

# Routing statyczny Cisco

## *Sprawozdanie nr 6 z przedmiotu Sieci Komputerowe*

Maciej Kaszkowiak, 151856, zadania wykonane 3 czerwca 2023

### Spis treści

<b>1</b>	<b>Przy pomocy programu netstat zbadaj, na jakich portach uruchamiane są serwery:</b>	<b>2</b>
<b>2</b>	<b>Zadanie 3</b>	<b>2</b>
2.1	Przeprowadzić analizę komunikacji (Wireshark) na poziomie warstwy transportowej podczas przykładowej transmisji w systemie WWW. . . . .	2
2.2	Określić adresy IPv4 i numery portów (klienta i serwera) używanych podczas transmisji. . . . .	2
2.3	Przeprowadzić analizę segmentów TCP przesyłanych w trakcie nawiązywania połączenia między klientem a serwerem HTTP. Określić dla każdego segmentu: . . .	3
2.4	Przeprowadzić analizę segmentów TCP przesyłanych w trakcie przesyłania zawartości strony download.html między serwerem HTTP a klientem. Określić dla każdego segmentu: . . . . .	3
2.5	Przeprowadzić analizę segmentów TCP przesyłanych w trakcie rozłączania. Określić dla każdego segmentu: . . . . .	4

## 1 Przy pomocy programu netstat zbadaj, na jakich portach uruchamiane są serwery:

- HTTP - 80
- HTTPS - 443
- FTP - 21
- telnet - 23
- DNS - 53

Oraz kilka dodatkowych:

- ssh - 22
- PostgreSQL - 5432
- Redis - 6379
- MongoDB - 27019

Zauważyłem również, że Google Chrome jako *klient* ma losowo przydzielane porty.

## 2 Zadanie 3

### 2.1 Przeprowadzić analizę komunikacji (Wireshark) na poziomie warstwy transportowej podczas przykładowej transmisji w systemie WWW.

Załadowałem plik http.cap do programu Wireshark.

### 2.2 Określić adresy IPv4 i numery portów (klienta i serwera) używanych podczas transmisji.

W przypadku połączenia o download.html, IPv4 serwera to 65.208.228.223, zaś IP klienta to 145.254.160.237. Port serwera to 80 (HTTP), a port klienta (przeglądarki) to 3372. Możemy zauważyć również połączenie na porcie 3371 - jest to drugie zapytanie ujęte w pliku http.cap, odpytujące o reklamy.

Source	Destination	Protocol	Length	Info
145.254.160.237	65.208.228.223	TCP	62	3372 → 80
65.208.228.223	145.254.160.237	TCP	62	80 → 3372
145.254.160.237	65.208.228.223	TCP	54	3372 → 80
145.254.160.237	65.208.228.223	HTTP	533	GET /downl
65.208.228.223	145.254.160.237	TCP	54	80 → 3372
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372
145.254.160.237	65.208.228.223	TCP	54	3372 → 80
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372
145.254.160.237	65.208.228.223	TCP	54	3372 → 80
145.254.160.237	145.253.2.203	DNS	89	Standard q
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372
145.254.160.237	65.208.228.223	TCP	54	3372 → 80
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372
145.253.2.203	145.254.160.237	DNS	188	Standard q
145.254.160.237	216.239.59.99	HTTP	775	GET /pagea

Rysunek 1: Adresy oraz nr portów

## 2.3 Przeprowadzić analizę segmentów TCP przesyłanych w trakcie nawiązywania połączenia między klientem a serwerem HTTP. Określić dla każdego segmentu:

- kierunek transmisji
- numery portów (źródłowego, docelowego)
- znaczniki
- wartość pola sequence number
- wartość pola acknowledge number
- wartość pola window size
- wartości opcji

Source	Destination	Protocol	Length	Info
145.254.160.237	65.208.228.223	TCP	62	3372 → 80 [SYN] Seq=0 win=8760 Len=0 MSS=1460 SACK_PERM=1
65.208.228.223	145.254.160.237	TCP	62	80 → 3372 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1380 SACK_PERM=1
145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=1 Ack=1 Win=9660 Len=0

Rysunek 2: Nawiązywanie połączenia pomiędzy klientem a serwerem HTTP

Na rzucie ekranu możemy zaobserwować TCP handshake w postaci trzech pakietów (SYN, SYN ACK, ACK). Pakiet SYN trafia od klienta do serwera, SYN ACK od serwera do klienta, zaś ACK ponownie od klienta do serwera. Po wymianie pakietów transmisja zostaje nawiązana.

Poniżej zamieściłem informacje z Wiresharka. Przykładowo, pierwszy pakiet informuje o przesyłaniu danych z adresu 145.254.160.237:3372 na adres 65.208.228.223:80, ze znacznikiem SYN, z polem sequence number o wartości 0, bez wartości acknowledge number, z polem window size równym 8760, z opcjami MSS = 1460 (maksymalny rozmiar segmentu) oraz SACK PERM = 1.

```
145.254.160.237 65.208.228.223 TCP 62 3372 → 80 [SYN] Seq=0
Win=8760 Len=0 MSS=1460 SACK_PERM=1
```

```
65.208.228.223 145.254.160.237 TCP 62 80 → 3372 [SYN, ACK] Seq=0 Ack=1
Win=5840 Len=0 MSS=1380 SACK_PERM=1
```

```
145.254.160.237 65.208.228.223 TCP 54 3372 → 80 [ACK] Seq=1 Ack=1
Win=9660 Len=0
```

Rysunek 3: Przedruk z Wiresharka

Poniżej załączam przedruk z RFC odnośnie opcji SACK\_PERM:

### 2. Sack-Permitted Option

This two-byte option may be sent in a SYN by a TCP that has been extended to receive (and presumably process) the SACK option once the connection has opened. It MUST NOT be sent on non-SYN segments.

Rysunek 4: RFC 1818: TCP Selective Acknowledgment Options

## 2.4 Przeprowadzić analizę segmentów TCP przesyłanych w trakcie przesyłania zawartości strony download.html między serwerem HTTP a klientem. Określić dla każdego segmentu:

- kierunek transmisji
- numery portów (źródłowego, docelowego)

- znaczniki
- wartość pola sequence number
- wartość pola acknowledgement number
- wielkość pola danych.

Source	Destination	Protocol	Length	Info
145.254.160.237	65.208.228.223	TCP	62	3372 → 80 [SYN] Seq=0 Win=8760 Len=0 MSS=1460 SACK_PERM=1
65.208.228.223	145.254.160.237	TCP	62	80 → 3372 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1380 SACK_PERM=1
145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=1 Ack=1 Win=9660 Len=0

Rysunek 5: Połączenie HTTP pomiędzy klientem a serwerem HTTP

145.254.160.237	65.208.228.223	TCP	62	3372 → 80 [SYN] Seq=0 Win=8760 Len=0 MSS=1460 SACK_PERM=1
65.208.228.223	145.254.160.237	TCP	62	80 → 3372 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1380 SACK_PERM=1
145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=1 Ack=1 Win=9660 Len=0
145.254.160.237	65.208.228.223	HTTP	533	GET /download.html HTTP/1.1
65.208.228.223	145.254.160.237	TCP	54	80 → 3372 [ACK] Seq=1 Ack=480 Win=6432 Len=0
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=1 Ack=480 Win=6432 Len=1380 [TCP Reset]
145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=1381 Win=9660 Len=0
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=1381 Ack=480 Win=6432 Len=1380
145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=2761 Win=9660 Len=0
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=2761 Ack=480 Win=6432 Len=1380
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [PSH, ACK] Seq=4141 Ack=480 Win=6432 Len=1380
145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=5521 Win=9660 Len=0
145.254.160.237	145.253.2.203	DNS	89	Standard query 0x0023 A pagead2.googleusercontent.com
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=5521 Ack=480 Win=6432 Len=1380
145.254.160.237	65.208.228.223	TCP	54	3372 → 80 [ACK] Seq=480 Ack=6901 Win=9660 Len=0
65.208.228.223	145.254.160.237	TCP	1434	80 → 3372 [ACK] Seq=6901 Ack=480 Win=6432 Len=1380
145.253.2.203	145.254.160.237	DNS	188	Standard query response 0x0023 A pagead2.googleusercontent.com

  

```

Header Checksum: 0x9010 [validation disabled]
[Header checksum status: Unverified]
Source Address: 145.254.160.237
Destination Address: 65.208.228.223
✓ Transmission Control Protocol, Src Port: 3372, Dst Port: 80, Seq: 1, Ack: 1, Len: 479
  Source Port: 3372
  Destination Port: 80
  [Stream index: 0]
  [Conversation completeness: Complete, WITH_DATA (31)]
  [TCP Segment Len: 479]
  Sequence Number: 1 (relative sequence number)
  Sequence Number (raw): 951057940
  [Next Sequence Number: 480 (relative sequence number)]
  Acknowledgment Number: 1 (relative ack number)
  Acknowledgment number (raw): 290218380
  0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x018 (PSH, ACK)
  Window: 9660
  [Calculated window size: 9660]
  [Window size scaling factor: -2 (no window scaling used)]

```

Rysunek 6: Żądanie HTTP

Na zrzutach ekranu możemy zaobserwować, że początkowo przesyłany jest pakiet z flagami PSH/ACK z żądaniem GET o pobranie pliku. W odpowiedzi serwer wysyła pakiety z danymi oraz flagą ACK. Klient zaś potwierdza każdy pakiet odpowiadając pakietem ACK bez danych. Wartości sequence number rosną wraz z przesyłem danych, a acknowledgement number odpowiada odpowiednim polom sequence number. Adresy IP oraz numery portów pozostają bez zmian. IPv4 serwera WWW to 65.208.228.223, zaś IP klienta (przeglądarki) to 145.254.160.237. Port serwera to 80, a port klienta to 3372.

## 2.5 Przeprowadzić analizę segmentów TCP przesyłanych w trakcie rozłączania. Określić dla każdego segmentu:

- kierunek transmisji
- numery portów (źródłowego, docelowego)
- znaczniki
- wartość pola sequence number
- wartość pola acknowledge number.

65.208.228.223	145.254.160.237	TCP	54 80 → 3372 [FIN, ACK] Seq=18365 Ack=480 Win=6432 Len=0
145.254.160.237	65.208.228.223	TCP	54 3372 → 80 [ACK] Seq=480 Ack=18366 Win=9236 Len=0
145.254.160.237	65.208.228.223	TCP	54 3372 → 80 [FIN, ACK] Seq=480 Ack=18366 Win=9236 Len=0
65.208.228.223	145.254.160.237	TCP	54 80 → 3372 [ACK] Seq=18366 Ack=481 Win=6432 Len=0

  

...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 47
Protocol: TCP (6)
Header Checksum: 0x3187 [validation disabled]
[Header checksum status: Unverified]
Source Address: 65.208.228.223
Destination Address: 145.254.160.237
✓ Transmission Control Protocol, Src Port: 80, Dst Port: 3372, Seq: 18365, Ack: 480, Len: 0
Source Port: 80
Destination Port: 3372
[Stream index: 0]
[Conversation completeness: Complete, WITH_DATA (31)]
[TCP Segment Len: 0]
Sequence Number: 18365 (relative sequence number)
Sequence Number (raw): 290236744
[Next Sequence Number: 18366 (relative sequence number)]
Acknowledgment Number: 480 (relative ack number)
Acknowledgment number (raw): 951058419
0101 .... = Header Length: 20 bytes (5)
✓ Flags: 0x011 (FIN, ACK)

  

0000	00 00 01 00 00 00 fe ff	20 00 01 00 08 00 45 00	.....E.
0010	00 28 c0 ad 40 00 2f 06	31 87 41 d0 e4 df 01 f6	(. @ ./ 1 A ...
0020	00 50 0d 2c 11 4c	a9 48 38 af ff f3 50 11	P , L H8 ...P
0030	19 20 3c 64 00 00		<d ..

Rysunek 7: Segmenty TCP

Możemy zaobserwować, że koniec transmisji odbywa się zgodnie z następującym schematem: Klient przesyła do serwera pakiet FIN ACK, na co serwer odpowiada pakietem ACK. Następnie serwer również potwierdza zakończenie transmisji, przysyłając pakiet FIN ACK, na co klient odpowiada pakietem ACK. Adresy IP oraz numery portów pozostają bez zmian. IPv4 serwera WWW to 65.208.228.223, zaś IP klienta (przeglądarki) to 145.254.160.237. Port serwera to 80, a port klienta to 3372.