



**AKADEMIA GÓRNICZO-HUTNICZA  
IM. STANISŁAWA STASZICA W KRAKOWIE**

# **Podstawy telekomunikacji**

## **Lab 3. Techniki i protokoły sieciowe, Wireshark**

**dr inż. Szymon Szott**

**Wydział Informatyki, Elektroniki i Telekomunikacji  
Katedra Telekomunikacji**

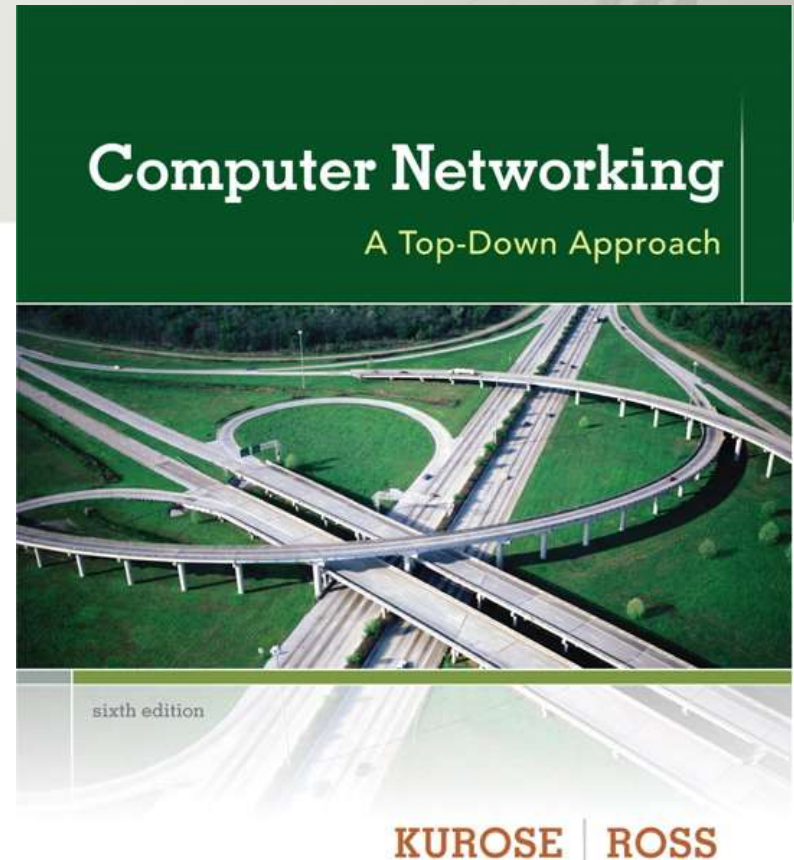
**Kraków, 2013-10-14**

## Przebieg zajęć

- Podstawy programu Wireshark
- Ćwiczenie z ICMP
  - ping
  - traceroute
- Ćwiczenie z Ethernet
  - Nagłówek Ethernet
  - Protokół ARP
- Propozycje dodatkowe ćwiczeń

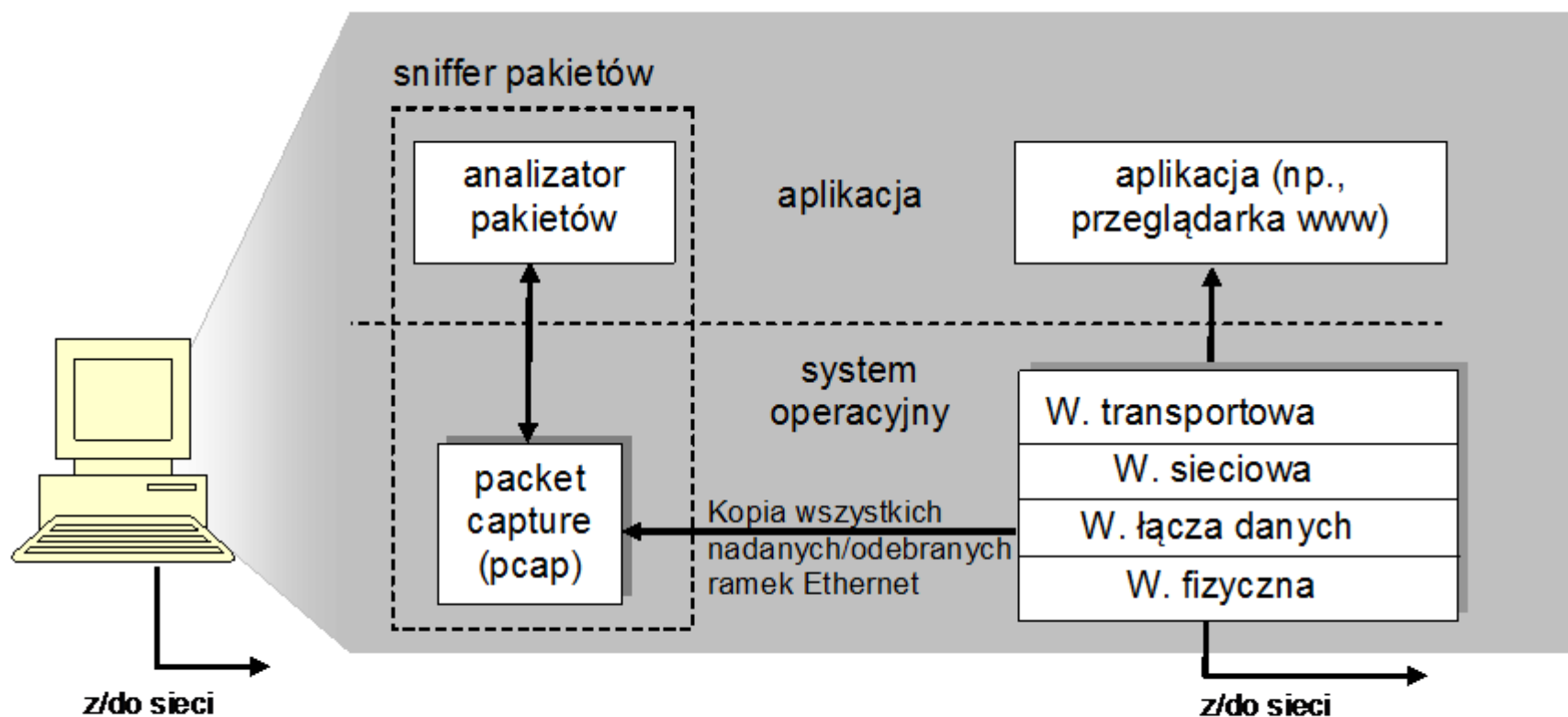
J.F. Kurose and K.W. Ross  
„Computer Networking:  
A Top-Down Approach”

<http://gaia.cs.umass.edu/wireshark-labs/>

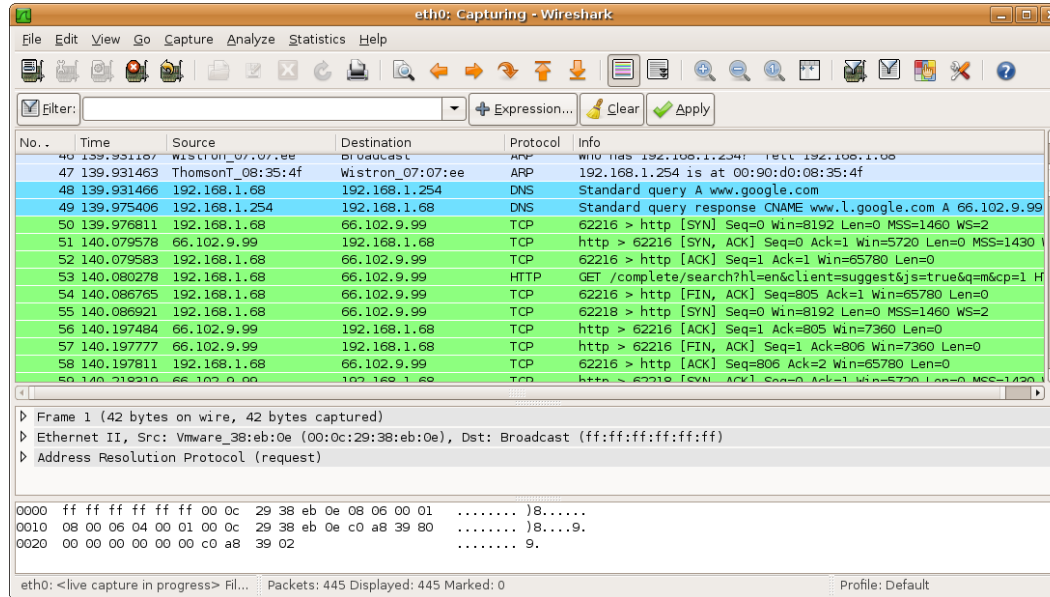


# **PODSTAWY PROGRAMU WIRESHARK**

# Sniffer pakietów

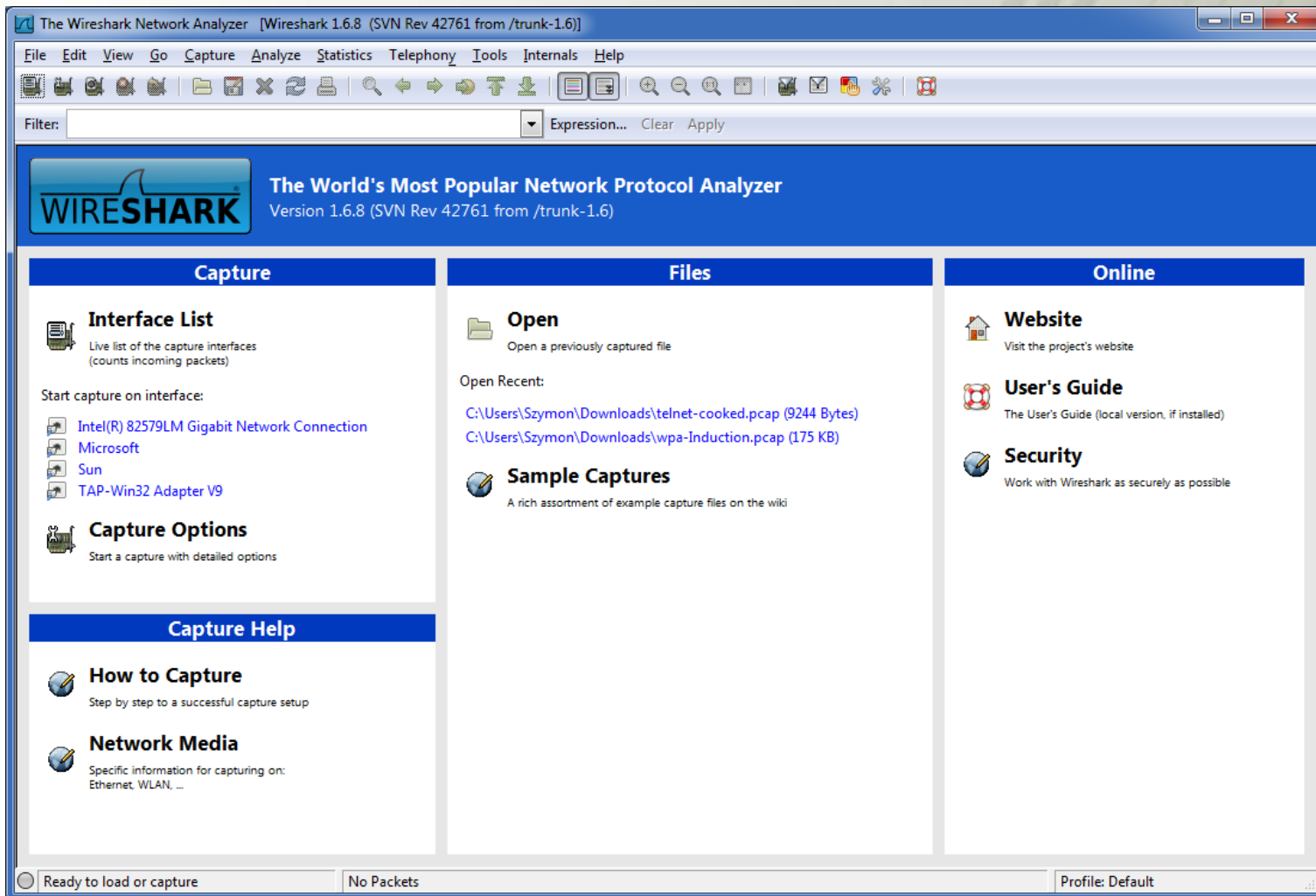


# Wireshark

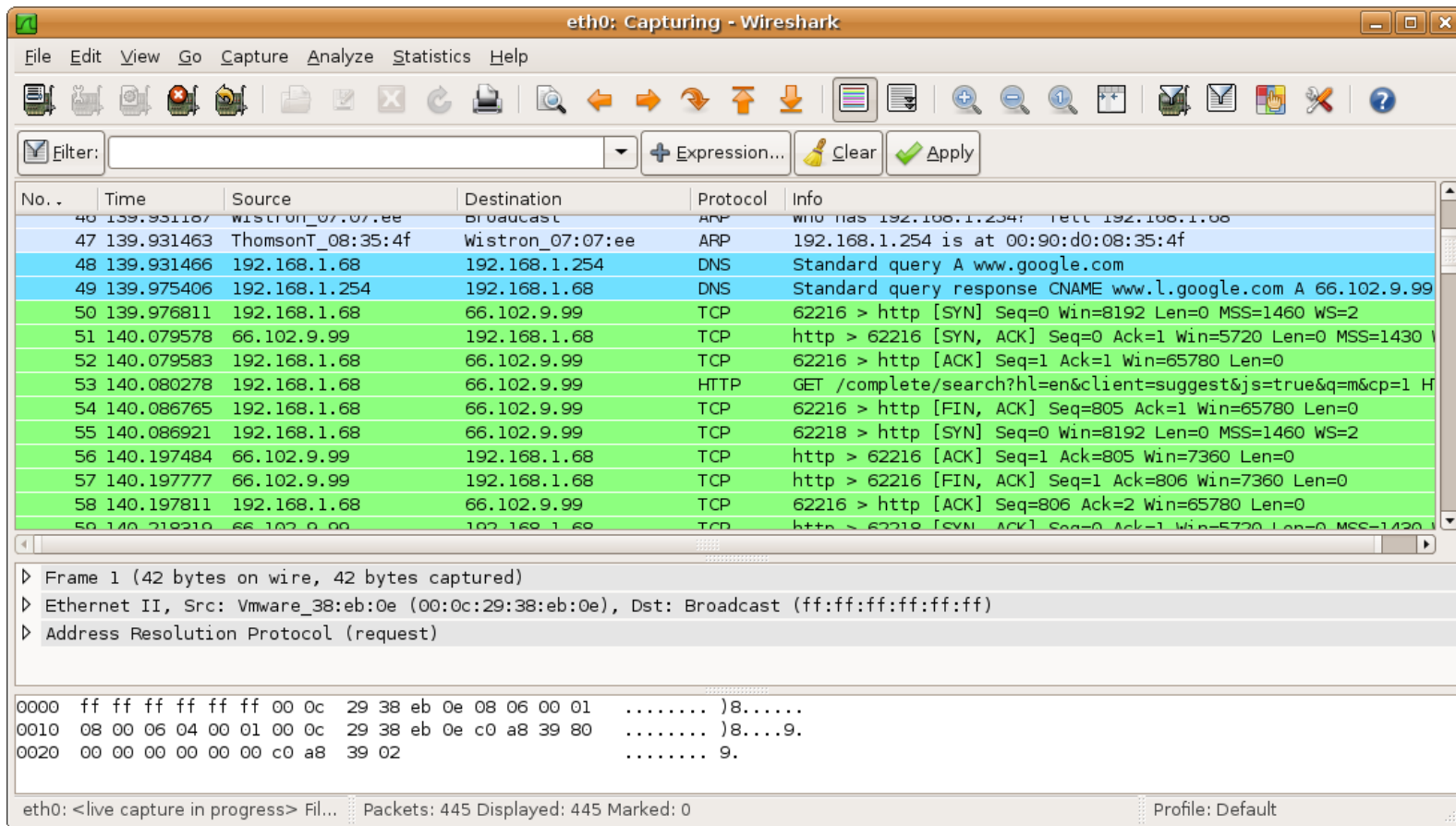


<http://www.wireshark.org/>

# Ekran powitalny



# Interfejs użytkownika



**eth0: Capturing - Wireshark**

File Edit View Go Capture Analyze Statistics Help

Filter:  + Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
46	139.931167	Wistron_07:07:ee	broadcast	ARP	who has 192.168.1.254? Tell 192.168.1.68
47	139.931463	ThomsonT_08:35:4f	Wistron_07:07:ee	ARP	192.168.1.254 is at 00:90:d0:08:35:4f
48	139.931466	192.168.1.68	192.168.1.254	DNS	Standard query A www.google.com
49	139.975406	192.168.1.254	192.168.1.68	DNS	Standard query response CNAME www.l.google.com A 66.102.9.99
50	139.976811	192.168.1.68	66.102.9.99	TCP	62216 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
51	140.079578	66.102.9.99	192.168.1.68	TCP	http > 62216 [SYN, ACK] Seq=0 Ack=1 Win=5720 Len=0 MSS=1430
52	140.079583	192.168.1.68	66.102.9.99	TCP	62216 > http [ACK] Seq=1 Ack=1 Win=65780 Len=0
53	140.080278	192.168.1.68	66.102.9.99	HTTP	GET /complete/search?hl=en&client=suggest&js=true&q=m&cp=1 H
54	140.086765	192.168.1.68	66.102.9.99	TCP	62216 > http [FIN, ACK] Seq=805 Ack=1 Win=65780 Len=0
55	140.086921	192.168.1.68	66.102.9.99	TCP	62218 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
56	140.197484	66.102.9.99	192.168.1.68	TCP	http > 62216 [ACK] Seq=1 Ack=805 Win=7360 Len=0
57	140.197777	66.102.9.99	192.168.1.68	TCP	http > 62216 [FIN, ACK] Seq=1 Ack=806 Win=7360 Len=0
58	140.197811	192.168.1.68	66.102.9.99	TCP	62216 > http [ACK] Seq=806 Ack=2 Win=65780 Len=0
59	140.218210	66.102.9.99	192.168.1.68	TCP	http > 62218 [SYN, ACK] Seq=0 Ack=1 Win=5720 Len=0 MSS=1430

Frame 1 (42 bytes on wire, 42 bytes captured)

Ethernet II, Src: Vmware\_38:eb:0e (00:0c:29:38:eb:0e), Dst: Broadcast (ff:ff:ff:ff:ff:ff)

Address Resolution Protocol (request)

```

0000  ff ff ff ff ff ff 00 0c 29 38 eb 0e 08 06 00 01  .... 8.....
0010  08 00 06 04 00 01 00 0c 29 38 eb 0e c0 a8 39 80  .... 8....9.
0020  00 00 00 00 00 00 c0 a8 39 02  .... 9.

```

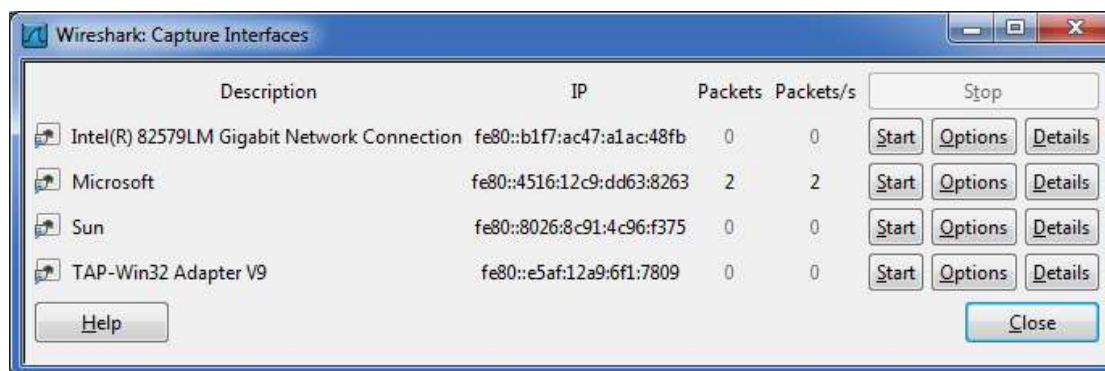
eth0: <live capture in progress> Fil... Packets: 445 Displayed: 445 Marked: 0 Profile: Default



# Ćwiczenie z podstaw Wiresharka

## Próbne przechwycenie pakietów

1. Uruchomić przeglądarkę Internetową
2. W Wireshark  
Capture → Interfaces → Start

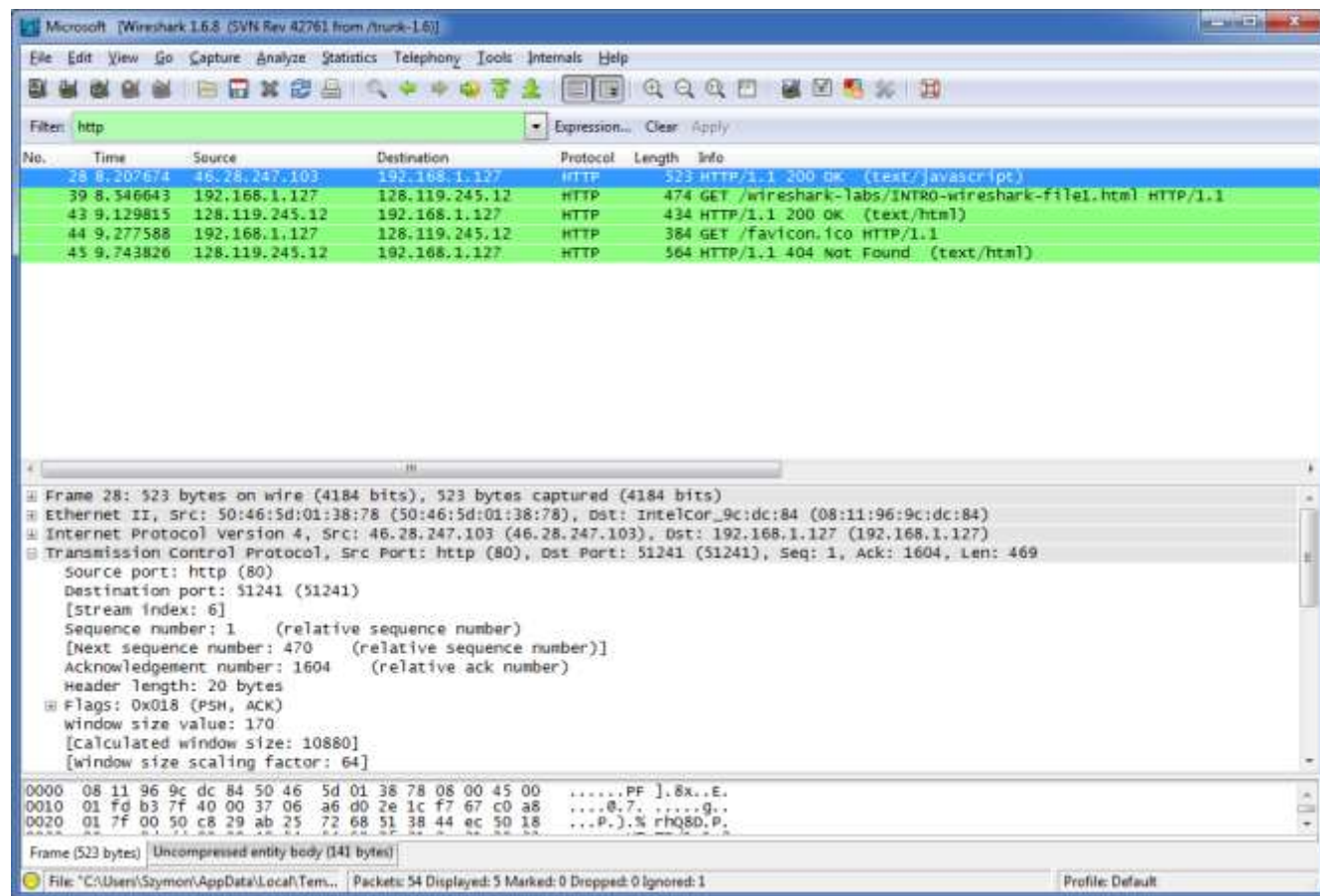


3. W przeglądarce otworzyć adres  
<http://kt.agh.edu.pl/~szott/pt/wireshark.html>
4. Wireshark: Capture → Stop

# Ćwiczenie z podstaw Wiresharka

## Filtrowanie pakietów

1. Filtr: http
2. Apply



# Ćwiczenie z podstaw Wiresharka

## Pytania

1. Jaką można zaobserwować enkapsulację protokołów?
2. Jakie są podstawowe komendy protokołu HTTP?
3. Ile czasu zajęło od wysłania zapytania HTTP do otrzymania odpowiedzi? Od czego to zależy?
4. Jaki jest Twój adres IP? Jaki jest adres serwera HTTP?
5. Oprócz zapytania o plik HTML pojawia się druga para komunikatów HTTP. Czego dotyczą?

# ĆWICZENIA Z ICMP

# ICMP

- Internetowy protokół komunikatów kontrolnych ICMP
  - Internet Control Message Protocol
- Protokół warstwy sieciowej
- Bezpołączeniowy
- Zastosowanie
  - Diagnostyka
  - Trasowanie (m.in. ping)

## Ćwiczenie z ICMP (ping)

- Użyj komendy ping w celu wygenerowania 10 wiadomości do wybranego serwera
  - Zaobserwuj pakiety w Wiresharku (filtr icmp)
1. Wireshark, Capture, Start
    - Przechwytywanie odbędzie się na ostatnio wybranym interfejsie
  2. W linii poleceń (Start, Uruchom, cmd.exe)
    - ping -n 10 [www.example.com](http://www.example.com)
  3. Wireshark, Capture, Stop

# Ćwiczenie z ICMP (ping)

## Pytania

1. Ile widać pakietów? Dlaczego?
2. Jaki jest Twój adres IP a jaki serwera docelowego?
3. Nagłówek ICMP
  - a) Ile bajtów zajmuje ten nagłówek?
  - b) Jak zakodowana jest komenda ping?
  - c) W jaki sposób realizowana jest detekcja błędów?
4. Co jest wysyłane w polu ICMP Data?
5. Jakie są podobieństwa i różnice w pakietach request i reply?

## Ćwiczenie z ICMP (traceroute)

- Użyj komendy `tracert` (Windows) lub `tracpath` (Linux) celu zbadania łączności z wybranym serwerem (np. `www.example.com`)
- Zaobserwuj pakiety w Wiresharku (filtr `icmp`)

### Pytania

1. Na jakiej zasadzie działa traceroute?
2. Czy są różnice w pakiecie echo request w stosunku do poprzedniego przykładu?
3. Jakie dodatkowe informacje zawiera komunikat błędu ICMP?
4. Czym się różnią ostatnie otrzymane pakiety ICMP?



# ETHERNET I ARP

# Ćwiczenie z analizy ramek Ethernet

1. Uruchom przeglądarkę, usuń pamięć podręczną
  - [Instrukcje dla Firefox](#)
  - [Instrukcja dla Chrome](#)
2. Wireshark, Capture, Start
3. Wejdź na stronę  
<http://kt.agh.edu.pl/~szott/pt/ethernet.txt>
4. Wireshark, Capture, Stop
5. Zapisz numer pakietu z HTTP GET
6. Wyłącz analizę protokołu IPv4
  - Wireshark, Analyze, Enabled protocols

## **Analiza ramek Ethernet**

### **Pytania**

Na podstawie wyznaczonej ramki Ethernet:

1. Jaki jest adres docelowy? Czy jest to adres serwera `kt.agh.edu.pl`?
2. Jaki jest adres źródłowy? Porównaj ze swoim adresem MAC.
3. Jaka jest trzecia informacja w nagłówku ramki Ethernet? O czym świadczy?
4. Za pomocą podglądu danych binarnych/ASCII, znajdź w polu Data komendę HTTP GET. Dlaczego pojawia się tak późno?

## Ćwiczenie z ARP

1. Wświetl tablicę ARP (`arp -a`) a następnie ją wyczyść (`arp -d *`)
2. Uruchom przeglądarkę, usuń pamięć podręczną
3. Wireshark, Capture, Start
4. Wejdź na stronę  
<http://kt.agh.edu.pl/~szott/pt/ethernet.txt>
5. Wireshark, Capture, Stop
6. Wyłącz analizę protokołu IP
  - Analyze, Enabled protocols
7. Włącz filtr arp

## Analiza ramek ARP

### Pytania

1. Jakie są adresy (źródłowy/docelowy) w ARP request?
2. Jaka jest wartość pola Type w nagłówce ramki Ethernet?
3. Na podstawie [formatu komunikatu ARP](#) odpowiedz jakie są wartości pól *opcode*, oraz *sender/target MAC/IP address*?
4. W którym polu pojawia się zapytanie? Porównaj z polem Info w liście pakietów Wiresharka.
5. Dla powyższych pytań, porównaj wartości pól w ramach ARP request i reply.

# **DODATKOWE ĆWICZENIA**

## Program tcpdump

- Program linii komend w systemach Linux
- Przykładowe zastosowanie

```
#tcpdump & ping wikipedia.org
```

```
05:25:44.622643 IP ai_k32 > rr.pmtpa.wikimedia.org:  
    ICMP echo request, id 37720, seq 2, length 64  
05:25:44.708387 IP rr.pmtpa.wikimedia.org > k26: ICMP  
    echo reply, id 37720, seq 2, length 64  
05:25:45.622624 IP ai_k32 > rr.pmtpa.wikimedia.org:  
    ICMP echo request, id 37720, seq 3, length 64  
05:25:45.708135 IP rr.pmtpa.wikimedia.org > k26: ICMP  
    echo reply, id 37720, seq 3, length 64
```

## Przechwytywanie zaszyfrowanych pakietów

- Przechwyć a następnie porównaj przechwycone ramki danych dla dwóch stron (jedna z http, druga z https)
- Przykładowo
  - <http://www.mbank.com.pl>
  - <https://www.mbank.com.pl>
- Jaka jest różnica w tym co można odczytać?



## Pozostałe ćwiczenia

- Na stronie <http://gaia.cs.umass.edu/wireshark-labs/> dostępne są dodatkowe ćwiczenia m.in. z
  - DNS
  - DHCP
  - TCP/UCP
  - IP
  - HTTP
  - SSL
- Dostępne są również pliki *trace* (.pcap) do w/w ćwiczeń

## Podsumowanie

- Podstawy programu Wireshark
- Ćwiczenia z
  - ICMP
  - Ethernet, ARP
- Opis dodatkowych ćwiczeń