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

ПОЯСНЮВАЛЬНА ЗАПИСКА

до випускної кваліфікаційної роботи бакалавра

на тему: «Проєкт побудови масштабованої системи автоматизованого розгортання операційних систем»

Виконав: студент 4-го курсу групи КІ-22-1
спеціальності 123 «Комп’ютерна інженерія»
_____ Євгеній Гречаний
Керівник: Старший викладач кафедри КІ та КБ
_____ Ігор Фальковський
Рецензент: Доцент кафедри КІ та КБ, к.пед.н., доцент
_____ Олена Головня

Житомир – 2026

Державний університет «Житомирська політехніка»
Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки
Спеціальність 123 «Комп'ютерна інженерія»
Освітня програма «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ
Завідувач
кафедри комп'ютерної
інженерії та кібербезпеки
Андрій Єфіменко
2 лютого 2026 р.

ЗАВДАННЯ
на випускну кваліфікаційну роботу бакалавра

Студент: Гречаний Євгеній Сергійович

Тема роботи: «Проєкт побудови масштабованої системи автоматизованого розгортання операційних систем.»

затверджена Наказом університету від 2 лютого 2026 р. №57/с

Термін здачі студентом закінченої роботи 9 червня 2026 р.

Вихідні дані роботи (зазначається предмет і об'єкт дослідження): Технології автоматизованого розгортання операційних систем WDS, MDT, PXE та серверна інфраструктура Windows Server.

Обладнання та програмне забезпечення:

1. Персональний комп'ютер із підтримкою віртуалізації під управлінням ОС Winbdows 10 ProD (Intel Core I7(4) 16 Gb RAM).
2. Серверна операційна система Windows Server 2022 Standard Evaluation.
3. Клієнтська операційна система Windows 10 Pro.
4. Програмне забезпечення Microsoft Deployment Toolkit (MDT) 6.3.8456.1000, Windows ADK 10.1.20348.3694, Windows ADK PE 10.1.20348.3694.
5. Гіпервізор Oracle VM VirtualBox 7.2.4 r170995 (Qt6.8.0 on windows).

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

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Ігор Фальковський	29.03.2026	29.03.2026
2	Ігор Фальковський	18.04.2026	18.04.2026
3	Ігор Фальковський	27.05.2026	27.05.2026

Студент _____ Євгеній Гречаний
Керівник _____ Ігор Фальковський

Календарний план

№ з/п	Назви етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постановка завдання	10.02.2026	виконав
2	Пошук, огляд та аналіз літературних джерел	22.02.2026	виконав
3	Формулювання та постановка технічного завдання	04.03.2026	виконав
4	Розробка плану кваліфікаційної роботи	14.03.2026	виконав
5	Огляд та порівняння аналогічних технологій	29.03.2026	виконав
6	Проектування архітектури масштабованої системи розгортання	16.04.2026	виконав
7	Практична реалізація автоматизованого розгортання	27.04.2026	виконав
8	Тестування роботи системи	18.05.2026	виконав
9	Написання пояснювальної записки	28.05.2026	виконав
10	Захист	16.06.2026	

Студент _____ Євгеній Гречаний
Керівник _____ Ігор Фальковський

АНОТАЦІЯ

У кваліфікаційній роботі бакалавра представлено розробку масштабованої системи автоматизованого розгортання операційних систем на базі технологій Microsoft Deployment Toolkit та Windows Deployment Services.

Пояснювальна записка складається з трьох розділів, висновків, списку використаної літератури та додатків. Кваліфікаційна робота містить 109 сторінок, 95 рисунків, 9 таблиць. Список використаних джерел налічує 21 найменування і розміщений на 2 сторінках.

У першому розділі проаналізовано існуючі підходи до розгортання операційних систем, розглянуто сучасні інструменти та технології, а також особливості автоматизації в корпоративних середовищах.

У другому розділі виконано проєктування та розгортання серверної інфраструктури системи автоматизованого встановлення ОС. Обґрунтовано вибір віртуалізації та програмного забезпечення, розроблено логічну мережеву структуру та IP-адресацію, а також налаштовано базові мережеві сервіси й серверну інфраструктуру для роботи WDS і MDT.

У третьому розділі наведено практичну реалізацію системи розгортання, результати її тестування, а також оцінку продуктивності та ефективності автоматизованого розгортання.

Ключові слова: АВТОМАТИЗОВАНЕ РОЗГОРТАННЯ ОС, MDT, WDS, PXE, ACTIVE DIRECTORY, WINDOWS SERVER, МЕРЕЖЕВЕ РОЗГОРТАННЯ, ВІРТУАЛІЗАЦІЯ, ІНФРАСТРУКТУРА, СИСТЕМНЕ АДМІНІСТРУВАННЯ.

					ККІКБ.КР.Б – 123 – 03 – ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Гречаний Є.С.		8.06.26	Проєкт побудови масштабованої системи автоматизованого розгортання операційних систем	Літ.	Арк. 4 / Аркушів 109
Керівник		Фальковський І.Г.		8.06.26			
Реценз.		Головня О.С.		8.06.26		ФІКТ, Гр. КІ-22-1	
Н. контр.		Фальковський І.Г.		8.06.26			
Затверд.		Єфіменко А.А.		8.06.26			

ANNOTATION

In the bachelor's qualification thesis, a scalable system for automated operating system deployment based on Microsoft Deployment Toolkit and Windows Deployment Services technologies is presented.

The explanatory note consists of three chapters, conclusions, a list of references, and appendices. The thesis contains 109 pages, 95 figures, and 9 tables. The list of references includes 21 sources and is presented on 2 pages.

The first chapter analyzes existing approaches to operating system deployment, reviews modern tools and technologies, as well as features of automation in corporate environments.

The second chapter presents the design and deployment of the server infrastructure for the automated operating system deployment system. The choice of virtualization environment and software is justified, a logical network structure and IP addressing scheme are developed, and basic network services and server infrastructure required for WDS and MDT operation are configured.

The third chapter presents the practical implementation of the deployment system, test results, and an assessment of the performance and efficiency of the automated deployment process.

Keywords: AUTOMATED OS DEPLOYMENT, MDT, WDS, PXE, ACTIVE DIRECTORY, WINDOWS SERVER, NETWORK DEPLOYMENT, VIRTUALIZATION, INFRASTRUCTURE, SYSTEM ADMINISTRATION.

ЗМІСТ

ВСТУП..... 9
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ АВТОМАТИЗОВАНОГО РОЗГОРТАННЯ ОС 11
1.1 Огляд сучасних методів та стратегій розгортання операційних систем..... 11
1.2 Архітектура та теоретичні принципи роботи Windows Deployment Services 15
1.3 Архітектура та теоретичні принципи роботи Microsoft Deployment Toolkit. 23
1.4 Аналіз архітектури системи розгортання ОС та обґрунтування вимог на базі WDS та MDT 28
1.5 Аналіз переваг і недоліків WDS та MDT та їх порівняння з Microsoft Endpoint Configuration Manager..... 32
Висновки до першого розділу..... 35
РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА ПОБУДОВА СЕРВЕРНОЇ ІНФРАСТРУКТУРИ СИСТЕМИ РОЗГОРТАННЯ ОС 37
2.1 Вибір середовища віртуалізації для побудови лабораторного стенду..... 37
2.2 Аналіз та вибір програмного забезпечення для побудови інфраструктури .. 42
2.3 Проєктування логічної структури та IP-адресації розподіленої системи розгортання ОС 45
2.4 Налаштування базової мережевої інфраструктури та серверного забезпечення 51
Висновки до другого розділу 59
РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ РОЗПОДІЛЕНОГО РОЗГОРТАННЯ..... 61
3.1 Встановлення Active Directory Domain Services та встановлення основних ролей інфраструктури..... 61
3.2 Створення структури домену та базові групові політики. 67
3.3 Розгортання та базова конфігурація розподіленої архітектури 72

3.4 Налаштування автоматизованого розгортання ОС із використанням Task Sequence, драйверів та прикладного ПЗ 77
3.4.1 Створення та налаштування Task Sequence 77
3.4.2 Інтеграція драйверів у середовище розгортання 82
3.4.3 Додавання та конфігурація прикладного програмного забезпечення 84
3.4.4 Автоматизація процесу розгортання операційної системи на основі скриптів 86
3.5 Тестування системи, моніторинг її роботи та аналіз метрик ефективності .. 88
3.5.1 Тестування розгортання системи 88
3.5.2 Моніторинг роботи системи 93
3.5.3 Аналіз метрик ефективності 97
Висновки до третього розділу 99
ВИСНОВКИ 101
Список використаних джерел 102
ДОДАТКИ 104

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ОС – операційна система

ПЗ – програмне забезпечення

MDT – Microsoft Deployment Toolkit

WDS – Windows Deployment Services

ADK – Windows Assessment and Deployment Kit

PXE – Preboot Execution Environment

DHCP – Dynamic Host Configuration Protocol

OU – Organizational Units

DNS – Domain Name System

MECM – Microsoft Endpoint Configuration Manager

WIM – Windows Imaging Format

VM – віртуальна машина

BIOS – базова система введення/виведення

UEFI – уніфікований розширений інтерфейс прошивки

PE – Preinstallation Environment

ВСТУП

У сучасних корпоративних мережах процес встановлення та налаштування операційних систем потребує значних витрат часу та ресурсів, особливо при обслуговуванні великої кількості робочих станцій. Ручне розгортання ОС ускладнює стандартизацію програмного середовища, підвищує ймовірність помилок і збільшує навантаження на системних адміністраторів. Використання автоматизованих засобів розгортання, таких як Microsoft Deployment Toolkit та Windows Deployment Services, дозволяє централізувати процес встановлення операційних систем, скоротити час підготовки робочих місць, підвищити масштабованість інфраструктури та забезпечити уніфікованість конфігурацій у корпоративному середовищі.

Метою кваліфікаційної роботи є проєктування та реалізація масштабованої системи автоматизованого розгортання операційних систем на базі технологій Windows Deployment Services та Microsoft Deployment Toolkit для підвищення ефективності адміністрування корпоративної інфраструктури.

Визначена мета обумовлює наступні завдання:

- проаналізувати сучасні методи та технології розгортання операційних систем;
- дослідити принципи роботи та архітектуру WDS і MDT;
- спроєктувати серверну інфраструктуру системи автоматизованого розгортання ОС;
- реалізувати та налаштувати систему автоматизованого мережевого розгортання операційних систем;
- провести тестування системи та оцінити ефективність її роботи.

Об’єктом дослідження є процес автоматизованого розгортання операційних систем у корпоративних комп’ютерних мережах.

Предметом дослідження є методи, засоби та технології автоматизованого розгортання операційних систем із використанням Windows Deployment Services, Microsoft Deployment Toolkit, PXE-завантаження та серверної інфраструктури Windows Server.

У процесі виконання кваліфікаційної роботи було розроблено проект масштабованої системи автоматизованого розгортання операційних систем на базі WDS та MDT у віртуальному середовищі. Реалізована система забезпечує централізоване мережеве встановлення ОС, автоматизацію налаштування клієнтських систем та підтримку масштабування інфраструктури.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ АВТОМАТИЗОВАНОГО РОЗГОРТАННЯ ОС

1.1 Огляд сучасних методів та стратегій розгортання операційних систем

У сучасних інформаційних системах розгортання операційних систем (ОС) є основним етапом до підготовки робочих станцій та серверів. Від вибору методу встановлення залежить швидкість введення обладнання в експлуатацію, стандартизація налаштувань, а також загальна надійність та безпека корпоративної інфраструктури. У сучасному світі існує безліч методів встановлення та розгортання операційних систем, але кожен метод має свої недоліки та переваги.

1. Ручне (традиційне) встановлення операційної системи

Ручне встановлення передбачає інсталяцію ОС із фізичних носіїв – USB-накопичувача чи оптичного диска. Усі кроки виконуються адміністратором: вибір розділів диска, введення параметрів, установка драйверів, базове налаштування системи та програмного забезпечення.

Основні переваги ручного встановлення – простота та доступність. Цей метод встановлення не потребує додаткових серверів та складної мережевої інфраструктури.

Основними недоліками цього методу можна вважати значні витрати часу, висока ймовірність помилки, низька масштабованість, неможливість стандартизувати конфігурацію на великій кількості систем.

Ручне встановлення може підійти для користувачів, які використовують операційну систему для дому або може використовуватись у невеликих офісах та компаніях.



Рисунок – 1.1 Фізичний носій флешка для встановлення ОС

2. Використання відповідних файлів (напівавтоматичне розгортання)

Цей метод, у порівнянні з ручним встановленням, є більш гнучким та зручним, він передбачає створення файлу відповідей, у якому зберігаються основні параметри встановлення (мовні параметри, ім’я комп’ютера, часовий пояс, параметри дисків). У процесі інсталяції система автоматично зчитує ці налаштування, зменшуючи кількість дій адміністратора.

Основні переваги цього методу – мінімізація ручного введення даних, які використовуються під час встановлення та зменшення помилок під час встановлення, які частіше виникають при ручному встановленні.

Основними недоліками цього методу можна вважати, що потрібно попереднє створення конфігураційних файлів, не підходить для масового одночасного розгортання, потребує локальних носіїв.

Напівавтоматичне розгортання операційної системи можна вважати більш зручним рішенням у порівнянні з ручним встановленням, але цей метод також має свої недоліки[21].

3. Клонування дисків та образів системи

Клонування передбачає створення повного образу операційної системи з усіма драйверами, програмами та налаштуваннями. Такий образ переноситься на інші комп’ютери за допомогою програм Acronis, Clonezilla, Norton Ghost тощо.

Основні переваги цього методу – дуже швидке відновлення або встановлення, дозволяє повністю дублювати робоче середовище.

Основними недоліками цього методу можна вважати залежність від апаратного забезпечення, потреба в перезаписі всього диска, складність оновлень та зміни конфігурації, труднощі при масштабуванні.

Метод широко застосовується для швидкого відновлення систем, але в корпоративному розгортанні його витіснили більш гнучкі технології [1].



Рисунок – 1.2 Приклад програмного забезпечення для клонування дисків

4. Мережеве розгортання (Network Deployment) через PXE

Мережевий метод на сьогодні є сучасним та ефективним рішенням. Він використовує сервер мережевого завантаження та технологію PXE (Preboot eXecution Environment), що дозволяє комп'ютеру завантажувати інсталятор ОС безпосередньо з мережі.

Основні інструменти для мережевого розгортання:

- Windows Deployment Services (WDS);
- Microsoft Deployment Toolkit (MDT);
- System Center Configuration Manager (SCCM);
- Microsoft Intune;
- Windows Autopilot;
- FOG Project;
- Clonezilla Server Edition;
- Acronis Snap Deploy;
- Symantec Ghost Solution Suite;
- SmartDeploy.

Основні переваги цього методу – можливість одночасного розгортання десятків і сотень систем, централізоване зберігання образів, повна автоматизація процесу, стандартизація конфігурації робочих станцій, не потребує фізичних носіїв.

Основним недоліком цього підходу є високі витрати на серверну інфраструктуру та складність налаштувань, які виникають у системних адміністраторів.

Мережеве розгортання є стандартом у сучасних IT-підрозділах та широко використовується в середовищах, де важлива швидкість, безпека й автоматизація [2].

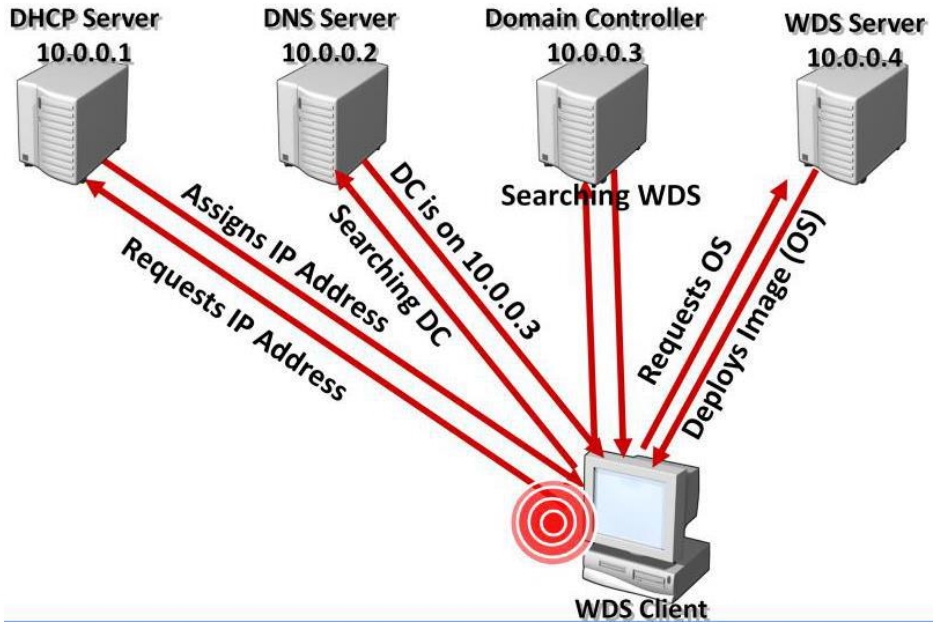


Рисунок – 1.3 Принцип роботи мережевого розгортання на прикладі WDS

5. Хмарне розгортання (Cloud Deployment)

Хмарне розгортання – це сучасний метод встановлення ОС, що використовує сервіси Microsoft Intune, Windows Autopilot та Azure AD. Він дозволяє виконувати підготовку пристрою “з хмари”, без використання локальних серверів. Новий або скинутий ПК автоматично отримує політики безпеки, налаштування та необхідні програми одразу після підключення до Інтернету, що значно зменшує навантаження на IT-відділ.

Основною перевагою методу є централізоване управління та зручність для мобільних і віддалених співробітників. Адміністратор може застосовувати політики, встановлювати оновлення, контролювати статус пристрою або виконувати віддалене скидання.

Разом із тим, хмарне розгортання залежить від стабільного інтернет-з’єднання і доступності хмарних сервісів. Також організація має забезпечити безпеку облікових записів та відповідність інфраструктури вимогам Microsoft.

Попри обмеження, це один із найперспективніших методів, що поступово доповнює і частково замінює класичні підходи до корпоративного розгортання ОС [3].

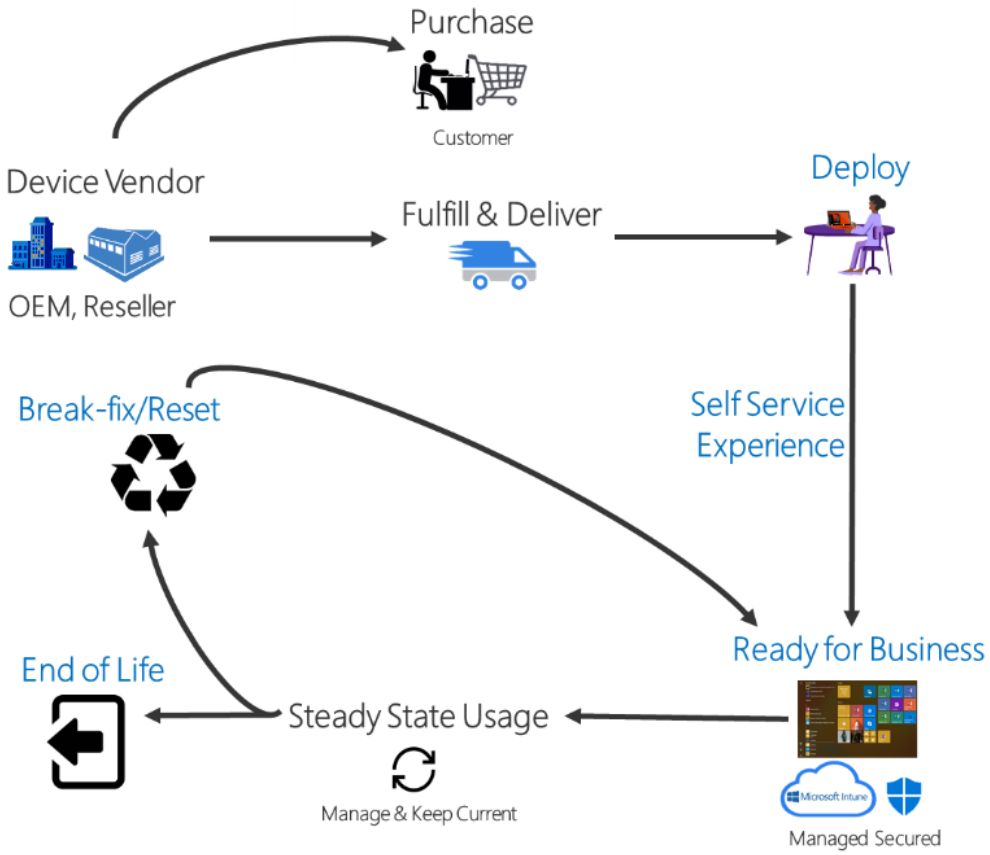


Рисунок – 1.4 Схема використання Windows Autopilot для розгортання

1.2 Архітектура та теоретичні принципи роботи Windows Deployment Services

Служби розгортання Windows (Windows Deployment Services, WDS) – це компонент операційної системи Windows Server, призначений для централізованого мережевого розгортання операційних систем на клієнтських комп'ютерах без використання фізичних носіїв інформації. Дана технологія є наступником служби віддаленої інсталяції Remote Installation Services (RIS) та значно розширює її функціональні можливості.

WDS офіційно підтримує розгортання операційних систем, починаючи з Windows Vista та Windows Server 2008, а також усіх наступних версій. Завдяки використанню образів дисків у форматі Windows Imaging Format (WIM), система забезпечує гнучкість у процесі встановлення та дозволяє розгортати різні операційні системи, включаючи не лише продукти Microsoft.

На відміну від технології RIS, яка базувалася на автоматизації процесу інсталяції з використанням інсталяційних файлів, WDS використовує образи операційних систем, що значно підвищує швидкість розгортання та спрощує процес адміністрування. Крім того, застосування образів дозволяє забезпечити уніфікованість налаштувань та програмного середовища на всіх клієнтських пристроях.

Служба WDS надає можливість встановлення операційних систем через мережу з використанням технології PXE (Preboot Execution Environment), що дозволяє відмовитися від застосування оптичних дисків або інших фізичних носіїв інформації для кожного окремого комп'ютера. Такий підхід є особливо ефективним у середовищах із великою кількістю робочих станцій, де критично важливими є швидкість розгортання, можливість масштабування та централізоване управління процесом встановлення операційних систем.

Для розгортання архітектури WDS необхідно встановити та налаштувати такі компоненти:

- Сервер WDS і Active Directory Domain Services (AD DS) – сервер, на якому розгортається WDS, повинен бути приєднаний до домену AD DS або виконувати функції контролера домену. Це забезпечує коректну інтеграцію служби з доменною інфраструктурою та можливість централізованого керування клієнтами.
- DHCP-сервер – у мережі повинен бути налаштований і функціонувати сервер Dynamic Host Configuration Protocol, який відповідає за автоматичне призначення IP-адрес. Це є критично важливим, оскільки технологія PXE, що використовується WDS, базується на отриманні мережевих параметрів через DHCP.
- DNS-сервер – для забезпечення коректного функціонування доменної інфраструктури та вирішення імен у мережі необхідна наявність працюючого сервера системи доменних імен (Domain Name System).
- Файлова система NTFS – на сервері WDS має бути доступний розділ із файловою системою NTFS, який використовується для зберігання інсталяційних та завантажувальних образів операційних систем.

- Облікові права адміністратора – для встановлення та налаштування ролі WDS користувач повинен мати права локального адміністратора або адміністратора домену, що забезпечує можливість внесення необхідних змін у систему.

Windows Deployment Services (WDS) для мережевого розгортання операційних систем використовує кілька ключових протоколів, які забезпечують завантаження клієнтів, передачу образів та управління процесом інсталяції. Використовує такі протоколи PXE, DHCP, TFTP, SMB / HTTP / HTTPS, BINL [4].

PXE (Preboot Execution Environment) – це базовий протокол мережевого завантаження, який дозволяє комп'ютеру отримати образ завантажувальної операційної системи ще до старту локальної ОС. Завдяки цьому забезпечується можливість централізованого розгортання операційних систем у мережевому середовищі без використання фізичних носіїв інформації.

Концепція PXE сформувалася на основі розвитку мережевих протоколів BOOTP, DHCP та TFTP, які забезпечували базові механізми завантаження систем через мережу. Станом на сучасний етап розвитку інформаційних технологій PXE інтегровано в стандарт UEFI (Unified Extensible Firmware Interface), що дозволяє використовувати його на рівні сучасного мікропрограмного забезпечення. У центрах обробки даних PXE є одним із найпоширеніших рішень для мережевого завантаження, інсталяції та масового розгортання операційних систем.

З моменту появи комп'ютерних мереж виникла необхідність у створенні клієнтських систем, здатних отримувати операційні образи та конфігураційні параметри з віддалених серверів. Для реалізації цієї задачі використовуються служби попереднього завантаження, що базуються на стандартних мережевих протоколах. При цьому програма мережевого завантаження (Network Bootstrap Program, NBP) виконується на рівні мікропрограмного забезпечення пристрою, що забезпечує апаратно-незалежний доступ до мережевих ресурсів.

Першими стандартами, що заклали основу для мережевого завантаження, були TFTP (RFC 906, 1984), який визначив протокол передачі файлів для завантаження, та BOOTP (RFC 951, 1985), що дозволяв бездисковим клієнтам отримувати

IP-адресу, адресу TFTP-сервера та ім’я завантажувального модуля NBP. Згодом для спрощення та автоматизації мережевої конфігурації було розроблено протокол DHCP (RFC 2131, 1997), який частково замінив BOOTP та розширив його функціональні можливості [8].

Середовище PXE, розроблене компанією Intel у рамках ініціативи Wired for Management, використовує поєднання протоколів DHCP і TFTP для завантаження клієнтських систем. Реалізація PXE здійснюється через BIOS або UEFI та включає необхідні компоненти для ініціалізації мережевого з’єднання, зокрема стек UDP/IP та інтерфейс UNDI (Universal Network Device Interface). Це забезпечує можливість взаємодії програми мережевого завантаження (NBP) із серверною інфраструктурою.

З метою підвищення продуктивності передачі даних у PXE було впроваджено вдосконалення протоколу TFTP, зокрема збільшення розміру блоків передачі даних (RFC 2348, 1998), а також подальші оптимізації, визначені у RFC 7440 (2015). Ці покращення дозволили значно підвищити швидкість завантаження образів операційних систем у мережевих середовищах. [9].

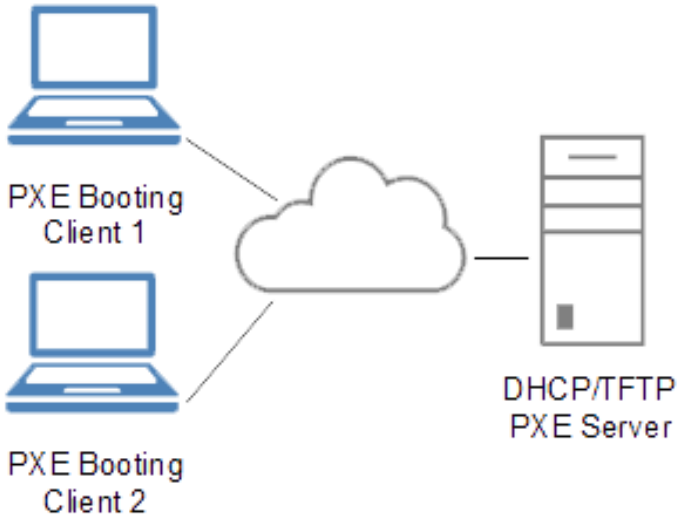


Рисунок – 1.5 Принцип роботи протоколу PXE

Також в роботі WDS використовується протокол DHCP

DHCP (Dynamic Host Configuration Protocol) – це мережевий протокол, який автоматично призначає IP-адреси та інші параметри мережі (маску підмережі, шлюз за замовчуванням, DNS-сервери) пристроям у локальній або корпоративній

мережі. Завдяки DHCP адміністратору не потрібно вручну налаштовувати мережеві параметри на кожному комп'ютері, що значно спрощує управління мережею та підключення нових пристроїв.

Процес призначення IP-адреси відбувається за чотирма етапами, відомими як DORA: Discover – клієнт надсилає широкомовний запит для пошуку DHCP-сервера; Offer – сервер пропонує IP-адресу та мережеві параметри; Request – клієнт підтверджує отримання пропозиції; Acknowledge – сервер резервує адресу та підтверджує її клієнту.

DHCP-сервер містить пул доступних IP-адрес та додаткових налаштувань і видає їх клієнтам на певний час, який називається лізингом (lease). Після закінчення цього періоду клієнт повинен оновити або отримати нову адресу. Сервер також може підтримувати резервування IP для конкретних пристроїв і взаємодіяти з DNS для динамічного оновлення записів.

Хоча DHCP значно спрощує адміністрування мережі, протокол має деякі вразливості. Наприклад, можливі атаки типу "DHCP starvation", коли пул адрес виснажується, або використання фальшивих DHCP-серверів, що надають некоректні параметри клієнтам [7].

Завдяки своїй функціональності DHCP є важливою складовою сучасних мереж та невід'ємною частиною процесу автоматизованого розгортання операційних систем, зокрема у середовищах WDS та PXE.

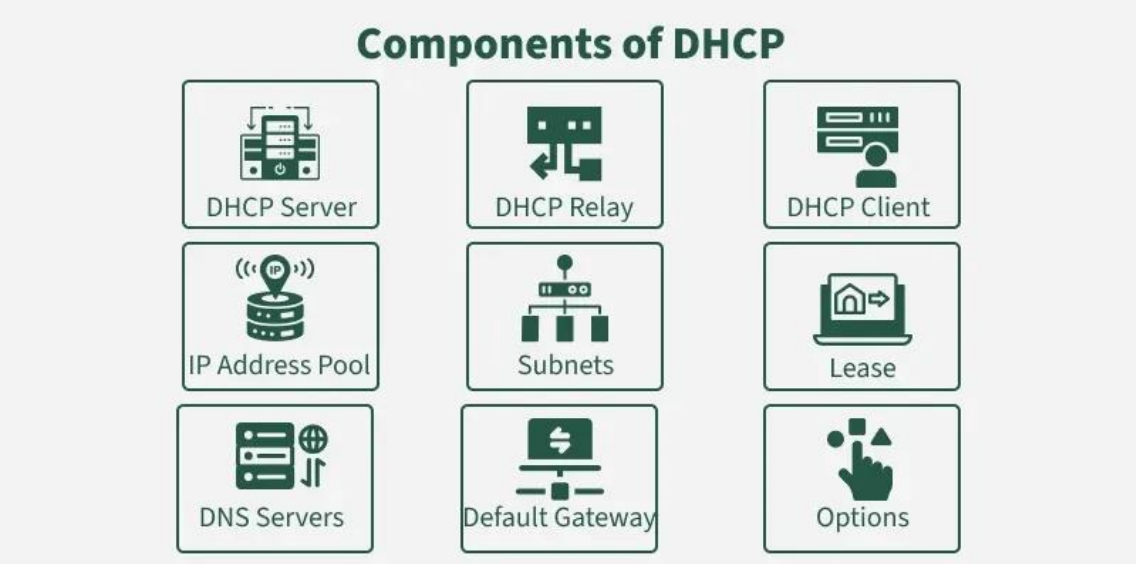


Рисунок – 1.6 Основні компоненти протоколу DHCP

TFTP (Trivial File Transfer Protocol) – це простий протокол передачі файлів, який використовується для базового обміну даними між клієнтом і сервером у мережі. Він працює поверх транспортного протоколу UDP та використовує порт 69 для ініціації з'єднання. Основною особливістю TFTP є його мінімалістична реалізація, що не передбачає складних механізмів керування з'єднанням або автентифікації користувачів. Відповідно до специфікації RFC 1350, протокол орієнтований на прості сценарії передачі файлів у локальних мережах, де важлива швидкість і простота реалізації.

У контексті технології Windows Deployment Services (WDS) протокол TFTP відіграє важливу роль на етапі початкового завантаження клієнтських систем. Він використовується для передачі завантажувальних компонентів і конфігураційних файлів від сервера до клієнта, який ініціює процес завантаження через PXE (Preboot Execution Environment). Саме за допомогою TFTP клієнт отримує необхідний завантажувальний образ (наприклад, boot.wim), після чого запускається середовище встановлення операційної системи.

Завдяки своїй простоті TFTP забезпечує швидкий обмін невеликими файлами на ранніх етапах завантаження системи, коли ще не ініціалізовано повноцінний стек операційної системи. Це робить його ефективним інструментом для використання в корпоративних мережах, де застосовуються технології централізованого розгортання ОС.

Водночас протокол має певні обмеження, зокрема відсутність шифрування, автентифікації та контрольованого управління передачею даних, що обмежує його використання лише локальними та довіреними мережами. Попри це, у поєднанні з DHCP і PXE, TFTP є ключовим елементом процесу мережевого завантаження та широко застосовується у середовищах WDS для автоматизованого розгортання операційних систем [RFC 1350] [8].

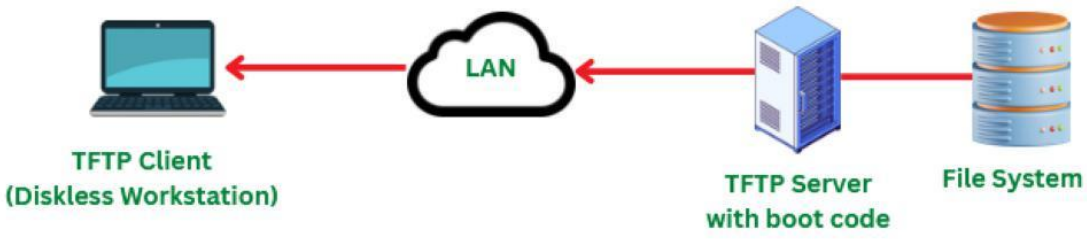


Рисунок – 1.7 Схема роботи протоколу TFTP

BINL (Boot Information Negotiation Layer) – це протокол і сервіс у складі Windows Deployment Services (WDS), який відповідає за обробку PXE-запитів від клієнтів, що завантажуються через мережу. Його основне завдання – передати клієнтам інформацію про доступні завантажувальні образи та параметри інсталяції, забезпечуючи правильну координацію процесу мережевого розгортання операційних систем.

BINL тісно інтегрований з DHCP і TFTP: через DHCP клієнт отримує мережеві параметри і адресу сервера, а через TFTP – початковий завантажувальний образ (boot.wim). Протокол також взаємодіє з Active Directory для надання клієнтам індивідуальних політик доступу та списку доступних образів [9].

Для функціонування Windows Deployment Services (WDS) використовується спеціальне попереднє середовище встановлення – Windows Preinstallation Environment (Windows PE, WinPE). Це мінімалістична операційна система, призначена для підготовки, інсталяції, розгортання та відновлення операційних систем сімейства Windows Server і Windows.

Windows PE забезпечує виконання широкого спектра завдань, пов’язаних із розгортанням ОС, зокрема:

- підготовку жорсткого диска перед інсталяцією операційної системи;
- встановлення Windows із мережевих або локальних джерел за допомогою скриптів і програм;
- створення та розгортання образів операційних систем;
- роботу з офлайн-версією Windows для внесення змін до системи;

- налаштування механізмів автоматичного відновлення;
- відновлення даних із пристроїв, що не завантажуються;
- можливість додавання власної оболонки або графічного інтерфейсу для автоматизації процесів.

Середовище Windows PE підтримує роботу з образами операційних систем через інструменти DISM, а також забезпечує мережеву взаємодію за протоколами TCP/IP. Крім того, воно підтримує роботу з файловими системами, такими як NTFS, та інструментами керування дисками, зокрема DiskPart і BCDBoot. Додатково Windows PE може взаємодіяти з технологіями безпеки, включаючи BitLocker і TPM, а також інтегруватися з віртуалізаційною платформою Hyper-V для роботи з VHD-образами та мережевими ресурсами у віртуальному середовищі.

Вимоги до апаратного забезпечення Windows PE загалом подібні до вимог стандартної операційної системи Windows, однак мають певні особливості. Зокрема, використання жорсткого диска не є обов'язковим, оскільки середовище може повністю завантажуватися в оперативну пам'ять. Мінімальний обсяг ОЗП для базової конфігурації становить приблизно 512 МБ, однак при додаванні драйверів або додаткових компонентів вимоги до пам'яті можуть зростати [17].

Для завантаження Windows PE в оперативну пам'ять необхідна безперервна область ОЗП, достатня для розміщення WIM-образу. У зв'язку з цим виробникам апаратного забезпечення рекомендується резервувати відповідні ділянки пам'яті на рівні BIOS або UEFI для забезпечення стабільної роботи середовища.

32-розрядна версія Windows PE підтримує запуск на 32-розрядних системах з BIOS або UEFI, а також на 64-розрядних системах із BIOS. У свою чергу, 64-розрядна версія призначена для роботи на сучасних платформах із 64-розрядними BIOS та UEFI [10].

Варто зазначити, що Windows PE не є повноцінною операційною системою загального призначення. Вона не призначена для постійного використання, тестування апаратного забезпечення або функціонування як тонкий клієнт. Основне її призначення полягає виключно у виконанні завдань розгортання, інсталяції та відновлення операційних систем.

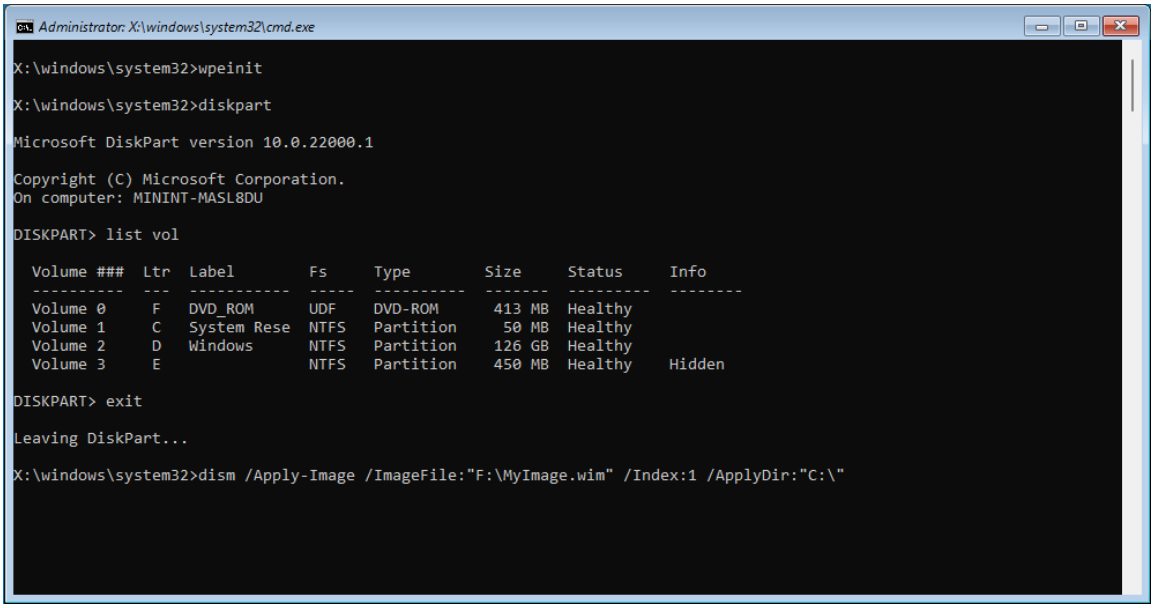


Рисунок – 1.8 Приклад вікна командного рядка Windows PE

1.3 Архітектура та теоретичні принципи роботи Microsoft Deployment Toolkit

Для автоматизації процесу розгортання операційних систем у корпоративних мережах часто використовується спільна архітектура Windows Deployment Services (WDS) та Microsoft Deployment Toolkit (MDT). Такий підхід дозволяє реалізувати централізоване та повністю автоматизоване встановлення операційних систем на клієнтських пристроях.

У цій архітектурі WDS забезпечує мережеве завантаження клієнтських комп’ютерів із використанням технології PXE (Preboot Execution Environment), що дозволяє відмовитися від використання фізичних носіїв інформації, таких як DVD-диски або USB-накопичувачі. Завдяки цьому клієнтський пристрій отримує завантажувальний образ безпосередньо з мережевого сервера.

У свою чергу, MDT відповідає за логіку та автоматизацію процесу інсталяції операційної системи. Він формує завантажувальне середовище Windows PE (WinPE), до якого інтегруються необхідні компоненти, зокрема драйвери пристроїв, сценарії встановлення, оновлення та прикладне програмне забезпечення. Це дозволяє забезпечити уніфікований процес розгортання операційної системи з мінімальним втручанням адміністратора.

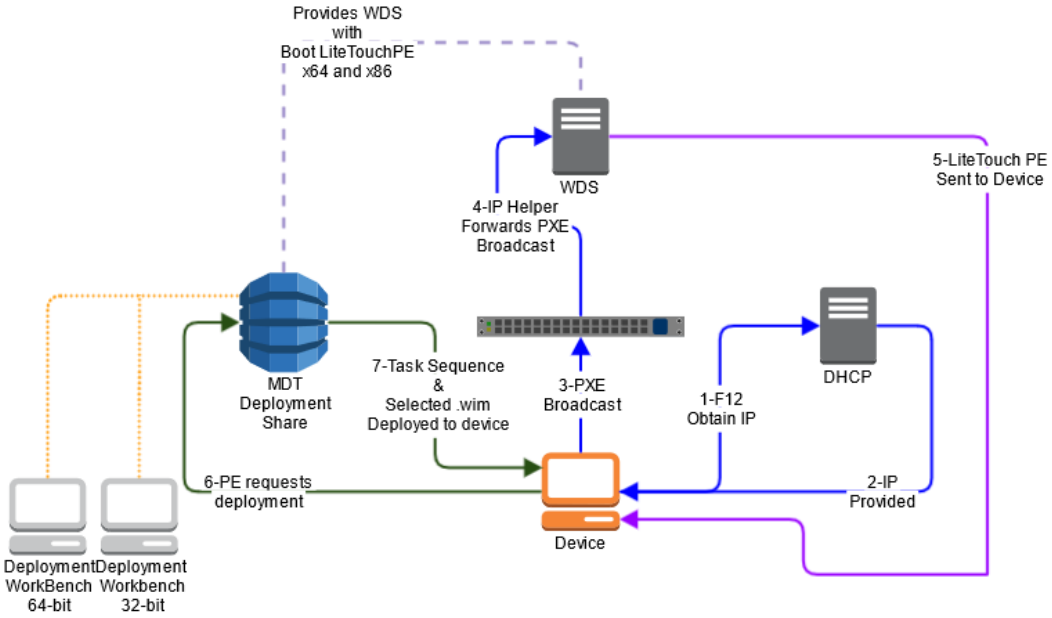


Рисунок – 1.9 Інтеграція WDS з MDT

Microsoft Deployment Toolkit (MDT) – це інструмент, що належить до категорії «прискорювачів рішень» (Solution Accelerator) і призначений для проєктування, автоматизації та керування процесами розгортання операційних систем і прикладного програмного забезпечення. MDT дозволяє створювати уніфіковані та масштабовані сценарії інсталяції, які можуть застосовуватися в корпоративних мережах різного рівня складності.

У більшості випадків MDT використовується у взаємодії з Windows Assessment and Deployment Kit (ADK) (раніше – Windows Automated Installation Kit, WAIK), що забезпечує необхідні інструменти для створення, налаштування та обробки образів операційних систем. Така інтеграція дозволяє реалізовувати гнучкі та повністю автоматизовані процеси розгортання.

MDT базується на принципі task-based deployment (розгортання на основі завдань), що забезпечує детальний контроль над кожним етапом інсталяції операційної системи. Це дозволяє адміністратору визначати послідовність дій, які виконуються під час розгортання, включаючи підготовку диска, встановлення операційної системи, інтеграцію драйверів, оновлень та прикладного програмного забезпечення.

MDT дозволяє використовувати «тонкий» базовий образ операційної системи, який не містить додаткового програмного забезпечення та налаштувань. Такий підхід передбачає додавання необхідних компонентів безпосередньо під час процесу розгортання відповідно до заздалегідь визначених критеріїв. Це значно підвищує гнучкість процесу інсталяції та зменшує потребу у створенні великої кількості спеціалізованих образів.

Завдяки можливості динамічного додавання драйверів, програмного забезпечення та конфігураційних параметрів під час розгортання, один і той самий базовий образ може використовуватися для різних підрозділів або середовищ. При цьому за рахунок використання різних task sequence забезпечується формування різних конфігурацій програмного забезпечення та системних налаштувань без необхідності створення окремих інсталяційних образів.

У порівнянні з MDT, технологія Windows Deployment Services (WDS) орієнтована переважно на роботу з так званими «товстими» образами, які містять попередньо налаштовану операційну систему разом із драйверами та програмним забезпеченням. Такі образи можуть бути захоплені з еталонного пристрою та розгорнуті на інших комп’ютерах через мережу з використанням технології PXE.

MDT, у свою чергу, є інструментом проєктування процесу розгортання, який дозволяє створювати гнучкі покрокові сценарії інсталяції з високим рівнем автоматизації. Він забезпечує детальний контроль над усіма етапами встановлення операційної системи та програмного забезпечення.

MDT підтримує кілька основних моделей розгортання, зокрема:

Zero Touch Installation (ZTI) – повністю автоматизоване розгортання, яке зазвичай використовується у зв’язці з System Center Configuration Manager (SCCM);

Lite Touch Installation (LTI) – напівавтоматичний режим, що потребує мінімальної участі користувача;

User Driven Installation (UDI) – модель, у якій користувач самостійно вибирає параметри встановлення через графічний інтерфейс.

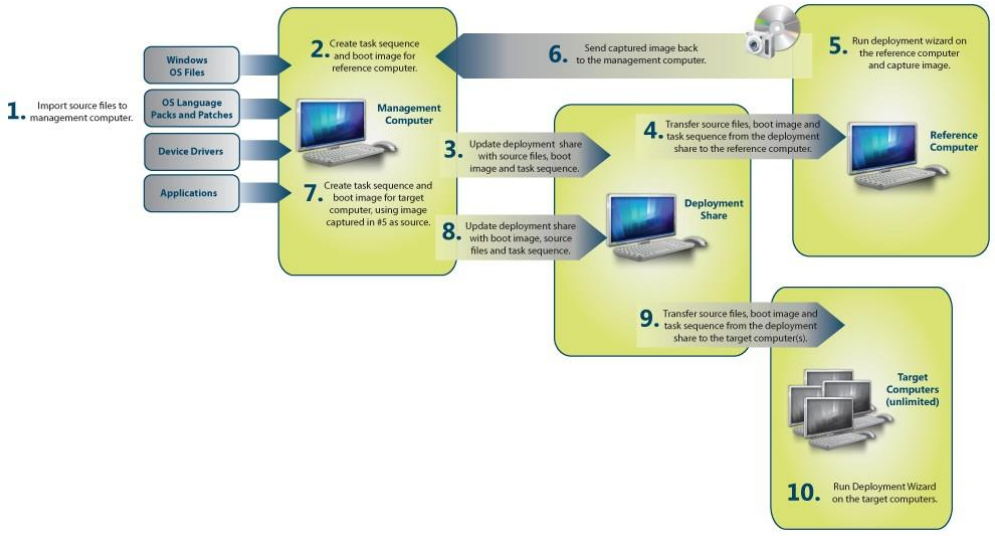


Рисунок – 1.10 Високорівневі процеси розгортання LTI, ZTI та UDI

Високорівневий процес розгортання LTI, ZTI та UDI

Процес розгортання операційних систем у середовищі Microsoft Deployment Toolkit (MDT) включає кілька послідовних етапів, які можуть виконуватися відповідно до обраної моделі інсталяції: Lite Touch Installation (LTI), Zero Touch Installation (ZTI) та User Driven Installation (UDI).

На першому етапі здійснюється підготовка необхідних компонентів MDT. До них належать вихідні інсталяційні файли або готові образи операційної системи Windows, мовні пакети, драйвери пристроїв для еталонних і цільових комп'ютерів, а також додаткові інсталяційні пакети програмного забезпечення.

Далі виконується формування початкової конфігурації середовища розгортання. На цьому етапі створюються базові образи операційної системи, визначаються параметри встановлення та налаштовуються послідовності завдань (Task Sequences), які будуть використовуватися для автоматизації процесу інсталяції.

Наступним кроком є розгортання підготовленого образу на еталонній (reference) машині. Після інсталяції здійснюється її конфігурація відповідно до вимог організації, після чого виконується захоплення (capture) налаштованого образу операційної системи.

Після цього формується оновлена конфігурація розгортання, яка включає нові параметри та послідовності завдань для встановлення захопленого еталонного образу на цільових пристроях.

Завершальним етапом є безпосереднє розгортання підготовленого образу на клієнтських комп’ютерах. Залежно від обраної моделі (LTI, ZTI або UDI) процес може бути повністю автоматизованим або вимагати часткової участі користувача [11].

Таблиця 1.1

Переваги та недоліки товстих, тонких та гібридних образів

Тип образу	Перевага	Недолік
Товстий	може бути простіше розгорнути, оскільки всі програми та мовні пакети є в образі	потрібно більше місця для кожного образу
	зменшено початкову складність, оскільки розширені скрипти зазвичай не потрібні.	потрібно більше часу для завантаження через підключення до мережі, ніж тонкі або гібридні образи.
	програми та мовні пакети доступні після завершення розгортання.	потрібно збільшити обсяг обслуговування образів, оскільки для будь-яких оновлень операційних систем, драйверів пристроїв, програм або мовних пакетів потрібно створити новий образ.
Тонкий	потрібно менше місця для кожного образу.	Створення може бути складнішим, оскільки під час створення образу потрібні додаткові дії.
	потрібно менше часу, ніж товсті зображення, щоб завантажити мережні з'єднання.	може бути складнішою, оскільки може знадобитися розширений скрипт.
	скорочено витрати на обслуговування образу, оскільки образ містить менше компонентів	Програми та мови стають недоступними після завершення розгортання зображення
Гібридний	потрібно менше пам'яті, ніж товсті зображення кожного зображення	може бути складнішим, ніж товстий образ (але не більш тонкого), оскільки під час створення образу потрібні додаткові дії.
	потрібно менше часу, ніж товстий товстий образ для завантаження по мережевим підключенням.	можливо, ускладнюється, оскільки може бути потрібний розширений скрипт (хоча й не такий складний, як у тонких зображеннях

Продовження таблиці 1.1

	скоротилися витрати на обслуговування образу, оскільки образ містить менше компонентів.	сторінки та мови стають недоступними одразу після завершення розгортання зображення

1.4 Аналіз архітектури системи розгортання ОС та обґрунтування вимог на базі WDS та MDT

У сучасних умовах розвитку інформаційних технологій автоматизація процесів розгортання операційних систем є важливим складником ефективного функціонування IT-інфраструктури підприємства. Зі зростанням кількості робочих станцій і серверів виникає потреба у впровадженні рішень, які дозволяють централізовано керувати процесом встановлення операційних систем, мінімізувати участь адміністратора та забезпечити стандартизацію програмного середовища. Одним із таких рішень є використання інтегрованої архітектури Windows Deployment Services (WDS) та Microsoft Deployment Toolkit (MDT).

Особливої актуальності набуває застосування розподіленої архітектури розгортання, яка передбачає розділення функціональних компонентів системи між кількома серверами. Такий підхід дозволяє не лише підвищити продуктивність, але й забезпечити масштабованість системи, оптимізувати використання мережевих ресурсів та закласти основу для подальшого розвитку інфраструктури.

Розподілена архітектура WDS та MDT базується на принципі функціонального розділення ролей між серверами. У цій моделі один сервер виконує функцію мережевого завантаження та відповідає за передачу початкових образів клієнтським пристроям, тоді як інший забезпечує логіку розгортання, включаючи управління сценаріями встановлення, образами операційних систем, драйверами та програмним забезпеченням. Таким чином, WDS виступає як транспортний компонент, який забезпечує доступ до середовища розгортання, а MDT визначає послідовність і правила виконання процесу інсталяції [5].

Процес розгортання операційної системи в такій архітектурі відбувається поетапно. Спочатку клієнтський пристрій ініціює мережеве завантаження за допомогою технології PXE та отримує мережеві параметри від DHCP-сервера. Після

цього встановлюється з’єднання із сервером WDS, який передає завантажувальний образ середовища WinPE. Завантажившись у це середовище, клієнт звертається до Deployment Share, розміщеного на сервері MDT, звідки отримує необхідні ресурси для подальшого встановлення. Далі запускається сценарій розгортання, що визначає повний цикл встановлення операційної системи, включаючи підготовку диска, копіювання образу, встановлення драйверів, оновлень та прикладного програмного забезпечення. Завершальним етапом є перезавантаження системи та її інтеграція в корпоративне середовище.

У межах аналізу архітектури доцільно розглядати різні варіанти її побудови. Найпростішим є централізований підхід, за якого всі компоненти розміщені на одному сервері, однак він має обмеження щодо продуктивності та подальшого масштабування. Більш ефективним є розподілений варіант, у якому ролі WDS і MDT рознесені між різними серверами, що дозволяє зменшити навантаження та підвищити стабільність роботи системи. У великих організаціях може застосовуватися ієрархічна модель, яка передбачає використання центрального сервера розгортання та декількох серверів мережевого завантаження в різних сегментах мережі.

З позиції вимог до сучасних інформаційних систем, розподілена архітектура повинна забезпечувати виконання як функціональних, так і нефункціональних характеристик. У функціональному аспекті система має підтримувати повністю автоматизоване розгортання операційних систем через мережу, централізоване управління образами та сценаріями встановлення, а також інтеграцію з доменною інфраструктурою. Це дозволяє значно скоротити час підготовки нових робочих місць і забезпечити уніфікованість програмного середовища.

Не менш важливими є вимоги до надійності та відмовостійкості. У теоретичному плані така система повинна передбачати можливість резервування ключових компонентів, дублювання серверів та організацію відмовостійкої інфраструктури. Це дозволяє мінімізувати ризики простою та забезпечити безперервність процесу розгортання навіть у разі часткових відмов. Окремо слід відзначити можливість швидкого відновлення працездатності клієнтських систем шляхом повторного розгортання операційної системи з подальшим відновленням даних із резервних копій,

що є важливим елементом сучасних підходів до забезпечення безперервності бізнес-процесів.

Масштабованість – це здатність системи ефективно збільшувати свою продуктивність і пропускну здатність при зростанні навантаження (кількості клієнтів, обсягу даних або кількості одночасних розгортань) шляхом додавання ресурсів або вузлів.

У контексті системи автоматизованого розгортання операційних систем масштабованість означає можливість обслуговування більшої кількості клієнтських пристроїв без суттєвого зниження швидкості розгортання та стабільності роботи.

У сучасних інформаційних системах застосовуються два основні підходи до масштабування: вертикальне та горизонтальне.

Вертикальне масштабування (scale-up) – це спосіб підвищення продуктивності системи шляхом збільшення ресурсів одного сервера. До таких ресурсів належать:

- оперативна пам’ять (RAM);
- обчислювальна потужність процесора (CPU);
- швидкість та обсяг дискової підсистеми (SSD/NVMe).

Перевагами вертикального масштабування є простота реалізації та відсутність необхідності змінювати архітектуру системи. Проте цей підхід має обмеження, оскільки ресурси одного сервера не є безмежними, а також не забезпечує відмовостійкість – у разі збою сервера система стає недоступною.

Горизонтальне масштабування (scale-out) – це спосіб збільшення продуктивності шляхом додавання нових серверів до системи та розподілу навантаження між ними.

Основні характеристики:

- додавання нових вузлів (серверів) у систему;
- балансування навантаження між серверами;
- можливість паралельної обробки запитів.

Переваги горизонтального масштабування:

- висока відмовостійкість (при виході з ладу одного вузла система продовжує працювати);
- практично необмежене розширення;
- ефективна робота при великій кількості клієнтів.

Недоліком є складніша реалізація, яка потребує налаштування синхронізації даних, балансування навантаження та мережевої інфраструктури [15].

Таблиця 1.2

Порівняння горизонтального та вертикального масштабування

Характеристика	Горизонтальне масштабування	Вертикальне масштабування
Визначення	Додавання нових серверів (вузлів) до системи для розподілу навантаження	Збільшення обчислювальних ресурсів існуючого сервера
Переваги	Висока відмовостійкість, гнучкість, можливість практично необмеженого розширення	Простота реалізації, відсутність складної розподіленої архітектури
Недоліки	Складніше адміністрування, потреба в балансуванні навантаження та синхронізації даних	Обмеження апаратних ресурсів, наявність єдиної точки відмови
Сфери застосування	Розподілені системи, веб-сервіси, корпоративні мережі з великою кількістю клієнтів	Локальні сервери, бази даних, системи з високими вимогами до продуктивності одного вузла

Продуктивність системи визначається швидкістю розгортання операційних систем і ефективністю використання мережевих та обчислювальних ресурсів. Розподілена архітектура дозволяє оптимізувати передачу даних, зменшити затримки та забезпечити одночасне обслуговування великої кількості клієнтів. Розміщення серверів ближче до користувачів також сприяє зменшенню навантаження на мережу та скороченню часу встановлення.

Важливим аспектом є забезпечення безпеки системи. Розподіл компонентів дозволяє чітко розмежувати доступ до ресурсів, обмежити доступ до сховищ образів та контролювати процес автентифікації користувачів. Це знижує ризик несанкціонованого доступу та підвищує загальний рівень захищеності інфраструктури.

Незважаючи на значні переваги, розподілена архітектура має і певні недоліки. Передусім це складність впровадження та налаштування, оскільки необхідно забезпечити коректну взаємодію між усіма компонентами системи. Крім того, ефективність її роботи значною мірою залежить від стабільності мережевої інфраструктури. У разі виникнення проблем із мережею процес розгортання може сповільнюватися або перериватися. Також збільшується кількість потенційних точок відмови, що потребує додаткових заходів для підвищення відмовостійкості. Адміністрування такої системи вимагає вищого рівня підготовки персоналу та додаткових витрат часу на її підтримку.

Таким чином, розподілена архітектура WDS та MDT є ефективним рішенням для автоматизації розгортання операційних систем у корпоративних середовищах. Вона поєднує високу продуктивність, масштабованість і гнучкість, а також дозволяє реалізувати сучасні підходи до забезпечення надійності, безпеки та безперервності роботи інформаційних систем.

1.5 Аналіз переваг і недоліків WDS та MDT та їх порівняння з Microsoft Endpoint Configuration Manager

Windows Deployment Services забезпечує можливість мережевого розгортання операційних систем Windows, що значно спрощує процес інсталяції у порівнянні з ручним встановленням. Це зменшує витрати часу й ресурсів, особливо у великих IT-інфраструктурах, де потрібно швидко підготувати велику кількість робочих станцій.

WDS підтримує роботу у змішаних середовищах, дозволяючи розгортати образи різних версій Windows –від Windows 7 і Windows Server 2008 R2 до сучасніших систем, таких як Windows 8.1 та Windows Server 2012 R2. Така сумісність робить службу універсальним інструментом для організацій з різнорідною інфраструктурою.

Платформа базується на стандартних технологіях Windows-розгортання: Windows Preinstallation Environment (Windows PE), файлах образів формату .wim та інсталяції на основі образів. Це гарантує стабільність, гнучкість і можливість інтеграції з іншими інструментами, такими як MDT або Configuration Manager.

Однією з ключових технічних переваг є підтримка багатоадресної розсилки (multicast), яка дозволяє передавати образи декільком комп'ютерам одночасно. Це зменшує навантаження на мережу та пришвидшує розгортання у великих середовищах.

WDS також надає зручні засоби для роботи з образами. Зокрема, вбудований майстер захоплення образів дозволяє швидко створювати еталонні зображення системи без необхідності використовувати сторонні інструменти, такі як ImageX.

Крім того, служба підтримує централізоване керування драйверами: адміністратор може додавати драйвери на сервер WDS і налаштовувати їх автоматичну інтеграцію у процес інсталяції певних образів. Це забезпечує сумісність обладнання та усуває проблеми з відсутніми драйверами на цільових комп'ютерах.

Основнивним аналогом WDS є інструмент SCCM. Він схожий за функціоналом, але має свої переваги та недоліки.

Microsoft Endpoint Configuration Manager (MECM) – це комплексна система керування IT-інфраструктурою, яка дозволяє централізовано розгортати операційні системи, встановлювати та оновлювати програмне забезпечення, керувати налаштуваннями безпеки, виконувати інвентаризацію та здійснювати моніторинг стану пристроїв [13].

MECM забезпечує масштабне корпоративне управління Windows-комп'ютерами, серверами та мобільними пристроями. На відміну від WDS, який призначений переважно для розгортання ОС, MECM охоплює повний цикл життєвого циклу пристроїв: від інсталяції до оновлення та управління їх конфігурацією.

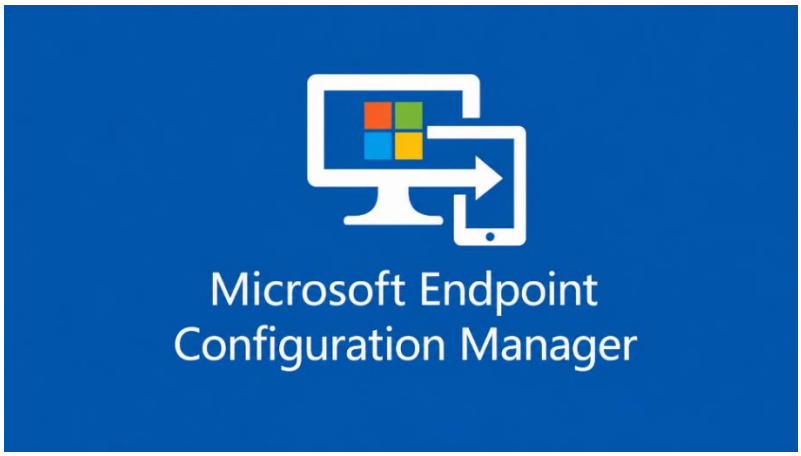


Рисунок – 1.11 Програмний продукт Microsoft Endpoint Configuration Manager

Основні переваги MECM:

Забезпечує повне управління пристроями в організації. Це включає розгортання операційних систем, встановлення та оновлення програмного забезпечення, управління безпекою, моніторинг і інвентаризацію. Завдяки цьому можна централізовано керувати великою кількістю комп'ютерів та серверів, що значно спрощує адміністрування IT-інфраструктури.

Система підтримує мережеве розгортання через PXE-завантаження, Multicast і Task Sequences, що дозволяє автоматизувати встановлення ОС та програм на численних пристроях одночасно. Крім того, MECM інтегрується з хмарними сервісами Microsoft 365, такими як Intune і Entra ID, що дає змогу керувати пристроями як у локальній мережі, так і за її межами.

Ще однією перевагою є централізоване керування оновленнями та програмами, а також потужні засоби звітності й аналізу. Інструменти типу CMPivot дозволяють отримувати дані з клієнтів у реальному часі, а SQL-звіти допомагають відстежувати стан систем та використання ПЗ. Крім того, MECM інтегрується з Microsoft Defender, що забезпечує управління безпекою та контроль брандмауера на всіх пристроях.

Основні недоліки MECM:

Головний недолік MECM – це складність його налаштування та підтримки. Для роботи потрібні SQL Server, розгорнуті ролі сайтів та інші компоненти інфраструктури, що робить систему важкою для швидкого впровадження.

MECM має високі вимоги до ресурсів сервера та мережі, особливо при обслуговуванні великої кількості клієнтів, і потребує відповідних ліцензій, що збільшує загальні витрати на впровадження.

Для малих організацій MECM може бути надмірним рішенням. У невеликих мережах простіше використовувати WDS у поєднанні з MDT, оскільки цього достатньо для базового розгортання ОС і програм без складної інфраструктури.

З іншого боку, Microsoft Endpoint Configuration Manager – це комплексне рішення для управління, яке забезпечує єдину інфраструктуру для керування та розгортання програмного забезпечення, апаратного забезпечення та операційних

систем. MECM OSD (Operating System Deployment – розгортання операційної системи) – це функція MECM, яка дозволяє адміністраторам розгортати операційні системи Windows на комп’ютерах в організації.

WDS та MDT більше підходять для середовищ малого масштабу, MECM OSD – це більш комплексне рішення, яке краще підходить для середовищ великого масштабу. MDT є безкоштовним і чудово підходить для невеликих середовищ без великої кількості комп'ютерів для міграції, тоді як MECM – це набагато складніший інструмент, який дозволяє робити набагато більше, ніж MDT.

Висновки до першого розділу

У першому розділі було проведено комплексний аналіз теоретичних основ розгортання операційних систем та сучасних підходів до автоматизації цього процесу. Розглянуто основні методи встановлення ОС, починаючи від традиційного ручного способу до сучасних автоматизованих і хмарних рішень, що дозволило визначити їх переваги, недоліки та сфери доцільного застосування.

Особливу увагу приділено технологіям мережевого розгортання, зокрема службам Windows Deployment Services (WDS) та Microsoft Deployment Toolkit (MDT). Проаналізовано їх архітектуру, принципи роботи, а також ключові компоненти та протоколи, які забезпечують функціонування системи (PXE, DHCP, TFTP тощо). Визначено роль кожного з інструментів у процесі автоматизованого встановлення операційних систем та їх взаємодію у спільній інфраструктурі.

Також у розділі досліджено розподілену архітектуру розгортання WDS та MDT, що дозволяє підвищити продуктивність, масштабованість і гнучкість системи. Проведено аналіз її переваг і недоліків, а також розглянуто різні варіанти побудови інфраструктури розгортання. Окремо виконано порівняння WDS і MDT із більш комплексним рішенням – System Center Configuration Manager (SCCM), що дало змогу визначити їх місце в сучасних IT-середовищах.

Отже, за результатами розділу встановлено, що використання автоматизованих засобів розгортання операційних систем є ефективним підходом для сучасних

організацій, оскільки забезпечує стандартизацію, скорочення часу налаштування та зменшення кількості помилок. Отримані теоретичні знання створюють основу для подальшої практичної реалізації системи автоматизованого розгортання у наступних розділах роботи.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА ПОБУДОВА СЕРВЕРНОЇ ІНФРАСТРУКТУРИ СИСТЕМИ РОЗГОРТАННЯ ОС

2.1 Вибір середовища віртуалізації для побудови лабораторного стенду

У сучасних IT-інфраструктурах автоматизоване розгортання операційних систем за допомогою таких інструментів, як WDS та MDT, широко застосовується в реальних корпоративних середовищах. Це дозволяє централізовано встановлювати операційні системи, зменшувати витрати часу на налаштування робочих станцій і забезпечувати єдині стандарти конфігурації в організації.

Однак розгортання повноцінної серверної інфраструктури потребує значних апаратних ресурсів, окремого мережевого середовища та відповідного обладнання. Використання віртуальних машин дозволяє змоделювати реальну мережеву інфраструктуру, включаючи сервери та клієнтські пристрої, без необхідності використання фізичного обладнання. Буде доцільно провести порівняльний аналіз різних середовищ віртуалізації, щоб визначити їх основні переваги і недоліки.

VirtualBox – це гіпервізор, призначений для запуску кількох операційних систем у так званому віртуальному середовищі – віртуальних машинах, які працюють поверх уже встановленої на комп’ютері хост-операційної системи. Цей програмний продукт постійно вдосконалюється та отримує нові функціональні можливості, що робить його актуальним інструментом у сфері віртуалізації.

Керування віртуальними машинами в VirtualBox може здійснюватися як через графічний інтерфейс, побудований на базі бібліотеки Qt, так і через консольні утиліти (у режимі headless або за допомогою SDL), що дозволяє автоматизувати процеси запуску та керування віртуальними середовищами.

Для більш тісної інтеграції гостьової операційної системи з хост-системою у складі VirtualBox використовуються спеціальні гостьові доповнення (Guest Additions). Вони забезпечують такі можливості, як спільний доступ до папок, обмін даними через буфер обміну, апаратне відеоприскорення, а також більш зручну роботу з вікнами завдяки режиму безшовної інтеграції.

VirtualBox підтримується основними операційними системами (Linux, FreeBSD, Mac OS X, Windows тощо) та дозволяє запускати гостьові ОС (FreeBSD, Linux, OpenBSD, OS/2 Warp, Windows і Solaris) [14].

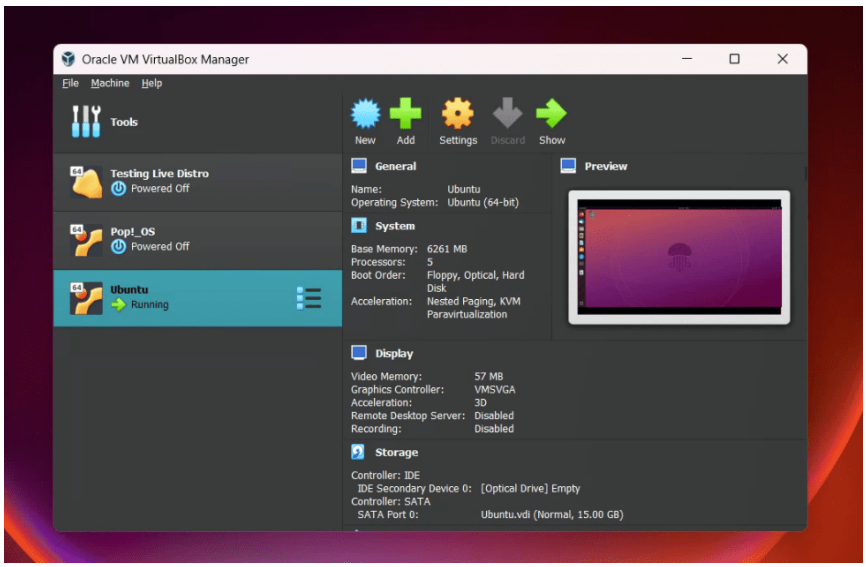


Рисунок – 2.1 Середовище віртуалізації VirtualBox

Використання Oracle VM VirtualBox як середовища віртуалізації має як переваги, так і певні недоліки, які варто враховувати під час побудови тестової або навчальної інфраструктури.

До основних переваг VirtualBox належить його безкоштовність і відкритість, що робить його доступним для студентів та невеликих проєктів. Програма підтримує широкий спектр операційних систем, включаючи різні версії Windows, Linux та інші платформи, що забезпечує гнучкість у використанні. Важливою перевагою є простота встановлення та налаштування – інтерфейс інтуїтивно зрозумілий, що дозволяє швидко створювати та конфігурувати віртуальні машини. Також VirtualBox надає можливість створення знімків (snapshots), що дозволяє зберігати стан системи та повертатися до нього у разі помилок. Додатково слід відзначити підтримку різних мережевих режимів (NAT, Bridged, Host-Only), що є важливим для моделювання мережевих сценаріїв, зокрема при роботі з WDS та MDT.

VirtualBox має і певні недоліки. Насамперед, це нижча продуктивність у порівнянні з такими рішеннями, як VMware або Hyper-V, особливо при роботі з ресурсоємними системами. Обмежена підтримка деяких сучасних функцій

віртуалізації та апаратного прискорення може впливати на стабільність і швидкодію. Крім того, у складних мережевих конфігураціях можуть виникати труднощі з налаштуванням, зокрема при емуляції корпоративної інфраструктури. Також VirtualBox менш придатний для використання у великих виробничих середовищах, оскільки не забезпечує такого рівня масштабованості та централізованого управління, як професійні платформи.

Hyper-V – це платформа віртуалізації від Microsoft, яка використовується для створення та керування віртуальними машинами в середовищах Windows. Вона входить до складу Windows Server, а також доступна в деяких версіях клієнтської операційної системи Windows. Основною особливістю Hyper-V є те, що він працює як гіпервізор першого типу, тобто безпосередньо взаємодіє з апаратним забезпеченням. Завдяки цьому досягається висока продуктивність і ефективне використання ресурсів у порівнянні з програмними рішеннями, які працюють поверх операційної системи.

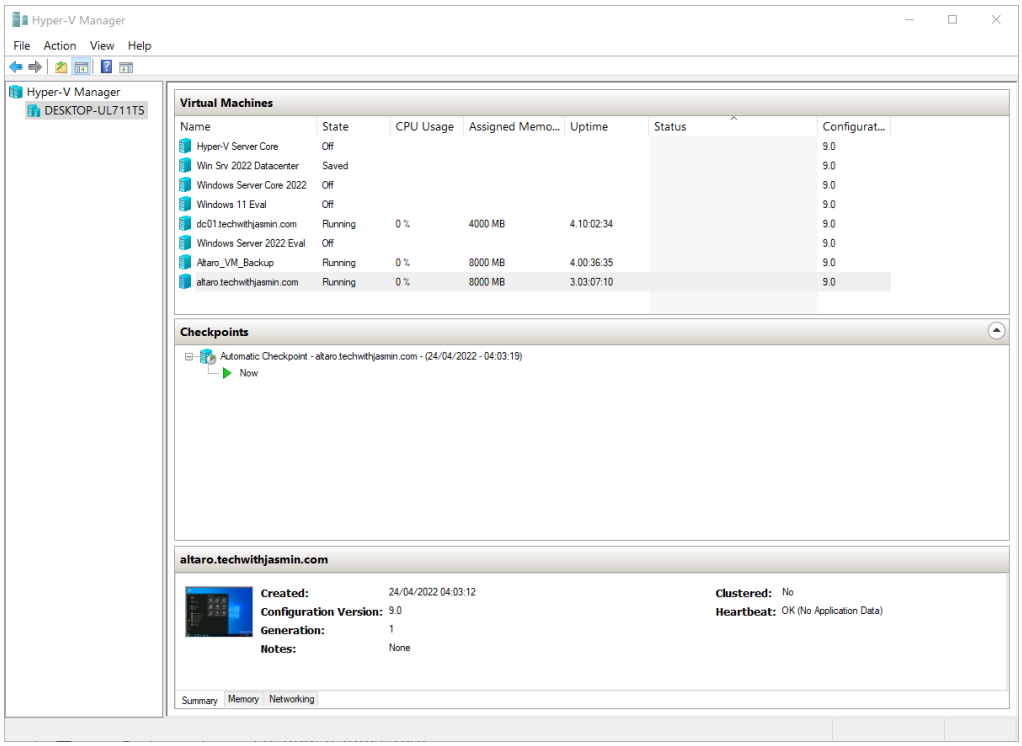


Рисунок – 2.2 Середовище віртуалізації Hyper-V

У середовищі Hyper-V можна створювати кілька віртуальних машин, кожна з яких функціонує як окремий комп’ютер зі своєю операційною системою та ресурсами. Платформа підтримує не лише Windows, а й різні дистрибутиви Linux, що

робить її універсальним інструментом для різних задач. Також Hyper-V надає широкі можливості для керування ресурсами – можна гнучко налаштовувати використання процесора, оперативної пам’яті та мережі, що особливо важливо в корпоративних середовищах.

Суттєвою перевагою Hyper-V є його глибока інтеграція з іншими технологіями Microsoft. Це спрощує адміністрування, особливо якщо вся інфраструктура побудована на базі Windows Server. Крім того, платформа підтримує такі можливості, як жива міграція віртуальних машин без зупинки, реплікація для підвищення відмовостійкості та динамічне виділення пам’яті. Усе це робить Hyper-V придатним не лише для тестування, а й для використання у великих виробничих системах.

Водночас Hyper-V має і певні недоліки. Для його ефективного використання потрібне сумісне апаратне забезпечення з підтримкою апаратної віртуалізації, а також достатня кількість ресурсів. Налаштування деяких функцій може бути складнішим для початківців у порівнянні з більш простими рішеннями, такими як VirtualBox. Крім того, Hyper-V орієнтований переважно на екосистему Microsoft, тому його використання в змішаних або нестандартних середовищах може бути менш зручним.

VMware – це одна з найвідоміших платформ для віртуалізації, яка використовується як у навчальних цілях, так і в реальних IT-інфраструктурах. Вона дозволяє створювати віртуальні машини та запускати на одному фізичному комп’ютері декілька операційних систем одночасно. Найчастіше для локального використання застосовують такі продукти, як VMware Workstation або VMware Player, які встановлюються як звичайні програми на операційну систему.

Основною перевагою VMware є висока продуктивність і стабільність роботи. У порівнянні з багатьма іншими рішеннями, ця платформа краще оптимізована та забезпечує більш ефективне використання апаратних ресурсів. Також VMware добре підтримує різні операційні системи, включаючи Windows і Linux, та забезпечує їх коректну роботу у віртуальному середовищі. Інтерфейс програми є досить зручним, що дозволяє швидко створювати та налаштовувати віртуальні машини навіть без глибоких знань.

Крім того, VMware має корисні функції, такі як створення знімків системи (snapshots), можливість клонування віртуальних машин і гнучке налаштування мережі. Це дозволяє легко тестувати різні сценарії, зокрема мережеві конфігурації, що важливо при дослідженні технологій розгортання операційних систем.

Також платформа має і недоліки. Основним з них є те, що повнофункціональні версії VMware є платними, що може бути обмеженням для студентів або невеликих проєктів. Також програма досить вимоглива до ресурсів комп'ютера, особливо при одночасному запуску кількох віртуальних машин. У деяких випадках налаштування складних мережевих сценаріїв може бути менш інтуїтивним у порівнянні з простішими рішеннями.

У межах даної роботи як основне середовище віртуалізації було обрано Oracle VM VirtualBox версії 7.2.4 r170995 (Qt6.8.0 on windows). Це рішення обумовлене насамперед його доступністю та простотою використання, що є важливим фактором при виконанні навчальних і дослідницьких проєктів. VirtualBox є безкоштовним програмним продуктом із відкритим вихідним кодом, що дозволяє використовувати його без додаткових фінансових витрат, на відміну від деяких комерційних аналогів, таких як VMware Workstation.

Крім того, VirtualBox забезпечує достатній функціонал для моделювання мережевої інфраструктури, необхідної для дослідження технологій автоматизованого розгортання операційних систем за допомогою WDS та MDT.

Важливим фактором є також невисокі вимоги до апаратних ресурсів у порівнянні з більш потужними платформами, такими як Microsoft Hyper-V, що дозволяє запускати кілька віртуальних машин навіть на персональному комп'ютері середнього рівня. Простота встановлення, зрозумілий інтерфейс і наявність функцій створення знімків (snapshots) значно спрощують процес тестування та налагодження системи.

Таким чином, використання VirtualBox у даній роботі є доцільним, оскільки воно повністю задовольняє вимоги до створення навчальної віртуальної інфраструктури, забезпечуючи баланс між функціональністю, зручністю та доступністю.

2.2 Аналіз та вибір програмного забезпечення для побудови інфраструктури

Для створення проєкту на базі програмного середовища Oracle VM VirtualBox було використано серверну операційну систему Windows Server 2022 Standard Evaluation. та клієнтську ОС Windows 10 Pro. Такий вибір обумовлений необхідністю моделювання реального корпоративного середовища, у якому сервер виконує функції централізованого керування, а клієнтські машини виступають об’єктами розгортання та адміністрування.

Windows Server 2022 – це сучасна, гнучка та продуктивна серверна операційна система для розгортання й підтримки різноманітних додатків і служб. Вона оснащена розширеною багаторівневою системою безпеки, а також підтримує гібридне підключення до середовища Microsoft Azure, що дозволяє організаціям поєднувати локальну інфраструктуру з хмарними сервісами.

Окрім цього, в Windows Server 2022 значно покращено безпеку мережевих з’єднань. Доступні нові механізми захищеного з’єднання, які забезпечують більш швидку та безпечну обробку HTTPS-запитів, а також використання сучасного стандарту SMB з шифруванням AES-256, що гарантує надійний захист даних під час передавання у мережі. Такі зміни особливо важливі для корпоративних середовищ, де безпека інформації має першочергове значення.

Операційна система Windows Server 2022 також значно спрощує управління гібридною інфраструктурою. Можливості адміністрування були суттєво розширені за рахунок удосконалення Windows Admin Center: покращено інструменти керування віртуальними машинами, оптимізовано перегляд журналів подій, а також додано нові функції моніторингу й діагностики стану системи.

Велику увагу розробники також приділили контейнеризації. У новій версії суттєво зменшено розміри образів контейнерів, що прискорює їх завантаження і розгортання. Також спрощено процес створення та впровадження мережевих політик, покращено сумісність і оптимізовано інструменти контейнеризації для застосунків, розроблених на платформі .NET. Це робить Windows Server 2022 ефективним рішенням для сучасних DevOps-процесів та мікросервісних архітектур.



Рисунок – 2.3 Операційна система Windows Server 2022

Основні редакції Windows Server 2022:

Windows Server 2022 Essentials орієнтований на малий бізнес і призначе-ний для середовищ з до 25 користувачами та 50 пристроями. Він забезпечує спрощене керування сервером та не потребує CAL-ліцензій (Client Access Licenses). Однак ця редакція має обмежений набір серверних ролей у порівнянні з Standard або Datacenter, що робить її оптимальною для невеликих організацій із базовими потребами.

Windows Server 2022 Standard призначений для фізичних та мінімально віртуалізованих середовищ. Він підтримує до 2 віртуальних машин та основні серверні ролі, такі як Active Directory Domain Services (AD DS), DNS, DHCP, файлові служби та інші. Для використання цієї редакції необхідні CAL-ліцензії (Client Access Licenses).

Windows Server 2022 Datacenter орієнтований на великі центри обробки даних та середовища з інтенсивною віртуалізацією. Ця редакція дозволяє запускати необмежену кількість віртуальних машин, підтримує Storage Spaces Direct (S2D) та Software-Defined Networking. Datacenter забезпечує максимальний функціонал сервера та також потребує CAL-ліцензій.

Windows Server 2022 Datacenter: Azure Edition розроблений спеціально для запуску в Microsoft Azure та Azure Stack HCI. Він містить усі можливості редакції Datacenter, а також додаткові хмарні функції, такі як Hotpatch та Extended Security Updates. Ця редакція не призначена для звичайної локальної установки.

Після порівняння цих основних редакцій було визначено, що для цього проекту підходить редакція Windows Server 2022 Standard. Для зручності налаштувань було використано версію з графічним інтерфейсом.

Під час вибору клієнтської операційної системи було обрано Windows 10 Pro.

Windows 10 – це операційна система сімейства Windows NT від компанії Microsoft, яка є безпосереднім наступником Windows 8.1, випущеної майже два роки раніше. Виробнича версія Windows 10 була представлена 15 липня 2015 року, а в роздрібну продаж надійшла 29 липня 2015 року.

Основні редакції Windows 10:

Windows 10 Home – призначена для домашніх користувачів та персонального використання. Має базовий набір функцій, включно з Microsoft Edge, Cortana, Windows Hello та підтримкою ігор. Не підтримує деякі корпоративні функції, такі як шифрування BitLocker чи приєднання до домену.

Windows 10 Pro – орієнтована на малий бізнес та професійних користувачів. Включає всі функції Home, а також додаткові можливості для безпеки та управління: BitLocker, Remote Desktop, приєднання до домену та Azure Active Directory. Підтримує управління груповими політиками та функції для віддаленого адміністрування.

Windows 10 Pro for Workstations – розширена версія Pro для потужних робочих станцій. Підтримує високопродуктивні апаратні платформи, файлову систему ReFS, більш швидкий доступ до даних через SMB Direct та управління великими обсягами оперативної пам’яті.

Windows 10 Enterprise – орієнтована на великі організації та корпоративні середовища. Має всі функції Pro, а також додаткові можливості для безпе-ки, управління та розгортання в масштабі підприємства. Підтримує AppLocker, Windows Defender Advanced Threat Protection (ATP) та розширене управління оновленнями.

Windows 10 Education – спеціальна редакція для навчальних закладів. Включає функціонал Enterprise і додаткові інструменти для навчання та управління комп’ютерами в освітніх мережах. Часто постачається через академічні ліцензії.

Оскільки версія Windows 10 Home не підтримує приєднання до домену, то обрано версію Windows 10 Pro, яка є найбільш функціональна та оптимальна для роботи.

Окрім цього, Windows 10 Pro має стабільну роботу у доменній інфраструктурі, добре інтегрується з серверними версіями Windows (Windows Server 2016/2019/2022) та забезпечує сумісність із більшістю професійного програмного забезпечення. Враховуючи баланс між розширеним функціоналом і помірними системними вимогами, саме ця версія є найбільш доцільною та оптимальною для побудови та тестування доменної мережі.



Рисунок – 2.4 Операційна система Windows 10

2.3 Проєктування логічної структури та IP-адресації розподіленої системи розгортання ОС

На початку проектування потрібно створити архітектуру автоматизованого розгортання ОС із використанням ролей Domain Controller, Windows Deployment Services (WDS) та Microsoft Deployment Toolkit (MDT). Кожен із серверів виконує окрему функцію в загальному процесі: від видачі мережевих параметрів і ініціалізації завантаження до повноцінного встановлення операційної системи та її налаштування.

Розроблена архітектура автоматизованого розгортання операційних систем повинна вирішувати такі задачі:

1. Централізація управління

Усі основні компоненти системи (образи ОС, драйвери, програми, сценарії встановлення) повинні зберігатися та керуватися централізовано на сервері розгортання. Це спрощує адміністрування та оновлення системи.

2. Автоматизація процесів

Процес встановлення операційної системи має виконуватись автоматично без або з мінімальною участю користувача. Це досягається за рахунок використання сценаріїв (Task Sequence) та попередньо налаштованих параметрів.

3. Масштабованість

Архітектура повинна підтримувати розгортання ОС як на одному, так і на великій кількості клієнтських комп'ютерів одночасно без значного зниження продуктивності.

У системі автоматизованого розгортання операційних систем використовується підхід горизонтального масштабування, який полягає у збільшенні продуктивності та пропускної здатності системи шляхом додавання нових серверів до існуючої інфраструктури. Основна ідея цього підходу полягає в тому, що навантаження не концентрується на одному сервері, а розподіляється між кількома вузлами, що дозволяє системі ефективно працювати при зростанні кількості клієнтських пристроїв.

Застосування горизонтального масштабування у даній системі дозволяє створювати декілька серверів розгортання, які виконують однакові або розподілені функції. Наприклад, можуть бути розгорнуті кілька WDS-серверів для обробки PXE-запитів, а також кілька MDT-середовищ із власними Deployment Share. Це дає можливість рівномірно розподіляти навантаження під час масового розгортання операційних систем.

Крім того, горизонтальне масштабування передбачає використання розподіленого файлового сховища, де образи операційних систем, драйвери та пакети оновлень можуть зберігатися на декількох серверах або в централізованій системі з реплікацією даних. Також DHCP/PXE-інфраструктура може бути розділена

між кількома серверами, що підвищує надійність отримання мережевих параметрів та завантаження клієнтів.

4. Модульність та розподіл ролей

Кожен сервер виконує чітко визначену функцію (DC, WDS, MDT), що дозволяє гнучко змінювати або масштабувати окремі компоненти без впливу на всю систему.

5. Уніфікація та стандартизація

Встановлення ОС повинно виконуватись за єдиними шаблонами, що гарантує однакову конфігурацію всіх клієнтських ПК незалежно від кількості та часу розгортання.

6. Надійність та відмовостійкість

Система повинна забезпечувати стабільну роботу навіть при навантаженні або часткових збоях, а також мати можливість швидкого відновлення.

7. Ефективне використання мережевих ресурсів

Передача образів і даних повинна бути оптимізована (наприклад, за допомогою multicast), щоб зменшити навантаження на мережу під час масового розгортання.

8. Гнучкість налаштування

Архітектура повинна дозволяти змінювати сценарії встановлення, додавати нові образи ОС, драйвери та програми без значної перебудови системи.

Призначення та ролі в архітектурі:

Domain Controller (Srv-40-3-1-DC-DHCP-SERVER)

Це основний сервер, який відповідає за роботу всієї мережі.

Основне призначення цього сервера – керування доменом через Active Directory: зберігання облікових записів користувачів і комп’ютерів, а також налаштування та адміністрування групових політик і дискових квот.

На сервері розгорнута служба Active Directory Domain Services (AD DS), яка забезпечує централізовану автентифікацію користувачів і комп’ютерів у домені, а також контроль доступу до мережевих ресурсів.

Також важливою складовою є служба DNS, яка інтегрована з Active Directory. Вона відповідає за перетворення доменних імен у IP-адреси та навпаки. Завдяки DNS клієнтські комп'ютери можуть знаходити сервери в мережі (зокрема MDT і WDS), а також коректно працює процес приєднання до домену.

Крім цього, на сервері працює DHCP-сервер. Коли клієнт вмикається, він отримує IP-адресу саме звідси. Разом з IP клієнт також отримує інформацію, куди звертатися для мережевого завантаження (PXE).

Transport Server (Srv-40-3-3-WDS-TRANSPORT)

Цей сервер відповідає за початкове завантаження комп'ютера по мережі.

На даному сервері встановлена роль Windows Deployment Services (WDS), яка забезпечує передачу завантажувальних образів клієнтам. У даній архітектурі сервер функціонує в автономному режимі та не інтегрований у домен Active Directory, що дозволяє використовувати його виключно для виконання транспортних функцій

Коли клієнт отримав IP-адресу від DHCP-сервера, він відправляє PXE-запит на WDS. У відповідь сервер передає файл boot.wim – це спеціальний завантажувальний образ, який містить мінімальне середовище WinPE для подальшого запуску процесу встановлення.

Передача даних на початковому етапі виконується за допомогою протоколу TFTP, після чого клієнт отримує можливість підключитися до сервера розгортання (MDT).

Deployment Server (Srv-40-3-2-MDT-DEPLOYMENT)

Це головний сервер, який виконує розгортання операційної системи.

Після запуску WinPE клієнт підключається до MDT і отримує інструкції, що робити далі. На цьому сервері зберігається все необхідне:

1. образ Windows (install.wim)
2. драйвери
3. програми
4. сценарії встановлення (Task Sequence)

MDT автоматично виконує весь процес:

- розбиває диск;
- копіює і встановлює Windows;
- ставить драйвери під конкретне обладнання;
- встановлює програми;
- додає комп’ютер у домен.

Клієнтський ПК (тестова віртуальна машина)

Це клієнтський комп’ютер, на який виконується централізоване автоматизоване розгортання операційної системи через мережеве завантаження PXE. Після увімкнення пристрій ініціалізує мережеву карту та запускає PXE-boot, після чого отримує IP-адресу, параметри мережі та шлях до завантажувальних файлів від DHCP-сервера, що працює на контролері домену.

Далі система завантажує середовище попередньої інсталяції WinPE з сервера WDS, яке забезпечує базове середовище для запуску процесу розгортання. Після старту WinPE клієнт підключається до інфраструктури MDT, де виконується вибір або автоматичне застосування відповідного сценарію встановлення. У процесі інсталяції автоматично розгортається операційна система, встановлюються необхідні драйвери та прикладне програмне забезпечення згідно з попередньо налаштованим завданням розгортання.

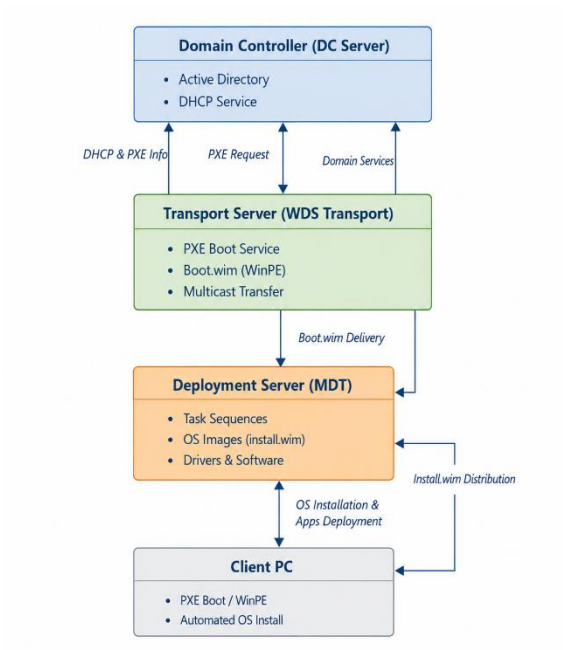


Рисунок – 2.5 Логічна схема мережі

Для правильної роботи та функціювання мережевої інфраструктури необхідно правильно спланувати IP-адресацію. У даному проєкті було виконано розрахунок параметрів підмережі на основі заданої IP-адреси та маски. Визначено адресу мережі, діапазон доступних IP-адрес для хостів, широкомовну адресу та шлюз за замовчуванням. Результати розрахунку IP-адрес для підмережі наведено в таблиці 2.1.

Таблиця 2.1

Розрахунок IP-адрес для підмережі

Призначення	IP-адреса
Адреса мережі	192.168.3.0
Маска мережі	255.255.255.224/27
Інвертована маска	0.0.0.31
Мінімальна адреса	192.168.3.1
Максимальна адреса	192.168.3.30
Широкомовна адреса	192.168.3.31
Шлюз по замовчуванню	192.168.3.1
Кількість адрес для ПК	30
Кількість всіх доступних адрес	32

У таблиці 2.2 наведено схему IP-адресації вузлів розробленої комп'ютерної мережі. Базовою адресою мережі є 192.168.3.0/27 із маскою підмережі 255.255.255.224. Використання префікса /27 обмежує ємність мережі до 30 доступних адрес для хостів. Таке рішення є архітектурно обґрунтованим для тестового середовища, оскільки воно дозволяє ефективно використовувати адресний простір та мінімізувати розмір широкомовного домену.

Враховуючи те, що інфраструктура розгортається у середовищі віртуалізації (VirtualBox), перші дві IP-адреси (192.168.3.1 та 192.168.3.2) зарезервовані гіпервізором для власних потреб. Зокрема, адреса 192.168.3.1 виконує функцію шлюзу за замовчуванням (Default Gateway) і забезпечує маршрутизацію трафіку для всіх вузлів інфраструктури.

Таблиця 2.2

Схема адресації пристроїв мережі

Мережа / Пристрій	Інтерфейс / Мережний адаптер / Шлюз	IP-адреса	Маска
Мережа	-	192.168.3.0/27	255.255.255.224
	Зарезервовані адреси VBox	192.168.3.1 192.168.3.2	255.255.255.224
Windows сервер Srv-40-3-1-DC-DHCP-SERVER	Мережний адаптер	192.168.3.3	255.255.255.224
	Шлюз за замовчуванням / Public DNS	192.168.3.1	-
Windows сервер Srv-40-3-2-MDT-DE-PLOYMENT	Мережний адаптер	192.168.3.4	255.255.255.224
	Шлюз за замовчуванням / Public DNS	192.168.3.1	-
Srv-40-3-3-WDS-TRANSPORT	Мережний адаптер	192.168.3.5	255.255.255.224
	Шлюз за замовчуванням / Public DNS	192.168.3.1	-
Робоча станція Windows 10	Мережний адаптер	DHCP	
	Шлюз за замовчуванням / Public DNS		

Після визначення основних параметрів підмережі було сформовано пул DHCP-адрес. Частина IP-адрес зарезервована для мережевого обладнання та серверів, а решта адрес автоматично видається робочим станціям у локальній мережі. Розподіл динамічних і зарезервованих IP-адрес наведено в таблиці 2.3.

Таблиця 2.3

Пул DHCP-адрес

IP-адреса	Статус адрес
192.168.3.6 - 192.168.3.30	Виділені адреси для робочих станцій
192.168.3.1 - 192.168.3.5	Зарезервованні адреси

2.4 Налаштування базової мережевої інфраструктури та серверного забезпечення

У межах кваліфікаційної роботи віртуальна мережа типу NAT Network у середовищі VirtualBox розглядається як ізольоване мережеве середовище, призначене для взаємодії виключно між віртуальними машинами. У даній конфігурації доступ до зовнішньої мережі Інтернет не використовується або

обмежується, що дозволяє створити контрольований сегмент для тестування та розгортання служб.

Віртуальні машини, підключені до NAT Network, функціонують у межах окремої підмережі з власною адресацією, що не маршрутизується у фізичну мережу. Обмін трафіком відбувається лише між вузлами всередині цієї віртуальної інфраструктури, що забезпечує її логічну ізоляцію від зовнішніх впливів.



Рисунок – 2.6 Створення NAT Networks та налаштування параметрів

Після створення віртуальної мережі типу NAT Network у середовищі VirtualBox було виконано підключення всіх віртуальних машин, що входять до складу розроблюваної інфраструктури. Для кожної віртуальної машини у налаштуваннях мережевого адаптера обрано режим підключення «Мережа NAT» із вказанням створеної мережі NatNetwork (WDS).

Для забезпечення сумісності та стабільної роботи обрано тип мережевого адаптера Intel PRO/1000 MT Desktop (82540EM), який повністю підтримується більшістю серверних операційних систем.

Аналогічна конфігурація була застосована до всіх інших віртуальних машин, що входять до складу системи (контролер домену, сервер розгортання, транспортний сервер, клієнтські машини тощо). Усі вузли підключено до однієї віртуальної мережі NatNetwork (WDS), що забезпечує їхню взаємодію в межах єдиного ізольованого середовища.

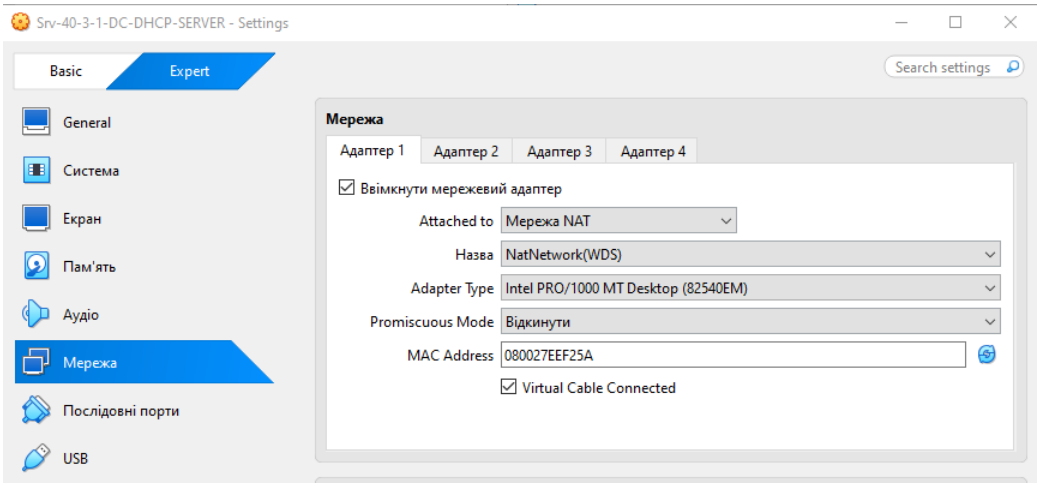


Рисунок – 2.7 Підкдючення сервера до NAT Networks

Для забезпечення коректної та стабільної роботи віртуальних машин, на яких встановлено серверні операційні системи, необхідно виділити достатній обсяг оперативної пам’яті. У межах даного проєкту для таких віртуальних машин було рекомендовано виділяти не менше 3 ГБ оперативної пам’яті.

Це обумовлено тим, що сучасні серверні операційні системи мають підвищені вимоги до апаратних ресурсів, особливо при одночасному використанні кількох ролей і служб, таких як DHCP, DNS, Active Directory.

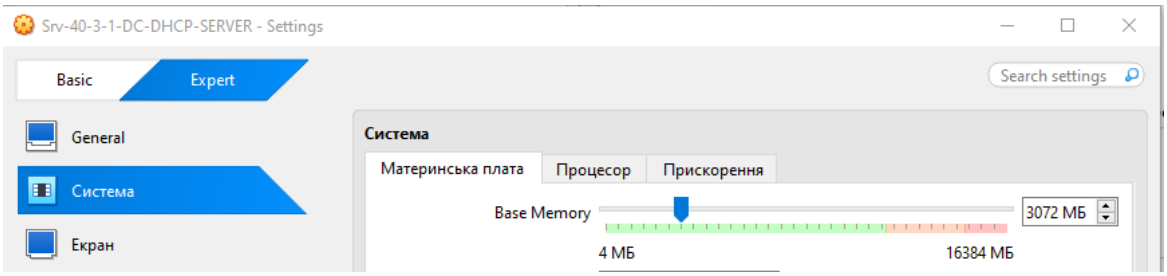


Рисунок – 2.8 Налаштування кількості оперативної пам’яті

На прикладі DC показано вибір 3 логічних процесорів є ідеальним балансом: ця кількість знаходиться в безпечній «зеленій зоні» налаштувань, що залишає ресурси для стабільної роботи фізичного комп'ютера і запобігає зависанням. Одночасно з цим, оскільки віртуальна машина виконує легку роль DHCP-сервера, трьох ядер більш ніж достатньо для миттєвої обробки мережевих запитів, швидкої роботи самої операційної системи та встановлення оновлень.

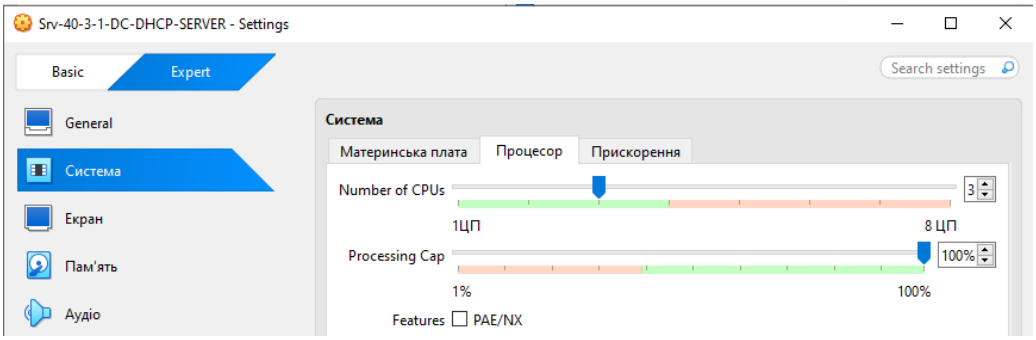


Рисунок – 2.9 Налаштування кількості логічних процесорів

На початковому етапі розгортання сервера у середовищі віртуалізації Oracle VirtualBox було запущено інсталяцію операційної системи. Першим кроком виконано налаштування базових регіональних параметрів: мови інтерфейсу, формату часу і грошових одиниць, а також розкладки клавіатури. Для уніфікації середовища обрано стандартні параметри English (United States).

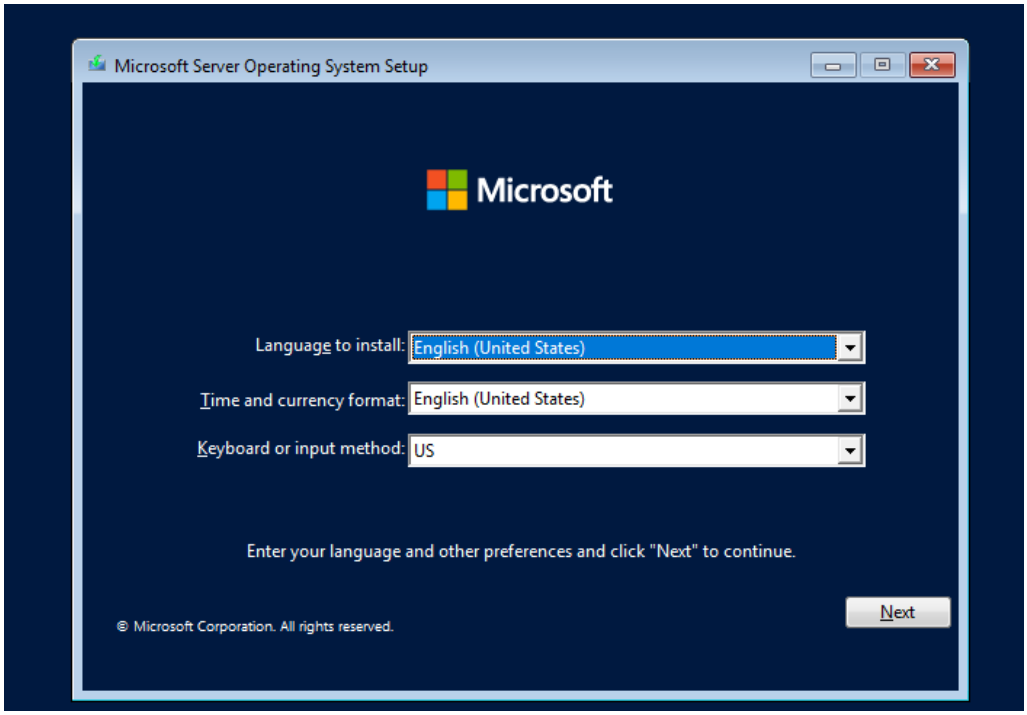


Рисунок – 2.10 Налаштування мовних параметрів та часового формату

Далі, зі списку доступних редакцій, для встановлення обрано Windows Server 2022 Standard Evaluation (Desktop Experience). Використання версії з графічним інтерфейсом (GUI) обумовлене необхідністю спрощення процесів подальшого налаштування, адміністрування та тестування серверних ролей.

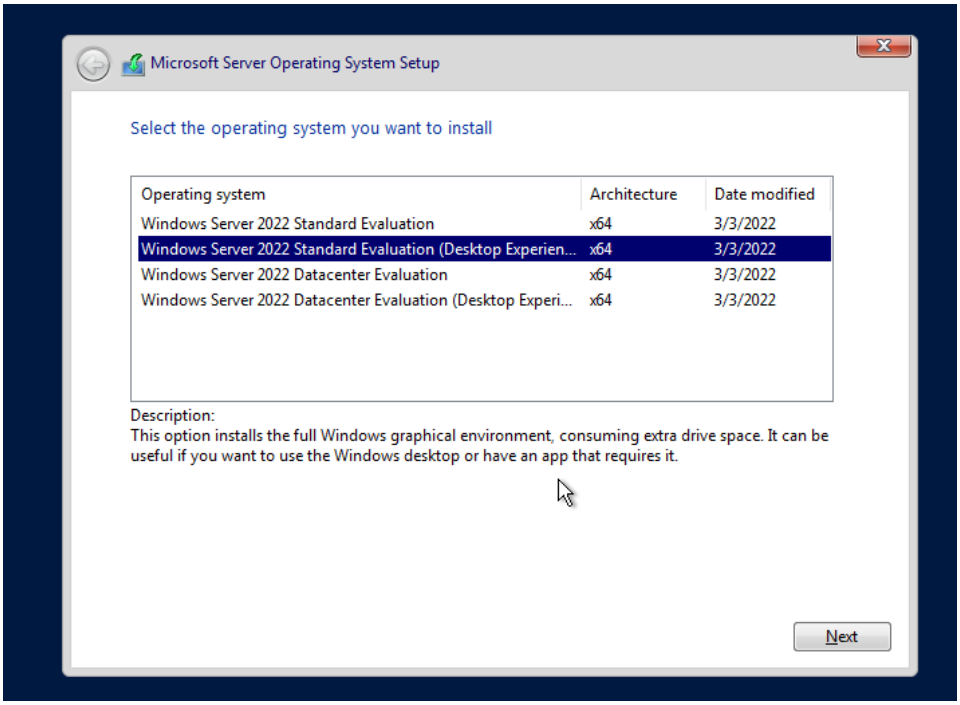


Рисунок – 2.11 Вибір необхідної редакції під час встановлення

Після ознайомлення з умовами ліцензійної угоди (Microsoft Software License Terms) та їх прийняття було обрано тип інсталяції Custom: Install Microsoft Server Operating System only (advanced). Даний режим забезпечує виконання «чистого» встановлення операційної системи та надає повний контроль над розподілом дискового простору.

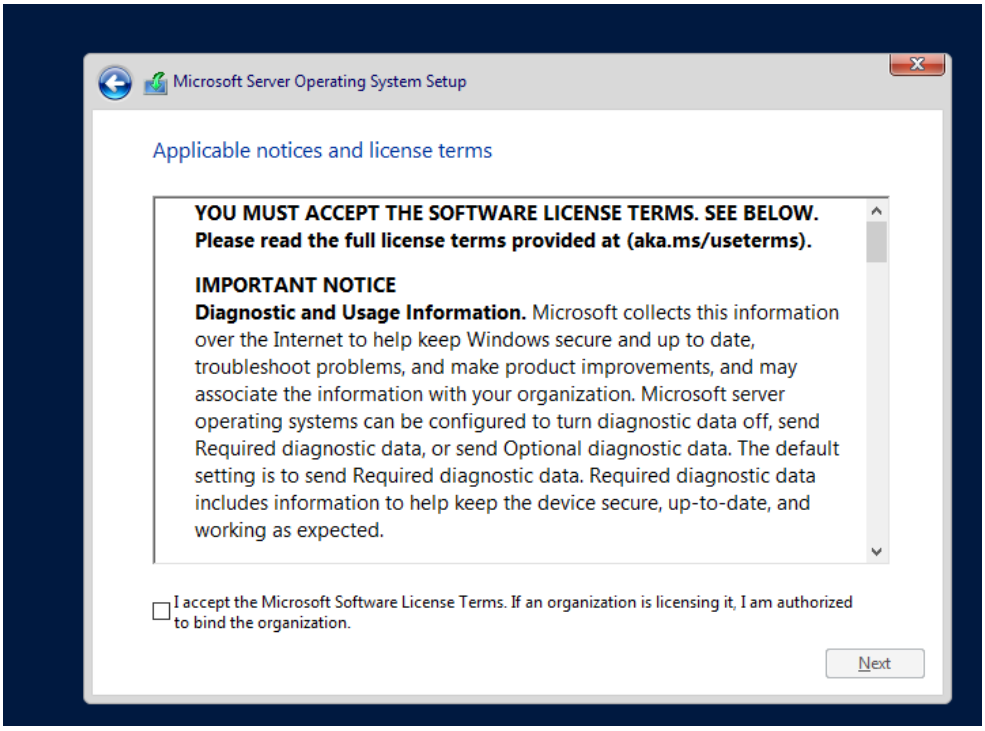


Рисунок – 2.12 Прийняття ліцензійної угоди

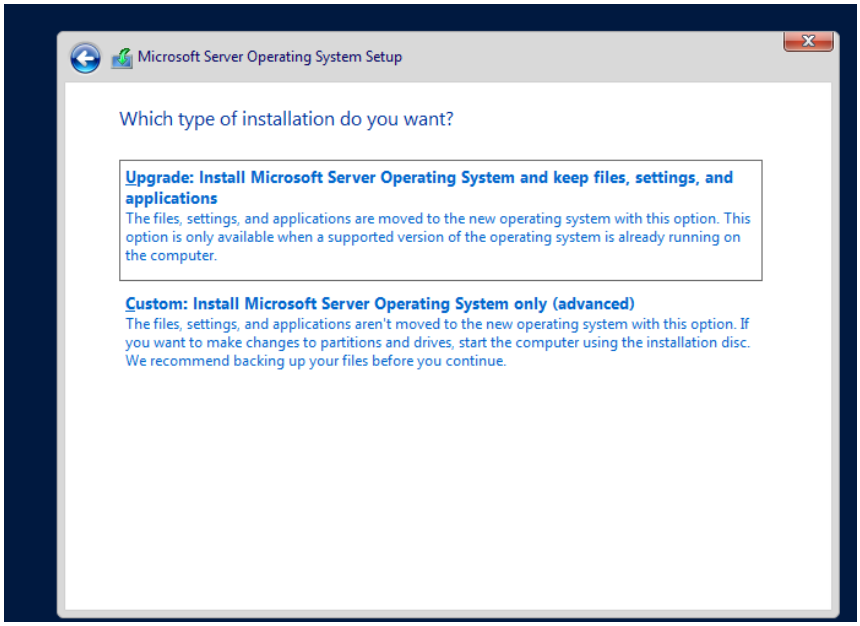


Рисунок – 2.13 Вибір типу інсталяції – Custom

На етапі налаштування дискової підсистеми для встановлення було використано нерозмічений простір віртуального жорсткого диска. Під час створення основного розділу система автоматично сформувала службовий розділ System Reserved обсягом близько 100 МБ, який використовується для зберігання завантажувальних файлів і системних компонентів. Основний розділ обсягом приблизно 50 ГБ було обрано для інсталяції операційної системи.

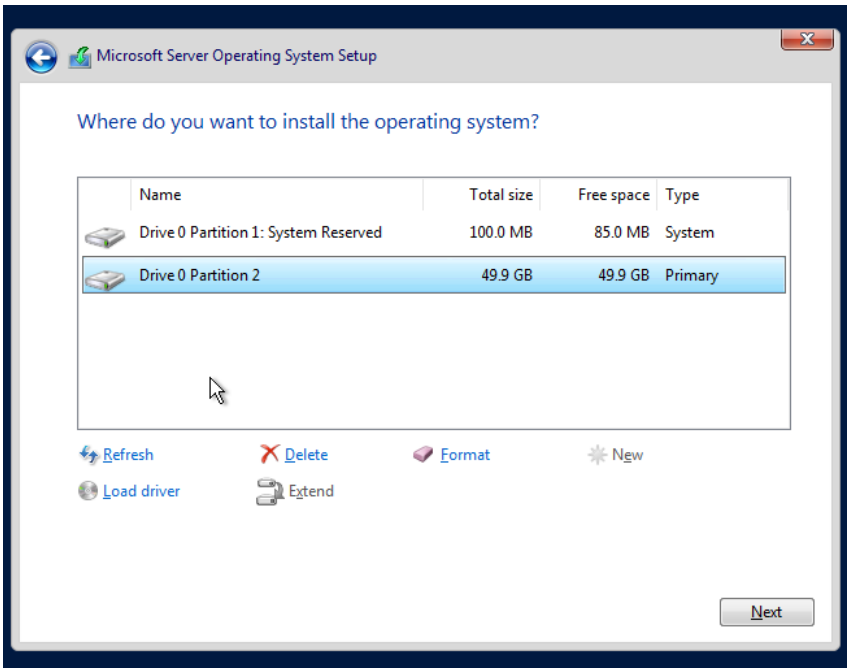


Рисунок – 2.14 Створення стандартних дискових томів під час встановлення Windows Server

Після підтвердження параметрів диска розпочався автоматизований процес встановлення, що включає копіювання файлів, їх підготовку та інсталяцію компонентів Windows. Завершальним етапом стало налаштування параметрів безпеки: у вікні персоналізації було задано складний пароль для вбудованого облікового запису адміністратора .

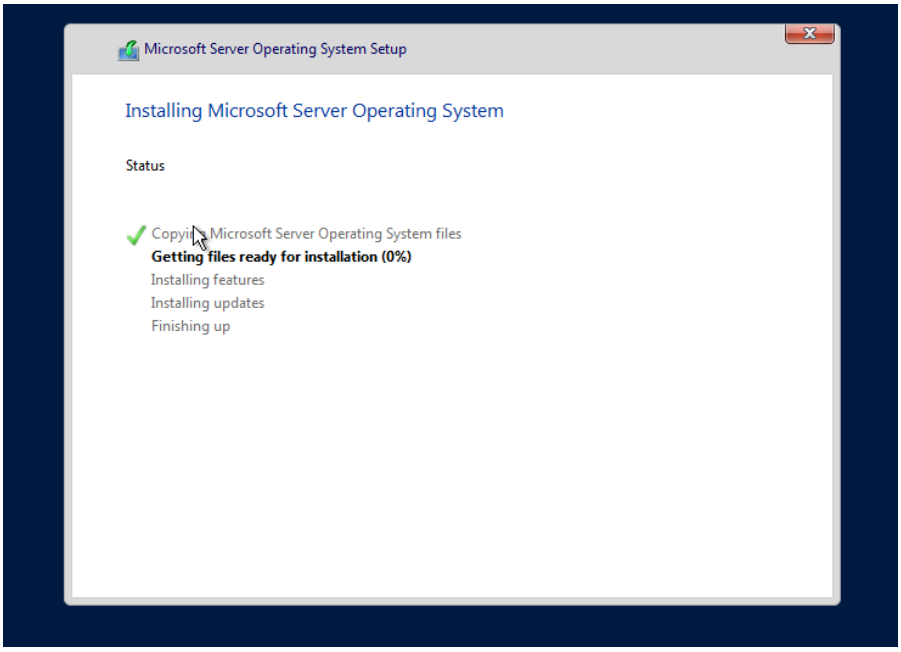


Рисунок – 2.14 Процес встановлення Windows Server 2022

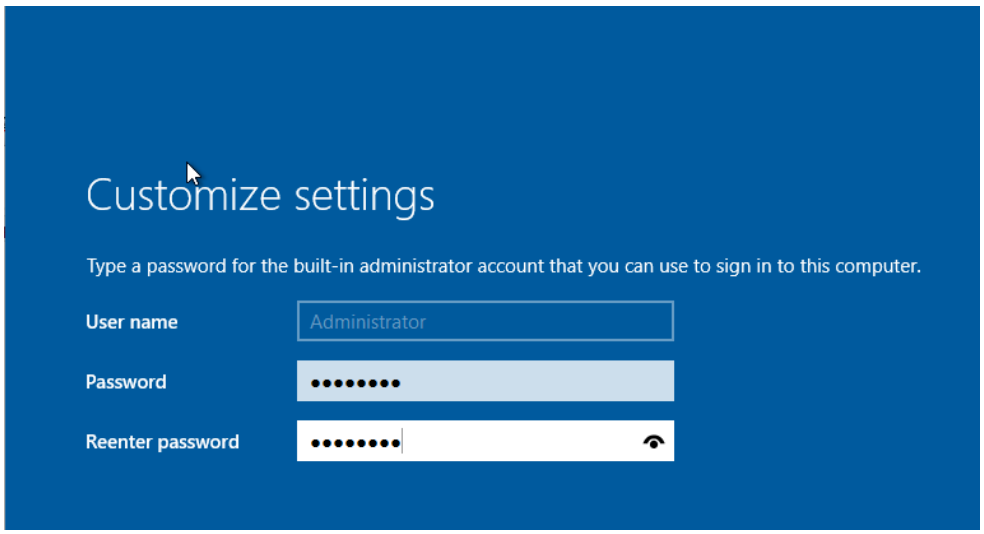


Рисунок – 2.15 Встановлення пароля для локального адміністратора

Іменування вузлів у межах даного проєкту відіграє важливу роль у забезпеченні зручності адміністрування, ідентифікації та підтримки мережевої інфраструктури.

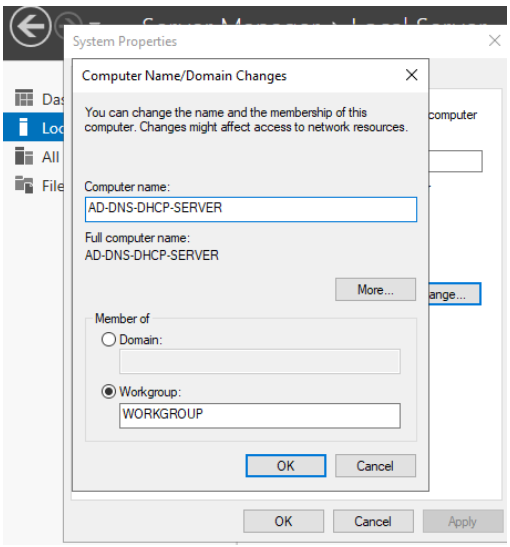


Рисунок – 2.16 Налаштування іменування серверів

Згідно з налаштуваннями, віртуальній машині призначено статичну IP-адресу 192.168.3.3 із маскою підмережі 255.255.255.224 та шлюзом за замовчуванням 192.168.3.1. Така конфігурація відповідає підмережі з обмеженою кількістю вузлів (до 30 хостів), що використовується в лабораторних або тестових середовищах. У параметрах DNS налаштовано використання локального сервера з адресою 127.0.0.1 як основного, а також альтернативного DNS-сервера 192.168.3.1 для резервного доступу до мережевих ресурсів.

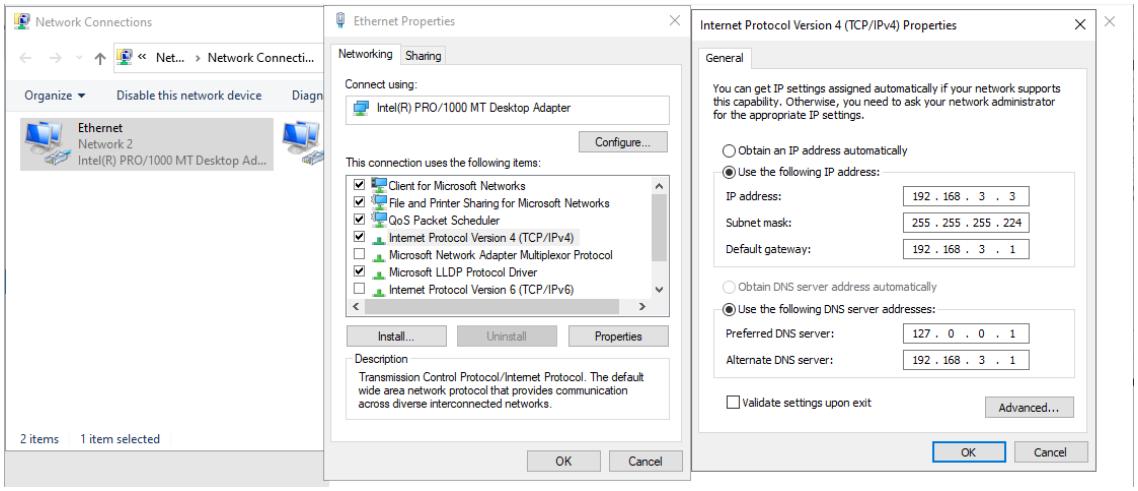


Рисунок – 2.17 Приклад налаштування параметрів адресації на DC

Після виконання необхідних налаштувань на обох серверах було додатково налаштовано брандмауер Windows. Зокрема, у правилах вхідного трафіку дозволено ICMPv4-пакети, що забезпечує коректну перевірку мережевої доступності вузлів та діагностику мережевих з'єднань у межах інфраструктури.

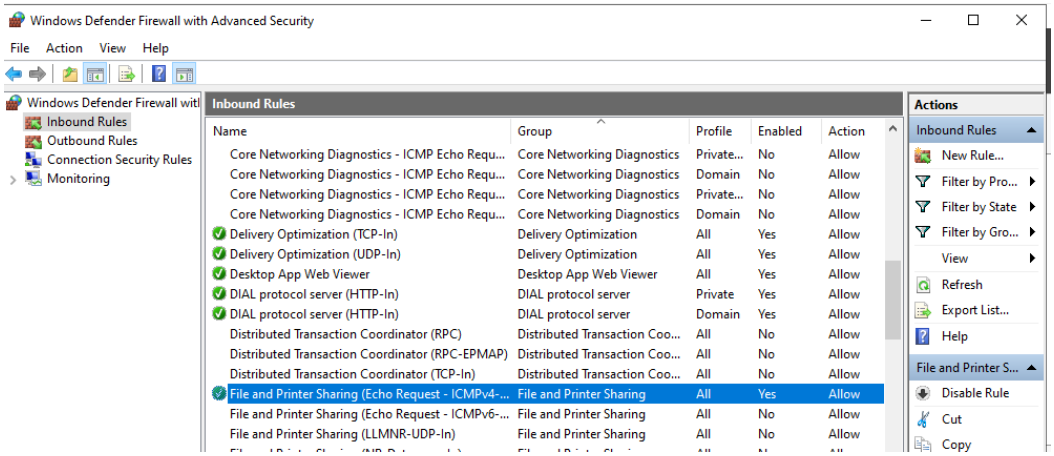


Рисунок – 2.18 Налаштування вхідного ICMPv4-трафіку у брандмауері Windows

На рис. 2.19 виконано перевірку мережевого з'єднання за допомогою команди ping до IP-адрес 192.168.3.4 та 192.168.3.1. Адреса 192.168.3.4 відповідає серверу в межах підмережі, тоді як 192.168.3.1 є шлюзом за замовчуванням. Обидва вузли успішно відповідають на запити, що підтверджує наявність коректного мережевого з'єднання між пристроями.

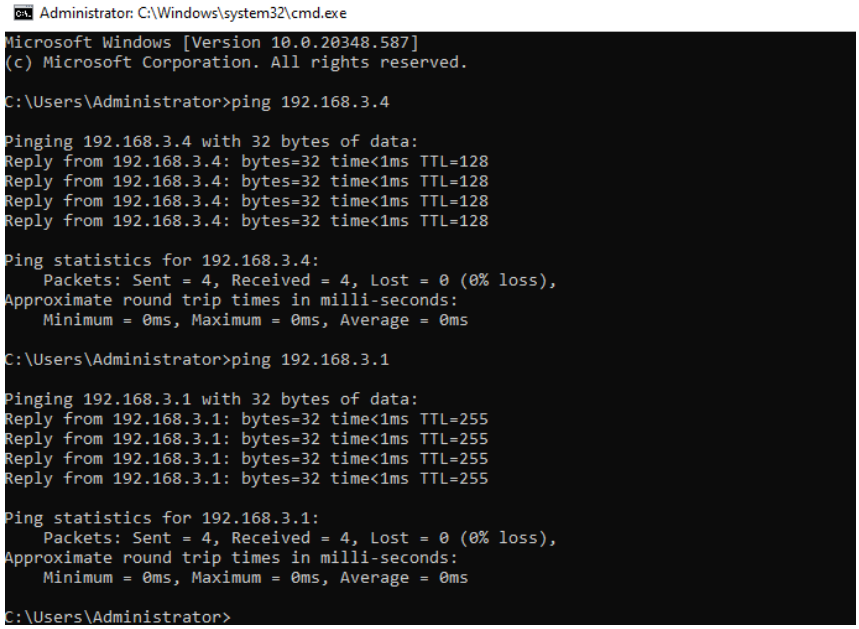


Рисунок – 2.19 Перевірка зв'язку між вузлами мережі

Висновки до другого розділу

У другому розділі виконано проєктування та розгортання серверної інфраструктури для автоматизованого встановлення операційних систем. Як середовище віртуалізації обрано VirtualBox, що дозволило реалізувати лабораторний стенд із

використанням віртуальних машин та відтворити роботу мережевої інфраструктури без застосування фізичного обладнання.

У процесі роботи проаналізовано та впроваджено програмні засоби WDS і MDT, визначено їх функціональне призначення та взаємодію в межах єдиної системи розгортання. Спроєктовано логічну структуру мережі, виконано розподіл серверних ролей і реалізовано схему IP-адресації, що забезпечує коректну маршрутизацію та доступність мережевих сервісів.

Налаштовано базові мережеві служби, зокрема IP-конфігурацію та DNS, а також виконано перевірку мережевої доступності між вузлами. Результати тестування підтвердили коректність налаштувань і стабільну взаємодію компонентів інфраструктури.

Таким чином, у межах розділу реалізовано працездатну серверну інфраструктуру, готову до використання для автоматизованого розгортання операційних систем у наступному етапі роботи.

РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ РОЗПОДІЛЕНОГО РОЗГОРТАННЯ

3.1 Встановлення Active Directory Domain Services та встановлення основних ролей інфраструктури

Процес підготовки до створення контролера домену в середовищі Windows Server починається з активації ролі Active Directory Domain Services (AD DS) через майстер додавання ролей і компонентів. Як показано на рисунку 3.1, під час вибору цієї ролі система автоматично пропонує встановити необхідні залежності, зокрема консоль керування груповими політиками та інструменти віддаленого адміністрування сервера (RSAT).

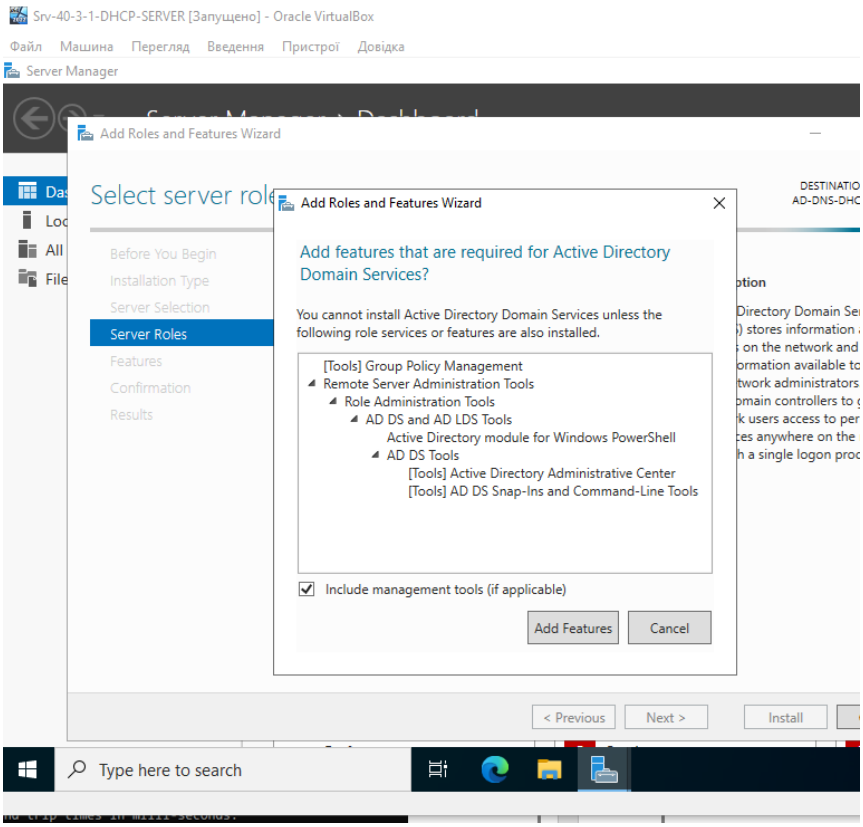


Рисунок – 3.1 Додавання ролі AD DS

На рисунку 3.2 показано наступний етап налаштування контролера домену – конфігурацію розгортання (Deployment Configuration) у майстрі Active Directory Domain Services. На цьому кроці було обрано створення нового лісу (Add a new forest), тобто розгортання доменної інфраструктури з нуля.

У полі «Root domain name» вказано ім’я кореневого домену –corp.net, яке буде використовуватися як основне ім’я для всієї мережі. Воно потрібне для ідентифікації користувачів, комп’ютерів і ресурсів у створеному домені.

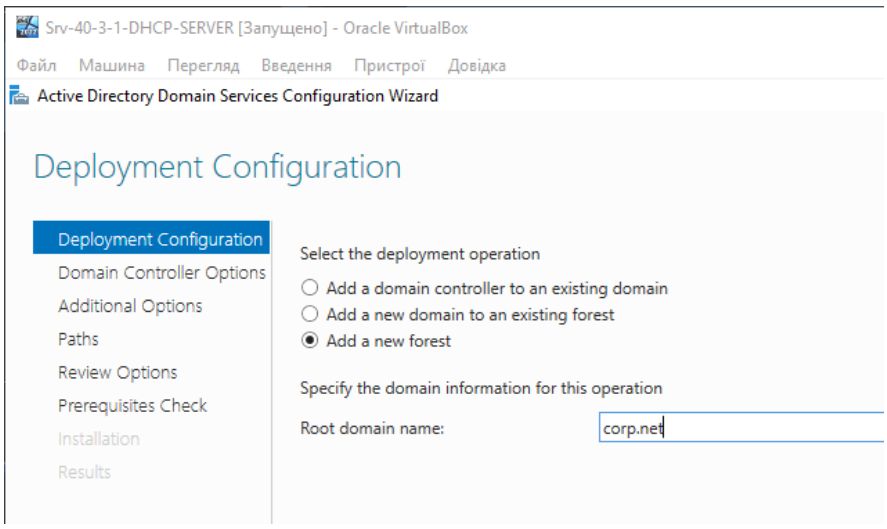


Рисунок – 3.2 Виконання Post-Deployment Configuration

На рисунку 3.3 представлено головне вікно диспетчера серверів (Server Manager Dashboard) після успішного завершення процесу інсталяції та первинної конфігурації. У розділі «Roles and Server Groups» відображається перелік активованих ролей, серед яких –AD DS та DNS.

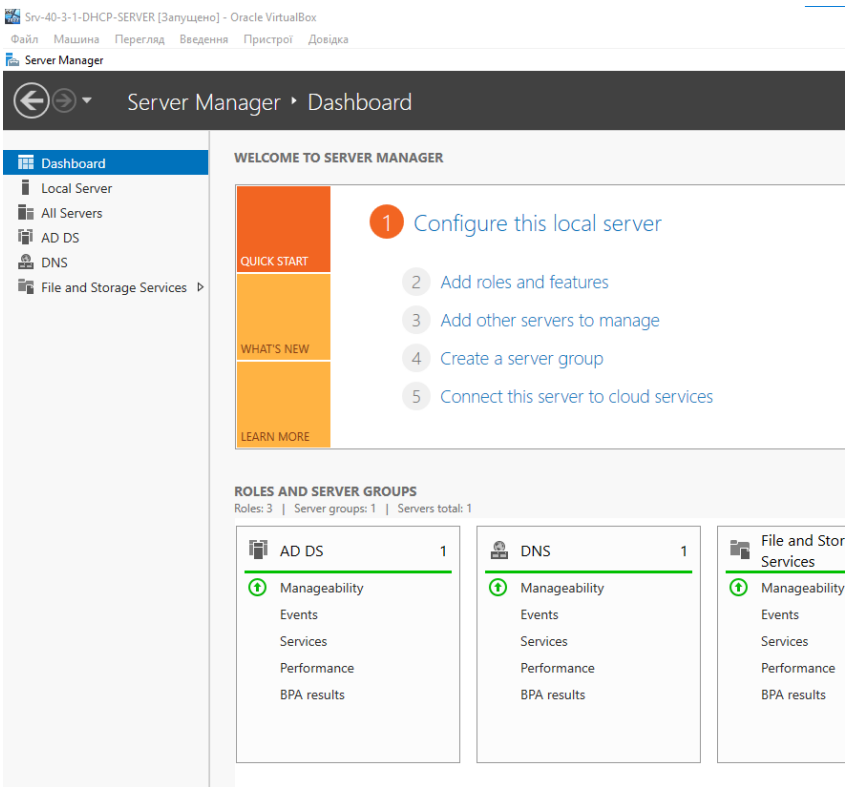


Рисунок – 3.3 Успішне встановлення AD DS та DNS

Для забезпечення роботи PXE-завантаження необхідна наявність DHCP-сервера, оскільки саме він відповідає за автоматичну видачу IP-адрес клієнтським пристроям, а також передачу параметрів мережевого завантаження.

У даному проєкті в якості DHCP-сервера використовується контролер домену, на якому додатково встановлюється відповідна серверна роль. Такий підхід є доцільним для невеликих або тестових мереж, оскільки дозволяє спростити інфраструктуру та зменшити кількість задіяних серверів.

На рисунку 3.4 показано процес додавання ролі DHCP-сервера за допомогою майстра «Add Roles and Features Wizard». На цьому етапі система автоматично пропонує встановити додаткові компоненти, необхідні для керування цією роллю, зокрема DHCP Server Tools, які входять до складу засобів віддаленого адміністрування сервера (RSAT).

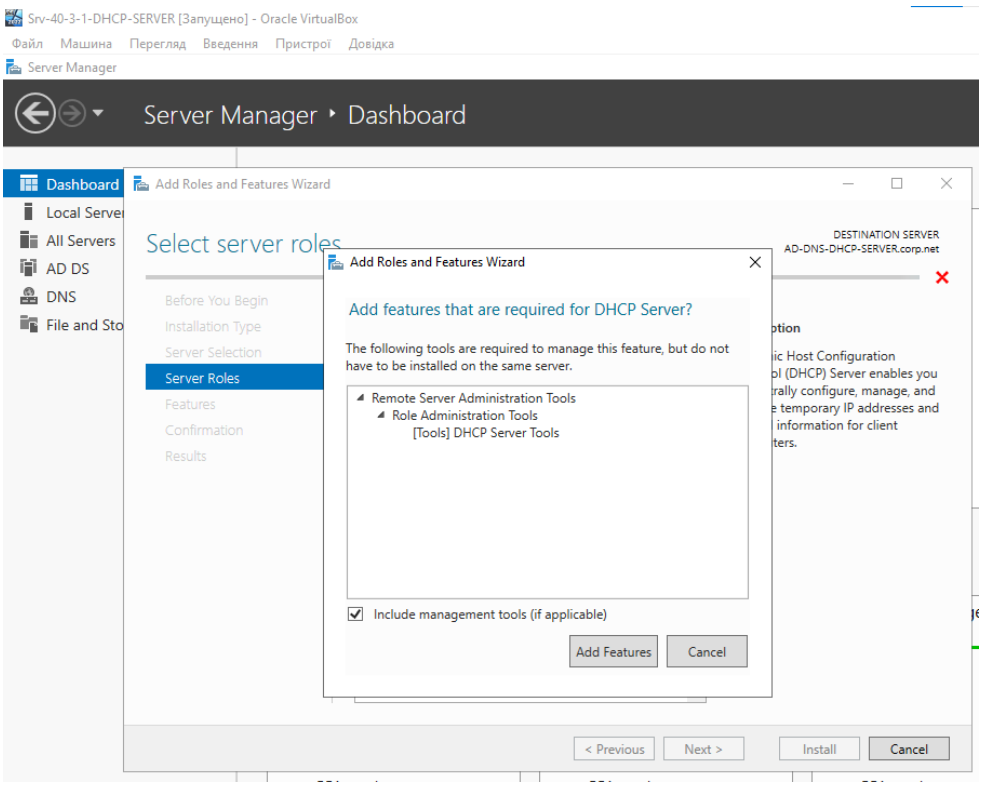


Рисунок – 3.4 Встановлення ролі DHCP-сервера

Процес створення нової області розподілу (пулу) IP-адрес виконується у консолі керування DHCP за допомогою майстра New Scope Wizard. На цьому етапі адміністратор задає основні ідентифікаційні параметри області: у полі «Name» вказано назву wds project, а в полі «Description» –короткий опис wds scope.

Створення області є важливим етапом налаштування DHCP-сервера, оскільки вона визначає діапазон IP-адрес, які будуть автоматично видаватися клієнтським пристроям. Це дозволяє логічно розділити мережу на окремі сегменти та забезпечити централізоване призначення таких параметрів, як IP-адреса, маска підмережі, шлюз за замовчуванням і DNS-сервер для визначеної групи пристроїв.

На рисунку 3.5 представлено відповідний етап налаштування області в DHCP-сервері.

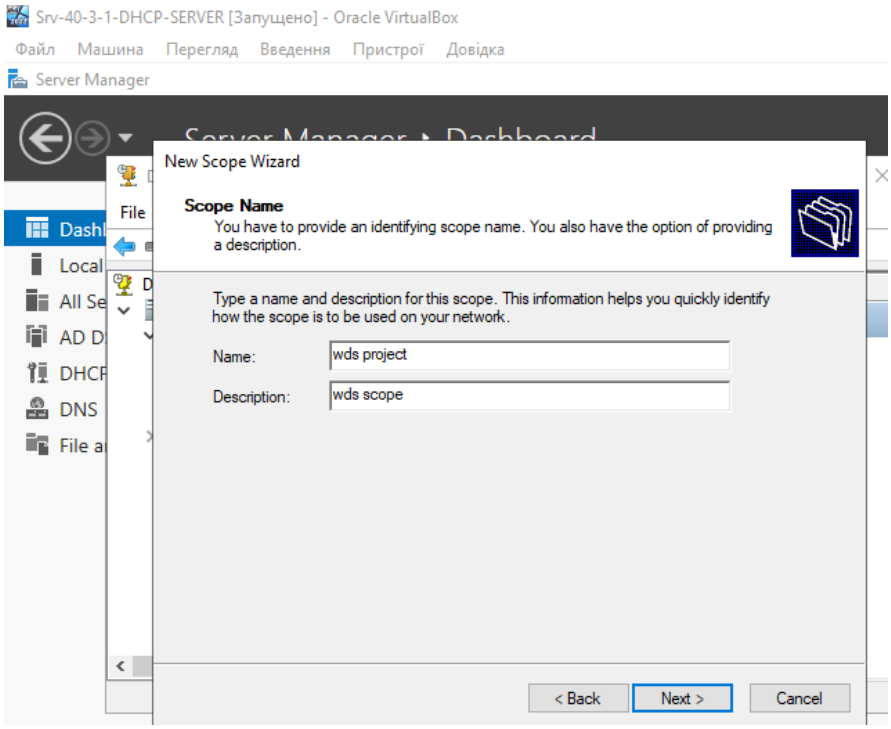


Рисунок – 3.5 Налаштування іменування пулу

Перший етап налаштування області DHCP передбачає визначення діапазону доступних IP-адрес для клієнтських пристроїв (рисунок 3.6). У відповідному вікні задається початкова адреса 192.168.3.6 та кінцева адреса 192.168.3.30, що формують пул адрес, які сервер буде автоматично призначати пристроям у мережі. Також встановлюється маска підмережі /27 (255.255.255.224), яка визначає розмір сегмента та кількість доступних адрес. Таке налаштування дозволяє обмежити кількість динамічних IP-адрес і раціонально використовувати адресний простір у межах локальної мережі.

На наступному етапі виконується налаштування тривалості оренди IP-адреси (Lease Duration) (рисунок 3.7). Для даної області встановлено термін оренди 30 днів, протягом якого клієнтський пристрій зберігає закріплену за ним IP-адресу без

необхідності її повторного отримання. Використання більш тривалого періоду оренди є доцільним для стабільних локальних мереж зі стаціонарними робочими станціями, оскільки це зменшує навантаження на DHCP-сервер і кількість службових запитів у мережі, забезпечуючи стабільність мережевих налаштувань.

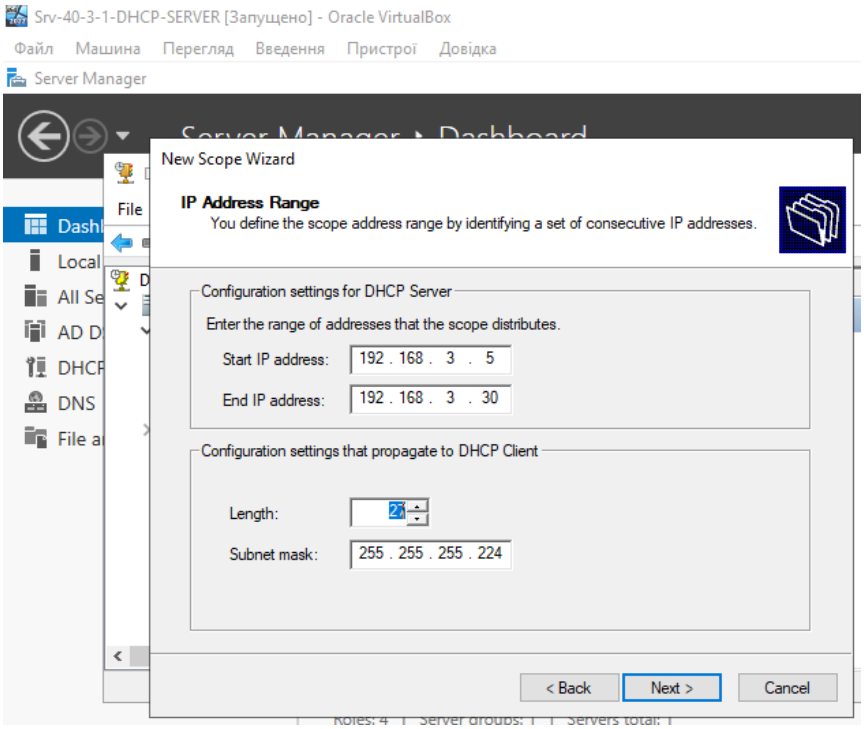


Рисунок – 3.6 Налаштування діапазону для пулу адрес

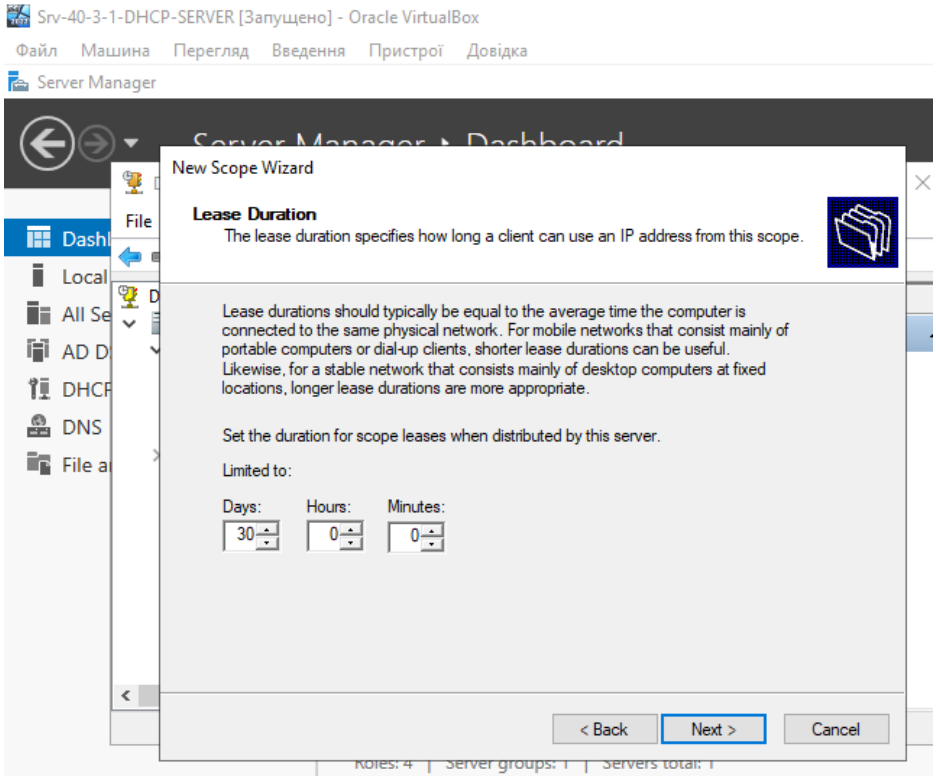


Рисунок – 3.7 Налаштування терміну аренди адреси

На рисунку 3.8 показано етап налаштування параметрів маршрутизації в майстрі New Scope Wizard. У даному вікні виконується конфігурація основного шлюзу (Default Gateway), який буде автоматично передаватися клієнтським пристроям разом із IP-адресою. Вказана адреса 192.168.3.1 визначає вузол (маршрутизатор), через який здійснюватиметься вихід трафіку за межі поточної підмережі та доступ до зовнішніх мережевих ресурсів.

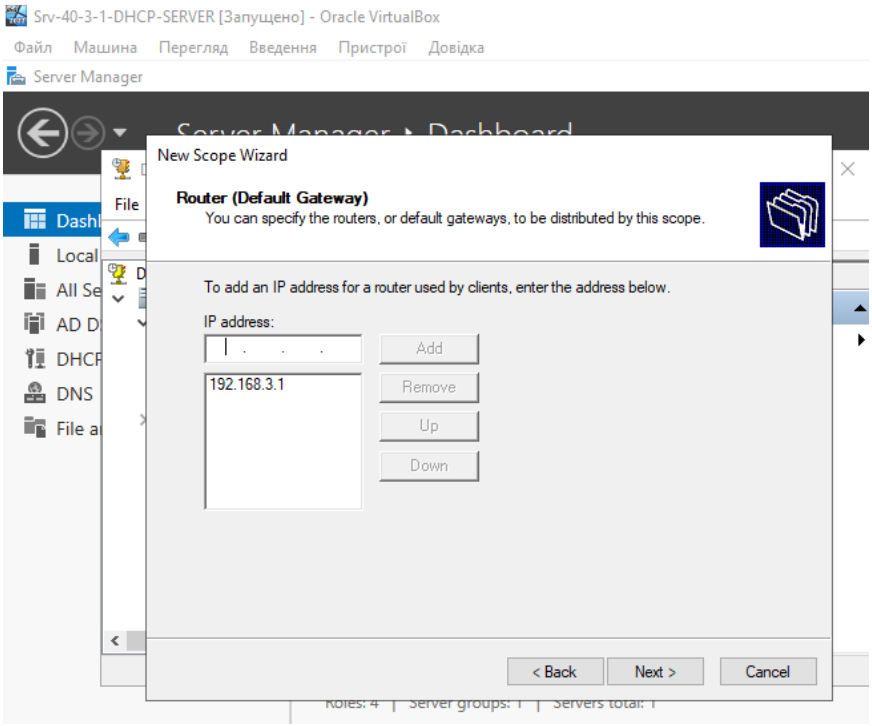


Рисунок – 3.8 Додавання адреси шлюзу

На рисунку 3.9 показано консоль керування DHCP з налаштованими параметрами області (Scope Options). Крім стандартних параметрів, таких як шлюз за замовчуванням (003), DNS-сервер (006) та доменне ім’я (015), додатково налаштовано опції для підтримки PXE-завантаження та служби розгортання операційних систем (WDS).

Зокрема, у параметрі 066 Boot Server Host Name вказано IP-адресу сервера завантаження –192.168.3.5, а в параметрі 067 Bootfile Name задано шлях до завантажувального файлу –boot\x64\wdsmgfw.efi.

Налаштування цих параметрів забезпечує коректну роботу PXE-завантаження, завдяки чому клієнтські комп’ютери автоматично отримують необхідну інформацію для запуску інсталяції операційної системи з мережі. Це є важливою частиною автоматизованого розгортання ОС у рамках проєкту.

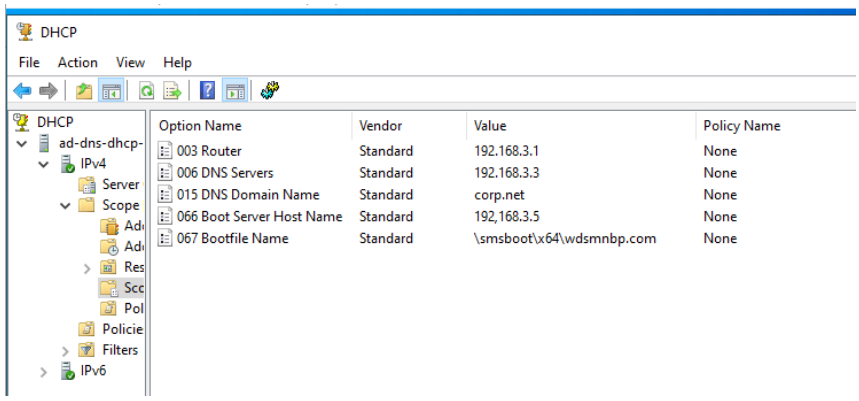


Рисунок – 3.9 Додані опції 66 та 67 для роботи транспортного-сервера

3.2 Створення структури домену та базові групові політики.

Для даного проєкту було розроблено структуру домену з використанням організаційних одиниць (Organizational Units, OU). Такий підхід забезпечує ієрархічну організацію об’єктів домену та спрощує процес адміністрування.

На рисунку 3.10 наведено логічну схему структури користувачів домену, яка відображає розподіл облікових записів за філіями та підрозділами.

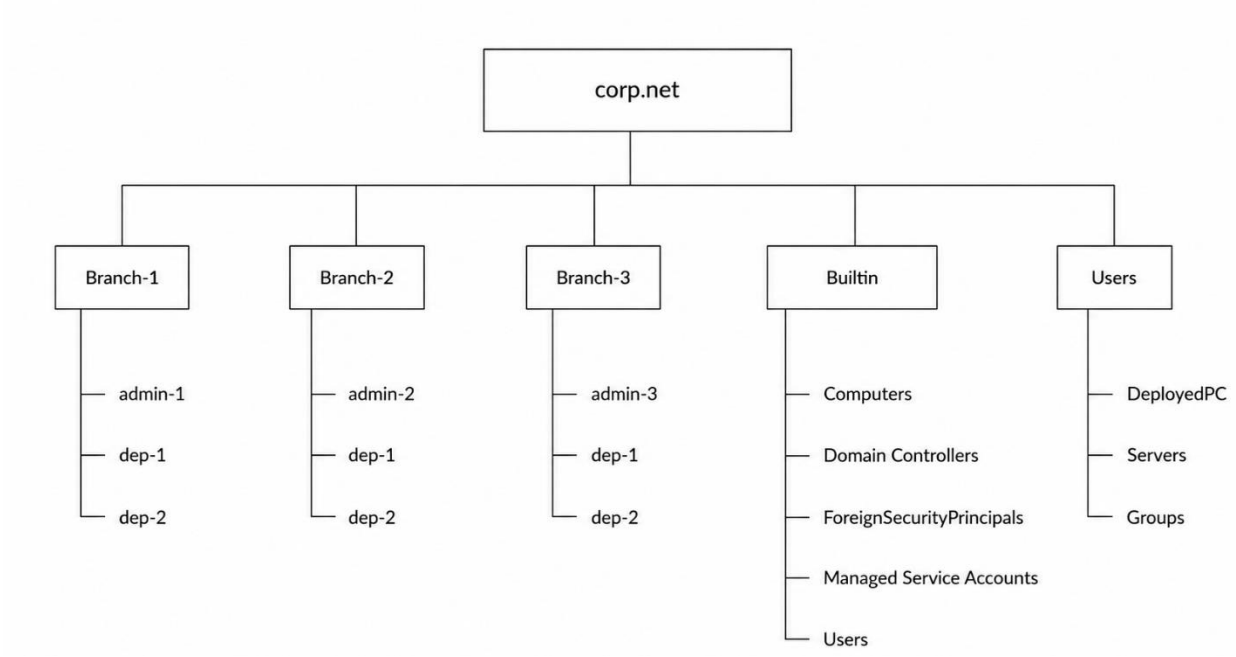


Рисунок – 3.10 Логічна схема організаційних одиниць домену

На рисунку 3.11 показано створення окремої організаційної одиниці, яка використовується для централізованого розміщення облікових записів робочих станцій, доданих після автоматизованого розгортання операційної системи.

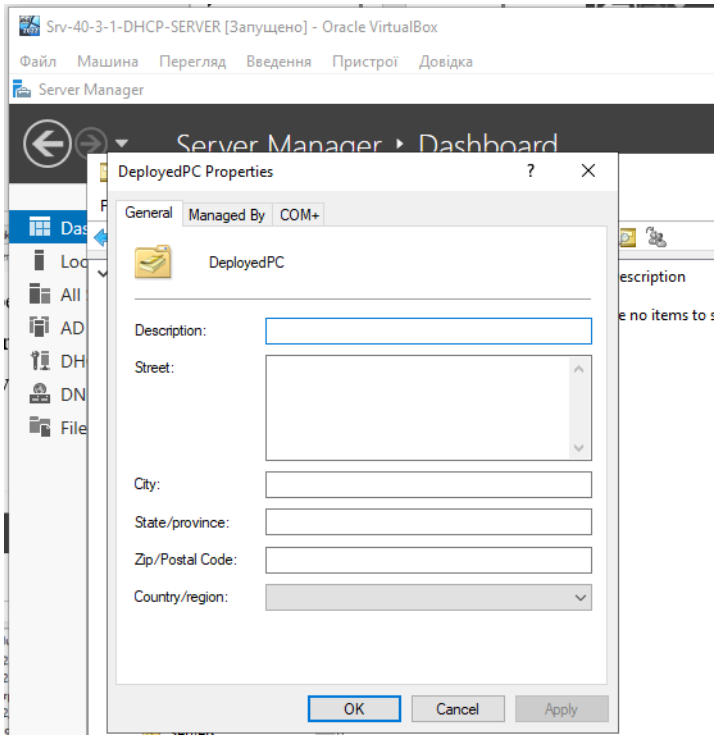


Рисунок 3.11 – Організаційна одиниця для розгорнутих робочих станцій.

На рисунку 3.12 показано процес створення групи користувачів. Для розмежування прав потрібно створити групу локальних адмінстраторів, яка буде стосуватися конкретного сервера розгортання.

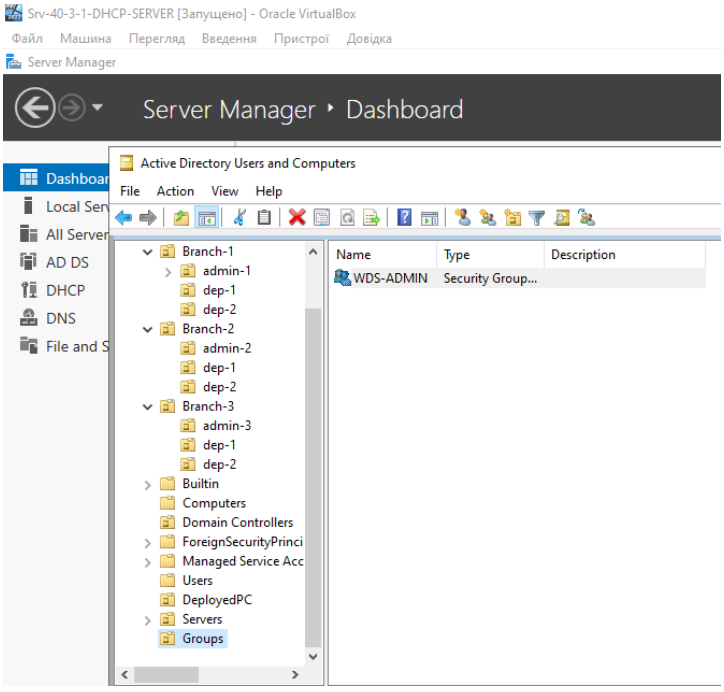


Рисунок – 3.12 Створена група локальних адміністраторів

На рисунку 3.13 створення політики, яка застосовується для сервера розгортання. Вона додає до групи локальних адміністраторів групу, яка була створена на рисунку 3.12.

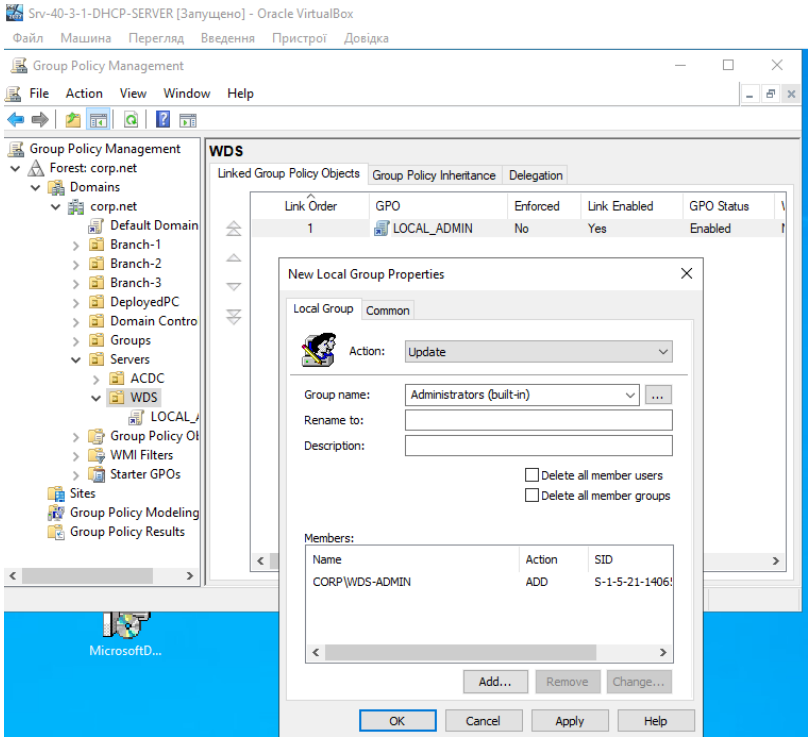


Рисунок – 3.13 Створена GPO для додавання групи локальних адміністраторів

З метою забезпечення автоматизованого приєднання робочих станцій до домену необхідно створити обліковий запис користувача та надати йому розширені адміністративні привілеї шляхом включення до групи Domain Admins, як показано на рисунку 3.14 та 3.15.

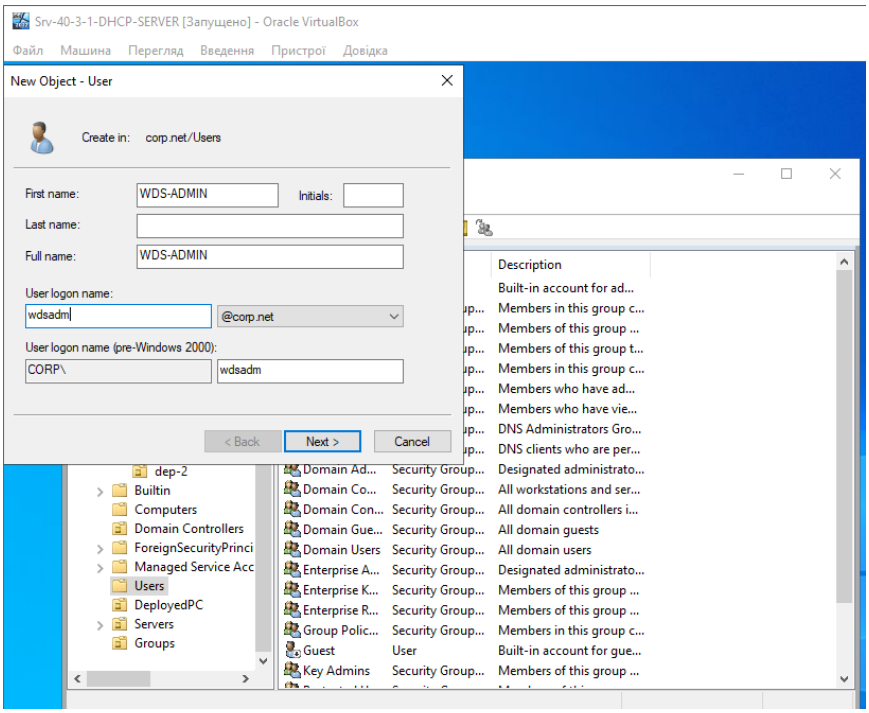


Рисунок – 3.14 Створення користувача з розширеними адміністративними привілеями

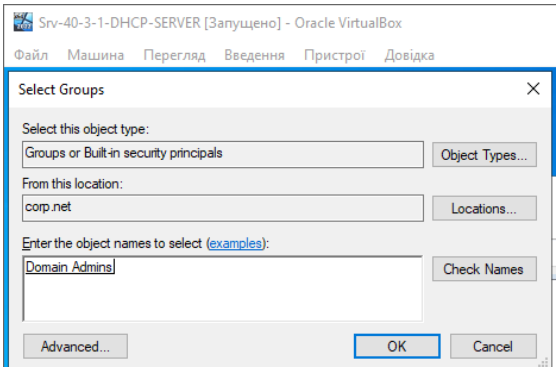


Рисунок – 3.15 Додавання користувача до групи Domain Admins

Крім того, для кожної гілки доменної структури передбачено створення базового набору групових політик (GPO), які застосовуються до користувачів, що працюють у межах відповідних організаційних одиниць, з метою уніфікації налаштувань та підвищення рівня безпеки, приклад створення таких політик показано на рисунках 3.16 – 3.20.

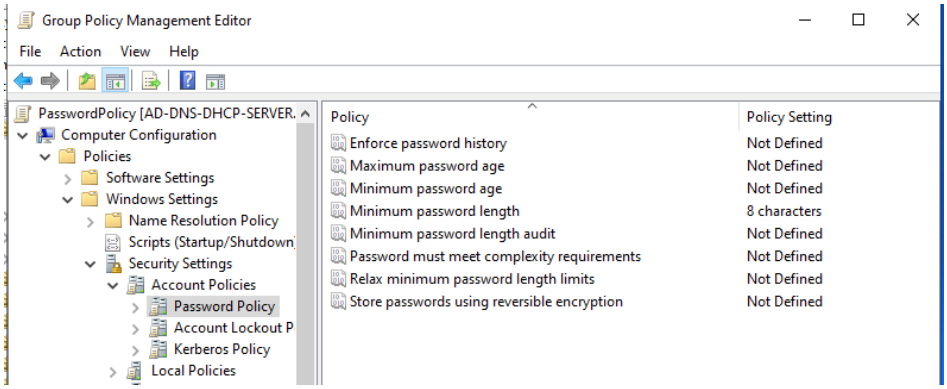


Рисунок – 3.16 Встановлення політики для користувачів на мінімальний пароль від 8 символів

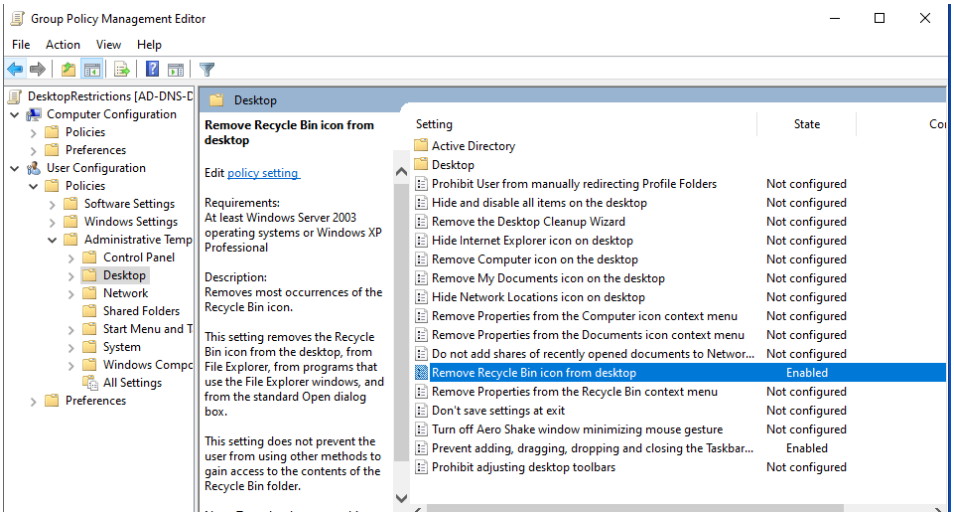


Рисунок – 3.17 Встановлення політики для користувачів, яка забороняє деякі дії з робочим столом

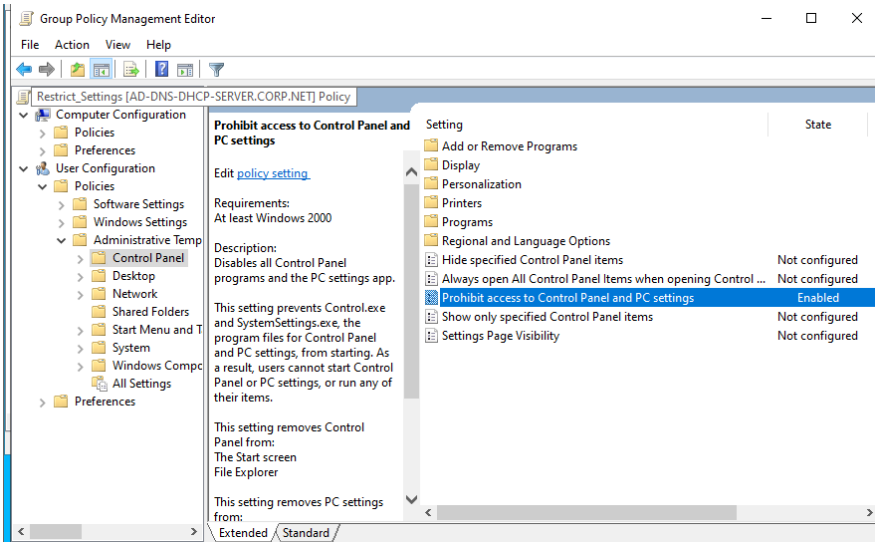


Рисунок – 3.18 Встановлення політики для користувачів, яка забороняє повністю відкривати панель керування та налаштування Windows

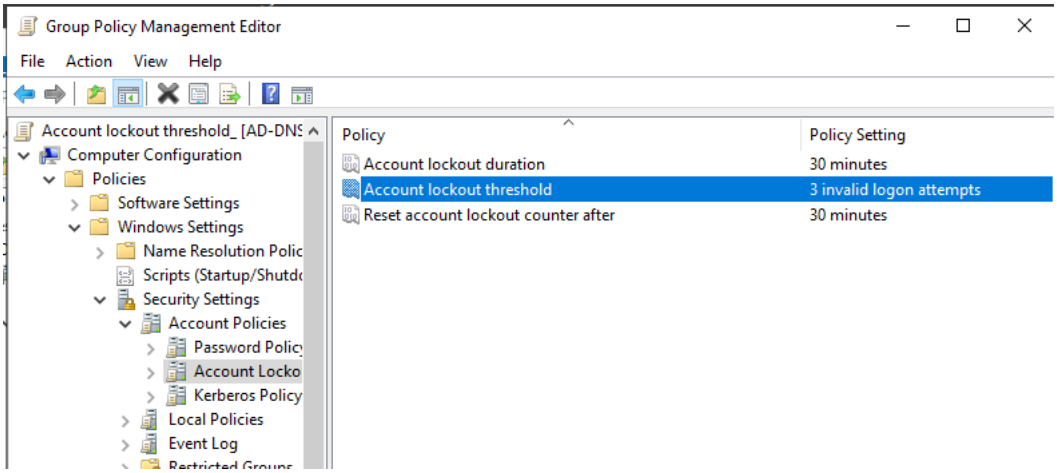


Рисунок – 3.19 Встановлення політики для користувачів, яка обмежує кількість невдалих спроб входу до системи

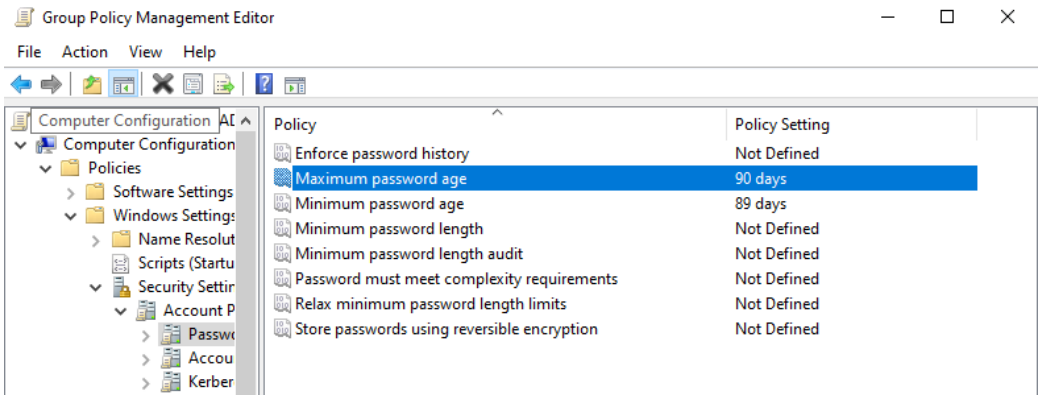


Рисунок – 3.20 Встановлення політики для користувачів, яка вимагає регулярну зміну пароля через визначений проміжок часу

3.3 Розгортання та базова конфігурація розподіленої архітектури

Для коректної роботи розподіленої архітектури на транспортному сервері виконується встановлення ролі Windows Deployment Services (WDS), яка забезпечує мережеве розгортання операційних систем.

Роль Windows Deployment Services (WDS) використовується для автоматизованого розгортання операційних систем у корпоративній мережі. Цю роль потрібно встановити на сервері, щоб передавати налаштовані образи через PXE.

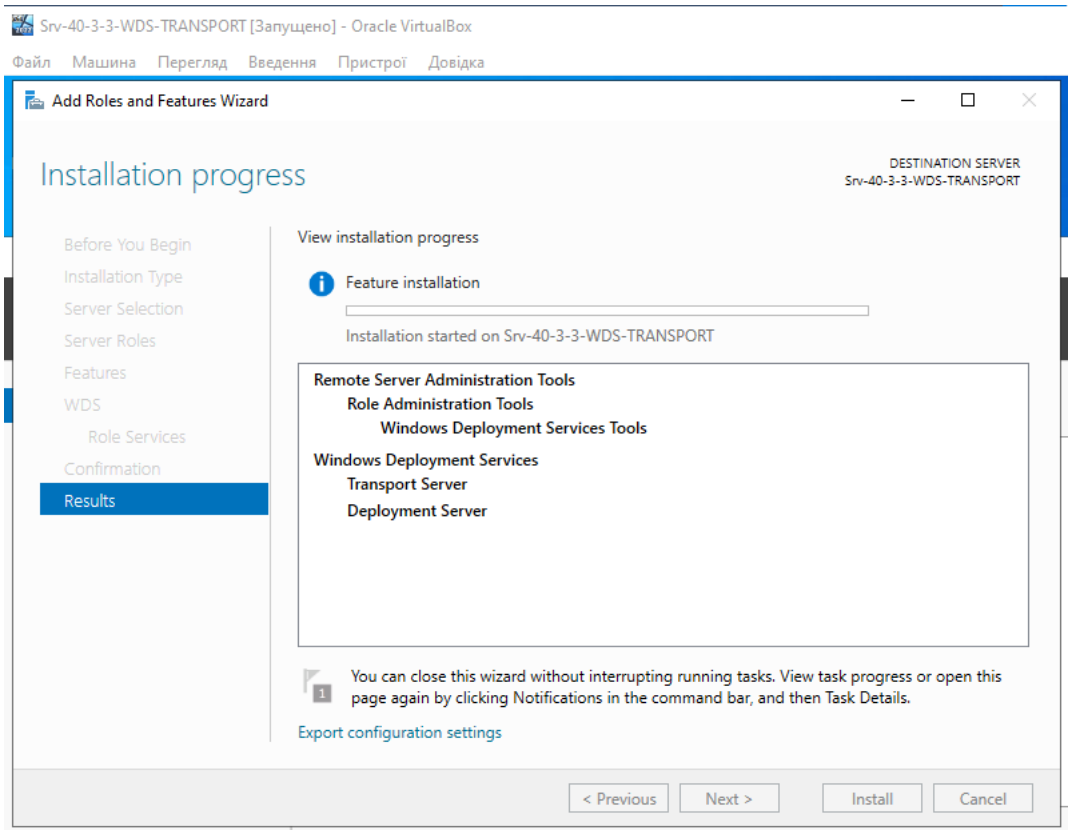


Рисунок – 3.21 Встановлення ролі Windows Deployment Services на транспортному сервері

На рисунку 3.22 зображено етап вибору режиму інсталяції сервера WDS у майстрі конфігурації. Для даної реалізації обрано варіант Standalone server. Цей режим дозволяє розгортати операційні системи без обов'язкової інтеграції з Active Directory, що спрощує структуру сервісу та забезпечує незалежність служби завантаження від доменної інфраструктури в конкретних сегментах мережі.

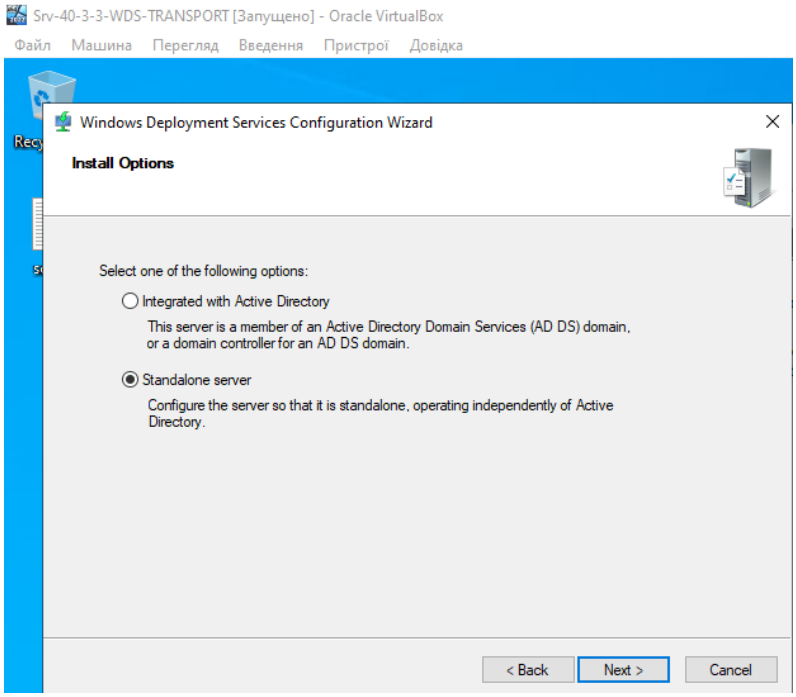


Рисунок – 3.22 Встановлення режиму Standalone server

На рисунку 3.23 показано етап вибору каталогу для зберігання файлів служби Windows Deployment Services у майстрі налаштування WDS. На цьому етапі визначається шлях до папки віддаленого встановлення, яка за замовчуванням має адресу C:\RemoteInstall. У зазначеному каталозі зберігатимуться завантажувальні та інсталяційні образи, файли PXE-завантажувача, а також службові інструменти керування.

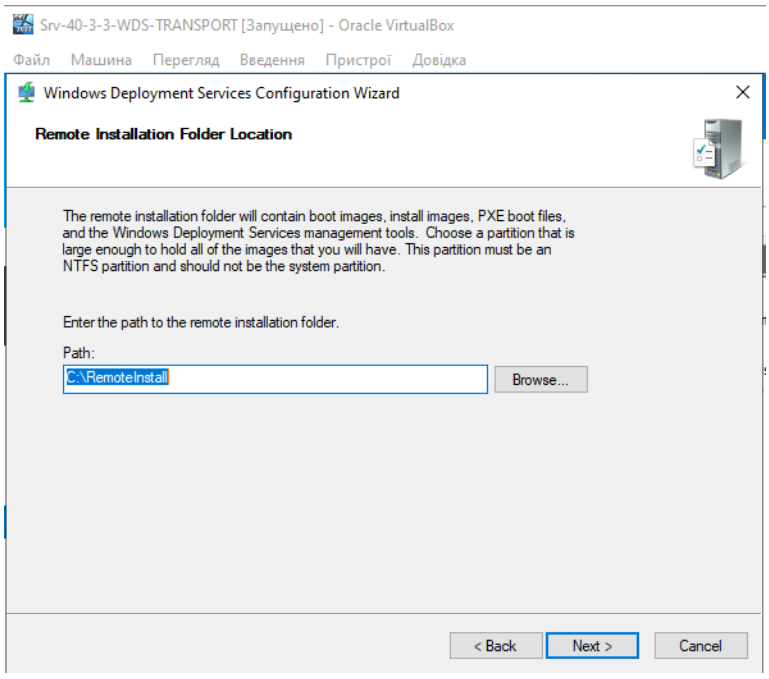


Рисунок – 3.23 Налаштування папки для зберігання файлів служби WDS

На рисунку 3.24 показано конфігурування початкових параметрів PXE-сервера. Обрано параметр Respond to all client computers (known and unknown), що дозволяє серверу відповідати на запити завантаження від будь-яких клієнтських комп'ютерів у мережі. Таке налаштування є оптимальним для масового розгортання робочих станцій «з нуля», оскільки воно автоматизує процес початку встановлення ОС без необхідності попередньої ручної реєстрації кожного пристрою.

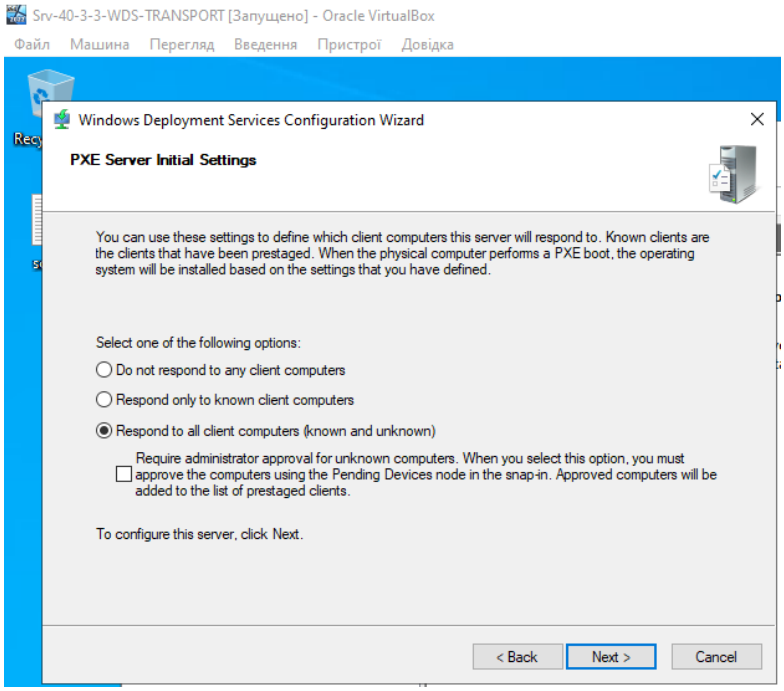


Рисунок – 3.24 Налаштування опції Respond to all client computers

Після виконання всіх необхідних налаштувань транспортний сервер переходить у робочий стан і готовий до завантаження та розповсюдження образів операційних систем.

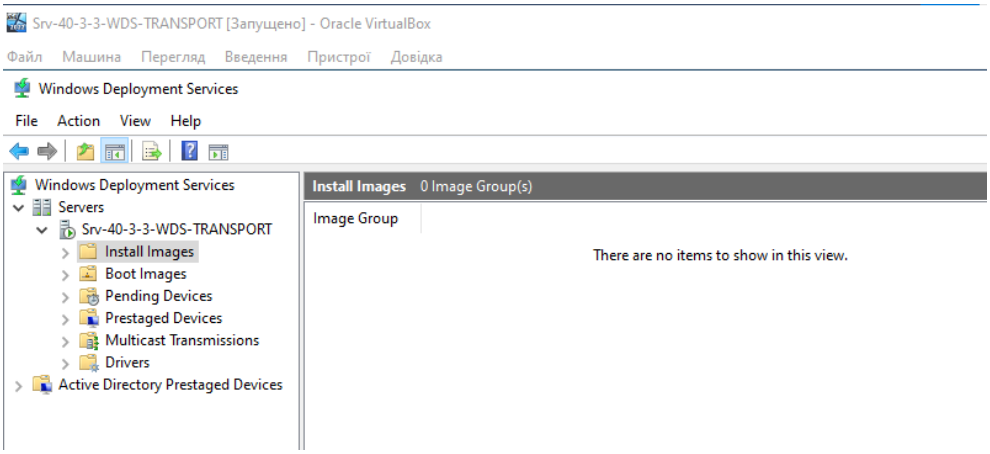


Рисунок – 3.25 Налаштована служба WDS

Наступним кроком є розгортання програмного забезпечення для сервера розгортання, яке забезпечує автоматизоване створення, налаштування та розповсюдження образів операційних систем у мережі. Після цього можна встановити Microsoft Deployment Toolkit, як показано на рисунку 3.26

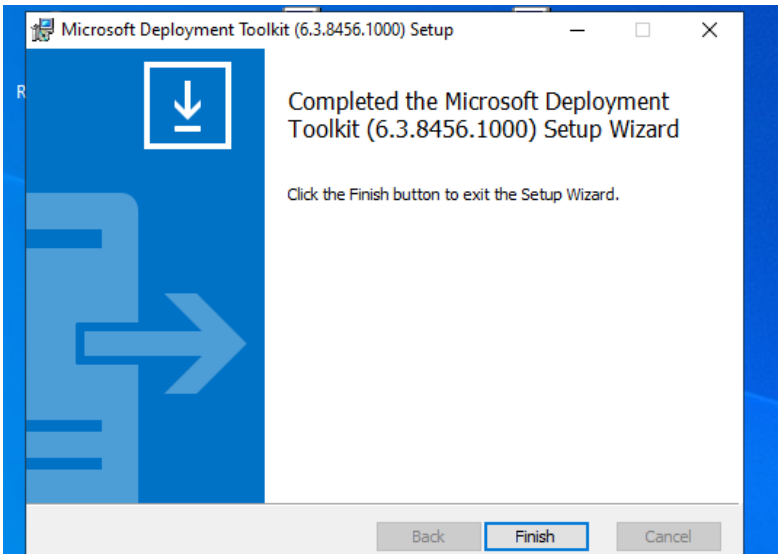


Рисунок – 3.26 Успішне встановлення Microsoft Deployment Toolkit.

Для встановлення та подальшої роботи інструментів підготовки та розгортання операційних систем необхідно інсталювати Windows Assessment and Deployment Kit (Windows ADK), а також додаток Windows Preinstallation Environment (Windows PE). На рисунку 3.27 наведено процес встановлення Windows ADK, де користувач обирає необхідні компоненти, зокрема Deployment Tools, які забезпечують роботу з образами операційних систем.

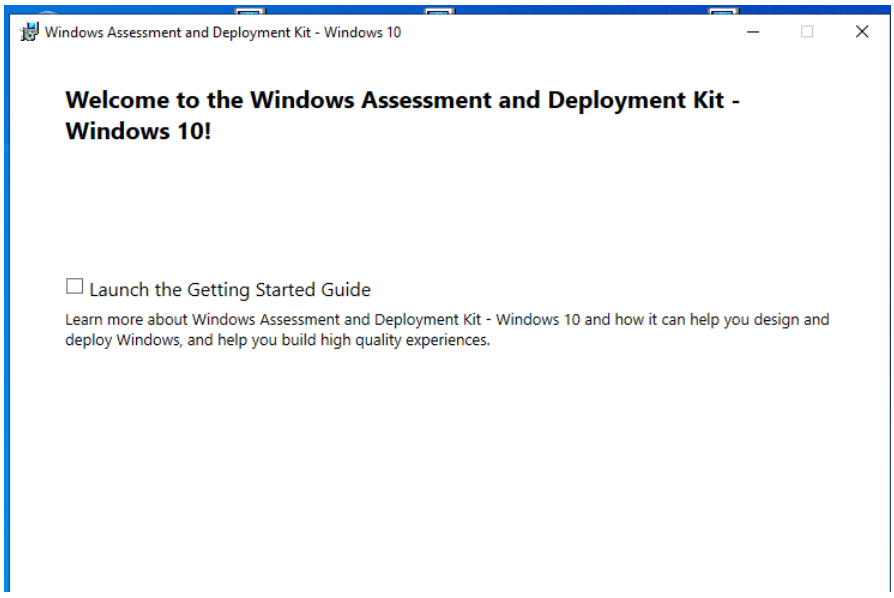


Рисунок – 3.27 Успішне встановлення Windows Assessment and Deployment Kit

На рисунку 3.28 показано етап інсталяції Windows PE Add-on, що розширює функціонал ADK та дозволяє створювати завантажувальні середовища для мережевого розгортання ОС.

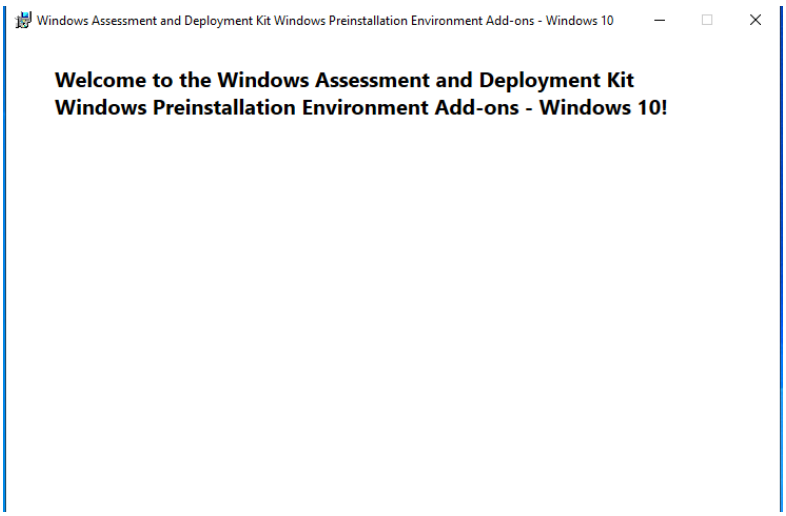


Рисунок – 3.28 Успішне встановлення Windows PE Add-on для Windows ADK

Після завершення встановлення зазначених компонентів формується повноцінне середовище для підготовки та кастомізації завантажувальних образів, які використовуються у процесі автоматизованого розгортання операційних систем.

Наступним етапом було створення та ініціалізація робочого каталогу C:\DeploymentShare у середовищі Microsoft Deployment Toolkit (MDT), що показано на рисунку 3.29. Дана папка є центральним мережевим репозиторієм (Deployment Share), у якому зберігається повний набір компонентів для автоматизованого розгортання операційних систем, а саме: образи операційних систем, драйвери, пакети оновлень, сценарії встановлення, додаткові утиліти та конфігураційні файли, необхідні для виконання процесу розгортання.

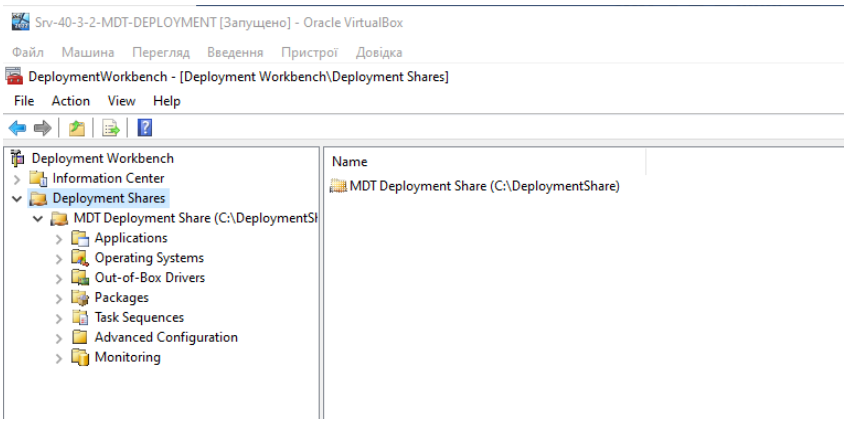


Рисунок – 3.29 Створення MDT Deployment Share

3.4 Налаштування автоматизованого розгортання ОС із використанням Task Sequence, драйверів та прикладного ПЗ

3.4.1 Створення та налаштування Task Sequence

Перед створенням Task Sequence необхідно імпортувати образ операційної системи в середовище Microsoft Deployment Toolkit. Цей образ використовується як основа для подальшого розгортання, оскільки саме на його базі налаштовуються параметри встановлення ОС, а також додаються драйвери та прикладне програмне забезпечення.

На рисунку 3.30 показано процес імпорту інсталяційних образів операційних систем у середовище Microsoft Deployment Toolkit (Deployment Workbench). У розділі Operating Systems додано декілька редакцій Windows 10 (Home, Pro, Education та інші), які надалі можуть бути використані під час створення та налаштування Task Sequence.

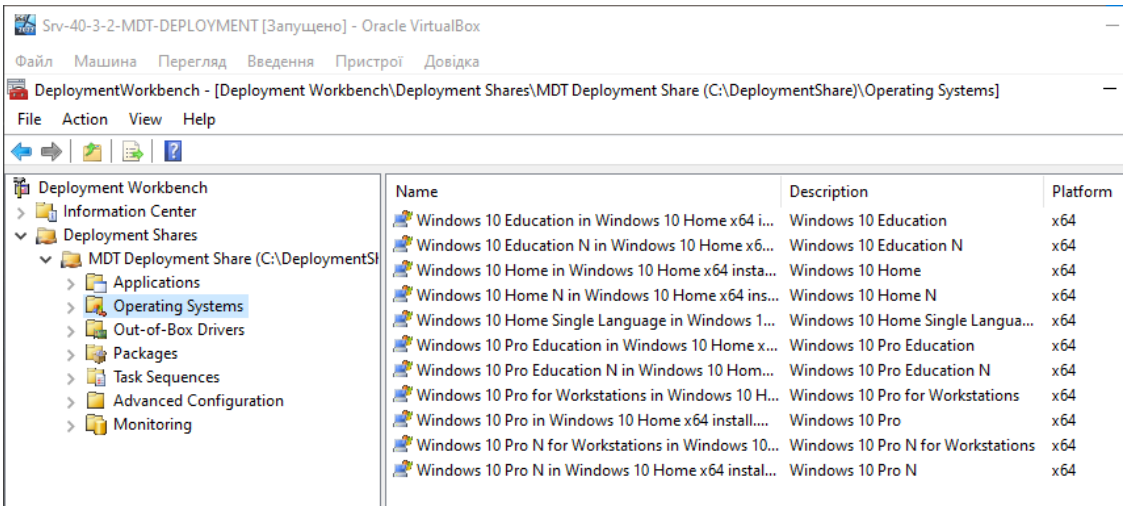


Рисунок – 3.30 Імпортований образ ОС з відповідними редакціями

Наступним кроком є створення Task Sequence – послідовності завдань, яка містить набір дій для автоматизованого розгортання операційної системи. На рисунку 3.31 показано процес створення та налаштування нової послідовності завдань у Microsoft Deployment Toolkit.

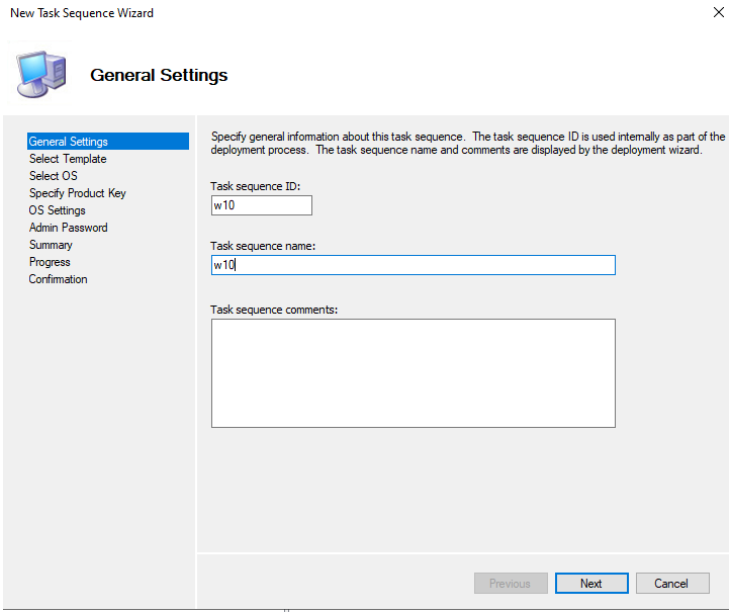


Рисунок – 3.31 Створення нового Task Sequences та його іменування

На рисунку 3.32 показано етап вибору редакції операційної системи для створення Task Sequence. Для подальшого розгортання було обрано редакцію Windows 10 Pro, оскільки вона підтримує підключення до домену, що є необхідним для роботи в доменній інфраструктурі організації.

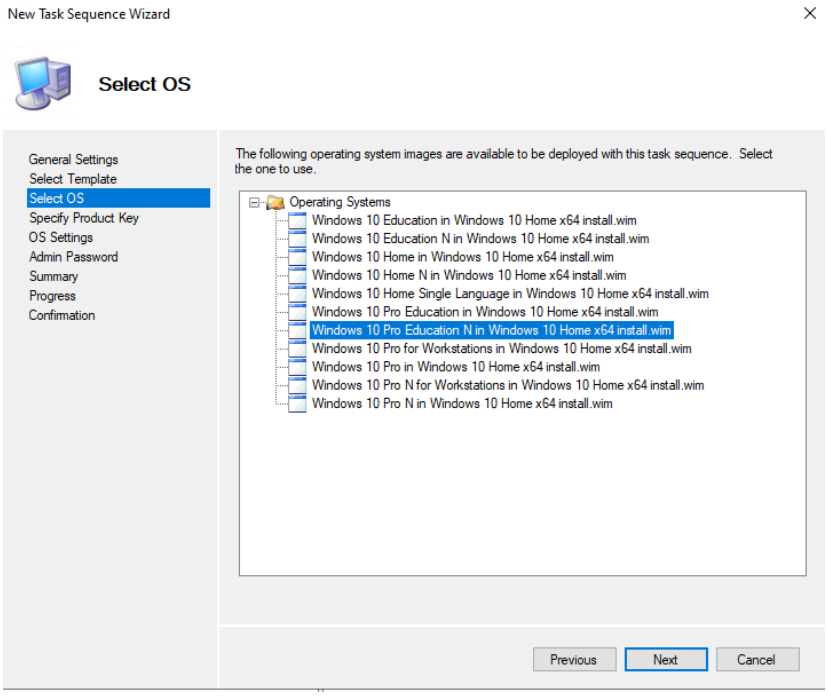


Рисунок – 3.32 Вибір редакції операційної системи для розгортання

На рисунку 3.33 показано етап налаштування параметрів операційної системи під час створення Task Sequence у MDT. У даному вікні задаються основні

відомості, які будуть автоматично застосовані після встановлення Windows, зокрема ім’я користувача, назва організації та домашня сторінка браузера.

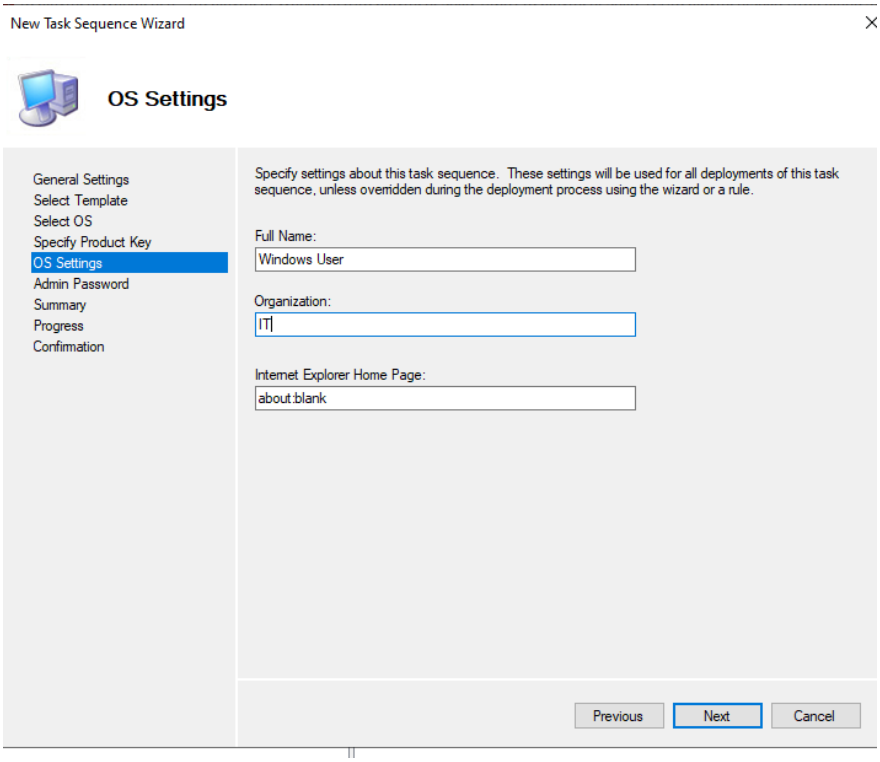


Рисунок – 3.33 Налаштування параметрів операційної системи в Task Sequence

Після всіх виконанних налаштувань було успішно створено послідовність завдань, що показано на рисунку 3.34.

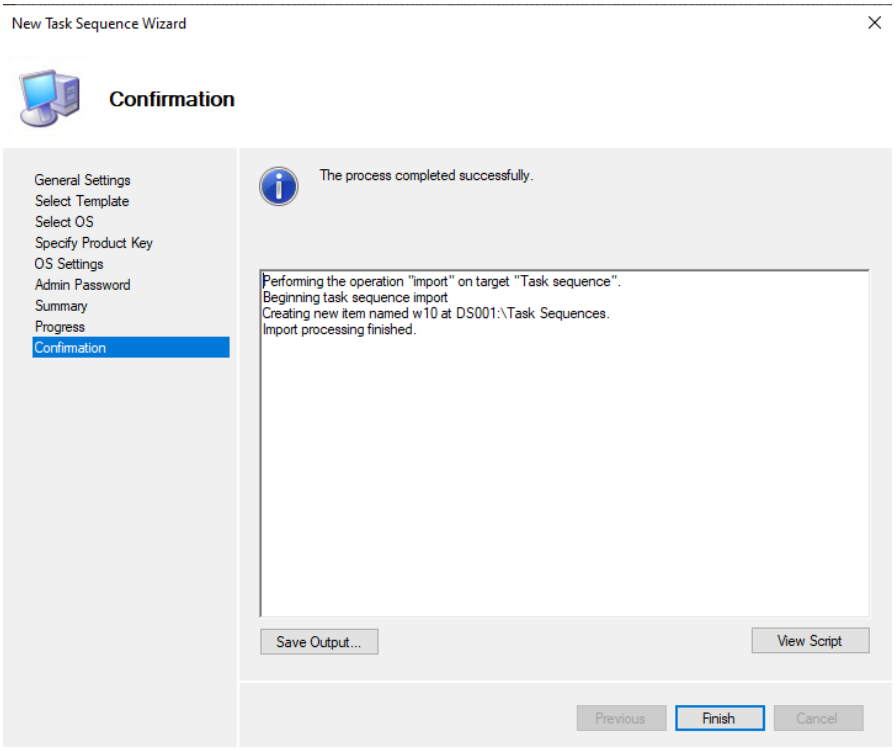


Рисунок – 3.34 Успішний створенний Task Sequences

На рисунку 3.35 показано вікно властивостей Deployment Share у Microsoft Deployment Toolkit. У даному вікні налаштовуються основні параметри середовища розгортання, зокрема мережевий та локальний шляхи до файлів встановлення операційних систем. Також активовано підтримку архітектури x64 для роботи із 64-бітними операційними системами.

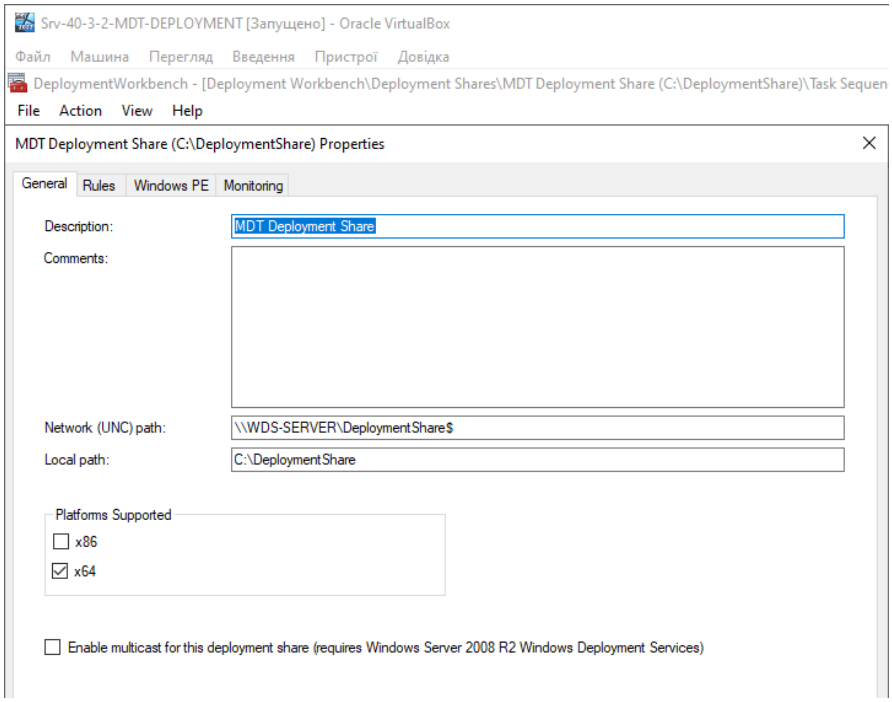


Рисунок – 3.35 Активація підтримки 64-бітної архітектури

Файл CustomSettings.ini використовується як один із ключових компонентів повної автоматизації процесу розгортання операційної системи Windows за допомогою Microsoft Deployment Toolkit (MDT). Його основне призначення полягає у централізованому визначенні параметрів встановлення, що дозволяє мінімізувати участь адміністратора на етапі інсталяції та забезпечити стандартизоване розгортання операційних систем у межах організації.

У цьому файлі задаються основні налаштування середовища інсталяції, зокрема мова інтерфейсу системи, регіональні параметри, формат часу та дати, часовий пояс, а також параметри клавіатури. Окрім цього, у CustomSettings.ini визначаються правила автоматичного приєднання комп'ютера до домену, що дозволяє одразу інтегрувати нові робочі станції в корпоративну інфраструктуру без додаткових ручних дій адміністратора.

Після завершення процесу інсталяції система автоматично виконує приєднання до домену corp.net, після чого комп'ютер розміщується у заздалегідь визначеній організаційній одиниці (OU) в Active Directory. Це забезпечує автоматичне застосування необхідних групових політик (GPO), політик безпеки, а також інших корпоративних налаштувань, що відповідають вимогам організації.

Використання CustomSettings.ini є необхідним етапом при побудові повністю автоматизованого середовища розгортання, оскільки дозволяє суттєво спростити процес встановлення операційних систем, зменшити кількість можливих помилок, які можуть виникати при ручній конфігурації, та забезпечити уніфіковану конфігурацію всіх робочих станцій.

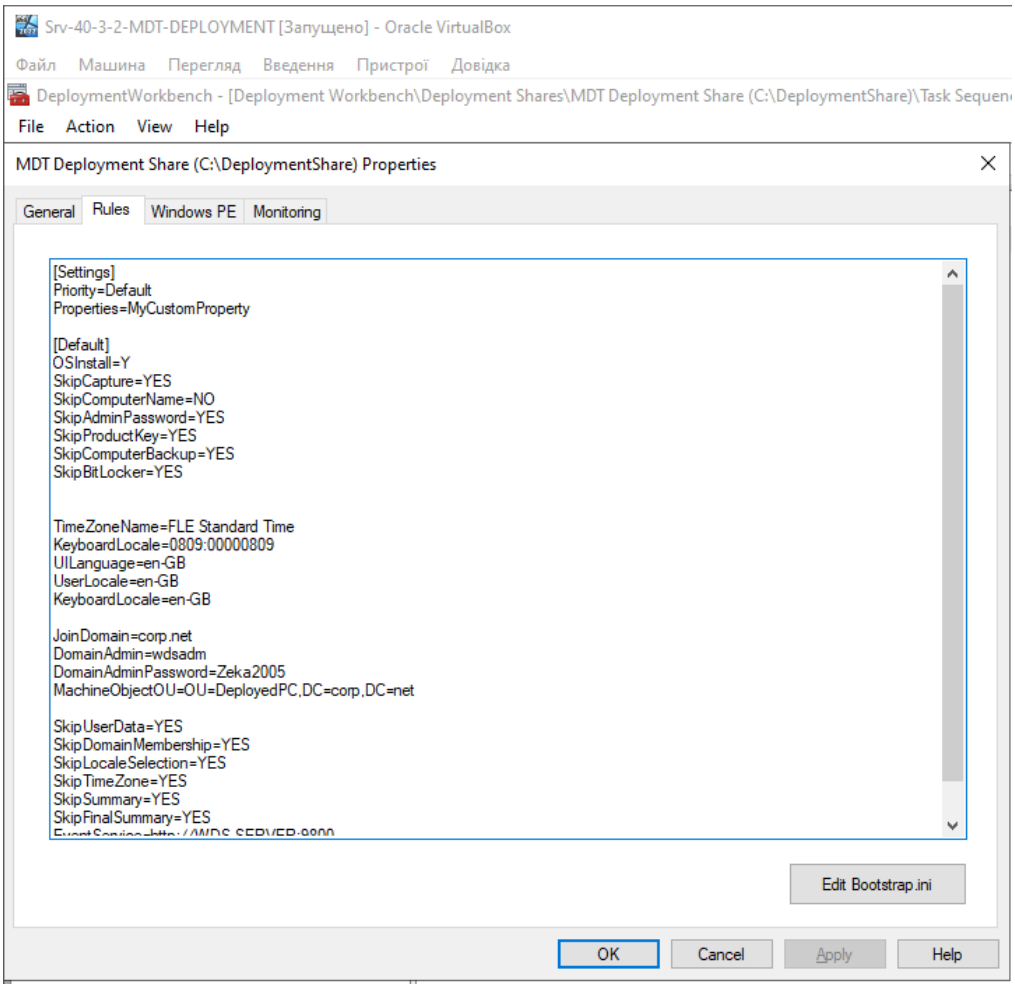


Рисунок – 3.36 Вміст файлу CustomSettings.ini

На рисунку 3.37 показано вікно налаштування моніторингу спільного ресурсу розгортання у вкладці Monitoring, яке дозволяє відстежувати процес інсталяції операційної системи на клієнтських машинах у реальному часі.

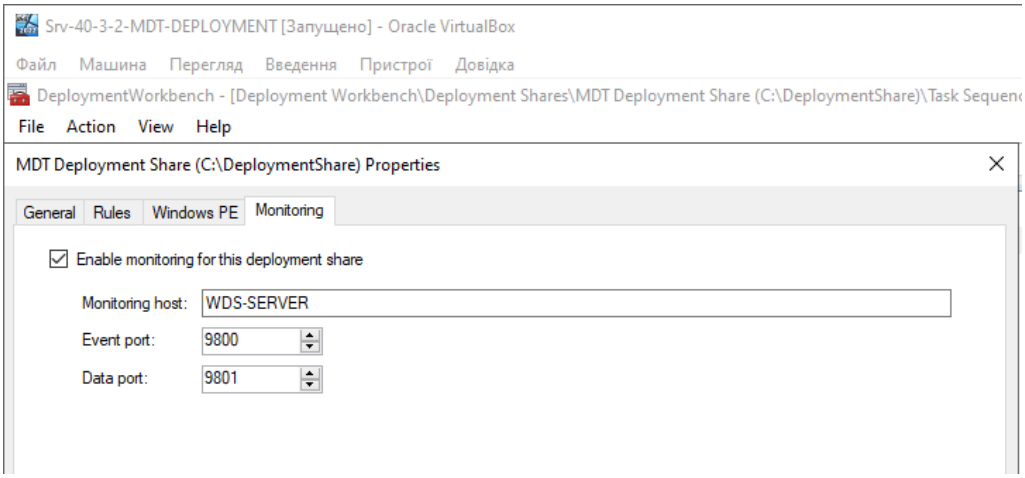


Рисунок – 3.37 Ввімкнення вбудованого моніторингу

3.4.2 Інтеграція драйверів у середовище розгортання

У процесі налаштування автоматизованого розгортання операційної системи важливим етапом є інтеграція драйверів для забезпечення коректної роботи апаратного забезпечення. Проте в умовах використання віртуалізованого середовища, зокрема VirtualBox, застосування типових драйверів обладнання є обмеженим, оскільки віртуальні машини використовують уніфіковані (емуляційні) апаратні компоненти, що не потребують встановлення специфічних драйверів у середовищі розгортання.

У зв’язку з цим, для демонстрації можливостей інтеграції драйверів у процесі розгортання доцільно використовувати драйвери периферійних пристроїв, наприклад, драйвери принтерів. Такий підхід дозволяє перевірити коректність додавання, обробки та встановлення драйверів у межах Task Sequence без залежності від фізичного апаратного забезпечення.

На рисунку 3.38 показано процес імпорту драйверів до MDT за допомогою Import Driver Wizard, який забезпечує сумісність ОС із апаратним забезпеченням робочих станцій.

У вікні Specify Directory вказується шлях до каталогу з драйверами, після чого система автоматично знаходить файли .inf та додає їх до сховища розгортання. Процес дозволяє централізовано керувати драйверами, що використовуються під час інсталяції Windows.

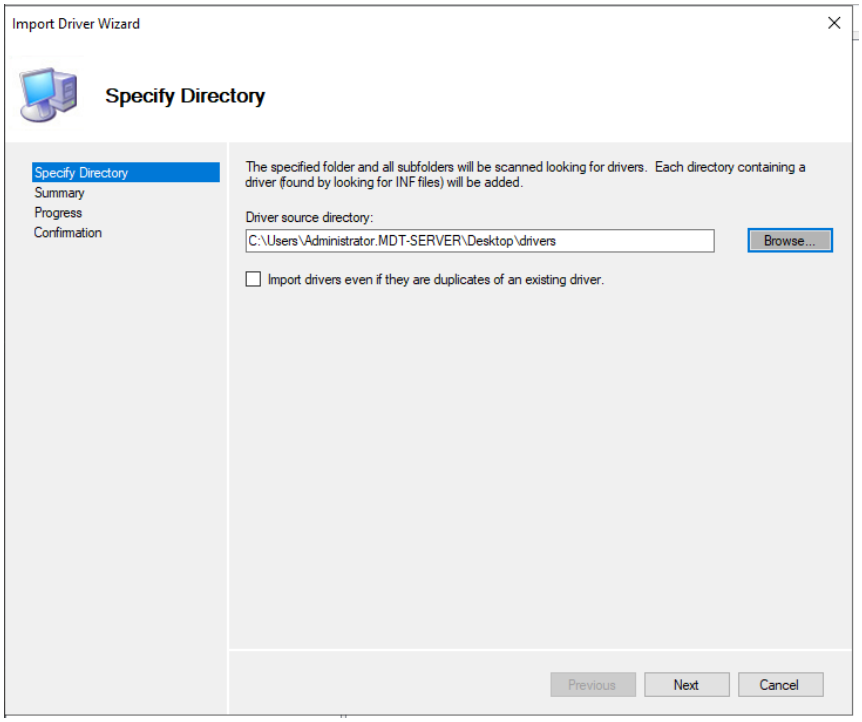


Рисунок – 3.38 Вибір шляху з драйверами

На рисунку 3.39 показано вікно редагування Task Sequence, де виконується налаштування етапів автоматизованого встановлення операційної системи. У правій частині вказано параметри цього кроку, зокрема вибір профілю драйверів та режим їх автоматичної інсталяції, що забезпечує контрольований процес розгортання системи.

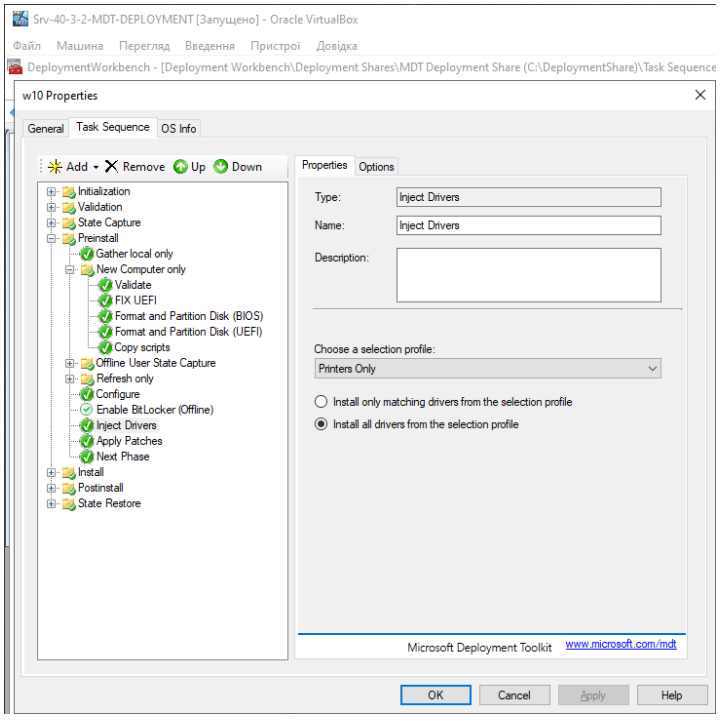


Рисунок – 3.39 Вибір профілю з драйверами для принтера

3.4.3 Додавання та конфігурація прикладного програмного забезпечення

Після налаштування базових компонентів середовища розгортання, таких як параметри операційної системи, драйвери та послідовність завдань, наступним етапом є підготовка та конфігурація прикладного програмного забезпечення. Саме на цьому рівні формується повноцінний набір інструментів, необхідних для роботи кінцевого користувача.

У даному підрозділі буде розглянуто процес додавання програм до середовища розгортання та подальші налаштування, які забезпечують їх автоматичну інсталяцію під час встановлення операційної системи. Це дозволяє створити єдине стандартизоване робоче середовище та повністю автоматизувати процес підготовки робочих станцій.

Як видно на рисунках 3.40 – 3.42, за допомогою команди з ключем /? відкривається довідкова інформація інсталятора, що дозволяє визначити доступні параметри запуску. Зокрема, використано ключі /quiet для повністю автоматичного встановлення та /passive для режиму з відображенням прогресу без участі користувача.

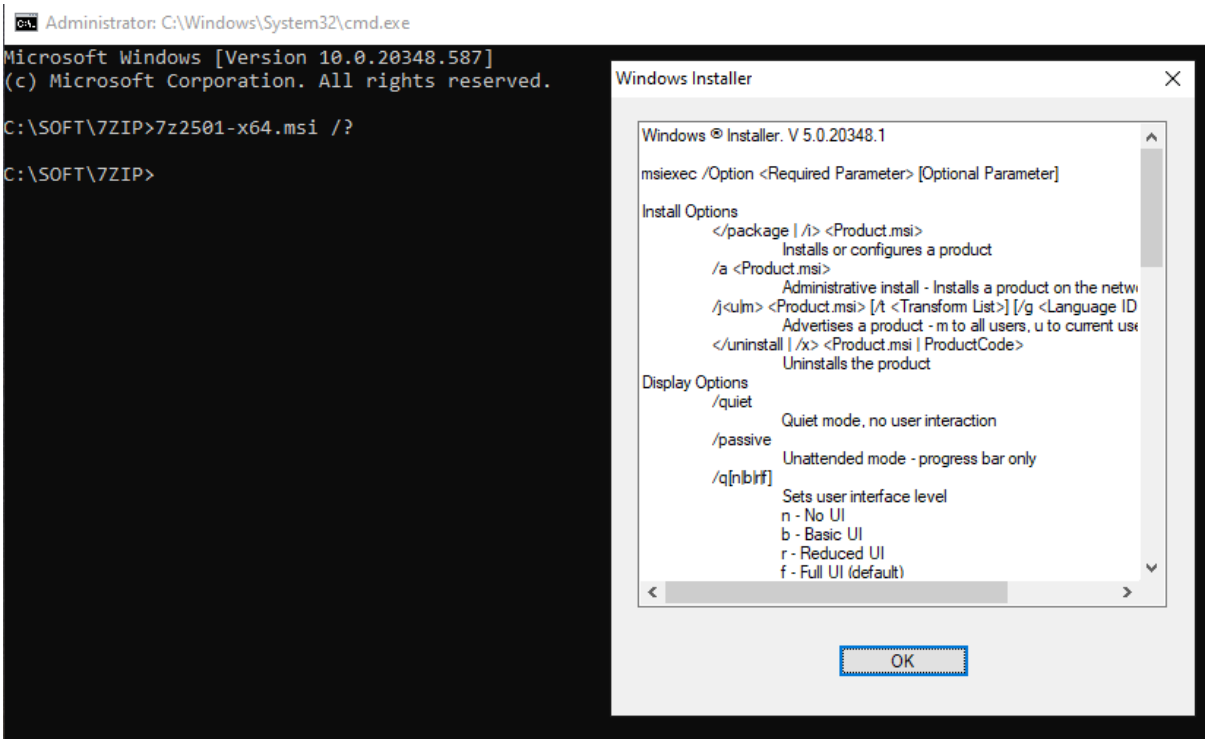


Рисунок – 3.40 Визначення параметрів «тихого» встановлення 7-Zip через командний рядок Windows

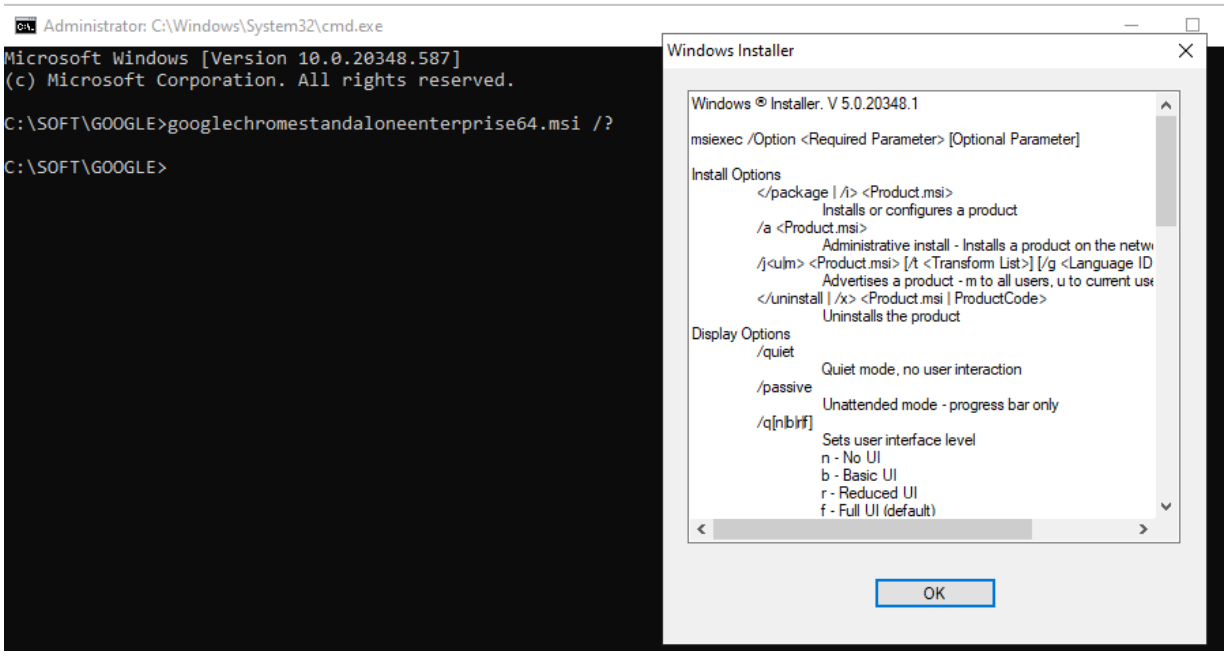


Рисунок – 3.41 Визначення параметрів «тихого» встановлення Google Chrome через командний рядок Windows

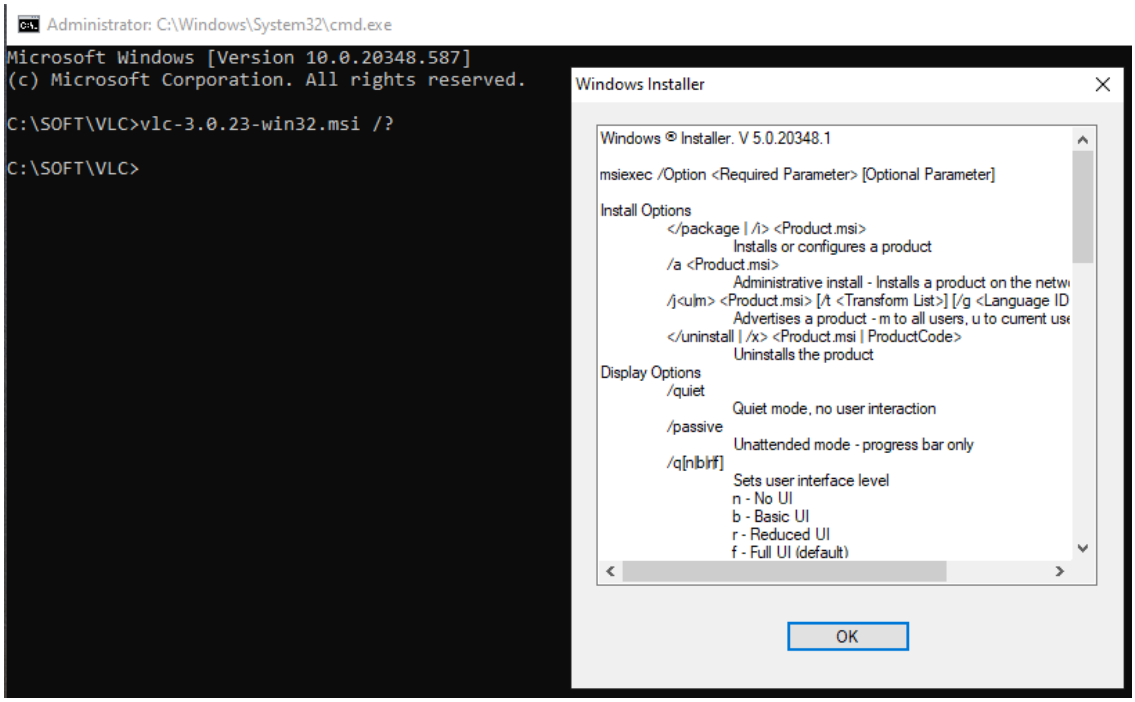


Рисунок – 3.42 Визначення параметрів «тихого» встановлення VLC media player через командний рядок Windows

На рисунку 3.43 показано консоль Deployment Workbench з налаштованим переліком прикладного програмного забезпечення. У розділі Applications сформовано базу програм, які автоматично встановлюються під час розгортання операційної системи.

Як видно на рисунку 3.43, до сховища додано основні застосунки, зокрема 7-Zip, Google Chrome та VLC media player. Ліва частина вікна відображає структуру категорій для зручної організації програм, а права панель містить список доступних інсталяційних пакетів.

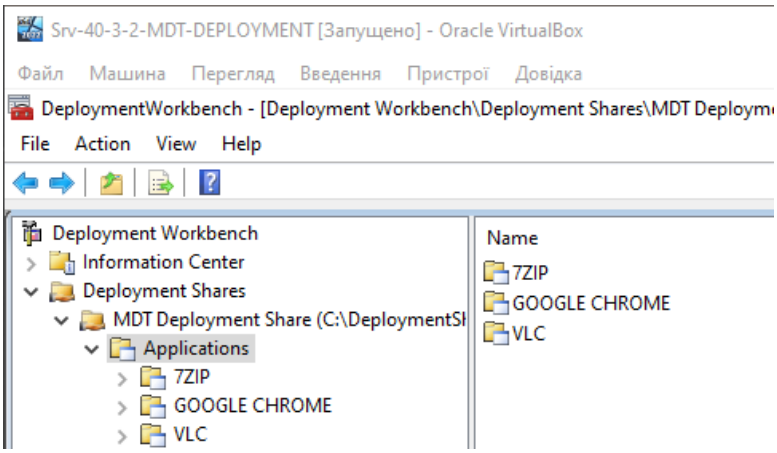


Рисунок – 3.43 Вибір прикладних програм для автоматизованого встановлення

3.4.4 Автоматизація процесу розгортання операційної системи на основі скриптів

Після налаштування основних компонентів середовища розгортання, інтеграції прикладного програмного забезпечення та визначення параметрів встановлення операційної системи виникає необхідність автоматизації завершальних етапів інсталяції. Саме на цьому рівні забезпечується виконання додаткових системних налаштувань, підготовка робочого середовища користувача та фіксація результатів розгортання.

У даному підрозділі розглядається використання скриптів для автоматизації процесу розгортання операційної системи, що дозволяє виконувати постінсталяційні дії без участі адміністратора та забезпечує повністю керований і стандартизований процес налаштування робочих станцій.

На рисунку 3.44 показано процес автоматизації додаткових налаштувань системи за допомогою сценаріїв PowerShell, що дозволяє розширити стандартні можливості MDT і виконувати спеціалізовані завдання під час розгортання операційної системи.

Як видно на рисунку 3.45 показано каталог Scripts, у якому збережено набір службових сценаріїв: CreateUser.ps1, SystemInfo.ps1, SecurityConfig.ps1 та

DeploymentLog.ps1. Вони використовуються для збору системної інформації, налаштування параметрів безпеки та ведення журналів процесу розгортання. Повний лістинг скриптів наведено в додатку Г.

Інтеграція цих скриптів у MDT забезпечує автоматизацію постінсталяційних дій, зменшує обсяг ручної роботи та гарантує уніфіковану конфігурацію всіх робочих станцій.

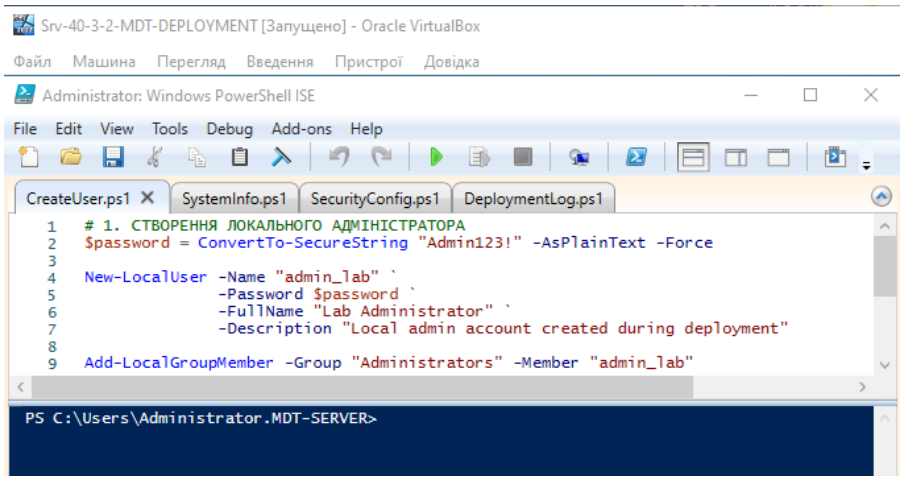


Рисунок – 3.44 Створення скриптів у середовищі PowerShell ISE

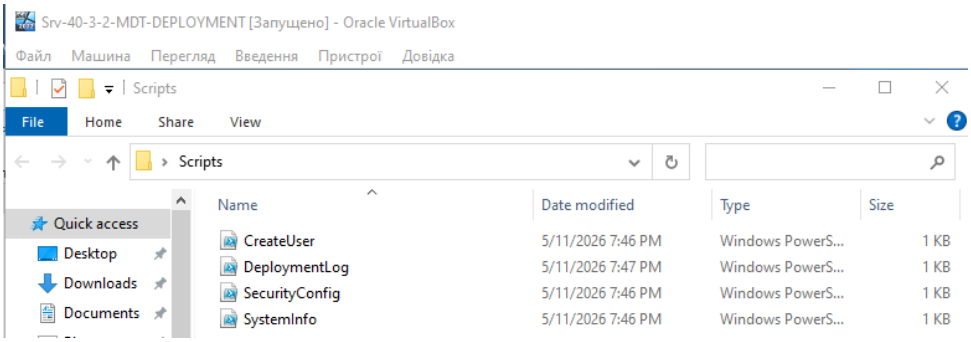


Рисунок – 3.45 Тека з розміщенними скриптами

Додавання скриптів у послідовність завдань (Task Sequence) є завершальним етапом налаштування автоматизованого розгортання операційної системи. На цьому етапі PowerShell-скрипти інтегруються у відповідні стадії встановлення ОС, що дозволяє виконувати додаткові системні налаштування без участі адміністратора.

Як показано на рисунку 3.46, скрипти додаються до розділу State Restore, який відповідає за постінсталяційну конфігурацію системи. Саме тут виконуються такі завдання, як створення локальних користувачів, налаштування параметрів безпеки, збір системної інформації та ведення журналів розгортання.

Кожен скрипт підключається через дію Run PowerShell Script, де вказується шлях до відповідного файлу в каталозі Scripts. Такий підхід забезпечує централізоване керування процесом виконання сценаріїв та їх послідовність відповідно до логіки розгортання.

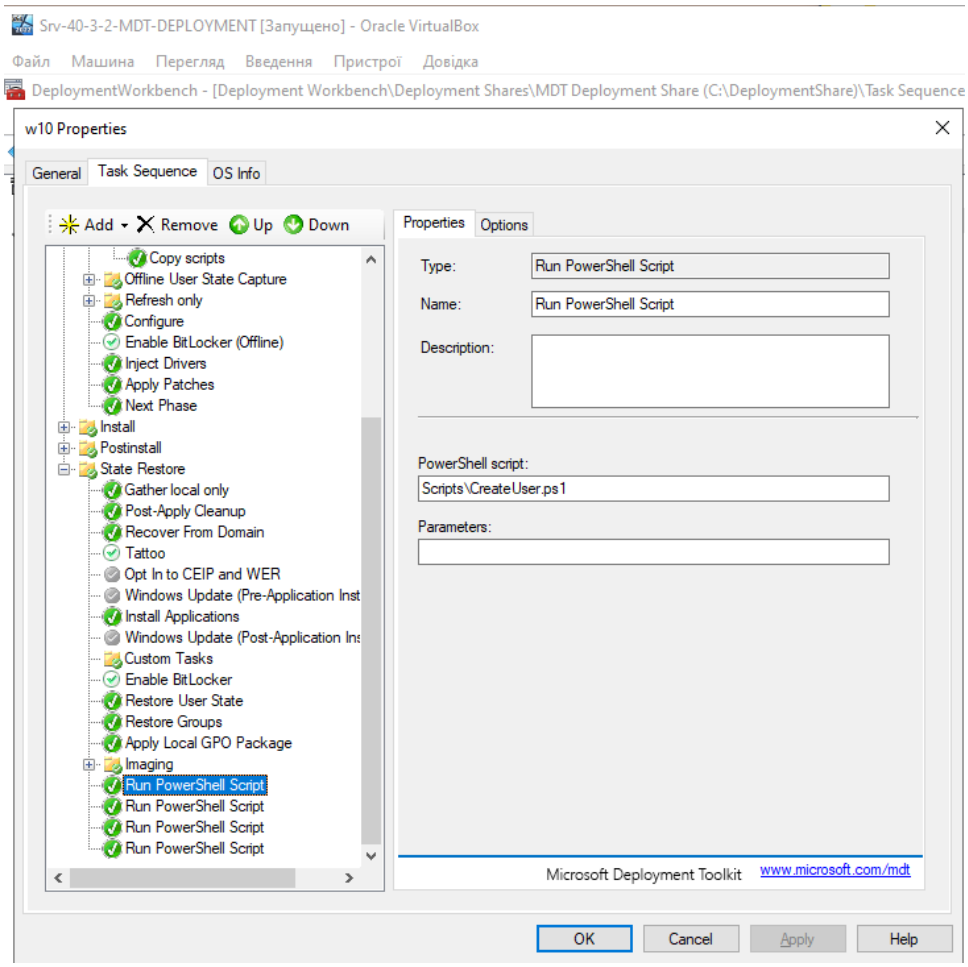


Рисунок – 3.46 Додані скрипти в Task Sequence

3.5 Тестування системи, моніторинг її роботи та аналіз метрик ефективності

3.5.1 Тестування розгортання системи

Перед початком процесу розгортання операційної системи було створено віртуальну машину у середовищі VirtualBox (рисунок 3.47). Під час налаштування визначено обсяг оперативної пам’яті та кількість логічних процесорів відповідно до вимог системи.

Крім того, у параметрах віртуальної машини було увімкнено можливість завантаження з мережі (PXE), що дозволяє виконувати автоматизоване розгортання операційної системи з сервера.

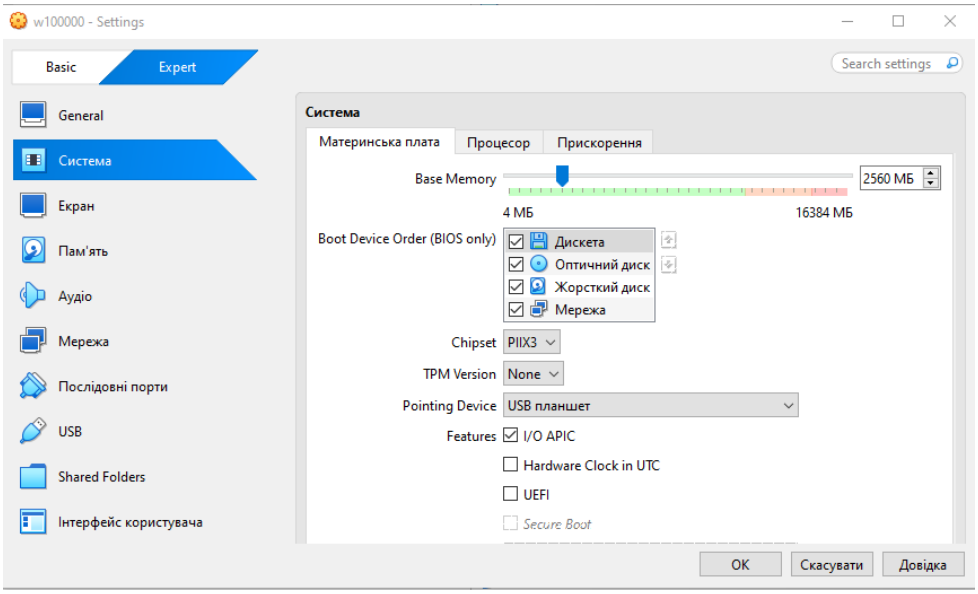


Рисунок – 3.47 Ввімкнення опції мережевого завантаження

Після запуску створеної віртуальної робочої станції було ініційовано процес мережевого завантаження (рисунок 3.48). На початковому етапі клієнтська машина автоматично отримала динамічну IP-адресу від DHCP-сервера, що підтверджує коректність налаштування мережевої інфраструктури та почалось завантаження образу.

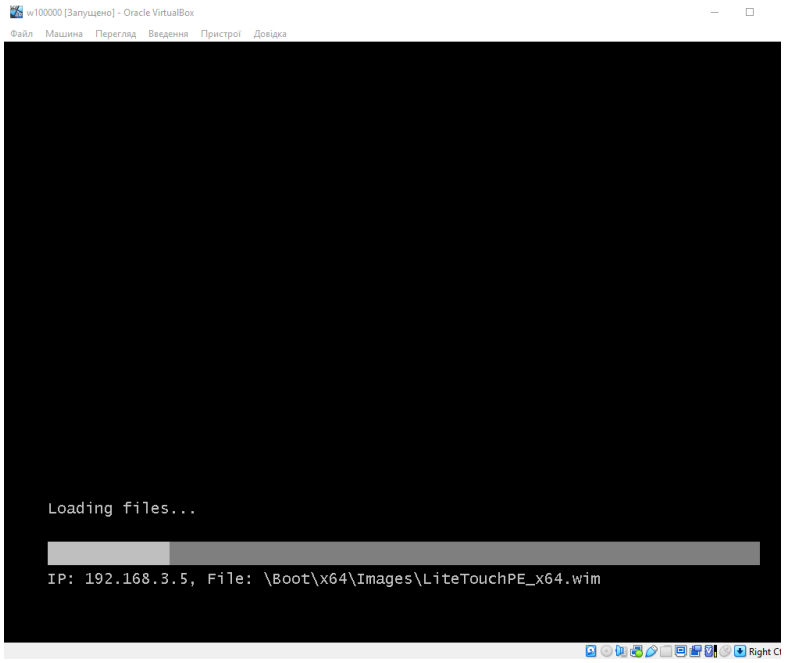


Рисунок – 3.48 Початок завантаження ОС по мережі

На цьому етапі необхідно ввести облікові дані адміністратора. Було використано обліковий запис користувача, який має права локального адміністратора на сервері.

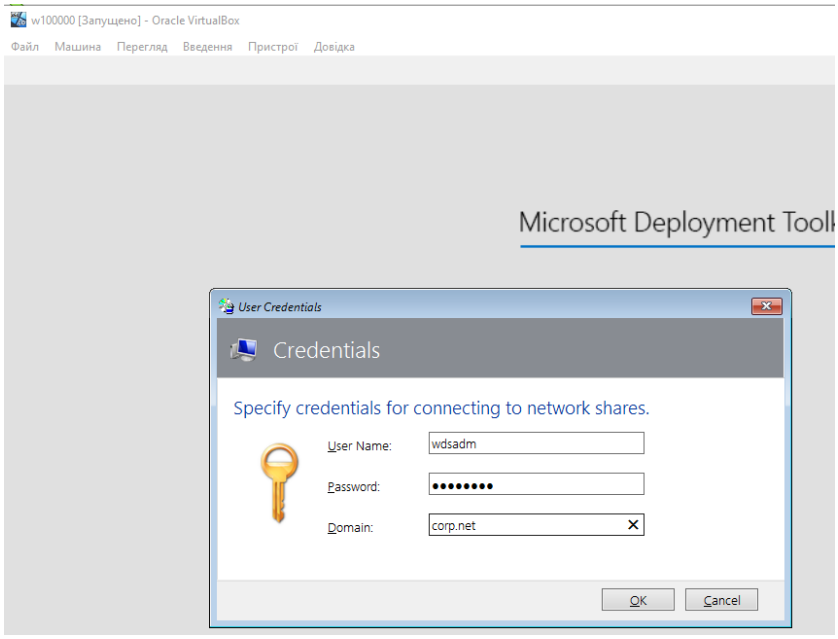


Рисунок – 3.49 Введення облікового запису адміністратора та домену

На рисунку 3.50 показано вікно вибору програм у середовищі Windows Deployment Wizard під час автоматизованого встановлення операційної системи. У цьому меню можна вибрати додаткове програмне забезпечення, яке буде автоматично встановлене на комп’ютер після завершення розгортання системи.

На рисунку видно, що для встановлення обрано програми 7-Zip, Google Chrome та VLC Media Player.

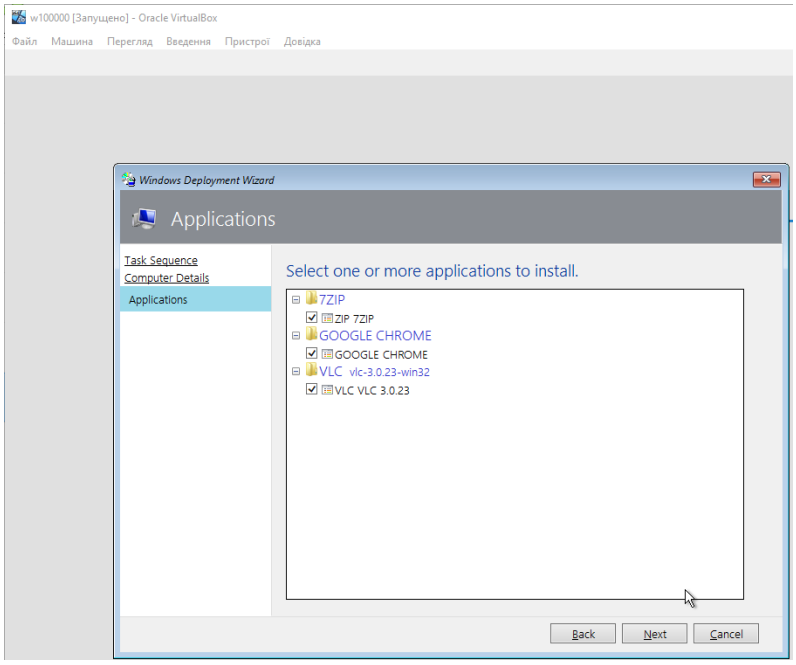


Рисунок – 3.50 Вибір прикладного програмного забезпечення під час встановлення

На рисунку 3.51 зображено вікно Windows PowerShell, запущене від імені адміністратора, у якому виконано команду Get-CimInstance -ClassName Win32_Product. Дана команда використовується для отримання інформації про встановлене програмне забезпечення в операційній системі.

У результаті виконання команди було отримано список встановлених програм, серед яких VLC Media Player, 7-Zip та Google Chrome. У вікні PowerShell також відображаються назва програмного забезпечення, його версія та унікальний ідентифікатор продукту. Отримані дані можуть використовуватись для перевірки коректності автоматизованого встановлення програм після розгортання операційної системи.

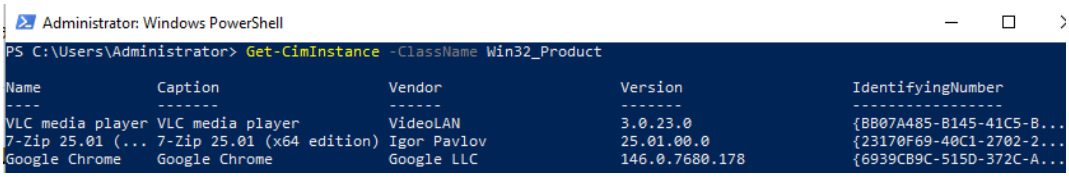


Рисунок – 3.51 Перегляд встановленого ПЗ

Для відображення списку драйверів, встановлених у системі під час розгортання операційної системи, було використано вбудовану консольну утиліту pnputil. Дана команда дозволяє отримати інформацію про всі драйвери, що містяться у Driver Store, включаючи їх постачальника, клас пристрою, версію та ідентифікатор пакета (INF-файл).

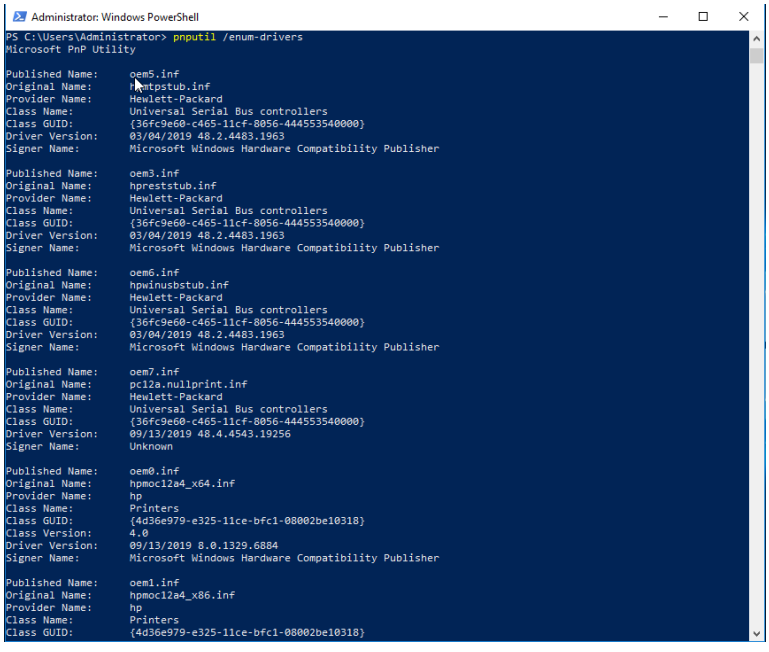


Рисунок – 3.52 Перевірка встановлення драйверів

Як показано на рисунку 3.53 було перевірено роботу скриптів. Вони успішно виконались та згенерували відповідні файли, де зберігається системна інформація та результати встановлення ОС.

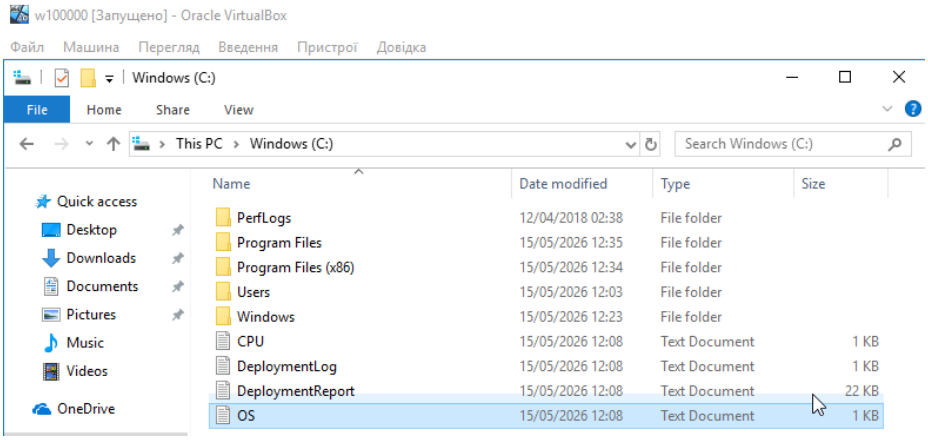


Рисунок – 3.53 Перевірка роботи скриптів

Оскількі в файлі CustomSettings.ini було налаштовано, щоб робочі станції переміщувалися в окрему організаційну одиницю після встановлення ОС. На рисунку 3.54 показано, що робоча станція наявна в цій організаційній одиниці.

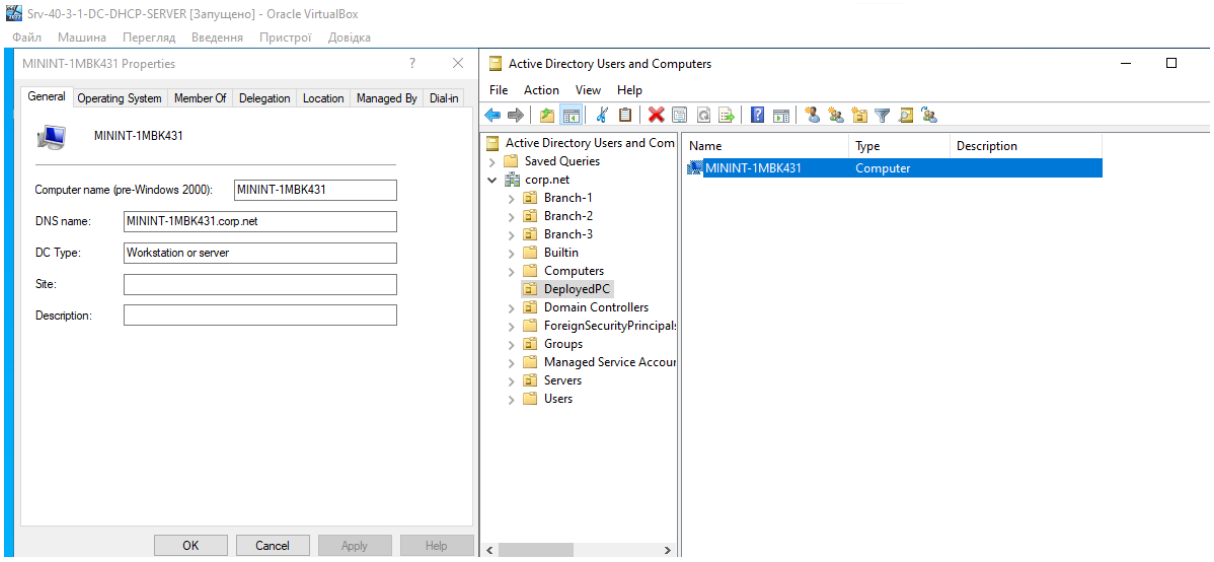


Рисунок – 3.54 Додавання робочої станції в окрему OU

Після завершення процесу розгортання операційної системи було проведено тестування групових політик (Group Policy Objects, GPO) з метою перевірки коректності їх застосування до клієнтських комп'ютерів доменної інфраструктури.

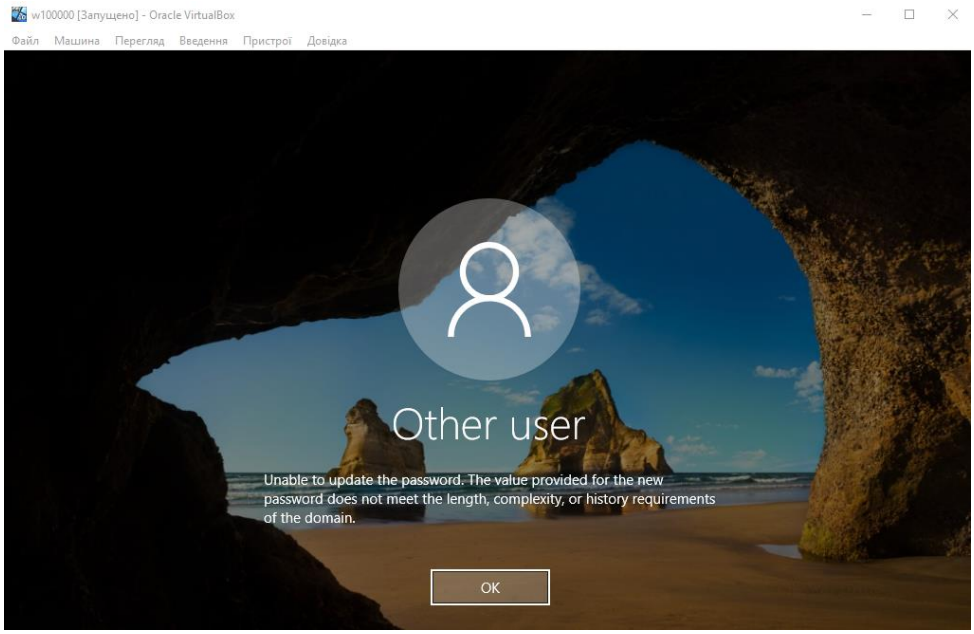


Рисунок – 3.55 Перевірка політики мінімальної довжини пароля

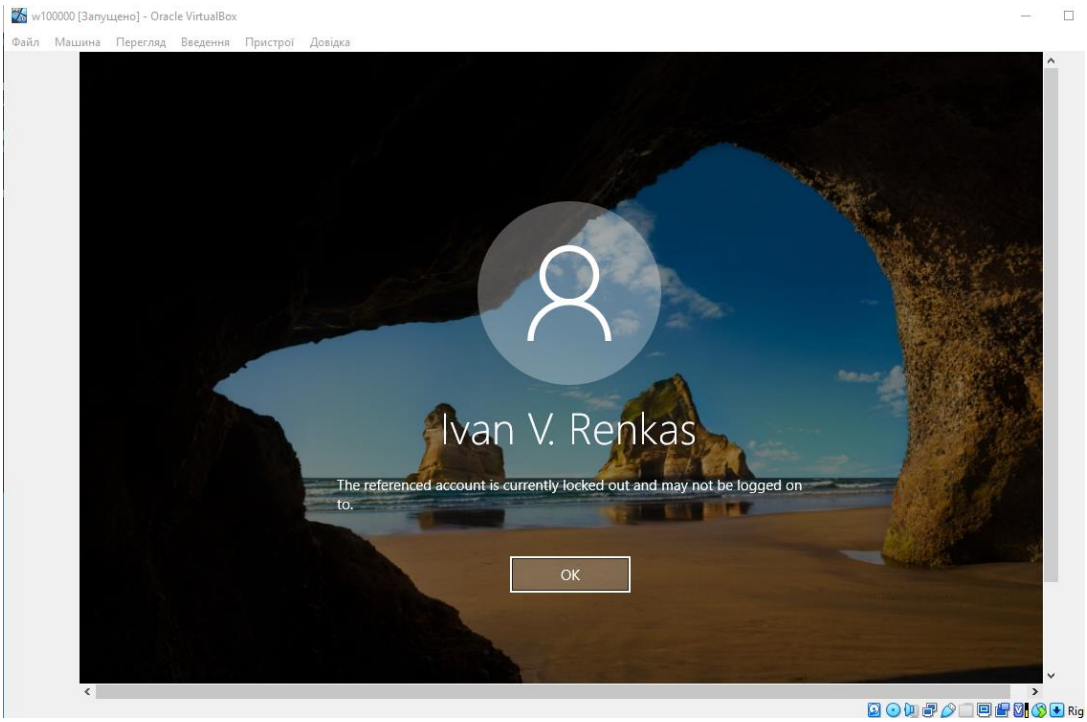


Рисунок – 3.56 Перевірка політики кількості невдалих спроб

3.5.2 Моніторинг роботи системи

Для моніторингу роботи системи в рамках виконання дослідження використовувались вбудовані інструменти операційної системи Windows Server, зокрема Performance Monitor (PerfMon) та Event Viewer (Перегляд подій). Застосування саме стандартних засобів адміністрування дозволило отримати об’єктивні та відтворювані результати без необхідності встановлення додаткового програмного забезпечення, що могло б вплинути на показники продуктивності системи.

Моніторинг продуктивності за допомогою Performance Monitor

Інструмент Performance Monitor використовувався для детального аналізу ключових системних ресурсів у режимі реального часу та з можливістю подальшого збереження логів для аналізу. Основна увага приділялась таким показникам:

- завантаження центрального процесора (CPU Usage, % Processor Time);
- використання оперативної пам'яті (Available MBytes, Committed Bytes, Cache);
- активність підсистеми введення-виведення;
- мережеве навантаження (Bytes Total/sec, Packets/sec).

Аналіз системних подій через Event Viewer

Додатково використовувався інструмент Event Viewer, який дозволяє здійснювати аналіз системних, безпекових та прикладних подій. У ході дослідження особлива увага приділялась:

- критичним системним помилкам (Critical);
- попередженням (Warning), що можуть свідчити про потенційні проблеми;
- інформаційним подіям (Information), які фіксують штатну роботу служб.

На рисунку 3.57 показано порівняння навантаження на процесор, де метрики продуктивності були зафіксовані до початку процесу мережевого розгортання операційних систем.

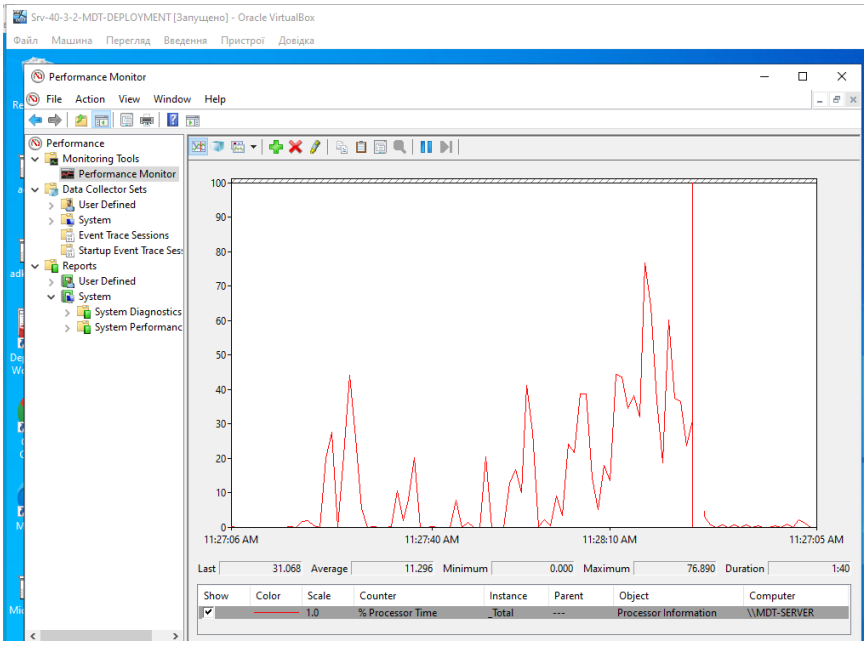


Рисунок – 3.57 Використання CPU під час мережевого розгортання

На рисунку 3.58 зображено вкладку «Продуктивність» системної утиліти Task Manager в операційній системі Windows. У даному вікні відображається моніторинг використання системних ресурсів комп'ютера в реальному часі.

У центральній частині вікна показано графік активності мережевого адаптера Intel(R) PRO/1000 MT Desktop Adapter. Графік демонструє рівень використання мережі за останні 60 секунд. На момент зняття показників швидкість передачі даних становила 55,9 Mbps, а швидкість отримання –760 Kbps, що свідчить про активний процес передавання даних мережею.

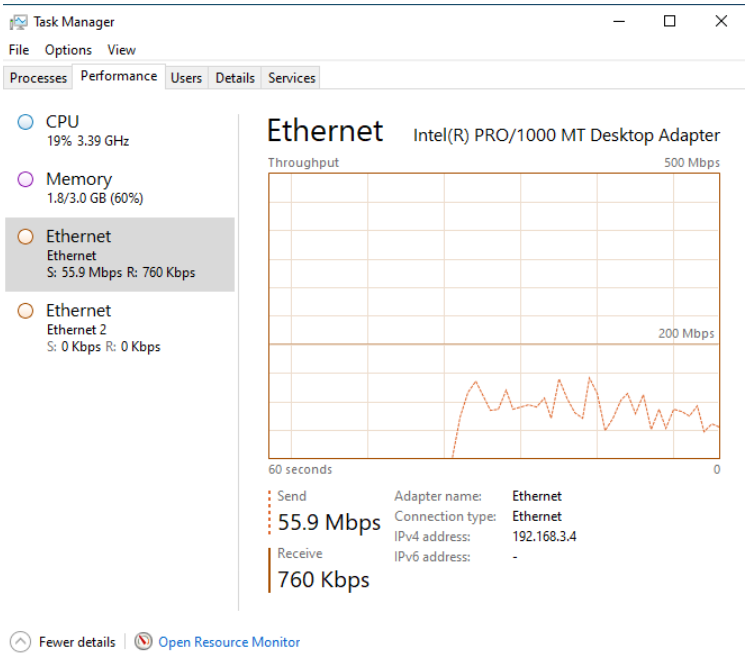


Рисунок – 3.58 Моніторинг мережевої активності

На рисунку 3.59 показано перегляд журналу подій Application у системній утиліті Event Viewer на сервері MDT. У центральній частині вікна відображається список зареєстрованих подій із зазначенням рівня події, дати та часу, джерела, ідентифікатора події та категорії задачі.

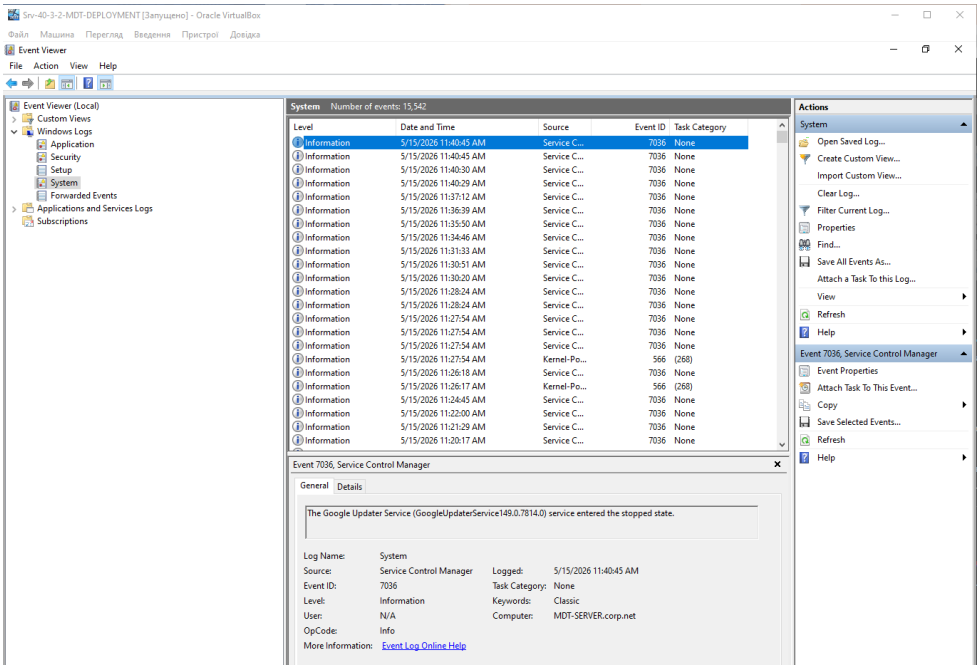


Рисунок – 3.59 Перегляд подій в Event Viewer

Для базового моніторингу процесу встановлення ОС, використано вкладку Monitoring в Microsoft Deployment Toolkit, що показано на рисунку 3.60.

На момент зняття показників процес розгортання перебував у стані Running, а прогрес виконання становив 57 % (55 із 95 кроків). Також видно, що під час встановлення не було зафіксовано помилок або попереджень, що свідчить про коректну роботу системи розгортання.

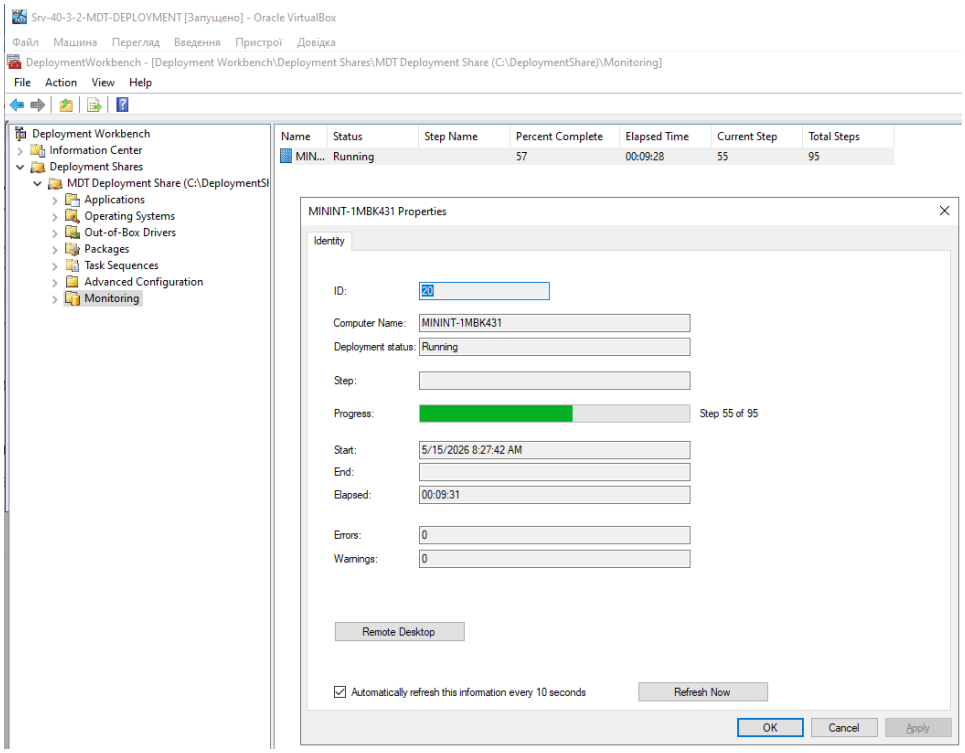


Рисунок – 3.60 Перегляд процесу встановлення через базовий моніторинг

3.5.3 Аналіз метрик ефективності

Для оцінки ефективності впровадженого рішення доцільно провести порівняльний аналіз ручного та автоматизованого способів встановлення операційної системи. Такий підхід дозволяє визначити основні переваги використання системи розгортання MDT/WDS з точки зору часу виконання, рівня автоматизації та навантаження на адміністратора. Результати порівняння наведено в таблиці 3.1.

Таблиця 3.1

Порівняння ручного та автоматизованого встановлення ОС

Параметр	**Ручне встановлення**	**Автоматизоване**
Час встановлення	40–60 хв	30–35 хв
Участь адміністратора	Постійна	Мінімальна
Ймовірність помилки	Висока	Низька
Масове розгортання	Неефективне	Ефективне
Налаштування системи	Вручну	Автоматично

Для детального аналізу продуктивності системи автоматизованого розгортання було проведено дослідження завантаження основних ресурсів серверів та мережевої інфраструктури. Оцінювалися показники використання процесора, оперативної пам’яті та мережевого трафіку на різних етапах розгортання операційної системи. Результати аналізу наведено в таблицях 3.2–3.4.

Таблиця 3.2

Завантаження процесора (CPU)

Стан системи	**Мін**	**Середнє**	**Макс**	**Опис**
У стані спокою	2%	7%	15%	Фонові процеси
Отримання IP	10%	18%	25%	DHCP запити
Завантаження образу	40%	55%	75%	PXE навантаження
Інсталяція ОС	30%	50%	70%	Розпакування
Після завершення	3%	6%	12%	Очікування

Після аналізу таблиці 3.2 було визначено, що система демонструє очікуваний профіль навантаження з короткочасними піками під час розгортання образу та інсталяції ОС, при цьому в режимі очікування споживання ресурсів є мінімальним, що підтверджує її ефективність і стабільність.

Таблиця 3.3

Використання оперативної пам’яті (RAM)

Стан системи	**Мін**	**Середнє**	**Макс**	**Опис**
Спокій	1.2 GB	1.3 GB	2 GB	Базові служби
PXE запуск	1.6 GB	2 GB	2.3 GB	Ініціалізація
Завантаження образу	2.4 GB	2.6 GB	2.9 GB	Кешування
Інсталяція	3.0 GB	3.0 GB	3.0 GB	Розпакування
Завершення	2.0 GB	2.5 GB	3.0 GB	Стабілізація

Згідно з даними таблиці 3.3, спостерігається закономірне зростання споживання оперативної пам’яті під час активних етапів розгортання із подальшим поверненням до стабільного рівня після завершення процесів, що свідчить про ефективне використання системних ресурсів.

Таблиця 3.4

Мережевий трафік

Стан системи	**Мін**	**Середнє**	**Макс**	**Опис**
Спокій	0	0.2 Мбіт/с	1 Мбіт/с	Фонові процеси
DHCP	0.1 Мбіт/с	0.5 Мбіт/с	2 Мбіт/с	Отримання IP
PXE	20 Мбіт/с	80 Мбіт/с	150 Мбіт/с	Завантаження
Інсталяція	30 Мбіт/с	110 Мбіт/с	200 Мбіт/с	Передача образу
Завершення	1 Мбіт/с	5 Мбіт/с	15 Мбіт/с	Фінальні запити

У цілому результати підтверджують, що основне навантаження на мережевий канал припадає на етапи PXE-завантаження та інсталяції, тоді як інші фази характеризуються мінімальним впливом на пропускну здатність мережі, що показано в таблиці 3.4.

Висновки до третього розділу

У третьому розділі було виконано практичну реалізацію системи автоматизованого розгортання операційних систем на базі технологій Windows Deployment Services та Microsoft Deployment Toolkit.

У процесі виконання розділу було розгорнуто доменну інфраструктуру з використанням служби Active Directory Domain Services, створено структуру домену та налаштовано базові групові політики, що забезпечують централізоване управління клієнтськими пристроями. Також реалізовано серверну архітектуру системи розгортання, що включає налаштування служб WDS і MDT та їх інтеграцію.

Було виконано налаштування процесу автоматизованого розгортання операційних систем із використанням Task Sequence, інтеграції драйверів і додавання прикладного програмного забезпечення. Окрему увагу приділено автоматизації процесу встановлення за допомогою скриптів, що дозволило мінімізувати втручання адміністратора та зменшити ймовірність помилок.

Проведене тестування системи підтвердило її працездатність та ефективність. У ході тестування було встановлено, що автоматизоване розгортання значно скорочує час інсталяції операційної системи порівняно з ручним методом, забезпечує стабільність процесу та уніфікованість конфігурацій. Моніторинг роботи системи показав стабільне використання ресурсів і відсутність критичних навантажень на серверну інфраструктуру.

Аналіз метрик ефективності дозволив зробити висновок, що впроваджена система є масштабованою, зручною в адмініструванні та придатною для використання в корпоративних мережах із великою кількістю робочих станцій.

Таким чином, у третьому розділі підтверджено доцільність використання технологій WDS та MDT для побудови систем автоматизованого розгортання операційних систем, що забезпечують підвищення ефективності адміністрування та зменшення витрат часу на підготовку робочих місць.

ВИСНОВКИ

У кваліфікаційній роботі було розглянуто теоретичні та практичні аспекти побудови масштабованої системи автоматизованого розгортання операційних систем на базі технологій Windows Deployment Services та Microsoft Deployment Toolkit. Проведено аналіз сучасних методів розгортання ОС, досліджено принципи роботи WDS, MDT, PXE, DHCP і TFTP, а також визначено особливості їх використання в корпоративних мережах.

У процесі виконання роботи було спроєктовано та реалізовано серверну інфраструктуру у віртуальному середовищі Oracle VM VirtualBox. Налаштовано доменну інфраструктуру Active Directory, служби WDS і MDT, а також реалізовано автоматизоване мережеве розгортання операційних систем із використанням Task Sequence, драйверів, прикладного програмного забезпечення та PowerShell-скриптів.

У межах практичної частини проведено тестування працездатності системи та аналіз її ефективності. Отримані результати підтвердили, що використання автоматизованого розгортання дозволяє скоротити час встановлення операційних систем, зменшити кількість помилок під час налаштування та забезпечити стандартизацію програмного середовища на клієнтських пристроях.

Розроблена система має можливість подальшого масштабування та модернізації. Зокрема, система може бути масштабована шляхом додавання додаткових серверів розгортання для обслуговування віддалених філій або збільшення кількості одночасних PXE-клієнтів. Також можливе впровадження механізмів балансування навантаження, резервування серверів та інтеграції з хмарними сервісами Microsoft.

Отже, поставлена мета кваліфікаційної роботи була досягнута, а розроблена система автоматизованого розгортання операційних систем може використовуватись як основа для побудови сучасної масштабованої інфраструктури в корпоративному середовищі.

Список використаних джерел

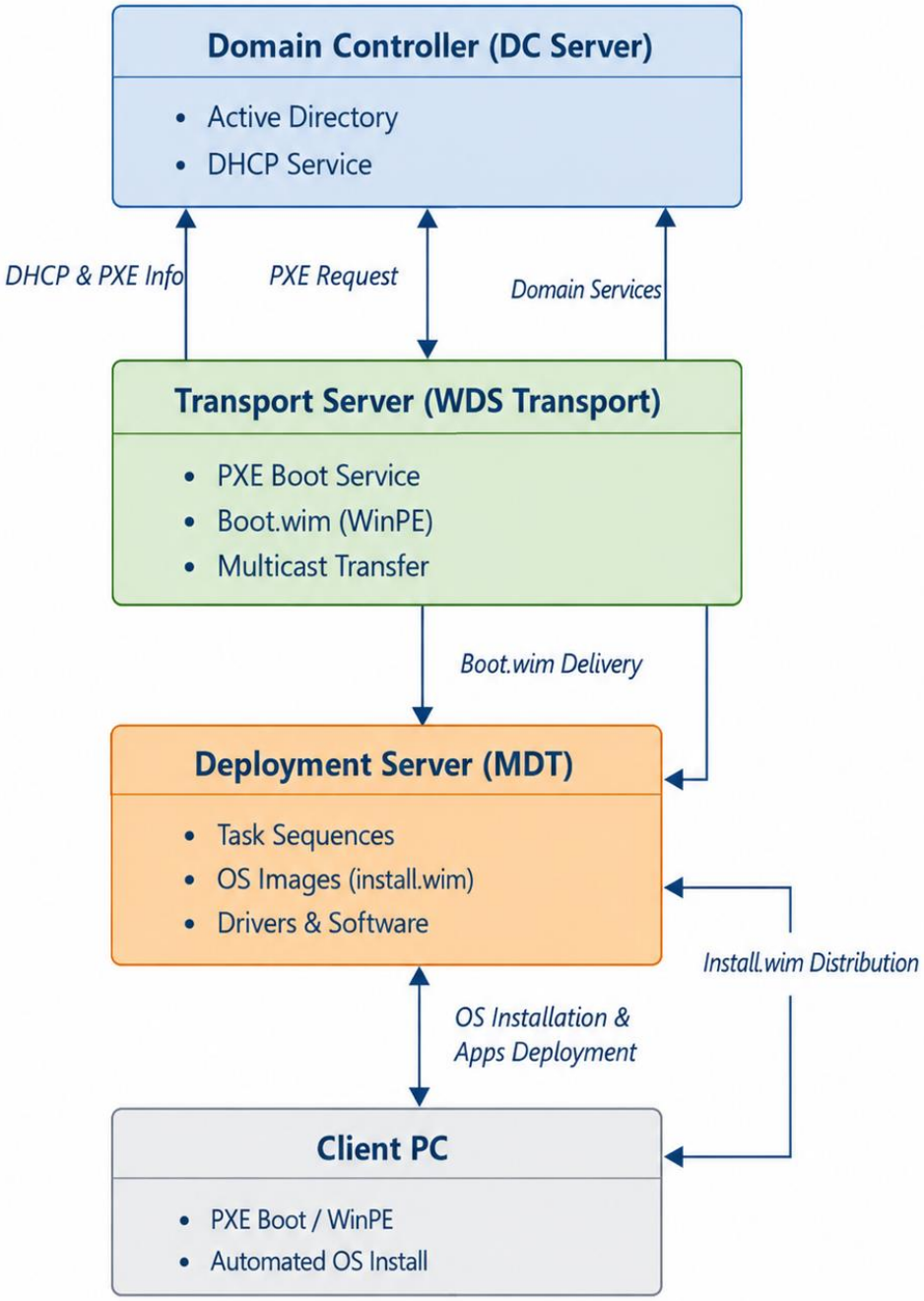
1. Robert Sheldon, Paul Crocetti. What is ghost imaging (disk imaging)? [Електронний ресурс]. – Режим доступу: https://www.techtarget.com/searchdatabackup/definition/ghost-imaging.
2. VPN Unlimited. PXE-сервер: що це таке та як він працює [Електронний ресурс]. – Режим доступу: https://surl.li/gdhaqg.
3. Microsoft. Розгортання пристроїв за допомогою Windows Autopilot [Електронний ресурс]. – Режим доступу:.https://learn.microsoft.com/uk-ua/training/modules/windows-autopilot-2026/.
4. Microsoft Windows Deployment Services [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/hh831764(v=ws.11).
5. Microsoft. Windows Deployment Services Getting Started Guide — Prerequisites for installing Windows Deployment Services [Електронний ресурс]. –Режим доступу: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/jj648426(v=ws.11).
6. Preboot Execution Environment (PXE) [Електронний ресурс]. – Режим доступу: https://github.com/tianocore/tianocore-wiki.github.io.
7. RFC 2131: Dynamic Host Configuration Protocol [Електронний ресурс]. – Режим доступу: https://www.rfc-editor.org/rfc/rfc2131.html.
8. RFC 1350: The TFTP Protocol (Revision 2) [Електронний ресурс]. – Режим доступу: https://datatracker.ietf.org/doc/html/rfc1350.
9. PXE/BINL – Windows Network Install – Basis [Електронний ресурс]. – Режим доступу:https://www.vercot.com/~serva/an/WindowsPXE1.html.
10. Microsoft. Windows PE (WinPE) [Електронний ресурс]. – Режим доступу:https://surl.li/fhfnuu.
11. Microsoft. Using the Microsoft Deployment Toolkit [Електронний ресурс]. –Режим доступу: https://learn.microsoft.com/uk-ua/intune/configmgr/mdt/.

12. Microsoft. What is Configuration Manager? [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/en-us/intune/configmgr/core/understand/introduction.
13. Microsoft. What is Configuration Manager? [Електронний ресурс]. – Режим доступу: https://www.colorado.edu/amath/microsoft-endpoint-configuration-manager-mecm.
14. Oracle. VirtualBox User Manual. VirtualBox Documentation. – Режим доступу: https://www.virtualbox.org/manual/.
15. ITRex Group. What is software scalability? [Електронний ресурс]. – Режим доступу: https://itrexgroup.com/blog/what-is-software-scalability/.
16. Microsoft. Windows Assessment and Deployment Kit (Windows ADK) [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/uk-ua/windows-hardware/get-started/adk-install.
17. Microsoft. Create a bootable Windows PE media [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/en-us/windows-hardware/manufacture/desktop/winpe-create-usb-bootable-drive.
18. Microsoft. Windows Unattended Setup Reference [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/en-us/windows-hardware/customize/desktop/unattend/.
19. Microsoft. Windows Imaging and Configuration Designer (ICD) [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/en-us/windows/configuration/provisioning-packages/provisioning-create-package.
20. Microsoft. DISM (Deployment Image Servicing and Management) [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/en-us/windows-hardware/manufacture/desktop/what-is-dism.
21. Microsoft. Answer files (unattend.xml) [Електронний ресурс]. – Режим доступу: https://learn.microsoft.com/en-us/windows-hardware/manufacture/desktop/update-windows-settings-and-scripts-create-your-own-answer-file-sxs?view=windows-11.

ДОДАТКИ

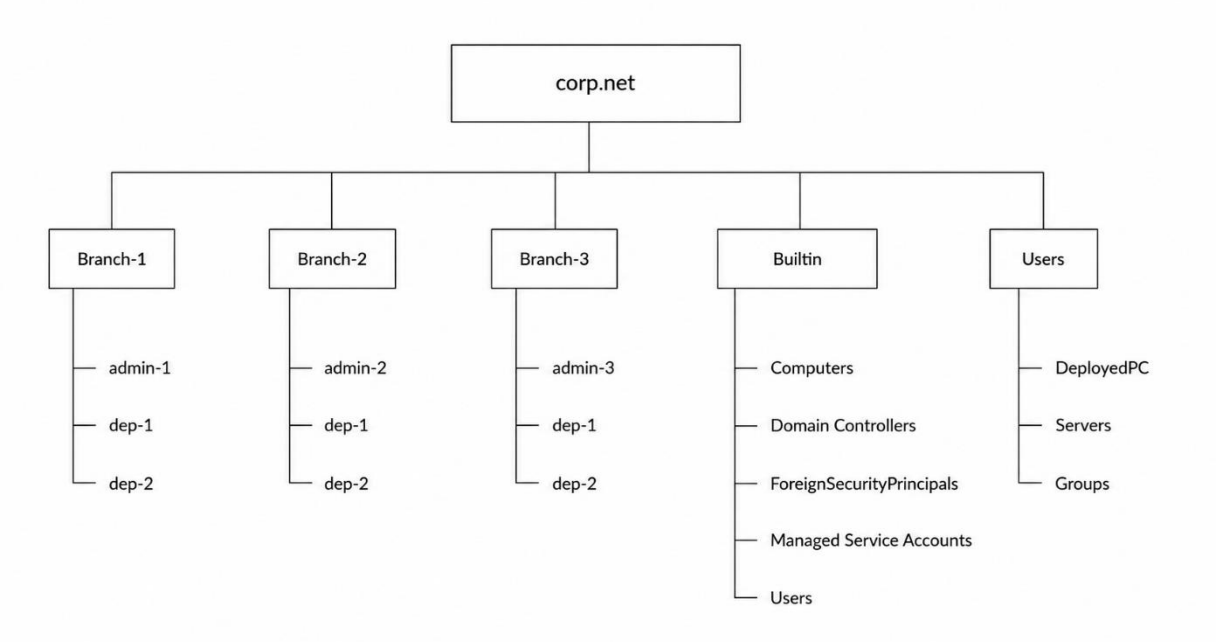
Додаток А

Логічна схема мережі



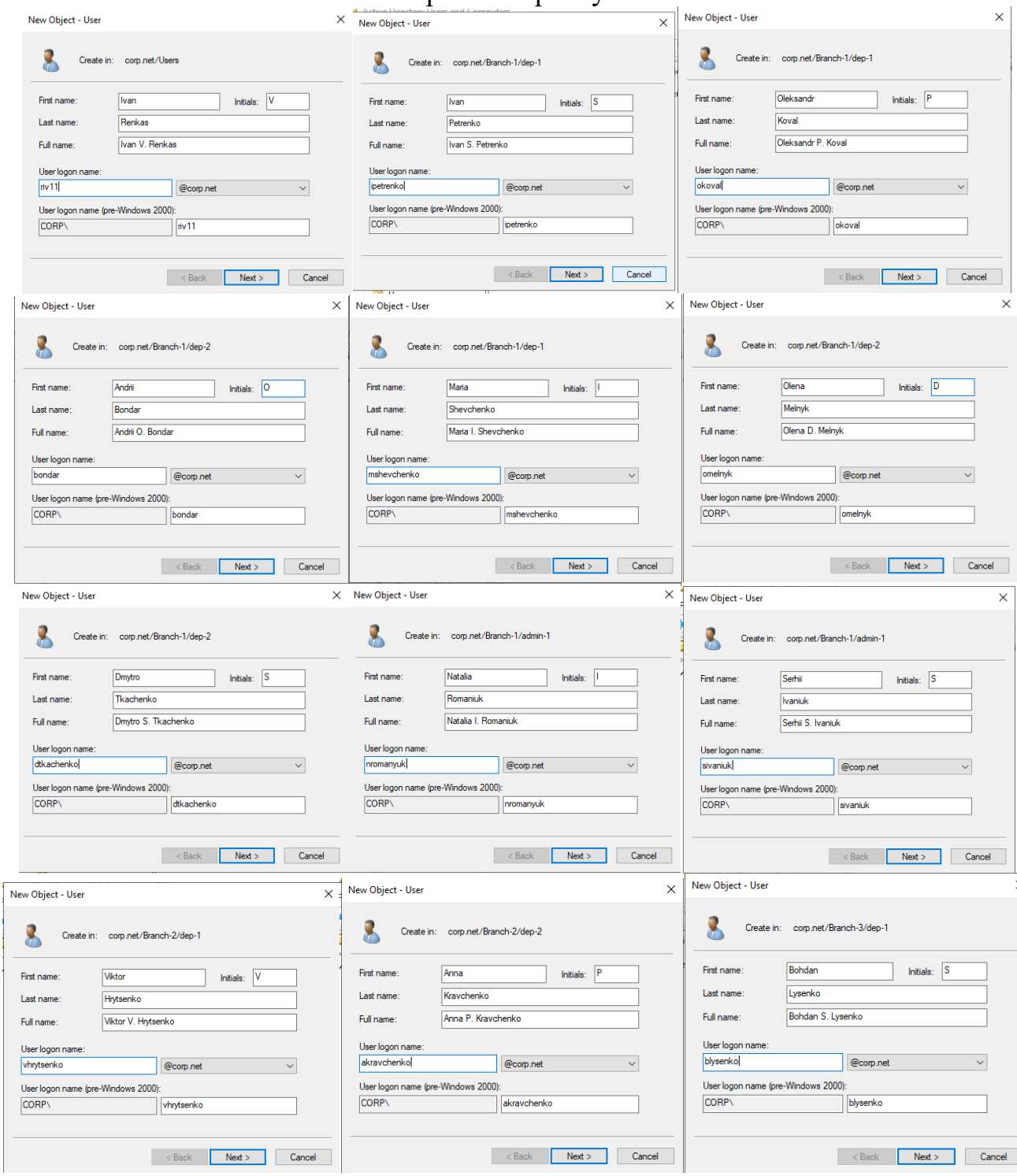
Додаток Б

Логічна структура організаційних одиниць



Додаток В

Створення користувачів



Додаток Г

Використанні скрипти

Скрипт CreateUser.ps1

```
# 1. СТВОРЕННЯ ЛОКАЛЬНОГО АДМІНІСТРАТОРА
$password = ConvertTo-SecureString "Admin123!" -AsPlainText -Force
New-LocalUser -Name "admin_lab" `
        -Password $password `
        -FullName "Lab Administrator" `
        -Description "Local admin account created during deployment"
Add-LocalGroupMember -Group "Administrators" -Member "admin_lab"
```

Скрипт DeploymentLog.ps1

```
# 2. ЗБІР ІНФОРМАЦІЇ ПРО СИСТЕМУ
Get-ComputerInfo | Out-File "C:\DeploymentReport.txt"
Get-WmiObject Win32_Processor |
Select-Object Name, NumberOfCores, NumberOfLogicalProcessors, MaxClockSpeed |
Out-File "C:\CPU.txt"
Get-WmiObject Win32_OperatingSystem |
Select-Object Caption, Version, BuildNumber, OSArchitecture |
Out-File "C:\OS.txt"
```

Скрипт SecurityConfig.ps1

```
# 3. БАЗОВІ НАЛАШТУВАННЯ БЕЗПЕКИ
Set-NetFirewallProfile -Profile Domain,Public,Private -Enabled True
# Вимкнення застарілого протоколу SMBv1
Disable-WindowsOptionalFeature -Online -FeatureName SMB1Protocol -NoRestart
# Додаткове посилення політики брандмауера
Set-NetFirewallProfile -DefaultInboundAction Block -DefaultOutboundAction Allow
```

Скрипт SystemInfo.ps1

```
# 4. БАЗОВЕ ЛОГУВАННЯ ЗАВЕРШЕННЯ
$time = Get-Date
Add-Content "C:\DeploymentLog.txt" "Deployment completed at $time"
# 5. РОЗШИРЕНЕ ЛОГУВАННЯ ПРОЦЕСУ
Add-Content "C:\DeploymentLog.txt" "============================="
Add-Content "C:\DeploymentLog.txt" "DEPLOYMENT SUMMARY LOG"
Add-Content "C:\DeploymentLog.txt" "Timestamp: $time"
Add-Content "C:\DeploymentLog.txt" "-----------------------------"
Add-Content "C:\DeploymentLog.txt" "Local user admin_lab created and added to
Administrators"
Add-Content "C:\DeploymentLog.txt" "System information collected successfully"
Add-Content "C:\DeploymentLog.txt" "Firewall enabled for all profiles"
Add-Content "C:\DeploymentLog.txt" "SMBv1 disabled for security hardening"
Add-Content "C:\DeploymentLog.txt" "Deployment process completed successfully"
Add-Content "C:\DeploymentLog.txt" "============================="
```

Додаток Ґ

Лістинг конфігураційний файлів

Лістинг CustomSettings.ini:

```
[Settings]
Priority=Default
Properties=MyCustomProperty
[Default]
OSInstall=Y
SkipCapture=YES
SkipComputerName=NO
SkipAdminPassword=YES
SkipProductKey=YES
SkipComputerBackup=YES
SkipBitLocker=YES
TimeZoneName=FLE Standard Time
KeyboardLocale=0809:00000809
UILanguage=en-GB
UserLocale=en-GB
KeyboardLocale=en-GB
JoinDomain=corp.net
DomainAdmin=wdsadm
DomainAdminPassword=Zeka2005
MachineObjectOU=OU=DeployedPC, DC=corp,DC=net
SkipUserData=YES
SkipDomainMembership=YES
SkipLocaleSelection=YES
SkipTimeZone=YES
SkipSummary=YES
SkipFinalSummary=YES
EventService=http://WDS-SERVER:9800
```

Лістинг Bootstrap.ini:

```
[Settings]
Priority=Default
[Default]
DeployRoot=\\WDS-SERVER\DeploymentShare$
SkipBDDWelcome=YES
```