

Warstwa sieciowa

Sprawozdanie nr 4 z przedmiotu Sieci Komputerowe

Maciej Kaszkowiak, 151856, zadania wykonane 13 maja 2023

Spis treści

1	Zadanie 1	2
1.1	Zaloguj ruch przy pomocy programu wireshark. Jakie zazwyczaj są ustawione flagi w pakietach?	2
1.2	Powtórz ćwiczenie otwierając w międzyczasie jakąś stronę internetową przy użyciu protokołu https.	2
1.3	Powtórz ćwiczenie w międzyczasie wykonując polecenie ping -s 4000 na wybrany adres IP.	2
2	Zadanie 2	3
2.1	Jak działa program traceroute (ewentualnie skorzystaj z opcji -I lub -T)?	3
2.2	Zaloguj ruch przy pomocy programu wireshark i zbadaj nagłówki pakietów generowanych przez program traceroute.	4
3	Zadanie 3	4
3.1	Podłącz swój komputer (poprzez port p4p1) do koncentratora (na zapleczu).	4
3.2	Skonfiguruj interfejs p4p1, tak by wszystkie komputery w rzędzie działały w jednej sieci (unikalne sieci między rzędami).	4
3.3	Zbadaj jak zmienia się tablica ARP, gdy uruchamiasz program ping z argumentami będącymi adresami IP komputerów z Twojej sieci i adresami komputerów w innych rzędach (należących do innych sieci).	5

1 Zadanie 1

1.1 Zaloguj ruch przy pomocy programu wireshark. Jakiej zazwyczaj są ustawione flagi w pakietach?

Możemy zaobserwować 2 flagi w pakietach: "Don't fragment" oraz "More fragments".

```

Internet Protocol Version 4, Src: 150.254.32.68, Dst: 35.241.46.245
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 52
    Identification: 0x7fb8 (32696)
  ✓ Flags: 0x40, Don't fragment
    0... .... = Reserved bit: Not set
    .1.. .... = Don't fragment: Set
    ..0. .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: TCP (6)
    Header Checksum: 0xb0e3 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 150.254.32.68
    Destination Address: 35.241.46.245
  > Transmission Control Protocol, Src Port: 53314, Dst Port: 443, Seq: 1, Ack: 1, Len: 0

```

Rysunek 1: Flagi w pakietach

1.2 Powtórz ćwiczenie otwierając w międzyczasie jakąś stronę internetową przy użyciu protokołu https.

Możemy zauważyć, że flaga Don't fragment w pakiecie została wyłączona.

34.120.52.64	150.254.32.68	TCP	66 443 → 53840 [ACK] Seq=400 Ack=955 Win=409 Len=0 TSval=2016496352 TSecr=3666692685
34.120.52.64	150.254.32.68	TLSv1.2	105 Application Data
150.254.32.68	34.120.52.64	TCP	66 53840 → 443 [ACK] Seq=955 Ack=439 Win=2875 Len=0 TSval=3666692613 TSecr=2016496352
34.120.52.64	150.254.32.68	TLSv1.2	344 Application Data
150.254.32.68	34.120.52.64	TCP	66 53840 → 443 [ACK] Seq=955 Ack=717 Win=2873 Len=0 TSval=3666692782 TSecr=2016496521
34.120.52.64	150.254.32.68	TLSv1.2	432 Application Data
150.254.32.68	34.120.52.64	TCP	66 53840 → 443 [ACK] Seq=955 Ack=1083 Win=2871 Len=0 TSval=3666692782 TSecr=2016496521
34.120.52.64	150.254.32.68	TLSv1.2	140 Application Data
150.254.32.68	34.120.52.64	TCP	66 53840 → 443 [ACK] Seq=955 Ack=1157 Win=2871 Len=0 TSval=3666692782 TSecr=2016496521
34.120.52.64	150.254.32.68	TLSv1.2	105 Application Data
150.254.32.68	34.120.52.64	TCP	66 53840 → 443 [ACK] Seq=955 Ack=1196 Win=2871 Len=0 TSval=3666692782 TSecr=2016496521

```

Ethernet II, Src: ExtremeN_9b:f0 (00:04:96:9b:f0), Dst: Dell_a5:98:cc (e4:54:e8:a5:98:cc)
Internet Protocol Version 4, Src: 34.120.52.64, Dst: 150.254.32.68
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 52
    Identification: 0x257c (9596)
  ✓ Flags: 0x00
    0... .... = Reserved bit: Not set
    .0.. .... = Don't fragment: Not set
    ..0. .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 121
    Protocol: TCP (6)
    Header Checksum: 0x0e4e [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 34.120.52.64
    Destination Address: 150.254.32.68
  > Transmission Control Protocol, Src Port: 443, Dst Port: 53840, Seq: 400, Ack: 955, Len: 0

```

Rysunek 2: Flagi pozwalające na fragmentację pakietu

1.3 Powtórz ćwiczenie w międzyczasie wykonując polecenie ping -s 4000 na wybrany adres IP.

Możemy zauważyć, że flaga Don't fragment w pakiecie jest wyłączona, a flaga More fragments w niektórych pakietach jest aktywna.

```

✓ Internet Protocol Version 4, Src: 150.254.32.68, Dst: 75.2.92.173
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 1500
  Identification: 0x3632 (13874)
  ✓ Flags: 0x20, More fragments
    0... .... = Reserved bit: Not set
    .0.. .... = Don't fragment: Not set
    ..1. .... = More fragments: Set
  ...0 0000 1011 1001 = Fragment Offset: 1480
  Time to Live: 64
  Protocol: ICMP (1)
  Header Checksum: 0xbf44 [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 150.254.32.68
  Destination Address: 75.2.92.173
  [Reassembled IPv4 in frame: 145]

```

Rysunek 3: Flagi w pofragmentowanym pakiecie

2 Zadanie 2

2.1 Jak działa program traceroute (ewentualnie skorzystaj z opcji -I lub -T)?

Program traceroute opiera się na mechanizmie TTL w celu wyznaczenia poszczególnych węzłów na ścieżce pakietu wiodącego do wskazanego przez nas adresu IP. Program nasłuchuje na odpowiedzi ICMP informujące o porzuceniu pakietu ze względu na wyzerowanie jego wartości TTL.

TRACEROUTE(8)	Traceroute For Linux	TRACEROUTE(8)
NAME traceroute - print the route packets trace to network host		
SYNOPSIS traceroute [-46dFITUnreAV] [-f first_ttl] [-g gate,...] [-i device] [-m max_ttl] [-p port] [-s src_addr] [-q nqueries] [-N squeries] [-t tos] [-l flow_label] [-w waittime] [-z sendwait] [-UL] [-D] [-P proto] [--sport=port] [-M method] [-O mod_options] [--mtu] [--back] host [packet_len] traceroute6 [options]		
DESCRIPTION <u>traceroute</u> tracks the route packets taken from an IP network on their way to a given host. It utilizes the IP protocol's time to live (TTL) field and attempts to elicit an ICMP TIME_EXCEEDED response from each gateway along the path to the host. <u>traceroute6</u> is equivalent to <u>traceroute</u> -6		
Manual page traceroute(8) line 1 (press h for help or q to quit)		

Rysunek 4: Manpages programu traceroute

2.2 Zaloguj ruch przy pomocy programu wireshark i zbadaj nagłówki pakietów generowanych przez program traceroute.

```
student@lab-sec-3:~> traceroute onet.pl
traceroute to onet.pl (99.83.207.202), 30 hops max, 60 byte packets
 1 150.254.32.65 (150.254.32.65) 1.528 ms 1.585 ms 1.681 ms
 2 150.254.30.129 (150.254.30.129) 1.044 ms 1.263 ms 1.501 ms
 3 * * *
```

Rysunek 5: Wykonana komenda

Program generuje pakiety UDP z rosnącymi wartościami TTL, począwszy od TTL = 1. W odpowiedzi otrzymywane są pakiety ICMP informujące o porzuceniu pakietu.

150.254.32.68	99.83.207.202	UDP	74 53698 → 33434 Len=32
150.254.32.68	99.83.207.202	UDP	74 46512 → 33435 Len=32
150.254.32.68	99.83.207.202	UDP	74 44200 → 33436 Len=32
150.254.32.68	99.83.207.202	UDP	74 36502 → 33437 Len=32
150.254.32.68	99.83.207.202	UDP	74 34130 → 33438 Len=32
150.254.32.68	99.83.207.202	UDP	74 39615 → 33439 Len=32
150.254.32.68	99.83.207.202	UDP	74 49051 → 33440 Len=32
150.254.32.68	99.83.207.202	UDP	74 50359 → 33441 Len=32
150.254.32.68	99.83.207.202	UDP	74 43067 → 33442 Len=32
150.254.32.68	99.83.207.202	UDP	74 51700 → 33443 Len=32
150.254.32.68	99.83.207.202	UDP	74 51001 → 33444 Len=32
150.254.32.68	99.83.207.202	UDP	74 48670 → 33445 Len=32
150.254.32.68	99.83.207.202	UDP	74 46108 → 33446 Len=32
150.254.32.68	99.83.207.202	UDP	74 42997 → 33447 Len=32

> Frame 13: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface br0, id 0	
> Ethernet II, Src: Dell_a5:98:cc (e4:54:e8:a5:98:cc), Dst: ExtremeN_9b:9b:f0 (00:04:96:9b:9b:f0)	
✓ Internet Protocol Version 4, Src: 150.254.32.68, Dst: 99.83.207.202	
0100 = Version: 4	
.... 0101 = Header Length: 20 bytes (5)	
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)	
Total Length: 60	
Identification: 0xa555 (42325)	
✓ Flags: 0x00	
0... = Reserved bit: Not set	
.0.. = Don't fragment: Not set	
..0. = More fragments: Not set	
...0 0000 0000 0000 = Fragment Offset: 0	
> Time to Live: 3	
Protocol: UDP (17)	
Header Checksum: 0x27fc [validation disabled]	
[Header checksum status: Unverified]	
Source Address: 150.254.32.68	
Destination Address: 99.83.207.202	
> User Datagram Protocol, Src Port: 43067, Dst Port: 33442	

Rysunek 6: Nagłówki pakietów generowanych przez program traceroute

3 Zadanie 3

3.1 Podłącz swój komputer (poprzez port p4p1) do koncentratora (na zapleczu).

Podłączyłem swój komputer poprzez port o numerze 99 i interfejs p4p1.

3.2 Skonfiguruj interfejs p4p1, tak by wszystkie komputery w rzędzie działały w jednej sieci (unikalne sieci między rzędami).

Połączenie działa:

```
lab-sec-3:/homex/student # ip addr add 192.168.1.3/24 dev p4p1
lab-sec-3:/homex/student # ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=1.22 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.634 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.704 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=64 time=0.496 ms
```

Rysunek 7: Wykonana komenda

3.3 Zbadaj jak zmienia się tablica ARP, gdy uruchamiasz program ping z argumentami będącymi adresami IP komputerów z Twojej sieci i adresami komputerów w innych rzędach (należących do innych sieci).

Możemy zaobserwować, że tablica ARP uzupełnia się po wykonaniu pingu do komputera w naszej sieci.

```
lab-sec-3:/homex/student # arp -d 192.168.1.2
lab-sec-3:/homex/student # arp -n
```

Address	HWtype	HWaddress	Flags	Mask	Iface
150.254.32.65	ether	00:04:96:9b:9b:f0	C		br0
150.254.32.120	ether	52:54:00:7d:97:53	C		br0
192.168.1.1	ether	e4:54:e8:a5:98:c6	C		br0
192.168.1.1		(incomplete)			p4p1
150.254.32.126	ether	00:25:64:3b:c1:d0	C		br0

```
lab-sec-3:/homex/student # ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=1.44 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.774 ms
^C
--- 192.168.1.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.774/1.104/1.435/0.330 ms
lab-sec-3:/homex/student # arp -n
```

Address	HWtype	HWaddress	Flags	Mask	Iface
150.254.32.65	ether	00:04:96:9b:9b:f0	C		br0
192.168.1.2	ether	00:10:18:b4:e0:24	C		p4p1
150.254.32.120	ether	52:54:00:7d:97:53	C		br0
192.168.1.1	ether	e4:54:e8:a5:98:c6	C		br0
192.168.1.1		(incomplete)			p4p1
150.254.32.126	ether	00:25:64:3b:c1:d0	C		br0

Rysunek 8: Wykonana komenda

Komputery w innych rzędach są nieosiągalne, przez co tablica ARP się nie uzupełnia - nie znamy adresów MAC nieosiągalnych węzłów.

```
lab-sec-3:/homex/student # ping 192.168.5.14
PING 192.168.5.14 (192.168.5.14) 56(84) bytes of data.
From 150.254.4.66 icmp_seq=6 Destination Net Unreachable
^C
--- 192.168.5.14 ping statistics ---
9 packets transmitted, 0 received, +1 errors, 100% packet loss, time 8168ms

lab-sec-3:/homex/student # ping 192.168.5.15
PING 192.168.5.15 (192.168.5.15) 56(84) bytes of data.
From 150.254.4.66 icmp_seq=6 Destination Net Unreachable
From 150.254.4.66 icmp_seq=7 Destination Net Unreachable
^C
--- 192.168.5.15 ping statistics ---
7 packets transmitted, 0 received, +2 errors, 100% packet loss, time 6129ms

lab-sec-3:/homex/student # arp -n
Address                HWtype  HWaddress           Flags Mask            Iface
150.254.32.65          ether    00:04:96:9b:9b:f0    C                     br0
192.168.1.2             ether    00:10:18:b4:e0:24    C                     p4p1
150.254.32.120         ether    52:54:00:7d:97:53    C                     br0
192.168.1.1             ether    e4:54:e8:a5:98:c6    C                     br0
192.168.1.1             ether    00:10:18:aa:bd:7c    C                     p4p1
150.254.32.126         ether    00:25:64:3b:c1:d0    C                     br0
```

Rysunek 9: Wykonana komenda

Zadanie 4 nie zostało wykonane zgodnie z poleceniem prowadzącego.