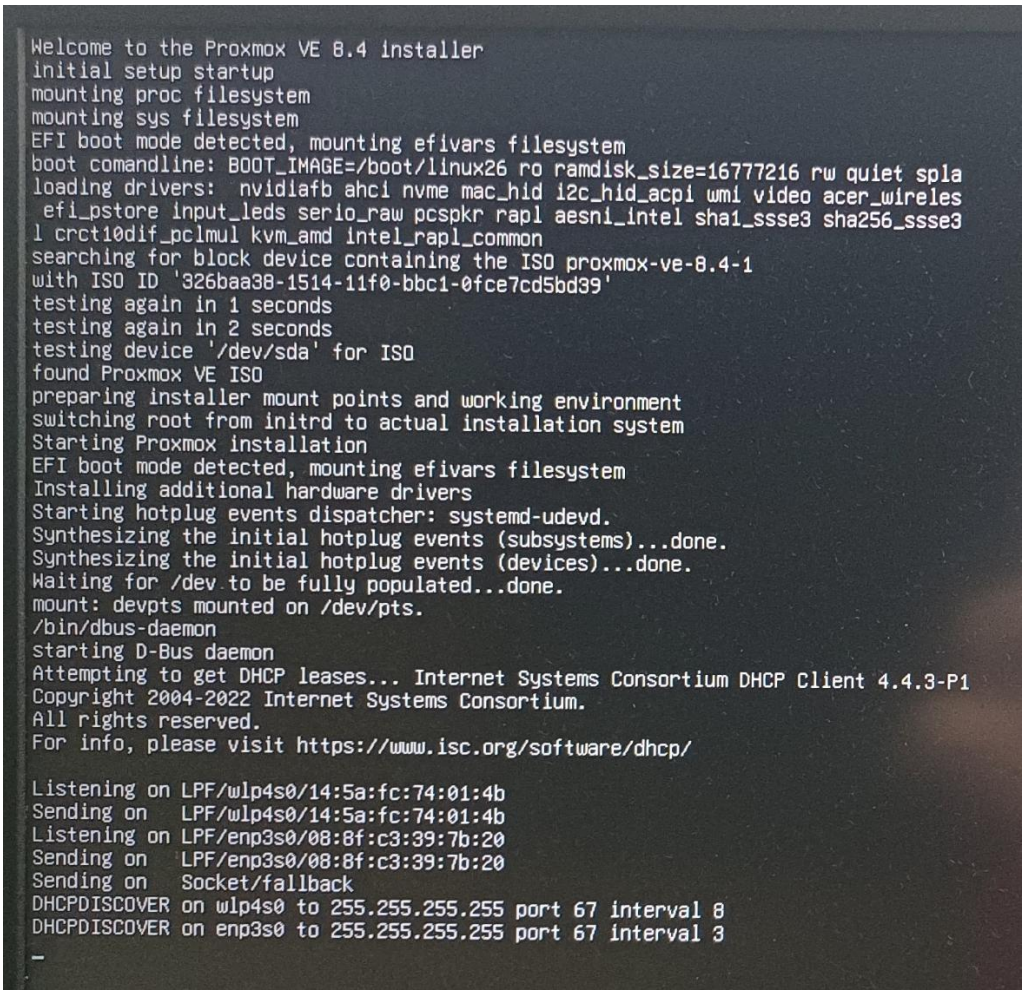


Рисунок 3.1(б) – Інсталятор Proxmox Virtual Environment

Після запуску інсталятора, нас зустрічає ліцензійна угода. Приймаємо її:

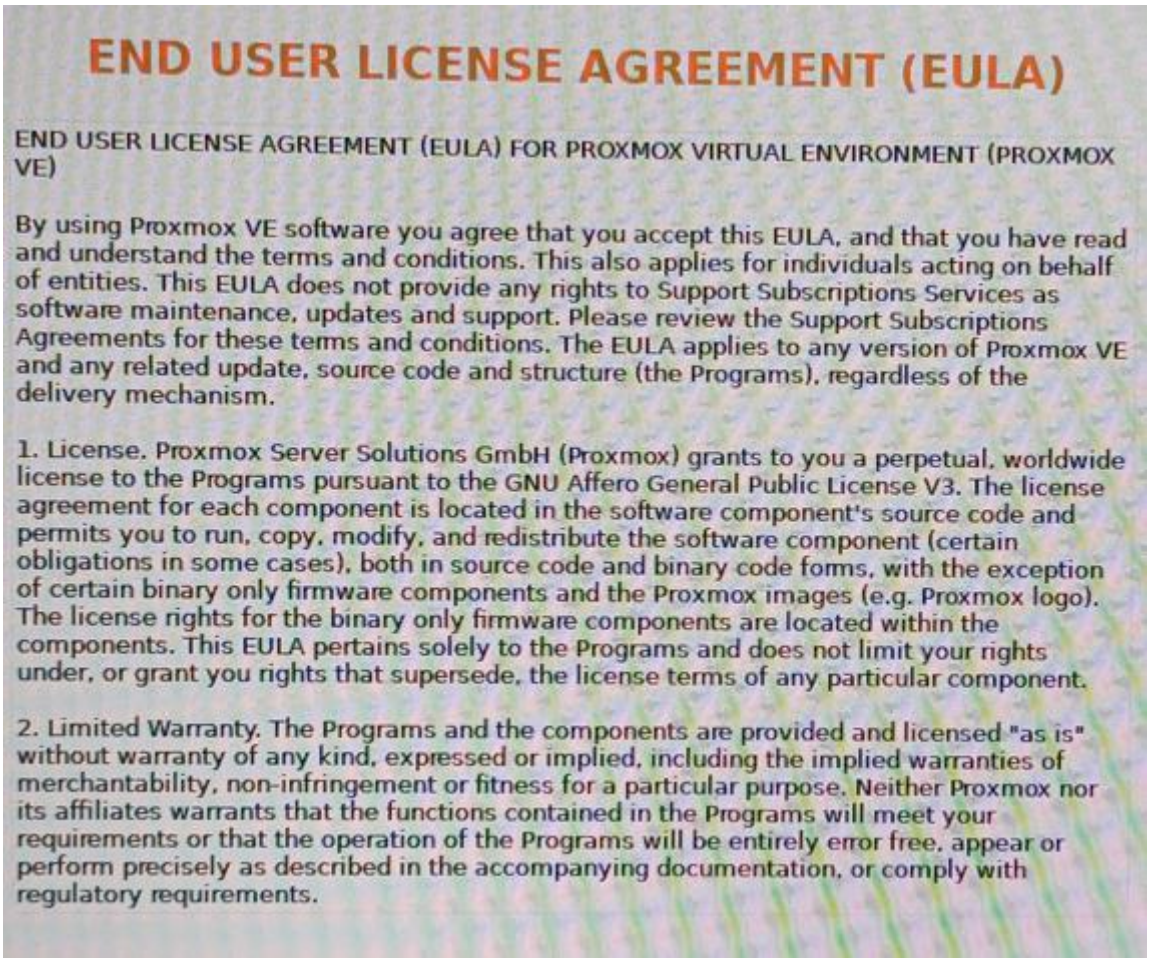


Рисунок 3.2 – USER LICENSE AGREEMENT

Обираємо Options та у Filesystem обираємо zfs (RAID0). zfs RAID0 ми обираємо тому що ми будуємо кластер. У Harddisk ми обираємо носії на яких пройде інсталяція Proxmox:

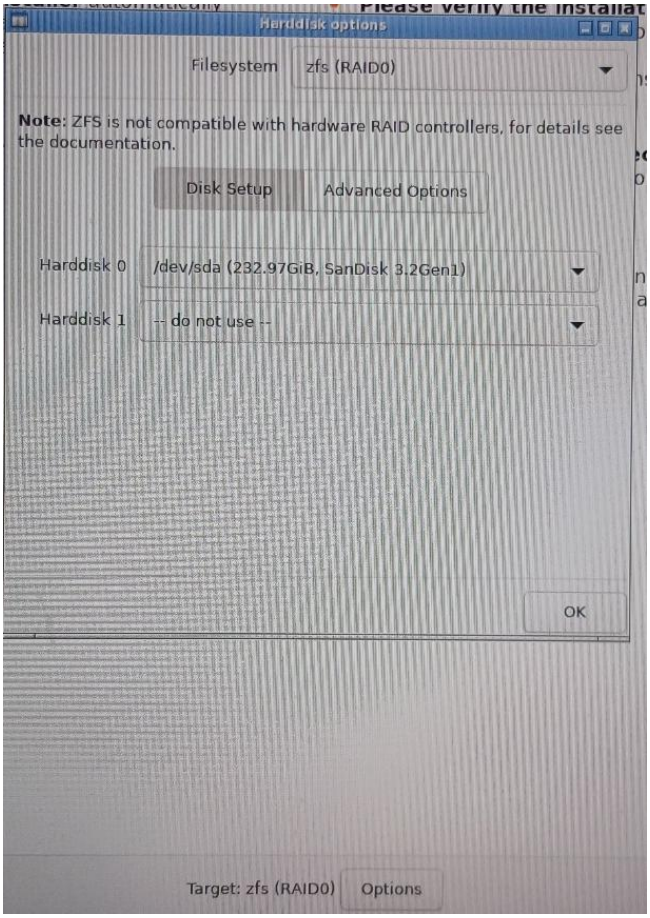


Рисунок 3.3 – Harddisk options

Далі ми вводимо назву країни, часовий пояс та мову розкладки клавіатури:

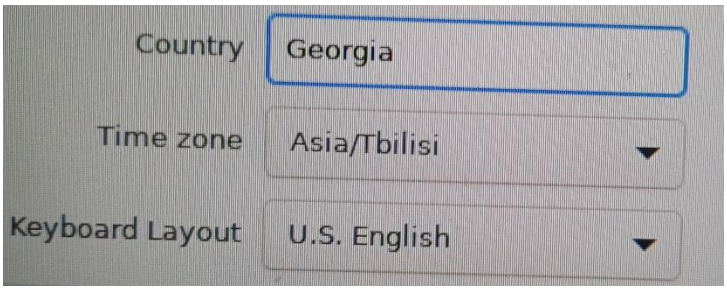


Рисунок 3.4 – Region options

Задаємо нашому серверу пароль та вводимо власний e-mail:

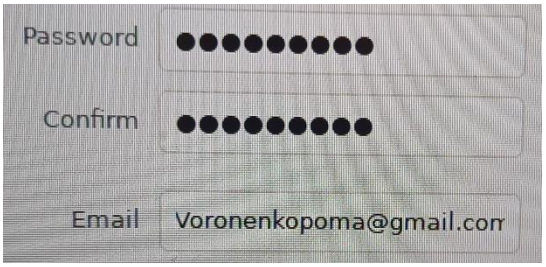


Рисунок 3.5 – Password

Після того як ми ввели пароль та емеіл, ми вводимо дані нашого способу інтернет підключення (дротове/бездротове). Додаємо ім’я хоста. Атоматично буде додано IP адресса, Getaway, DNS сервера. Також ми можемо змінити дані та вписати свої:

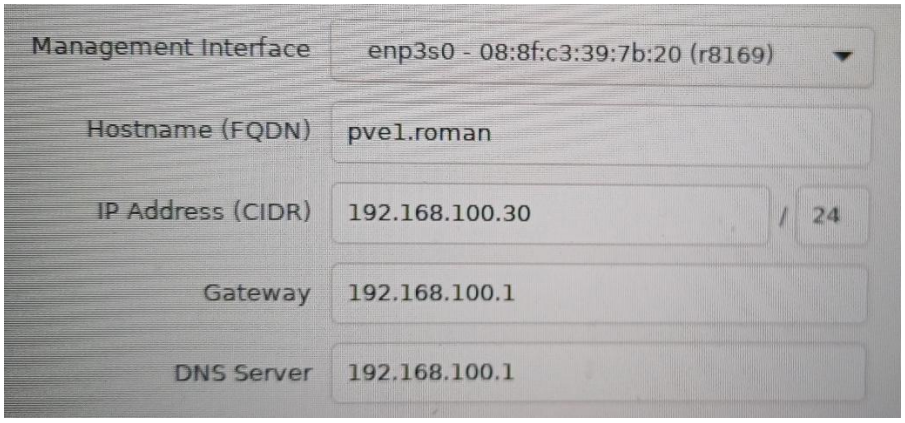


Рисунок 3.6 – Вихід в мережу

На рисунку 3.7 Proxmox нам пропонує перевірити корректність введених нами даних та якщо результат задовільний, ми розпочинаємо інсталяцію нашого серверу.

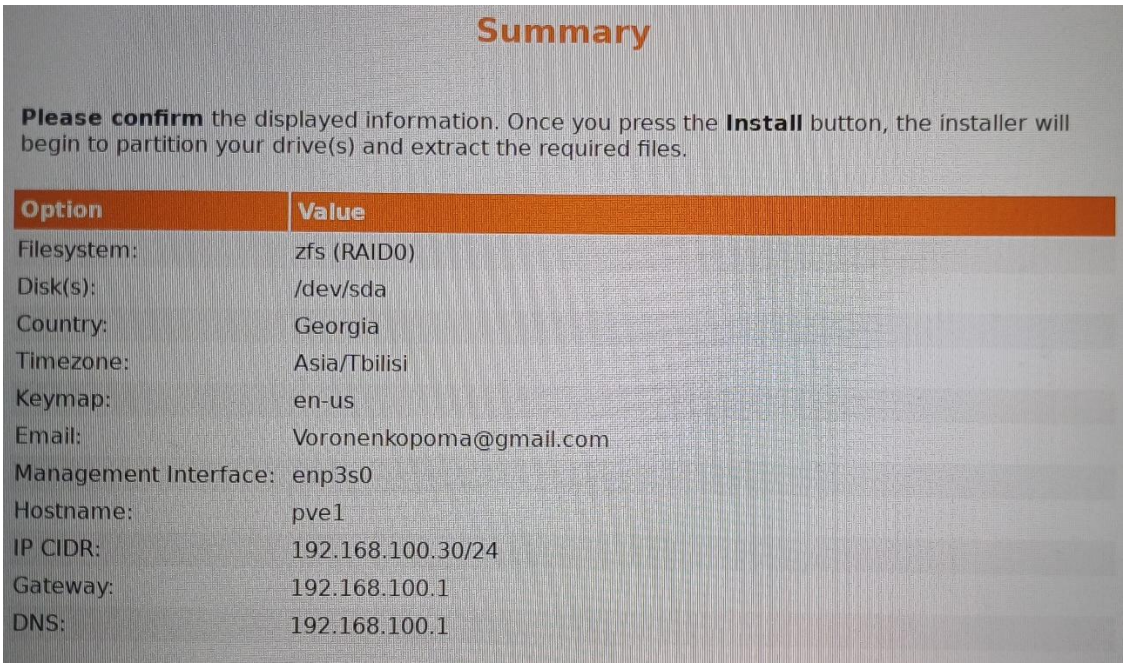


Рисунок 3.7 – Summary

Після інсталяції сервер буде запущено (рисунок 3.8). Вводимо логін (за замовчуванням логін: root). Вводимо нами створений пароль та натискаємо Enter

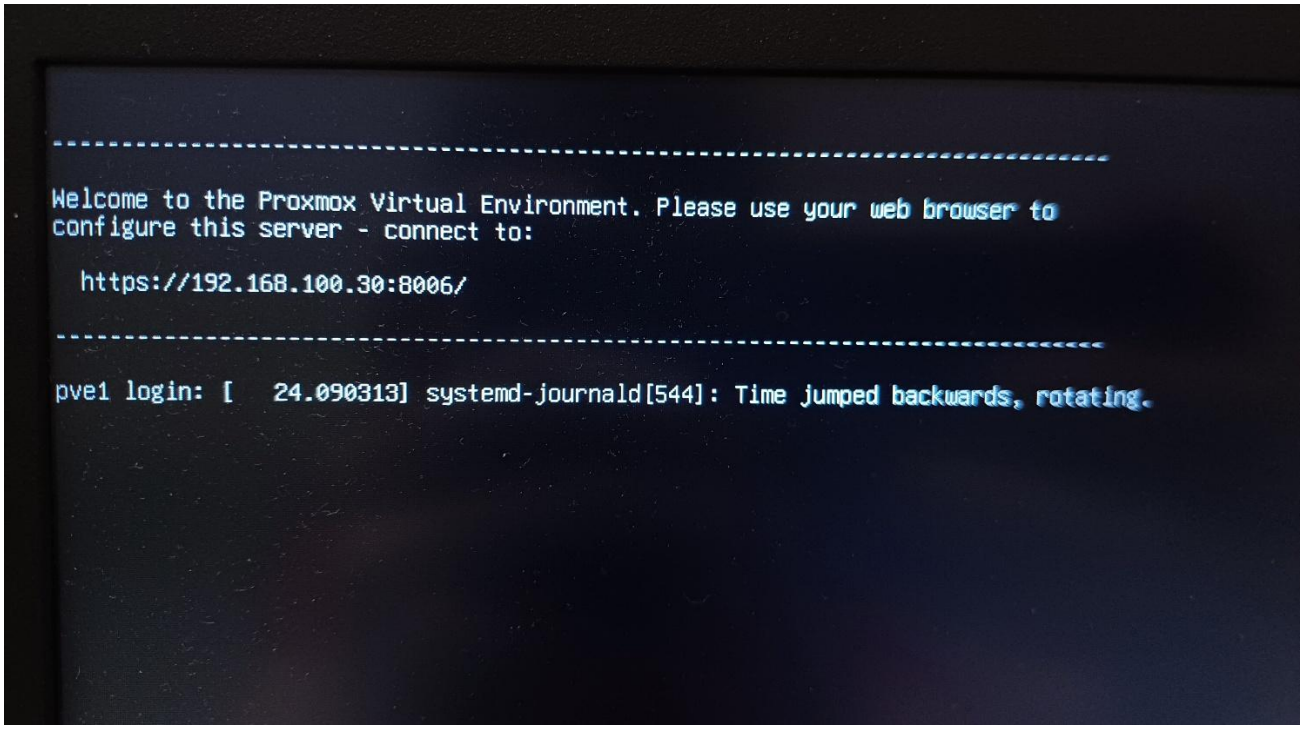


Рисунок 3.8 – Сервер запущено

3.2 Створення та налаштування кластеру

Сервер працює, але тепер нам треба до нього доєднатись. На іншому пристрої потрібно відкрити у будь-якому браузері посилання на сервер:

https://192.168.100.30:8006/

Можливо браузер буде писати про небезпечне з'єднання, але не звертаємо на це увагу та відкриваємо посилання. Невідомі посилання браузер сприймає як небезпечні. Перед входом на сервер, обираємо мову інтерфейсу та вводимо дані Ім'я користувача та пароль:

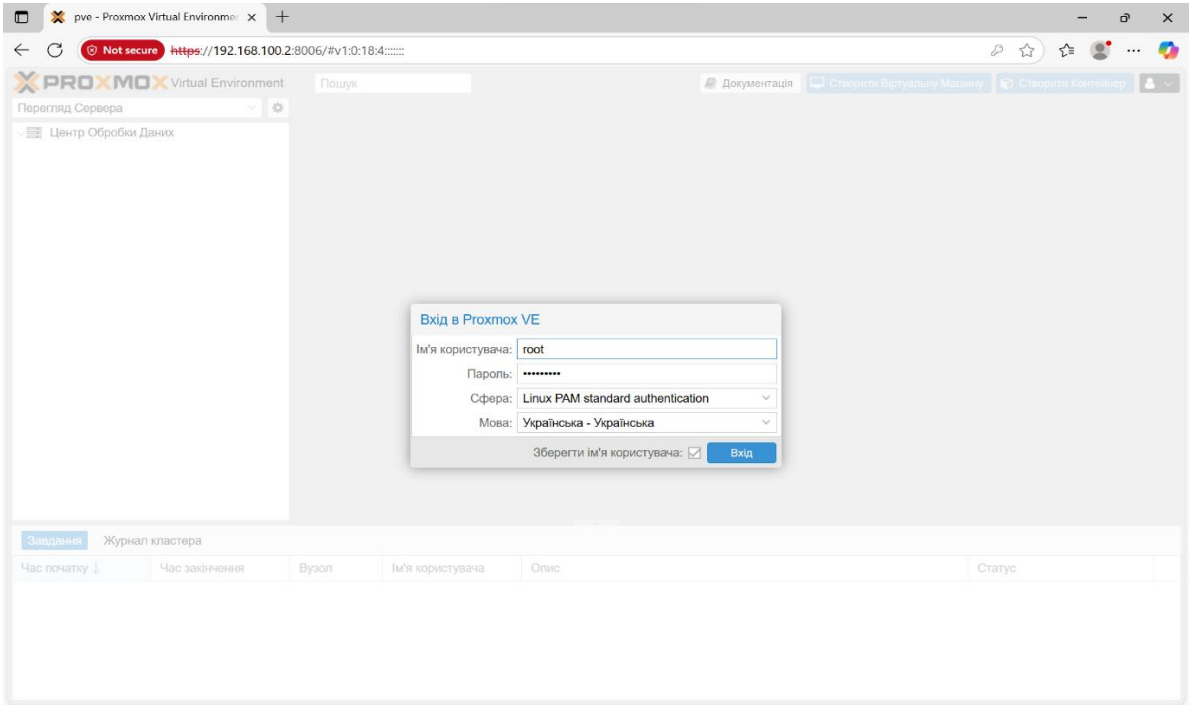


Рисунок 3.9(а) – Сервер

Тепер ми маємо один робочий сервер (рисунок 3.10)

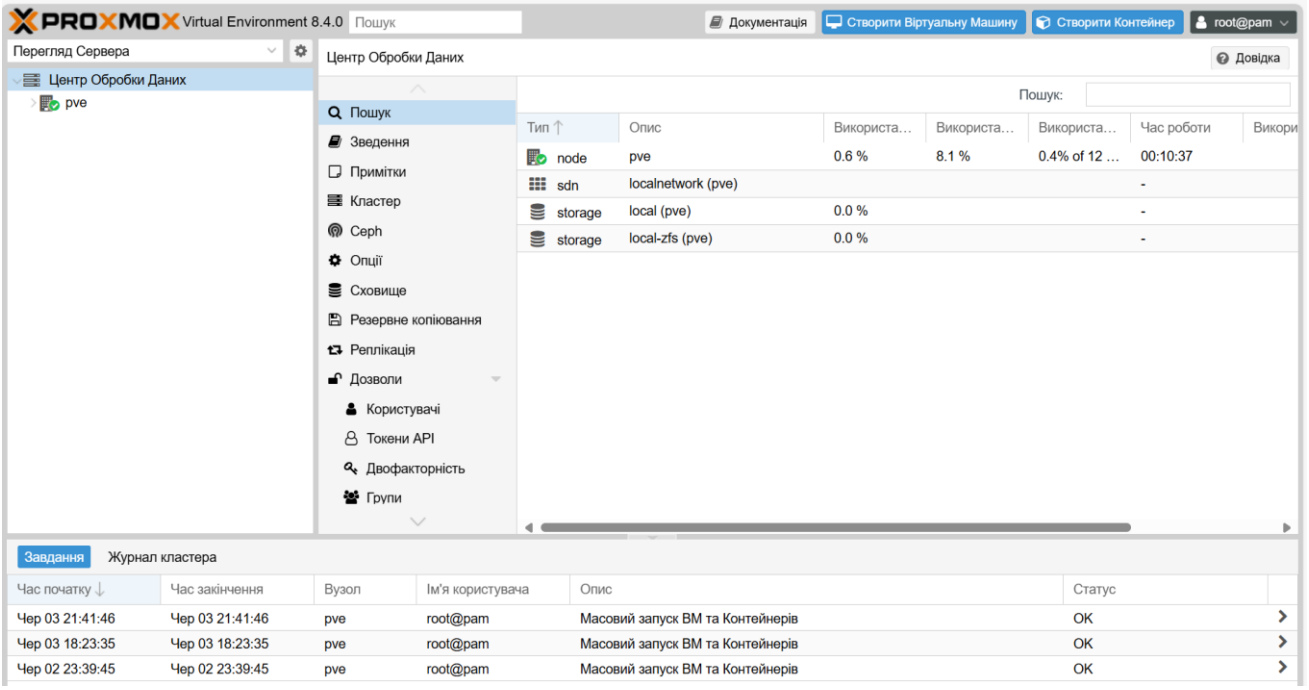


Рисунок 3.9(б) – Сервер

Щоб побудувати кластер нам потрібно повторити встановлення Proxmox, але вже на інший флеш-накопичувач. Після повторення процедури ми на іншій вкладці браузера вводимо посилання на інший сервер. Заходимо на сервер. Щоб об’єднати наші сервера у кластер, нам знадобиться зайти у Центр обробки даних – Кластер – Створити кластер:

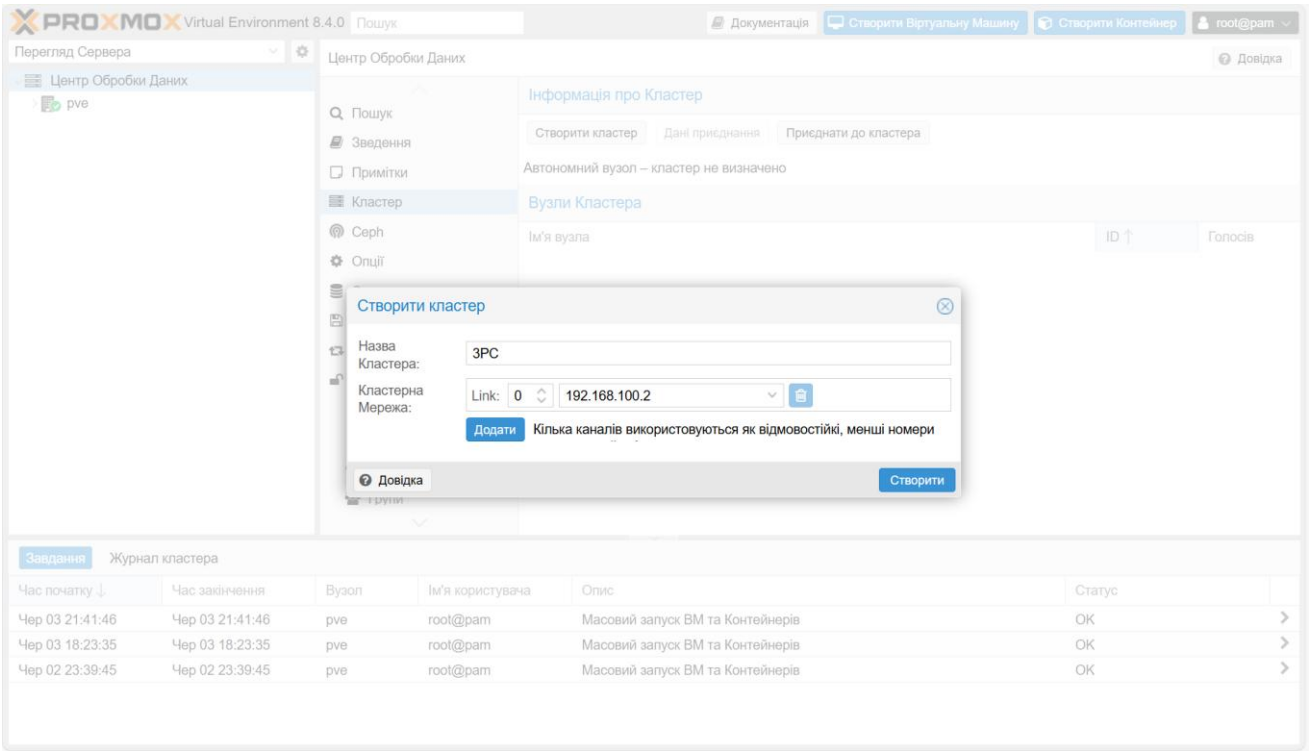


Рисунок 3.10 – Створення кластеру

Створюємо кластер. Після створення кластеру ми отримаємо дані (рисунок 3.11) для приєднання на сервер. Ці дані ми копіюємо та у іншому сервері вводимо. Розпочнеться приєднання до кластера, чекаємо поки не відбудеться приєднання (рисунок 3.12).

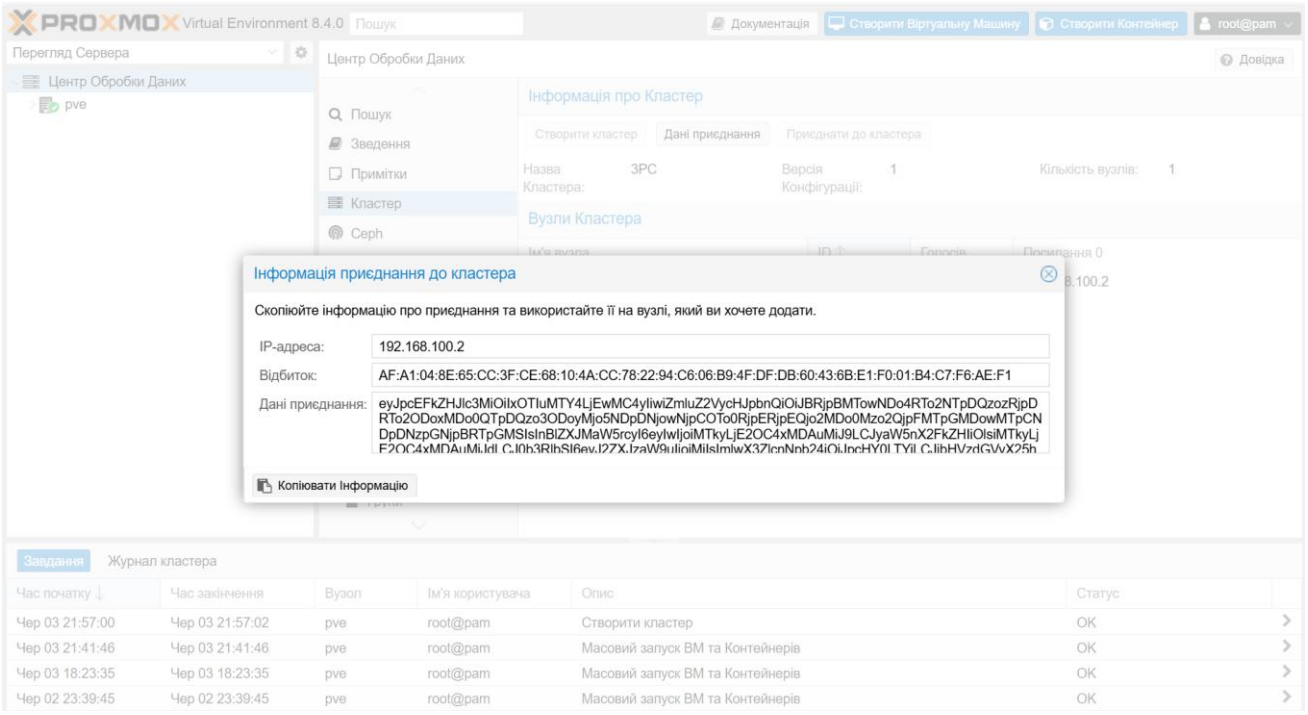


Рисунок 3.11 – Інформація приєднання до кластеру

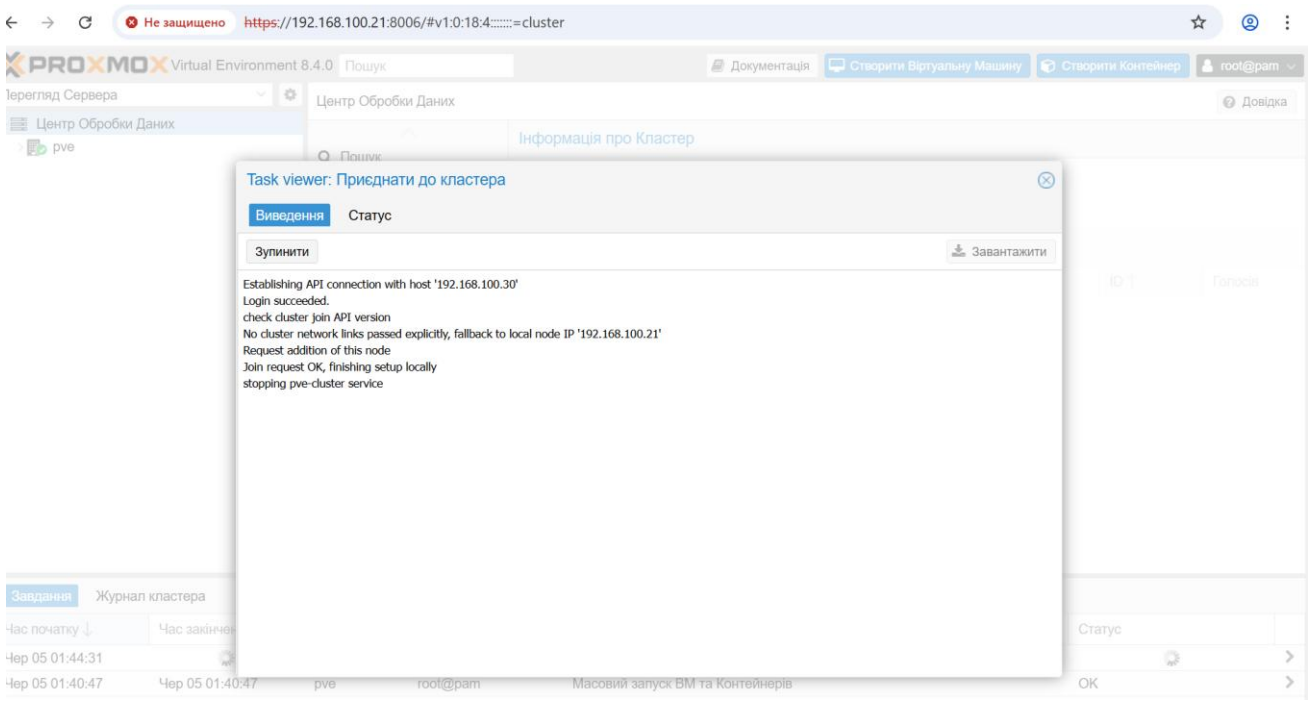


Рисунок 3.12 – Приєднання до кластеру

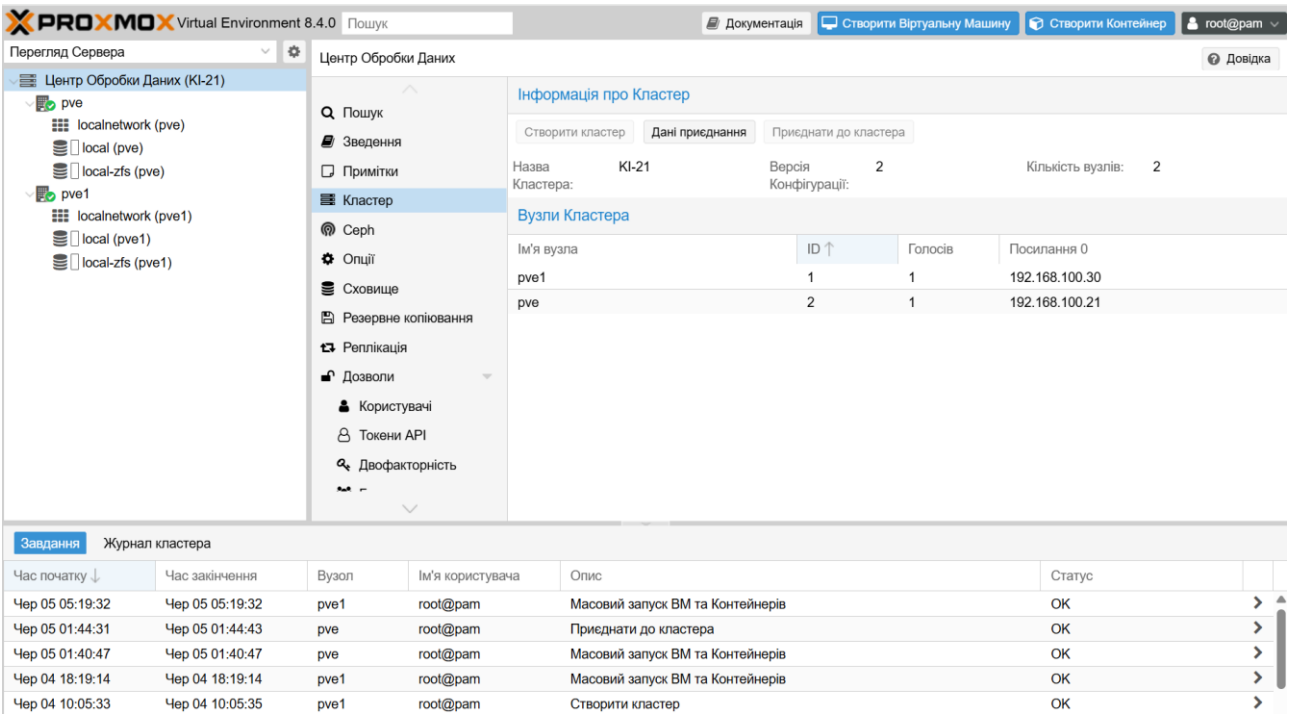


Рисунок 3.13 – Кластер

Завантажимо на наш кластер Ubuntu. У вузлі pve1 відкриваємо local – ISO Образи – Завантажити та завантажуємо на наш кластер операційну систему Ubuntu як показано на рисунку 3.14:

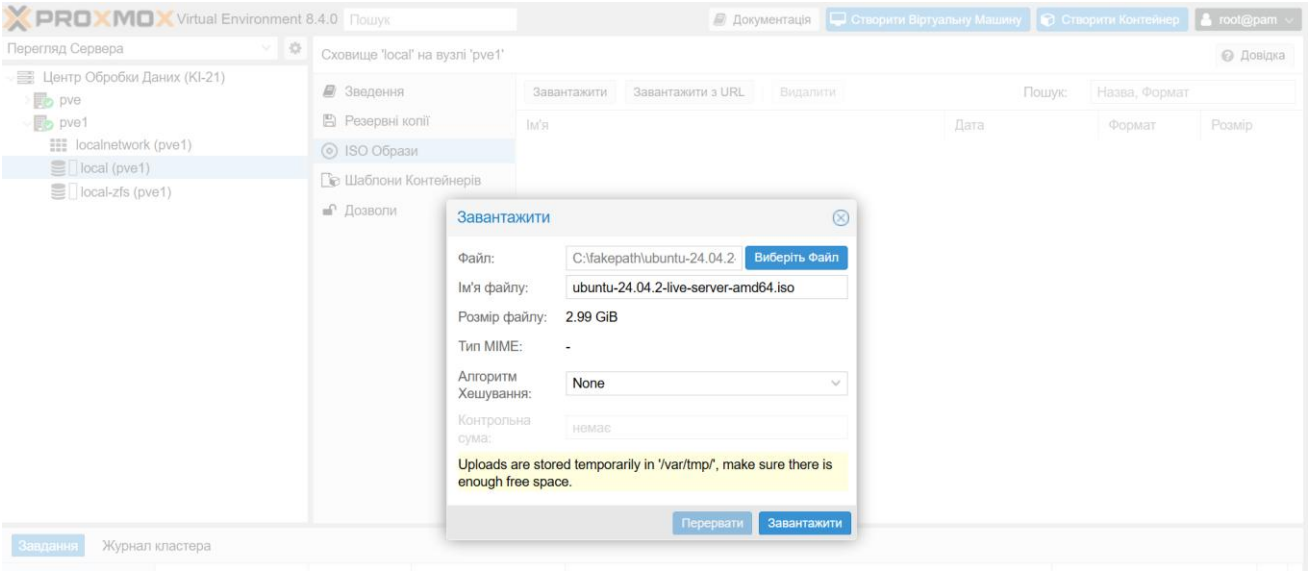


Рисунок 3.14 – Завантаження ISO образу

Далі ми створимо віртуальну машину. Для цього ми натискаємо Створити віртуальну машину та виконаємо наступні кроки як на рисунках 3.15 – 3.21:

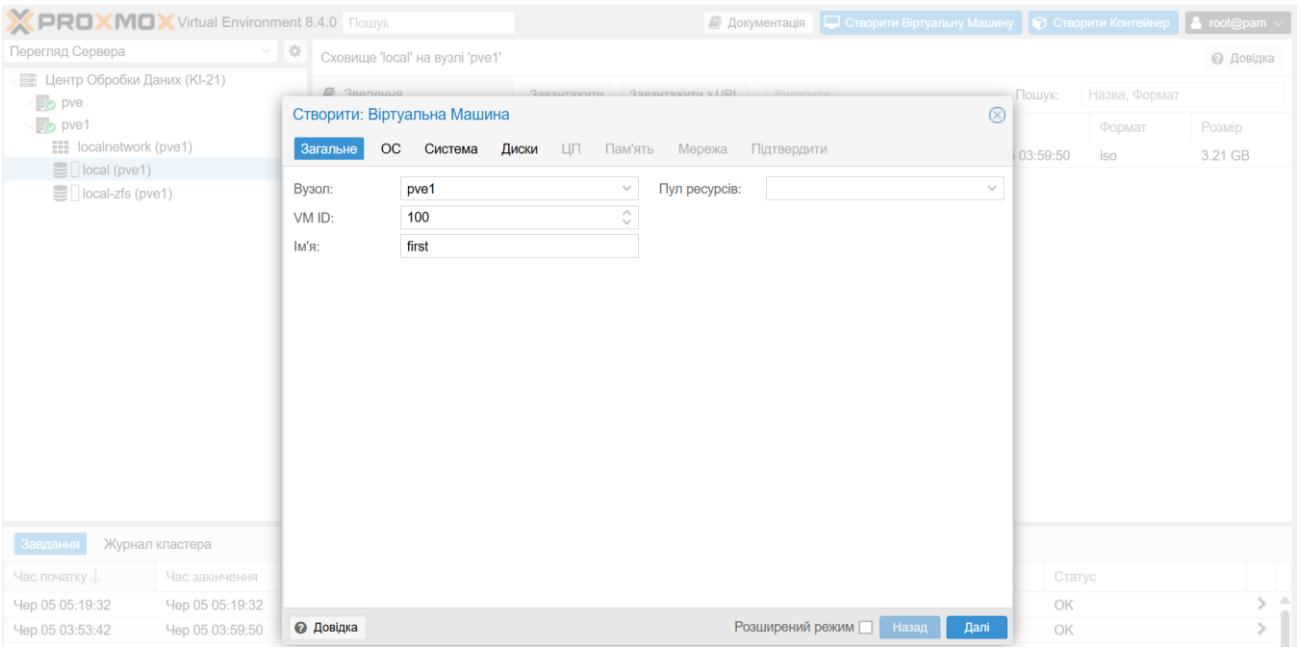


Рисунок 3.15 – Створення віртуальної машини. Загальне

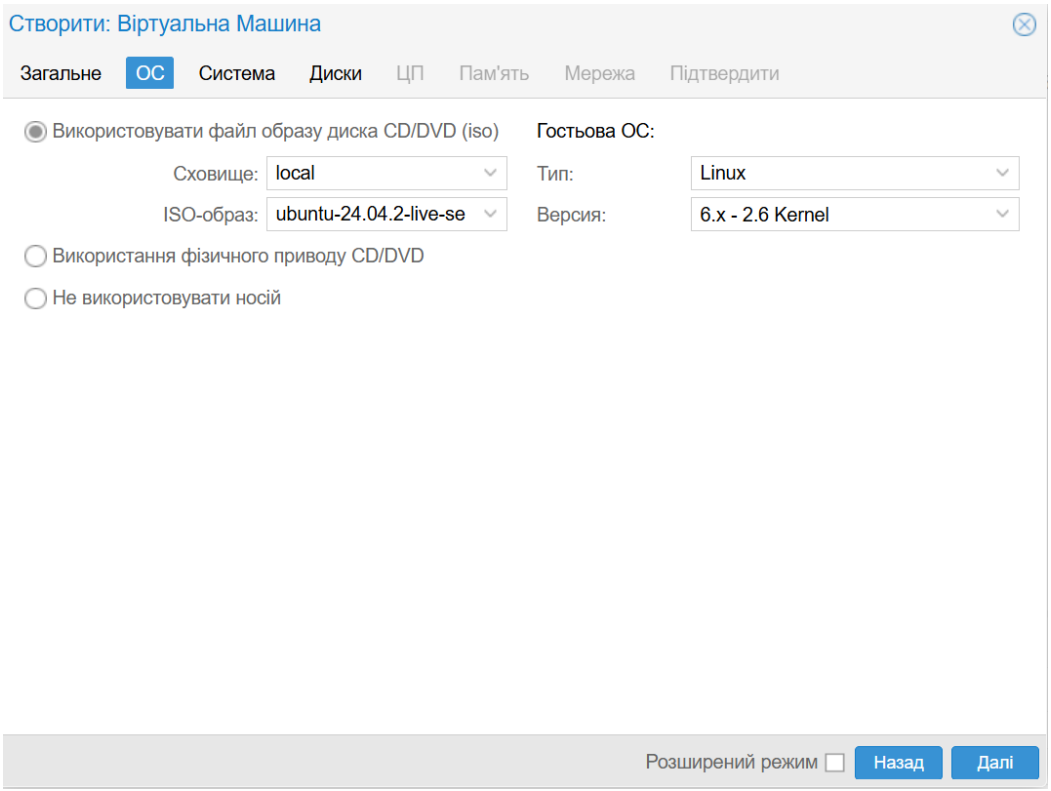


Рисунок 3.16 – Створення віртуальної машини. Операційна система

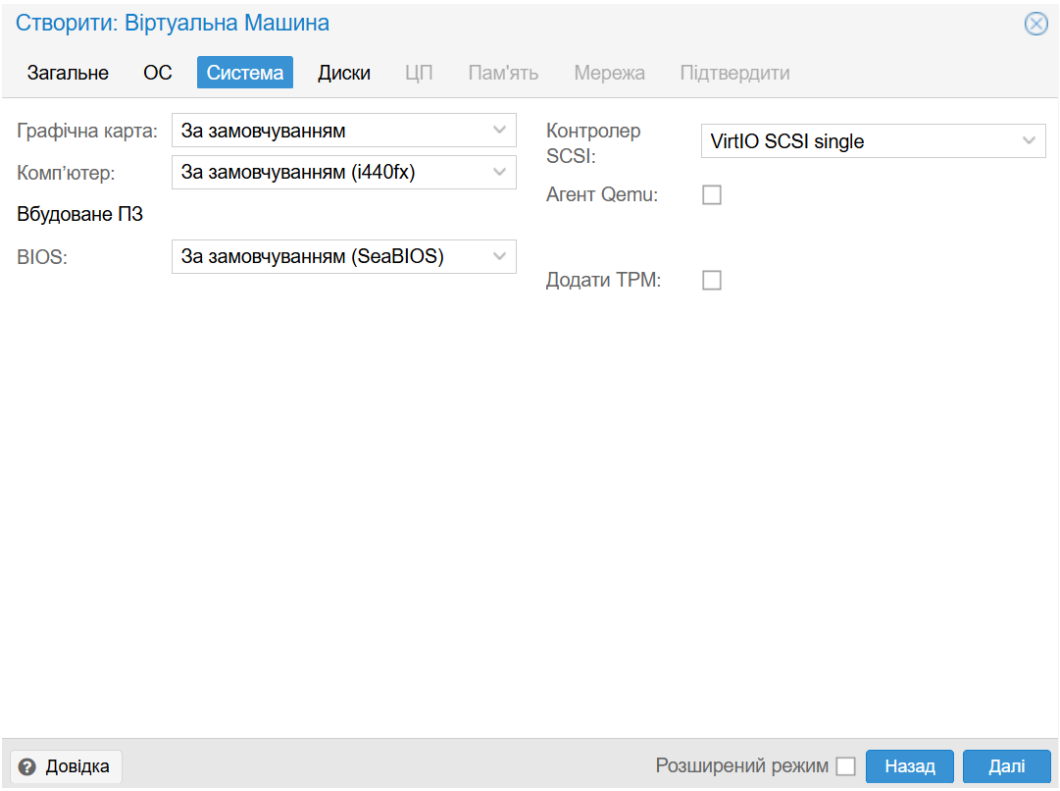


Рисунок 3.17 – Створення віртуальної машини. Система

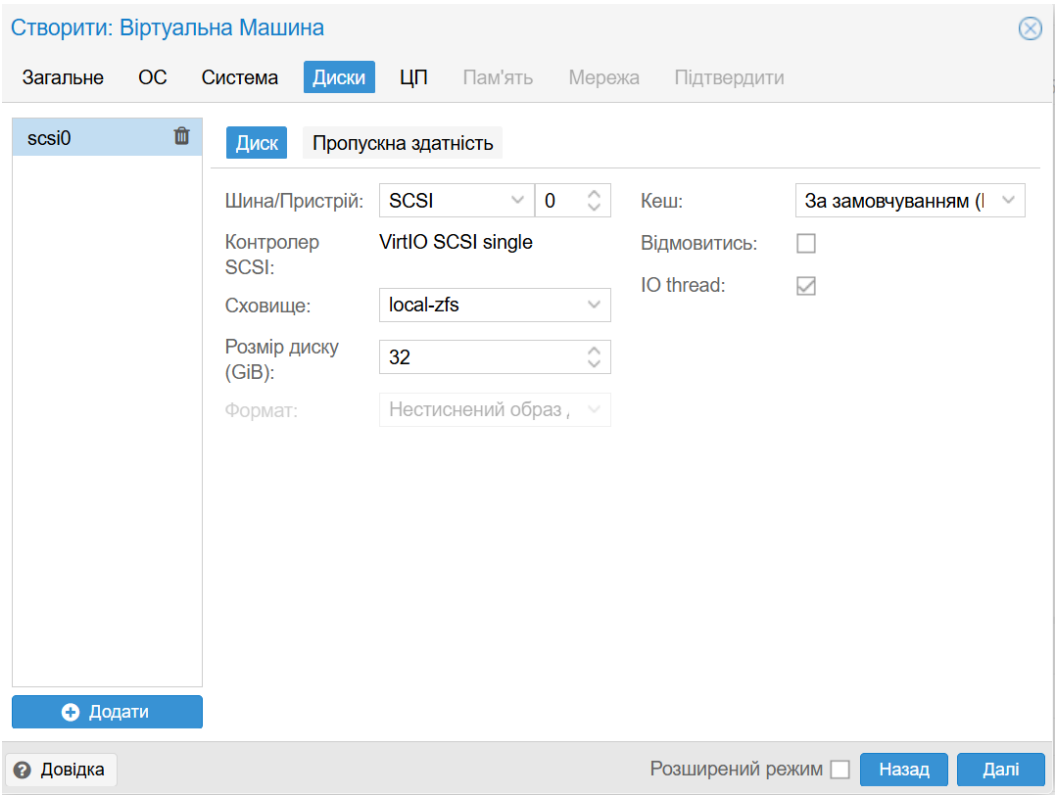


Рисунок 3.18 – Створення віртуальної машини. Диски

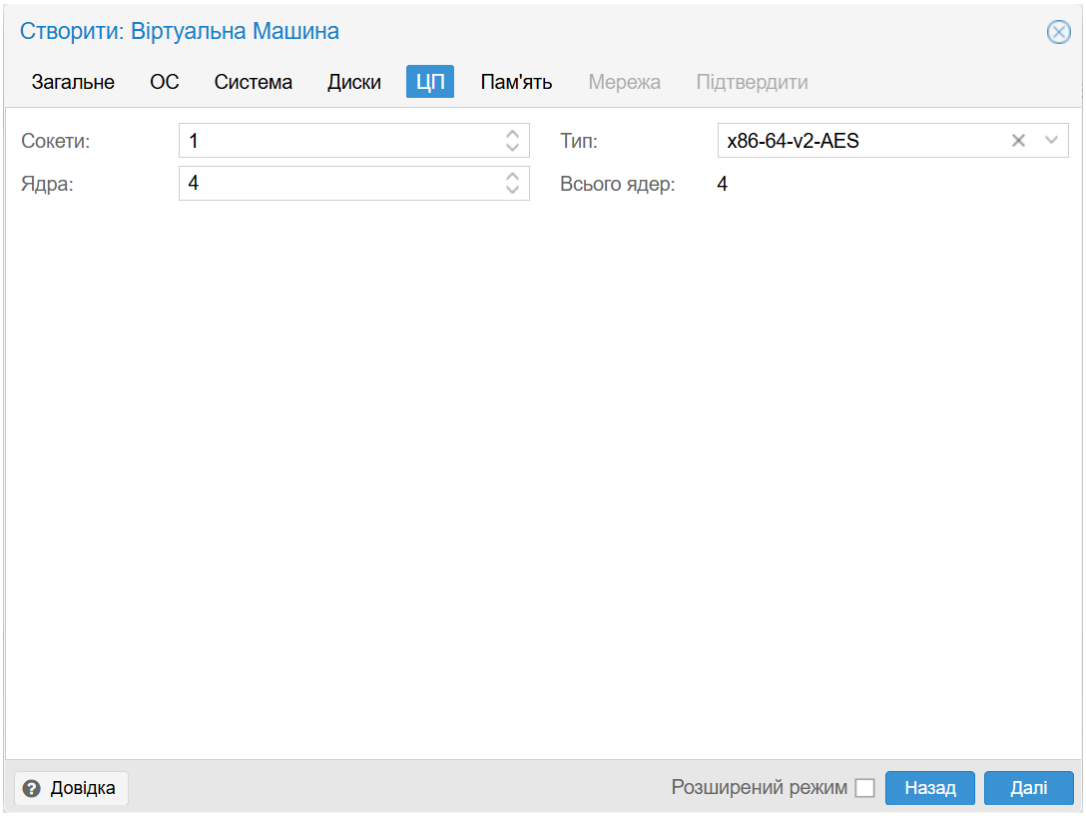


Рисунок 3.19 – Створення віртуальної машини. Центральний процесор

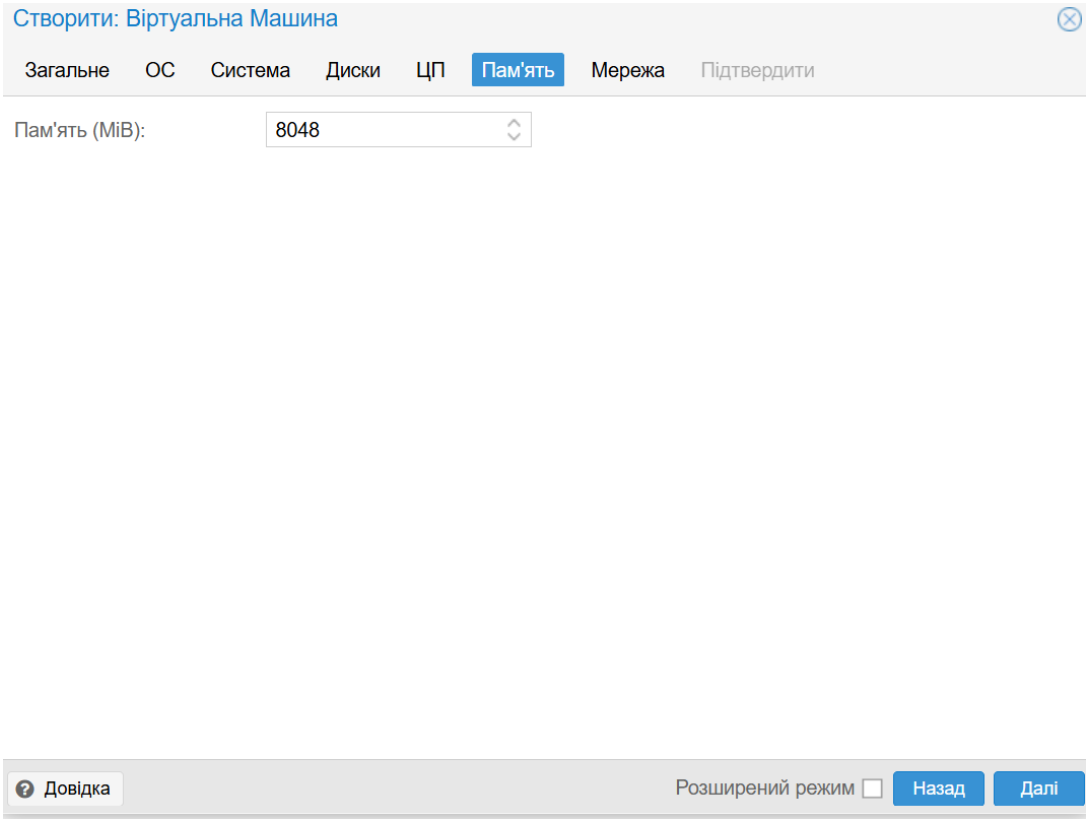


Рисунок 3.20 – Створення віртуальної машини. Оперативна пам’ять

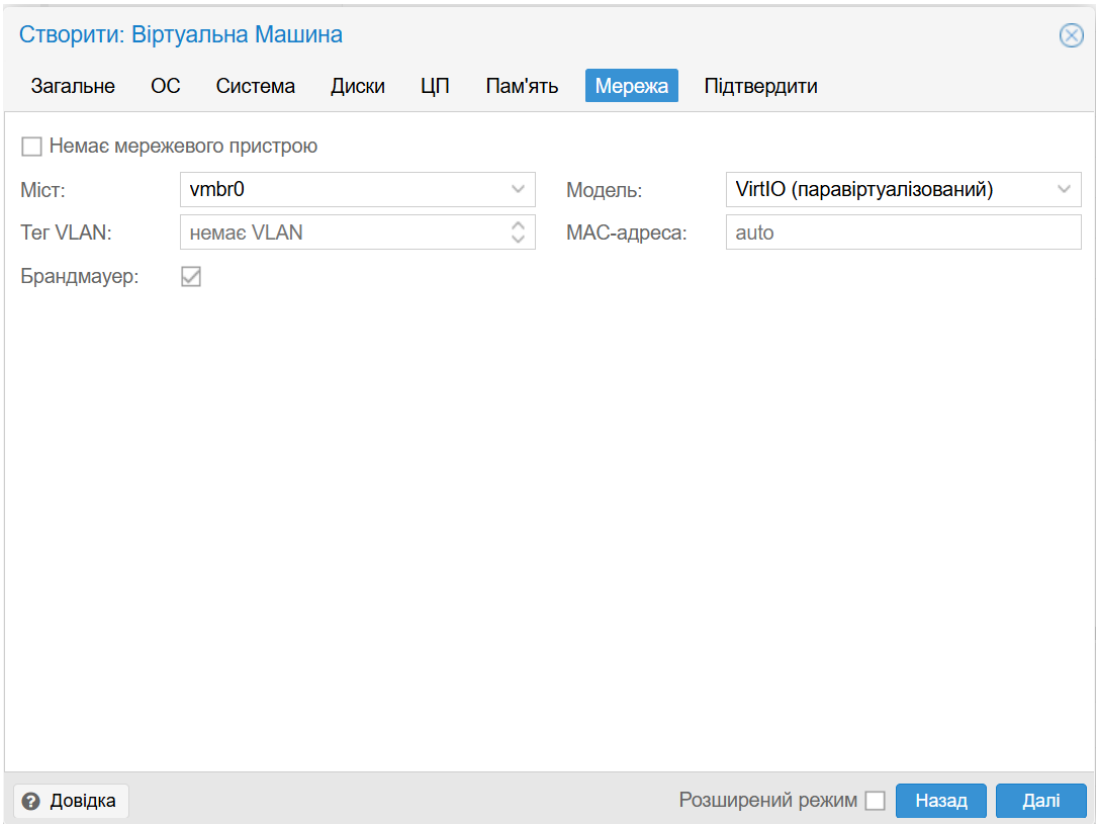


Рисунок 3.21 – Створення віртуальної машини. Мережа

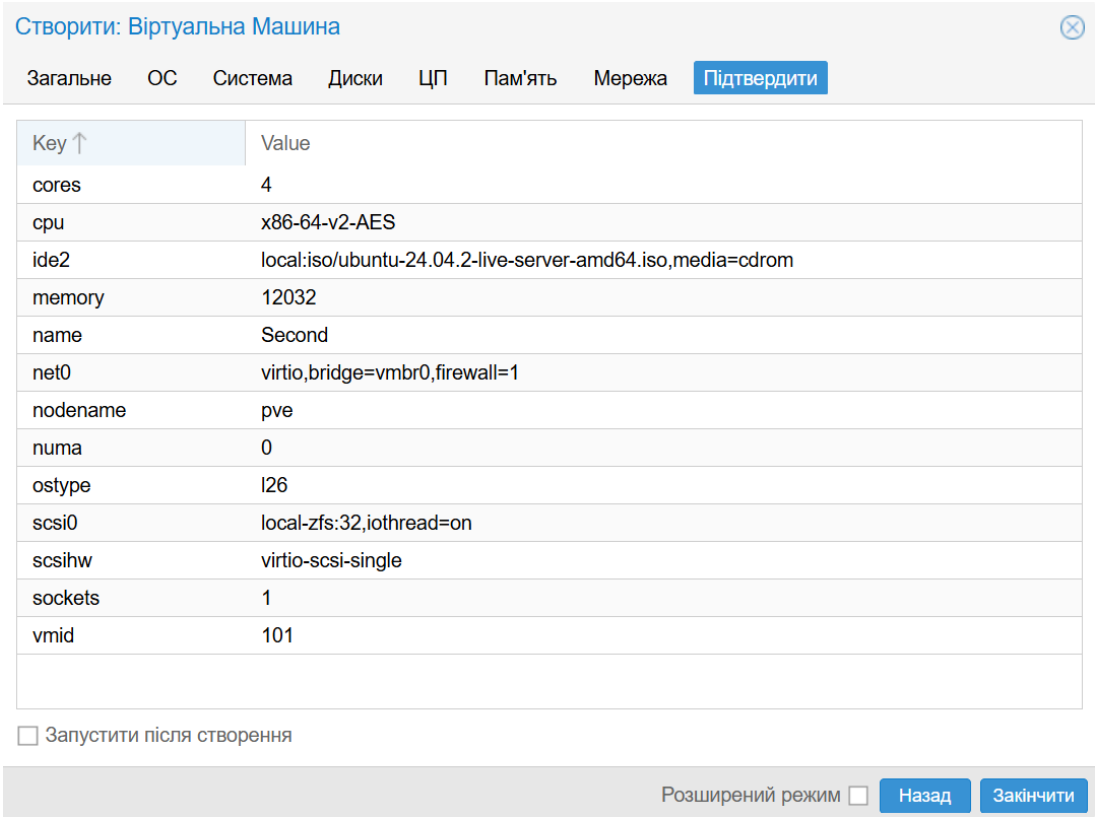


Рисунок 3.22 – Створення віртуальної машини. Загальна інформація

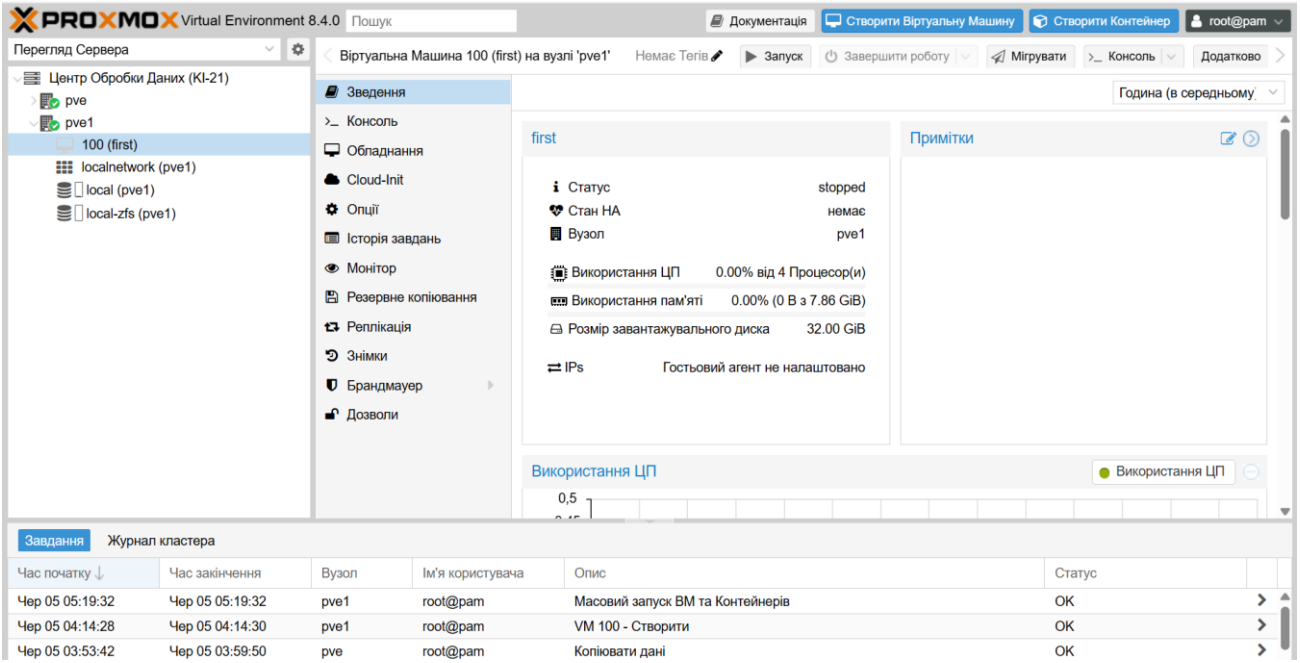


Рисунок 3.21 – Створена віртуальна машина

3.3 Тестування кластеру

Протестуємо кластер. Тестування кластера потрібне для того, щоб гарантувати його надійність, стабільність і безпечну роботу в реальних умовах. Тест дозволяє знайти конфігураційні або апаратні проблеми до того, як вони вплинуть на користувачів або бізнес. Дає впевненість, що система витримає збій вузла, мережі або сховища – без втрати даних і сервісів. Допомагає оцінити, чи здатна система рівномірно розподіляти навантаження та адаптуватись до росту. Можна побачити, скільки ресурсів реально доступно, як кластер справляється з піковим навантаженням. Тестування доступу, ролей, шифрування – щоб уникнути несанкціонованого доступу. Дає змогу виробити процедури відновлення і автоматизацію (HA, резервне копіювання). Перед оновленням Proxmox чи ядер – тест на ізольованому кластері дозволяє уникнути проблем. Тестування – це гарантія того, що твій кластер буде працювати стабільно в будь-якій ситуації. Кластер тестується за певними критеріями:

- Відмово стійкість
- Балансування навантаження
- Продуктивність
- Консистентність даних
- Логування та моніторинг
- Коректність роботи серверів
- Безпека
- Тестування під навантаженням

Розпочнімо тестування.

1. Відмово стійкість.Симулюємо випадковий розрив вузла:

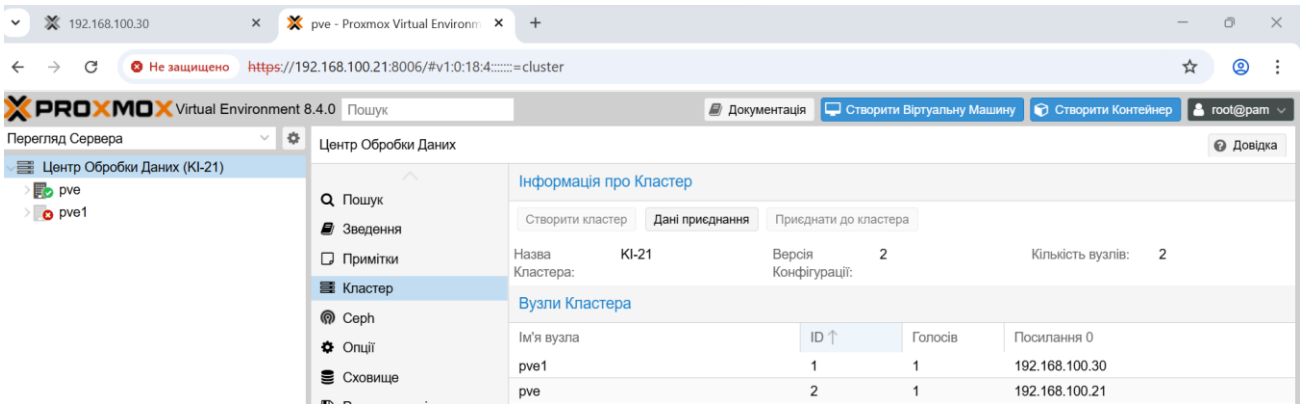


Рисунок 3.22 – Втрачено сервер

Результат:

– Інтерфейс Proxmox залишається доступним.
– У журналі та у перегляді сервера є повідомлення про втрату вузла, але кластер функціонує.

2. Балансування навантаження:

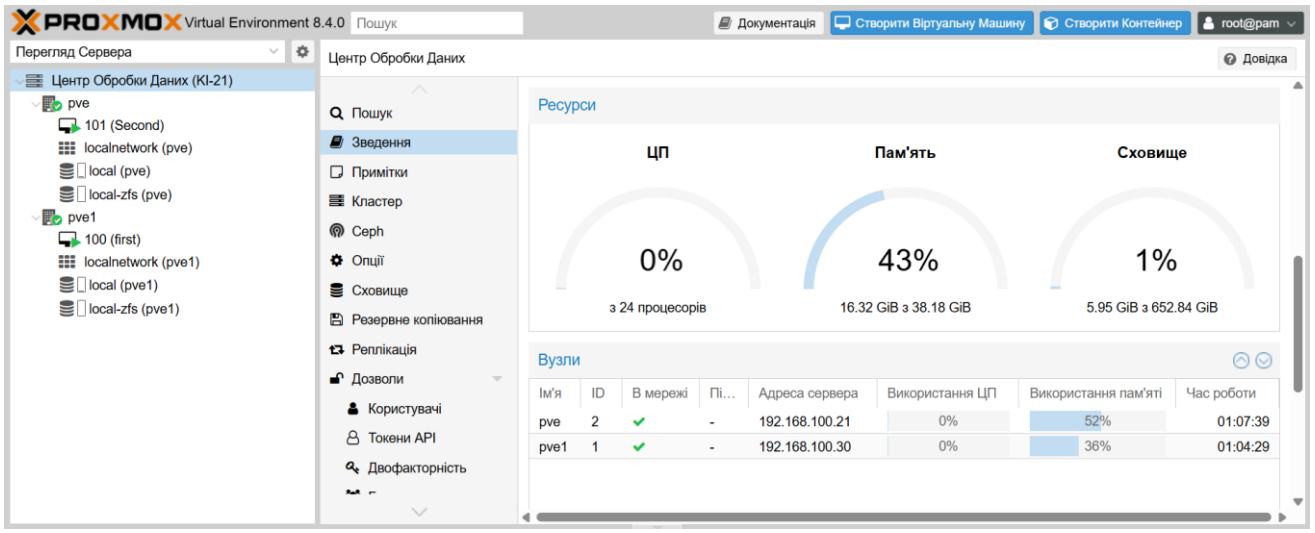


Рисунок 3.23 – Ресурси. Використання та розподілення ресурсів між серверами

На pve навантаження приходиться більше тому, що має більший об’єм pve ОП.

3. Продуктивність. Результати роботи кластера протягом часу:

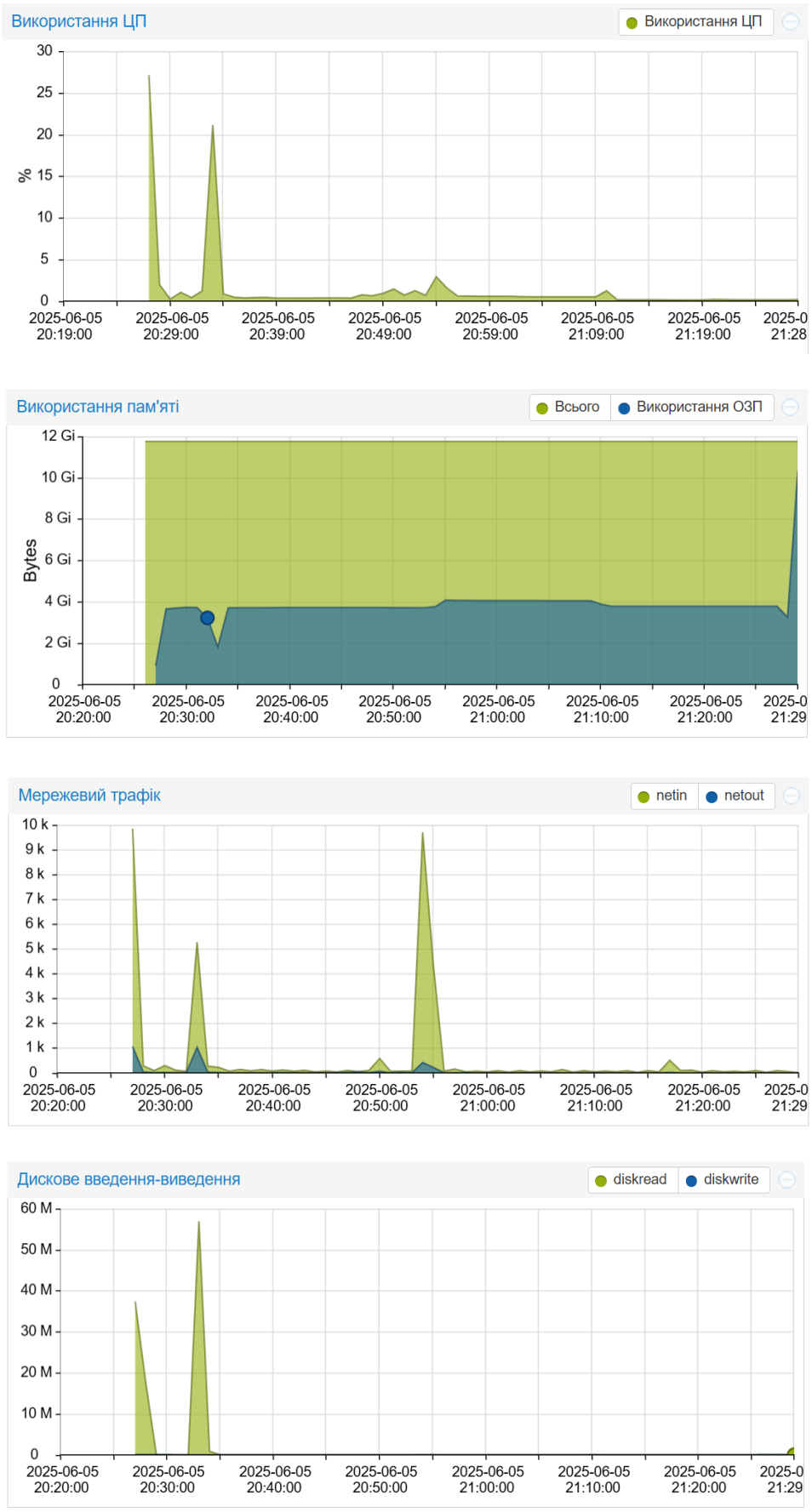
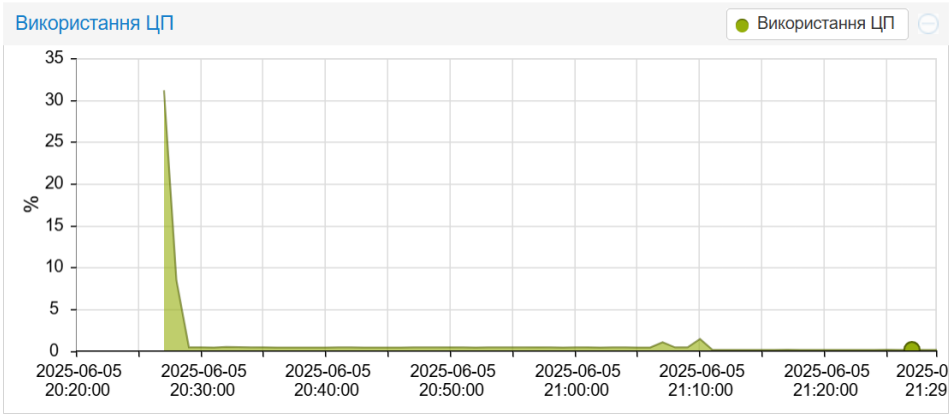
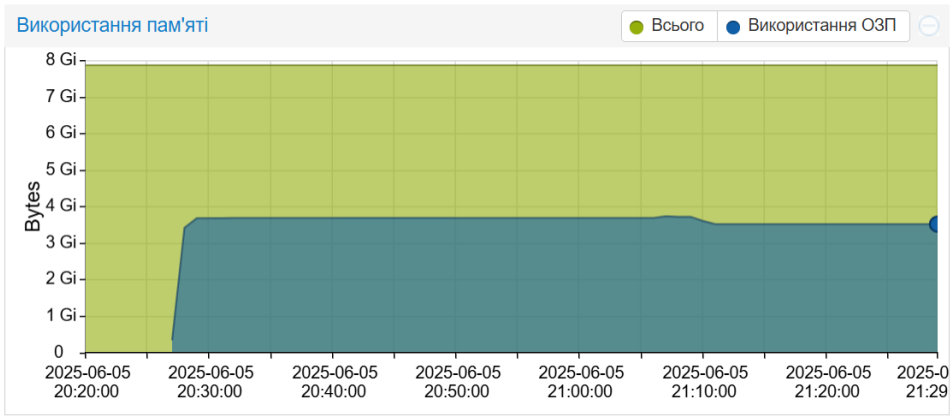
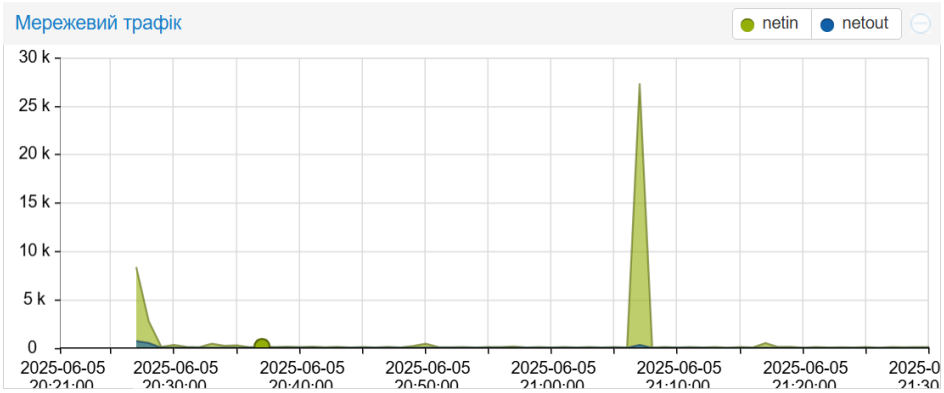


Рисунок 3.24(а) – Продуктивність протягом години pve сервера







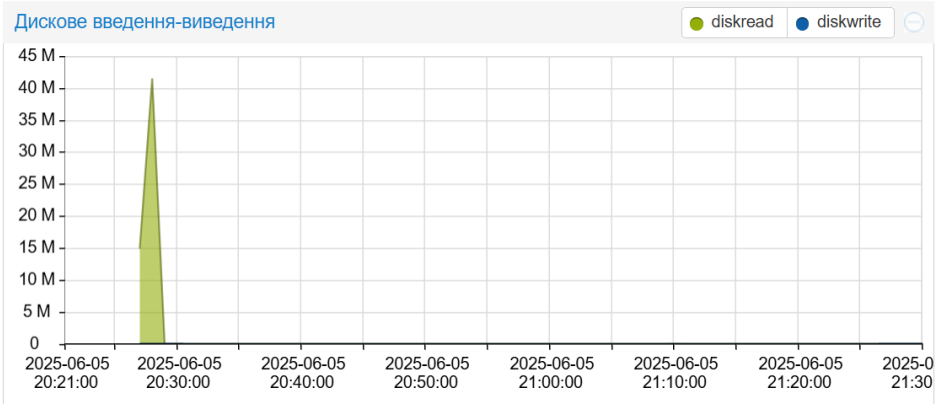


Рисунок 3.24(б) – Продуктивність протягом години pve1 сервера

4. Тестування під навантаженням. Запустивши Ubuntu Server на віртуальній машині, проводимо тест у консолі:

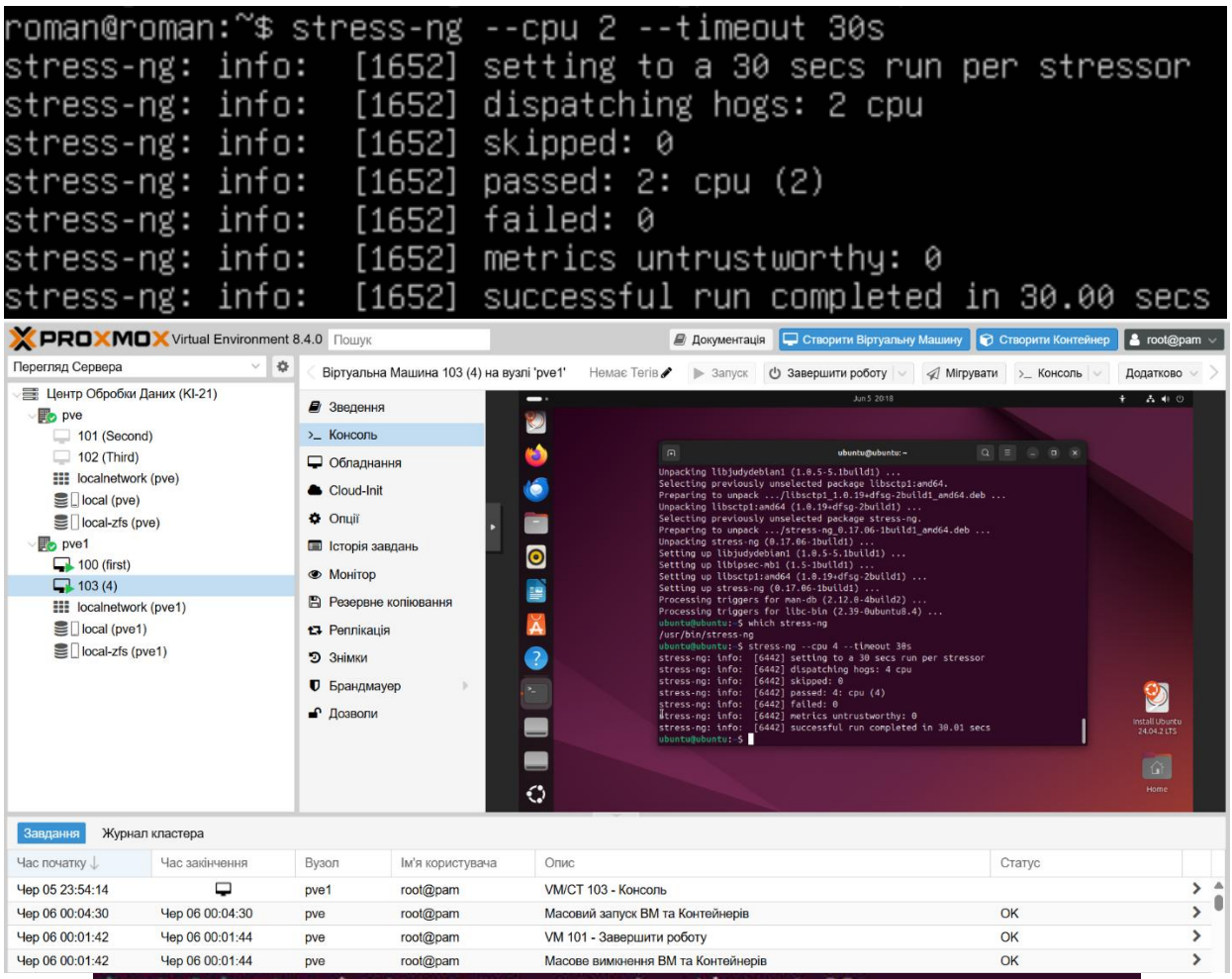


Рисунок 3.25 – Стрес тест протягом 30 секунд

5. Тест пам’яті:

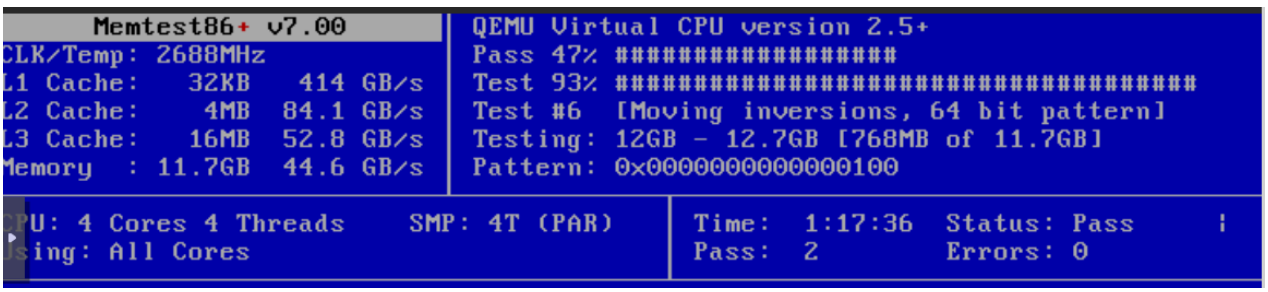


Рисунок 3.26 – Тест пам’яті протягом однієї години

6. Перевірка доступності вузлів:

```
ubuntu@ubuntu:~$ ping 192.168.100.21
PING 192.168.100.21 (192.168.100.21) 56(84) bytes of data.
64 bytes from 192.168.100.21: icmp_seq=1 ttl=64 time=1.17 ms
64 bytes from 192.168.100.21: icmp_seq=2 ttl=64 time=1.07 ms
64 bytes from 192.168.100.21: icmp_seq=3 ttl=64 time=0.759 ms
64 bytes from 192.168.100.21: icmp_seq=4 ttl=64 time=0.708 ms
64 bytes from 192.168.100.21: icmp_seq=5 ttl=64 time=0.485 ms
64 bytes from 192.168.100.21: icmp_seq=6 ttl=64 time=0.612 ms
64 bytes from 192.168.100.21: icmp_seq=7 ttl=64 time=0.941 ms
64 bytes from 192.168.100.21: icmp_seq=8 ttl=64 time=1.59 ms
64 bytes from 192.168.100.21: icmp_seq=9 ttl=64 time=1.01 ms
64 bytes from 192.168.100.21: icmp_seq=10 ttl=64 time=1.61 ms
64 bytes from 192.168.100.21: icmp_seq=11 ttl=64 time=1.29 ms
64 bytes from 192.168.100.21: icmp_seq=12 ttl=64 time=0.577 ms
64 bytes from 192.168.100.21: icmp_seq=13 ttl=64 time=1.45 ms
64 bytes from 192.168.100.21: icmp_seq=14 ttl=64 time=1.46 ms
64 bytes from 192.168.100.21: icmp_seq=15 ttl=64 time=1.02 ms
64 bytes from 192.168.100.21: icmp_seq=16 ttl=64 time=1.07 ms
64 bytes from 192.168.100.21: icmp_seq=17 ttl=64 time=0.840 ms
64 bytes from 192.168.100.21: icmp_seq=18 ttl=64 time=1.39 ms
64 bytes from 192.168.100.21: icmp_seq=19 ttl=64 time=1.03 ms
64 bytes from 192.168.100.21: icmp_seq=20 ttl=64 time=0.529 ms
64 bytes from 192.168.100.21: icmp_seq=306 ttl=64 time=1.35 ms
^C
--- 192.168.100.21 ping statistics ---
306 packets transmitted, 306 received, 0% packet loss, time 307341ms
rtt min/avg/max/mdev = 0.272/1.077/1.681/0.303 ms
```

Рисунок 3.27 – Перевірка доступності вузлів

7. Час працювання системи, системне навантаження:

```
ubuntu@ubuntu:~$ uptime
 20:36:34 up 41 min,  1 user,  load average: 0.02, 0.05, 0.08
```

Рисунок 3.28 – Час працювання системи та системне навантаження

8. Навантаження на диск:

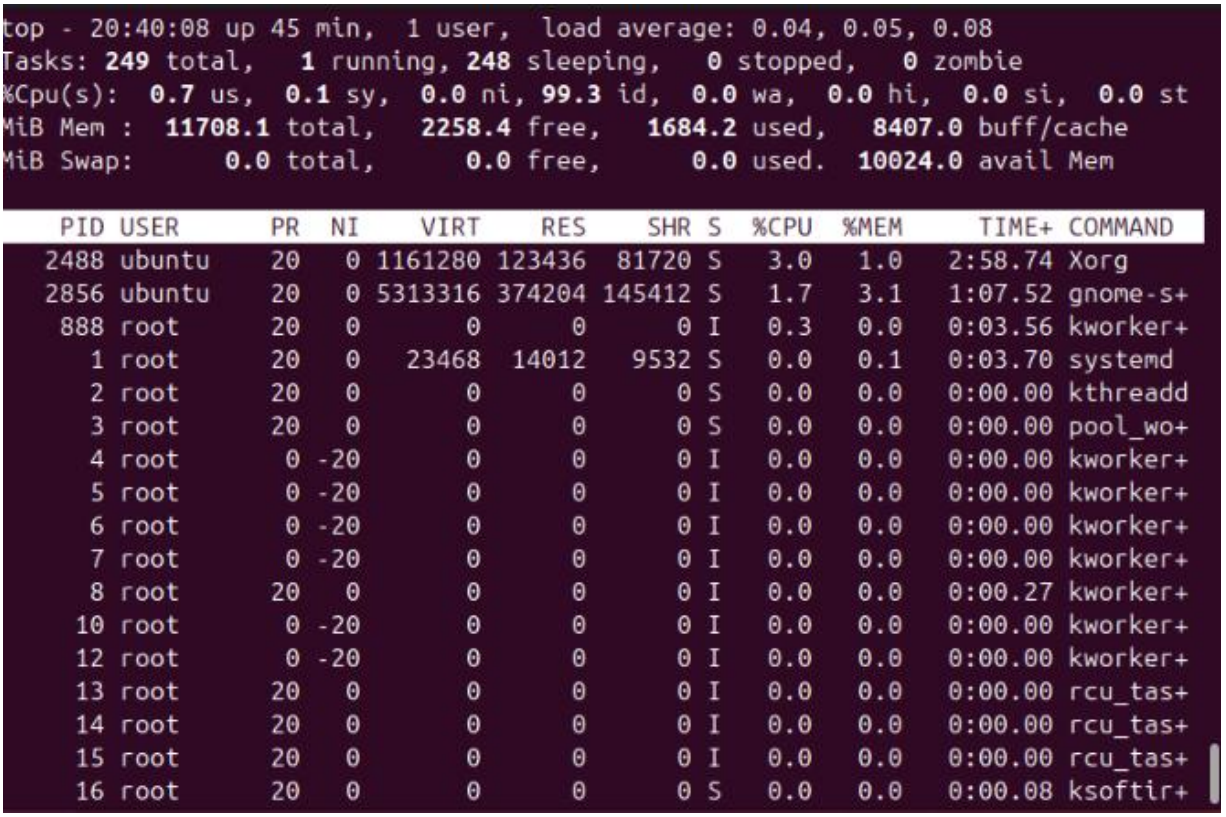


Рисунок 3.29 – Навантаження на диск

9. Безпека:

Відкритим був лише порт 8006 так як він був необхідним для завантаження Proxmox сторонніх портів не було виявлено.

ВИСНОВОК

У процесі дослідження було здійснено повноцінний аналіз і практичне тестування системи віртуалізації Proxmox VE з позицій її технічної реалізації, ефективності та доцільності впровадження у сучасних IT-інфраструктурах.

У першій частині роботи розглянуто сучасні підходи до віртуалізації, а також проведено порівняння Proxmox VE з альтернативними системами — VirtualBox, VMware vSphere, Hyper-V, XenServer. Встановлено, що завдяки підтримці KVM і LXC, відкритому коду, централізованому керуванню та гнучкості налаштування, Proxmox VE є потужною платформою для корпоративного середовища. Також визначено її релевантність для українських підприємств, зокрема у воєнний час, як інструмент імпортозаміщення та підвищення кібербезпеки.

Другий розділ містить проєктування апаратної частини кластерної інфраструктури, вибір обладнання (на прикладі трьох різних вузлів), побудову топології та схеми розгортання Proxmox VE 8.4. Наведено методику налаштування кластера, резервного копіювання, а також сформульовано критерії оцінки ефективності системи — зокрема за показниками надійності, масштабованості, безпеки та економічної доцільності. Було обґрунтовано, що навіть за використання доступного обладнання, можна досягти високої продуктивності кластерної інфраструктури з підтримкою HA (високої доступності).

У третьому розділі проведено практичні вимірювання: навантажувальне тестування CPU, пам’яті та диску, перевірку доступності вузлів, моніторинг системних ресурсів. Виявлено, що система демонструє стабільну роботу під час стрес-тестів та ефективно перерозподіляє навантаження. Безпековий аналіз засвідчив мінімальний набір відкритих портів, а також стабільність роботи кластерної структури. Також були виявлені окремі недоліки, пов’язані з потребою у точнішій обробці збоїв гостьових ОС і оптимізації плагінів збереження, що були враховані в рекомендаціях щодо вдосконалення системи

Впровадження Proxmox VE дозволяє істотно скоротити витрати на IT-інфраструктуру, забезпечити високий рівень автономності, безпеки, керованості та масштабованості, що робить цю систему надзвичайно перспективною для широкого кола застосувань в Україні — від бізнесу до державного сектору.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] Tonya. Віртуалізація 2024. Революція в IT-сфері. URL: https://ucloud.ua/virtualizacziya/ (дата звернення 31.03.2025)

[2] Детально про віртуалізацію: типи, переваги та рішення. URL: https://onbiz.biz/about-virtualization/ (дата звернення 31.03.2025)

[3] Хмарна вікіпедія. Що таке віртуалізація та які переваги вона надає. URL: https://gigacloud.ua/articles/shho-take-virtualizacziya-ta-yaki-perevagy-vona-nadaye/ (дата звернення 31.03.2025)

[4] Virendra Soni. Порівнюємо найкраще програмне забезпечення для віртуалізації у 2020 році: Hyper-V, KVM, vSphere та XenServer. URL: https://habr.com/ru/companies/otus/articles/510682/ (дата звернення 31.03.2025)

Рис. 1.2.1 URL: https://www.virtualizationhowto.com/2019/12/esxi-vs-vsphere-vs-vcenter/ (дата звернення 31.03.2025)

Рис. 1.2.2 URL: https://en.wikipedia.org/wiki/Hyper-V#/media/File:Hyper-V.png (дата звернення 31.03.2025)

Рис. 1.2.3 URL: https://www.citrix.com/blogs/2016/06/30/xenserver-7-building-the-foundations-of-a-great-future/ (дата звернення 31.03.2025)

Рис. 1.2.4 URL: https://www.researchgate.net/figure/The-architecture-of-KVM-full-virtualized-network-I-O_fig2_294105897 (дата звернення 31.03.2025)

[5] Stackshare. KVM vs LXC. URL: https://stackshare.io/stackups/kvm-vs-lxc (дата звернення 31.03.2025)

[6] DiskInternals Research. Proxmox vs VirtualBox: A Comprehensive Comparison. URL: https://www.diskinternals.com/vmfs-recovery/proxmox-vs-virtualbox/ (дата звернення 31.03.2025)

[7] Офіційний сайт Proxmox. System Requirements. URL: https://www.proxmox.com/en/products/proxmox-virtual-environment/requirements (дата звернення 31.03.2025)

[8] Дорожня карта. Історя випусків. URL: https://pve.proxmox.com/wiki/Roadmap#Proxmox_VE_8.4 (дата звернення 27.04.2025)

[9] Онлайн магазин ROZETKA. USB флеш-накопичувач SanDisk Ultra USB 3.0. Обсягом 256 GB. URL: https://rozetka.com.ua/ua/303090688/p303090688/ (дата звернення 05.05.2025)

[10] Онлайн магазин ROZETKA. USB флеш-накопичувач Kingston DataTraveler Exodia. Обсягом 128 GB. URL: https://rozetka.com.ua/ua/kingston-dtxm-128gb/p340729138/ (дата звернення 05.05.2025)

[11] Онлайн магазин ROZETKA. USB флеш-накопичувач SanDisk SDCZ74. Обсягом 64 GB. URL: https://rozetka.com.ua/ua/sandisk_sdcz74_064g_g46/p130660570/ (дата звернення 05.05.2025)

[12] Онлайн магазин ROZETKA. USB флеш-накопичувач TOSHIBA U401. Обсягом 8GB. URL: https://i-mobile.ge/products/2292 (дата звернення 05.05.2025)
[13] ACER. Ноутбук Acer Aspire 7 A715-42G. URL: https://www.acer.com/ua-uk/laptops/aspire/aspire-7-amd/pdp/NH.QBFEU.00C (дата звернення 05.05.2025)
[14] ASUS. Ноутбук Asus TUF Gaming F17. URL: https://www.asus.com/laptops/for-gaming/tuf-gaming/asus-tuf-gaming-f17/ (дата звернення 05.05.2025)
[15] HP. Ультрабук hp spectre x360 14 2-in-1 laptop. URL: https://www.hp.com/us-en/shop/mdp/envy---omen/hp-spectre-x360-14-3074457345617526676--1 (дата звернення 05.05.2025)
[16] KVANTECH. Оптичний термінал Huawei EchoLife EG8012H5. URL: https://kvan.tech/catalog/setevoe-oborudovanie-i-organizatsiya-setey/oborudovanie-opticheskikh-liniy-svyazi/opticheskie-abonentskie-terminaly/opticheskiy-terminal-huawei-echolife-eg8012h5/ (дата звернення 05.05.2025)
[17] Онлайн магазин ROZETKA. Патч корд Cablexpert CAT6 UTP Сірий (PP6U-1M). URL: https://rozetka.com.ua/ua/cablexpert_pp6u_1m/p241174099/ (дата звернення 05.05.2025)
[18] Онлайн магазин ROZETKA. Патч корд Cablexpert CAT5e UTP 1.5 м Синій (PP12-1.5M/B). URL: https://rozetka.com.ua/ua/cablexpert_pp12_15m_b/p15852133/ (дата звернення 05.05.2025)
[19] PROXMOX. Proxmox VE 8.4 ISO Installer. URL: https://www.proxmox.com/en/downloads/proxmox-virtual-environment/iso/proxmox-ve-8-4-iso-installer (дата звернення 05.05.2025)
[20] Ubuntu. Ubuntu Server. URL: https://ubuntu.com/download/server#release-notes-lts (дата звернення 05.05.2025)