



CURSO DE HACKING ÉTICO

AUDITORÍA CREDENCIALES DE WINDOWS

Tipos de ataques de contraseña

- ☐ Password Guessing
- ☐ Password Cracking
- ☐ Password Cracking (Rainbow Tables)
- ☐ Pass The Hash

Password Guessing

Es el nombre que se utiliza al intento de obtener credenciales válidas tratando de adivinar contraseñas de un sistema remoto mediante fuerza bruta. Estos ataques son siempre eficaces aunque se enfrentan a los siguientes problemas:

- ❑ **Lentitud:** Al tratarse de un ataque contra equipos remotos que requiere comprobación mediante ensayo y error la ejecución es lenta, además en ocasiones es necesario añadir tiempos de espera para evitar otros problemas que se describen a continuación.
- ❑ **Registros:** Como se lanzan cientos de pruebas los registros del sistema se llenarán rápidamente con múltiples líneas revelando nuestras intenciones. Aunque no es un problema en sí, ya que estamos autorizados a lanzar estas pruebas, se pueden generar otros relacionados, como por ejemplo que se llene un volumen de datos y el equipo deje de registrar eventos importantes o inutilice el servicio.
- ❑ **Medidas preventivas:** Para evitar este tipo intrusión, los sistemas y aplicaciones en ocasiones implementan algunas medidas preventivas, como por ejemplo el uso de [captchas](#), la prolongación del tiempo hasta que informa de si es incorrecto o no, la inhabilitación o filtrado en el cortafuegos de la dirección IP de origen o el bloqueo del usuario para que no pueda acceder.
- ❑ **Bloqueo de cuentas:** Merece especial atención esta medida, ya que una configuración y planificación incorrecta del ataque podría bloquear el acceso a todas las cuentas de la red, dejando completamente inaccesible los sistemas incluso para los administradores.

Password Guessing

Para evitar este bloqueo de usuarios, antes de lanzar el análisis se tomarán los siguientes datos y precauciones para configurar la herramienta:

- ❑ **Número de intentos antes de bloqueo:** Con una cuenta de prueba se contarán el número de veces que permite incorrectamente una contraseña antes de bloquear e inutilizar el usuario. El ataque se hará de ese número-1, para evitar la protección y no llegar nunca a bloquear a nadie.
- ❑ **Duración del bloqueo:** En algunos productos si se produce un bloqueo este es desactivado automáticamente una vez ha transcurrido un tiempo. El bloqueo puede ser al usuario o a nuestra dirección IP. Se ha de averiguar si existe este tiempo y que duración tiene para considerarlo y añadir esperas entre los bancos de pruebas.
- ❑ **Duración antes que no se considere un nuevo intento:** En ocasiones si los intentos de acceso son cada mucho tiempo el contador de intentos no suma la nueva prueba fallida. Esto es útil para jugar nuevamente con los tiempos y añadir tiempos de retardo o probar otros usuarios y otras contraseñas mientras transcurre esa ventana de tiempo.
- ❑ **Prueba inversa:** O *user guessing*, es un método de evitar bloqueos que funciona en la mayoría de los casos. Consiste en dada una contraseña de uso común, probar nombres de usuario que la hayan utilizado. El mismo proceso pero en orden inverso.
- ❑ **Reinicio del contador de bloqueo en caso de usuario correcto:** Se ha de contemplar la posibilidad de que el contador de intentos fallidos se reinicie si se introduce un usuario y contraseña válido. Si fuese así, se probará la fuerza bruta contra un usuario mientras se intercalan accesos válidos con uno de prueba o que ya se haya obtenido.

Password Guessing

Para que el ataque este lo más optimizado posible a la plataforma es conveniente conocer la política de usuarios y contraseñas que sigue el servicio analizado. Los factores a controlar son:

- ☐ **Longitud mínima y máxima de la contraseña:** De esta forma el diccionario tendrá que cumplir los requisitos.
- ☐ **Caracteres obligatorios:** Así se podrá determinar si es obligatorio algún tipo de combinación y si tiene sentido el uso de un diccionario o no.
- ☐ **¿Permite que la contraseña sea igual al usuario?:** Ya que este tipo de ataque es habitual se debe conocer si se contempla esta posibilidad o no.

Herramientas y referencias

- ☐ CeWL, Custom word List Generator - <http://www.digininja.org/projects/cewl.php>
- ☐ Wyp - http://www.remote-exploit.org/?page_id=418
- ☐ Diccionarios - <http://www.skullsecurity.org/wiki/index.php/Passwords>
- ☐ ncrack - <http://nmap.org/ncrack/man.html>
- ☐ THC-Hydra - <http://freeworld.thc.org/thc-hydra/>
- ☐ Medusa - <http://www.foofus.net/jmk/medusa/medusa.html>

Password Cracking

El cracking de contraseñas consiste en intentar obtener el valor en claro de algún elemento cifrado.

Lo más común en un test de intrusión es ejecutar el ataque contra los ficheros de contraseñas de los sistemas operativos y aplicaciones en los que se obtenga acceso mediante exploits, errores de configuración o accesos físicos.

Las ventajas de este sistema es que ninguna cuenta será bloqueada y pueden utilizarse uno o varios sistemas para paralelizar las tareas y ganar velocidad. En general una vez extraídas de la fuente, el proceso se llevará acabo localmente.












Los archivos de contraseñas más comunes en sistemas Unix son:

- ❑ **Linux, Solaris**: /etc/passwd y /etc/shadow
- ❑ **AIX**: /etc/security/passwd
- ❑ **HPUX**: /tcb/files/auth/*

Password Cracking

Aunque siempre se puede consultar en guías del estilo Unixguide.net otras rutas. Para todos los casos es necesario disponer de un usuario con **privilegios de root**.

Los cifrados más típicos para estos ficheros son 3Des, MD5 (que comenzarán por \$1\$), BSDi, SHA256 (\$5\$) y SHA512 (\$6\$). Para lo que lo normal es utilizar la aplicación [John The Ripper](#), aunque aún no tiene soporte nativo para SHA512 o SHA256 usados en los últimos Linux. En este caso será necesario aplicar un [parche](#).

Patch	Author	Description
 generic crypt(3) support	Solar Designer	Status: separate patch, currently relevant. Normally, JtR uses its own optimized crypto code, but this patch lets it also use the underlying Unix-like system's crypt(3) function. This is particularly useful for  cracking glibc 2.7+ "SHA-crypt" hashes (such as on recent Fedora and Ubuntu) until proper "native" support for that is implemented.  Other uses are possible as well.
 MySQL(323)-old fast algorithm for 1.7.2	Balázs Bucsay	Status: integrated into jumbo patch
 raw-MD5 fast algorithm for 1.7.2	Balázs Bucsay	Status: integrated into jumbo patch
 Oracle 11g for 1.7.2	Alexandre Hamelin	Status: integrated into jumbo patch
 Oracle 11g for 1.7.3.1	Alexandre Hamelin	Status: integrated into jumbo patch
 mpi10 patch for 1.7.3.1	RB	Status: separate patch, currently relevant. Latest stripped version of bindshell.net MPI patch.
 mpi10 patch for 1.7.5	magnum	Status: separate patch, currently relevant. This is just the same as above after hand-editing the rejects
 Extended MPI for 1.7.5	magnum	Status: separate patch, currently relevant. This is version 3. Apply after jumbo-2 See  mailing list announcement

Password Cracking

Si disponemos de un **hash** del que desconocemos el algoritmo de origen, se puede consultar la [web de John](#) para tratar de averiguarlo por su formato.

En el caso de los sistemas Windows se definen dos algoritmos:

- ❑ **LANMAN (LM)**: En sistemas hasta XP.
- ❑ **NT Hash (NTLM)**: En 2008, Vista y windows 7.
- ❑ **LANMAN** tiene varias debilidades importantes, no diferencia entre mayúsculas y minúsculas, lo que reduce el número posible de combinaciones y añade datos a contraseñas inferiores de 8 caracteres, por lo que observando la terminación del hash (AAD3B435B51404EE) es posible saber si es mayor o menor que este tamaño.
- ❑ **AES**: Windows 10, 2016 y 2019 a partir año 2015 con Windows Anniversary Update

Además de ser Administrador o SYSTEM, para volcar las contraseñas en sistemas Windows es necesario utilizar aplicaciones del estilo Pwdump en alguna de sus [múltiples versiones](#).

Las opciones más comunes y cómodas para el cracking de contraseñas son el uso de técnicas "time-memory-trade-off" o rainbow tables. Con las que previamente se almacenan en ficheros todas las combinaciones posibles para posteriormente ser consultadas.

Password Cracking

Las herramientas para la generación de Rainbow tables:

- ❑ **rtgen:** <http://project-rainbowcrack.com/>
- ❑ **Cain&Abel (winrtgen):** <http://www.oxid.it>
- ❑ **Precomp (Ophcrack):** <http://ophcrack.sourceforge.net>
- ❑ Aunque estas tablas también pueden descargarse de sitios como: Free Rainbow Tables: <http://www.freerainbowtables.com/>
- ❑ **Shmoo group:** <http://rainbowtables.shmoo.com/>
- ❑ **Ophcrack:** <http://ophcrack.sourceforge.net/>



ophcrack



Pass the Hash

Pass The Hash es una técnica que permite al atacante, a través del protocolo NTLM, un intento de acceso a un sistema remoto sin la necesidad de conocer la contraseña del usuario.

Windows almacena diferentes tipos de valores hash, derivados de la contraseña del usuario, para permitir el acceso a diferentes servicios sin la necesidad de volver a ingresar la contraseña.

El protocolo NTLM utiliza el hash NT para la autenticación y no hace un 'salt' de la contraseña, lo que a su vez significa que si uno toma el valor del hash, la autenticación se puede realizar sin conocer la contraseña real.

Pass the Hash

Un atacante por lo tanto puede autenticarse sin una contraseña pasando el hash. Para ello tendrá dos opciones:

- ☐ Inyectar el hash en LSASS.exe y abrir sesión con el hash inyectado.
- ☐ Implementar parte del protocolo NTLM para la autenticación con el hash y enviar comandos a través de la red con protocolos como SMB, WMI, etc.

La principal diferencia entre pasar el hash a una conexión NTLM legítima es el uso de una contraseña. La búsqueda de inicios de sesión de usuarios legítimos, donde se usó la contraseña antes de la conexión NTLM, puede ayudar a filtrar todos los inicios de sesión legítimos y dejar solo el inicio de sesión sospechoso.

Pass the Hash

Conexiones legítimas desde el visor de eventos

El visor de eventos contiene una cantidad de registros que indican inicios de sesión interactivos:

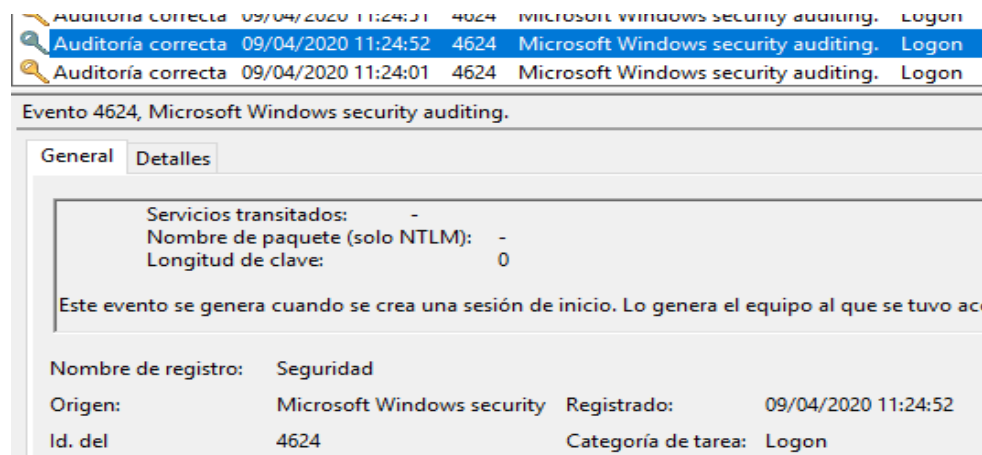
- 4768 - Se solicitó un ticket de autenticación de Kerberos (TGT)
- 4769 - Se solicitó un ticket de servicio Kerberos (TGS)
- 4648 - Se intentó un inicio de sesión utilizando credenciales explícitas
- 4624 - Se inició sesión correctamente en una cuenta

Tipos de inicio de sesión: 2 (Interactivo), 7 (Desbloqueo), 10 (RemoteInteractive) o 11 (CachedInteractive).

Comprobar que cada conexión NTLM tenía un inicio de sesión interactivo con la misma cuenta antes de la conexión, basándose en los registros anteriores, puede ayudar a distinguir entre un atacante que usa el hash y un usuario normal que usa la contraseña.

Pass the Hash

La empresa israelí [CyberArk](#) ha creado la herramienta **Ketshash** que automatiza la detección de conexiones NTLM con privilegios sospechosos, en particular ataques Pass-The-Hash, en función de los registros del visor de eventos.



```
PS C:\> Invoke-DetectPth -TargetComputers @"MARS-7", "MARS-10"
```

```
[*] TID: 5
[*] User Sid: S-1-5-21-2525669019-4238447758-834479404-1121
[*] Source computer name: MARS-10
[*] Target computer name: MARS-7
[*] User: CYBER\legit1
[*] Time: 11/26/2017 14:12:55
[*] Found a logon attempt using explicit credentials
[*] Legit logon
```