

# Wykrywanie zagrożeń i reakcja na incydenty

## Laboratorium 2

Tomasz Jarząbek 272279  
Wiktoria Migasiewicz 272177

23.03.2025

## Spis treści

<b>1</b>	<b>Opis laboratorium</b>	<b>3</b>
<b>2</b>	<b>Rozwiązania</b>	<b>3</b>
2.1	Konfiguracja maszyn . . . . .	3
2.2	TCPDump – wprowadzenie . . . . .	5
2.3	Przechwycenie pakietów ICMP ze zmianą parametrów . . . . .	6
2.4	Przechwycenie ruchu SSH. . . . .	7
2.5	Tcpdump – tworzenie plików PCAP . . . . .	11
2.6	Konfiguracja narzędzia Wireshark . . . . .	13
2.7	Wireshark – wprowadzenie . . . . .	15

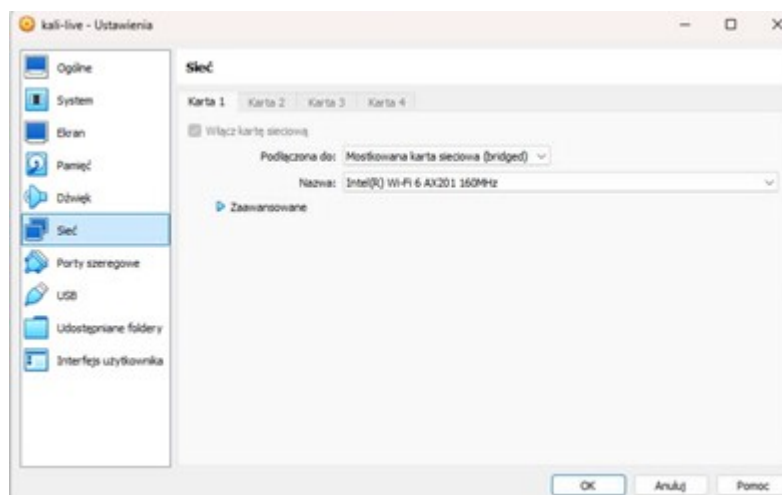
## 1 Opis laboratorium

Laboratorium polegało na użyciu narzędzia TCPDump do przechwytywania ruchu między maszynami wirtualnymi Kali Linux (Kali VM) i Kali Live, które obie były umieszczone we wspólnej podsieci. Przechwycony ruch SSH oraz ICMP zostały zapisane następnie do pliku .pcap i analizowane za pomocą aplikacji Wireshark.

## 2 Rozwiązania

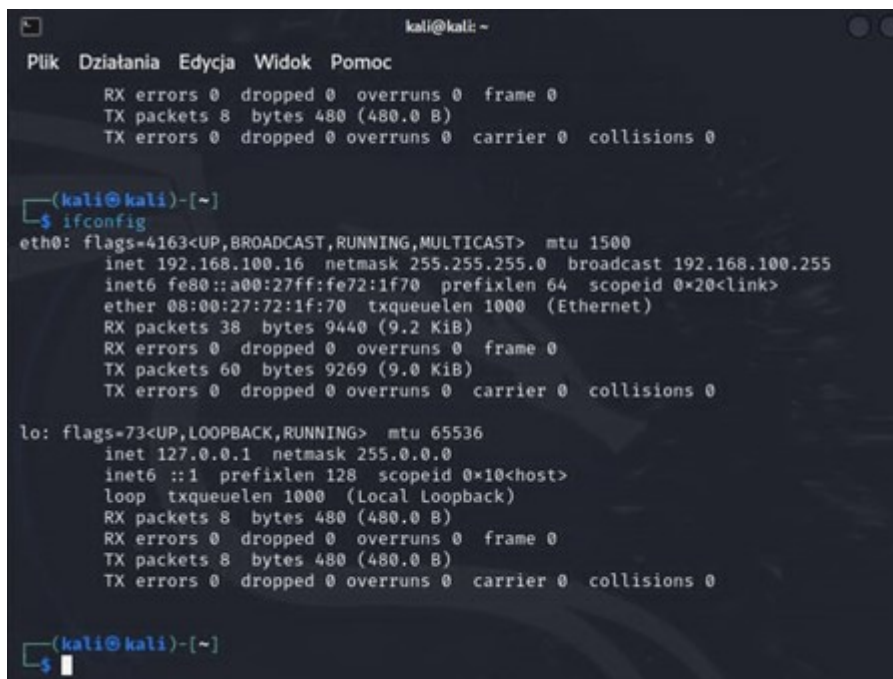
### 2.1 Konfiguracja maszyn

Ustawienia sieciowe maszyn:



Rysunek 1: Ustawienie obu maszyn w jednej wspólnej sieci..

Maszyna kali-linux: *adres IP 192.168.100.16*



```
kali@kali: ~
Plik Działania Edycja Widok Pomoc
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 8 bytes 480 (480.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

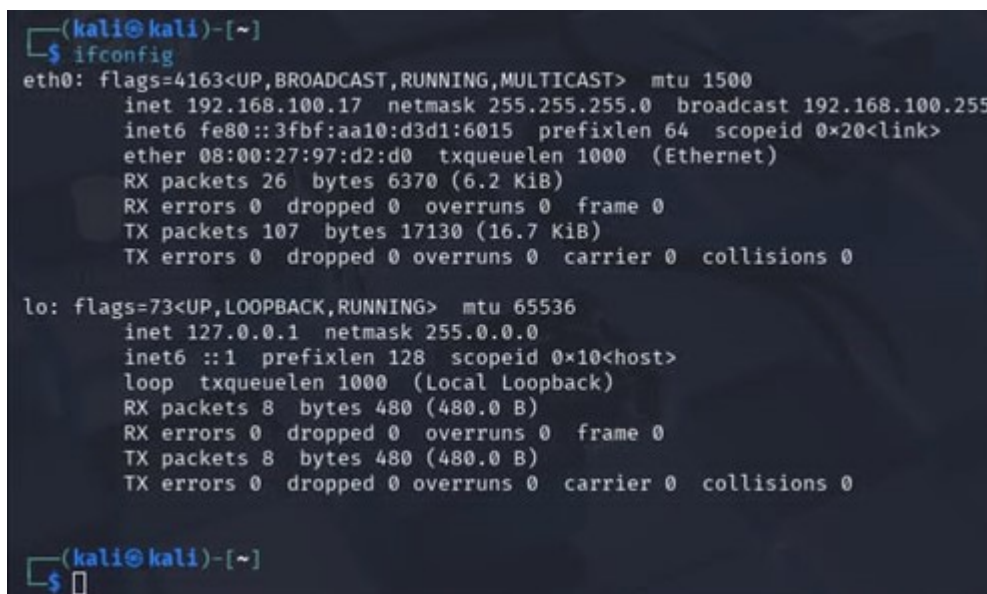
(kali@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.16 netmask 255.255.255.0 broadcast 192.168.100.255
    inet6 fe80::a00:27ff:fe72:1f70 prefixlen 64 scopeid 0<link>
    ether 08:00:27:72:1f:70 txqueuelen 1000 (Ethernet)
    RX packets 38 bytes 9440 (9.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 60 bytes 9269 (9.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 8 bytes 480 (480.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8 bytes 480 (480.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(kali@kali)-[~]
$
```

Rysunek 2: Wynik komendy ip a na Kali VM (kali linux)

Maszyna kali-live: *adres IP 192.168.100.17*



```
(kali@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.17 netmask 255.255.255.0 broadcast 192.168.100.255
    inet6 fe80::3fbf:aa10:d3d1:6015 prefixlen 64 scopeid 0<link>
    ether 08:00:27:97:d2:d0 txqueuelen 1000 (Ethernet)
    RX packets 26 bytes 6370 (6.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 107 bytes 17130 (16.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 8 bytes 480 (480.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8 bytes 480 (480.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(kali@kali)-[~]
$
```

Rysunek 3: Wynik komendy ip a na Kali Live.

Łączność między maszynami:

```
(kali@kali)-[~]  
$ ping 192.168.100.16  
PING 192.168.100.16 (192.168.100.16) 56(84) bytes of data.  
64 bytes from 192.168.100.16: icmp_seq=1 ttl=64 time=6.31 ms  
64 bytes from 192.168.100.16: icmp_seq=2 ttl=64 time=3.08 ms
```

Rysunek 4: Wykazanie łączności między obiema maszynami.

```
(kali@kali)-[~]  
$ ping 192.168.100.17  
PING 192.168.100.17 (192.168.100.17) 56(84) bytes of data.  
64 bytes from 192.168.100.17: icmp_seq=1 ttl=64 time=2.21 ms  
64 bytes from 192.168.100.17: icmp_seq=2 ttl=64 time=2.82 ms
```

Rysunek 5: Wykazanie łączności między obiema maszynami.

## 2.2 TCPDump – wprowadzenie

**Polecenie:** Proszę przy użyciu narzędzia „tcpdump” sprawdzić listę dostępnych interfejsów sieciowych na maszynie „Kali VM”. Wykorzystana komenda: *tcpdump -D*

**Wynik:**

```
(kali@kali)-[~]  
$ tcpdump -D  
1.eth0 [Up, Running, Connected]  
2.any (Pseudo-device that captures on all interfaces) [Up, Running]  
3.lo [Up, Running, Loopback]  
4.bluetooth-monitor (Bluetooth Linux Monitor) [Wireless]  
5.nflog (Linux netfilter log (NFLOG) interface) [none]  
6.nfqueue (Linux netfilter queue (NFQUEUE) interface) [none]  
7.dbus-system (D-Bus system bus) [none]  
8.dbus-session (D-Bus session bus) [none]
```

Rysunek 6: Dostępne interfejsy sieciowe na Kali VM (kali linux).

**Polecenie:** Proszę z wykorzystaniem narzędzia „tcpdump” uruchomić nasłuchiwanie na pakiety ICMP na maszynie „Kali VM”.

Wykorzystana komenda:

*sudo tcpdump -i eth0 icmp*

**Wynik:**

```
(kali@kali)-[~]
└─$ sudo tcpdump -i eth0 icmp
[sudo] hasło użytkownika kali:
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
█
```

Rysunek 7: Nasłuchiwanie na Kali VM na pakiety ICMP (ping).

**Polecenie:** Proszę wysłać pakiety ICMP z maszyny „Kali Live” do maszyny „Kali VM” i zaprezentować przechwycenie pakietów.

Wykorzystane komendy:

*Kali-linux: sudo tcpdump -i eth0 icmp Kali-live: ping 192.168.100.16*

**Wynik na maszynie kali-linux:**

```
(kali@kali)-[~]
└─$ sudo tcpdump -i eth0 icmp
[sudo] hasło użytkownika kali:
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
15:05:32.906509 IP 192.168.100.17 > 192.168.100.16: ICMP echo request, id 2, seq 1, length 64
15:05:32.906832 IP 192.168.100.16 > 192.168.100.17: ICMP echo reply, id 2, seq 1, length 64
15:05:33.917043 IP 192.168.100.17 > 192.168.100.16: ICMP echo request, id 2, seq 2, length 64
15:05:33.917132 IP 192.168.100.16 > 192.168.100.17: ICMP echo reply, id 2, seq 2, length 64
15:05:34.922912 IP 192.168.100.17 > 192.168.100.16: ICMP echo request, id 2, seq 3,
```

Rysunek 8: Otrzymane pakiety z maszyny Kali Live.

## 2.3 Przechwycenie pakietów ICMP ze zmianą parametrów

**Polecenie:** Proszę zmodyfikować parametry „tcpdump” tak, aby:

- złapane pakiety posiadały czas w formacie: „rok-miesiąc-dzień godzina”
- nazwy hostów i portów mają pozostać niezmodyfikowane (postać liczbową)
- złapane pakiety były przedstawione w postaciach HEX i ASCII. Proszę uwzględnić nagłówki warstwy łącza danych.
- pakiety były łapane jedynie na jednym interfejsie
- łapane były tylko pakiety ICMP

Wykorzystana komenda:

`tcpdump -i eth0 icmp -tttt -n -X -e`

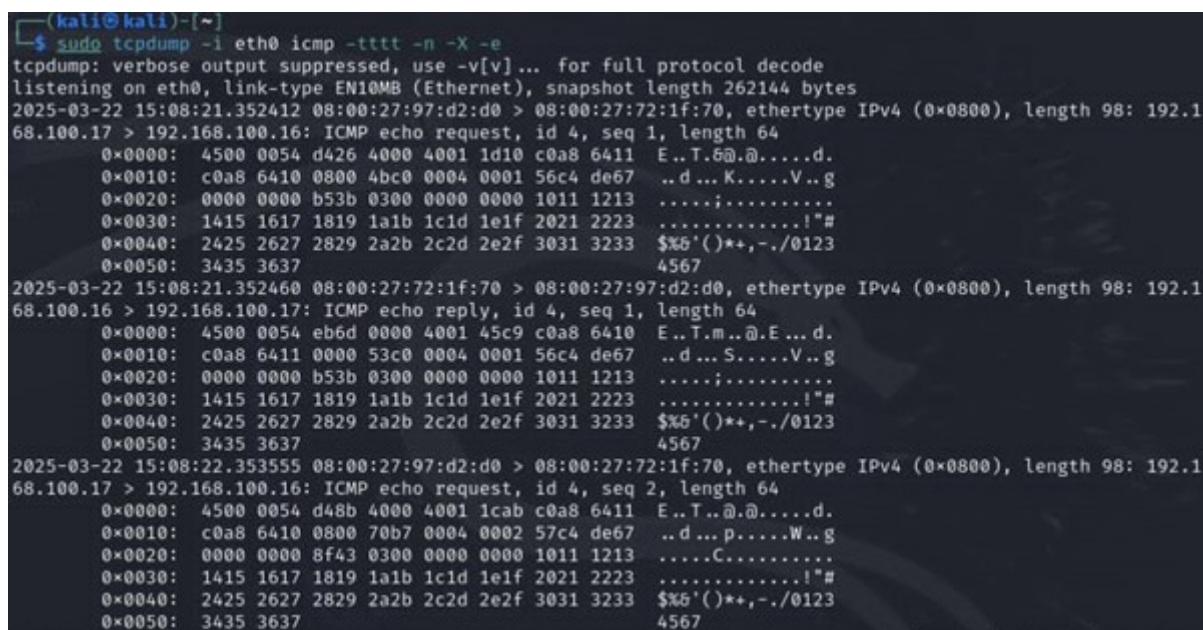
- `-i eth0` → nasłuchiwanie tylko na podanym interfejsie
- `-tttt` → wyświetlanie czasu w formacie "rok-miesiąc-dzień godzina"
- `-n` → pozostawienie adresów IP i portów w postaci liczbowej
- `-X` → wyświetlanie pakietów w HEX + ASCII
- `-e` → uwzględnienie nagłówków warstwy łącza danych
- `icmp` → filtrowanie tylko pakietów ICM

**Polecenie:** Proszę zainicjować odpowiedni ruch i zaprezentować przechwycenie pakietów.

Wykorzystane komendy:

*Kali-linux:* `tcpdump -i eth0 icmp -tttt -n -X -e` *Kali-live:* `ping 192.168.100.16`

Wynik:



```
(kali@kali)-[~]
$ sudo tcpdump -i eth0 icmp -tttt -n -X -e
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
2025-03-22 15:08:21.352412 08:00:27:97:d2:d0 > 08:00:27:72:1f:70, ethertype IPv4 (0x0800), length 98: 192.1
68.100.17 > 192.168.100.16: ICMP echo request, id 4, seq 1, length 64
 0x0000: 4500 0054 d426 4000 4001 1d10 c0a8 6411  E..T.6@.@.....d.
 0x0010: c0a8 6410 0800 4bc0 0004 0001 56c4 de67  ..d...K....V..g
 0x0020: 0000 0000 b53b 0300 0000 0000 1011 1213  ....j.....
 0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!""#
 0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
 0x0050: 3435 3637 4567
2025-03-22 15:08:21.352460 08:00:27:72:1f:70 > 08:00:27:97:d2:d0, ethertype IPv4 (0x0800), length 98: 192.1
68.100.16 > 192.168.100.17: ICMP echo reply, id 4, seq 1, length 64
 0x0000: 4500 0054 eb6d 0000 4001 45c9 c0a8 6410  E..T.m..@.E...d.
 0x0010: c0a8 6411 0000 53c0 0004 0001 56c4 de67  ..d...S....V..g
 0x0020: 0000 0000 b53b 0300 0000 0000 1011 1213  ....j.....
 0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!""#
 0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
 0x0050: 3435 3637 4567
2025-03-22 15:08:22.353555 08:00:27:97:d2:d0 > 08:00:27:72:1f:70, ethertype IPv4 (0x0800), length 98: 192.1
68.100.17 > 192.168.100.16: ICMP echo request, id 4, seq 2, length 64
 0x0000: 4500 0054 d48b 4000 4001 1cab c0a8 6411  E..T..@.@.....d.
 0x0010: c0a8 6410 0800 70b7 0004 0002 57c4 de67  ..d...p....W..g
 0x0020: 0000 0000 8f43 0300 0000 0000 1011 1213  ....C.....
 0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!""#
 0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
 0x0050: 3435 3637 4567
```

Rysunek 9: Przechwycone pakiety ICMP.

## 2.4 Przechwycenie ruchu SSH.

**Polecenie:**

- Proszę zmodyfikować parametry „tcpdump” i uruchomić je na maszynie Kali Linux tak, aby:
- złapane pakiety posiadały czas w formacie: „rok-miesiąc-dzień godzina”

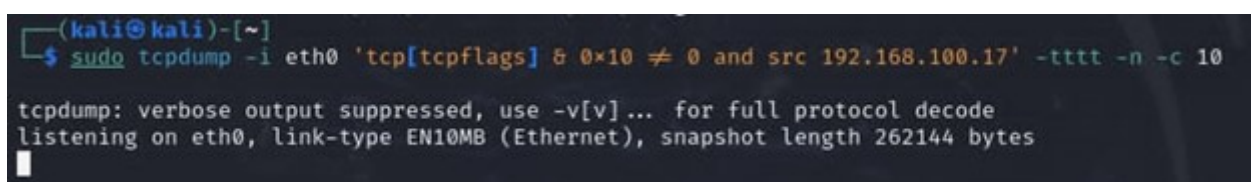


- nazwy hostów i portów mają pozostać niezmodyfikowane (postać liczbową)
- pakiety były łapane jedynie na jednym interfejsie
- złapane zostało jedynie 10 pakietów
- pakiety pochodziły tylko z adresu IP maszyny „Kali Live”
- złapane zostały jedynie pakiety TCP z flagą „ACK”

Wykorzystana komenda:

```
tcpdump -i eth0 'tcp[tcpflags] & 0x10 != 0 and src 192.168.100.17' -tttt -n -c 10
```

Wynik:



```
(kali@kali)-[~]  
$ sudo tcpdump -i eth0 'tcp[tcpflags] & 0x10 != 0 and src 192.168.100.17' -tttt -n -c 10  
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode  
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

Rysunek 10: Wynik rozpoczęcia słuchania ruchu SSH z odpowiednimi flagami.

**Polecenie:** Proszę utworzyć pliki zawierające listy użytkowników oraz haseł. Proszę upewnić się, że listy są stosunkowo krótkie – posiadają maksymalnie kilka rekordów.

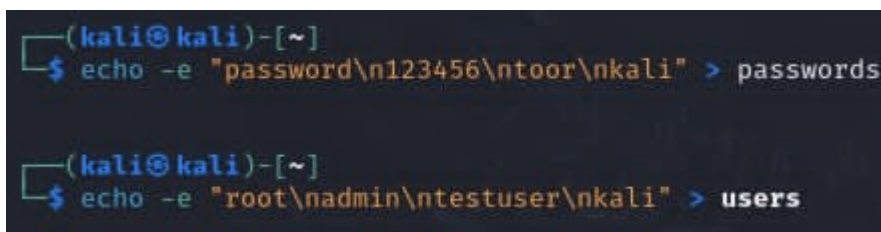
- users
- passwords

Wykorzystane komendy:

```
echo -e "password\n123456\ntoor\nkali" > passwords
```

```
echo -e "root\nadmin\ntestuser\nkali" > users
```

Wynik:



```
(kali@kali)-[~]  
$ echo -e "password\n123456\ntoor\nkali" > passwords  
  
(kali@kali)-[~]  
$ echo -e "root\nadmin\ntestuser\nkali" > users
```

Rysunek 11: Wynik zapisu do plików.

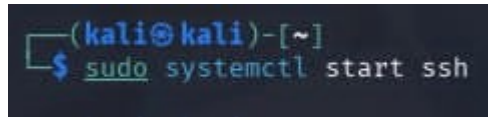
**Polecenie:** Proszę uruchomić usługę SSH wykorzystując komendę „systemctl” na maszynie Kali Linux.



Wykorzystana komenda:

```
sudo systemctl start ssh
```

**Wynik:**



Rysunek 12: Start usługi ssh.

**Polecenie:** Proszę wykonać poniższe polecenie na maszynie Kali Live:

- „nmap -p 22 -script ssh-brute -script-args userdb=users,passdb=passwords [IP]”, gdzie w miejsce „[IP]” proszę podać adres IP maszyny utworzonej w ramach pierwszej instrukcji. Proszę wykorzystać wcześniej zbudowane pliki.

Wykorzystana komenda:

```
nmap -p 22 -script ssh-brute -script-args userdb=users,passdb=passwords 192.168.100.16
```

**Wynik:**

```
(kali@kali)-[~]
$ nmap -p 22 --script ssh-brute --script-args userdb=users,passdb=passwords 192.168.100.16
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-22 14:34 UTC
NSE: [ssh-brute] Trying username/password pair: root:root
NSE: [ssh-brute] Trying username/password pair: admin:admin
NSE: [ssh-brute] Trying username/password pair: testuser:testuser
NSE: [ssh-brute] Trying username/password pair: kali:kali
NSE: [ssh-brute] Trying username/password pair: root:password
NSE: [ssh-brute] Trying username/password pair: admin:password
NSE: [ssh-brute] Trying username/password pair: testuser:password
NSE: [ssh-brute] Trying username/password pair: root:123456
NSE: [ssh-brute] Trying username/password pair: admin:123456
NSE: [ssh-brute] Trying username/password pair: testuser:123456
NSE: [ssh-brute] Trying username/password pair: root:toor
NSE: [ssh-brute] Trying username/password pair: admin:toor
NSE: [ssh-brute] Trying username/password pair: testuser:toor
NSE: [ssh-brute] Trying username/password pair: root:kali
NSE: [ssh-brute] Trying username/password pair: admin:kali
NSE: [ssh-brute] Trying username/password pair: testuser:kali
Nmap scan report for 192.168.100.16
Host is up (0.0012s latency).

PORT      STATE SERVICE
22/tcp    open  ssh
| ssh-brute:
|   Accounts:
|     kali:kali - Valid credentials
|_ Statistics: Performed 16 guesses in 8 seconds, average tps: 2.0
MAC Address: 08:00:27:72:1F:70 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 8.83 seconds

(kali@kali)-[~]
$
```

Rysunek 13: Nmap SSH dla IP Kali Linux na Kali Live.

**Polecenie:** Proszę zaprezentować przechwycenie pakietów w trakcie skanowania usługi SSH.

Wykorzystana komenda:

```
tcpdump -i eth0 'tcp[tcpflags] & 0x10 != 0 and src 192.168.100.17' -tttt -n -c 10
```

**Wynik:**

```

(kali@kali)~$ sudo tcpdump -i eth0 'tcp[tcpflags] & 0x10 & 0 and src 192.168.100.17' -tttt -n -c 10
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
2025-03-22 15:34:11.472391 IP 192.168.100.17.57326 > 192.168.100.16.22: Flags [.], ack 1577359453, win 502, options
[nop,nop,TS val 1009786037 ecr 314758708], length 0
2025-03-22 15:34:11.472968 IP 192.168.100.17.57326 > 192.168.100.16.22: Flags [P.], seq 0:24, ack 1, win 502, optio
ns [nop,nop,TS val 1009786038 ecr 314758708], length 24: SSH: SSH-2.0-libssh2_1.11.1
2025-03-22 15:34:11.525965 IP 192.168.100.17.57326 > 192.168.100.16.22: Flags [.], ack 33, win 502, options [nop,no
p,TS val 1009786091 ecr 314758762], length 0
2025-03-22 15:34:11.527394 IP 192.168.100.17.57326 > 192.168.100.16.22: Flags [P.], seq 24:1664, ack 33, win 502, o
ptions [nop,nop,TS val 1009786093 ecr 314758762], length 1640
2025-03-22 15:34:11.538375 IP 192.168.100.17.57326 > 192.168.100.16.22: Flags [P.], seq 1664:1712, ack 1201, win 52
4, options [nop,nop,TS val 1009786103 ecr 314758771], length 48
2025-03-22 15:34:11.564361 IP 192.168.100.17.57326 > 192.168.100.16.22: Flags [P.], seq 1712:1772, ack 1765, win 53
9, options [nop,nop,TS val 1009786129 ecr 314758798], length 60
2025-03-22 15:34:11.569563 IP 192.168.100.17.57326 > 192.168.100.16.22: Flags [P.], seq 1772:1832, ack 1809, win 53
9, options [nop,nop,TS val 1009786134 ecr 314758802], length 60
2025-03-22 15:34:11.584923 IP 192.168.100.17.57326 > 192.168.100.16.22: Flags [F.], seq 1832, ack 1861, win 539, op
tions [nop,nop,TS val 1009786150 ecr 314758818], length 0
2025-03-22 15:34:11.588318 IP 192.168.100.17.57338 > 192.168.100.16.22: Flags [.], ack 3714728556, win 502, options
[nop,nop,TS val 1009786153 ecr 314758824], length 0
2025-03-22 15:34:11.588319 IP 192.168.100.17.57342 > 192.168.100.16.22: Flags [.], ack 3340275212, win 502, options
[nop,nop,TS val 1009786153 ecr 314758824], length 0
10 packets captured
18 packets received by filter
0 packets dropped by kernel
(kali@kali)~$

```

Rysunek 14: Przechwytywanie pakietów SSH.

## 2.5 Tcpcmdump – tworzenie plików PCAP

**Polecenie:** Proszę zmodyfikować parametry „tcpdump” tak, aby:

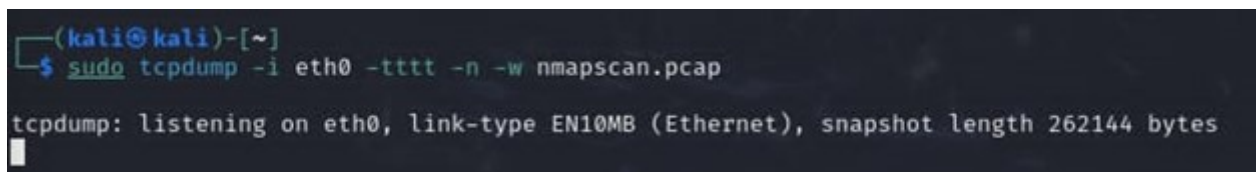
- złapane pakiety posiadały czas w formacie: „rok-miesiąc-dzień godzina”
- nazwy hostów i portów mają pozostać niezmodyfikowane (postać liczbową)
- pakiety były łapane jedynie na jednym interfejsie
- przechwycony ruch został zapisany do pliku „nmapscan.pcap”

Wykorzystana komenda:

```
sudo tcpdump -i eth0 -tttt -n -w nmapscan.pcap
```

- -i eth0 → nasłuchiwanie na interfejsie eth0
- -tttt → format czasu: rok-miesiąc-dzień godzina
- -n → wyświetlanie adresów IP i numerów portów w postaci liczbowej
- -w nmapscan.pcap → zapis przechwyconych pakietów do pliku nmapscan.pcap

**Wynik:**



```
(kali㉿kali)-[~]  
$ sudo tcpdump -i eth0 -tttt -n -w nmapscan.pcap  
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes  
█
```

Rysunek 15: Słuchanie pakietów i zapis do pliku nmapscan.pcap.

**Polecenie:** Proszę ponownie wykonać skanowanie usługi SSH z uruchomionym skryptem „ssh-brute” i zaprezentować przechwycenie pakietów

Wykorzystana komenda:

```
nmap -p 22 -script ssh-brute --script-args userdb=users,passdb=passwords 192.168.100.16
```

**Wynik:**

```
(kali@kali)-[~]
$ nmap -p 22 --script ssh-brute --script-args userdb=users,passdb=passwords 192.168.100.16
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-22 14:37 UTC
NSE: [ssh-brute] Trying username/password pair: root:root
NSE: [ssh-brute] Trying username/password pair: admin:admin
NSE: [ssh-brute] Trying username/password pair: testuser:testuser
NSE: [ssh-brute] Trying username/password pair: kali:kali
NSE: [ssh-brute] Trying username/password pair: root:password
NSE: [ssh-brute] Trying username/password pair: admin:password
NSE: [ssh-brute] Trying username/password pair: testuser:password
NSE: [ssh-brute] Trying username/password pair: root:123456
NSE: [ssh-brute] Trying username/password pair: admin:123456
NSE: [ssh-brute] Trying username/password pair: testuser:123456
NSE: [ssh-brute] Trying username/password pair: root:toor
NSE: [ssh-brute] Trying username/password pair: admin:toor
NSE: [ssh-brute] Trying username/password pair: testuser:toor
NSE: [ssh-brute] Trying username/password pair: root:kali
NSE: [ssh-brute] Trying username/password pair: admin:kali
NSE: [ssh-brute] Trying username/password pair: testuser:kali
Nmap scan report for 192.168.100.16
Host is up (0.0013s latency).

PORT      STATE SERVICE
22/tcp    open  ssh
| ssh-brute:
|   Accounts:
|     kali:kali - Valid credentials
|_ Statistics: Performed 16 guesses in 8 seconds, average tps: 2.0
MAC Address: 08:00:27:72:1F:70 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 8.38 seconds

(kali@kali)-[~]
$
```

Rysunek 16: Ponowne skanowanie SSH z użyciem ssh-brute.



Rysunek 17: Wygenerowany plik .pcap.

## 2.6 Konfiguracja narzędzia Wireshark

**Polecenie:** Proszę otworzyć nmapscan.pcap i zmodyfikować okno programu Wireshark:



- Proszę wyłączyć kolumny „No.”, „Protocol” i „Length”
- Proszę usunąć kolumnę „Length”
- Proszę dodać kolumny „Source Port” i „Destination Port”
- Proszę zmodyfikować kolumnę „Time” tak, by wyświetlała czas w formacie UTC (1970-01-01 01:02:03.123456)
- Proszę zaprezentować wynik końcowy

Wynik:

No.	Time	Source	Destination	Protocol	Length
1	0.000000	192.168.100.10	239.255.255.250	SSDP	212
2	1.002118	192.168.100.10	239.255.255.250	SSDP	212
3	2.002532	192.168.100.10	239.255.255.250	SSDP	212
4	2.524147	PCSSystemtec_97:d2:...	Broadcast	ARP	60
5	2.524238	PCSSystemtec_72:1f:...	PCSSystemtec_97:d2:...	ARP	42
6	2.608735	192.168.100.17	192.168.100.16	TCP	60
7	2.609354	192.168.100.16	192.168.100.17	TCP	58
8	2.610720	192.168.100.17	192.168.100.16	TCP	60
9	2.636673	192.168.100.17	192.168.100.16	TCP	74
10	2.636712	192.168.100.16	192.168.100.17	TCP	74
11	2.637832	192.168.100.17	192.168.100.16	TCP	60

<p>▶ Frame 1: 212 bytes on wire (1696 bits),</p> <p>▶ Ethernet II, Src: Intel_e4:ee:04 (9c:fc:00:00:00:00), Dst: 01:00:5e:7f:ff:fa</p> <p>▶ Internet Protocol Version 4, Src: 192.168.100.10, Dst: 239.255.255.250</p> <p>▶ User Datagram Protocol, Src Port: 58076, Dst Port: 1900</p> <p>▶ Simple Service Discovery Protocol</p>	<pre> 0000  01 00 5e 7f ff fa 9c fc e8 e4 ee 0010  00 c6 92 94 00 00 01 11 11 e6 c0 0020  ff fa e2 dc 07 6c 00 b2 ae 92 4d 0030  43 48 20 2a 20 48 54 54 50 2f 31 0040  4f 53 54 3a 20 32 33 39 2e 32 35 0050  2e 32 35 30 3a 31 39 30 30 0d 0a 0060  22 73 73 64 70 3a 64 69 73 63 6f 0070  0a 4d 58 3a 20 31 0d 0a 53 54 3a 0080  64 69 61 6c 2d 6d 75 6c 74 69 73 0090  2d 6f 72 67 3a 73 65 72 76 69 63 00a0  6c 3a 31 0d 0a 55 53 45 52 2d 41 00b0  20 43 68 72 6f 6d 69 75 6d 2f 31 </pre>
--	--

nmapscan.pcap      Packets: 647      Profile: Default

Rysunek 18: Interfejs Wireshark z nmapscan.pcap przed zmianami.

Time	Source	Destination	Info
2025-03-22 14:37:26.252153	192.168.100.10	239.255.255.250	M-SEARCH * HTTP/1.1
2025-03-22 14:37:27.254271	192.168.100.10	239.255.255.250	M-SEARCH * HTTP/1.1
2025-03-22 14:37:28.254685	192.168.100.10	239.255.255.250	M-SEARCH * HTTP/1.1
2025-03-22 14:37:28.776300	PCSSystemtec_97:d2:...	Broadcast	Who has 192.168.100.16
2025-03-22 14:37:28.776391	PCSSystemtec_72:1f:...	PCSSystemtec_97:d2:...	192.168.100.16 is a
2025-03-22 14:37:28.860888	192.168.100.17	192.168.100.16	37631 → 22 [SYN] Seq=
2025-03-22 14:37:28.861507	192.168.100.16	192.168.100.17	22 → 37631 [SYN, ACK]
2025-03-22 14:37:28.862873	192.168.100.17	192.168.100.16	37631 → 22 [RST] Seq=
2025-03-22 14:37:28.888826	192.168.100.17	192.168.100.16	52052 → 22 [SYN] Seq=
2025-03-22 14:37:28.888865	192.168.100.16	192.168.100.17	22 → 52052 [SYN, ACK]
2025-03-22 14:37:28.889985	192.168.100.17	192.168.100.16	52052 → 22 [ACK] Seq=

Frame 1: 212 bytes on wire (1696 bits), 212 bytes captured (1696 bits) on interface 0	0000	01 00 5e 7f ff fa 9c fc e8 e4 ee 04 08
Ethernet II, Src: Intel_E4:ee:04 (9c:fc:e8:e4:ee:04), Dst: Broadcast (ff:ff:ff:ff:ff:ff)	0010	00 c6 92 94 00 00 01 11 11 e6 c0 a8 64
Internet Protocol Version 4, Src: 192.168.100.10, Dst: 239.255.255.250	0020	ff fa e2 dc 07 6c 00 b2 ae 92 4d 2d 53
User Datagram Protocol, Src Port: 58076, Dst Port: 1900	0030	43 48 20 2a 20 48 54 54 50 2f 31 2e 31
Simple Service Discovery Protocol	0040	4f 53 54 3a 20 32 33 39 2e 32 35 35 2e
	0050	2e 32 35 30 3a 31 39 30 30 0d 0a 4d 41
	0060	22 73 73 64 70 3a 64 69 73 63 6f 76 65
	0070	0a 4d 58 3a 20 31 0d 0a 53 54 3a 20 75
	0080	64 69 61 6c 2d 6d 75 6c 74 69 73 63 72
	0090	2d 6f 72 67 3a 73 65 72 76 69 63 65 3a
	00a0	6c 3a 31 0d 0a 55 53 45 52 2d 41 47 45
	00b0	20 43 68 72 6f 6d 69 75 6d 2f 31 33 31

nmmapscan.pcap Packets: 647 Profile: Default

Rysunek 19: Interfejs Wireshark zmieniony wedle instrukcji.

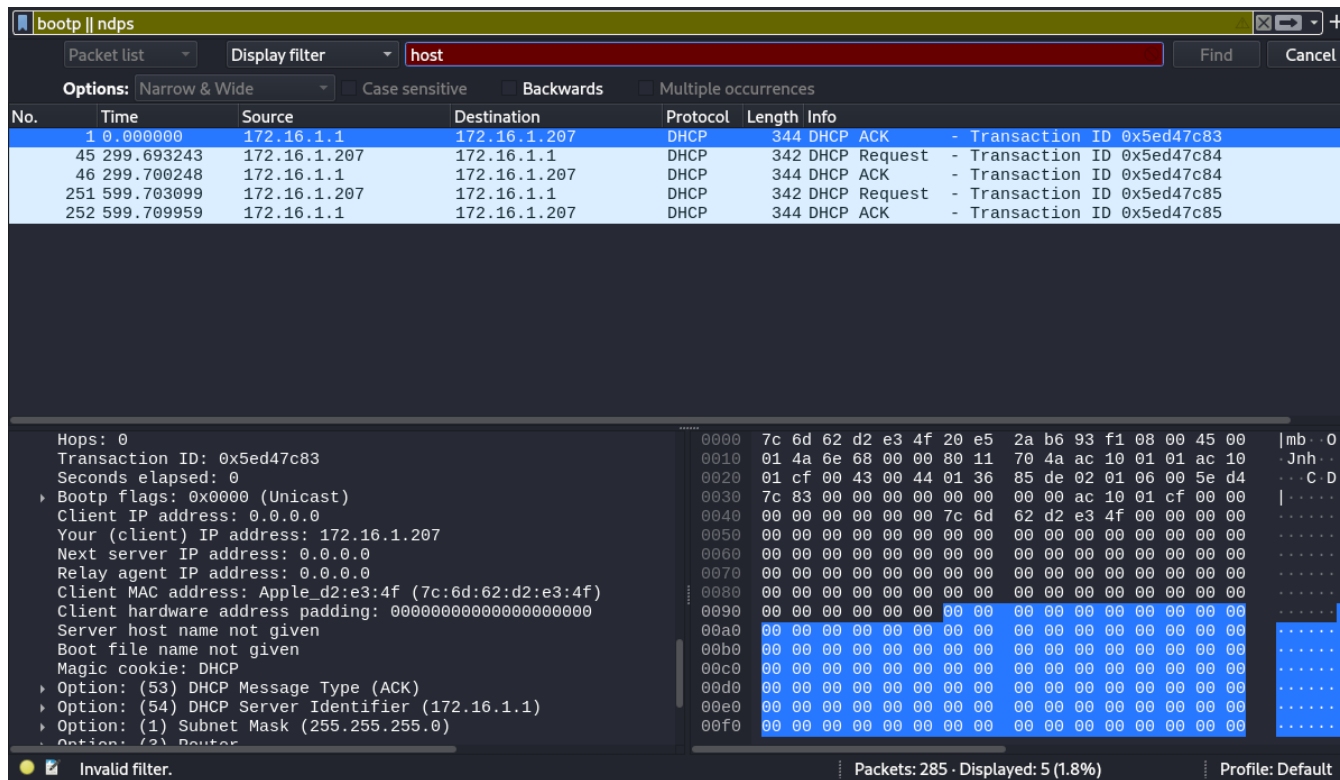
## 2.7 Wireshark – wprowadzenie

### Polecenie:

Proszę w plikach „pcap-01.pcap” i „pcap-02.pcap” odfiltrować wszystkie pakiety DHCP i NBNS, a następnie na podstawie pozyskanych danych zidentyfikować dane hostów: nazwa hosta, adres IP oraz adres MAC.

### Wynik:

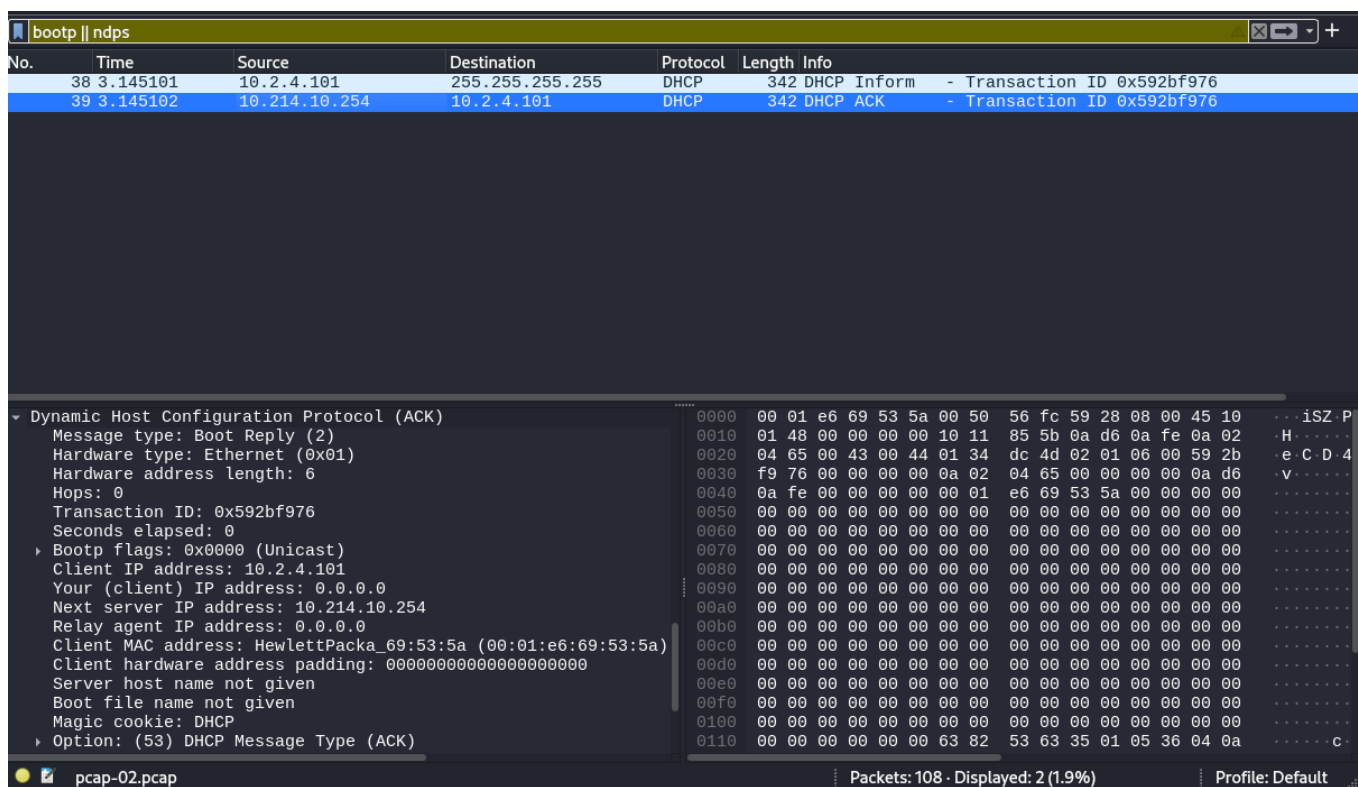




Rysunek 20: Wynik analizy pakietu w pcap-01.pcap.

Jak widać na załączonym zrzucie ekranu (Rysunek 20), szukane parametry w pliku pcap-01.pcap to:

- Nazwa hosta: Apple\_d2:e3:4f
- Adres IP: 172.16.1.207
- Adres MAC: 7c:6d:62:d2:e3:4f



Rysunek 21: Wynik komendy ip neigh.

Jak widać na załączonym zrzucie ekranu (Rysunek 21), szukane parametry w pliku pcap-02.pcap to:

- Nazwa hosta: HewlettPacka\_69:53:5a
- Adres IP: 10.2.4.101
- Adres MAC: 00:01:e6:69:53:5a

Obie nazwy zostały automatycznie wygenerowane na podstawie producenta karty sieciowej oraz fragmentu adresu MAC. **Polecenie:**

Proszę w pliku „pcap-03.pcap” odnaleźć odpowiednie zapytanie http i za pomocą opcji „HTTP stream” zidentyfikować system operacyjny.

**Wynik:**

http.request						
No.	Time	Source	Destination	Protocol	Length	Info
6	0.179611	10.0.0.114	17.253.21.208	HTTP	199	GET /hotspot-detect.html HTTP/1.1
146	2.229101			HTTP	364	GET /MFYwVKADAgEAME0wSzBJM/
153	2.236209			HTTP	368	GET /MFYwVKADAgEAME0wSzBJM/
154	2.236211			HTTP	364	GET /MFYwVKADAgEAME0wSzBJM/
155	2.236442			HTTP	368	GET /MFYwVKADAgEAME0wSzBJM/
407	5.031218			HTTP	366	GET /MFYwVKADAgEAME0wSzBJM/
443	14.382135			HTTP	466	GET /lirr/about/Bicycles/
459	14.485785			HTTP	434	GET /css/base.css HTTP/1.1
471	14.541486			HTTP	445	GET /css/jquery.datepick.cs
478	14.549715			HTTP	434	GET /css/grid.css HTTP/1.1
479	14.549716			HTTP	436	GET /css/topbar.css HTTP/1.1
480	14.549833			HTTP	439	GET /css/formalize.css HTTP/1.1
488	14.611333			HTTP	429	GET /js/jquery-1.4.4.min.js
501	14.617850			HTTP	438	GET /css/template.css HTTP/1.1
502	14.617972			HTTP	438	GET /css/homepage.css HTTP/1.1
503	14.618296			HTTP	442	GET /lirr/assets/lirr.css
545	14.681070			HTTP	430	GET /is/csshorizontalmenu.c
Frame 6: 199 bytes on wire (1592 bits), 199 bytes captured (1592 bits) on interface 0				Follow	HTTP Stream	Ctrl+Alt+Shift+H
Encapsulation type: Ethernet (1)				Copy	TCP Stream	Ctrl+Alt+Shift+T
Arrival Time: Feb 5, 2019 01:23:15.905855000 UTC				Protocol Preferences		
UTC Arrival Time: Feb 5, 2019 01:23:15.905855000 UTC				Decode As...		
Epoch Arrival Time: Feb 5, 2019 01:23:15.905855000 UTC				Show Packet in New Window		
[Time shift for this packet: 0.179611000 seconds]						
[Time delta from previous packet: 0.179611000 seconds]						
[Time delta from previous packet: 0.179611000 seconds]						
[Time since reference or first frame: 0.179611000 seconds]						
Frame Number: 6						
					0030 08 05 b5 a8 00 00 01 01 08 0a	
					0040 0e 18 47 45 54 20 2f 68 6f 74	
					0050 65 74 65 63 74 2e 68 74 6d 6c	
					0060 31 2e 30 0d 0a 48 6f 73 74 3a	
					0070 76 65 2e 61 70 70 6c 65 2e 63	
					0080 6e 6e 65 63 74 69 6f 6e 3a 20	
					0090 0a 55 73 65 72 2d 41 67 65 6e	

Rysunek 22: Włączenie opcji HTTP stream w pakiecie protokołu http.

tcp.stream eq 0			
No.	Time	Source	Destination
3	0.127798	10.0.0.114	17.253.21.208
4	0.175765	17.253.21.208	10.0.0.114
5	0.179195	10.0.0.114	17.253.21.208
6	0.179611	10.0.0.114	17.253.21.208
7	0.232244	17.253.21.208	10.0.0.114
8	0.232716	17.253.21.208	10.0.0.114
9	0.232791	17.253.21.208	10.0.0.114
10	0.235749	10.0.0.114	17.253.21.208
11	0.235828	10.0.0.114	17.253.21.208
115	1.970930	10.0.0.114	17.253.21.208
131	2.021972	17.253.21.208	10.0.0.114
Frame 6: 199 bytes on wire (1592 bits), 199 bytes captured (1592 bits) on interface 0			
Encapsulation type: Ethernet (1)			
Arrival Time: Feb 5, 2019 01:23:15.905855000 UTC			
UTC Arrival Time: Feb 5, 2019 01:23:15.905855000 UTC			

GET /hotspot-detect.html HTTP/1.0	
Host: captive.apple.com	
Connection: close	
User-Agent: CaptiveNetworkSupport-355.200.27 wispr	
HTTP/1.0 200 OK	
x-amz-id-2: wjSTV63JQZ8YVRgLnR8JGp3/gEzYXivcI/gGIXguntnJ9sq6ToI	
I=	
x-amz-request-id: 5F172F820C7D4D8E	
Date: Tue, 05 Feb 2019 01:18:20 GMT	
Last-Modified: Fri, 17 Feb 2017 20:36:28 GMT	
Cache-Control: max-age=300	
Accept-Ranges: bytes	
Content-Type: text/html	
Content-Length: 69	
Server: ATS/8.0.2	
Via: http/1.1 usqas2-edge-lx-002.ts.apple.com (ApacheTrafficServer/8.0.2)	
CDNUUID: 947d7c68-2dfd-4367-bfcd-25cc1ee3e13e-236362256	
X-Cache: hit-fresh, hit-fresh	
Etag: "41ba060eb1c0898e0a4a0cca36a8ca91"	
Age: 296	
<HTML><HEAD><TITLE>Success</TITLE></HEAD><BODY>Success</BODY></	

Rysunek 23: Wynik HTTP stream na badanym pakiecie GET/POST.

System operacyjny został zidentyfikowany jako macOS, ponieważ "CaptiveNetworkSupport" jest charakterystyczne dla urządzeń iPhone.