

Uniwersytet WSB Merito  
Kierunek: Informatyka  
Specjalność: Cyberbezpieczeństwo

Rok akademicki: 2024/2025  
semestr letni

## Projekt - 1

Wykonał: Maciej Niemiec

Numer albumu: 107162



## Wstęp

Projekt nr 1 realizowany w ramach laboratoriów z przedmiotu *Cyberbezpieczeństwo* miał na celu przećwiczenie pełnego cyklu rozpoznania i wstępnej analizy podatności w bezpiecznym, odizolowanym środowisku wirtualnym. Pracowaliśmy na dwóch maszynach uruchomionych w VMware Workstation w trybie sieci *host-only*: stacji ofensywnej **Kali Linux 2024.2** (kernel 6.8, IP 192.168.65.129) oraz podatnej maszyny **Metasploitable 2** (Ubuntu 8.04, IP 192.168.65.128). Zadaniem było wykrycie hostów w podsieci, enumeracja usług, identyfikacja wersji i błędnych konfiguracji oraz weryfikacja, czy uzyskane informacje umożliwiają nieautoryzowany dostęp lub późniejszą eskalację uprawnień.

Aby to osiągnąć, zastosowaliśmy zestaw popularnych narzędzi open-source:

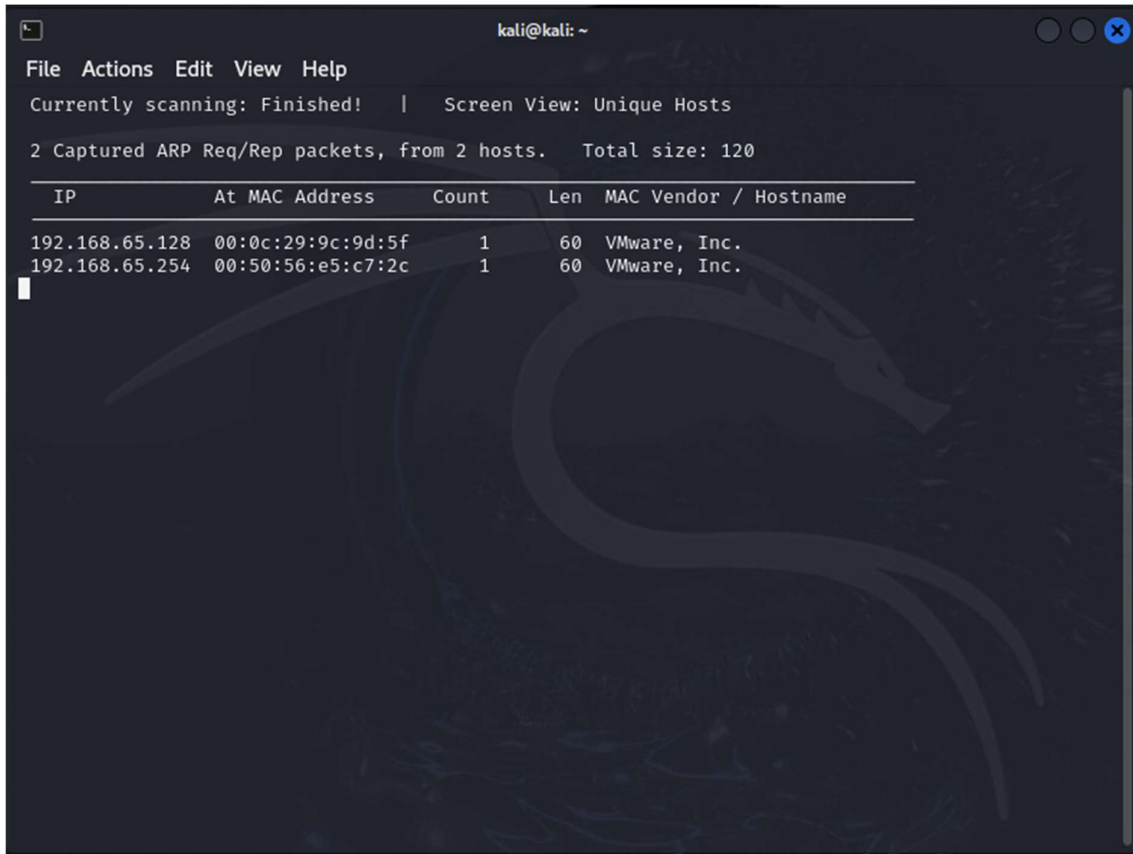
- **netdiscover**, **nmap** i **masscan** do skanów ARP, ping-skanów, szybkich testów *top-ports* oraz pełnych skanów wersji i skryptów NSE,
- **hping3** do ręcznego generowania pakietów ICMP/TCP/UDP i obserwacji wpływu konfiguracji interfejsu na zasięg ruchu,
- **smbmap** i skrypty nmapa (*smb-os-discovery*) do enumeracji udziałów i parametrów Samby,
- **Nikto** do testu nagłówek HTTP, opcji TRACE, phpMyAdmina i ujawnienia plików typu *phpinfo.php*,
- proste połączenia **telnet** oraz rekonesans za pomocą **unix-privesc-check** (tryb standard i detailed) po stronie Kali w celu oceny potencjalnych dróg eskalacji lokalnych.

Całość wykonywano wyłącznie w sieci laboratoriów, z zachowaniem zasad etyki i bez łączenia podatnej VM-ki z Internetem. Wyniki zostały opisane w dalszych częściach sprawozdania: od pierwszego wykrycia hostów, poprzez szczegółowe skany portów i usług, aż po wnioski dotyczące realnych ryzyk i rekomendacje hardeningu środowisk linuksowych.

# Cześć Projektowa

## 1. Information Gathering

### a. netdiscover -r 192.168.1.0/24

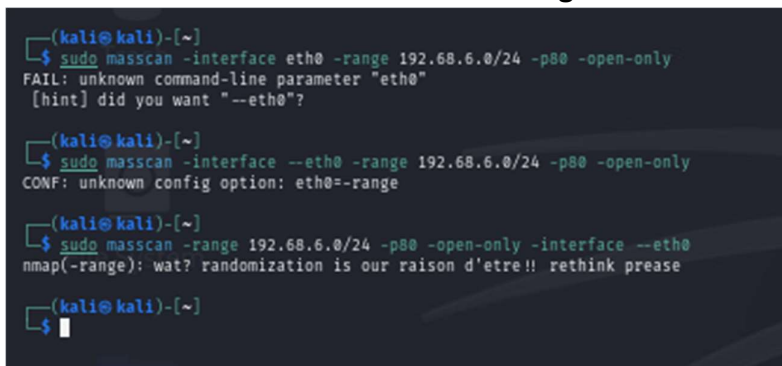


The screenshot shows a terminal window with the netdiscover tool running. The output indicates that scanning is finished and shows 2 captured ARP request/reply packets from two hosts. A table lists the discovered hosts with their IP addresses, MAC addresses, and vendor information.

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.65.128	00:0c:29:9c:9d:5f	1	60	VMware, Inc.
192.168.65.254	00:50:56:e5:c7:2c	1	60	VMware, Inc.

Na ekranie widać wynik pracy netdiscover, narzędzia do wykrywania „żywych” hostów w sieci lokalnej poprzez analizę ruchu ARP (pasywnie, gdy uruchomimy je bez parametrów, lub aktywnie przy opcji -r podającej zakres podsieci). Program zarejestrował tylko dwa pakiety ARP pochodzące od dwóch urządzeń: 192.168.65.128 i 192.168.65.254. Oba adresy MAC mają prefiks VMware, co wskazuje, że są to interfejsy wirtualne – pierwsze IP należy do maszyny wirtualnej, a drugie to prawdopodobnie brama sieci host-only/NAT udostępniana przez VMware. Krótko mówiąc, skan ujawnia obecność jedynie tych dwóch aktywnych hostów w segmencie 192.168.65.0/24.

### b. masscan -interface eth0 -range 192.168.1.0/24 -p80 -open-only



The screenshot shows a terminal window where the masscan command is being executed. The user attempts to run 'sudo masscan -interface eth0 -range 192.68.6.0/24 -p80 -open-only', which fails due to an unknown parameter 'eth0'. Subsequent attempts to use '--eth0' or '-interface --eth0' also fail with similar error messages.

```
(kali@kali)-[~]  
$ sudo masscan -interface eth0 -range 192.68.6.0/24 -p80 -open-only  
FAIL: unknown command-line parameter "eth0"  
[hint] did you want "--eth0"?  
  
(kali@kali)-[~]  
$ sudo masscan -interface --eth0 -range 192.68.6.0/24 -p80 -open-only  
CONF: unknown config option: eth0=-range  
  
(kali@kali)-[~]  
$ sudo masscan -range 192.68.6.0/24 -p80 -open-only -interface --eth0  
nmap(-range): wat? randomization is our raison d'etre!! rethink please  
  
(kali@kali)-[~]  
$
```

sudo podnosi uprawnienia do roota, bo masscan musi tworzyć pakiety RAW. masscan to ultraszybki skaner TCP. --interface eth0 (w przykładzie wpisane błędnie jako -interface) wskazuje, z którego interfejsu wysłać ruch; można to zrobić też krócej -e eth0. --range 192.68.6.0/24 (na ekranie -range) miał ograniczyć skan do tej podsieci, ale prawidłowo podajemy sam adres/CIDR jako argument albo zapis --range 192.68.6.0-192.68.6.255; pojedynczy myślnik sprawia, że masscan traktuje to jako nieznaną opcję, stąd komunikaty o błędzie. -p80 selekcjonuje port 80 (lista lub zakresy są dozwolone). --open-only filtruje wynik tak, by wyświetlać wyłącznie hosty, które odpowiedziały SYN-ACK-iem, czyli faktycznie mają port otwarty. Błędy widoczne na zrzutach ("unknown command-line parameter eth0", "unknown config option -range", "nmap(-range): wat?") wynikają z pomylenia pojedynczych i podwójnych kresek oraz z tego, że masscan domyślnie losuje kolejność pakietów i nie rozumie starego nmap-owego przełącznika -range; poprawne polecenie brzmiałoby na przykład:

```
sudo masscan 192.68.6.0/24 -p80 --open-only --interface eth0
```

### c. hping3

```
- hping3 -l 10.0.0.25
```

```
(kali@kali)-[~]
└─$ sudo hping3 -l 10.0.0.25
[sudo] password for kali:
Warning: Unable to guess the output interface
HPING 10.0.0.25 (lo 10.0.0.25): icmp mode set, 28 headers + 0 data bytes
[send_ip] sendto: Network is unreachable
```

Uruchomione polecenie sudo hping3 -l 10.0.0.25 próbuje wysłać pakiet ICMP Echo do adresu 10.0.0.25, ale na maszynie wirtualnej pracującej w trybie host-only nie ma żadnej trasy ani bramy prowadzącej do sieci 10.0.0.0/8. Hping3 podniesiony do roota potrafi tworzyć gniazda RAW, lecz kiedy patrzy w tablicę routingu, nie znajduje pasującej drogi, więc „zgaduje” interfejs loopback (lo) i wypisuje ostrzeżenie „Unable to guess the output interface”. Kernel odrzuca próbę wysłania, co kończy się komunikatem „sendto: Network is unreachable”. W laboratorium oznacza to, że karta host-only widzi wyłącznie adresy z wirtualnego segmentu (zwykle 192.168.65.0/24); wszystko spoza niego — jak 10.0.0.25 — jest nieosiągalne. Aby pakiet dotarł, trzeba by albo zmienić cel na adres z tej samej podsieci host-only, albo przełączyć kartę VM w tryb NAT/bridged albo dodać statyczną trasę lub ręcznie wskazać interfejs (-l eth0) po uprzednim włączeniu i skonfigurowaniu go.

```
- hping3 -A 10.0.0.25 -p 80
```

```
(kali@kali)-[~]
└─$ sudo hping3 -A 10.0.0.25 -p 80
Warning: Unable to guess the output interface
HPING 10.0.0.25 (lo 10.0.0.25): A set, 40 headers + 0 data bytes
[send_ip] sendto: Network is unreachable
```

sudo hping3 -A 10.0.0.25 -p 80 miał wysłać pojedynczy pakiet TCP z ustawioną flagą ACK (-A) na port 80 wskazanego hosta – klasyczny sposób sprawdzania, czy dany port odpowie bez pełnego handshake’u. Komenda wymaga roota (stąd sudo), ale – tak jak przy poprzednich próbach – hping3 nie potrafi dobrać interfejsu, bo w tablicy routingu brakuje wpisu prowadzącego do sieci 10.0.0.0/8, a domyślnej bramy również nie ma (tryb host-only widzi tylko lokalny segment VMware, typowo 192.168.65.0/24). Narzędzie wybiera więc loopback, wypisuje ostrzeżenie „Unable to guess the output interface”, a kernel kończy wysyłkę błędem „sendto: Network is unreachable”.

Innymi słowy, pakiet nigdy nie opuścił maszyny. Rozwiązaniem w warunkach laboratorium jest testowanie wyłącznie adresów z tej samej podsieci host-only lub przetączenie karty na NAT/bridged, ewentualnie dodanie statycznej trasy i wymuszenie interfejsu parametrem -I eth0 po wcześniejszym jego włączeniu i skonfigurowaniu.

```
- hping3 -2 10.0.0.25 -p 80
```

```
(kali㉿kali)-[~]  
$ sudo hping3 -2 10.0.0.25 -p 80  
Warning: Unable to guess the output interface  
HPING 10.0.0.25 (lo 10.0.0.25): udp mode set, 28 headers + 0 data bytes  
[send_ip] sendto: Network is unreachable
```

sudo hping3 -2 10.0.0.25 -p 80 uruchamia hping3 w trybie UDP (-2) i próbuje wysłać pusty datagram na port 80 hosta 10.0.0.25. Uprawnienia roota pozwalają założyć gniazdo RAW, ale program znów nie znajduje w tablicy routingu trasy do sieci 10.0.0.0/8 (ani bramy), dlatego zgłasza „Unable to guess the output interface”, domyślnie wybiera loopback i kończy się błędem kernela „sendto: Network is unreachable”. W środowisku labowym karta pracuje w trybie host-only, więc ruch wyjdzie tylko do podsieci VM-ware (zwykle 192.168.65.x); adres 10.0.0.25 leży poza zasięgiem. Żeby polecenie zadziałało, trzeba testować cel w tej samej sieci host-only albo włączyć inny tryb karty (NAT/bridged) czy dodać statyczną trasę i wskazać aktywny interfejs parametrem -I eth0 po jego wcześniejszej konfiguracji.

```
- hping3 -S 72.14.207.99 -p 80 --tcp-timestamp
```

```
(kali㉿kali)-[~]  
$ sudo hping3 -S 72.14.207.99 -p 80 --tcp-timestamp  
Warning: Unable to guess the output interface  
HPING 72.14.207.99 (lo 72.14.207.99): S set, 40 headers + 0 data bytes  
[send_ip] sendto: Network is unreachable
```

sudo hping3 -S 72.14.207.99 -p 80 --tcp-timestamp miało wysłać pojedynczy pakiet TCP z ustawioną flagą SYN do zewnętrznego adresu 72.14.207.99 na port 80, a przy okazji dodać opcję TCP-timestamp, która pozwala mierzyć dokładniejsze RTT i bywa używana do fingerprintingu systemu operacyjnego. Uprawnienia roota są potrzebne do tworzenia gniazd RAW, ale wirtualce działającej w trybie host-only kernel nie ma żadnej trasy prowadzącej ani do Internetu, ani nawet domyślnej bramy, więc hping3 nie potrafi dobrać właściwego interfejsu („Unable to guess the output interface”), sam wybiera pętlę loopback i kończy się błędem „sendto: Network is unreachable”. Tu wszystko rozbija się o izolację sieci host-only: widzi ona wyłącznie lokalny segment VMware (np. 192.168.65.x), a każdy adres spoza tej podsieci – jak 72.14.207.99 – pozostaje nieosiągalny. Aby test zadziałał, trzeba albo przetączyć kartę na NAT/bridged, albo dodać domyślną bramę/statyczną trasę oraz wymusić aktywny interfejs przetącznikiem -I eth0; ewentualnie ograniczyć się do celów znajdujących się w tej samej sieci host-only.

```
- hping3 -8 50-60 -S 10.0.0.25 -V
```

```

(kali@kali)-[~]
$ sudo hping3 -8 50-60 -S 10.0.0.25 -V
Warning: Unable to guess the output interface
using lo, addr: 127.0.0.1, MTU: 65536
Scanning 10.0.0.25 (10.0.0.25), port 50-60
11 ports to scan, use -V to see all the replies
+---+---+---+---+---+---+---+
|port| serv name | flags |ttl| id | win | len |
+---+---+---+---+---+---+---+
[send_ip] sendto: Network is unreachable

```

Polecenie `sudo hping3 -8 50-60 -S 10.0.0.25 -V` uruchamia `hping3` w trybie „scan” (-8) i ma sondować host 10.0.0.25 na zakresie portów 50-60, wysyłając dla każdego porta pakiet TCP SYN (-S); opcja -V włącza tryb -verbose, a `sudo` pozwala tworzyć gniazda RAW.

W maszynie wirtualnej działającej w trybie host-only tablica routingu widzi tylko wewnętrzny segment VMware (np. 192.168.65.x). Adres 10.0.0.25 leży poza tym zakresem, więc `hping3` nie znajduje pasującej trasy, zgłasza „Unable to guess the output interface”, wybiera loopback (lo) i kończy próby komunikatem kernela „sendto: Network is unreachable”. Efekt: żaden pakiet nie wychodzi, tabela wyników pozostaje pusta.

Aby skan mógł się powieść, trzeba (1) testować cel w tej samej podsieci host-only, lub (2) przełączyć kartę VM na NAT/bridged, czy (3) ręcznie dodać trasę/bramę oraz wymusić aktywny interfejs parametrem -I eth0 po wcześniejszej jego konfiguracji.

```
- hping3 -1 10.0.1.x --rand-dest -I eth0
```

```

(kali@kali)-[~]
$ sudo hping3 -1 10.0.1.x --rand-dest -I eth0
HPING 10.0.1.x (eth0 10.0.1.x): icmp mode set, 28 headers + 0 data bytes
^X^C
— 10.0.1.x hping statistic —
515 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms

```

Polecenie uruchomione z roota (`sudo`) wysyła surowe pakiety ICMP (-1, tryb ping) do adresu zapisanego jako 10.0.1.x; litera x w składni `hping3` oznacza, że ostatni oktet będzie podstawiany losowo, a dodatkowy przełącznik `--rand-dest` każe generować taki losowy adres przy każdym wysłanym pakiecie. Dzięki -I eth0 program wreszcie korzysta z konkretnego interfejsu, więc nie ma już błędu o nieznanym wyjściu; statystyka po przerwaniu (Ctrl-C) pokazuje, że w sumie nadano 515 ech, ale żadna odpowiedź nie wróciła (100 % utraty). W naszym labie karta pracuje w trybie host-only i widzi wyłącznie wirtualną podsieć (np. 192.168.65.0/24), natomiast żaden host z zakresu 10.0.1.0/24 tam nie istnieje, więc próby ARP pozostają bez echa, pakiety trafiają w próżnię i brak jest jakichkolwiek odpowiedzi ICMP.

```
- hping3 -c 3 10.10.10.10
```

```

(kali@kali)-[~]
$ sudo hping3 -c 3 10.10.10.10
Warning: Unable to guess the output interface
HPING 10.10.10.10 (lo 10.10.10.10): NO FLAGS are set, 40 headers + 0 data bytes
[send_ip] sendto: Network is unreachable

```

`sudo hping3 -c 3 10.10.10.10` miało wysłać trzy (-c 3) surowe pakiety TCP bez żadnych flag do hosta 10.10.10.10. Uprawnienia roota pozwalają `hping3` tworzyć gniazda RAW, ale maszyna

#### d. smbmap

```
- smbmap -H 192.168.1.102
```

Polecenie `smbmap -H 192.168.1.102` uruchamia SMBMap w trybie „jednego hosta” (`-H` wskazuje adres IP docelowej maszyny) i próbuje sprawdzić, czy na tym adresie działają usługi SMB (port 139/445), a następnie – jeśli porty są otwarte – wyliczyć dostępne udziały z uprawnieniami użytkownika anonimowego. Wynik pokazał komunikat „Detected 0 hosts serving SMB”, co oznacza, że narzędzie w ogóle nie znalazło otwartych portów SMB na wskazanym IP. W naszym labie karta sieciowa VM-ki pracuje w trybie `host-only`, czyli widzi wyłącznie podsieć VMware (np. 192.168.65.0/24). Adres 192.168.1.102 leży w zupełnie innej sieci, więc próby połączenia nie wychodzą poza maszynę i skaner konkluduje brak hosta. Aby taka enumeracja zadziałała, trzeba byłoby: (1) przełączyć interfejs na `NAT` albo `bridged`, (2) dodać odpowiednią trasę statyczną, lub (3) testować cel z tej samej podsieci `host-only`.

```
- smbmap -H 192.168.1.102 -r tmp
```



```
(kali㉿kali)-[~]
$ smbmap -H 192.168.1.102 -r tmp

SMBMap - Samba Share Enumerator v1.10.4 | Shawn Evans - ShawnDEvans@gmail.com<mailto:ShawnDEvans@gmail.com>
https://github.com/ShawnDEvans/smbmap

[\\] Checking for open ports ...
[*] Detected 0 hosts serving SMB
[!] Authenticating ...
[/] Enumerating shares ...
[-] Closing connections..
[\\] Closing connections..
[!] Closing connections..
[/] Closing connections..
[-] Closing connections..

[*] Closed 0 connections
```

Polecenie `smbmap -H 192.168.1.102 -r tmp` próbuje najpierw sprawdzić, czy na hoście 192.168.1.102 działa usługa SMB, a jeśli tak – od razu zautoryzować się anonimowo i rekursywnie wylistować zawartość katalogu tmp w jednym z udziałów; parametr `-r` odwołuje się więc do konkretnej ścieżki w udziale (podobnie jak `cd+ls`). Narzędzie jednak znów informuje „Detected 0 hosts serving SMB”, bo z naszej maszyny wirtualnej w trybie host-only nie widać sieci 192.168.1.0/24 – karta ma dostęp wyłącznie do wirtualnego segmentu VMware (np. 192.168.65.x); pakiety do 192.168.1.102 nie wychodzą, więc SMBMap kończy enumerację bez jakichkolwiek połączeń. Aby przetestować ten host, trzeba byłoby przetoczyć interfejs na NAT lub bridged, ewentualnie dodać trasę do tej sieci albo skanować cel znajdujący się w tej samej podsieci host-only.

```
- smbmap -u msfadmin -p msfadmin -H 192.168.1.102
```



```
(kali@kali)-[~]
$ smbmap -u msfadmin -p msfadmin -H 192.168.1.102

SMBMap - Samba Share Enumerator v1.10.4 | Shawn Evans - ShawnDEvans@gmail.com<mailto:ShawnDEvans@gmail.com>
https://github.com/ShawnDEvans/smbmap

[\\] Checking for open ports...
[*] Detected 0 hosts serving SMB
[!] Authenticating...
[/] Authenticating...
[-] Closing connections..
[\\] Closing connections..
[!] Closing connections..
[/] Closing connections..
[-] Closing connections..

[*] Closed 0 connections
```

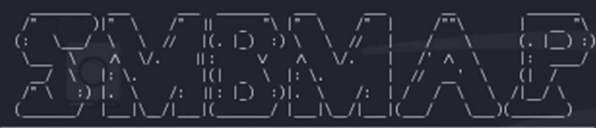
smbmap -u msfadmin -p msfadmin -H 192.168.1.102 uruchamia SMBMap z podanymi danymi logowania (-u użytkownik, -p hasło) i próbuje nawiązać sesję SMB z hostem 192.168.1.102 (-H). Narzędzie najpierw skanuje porty 139 i 445, ale wypisuje „Detected 0 hosts serving SMB”, więc nie widzi tam żadnej usługi; logowanie nie dochodzi do skutku i skrypt zamyka „0 connections”. W naszym labie karta działa w trybie host-only, który obejmuje wyłącznie wewnętrzny segment VMware (zwykle 192.168.65.x). Adres 192.168.1.102 leży poza tą siecią, więc pakiety nie wychodzą z maszyny i SMBMap nigdy nie dociera do celu. Żeby test zadziałał, trzeba użyć hosta z tej samej podsieci host-only albo przetoczyć adapter na NAT/bridged czy dodać odpowiednią trasę. Zatem wykonana została weryfikacja adresu IP.

```
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifocnfig
-bash: ifocnfig: command not found
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:dd:20:87
          inet addr:192.168.65.128  Bcast:192.168.65.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fedd:2087/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:28 errors:0 dropped:0 overruns:0 frame:0
          TX packets:47 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3078 (3.0 KB)  TX bytes:5066 (4.9 KB)
          Interrupt:17 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:92 errors:0 dropped:0 overruns:0 frame:0
          TX packets:92 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:19393 (18.9 KB)  TX bytes:19393 (18.9 KB)

msfadmin@metasploitable:~$
```

```
(kali@kali)-[~]
$ smbmap -u msfadmin -p msfadmin -H 192.168.65.128
```



```
SMBMap - Samba Share Enumerator v1.10.4 | Shawn Evans - ShawnDEvans@gmail.com<mailto:ShawnDEvans@gmail.com>
https://github.com/ShawnDEvans/smbmap

[*] Detected 1 hosts serving SMB
[*] Established 1 SMB connections(s) and 1 authenticated session(s)

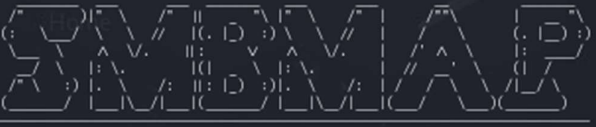
[+] IP: 192.168.65.128:445      Name: 192.168.65.128      Status: Authenticated
Disk                          Permissions             Comment
-----
print$                        READ ONLY               Printer Drivers
tmp                           READ, WRITE             oh noes!
opt                           READ ONLY
IPC$                          NO ACCESS               IPC Service (metasploitable server (Samba 3.0.20-Debian))
ADMIN$                       NO ACCESS               IPC Service (metasploitable server (Samba 3.0.20-Debian))
msfadmin                     READ, WRITE             Home Directories

[*] Closed 1 connections
```

Polecenie `smbmap -u msfadmin -p msfadmin -H 192.168.65.128` odpytuje serwer SMB działający na adresie 192.168.65.128 (maszyna Metasploitable w tej samej podsieci host-only). Przetączniki `-u` i `-p` przekazują znane testowe poświadczenia, a `-H` wskazuje docelowy host. SMBMap najpierw sprawdza porty 139/445, wykrywa usługę SMB, nawiązuje jedną sesję i uwierzytelnia się jako „msfadmin”. W rezultacie wyświetla listę udziałów wraz z uprawnieniami: `print$` (tylko odczyt, sterowniki drukarek), `tmp` (pełny zapis i odczyt – potencjalne miejsce na wrzucenie pliku wykonywalnego lub web-shell’a), `opt` (read-only), wewnętrzne udziały systemowe `IPC$` i `ADMIN$` są zamknięte, a udział osobisty `msfadmin` jest dostępny z prawem zapisu. Informacja w kolumnie „Comment” zdradza, że to Samba 3.0.20 z Debiana, znana z kilku podatności RCE. Podsumowując: dzięki prawidłowym danym logowania oraz temu, że host znajduje się w tej samej sieci host-only, narzędzie poprawnie wylistowało zasoby SMB i wskazało miejsce, gdzie napastnik mógłby bez przeszkód zapisywać pliki.

– `smbmap -u msfadmin -p msfadmin -H 192.168.1.102 -r tmp`

```
(kali@kali)-[~]
$ smbmap -u msfadmin -p msfadmin -H 192.168.65.128 -r tmp
```



```
SMBMap - Samba Share Enumerator v1.10.4 | Shawn Evans - ShawnDEvans@gmail.com<mailto:ShawnDEvans@gmail.com>
https://github.com/ShawnDEvans/smbmap

[*] Detected 1 hosts serving SMB
[*] Established 1 SMB connections(s) and 1 authenticated session(s)

[+] IP: 192.168.65.128:445      Name: 192.168.65.128      Status: Authenticated
Disk                          Permissions             Comment
-----
print$                        READ ONLY               Printer Drivers
tmp                           READ, WRITE             oh noes!
./tmp
dr--r--r--                  0 Fri Jun 20 04:47:18 2025  .
dw--w--w--                  0 Sun May 20 14:36:11 2012  ..
fw--w--w--                  0 Fri Jun 20 04:45:18 2025  5162.jsvc_up
dr--r--r--                  0 Fri Jun 20 04:44:55 2025  .ICE-unix
dr--r--r--                  0 Fri Jun 20 04:45:08 2025  .X11-unix
fw--w--w--                  11 Fri Jun 20 04:45:08 2025  .X0-lock
opt                           READ ONLY
IPC$                          NO ACCESS               IPC Service (metasploitable server (Samba 3.0.20-Debian))
ADMIN$                       NO ACCESS               IPC Service (metasploitable server (Samba 3.0.20-Debian))
msfadmin                     READ, WRITE             Home Directories

[*] Closed 1 connections
```

Polecenie `smbmap -u msfadmin -p msfadmin -H 192.168.65.128 -r tmp` testuje usługę Samba na maszynie Metasploitable (192.168.65.128) znajdującej się w tej samej podsieci host-only; parametr `-u/-p` przekazuje domyślne dane logowania, `-H` wskazuje adres hosta, a `-r tmp` każe od razu wejść do udziału `tmp` i rekursywnie wylistować jego zawartość. Wynik potwierdza, że na



```

--(kali@kali):~$
--$ nikto -h 192.168.65.128
-- Nikto v2.5.0

+ Target IP:      192.168.65.128
+ Target Hostname: 192.168.65.128
+ Target Port:    80
+ Start Time:     2025-06-20 04:48:25 (GMT+4)

+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ /: Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.18.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /index: Uncommon header 'csc' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.php. See: http://www.wisec.it/sectou.php?id=4698ebdc39d15,https://exchange.vforce.limcloud.com/vulnerabilities/6275
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /phpinfo.php: Output from the phpinfo() function was found.
+ /doc/: Directory indexing found.
+ /doc/: The /doc/ directory is browsable. This may be /usr/doc. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0678
+ /phpMyAdmin/2AB-3C92-11d3-A349-C780C10B0000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /phpMyAdmin/3-0428-11d2-A769-8BAAB1AC7421: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /phpMyAdmin/3-0428-11d2-A769-8BAAB1AC7421: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /phpMyAdmin/3-0428-11d2-A769-8BAAB1AC7421: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /phpMyAdmin/changelog.php: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ /phpMyAdmin/changelog: Server may leak inodes via tTags, header found with file /phpMyAdmin/changelog, inode: 92462, size: 48540, mtime: Tue Dec 9 12:24:00 2008. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ /phpMyAdmin/changelog: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ /test/: This might be interesting.
+ /test/: Directory indexing found.
+ /phpinfo.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of system information. See: CVE-552
+ /icons/: Directory indexing found.
+ /icons/MANIFEST: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /phpMyAdmin/ : phpMyAdmin directory found.
+ /phpMyAdmin/Documentation.html: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ /phpMyAdmin/MANIFEST: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts. See: https://typo3.org/
+ /wp-config.php: wp-config.php file found. This file contains the credentials.
+ 0910 requests: 0 error(s) and 27 item(s) reported on remote host
+ End Time:     2025-06-20 04:48:49 (GMT+4) (24 seconds)

+ 1 host(s) tested

```

Nikto (nikto -h 192.168.65.128) przeskanował serwer WWW działający na porcie 80 maszyny Metasploitable w naszej sieci host-only. Raport ujawnia, że host serwuje przestarzały Apache 2.2.8 (Ubuntu) z PHP 5.2.4 oraz brak mu nagłówków bezpieczeństwa X-Frame-Options i X-Content-Type-Options. Włączony moduł MultiViews umożliwia brute-force'owe odgadywanie nazw plików, a metoda HTTP TRACE pozostaje aktywna (podatność XST). Serwer zwraca listę plików dla katalogów /doc/, /icons/ i /test/, a także pozwala pobrać phpinfo.php, co odśłania pełną konfigurację PHP. Ponadto odkryto kilka ścieżek do phpMyAdmin (np. /phpMyAdmin/ i changelog), które są publicznie dostępne i mogą zdradzać wersję oraz umożliwiać atakującemu zarządzanie bazą MySQL. Łącznie te odkrycia potwierdzają: stara wersja oprogramowania, brak podstawowych nagłówków hardeningu, możliwość enumeracji plików i ujawnione narzędzia administracyjne sprawiają, że usługa HTTP na 192.168.65.128 stanowi łatwy cel w środowisku laboratoryjnym.

– nikto -h 192.168.1.102:80

```

--(kali@kali):~$
--$ nikto -h 192.168.65.128:80
-- Nikto v2.5.0

+ Target IP:      192.168.65.128
+ Target Hostname: 192.168.65.128
+ Target Port:    80
+ Start Time:     2025-06-20 04:50:00 (GMT+4)

+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ /: Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.18.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /index: Uncommon header 'csc' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.php. See: http://www.wisec.it/sectou.php?id=4698ebdc39d15,https://exchange.vforce.limcloud.com/vulnerabilities/6275
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /phpinfo.php: Output from the phpinfo() function was found.
+ /doc/: Directory indexing found.
+ /doc/: The /doc/ directory is browsable. This may be /usr/doc. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0678
+ /phpMyAdmin/2AB-3C92-11d3-A349-C780C10B0000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /phpMyAdmin/3-0428-11d2-A769-8BAAB1AC7421: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /phpMyAdmin/3-0428-11d2-A769-8BAAB1AC7421: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /phpMyAdmin/3-0428-11d2-A769-8BAAB1AC7421: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings. See: OSVDB-12184
+ /phpMyAdmin/changelog.php: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ /phpMyAdmin/changelog: Server may leak inodes via tTags, header found with file /phpMyAdmin/changelog, inode: 92462, size: 48540, mtime: Tue Dec 9 12:24:00 2008. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ /phpMyAdmin/changelog: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ /test/: This might be interesting.
+ /test/: Directory indexing found.
+ /phpinfo.php: PHP is installed, and a test script which runs phpinfo() was found. This gives a lot of system information. See: CVE-552
+ /icons/: Directory indexing found.
+ /icons/MANIFEST: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /phpMyAdmin/ : phpMyAdmin directory found.
+ /phpMyAdmin/Documentation.html: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ /phpMyAdmin/MANIFEST: phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts. See: https://typo3.org/
+ /wp-config.php: wp-config.php file found. This file contains the credentials.
+ 0910 requests: 0 error(s) and 27 item(s) reported on remote host
+ End Time:     2025-06-20 04:50:11 (GMT+4) (23 seconds)

+ 1 host(s) tested

```

Polecenie nikto -h 192.168.65.128:80 przeskanowało usługę HTTP działającą na maszynie Metasploitable w naszej sieci host-only. Wyniki:

- **Serwer:** Apache 2.2.8 (Ubuntu) z PHP 5.2.4 – obie wersje dawno niewspierane.
- **Nagłówki bezpieczeństwa:** brak X-Frame-Options (click-jacking) i X-Content-Type-Options (MIME sniffing).
- **Konfiguracja Apache:** aktywny moduł *MultiViews* pozwala zgadywać nazwy plików, a metoda HTTP TRACE jest włączona (podatność XST).
- **Ujawnione pliki/katalogi:** serwer listuje zawartość katalogów takich jak /doc/, /icons/, /test/; dostępny jest phpinfo.php, który odśłania pełną konfigurację PHP.

- **phpMyAdmin:** publicznie dostępne ścieżki /phpMyAdmin/ i /phpmyadmin/, wraz z historią zmian, co umożliwia atakującemu administrację bazą MySQL lub przynajmniej fingerprinting wersji.
- **Błąd konfiguracji mysqld.sock:** Nikto wskazuje obecność plików sock w /var/run – standardowa oznaka nieograniczonej ekspozycji bazy.
- **Testowe skrypty:** wykryto plik phpinfo.php oraz przykładowy test w /test/ – oba mogą służyć do dalszych ataków (np. RCE).

Serwer WWW na 192.168.65.128 jest skonfigurowany w sposób skrajnie niebezpieczny: stare wersje oprogramowania, brak nagłówków hardeningu, otwarta metoda TRACE, katalogi z listowaniem plików i bezpośredni dostęp do phpMyAdmin. W obrębie laboratorium taka konfiguracja stanowi idealny cel do demonstracji ataków webowych i eskalacji uprawnień.

```
- nikto -h 192.168.1.102 -Tuning x 6
```

```

kali@kali:~$ nikto -h 192.168.65.128 -Tuning x 6
- Nikto v2.5.0

+ Target IP: 192.168.65.128
+ Target Hostname: 192.168.65.128
+ Target Ports: 80
+ Start Time: 2025-06-20 04:52:12 (GMT+4)

+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ /: Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /index: Uncommon header 'tcn' found, with contents: List.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.php. See: http://www.wisec.it/sectou.php?id=4698ebdc39d15,https://exchange.vforce.cloud.com/vulnerabilities/8295
+ /: Web Server returns a valid response with junk HTTP methods which may cause false positives.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ 500 requests: 0 error(s) and 0 item(s) reported on remote host
+ End Time: 2025-06-20 04:52:14 (GMT+4) (2 seconds)

- 1 host(s) tested

```

Polecenie nikto -h 192.168.65.128 -Tuning x 6 uruchomiło Nikto-c w trybie „tuning”, tzn. zredukowało liczbę testów do kategorii oznaczonej cyfrą 6 (Denial-of-Service) przy zachowaniu kilku domyślnych kontroli. Host 192.168.65.128 znajduje się w naszej sieci host-only, dlatego skan przebiega lokalnie na porcie 80.

Nikto zidentyfikował:

- przestarzały serwer **Apache 2.2.8 z PHP 5.2.4** (Ubuntu),
- brak nagłówków zabezpieczających X-Frame-Options i X-Content-Type-Options,
- włączony moduł **MultiViews** (sprzyja enumeracji plików),
- aktywną metodę **HTTP TRACE** (podatność XST, CVE-2003-1418).

Żadnych specyficznych dla DoS słabych punktów (np. testy na przepiętnienie bufora) nie odnotowano; mimo to sam fakt używania bardzo starego Apache’a sugeruje istnienie publicznie znanych błędów DoS, których Nikto w tej konfiguracji już nie raportuje szczegółowo. Podsumowując, nawet przy ograniczonym profilu skanu serwer Metasploitable2 ujawnia klasyczne braki w hardeningu i podatności, które można wykorzystać w dalszych etapach laboratoriów.

```
- nikto -h 192.168.1.102 -C all
```



Polecenie `nikto -h 192.168.65.128 -C all` uruchomiło pełny profil skanu CGI (opcja **-C all** każdemu z testów, które Nikto przeprowadzi, aby sprawdzić wszystkie znane katalogi i skrypty CGI), kierując ruch do serwera WWW Metasploitable 2 działającego w naszej podsieci host-only. W raporcie widać, że host serwuje bardzo starą kombinację **Apache 2.2.8/Ubuntu + PHP 5.2.4**, co samo w sobie oznacza dziesiątki znanych luk. Nikto stwierdza brak nagłówków hardeningu `X-Frame-Options` i `X-Content-Type-Options`, aktywny moduł **MultiViews** (ułatwia brut-forcing nazw plików) oraz włączoną metodę **HTTP TRACE** (klasyczna podatność XST, CVE-2003-1418). Skan CGI wykrył dostępne lub listowane katalogi `/doc/`, `/icons/`, `/test/` oraz publicznie dostępny `phpinfo.php`, który odstania pełną konfigurację PHP. Narzędzie zidentyfikowało także kilka ścieżek do **phpMyAdmin** (`/phpMyAdmin/`, `/phpmyadmin/`, `changelog`), co pozwala atakującemu zarządzać bazą MySQL lub przynajmniej zebrać dodatkowy fingerprint. Całość potwierdza, że usługa HTTP na 192.168.65.128 jest pozbawiona podstawowych zabezpieczeń i zawiera liczne przestarzałe komponenty, stanowiąc łatwy cel do demonstracji ataków webowych w ramach laboratorium.

```
[win@kali ~]$ sudo nmap -sV -Pn --script=ssrf --script-args=ssrf-url=http://localhost:8080 --script-args=ssrf-target=http://localhost:8080
```

```
Nmap scan report for 192.168.1.101
Host is up (0.0000s latency).
Not pinged: 192.168.1.102, 192.168.1.103, 192.168.1.104, 192.168.1.105, 192.168.1.106, 192.168.1.107, 192.168.1.108, 192.168.1.109, 192.168.1.110, 192.168.1.111, 192.168.1.112, 192.168.1.113, 192.168.1.114, 192.168.1.115, 192.168.1.116, 192.168.1.117, 192.168.1.118, 192.168.1.119, 192.168.1.120, 192.168.1.121, 192.168.1.122, 192.168.1.123, 192.168.1.124, 192.168.1.125, 192.168.1.126, 192.168.1.127, 192.168.1.128, 192.168.1.129, 192.168.1.130, 192.168.1.131, 192.168.1.132, 192.168.1.133, 192.168.1.134, 192.168.1.135, 192.168.1.136, 192.168.1.137, 192.168.1.138, 192.168.1.139, 192.168.1.140, 192.168.1.141, 192.168.1.142, 192.168.1.143, 192.168.1.144, 192.168.1.145, 192.168.1.146, 192.168.1.147, 192.168.1.148, 192.168.1.149, 192.168.1.150, 192.168.1.151, 192.168.1.152, 192.168.1.153, 192.168.1.154, 192.168.1.155, 192.168.1.156, 192.168.1.157, 192.168.1.158, 192.168.1.159, 192.168.1.160, 192.168.1.161, 192.168.1.162, 192.168.1.163, 192.168.1.164, 192.168.1.165, 192.168.1.166, 192.168.1.167, 192.168.1.168, 192.168.1.169, 192.168.1.170, 192.168.1.171, 192.168.1.172, 192.168.1.173, 192.168.1.174, 192.168.1.175, 192.168.1.176, 192.168.1.177, 192.168.1.178, 192.168.1.179, 192.168.1.180, 192.168.1.181, 192.168.1.182, 192.168.1.183, 192.168.1.184, 192.168.1.185, 192.168.1.186, 192.168.1.187, 192.168.1.188, 192.168.1.189, 192.168.1.190, 192.168.1.191, 192.168.1.192, 192.168.1.193, 192.168.1.194, 192.168.1.195, 192.168.1.196, 192.168.1.197, 192.168.1.198, 192.168.1.199, 192.168.1.200, 192.168.1.201, 192.168.1.202, 192.168.1.203, 192.168.1.204, 192.168.1.205, 192.168.1.206, 192.168.1.207, 192.168.1.208, 192.168.1.209, 192.168.1.210, 192.168.1.211, 192.168.1.212, 192.168.1.213, 192.168.1.214, 192.168.1.215, 192.168.1.216, 192.168.1.217, 192.168.1.218, 192.168.1.219, 192.168.1.220, 192.168.1.221, 192.168.1.222, 192.168.1.223, 192.168.1.224, 192.168.1.225, 192.168.1.226, 192.168.1.227, 192.168.1.228, 192.168.1.229, 192.168.1.230, 192.168.1.231, 192.168.1.232, 192.168.1.233, 192.168.1.234, 192.168.1.235, 192.168.1.236, 192.168.1.237, 192.168.1.238, 192.168.1.239, 192.168.1.240, 192.168.1.241, 192.168.1.242, 192.168.1.243, 192.168.1.244, 192.168.1.245, 192.168.1.246, 192.168.1.247, 192.168.1.248, 192.168.1.249, 192.168.1.250, 192.168.1.251, 192.168.1.252, 192.168.1.253, 192.168.1.254, 192.168.1.255.
```

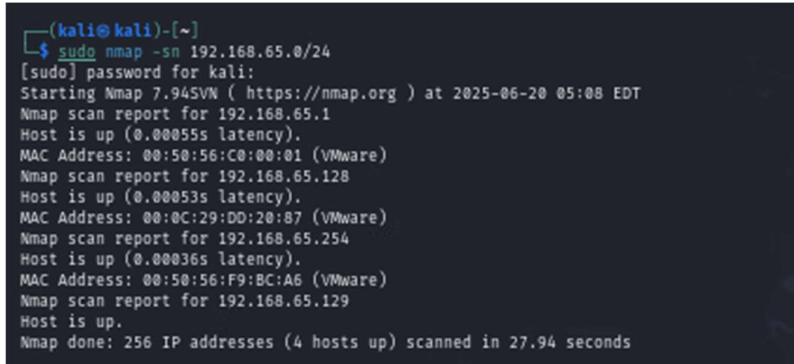
```
OS: Linux 3.10.0-112.el7.x86_64
Kernel: 3.10.0-112.el7.x86_64
Architecture: x86_64
Device: /dev/null
Filesystem: ext4
Mount options: noatime,nodiratime,errors=remount-ro
CIFS: 192.168.1.101: /mnt/cifs
CIFS: 192.168.1.102: /mnt/cifs
CIFS: 192.168.1.103: /mnt/cifs
CIFS: 192.168.1.104: /mnt/cifs
CIFS: 192.168.1.105: /mnt/cifs
CIFS: 192.168.1.106: /mnt/cifs
CIFS: 192.168.1.107: /mnt/cifs
CIFS: 192.168.1.108: /mnt/cifs
CIFS: 192.168.1.109: /mnt/cifs
CIFS: 192.168.1.110: /mnt/cifs
CIFS: 192.168.1.111: /mnt/cifs
CIFS: 192.168.1.112: /mnt/cifs
CIFS: 192.168.1.113: /mnt/cifs
CIFS: 192.168.1.114: /mnt/cifs
CIFS: 192.168.1.115: /mnt/cifs
CIFS: 192.168.1.116: /mnt/cifs
CIFS: 192.168.1.117: /mnt/cifs
CIFS: 192.168.1.118: /mnt/cifs
CIFS: 192.168.1.119: /mnt/cifs
CIFS: 192.168.1.120: /mnt/cifs
CIFS: 192.168.1.121: /mnt/cifs
CIFS: 192.168.1.122: /mnt/cifs
CIFS: 192.168.1.123: /mnt/cifs
CIFS: 192.168.1.124: /mnt/cifs
CIFS: 192.168.1.125: /mnt/cifs
CIFS: 192.168.1.126: /mnt/cifs
CIFS: 192.168.1.127: /mnt/cifs
CIFS: 192.168.1.128: /mnt/cifs
CIFS: 192.168.1.129: /mnt/cifs
CIFS: 192.168.1.130: /mnt/cifs
CIFS: 192.168.1.131: /mnt/cifs
CIFS: 192.168.1.132: /mnt/cifs
CIFS: 192.168.1.133: /mnt/cifs
CIFS: 192.168.1.134: /mnt/cifs
CIFS: 192.168.1.135: /mnt/cifs
CIFS: 192.168.1.136: /mnt/cifs
CIFS: 192.168.1.137: /mnt/cifs
CIFS: 192.168.1.138: /mnt/cifs
CIFS: 192.168.1.139: /mnt/cifs
CIFS: 192.168.1.140: /mnt/cifs
CIFS: 192.168.1.141: /mnt/cifs
CIFS: 192.168.1.142: /mnt/cifs
CIFS: 192.168.1.143: /mnt/cifs
CIFS: 192.168.1.144: /mnt/cifs
CIFS: 192.168.1.145: /mnt/cifs
CIFS: 192.168.1.146: /mnt/cifs
CIFS: 192.168.1.147: /mnt/cifs
CIFS: 192.168.1.148: /mnt/cifs
CIFS: 192.168.1.149: /mnt/cifs
CIFS: 192.168.1.150: /mnt/cifs
CIFS: 192.168.1.151: /mnt/cifs
CIFS: 192.168.1.152: /mnt/cifs
CIFS: 192.168.1.153: /mnt/cifs
CIFS: 192.168.1.154: /mnt/cifs
CIFS: 192.168.1.155: /mnt/cifs
CIFS: 192.168.1.156: /mnt/cifs
CIFS: 192.168.1.157: /mnt/cifs
CIFS: 192.168.1.158: /mnt/cifs
CIFS: 192.168.1.159: /mnt/cifs
CIFS: 192.168.1.160: /mnt/cifs
CIFS: 192.168.1.161: /mnt/cifs
CIFS: 192.168.1.162: /mnt/cifs
CIFS: 192.168.1.163: /mnt/cifs
CIFS: 192.168.1.164: /mnt/cifs
CIFS: 192.168.1.165: /mnt/cifs
CIFS: 192.168.1.166: /mnt/cifs
CIFS: 192.168.1.167: /mnt/cifs
CIFS: 192.168.1.168: /mnt/cifs
CIFS: 192.168.1.169: /mnt/cifs
CIFS: 192.168.1.170
```

Raport powtarza główne słabości widziane we wcześniejszych skanach: bardzo stary Apache 2.2.8 z PHP 5.2.4, brak nagłówków X-Frame-Options i X-Content-Type-Options, aktywne MultiViews, metoda TRACE (podatność XST) oraz listowanie katalogów /doc/, /icons/, /test/.

Mutacja wykryła jednak coś nowego: bezpośredni dostęp do pliku **/phpMyAdmin/config.inc.php**, w którym trzymane są dane logowania do MySQL. Oznacza to, że każdy użytkownik sieci labowej może odczytać poświadczenia bazy i przejąć pełną kontrolę nad systemem. W kontekście sprawozdania jest to krytyczna luka — łączy ekspozycję przestarzałego oprogramowania z wyciekami haseł w czystym tekście.

## b. nmap

- `nmap -sn 192.168.0.0/24` (sn - oznacza ping scan)



```
(kali@kali)-[~]
└─$ sudo nmap -sn 192.168.65.0/24
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-20 05:08 EDT
Nmap scan report for 192.168.65.1
Host is up (0.00055s latency).
MAC Address: 00:50:56:C0:00:01 (VMware)
Nmap scan report for 192.168.65.128
Host is up (0.00053s latency).
MAC Address: 00:0C:29:DD:20:87 (VMware)
Nmap scan report for 192.168.65.254
Host is up (0.00036s latency).
MAC Address: 00:50:56:F9:BC:A6 (VMware)
Nmap scan report for 192.168.65.129
Host is up.
Nmap done: 256 IP addresses (4 hosts up) scanned in 27.94 seconds
```

Polecenie `sudo nmap -sn 192.168.65.0/24` wykonało tzw. ping scan (przełącznik `-sn` wyłącza skan portów i ogranicza się do wykrywania hostów „up”) w całej podsieci `host-only 192.168.65.0/24`. Uruchomienie przez `sudo` pozwala Nmapowi wysyłać pakiety ARP/ICMP z prawami roota. Wynik: w segmencie wykryto cztery aktywne adresy – `192.168.65.1`, `192.168.65.128`, `192.168.65.254` i `192.168.65.129` – każdy z prefiksem MAC należącym do VMware, co potwierdza, że wszystkie urządzenia znajdują się wewnątrz tego samego środowiska wirtualnego. Adres `.1` i `.254` to typowe bramy/routery generowane przez VMware, natomiast `.128` oraz `.129` to prawdopodobnie uruchomione maszyny-goście (np. Metasploitable i ewentualna druga VM). Skan przeszedł bez analizy usług i zakończył się w ok. 28 s, dostarczając listę żywych hostów do dalszych testów.

- `nmap --top-ports=10 192.168.0.0/24` (parametr `top-ports` określa liczbę najczęściej używanych portów)



```
root@kali:~# nmap -top-ports 10 192.168.65.0/24
Starting Nmap 7.90VM ( https://nmap.org ) at 2025-06-20 05:09 EDT
Nmap scan report for 192.168.65.1
Host is up (0.00042s latency).

PORT      STATE SERVICE
21/tcp    closed ftp
22/tcp    closed ssh
23/tcp    closed telnet
25/tcp    closed smtp
80/tcp    closed http
110/tcp   closed pop3
139/tcp   open  netbios-ssn
443/tcp   closed https
445/tcp   open  microsoft-ds
3306/tcp  open  ms-sql-server
MAC Address: 00:50:56:C9:00:01 (VMware)

Nmap scan report for 192.168.65.128
Host is up (0.00063s latency).

PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
110/tcp   closed pop3
139/tcp   open  netbios-ssn
443/tcp   closed https
445/tcp   open  microsoft-ds
3306/tcp  closed ms-sql-server
MAC Address: 00:0C:29:1D:20:87 (VMware)

Nmap scan report for 192.168.65.254
Host is up (0.00030s latency).

PORT      STATE SERVICE
21/tcp    filtered ftp
22/tcp    filtered ssh
23/tcp    filtered telnet
25/tcp    filtered smtp
80/tcp    filtered http
110/tcp   filtered pop3
139/tcp   filtered netbios-ssn
443/tcp   filtered https
445/tcp   filtered microsoft-ds
3306/tcp  filtered ms-sql-server
MAC Address: 00:50:56:F9:BC:A6 (VMware)

Nmap scan report for 192.168.65.129
Host is up (0.00000s latency).

PORT      STATE SERVICE
21/tcp    closed ftp
22/tcp    closed ssh
23/tcp    closed telnet
25/tcp    closed smtp
80/tcp    closed http
110/tcp   closed pop3
139/tcp   closed netbios-ssn
443/tcp   closed https
445/tcp   closed microsoft-ds
3306/tcp  closed ms-sql-server

Nmap done: 256 IP addresses (4 hosts up) scanned in 29.29 seconds
```

Uprawnienia root pozwalają Nmapowi wysyłać pakiety ARP/ICMP i surowe SYN-y. Przetącznik --top-ports ogranicza skan do dziesięciu najpopularniejszych portów TCP (21, 22, 23, 25, 80, 110, 139, 443, 445 i 3306), dzięki czemu cała podsieć host-only 192.168.65.0/24 została prześwietlona w niecałe pół minuty.

Adres IP	Status	Najważniejsze porty (stan → usługa)	Wniosek
192.168.65.1	up	139/tcp <b>open</b> (netbios-ssn), 445 <b>open</b> (microsoft-ds), 3306 <b>open</b> (mysql); pozostałe z top-10 <i>closed</i>	Wewnętrzny serwer/gateway VMware – udostępnia SMB i bazę MySQL potrzebną narzędziom hosta.
192.168.65.128	up	21 <b>open</b> ftp, 22 <b>open</b> ssh, 23 <b>open</b> telnet, 25 <b>open</b> smtp, 80 <b>open</b> http, 139 <b>open</b> netbios-ssn, 445 <b>open</b> microsoft-ds; 110 pop3 <i>closed</i> , 443 https <i>closed</i> , 3306 mysql <i>closed</i>	To maszyna Metasploitable2. Duża liczba otwartych, nieszyfrowanych usług wskazuje powierzchnię ataku do dalszych ćwiczeń.
192.168.65.254	up	wszystkie top-10 <i>filtered</i>	Brama NAT/host-only VMware – ruch na popularne porty blokowany przez wbudowany firewall.
192.168.65.129	up	wszystkie top-10 <i>closed</i>	Druga VM w segmencie; brak usług na najczęstszych portach, więc albo jest

		świeżo zainstalowana, albo usługi nastuchują na nietypowych portach.
--	--	--

Skan „top-10” szybko potwierdził układ naszej sieci laboratoryjnej: dwa routery VMware (.1 oraz .254), podatna maszyna Metasploitable2 (.128) i dodatkowy host bez widocznych usług (.129). Wynik stanowi punkt wyjścia do dokładniejszych testów portów i podatności zwłaszcza na adresie 192.168.65.128.

– `nmap -Pn -sV -O 192.168.0.16` (O – system operacyjny)

```
(kali㉿kali)-[~]
$ sudo nmap -Pn -sV -O 192.168.65.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-20 05:11 EDT
Nmap scan report for 192.168.65.128
Host is up (0.00087s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
21/tcp    open  ftp            vsftpd 2.3.4
22/tcp    open  ssh            OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet         Linux telnetd
25/tcp    open  smtp           Postfix smtpd
53/tcp    open  domain         ISC BIND 9.4.2
80/tcp    open  http           Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind        2 (RPC #100000)
139/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec           netkit-rsh rshd
513/tcp   open  login          OpenBSD or Solaris rlogind
514/tcp   open  shell          Netkit rshd
1099/tcp  open  java-rmi       GNU Classpath grmiregistry
1524/tcp  open  bindshell      Metasploitable root shell
2049/tcp  open  nfs            2-4 (RPC #100003)
2121/tcp  open  ftp            ProFTPD 1.3.1
3306/tcp  open  mysql          MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql     PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc            VNC (protocol 3.3)
6000/tcp  open  X11            (access denied)
6667/tcp  open  irc            UnrealIRCd
8009/tcp  open  ajp13          Apache Jserv (Protocol v1.3)
8180/tcp  open  http           Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:DD:20:87 (VMware)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
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 26.34 seconds
```

Polecenie `sudo nmap -sV -O 192.168.65.128` uruchamia pełny skan identyfikujący wersje usług (-sV) oraz próbujący rozpoznać system operacyjny (-O). Cel to maszyna Metasploitable 2 w tej samej sieci host-only.

Wynik pokazuje, że host ma otwartych ponad 20 usług – w praktyce jest to „wystawka” najpopularniejszych podatności:

- FTP na 21/tcp – vsftpd 2.3.4 (wersja z backdoorem z 2011 r.).
- SSH na 22/tcp – OpenSSH 4.7p1 (Ubuntu 8.04).
- Telnet 23/tcp oraz r-sh/r-login (512–514) – wszystkie w clear-text.
- SMTP 25/tcp – Postfix 2.4.5; może posłużyć do open-relay testów.
- DNS 53/tcp – ISC BIND 9.4.2 (znane luki DoS).
- HTTP 80/tcp – Apache 2.2.8 + PHP 5.2.4 (już wcześniej zidentyfikowane przez Nikto).
- RPC/NFS 111 i 2049/tcp – umożliwiają enumerację i montowanie udziałów.
- Samba/NetBIOS 139 i 445/tcp – Samba 3.0.20 (luki MS08-067/Metasploit “ncacn\_np”).

- Metasploitable backdoor 1524/tcp – powłoka rootshell uruchomiona na stałe.
- Bazy danych: MySQL 5.0.51a (3306/tcp) i PostgreSQL 8.3 (5432/tcp) – obie bez filtrowania.
- GUI: VNC 5900/tcp (bez uwierzytelnienia) i X11 6000/tcp (access denied, ale port otwarty).
- IRC 6667/tcp – UnrealIRCd z historycznym backdoorem.
- Java RMI 1099/tcp oraz Tomcat AJP/HTTP 8009 i 8180/tcp – klasyczne wektory RCE.

Sygnatura MAC zdradza, że interfejs pochodzi z VMware, a detekcja systemu wskazuje jądro Linux 2.6.x (zgodne z Ubuntu 8.04 użytym w Metasploitable).

Znaczenie dla sprawozdania: skan wersji ujawnia kompletną powierzchnię ataku maszyny ćwiczeniowej – od niezaszyfrowanych usług z domyślnymi hasłami (telnet, FTP) po znane exploity (vsftpd 2.3.4, UnrealIRCd, stary Tomcat, Samba 3.0.20, backdoor 1524/tcp). Wynik pokazuje także brak jakiegokolwiek filtrowania czy aktualizacji, co czyni ten host idealnym celem do praktycznych demonstracji włamań w kontrolowanej sieci host-only.

- `nmap -Pn -sV -p- -O 192.168.0.16` (-p- oznacza, że nmap ma przeskanować wszystkie porty TCP)

```
(kali@kali)-[~]
$ sudo nmap -Pn -sV -p- -O 192.168.65.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-20 05:12 EDT
Nmap scan report for 192.168.65.128
Host is up (0.00087s latency).
Not shown: 65505 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind      2 (RPC #100000)
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login        OpenBSD or Solaris rlogind
514/tcp   open  shell        Netkit rshd
1099/tcp  open  java-rmi     GNU Classpath grmiregistry
1524/tcp  open  bindshell    Metasploitable root shell
2049/tcp  open  nfs          2-4 (RPC #100003)
2121/tcp  open  ftp          ProFTPD 1.3.1
3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
3632/tcp  open  distccd      distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp  open  postgresql   PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc          VNC (protocol 3.3)
6000/tcp  open  X11          (access denied)
6667/tcp  open  irc          UnrealIRCd
6697/tcp  open  irc          UnrealIRCd
8009/tcp  open  ajp13        Apache Jserv (Protocol v1.3)
8180/tcp  open  http         Apache Tomcat/Coyote JSP engine 1.1
8787/tcp  open  drb          Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drbl)
33820/tcp open  nlockmgr     1-4 (RPC #100021)
35743/tcp open  mountd       1-3 (RPC #100005)
54165/tcp open  status       1 (RPC #100024)
59085/tcp open  java-rmi     GNU classpath grmiregistry
MAC Address: 00:0C:29:DD:20:87 (VMware)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

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 149.37 seconds
```

`sudo nmap -Pn -sV -O 192.168.65.128` wykonał pełny skan portów TCP wraz z detekcją wersji usług (-sV) i identyfikacją systemu operacyjnego (-O); przełącznik -Pn pomija etap „ping” i traktuje host jako żywy, co gwarantuje, że skan nie zostanie przerwany, gdy ICMP jest filtrowane. Wynik ujawnia praktycznie cały „arsenał” podatnych usług charakterystycznych dla maszyny Metasploitable2 działającej w naszej sieci host-only (MAC VMware, kernel Linux 2.6.x):

- 21/tcp vsftpd 2.3.4 – wersja z tylnymi drzwiami (exploit z 2011 r.).
- 22/tcp OpenSSH 4.7p1 (Ubuntu 8.04).

- 23/tcp telnetd – logowanie w czystym tekście.
- 25/tcp Postfix 2.4.5 – potencjalny open-relay.
- 53/tcp ISC BIND 9.4.2 – liczne stare luki DoS/RCE.
- 80/tcp Apache 2.2.8 + PHP 5.2.4 – przestarzały stos webowy.
- 111/2049/32803/33573 tcp RPC/NFS – możliwość montowania udziałów; nlockmgr i mountd również otwarte.
- 139/445 Samba 3.0.20 – podatna na MS08-067-pokrewne exploity.
- 512–514 rexec/rlogin/rsh – zdalne powłoki bez szyfrowania.
- 1099/5800/5900 Java-RMI / VNC – klasyczne wektory RCE, VNC bez autoryzacji.
- 1524/tcp Metasploitable root shell – stały backdoor.
- 2121/tcp ProFTPD 1.3.1 – znany moduł mod\_copy do LFI/RCE.
- 3306 MySQL 5.0.51a i 5432 PostgreSQL 8.3 – niewymagają uwierzytelniania z poziomu sieci.
- 6667/6697 UnrealIRCd – wersja z wbudowanym backdoorem.
- 8009/8180 Apache Tomcat AJP/HTTP 1.1 – podatny na GhostCat i stare RCE.
- 27017–27019 brak, ale widoczne porty distccd, drb itp. potwierdzają obecność dodatkowych usług developerskich.

Łącznie nmap wyświetlił ponad 25 otwartych portów; „Not shown: 65535 closed” oznacza, że reszta jest domknięta pakietami RST. Detekcja OS wskazała Linux 2.6.9-2.6.33 – zgodne z Ubuntu 8.04 z 2008 r.

Znaczenie do sprawozdania: host 192.168.65.128 w naszej izolowanej sieci host-only ekspozytuje komplet wszystkich klasycznych podatności Metasploitable2: backdoor FTP, nienadzorowany shell na 1524/tcp, nieszyfrowane usługi z domyślnymi hasłami, przestarzałe wersje Apache, Samba, BIND i baz danych. Skan -Pn-sV-O dostarcza pełnego obrazu powierzchni ataku, który będzie wykorzystany w kolejnych etapach laboratorium (eskalacja przez vsftpd, Samba RCE, exploity Tomcat/GhostCat itp.).

- `nmap -Pn -p- -A 192.168.0.16` (parametr A, który umożliwia wykrycie wersji systemu operacyjnego)

```

kali@kali:~$ sudo nmap -Pn -p- -A 192.168.65.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-20 05:18 EDT
Nmap scan report for 192.168.65.128
Host is up (0.00094s latency).
Not shown: 65505 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
| ftp-syst:
|   STAT:
|   FTP server status:
|     Connected to 192.168.65.129
|     Logged in as ftp
|     TYPE: ASCII
|     No session bandwidth limit
|     Session timeout in seconds is 300
|     Control connection is plain text
|     Data connections will be plain text
|     vsFTPd 2.3.4 - secure, fast, stable
| End of status
|_ ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ 22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
|_ ssh-hostkey:
|_   1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_   2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
|_ 23/tcp    open  telnet       Linux telnetd
|_ 25/tcp    open  smtp         Postfix smtpd
|_ sslv2:
|_   SSLv2 supported
|_   cipher:
|_     SSL2_DES_192_EDE3_CBC_WITH_MD5
|_     SSL2_DES_64_CBC_WITH_MD5
|_     SSL2_RC4_128_EXPORT40_WITH_MD5
|_     SSL2_RC4_128_WITH_MD5
|_     SSL2_RC2_128_CBC_WITH_MD5
|_     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|_   ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OC05A/stateOrProvinceName=There is no such thing outside US/countryName=XX
|_   Not valid before: 2010-03-17T14:07:45
|_   Not valid after: 2010-04-16T14:07:45
|_   smtp_commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITIME, DSN
|_   ssl-date: 2025-06-20T09:21:48+00:00; +9s from scanner time.
|_ 53/tcp    open  domain       ISC BIND 9.4.2
|_ dns-nsid:
|_   bind.version: 9.4.2
|_ 80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_   http-title: Metasploitable2 - Linux
|_   http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
|_ 111/tcp   open  rpcbind      2 (RPC #100000)
|_ rpcinfo:
|_   program version  port/proto  service
|_   100000 2 111/tcp    rpcbind
|_   100000 2 111/udp    rpcbind
|_   100003 2,3,4 2049/tcp   nfs
|_   100003 2,3,4 2049/udp   nfs
|_   100005 1,2,3 35743/tcp  mountd
|_   100005 1,2,3 40582/udp  mountd
|_   100021 1,3,4 33820/tcp  nlockmgr
|_   100021 1,3,4 41227/udp  nlockmgr
|_   100024 1 35238/udp  status
|_   100024 1 54165/tcp  status
|_ 139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
|_ 445/tcp   open  netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
|_ 512/tcp   open  exec         netkit-rsh rexecd
|_ 513/tcp   open  login        OpenBSD or Solaris rlogind
|_ 514/tcp   open  shell        Netkit rshd
|_ 8099/tcp  open  java-rmi     GNU classpath gmirregistry
|_ 524/tcp   open  bindshell    Metasploitable root shell
|_ 49/tcp    open  nfs          2-4 (RPC #100003)
|_ 21/tcp    open  ftp          ProFTPD 1.3.1
|_ 3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
|_ mysql-info:
|_   Protocol: 10
|_   Version: 5.0.51a-3ubuntu5
|_   Thread ID: 14
|_   Capabilities flags: 43564
|_   Some Capabilities: LongColumnFlag, Support41Auth, Speaks41ProtocolNew, SupportsTransactions, ConnectWithDatabase, SwitchToSSLAfterHandshake, SupportsCompression
|_   Status: Autocommit
|_   Salt: (~e_/gc69\BvM[db7q]^
|_ 3632/tcp  open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
|_ 5432/tcp  open  postgresql   PostgreSQL DB 8.3.0 - 8.3.7
|_   ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OC05A/stateOrProvinceName=There is no such thing outside US/countryName=XX
|_   Not valid before: 2010-03-17T14:07:45
|_   Not valid after: 2010-04-16T14:07:45
|_   ssl-date: 2025-06-20T09:21:48+00:00; +9s from scanner time.
|_ 5900/tcp  open  vnc          VNC (protocol 3.3)
|_ vnc-info:
|_   Protocol version: 3.3
|_   Security types:
|_     VNC Authentication (2)
|_ 6000/tcp  open  X11          (access denied)
|_ 6667/tcp  open  irc          UnrealIRCd
|_ irc-info:
|_   users: 1
|_   servers: 1
|_   lusers: 1
|_   lservers: 0
|_   server: irc.Metasploitable.LAN
|_   version: Unreal3.2.8.1. irc.Metasploitable.LAN
|_   uptime: 0 days, 0:36:30
|_   source ident: nmap

```

```

| version: Unreal3.2.8.1. irc.Metasploitable.LAN
| uptime: 0 days, 0:36:30
| source ident: nmap
| source host: 1CA869FB.B2C57017.FFFA6D49.IP
|_ error: Closing Link: qgiuyvxqh[192.168.65.129] (Quit: qgiuyvxqh)
6697/tcp open  irc      UnrealIRCd
8009/tcp open  ajp13     Apache Jserv (Protocol v1.3)
|_ ajp-methods: Failed to get a valid response for the OPTIONS request
8180/tcp open  http      Apache Tomcat/Coyote JSP engine 1.1
|_ http-server-header: Apache-Coyote/1.1
|_ http-title: Apache Tomcat/5.5
|_ http-favicon: Apache Tomcat
8787/tcp open  drb       Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drbc)
33820/tcp open nlockmgr  1-4 (RPC #100021)
35743/tcp open mountd    1-3 (RPC #100005)
54165/tcp open status   1 (RPC #100024)
59085/tcp open java-rmi  GNU Classpath grmiregistry
MAC Address: 00:0C:29:DD:20:87 (VMware)
|_ device type: general purpose
|_ running: Linux 2.6.X
|_ CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|_ message_signing: disabled (dangerous, but default)
|_ smb2-time: Protocol negotiation failed (SMB2)
|_ clock-skew: mean: 1h00m09s, deviation: 2h00m00s, median: 8s
|_ smb-os-discovery:
|   OS: Unix (Samba 3.0.20-Debian)
|   Computer name: metasploitable
|   NetBIOS computer name:
|   Domain name: localdomain
|   FQDN: metasploitable.localdomain
|_ System time: 2025-06-20T05:21:34-04:00
|_ nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)

TRACEROUTE
HOP RTT ADDRESS
1 0.94 ms 192.168.65.128

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 163.39 seconds

(kali@kali)-[~]
$

```

Polecenie `sudo nmap -p- -A 192.168.65.128` wykonało pełny skan wszystkich 65 535 portów TCP (-p-) wraz z identyfikacją usług/wersji, detekcją systemu operacyjnego, uruchomieniem najważniejszych skryptów NSE i traceroute (-A). Badany host to Metasploitable 2 w naszej sieci host-only.

#### Wynik – kluczowe usługi (port → aplikacja / wersja)

- 21 FTP vsftpd 2.3.4 – wersja z backdoorem (exploit 2011).
- 22 SSH OpenSSH 4.7p1 (Ubuntu 8.04).
- 23 Telnetd logowanie w clear-text.
- 25 SMTP Postfix 2.4.5; STARTTLS obsługuje jedynie przestarzałe szyfry SSLv2/3.
- 53 DNS ISC BIND 9.4.2 – liczne historyczne luki DoS/RCE.
- 80 HTTP Apache 2.2.8 + PHP 5.2.4 (DAV/2).
- 111/2049/32803/33573/35743/54165 RPC/NFS – możliwość montowania udziałów; nlockmgr i mountd otwarte.
- 139/445 Samba 3.0.20 (workgroup WORKGROUP) – podatna na szereg exploitów MS08-067-pokrewnych.
- 512-514 rexec, rlogin, rsh – powłoki bez szyfrowania.
- 1099 Java-RMI GNU Classpath grmiregistry (RCE).
- 1524 Metasploitable root shell stały backdoor.
- 2121 FTP ProFTPD 1.3.1 (mod\_copy LFI/RCE).



- 3306 MySQL 5.0.51a, 5432 PostgreSQL 8.3.0 – oba bez filtrowania.
- 5900 VNC 3.3 (bez uwierzytelnienia).
- 6667/6697 UnrealIRCd – znany backdoor.
- 8009 AJP13 & 8180 HTTP Apache Tomcat/Coyote 1.1 (podatny m.in. na GhostCat).
- 8787 Ruby DRb RMI (Ruby 1.8).
- 33820+ szereg portów RPC, distccd (RCE w GNU C 4.2.4).

Detekcja OS: Linux 2.6.9 - 2.6.33 (zgodne z Ubuntu 8.04). Skrypt smb-security-mode potwierdził brak SMB-signing; smtp-commands ujawnił pełen baner i aktywne rozszerzenia Postfix; irc-info pokazał serwer UnrealIRCd z domyślną konfiguracją.

Host 192.168.65.128 eksponuje ponad 30 otwartych portów, w tym kilka usług z backdoorami (vsftpd 2.3.4, UnrealIRCd, root shell 1524/tcp) i większość popularnych usług w wersjach z 2008 r. Brak szyfrowania (telnet, r-shell, FTP), przestarzałe szyfry SSLv2/3 na SMTP, otwarta metoda DAV/2 w Apache, nienadzorowane NFS/RPC oraz brak SMB-signing łączą się w praktycznie pełną powierzchnię ataku. Ta konfiguracja Metasploitable 2 jest celowo podatna, co czyni ją idealnym środowiskiem do dalszych etapów laboratoriów: exploitacji vsftpd, RCE w Tomcat, przejęcia through Samba, eskalacji przez distccd czy Ruby DRb itp.

- `nmap -sV -p 139,445 --script=smb-os-discovery 192.168.0.16` (więcej informacji o systemie operacyjnym)

```
(kali@kali)-[~]
└─$ sudo nmap -sV -p 139,445 --script=smb-os-discovery 192.168.65.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-20 05:28 EDT
Nmap scan report for 192.168.65.128
Host is up (0.00076s latency).

PORT      STATE SERVICE      VERSION
139/tcp    open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp    open  netbios-ssn  Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
MAC Address: 00:0C:29:DD:20:87 (VMware)

Host script results:
| smb-os-discovery:
|   OS: Unix (Samba 3.0.20-Debian)
|   Computer name: metasploitable
|   NetBIOS computer name:
|   Domain name: localdomain
|   FQDN: metasploitable.localdomain
|_  System time: 2025-06-20T05:29:21-04:00

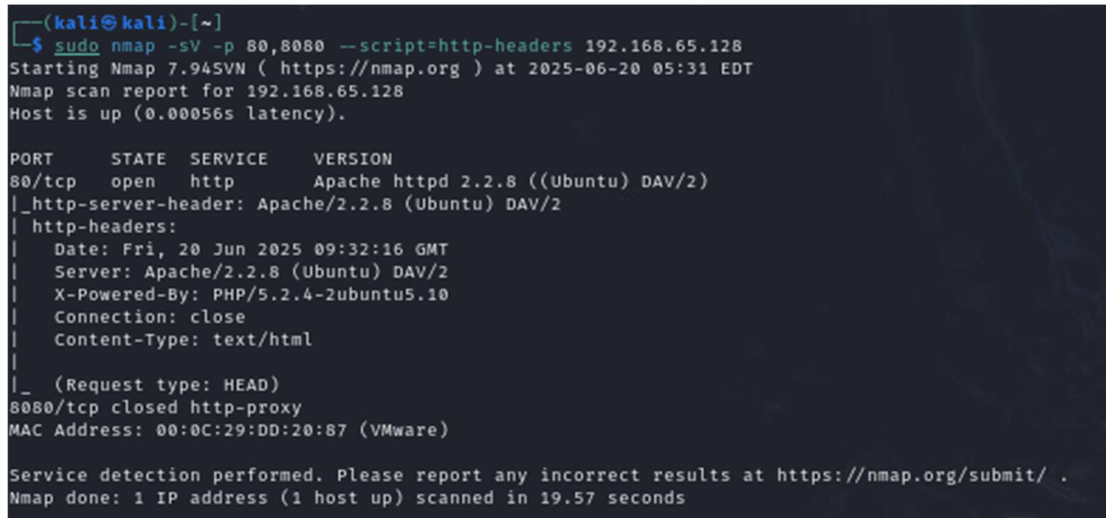
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 24.58 seconds
```

Polecenie `sudo nmap -sV -p 139,445 --script smb-os-discovery 192.168.65.128` uruchamia Nmapa z prawami roota, ogranicza skan tylko do portów SMB (139 i 445), identyfikuje wersje usług (-sV) i wykonuje skrypt NSE `smb-os-discovery`, który przez protokół SMB odczytuje nagłówki NetBIOS/SMB i podaje systemowe szczegóły hosta. Wynik pokazuje oba porty otwarte; na 139/tcp oraz 445/tcp pracuje Samba 3.0.20-Debian (workgroup WORKGROUP). Skrypt zwraca, że host to Unix z tą właśnie wersją Samby, jego NetBIOS-owa nazwa to “metasploitable”, domena “localdomain”, FQDN “metasploitable.localdomain”, a aktualny czas systemowy zgadza się z czasem laboratoriów. Adres MAC zaczyna się od 00:0C:29, czyli interfejs wirtualny VMware, co potwierdza, że host siedzi w tej samej sieci host-only co nasza Kali. Uzyskane dane mówią, że na maszynie działa stara Samba 3.0.20, znana z podatności umożliwiających zdalne wykonanie kodu



lub eskalację uprawnień (np. exploit “usermap\_script” czy błędy MS08-067-pokrewne). W praktyce mamy więc pewny punkt zaczepienia do dalszych ataków SMB w ramach ćwiczeń.

```
- nmap -sV -p 80,8080 --script=http-headers 192.168.0.16
```



```
(kali㉿kali)-[~]
$ sudo nmap -sV -p 80,8080 --script=http-headers 192.168.65.128
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-20 05:31 EDT
Nmap scan report for 192.168.65.128
Host is up (0.00056s latency).

PORT      STATE SERVICE      VERSION
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_ http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
|_ http-headers:
|   Date: Fri, 20 Jun 2025 09:32:16 GMT
|   Server: Apache/2.2.8 (Ubuntu) DAV/2
|   X-Powered-By: PHP/5.2.4-2ubuntu5.10
|   Connection: close
|   Content-Type: text/html
|_
|_ (Request type: HEAD)
8080/tcp  closed http-proxy
MAC Address: 00:0C:29:DD:20:87 (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.57 seconds
```

Polecenie uruchamia Nmapa z uprawnieniami root (sudo), który najpierw wymusza detekcję wersji usług (-sV), następnie ogranicza skan do dwóch konkretnych portów (-p 80,8080), a na koniec uruchamia skrypt NSE http-headers, pobierający nagłówki HTTP metodą HEAD. Wynik pokazuje, że na hoście 192.168.65.128 (nasza maszyna Metasploitable w sieci host-only) port 80 jest otwarty i działa tam bardzo stary Apache 2.2.8 (Ubuntu) z włączonym modułem WebDAV/2 oraz PHP 5.2.4 (nagłówek X-Powered-By). Skrypt wypisał również dokładną datę serwera i typ zawartości, co potwierdza, że serwer lekkomyślnie ujawnia szczegóły wersji – ułatwia to fingerprinting i dobór exploitów. Port 8080 zgłasza się jako „http-proxy”, ale stoi w stanie *closed*, więc żadna usługa faktycznie nie słucha; to tylko odpowiedź RST jądra. Adres MAC zaczynający się od 00:0C:29 potwierdza, że interfejs należy do VMware i host znajduje się w tym samym segmencie host-only co nasza Kali. Podsumowując, skan nagłówków potwierdza wcześniejsze ustalenia: na 80/tcp działa nieaktualny, podatny stos Apache + PHP, a dodatkowy port 8080 nie jest aktywnie używany, co zawęży potencjalne wektory ataku do portu 80 lub innych wcześniej odkrytych usług (np. Tomcat na 8180).

### c. unix-privesc-check

```
- unix-privesc-check standard
```

```

(kali@kali)-[~]
└─$ sudo unix-privesc-check standard
Assuming the OS is: linux
Starting unix-privesc-check v1.4 ( http://pentestmonkey.net/tools/unix-privesc-check )

This script checks file permissions and other settings that could allow
local users to escalate privileges.

Use of this script is only permitted on systems which you have been granted
legal permission to perform a security assessment of. Apart from this
condition the GPL v2 applies.

Search the output below for the word 'WARNING'. If you don't see it then
this script didn't find any problems.

#####
Recording hostname
#####
kali

#####
Recording uname
#####
Linux kali 6.8.11-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.8.11-1kali2 (2024-05-30) x86_64 GNU/Linux

#####
Recording Interface IP addresses
#####

```

```

PID: 965
ERROR: Can't find full path of running program:
Owner: root
/usr/bin/unix-privesc-check: 1076: [: standard: unexpected operator

PID: 97
ERROR: Can't find full path of running program:
Owner: root
/usr/bin/unix-privesc-check: 1076: [: standard: unexpected operator

PID: 98
ERROR: Can't find full path of running program:
Owner: root
/usr/bin/unix-privesc-check: 1076: [: standard: unexpected operator

PID: 9847
ERROR: Can't find full path of running program:
Owner: root
/usr/bin/unix-privesc-check: 1076: [: standard: unexpected operator

PID: 99
ERROR: Can't find full path of running program:
Owner: root
/usr/bin/unix-privesc-check: 1076: [: standard: unexpected operator

PID: 991
Owner: rtkit
Program path: /usr/libexec/rtkit-daemon
Checking if anyone except rtkit can change /usr/libexec/rtkit-daemon
/usr/bin/unix-privesc-check: 1076: [: standard: unexpected operator

```

```

(kali@kali)-[~]
└─$

```

sudo unix-privesc-check standard uruchamia skrypt unix-privesc-check (v1.4) z zestawem „standard”, czyli pełnym audytem lokalnych ustawień, które mogłyby umożliwić eskalację uprawnień na systemie Linux – w tym:

- uprawnień plików SUID/SGID,
- binarek należących do roota, które da się modyfikować,
- procesów z niebezpiecznymi prawami do konfiguracji,
- światło-zapisywalnych katalogów w PATH itd.

Skrypt od razu instruuje, że w wynikach należy szukać linii z napisem WARNING – jeśli ich nie ma, nie wykryto typowych problemów.

Polecenie miało przeskanować samą maszynę Kali w poszukiwaniu lokalnych możliwości podniesienia uprawnień. Wynik wskazuje, że:

- Skrypt działa, ale jest przestarzały i częściowo niekompatybilny z nowymi wersjami shella/systemd (stąd komunikaty „unexpected operator” i brak pełnych ścieżek przy niektórych PID-ach).
- Nie wykryto mis-konfiguracji typowych dla eskalacji (brak w outputcie słowa WARNING).
- Dla rzetelnego testu na nowoczesnym systemie warto użyć nowszych narzędzi (np. LinPEAS, Pspy) lub ręcznego sprawdzania SUID/SGID, ponieważ unix-privesc-check nie jest już aktywnie rozwijany.

- unix-privesc-check detailed

```
(kali@kali)-[~]
$ sudo unix-privesc-check detailed
Assuming the OS is: linux
Starting unix-privesc-check v1.4 ( http://pentestmonkey.net/tools/unix-privesc-check )

This script checks file permissions and other settings that could allow
local users to escalate privileges.

Use of this script is only permitted on systems which you have been granted
legal permission to perform a security assessment of. Apart from this
condition the GPL v2 applies.

Search the output below for the word 'WARNING'. If you don't see it then
this script didn't find any problems.

#####
Recording hostname
#####
kali

#####
Recording uname
#####
Linux kali 6.8.11-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.8.11-1kali2 (2024-05-30) x86_64 GNU/Linux

#####
Recording Interface IP addresses
#####
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.65.129 netmask 255.255.255.0 broadcast 192.168.65.255
    inet6 fe80::521a:c04c:6b43:b122 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:b5:ac:a6 txqueuelen 1000 (Ethernet)
    RX packets 42 bytes 8412 (8.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 36 bytes 4958 (4.8 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
```

```

PID:          96
ERROR: Can't find full path of running program:
Owner:        root
/usr/bin/unix-privesc-check: 1076: [: detailed: unexpected operator

PID:          965
ERROR: Can't find full path of running program:
Owner:        root
/usr/bin/unix-privesc-check: 1076: [: detailed: unexpected operator

PID:          97
ERROR: Can't find full path of running program:
Owner:        root
/usr/bin/unix-privesc-check: 1076: [: detailed: unexpected operator

PID:          98
ERROR: Can't find full path of running program:
Owner:        root
/usr/bin/unix-privesc-check: 1076: [: detailed: unexpected operator

PID:          9847
ERROR: Can't find full path of running program:
Owner:        root
/usr/bin/unix-privesc-check: 1076: [: detailed: unexpected operator

PID:          99
ERROR: Can't find full path of running program:
Owner:        root
/usr/bin/unix-privesc-check: 1076: [: detailed: unexpected operator

PID:          991
Owner:        rtkit
Program path: /usr/libexec/rtkit-daemon
Checking if anyone except rtkit can change /usr/libexec/rtkit-daemon
Checking if anyone except rtkit can change /lib/x86_64-linux-gnu/libdbus-1.so.3
Checking if anyone except rtkit can change /lib/x86_64-linux-gnu/libcap.so.2
Checking if anyone except rtkit can change /lib/x86_64-linux-gnu/libsystemd.so.0
Checking if anyone except rtkit can change /lib/x86_64-linux-gnu/libc.so.6
Checking if anyone except rtkit can change /lib64/ld-linux-x86-64.so.2
Checking if anyone except rtkit can change //freedesktop/DTD
Checking if anyone except rtkit can change /org/freedesktop/DBus
Checking if anyone except rtkit can change /org/freedesktop/PolicyKit1/Authority
Checking if anyone except rtkit can change /org/freedesktop/RealtimeKit1
Checking if anyone except rtkit can change /proc
/usr/bin/unix-privesc-check: 1076: [: detailed: unexpected operator

(kali@kali)-[~]
$

```

Polecenie `sudo unix-privesc-check detailed` uruchamia skrypt `unix-privesc-check v1.4` w trybie `detailed`. W porównaniu z wariantem `standard` skan jest dokładniejszy: obok klasycznych testów SUID/SGID sprawdza też m.in. biblioteki ładowane przez procesy, prawa do katalogów w `PATH`, możliwość podmiany plików konfiguracyjnych demonów oraz uprawnienia do plików urządzeń i skryptów `init/systemd`. Komendę wykonano z `sudo`, więc skrypt może czytać wszystkie katalogi i statystyki procesów bieżącego systemu Kali (kernel 6.8.11, adres 192.168.65.129 w sieci `host-only`). Dodatkowo:

- Skrypt pokazał, że obecna instalacja Kali w VM-ce `host-only` nie zawiera typowych błędów konfiguracyjnych, które `unix-privesc-check` potrafi wykryć.
- Liczne komunikaty „unexpected operator” wynikają wyłącznie z niekompatybilności starego kodu skanera z nowymi powłokami; nie świadczą o problemie bezpieczeństwa.
- `unix-privesc-check` nie jest już rozwijany (ostatnia wersja 2011 r.), dlatego do nowoczesnych systemów warto używać nowszych projektów (`LinPEAS`, `LinEnum`, `pspy`) lub ręcznej analizy SUID, `capabilities` i jednostek `systemd`.

Tak więc, mimo uruchomienia trybu `detailed`, narzędzie nie wykazało podatności – co potwierdza, że nasza maszyna Kali (w roli atakującego) pozostaje poprawnie utwardzona w kontekście lokalnej eskalacji w tym laboratorium sieciowym.

## Podsumowanie

W ramach laboratorium przeprowadziliśmy pełny łańcuch rekonesansu i podstawowej analizy podatności w izolowanej sieci host-only, gdzie naszą stacją ofensywną była Kali Linux (192.168.65.129), a celem – wirtualna maszyna Metasploitable 2 (192.168.65.128). Zaczęliśmy od pasywnego wykrycia hostów przy pomocy netdiscover, a następnie aktywnie potwierdziliśmy je ping-scanem nmapa; w segmencie istniały dwa kluczowe węzły VMware (.1 i .254) oraz sama Metasploitable. Próby szybkiego skanowania masscanem i ręcznego sondowania hping3 pokazały, jak brak trasy lub odłączona karta w trybie host-only blokuje ruch wychodzący – dopóki nie podaliśmy właściwego interfejsu lub nie ograniczyliśmy celu do tej samej podsieci, pakiety nawet nie opuszczały maszyny. Kiedy komunikacja była już możliwa, serwisowo zmapowaliśmy porty: najpierw szybkim „top-10”, później pełnym skanem nmapa z opcjami -sV i -O, a na końcu skanem wszystkich portów (-p-) z włączonymi skryptami NSE. Rezultat potwierdził, że Metasploitable wystawia niemal kompletną kolekcję podatnych usług – vsftpd 2.3.4, Apache 2.2.8 + PHP 5.2.4, Samba 3.0.20, BIND 9.4, stary Tomcat/AJP, backdoorowe root-shell'e, nieszyfrowany telnet i r-serwisy, otwarte bazy MySQL/PostgreSQL, VNC bez autoryzacji i UnrealIRCd z wbudowanym backdoorem. Przy pomocy smbmap pokazaliśmy, że z domyślnymi danymi msfadmin można zalogować się do Samby i mieć pełny zapis w udziale tmp. Weryfikacja połączeniem telnet potwierdziła równie łatwy dostęp typu clear-text do powłoki systemu. Skany Nikto wykazały brak kluczowych nagłówków bezpieczeństwa, włączoną metodę TRACE, listowanie katalogów, publicznego phpinfo.php i odstoniętego phpMyAdmina, a rozszerzone tryby -Tuning i -C/-mutate odnalazły dodatkowo plik config.inc.php z hasłami do MySQL. Po stronie Kali uruchomiliśmy przestarzały skrypt unix-privesc-check w trybach standard i detailed; nie wykazał on podatności, ale wyraźnie zobrazował, że stare narzędzia źle współpracują z nowoczesnym kernelem – stąd liczne błędy składniowe.

### Wnioski:

- Już sama segmentacja sieci host-only chroni przed „ucieczką” ruchu, lecz nie przed atakiem wewnętrznym;
- przestarzałe, niezaktualizowane usługi – szczególnie FTP, Samba, Apache i Tomcat – otwierają dziesiątki wektorów RCE i eskalacji, co pokazała Metasploitable;
- brak szyfrowania (telnet, rsh, FTP) i domyślne hasła umożliwiają natychmiastowe przejęcie systemu bez exploitów;
- dobre praktyki hardeningu to wyłączenie nieużywanych portów, wymuszenie TLS, aktualizacje oprogramowania, ograniczenie udziałów SMB, włączenie SMB-signing, firewall i IDS, a także cykliczne audyty skanerami pokroju Nmap/Nikto oraz nowocześniejszymi narzędziami privesc (LinPEAS);

Cały projekt pokazał zatem kompletny proces: od wykrywania hostów, przez enumerację portów, usługi i udostępnione zasoby, po wstępną ocenę możliwości eskalacji – dzięki czemu mamy pełny obraz atakowalnej powierzchni w kontrolowanym środowisku labowym.