

Instrukcja programu Wireshark (wersja 1.8.3) w zakresie TCP/IP

I. Na początek

Czym jest analizator sieciowy jakim jest Wireshark?

Analizator sieciowy pozwala na przechwytywanie i analizę danych, które są przesyłane przez sieć. W łatwy sposób możemy "obejrzeć" cały ruch w naszej sieci. Możemy sprawdzić jakie porty wykorzystuje dane oprogramowanie oraz wysyłany/odbierany ruch przez nasz komputer.

Wireshark jest najbardziej popularnym analizatorem sieciowym. To silne narzędzie dostarcza sieciowe i wyższych protokołów informacyjnych dotyczące zrzuconych danych w sieci.

Do roku 2006 Wireshark nosił nazwę Ethereal.

Ważna uwaga:

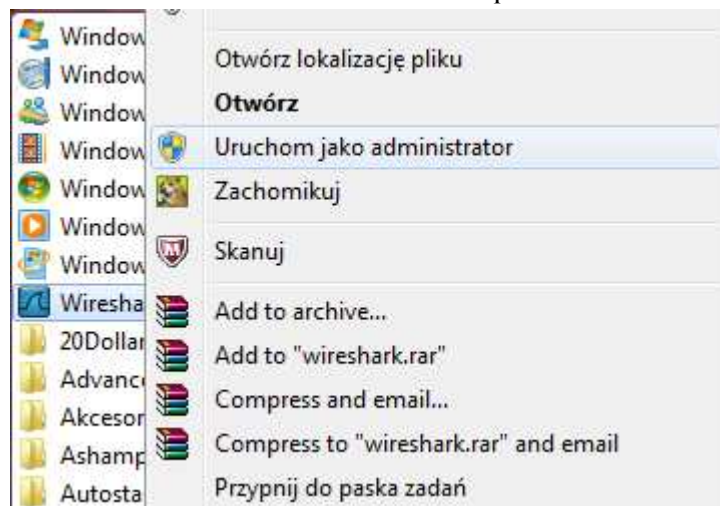
Jeśli szukasz programu do włamywania się, podrzucania wirusów, wykradania haseł czy czegoś w tym stylu - to NIE TO. Wireshark tylko wyświetla w czytelny sposób to co już i tak dociera do twojego komputera (zwykle oznacza to tylko ruch własny). Nie spowoduje, że nagle zobaczysz hasło do banku swojego szefa, albo kody do odpalenia amerykańskich, czy rosyjskich rakiet balistycznych.

Do czego może służyć Wireshark?

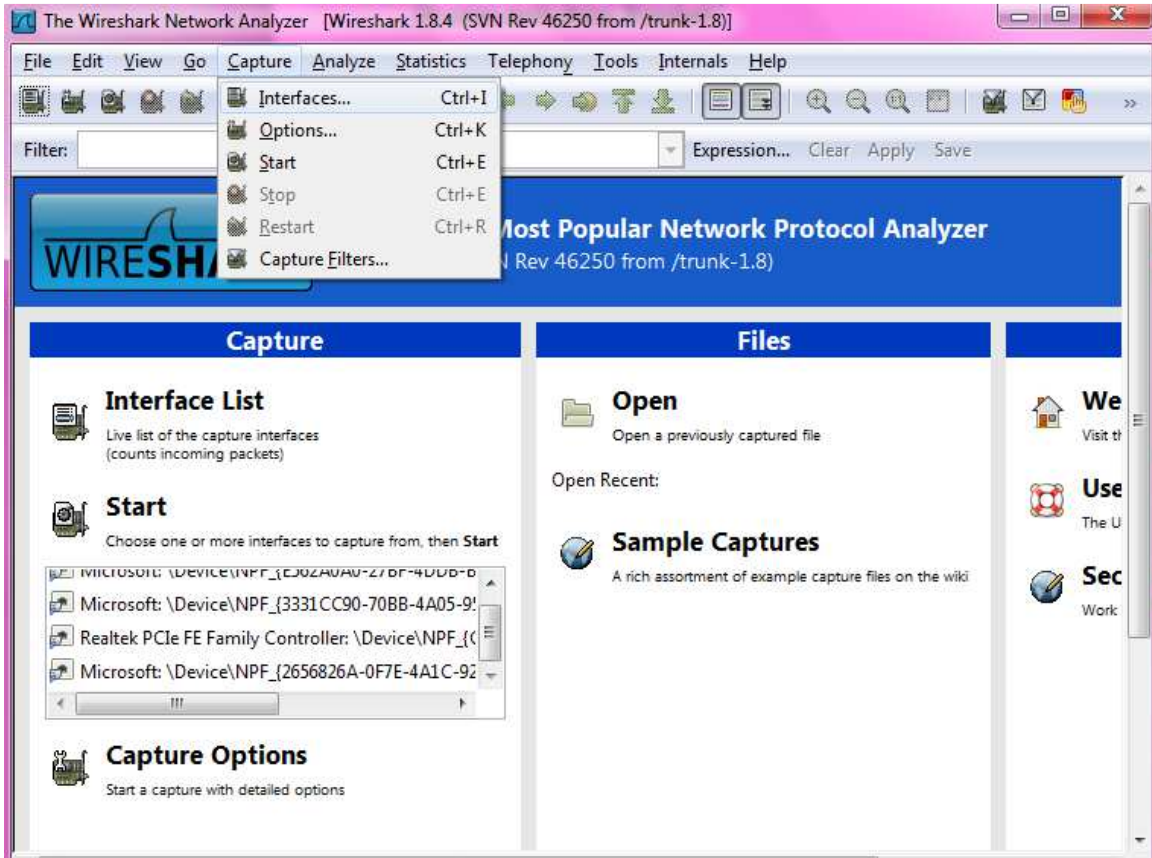
- Głównie do diagnostyki: np. nie można połączyć się z jakimś serwerem - on nie działa, czy mój komputer w ogóle nie próbuje, a może DNS podał zły adres? To oznacza, że raczej trzeba już wiedzieć czego się szuka, a wtedy instrukcja obsługi nie jest potrzebna ;)
- Jest doskonałym narzędziem do nauki - żaden wykład teoretyczny nie zastąpi popatrzenia sobie w praktyce jak te pakiety faktycznie wyglądają.
- No dobrze, można też podsłuchiwać. Jest to możliwe, kiedy komputer "słyszy" nie swoje pakiety. Dzieje się tak np. gdy mamy sieć zbudowaną na HUBach (a nie switchach), bezprzewodową (choć wtedy mogą być problemy z driverami i i tak się nie uda) lub nasz komputer robi za router ("udostępnia sieć" innym).

II. Przechwytywanie

- Uruchom **Wiresharka** z prawami Administratora



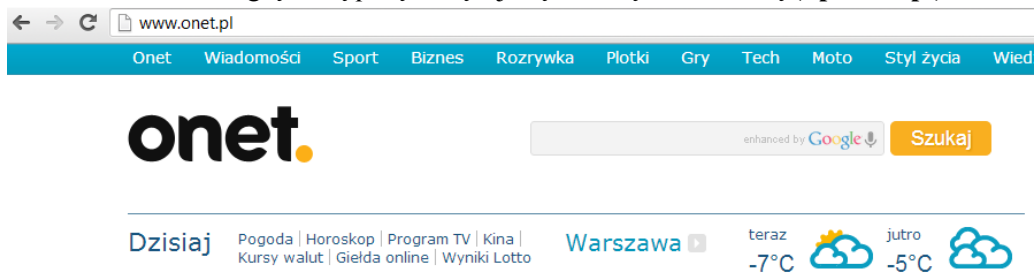
➤ Następnie wybierz **Capture - Interfaces**



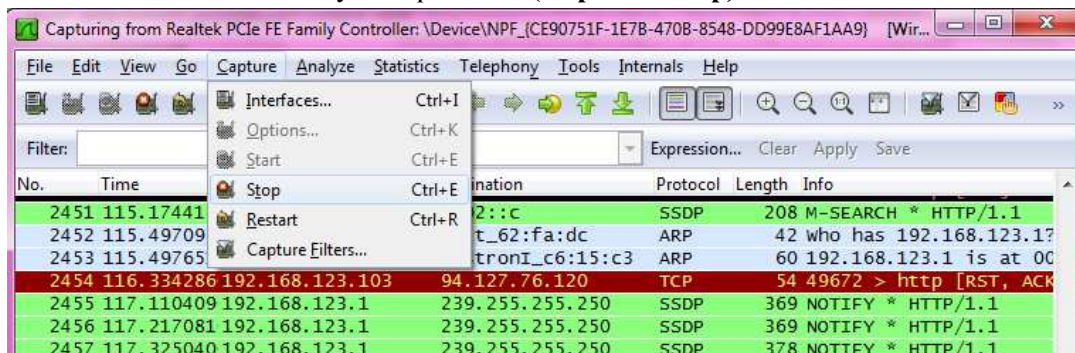
➤ Wybierz interfejs którym twój komputer łączy się z siecią (zwykle **eth0**) i naciśnij start.



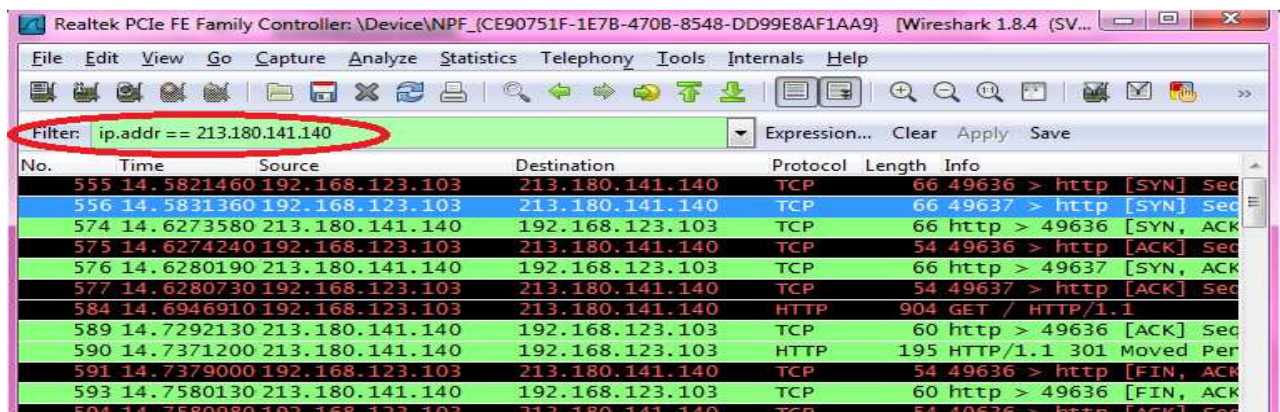
- Przeglądarką połącz się z jakąś stroną internetową (np. onet.pl).



- Skończ chwywanie pakietów (Capture - Stop)

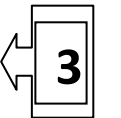
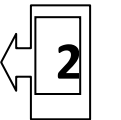
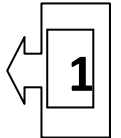
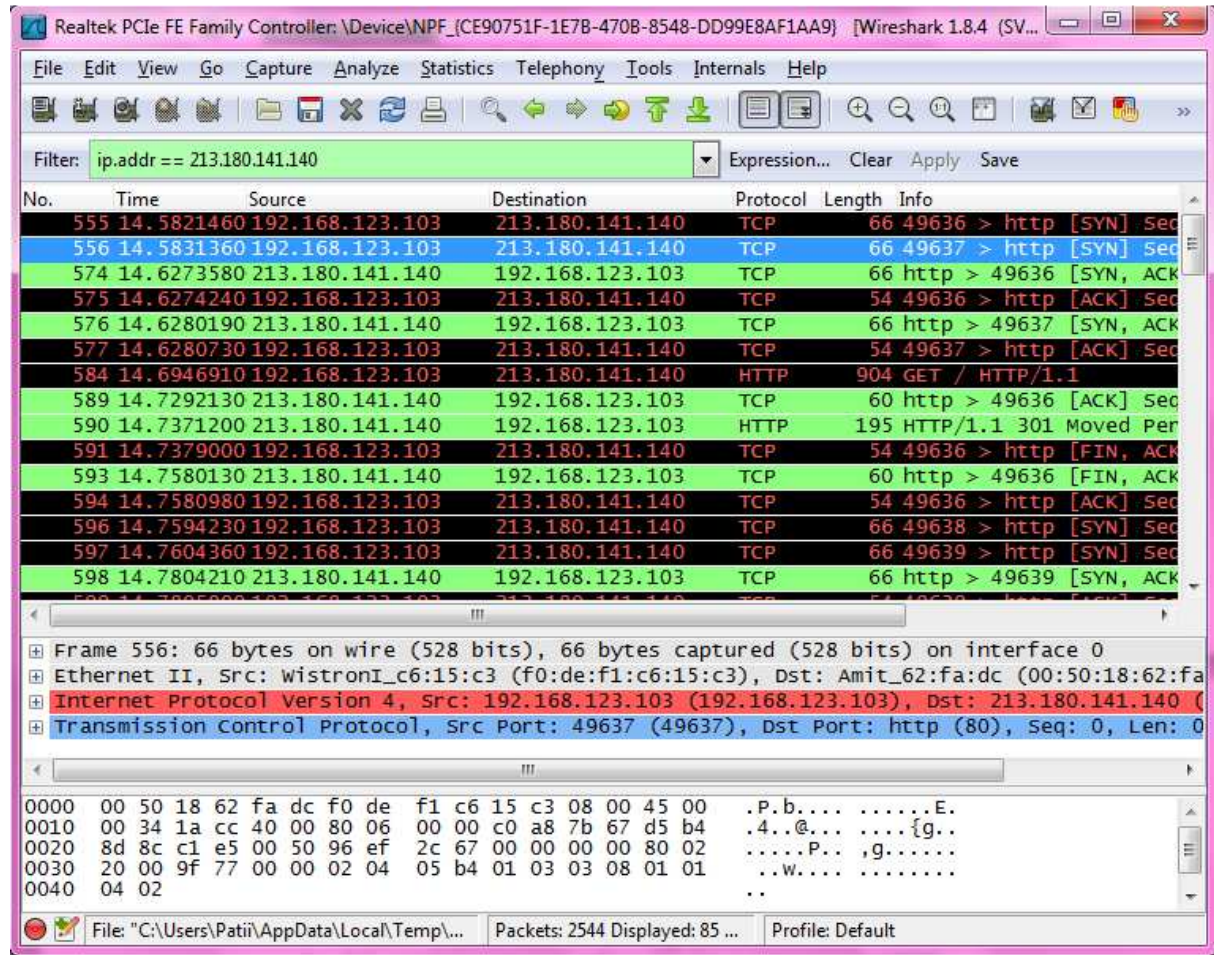


- Przefiltruj listę pakietów by zostawić tylko te opisujące komunikację z hostem www.onet.pl (IP: 213.180.141.140): w pasku „Filter” u góry okna programu wpisz: `ip.addr == 213.180.141.140`.



III. Analiza pakietów

- Okno Wireshark podzielone jest na trzy części:
 - górna część zawiera listę złapanych pakietów (przefiltrowanych przez filtr) - **1**
 - środkowa część to analiza wybranego pakietu - **2**
 - dolna część to zawartość pakietu (binarna i reprezentacja ASCII) - **3**



- Zadania:

- Kliknij w wiersz tabelki opisujący pierwszy pakiet protokołu HTTP.

Realtek PCIe FE Family Controller: \Device\NPF_{CE90751F-1E7B-470B-8548-DD99E8AF1AA9} [Wireshark 1.8.4 (SV...)]

Filter: ip.addr == 213.180.141.140

No.	Time	Source	Destination	Protocol	Length	Info
555	14.5821460	192.168.123.103	213.180.141.140	TCP	66	49636 > http [SYN] Seq...
556	14.5831360	192.168.123.103	213.180.141.140	TCP	66	49637 > http [SYN] Seq...
574	14.6273580	213.180.141.140	192.168.123.103	TCP	66	http > 49636 [SYN, ACK] Seq...
575	14.6274240	192.168.123.103	213.180.141.140	TCP	54	49636 > http [ACK] Seq...
576	14.6280190	213.180.141.140	192.168.123.103	TCP	66	http > 49637 [SYN, ACK] Seq...
577	14.6280730	192.168.123.103	213.180.141.140	TCP	54	49637 > http [ACK] Seq...
584	14.6946910	192.168.123.103	213.180.141.140	HTTP	904	GET / HTTP/1.1
589	14.7292130	213.180.141.140	192.168.123.103	TCP	60	http > 49636 [ACK] Seq...
590	14.7371200	213.180.141.140	192.168.123.103	HTTP	195	HTTP/1.1 301 Moved Per...
591	14.7379000	192.168.123.103	213.180.141.140	TCP	54	49636 > http [FIN, ACK] Seq...
593	14.7580130	213.180.141.140	192.168.123.103	TCP	60	http > 49636 [FIN, ACK] Seq...
594	14.7580980	192.168.123.103	213.180.141.140	TCP	54	49636 > http [ACK] Seq...
596	14.7594230	192.168.123.103	213.180.141.140	TCP	66	49638 > http [SYN] Seq...
597	14.7604360	192.168.123.103	213.180.141.140	TCP	66	49639 > http [SYN] Seq...
598	14.7804210	213.180.141.140	192.168.123.103	TCP	66	http > 49639 [SYN, ACK] Seq...

Frame 590: 195 bytes on wire (1560 bits), 195 bytes captured (1560 bits) on interface 0

- Ethernet II, Src: Amit_62:fa:dc (00:50:18:62:fa:dc), Dst: wistronI_c6:15:c3 (f0:de:f1:c6:15:c3)
- Internet Protocol Version 4, Src: 213.180.141.140 (213.180.141.140), Dst: 192.168.123.103 (192.168.123.103)
- Transmission Control Protocol, Src Port: http (80), Dst Port: 49636 (49636), Seq: 1, Ack: 851, Len: 141
- Hypertext Transfer Protocol

File: "C:\Users\Pati\AppData\Local\Temp\..." Packets: 2544 Displayed: 85 ... Profile: Default

- Frame 590: 195 bytes on wire (1560 bits), 195 bytes captured (1560 bits) on interface 0
- Ethernet II, Src: Amit_62:fa:dc (00:50:18:62:fa:dc), Dst: wistronI_c6:15:c3 (f0:de:f1:c6:15:c3)
- Internet Protocol Version 4, Src: 213.180.141.140 (213.180.141.140), Dst: 192.168.123.103 (192.168.123.103)
- Transmission Control Protocol, Src Port: http (80), Dst Port: 49636 (49636), Seq: 1, Ack: 851, Len: 141
- Hypertext Transfer Protocol

- Obejrzyj szczegółowe informacje w środkowej części okna.

- Transmission Control Protocol, Src Port: http (80), Dst Port: 49636 (49636), Seq: 1, Ack: 851, Len: 141
- Hypertext Transfer Protocol
 - HTTP/1.1 301 Moved Permanently\r\n
 - [Expert Info (Chat/Sequence): HTTP/1.1 301 Moved Permanently\r\n]
 - Request version: HTTP/1.1
 - Status Code: 301
 - Response Phrase: Moved Permanently
 - Location: http://www.onet.pl\r\n
 - server: edgeserver\r\n
 - connection: keep-alive\r\n
 - Transfer-Encoding: chunked\r\n
 - \r\n
 - HTTP chunked response
 - End of chunked encoding
 - Chunk size: 0 octets
 - Chunk boundary

➤ Kliknij w Hypertext Transfer Protocol w środkowej części okna.

```
Transmission Control Protocol, Src Port: http (80), Dst Port: 49636 (49636), Seq: 1, Ack: 851, Len: 141
Hypertext Transfer Protocol
  HTTP/1.1 301 Moved Permanently\r\n
    [Expert Info (Chat/Sequence): HTTP/1.1 301 Moved Permanently\r\n]
    Request Version: HTTP/1.1
    Status Code: 301
    Response Phrase: Moved Permanently
    Location: http://www.onet.pl\r\n
    Server: edgserver\r\n
    Connection: keep-alive\r\n
    Transfer-Encoding: chunked\r\n
    \r\n
  HTTP chunked response
    End of chunked encoding
      Chunk size: 0 octets
      Chunk boundary
```

➤ Obejrzyj które bajty w pakiecie odpowiadają za zawartość HTTP, a które za nagłówki (Ethernet, IP, TCP).

```
Transmission Control Protocol, Src Port: http (80), Dst Port: 49636 (49636), Seq: 1, Ack: 851, Len: 141
  Source port: http (80)
  Destination port: 49636 (49636)
  [Stream index: 19]
  Sequence number: 1 (relative sequence number)
  [Next sequence number: 142 (relative sequence number)]
  Acknowledgment number: 851 (relative ack number)
  Header length: 20 bytes
  Flags: 0x018 (PSH, ACK)
  Window size value: 32
  [Calculated window size: 16384]
  [Window size scaling factor: 512]
  Checksum: 0xefe6 [validation disabled]
  [SEQ/ACK analysis]
```

```
Ethernet II, Src: Amit_62:fa:dc (00:50:18:62:fa:dc), Dst: wistronI_c6:15:c3 (f0:de:f1:c6:15:c3)
  Destination: wistronI_c6:15:c3 (f0:de:f1:c6:15:c3)
  Source: Amit_62:fa:dc (00:50:18:62:fa:dc)
  Type: IP (0x0800)
```

```
Internet Protocol Version 4, Src: 213.180.141.140 (213.180.141.140), Dst: 192.168.123.103 (192.168.123.103)
  Version: 4
  Header length: 20 bytes
  Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
  Total Length: 181
  Identification: 0x1451 (5201)
  Flags: 0x02 (Don't Fragment)
  Fragment offset: 0
  Time to live: 54
  Protocol: TCP (6)
  Header checksum: 0x90a1 [correct]
  Source: 213.180.141.140 (213.180.141.140)
  Destination: 192.168.123.103 (192.168.123.103)
  [Source GeoIP: Unknown]
  [Destination GeoIP: Unknown]
```

➤ Jaki jest lokalny numer portu? Jaki jest numer portu serwera?

```
Transmission Control Protocol, Src Port: http (80), Dst Port: 49636 (49636), Seq: 1, Ack: 851, Len: 141
  Source port: http (80)
  Destination port: 49636 (49636)
  [Stream index: 19]
  Sequence number: 1 (relative sequence number)
```

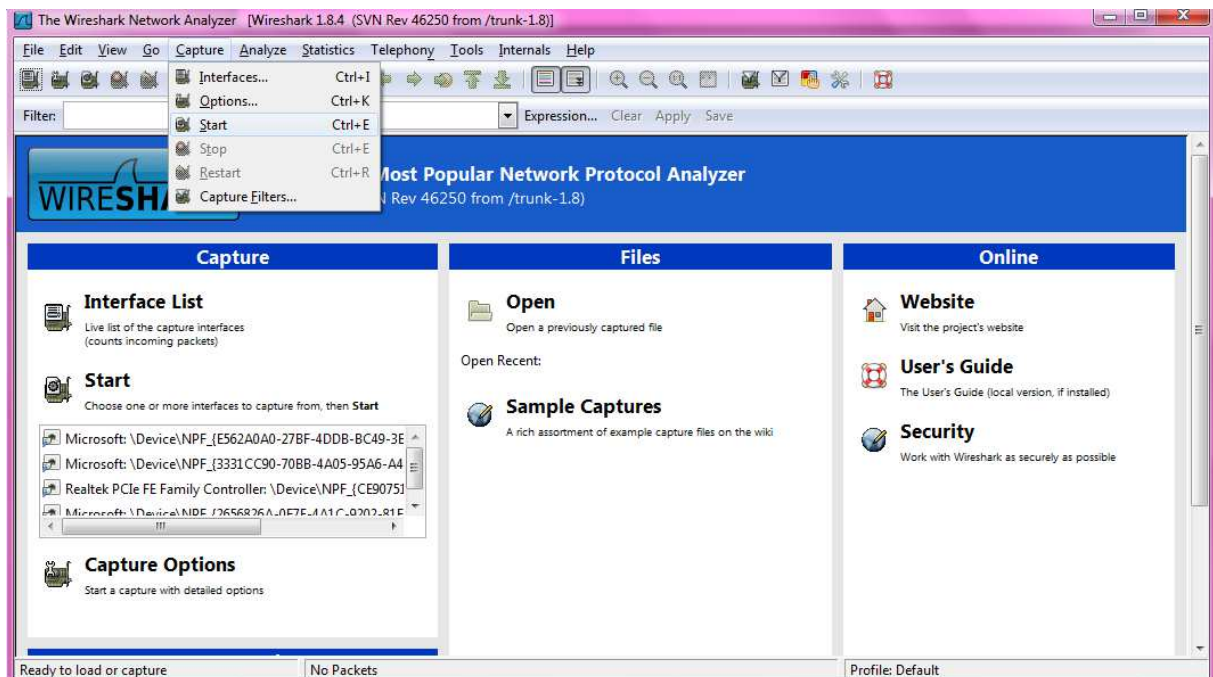

- Znajdź pakiet z odpowiedzią serwera.

```
Frame 590: 195 bytes on wire (1560 bits), 195 bytes captured (1560 bits) on interface 0
Interface id: 0
WTAP_ENCAP: 1
Arrival Time: Dec 8, 2012 18:16:38.678479000 Środzkowoeuropejski czas stand.
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1354986998.678479000 seconds
[Time delta from previous captured frame: 0.007907000 seconds]
[Time delta from previous displayed frame: 0.007907000 seconds]
[Time since reference or first frame: 14.737120000 seconds]
Frame Number: 590
Frame Length: 195 bytes (1560 bits)
Capture Length: 195 bytes (1560 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: eth:ip:tcp:http:data]
[Coloring Rule Name: HTTP]
[Coloring Rule String: http || tcp.port == 80]
```

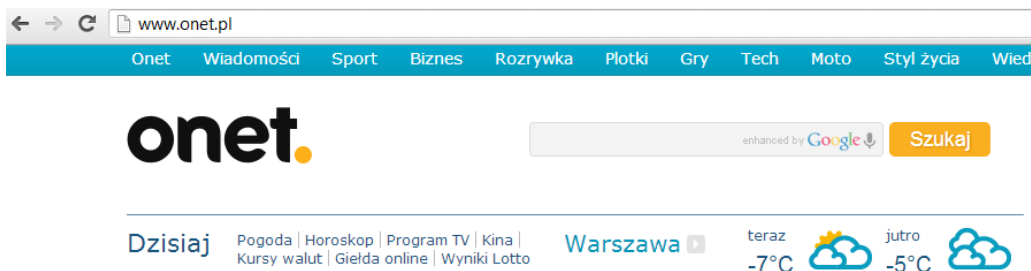
- Z jakiego portu została wysłana odpowiedź? Na jaki port w naszym hoście?
- Jaka jest zawartość odpowiedzi serwera?

IV. Cache'owanie w HTTP

- Tą część trzeba wykonać z włączonym łapaniem pakietów [po wstępnym wybraniu interfejsu, łapanie pakietów można włączać przez Capture – Start]



- Połącz się jeszcze raz z Onet.pl (lub inną wybraną stroną)



- Ogranicz pakiety do http (filtr: http && ip.addr == 213.180.141.140)

Filter: http && ip.addr == 213.180.141.140

- Co zmieniło się w odpowiedzi serwera? Dlaczego serwer http mógł tak odpowiedzieć?

Capturing from Realtek PCIe FE Family Controller: \Device\NPF_{CE90751F-1E7B-470B-8548-DD99E8AF1AA9} [Wireshark 1.8.4 (SVN Rev 46250 from /trunk-1.8)]

Filter: http && ip.addr == 213.180.141.140

No.	Time	Source	Destination	Protocol	Length	Info
44	9.66981600	192.168.123.103	213.180.141.140	HTTP	1085	GET / HTTP/1.1
46	9.70327200	213.180.141.140	192.168.123.103	HTTP	195	HTTP/1.1 301 Moved Permanently
50	9.70803600	192.168.123.103	213.180.141.140	HTTP	1089	GET / HTTP/1.1
148	9.94190900	213.180.141.140	192.168.123.103	HTTP	1514	[TCP Retransmission] HTTP/1.1 200 OK (text/html)
150	10.0884670	192.168.123.103	213.180.141.140	HTTP	1073	GET /cdf/client.js HTTP/1.1
156	10.1486560	213.180.141.140	192.168.123.103	HTTP	577	HTTP/1.1 200 OK (text/javascript)
205	10.1895010	192.168.123.103	213.180.141.140	HTTP	566	POST /cdf/render HTTP/1.1 (application/x-www-form-urlencoded)
236	10.2845960	213.180.141.140	192.168.123.103	HTTP	401	HTTP/1.1 200 OK (text/html)
270	10.5043840	213.180.141.140	192.168.123.103	HTTP	401	[TCP Retransmission] HTTP/1.1 200 OK (text/html)

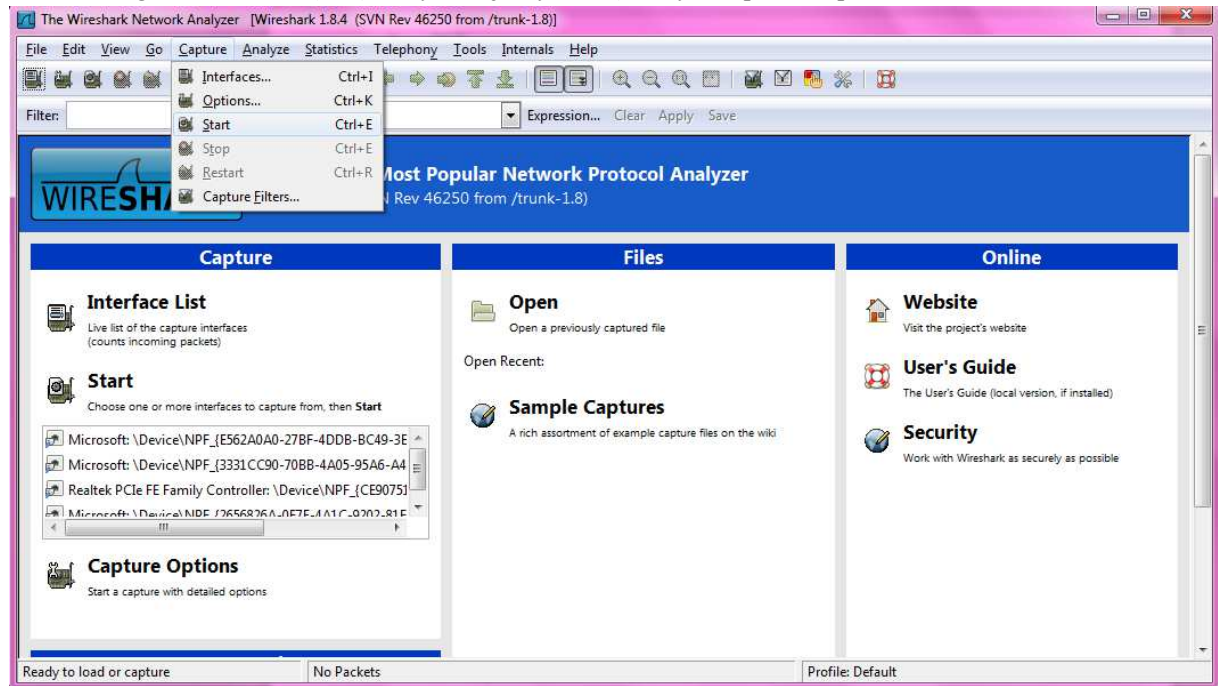
Frame 44: 1085 bytes on wire (8680 bits), 1085 bytes captured (8680 bits) on interface 0
Ethernet II, Src: WistronI_c6:15:c3 (f0:de:f1:c6:15:c3), Dst: Amit_62:fa:dc (00:50:18:62:fa:dc)
Internet Protocol Version 4, Src: 192.168.123.103 (192.168.123.103), Dst: 213.180.141.140 (213.180.141.140)

```
0000  00 50 18 62 fa dc f0 de f1 c6 15 c3 08 00 45 00  .P.b....E.  
0010  04 2f 25 f3 40 00 80 06 00 00 c0 a8 7b 67 d5 b4  .%.@...{g.  
0020  8d 8c c2 70 00 50 a6 74 e3 6e ef 1c 0d 4a 50 18  ...p.P.t.n..JP.  
0030  01 00 a3 72 00 00 47 45 54 20 2f 20 48 54 54 50  ...r..GET / HTTP  
0040  2f 31 2e 31 0d 0a 48 6f 73 74 3a 20 6f 6e 65 74  /1..Host: onet  
0050  2a 70 6c 04 02 43 6f 6a 6a 65 62 74 60 6f 6a 3a  a..con.net
```

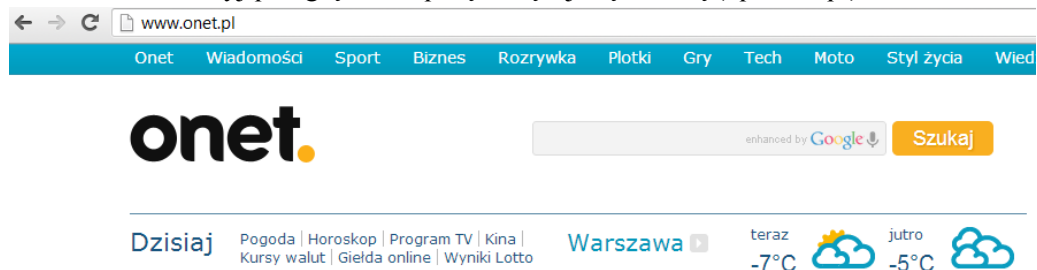
Realtek PCIe FE Family Controller: \Device\N... Packets: 1158 Displayed: 9 Marked: 0 Profile: Default

V. TCP

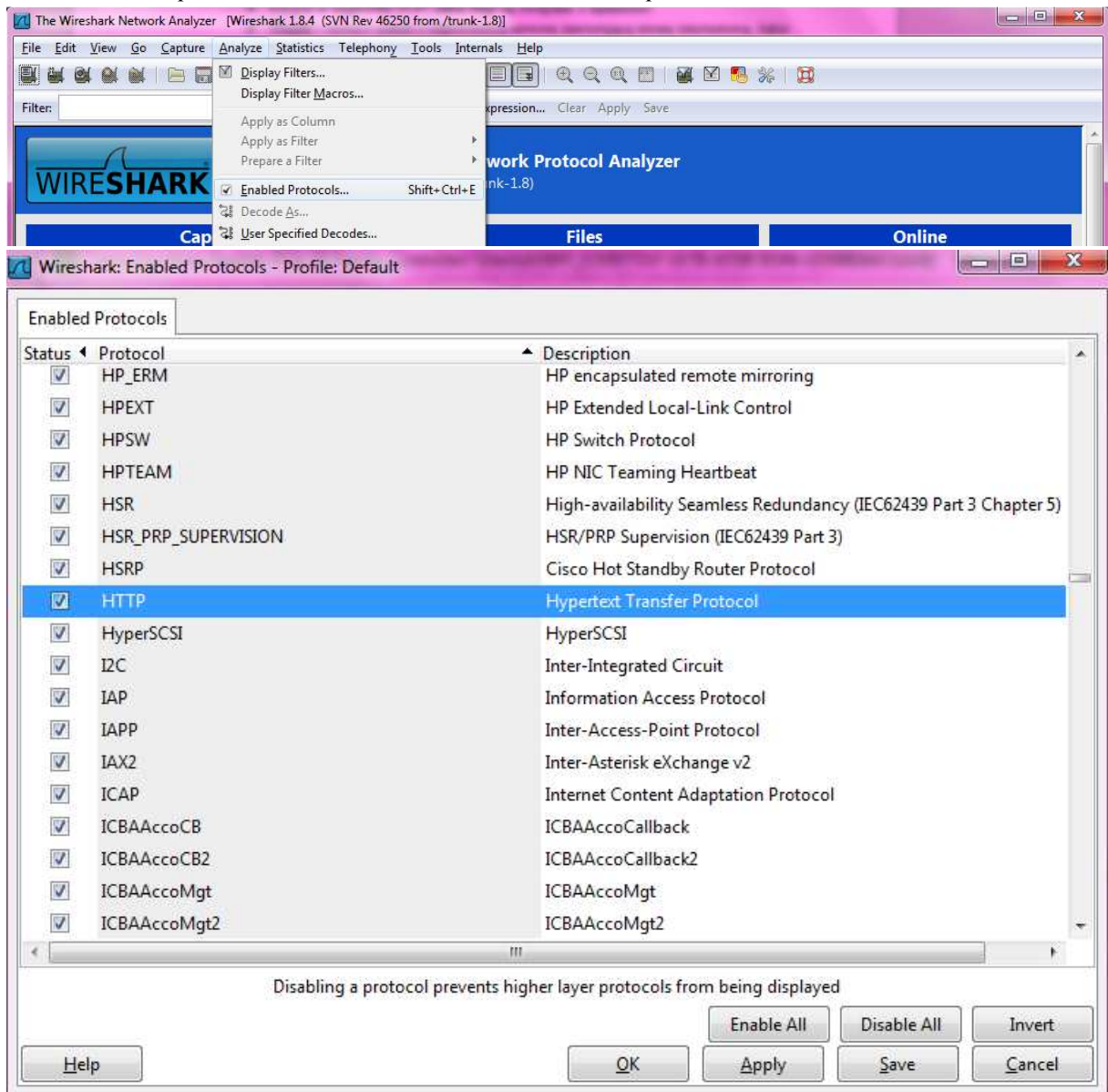
- Nagłówek TCP (tą część wykonujemy z włączonym łapaniem pakietów)



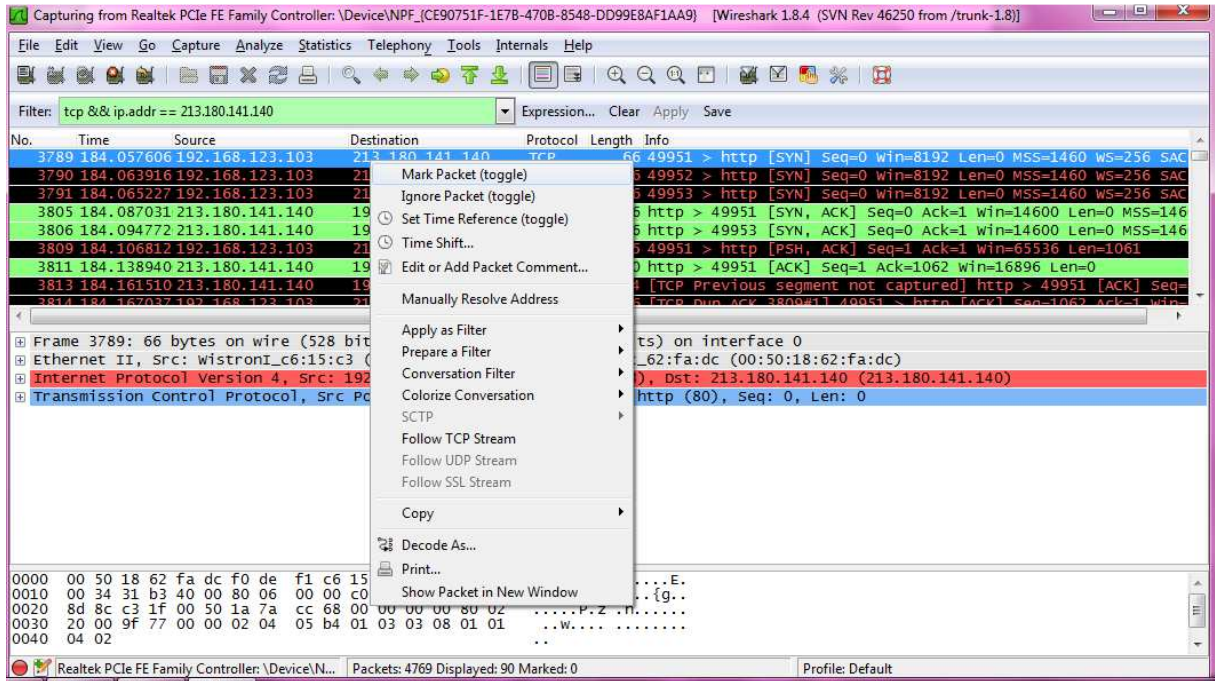
➤ użyj przeglądarki i połącz się z jakąś stroną (np. onet.pl)



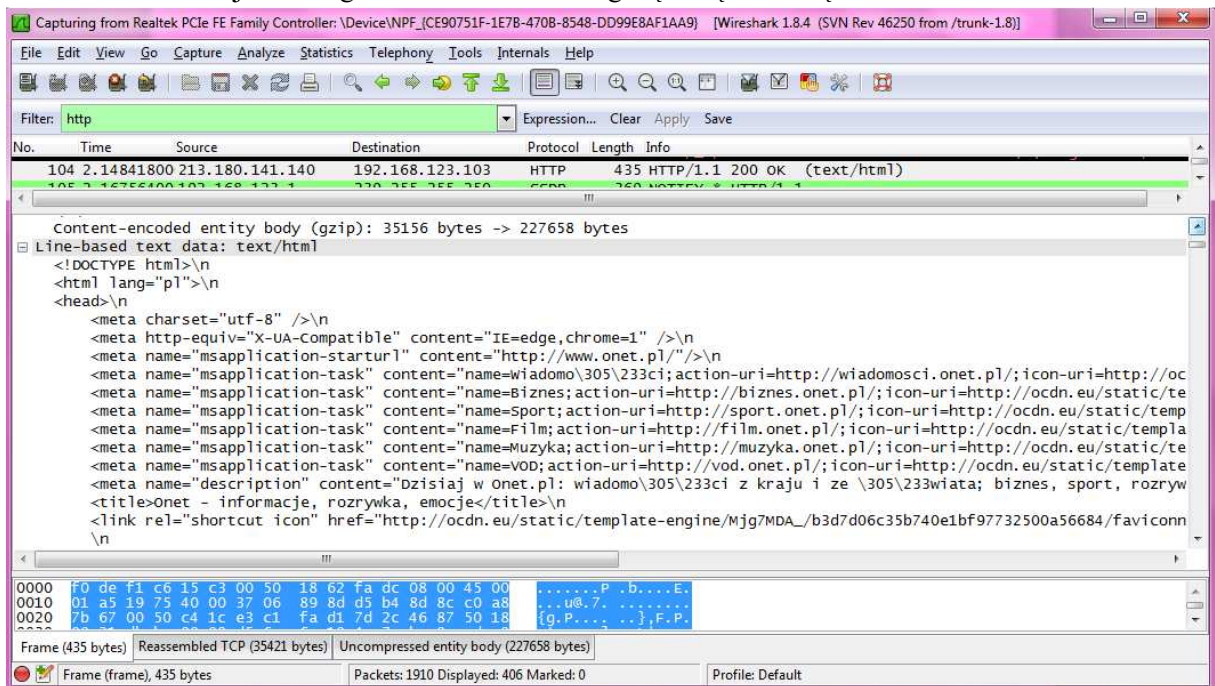
- Wyłącz analizę protokołu HTTP (Analyze - Enabled Protocols). Ustaw filtr na protokół TCP i adres IP serwera www.onet.pl



- Znajdź i zaznacz (Edit - Mark Packet) pakiet w którym przeglądarka przesyła żądanie do serwera.



- Co jest w nagłówku TCP? Jakie flagi są związane z żądaniem?



- Znajdź i zaznacz pakiet z odpowiedzią serwera zawierającą stronę internetową. Jakie flagi są ustawione? Czy jest to jedyna odpowiedź TCP serwera? Jeśli nie - dlaczego jest ich więcej? Czym różnią się pozostałe odpowiedzi?

- ☐ **Flags: 0x02 (Don't Fragment)**
 - 0... .. = Reserved bit: Not set
 - .1... .. = Don't fragment: Set
 - ..0. = More fragments: Not set
- Fragment offset: 0
- Time to live: 128
- Protocol: TCP (6)

- Zestawianie i zamykanie połączenia TCP

- Przeanalizuj pakiety tcp poprzedzające pakiet z żądaniem http.

No.	Time	Source	Destination	Protocol	Length	Info
44	9.66981600	192.168.123.103	213.180.141.140	HTTP	1085	GET / HTTP/1.1
46	9.70327200	213.180.141.140	192.168.123.103	HTTP	195	HTTP/1.1 301 Moved Permanently
50	9.70801600	192.168.123.103	213.180.141.140	HTTP	1089	GET / HTTP/1.1
148	9.94190900	213.180.141.140	192.168.123.103	HTTP	1514	[TCP Retransmission] HTTP/1.1 200 OK (text/html)
150	10.0884670	192.168.123.103	213.180.141.140	HTTP	1073	GET /cdn/client.js HTTP/1.1
156	10.1486560	213.180.141.140	192.168.123.103	HTTP	577	HTTP/1.1 200 OK (text/javascript)
205	10.1895010	192.168.123.103	213.180.141.140	HTTP	586	POST /cdn/render HTTP/1.1 (application/x-www-form-urlencoded)
236	10.2845960	213.180.141.140	192.168.123.103	HTTP	401	HTTP/1.1 200 OK (text/html)
270	10.5043840	213.180.141.140	192.168.123.103	HTTP	401	[TCP Retransmission] HTTP/1.1 200 OK (text/html)

- Ile jest takich pakietów?
- Kto inicjuje połączenie?

Source

192.168.123.103
213.180.141.140
192.168.123.103
213.180.141.140
192.168.123.103
213.180.141.140
192.168.123.103
213.180.141.140
213.180.141.140

- Jakie flagi są ustawiane?

Flags: 0x02 (Don't Fragment)
 0... .. = Reserved bit: Not set
 .1.. .. = Don't fragment: Set
 ..0. .. = More fragments: Not set
 Fragment offset: 0
 Time to live: 128
 Protocol: TCP (6)

- Przeanalizuj pakiety tcp po pakiecie z odpowiedzią http.

No.	Time	Source	Destination	Protocol	Length	Info
44	9.66981600	192.168.123.103	213.180.141.140	HTTP	1085	GET / HTTP/1.1
46	9.70327200	213.180.141.140	192.168.123.103	HTTP	195	HTTP/1.1 301 Moved Permanently
50	9.70801600	192.168.123.103	213.180.141.140	HTTP	1089	GET / HTTP/1.1
148	9.94190900	213.180.141.140	192.168.123.103	HTTP	1514	[TCP Retransmission] HTTP/1.1 200 OK (text/html)
150	10.0884670	192.168.123.103	213.180.141.140	HTTP	1073	GET /cdn/client.js HTTP/1.1
156	10.1486560	213.180.141.140	192.168.123.103	HTTP	577	HTTP/1.1 200 OK (text/javascript)
205	10.1895010	192.168.123.103	213.180.141.140	HTTP	586	POST /cdn/render HTTP/1.1 (application/x-www-form-urlencoded)
236	10.2845960	213.180.141.140	192.168.123.103	HTTP	401	HTTP/1.1 200 OK (text/html)
270	10.5043840	213.180.141.140	192.168.123.103	HTTP	401	[TCP Retransmission] HTTP/1.1 200 OK (text/html)

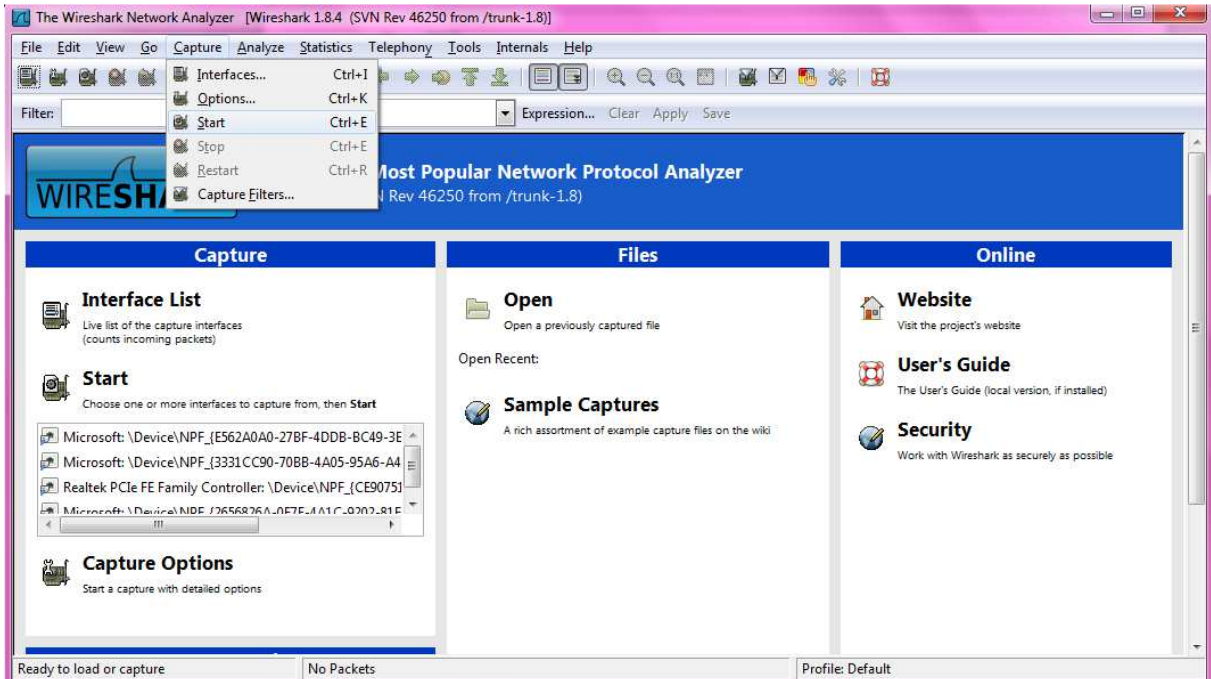
- Ile jest takich pakietów?
- Kto zaczyna proces zamykania połączenie?

213.180.141.140
192.168.123.103
213.180.141.140
192.168.123.103
213.180.141.140
192.168.123.103
213.180.141.140
192.168.123.103
192.168.123.103

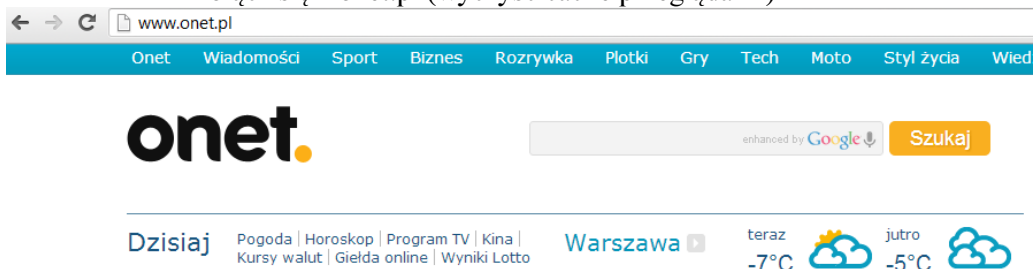
➤ Jakie flagi są ustawiane?

- Flags: 0x02 (Don't Fragment)
 - 0... = Reserved bit: Not set
 - .1.. = Don't fragment: Set
 - ..0. = More fragments: Not set
- Fragment offset: 0
- Time to live: 128
- Protocol: TCP (6)

- Połączenia keep-alive w http - zobaczymy jak przeglądarka i serwer obsługują strony wymagające więcej niż jednego żądania http. (Tą część wykonujemy z włączonym łapaniem pakietów)



➤ Połącz się z onet.pl (wyczyść cache przeglądarki)



➤ Znajdź i zaznacz pakiety zawierające dwa żądania http

The screenshot shows the Wireshark interface with a filter set to `http && ip.addr == 213.180.141.140`. The packet list pane displays several HTTP packets:

No.	Time	Source	Destination	Protocol	Length	Info
325	65.7133740	192.168.123.103	213.180.141.140	HTTP	1115	GET / HTTP/1.1
412	66.2662710	213.180.141.140	192.168.123.103	HTTP	460	HTTP/1.1 200 OK (text/html)
414	66.2804110	192.168.123.103	213.180.141.140	HTTP	1099	GET /_cdf/client.js HTTP/1.1
420	66.3408530	213.180.141.140	192.168.123.103	HTTP	610	HTTP/1.1 200 OK (text/javascript)
436	69.1592840	192.168.123.103	213.180.141.140	HTTP	566	POST /_cdf/render HTTP/1.1 (application/x-www-form-urlencoded)
440	69.2144320	213.180.141.140	192.168.123.103	HTTP	538	HTTP/1.1 200 OK (text/html)
558	69.4804690	213.180.141.140	192.168.123.103	HTTP	538	[TCP Retransmission] HTTP/1.1 200 OK (text/html)
1293	185.062849	192.168.123.103	213.180.141.140	HTTP	1115	GET / HTTP/1.1
1374	185.311272	213.180.141.140	192.168.123.103	HTTP	473	HTTP/1.1 200 OK (text/html)

The packet details pane for packet 325 shows the following layers:

- Frame 325: 1115 bytes on wire (8920 bits), 1115 bytes captured (8920 bits) on interface 0
- Ethernet II, Src: WistronI_c6:15:c3 (f0:de:f1:c6:15:c3), Dst: Amit_62:fa:dc (00:50:18:62:fa:dc)
- Internet Protocol Version 4, Src: 192.168.123.103 (192.168.123.103), Dst: 213.180.141.140 (213.180.141.140)
- Transmission Control Protocol, Src Port: 50013 (50013), Dst Port: http (80), Seq: 1, Ack: 1, Len: 1061
- Hypertext Transfer Protocol

The packet bytes pane shows the raw data in hexadecimal and ASCII:

```

0000 00 50 18 62 fa dc f0 de f1 c6 15 c3 08 00 45 00  .P.b....E.
0010 04 4d 35 ba 40 00 80 06 00 00 c0 a8 7b 67 d5 b4  .M5.@...{g.
0020 8d 8c 3c 5d 00 50 03 3c 56 87 8c 91 a2 39 50 18  ...].P.<V...9P.
0030 01 00 a3 90 00 00 47 45 54 20 2f 20 48 54 54 50  .....GET / HTTP
0040 2f 31 2e 31 0d 0a 48 6f 73 74 3a 20 77 77 77 2e  /1.1..Host: www.
0050 ef 6a 65 74 2a 70 67 04 03 43 6f 6a 65 63 74  _POST
  
```


- Czy dla każdego żądania nawiązywana była oddzielna sesja TCP, czy też oba żądania były obsłużone w jednej sesji?

The screenshot shows a Wireshark capture of network traffic. The filter is set to 'http && ip.addr == 213.180.141.140'. The packet list shows several HTTP requests and responses:

No.	Time	Source	Destination	Protocol	Length	Info
325	65.7133740	192.168.123.103	213.180.141.140	HTTP	1115	GET / HTTP/1.1
412	66.2662710	213.180.141.140	192.168.123.103	HTTP	460	HTTP/1.1 200 OK (text/html)
414	66.2804110	192.168.123.103	213.180.141.140	HTTP	1099	GET /cdf/client.js HTTP/1.1
420	66.3408530	213.180.141.140	192.168.123.103	HTTP	610	HTTP/1.1 200 OK (text/javascript)
436	69.1592840	192.168.123.103	213.180.141.140	HTTP	566	POST /cdf/render HTTP/1.1 (application/x-www-form-urlencoded)
440	69.2144320	213.180.141.140	192.168.123.103	HTTP	538	HTTP/1.1 200 OK (text/html)
558	69.4804690	213.180.141.140	192.168.123.103	HTTP	538	[TCP Retransmission] HTTP/1.1 200 OK (text/html)
1293	185.062849	192.168.123.103	213.180.141.140	HTTP	1115	GET / HTTP/1.1
1374	185.311272	213.180.141.140	192.168.123.103	HTTP	473	HTTP/1.1 200 OK (text/html)

The packet details pane shows a Transmission Control Protocol (TCP) segment with the following information:

- Source port: http (80)
- Destination port: 50013 (50013)
- [Stream index: 19]
- Sequence number: 35041 (relative sequence number)
- [Next sequence number: 35447 (relative sequence number)]
- Acknowledgment number: 1062 (relative ack number)
- Header length: 20 bytes
- Flags: 0x018 (PSH, ACK)
- window size value: 33
- [Calculated window size: 16896]

The packet bytes pane shows the raw data of the TCP segment, including the sequence number 35041 and the acknowledgment number 1062.

The screenshot shows a Wireshark capture of network traffic. The filter is set to 'http && ip.addr == 213.180.141.140'. The packet list shows several HTTP requests and responses:

No.	Time	Source	Destination	Protocol	Length	Info
325	65.7133740	192.168.123.103	213.180.141.140	HTTP	1115	GET / HTTP/1.1
412	66.2662710	213.180.141.140	192.168.123.103	HTTP	460	HTTP/1.1 200 OK (text/html)
414	66.2804110	192.168.123.103	213.180.141.140	HTTP	1099	GET /cdf/client.js HTTP/1.1
420	66.3408530	213.180.141.140	192.168.123.103	HTTP	610	HTTP/1.1 200 OK (text/javascript)
436	69.1592840	192.168.123.103	213.180.141.140	HTTP	566	POST /cdf/render HTTP/1.1 (application/x-www-form-urlencoded)
440	69.2144320	213.180.141.140	192.168.123.103	HTTP	538	HTTP/1.1 200 OK (text/html)
558	69.4804690	213.180.141.140	192.168.123.103	HTTP	538	[TCP Retransmission] HTTP/1.1 200 OK (text/html)
1293	185.062849	192.168.123.103	213.180.141.140	HTTP	1115	GET / HTTP/1.1
1374	185.311272	213.180.141.140	192.168.123.103	HTTP	473	HTTP/1.1 200 OK (text/html)

The packet details pane shows a Transmission Control Protocol (TCP) segment with the following information:

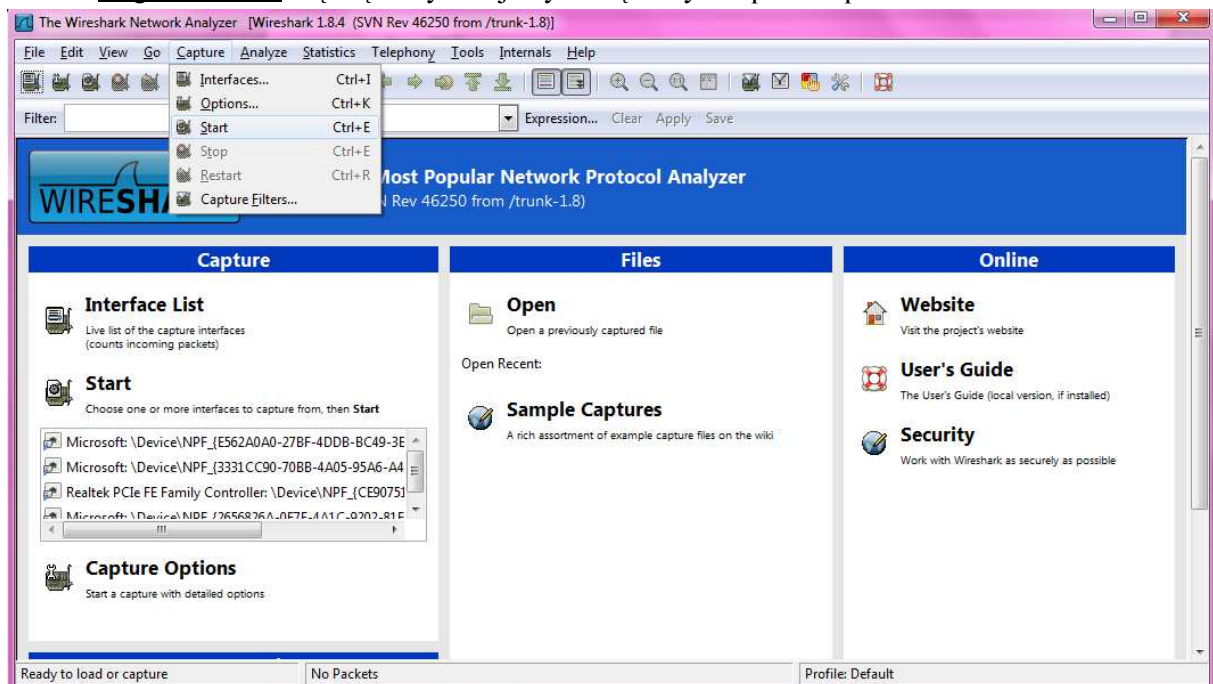
- Source port: 50013 (50013)
- Destination port: http (80)
- [Stream index: 19]
- Sequence number: 1062 (relative sequence number)
- [Next sequence number: 2107 (relative sequence number)]
- Acknowledgment number: 35447 (relative ack number)
- Header length: 20 bytes
- Flags: 0x018 (PSH, ACK)
- window size value: 255
- [Calculated window size: 65280]
- [window size scaling factor: 256]

The packet bytes pane shows the raw data of the TCP segment, including the sequence number 1062 and the acknowledgment number 35447.

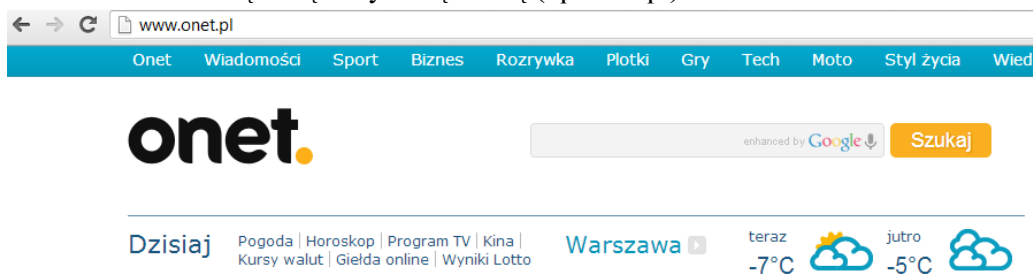
- Co daje niezamykanie połączeń (czyli http w trybie keep-alive)? Jakie są możliwe problemy?

VI. IP

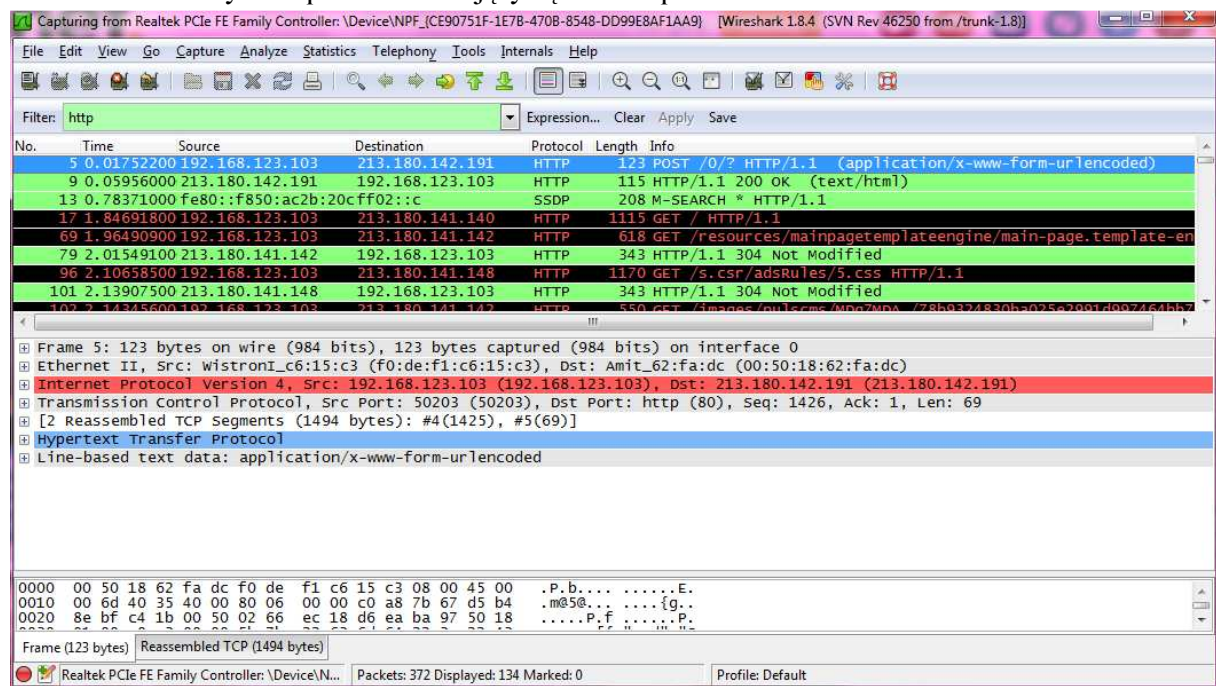
- Nagłówek IPv4 - tą część wykonujemy z włączonym łapaniem pakietów



- Połącz się z wybraną stroną (np. onet.pl)



- Wybierz pakiet zawierający żądanie http



➤ Przeanalizuj zawartość nagłówka IP

Capturing from Realtek PCIe FE Family Controller: \Device\NPF_{CE90751F-1E7B-470B-8548-DD99E8AF1AA9} [Wireshark 1.8.4 (SVN Rev 46250 from /trunk-1.8)]

Filter: http

Frame 5: 123 bytes on wire (984 bits), 123 bytes captured (984 bits) on interface 0

Ethernet II, Src: WlstronI_c6:15:c3 (f0:de:f1:c6:15:c3), Dst: Amit_62:fa:dc (00:50:18:62:fa:dc)

Internet Protocol Version 4, Src: 192.168.123.103 (192.168.123.103), Dst: 213.180.142.191 (213.180.142.191)

- Version: 4
- Header length: 20 bytes
- Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-capable Transport))
- Total Length: 109
- Identification: 0x4035 (16437)
- Flags: 0x02 (Don't Fragment)
- Fragment offset: 0
- Time to live: 128
- Protocol: TCP (6)
- Header checksum: 0x0000 [incorrect, should be 0x19d2 (may be caused by "IP checksum offload"?)]
- Source: 192.168.123.103 (192.168.123.103)
- Destination: 213.180.142.191 (213.180.142.191)
- [Source GeoIP: Unknown]
- [Destination GeoIP: Unknown]
- Transmission Control Protocol, Src Port: 50203 (50203), Dst Port: http (80), Seq: 1426, Ack: 1, Len: 69
- [2 Reassembled TCP Segments (1494 bytes): #4(1425), #5(69)]
- Hypertext Transfer Protocol
- Line-based text data: application/x-www-form-urlencoded

0000 00 50 18 62 fa dc f0 de f1 c6 15 c3 08 00 45 00 .P.b....E:
 0010 00 0d 40 35 40 00 80 06 00 00 c0 a8 7b 67 d5 b4 {g.P.!. [.P.
 0020 8e bf c4 1b 00 50 02 66 ec 18 d6 ea ba 97 50 18P.f .P.
P.

Frame (123 bytes) Reassembled TCP (1494 bytes)

Internet Protocol Version 4 (ip), 20 bytes Packets: 586 Displayed: 177 Marked: 0 Profile: Default

➤ Wybierz pakiet zawierający odpowiedź http

No.	Time	Source	Destination	Protocol	Length	Info
5	0.01752200	192.168.123.103	213.180.142.191	HTTP	123	POST /0/? HTTP/1.1 (application/x-www-form-urlencoded)
9	0.05956000	213.180.142.191	192.168.123.103	HTTP	115	HTTP/1.1 200 OK (text/html)
13	0.78371000	fe80::f850:ac2b:20c:ff02::c		SSDP	208	M-SEARCH * HTTP/1.1
17	1.84691800	192.168.123.103	213.180.141.140	HTTP	1115	GET / HTTP/1.1
69	1.96490900	192.168.123.103	213.180.141.142	HTTP	618	GET /resources/mainpagetemplateengine/main-page.templa
79	2.01549100	213.180.141.142	192.168.123.103	HTTP	343	HTTP/1.1 304 Not Modified
96	2.10658500	192.168.123.103	213.180.141.148	HTTP	1170	GET /s.csr/adsRules/5.css HTTP/1.1
101	2.13907500	213.180.141.148	192.168.123.103	HTTP	343	HTTP/1.1 304 Not Modified
102	2.14345600	192.168.123.103	213.180.141.142	HTTP	550	GET /images/pulscms/MDg7MDA_/78b9324830ba025e2991d9974
103	2.14486300	192.168.123.103	213.180.141.145	HTTP	536	GET /_m/55fade2068e/503eae8d7ddf5eb6bd09,0,29.gif HTTP
104	2.14841800	213.180.141.140	192.168.123.103	HTTP	435	HTTP/1.1 200 OK (text/html)
105	2.16756400	192.168.123.1	239.255.255.250	SSDP	369	NOTIFY * HTTP/1.1
106	2.17232500	213.180.141.142	192.168.123.103	HTTP	301	HTTP/1.1 304 Not Modified
108	2.17440000	213.180.141.145	192.168.123.103	HTTP	309	HTTP/1.1 304 Not Modified

➤ Jakie jest TTL?

Capturing from Realtek PCIe FE Family Controller: \Device\NPF_{CE90751F-1E7B-470B-8548-DD99E8AF1AA9} [Wireshark 1.8.4 (SVN Rev 46250 from /trunk-1.8)]

Filter: http

No.	Time	Source	Destination	Protocol	Length	Info
104	2.14841800	213.180.141.140	192.168.123.103	HTTP	435	HTTP/1.1 200 OK (text/html)
105	2.16756400	192.168.123.1	239.255.255.250	SSDP	369	NOTIFY * HTTP/1.1
106	2.17232500	213.180.141.142	192.168.123.103	HTTP	301	HTTP/1.1 304 Not Modified
108	2.17440000	213.180.141.145	192.168.123.103	HTTP	309	HTTP/1.1 304 Not Modified

Internet Protocol Version 4, Src: 213.180.141.145 (213.180.141.145), Dst: 192.168.123.103 (192.168.123.103)

- Version: 4
- Header length: 20 bytes
- Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-capable Transport))
- Total Length: 295
- Identification: 0x892f (35119)
- Flags: 0x02 (Don't Fragment)
- Fragment offset: 0
- Time to live: 56**
- Protocol: TCP (6)
- Header checksum: 0x194c [correct]
- Source: 213.180.141.145 (213.180.141.145)
- Destination: 192.168.123.103 (192.168.123.103)
- [Source GeoIP: Unknown]
- [Destination GeoIP: Unknown]
- Transmission Control Protocol, Src Port: http (80), Dst Port: 50209 (50209), Seq: 1, Ack: 483, Len: 255
- Hypertext Transfer Protocol

0010 01 27 89 2f 40 00 06 19 4c d5 b4 8d 91 c0 a8 .:/@. .L.....
 0020 7b 67 00 50 c4 21 f7 1e 5b 5b da 10 c8 d5 50 18 {g.P.!. [.P.
 0030 00 0e 40 1b 00 00 48 54 54 50 2f 31 2e 31 20 33 ..@...HT P/1.1 3
 0040 30 34 20 4e 6f 74 20 4d 6f 64 69 66 69 65 64 0d 04 Not Modified.
 0050 0a 44 61 74 65 3a 20 53 61 74 2c 20 30 38 20 44 .Date: Sat, 08 D
 0060 65 62 70 72 20 31 22 20 21 28 22 20 20 22 24 22 cc 2012 18:00:42

Time to live (ip.ttl), 1 byte Packets: 5093 Displayed: 711 Marked: 0 Profile: Default

➤ W jakim celu wprowadzono pole Protocol?

The screenshot shows a packet capture in Wireshark. The filter is set to 'http'. The packet list shows four packets, with packet 104 selected. The packet details pane shows the following layers:

- Internet Protocol Version 4, Src: 213.180.141.145 (213.180.141.145), Dst: 192.168.123.103 (192.168.123.103)
 - Version: 4
 - Header length: 20 bytes
 - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
 - Total Length: 295
 - Identification: 0x892f (35119)
 - Flags: 0x02 (Don't Fragment)
 - Fragment offset: 0
 - Time to live: 56
 - Protocol: TCP (6)**
 - Header checksum: 0x194c [correct]
 - Source: 213.180.141.145 (213.180.141.145)
 - Destination: 192.168.123.103 (192.168.123.103)
 - [Source GeoIP: Unknown]
 - [Destination GeoIP: Unknown]
- Transmission Control Protocol, Src Port: http (80), Dst Port: 50209 (50209), Seq: 1, Ack: 483, Len: 255
- Hypertext Transfer Protocol

The packet bytes pane shows the raw data of the packet, including the IP header, TCP header, and the beginning of the HTTP request body.

➤ Co sugeruje obecność pola "Fragment Offset"?

The screenshot shows a packet capture in Wireshark. The filter is set to 'http'. The packet list shows four packets, with packet 104 selected. The packet details pane shows the following layers:

- Internet Protocol Version 4, Src: 213.180.141.145 (213.180.141.145), Dst: 192.168.123.103 (192.168.123.103)
 - Version: 4
 - Header length: 20 bytes
 - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
 - Total Length: 295
 - Identification: 0x892f (35119)
 - Flags: 0x02 (Don't Fragment)
 - Fragment offset: 0**
 - Time to live: 56
 - Protocol: TCP (6)
 - Header checksum: 0x194c [correct]
 - Source: 213.180.141.145 (213.180.141.145)
 - Destination: 192.168.123.103 (192.168.123.103)
 - [Source GeoIP: Unknown]
 - [Destination GeoIP: Unknown]
- Transmission Control Protocol, Src Port: http (80), Dst Port: 50209 (50209), Seq: 1, Ack: 483, Len: 255
- Hypertext Transfer Protocol

The packet bytes pane shows the raw data of the packet, including the IP header, TCP header, and the beginning of the HTTP request body.