uc3m | Universidad Carlos III de Madrid

Máster Universitario en Ciberseguridad

2018-2019

Trabajo Fin de Máster

“Ciberejercicios para evaluar Active Directory en sus distinas versiones. ”

Borja Lorenzo Fernádez

Tutor

Andrés Marín López

Agradecimientos

Abstract

Microsoft Active Directory se ha convertido en una de las partes fundamentales de las empresas en la actualidad y se le considera como el *core* de la organización. Active Directory permite a los administradores gestionar de manera eficiente la información de la empresa y los límites de la misma. Esta información puede englobar usuarios, clientes, proveedores, dispositivos como ordenadores, servidores o impresoras, servicios y aplicaciones y la forma de interacción entre ellos así como los diferentes permisos que tiene cada usuario o grupo de usuarios. Esta información, como se puede deducir, supone un importante objetivo para atacantes y ciberdelincuentes y un gran problema de seguridad para las empresas si estos sistemas de gestión se ven comprometidos. Esto a llevado a cabo un gran aumento de ciberataques contra Active Directory con el objetivo principal de recolectar dicha información y comprometer la seguridad total de una empresa. Además, en los últimos años se han perfeccionado las técnica principales de ataque así como la implementación de novedosas herramientas que facilitan todo tipo de ataques.

Con el fin de contribuir al desarrollo y mejora de la seguridad de Active Directory y por ende, de las empresas y particulares que hacen uso de ello, el trabajo realizado se centra en el estudio de los principales componentes que engloban la seguridad y gestión del sistema de directorio Active Directory a través de la creación de un laboratorio, de manera local, y el análisis y revisión de los principales ataques y amenazas usadas en la actualidad para vulnerar dicho sistema de gestión. Además, se ha implementado una topología que simula una empresa ficticia e implementa las últimas versiones proporcionadas por Microsoft que gestionan cada dominio.

Palabras Clave:

Microsoft Active Directory, Domain Controller, Kerberos, Ciberseguridad, Windows Server, Pentesting, Red Team

Índice general

Índice de figuras

Índice de tablas

1. Introducción

En los últimos años, empresas y organizaciones se han visto en la necesidad de gestionar de una manera eficiente y centralizada la información y recursos en red que disponen, activo fundamental para el correcto funcionamiento del negocio. El aumento masivo de dicha información además de la necesidad de crear, distribuir y manipular tal cantidad de datos, ya sea a través de servicios de bases de datos como puede ser servicios MySQL, la obtención de servidores y dispositivos para su almacenamiento, la creación de aplicaciones web y servicios que permitan su distribución o la gestión de los usuarios que puedan manipularla o consultarla supone una gran cantidad de agentes que intervienen en el funcionamiento que es necesario controlar y regular. Microsoft Active Directory supone una solución a esa problemática a través de un servicio de directorio como base de datos distribuida que permite la gestión, administración y localización de todos los recursos en red [1].

Dentro del panorama actual, los servicios de Microsoft Active Directory se ha convertido en uno de los pilares que sostienen la organización de los recursos en red de la mayoría de las empresas vigentes así como uno de los principales objetivos para atacantes debido a dicha importancia. Esto se puede comprobar en el principal interés que tienen los principales grupos de ciberdelincuentes como APT28, Cobalt Strike... por esta infraestructura o los ataques de ransomware WannaCry, NotPetya, MBR-ONI, etc, que ponen a Active Directory en el punto de mira y centro de sus ataques [2]. En los últimos años, se ha visto un aumento considerable de vulnerabilidades críticas que afectan a la seguridad y que detectan y hacen considerablemente más sencillo su explotación.

Por todo ello y con el fin de abordar esta problemática, el trabajo realizado se ha centrado en la revisión, análisis y prueba en profundidad de las principales amenazas que suponen un problema de seguridad para Active Directory en sus diferentes versiones, técnicas como Pass-The-Hass, NTLM Relay, Kerberoast, etc que serán detalladas en los capítulos posteriores. Además, también se proporciona las directrices para la creación de un laboratorio local que permite la prueba de los ataques detallados así como la posibilidad de probar nuevas técnicas y ataques sin poner en riesgo la seguriidad de ningún entorno real u organización.

1.1. Estado del Arte

Para el desarrollo del trabajo, se ha considerado las siguientes versiones proporcionadas por Microsoft para la instalación de los dominios que van a formar partes del Active

Directory y van a ejecutar los Domain Controllers:

- **Windows Server 2019**
- **Windows Server 2016**
- **Windows Server 2012 R2**

Por lo tanto, se ha dejado la versión más antigua Windows Server 2008 como objeto de estudio o posible implantación en trabajo futuro al ser la versión más obsoleta. Aunque esto no implica que no haya empresas que aún siguen usando dicha versión.

En cuanto a los ataques a analizar de manera detallada se han considerado los siguientes:

- **Pass-The-Hash**
- **NTLM Relay**
- **Overpass-The-Hash**
- **Pass-The-Ticket**
- **Golden/Silver Ticket**
- **Kerberoast**

Como se puede observar, la mayoría de los ataques no son específicos de Active Directory si no que atacan a los protocolos NTLM y Kerberos, por lo que, también se van a detallar en profundidad estos protocolos en los capitulos siguientes.

1.2. Objetivos

El objetivo principal de este trabajo es la creación de un laboratorio que provea la infraestructura necesaria para la replicación de las principales técnicas de ataque a Active Directory así como la revisión de las mismas sobre las diferentes versiones proporcionadas por Microsoft. La creación del laboratorio posibilita tanto a *Pentesters* o expertos en seguridad ofensiva realizar ejercicios de *Read Team* en entornos controlados o réplicas de un entorno real como a administradores de sistemas para probar nuevas configuraciones y reglas para equipos de *Blue Team*.

Además, este trabajo tiene como objetivo la adquisición de conocimiento sobre Active Directory como punto de partida para futuras investigaciones. Como se ha visto en la

introducción Active Directory es una parte fundamental de una empresa y es uno de los principales objetivos de atacantes, conocer los principales ataques y cómo está organizado es de gran importancia hoy en día, permitiendo aasí una correcta implementación que minimice los riesgos a los que está sometido.

1.3. Organización del Proyecto

El presente documento se divide en 7 capítulos, en los cuales en primera instancia se detallan los aspectos a tener en cuenta relacionados con Active Directoy, se detalla el laboratorio implementado, las pruebas que se van a realizar, la experimentación realizada así como los resultados obtenidos durante el trascurso:

En el Capítulo 2 se detalla los aspectos relacionados con Active Directory, en primer lugar se define la autenticación y autorización en sistemas Windows tanto localmente como en dominio, se analizan los protocolos NT Lan Manager y Kerberos y la terminología relacionada con Active Directory.

En el Capitulo 3 se desarrolla la topología que se ha elegido para la creación del laboratorio de pruebas con las diferentes versiones de Windows y su implementación.

En el Capítulo 4 se detallan los ataques elegidos para realizar la experimentación.

En el Capítulo 5, una vez detalladas tanto el laboratorio como las ataaques principales objeto de estudio, se muestran las diferentes pruebas realizas en las versiones de Windows especificadas en el estado del arte.

En el Capítulo 6 se presentan y se discuten los resultados obtenidos.

En el Capítulo 7, para finalizar el proyecto, se lleva a cabo una reflexión sobre el esfuerzo realizado y sus diferentes líneas de trabajo futuro.

2. Aspectos Clave

Este capítulo aborda los términos y definiciones a tener en cuenta y sirve como introducción a Active Directory. En primer lugar, se define la forma en la que los Sistemas Windows gestioana la autenticación y la auterización. Posteriormente, se definen los protocolos de seguridad para la verificación de la autenticación NT Lan Manager y Kerberos. Finalmente, se presenta Active Directory y la terminología necesaria relativa a este.

2.1. Autenticación y Autorización

Uno de los principales requisitos a la hora de entender como funcionan la mayoría de los ataques contra Sistemas Windows pasa por la gstión de la autenticación y autorización de los usuarios que inician sesión en el ordenador ya sea a nivel local o en red.

Por un lado, la **autenticación** consiste en la verificación de la identidad de un usuario, dicho con otras palabras, que el sistema de autenticación se asegure de que un usuario es quién dice ser. Por ejemplo, conociendo la contraseña del usuario que dice ser.

Por otro lado, la **autorización** consiste en establacer y delimitar los recursos a los que puede acceder, o no puede acceder ya que los tiene restringidos un usuario (o grupos de usuarios).

2.1.1. Interactive Logon

El proceso de autenticación a través de inicio de sesión interactivo del inglés *Interactive Logon*, a diferencia del inicio de sesión en red o *Network Logon*, es llevado a cabo por el proceso *WinLogon* que se encarga de recoger las credenciales introducidas por el usuario y su posterior validación. Un usuario que inicia sesión en un equipo ya sea localmente o un inicio de sesión en red introduce el usuario y la contraseña (denominado credenciales de usuario) y sirve para verificar la identidad del usuario. Por otro lado, cuando se inicia sesión a través de una Smart Card *(Smart Card Logon)* las credenciales están almacenadas en el chip de la tarjeta y estas son leídas por un dispositivo externo y el usuario introduce el *Personal Identification Number (PIN)* [3] [4].

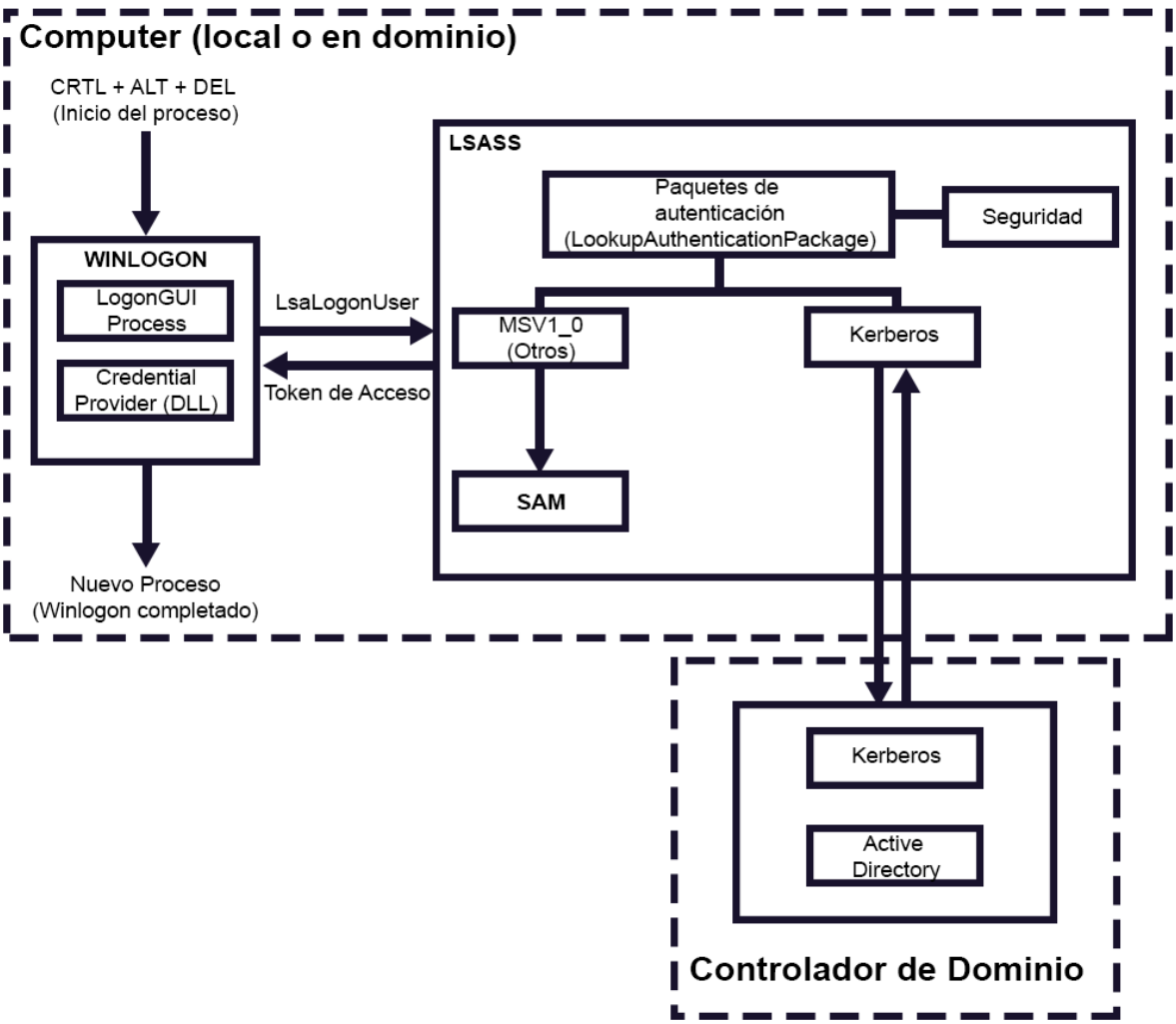

Fig. 2.1. Proceso de inicio de sesión interactivo (WinLogon).

Proceso WinLogon

WinLogon es el proceso encargado de coordinar el inicio de sesión. Además, este proceso también se encarga de gestionar el *logout*, lanzar los procesos necesarios para la autenticación de un usuario, cambiar las contraseñas, bloquear y desbloquear un equipo y proporcionar la seguridad necesaria para que ningún otro proceso pueda acceder a información sensible cuando estos procedimientos se están llevando a cabo.

Como se puede ver en la Figura 2.1 el proceso de inicio iterativo consta de varias fases [5]:

1. En primer lugar, el proceso de inicio de sesión comienza con una secuencia denominada *Secure Attention Sequence (SAS)*, esta secuencia es *CTRL + ALT + DEL* por defecto e inicia el proceso WinLogon.

2. Una vez iniciado el proceso WinLogon, este ejecuta el proceso *LogonUI* que proporciona la interfaz por defecto para introducir las credenciales y a su vez carga las bibliotecas

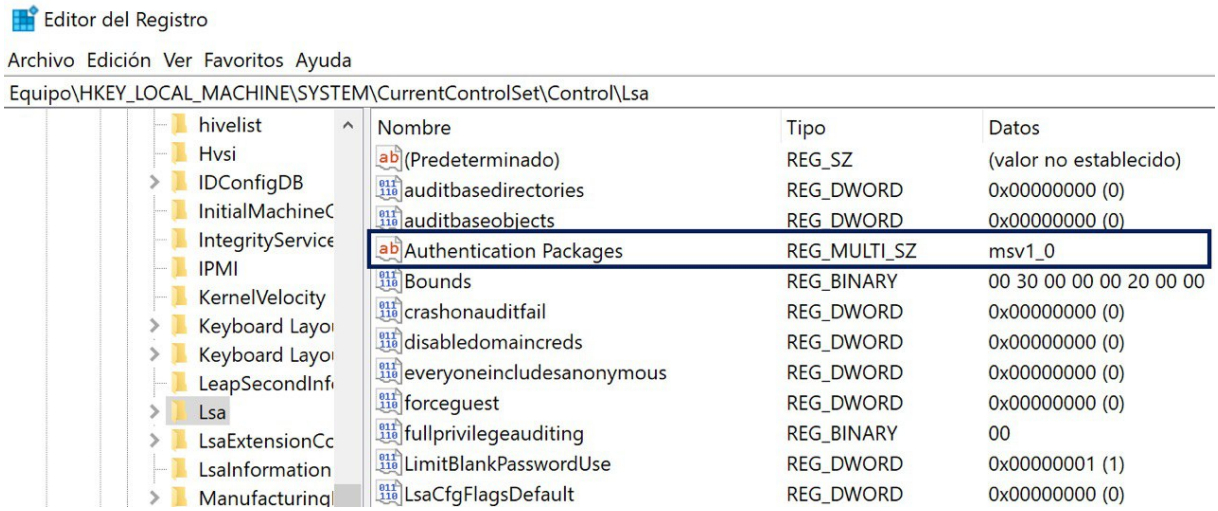

Fig. 2.2. Clave de registro sobre los paquetes de autenticación.

de enlace dinámico, del inglés *Dynamic-Link Library (DLL)* que se encargan de recoger las credenciales y pasarlas al proceso denominado Servicio Subsistema de Autoridad de Seguridad Local del inglés *Local Security Authority Subsystem Service* (LSASS). Estas DLLs denominadas *Credential Providers* se encuentran en [1] o en [2] (Si se trata de un inicio de sesión con Smart Cards).

3. Al ejecutarse Winlogon, también se crea un número identificador de seguridad del inglés *Security Identifier* (SID), [6] estructura de datos que identifica a un usuario, grupo y cuentas. Cada cuenta en dominio tiene un único SID que le identifica. Los procesos de Windows utilizar el SID asociado en lugar del nombre de usuario o el grupo al que pertenece, este número se pasa como argumento en la llamada *LsaLogonUser* y será incluido en el Token de Acceso *(Access Token)* si la autenticación se procesa correctamente.

4. Una vez introducido usuario y contraseña, WinLogon llama al proceso LSASS a través de la función *LsaLookupAuthenticationPackage*. Esta función tiene como objetivo obtener los paquetes de autenticación disponibles en el sistema a través de la clave de registro [3] como se puede observar en la Figura 2.2.

5. Posteriormente, se envían las credendiales a través de la función *LsaLogonUser*. Si algún paquete de autenticación autentica el usuario el proceso continua, en cambio, si ningún paquete indica que se ha iniciado sesión correctamente el proceso acaba.

6. Una vez autenticado, el proceso LSASS comprobará en la base de datos de políticas locales si el usaurio autenticado tiene los permisos suficientes para realizar la acción que está solicitando. Si el inicio de sesión no coincide el proceso de autenticación

[1] %SystemRoot%\System32\authui.dll
[2] %SystemRoot%\System32\SmartcardCredentialProvider.dll
[3] HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa

acaba y LSASS elimina cualquier estructura de datos creada y lo notifica a WinLogon. Si el acceso está permitido, LSASS agrega los IDs de seguridad correspondientes, busca en la base de datos los permisos asociados a los usuarios del mismo grupo del SID del usuario y los añade al token de acceso (*Access Token*) y crea el Token que será enviado a Winlogon con un mensaje de inicio de sesión correcto.

7. Por último, Winlogon mira en el registro [4] y crea un proceso con el valor que haya contenido en el registro. El valor por defecto es *Userinit.exe* que carga el perfil del usuario autenticado.

Una vez definido el proceso de inicio interactivo a grandes rasgos, se va a pasar a detallar los componentes mencionados que forman parte de dicho proceso.

2.1.2. Local Security Authority (LSA)

El subsistema protegido *Local Security Authority* [7] en Sistemas Windows se encarga de validad el acceso a los objetos, comprobar si un usuario tiene permisos suficientes y generar mensajes de auditoría. Es decir, LSA se encarga de las siguientes acciones:

- Autenticar y registrar los usuarios en un sistema local, es decir, se encarga del proceso visto anteriormente.
- Administrar la politica de seguridad local de un sistema, del inglés *Local Security Policy*.
- Proporcionar los servicios necesarios tanto para la autenticación de un usuario, como para la generación de los tokens de acceso correspondientes.
- Gestionar los servicios necesarios para mantener la relación entre nombres y SIDs.

Local Security Authority Subsystem Service (LSASS)

El proceso *Local Security Authority Subsystem Service (LSASS)* se encarga de instanciar las politicas de seguridad en el sistema, realizar un seguimiento de las políticas de seguridad de las cuentas activas, modificar credenciales y crear tokens de acceso y almacenar las credenciales de los usuarios activos del sitema, esto permite que un usuario no tenga que introducir las credenciales cada vez que accede a un recurso, esto se denomina *Single Sign-On* [8].

Este proceso es de gran interés para los atacantes ya que LSASS puede almacenar credenciales como Tikets de Kerberos, Hashes NT, Hashes LM o credenciales con algoritmos

[4] HKLM\SOFTWARE\Microsoft\Windows NT\Current Version\Winlogon\Userinit

de cifrado débiles que se puede obtener la contraseña en texto claro.

Las contraseñas son almacenadas en LSASS cuando:

- Se inicia sesión localmente (o remotamente a través de *Remote Desktop Protocol (RDP)*).
- Se ejecuta un proceso o tarea usando el comando *RunAs*.
- Se ejecuta un servicio de Windows que necesite mayores privilegios de los actuales y se requiere autenticación.
- Se ejecuta una tarea programada para la que es necesario autenticarse.
- Se ejecuta una tarea local usando una herramienta de administración remota.

2.1.3. Security Support Provider Interface (SSPI)

Security Support Provider Interface (SSPI) [9] es una API que permite que una aplicación pueda utilizar varios modelos de seguridad, es decir, abstrae las llamadas necesarias en el proceso de autenticación y permite que una aplicación lleve a cabo un proceso de autenticación sin especificar los protocolos de autenticación, denominados paquetes de autenticación que se verán en detalle en la siguiente sección de este capítulo.

En primer lugar, se negocia el protocolo a utilizar, para que el proceso se complete correctamente ambas máquinas deben aceptar mismo *Security Support Provider (SSP)*. Un SSP es un DLL que implementa SSPI y permite la ejecució de los paquetes de autenticación. Cada paquete proporciona un "mapeo.[en]tre las llamadas de funciones SSPI de una aplicación y las funciones de un modelo de seguridad.

Los principales SPP son:

- Kerberos.

```
%SystemRoot\%\Windows\System32\kerberos.dll
```

- NT Lan Manager (NTLM): NTLMv1 y NTLMv2.

```
%SystemRoot%\Windows\System32\msv1_0dll
```

- Digest.

```
%SystemRoot%\Windows\System32\Wdigest.dll
```

- Schanell.

```
%SystemRoot%\Windows\System32\Schannel.dll
```

- Negotiate.

```
%SystemRoot%\Windows\System32\lsasrv.dll
```

Negotiate

Microsoft Negotiate [10] es un SSP que actua como intermediario entre la API SPPI y otro SSP. Cuando una aplicación requiera algún tipo de autenticación, se envía una petición a Negotiate con los armunetos necesarios (parámetros, credenciales, SSPs a utilizar...), este lo examinará y pasará la petición al SSP correspondiente que llevará a cabo la autenticación.

Actualmente, Neogtiate elige entre Kerberos y NTLM. Seleccionará el primero simpre y cuando haya conexión entre las dos partes implicadas en el proceso y el usuario haya especificado el Service Principal Name (SPN), un User Principal Name (UPN), o una cuenta de NetBIOS. En cambio, si se trata de una autenticación local utilizará NTLM.

2.1.4. Authentication Packages

Windows utiliza dos paquetes de autenticación de forma estandar, Kerberos y MSV1_0. Como se ha comentado anteriormente Microsoft Windows utilizará el paquete MSV1_0 para sistemas independientes (que no están unidos a un dominio). Este protocolo implementa la versión 2 del protocolo Lan Manager. Además, este paquete se utiliza para sistemas en red unidos a un dominio cuya versión sea anterior a Windows 2000. Por otro lado, el paquete de autenticación Kerberos, se utiliza en sistemas que son parte de un dominio. El paquete de Windows Kerberos se comunicará con un Controlador de Dominio donde se ejecutará y comprobará la validación de las credenciales. Kerberos está desarrolado siguiendo el RFC4120 [11]. Los protocolos soportados por ambos paquetes de autenticación serán detallados en los siguientes capítulos.

2.1.5. Security Account Manager (SAM)

Uno de los principales elementos que forman parte de la autenticación local es *Security Account Manager (SAM)*. La SAM es un archivo que tiene todos los Sistemas Windows y consiste en una base de datos que almacena las credenciales de los usuarios locales. Este fichero almacena el identificador de usuario, nombre de usuario y el hash de la contraseña. Este último puede ser LM o NT. El fichero se encuentra en:

```
%SystemRoot%\Windows\System32\config\SAM
```

2.1.6. Access Token

Una vez validada la autenticación, se crea un *Access Token*, este objeto describe el contexto de seguridad de un proceso o de un hilo (*thread*) [12]. Para SSPI se denomina contexto de seguridad a una estructura de datos que contiene datos relevantes de seguridad como puede ser la clave de sesión o la duración de dicha sesión. Cada proceso ejecutado por un usuario dispone de una copia del token de acceso de ese usuario. Windows utiliza estos tokens para identificar a un usuario cuando ejecuta un hilo que necesite privilegios de ese usuario. La información contenida en un *Access Token* es:

- El SID de la cuenta del usuario en cuestión.
- El SID del grupo de usuarios de los que el usuario es miembro.
- El SID de la session (logon session).
- Los privilegios del usuario y del grupo de usuarios al que pertenece.
- El SID del grupo primario.
- El *Discretionary Access Control List* (DACL) [13] por defecto que se utiliza cuando el usuario crea un proceso sin especificar el descriptor de seguridad.
- La procedencia del token de acceso.
- Si el token de acceso es primario o es una suplantación.
- Una lista de SIDs restrictivos.
- Niveles de suplatación actuales.
- Otras estadísticas.

Cuando un administrador local inicia sesión en una máquina, se crean dos tokens diferentes: Un token primario que contendrá el contexto de seguridad del usuario y un token de administrador. Esto es debido a la política de Windows de mínimo privilegio posible, esto significa que el sistema usará por defecto el token primario cuando un proceso o hilo interactue con un *Securable Objects* [14]. Esto es importante a la hora de entender *User Account Control (UAC)*.

Para listar las sesiones logon en un sistema se puede utilizar el comando *logonsessions* de *Windows Sysinternals* [15] como se puede ver en la Figura 2.3.

```
[11] Logon session 00000000:00126688:
    User name:    WIN-R79MCJAAKB6\Cliente01
    Auth package: NTLM
    Logon type:   Interactive
    Session:      0
    Sid:          S-1-5-21-3305109258-3633115399-2278369259-1003
    Logon time:   18/08/2019 13:59:23
    Logon server: WIN-R79MCJAAKB6
    DNS Domain:
    UPN:
```

Fig. 2.3. Salida del comando Logonsessions.

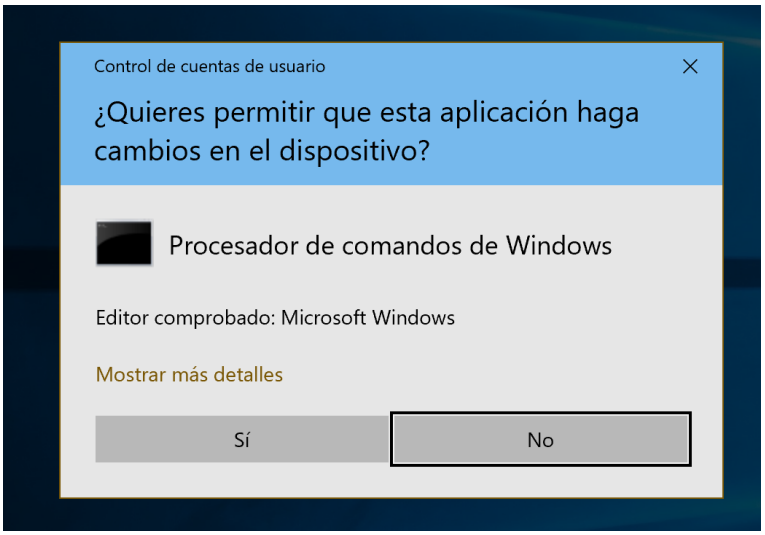

Fig. 2.4. Control de Cuentas de Usuario al ejecutar cmd.exe

2.1.7. User Account Control (UAC)

User Account Control (UAC) [16] [17]sirve para controlar cuando se están usando privilegios de adminstración. Como se ha comentado anteriormente al iniciar sesión desde una cuenta administrativa, se crean dos tokens de usuario: uno denominado *full access token* y un token secundario denominado *filtered access token*. Esto permite que los procesos ejecutados por este usuario se ejecuten con el segundo token siempre y cuando no necesiten privilegios de administración y así cumplir la política de mínimo privilegio posible. Esto lo podemos observar cuando se completa un inicio de sesión válido, el proceso *Explorer.exe* se ejecuta con el *filtered access token*. Cuando un proceso concreto necesita privilegios de administración, se requerirá una elvación de privilegios realizando una elevación de UAC. Como se puede ver en la Figura 2.4 se ha ejecutado una terminal *(cmd.exe)* con privilegios administrativos y aparece el mensaje de Control de Cuentas de Usuario para elevar de privilegios.

Para controlar qué procesos necesitan privilegios especiales o cuales no, los Sistemas Windows hacen uso de *Mandatory Integrity Control* [18], un mecanismo para controlar el acceso a los *Securable Objects* [14], para ello, el MIC utiliza niveles de integridad para evaluar el acceso a un proceso. Estos niveles son: *untrusted, low, medium, high, system e installer* [19] [20]. Esto implica que un objeto con integridad baja (low) no puede escribir en un objeto de integridad medio.

- **Untrusted:** Son aquellos procesos iniciados de forma anónima, como puede ser procesos lanzados desde cuentas de Invitados.
- **Low:** Nivel de integridad usado por defecto para la interacción con internet. Cuando se lanza *Internet Explorer* se utiliza este modo, por lo tanto, todos los archivos y procesos asociados a este se le asignan un nivel de integridad bajo.
- **Medium:** Nivel de integridad usado por la mayoría de los procesos, es el nivel designado por defecto siempre y cuando no se especifique explícitamente un nivel inferior o superior.
- **High:** Nivel de integridad destinado para aquellos procesos que necesitan privilegios de administración. Los objetos con este nivel de integridad solicitarán una elevaciñon de privilegios a través de UAC.
- **System:** Nivel de integridad reservado para los objetos del sistema, estos objetos engloban el kernel de Windows y servicios del *core*.
- **Installer:** Nivel de integridad especial utilizado para la instalación de software. Este nivel es igual o superior a todos los niveles anteiores, lo que permite que este nivel puede desinstalar los demás objetos.

2.2. NT Lan Manager (NTLM)

Como se ha comentado anteriormente, cuando un usuario inicia sesión localmente en Sistemas Windows utiliza el paquete de autenticacióni MSV1_0 [21]. LSA llama a este paquete para procesar los datos de inicio de sesión recogidos en el proceso WinLogon para su posteior comprobación con la información contenida en la base de datos SAM. En este apartado se va a detallar cómo gestiona Windows el almacenamiento de dichos datos a través de NT Lan Manager (NTLM).

2.2.1. (

Windows Hashes)

Para almacenar los contraseñas en la base de datos, los Sistemas Windows utilizan Hashes *Lan Manager (LM)* y NT.

LM

Los hashes LM son la versión que s

2.3. Kerberos

2.4. Active Directory

Bibliografía

[1] Microsoft, "Introducción a active directory," 2000. https://support.microsoft.com/es-es/help/196464.

[2] M. Bresman, "Wannacry, notpetya, mbr-oni and friends: Tales of wiper attacks and active directory destruction," 2018. https://www.semperis.com/blog/wannacry-notpetya-wiper-attacks-active-directory/.

[3] Microsoft, "How interactive logon works," 2017. https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc780332(v=ws.10).

[4] Microsoft, "Windows server 2008 r2 and windows 7 authentication architecture," 2013. https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/dn169016%28v%3dws.10%29.

[5] B. Catlin, P. Yosifovich, J. Hanrahan, M. Russinovich, A. Ionescu, and D. Solomon, *Windows Internals: User Mode*. Windows internals ; Part 1, Microsoft Press, 2017.

[6] Microsoft, "Security identifiers," 2018. https://docs.microsoft.com/es-es/windows/win32/secauthz/security-identifiers.

[7] Microsoft, "Lsa authentication," 2013. https://docs.microsoft.com/en-us/windows/win32/secauthn/lsa-authentication.

[8] Microsoft, "Single sign-on in windows 2000 networks," 2009. https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/bb742456(v=technet.10).

[9] Microsoft, "Sspi," 2018. https://docs.microsoft.com/en-us/windows/win32/secauthn/sspi.

[10] Microsoft, "Microsoft negotiate," 2018. https://docs.microsoft.com/en-us/windows/win32/secauthn/microsoft-negotiate.

[11] K. R. C. Neuman, S. Hartman, "The kerberos network authentication service (v5)," 2005. https://tools.ietf.org/html/rfc4120.

[12] Microsoft, "Access tokens," 2018. https://docs.microsoft.com/es-es/windows/win32/secauthz/access-tokens.

[13] Microsoft, "Access control lists," 2018. https://docs.microsoft.com/es-es/windows/win32/secauthz/access-control-lists.

[14] Microsoft, "Securable objects," 2018. https://docs.microsoft.com/es-es/windows/win32/secauthz/securable-objects.

[15] Microsoft, "Windows sysinternals," 2017. https://docs.microsoft.com/es-es/sysinternals.

[16] Microsoft, "User account control," 2018. https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc731416(v=ws.10).

[17] Microsoft, "Cómo funciona el control de cuentas de usuario," 2018. https://docs.microsoft.com/es-es/windows/security/identity-protection/user-account-control/how-user-account-control-works.

[18] Microsoft, "Mandatory integrity control," 2018. https://docs.microsoft.com/en-us/windows/win32/secauthz/mandatory-integrity-control.

[19] T. Bradley, "Introduction to windows integrity control," 2007. https://www.symantec.com/connect/articles/introduction-windows-integrity-control.

[20] Microsoft, "Modify an object label," 2017. https://docs.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/modify-an-object-label.

[21] Microsoft, "Msv1_0 authentication package," 2018. https://docs.microsoft.com/en-us/windows/win32/secauthn/msv1-0-authentication-package.