ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЗАХИСТУ ІНФОРМАЦІЇ

КАФЕДРА ІНФОРМАЦІЙНОЇ ТА КІБЕРНЕТИЧНОЇ БЕЗПЕКИ

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Дослідження та розробка рекомендацій щодо виявлення та усунення вразливостей служби каталогів Active Directory шляхом тестування»

зі спеціальності *125 Кібербезпека*
(код, найменування спеціальності)

освітньо-професійної програми *Інформаційна та кібернетична безпека*
(назва програми)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ Анастасія ЧАБАН
(підпис)

Виконав: здобувач вищої освіти групи БСД-41
ЧАБАН Анастасія
(прізвище, ім'я)

Керівник старший викладач РАБЧУН Дмитро
(науковий ступінь, вчене звання, прізвище, ім'я)

Рецензент _____
(науковий ступінь, вчене звання, прізвище, ім'я)

Київ 2024

ЗМІСТ

Стор.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ……. 8

ВСТУП ……..……... 9

1 **ДОСЛІДЖЕННЯ ПРОБЛЕМИ ЗАХИСТУ СЛУЖБИ КАТАЛОГІВ ACTIVE DIRECTORY** ….……..…… 11

1.1 Загальні відомості про службу каталогів Active Directory……. 11
1.2 Загальні відомості про тестування на захищеність……. 22
1.3 Аналіз відомих вразливостей та наслідків експлуатації……. 27

2 **АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ТЕСТУВАННЯ ЗАХИЩЕННОСТІ СЛУЖБИ КАТАЛОГІВ ACTIVE DIRECTORY** 35

2.1 Огляд можливостей утиліти та отримання тримання початкового доступу за допомогою утиліти Responder…… 35
2.2 Аналіз структури організації та пошук шляхів до компрометації домена за допомогою утиліти Bloodhound…… 43
2.3 Огляд можливостей підвищення доменних привілеїв та здійснення бічного переміщення за допомогою утиліти Mimikatz…… 47

3 **РОЗРОБКА РЕКОМЕНДАЦІЙ ЩОДО ЗАСТОСУВАННЯ МЕТОДІВ ТА ЗАСОБІВ ЗАХИСТУ КІНЦЕВИХ ТОЧОК ОРГАНІЗАЦІЇ** ……... 52

3.1 Методологія тестування на захищеність службі каталогів Active Directory…… 52
3.2 Загальні рекомендації щодо оцінювання та усунення вразливостей…… 59

ВИСНОВКИ …….. 66

ПЕРЕЛІК ПОСИЛАНЬ 68

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація) 72

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

AD – Active Directory

LDAP – Lightweight Directory Access Protocol

DNS – Domain Name System

AD DS – Active Directory Domain Services

OU – Organizational Units

KDC – Key Distribution Center

TGS – Ticket Granting Service

TGT – Ticket Granting Ticket

SASL – Simple Authentication and Security Layer

NTLM – New Technology LAN Manager

SID – Security Identifier

MDNS – Multicast DNS

ESS – Extended Protection for Authentication (Enhanced Session Security)

ВСТУП

Актуальність дослідження. Виявлення вразливостей та забезпечення безпеки служби каталогів Active Directory (AD) організації має вирішальне значення з кількох причин.

По-перше, активи служби каталогів містять конфіденційні дані, а доступ до них надає контролер домену, який автентифікує та авторизує користувачів. Забезпечення безпеки цієї служби є важливим для запобігання несанкціонованому доступу, витоку та втрати даних.

По-друге, служба каталогів Active Directory є привабливим об'єктом для кібератак, оскільки організації часто працюють з великими обсягами інформації та ресурсів і мають велику кількість працівників, що потребує масштабування. Через це масштабування стає складніше стежити за коректними налаштуваннями, і існує безліч атак, які використовують не лише відомі вразливості у програмному забезпеченні, протоколах та операційних системах, але й помилки в конфігураціях.

Дослідження методів та засобів тестування на захищеність служби каталогів Active Directory є важливим для виявлення можливих вразливостей та розробки рекомендацій щодо їх усунення. Це дослідження спрямоване на аналіз методів та засобів тестування захищеності цієї служби, визначення її призначення, основних функцій, а також розробку варіантів тестування захищеності служби каталогів Active Directory та надання загальних рекомендацій щодо усунення вразливостей.

Вищесказане визначає актуальність теми даної кваліфікаційної роботи, основний зміст якої становлять дослідження методів та засобів тестування на захищеність служби каталогів Active Directory.

Об’єкт дослідження – захищеність служби каталогів Active Directory.

Предмет дослідження – методи та засоби тестування захищеності служби каталогів Active Directory.

Мета роботи розробити рекомендації щодо виявлення та усунення вразливостей служби каталогів Active Directory шляхом тестування.

Наукові завдання:

дослідити сутність проблеми захисту служби каталогів Active Directory;

проаналізувати методи та засоби тестування захищеності служби каталогів Active Directory;

проаналізувати існуючі рішення зі тестування захищеності служби каталогів Active Directory;

розробити рекомендації щодо виправлення виявлених вразливостей служби каталогів Active Directory;

Методи дослідження – опрацювання методів тестування на безпеку, аналіз вразливостей, системного підходу до вдосконалення методів тестування захищеності служби каталогів Active Directory та усунення вразливостей.

Практичне значення одержаних результатів: розроблено рекомендації щодо застосування методів та засобів для виявлення вразливостей шляхом тестування на захищеність Active Directory, а також розроблено рекомендації фахівцям з кібербезпеки щодо усунення вразливостей.

1 ДОСЛІДЖЕННЯ ПРОБЛЕМИ ЗАХИСТУ СЛУЖБИ КАТАЛОГІВ ACTIVE DIRECTORY

1.1. Загальні відомості про службу каталогів Active Directory

Перш ніж перейти до аналізу методів тестування на захищенність та аналізу вразливостей Active Directory ознайомимось з призначенням та базовими поняттями та компонентами цієї служби.

Active Directory (AD) – це служба каталогів на базі Windows, яка забезпечує централізоване управління обліковими записами та ресурсами в мережі та дозволяє адміністраторам ефективно впроваджувати управління доступом на основі ролей та принципу найменших привілеїв.

Active Directory використовується для зберігання даних про ресурси, такі як облікові записи користувачів, комп'ютери, принтери та файли, в ієрархічній структурі, відомій як «дерево каталогів». Ця структура дозволяє адміністраторам легко управляти та організовувати ресурси.

AD використовує протокол LDAP (Lightweight Directory Access Protocol) для спілкування між клієнтами та серверами. AD також використовує DNS (Domain Name System) для іменування та визначення місця ресурсів в мережі.

Однією з ключових особливостей AD є можливість делегування управління. Це означає, що адміністратори можуть надавати обмежені привілеї іншим користувачам або групам користувачів, дозволяючи їм виконувати певні задачі, такі як управління обліковими записами користувачів або ресурсами.

AD також включає в себе політики групи, які дозволяють адміністраторам управляти налаштуваннями безпеки та поведінкою системи на рівні домену. Це може включати налаштування, такі як вимоги до паролів, налаштування мережевого екрану, правила доступу до файлів та багато іншого.

В цілому, Active Directory – це потужний інструмент для централізованого управління ресурсами в мережі, який допомагає організаціям підтримувати

безпеку, ефективність та організацію. Однак, як і будь-яка складна система, вона вимагає відповідного розуміння та управління для ефективного використання.

Загалом, Active Directory складається з наступних ключових компонентів:

- *Об'єкт:* Об'єкт може бути визначений як будь-який ресурс, присутній в середовищі Active Directory, такі як:
    - *Користувач:* Об'єкт користувача в AD представляє реального користувача, який є частиною мережі AD організації. Він є листковим об'єктом, що означає, що він не може містити інші об'єкти AD всередині себе. Користувач може бути працівником організації, таким як менеджер, спеціаліст в HR або адміністратор IT, який зазвичай має підвищені права в порівнянні з іншими користувачами. Об'єкт користувача є безпечним принципом, що означає, що він має ідентифікатор безпеки (SID) окрім глобального унікального ідентифікатора (GUID). Об'єкт користувача в AD має атрибути, що містять інформацію, таку як канонічні імена, ім'я, по батькові, прізвище, облікові дані для входу, номер телефону, керівник, якому він або вона підпорядковані, адреса, хто є його підлеглими та інше.
    - *Принтер:* Об'єкт принтера в AD є вказівником, який вказує на реальний принтер у мережі AD. Він є листковим об'єктом, що означає, що він не може містити інші об'єкти AD всередині себе. Об'єкт принтера в AD не є безпечним принципом, тому він має лише GUID. Об'єкт принтера в AD має атрибути, що містять інформацію, таку як назва принтера, назва драйвера, режим кольору, номер порту та інше.
    - *Об'єкт комп'ютера:* Об'єкт комп'ютера в AD представляє комп'ютер, який є частиною мережі AD організації. Користувач може бути будь-яким із працівників організації. Він є листковим об'єктом, що означає, що він не може містити інші об'єкти AD всередині себе. Об'єкт комп'ютера в AD також є безпечним принципом, подібно до об'єкту користувача. Таким чином, комп'ютери також мають SIDs окрім GUIDs. Об'єкт комп'ютера в AD має атрибути, що містять інформацію,

таку як назва комп'ютера, комп'ютерне ім'я (до Windows 2000), його унікальний ідентифікатор, DNS-ім'я, роль, опис, місцезнаходження, хто управляє комп'ютером, версія операційної системи, на якій він працює, та інше.

- *Спільна папка.* Об'єкт спільної папки в AD є вказівником, що вказує на спільну папку на комп'ютері, на якому зберігається папка. Спільна папка - це папка, яка спільно використовується між членами мережі AD, і тільки ці члени можуть переглядати вміст папки, тоді як іншим членам буде відмовлено в доступі. Це листовий об'єкт, що означає, що він не може містити інші об'єкти AD у собі. Об'єкт спільної папки в AD не є об'єктом безпеки, тому він має тільки GUID. Об'єкт спільної папки в AD має атрибути, що містять інформацію, таку як ім'я папки, місцезнаходження, привілеї доступу і багато іншого. Об'єкт спільної папки в AD є вказівником, який вказує на спільну папку на комп'ютері, де зберігається папка. Спільна папка – це папка, яка спільно використовується між членами мережі AD, і тільки ці члени можуть переглядати вміст папки, тоді як іншим членам буде відмовлено в доступі. Він є листковим об'єктом, що означає, що він не може містити інші об'єкти AD всередині себе. Об'єкт спільної папки в AD не є безпечним принципом, тому він має лише GUID. Об'єкт спільної папки в AD має атрибути, що містять інформацію, таку як назва папки, місцезнаходження, права доступу та інше.
- *Група:* Об'єкт групи в AD - це об'єкт, що може містити інші об'єкти AD, такі як інші групи, користувачі та комп'ютери. Таким чином, об'єкт групи є контейнерним об'єктом. Об'єкт групи в AD також є безпечним принципом, подібно до об'єктів користувача та комп'ютера. Таким чином, об'єкти групи також мають SIDs окрім GUIDs. Об'єкт групи використовується для надання дозволів членам об'єктів AD всередині групи. Об'єкт групи в AD має атрибути, що містять інформацію, таку як назва групи, об'єкти-члени в групі та інше.

- *Організаційна одиниця:* Організаційні одиниці (OU) в домені, керованому службами доменів Active Directory Domain Services (AD DS), дозволяють логічно групувати такі об'єкти, як облікові записи користувачів, службові облікові записи або облікові записи комп'ютерів [4]. OU в AD також є безпечним принципом, подібно до об'єктів користувача, комп'ютера та групи. Таким чином, OU також мають SIDs окрім GUIDs. OU використовується для делегування ролей членам об'єктів AD всередині групи. Організаційна одиниця в AD має атрибути, що містять інформацію, таку як її назва, об'єкти-члени в OU та інше.
- *Контролер домену:* Об'єкт контролера домену (DC) в AD посилається на сервер, який діє як контролер домену для домену, в якому він розташований. Контролер домену зберігає політики, автентифікує користувачів AD та також відповідає за ролі, які повинні виконувати всі контролери доменів в домені.
- *Об'єкти сайту:* Об'єкти сайту в AD – це об'єкти, які реалізовані в мережі Active Directory для управління та полегшення процесу реплікації [2].
- Атрибути: У кожному об'єкті в Active Directory є пов'язаний набір атрибутів, що використовуються для визначення його характеристик. Об'єкт комп'ютера містить атрибути, такі як ім'я хоста та DNS-ім'я.
- Усі атрибути в AD мають пов'язане ім'я LDAP, яке можна використовувати при виконанні LDAP-запитів, наприклад displayName для Повного імені та givenName для Імені.

- *Схема:* Схема Active Directory по суті є планом будь-якого корпоративного середовища. Вона визначає, які типи об'єктів можуть існувати в базі даних AD та їх пов'язані атрибути. Вона перераховує визначення, відповідні об'єктам AD, і містить інформацію про кожен об'єкт. Наприклад, користувачі в AD належать до класу «user», а об'єкти комп'ютерів до «computer» та інших.
- Кожен об'єкт має свою власну інформацію (деякі обов'язкові до

встановлення, інші необов'язкові), яка зберігається у властивостях. Коли об'єкт створюється з класу, це називається інстанціюванням, а об'єкт, створений з певного класу, називається екземпляром цього класу.

- *Контейнер:* Об'єкти-контейнери містять інші об'єкти і мають визначене місце в ієрархії піддерев каталогу.
- *Домен* – логічна група об'єктів, таких як комп'ютери, користувачі, ОО, групи тощо. Можна уявляти кожен домен як різні міста у межах штату або країни. Домени можуть функціонувати абсолютно незалежно один від одного або бути пов'язаними через довірчі відносини.
- *Дерево* – це колекція доменів Active Directory, яка починається з одного кореневого домену. Ліс – це колекція дерев Active Directory. Кожен домен у дереві ділить кордон з іншими доменами. Взаємовідношення «батьківсько-дочірній» утворюється, коли домен додається під інший домен у дереві. Два дерева в одному лісі не можуть мати спільне ім'я (простір імен).
- Як приклад, є два дерева в лісі AD: blue.com та red.com. Дочірній домен першого буде sales.blue.com, тоді як дочірній домен другого може бути accounts.red.com. Усі домени в дереві спільно використовують стандартний Глобальний каталог, що містить всю інформацію про об'єкти, які належать до дерева.
- Ліс – це колекція доменів Active Directory, верхній контейнер, що містить всі об'єкти AD, включаючи, але не обмежуючись, доменами, користувачами, групами, комп'ютерами та об'єктами об'єктів політики групи (GPOs). Ліс може містити один або кілька доменів і може бути уявлений як штат у США або країна в ЄС. Кожен ліс працює незалежно, але може мати різні довірчі відносини з іншими лісами. [1][3]

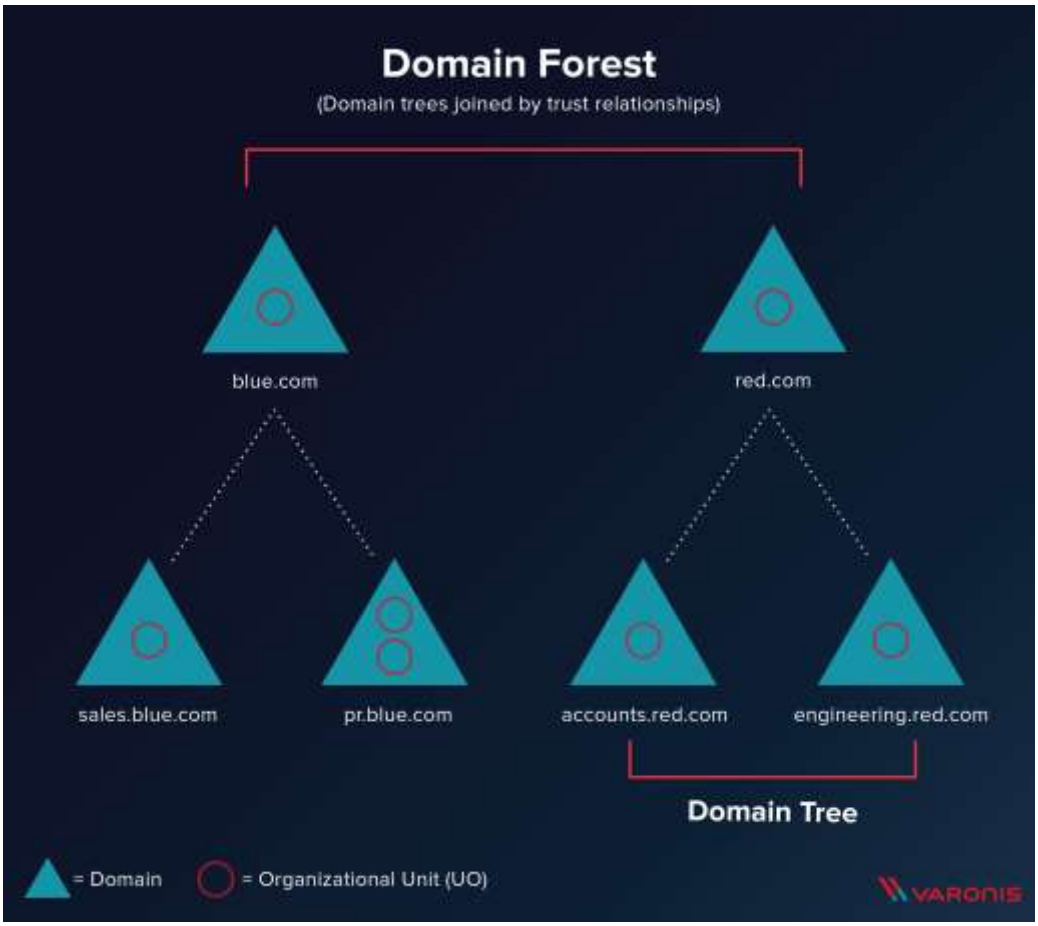

Рис 1.1 – Схема структури лісу, дерев та дочірніх доменів Active Directory [3]

- DNS: Domain Name System (DNS) є одним з найважливіших сервісів для Active Directory. DNS забезпечує послуги розв'язування імен для Active Directory, перетворюючи імена хостів, URL-адреси та повністю кваліфіковані доменні імена (FQDN) в IP-адреси.

  Active Directory Domain Services (AD DS) використовує DNS як механізм розташування контролера домену. Коли виконується будь-яка основна операція Active Directory, така як аутентифікація, оновлення або пошук, комп'ютери використовують DNS для пошуку контролерів домену Active Directory. Крім того, контролери домену використовують DNS для пошуку один одного.

  AD DS надає вбудований метод зберігання та реплікації записів DNS за допомогою зон DNS, інтегрованих з Active Directory. Всі записи та дані зони, збережені в зоні, реплікуються на інші сервери DNS за допомогою рідного сервісу реплікації AD DS.

DNS є частиною протоколу TCP/IP, і разом DNS-клієнт та DNS-сервер надають послуги розв'язування імен комп'ютерів та користувачів3. Коли ви налаштовуєте з'єднання TCP/IP з IP-адресою DNS-сервера, DNS-клієнт запитує DNS-сервер для виявлення контролерів домену та для розв'язування імен комп'ютерів в IP-адреси.

Наприклад, коли мережевий користувач з обліковим записом користувача Active Directory входить в домен Active Directory, служба DNS-клієнта запитує DNS-сервер для розташування контролера домену для домену Active Directory3. Коли DNS-сервер відповідає на запит та надає IP-адресу контролера домену клієнту, клієнт зв'язується з контролером домену, і може початися процес аутентифікації.

Тепер, коли було розглянуто основні компоненти служби каталогів Active Directory переходимо до ознайомлення того, як відбувається автентифікація в Active Directory.

Автентифікація в Active Directory (AD) працює через кілька протоколів, серед яких основні – Kerberos, LDAP та NTLM.

Kerberos: Це протокол безпеки мережевого рівня, який використовується для автентифікації довірених пристроїв у мережі. При автентифікації в AD за допомогою Kerberos, після того, як користувач увійшов до системи за своїми обліковими даними AD, для нього створюється сеанс. Сеанс має кілька атрибутів, і можна налаштувати час його дії відповідно до потреб організації. У Kerberos є три складові: сервер, клієнт та центр розподілу ключів (KDC). Kerberos має дві функції - автентифікацію та надання квитків. Він використовує криптографію з секретним ключем для автентифікації ідентичності користувача. Якщо користувач або клієнт бажає отримати доступ до будь-яких ресурсів компанії, спочатку користувачі повинні автентифікуватися в KDC. KDC містить сервер автентифікації та сервер надання квитків. Користувач надсилає запит на авторизацію до AS, який видаватиме квиток для надання прав (TGT) клієнту. Клієнт робить другий виклик авторизації разом з TGT на сервер надання квитків (TGC). TGS надсилає ключ на цільовий сервер та надає токен клієнту. Клієнт потім звертається до цільового

сервера з токеном, а цільовий сервер автентифікує його, використовуючи ключ, який був обмінений TGS. Таким чином, не передається жодне ім'я користувача або пароль, а все ж користувача автентифікують.

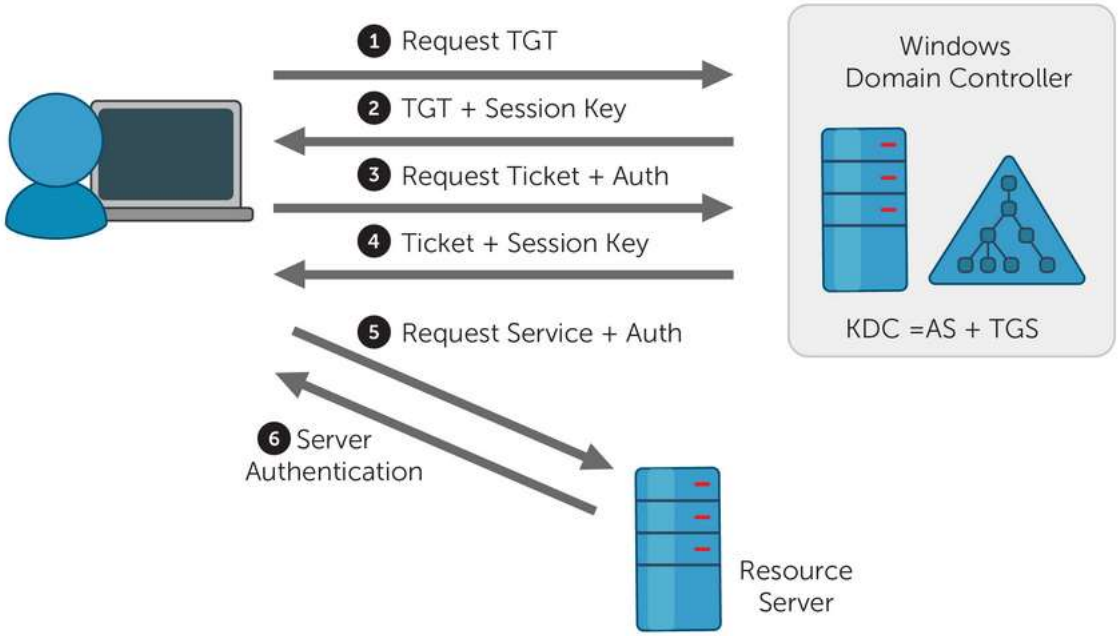

Рис. 1.2 – Kerberos authentication protocol [7]

LDAP автентифікація: LDAP (Lightweight Directory Access Protocol) - це відкритий, крос-платформовий програмний протокол, який використовується для аутентифікації та комунікації в службах каталогів. Він забезпечує мову, яку застосунки використовують для взаємодії в службах каталогів, де зберігаються облікові записи комп'ютерів, користувачів та паролі, та обмінюються ними з іншими сутностями в мережах.

LDAP має широкий спектр використання, але найпопулярнішим є як центральний вузол для організацій для управління аутентифікацією. Він дуже ефективний для допомоги організаціям зберігати, керувати та отримувати доступ до імен користувачів та паролів по всій їхній мережі та додатках. Якщо організації використовують відповідні плагіни, LDAP дозволяє їм зберігати та перевіряти облікові дані кожного разу, коли користувач намагається отримати доступ до додатків, каталогів та систем.

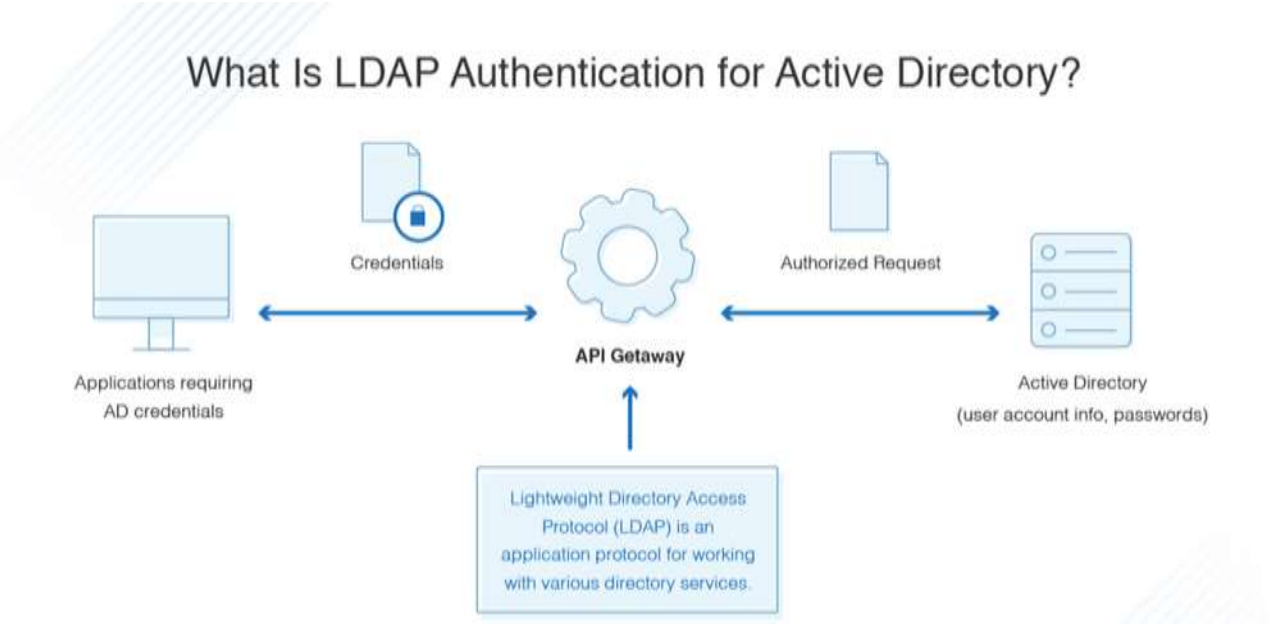

Рис. 1.3 – Принцип роботи протоколу LDAP[8]

Аутентифікація LDAP працює шляхом прив'язки користувачів до сервера. Коли клієнт надсилає запит на певну інформацію, таку як облікові дані користувача, сервер LDAP обробляє його за допомогою внутрішньої мови, а потім взаємодіє зі службами каталогів перед відправленням відповіді. LDAPv2 пропонує дві форми аутентифікації, LDAPv2 пропонує дві форми автентифікації – просту та з використанням простого рівня автентифікації та безпеки (SASL)[5].

Проста аутентифікація в LDAP працює наступним чином:

Клієнт надсилає серверу LDAP повністю кваліфіковане DN (ім’я) клієнта (користувача) та пароль клієнта у відкритому тексті.

Цей механізм має проблеми з безпекою, оскільки пароль можна прочитати з мережі. Для уникнення розкриття пароля використовується простий механізм автентифікації в зашифрованому каналі (наприклад, SSL), за умови, що це підтримується LDAP-сервером. [6].

SASL (Simple Authentication and Security Layer) – це рамка для аутентифікації та захисту даних в інтернет-протоколах. SASL відокремлює механізми аутентифікації від протоколів застосувань, що дозволяє використовувати будь-який механізм аутентифікації, який підтримує SASL, в будь-якому протоколі застосування, який використовує SASL.

Автентифікація SASL працює шляхом прив'язки сервера LDAP до окремого

процесу автентифікації, наприклад, Kerberos. Потім сервер LDAP використовує протокол LDAP для надсилання повідомлення процесу автентифікації Kerberos. Це запускає серію повідомлень-відповідей, які або повідомляють про успішну автентифікацію, або про помилку автентифікації. За замовчуванням всі ці повідомлення надсилаються відкритим текстом, що означає, що будь-хто, хто їх підглядає, зможе їх прочитати. Тому дуже важливо додати заходи безпеки, такі як шифрування, до процесу автентифікації, щоб гарантувати, що дані користувача і дані, до яких надається спільний доступ, захищені.[5]

NTLM автентифікація: NTLM (New Technology LAN Manager) – це протокол аутентифікації, розроблений Microsoft. Він використовує механізм виклику-відповіді для перевірки автентичності користувача. Ось як працює аутентифікація NTLM:

1) Користувач надає своє ім’я користувача, пароль та доменне ім’я клієнту.
2) Клієнт створює засекречену версію пароля або хеш і видаляє повний пароль.
3) Клієнт передає текстову версію імені користувача відповідному серверу.
4) Сервер відповідає клієнту викликом, який є 16-байтовим випадковим числом.
5) У відповідь клієнт надсилає виклик, зашифрований за допомогою хешу пароля користувача.
6) Сервер потім надсилає виклик, відповідь та ім’я користувача до контролера домену (DC).
7) DC отримує пароль користувача з бази даних і використовує його для шифрування виклику.
8) DC потім порівнює зашифрований виклик та відповідь клієнта. Якщо ці дві частини співпадають, то користувача автентифікують і надають доступ.

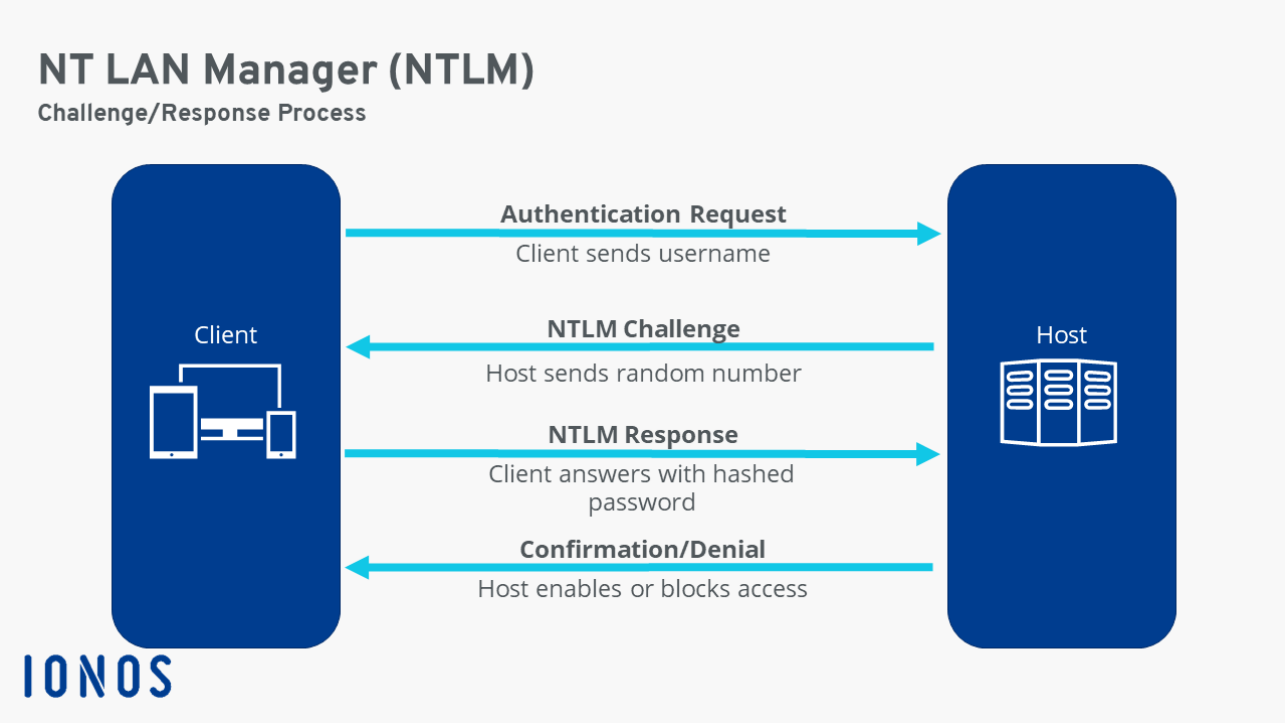

Рис. 1.4 – NTLM authentication protocol [9]

NTLM був замінений Kerberos як протоколом аутентифікації за замовчуванням у Windows 2000 та наступних доменах Active Directory (AD). Однак NTLM все ще використовується як резервний протокол, якщо Kerberos не вдається аутентифікувати користувача.

Авторизація в Active Directory: Тепер проаналізуємо, як працює авторизація в Active Directory. Коли користувач входить у систему, він надає свої облікові дані для взаємодії з ресурсами.

Active Directory перевіряє доступ до ресурсу на основі безпечного токена користувача. У безпечному токені, по суті, відбувається перевірка того, чи входить користувач до списку управління доступом (ACL) для запитуваного об'єкта або ресурсу.

Деякі типи атрибутів, які можуть міститися в безпечному токені, включають:

- група користувачів;
- володіння;
- адміністративні привілеї.

Основним засобом ідентифікації безпечного об'єкта під час спроби доступу до мережевих ресурсів є ідентифікатор безпеки (SID).

Атрибут SID унікальний для кожного користувача або групи безпеки.

Список вибіркового управління доступом (DACL) визначає безпечних агентів, яким дозволено або заборонено доступ до об'єкта.

Коли користувач або процес намагається отримати доступ до об'єкта, система перевіряє записи управління доступом (ACE) у DACL об'єкта, щоб визначити, чи слід надати доступ до нього.

Список контролю доступу до системи (SACL) дає змогу адміністраторам реєструвати спроби доступу до захищеного об'єкта.

Кожен запис управління доступом (ACE) визначає типи спроб доступу заданого агента, які призводять до створення запису в журналі подій безпеки.

ACE в SACL може створювати аудиторські записи в разі невдалої спроби доступу, в разі успішної спроби або в обох випадках.

Авторизація в Active Directory працює таким чином:

1) Учасник безпеки (Security Principal) отримують ідентифікатори безпеки (SIDs) під час створення облікового запису.
2) Загальні ресурси в мережі включають списки управління доступом (ACL), які визначають, хто може отримати доступ до ресурсу.
3) Під час аутентифікації користувацькі облікові записи отримують безпечні маркери, які включають SID користувача і всі пов'язані з ним SID груп.
4) Маркер безпеки порівнюється зі списком управління доступом (ACL) на основі розсуду (DACL) на ресурсі, і доступ дозволяється або забороняється.

1.2. Загальні відомості про тестування на захищеність

Тестування на проникнення (також відоме як «пентест» або «етичний хакінг») − це систематичний процес виявлення вразливостей у мережах (інфраструктури) та додатках (програмне забезпечення) з використанням різноманітних методів та інструментів для імітації атак.

Основна мета пентесту полягає у виявленні слабких місць, які можуть призвести до несанкціонованого доступу, оцінці ефективності існуючих заходів

безпеки, наданні рекомендацій щодо покращення захисту та підготовці до потенційних атак шляхом імітації дій зловмисників.

Основне завдання тесту на проникнення полягає в тому, щоб, цілком відтворюючи дії зловмисника, здійснити атаки на IT-інфраструктуру, включаючи мережеві та програмні компоненти, системи зберігання даних і людський фактор, що може виконуватися як частина аудиту на відповідність стандартам або як окреме завдання для виявлення та усунення вразливостей.

Основні види пентестів:

Кількість інформації, наданої перед тестуванням, може значно вплинути на його результати. Стиль тестування зазвичай визначається як White box, Black box або Grey box penetration testing [10].

- White box penetration testing, передбачає надання повної інформації про мережу та систему тестувальнику, включаючи карти мережі та облікові дані. Це допомагає заощадити час і знизити загальні витрати на тестування. White box penetration testing корисне для моделювання цільової атаки на конкретну систему з використанням максимально можливої кількості векторів атаки.
- У Black box penetration testing тестувальнику взагалі не надається жодної інформації. У цьому випадку пентестер діє як неперевірений зловмисник, від початкового доступу та виконання до експлуатації. Цей сценарій можна вважати найавтентичнішим, оскільки він демонструє, як противник без внутрішніх знань буде цілитися і компрометувати організацію. Однак це зазвичай робить його найдорожчим варіантом.
- У Grey box penetration testing тестувальнику надається лише обмежена інформація. Зазвичай це логін та облікові дані. Grey box testing корисне для розуміння рівня доступу, який може отримати привілейований користувач, і потенційної шкоди, яку він може завдати. Grey box testing балансують між глибиною та ефективністю і можуть бути використані для моделювання як внутрішньої загрози, так і атаки, що порушила б периметр мережі. У більшості реальних атак наполегливий противник проводить розвідку цільового середовища, що дає йому знання, подібні до інсайдерських. Grey

box testing часто обирається замовниками як найкращий баланс між ефективністю та автентичністю, оскільки виключає потенційно тривалий етап розвідки [11].

Процес тестування на проникнення можна спростити до наступних п’яти етапів:

- Розвідка. Є основою всього процесу. На цьому етапі тестувальник починає збір інформації про цільову систему. Збір може охоплювати різноманітні дані, включаючи інформацію про IP-адреси, деталі домену, мережеві служби, поштові сервери та топологію мережі.

  Цей проактивний збір розвідданих допомагає скласти детальний план цільового середовища. Озброєний цією інформацією, тестувальник може розробити обґрунтовану стратегію тестування, яка ефективно перевірятиме вразливості, підготувавши основу для наступних етапів процесу тестування на проникнення.

- Сканування. Ця фаза включає детальний технічний огляд цільової системи. Використовуються автоматизовані інструменти, такі як сканери вразливостей, мережеві мапери та інші для розуміння того, як цільова система реагує на різні втручання.

  Сканування дозволяє тестувальникам визначити, як цільова програма поводиться за різних умов, та ідентифікувати потенційні слабкі місця, які можуть бути використані. Це допомагає створити цифрову карту системи, дозволяючи тестувальнику знайти можливі точки входу, які може використати зловмисник.

- Оцінки вразливостей. Після ретельного сканування цільової системи процес переходить до етапу Оцінки вразливостей. Ця фаза є ретельним аналізом цільової системи для визначення можливих точок експлуатації.

  Використовуючи комбінацію автоматизованих інструментів та ручних методологій, тестувальник ретельно перевіряє безпеку систем, ідентифікуючи будь-які можливі лазівки. Ця скрупульозна оцінка забезпечує повне розуміння стану безпеки системи, вказуючи на потенційні вразливості, які можуть бути використані кіберзлочинцями.

- Експлуатація. Після завершення Оцінки вразливостей наступним етапом є Експлуатація. У цій критичній фазі тестувальник намагається використати виявлені вразливості. Мета полягає не в тому, щоб завдати шкоди, а в тому, щоб визначити глибину вразливості та оцінити потенційні збитки, які вона може спричинити.

  Експлуатація може включати витоки даних, порушення роботи сервісів або несанкціонований доступ до конфіденційної інформації. Ця стадія повинна бути ретельно контрольованою та моніторинговою, щоб забезпечити, що система не буде випадково пошкоджена під час процесу. Це тонкий баланс між розширенням меж та збереженням цілісності системи.
- Звітність. Остаточний етап – Звітність, де тестувальник складає детальний звіт про свої знахідки. Це включає виявлені вразливості, використані дані та успішність симульованого порушення. Але звіт – це не просто перелік проблем. Він також пропонує рекомендації щодо усунення вразливостей, включаючи виправлення програмного забезпечення, зміни конфігурації та покращені політики безпеки [12].

Фахівці з тестування безпеки використовують наступні методології пентесту:

1. OSSTMM: Методологічний посібник з тестування безпеки з відкритим вихідним кодом (OSSTMM) – це методологія пентестингу, яка пройшла рецензування (Institute for Security and Open Methodologies, 2010). Вона надає наукову основу для тестування мереж і оцінки вразливостей, а також пропонує всебічний посібник, який може бути належним чином використаний сертифікованими пентестерами. OSSTMM охоплює п'ять категорій (Rounsavall, 2017):

- Контроль даних та інформації;
- Обізнаність про кібербезпеку серед персоналу;
- Контроль шахрайства та соціальної інженерії;
- Контроль мережевих пристроїв, включаючи комп'ютери та бездротові пристрої;
- Фізичний контроль безпеки.

Однією з основних переваг OSSTMM є її висока гнучкість. Якщо пентестери правильно застосовують OSSTMM, вони можуть використовувати її для вирішення вразливостей, виявлених на різних пристроях, включаючи комп'ютери, сервери, бездротові пристрої тощо.

2. OWASP: Фонд Open Web Application Security Project (OWASP) (2020, 2021, 2022) підтримує методології пентестингу та всебічні посібники для тестування веб-додатків, мобільних додатків та прошивки пристроїв. Коли методології OWASP виконуються належним чином, вони можуть допомогти пентестерам виявити ряд вразливостей у мікропрограмах мережі та мобільних або веб-додатках.

3. NIST: Національний інститут стандартів і технологій (NIST; 2022) є агентством в межах Міністерства торгівлі США. Метою NIST щодо стандартів інформаційної безпеки є не встановлення однієї конкретної методології, а створення серії стандартів для пентестингу (Scarfone et al., 2008). Хоча федеральний уряд зобов'язаний дотримуватися стандартів NIST, інші мережі часто також дотримуються їх.

Стандарти NIST слід розглядати як абсолютний мінімум, а не як єдині стандарти, яких повинні дотримуватися бізнес чи інші організації. Будь-який сертифікований пентестер повинен бути знайомий з методологіями пентестингу мереж та додатків, створеними NIST.

4. PTES: Стандарт виконання пентестингу (PTES; 2014) – це методологія пентестингу, яка охоплює сім розділів:

- Взаємодія перед початком тестування
- Збір розвідувальної інформації
- Моделювання загроз
- Аналіз вразливостей
- Експлуатація
- Пост-експлуатація
- Звітність

PTES (2012) також надає розширений технічний посібник, який дозволяє пентестерам виконувати методологію.

5. ISSAF: Рамкова структура оцінки безпеки інформаційних систем (ISSAF) є спеціалізованим підходом до пентестингу (Open Information Systems Security Group, 2006). Її розширений посібник, який нараховує понад 1200 сторінок, викладає основу цієї методології тестування. Зрозумілий підхід ISSAF легко адаптувати для індивідуальних організацій і пентестерів, дозволяючи створювати персоналізовані плани тестування. Будь-який пентестер, який використовує кілька інструментів, повинен дотримуватися методології ISSAF.

Варто зазначити, що ISSAF виходить далеко за межі простого пентестингу: вона також охоплює створення інструментів, які можна використовувати для навчання інших осіб, які мають доступ до мережі. Вона також забезпечує дотримання відповідних правових стандартів особами, які користуються даною мережею [13].

1.3 Аналіз відомих вразливостей служби каталогів Active Directory та наслідків експлуатації

Ця тема зосереджена на аналізі відомих вразливостей у службі каталогів Active Directory та потенційних наслідках їх експлуатації. У контексті сучасного кіберпростору розуміння цих вразливостей є критично важливим для забезпечення надійного захисту інформаційних систем.

Повторне використання паролів. Повторне використання паролів є однією з небезпечних практик в адмініструванні Active Directory. Коли користувачі використовують один і той самий пароль для кількох облікових записів або систем, це створює значний ризик для безпеки всієї мережі. Наприклад, якщо зловмисник отримує доступ до одного з облікових записів через викрадений або зламаний пароль, він може використовувати ті ж облікові дані для входу в інші системи, які використовують той же пароль. Ця атака відома як «credential stuffing». Такі атаки можуть призвести до масового порушення безпеки, оскільки зловмисники можуть отримати доступ до корпоративної електронної пошти, внутрішніх баз даних, фінансових систем та інших критично важливих ресурсів компанії. У випадку компрометації адміністраторських облікових записів, наслідки можуть бути ще

більш катастрофічними, оскільки зловмисники можуть отримати повний контроль над всією IT-інфраструктурою, викрадати конфіденційні дані, змінювати налаштування безпеки або видаляти важливу інформацію. Така ситуація може призвести до серйозних фінансових втрат, втрати репутації та правових наслідків для компанії.

Використання слабких паролів. Використання слабких паролів є також однією серйозною проблемою безпеки в Active Directory. Слабкі паролі, такі як «123456», «password» або «qwerty», легко зламуються за допомогою простих атак перебору (brute force) або словникових атак (dictionary attacks). Зловмисники можуть використовувати автоматизовані інструменти для швидкого підбору таких паролів, що дозволяє їм отримати доступ до облікових записів користувачів. У контексті Active Directory це особливо небезпечно, оскільки компрометація одного облікового запису може надати зловмиснику доступ до багатьох інших ресурсів мережі. Наприклад, якщо зловмисник зламає пароль користувача, він може отримати доступ до електронної пошти, внутрішніх систем управління проектами, фінансових звітів та інших конфіденційних даних. Якщо зламаний пароль належить адміністратору, наслідки можуть бути ще серйознішими, оскільки зловмисник може отримати повний контроль над всією мережею, змінювати конфігурації систем, видаляти або змінювати дані, створювати нові облікові записи для подальших атак і навіть вимкнути заходи безпеки.

Зберігання паролів на шарах зі спільним доступом. Зберігання паролів на шарах зі спільним доступом є серйозною проблемою безпеки, яка може призвести до легкої компрометації облікових даних. Коли паролі зберігаються у відкритому вигляді на загальнодоступних шарах або у спільних папках, вони стають вразливими до зловмисних дій. Наприклад, якщо паролі зберігаються у текстових файлах на робочому столі або в папках, до яких мають доступ кілька користувачів, будь-хто, хто має доступ до цих файлів, може їх викрасти. Це особливо небезпечно у випадку, якщо до цих паролів мають доступ сторонні особи або навіть співробітники, які можуть бути невдоволеними або мати зловмисні наміри. Викрадені паролі можуть бути використані для несанкціонованого доступу до

облікових записів і систем, що може призвести до витоку конфіденційної інформації, викрадення даних клієнтів, фінансових звітів, стратегічних планів компанії та інших критично важливих даних. Це може спричинити серйозні фінансові втрати, втрату довіри клієнтів і партнерів, а також негативні правові наслідки.

ZeroLogon. Zerologon – це назва вразливості, ідентифікованої як CVE-2020-1472, що виникає через помилку в процесі аутентифікації: ініціалізаційний вектор (IV) завжди встановлений на всі нулі, тоді як IV повинен бути випадковим числом. Ця небезпечна вразливість має рейтинг 10 із 10 за шкалою CVSS (v3.1) за ступенем серйозності.

Zerologon дозволяє неавтентифікованому зловмиснику з мережевим доступом до контролера домену встановити вразливу сесію Netlogon і в кінцевому результаті отримати привілеї адміністратора домену. Вразливість особливо серйозна, оскільки єдиною вимогою для успішної експлуатації є здатність встановити з'єднання з контролером домену. Ця вразливість дозволяє хакеру взяти під контроль контролер домену (DC), включаючи кореневий DC, змінюючи або видаляючи пароль для службового облікового запису на контролері. Зловмисник може викликати відмову в обслуговуванні або повністю захопити і контролювати всю мережу.

Суть вразливості полягає в поганій реалізації виклику ComputeNetlogonCredential протоколу Netlogon Remote Protocol (MS-NRPC). ComputeNetlogonCredential приймає на вхід 8-байтовий виклик, виконує криптографічну трансформацію за допомогою сесійного ключа (що підтверджує знання секрету комп’ютера) і видає 8-байтовий результат. Проблема полягає у вразливості в новішому методі AES-CFB8 (який є єдиним дозволеним у новіших версіях Windows), що використовується для виконання цієї трансформації.

Для безпечного використання AES-CFB8 необхідно генерувати випадковий ініціалізаційний вектор (IV) для кожного тексту, який буде зашифрований за допомогою того ж самого ключа. Однак функція ComputeNetlogonCredential встановлює IV на фіксоване значення з 16 нульових байтів. Це призводить до

криптографічної вразливості, при якій шифрування 8 байтів нулів може дати шифротекст з нулями з імовірністю 1 до 256. Ще одна проблема реалізації, що дозволяє цю атаку, полягає в тому, що незашифровані сесії Netlogon не відхиляються серверами (за замовчуванням). Поєднання цих двох вразливостей може дозволити зловмиснику повністю скомпрометувати аутентифікацію і таким чином видавати себе за обраний сервер.

Ось короткий огляд кроків експлуатації:

1. Встановити незахищений канал Netlogon з контролером домену, виконавши атаку грубою силою, використовуючи 8 нульових байтів виклику та шифротексту, видаючи себе за той самий контролер домену. Це потребуватиме в середньому 256 спроб (з огляду на ймовірність успіху 1 до 256).

2. Використати виклик NetrServerPasswordSet2 для встановлення пароля облікового запису контролера домену, збереженого в Active Directory, на порожній. Це порушує деякі функції контролера домену, оскільки пароль, збережений у реєстрі контролера домену, не змінюється (з цієї причини виконується четвертий крок, зазначений нижче).

3. Використати порожній пароль для підключення до того самого контролера домену та вивантаження додаткових хешів за допомогою протоколу Domain Replication Service (DRS).

4. Повернути пароль контролера домену до початкового, як збережено в локальному реєстрі, щоб уникнути виявлення.

5. Використати хеші, вивантажені на етапі 3, для виконання будь-якої бажаної атаки, наприклад, атаки "Золотий квиток" або передавання хешу з використанням облікових даних адміністратора домену [14][15].

Zerologon використовується як швидкий шлях до повної компрометації домену. Оскільки ця вразливість може викликати відмову в обслуговуванні, на реальних проектах її ні в якому разі не можна експлуатувати, а потрібно лише підтвердити наявність за допомогою допоміжних утиліт, які просто засвідчують наявність вразливості.

CVE-2014-1812. CVE-2014-1812 – це вразливість в реалізації групових

політик у Microsoft Windows, яка може дозволити підвищення привілеїв, якщо для розповсюдження паролів по домену використовуються Active Directory Group Policy Preferences (GPP). Group Policy Preferences (GPP) дозволяють системним адміністраторам встановлювати паролі за допомогою різних розширень GPP, таких як локальний користувач і група, змонтовані диски, служби, заплановані завдання та джерела даних. Наприклад, групова політика може встановити пароль локального адміністратора для всіх пристроїв, приєднаних до домену, в межах дії об'єкта групової політики (GPO).

Проблема виникає, коли ці паролі зберігаються у файлах XML в системному томі (SYSVOL) на контролерах домену. Ці паролі знаходяться в полі cpassword. Основна проблема полягає в тому, що ці паролі не зашифровані, а лише злегка закамуфльовані. Це означає, що автентифікований користувач може виявити та отримати доступ до цих паролів, просто переглянувши SYSVOL.

Наслідком експлуатації цієї вразливості є підвищення привілеїв. Зловмисник, який має доступ до домену, може знайти XML-файли в SYSVOL, отримати з них паролі та використати їх для підвищення своїх привілеїв. Наприклад, отримання паролю локального адміністратора дозволяє зловмиснику отримати повний контроль над усіма машинами, які цей пароль використовують. Виявлення та використання цих паролів може призвести до повної компрометації мережі, що дозволяє зловмиснику виконувати будь-які дії від імені адміністратора на всіх підключених пристроях.

Вразливість протоколу WDigest. Digest Authentication – це протокол виклику/відповіді, який в основному використовувався в Windows Server 2003 для автентифікації через LDAP та веб-автентифікацію. Він використовує обміни Hypertext Transfer Protocol (HTTP) та Simple Authentication Security Layer (SASL) для автентифікації.

На високому рівні процес виглядає так: клієнт запитує доступ до чогось, сервер автентифікації викликає клієнта, а клієнт відповідає на виклик, шифруючи свою відповідь ключем, отриманим з пароля. Шифрована відповідь порівнюється зі збереженою відповіддю на сервері автентифікації, щоб визначити, чи має користувач правильний пароль.

WDigest зберігає паролі у відкритому текстовому форматі в пам'яті. Тому, якщо зловмисник має доступ до кінцевого пристрою, він може використати інструмент, такий як Mimikatz, щоб отримати не лише хеші паролів, які зберігаються в пам'яті, але й самі паролі у відкритому тексті. У результаті зловмисник не буде обмежений атаками типу Pass-the-Hash; він також зможе увійти до Exchange, внутрішніх веб-сайтів та інших ресурсів, що потребують введення ідентифікатора користувача та паролях[16]. Така вразливість надає зловмиснику легкий шлях для горизонтального переміщення по доменній мережі, або взагалі до повної компрометації домену.

Важливо зазначити, що для проведення цієї атаки потрібні привілеї локального адміністратора на системі, оскільки взаємодія відбуватиметься з системним процесом LSASS. Це стосується переважно застарілих версій Windows. У новіших версіях Windows, починаючи з Windows 8.1 і Windows Server 2012, LSASS не дозволяє зберігати облікові дані у вигляді відкритого тексту. Однак це обмеження можна обійти шляхом модифікації реєстру.

Вразливість до LLMNR/ NBT-NS Poisoning. LLMNR (Link-Local Multicast Name Resolution) — це протокол, який дозволяє вузлам як з IPv4, так і з IPv6 виконувати розв'язання імен для вузлів в одній локальній мережі без необхідності використання DNS-сервера або налаштування DNS.

Коли DNS-запит вузла зазнає невдачі (тобто DNS-сервер не знає імені), вузол транслює LLMNR-запит у локальній мережі, щоб дізнатися, чи може будь-який інший вузол відповісти на нього.

LLMNR є наступником NetBIOS. NetBIOS (Network Basic Input/Output System) — це старіший протокол, який широко використовувався в ранніх версіях мереж Windows. NBT-NS є компонентом NetBIOS через TCP/IP (NBT) і відповідає за реєстрацію та розв'язання імен. Як і LLMNR, NBT-NS є резервним протоколом, коли розв'язання DNS зазнає невдачі. Він дозволяє локальне розв'язання імен у межах LAN.

LLMNR не має механізму аутентифікації. Будь-хто може відповісти на запит LLMNR, що відкриває можливості для потенційних атак. Коли комп’ютер

намагається розв'язати доменне ім’я і не вдається зробити це стандартними методами (наприклад, DNS), він надсилає LLMNR-запит у локальну мережу. Зловмисник може прослуховувати ці запити та відповідати на них, що призводить до потенційного несанкціонованого доступу [17].

Кроки експлуатації вразливості LLMNR:

1) Зловмисник запускає спеціалізований інструмент, який прослуховує запити LLMNR у локальній мережі.
2) В мережі відбувається подія, під час якої комп’ютер не може розв'язати ім'я через DNS і відправляє LLMNR-запит.
3) Інструмент зловмисника перехоплює цей запит і відповідає на нього так, ніби він є легітимним пристроєм у мережі.
4) Зловмисник отримує конфіденційну інформацію від жертви, таку як IP-адреса, доменне ім'я, ім'я користувача та хеш пароля.
5) Використовуючи отриманий хеш пароля, зловмисник намагається зламати його, щоб отримати справжній пароль користувача.

На рисунку показана схема процесу експлуатації вразливості LLMNR:

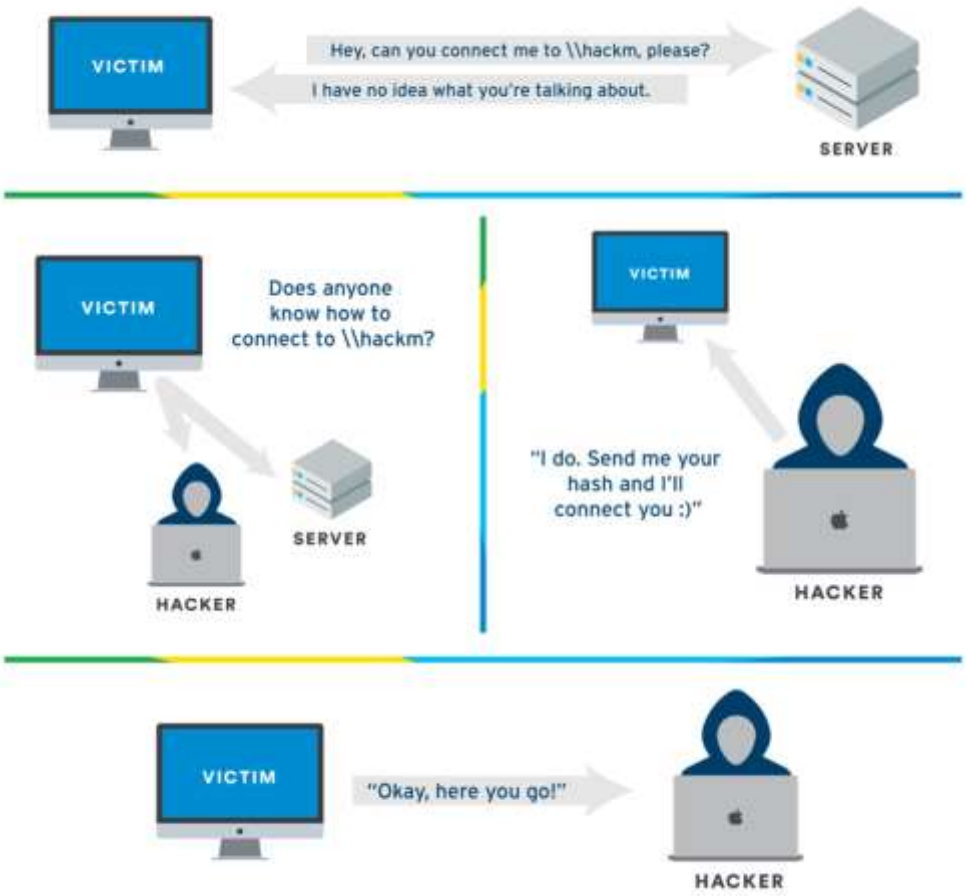

Рис. 1.5 – Процес експлуатації вразливості LLMNR [17]

В результаті експлуатації вразливості LLMNR зловмисник може отримати доступ до конфіденційної інформації та потенційно проникнути в мережу організації, що призводить до серйозних загроз безпеці.

2 АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ТЕСТУВАННЯ ЗАХИЩЕННОСТІ СЛУЖБИ КАТАЛОГІВ ACTIVE DIRECTORY

2.1 Огляд можливостей утиліти та отримання тримання початкового доступу за допомогою утиліти Responder

У цій роботі досліджено можливості зловживання стандартними службами розпізнавання імен LLMNR та NBT-NS у Microsoft Windows з метою викрадення автентифікаційних даних, отримання початкового доступу або здійснення бічного переміщення. Подібні атаки часто застосовуються в середовищі Active Directory з використанням утиліти Responder.

Responder – це потужна утиліта для атак на базі мережевих протоколів розроблена Trustwave SpiderLabs. Вона надає можливість перехоплювати та аналізувати різні мережеві протоколи, такі як LLMNR, NBT-NS, MDNS (Multicast DNS) та багато інших. Утиліта Responder дозволяє здійснювати різноманітні атаки на автентифікацію, що робить її важливим інструментом як для тестування безпеки, так і для потенційних зловмисників.

Утиліта Responder має такі ключові особливості:

- Методи отруєння: Responder використовує різні методи отруєння для маніпуляції мережевим трафіком, включаючи отруєння NetBIOS і LLMNR. Відповідаючи на мережеві запити зловмисними відповідями, Responder може захоплювати облікові дані та іншу конфіденційну інформацію.
- Пасивне виявлення мережі: Responder може пасивно ідентифікувати активні вузли в мережі, спостерігаючи за обміном даними між пристроями. Ця функція є незамінною для етичних хакерів, які прагнуть створити карту мережі та виявити потенційні цілі для подальшого аналізу.
- Збір облікових даних: Однією з основних функцій Responder є збір облікових даних, що передаються під час спроб аутентифікації. Це включає у себе паролі у відкритому вигляді, які етичні хакери можуть використовувати для оцінки безпеки мережі та виявлення потенційних слабких місць.

- Інтеграція з іншими інструментами: Responder розроблений для безперебійної роботи з іншими інструментами для етичного злому. Інтеграція з такими фреймворками, як Metasploit, підвищує його можливості, дозволяючи етичним хакерам автоматизувати завдання та спрощувати загальний процес тестування на проникнення [18].

Нижче наведено короткий опис реалізовані функції Responder:

- Вбудований сервер автентифікації SMB: Responder підтримує хеші NTLMv1, NTLMv2 з розширеною безпекою NTLMSSP за замовчуванням. Ця функціональність успішно працює на платформах від Windows 95 до Server 2012 RC, а також на Samba та Mac OS X Lion. Паролі у відкритому тексті підтримуються для NT4, а пониження хешування LM можливе при встановленні опції --lm. Ця функція активується автоматично при запуску інструмента.
- Вбудований сервер автентифікації MSSQL: Для перенаправлення SQL автентифікації на цей інструмент необхідно встановити опцію -r для систем старших за Windows Vista, оскільки LLMNR використовується для Vista та новіших версій. Цей сервер підтримує хеші NTLMv1 та LMv2. Функціональність успішно протестована на Windows SQL Server 2005 та 2008.
- Вбудований сервер автентифікації HTTP: Для перенаправлення HTTP автентифікації на цей інструмент необхідно встановити опцію `-r` для Windows версій старших за Vista, оскільки NBT-NS запити для пошуку HTTP сервера відправляються з використанням суфіксу імені сервісу Workstation (NetBIOS імені). Для Vista та новіших версій використовується LLMNR. Цей сервер підтримує хеші NTLMv1, NTLMv2 та базову автентифікацію. Успішно працює з Internet Explorer версіях 6 до 10, Firefox, Chrome та Safari. Цей модуль також працює для WebDav NTLM автентифікації, що випускається клієнтами Windows WebDav (WebClient), що дозволяє надсилати користувацькі файли жертві.
- Вбудований сервер автентифікації HTTPS: Аналогічно до попереднього

пункту, папка «certs/» містить два стандартні ключі, включаючи фіктивний приватний ключ, що дозволяє Responder працювати "з коробки". Також, можна використовувати власну пару ключів самопідписаного сертифіката.

- Вбудований сервер автентифікації LDAP: Для перенаправлення LDAP автентифікації на цей інструмент необхідно встановити опцію `-r` для Windows версій старших за Vista, оскільки NBT-NS запити для пошуку LDAP сервера відправляються з використанням суфіксу імені сервісу Workstation (NetBIOS імені). Для Vista та новіших версій використовується LLMNR. Цей сервер підтримує хеші NTLMSSP та просту автентифікацію (автентифікація у відкритому тексті).
- Вбудовані сервери автентифікації FTP, POP3, IMAP, SMTP: Ці модулі збирають облікові дані у відкритому тексті.
- Вбудований DNS сервер: Цей сервер відповідає на запити типу A. Він дуже зручний у поєднанні з ARP спуфінгом.
- Вбудований WPAD Proxy сервер: Цей модуль захоплює всі HTTP запити від будь-кого, хто запускає Internet Explorer в мережі з увімкненою опцією «Автоматично визначати параметри». Цей модуль є дуже ефективним. Можна налаштувати власний PAC скрипт у Responder.conf та впроваджувати HTML у відповіді сервера.
- Прослуховувач браузера: Цей модуль дозволяє знайти основний контролер домену (PDC) у скритому режимі.
- Зняття відбитків: При використанні опції -f Responder знімає відбитки з кожної системи, яка сгенерувала запит LLMNR/NBT-NS. Усі модулі захоплення працюють у режимі зняття відбитків.
- Режим аналізу: Цей модуль дозволяє бачити запити NBT-NS, BROWSER, LLMNR, DNS у мережі без отруєння відповідей. Також можна пасивно картографувати домени, MSSQL сервери, робочі станції та бачити, чи можливі атаки ICMP Redirect на вашій підмережі[19].

Тепер переходимо до практичного застосування атаки LLMNR Poisoning та розглянемо наступні атаки: LLMNR/NBT-NS Poisoning через SMB, атаки

LLMNR/NBT-NS Poisoning через WPAD та пониження рівня захисту NTLMv2-SSP до NTLMv2.

LLMNR/NBT-NS Poisoning через SMB: Коли клієнтська система Windows не може вирішити ім'я хоста за допомогою DNS, вона використовує протокол LLMNR для запиту у сусідніх комп'ютерів. LLMNR може використовуватись для вирішення як IPv4, так і IPv6 адрес. Якщо вирішення за допомогою LLMNR зазнало невдачі, система переходить до використання NBT-NS. Цей протокол аналогічний до LLMNR і служить для тієї ж мети, але працює виключно з IPv4. У випадках, коли для вирішення запиту використовуються LLMNR або NBT-NS, будь-який хост у мережі, який знає IP-адресу потрібного хоста, може відповісти. Навіть якщо хост відповідає на один із цих запитів з неправильною інформацією, відповідь все одно буде вважатися легітимною.

Утиліта Responder відповідає на запити LLMNR та NBT-NS, вказуючи власну IP-адресу як призначення для будь-якого запитуваного імені хоста.

Responder включає в себе SMB та HTTP сервери, на які користувачі можуть автентифікуватися. При автентифікації, Responder перехоплює облікові дані, включаючи хеші паролів які потім можна використовувати для відновлення паролю шляхом проведення brute-force атак або, якщо не вділось відновити пароль та в середовищі Active Directory використовується застаріла версія протоколу NTLM атак типу pass-the-hash. Нижче наведена схема того, як це працює:

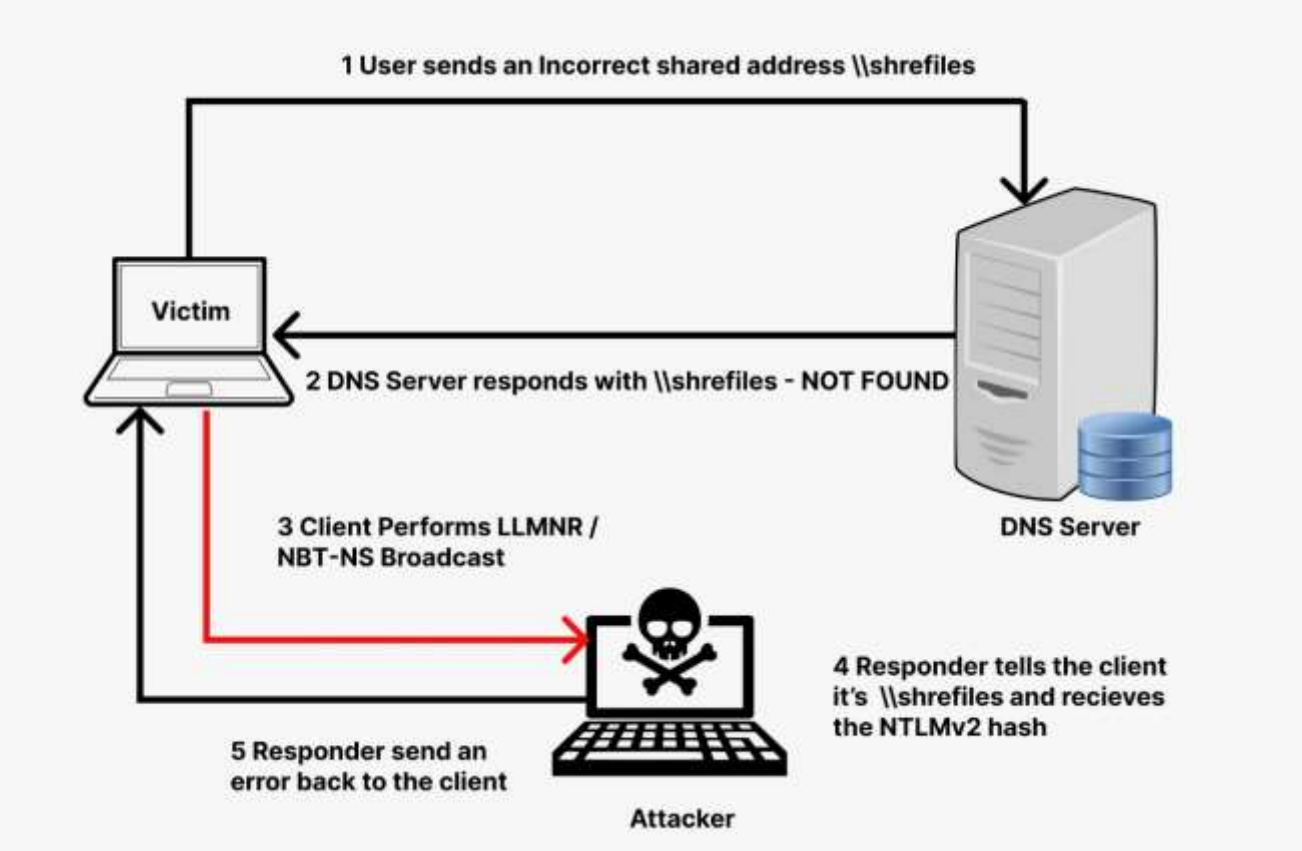

Рис.2.1 – Відбувається атака LLMNR/NBT-NS Poisoning [20]

Для експлуатації LLMNR/NBT-NS Poisoning потрібно запустити Responder з ключем -I та вказати назву мережевого інтерфейсу на якому буде запущене прослуховування та підняття необхідних зловмисних серверів. Важливо зазначити, що запускати утиліту необхідно з високими привілеями, адже вказані зловмисні сервери працюють на зареєстрованих портах.

```
kali@core: ~
File   Actions   Edit   View   Help
[*] [MDNS] Poisoned answer sent to 192.168.1.115   for name backsup.local
[*] [LLMNR]  Poisoned answer sent to 192.168.1.115 for name backsup
[SMB] NTLMv2-SSP Client   : 192.168.1.115
[SMB] NTLMv2-SSP Username : SYSTEM\John
[SMB] NTLMv2-SSP Hash     : John::SYSTEM:8a9e505ba616b7de:0130353BB67BDAF1A
701016F5F991F19:010100000000000C0653150DE09D201B39AAD663F43E0F900000000020
0080053004D0042003300010001E00570049004E002D0050005200480034003900320052005100
0041004600560004001400530004D0042003300 2E006C006F00630061006C000300340057004
9004E002D0050005200480034003900320052005100410046005600 2E0053004D0042003300
2E006C006F00630061006C00050014005300 4D0042003300 2E006C006F00630061006C00070
00800C0653150DE09D2010600040002000000080030003000000000000000010000000020000
006C2EDF7BB1941C6B84581C61C525DB0FC585CF1FFF552513FFBDBB9AA588713E0A0010000
0000000000000000000000000000000000900180063006900660073002F00620061006300 6B00
730075007000000000000000000
```

Рис 2.2 – Перехоплення хешу NTLMv2-SSP користувача John

Потім, необхідно зберегти отримані NTLMv1 хеші у файлі і використати програму hashcat для їх розшифрування. Для злому NTLMv1 хешів потрібно використовувати модуль, ідентифікований номером 5600.

Існують різні методи атаки для відновлення паролів за хешами, включаючи перебір за словником зі слабкими паролями та атаку за маскою. Однак, для використання останнього методу потрібна дуже потужна обчислювальна система, яка зможе перебрати велику кількість комбінацій за короткий час. Якщо парольна політика компанії включає в себе використання літер різних регістрів, цифр та спеціальних символів, то атака за маскою може бути дуже ефективною.

LLMNR/NBT-NS Poisoning через WPAD: WPAD (Web Proxy Auto-Discovery Protocol) – це протокол, що дозволяє браузерам автоматично знаходити і підключатися до кешуючих сервісів у мережі для швидшої доставки інформації. За замовчуванням WPAD використовує DHCP для виявлення кеш-сервісів, що спрощує процес підключення і розв'язання імен.

В організаціях, які використовують сервер WPAD, браузери отримують однакові конфігурації проксі-сервера через файл wpad.dat. Таким чином, будь-який запит з будь-якого браузера у корпоративній мережі спочатку знаходить wpad.dat, читає конфігурацію і потім надсилає запит за призначенням.

Коли вводиться некоректна URL-адреса і браузер не може завантажити сторінку через DNS, він надсилає запит LLMNR для пошуку проксі-сервера WPAD. Це трапляється в браузерах з увімкненою функцією "автоматичне визначення конфігурації", що часто використовується в корпоративних мережах для маршрутизації трафіку через проксі-сервер. Після цього браузер запитує файл wpad.dat, що містить дані для автоконфігурації проксі.

Зловмисник, відповідаючи на запит LLMNR, може створити фальшивий проксі-сервер WPAD, підробити запит і повідомити браузеру, що він має файл wpad.dat, запитуючи при цьому аутентифікацію. Коли користувач вводить свої облікові дані, їхні хеші потрапляють до зловмисника. Для налаштування фальшивого проксі-сервера WPAD використовується опція -w.

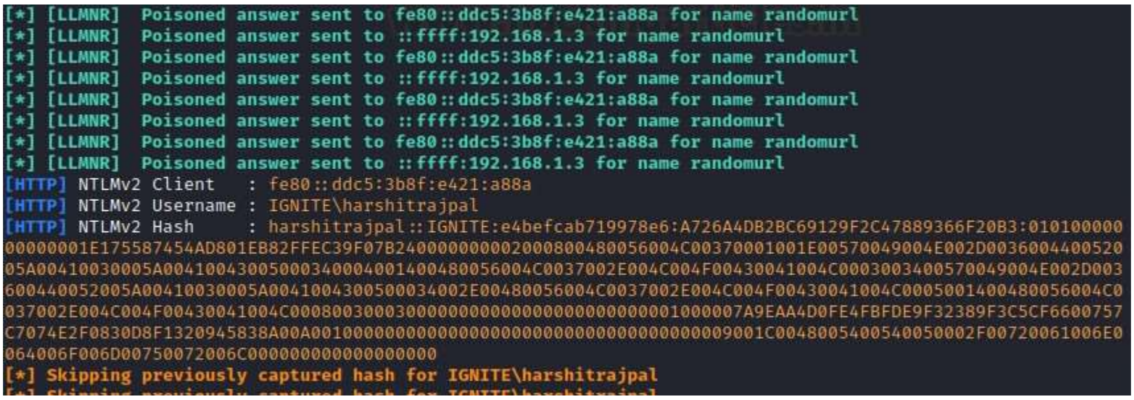

Рис. 2.4 – Перехоплення хешу NTLMv2-SSP у результаті успішної атаки LLMNR/NBT-NS Poisoning через WPAD

Коли користувач вводить некоректну URL-адресу, наприклад, randurl.local, браузер не може її знайти. Відповідач надсилає фальсифіковану DHCP-відповідь з IP-адресою WPAD зловмисника, і браузер намагається пройти аутентифікацію на сервері WPAD, видаючи запит на вхід у систему.

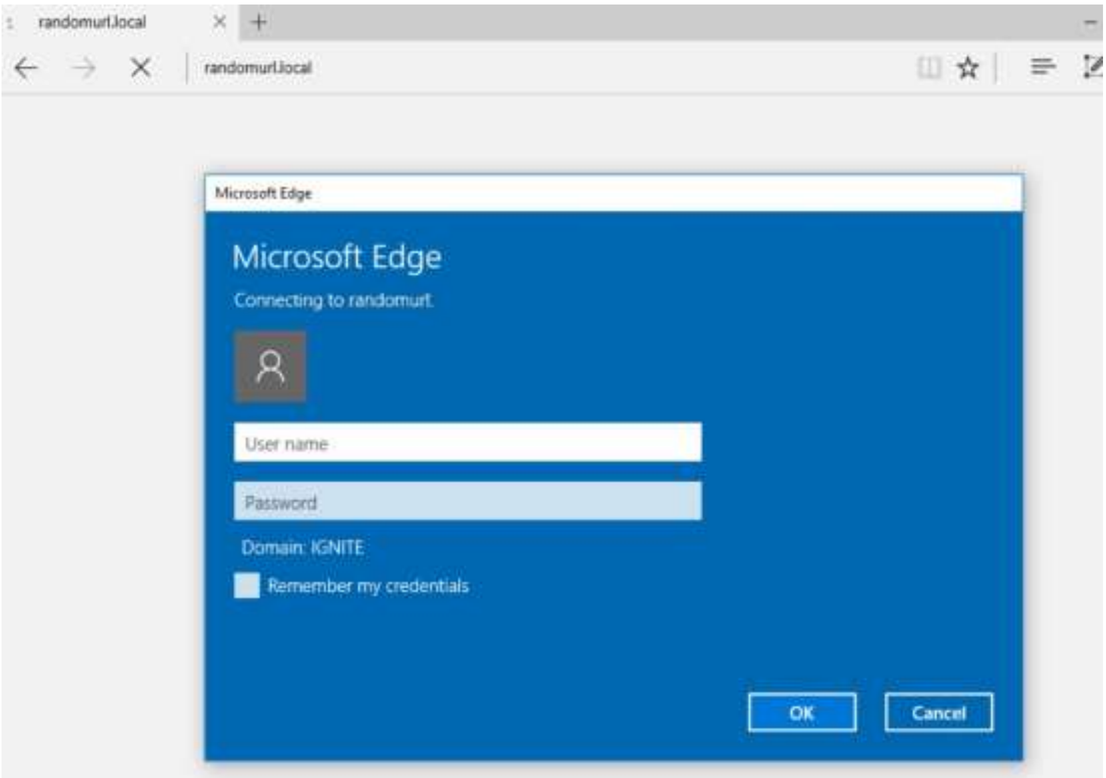

Рис. 2.5 – Запит на проходження аутентифікації на фальсифікованому сервері WPAD

Додатково, через цю атаку можна примусити користувачів мережі, які вводять некоректну URL-адресу, пройти базову автентифікацію, що дозволить отримати їхні облікові дані у відкритому вигляді. Для цього потрібно вказати опцію

-b, яка вказує на використання базової автентифікації, разом з опцією -F, яка використовується для примусової автентифікації.

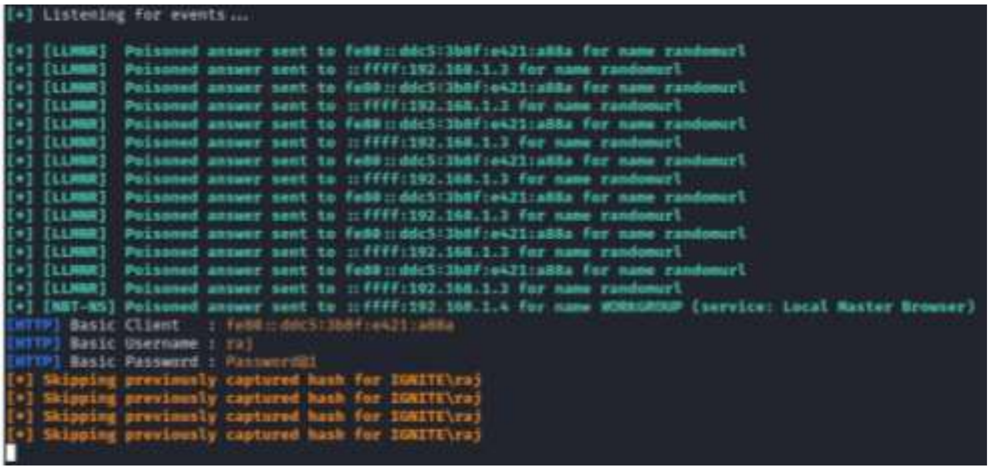

Рис. 2.6 — Перехоплення облікових даних у відкритому вигляді

Пониження рівня захисту NTLMv2-SSP до NTLMv2. Ця атака дозволить простіше розшифрувати перехоплений хеш:

У більшій кількості випадків, перехоплені хеші будуть формату NTLMv2-SSP. Застосування розширеного захисту сеансу (ESS) у NTLMv2, що додає до хеша прапорець «SSP», робить цей хеш більш складним для розшифрування.

У разі використання Responder для атаки на мережу, можна знизити рівень захисту NTLMv2-SSP до простого NTLMv2, що спрощує процес розшифрування хешів. Для досягнення цього мета необхідно використати прапорець –disable-ess, який вимикає функціонал ESS в Responder.

У Responder також існує прапорець --lm, який намагається примусити аутентифікацію NTLM до версії 1 замість версії 2. Проте, в новіших версіях Windows та Windows Server це неможливо. Таким чином, у випадку атаки використовується процедура атаки 2 з прапорцем --disable-ess.

Застосування простого NTLMv2 без ESS спрощує розшифрування хешів, що робить атаку з використанням Responder більш ефективною. Однак, важливо враховувати можливі обмеження, зокрема неможливість примусового переходу до NTLMv1 в певних операційних системах.

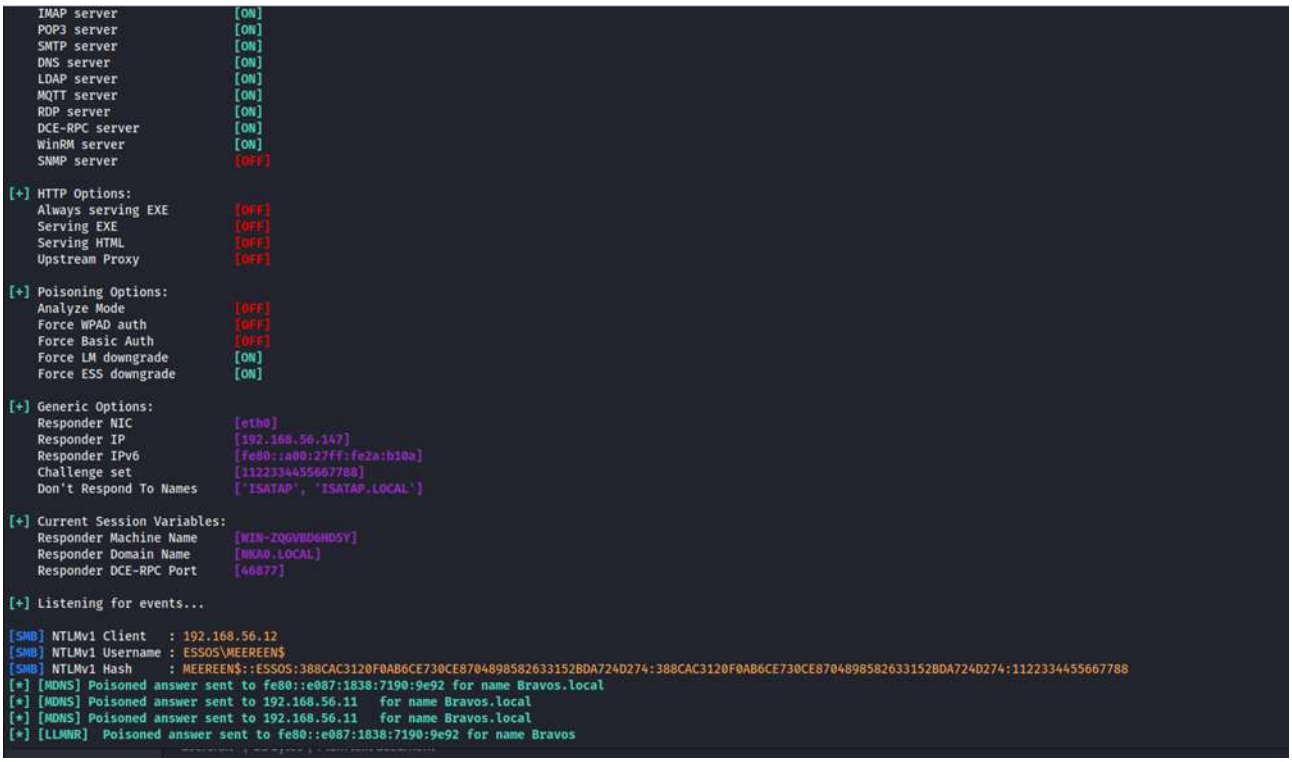

Рис. 2.6 — Результат спрацювання функції примусової аутентифікації NTLM версії 1

2.2 Аналіз структури організації та пошук шляхів до компрометації домена за допомогою утиліти Bloodhound

BloodHound є потужним інструментом для аналізу мережі та виявлення привілеїв у середовищі Active Directory. Він використовує теорію графів для виявлення складних зв'язків між об'єктами в AD та дозволяє як атакуючим, так і захисникам отримувати глибше розуміння структури привілеїв.

BloodHound – це веб-додаток на одній сторінці з базою даних Neo4J, що використовує теорію графів. У даній системі використовуються такі поняття:

- Вузли (Nodes): Представляють об'єкти AD, такі як користувачі, комп'ютери, групи тощо.
- Ребра (Edges): Представляють взаємозв'язки між об'єктами AD.
- Шляхи (Paths): Уявні ланцюги, що об'єднують вузли та ребра, і показують атакуючі маршрути.

Для роботи з BloodHound необхідно налаштувати базу даних Neo4J і

використовувати утиліти для збору даних, які збирають інформацію з домену та перетворюють її у зручний для BloodHound формат JSON. Потім ці файли можна імпортувати у веб-додаток BloodHound для подальшого аналізу доменної мережі. Зібрати всю інформацію про домен можливо лише за допомогою отримання первинних облікових даних із будь-якими привілеями. Це одна з найцікавіших особливостей у службі каталогів Active Directory, оскільки будь-який користувач може отримати всю інформацію про домен. Така функціональність дозволяє аналізувати структуру привілеїв та виявляти потенційні вразливості.

Інтерфейс BloodHound дозволяє вибирати вузли та запитувати шляхи між ними. Це дає можливість виявляти складні зв'язки та аналізувати маршрути, які можуть використовуватися для компрометації домену.

Як тільки налаштований екземпляр Neo4J та створена тестова база даних, BloodHound можна використовувати для представлення домену, наприклад, NORTH.SEVENKINGDOMS.LOCAL. Узли можуть бути запитані через рядок пошуку. Важливо зазначити, що узли відповідають об'єктам Active Directory.

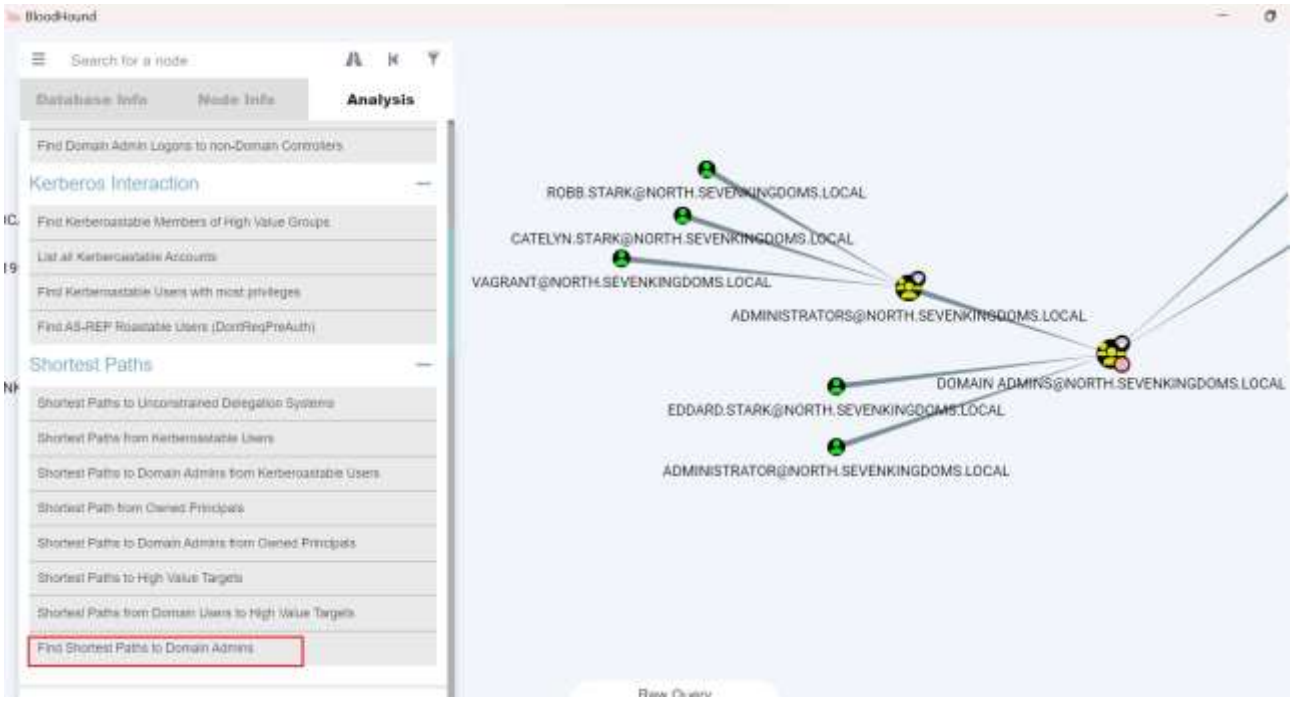

Рис.2.7. – Відображення коротких шляхів до компрометації доменного адміністратору

Вкладка «Analysis» дозволяє проаналізувати деякі вразливості служби каталогів Active Directory та наявність коротких шляхів до компрометації. В цьому випадку

шлях, що включає початкові узли та групу «Адміністратори домену», показує наступну інформацію:

1. Користувачі Robb.Stark, Catelyn.Stark та Vagrant є членами групи Administrators, яка. Це підтверджується описом самої групи та відміткою MemberOf, що вказує на приналежність користувачів до цієї групи.
2. Ця група має повний і необмежений доступ до комп'ютера або домену, на що вказує опис самої групи коли на неї натиснути.
3. WriteOwner вказує на те, що власники об'єктів зберігають можливість змінювати дескриптори безпеки об'єктів незалежно від дозволу на DACL об'єкта. Це означає, що існує можливість вилучення облікових даних цього користувача.
4. Відмітка WriteOwner вказує на те, що об'єкт (наприклад, користувач або група) має право змінювати власника іншого об'єкта в Active Directory. Це означає, що користувачі з правами WriteOwner можуть призначити себе або будь-кого іншого власником певного об'єкта, що надає їм повний контроль над цим об'єктом. У даному випадку, ми можемо додати цих користувачів до групи Domain Admins.
5. Користувачів Robb.Stark та Catelyn.Stark можна спробувати скомпрометувати шляхом виконання атаки типу Password Spraying, атак типу MiTM, соціальної інженерії, або шляхом додавання до SPN, щоб отримати зашифрований TGS (якщо на внесення цих змін є відповідні привілеї). Потім по цьому білену можна відновити пароль шляхом використання атаки типу брутфорс за словником або шаблоном.
6. Користувач Vagrant може мати пароль за замовчуванням. Загалом, необхідно перевіряти всі службові облікові записи або облікові записи для адміністрування на наявність встановлених паролів за замовчуванням.
7. Також можемо помітити, що іншим коротким шляхом до компрометації є компрометація користувачів Eddard.Stark та Administrator, які належать до групи Domain Admins, як показує нам BloodHound. Проте цей шлях може бути складнішим, оскільки такі облікові записи зазвичай краще захищені у

більшості випадків.

В загальному, потрібно максимально більше зібрати користувачів/систем з цікавими та небезпечними привілеями чи можливостями, які потім необхідно буде добавити в цілі.

Як приклад, відобразимо всіх користувачів з правами на виконання DCSync за допомогою стандартного запиту в Bloodhound:

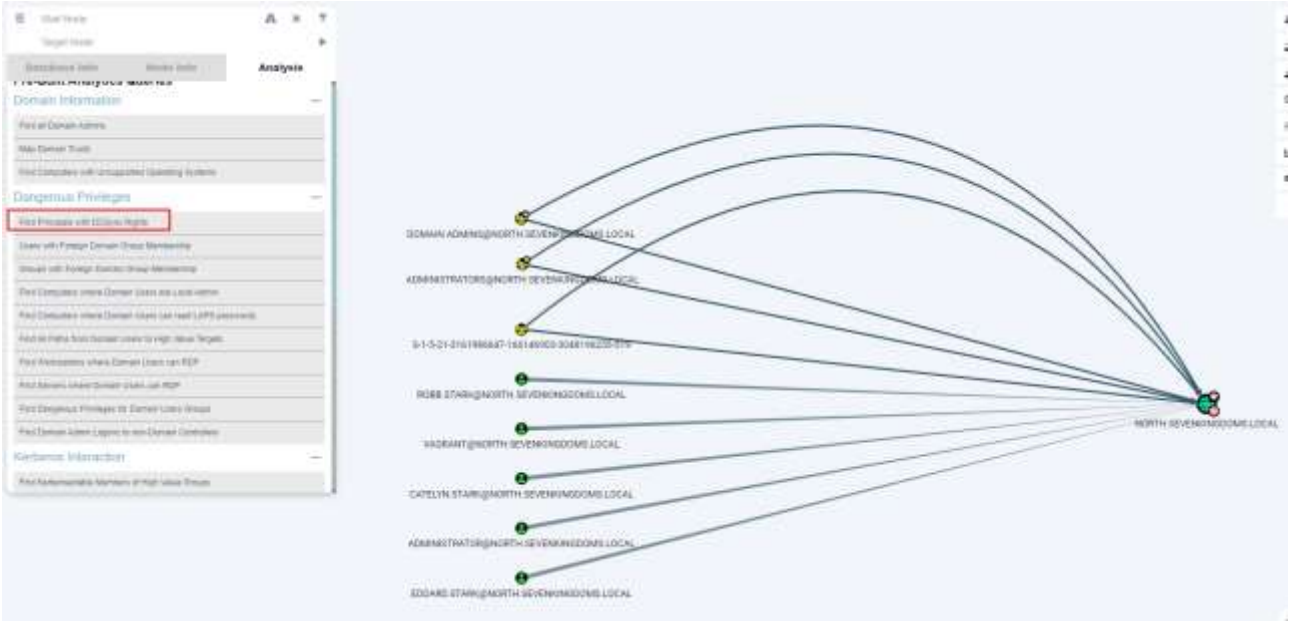

Рис 2.8 — Відображення користувачів з правами на виконання DCSync

З цих результатів можемо зробити висновок, що важливо ще також зосередитись на користувачах, які мають права на виконання DCSync. DCSync – це атака, яка дозволяє зловмисникам імітувати процес реплікації файлу ntds.dit з віддаленого контролера доменів (DC) та запитувати облікові дані від іншого DC [21]. Це дозволяє отримати доступ до облікових даних, таких як хеші паролів, без необхідності надавати повних адміністративних прав в межах домену. Найбільш критичним є те, що не потрібно намагатися відновити паролі за отриманими хешами користувачів. Потрібен лише хеш облікового запису krbtgt, який дозволить підробити квитки TGT на будь-якого користувача в домені, або навіть неіснуючого. Таким чином, такий аналіз допоможе значно скоротити час на знаходження шляху до компрометації доменної мережі Active Directory.

Ще однією зручною та потужною особливістю BloodHound є можливість використовувати власні запити, що дозволяє точніше визначати необхідну

інформацію. Якщо за замовчуванням відсутні необхідні запити, можна створити власні запити за допомогою мови запитів Cypher. Такий підхід дозволяє отримувати більш гнучкий та індивідуально налаштований аналіз доменної мережі за конкретними потребами.

Cypher – це декларативна мова запитів графів, створена для того, щоб бути простою і виразною, даючи змогу висловлювати складні запити простим способом. Спрощеним способом опису запитів Cypher є такий формат: (NODE)-[EDGE]->(NODE), де EDGE – це тип зв'язку, що з'єднує обидва вузли.

Для тестування запитів Cypher можна використовувати поле Raw Query в інтерфейсі BloodHound.

2.3 Огляд можливостей підвищення доменних привілеїв та здійснення бічного переміщення за допомогою утиліти Mimikatz

Mimikatz – це інструмент для перегляду та крадіжки облікових даних (які зберігаються в пам'яті), генерації квитків Kerberos та виконання атак. Mimikatz є інструментом пост-експлуатації, який дозволяє зловмисникам витягувати чутливу інформацію з систем Windows, включаючи паролі, квитки Kerberos та паролі у відкритому тексті, збережені в пам'яті. Атаки, які він може виконувати, включають:

Credential Dumping: це тип атаки, який дозволяє отримати облікові дані, що зберігаються в системній пам'яті. В результаті такої атаки, зловмисник може отримати доступ до паролів у відкритому вигляді, особливо якщо використовуються протоколи WDigest, або модулі TsPkg і LiveSSP, які можна легко розшифрувати [22]. Також можливе отримання хешів паролів, квитків Kerberos та інших облікових даних. Ці дані можуть бути використані для бічного переміщення по мережі, а також для горизонтального та вертикального підняття привілеїв.

Pass The Hash: це атака, яка використовується для бічного руху, з використанням хеша NTLM користувача з одного комп'ютера для отримання доступу до іншого комп'ютера [23]. Mimikatz дозволяє отримати цей хеш та виконати Pass The Hash для переходу на іншу систему.

Over Pass The Hash: ця атака є поєднанням двох інших атак: «pass-the-hash» і «pass-the-ticket». Зловмисник використовує хеш NTLM облікового запису користувача, щоб отримати квиток Kerberos, який може бути використаний для доступу до мережевих ресурсів. Ця техніка корисна, якщо ви не можете отримати пароль у відкритому тексті для облікового запису, але потрібна аутентифікація Kerberos для досягнення цілі. Ця атака може бути використана для виконання дій на локальних або віддалених серверах [24]. За допомогою, Mimikatz можна створити такий квиток та здійснювати аутентифікацію.

Pass The Ticket: це техніка, яка також використовується для бічного переміщення, в якій зловмисники крадуть квиток Kerberos з одного комп'ютера і використовують його для отримання доступу до іншого комп'ютера, повторно використовуючи вкрадений квиток [25].

Golden Ticket (золотий квиток): Golden Ticket (золотий квиток): дозволяє зловмиснику створити аутентифікаційний квиток Kerberos TGT за допомогою NTLM-хешу скомпрометованого облікового запису служби, який називається krbtgt, за допомогою Mimikatz [26]. Всі квитки TGT підписуються хешом облікового запису krbtgt, що дозволяє переконатися в легітимності аутентифікаційного квитка, однак цим можна зловживати. За замовчуванням золоті квитки дійсні протягом десяти років або до скидання пароля KRBTGT домену. У великих середовищах Active Directory це трапляється рідко через проблеми, які виникають у зв'язку з цим. Таким чином, золоті квитки забезпечують довготривалу стійкість і їх важко виправити захисникам [27].

Silver Ticket: включає в себе компрометацію облікових даних та зловживання протоколом Kerberos. Це дозволяє зловмиснику підробляти квитки для служб, які надають послугу з видачі квитків (TGS), але лише для конкретних служб. Квитки TGS шифруються за допомогою хешу пароля відповідної служби. Тому, якщо зловмисник вкраде хеш облікового запису служби, він може створювати квитки TGS для цієї служби [28].

Mimikatz розроблений для експлуатації слабких місць у механізмах безпеки Windows, зокрема способу зберігання паролів у пам'яті. Він працює шляхом ін'єкції

в процес LSASS, який відповідає за управління обліковими даними на системі Windows. Після ін'єкції, Mimikatz може витягувати облікові дані з пам'яті, включаючи паролі у відкритому тексті, хеші та квитки Kerberos. Він також може виконувати атаки типу pass-the-hash, коли зловмисник використовує викрадені хеші паролів для автентифікації на віддаленій системі. Для коректної роботи з цією утилітою потрібно запустити її під високими привілеями в системі.

Розглянемо використання Mimikatz для вилучення облікових даних:

Модуль sekurlsa дозволяє взаємодіяти з процесом LSASS та витягувати автентифікаційні дані, включаючи паролі, квитки Kerberos та хеші NTLM. Команда logonpasswords витягує паролі у текстовому форматі та хеші NTLM, пов'язані зі входами користувачів.

```
mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 401254 (00000000:00061f66)
Session           : Interactive from 1
User Name         : Administrator
Domain            : MARVEL
Logon Server      : HYDRA-DC
Logon Time        : 5/1/2023 6:53:45 AM
SID               : S-1-5-21-1354542597-2677663135-2645692803-500
        msv :
         [00000003] Primary
         * Username : Administrator
         * Domain   : MARVEL
         * NTLM     : f56a8399599f1be040128b1dd9623c29
         * SHA1     : 3edb384812cbe4c90713bca316eb3739fe2541f1
         * DPAPI    : 94c0b43db9fa4fd76eabdc2d178441af
        tspkg :
        wdigest :
         * Username : Administrator
         * Domain   : MARVEL
         * Password : (null)
        kerberos :
         * Username : Administrator
         * Domain   : MARVEL.LOCAL
         * Password : (null)
        ssp :
        credman :

Authentication Id : 0 ; 64774 (00000000:0000fd06)
Session           : Interactive from 1
User Name         : DWM-1
Domain            : Window Manager
Logon Server      : (null)
Logon Time        : 5/1/2023 6:49:58 AM
SID               : S-1-5-90-0-1
```

Рис 2.9 — Результат виконання команди logonpasswords

Ще однією важливою функцією є вилучення бази даних SAM. Фактично, ми можемо витягнути хеші SAM за допомогою команди lsadump::sam, як показано нижче:

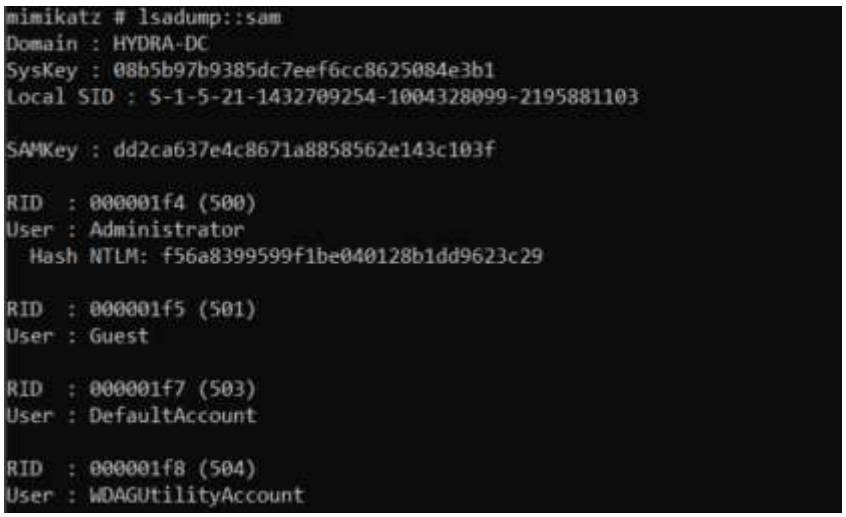

Рис 2.9 — Успішне вилучення хешів

Також, можна вилучити облікові дані і з процесу LSA. LSA (Local Security Authority) є захищеною підсистемою у системі аутентифікації Windows. Вона аутентифікує та створює журнали сеансів на локальному комп'ютері. В ній знаходяться імена користувачів разом з їх хешами NTLM. Цю інформацію можемо отримати за допомогою команди lsadump::lsa.

Після отримання облікових даних, наприклад, хешу, можна здійснити бічне переміщення за допомогою атаки Pass-The-Hash, використовуючи команду sekurlsa::pth. Після виконання цієї команди відкриється оболонка іншої системи, що дозволить продовжити подальші дії.

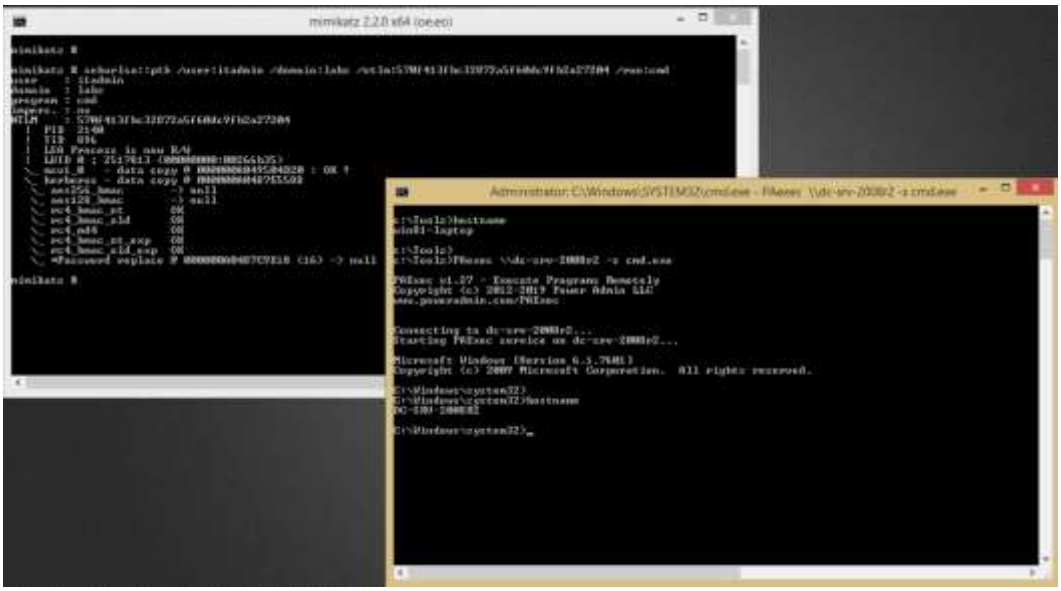

Рис. 2.10 — Успішне виконання атаки Pass-The-Hash та отримання оболонки іншої системи.

3 РОЗРОБКА РЕКОМЕНДАЦІЙ ЩОДО ЗАСТОСУВАННЯ МЕТОДІВ ТА ЗАСОБІВ ЗАХИСТУ КІНЦЕВИХ ТОЧОК ОРГАНІЗАЦІЇ

3.1. Методологія тестування на захищеність службі каталогів Active Directory

Дотримання послідовності в етапах тестування на захищеність відіграє важливу роль у знаходженні точок входу, просуванню в доменній мережі та компрометації служби каталогів Active Directory. Важливо також зазначити, що етапи є циклічними та продовжуватиметься це доки не буде досягнуто основної мети – повної компрометації зазначено домену.

Тестування на захищеність Active Directory включає в собі такі етапи:

- Розвідка.
- Отримання доступу.
- Перечислення.
- Локальне підвищення привілеїв.
- Бічне переміщення.
- Закріплення в домені.

Розвідка. Тестування в Active Directory з етапом розвідки зазвичай відноситься до випадку, коли у тестувальника немає облікових даних, які можна було б використати для перерахування об'єктів та вразливих конфігурацій служби каталогів Active Directory. Розвідка в цьому випадку поділяється на два етапи: зовнішню та внутрішню.

Зовнішня розвідка. На цьому етапі важливо знайти електронні пошти користувачів та можливі облікові дані по злитим базам даних. Додатково можна провести аналіз організації на сторінці LinkedIn, зокрема по співробітникам. Важливо, щоб на цьому етапі був готовий список можливих імен користувачів. Це може дуже допомогти у подальших етапах тестування.

Внутрішня розвідка. На цьому етапі необхідно зібрати якомога більше інформації про мережу, скласти карту мережі та визначити цікаві цілі, які можуть

знадобитися на наступному етапі. Зазвичай для цього завдання використовується утиліта Nmap, яка знаходить активні системи та додаткову інформацію про них. Важливо встановити режим виявлення версій операційних систем, версії сервісів, та додатково, який виконує зворотний DNS-запит за допомогою PTR-записів, щоб отримати доменне ім'я та FQDN (Fully Qualified Domain Name) систем. Ця інформація допоможе визначити призначення сегменту мережі, в якій наразі знаходиться система тестувальника – користувацька чи серверна. У більшості випадків в доменній мережі Active Directory налаштовуються зрозумілі назви, які визначають роль системи в доменній мережі.

Розуміння того, в якій мережі ми знаходимося, дає уявлення про те, які методи для отримання первинного доступу використовувати. Наприклад, для серверної мережі доцільніше використовувати різноманітні експлойти для характерних версій програмного забезпечення, тоді як для користувацької мережі доцільно буде запустити такий інструмент, як Responder, для перехоплення хешів.

Також, важливо буде знайти всі існуючі контролери домену мережі. Цю інформацію можна отримати проаналізувавши FQDN та запущені сервіси з номерами портів в результатах сканування утиліти Nmap. Зазвичай DC має прослуховує наступні порти:

- Порт 53 TCP/UDP - використовується для DNS.
- Порт 88 TCP/UDP - використовується для аутентифікації Kerberos.
- Порт 123 UDP - використовується для синхронізації часу W32Time.
- Порт 135 TCP - використовується для RPC Endpoint Mapper.
- Порти 137/138 UDP і Порт 139 TCP - використовуються для NetBIOS.
- Порт 389 TCP/UDP - використовується для LDAP.
- Порт 445 TCP - використовується для SMB.
- Порт 464 TCP/UDP - використовується для зміни пароля Kerberos.
- Порт 636 TCP - використовується для LDAP SSL.
- Порти 3268/3269 TCP - використовуються для LDAP Global Catalog / LDAP GC SSL.

Але самим швидким та точний результат може надати nslookup, створивши

спеціальний запит, який поверне імена та IP-адреси контролерів домену. Результат цього етапу повинен містити наступну інформацію:

Список імен користувачів організації або знайдені облікові дані;

- Ім’я домену;
- Призначення сегментів мережі;
- Карта мережі;
- Запущені сервіси, їх версії та відкриті порти;
- Версії операційних систем;
- Імена контролерів домену.

Етап розвідки є критично важливим першим кроком у тестуванні безпеки Active Directory (AD). На цьому етапі необхідно детально фіксувати знахідки, щоб забезпечити створення великої бази даних для подальших дій та виявлення можливих вразливостей.

Коли вся необхідна інформація зібрана, можна переходити до наступного етапу.

Отримання первинного доступу. Існує багато способів отримати дійсний обліковий запис користувача AD. Швидке рішення часто полягає у запуску такого інструменту, як Responder, щоб виконати LLMNR/NBT-NS poisoning, щоб отримати хеші паролів NTLMv1 або (більш поширений сьогодні) NTLMv2. Ці хеші можна або зламати офлайн за допомогою Hashcat. Також можна виконати атаку SMB Relay або атаку IPv6, щоб витягти дані домену або отримати локальний доступ адміністратора до системи.

Якщо виявимо, що один або більше контролерів домену налаштовані дозволяти SMB NULL сесії або анонімні зв’язки LDAP, то це дає можливість витягти список усіх користувачів AD разом із політикою паролів домену та здійснити атаку розпилювання паролів за допомогою такого інструменту, як Kerbrute. Якщо перші два варіанти не працюють, ми можемо використати список користувачів, який був складений на минулому етапі, для перечислення дійсних облікових записів користувачів AD за допомогою Kerbrute.

Якщо в компанії існує пароль за замовчуванням, який надається користувачеві для

першого входу в систему можна добавити його в список для розпилювання. Зазвичай достатньо скористатись пошуком в Google щоб знайти самі розповсюдженні слабкі паролі та обрати декілька, які дають більше шансів на успішне спрацювання. Якщо парольну політику ніяк не вдалось взнати, тоді краще перейти до інших методів, щоб не ризикувати блокуванням одного або багатьох облікових записів користувачів AD.

Якщо нам пощастить отримати політику паролів домену, зазвичай можна безпечно виконати декілька спроб розпилювання паролів, дотримуючись меж політики паролів і не блокуючи облікові записи. Нам потрібен лише один успіх (вдало підібраний пароль), щоб розпочати перечислення чи атаки на AD.

Якщо жоден із вищезазначених методів не спрацьовує, потрібно шукати вразливі версії сервісів чи операційної системи та скористатись експлойтом щоб отримати оболонку або отримання облікових даних.

Будь-яка кількість атак може бути використана для примушення до автентифікації NTLM. Один приклад – PetitPotam, який іноді може бути використаний для виконання атаки NTLM relaying без необхідності наявності дійсних облікових даних користувача AD, якщо AD CS присутній і налаштований на контролері домену, а не на окремому сервері.

Перечислення домену. Коли у нас є дійсний обліковий запис користувача Active Directory (AD), першим кроком зазвичай є ознайомлення з навколишнім середовищем та виявити потенційні вектори атак. Це можна зробити дуже ефективно за допомогою BloodHound і почати складати карти шляхів атак, таких як хости, на яких може бути локальний адміністратор, або потенційні користувачі/комп'ютери для цілей. Ми також можемо використовувати PowerView або багато інших інструментів для переліку інформації про ключові об'єкти AD та їх дозволи.

При перечисленні важливо зібрати таку інформацію:

- Короткий шлях до компрометації домену. Це можливо зробити за допомогою BloodHound. Важливо зафіксувати облікові записи, що проходять через цей шлях.

- Визначити імена користувачів доменних адміністраторів, щоб потім спробувати атаку з розпиленням паролю чи AS-REP Roasting.
- Звернути увагу на інформацію щодо того, коли останній раз змінювався пароль на цікавих для нас облікових записів. Якщо давно, то там може бути слабкий пароль.
- Виявити системи, де був залогінений доменний адміністратор, щоб у подальшому вилучити облікові дані з пам'яті.
- Перевірити наявність GPO та визначити, хто має можливість входити через RDP на контролер домену.
- Перевірити наявність користувачів типу віртуальних адміністраторів. Вони зазвичай мають високі привілеї та паролі за замовчуванням.
- Якщо у компанії використовується політика LAPS, то можна вилучити інформацію про дозволи на доступ до атрибута пароля LAPS, де зберігаються паролі у вигляді звичайного тексту (ms-Mcs-AdmPwd). З цією інформацією ми можемо визначити, хто має можливість переглядати паролі LAPS, і атакувати ці облікові записи.
- Перевірити наявність облікових записів SPN. Ці облікові записи можуть мати високі привілеї.
- AD CS представляє собою великий поверхню атаки, тому завжди варто перевірити, чи він присутній.

Локальне підняття привілеїв. Локальне підвищення привілеїв є важливим етапом пентестування Active Directory. На цьому етапі можна використати утиліту WinEnum для пошуку вразливих служб, некоректних налаштувань дозволів файлів або встановленого вразливого програмного забезпечення. Інша корисна утиліта, PowerUp, допомагає виявити та зловживати неправильною конфігурацією. Також можна перевірити файлову систему користувача на наявність збережених паролів у текстових файлах.

- Квитки TGT
- NTLM хеші
- Паролі у відкритому вигляді

Бічне переміщення. Бічне переміщення починається після отримання бази автентифікаційних даних. Після аналізу коротких шляхів, як згадувалось раніше, важливо проаналізувати, як скомпрометувати системи, що можуть призвести до компрометації доменного адміністратора. Результатом цього етапу повинен бути аналіз можливостей компрометування систем та оцінка шансів досягнення головної мети – компрометації домену.

Для здійснення бічного переміщення можна використовувати такі атаки, як Pass The Hash, Over Pass The Hash, Pass The Ticket, які можна виконати за допомогою тієї ж утиліти Mimikatz. Також, можна скористатися URL File Attacks або можливістю крадіжки активного сеансу RDP. Після проникнення на іншу систему можна використовувати різні протоколи, такі як RDP, WinRM, PsExec, WMI, для подальшого руху по мережі та здійснення атак.

Після отримання доступу до інших систем необхідно повернутися до етапу перечислення та підвищення привілеїв, щоб забезпечити дальше продвиження в мережі та досягнення поставлених цілей.

Підвищення привілеїв у домені. Служби сертифікації Active Directory (AD CS). Якщо AD CS присутній, то варто перелічити можливі неправильні налаштування за допомогою Certipy або Certify. Неправильна конфігурація може швидко призвести до доступу рівня Domain Admin або повного адміністративного контролю над доменом AD.

Атаки Roasting. Ми також можемо виконувати атаки типу "roasting", такі як Kerberoasting або ASREPRoasting. Успіх цих атак залежить від наявності слабкого пароля для облікового запису, але якщо ми можемо отримати та зламати хеш пароля, ми можемо отримати доступ до облікового запису, який допоможе нам просунутися далі до нашої мети або навіть безпосередньо до Domain Admin.

Інші можливі атаки. Ми можемо потенційно виконати атаку DACL, експлуатувати неправильно налаштовану GPO, використати членство в групі, що дозволяє доступ до віддаленого робочого столу (RDP) до хоста і потім підвищити привілеї, знайти пароль у полі опису користувача або комп'ютера, або знайти пароль у файлі параметрів групової політики.

Пошук даних у загальних ресурсах. Часта ситуація, коли користувачі можуть зберігати конфіденційні дані у спільних ресурсах, які зберігають всі свої паролі в текстовому файлі або електронній таблиці; файл web.config з паролем облікового запису служби MSSQL; резервна копія VMDK хоста, з якої ми можемо витягти хеш пароля локального адміністратора та повторно використовувати його на інших системах, і багато іншого. PowerView або CrackMapExec можуть бути використані для цієї мети.

Закріплення в домені. Закріплення в домені відіграє важливу роль у проведення тестування, оскільки це може демонструє реальний підхід зловмисника.

Одним з популярних варіантів закріплення в домені є створення Golden Ticket. Але це лише один з варіантів. Є багато інших способів, які можна використовувати.

- *Silver Ticket.* Можна отримати доступ до певних ресурсів без необхідності аутентифікації через контролер домену, але це забезпечує меншу видимість атаки порівняно з Golden Ticket та і дозволяє обійти деякі засоби моніторингу.
- *Skeleton Key.* Можна використати спеціальний модуль в Mimikatz для ін'єкції універсального пароля в пам'ять контролера домену. Це надає можливість входити в будь-який обліковий запис за допомогою цього універсального пароля.
- *AdminSDHolder.* Об'єкт AdminSDHolder в Active Directory забезпечує безпеку привілейованих груп (таких як Domain Admins і Enterprise Admins) шляхом застосування стандартного списку контролю доступу (ACL) до цих груп, щоб запобігти несанкціонованим змінам. Однак цією функцією можна зловживати змінюючи ACL об'єкта AdminSDHolder та надавши повний доступ та розширений контроль над усіма привілейованими групами звичайному користувачу.
- *Зловживання ACL.* Використовуючи права доменного адміністратору, можна надати спеціальні дозволи користувачеві, на прикладі GenericAll, що

фактично робить його на рівні доменного адміністратору.

- *DSRM.* Сама головна умова цього методу встановлення постійного доступу – що в організації не має стандарту по регулярній зміні паролю DSRM. Для успішної експлуатації потрібно вилучити NTLM хеш локального адміністратору та потім внести зміну в реєстрі надавши дозвіл входити за допомогою хешу DSRM.

Використання методів відмінних від Golden Ticket та Silver Ticket є більш опційним варіантом, оскільки вимагає змін у налаштуванні та відкриватиме нові вразливості. Тому краще з приводу цього етапу узгоджувати все з замовником та зрозуміти чого він очікує від проведення тестування.

3.2 Загальні рекомендації щодо оцінювання та усунення вразливостей

В більшості випадків у результаті тестування на проникнення використовується калькулятор CVSS для формування оцінки вразливості яка потім буде надана замовнику. Система оцінки загроз за уразливостями Common Vulnerability Scoring System (CVSS) зафіксовує основні технічні характеристики уразливостей програмного та апаратного забезпечення, а також вбудованого програмного забезпечення. Її результати включають числові показники, що вказують на важкість уразливості в порівнянні з іншими.

Переваги CVSS включають стандартизовану методику оцінки уразливостей, яка не залежить від постачальника чи платформи. Це відкритий фреймворк, що забезпечує прозорість щодо індивідуальних характеристик та методології, що використовуються для отримання оцінки.

CVSS складається з трьох груп метрик:

- Група базових метрик відображає внутрішні характеристики вразливості, які залишаються постійними з часом і в різних користувацьких середовищах. Вона складається з двох наборів метрик: метрик експлуатації та метрик впливу.

  Метрики експлуатації відображають легкість і технічні засоби, за допомогою

яких можна експлуатувати вразливість. Тобто вони представляють характеристики того, що є вразливим, що ми формально називаємо вразливим компонентом. Метрики впливу відображають прямий наслідок успішної експлуатації вразливості та представляють наслідки для того, що зазнає впливу, що ми формально називаємо вплинутим компонентом.

Хоча вразливим компонентом зазвичай є програмний додаток, модуль, драйвер тощо (або, можливо, апаратний пристрій), вплинутим компонентом може бути програмний додаток, апаратний пристрій або мережевий ресурс. Ця можливість вимірювання впливу вразливості, окрім вразливого компонента, була ключовою особливістю, введеною з CVSS версії 3.0. Ця властивість фіксується за допомогою метрики "Scope".

- Група тимчасових метрик відображає характеристики вразливості, які можуть змінюватися з часом, але не в різних користувацьких середовищах. Наприклад, наявність простого у використанні набору експлойтів підвищить CVSS-бал, тоді як створення офіційного патчу знизить його.
- Група метрик середовища відображає характеристики вразливості, які є релевантними та унікальними для конкретного користувацького середовища. До таких характеристик належать наявність засобів захисту, які можуть пом’якшити деякі або всі наслідки успішної атаки, та відносна важливість вразливої системи в технологічній інфраструктурі.

У цій роботі буде розглянуто оцінку Базової метрики, оскільки вона є найпоширенішою.

Для базових метрик існують метрики експлуатації – це підгрупа базових метрик, які оцінюють, наскільки легко можна експлуатувати вразливість. Вони включають вектор атаки, складність атаки та вимоги до автентифікації. Метрика експлуатації включає такі параметри:

Attack Vector: Ця метрика відображає контекст, в якому можливе використання вразливості. Значення цієї метрики (і відповідно, базовий бал) буде більшим, чим далі (логічно та фізично) може бути здійснений атакуючий для використання вразливої компоненти. Значення може бути:

- *Network (N):* Зловмисник може експлуатувати вразливість віддалено через мережу та не потребує фізичного доступу до системи і може здійснювати атаку з будь-якої точки в Інтернеті. Це найбільш небезпечний вектор атаки, оскільки він дозволяє зловмисникам атакувати системи з будь-якого місця в світі.
- *Adjacent (A):* Зловмисник повинен мати доступ до тієї ж мережі, що і цільова система, щоб експлуатувати вразливість. Для цього може використовуватись підключення по VPN або потрібно щоб атакуюча система була під'єднана до мережі.
- *Local (L):* Зловмисник повинен мати авторизований доступ до системи, щоб експлуатувати вразливість. Це може включати атаки, які використовують локальні ресурси системи, такі як файлові системи або місцеві служби.
- *Physical (P):* Зловмисник повинен мати безпосередній, фізичний доступ до системи, щоб експлуатувати вразливість. Це може включати атаки, які вимагають фізичного доступу до обладнання.

Attack Complexity оцінює, наскільки складно для зловмисника експлуатувати вразливість. Цей параметр має два можливі значення:

- *Low (L):* Атака не вимагає спеціальних умов. Наприклад, вразливість може бути використана за допомогою публічного експлойту, який не потребує особливих налаштувань. Це означає, що навіть люди без технічних знань можуть експлуатувати цю вразливість.
- *High (H):* Атака вимагає певних умов, які можуть складатися з різних факторів. Наприклад, вразливість може бути експлуатована лише за умови придбання експлойту, якого немає в публічному доступі, складне налаштування експлойту або для експлуатації вразливості потрібно спочатку експлуатувати іншу чи скомбінувати. Це також може включати самостійне написання коду експлойту чи складне налаштування інструментів для експлуатації.

Privileges Required в CVSS 3.1 оцінює рівень привілеїв, які зловмисник повинен мати перед успішною експлуатацією вразливості. Цей параметр має три

можливі значення:

- *None (N):* Зловмисник не має авторизації перед атакою, і тому не потребує доступу до попередньо системи для проведення атаки.
- *Low (L):* Зловмисник авторизований з привілеями, які надають базові можливості користувача.
- *High (H):* Зловмисник має високий рівень привілеїв, які зазвичай є рівня адміністратору чи системи.

Scope в CVSS 3.1 визначає, чи впливає експлуатація вразливості на компоненти системи, відмінні від того, де вразливість знаходиться. Цей параметр має два можливі значення:

- *Unchanged (U):* Вразливість не впливає на компоненти системи за межами того, де знаходиться вразливість. Наприклад, веб-додаток має SQL-ін'єкцію. Коли зловмисник експлуатує цю вразливість, він може отримати доступ до бази даних цього веб-додатку, але не може вийти за межі веб-додатку і вплинути на інші системи в мережі.
- *Changed (C):* Якщо «Scope» вказано як «Changed», це означає, що вразливість впливає на компоненти системи за межами того, де знаходиться вразливість. Наприклад, якщо веб-додаток має вразливість, яка дозволяє зловмиснику виконувати віддалено код на сервері з високими привілеями. Використовуючи цю вразливість, зловмисник може отримати доступ до всіх ресурсів, які виходять за межі ресурсів веб-додатка.

Метрики впливу в системі оцінки загальної вразливості (CVSS) відображають, як вразливість може вплинути на систему, якщо її вдалося експлуатувати. Вони включають:

- *Вплив на конфіденційність (Confidentiality Impact):* Ця метрика визначає, чи може вразливість призвести до витоку інформації з системи1. Наприклад, якщо атакувальник може отримати доступ до конфіденційних даних через вразливість, це матиме високий вплив на конфіденційність.
- *Вплив на цілісність (Integrity Impact):* Ця метрика визначає, чи може вразливість змінити або знищити дані в системі1. Наприклад, якщо

атакувальник може змінити дані в системі через вразливість, це матиме високий вплив на цілісність.

- *Вплив на доступність (Availability Impact):* Ця метрика визначає, чи може вразливість зупинити роботу системи або зробити її недоступною. Наприклад, якщо зловмисник може зупинити роботу системи через вразливість, це матиме високий вплив на доступність [29][30].

Система оцінки загальної вразливості (CVSS) дозволяє класифікувати вразливості за ступенем їх серйозності на шкалі від 0 до 10. Ця шкала допомагає зрозуміти, наскільки серйозна вразливість і як швидко її потрібно виправити. Нижче наведено категорії CVSS та рекомендації щодо дій для кожної з них.

None: 0.0: Вразливість не має впливу на систему. Це означає, що експлуатація такої вразливості не завдає жодної шкоди, не впливає на конфіденційність, цілісність або доступність системи.

Жодних заходів не вимагається. Оскільки така вразливість не представляє загрози та можна продовжувати нормальну роботу без змін.

Low: 0.1 - 3.9: Такі вразливості можуть бути прийняті організацією без вживання заходів щодо їх усунення, або навпаки можуть бути усунені. Вони мають незначний вплив на систему, і їх експлуатація зазвичай не призводить до значних наслідків.

Для вразливостей низького рівня потрібно оцінити контекст і вирішити, чи варто усувати вразливість. Якщо ресурсів або часу недостатньо, можна запланувати усунення на майбутнє або прийняти ризик.

Medium: 4.0 - 6.9: Вразливість може мати помірний вплив на систему, але для її експлуатації можуть знадобитися певні умови або додаткові права. Такі вразливості усуваються залежно від контексту та наслідків. Наприклад, атаки з підслуховуванням трафіку в мережі з наявним шифруванням доволі складно реалізувати, оскільки потрібно бути в тій же мережі, захоплювати багато пакетів, щоб мати успіх знайти в них конфіденційну інформацію (наприклад, облікові дані), та мати потужну обчислювальну систему для розшифрування трафіку. Такі атаки не дуже популярні серед зловмисників, тому вразливість зі слабким шифруванням

можна або прийняти через її складність та непопулярність, або усунути.

Для вразливостей середнього рівня потрібно визначити пріоритет усунення вразливості залежно від конкретних умов. Якщо вразливість може спричинити значні проблеми за певних умов, слід розглянути її усунення або застосувати тимчасові заходи для зменшення ризику, наприклад, підсилення шифрування або встановити оновлення від вендора, або усунути вразливість в коді чи конфігураціях.

High: 7.0 - 8.9: Вразливості цього рівня зазвичай вимагають негайного усунення. Ці вразливості повинні першочергово усуватися через високий вплив на конфіденційність, цілісність або доступність системи.

Вразливості високого рівня необхідно негайно усунути вразливість. Якщо це неможливо зробити швидко, слід застосувати тимчасові заходи для обмеження доступу або зменшення можливості експлуатації вразливості, наприклад, за допомогою брандмауера, додаткового контролю доступу або ізоляції вразливих компонентів системи.

Critical: 9.0 - 10.0: Вразливість має критичний вплив на систему, що може призвести до повної компрометації системи чи систем, у результаті чого це приведе до значних збитків. Прикладом може бути вразливість на контролері домену ZeroLogon, успішна експлуатація якої призведе до повної компрометації доменної мережі або її виходу з ладу, що впливає на всіх користувачів та системи у доменній мережі.

Вразливості критичного рівня необхідно негайно усунути вразливість. Якщо офіційне виправлення від вендора ще не доступне, необхідно прийняти заходи для ускладнення експлуатації вразливості. Це може включати налаштування додаткових правил брандмауера для обмеження доступу до вразливої служби, ізоляцію сегментів мережі, обмеження доступу до критичних компонентів системи, оновлення облікових даних та посилення політик безпеки.

У всіх випадках важливо проводити регулярні оцінки безпеки та аудити системи, щоб своєчасно виявляти нові вразливості та оперативно на них реагувати. Важливу роль відіграє тестування на захищеність, оскільки ця процедура

допомагає зрозуміти, на яких етапах вразливість може бути експлуатована, що саме є вразливим, і відповідно до отриманих результатів організація може розробити стратегію усунення вразливостей. Це дозволяє правильно розподілити ресурси на закриття вразливостей.

ВИСНОВКИ

У роботі проведено дослідження проблеми захищеності служби каталогів Active Directory (AD), визначено мету та завдання. На сьогодні проблеми захисту служби каталогів Active Directory залишаються актуальними, оскільки виявляються нові загрози, а також зберігаються вже відомі вразливості. Спостерігається зростання ризиків для безпеки AD, які посилюються недостатньою увагою організацій до безпечного адміністрування.

Проведено аналіз існуючих методів та засобів тестування захищеності служби каталогів Active Directory. Використання утиліт таких як Responder, Bloodhound та Mimikatz дозволяє виявляти вразливості, отримувати початковий доступ, аналізувати структуру організації, підвищувати доменні привілеї та здійснювати бічне переміщення. Ці утиліти відіграють визначальну роль у загальній системі виявлення можливих загроз безпеки інформаційної системи організації, дозволяючи оцінити поточний стан захищеності AD та виявити потенційні шляхи компрометації.

Розроблено рекомендації щодо етапів тестування на захищеність AD та загальні рекомендації, як оцінювати та усувати вразливості. Зокрема, було визначено, що ефективне тестування має включати кілька ключових етапів: розвідка, отримання доступу, перерахування домену, підвищення привілеїв, бічне переміщення та закріплення в домені. Було проаналізовано, які вразливості потрібно першочергово усувати, а які можна прийняти.

Отже, виявлення та усунення вразливостей у службі каталогів Active Directory повинно бути ключовим аспектом сучасної кібербезпеки, оскільки організації стикаються з великою кількістю кібератак. Комплексний підхід до захисту AD буде включати регулярне тестування на вразливості та впровадження рекомендацій щодо їх усунення. Це допоможе організаціям підвищити рівень безпеки своїх інформаційних ресурсів.

Таким чином, правильний підхід до тестування захищеності дозволить

правильно виявити та оцінити знайдені вразливості, що значно підвищить безвеку домену організації.

ПЕРЕЛІК ПОСИЛАНЬ

1. mrb3n. Active directory pentesting: cheatsheet and beginner guide. Hack The Box. URL: https://www.hackthebox.com/blog/active-directory-penetration-testing-cheatsheet-and-guide (date of access: 30.05.2024).

2. Active Directory Objects List - Windows Active Directory. Windows Active Directory. URL: https://www.windows-active-directory.com/active-directory-objects-list.html (date of access: 30.05.2024).

3. Buckbee M. What is an Active Directory Forest?. Varonis: Automated Data Security | DSPM | AI. URL: https://www.varonis.com/blog/active-directory-forest (date of access: 30.05.2024).

4. Create an organizational unit (OU) in Microsoft Entra Domain Services - Microsoft Entra ID. Microsoft Learn: Build skills that open doors in your career. URL: https://learn.microsoft.com/en-us/entra/identity/domain-services/create-ou (date of access: 30.05.2024).

5. What is LDAP Authentication? How Does It Work? | Fortinet. Fortinet. URL: https://www.fortinet.com/resources/cyberglossary/ldap-authentication (date of access: 30.05.2024).

6. Simple Authentication. Moved. URL: https://docs.oracle.com/javase/jndi/tutorial/ldap/security/simple.html (date of access: 30.05.2024).

7. Windows Event ID 4768 - A Kerberos authentication ticket was requested | ADAudit Plus. ManageEngine: IT security, operations & service management. URL: https://www.manageengine.com/products/active-directory-audit/kb/windows-security-log-event-id-4768.html (date of access: 30.05.2024).

8. Active Directory and LDAP Authentication Guide - DNSstuff. Software Reviews, Opinions, and Tips - DNSstuff. URL: https://www.dnsstuff.com/active-directory-ldap-authentication (date of access: 30.05.2024).

9. IONOS editorial team. NTLM (NT LAN Manager). IONOS Digital Guide. URL:

https://www.ionos.com/digitalguide/server/know-how/ntlm-nt-lan-manager/ (date of access: 30.05.2024).

10. Penetration Testing. IT Governance - Governance, Risk Management and Compliance for Information Technology. URL: https://www.itgovernance.co.uk/penetration-testing (date of access: 30.05.2024).

11. Types of Penetration Testing | Black Box vs White Box vs Grey Box. Redscan. URL: https://www.redscan.com/news/types-of-pen-testing-white-box-black-box-and-everything-in-between/ (date of access: 30.05.2024).

12. "Summarizing The Five Phases of Penetration Testing - Cybrary". Cybrary. 2015-05-06. Archived from the original on April 8, 2019. Retrieved 2018-06-25 (date of access: 30.05.2024).

13. Five Methodologies That Can Improve Your Penetration Testing ROI. Cybersecurity Exchange. URL: https://www.eccouncil.org/cybersecurity-exchange/penetration-testing/penetration-testing-methodology-improve-pen-testing-roi/ (date of access: 30.05.2024).

14. What Is Zerologon?. Trend Micro. URL: https://www.trendmicro.com/en_us/what-is/zerologon.html (date of access: 30.05.2024).

15. Zerologon (CVE-2020-1472): Overview, Exploit Steps and Prevention. crowdstrike.com. URL: https://www.crowdstrike.com/blog/cve-2020-1472-zerologon-security-advisory/ (date of access: 30.05.2024).

16. WDigest Clear-Text Passwords: Stealing More than a Hash. Netwrix Blog | Insights for Cybersecurity and IT Pros. URL: https://blog.netwrix.com/2022/10/11/wdigest-clear-text-passwords-stealing-more-than-a-hash/ (date of access: 30.05.2024).

17. LLMNR Poisoning and How to Prevent It - TCM Security. TCM Security - Penetration Testing & Consulting. URL: https://tcm-sec.com/llmnr-poisoning-and-how-to-prevent-it/ (date of access: 30.05.2024).

18. Unleashing the Power of Responder: An In-Depth Guide to a Versatile Ethical Hacking Tool. URL: https://medium.com/@S3Curiosity/unleashing-the-power-of-responder-an-in-depth-guide-to-a-versatile-ethical-hacking-tool-ffc398cf6c58/ (date of

access: 30.05.2024).

19. GitHub - SpiderLabs/Responder: Responder is a LLMNR, NBT-NS and MDNS poisoner, with built-in HTTP/SMB/MSSQL/FTP/LDAP rogue authentication server supporting NTLMv1/NTLMv2/LMv2, Extended Security NTLMSSP and Basic HTTP authentication. GitHub. URL: https://github.com/SpiderLabs/Responder/tree/master (date of access: 30.05.2024).

20. What is LLMNR Poisoning Attack? URL: https://systemweakness.com/what-is-llmnr-poisoning-attack-and-how-to-secure-against-it-417f3b415e51 (date of access: 30.05.2024).

21. DCSync Explanation. URL: https://medium.com/@sopheakchheng089/dcsync-explanation-1f41428ff8c2 (date of access: 30.05.2024).

22. Offense and Defense – A Tale of Two Sides: (Windows) OS Credential Dumping | FortiGuard Labs. Fortinet Blog. URL: https://www.fortinet.com/blog/threat-research/offense-and-defense-a-tale-of-two-sides-windows-os-credential-dumping (date of access: 30.05.2024).

23. Lateral movement security alerts - Microsoft Defender for Identity. Microsoft Learn: Build skills that open doors in your career. URL: https://learn.microsoft.com/en-us/defender-for-identity/lateral-movement-alerts (date of access: 30.05.2024).

24. Overpass-the-Hash Attack: Principles and Detection. Netwrix Blog | Insights for Cybersecurity and IT Pros. URL: https://blog.netwrix.com/2022/10/04/overpass-the-hash-attacks/ (date of access: 30.05.2024).

25. Lateral movement security alerts - Microsoft Defender for Identity. Microsoft Learn: Build skills that open doors in your career. URL: https://learn.microsoft.com/en-us/defender-for-identity/lateral-movement-alerts#suspected-identity-theft-pass-the-ticket-external-id-2018 (date of access: 30.05.2024).

26. Kerberosting and Golden Ticket exploit for Red Teamers!. Payatu. URL: https://payatu.com/blog/kerberosting-golden-ticket-red-team/#Golden_Ticket (date of access: 30.05.2024).

27. Kerberos Golden Ticket Attack Explained [2024 Complete Guide]. StationX. URL: https://www.stationx.net/golden-ticket-attack/ (date of access: 30.05.2024).

28. Silver Ticket Attack. Netwrix. URL: https://www.netwrix.com/silver_ticket_attack_forged_service_tickets.html (date of access: 30.05.2024).

29. CVSS v4.0 Specification Document. FIRST – Forum of Incident Response and Security Teams. URL: https://www.first.org/cvss/v4.0/specification-document (date of access: 30.05.2024).

30. What are CVSS Scores | Balbix. Balbix. URL: https://www.balbix.com/insights/understanding-cvss-scores/ (date of access: 30.05.2024).

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

(Презентація)