**REGIONAL ANTIOQUIA**

**CENTRO DE SERVICIOS Y GESTIÓN EMPRESARIAL**

**TECNOLOGÍA EN GESTIÓN EN REDES DE DATOS**

**(2803649)**

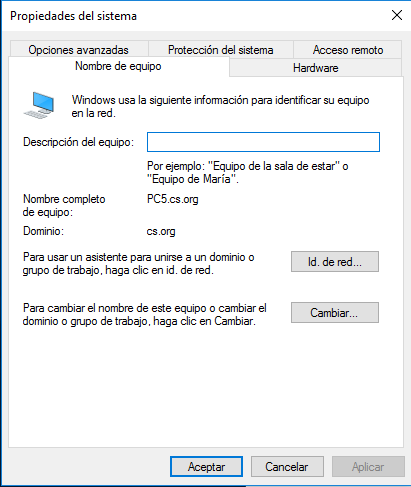
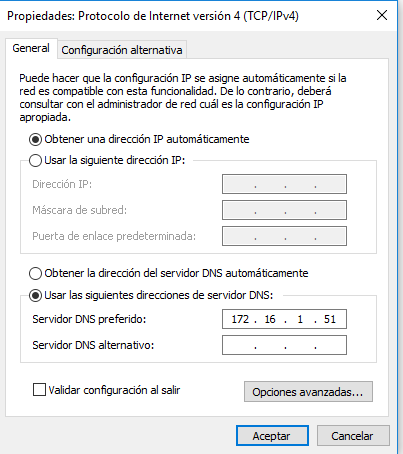
**Presentado por**:

Julián Montoya Posada

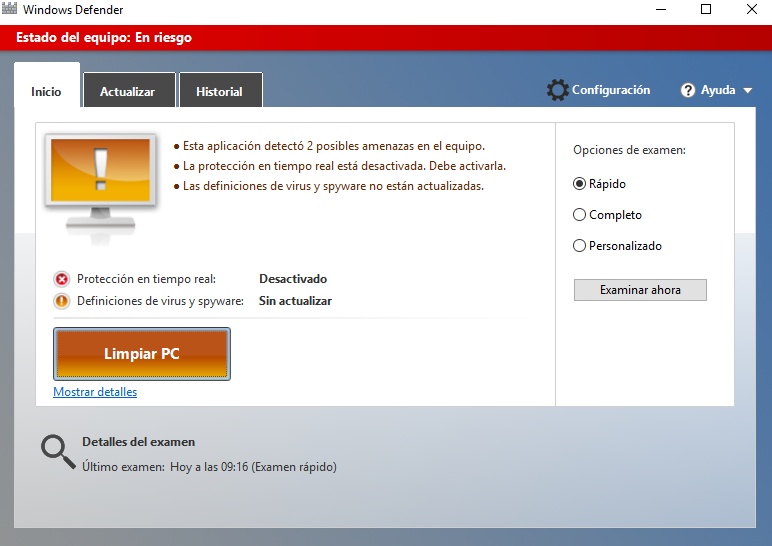
Medellín, 2024

**Active Directory**

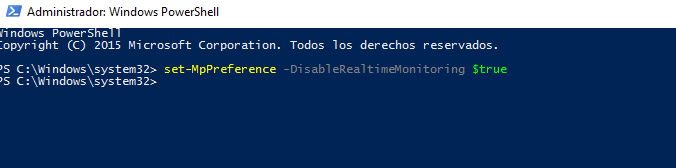
* Descargamos una máquina Windows 10 la cual va a estar unida a un dominio de directorio activo.
* Antes de unir la maquina Windows al dominio debemos de cambiarle el nombre del equipo, ya que nos vamos a enumerar en orden de números. En mi caso, el equipo se llama **PC5**, luego de esto se reinicia la máquina.
* Luego de esto vamos a poner en la tarjeta de red el DNS del dominio al cual nos vamos a unir, el DNS es **(172.16.1.51).** Si no hacemos esto no nos vamos a poder unir al dominio.
* Procedemos a unirnos al dominio llamado **(cs.org)**.

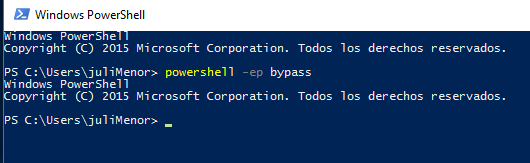
* Para esta actividad vamos a deshabilitar el Windows defender.



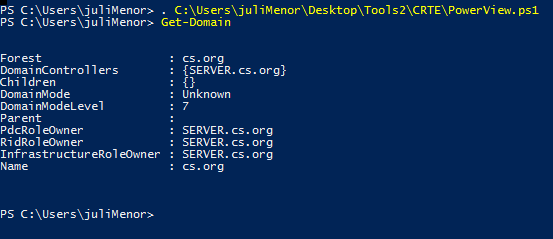
* Procedemos a familiarizarnos con la PowerShell.
* Primero ejecutamos la PowerShell como administrador para desactivar el Windows Defender.



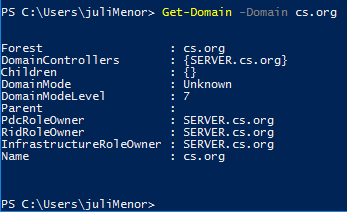
* Luego ejecutamos el comando **<powershell -ep bypass>** que tiene como propósito desactivar temporalmente las restricciones de ejecución de scripts en PowerShell.



* Procedemos a ejecutar el comando PowerView.ps1, este con el objetivo de cargar y realizar una enumeración del directorio activo, como usuarios, grupos y maquinas del dominio. **< . C:\Users\juliMenor\Desktop\Tools2\CRTE\PowerView.ps1 >**
* Vemos a que dominio estamos unidos con el comando **< Get-Domain>**



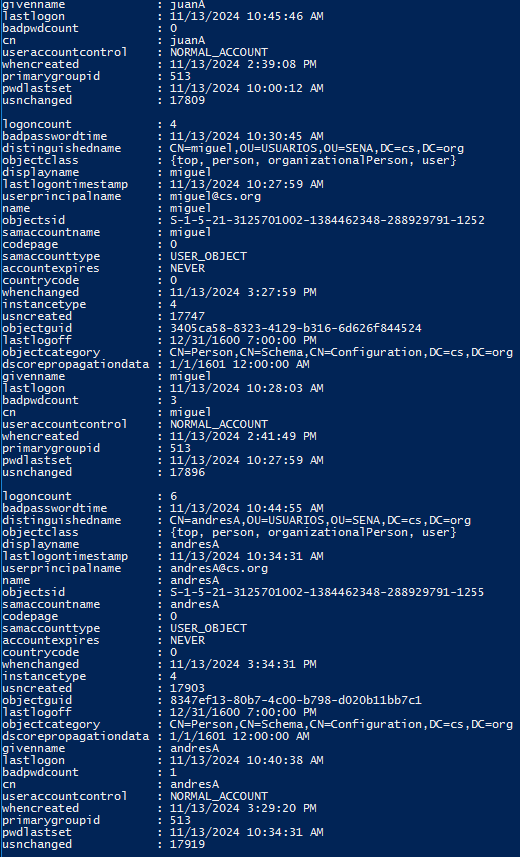
* Acá también vemos a que dominio estamos unidos, es diferente el comando y nos va arrojar la misma información.
* El comando utilizado es **< Get-Domain –Domain “nombre del dominio” >**



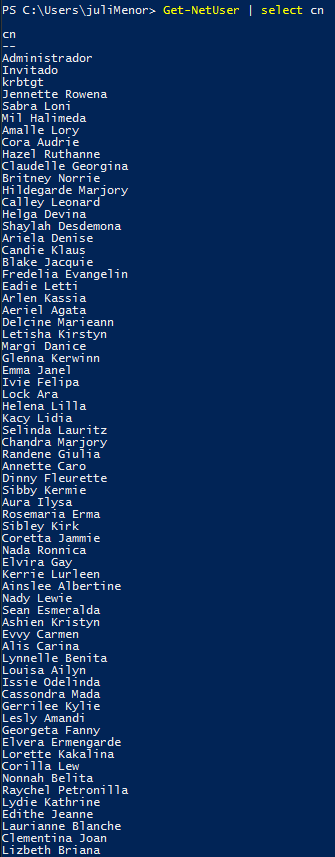
* Acá utilizamos el comando **< Get-DomainSID >.**
* El **SID** (Security Identifier) es un identificador único utilizado por Windows para representar objetos de seguridad, como usuarios, grupos y dominios.
* Cada dominio en Active Directory tiene su propio SID que se utiliza para identificarlo en la red.



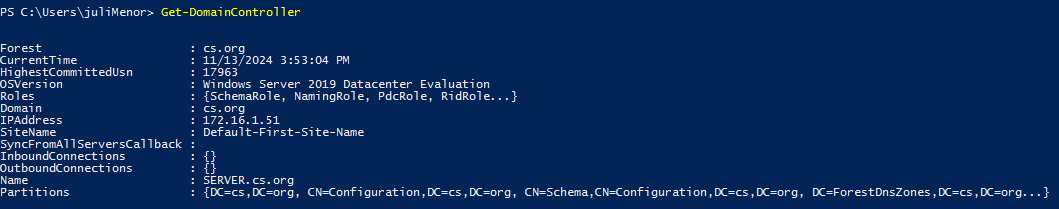
* Procedemos hacer la enumeración de cuentas de usuario de Active Directory con el comando **< Get-NetUser >.**
* Este nos va a listar todos los usuarios que tiene el dominio y nos va a decir más información sobre ellos.



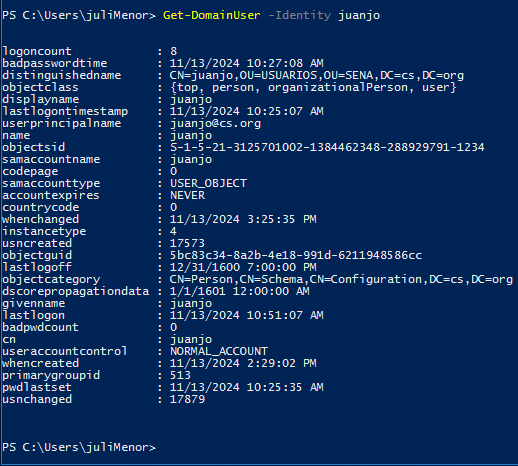
* Acá vamos a volver a listar los usuarios del AD pero solo nos va arrojar los nombres, no nos va a mostrar tanta información de los usuarios como en el paso anterior. Acá vamos a ver la información más específica.
* Utilizamos el comando **< Get-NetUser | select cn >.**



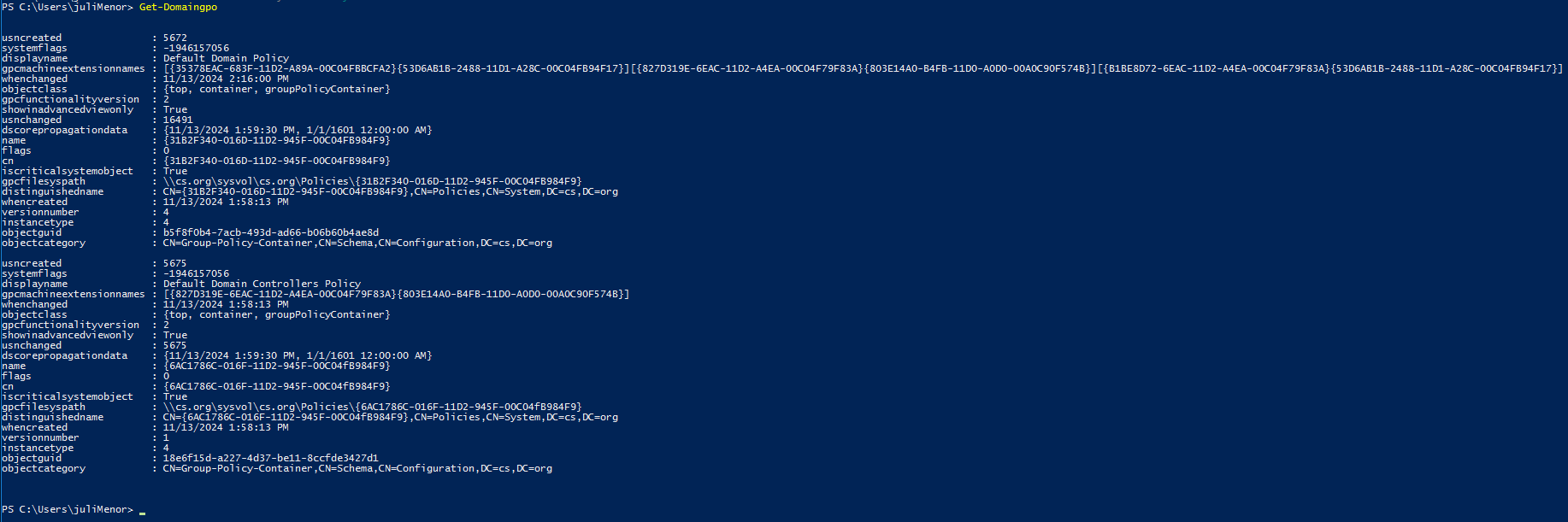
* Procedemos a listar y ver los detalles específicos del controlador de dominio.
* Utilizamos el comando **<Get-DomainController>**



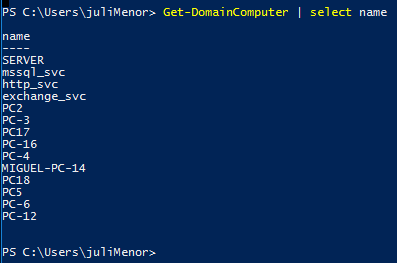
* Vemos información detallada sobre un usuario específico de Active Directory.
* Utilizamos el comando **< Get-DomainUser -Identity “nombre de usuario” >.**



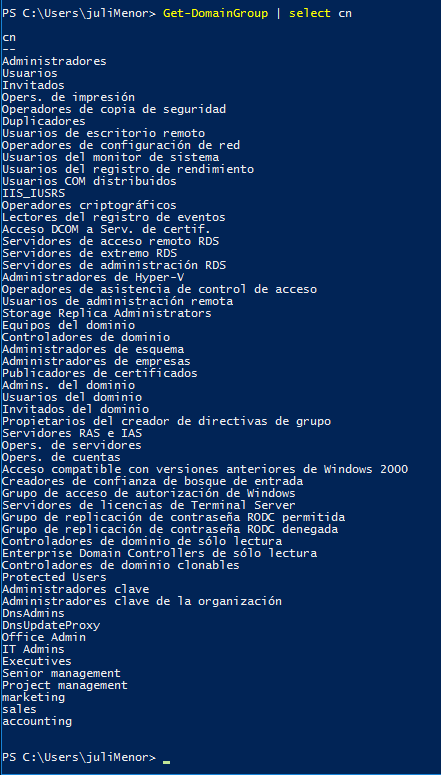
* Acá vamos a ver información sobre las Políticas de Grupo configuradas en el dominio.
* Utilizamos el comando **<Get-Domaingpo>**



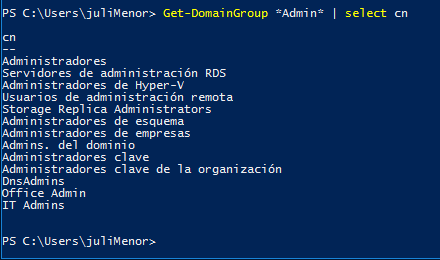
* Vamos a enumerar todos los equipos que forman parte de un dominio de Active Directory.
* Utilizamos el comando **< Get-DomainComputer | select name >.**



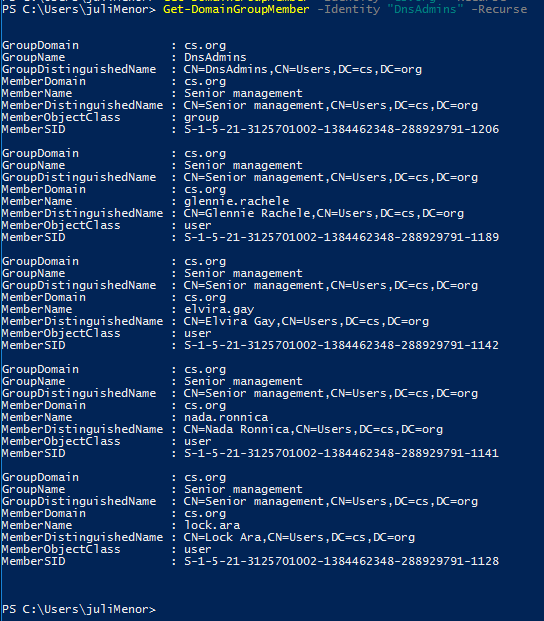
* Vamos a enumerar todos los grupos disponibles en un dominio de Active Directory.
* Utilizamos este comando para ello **< Get-DomainGroup | select cn >**



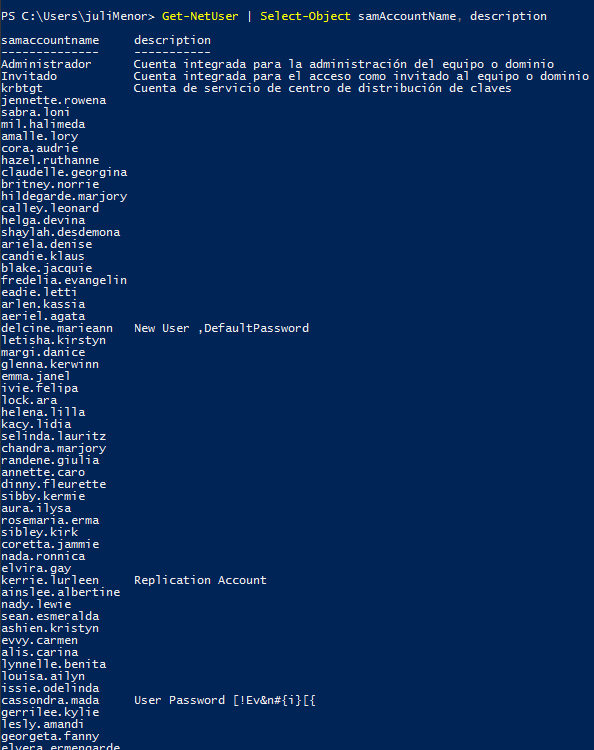
* Acá vamos a enumerar todos los grupos “Administradores” del Active Directory.
* Se utiliza el siguiente comando **< Get-DomainGroup \*Admin\* | select cn** **>**



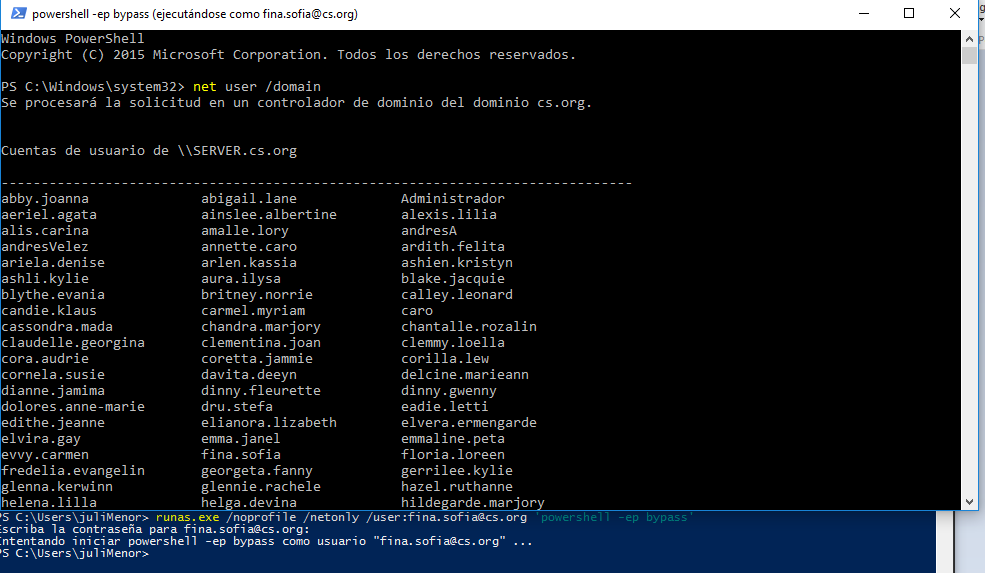
* Enumeramos todos los miembros de un grupo específico, en este caso, el grupo DNSAdmins, junto con los miembros indirectos (es decir, aquellos que pertenecen a grupos anidados
* Utilizamos el comando **< Get-DomainGroupMember –Identity “DNSAdmins” –Recurse >**



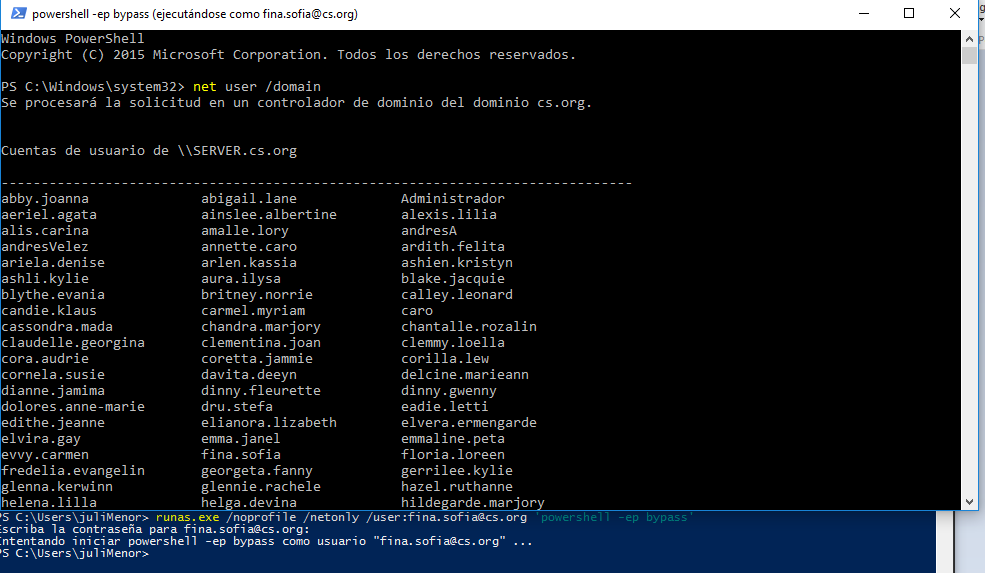
* Acá vamos a ver todos los usuarios del Active Directory con una descripción que seria la contraseña de algunos usuarios.
* Utilizamos el comando **< Get-NetUser | select-Object samAccountName, description >.**



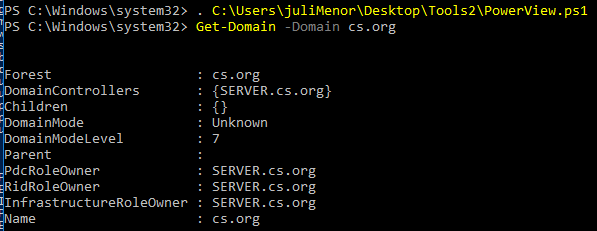
* Vamos a utilizar runas para ejecutar un comando con las credenciales de un usuario diferente en Windows.
* El comando utilizado fue **< runas.exe /noprofile /netonly /user:”usuario”@”dominio” powershell -ep bypass >**
* El comando intenta ejecutar una instancia de **PowerShell** utilizando las credenciales del usuario [fina.sofia@cs.org](mailto:fina.sofia@cs.org).
* La sesión de PowerShell se inicia con la política de ejecución configurada en **bypass**, lo que permite ejecutar cualquier script, sin restricciones.
* Dado que se utiliza **/netonly**, estas credenciales solo se autenticarán contra recursos remotos, como servidores de dominio o recursos en red, y no afectan al entorno local.



* Acá ya esta iniciada la PowerShell como usuario [fina.sofia@cs.org](mailto:fina.sofia@cs.org) y vamos a listar los usuarios del dominio.
* Utilizamos el comando **<Net user /domain >**



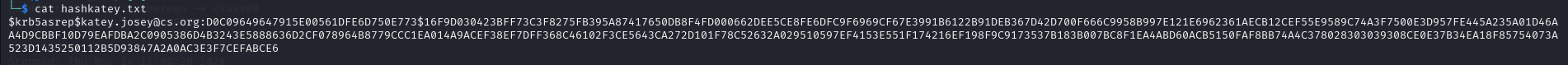
* En esta misma sesión de **“fina.sofia”** vamos a importar el PowerView para poder enumerar los recursos del dominio.
* Vamos a ver información sobre el dominio con el comando **< Get-Domain -Domain “dominio” >.**

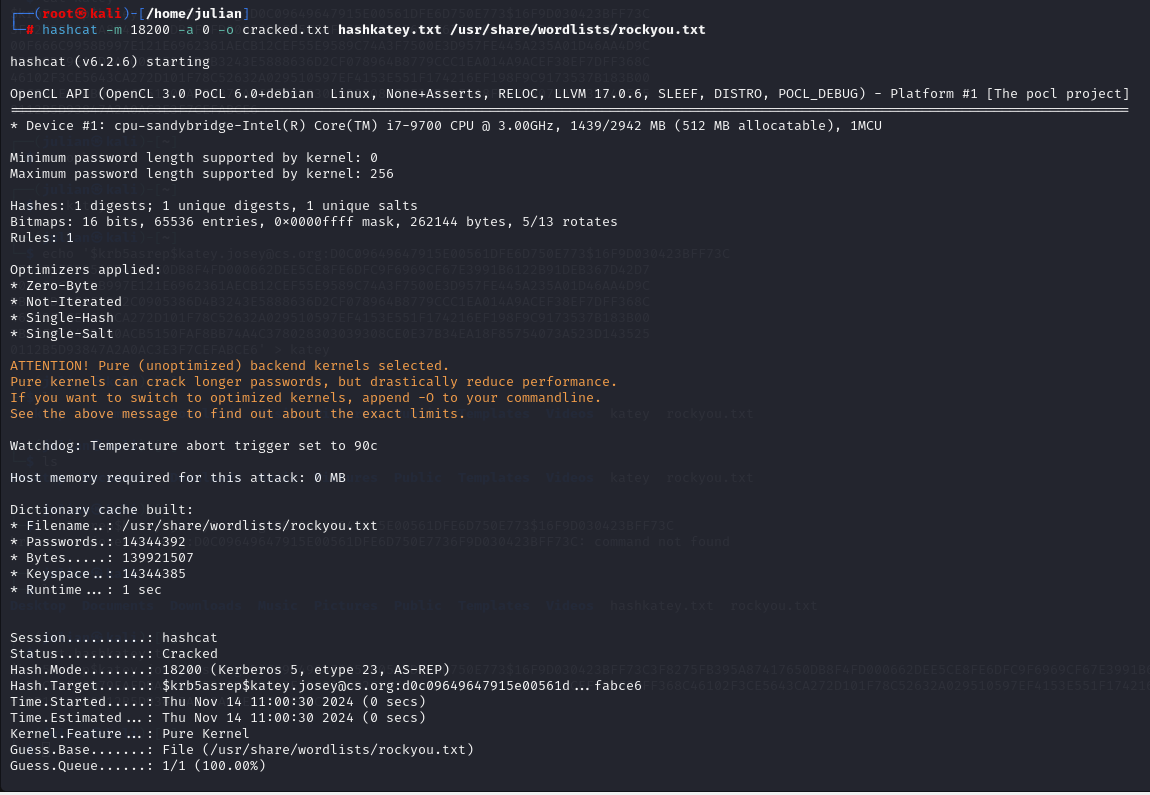


* **Rubeus:**
* Usamos la herramienta Rubeus, específicamente con el módulo AS-REP Roasting, para identificar cuentas en Active Directory que no requieren preautenticación Kerberos. Este fue utilizado para obtener hashes de contraseñas.

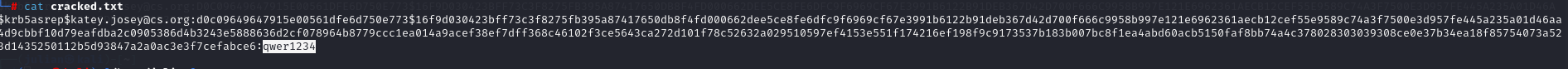


* **Hashcat:**
* Acá vamos a descifrar un hash de un usuario. Yo elegí el usuario katey.josey.
* Vamos a copiar el hash de ese usuario y lo vamos a guardar en un archivo de texto en el Kali. Lo guardamos como **hashkatey.txt.**
* El comando que utilizamos fue **< hashcat -m 18200 -o cracked.txt hashkatey.txt /usr/share/wordlists/rockyou.txt >**
* **hashcat:** Herramienta utilizada para realizar ataques de fuerza bruta o basados en diccionarios contra hashes.
* **-m 18200:** Especifica el modo de ataque. El valor 18200 indica que se está atacando un hash Kerberos 5 AS-REP (etype 23).
* **-o cracked.txt:** Define el archivo donde se guardarán las contraseñas descifradas.
* **hashkatey.txt:** Archivo que contiene el hash Kerberos obtenido (en este caso, asociado al usuario "katey").
* **/usr/share/wordlists/rockyou.txt:** Diccionario de contraseñas comúnmente que contiene millones de contraseñas posibles.

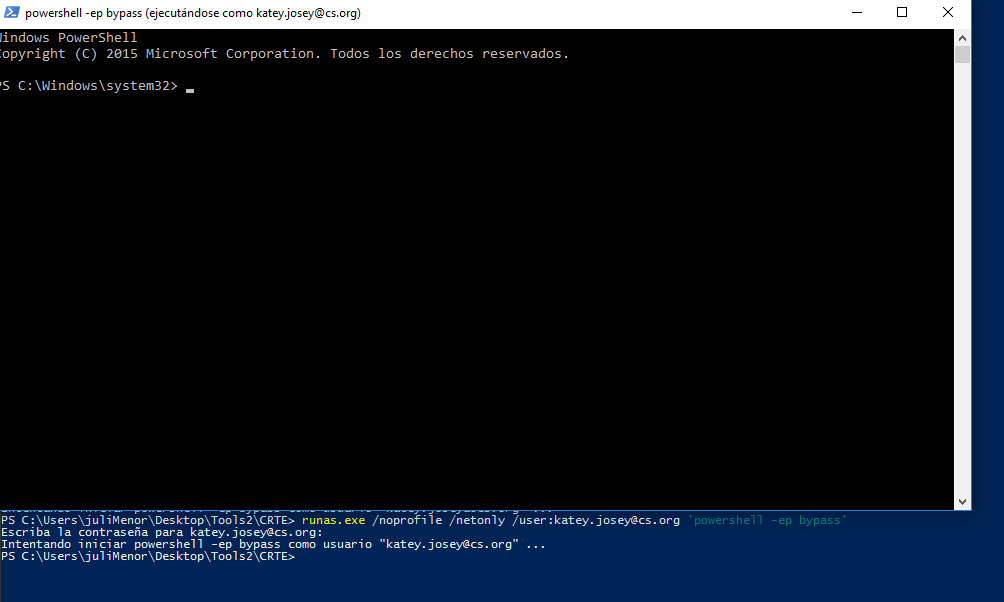


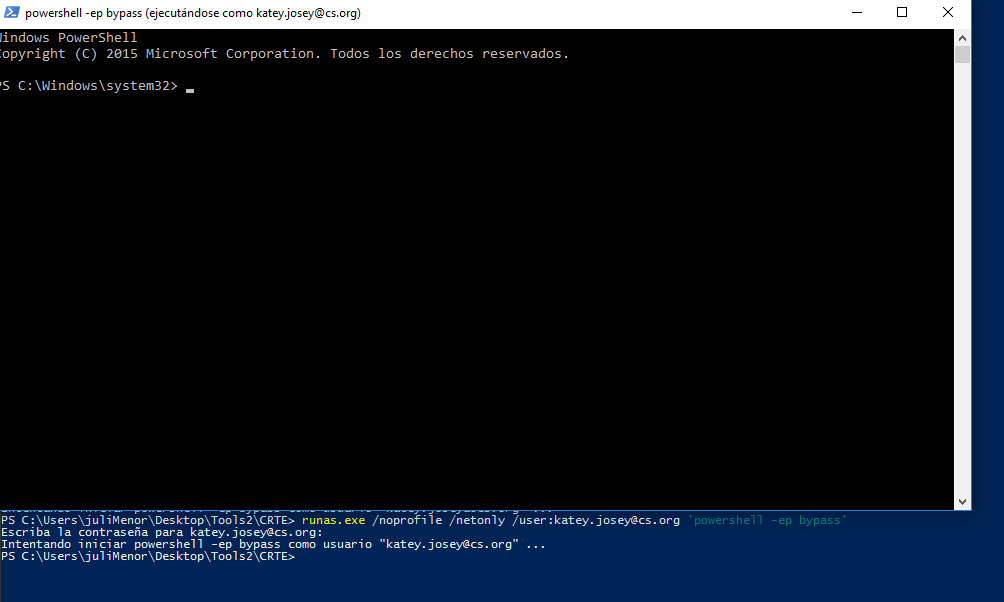


* Luego de ejecutar el hashcat vemos que la contraseña que nos dio de ese usuario fue **“qwer1234”**

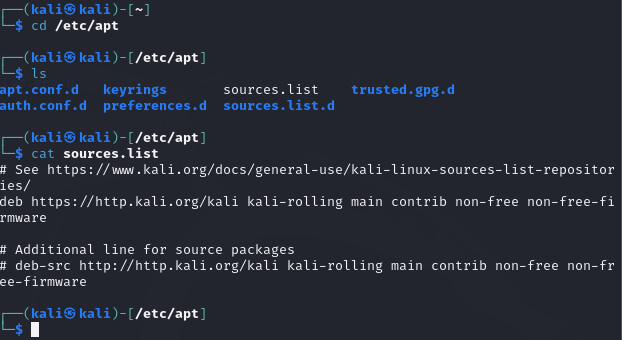


* Por medio de hashcat desciframos la contraseña del usuario katey.josey.
* Ya con esta contraseña que era lo único que nos faltaba, vamos a iniciar sesión en una PowerShell.
* Con el uso de la herramienta runas.exe para ejecutar un proceso de PowerShell bajo las credenciales del usuario [katey.josey@cs.org](mailto:katey.josey@cs.org) y su contraseña obtenida.
* El comando que se utilizo fue **< runas.exe /noprofile /netonly /user:katey.josey@cs.org "powershell -ep bypass" >.**

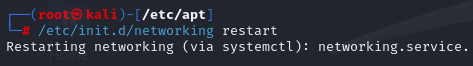




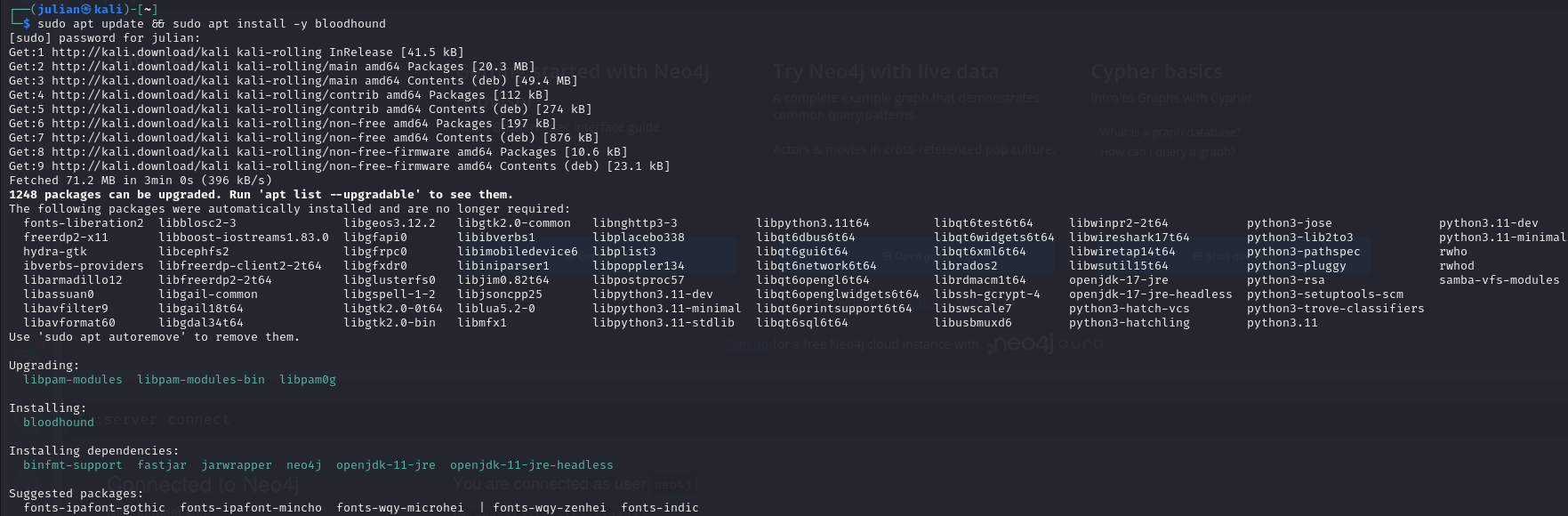
* **Bloodhund:**
* Instalamos Bloodhund en el Kali Linux.
* Primero vamos a ir a esta ruta **“cd /etc/apt”**, dentro de esta ruta vamos a ver el archivo **sources.list.**
* Dentro de este archivo vamos a editar el **“deb** [**http://http.kali.org/kali**](http://http.kali.org/kali) **kali-rolling main contrib non-free non-free-firmware”.**
* Lo vamos a cambiar agregándole la **s** al http y quedaría de esta forma **“deb** [**https://http.kali.org/kali**](https://http.kali.org/kali) **kali-rolling main contrib non-free non-free-firmware”.**



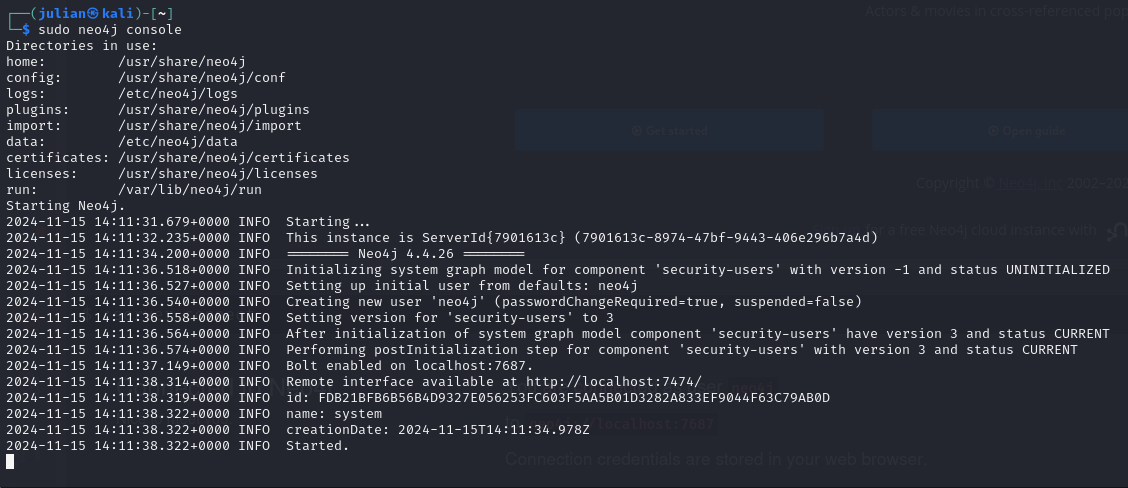
* Luego de editar el archivo sources.list, vamos a darle un restart al servicio networking
* /etc/init.d/networking restart.



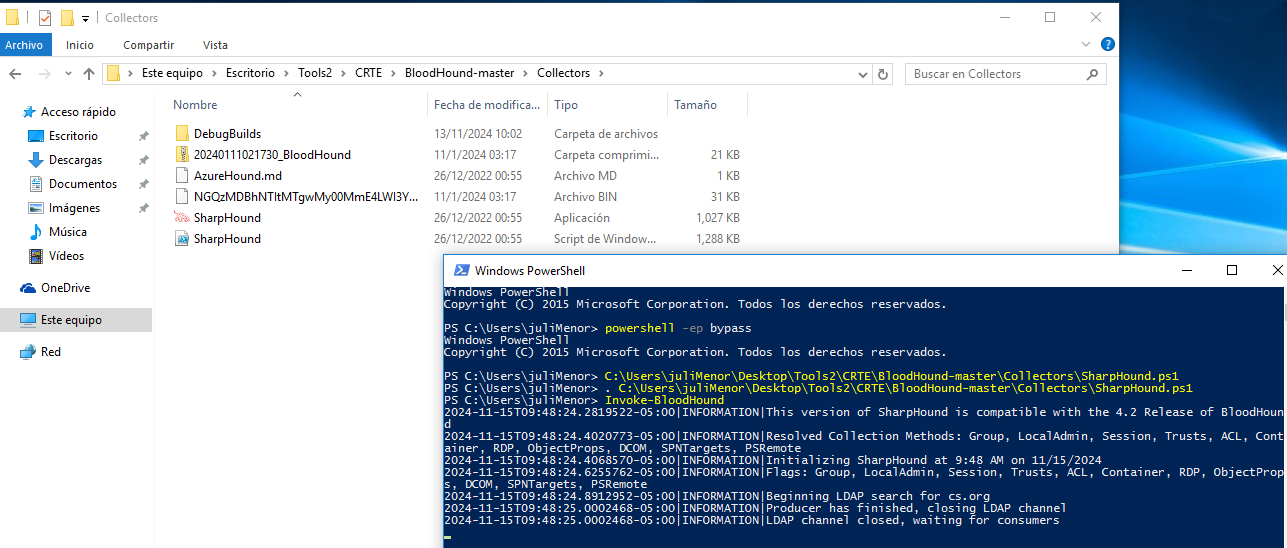
* Ya con la configuración anteriormente configurada vamos a instalar Bloodhund.

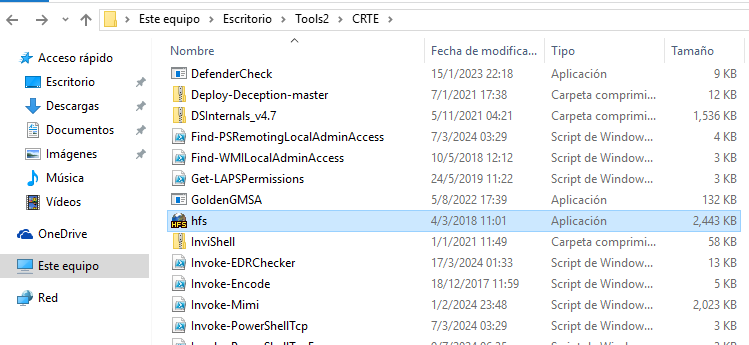


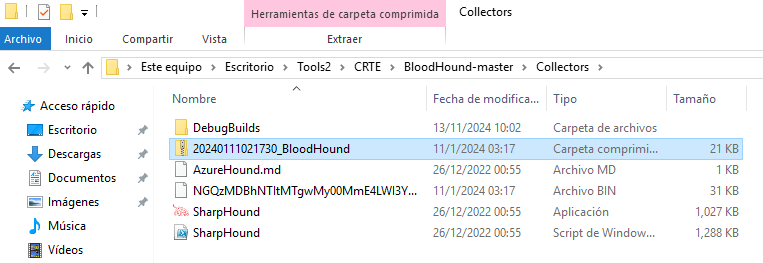
* Luego de tener Bloodhund correctamente instalado vamos a crear usuario y contraseña, luego vamos a iniciar la base de datos de bloodhund para poder ingresar a él.

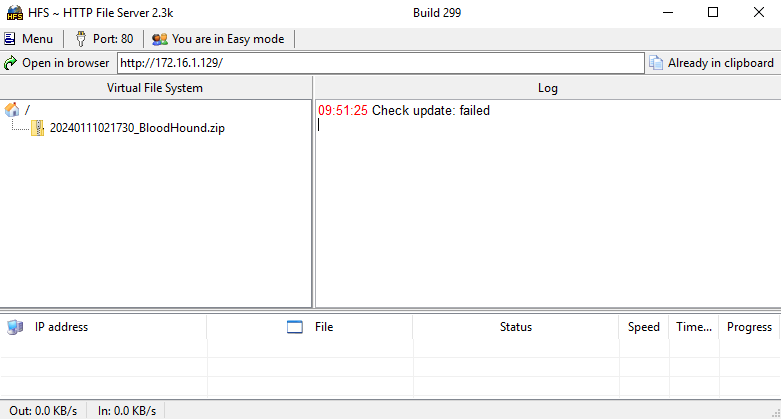


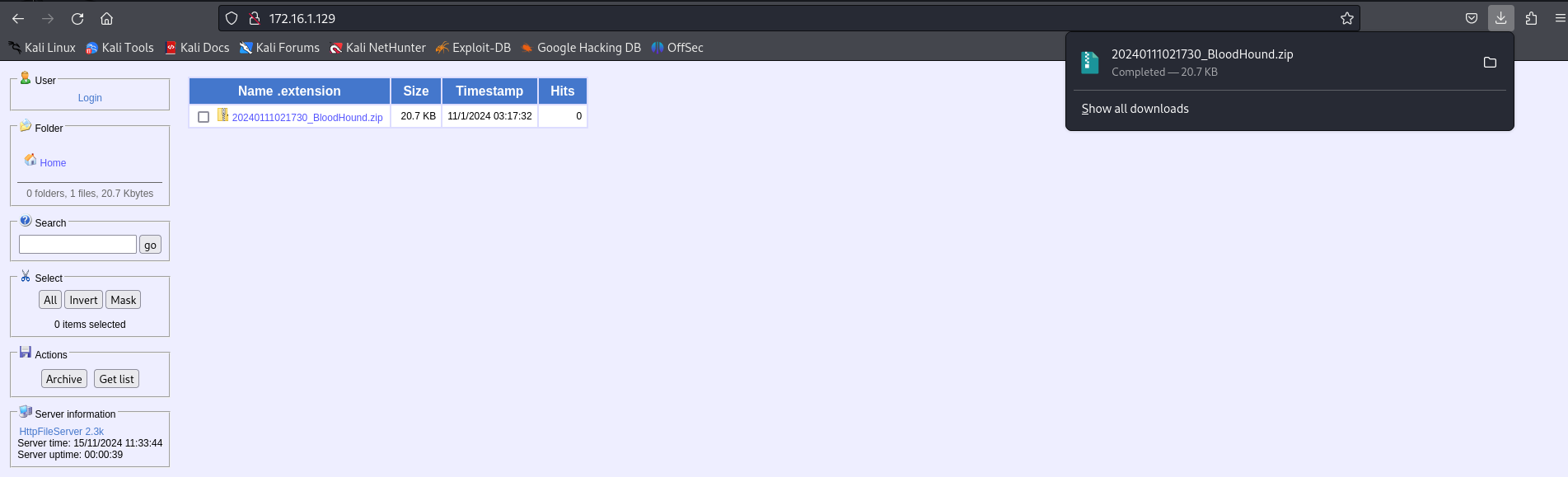
* Procedemos a ejecutar SharpHound

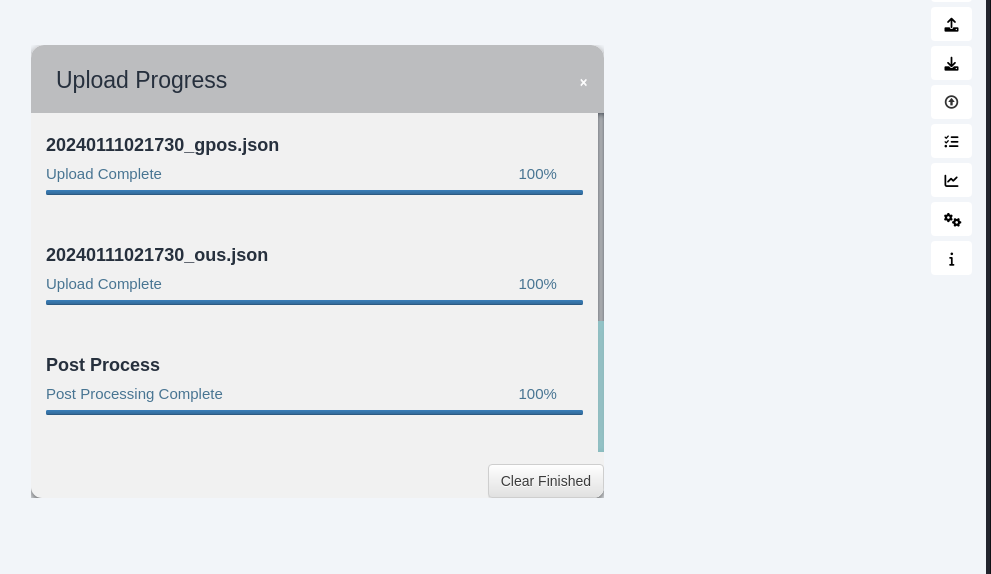


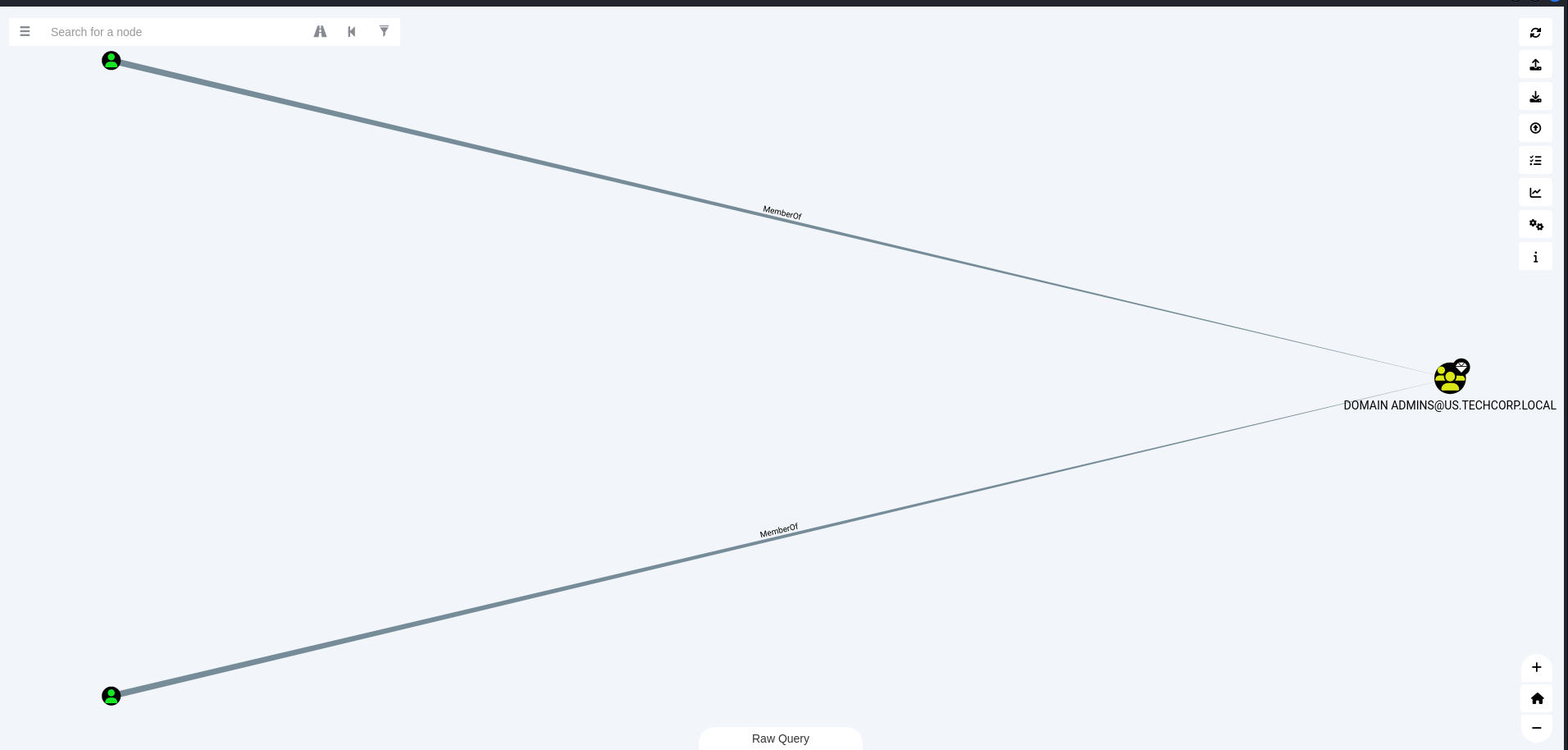


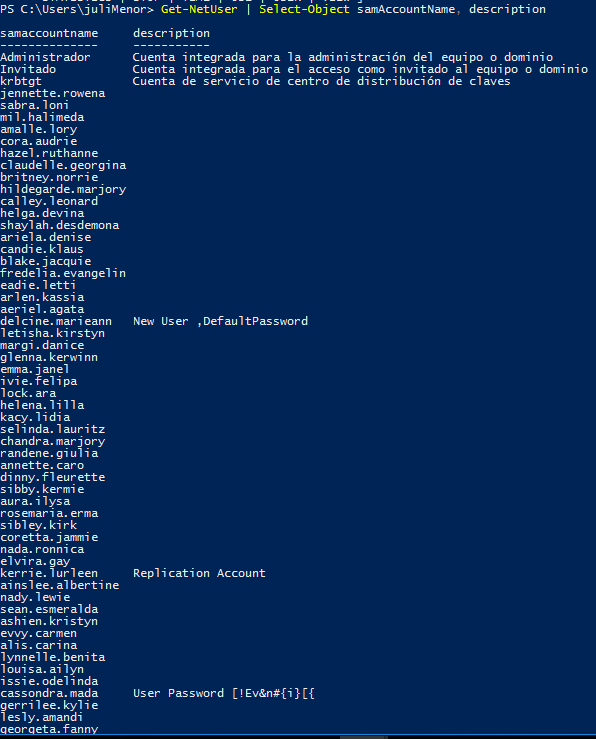


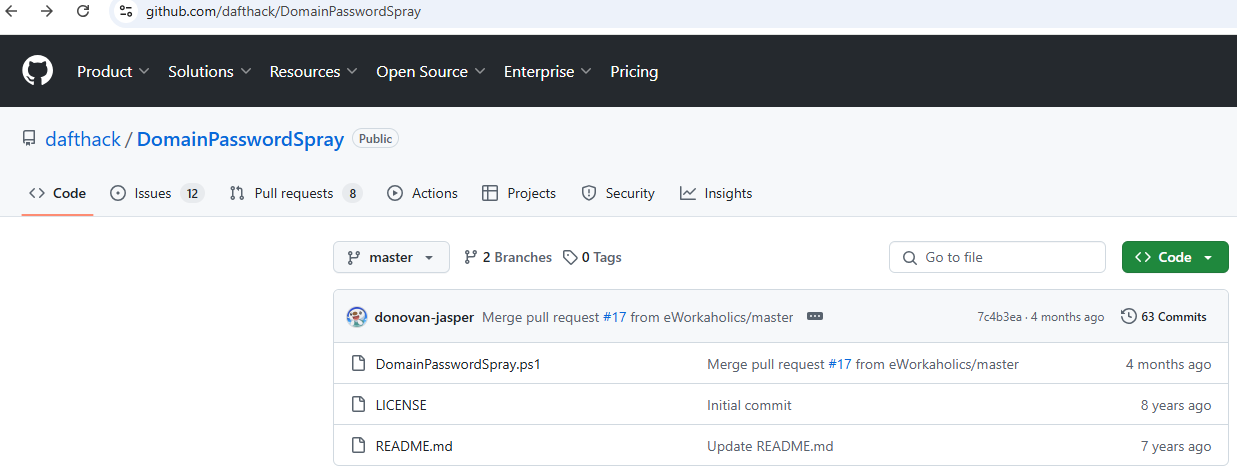


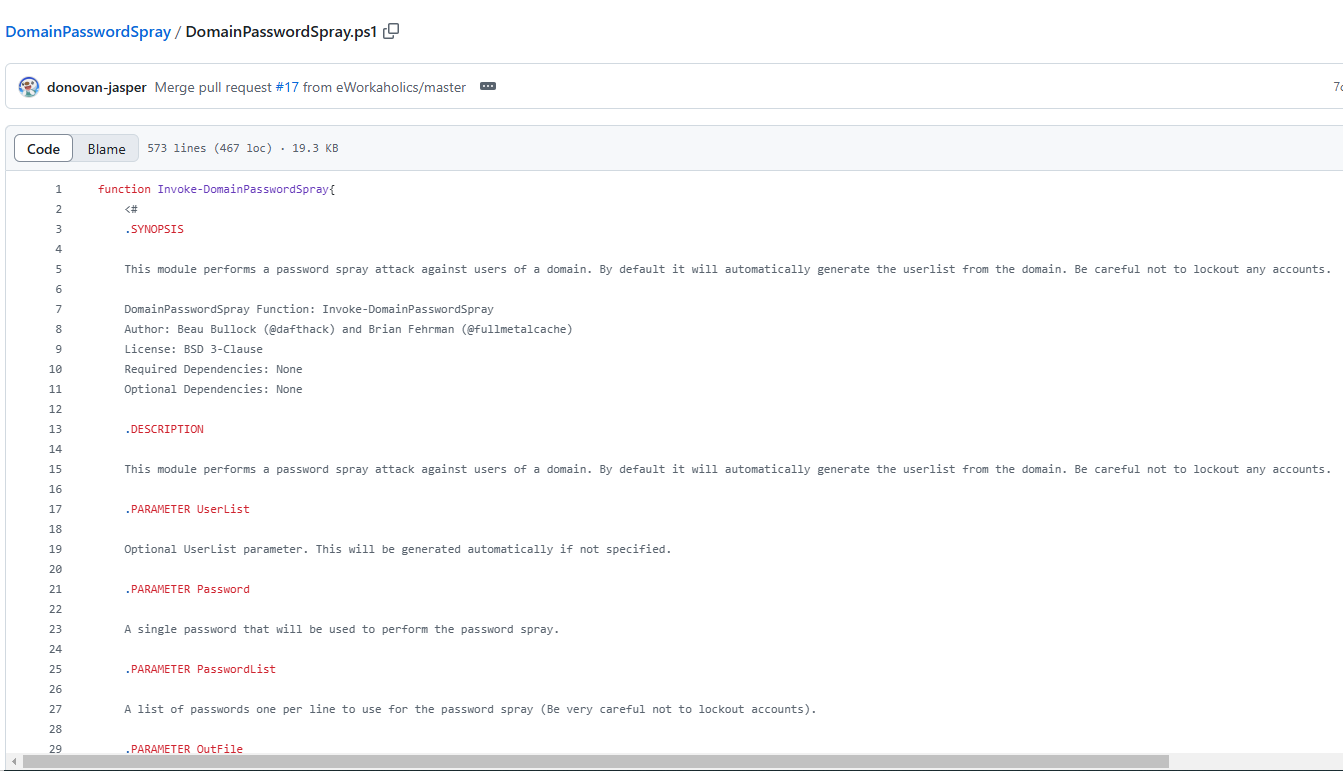


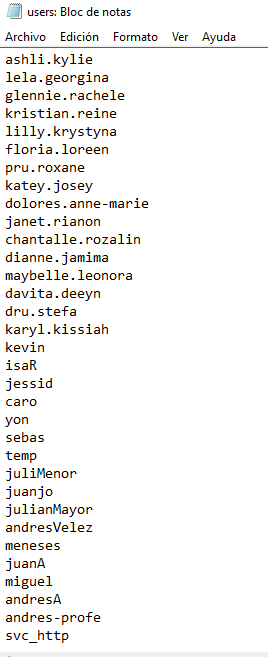


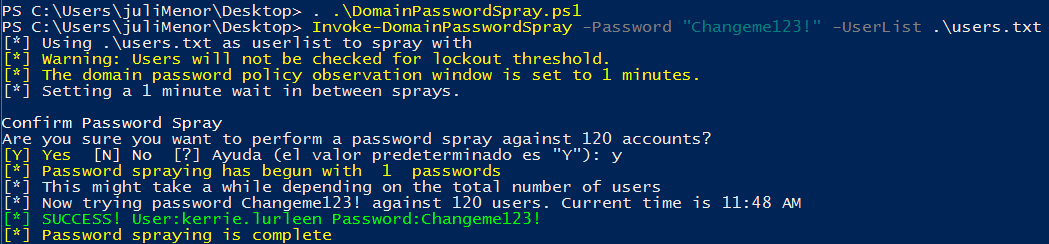




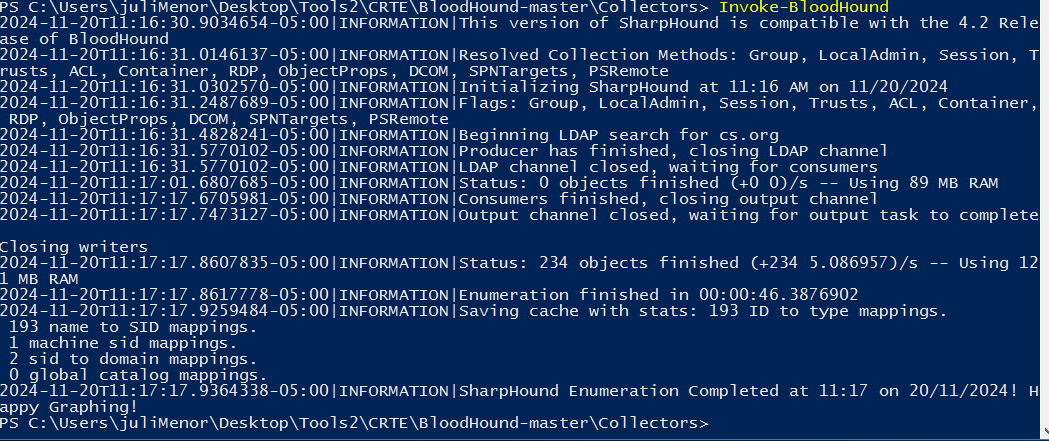


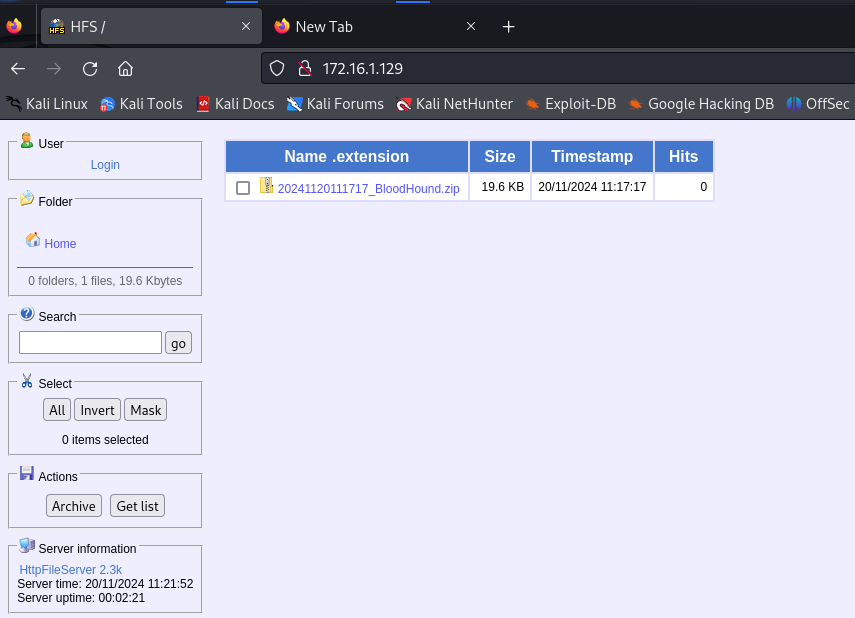


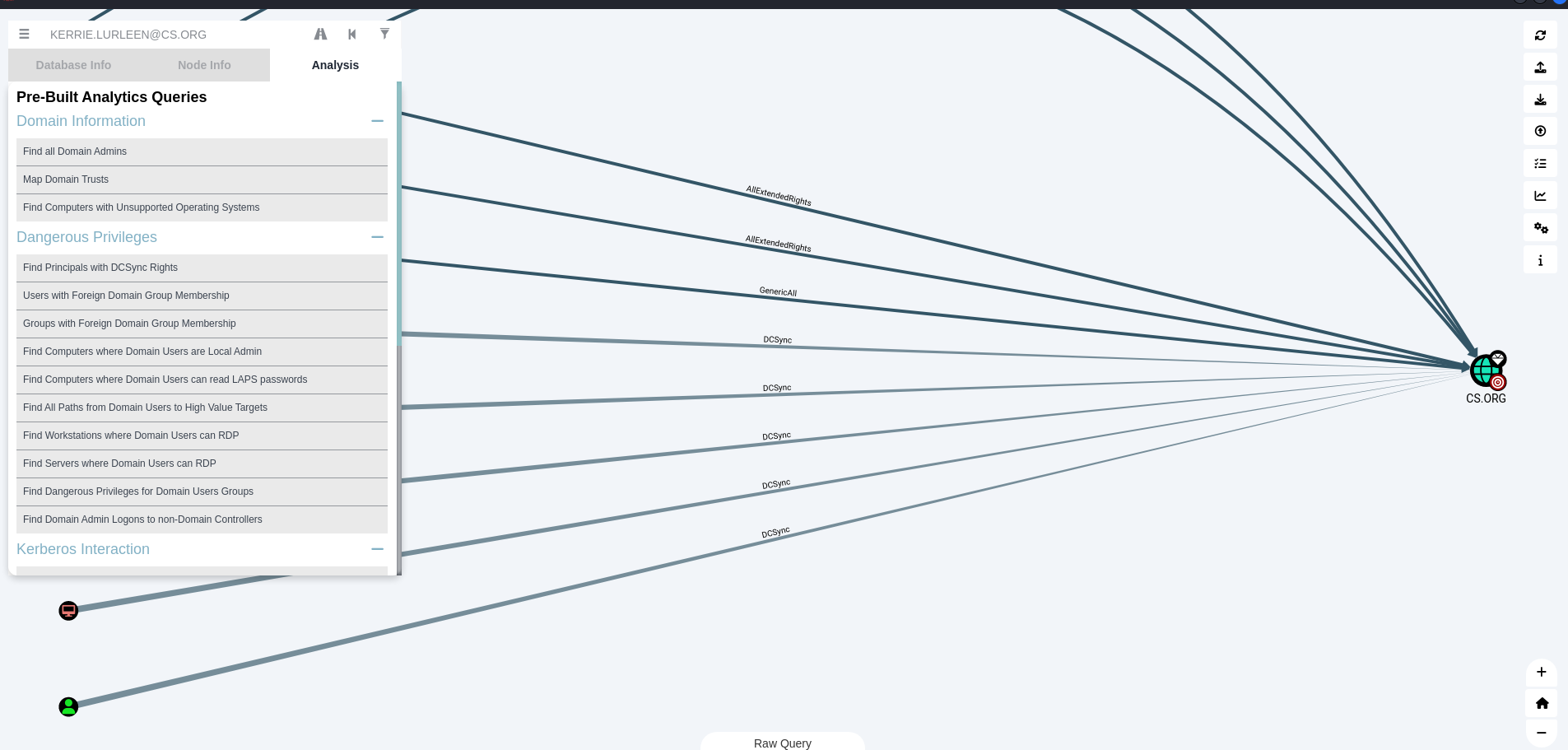




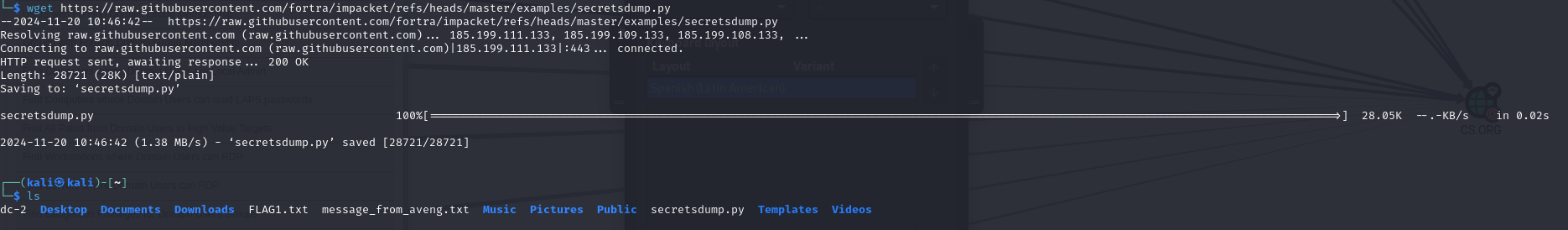
Montamos el directorio activo en bloodhound



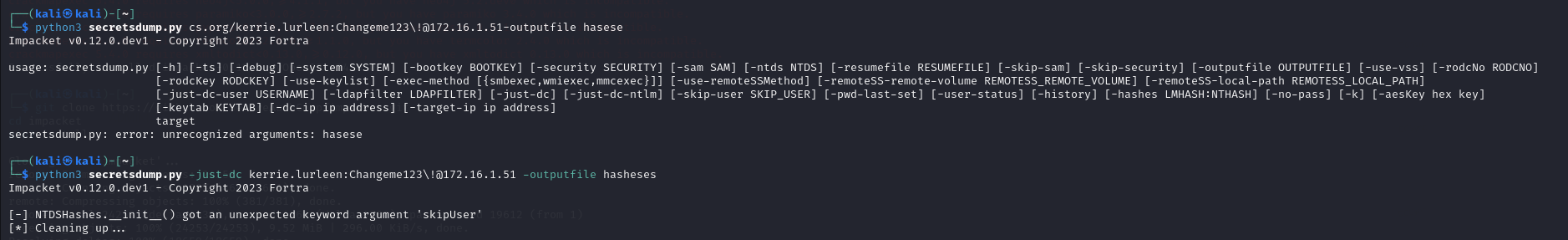








No funcionaba



**Actualizar Impacket:** Asegúrate de tener la versión más reciente de Impacket instalada. Para ello, ejecuta:

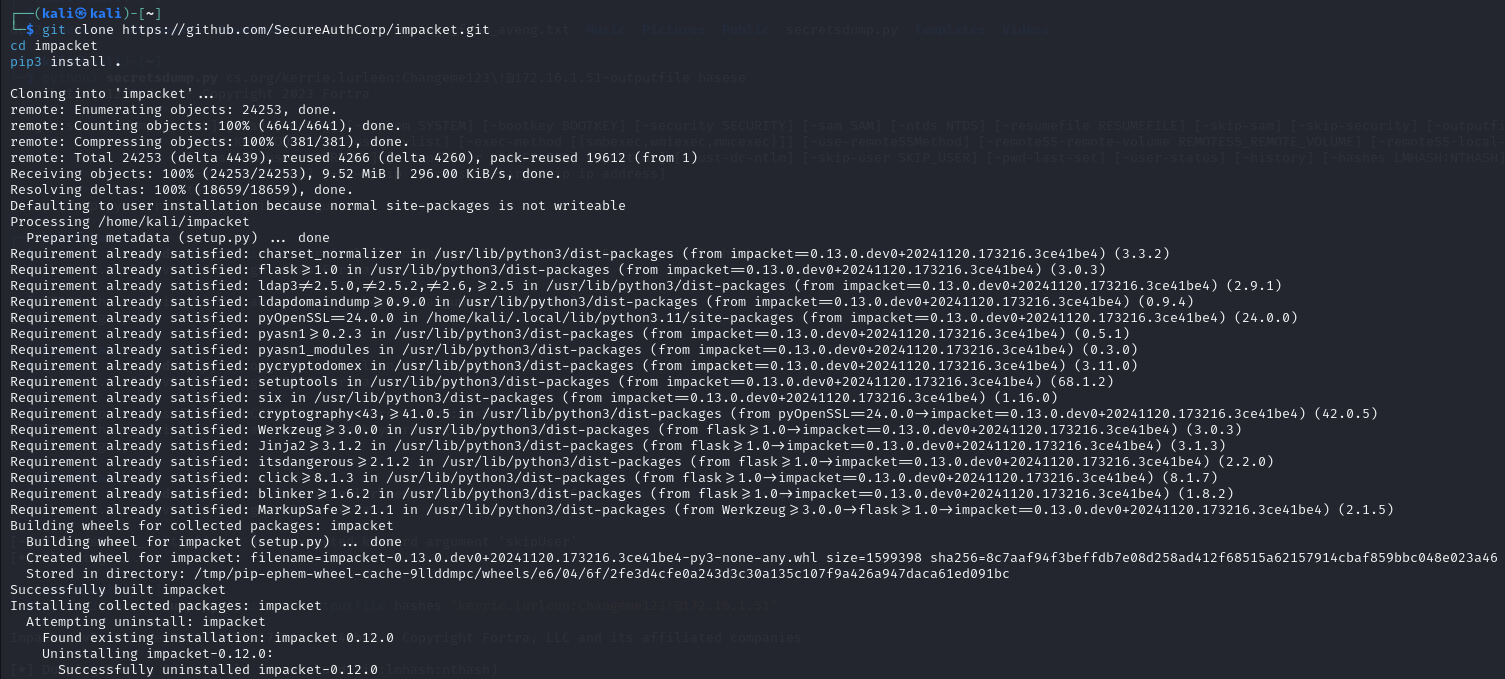


**Verifica la versión del script:** Si estás utilizando una copia modificada de secretsdump.py, asegúrate de que sea compatible con la versión de Impacket que tienes instalada. Descarga el script original desde el repositorio oficial si es necesario:

git clone https://github.com/SecureAuthCorp/impacket.git

cd impacket

pip3 install .

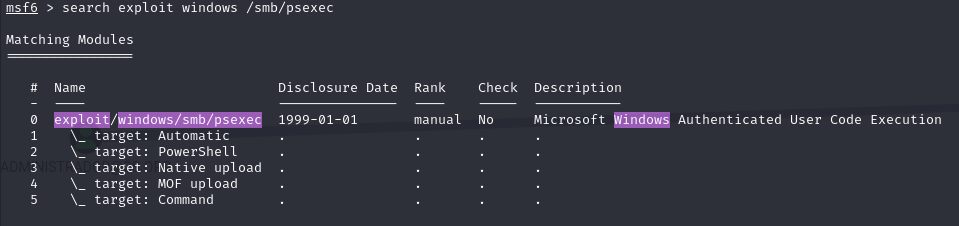


**Ejecuta el comando corregido:** Si el error se debe a un malentendido del argumento o formato, asegúrate de que el comando sea correcto. Por ejemplo:

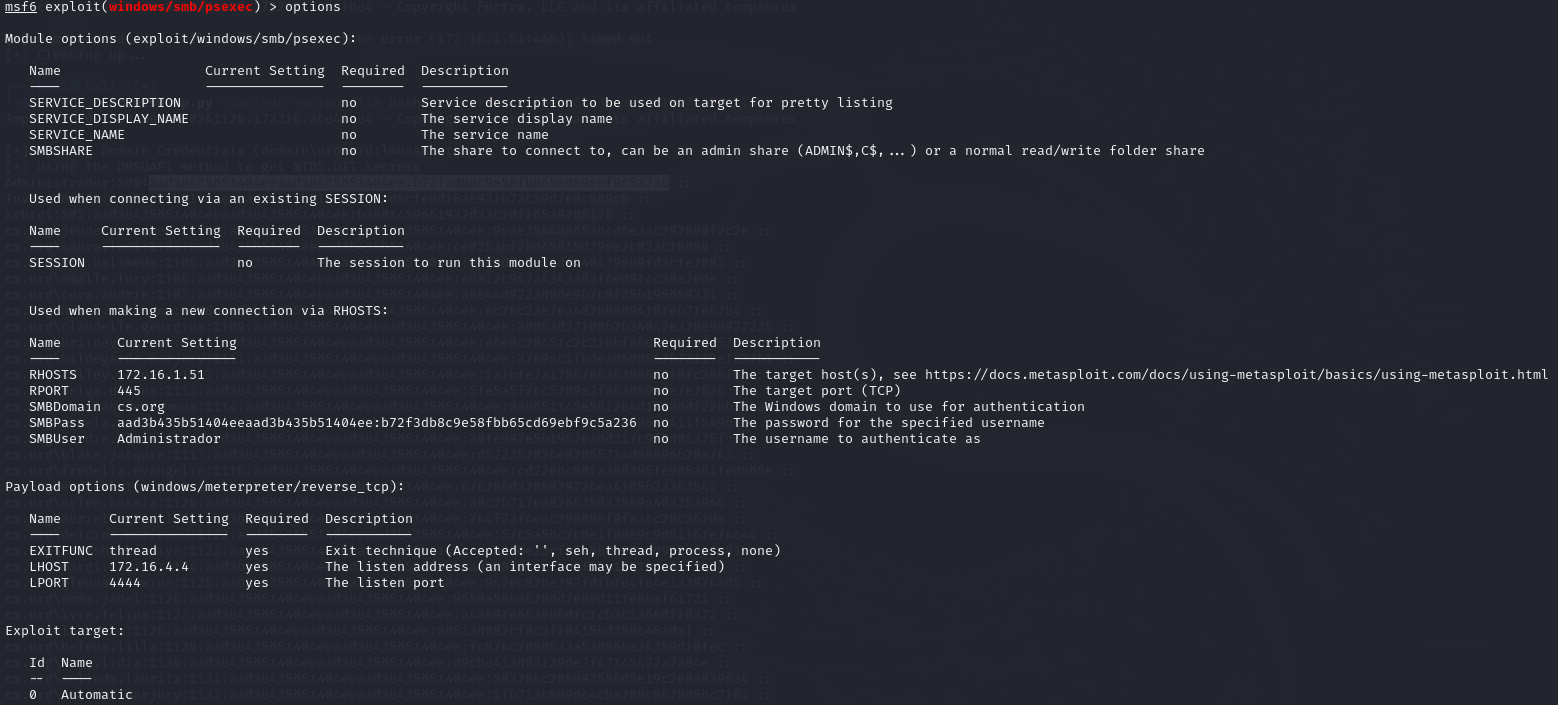
<python3 secretsdump.py -just-dc -outputfile hashes 'kerrie.lurleen:Changeme123!@172.16.1.51'>



Utilizamos el xploit 0

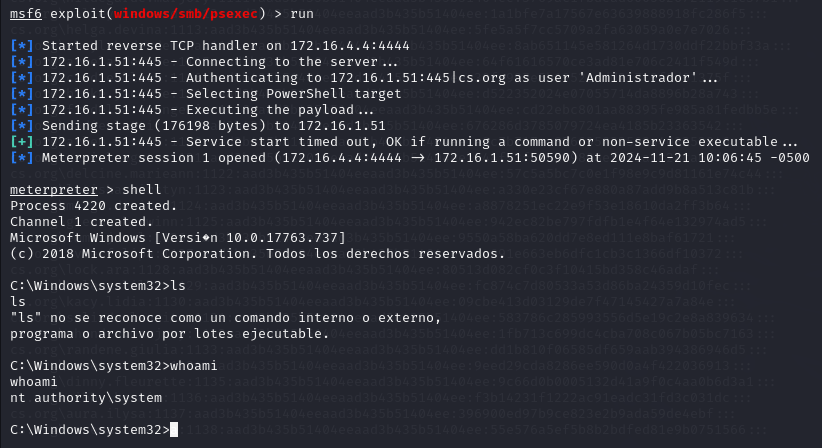


Le ponemos las opciones requeridas y el payload

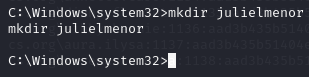




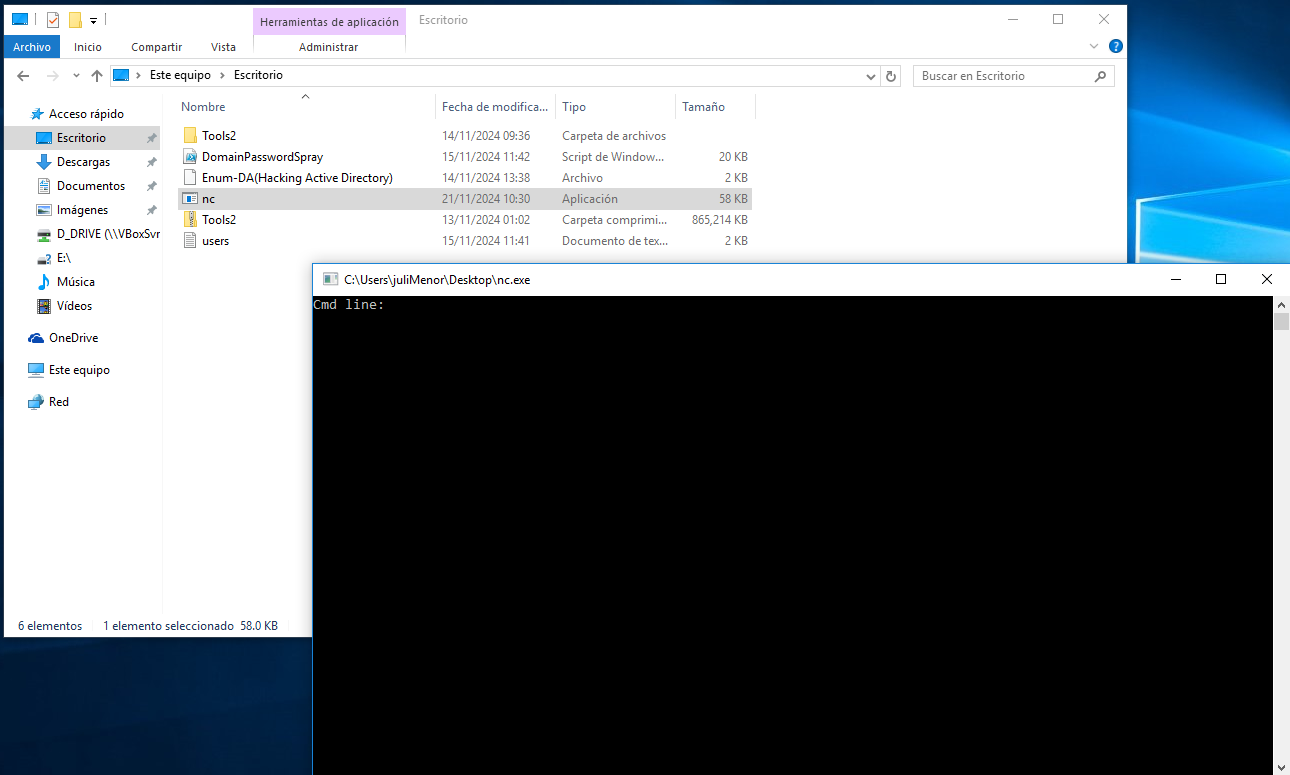
Corremos el exploit y vemos que ya estamos dentro del controlador de dominio



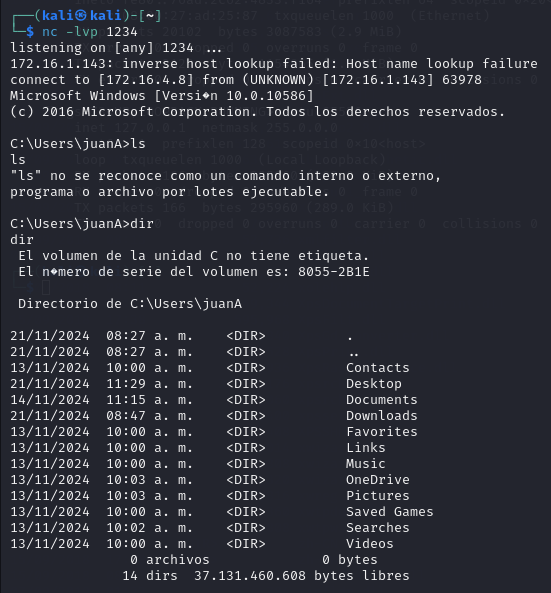
Creamos una carpeta



Descargamos el archivo netcat



* Vamos a conectarnos al equipo de un compañero



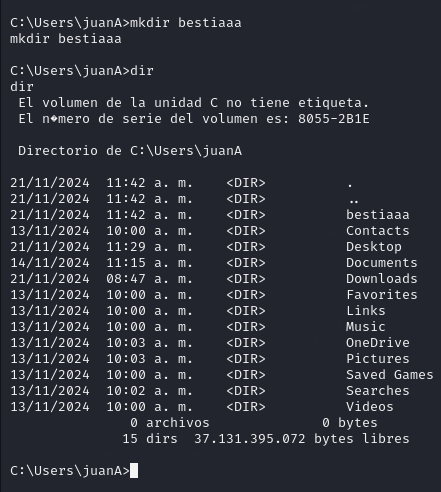


Foto juanan

