Active Directory para pentesters en 5 minutos

Active Directory para pentesters en 5 minutos

Por Miguel Ángel Villalobos
https://www.linkedin.com/in/m7villalobos/

1. Entendiendo Active Directory: ¿Qué es y Por Qué Importa?

Imagina una empresa grande. Necesitan una forma organizada y centralizada de gestionar:

- **Quién trabaja allí:** Cuentas de usuario (ej. `juan.perez`) con sus identidades y credenciales.
- **Qué ordenadores y servidores hay:** Identidades de equipo (ej. `PC-Contabilidad`, `ServidorWeb`), cada uno con su propia cuenta en AD.
- **Quién tiene permiso para hacer qué:** Control de acceso (Autorización - AuthZ) a archivos, impresoras, aplicaciones, recursos compartidos, etc.
- **Qué reglas de configuración y seguridad se aplican:** Políticas de grupo (GPOs) que definen desde la complejidad de contraseñas hasta qué software se puede ejecutar o qué configuración tiene el escritorio.

Active Directory (AD) es la solución de Microsoft para administrar todo esto. Es fundamentalmente una **base de datos jerárquica y distribuida** (almacenada en los Controladores de Dominio) junto con un conjunto de **servicios** que actúan como el sistema nervioso central para redes basadas en Windows.

Su importancia radica en:

- **Centralización:** Simplifica enormemente la administración de recursos y usuarios.
- **Seguridad:** Proporciona mecanismos robustos de **Autenticación** (AuthN - verificar quién eres, principalmente con Kerberos) y **Autorización** (AuthZ - definir qué puedes hacer, basado en permisos y membresías a grupos).
- **Escalabilidad:** Diseñado para funcionar desde pequeñas redes hasta organizaciones globales con millones de objetos.
- **Integración:** Innumerables aplicaciones y servicios empresariales (Exchange, SharePoint, SQL Server, muchas aplicaciones de terceros) dependen de AD para la gestión de identidades y accesos.

¿Por qué es un objetivo clave en seguridad ofensiva (Red Team / Pentesting)?

Porque AD contiene, literalmente, las "llaves del reino digital". Comprometer Active Directory a nivel de administrador (Domain Admin en un dominio o, el santo grial, Enterprise Admin en

el bosque) significa tener control prácticamente absoluto sobre la infraestructura tecnológica de la organización. Permite:

- Acceder a datos confidenciales en servidores de archivos.
- Desplegar software (incluyendo malware o ransomware) en toda la red.
- Crear cuentas de usuario (incluidas cuentas ocultas o con altos privilegios).
- Resetear contraseñas de cualquier usuario (incluidos otros administradores).
- Modificar políticas de seguridad.
- Moverse lateralmente sin restricciones por la red.

Por eso, obtener privilegios elevados en AD es casi siempre el **objetivo principal** de las operaciones de Red Team y las pruebas de penetración internas.

2. Los Componentes Esenciales de AD

Entender estas piezas es fundamental para identificar vectores de ataque y puntos débiles.

- **Bosque (Forest):** La estructura contenedora de más alto nivel en AD. Es el límite de seguridad y administrativo completo. Contiene uno o más Árboles de Dominios. La cuenta `Enterprise Admins` (que solo existe en el **dominio raíz del bosque**) tiene autoridad sobre todos los dominios del bosque.
- **Árbol (Tree):** Un conjunto de uno o más dominios que comparten un espacio de nombres DNS contiguo y jerárquico (ej. `empresa.local` como raíz del árbol y `ventas.empresa.local`, `dev.empresa.local` como dominios hijos).
- **Dominio (Domain):** La unidad principal de partición administrativa y de replicación. Cada dominio tiene su propia base de datos de AD (aunque se replica entre sus DCs), políticas específicas, usuarios, grupos y equipos. Ser `Domain Admin` te da control total sobre *ese* dominio específico.
- **Controladores de Dominio (DCs - Domain Controllers):** Los servidores **más críticos** de la infraestructura. Son los que:
  - Almacenan una copia completa (lectura/escritura, excepto en RODCs) de la base de datos del dominio (`NTDS.dit`). Este archivo contiene información sobre todos los objetos del dominio, incluyendo los **hashes de las contraseñas** de los usuarios y equipos.
  - Procesan las solicitudes de autenticación (Kerberos, NTLM).
  - Replican los cambios de AD entre sí.
  - Aplican las Políticas de Grupo (GPOs).
  - Son el objetivo número uno dentro de un dominio una vez se tienen ciertos privilegios.
- **Unidades Organizativas (OUs - Organizational Units):** Contenedores *dentro de un dominio* utilizados para organizar objetos (usuarios, grupos, equipos) de forma lógica (ej., por departamento, ubicación geográfica). Su propósito principal es:

    - **Delegar control administrativo:** Se pueden asignar permisos específicos sobre una OU a un grupo o usuario, permitiéndoles gestionar solo los objetos dentro de esa OU (ej., un helpdesk que solo puede resetear contraseñas para usuarios de la OU "Ventas"). *Las delegaciones mal configuradas son una fuente común de escalada de privilegios.*
    - **Aplicar Políticas de Grupo (GPOs):** Las GPOs se vinculan a OUs (también a dominios o sitios) para aplicar configuraciones específicas a los objetos contenidos en ellas.
- **Objetos:** Las entidades gestionadas por AD:
    - **Usuarios (Users):** Representan a personas o cuentas de servicio. Atributos clave: `samAccountName` (nombre de usuario), `SID` (Security Identifier - único e inmutable), `userAccountControl` (flags de estado), `memberOf` (grupos a los que pertenece), y el hash de su contraseña (no visible directamente).
    - **Grupos (Groups):** Colecciones de usuarios, equipos u otros grupos. Se usan para simplificar la asignación de permisos. Existen grupos de seguridad (con SID, usados para permisos) y grupos de distribución (para email). Son cruciales los **grupos privilegiados** como `Domain Admins`, `Enterprise Admins` (solo en el dominio raíz), `Administrators` (grupo local en cada máquina, pero también un grupo Builtin en AD), `Schema Admins`, `Backup Operators`, `Account Operators`, `Server Operators`, `DNSAdmins`, etc. *El objetivo final suele ser conseguir la membresía en uno de estos grupos.*
    - **Equipos (Computers):** Representan estaciones de trabajo y servidores unidos al dominio. Tienen su propia identidad, contraseña (gestionada automáticamente por defecto) y SID en AD.
- **Políticas de Grupo (GPOs - Group Policy Objects):** Conjuntos de configuraciones y reglas que definen aspectos del entorno de usuario y del sistema operativo (políticas de contraseña, mapeo de unidades, instalación de software, restricciones de seguridad, configuraciones de firewall, etc.). Se aplican a contenedores (Sitios, Dominios, OUs). Son extremadamente potentes. *Si un atacante con los permisos adecuados puede modificar una GPO vinculada a muchos sistemas (o a los DCs), puede comprometerlos masivamente.*
- **LDAP (Lightweight Directory Access Protocol):** El protocolo estándar utilizado para consultar y (si se tienen permisos) modificar la información almacenada en Active Directory. Funciona sobre TCP/IP, típicamente en el puerto **389/TCP** (generalmente sin cifrar, ¡cuidado!) o **636/TCP (LDAPS)** para conexiones cifradas con SSL/TLS. *Esencial para la fase de reconocimiento y enumeración.*
- **Kerberos:** El protocolo de autenticación **predeterminado y preferido** en entornos AD modernos. Se basa en un sistema de "tickets" emitidos por el Key Distribution Center (KDC, un servicio que corre en los DCs) para verificar la identidad de usuarios y servicios sin enviar contraseñas por la red. Aunque seguro conceptualmente, existen ataques específicos contra su implementación y uso (ver más abajo).

- **NTLM (NT LAN Manager):** Un protocolo de autenticación más antiguo, basado en un mecanismo de desafío-respuesta. Es considerado **menos seguro** que Kerberos. Todavía se utiliza por compatibilidad con sistemas antiguos o como mecanismo de fallback si Kerberos falla. Es vulnerable a ataques de retransmisión (NTLM Relay) y Pass-the-Hash.
- **Confianzas (Trusts):** Relaciones establecidas entre dominios (dentro del mismo bosque o entre bosques diferentes) que permiten a los usuarios y grupos de un dominio acceder a recursos en el otro dominio. Las confianzas pueden ser transitivas o no transitivas, unidireccionales o bidireccionales. *Pueden ser una vía para que un atacante "salte" de un dominio comprometido a otro si la confianza está mal configurada o si se obtienen credenciales válidas en el dominio confiado.*

3. Fases Típicas de un Ataque a AD

Un ataque dirigido a comprometer AD suele seguir estos pasos generales, aunque pueden variar:

1. **Reconocimiento / Enumeración (Reconnaissance & Enumeration):**
   - **Objetivo:** Mapear la estructura del dominio/bosque. Identificar usuarios (especialmente administradores y cuentas de servicio), grupos privilegiados, Controladores de Dominio, servidores críticos, relaciones de confianza, políticas de contraseña, posibles configuraciones erróneas y vulnerabilidades conocidas.
   - **Métodos:**
     - Consultas LDAP (con herramientas como `PowerView` (PowerShell), `SharpHound` (C#, el recolector de BloodHound), `AdExplorer` (Sysinternals), `ldapsearch` (Linux), o scripts propios).
     - Análisis de registros DNS para encontrar DCs y otros servidores.
     - Escaneo de red (`nmap`) para identificar puertos abiertos (LDAP, Kerberos, SMB, WinRM, RDP, etc.) en DCs y otros objetivos.
     - Uso intensivo de **`BloodHound`**: Esta herramienta es fundamental. Utiliza la información recopilada (generalmente vía `SharpHound`) para visualizar las relaciones de permisos (ACLs), pertenencias a grupos, sesiones activas y políticas de GPO en un grafo. Permite identificar rápidamente rutas de ataque para escalada de privilegios y movimiento lateral que serían muy difíciles de encontrar manualmente.
2. **Acceso Inicial (Initial Access):**
   - **Objetivo:** Obtener un primer punto de apoyo en la red, generalmente comprometiendo una estación de trabajo o un servidor con credenciales de un usuario con bajos privilegios.
   - **Métodos:** Phishing (obtener credenciales o ejecutar malware), explotación de vulnerabilidades en servicios expuestos a internet o en software de usuario, ataques de **Password Spraying** (probar contraseñas comunes o estacionales

como `Verano2024!` contra una lista grande de usuarios), adivinar contraseñas débiles.

3. **Acceso a Credenciales (Credential Access):**
   - **Objetivo:** Una vez dentro de una máquina, extraer credenciales (contraseñas en claro, hashes NTLM, tickets Kerberos) que puedan estar almacenados en memoria, registro o archivos. Estas credenciales pueden permitir el acceso a otras máquinas o la escalada de privilegios.
   - **Métodos:**
     - Uso de herramientas como **`Mimikatz`**: Capaz de extraer secretos del proceso LSASS (Local Security Authority Subsystem Service). Requiere privilegios de Administrador local o `SYSTEM` en la máquina objetivo. Puede obtener contraseñas en claro, hashes NTLM, y tickets Kerberos.
     - **Kerberoasting:** Técnica para solicitar tickets de servicio (TGS) para cuentas de usuario configuradas como Cuentas de Servicio (con un Service Principal Name - SPN). La parte cifrada del ticket (que contiene información firmada con el hash de la contraseña de la cuenta de servicio) puede ser extraída y crackeada offline. *No requiere privilegios elevados para solicitar el ticket inicialmente*, solo ser un usuario autenticado en el dominio.
     - **AS-REP Roasting:** Similar a Kerberoasting, pero dirigida a cuentas de usuario que tienen la opción "Do not require Kerberos preauthentication" habilitada. Permite solicitar directamente parte del TGT (AS-REP) cifrado con el hash del usuario y crackearlo offline. *Tampoco requiere privilegios elevados inicialmente.*
     - Búsqueda manual o automatizada de contraseñas en texto plano o archivos de configuración (`web.config`, scripts de PowerShell, archivos de historial, notas, etc.).
     - Dumping de la base de datos SAM local (para hashes locales, si no se está en dominio o como paso intermedio) o del archivo `NTDS.dit` del DC (si ya se tiene acceso privilegiado al DC).
4. **Escalada de Privilegios (Privilege Escalation):**
   - **Objetivo:** Aumentar el nivel de permisos, ya sea localmente en la máquina comprometida (de usuario estándar a Administrador/`SYSTEM`) o, más importante, a nivel de dominio (de usuario estándar a miembro de un grupo privilegiado como `Domain Admins`).
   - **Métodos:**
     - Utilizar las credenciales (hashes, tickets, contraseñas) robadas de cuentas con mayores privilegios.
     - **Abuso de Permisos/ACLs (Access Control Lists):** Buscar objetos en AD (usuarios, grupos, equipos, OUs, GPOs) donde la cuenta comprometida (o un grupo al que pertenece) tenga permisos de escritura peligrosos (ej. `WriteMembers` sobre un grupo, `WriteProperty` sobre ciertos atributos de usuario, `GenericAll`/`GenericWrite` sobre un objeto, permiso para modificar

una GPO). Herramientas como **`BloodHound`** y **`PowerView`** son esenciales para encontrar estas rutas de abuso.
    - **Abuso de GPOs:** Si se obtiene control sobre una GPO (o la OU/dominio donde se vincula), se pueden modificar políticas para ejecutar scripts, instalar software, añadir usuarios a grupos locales, etc., en todas las máquinas afectadas por esa GPO.
    - Explotación de vulnerabilidades del sistema operativo local o de servicios específicos de AD (ej. Zerologon, PrintNightmare, si no están parcheados).
    - Abuso de configuraciones inseguras de servicios (ej. servicios con `Unquoted Service Paths` o permisos débiles).
5. **Movimiento Lateral (Lateral Movement):**
    - **Objetivo:** Utilizar las credenciales o privilegios obtenidos para autenticarse y ejecutar código en otras máquinas de la red, expandiendo el control.
    - **Métodos:**
        - **Pass-the-Hash (PtH):** Usar un hash NTLM robado (ej. con `Mimikatz`) para autenticarse en otra máquina que acepte autenticación NTLM (común con SMB, WMI). Herramientas del framework **`Impacket`** (Python) como `psexec.py`, `wmiexec.py`, `smbexec.py` son muy usadas para esto.
        - **Pass-the-Ticket (PtT):** Usar un ticket Kerberos robado (TGT o TGS, ej. con `Mimikatz` o extraído de memoria) para autenticarse en servicios que usen Kerberos. Herramientas como **`Mimikatz`** ("kerberos::ptt") o **`Rubeus`** (C#) facilitan esto.
        - **Overpass-the-Hash (OtH):** Usar un hash NTLM para *solicitar* un ticket Kerberos (TGT) y luego usar ese ticket para autenticación Kerberos (PtT).
        - Uso de credenciales en claro robadas con herramientas estándar de administración remota como `PsExec` (Sysinternals), WMI (Windows Management Instrumentation), WinRM (Windows Remote Management), RDP (Remote Desktop Protocol).
6. **Persistencia y Dominación (Persistence & Domain Dominance):**
    - **Objetivo:** Establecer mecanismos para mantener el acceso a largo plazo, incluso si se cambian contraseñas o se detecta el acceso inicial, y finalmente alcanzar los objetivos finales (exfiltración de datos, despliegue de ransomware, sabotaje, etc.).
    - **Métodos (Persistencia Avanzada en AD):**
        - **Golden Ticket:** Un ataque devastador. Si se obtiene el hash NTLM de la cuenta **`krbtgt`** (una cuenta especial del dominio cuyo hash se usa para firmar todos los tickets Kerberos TGT), un atacante puede *forjar tickets TGT* para cualquier usuario (existente o no), con cualquier nivel de privilegio (ej. `Enterprise Admins`), y con largos periodos de validez. Requiere acceso a nivel de `Domain Admin` para obtener el hash `krbtgt`. Es muy sigiloso a nivel de autenticación.
        - **Silver Ticket:** Similar al Golden Ticket, pero se forja un ticket de servicio (TGS) para un *servicio específico* (ej. CIFS para acceso a archivos, HOST para

ejecución remota) utilizando el hash de la contraseña de la cuenta de servicio de ese servicio. Permite acceder a ese servicio específico como cualquier usuario. Requiere el hash de la cuenta de servicio.
    - **DCSync:** Abusar de los permisos de replicación de directorio (normalmente asignados a los DCs y a veces a otras cuentas) para solicitar directamente a un DC que replique la información de credenciales (hashes) de cualquier usuario, incluyendo la cuenta `krbtgt`. Herramientas como `Mimikatz` ("lsadump::dcsync") lo implementan.
    - Modificar ACLs de objetos críticos: Añadir control total para una cuenta controlada por el atacante sobre el objeto `AdminSDHolder` (cuyos permisos se propagan a todas las cuentas y grupos protegidos), sobre el objeto del Dominio, sobre GPOs importantes, o sobre OUs clave.
    - Crear cuentas ocultas, añadir cuentas a grupos privilegiados, modificar GPOs para ejecutar código periódicamente, crear tareas programadas en DCs, instalar puertas traseras (backdoors) o rootkits.
- **Acciones sobre el Objetivo (Actions on Objectives):** Una vez se tiene el control y la persistencia, realizar las acciones finales: exfiltrar datos sensibles, cifrar sistemas con ransomware, destruir información, etc.

4. Ataques Comunes Explicados (Resumen Rápido)

- **Password Spraying:** Probar 1-3 contraseñas (`Verano2024`, `Welcome1!`, `Password123`) contra CIENTOS o MILES de cuentas. Evita bloqueos por intentos fallidos en una sola cuenta. Muy efectivo contra políticas de contraseña débiles.
- **Kerberoasting:** Enumerar cuentas de servicio (SPNs). Solicitar un ticket de servicio (TGS) para ellas (cualquier usuario puede hacerlo). Extraer la parte cifrada con el hash de la cuenta de servicio. Crackear ese hash offline. Si la contraseña es débil, se obtiene acceso a la cuenta de servicio.
- **AS-REP Roasting:** Encontrar cuentas de usuario configuradas sin pre-autenticación Kerberos. Solicitar un AS-REP (parte del TGT inicial) cifrado con el hash del usuario. Crackear offline. Si la contraseña es débil, se obtiene acceso a la cuenta de usuario.
- **Mimikatz / LSASS Dumping:** Si eres Administrador Local / `SYSTEM` en una máquina, ejecutar `Mimikatz` (u herramientas similares) para volcar la memoria del proceso LSASS y extraer credenciales (texto plano, hashes NTLM, tickets Kerberos) de usuarios que hayan iniciado sesión en esa máquina.
- **Pass-the-Hash (PtH):** En lugar de usar una contraseña, usar directamente el hash NTLM robado para autenticarse en otro sistema a través del protocolo NTLM (ej. con `wmiexec.py –hashes :<ntlm_hash> target_ip`).
- **Pass-the-Ticket (PtT):** Inyectar un ticket Kerberos robado (TGT o TGS) en la sesión actual y usarlo para acceder a recursos/servicios que usen autenticación Kerberos.

- **Abuso de Permisos (ACLs / GPOs):** El "arte" de encontrar configuraciones erróneas o delegaciones excesivas. ¿Tiene un usuario normal permiso para añadir miembros al grupo `Domain Admins`? ¿Puede un grupo de bajo privilegio modificar una GPO crítica? **`BloodHound`** es la herramienta clave para visualizar y encontrar estas relaciones tóxicas.
- **Golden Ticket:** La "llave maestra" de Kerberos. Requiere el hash NTLM de la cuenta `krbtgt` (el secreto mejor guardado del dominio). Una vez obtenido (generalmente necesitando ser DA), permite crear TGTs falsos para cualquier usuario, otorgando acceso ilimitado y persistencia sigilosa.

5. ¿Qué Puede Frenar a un Atacante? (Defensas Clave)

Es crucial para un pentester conocer las defensas, tanto para saber cómo intentar sortearlas como para poder hacer recomendaciones útiles.

- **Contraseñas Fuertes y Únicas + Autenticación Multifactor (MFA):** La base absoluta. Políticas de contraseñas robustas y, sobre todo, MFA en todas partes (VPN, OWA, accesos privilegiados) dificultan enormemente el uso de credenciales robadas y ataques de fuerza bruta/spraying.
- **Parcheo Constante y Gestión de Vulnerabilidades:** Aplicar parches de seguridad rápidamente, especialmente los que afectan a AD y a los DCs (ej. Zerologon, PrintNightmare) cierra las puertas a la explotación directa.
- **Principio de Mínimo Privilegio (Least Privilege):** Dar a cada cuenta (usuario o servicio) _exactamente_los permisos que necesita para realizar su función, y nada más. Revisar periódicamente las membresías a grupos privilegiados y las delegaciones de permisos.
- **LAPS (Local Administrator Password Solution):** Herramienta gratuita de Microsoft que gestiona contraseñas aleatorias y únicas para la cuenta de Administrador local de cada estación de trabajo y servidor unido al dominio, rotándolas periódicamente. Complica enormemente el movimiento lateral basado en el reuso de la misma contraseña de administrador local.
- **Modelo de Tiers / Administración Segura (PAWs - Privileged Access Workstations):** Segregar estrictamente las cuentas y las estaciones de trabajo utilizadas para administrar sistemas críticos (Tier 0: DCs, AD; Tier 1: Servidores; Tier 2: Estaciones de trabajo) del entorno de usuario normal. Los administradores de AD solo deberían usar cuentas y máquinas dedicadas y altamente securizadas. Muy efectivo si se implementa correctamente.
- **Credential Guard & Remote Credential Guard:** Características de Windows (basadas en Virtualization-Based Security - VBS) que aíslan el proceso LSASS y protegen las credenciales almacenadas (hashes NTLM, TGTs Kerberos) contra el robo, incluso por parte de código con privilegios de `SYSTEM`. Dificulta mucho `Mimikatz` y ataques similares.

- **Monitorización y Detección Avanzada (EDR, SIEM, Microsoft Defender for Identity - MDI):** Usar herramientas de Endpoint Detection and Response (EDR) en los endpoints, un Security Information and Event Management (SIEM) para correlacionar logs, y soluciones específicas de monitorización de AD como **Microsoft Defender for Identity (MDI)** (anteriormente Azure ATP) que analizan el tráfico de autenticación y el comportamiento en AD para detectar patrones de ataque conocidos (Pass-the-Hash, Golden Ticket, Kerberoasting sospechoso, etc.) y alertar en tiempo real.
- **Segmentación de Red y Firewalls:** Limitar la comunicación entre diferentes segmentos de la red (ej. que las estaciones de trabajo no puedan comunicarse directamente entre sí por SMB, que solo ciertas máquinas puedan contactar a los DCs por los puertos necesarios).
- **Hardening de Sistemas y de AD:** Aplicar configuraciones de seguridad recomendadas: deshabilitar protocolos obsoletos (SMBv1, LM/NTLMv1), requerir firmas SMB y LDAP, configurar Kerberos de forma segura, habilitar logging avanzado, etc.

Active Directory es el corazón de la infraestructura de TI en la gran mayoría de las organizaciones que usan Windows. Su complejidad inherente y la multitud de formas en que puede ser configurado (y mal configurado) lo convierten en una superficie de ataque extensa y atractiva. Para cualquier profesional de la seguridad ofensiva (Red Team, pentesting), entender la arquitectura de AD, sus protocolos de autenticación, sus objetos clave y, sobre todo, las técnicas comunes para enumerarlo, explotar sus debilidades, escalar privilegios y moverse lateralmente, es absolutamente fundamental. Dominar estas técnicas no solo permite evaluar eficazmente la seguridad de una organización, sino también ayudarla a fortalecer sus defensas contra amenazas reales.

El siguiente paso lógico sería profundizar en las herramientas y comandos específicos para cada fase: reconocimiento (`PowerView` , `BloodHound` / `SharpHound`), obtención de credenciales (`Mimikatz` , `Rubeus`), movimiento lateral (`Impacket` , `PsExec`), etc. Lo veremos más adelante.
