МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

ВИПУСКНА РОБОТА

на тему:

«Система захисту комерційної таємниці в комп'ютерній мережі підприємства на основі Active Directory»

Завідувач випускаючої кафедри — **Довбиш А.С.**

Керівник роботи — **Кальченко В.В.**

Студента групи КБ – 71 — **Поздняков Р.О.**

СУМИ 2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

Кафедра комп'ютерних наук

Затверджую_____
Зав. кафедрою Довбиш А.С.
"_____"_____2021 р.

ЗАВДАННЯ
до випускної роботи

Студента четвертого курсу, групи КБ-71 спеціальності "Кібербезпека" денної форми навчання Позднякова Родіона Олеговича

Тема: " Система захисту комерційної таємниці в комп'ютерній мережі підприємства на основі Active Directory"

Затверджена наказом по СумДУ
№_____від_____2021 р.

Зміст пояснювальної записки:

1) аналіз предметної області;
2) постановка задачі;
3) реалізація серверної частини системи захисту інформації;
4) аналіз отриманих результатів.

Дата видачі завдання "_____"_____2021 р.
Керівник випускної роботи_____Кальченко В.В.
Завдання приняв до виконання_____Поздняков Р.О.

РЕФЕРАТ

Записка: стор. – 40, рис. – 19, табл. – 1, додатки – 3, джерел – 16.

Об’єкт дослідження — процес налаштування доменних служб та системи безпечного обміну інформацією за допомогою Active Directory.

Мета роботи — автоматизоване створення системи захисту комерційної таємниці в корпоративній мережі підприємства.

Методи дослідження — методи централізованого налаштування Active Directory та отримання інформування про несанкціоноване додавання користувачів у список адміністраторів служби каталогів та несанкціонований дступ до мережі.

Результати — розроблено скриптовий алгоритм дій розгортання та налаштування серверу корпоративної мережі офісу малого підприємства, було виконано розгортання і налаштування каталогів Active Directory на базі серверної операційної системи Windows Server 2019, логіка директорій для обміну інформацією між користувачами та розроблена система сповіщення адміністратора безпеки про несанкціонований доступ до ресурсів домену.

ACTIVE DIRECTORY, ДОМЕН, ЛІС, DHCP,
DNS, ГРУПИ БЕЗПЕКИ, СПІЛЬНІ ДИРЕКТОРІЇ, ТРИГЕРИ

ЗМІСТ

ВСТУП

Одним із найважливіших ресурсів будь-якого підприємства є комерційна таємниця, тобто інформація, що є секретною і цінною з точки зору бізнесу, доступ до якої повинні мати лише працівники підприємства або окремі підрозділи підприємства. Даний проект спрямований на вирішення проблеми захисту та адміністрування корпоративної мережі підприємства задля забезпечення і підтримки середовища для безпечного обміну інформацією.

Дана робота розглядає сучасні методи адміністрування і централізованої побудови захисту особистої інформації підприємства на основі готових рішень і сценаріїв автоматизації роботи даних рішень. Метою даних автоматизацій буде створення і налаштування середовища для авторизації співробітників підприємства на робочих станціях і встановлення директорій для обміну інформацією між структурними підрозділами, а також захист даних на основі дозволів і обмежень структурних одиниць.

Метою цього проекту є розробка функціонального середовища на основі готових рішень для малого та середнього бізнесу, що буде базою для корпоративної мережі підприємства. Дане середовище повинне відповідати критеріям захисту від спотворення, модифікації або викрадення цінної інформації, а також реалізована система повідомлень і моніторингу порушень задля швидкого реагування на несанкціонований доступ до мережі або інформації підприємства.

1 АНАЛІЗ ПРОБЛЕМИ ТА ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

1.1 Сучасні методи організації підприємства

Для вирішення задачі централізованого керування і захисту локальної мережі підприємства існує достатня кількість рішень, які мають різну реалізацію, підхід та різні методики використання. У сучасних реаліях вибір методу вирішення даної задачі залежить лише від вподобань підприємства, та задач, які воно виконує.

На базі великих і малих підприємств можна використовувати різну структуру адміністрування мережевих одиниць. Адміністрування локальної мережі можна вести від окремого кластеру структурної одиниці підприємства, так і адмініструвати повністю корпоративну мережу, використовуючи готові рішення для побудови централізованого керування мережею.

Класифікація мережі по способу її управління поділяється на однорангові мережі та мережі с виділеним сервером (рис.1.1). Основним недоліком однорангових мереж є слабкий захист інформації від несанкціонованого доступу, тому таку мережу не використовують на базі підприємства. Для малих і великих підприємства використовують мережі с виділеним сервером, тобто комп’ютером, який надає свої ресурси в спільне використання для клієнтів даної мережі, тобто, робочих станцій.

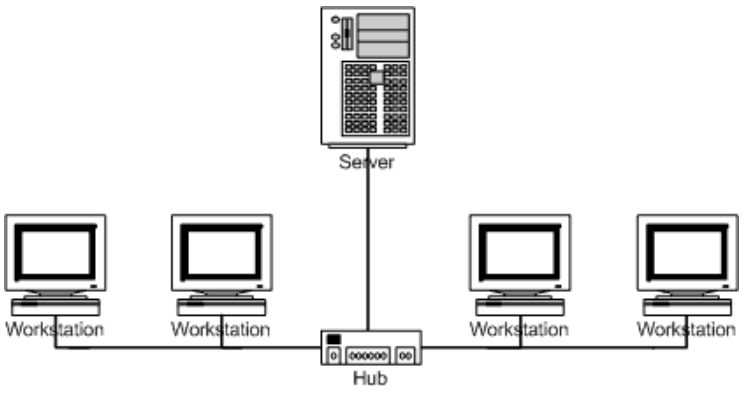

Рисунок 1.1 – Мережа з виділеним сервером

У даному типі мережі на сервері встановлюється мережева операційна система, до нього підключається всі мережеві зовнішні пристрої. Взаємодія між робочими станціями в мережі здійснюється через сервер, він і виконує роль центрального пристрою. Мережі на основі сервера здатні підтримувати тисячі користувачів. Мережами такого розміру, будь вони однорангові, було б неможливо централізовано керувати. Основним аргументом при виборі мережі на основі сервера є, як правило, захист даних. У таких мережах проблемами безпеки може займатися один адміністратор, він формує політику безпеки і застосовує щодо кожного користувача мережі.

1.2 Microsoft Active Directory на Windows Server

Операційна система Windows Server вкрай популярна в корпоративному сегменті, хоча більшість рядових користувачів асоціюють Windows виключно з версією для персональних комп’ютерів . Залежно від завдань і необхідної до підтримки інфраструктури зараз в експлуатації компаній знаходяться відразу кілька версій Windows Server, починаючи з Windows Server 2003 і закінчуючи останньою версією - Windows Server 2019.

Технологія Active Directory в Windows Server використовується для зберігання та організації об'єктів в мережі в ієрархічну захищену логічну структуру, наприклад користувачів, комп'ютерів або інших фізичних ресурсів.

Служби Active Directory – це рішення від компанії Microsoft дозволяє об'єднати різні об'єкти мережі, такі як комп'ютери, сервери, принтери, різні сервіси, в єдину систему. В даному випадку служби виступають в ролі каталогу, іншими словами, бази даних, в якій зберігається інформація про користувачів, персональних комп’ютерах, серверах, мережевих і периферійних пристроях підприємства.

Виділений сервер підприємства виступає контролером домену, на ньому розгортаються служби Active Directory. Ці служби будуть в подальшому виконувати функції перевірки автентичності користувачів у домені і реєстрацію кінцевих пристроїв, а в подальшому, виступати базою даних пристроїв і користувачів домену. Якщо будь-хто намагається використати об'єкт мережі, то буде виконуватись звернення до контролера домену підприємства, надалі, контролер або ж дозволяє доступ, або відхиляє (блокує) його, спираючись на свою базу даних.

Ліс і домен складають основу логічної структури Active Directory. Домени можуть бути структуровані в лісі, щоб забезпечити незалежність даних і сервісів та оптимізацію реплікації. Поділ логічних і фізичних структур поліпшує керованість системи і знижує адміністративні витрати, тому що на логічну структуру не впливають зміни в фізичному пристрої. Логічна структура дозволяє контролювати доступ до даних, тобто ви можете використовувати логічну структуру для контролю доступу до різних блокам даних.

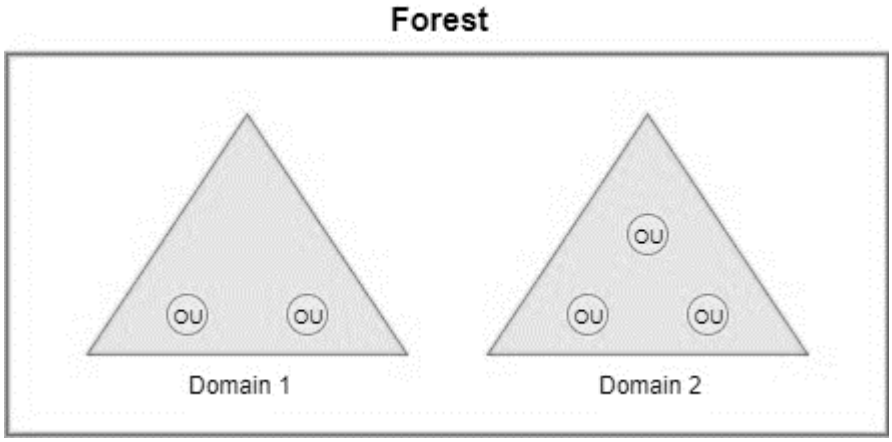

Рисунок 1.2 – Структура лісу в Active Directory

Ліс у контролері доменів Active Directory (рис.1.2) поділяється на логічні структури – домени, які по суті своїй мають в собі дані про технічний поділ підприємства, на якому вони реалізовані, через фізичну структуру, що містить в

собі базу даних, яка зберігається на всіх контролерах домену в лісі. У Active Directory об'єкти використовують каталоги для зберігання інформації, всі об'єкти мають визначеність в загальній схемі структури. Визначення об'єктів містять інформацію, таку як тип даних і синтаксис, яку каталог використовує, щоб гарантувати достовірність зберігання .Ніякі дані не можуть бути збережені в каталозі, поки вони не визначені в схемі. Схема за замовчуванням містить всі визначення і опису об'єктів, які необхідні для коректної роботи.

Сховище всередині Active Directory обробляє доступ до бази даних контролеру домену. Сховище цих даних складається з служб і фізичних файлів, що централізовано керують правами доступу до системи, процесами читання і запису даних усередині бази даних на жорсткому диску кожного контролера.

Служби Active Directory істотно збільшують захист корпоративної мережі. Так, всі дані, наприклад, облікові записи, зберігаються на контролерах домену, які захищені від зовнішнього доступу. Крім того, для аутентифікації в службах Active Directory використовується протокол Kerberos – це протокол для взаємної аутентифікації клієнта і сервера перед установкою з'єднання, в ньому враховано перехоплення і модифікації пакетів, що підвищує його надійність, який значно безпечніше аналога в робочих групах.

Active Directory пропонує зручне управління політиками. За допомогою даної служби можна поділити комп'ютери на різні робочі групи – організаційні підрозділи. Це істотно спрощує використання інфраструктури в двох випадках:

- Зміна існуючих налаштувань групи. Оскільки налаштування зберігаються в єдиній базі даних, при їх модифікації, вони будуть застосовані для всіх комп'ютерів, що належать до цієї групи.
- Додавання нового користувача. Він автоматично отримує встановлені для його групи налаштування, що істотно прискорює створення нового облікового запису.

Залежно від користувача, облікового запису, який використовується, і

його групи, можна ввести обмеження на використання функціоналу операційної системи. Наприклад, ви можете обмежити встановлення додатків для всіх, крім адміністраторів.

Служби каталогів дозволяють організувати все обладнання і сервіси в єдину систему. Дані, що зберігаються в Active Directory, можуть надходити з різних джерел. З великою кількістю різних джерел даних і безліччю різних типів даних Active Directory повинен використовувати деякий стандартизований механізм зберігання, щоб підтримувати цілісність інформації, що зберігається.

Основні переваги серверів під управлінням Windows - відносна простота адміністрування та досить великий пласт інформації, що підтримується системою. Крім того, ви не зможете обійтися без сервера на Windows, якщо в екосистемі компанії є програмне забезпечення або рішення, що використовують бібліотеки і частини ядра систем Microsoft. Крім того, Windows Server володіє полегшеною версією без графічного ядра з низьким споживанням ресурсів.

Дане рішення обирають, виходячи з простоти використання і налаштування у малих корпоративних мережах, де вся система робочих станцій побудована на Windows. Просте налаштування і графічний інтерфейс дасть змогу без проблемно адмініструвати персональні комп’ютери у робочій мережі офісу та централізованого управління, а також дозволить з легкістю реалізовувати технічну підтримку підприємства, при його розширенні.

1.3 Samba на Ubuntu Server

На базі родини операційних систем Ubuntu адміністрування виділених серверів підприємств можна встановити достатньо велику кількість готових рішень для адміністрування мережі малого підприємства з виділеним сервером. Серед популярних систем адміністрування вирізняють Ubuntu Server.

Linux Ubuntu Server – це безкоштовна серверна операційна система на базі ядра Linux. Ubuntu Server можна використовувати в якості платформи для серверів баз даних, DNS-серверів, файлових серверів та інших типів серверів. Для централізованого адміністрування і захисту, наприклад, корпоративної мережі використовують зв’язку Ubuntu Server і Samba.

Samba - це програмне забезпечення що спрямоване на організацію обміну файловими ресурсами і реалізацію роботи з цими ресурсами між комп'ютерами, що керуються ядром Linux / Unix. Samba складається з клієнтської і серверної частини:

- Клієнтська – дозволяє отримати доступ до мережевих файлів і директорій.
- Серверна – відкриває загальний доступ до системи Ubuntu для інших пристроїв у мережі.

За допомогою контролеру Samba можна перетворити діючий сервер, що працює під управлінням операційної системи сімейства Linux або UNIX, в контролер домену. Контролер домену підходить для централізованого адміністрування та зберігання даних облікових записів користувачів і комп'ютерів та правильного розмежування користувачів системи між собою.

Контролер домену виступає захистом робочих станцій мережі на основі перевірки доступу. Контролери домену мають задачу зберігати дані каталогу і керувати взаємодією між користувачем та доменом організації, основною задачею якого є процес перевірки входу користувача в систему, автентичність користувача і пошуки в каталозі, також, до цього можна додати деякі обмеження для користувачів, щоб захистити інформацію від витоку, що представляє собою комерційну таємницю підприємства. Це дозволяє централізовано захистити локальну мережу від несанкціонованого доступу через робочі станції користувачів та у разі виявлення несанкціонованого доступу обмежити доступ до важливої інформації.

Дане рішення підходить для всіх типів підприємств. Використання даної зв’язки, як централізованого контролеру локальної мережі підприємства зможе вирішувати проблеми налаштування та централізованого захисту робочих станцій в локальній мережі. Це дозволить додавати до існуючого захисту в перспективі більш складні рішення для захисту підприємства.

Крім того, Samba є програмним забезпеченням з відкритим вихідним кодом і розповсюджується за ліцензією вільною для копіювання, змінення та вдосконалення, навіть у промислових масштабах, а це в кінцевому рахунку дозволяє:

- знизити ризики, пов'язані з використанням імпортного програмного забезпечення;
- знизити сукупну вартість володіння інформаційною системою.

Для невеликих і середніх організацій, які планують організувати домен для зберігання і пошуку інформації про об'єкти інформаційних систем, а також для організацій, які по ряду причин планують перехід на системне програмне забезпечення, Samba може бути непоганою альтернативою Microsoft Active Directory.

На відміну від Microsoft Active Directory, якщо порівнювати дані рішення, то Samba має ряд деяких функціональних обмежень, таких як максимальний розмір бази даних Samba обмежений 4 Гб. Обмеження максимального розміру бази даних Samba пов'язано з 32-бітної архітектурою. Для великих організацій, с сотнями тисяч об'єктів в каталозі Active Dirtectory, перехід на Samba може виявитися неможливим. Таким чином для малої організації це рішення підійде, але лише до розширення підприємства і до побудови нових філіалів.

Серед інших обмежень, що будуть впливати на подальше розширення мережі в контролері доменів Samba можна виділити відсутність повноцінної підтримки, яка є повноцінно реалізованою у Windows Server На такому контролері домену можна зберегти екземпляр бази даних домену Active

Directory, доступний тільки для читання, але його функціональність набагато ширше, ніж у простого примірника бази даних, придатного тільки для читання.

Відсутність підтримки Kerberos у Samba. Kerberos – це мережевий протокол аутентифікації, який пропонує механізм взаємної перевірки клієнта домену, тобто користувача, і сервера, після чого встановлюється зв'язок між ними, причому в протоколі врахований той факт, що початковий обмін інформацією між клієнтом і сервером відбувається в незахищеною середовищі, а передані пакети можуть бути перехоплені і модифіковані.

В цілому контролер Samba – це достатній набір інструментів для адміністрування та централізованого захисту підприємства, але у реаліях підприємства дане рішення буде логічним, при малих розмірах підприємства, без майбутнього розширення та для побудови системи, дружньої з сімействами операційних систем UNIX.

2 ПОСТАНОВКА ЗАДАЧІ ТА МОДЕЛЮВАННЯ БАЗИ ПРОЕКТУ

2.1 Постановка задачі

Метою даного проекту є автоматизоване налаштування системи централізованого керування і захисту за допомогою сповіщень та моніторингу на базі малого підприємства з використанням Active Directory.

Розробка даної автоматизованої системи налаштування орієнтована на невеликі підприємства з робочою зоною в межах офісу, але з можливістю майбутнього розширення. Автоматизація дозволить швидко розгорнути мережу, додати користувачів та організувати діючі сповіщення системного адміністратора мережі або адміністратора безпеки про несанкціонований доступ до директорій адміністрування контролеру домену, а також показати історію підключень до домену, які мають сумнів в їх автентичності, як користувача, що допоможе завчасно відключити порушника з системи.

Розроблювана автоматизація має відповідати вимогам:

- автоматизоване встановлення і налаштування контролеру домену Active Directory;
- автоматизоване налаштування DHCP-ролі серверу;
- автоматизоване налаштування DNS-ролі серверу;
- додавання користувачів із сформованої бази співробітників підприємства;
- створення департаментів підприємства;
- створення і налаштування груп безпеки користувачів домену;
- налаштування спільних директорій для організації робочого процесу та обмеження доступу до цих директорій, задля розмежування потоку інформації між співробітниками;
- налаштування сповіщень адміністратора системи про несанкціоновану зміну груп адміністрації системи;

- здатність аналізу інформації про користувачів задля виявлення останніх підключень до домену;
- реалізована система повинна відповідати вимогам пункту 11 "Правил забезпечення захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах", затверджених Постановою Кабінету Міністрів України від 29.03.2006 №373.

Пункт 11 "Правил забезпечення захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах", затверджених Постановою Кабінету Міністрів України від 29.03.2006 №373, має в собі чіткі вказівки щодо вимог до розробленої системи захисту і обміну інформацією:

- У розробленій системі здійснюється обов'язкова реєстрація, що має результат ідентифікації та автентифікації користувачів.
- Має результат виконання користувачем системи операцій з обробки інформації.
- Реагування на спроби несанкціонованих дій з інформацією.
- Система має мати факт надання та позбавлення користувачів права доступу до певної інформації.
- Моє бути забезпеченою можливість проведення аналізу реєстраційних даних лише користувачем, що має привілегії до управління захистом інформації, тобто адміністратором системи, або адміністратором безпеки.
- Реєстрація в системі здійснюється автоматично, а дані захищені від модифікації та видалення сторонніми користувачами.
- Спроби несанкціонованих дій з інформацією, що становить комерційну або державну таємницю, повинна супроводжуватися повідомленням про це адміністратору безпеки, або системному адміністратору.

2.2 Моделювання структури підприємства

Даний проект спрямований на автоматизацію розгортання системи централізованого керування і захисту комерційної таємниці підприємства. Для моделювання бази виконання даного проекту обраний офіс малого архітектурного підприємства, який має в собі невеликий штат співробітників різних відділів і структур. Можна відмітити, що програмна автоматизація може бути реалізована на базі будь-якого малого підприємства, незалежно від вибору.

Штат підприємства має в собі 10 співробітників, що знаходяться у 6 відділах, незалежно від їх розташування у межах офісу (табл.2.1).

FirstName	Lastname	Department
Maxim	Curcumia	Managers
Maryna	Olephirenko	Architects
Oleg	Dobryi	GAPs
Roman	Viraz	GAPs
Boris	Campbell	Architects
Nicholas	Mirosh	Lawyers
Olga	Mirovova	Architects
Vlada	Brajnick	Architects
Valerii	Kabakov	Accountants
Rodion	Pozdniakov	SysEng

Таблиця 2.1 – Структура підприємства

Діаграма структури домену відображає відділи підприємства і групи безпеки, які будуть сформовані для обраних департаментів (рис.2.2).

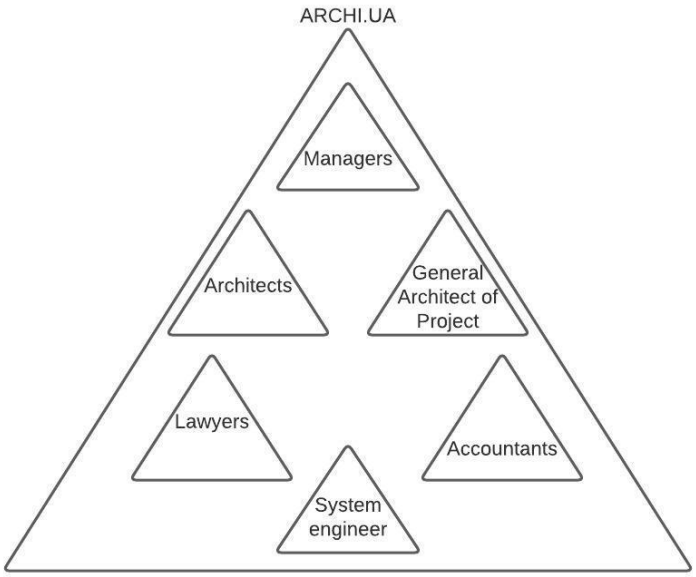

Рисунок 2.2 – Структура груп безпеки підприємства

Структури безпеки будуть відображати набір правил для відділу користувачів підприємства та об’єднувати їх у групи з різним рівнем допуску до директорій передачі інформації між департаментами.

3 РЕАЛІЗАЦІЯ СЕРВЕРНОЇ СИСТЕМИ НА БАЗІ КОНТРОЛЕРУ ДОМЕНУ ACTIVE DIRECTORY ТА СТВОРЕННЯ ЗАХИЩЕНОЇ СИСТЕМИ ОБМІНУ ІНФОРМАЦІЄЮ

3.1 Розгортання контролеру домену Active Directory

Active Directory, як контролер домену працюватиме під управлінням Windows Server 2019, основна його задачі – це збереження даних каталогів і управління взаємодією користувача та домену, такі як процеси авторизації користувача в системі і перевірка автентичності користувача у каталогах домену.

Контролер домену – це сервер, який містить в собі базу каталогів домену та служби, які дозволяють отримати доступ до даної бази. У процесі отримання доступу до бази використовується протокол LDAP. Він є відкритим протоколом, використовуваним для зберігання і отримання даних з каталогу з ієрархічною структурою. Зазвичай використовується для зберігання інформації про організацію, її активах і користувачів та доступу до каталогів X.500. Це простий протокол, що працює на TCP/IP, призначений для того щоб обробляти операції роботи з записами.

Контролер домену Active Directory (AD) складається із служб:

- Directory Services – служба організації бази даних каталогів.
- Certification Services – служба роботи із сертифікатами відповідності.
- Lightweight Directory Services – служба, створена для опрацювання додаткових додатків роботи екземплярів каталогів.
- Rights Management Services – служба організації і управління роботи з правами користувачів.
- Federations Services – служба налаштування довірчих відносин між організаціями у Active Directory.

Встановлення Active Directory:

1) Встановлення і розгортання служб налаштовується за допомогою сценарію написаного на Powershell.

Виконати за допомогою Powershell сценарій Install-forest.ps1 з правами адміністратора.

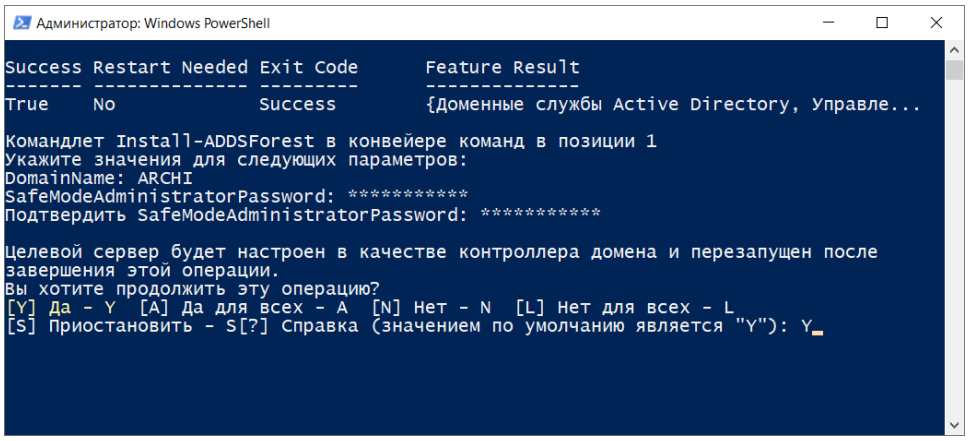

Рисунок 3.1 – Ініціалізація сценарію встановлення і налаштування AD

2) Виконується перевірка даних для встановлення нового лісу в AD. Після встановлення конфігурацій сервер перезавантажиться з встановленими службами. Встановлені обрані параметри для контролеру:

- При встановлення обрані параметри лісу узгодженні із структурою підприємства, домен має назву ARCHI.UA.
- DNS-делегування: Немає
- Директорія бази даних: C:\Windows\NTDS
- Режим роботи домену: WinThreshold
- Netbios ім’я: ARCHI
- Режим роботи лісу: WinThreshold
- DNS-сервер: Так
- Директорія файлів журналу: C:\Windows\NTDS
- Директорія SYSVOL: C:\Windows\SYSVOL

3.2 Налаштування DHCP та DNS

Служба DHCP – це роль сервера, що дозволяє мережевим пристроям автоматично отримувати IP-адресу та інші параметри, необхідні для роботи в мережі TCP / IP. Дана служба спрощує роботу системного адміністратора, тобто фахівцю не потрібно кожен раз вручну призначати ip-адреси новим комп'ютерам в мережі.

Служба DNS – це роль сервера, що дозволяє зіставити IP-адресу пристрою в мережі з його фізичною адресою. Дана служба виконує роль довідника для зіставлення імені з адресою.

Виділений сервер встановлює необхідні параметри для призначення мережевих адрес, для сервера обирається статистична IP-адреса.

Для коректної роботи DHCP та DNS ролі сервера обираються параметри та встановлюються області видачі адрес:

- Статистична IP-адреса: 10.12.34.49
- Основний шлюз: 10.12.34.250
- Початок області для DHCP: 10.12.34.50
- Кінець області для DHCP: 10.12.34.199
- Маска підмережі: 255.255.255.0

Сценарій Install_dhcp_staticip_renamepc.ps1 розгортає служби і сервер майже налаштований, але вже готовий до повноцінного використання (рис3.2). Домен не має фізичних користувачів системи, крім адміністратора.

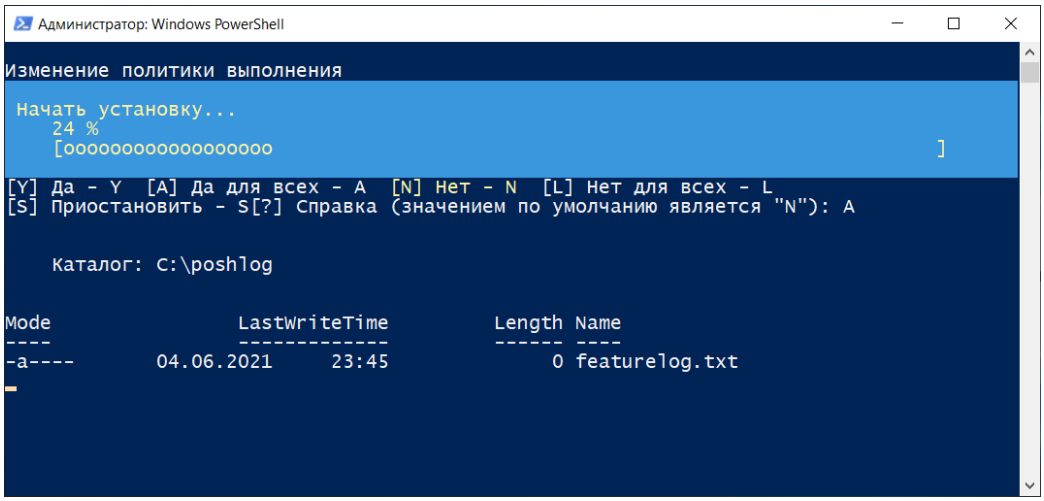

Рисунок 3.2 – Встановлення сценарію налаштування DHCP та DNS-ролі

Для автоматичного отримання IP-адрес станціям-користувачам системи створюється пул адрес, що не зачіпає робочі адреси важливих пристроїв домену. Такими пристроями можуть виступати сервер, шлюз мережі, мережеві пристрої виводу та вводу інформації (принтери, сканери і тд.). Область адрес для оренди (рис.3.3) у мережі не включає в себе пристрої із статистичними адресами серверу для уникнення конфлікту.

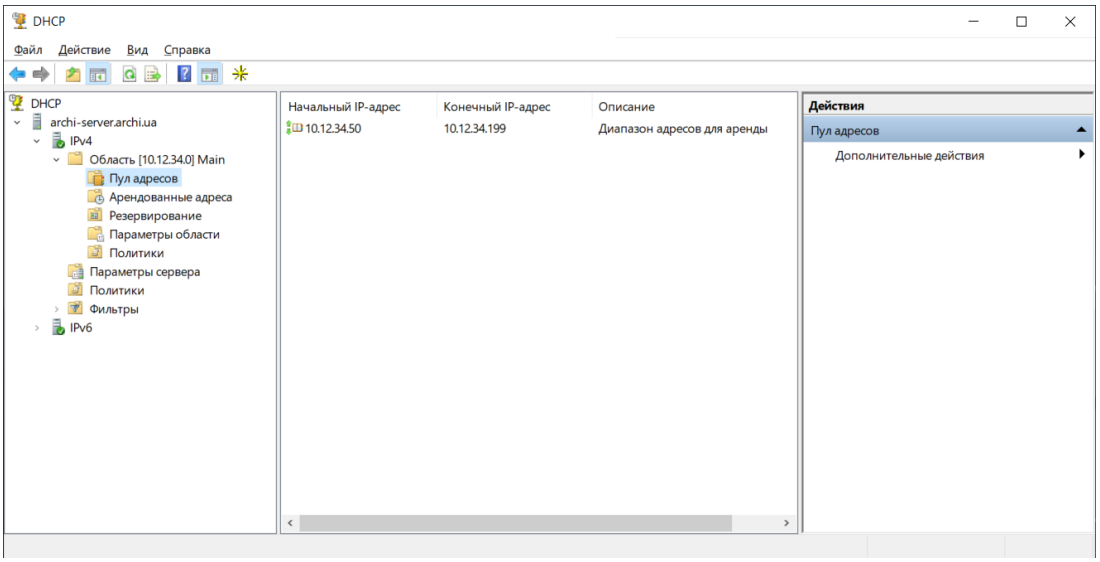

Рисунок 3.3 – Діапазон IP-адрес для видачі кінцевим пристроям

Служба DNS зіставляє значення орендованих IP-адрес пристроїв у мережі з їх назвою, виступаючи адресною книгою для серверу. Для наочності, орендованій IP-адресі 10.12.34.50 (рис.3.4) присвоєна назва кінцевого пристрою, тобто робочої станції клієнта з назвою “Archi-Workstation-1”.

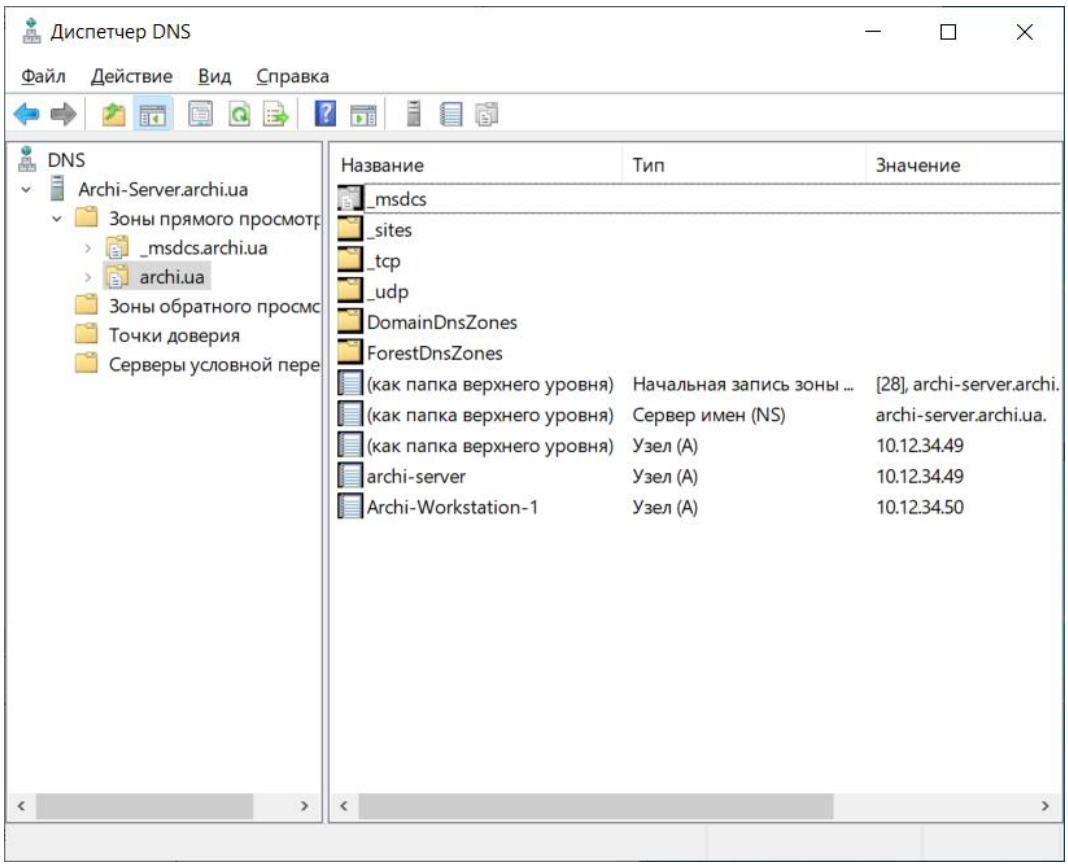

Рисунок 3.4 – Диспетчер DNS

Даний етап завершує налаштування ролей серверу (рис.3.5), мінімально-необхідні служби для опрацювання користувачів системи відповідають нормам адміністрування системи для подальшого додавання користувачів і розподілу їх по групам безпеки задля встановлення додаткових налаштувань.

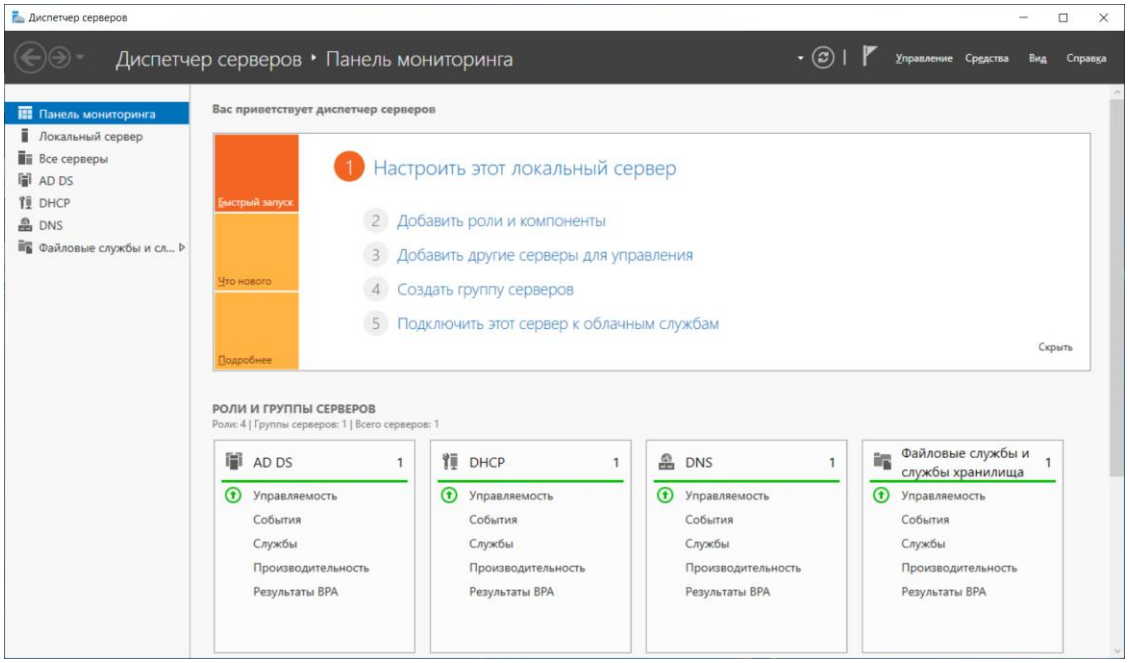

Рисунок 3.5 – Встановлені служби AD, DNS, DHCP

3.3 Додавання користувачів у домен та налаштування груп безпеки

В Active Directory існує два типи груп – безпеки і поширення.

Група поширення – застосовується для створення груп поштових розсилок. Лист відправлений на групу поширення дійде всім користувачам групи. Це група не призначена для роботи з наданням доступу на ресурси.

Група безпеки – застосовується для управління безпеки доступу до ресурсів за допомогою списків правил “ACL”. Тобто якщо ви хочете для мережевої папки створити групу, для цього необхідно створити групу безпеки. Так само за допомогою групи безпеки можна зробити поштову розсилку, але це не рекомендується робити оскільки для цього є група поширення.

Група безпеки використовується для розподілу правил користування ресурсами для певних користувачів. Найпоширеніше використання даних груп – розмежування робочої області та області спільних директорій у домені.

Крім груп існує три області дії для кожної групи:

- Локальна в домені – використовується для управління дозволами доступу до ресурсів в межах всього домену.
- Глобальна група – використовується для визначення колекції об'єктів доменів на підставі бізнес-правил і управління об'єктами, які вимагають щоденного використання.
- Універсальна група – Рекомендується використовувати в лісах з безліч доменів. За допомогою неї можна визначати ролі та управляти ресурсами, які розподілені на декількох доменах.

Згідно ієрархії користувачів домену у підприємства є список співробітників, їх посад та департаментів. Даний список потрібно імпортувати у файл CSV та інтерпретувати його у Active Directory, аби занести дані про користувачів домену та групи, до яких вони належатимуть. Дані про користувачів можна побачити у ДОДАТКУ Б.

Спираючись на структуру підприємства та відділи, в яких працюють співробітники, автоматизований сценарій Add-users.ps1 додасть до домену ARCHI.UA: 10 співробітників, і створить 6 відділів (рис.3.7).

Дані, які будуть імпортовані не є конфіденційними по відношенню до співробітників в середині підприємства. Єдина конфіденційна інформація, що буде імпортована – це паролі користувачів. Імпортований пароль доступу згенерований, як одноразовий і за замовчуванням при першій авторизації в домені виділений сервер надасть повідомлення що до зміни паролю (рис.3.6.).

Паралельно імпортуванню користувачів, на основі відділів буде створено 6 груп безпеки для кожного відділу (рис.3.7) – це дасть змогу далі налаштувати області передачі даних у домені між відділами.

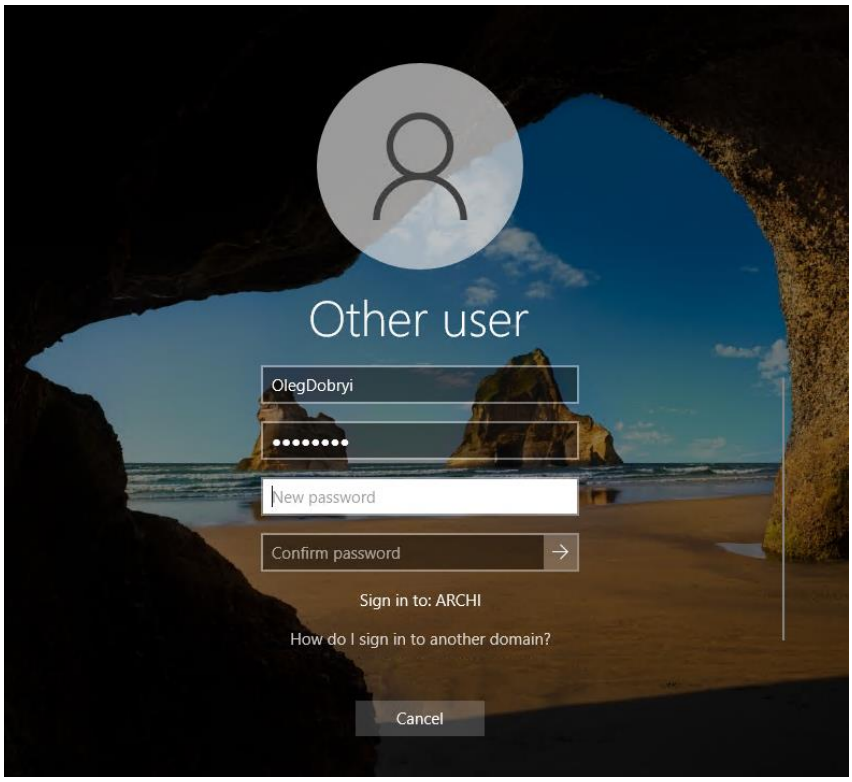

Рисунок 3.6 – Зміна тимчасового паролю за замовчуванням

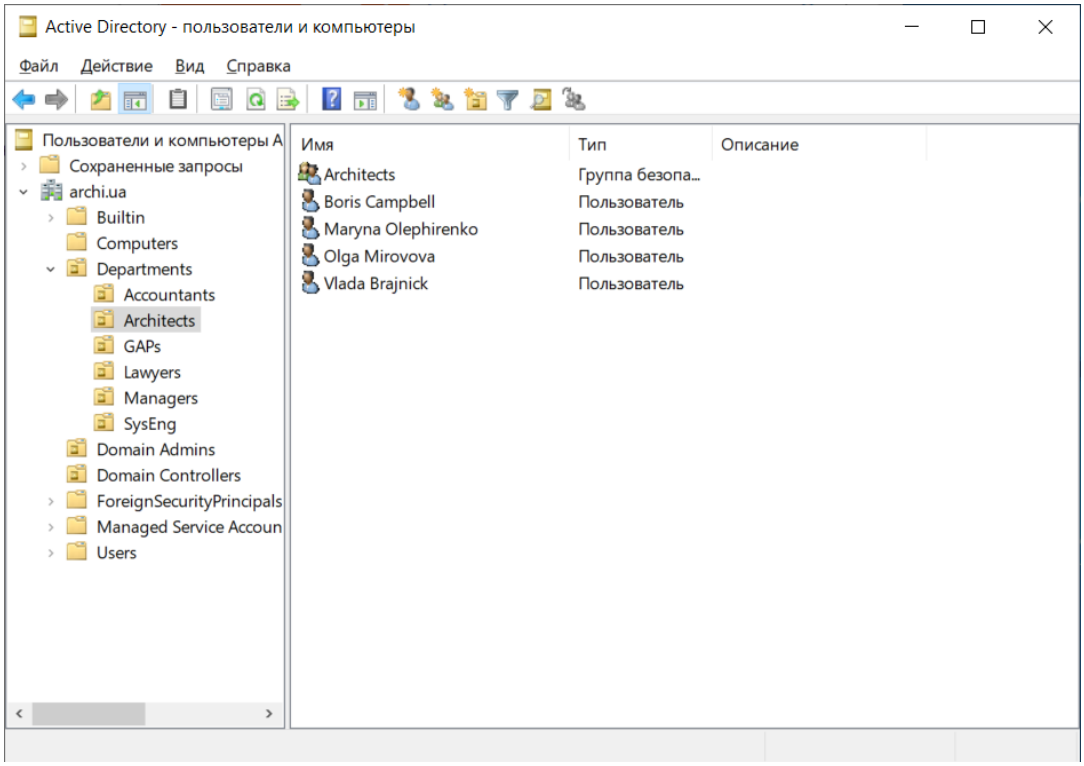

Рисунок 3.7 – Відділи з користувачами та групами безпеки

До Active Directory додані робочі станції співробітників (рис.3.8), кожна робоча станція не залежить від конкретного користувача.

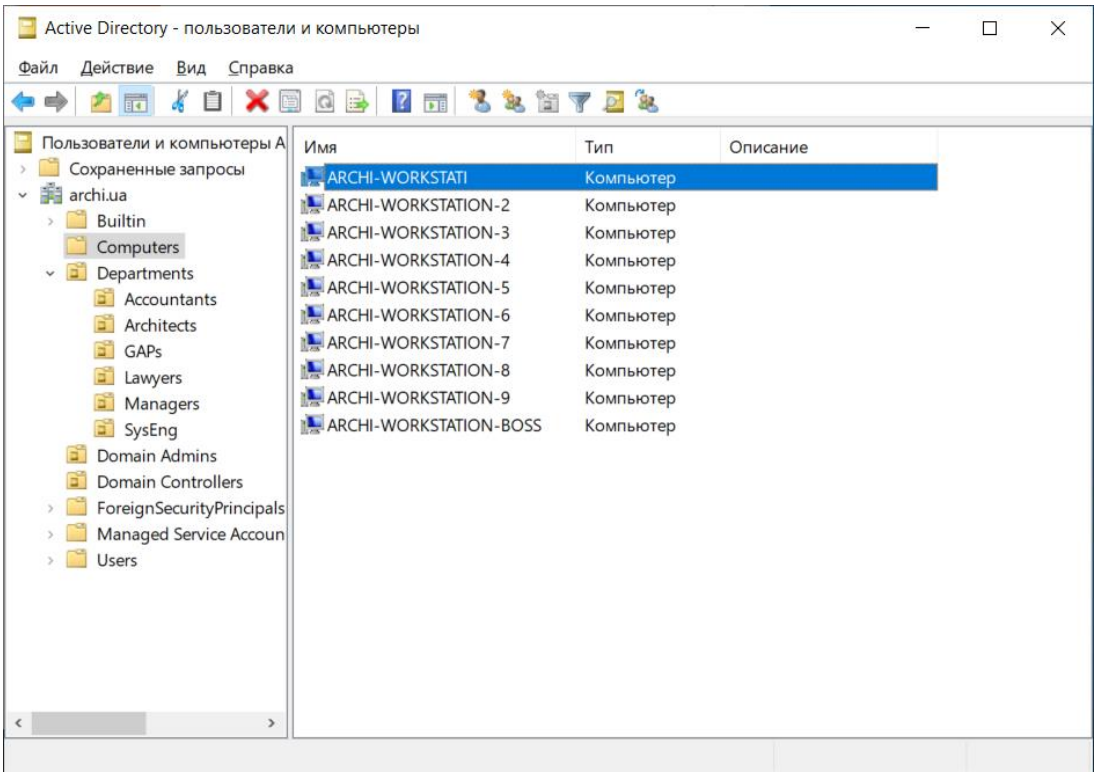

Рисунок 3.8 – Робочі станції мережі

3.4 Створення системи спільних директорій

Будь-яка директорія, для якої організований загальний доступ, включаючи папку DFS, може бути опублікована в Active Directory. Публікація полягає в створенні в Active Directory об'єкта типу “Загальна директорія”. Сама публікація не має на увазі автоматичне забезпечення загального доступу до директорії, тому процес публікації складається з двох етапів:

- Забезпечення загального доступу до папки.
- Її публікація в Active Directory у вигляді об'єкта каталогу.

Для кожної директорії забезпечується обмежений доступ, як правило для групи безпеки, де одна група має дозвіл на створення та редагування, а інша

лише на читання, або зовсім не має доступу. Така директорія розташовується на виділеному сервері мережі та публікується для кінцевих користувачів.

Логіка доступу до даних директорій відповідає структурному поділу підприємства, що забезпечує спільний доступ до ресурсів для ведення і розробки проектів, або спільного доступу до інформації (рис.3.9) від іншого департаменту підприємства, але обмежує доступ між ними за логічною структурою обміну конфіденційною інформацією, не призначеною для витоку між відділами.

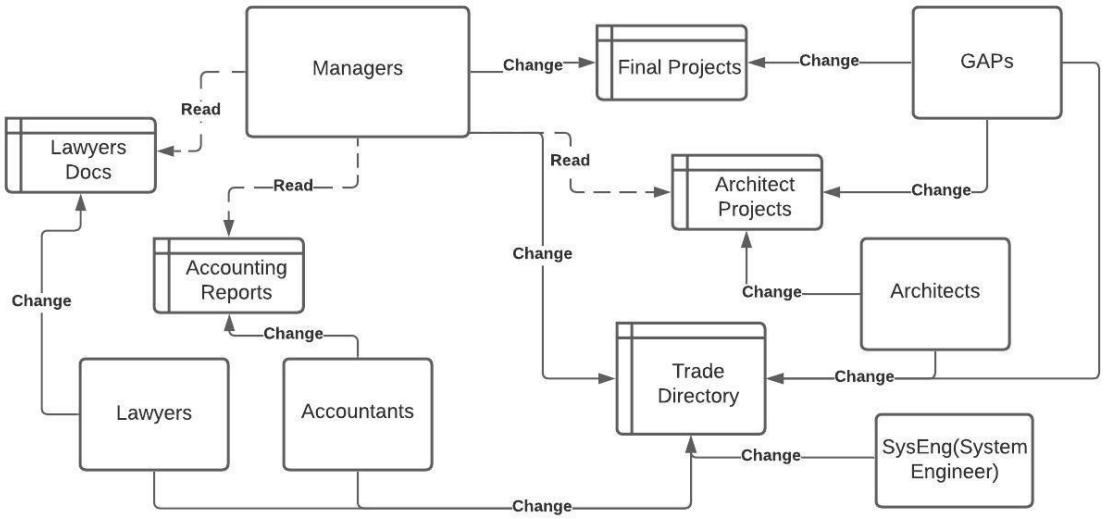

Рисунок 3.9 – Логіка доступу груп безпеки до директорій

Кожна група безпеки має індивідуальні налаштування політики доступу до директорії за допомогою автоматизованого сценарію створені директорії і рівні допуску до них:

- Директорія Lawyers Docs – це директорія для обміну документами юридичного відділу між дирекцією підприємства. Група безпеки Lawyers має змогу створювати і змінювати вміст директорії, група Managers має змогу на ознайомлення документації без можливості її редагування.
- Директорія Accounting Reports – це директорія для обміну звітами з

фінансів підприємства. Група безпеки Accountants має змогу створювати і змінювати вміст директорії, група Managers має змогу на ознайомлення документації без можливості її редагування.

- Директорія Architect Projects – це директорія для розробки проектів архітектури. Групи безпеки Architects і GAPs мають змогу створювати і змінювати вміст директорії, група Managers має змогу на ознайомлення документації без можливості її редагування.
- Директорія Final Projects – це директорія для здачі і фінального ухвалення готових робочих проектів. Групи безпеки Managers і GAPs мають рівноправні права доступу на читання та редагування вмісту директорії.
- Директорія Trade Directory – це директорія, що виступає спільною для всіх зоною обміну файлів і документів. Групи безпеки Managers, Architects, GAPs, SysEng, Accountants і Lawyers мають рівноправні права доступу, читання і редагування файлів.

Встановлення автоматизованого сценарію File-shared-system.ps1 створює директорії на виділеному сервері, що має шлях: \\ARCHI-SERVER\SharedFolders (рис.3.10).

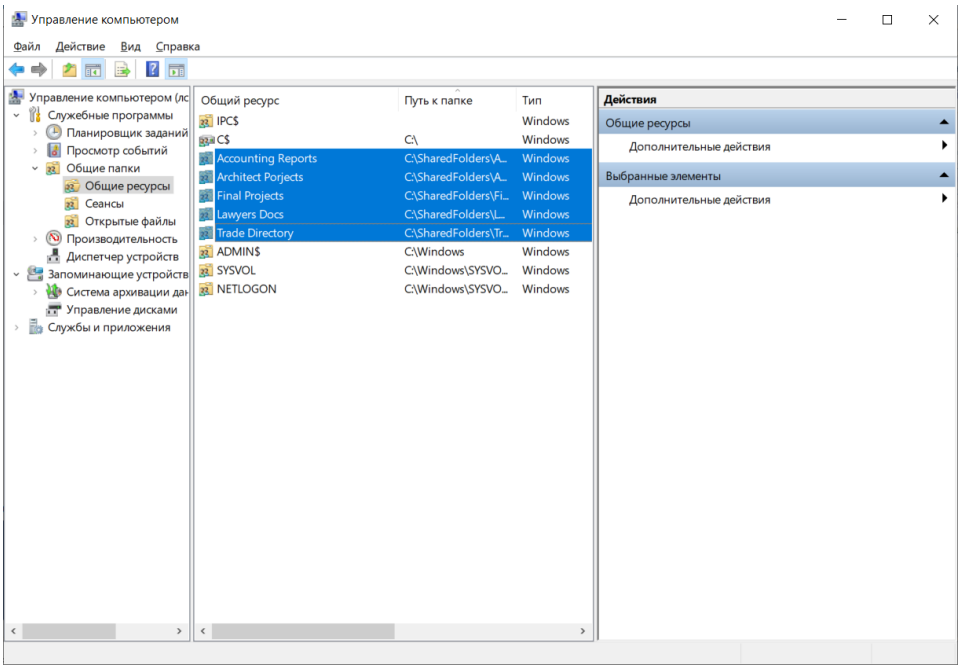

Рисунок 3.10 – Виділені директорії для спільного доступу

3.5 Налаштування сповіщень про несанкціоновану зміну адміністраторів AD

Система сповіщення (Тригер) - це функціонал, що дозволяє прив'язати завдання планувальника до будь-якої події в журналах системи. Завдяки цій можливості адміністратор може на будь-яку подію Windows призначити виконання певного сценарію або відправку повідомлення електронною поштою.

Можливість запуску завдань при настанні певних подій Windows заснована на тісній інтеграції Task Scheduler і Event Viewer. Призначити завдання планувальника на будь-яку подію Windows можна прямо з консолі журналу перегляду події (Event Viewer). Як реакція на подію, що відбулася планувальник може запустити скрипт або відправити поштове повідомлення адміністратору (або будь-якому іншому користувачу).

Завдання даної автоматизації - налаштувати оповіщення адміністратора безпеки про додавання в списки адміністраторів домену користувача Active Directory.

Сценарій для налаштування даної автоматизації складається з чотирьох частин:

- Створення еталонного списку адміністраторів домену на виділеному сервері підприємства (Select-domain-admins.ps1).
- Сценарій порівняння еталонного списку с фактичним списком адміністраторів (Comparison-domain-admins.ps1).
- Тригер-сповіщення, що реагує на не співпадіння списків Monitoring-domain-admins-trigger.ps1.
- Список моніторингу останніх авторизованих користувачів (Domain-

login-history.ps1).

Створення еталонного списку формується на основі звернення до підрозділу адміністраторів Domain Admins – це список користувачів, що мають привілеї до додавання, корегування доменними службами. Фактично даний тип користувачів має рівень Unlimited User, що дозволяє йому робити будь-які маніпуляції з доменом, файлами та користувачами. За замовчуванням цей користувач є Адміністратором, до даного типу користувачів додана група безпеки SysEng, що відповідає структурному підрозділу Системних адміністраторів та Адміністраторів безпеки підприємства.

Виконання автоматизованого сценарію створення еталонного списку адміністраторів (рис.3.11). Даний список зберігається у файловій системі серверу за шляхом: \\ARCHI-SERVER\C:\PS та оновлюється в ручному режимі при кожному запуску сценарію Select-domain-admins.ps1.

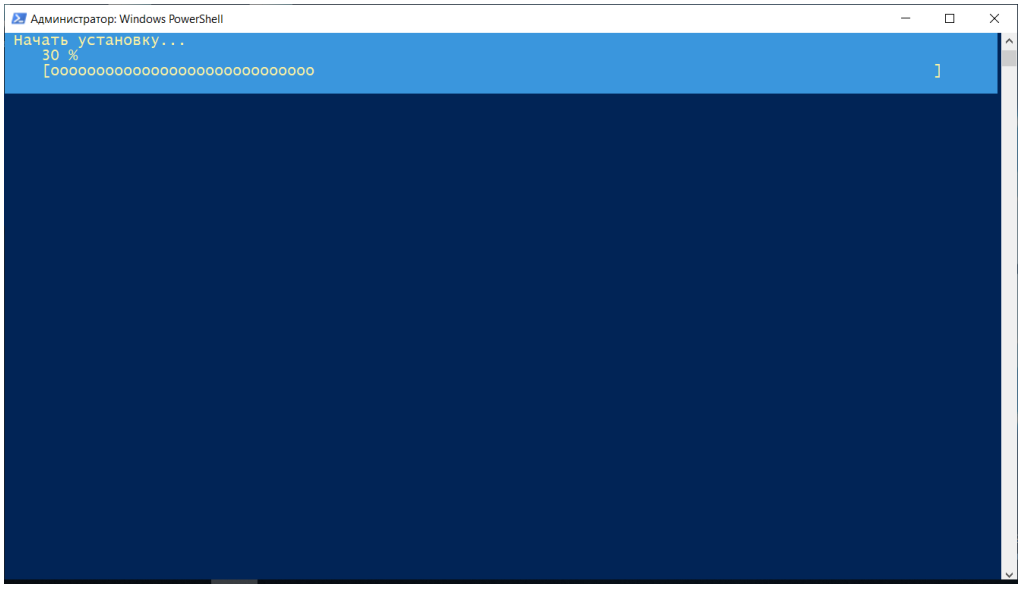

Рисунок 3.11 – Виділені директорії для спільного доступу

Результатом виконання сценарію є створення списку DomainAdminsCurrent (рис.3.12).

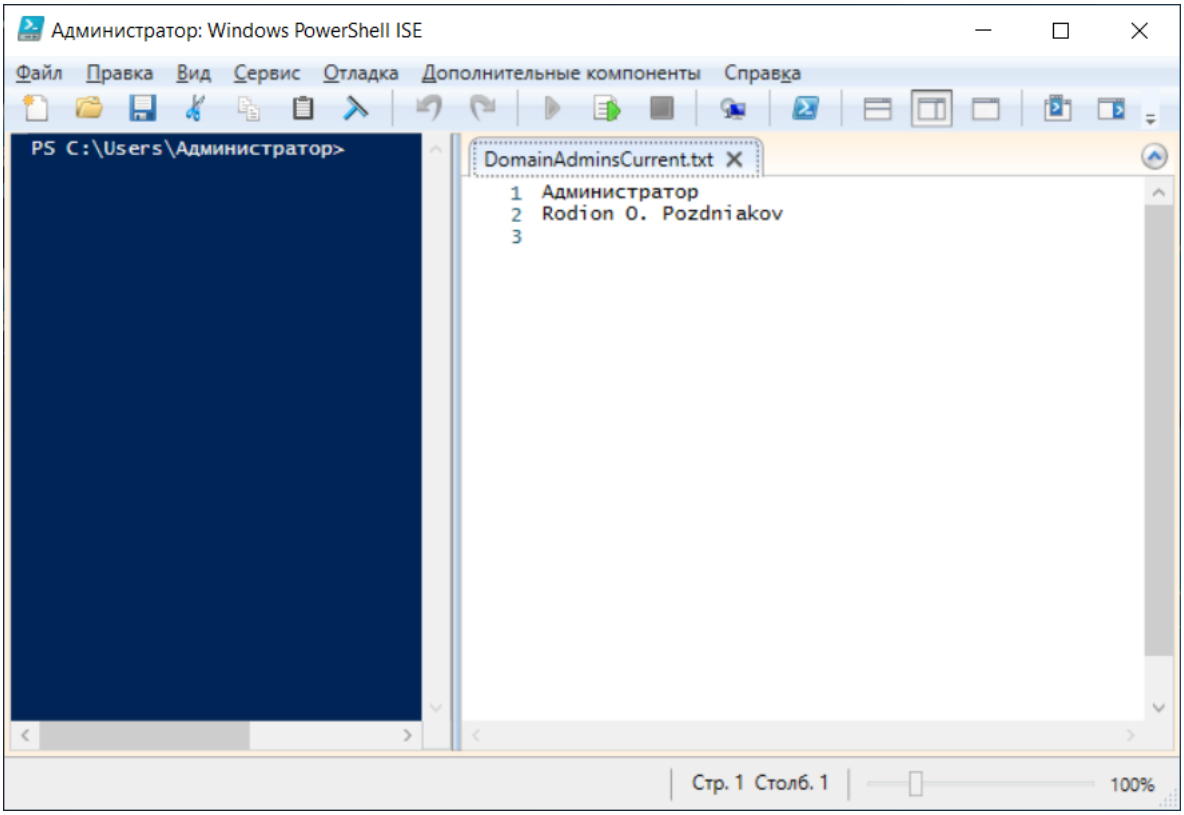

Рисунок 3.12 – Еталонний список адміністраторів домену

Наступною частиною є сценарій порівняння с еталонним списком фактичного списку адміністраторів домену Comparison-domain-admins.ps1. При запуску даної автоматизації буде створений список адміністраторів DomainAdmins та порівняння його зі списком DomainAdminsCurrent. При не співпадінні списків сценарій сповістить адміністратора безпеки за допомогою повідомлення. Також даний сценарій оновлює еталонний список адміністраторів кожен день на початку робочого дня.

Даний сценарій прив’язаний к тригеру Windows (рис.3.13), що створюється за допомогою скрипту Monitoring-domain-admins-trigger.ps1 і запускає перевірку на правильність списків кожну хвилину.

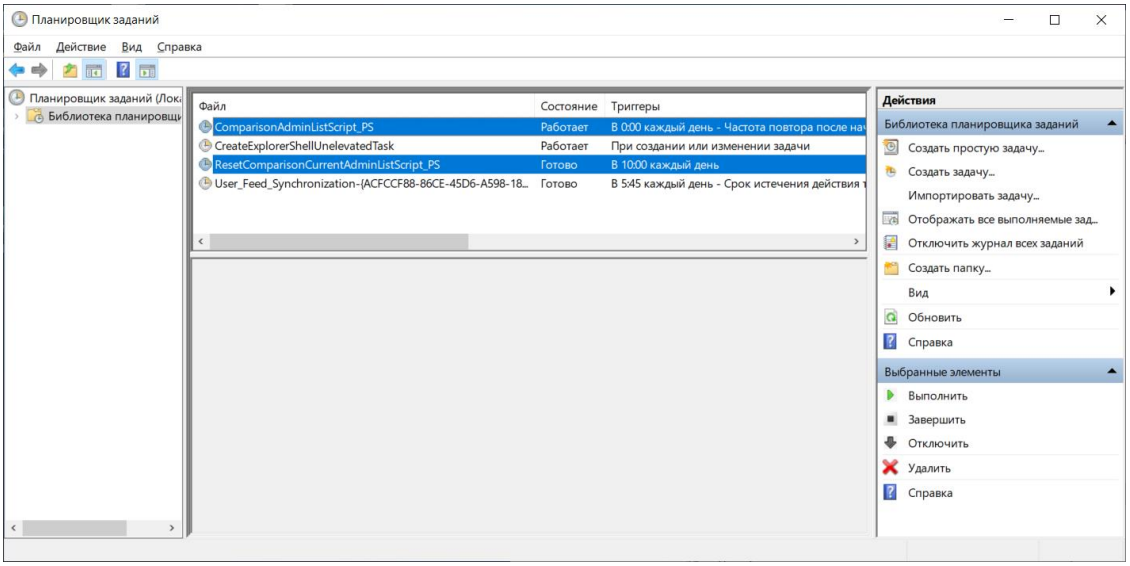

Рисунок 3.13 – Тригери запуску сценаріїв

Для перевірки системи сповіщень до списку адміністраторів системи у Active Directory був доданий новий користувач Hack Villiams. Тригер починає зчитувати фактичний список і порівнювати його з еталонним. Знайшовши не співпадіння тригер запускає функцію сповіщення за допомогою повідомлення (рис.3.14) і запускає фінальний скрипт для моніторингу останніх підключень до домену (рис.3.15).

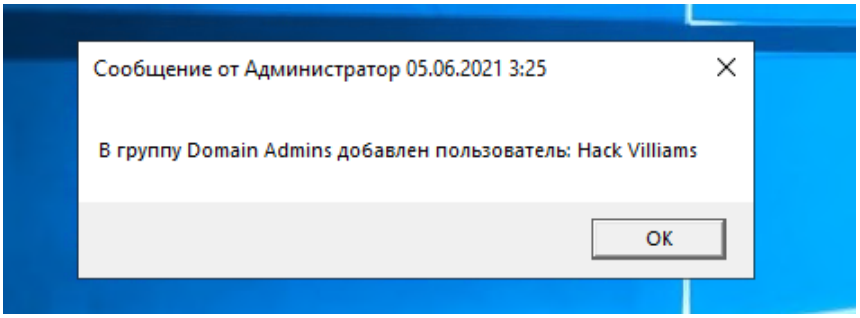

Рисунок 3.14 – Системне сповіщення додавання нового користувача

Паралельно сповіщенню починається пошук і фільтрація записів контролеру домену з останніх підключень до нього за подіями видачі квитка Kerberos при аутентифікації користувача, спроби скористуватися спільними

директоріями, які недоступні користувачеві, директорії і файли, до яких було отримано доступ, можливі привілегії, які були додані користувачеві та привілегії, отримані з причини додавання користувача в групу безпеки.

Можна отримати всі події з журналу контролера домену, відфільтрувати їх за потрібним кодом (EventID) і вивести дані про час, коли користувач розпізнаватися у домені, і комп'ютері, з якого виконаний вхід.

Перегляд подій – це компонент, що входить до складу операційних систем сімейства Windows, який надає змогу адміністраторам переглядати список подій на локальному комп'ютері або на віддаленій машині чи сервері. Список цих подій фільтрується і кожна подія має свій унікальний номер – EventID. Даний номер дозволяє ефективно налаштувати моніторинг подій у налаштованих системах та допомагає у проведенні аудиту корпоративних чи приватних мережах.

Аудит безпеки Windows – це технічні засоби і заходи, спрямовані на реєстрацію і систематичний регулярний аналіз подій, що впливають на безпеку інформаційних систем підприємства. Технічно, аудит безпеки в Windows реалізується через налаштування політик аудиту і налаштування аудиту об'єктів. Політика аудиту визначає які події і для яких об'єктів будуть генеруватися в журнал подій безпеки. Регулярний аналіз даних журналу безпеки відноситься до організаційних заходів, для підтримки яких може застосовуватися різне програмне забезпечення. У найпростішому випадку можна користуватися повним переглядом подій, але через повний запис усіх змін у системі ці події становлять великий об’єм даних, який складно обробити мануальним методом.

Оскільки в домені в майбутньому може бути кілька контролерів домену і потрібно отримати історію входів користувача з кожного з них, потрібно скористатися Командлети Get-ADDomainController (з модуля AD для Windows PowerShell).

PowerShell скрипт дозволяє отримати всі події входу користувача в домен AD зі всіх контролерів домену, спроби скористуватися директоріями і отриманими привілегіями. На виході отримується моніторинг-таблиця з історією і необхідною інформацією для визначення порушника.

Для отримання даного списку потрібна фільтрація по журналу контролеру домена за кодом EventID, що має значення відповідні типу запису у журналі:

- 4728 – долучення учаснику в захищену локальну групу безпеки.
- 5140 – реалізований доступ до спільного мережного ресурсу.
- 4768 – спроба підключення до домену.
- 4673 – виклик користувачем привілейованої служб.
- 4663 – проведена спроба доступу до об'єкта.

Паралельно з отриманням даних останніх подій зв’язаних з несанкціонованим доступом користувачів система повинна перевірити свої дані на наявність сторонніх втручань, з метою пошкодження працездатності сервера або домену. Наступні EventID позначають події, при яких зловмисник намагається або намагався нашкодити працездатній системі:

- 4612 – внутрішні ресурси, виділені на чергу аудиторських повідомлень, вичерпані, що призведе до втрати деяких аудитів.
- 5038 – цілісність код визначає, що хеш-файл файлу не є дійсним. Файл може бути пошкоджений через несанкціоновану модифікацію або недійсний хеш може вказувати на потенційну помилку дискового пристрою.
- 5057 – помилка криптографічної примітивної операції.
- 5060 – помилка операції перевірки.
- 6410 – цілісність коду визначає, що файл не відповідає вимогам безпеки для завантаження в процес.
- 1102 – видалення журналу безпеки.

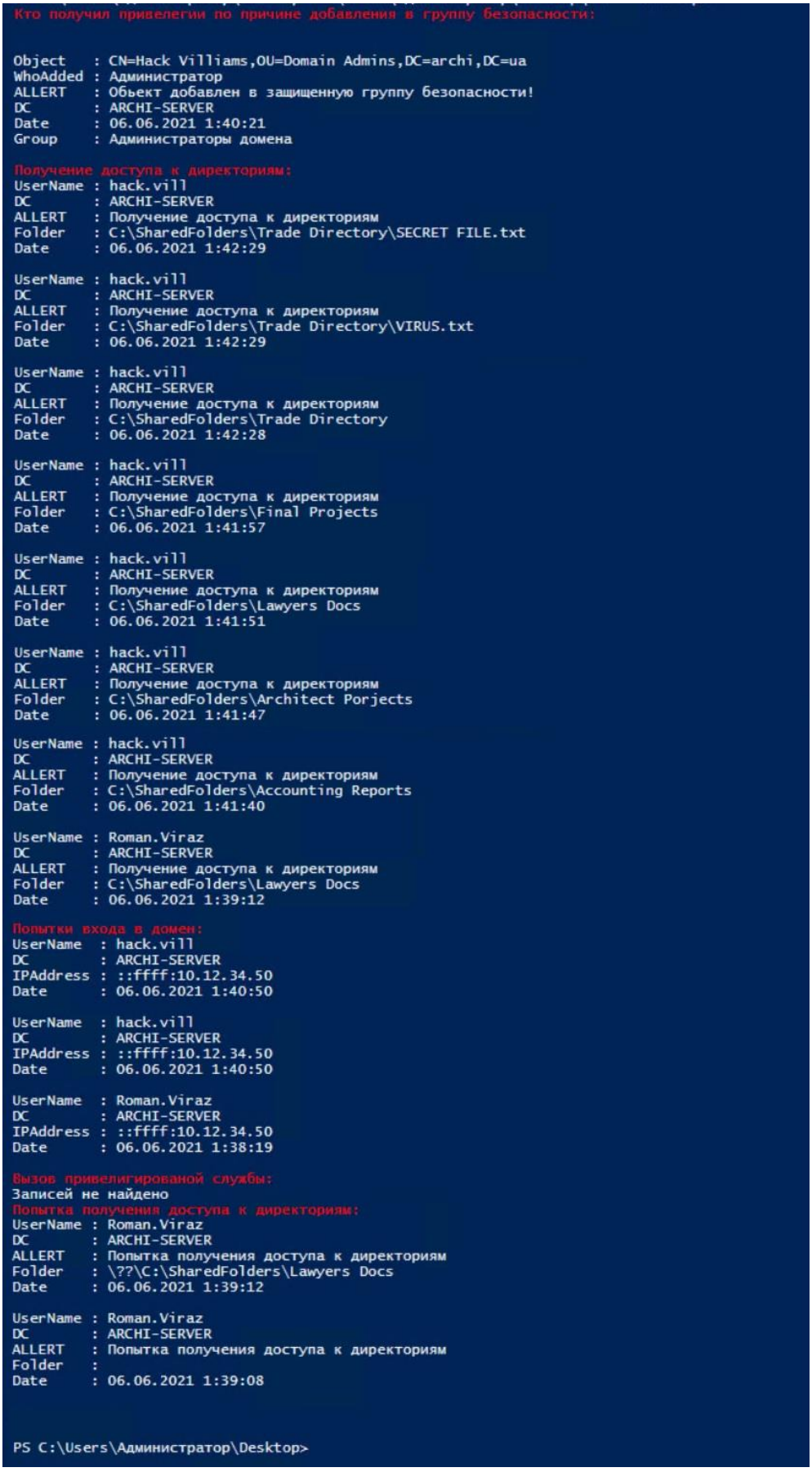

Рисунок 3.15 – Список останніх авторизацій

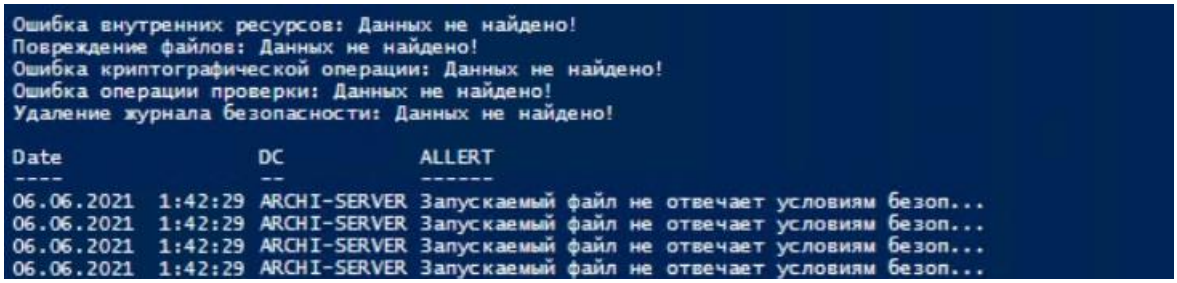

Рисунок 3.16 – Список зміни даних системи

Далі адміністратор домену матиме можливість визначити порушника системи та відключити його, або видалити зі списків керування доменом або з груп, що мають привілегії.

За даними результату роботи першого скрипту моніторингу (рис. 3.15) можна визначити, що несанкціонований користувач був доданий в групу безпеки адміністраторів домену, здійснив спробу входу в домен та отримав доступ до визначених директорій і файлів. Також можна побачити, що перед цим, з кінцевої робочої станції, була здійснена спроба доступу до директорій, але доступ був відхилений, після, з пристрою з тою ж самою фізичною адресою у мережі і проводилась атака на домен. Паралельно було перевірено систему серверу на наявність порушень кінцевих вказівок системи (рис. 3.16), яке виявило, що на пристрої, що вступає контролером домену намагався запуститись несанкціонований програмний код.

Адміністратор системи, за допомогою даних автоматизацій має змогу дослідити введені порушення системи, та виявити несанкціоновані дії, дослідивши історію записів журналу.

ВИСНОВКИ

У ході виконання випускної роботи були розглянуті і досліджені методи сучасного адміністрування систем на базі виділеного сервера для впровадження централізованого захисту комерційної таємниці і керування мережею підприємства. Була запропонована модель для впровадження серверної системи на базі Active Directory з прикладом її реалізації на базі малого архітектурного підприємства. Була розроблена модель організації директорій для обміну інформацією на базі розроблених структурних підрозділів і реалізована система сповіщень адміністратора безпеки на наявність несанкціонованих підключень до мережі підприємства, а також система моніторингу порушень.

Для роботи була обрана модель організації підприємства за допомогою системи домену та лісу. Та створені автоматизовані сценарії для налаштування розгортання і системи, ролі серверу для функціонування і налагодження робочих процесів підприємства.

Також було виконано тестове встановлення робочої моделі системи каталогів AD та тригерів для моніторингу сторонніх підключень до контролеру домену. Розроблені рішення були встановлені на виділеному серверів на базі операційної системи Windows Server 2019 у мережі офісу підприємства.

Аналіз розробленої системи відповідає положенням Кабінету Міністрів України про правила захисту і обміну комерційної інформації на підприємстві.

Розроблена у роботі система відповідає вимогам до систем захисту інформації, які висуваються "Правилами забезпечення захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах", затверджених Постановою Кабінету Міністрів України від 29.03.2006 №373. Система доменних служб Active Directory дозволяє реалізувати вимоги щодо обов’язкової реєстрації користувачів, подальшу авторизацію та автентифікацію у системі. За допомогою налаштувань груп

безпеки кожен з користувач має можливість виконувати лише ті дії, які йому були дозволені відповідно до політики безпеки.

Розроблений скрипт моніторингу дозволяє здійснювати спостереження за діями, пов’язаних з обробкою інформації в системі. Будь-яка спроба несанкціонованих дій з інформацією, що становить комерційну таємницю для підприємства одразу супроводжується інформуванням адміністратора безпеки підприємства. Крім того, розроблений скрипт дозволяє відслідковувати:

- результати ідентифікації та автентифікації користувачів у системі;
- результати виконання користувачем системи операцій з обробки інформації;
- спроби користувачів виконати несанкціоновані дії з інформацією;
- факти надання та позбавлення користувачів права доступу до певної інформації;
- результати перевірки цілісності засобів захисту інформації.

СПИСОК ЛІТЕРАТУРИ

1. Samba Docs And Books. Learning. [Електронний ресурс] – Режим доступу до ресурсу:

https://www.samba.org/samba/docs/ (дата звернення: 31.05.2021).

2. Хілл Операційна система Ubuntu Linux / Хілл, Б. Мако. - М .: Тріумф, 2008. - 384 с.

3. Active Directory, 4th Edition / [B. Desmond, J. Richards, R. Allen та ін.].,

2008. – 866 с

4. Оліфер, В. Комп'ютерні мережі. Принципи, технології, протоколи: Підручник / В. Олефір, Н. Оліфер. - СПб .: Пітер, 2016. - 318 с.

5. Мінас, Марк Windows Server 2012 R2. Повне керівництво. Том 1. встановлювати і налаштовувати сервера, мережі, DNS / Марк Мінас і ін. - Москва: 2017. - 960 с.

6. Windows Server documentation. [Електронний ресурс] – Режим доступу до ресурсу:

https://docs.microsoft.com/uk-ua/windows-server/ (дата звернення: 18.05.2021).

7. Васькевич Стратегії клієнт / сервер / Васькевич, Девід. - М .: Київ:

Діалектика, 2016. - 384 с.

8. Ренд Морімото, Кентон Гардін'єр, Майкл Ноел, Джо Кока. Microsoft Exchange Server 2003. Повне керівництво = Microsoft Exchange Server 2003 Unleashed. — М. : «Вильямс», 2006. — С. 1024

9. Станек У. Microsoft Windows Server 2012 R2: зберігання, безпека,

мережеві компоненти. Довідник адміністратора / У. Станек. - СПб .: BHV, 2015. – 416 с:

10. Розгортаємо Active Directory усіма можливими методами. [Електронний ресурс] – Режим доступу до ресурсу: https://habr.com/ru/company/ultravds/blog/480776/ (дата звернення 18.05.2021).

11. Крейг Хант. TCP / IP. Мережеве адміністрування, 3-е видання. - Пер. з англ. - СПБ: Сім-вол-Плюс, 2007. - 816 с.

12. Active Directory Security Groups. [Електронний ресурс] – Режим доступу до ресурсу: https://docs.microsoft.com/uk-ua/windows/security/identity-protection/access-control/active-directory-security-groups дата звернення 18.05.2021).

13. Левін М. Як стати системним адміністратором: Самовчитель: Науково-популярне видання / Левін Максим. - М .: Пізнавальна книга плюс, 2001. - 320 с.

14. PowerShell Documentation. [Електронний ресурс] – Режим доступу до ресурсу: https://docs.microsoft.com/en-us/powershell/ (дата звернення 21.05.2021)

15. Холмс Лі Windows PowerShell. Кишеньковий довідник; ЕКОМ Паблишерз -, 2013. - 160 с.

16. Постанова Кабінету міністрів України про затвердження Правил забезпечення захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах від 29 березня 2006 р. №373 [Електронний ресурс] – Режим доступу до ресурсу: https://zakon.rada.gov.ua/laws/show/373-2006-п#Text (дата звернення 05.05.2021).

ДОДАТОК А – СКРИПТ POWERSHELL ДЛЯ РОЗГОРТАННЯ ACTIVE DIRECTORY

Install-forest.ps1:

```
# import modules
Import-Module ServerManager
Import-Module ActiveDirectory

# install role AD DS
Add-WindowsFeature -Name AD-Domain-Services -
IncludeAllSubFeature -IncludeManagementTools
# import module ADDSDeployment
Import-Module ADDSDeployment

# install new forest
Install-ADDSForest `
# do not include delegation
-CreateDnsDelegation:$false `
# database path of AD
-DatabasePath "C:\Windows\NTDS" `
# domain functional mode
-DomainMode "WinThreshold" `
# set the domain name
-DomainName "archi.ua" `
# set short NetBIOS name
-DomainNetbiosName "ARCHI" `
# set the forest operating mode
-ForestMode "WinThreshold" `
# install the DNS server
-InstallDns:$true `
# Set the path to NTDS
-LogPath "C:\Windows\NTDS" `
# if a reboot is required, then reboot
-NoRebootOnCompletion:$false `
# set the path to the SYSVOL folder
-SysvolPath "C:\Windows\SYSVOL" `
-Force:$true
```

Install-dhcp-statip-renamepc.ps1:

```
     #import modules
Import-Module ServerManager
Import-Module ActiveDirectory


#rename the computer
$newname = “Archi-Server”
$dnsdomain = "archi.ua"
Rename-Computer -NewName $newname -force
#install features
```

```
$featureLogPath = "c:\poshlog\featurelog.txt"
New-Item $featureLogPath -ItemType file -Force
$addsTools = "RSAT-AD-Tools"
Add-WindowsFeature $addsTools
Get-WindowsFeature | Where installed >>$featureLogPath

#set static IP address
$ipaddress = "10.12.34.49"
$ipprefix = "24"
$ipgw = "10.12.34.250"
$ipdns = "10.12.34.49"
$ipif = (Get-NetAdapter).ifIndex
$zero = "10.12.34.0"
$startrange = "10.12.34.50"
$endrange = "10.12.34.199"
$mask = "255.255.255.0"
New-NetIPAddress -IPAddress $ipaddress -PrefixLength
$ipprefix `
-InterfaceIndex $ipif -DefaultGateway $ipgw

#install DHCP Role
Install-WindowsFeature -Name DHCP -IncludeManagementTools
Add-DhcpServerInDC -DnsName "archi.ua" -IPAddress 10.12.34.49
#set ip scope
Add-DHCPServervScope -EndRange $endrange -Name Main -
StartRange $startrange -SubnetMask $mask -State Active -
ComputerName $newname
Set-DHCPServervOptionValue -ComputerName $newname -DnsServer
$ipdns -DnsDomain $dnsdomain -Router $ipgw
Set-DHCPServervOptionValue -ComputerName $newname -ScopeId
$zero -DnsServer $ipdns -DnsDomain $dnsdomain -Router $ipgw
```

File-shared-system.ps1

```
#Create shared folder Accountants to Managers
New-SMBShare -Name Accounting Reports `
             -Path C:\SharedFolders `
             -FullAccess Accountants `
             -ChangeAccess Managers `
             -ReadAccess Users
#Create shared folder Lawyers to Managers
New-SMBShare -Name Lawyers Docs `
             -Path C:\SharedFolders `
             -FullAccess Lawyers `
             -ReadAccess Managers
#Create shared folder Architects to GAPs with Managers
New-SMBShare -Name Architect Porjects `
             -Path C:\SharedFolders `
             -FullAccess GAPs `
             -ChangeAccess Architects `
             -ReadAccess Managers
```

```
#Create shared folder GAPs to Managers
New-SMBShare -Name Final Projects `
             -Path C:\SharedFolders `
             -FullAccess GAPs `
             -ReadAccess Managers `
#Create shared folder for ALL GROUPS
New-SMBShare -Name Trade Directory `
             -Path C:\SharedFolders `
             -FullAccess GAPs, Managers, Architects,
Accountants, Lawyers, SysEng

$path = 'C:\Shared Folders'

Add-NTFSAudit -Path $path -AccessRights FullControl -Account
Everyone -AuditFlags Success -InheritanceFlags
ContainerInherit,ObjectInherit -PropagationFlags None
```

Select-domain-admins.ps1

```
#delete oldest version of list
Remove-item C:\PS\DomainAdmins.txt
#create new version of list
(Get-ADGroupMember -Identity "Администраторы домена" -
recursive).Name | Out-File C:\PS\DomainAdmins.txt
```

Comparison-domain-admins.ps1

```
#delete oldest version of list
Remove-item C:\PS\DomainAdminsCurrent.txt
#create new version of list
(Get-ADGroupMember -Identity "Администраторы домена" -
recursive).Name | Out-File C:\PS\DomainAdminsCurrent.txt
$oldadm=GC C:\PS\DomainAdmins.txt
$newadm=GC C:\PS\DomainAdminsCurrent.txt
#compare lists of domain administrators
$diff=Compare-Object -ReferenceObject $oldadm -
DifferenceObject $newadm | Select-Object -ExpandProperty
InputObject
write-host $diff
#result output window
$result=(Compare-Object -ReferenceObject $oldadm -
DifferenceObject $diff | Where-Object {$_.SideIndicator -eq
"=>"} | Select-Object -ExpandProperty InputObject) -join ", "
If ($result)
{msg * "В группу Domain Admins добавлен пользователь:
$result"
.\Unauthorized-access-monitor.ps1}
```

Monitoring-domain-admins-trigger.ps1

```
    #Every 1 min compare trigger
$Trigger= New-ScheduledTaskTrigger -At (Get-
Date) -RepetitionInterval (New-TimeSpan -Minutes
1)
    $User= "NT AUTHORITY\SYSTEM"
$Action= New-ScheduledTaskAction -Execute
"PowerShell.exe" -Argument "C:\PS\Comparison-
domain-admins.ps1"
Register-ScheduledTask -TaskName
"ComparisonAdminListScript_PS" -Trigger $Trigger
-User $User -Action $Action -RunLevel Highest -
Force
    #Daily update list trigger
$Trigger= New-ScheduledTaskTrigger -At 10:00am -
Daily
    $User= "NT AUTHORITY\SYSTEM"
$Action= New-ScheduledTaskAction -Execute
"PowerShell.exe" -Argument "C:\PS\Select-
domain-admins.ps1"
Register-ScheduledTask -TaskName
"ResetComparisonCurrentAdminListScript_PS" -
Trigger $Trigger -User $User -Action $Action -
RunLevel Highest -Force
```

ДОДАТОК Б – СКРИПТ POWERSHELL ДЛЯ СТВОРЕННЯ КОРИСТУВАЧІВ

Add-users.ps1:

```
#import Active Directory module
Import-Module ActiveDirectory

#add groups
New-ADOrganizationalUnit -Name:"Departments" -
Path:"DC=archi,DC=ua" -ProtectedFromAccidentalDeletion:$true
New-ADGroup -Name "Managers" -GroupCategory Security -
GroupScope Global -Path "OU=Departments,DC=archi,DC=ua"
New-ADGroup -Name "Architects" -GroupCategory Security -
GroupScope Global -Path "OU=Departments,DC=archi,DC=ua"
New-ADGroup -Name "Accountants" -GroupCategory Security -
GroupScope Global -Path "OU=Departments,DC=archi,DC=ua"
New-ADGroup -Name "Lawyers" -GroupCategory Security -
GroupScope Global -Path "OU=Departments,DC=archi,DC=ua"
New-ADGroup -Name "GAPs" -GroupCategory Security -GroupScope
Global -Path "OU=Departments,DC=archi,DC=ua"
New-ADGroup -Name "SysEng" -GroupCategory Security -
GroupScope Global -Path "OU=Departments,DC=archi,DC=ua"

#read the data from users.csv
$ADUsers = Import-Csv users.csv -Delimiter ";"

foreach ($User in $ADUsers)
{
#read user data from each field in each row and assign the
data to a variable as below
$Username = $User.username
$Password = $User.password
$Firstname = $User.firstname
$Lastname = $User.lastname
$CN = $User.CN
$email = $User.email
$streetaddress = $User.streetaddress
$city = $User.city
$zipcode = $User.zipcode
$country = $User.country
$telephone = $User.telephone
$jobtitle = $User.jobtitle
$company = $User.company
$department = $User.department
$Password = $User.Password

#check to see if the user already exists in the AD

if (Get-ADUser -F {SamAccountName -eq $Username})
```

```
 {
#if the user does exist, give a warning
Write-Warning "A user account with username $Username already
exists in Active Directory."
    }
else
{
#user does not exist then proceed to create the new user
account
New-ADUser `
-SamAccountName $Username `
-UserPrincipalName "$Username@" `
-Name "$Firstname $Lastname" `
-GivenName $Firstname `
-Surname $Lastname `
-Enabled $True `
-DisplayName "$Lastname, $Firstname" `
-Path $CN `
-City $city `
-Company $company `
-StreetAddress $streetaddress `
-OfficePhone $telephone `
-EmailAddress $email `
-Title $jobtitle `
-Department $department `
-AccountPassword (convertto-securestring $Password -
AsPlainText -Force)  `
-ChangePasswordAtLogon $True

}
}
```

Users.csv

```
FirstName;Initials;Lastname;Username;Email;StreetAddress;City
;ZipCode;Country;Department;Password;Telephone;JobTitle;Compa
ny;CN
Maxim;MC;Curcumia;Maxim.Curcumia;Maxim.Curcumia@archi.ua;Pleh
anivska
20A;Kharkiv;61000;Ukraine;Managers;maxguram2021;44123456780;B
oss;ARCHI;OU=Managers,CN=Managers,CN=Departments,DC=archi,DC=
ua
Maryna;MO;Olephirenko;Maryna.Olephirenko;Maryna.Olephirenko@a
rchi.ua;Plehanivska
20A;Kharkiv;61000;Ukraine;Architects;marco1997;44123456781;Ar
chitector-
Designer;ARCHI;OU=Architects,CN=Architects,CN=Departments,DC=
archi,DC=ua
Oleg;OD;Dobryi;OlegDobryi;OlegDobryi@archi.ua;Plehanivska
20A;Kharkiv;61000;Ukraine;GAPs;3VKr2.Wm;44123456782;General
Architect of
Project;ARCHI;OU=GAPs,CN=GAPs,CN=Departments,DC=archi,DC=ua
Roman;RV;Viraz;Roman.Viraz;Roman.Viraz@archi.ua;Plehanivska
20A;Kharkiv;61000;Ukraine;GAPs;)N3ZYJvS;44123456783;General
Architect of
```

```
Project;ARCHI;OU=Architects,CN=GAPs,CN=Departments,DC=archi,D
C=ua
Boris;BC;Campbell;Boris.Campbell;Boris.Campbell@archi.ua;Pleh
anivska
20A;Kharkiv;61000;Ukraine;Architects;9ZesQ]pq;44123456784;Arc
hitector-
Designer;ARCHI;OU=Architects,CN=Architects,CN=Departments,DC=
archi,DC=ua
Nicholas;NM;Mirosh;Nicholas.Mirosh;Nicholas.Mirosh@archi.ua;P
lehanivska
20A;Kharkiv;61000;Ukraine;Lawyers;KX*rB72p;44123456785;Lawyer
;ARCHI;OU=Lawyers,CN=Lawyers,CN=Departments,DC=archi,DC=ua
Olga;OM;Mirovova;Olga.Mironova;Olga.Mironova@archi.ua;Plehani
vska
20A;Kharkiv;61000;Ukraine;Architects;AJ+(}c3$;44123456786;Arc
hitector-
Designer;ARCHI;OU=Architects,CN=Architects,CN=Departments,DC=
archi,DC=ua
Vlada;VB;Brajnick;Vlada.Brajnick;Vlada.Brajnick@archi.ua;Pleh
anivska
20A;Kharkiv;61000;Ukraine;Architects;Jgv4{Bb$;44123456787;Arc
hitector;ARCHI;OU=Architects,CN=Architects,CN=Departments,DC=
archi,DC=ua
Valerii;VK;Kabakov;Valerii.Kabakov;Valerii.Kabakov@archi.ua;P
lehanivska
20A;Kharkiv;61000;Ukraine;Accountants;u*PQJAx5;44123456788;Ac
countant;ARCHI;OU=Accountants,CN=Accountants,CN=Departments,D
C=archi,DC=ua
Rodion;RP;Pozdniakov;Rodion.Pozdniakov;Rodion.Pozdniakov@arch
i.ua;Plehanivska
20A;Kharkiv;61000;Ukraine;SysEng;KickAss1999;44123456789;Syst
em
Engineer;ARCHI;OU=SysEng,CN=SysEng,CN=Departments,DC=archi,DC
=ua
```

ДОДАТОК В – СКРИПТ POWERSHELL ДЛЯ МОНІТОРИНГУ І АУДИТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Unauthorized-access-monitor.ps1

```
#Show table of user Logs Activity in Console
$alluserhistory = @()
$startDate = (get-date).AddDays(-3)
$DCs = Get-ADDomainController -Filter *

#who added to group
foreach ($DC in $DCs){
    $logonevents = Get-Eventlog -LogName Security -InstanceID 4728
-after $startDate -ComputerName $dc.HostName
        foreach ($event in $logonevents){
            if ($event.ReplacementStrings[0] -notlike '*$') {
                $userhistory = New-Object PSObject -Property @{
                    Object = $event.ReplacementStrings[0]
                    Group = $event.ReplacementStrings[2]
                    ALLERT = 'Обьект добавлен в защищенную группу
безопасности!'
                    WhoAdded = $event.ReplacementStrings[6]
                    Date = $event.TimeGenerated
                    DC = $dc.Name
}
$alluserhistory += $userhistory
}
}
}
Write-host "Кто получил привелегии по причине добавления в группу
безопасности:" -ForegroundColor Red
$alluserhistory

$alluserhistory = @()
$userhistory =  @()

#who tried to open folders/files
foreach ($DC in $DCs){
    $logonevents = Get-Eventlog -LogName Security -InstanceID 4663
-after $startDate -ComputerName $dc.HostName
        foreach ($event in $logonevents){
            if ($event.ReplacementStrings[1] -notlike 'ARCHI-
SERVER$') {
                $userhistory = New-Object PSObject -Property @{
                    UserName = $event.ReplacementStrings[1]
                    ALLERT = 'Получение доступа к директориям'
                    Folder = $event.ReplacementStrings[6]
                    Date = $event.TimeGenerated
```

```
                DC = $dc.Name
}
$alluserhistory += $userhistory
}
}
}
Write-host "Получение доступа к директориям:" -ForegroundColor Red
$alluserhistory

$alluserhistory = @()
$userhistory =  @()

#who login in domain
foreach ($DC in $DCs){
    $logonevents = Get-Eventlog -LogName Security -InstanceID 4768
-after $startDate -ComputerName $dc.HostName
        foreach ($event in $logonevents){
            if ($event.ReplacementStrings[1] -notlike 'ARCHI') {
                $userhistory = New-Object PSObject -Property @{
                   UserName = $event.ReplacementStrings[0]
                   ALLERT = 'Попытка входа в домен'
                   IPAddress = $event.ReplacementStrings[9]
                   Date = $event.TimeGenerated
                   DC = $dc.Name
}
$alluserhistory += $userhistory
}
}
}
Write-host "Попытки входа в домен:" -ForegroundColor Red
$alluserhistory

$alluserhistory = @()
$userhistory =  @()

#privelegue check
foreach ($DC in $DCs){
    $logonevents = Get-Eventlog -LogName Security -InstanceID 4673
-after $startDate -ComputerName $dc.HostName
        foreach ($event in $logonevents){
            if ($event.ReplacementStrings[1] -notlike 'ARCHI') {
                $userhistory = New-Object PSObject -Property @{
                    UserName = $event.ReplacementStrings[1]
                    ALLERT = 'Вызов службы привелегий'
                    Date = $event.TimeGenerated
                    DC = $dc.Name
}
$alluserhistory += $userhistory
}
}
}
Write-host "Вызов привелигированой службы:" -ForegroundColor Red
```

```
    if (!$alluserhistory) { Write-Host "Записей не найдено" }
    else {$alluserhistory}

$alluserhistory = @()
$userhistory =  @()

#who tried connect folders
foreach ($DC in $DCs){
    $logonevents = Get-Eventlog -LogName Security -InstanceID 5140
-after $startDate -ComputerName $dc.HostName
        foreach ($event in $logonevents){
            if ($event.ReplacementStrings[1] -notlike 'ARCHI') {
                $userhistory = New-Object PSObject -Property @{
                    UserName = $event.ReplacementStrings[1]
                    ALLERT = 'Попытка получения доступа к
директориям'
                    Folder = $event.ReplacementStrings[8]
                    Date = $event.TimeGenerated
                    DC = $dc.Name
}
$alluserhistory += $userhistory
}
}
}
Write-host "Попытка получения доступа к директориям:" -
ForegroundColor Red
$alluserhistory

#Integrity test
.\integrity-system-test.ps1
```

Integrity-system-test.ps1

```
#Integrity test

$allsystemhistory = @()
$null = @()
$startDate = (get-date).AddDays(-1)
$DCs = Get-ADDomainController -Filter *

foreach ($DC in $DCs){
    $events = Get-Eventlog -LogName * -InstanceID 4612 -after
$startDate -ComputerName $dc.HostName
        foreach ($event in $events){
            {
                $system4612 = New-Object PSObject -Property @
                {
                    ALLERT = 'Ошибка внутренних ресурсов!'
                    Date = $event.TimeGenerated
                    DC = $dc.Name
```

```
}
if ($system4612 = $null) {'Ошибка внутренних ресурсов: Данных не
найдено!'} else {$allsystemhistory += $system4612}
            }
}
}

foreach ($DC in $DCs){
    $events = Get-Eventlog -LogName * -InstanceID 5038 -after
$startDate -ComputerName $dc.HostName
        foreach ($event in $events){
            {
                $system5038 = New-Object PSObject -Property @
                {
                    ALLERT = 'Повреждение файлов!'
                    Date = $event.TimeGenerated
                    DC = $dc.Name
                }
if ($system5038 = $null) {'Повреждение файлов: Данных не
найдено!'} else {$allsystemhistory += $system5038}
            }
}
}

foreach ($DC in $DCs){
    $events = Get-Eventlog -LogName * -InstanceID 5057 -after
$startDate -ComputerName $dc.HostName
        foreach ($event in $events){
            {
                $system5057 = New-Object PSObject -Property @
                {
                    ALLERT = 'Ошибка криптографической операции!'
                    Date = $event.TimeGenerated
                    DC = $dc.Name
                }
if ($system5057 = $null) {'Ошибка криптографической операции:
Данных не найдено!'} else {$allsystemhistory += $system5057}
            }
}
}

foreach ($DC in $DCs){
    $events = Get-Eventlog -LogName * -InstanceID 5060 -after
$startDate -ComputerName $dc.HostName
        foreach ($event in $events){
            {
                $system5060 = New-Object PSObject -Property @
                {
                    ALLERT = 'Ошибка операции проверки!'
                    Date = $event.TimeGenerated
                    DC = $dc.Name
                }
```

```
if ($system5060 = $null) {'Ошибка операции проверки: Данных не
найдено!'} else {$allsystemhistory += $system5060}

            }
}
}

foreach ($DC in $DCs){
    $events = Get-Eventlog -LogName * -InstanceID 6410 -after
$startDate -ComputerName $dc.HostName
        foreach ($event in $events){
            {
                $system6410 = New-Object PSObject -Property @
                {
                    ALLERT = 'Запускаемый файл не отвечает
условиям безопасности!'
                    Date = $event.TimeGenerated
                    DC = $dc.Name
                }
if ($system6410 = $null) {'Запускаемый файл не отвечает условиям
безопасности: Данных не найдено!'} else {$allsystemhistory +=
$system6410}
            }
}
}

foreach ($DC in $DCs){
    $events = Get-Eventlog -LogName * -InstanceID 1102 -after
$startDate -ComputerName $dc.HostName
        foreach ($event in $events){
            {
                $system1102 = New-Object PSObject -Property @
                {
                    ALLERT = 'Удаление журнала безопасности!'
                    Date = $event.TimeGenerated
                    DC = $dc.Name
                }
if ($system1102 = $null) {'Удаление журнала безопасности: Данных
не найдено!'} else {$allsystemhistory += $system1102}
            }
}
}


$allsystemhistory
```