

Sprawozdanie

1. Ping

1.1. Opis programu

Ping – jest to program służący do testowania połączeń sieciowych. Jego najczęstszym zastosowaniem jest sprawdzenie czy istnieje połączenie pomiędzy dwoma komputerami. Poza tym pozwala również na sprawdzenie opóźnień podczas wysyłania pakietów czy monitorowania ilości zagubionych pakietów. Podstawą jego działania jest użycie protokołu ICMP (Internet Control Message Protocol). Działanie programu polega na wysłaniu pakietu ‘ICMP echo request’ i oczekiwaniu na odpowiedź ‘ICMP echo reply’. Większość obecnych komputerów wysyła takie odpowiedzi. Jednak część serwisów zdecydowała się na wyłączenie tej opcji ze względów bezpieczeństwa, ponieważ potwierdzenie obecności hosta pod pewnym adresem IP może być potraktowane jako zidentyfikowanie go potencjalny cel ataku. Stąd wniosek, że są takie hosty, które pomimo bycia podłączonymi do sieci nie odpowiedzą na ‘echo request’.

1.2. Przykładowe użycie

Pingowanie australijskiego serwera.

```
C:\Users\Kamil>ping -i 22 canberra.com.au

Pinging canberra.com.au [110.34.55.6] with 32 bytes of data:
Reply from 110.34.55.6: bytes=32 time=340ms TTL=48
Reply from 110.34.55.6: bytes=32 time=339ms TTL=48
Reply from 110.34.55.6: bytes=32 time=339ms TTL=48
Reply from 110.34.55.6: bytes=32 time=339ms TTL=48

Ping statistics for 110.34.55.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 339ms, Maximum = 340ms, Average = 339ms
```

1.3. Liczenie ilości węzłów do danego serwera

Używając programu ping można sprawdzić ile węzłów znajduje się na trasie do danego serwera. Na każdym węźle wartość TTL jest zmniejszana o 1, więc można ręcznie sprawdzić graniczną wartość TTL dla którego pakiet dotrze do celu. Za pomocą opcji -i można ustalić wartość TTL wysyłanego pakietu. Swoje doświadczenia przeprowadziłem będąc we Wrocławiu.

```

C:\Users\Kamil>ping -i 22 nus.edu.sg

Pinging nus.edu.sg [137.132.21.27] with 32 bytes of data:
Reply from 137.132.21.27: TTL expired in transit.
Reply from 137.132.21.27: TTL expired in transit.
Reply from 137.132.21.27: TTL expired in transit.
Reply from 137.132.21.27: TTL expired in transit.

Ping statistics for 137.132.21.27:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

C:\Users\Kamil>ping -i 23 nus.edu.sg

Pinging nus.edu.sg [137.132.21.27] with 32 bytes of data:
Reply from 137.132.21.27: bytes=32 time=194ms TTL=43
Reply from 137.132.21.27: bytes=32 time=194ms TTL=43
Reply from 137.132.21.27: bytes=32 time=193ms TTL=43
Reply from 137.132.21.27: bytes=32 time=194ms TTL=43

Ping statistics for 137.132.21.27:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 193ms, Maximum = 194ms, Average = 193ms

```

Postanowiłem sprawdzić ilość węzłów na trasie do serwera nus.edu.sg (National University of Singapore). Wartość TTL równa 22 była za mała aby pakiet dotarł do celu. Jednak wartość 23 okazała się w porządku. Stąd wniosek że ilość węzłów na trasie do serwera nus.edu.sg wynosi 23.

1.4. Trasy tam i z powrotem

Teraz spróbuję ustalić czy trasa do serwera jest taka sama jak trasa powrotna. Najpierw zdecydowałem się sprawdzić ilość węzłów na trasie tam i z powrotem, ponieważ jeśli stwierdzę różnicę między ich ilością to będzie to oznaczać, że trasy te są różne. Do testów postanowiłem użyć dwóch serwerów: onet.pl (Polska) oraz wikipedia.org (USA).

Host	Trasa tam (TTL)	Trasa powrotna (TTL)
onet.pl	11	10
wikipedia.org	10	10

Kiedy serwer wysyła odpowiedź TTL przyjmuje wartość domyślną dla danego serwera. Są to wartości 32/64/128/255. W moim doświadczeniu pakiet zwrotny miał wartość TTL równą 54 (w obu przypadkach), więc stwierdzam, że początkowa wartość TTL była równa 64. Teraz ilość węzłów na trasie powrotnej mogę obliczyć jako różnicę tych wartości

Adres hosta	Ilość węzłów na trasie do hosta	Ilość węzłów na drodze powrotnej
onet.pl	11	10
wikipedia.org	10	10

Wnioskiem z doświadczenia jest, że trasy powrotne nie są takie same, ponieważ ilość węzłów na trasie do serwera onet.pl była inna niż na trasie powrotnej. W przypadku serwera wikipedia.org widząc taką samą ilość węzłów na trasie tam i z powrotem nie jesteśmy w stanie

jednoznacznie powiedzieć czy trasa była taka sama, wiemy jedynie, że jej długość była taka sama.

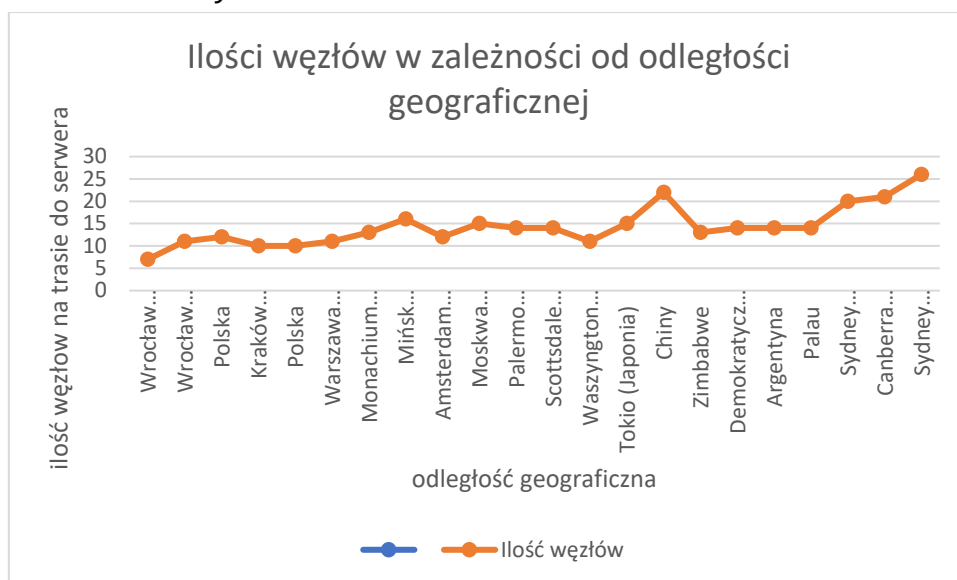
1.5. Ilość węzłów, czas odpowiedzi, a odległość geograficzna

Teraz sprawdzę zależność ilości węzłów i czasów odpowiedzi od odległości geograficznej. W tym celu będę sprawdzać ilość węzłów na trasach do serwerów z różnych regionów świata.

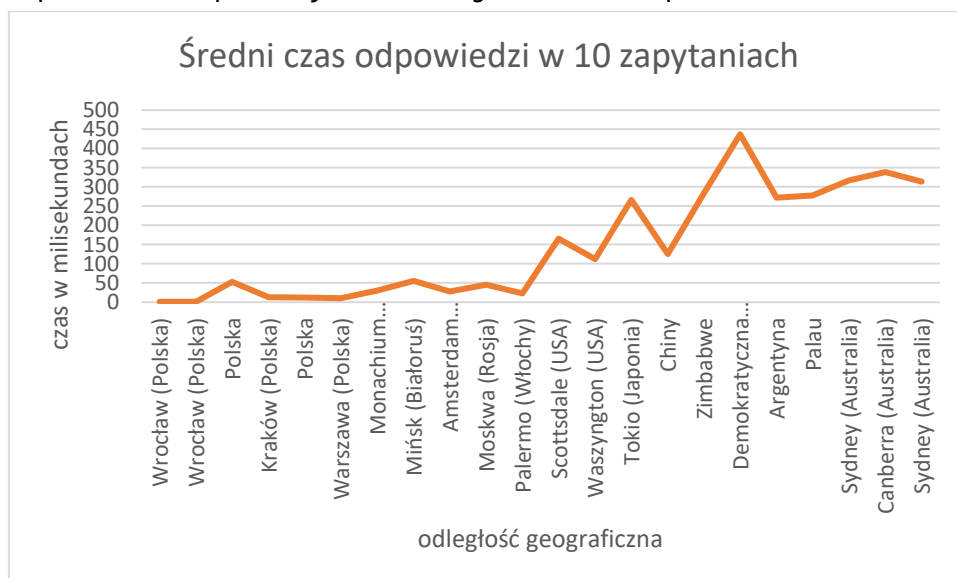
lokalizacja	Adres hosta	Czas(min max avg)*	Ilość węzłów (do hosta)	Ilość węzłów (od hosta) _[TTL]
Wrocław (Polska)	s.student.pwr.edu.pl	1 2 1	7	6[58]
Wrocław (Polska)	jsos.pwr.edu.pl	1 2 1	11	8[56]
Polska	wykop.pl	52 65 53	12	11[53]
Kraków (Polska)	krakow.pl	13 15 13	10	8[56]
Polska	wp.pl	12 13 12	10	8[56]
Warszawa (Polska)	pw.edu.pl	10 11 10	11	9[55]
Monachium (Niemcy)	tum.de	30 32 30	13	12[243]
Mińsk (Białoruś)	bsu.by	53 54 55	16	15[113]
Amsterdam (Holandia)	uva.nl	27 36 28	12	12[52]
Moskwa (Rosja)	msu.ru	44 46 45	15	13[51]
Palermo (Włochy)	palermo.repubblica.it	23 24 23	14	9[55]
Scottsdale (USA)	franklinepb.com	164 166 165	14	12[116]
Waszyngton (USA)	ryanair.com	110 117 112	11	11[53]
Tokio (Japonia)	tokyotimes.com	266 266 266	15	15[49]
Chiny	114.114.115.119	125 127 126	22	-[?]
Zimbabwe	77.246.56.247	284 285 284	13	12[243]
Demokratyczna Republika Kongo	31.209.128.129	429 444 437	14	14[50]
Argentyna	2.18.56.0	271 274 272	14	14[50]
Palau	103.30.248.18	278 279 278	14	15[113]
Sydney (Australia)	victoria.ac.nz	317 318 317	20	16[48]
Canberra (Australia)	canberra.com.au	339 346 339	21	16[48]
Sydney (Australia)	sydney.edu.au	314 315 314	26	25[230]

*wykonywane jest 10 zapytań (opcja -n 10)

Wnioskiem z powyższej tabeli jest to, że ilość pakietów nie zależy w znacznym stopniu od odległości geograficznej. Różnica była widoczna dopiero przy połączeniach z Sydney. Połączenie z Argentyną nie różniło się w ilości pakietów od połączenia z Monachium mimo bardzo znacznej różnicy w odległościach. Widać jednak było nieznacznie zmniejszoną ilość węzłów na trasie do wrocławskich serwerów – tych które są blisko. Połączenie do jednego z nich wymagało przejścia jedynie przez 7 routerów, co może być związane z faktem wykonywania tego doświadczenia na łączu internetowym w akademiku PWr.



Innym ważnym wnioskiem jest zależność czasu odpowiedzi od odległości geograficznej. Czas odpowiedzi był wyraźnie krótszy dla serwerów blisko, i wyraźnie dłuższy dla tych odległych. Jednak występowały też pewne odchylenia – polski serwer wykop.pl odpowiadał w czasie dłuższym niż ten w Palermo. Inny przykład to serwer na małej wyspie na Pacyfiku (Palau), który odpowiadał w podobnym czasie jak ten w Japonii.



1.6. Chińskie serwery

Na szczególną uwagę zasługuje zachowanie jednego z chińskich serwerów.

```
C:\Users\Kamil>ping -n 10 114.114.115.119

Pinging 114.114.115.119 with 32 bytes of data:
Reply from 114.114.115.119: bytes=32 time=126ms TTL=78
Reply from 114.114.115.119: bytes=32 time=126ms TTL=79
Reply from 114.114.115.119: bytes=32 time=126ms TTL=71
Reply from 114.114.115.119: bytes=32 time=127ms TTL=62
Reply from 114.114.115.119: bytes=32 time=126ms TTL=78
Reply from 114.114.115.119: bytes=32 time=126ms TTL=84
Reply from 114.114.115.119: bytes=32 time=126ms TTL=73
Reply from 114.114.115.119: bytes=32 time=126ms TTL=59
Reply from 114.114.115.119: bytes=32 time=126ms TTL=82
Reply from 114.114.115.119: bytes=32 time=126ms TTL=81

Ping statistics for 114.114.115.119:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 126ms, Maximum = 127ms, Average = 126ms
```

W każdej wiadomości zwrotnej serwer miał inną wartość TTL, co prawdopodobnie jest skutkiem umyślnego zaburzania tej wartości.

1.7. Wielkość pakietów, a czas przesyłania

W celu sprawdzenia zależności czasu przesyłania pakietu od wielkości pakietów wysyłałem pakiety o różnych rozmiarach do różnych serwerów. Jednym z serwerów była wikipedia.org (Kalifornia, USA), drugim uni.wroc.pl (Wrocław, Polska), a trzecim canberra.com.au (Sydney, Australia)

Tabela dla wikipedia.org.

Wielkość pakietu w bajtach	Czas(min max avg)*	Trasa do hosta	Trasa od hosta [TTL]
32	25 26 25	10	10[54]
500	25 27 25	10	10[54]
1000	26 26 26	10	10[54]
5000	27 28 27	10	10[54]
8000	27 28 27	10	10[54]
16000	29 30 29	10	10[54]
32000	32 33 32	10	10[54]
48000	34 36 34	10	10[54]
65500	37 40 37	10	10[54]

*wykonywane jest 10 zapytań (opcja -n 10)

Różnica między pierwszym, a drugim jest głównie w odległości geograficznej, jeśli chodzi o odległość w ilości węzłów to są one podobne. Trzeci host jest zdecydowanie dalej niż dwa pozostałe zarówno pod względem odległości geograficznej jak i ilości węzłów.

Tabela dla uni.wroc.pl.

Wielość pakietu w bajtach	Czas(min max avg)*	Trasa do hosta	Trasa od hosta [TTL]
32	1 2 1	11	9[55]
500	1 1 1	11	9[55]
1000	1 3 1	11	9[55]
5000	2 3 2	11	9[55]
8000	2 4 2	11	9[55]
16000	4 5 4	11	9[55]
32000	7 8 7	11	9[55]
48000	10 11 10	11	9[55]
65500	13 15 13	11	9[55]

*wykonywane jest 10 zapytań (opcja -n 10)

Tabela dla canberra.com.au.

Wielość pakietu w bajtach	Czas(min max avg)*	Trasa do hosta	Trasa od hosta [TTL]
32	339 340 339	21	16[48]
500	339 349 340	21	16[48]
1000	339 344 339	21	16[48]
5000	339 341 339	21	16[48]
8000	342 343 342	21	16[48]
16000	344 345 344	21	16[48]
32000	348 350 349	21	16[48]
48000	353 354 353	21	16[48]
65500	358 360 358	21	16[48]

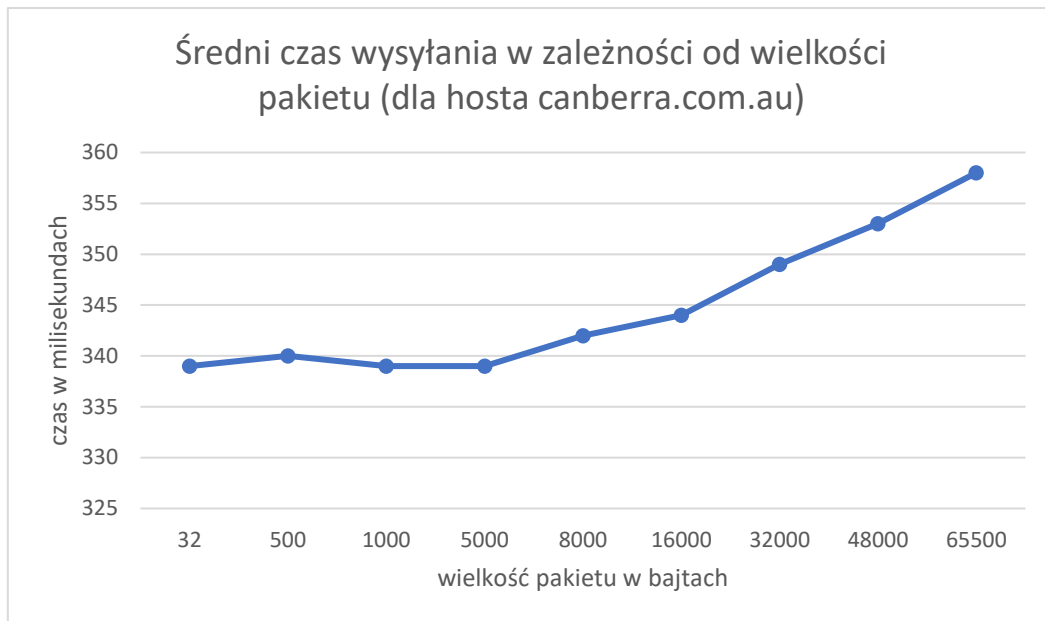
*wykonywane jest 10 zapytań (opcja -n 10)

Czas przesyłania dla pakietów poniżej 5000B praktycznie się nie różnił. Drobne różnice zaczęły się pojawiać dla tych od 5000B do 16000B. Dopiero od tych wartości czas wysyłania zaczął rosnąć. Ilość węzłów na trasie do serwera i z powrotem w żadnym przypadku nie uległa zmianie.

Maksymalny rozmiar wiadomości to 65500B.

```
C:\Users\Kamil>ping -n 10 -l 65501 canberra.com.au
Bad value for option -l, valid range is from 0 to 65500.
```

Maksymalna wielkość pakietu dla której nie jest potrzebna fragmentacja to 1472B. Wnioskiem z doświadczenia jest to, że fragmentacja wpływa nieznacznie na czas przesyłania pakietu. Dopiero przy bardzo dużych pakietach gdzie nastąpiło większe pofragmentowanie różnica w czasie jest widoczna.



Próba wysłania pakietu większego niż 1472 nie fragmentując go.

```
C:\WINDOWS\system32>ping -l 1472 -f wikipedia.org

Pinging wikipedia.org [91.198.174.192] with 1472 bytes of data:
Reply from 91.198.174.192: bytes=1472 time=27ms TTL=54
Reply from 91.198.174.192: bytes=1472 time=27ms TTL=54
Reply from 91.198.174.192: bytes=1472 time=27ms TTL=54
Reply from 91.198.174.192: bytes=1472 time=27ms TTL=54

Ping statistics for 91.198.174.192:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 27ms, Maximum = 27ms, Average = 27ms

C:\WINDOWS\system32>ping -l 1473 -f wikipedia.org

Pinging wikipedia.org [91.198.174.192] with 1473 bytes of data:
Packet needs to be fragmented but DF set.
Packet needs to be fragmented but DF set.
Packet needs to be fragmented but DF set.
Packet needs to be fragmented but DF set.

Ping statistics for 91.198.174.192:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

Pinging canberra.com.au [110.34.55.6] with 65500 bytes of data:
Reply from 110.34.55.6: bytes=65500 time=359ms TTL=48
Reply from 110.34.55.6: bytes=65500 time=358ms TTL=48
Reply from 110.34.55.6: bytes=65500 time=358ms TTL=48
Reply from 110.34.55.6: bytes=65500 time=358ms TTL=48
Reply from 110.34.55.6: bytes=65500 time=358ms TTL=48
Reply from 110.34.55.6: bytes=65500 time=359ms TTL=48
Reply from 110.34.55.6: bytes=65500 time=358ms TTL=48
Reply from 110.34.55.6: bytes=65500 time=358ms TTL=48
Reply from 110.34.55.6: bytes=65500 time=360ms TTL=48
Reply from 110.34.55.6: bytes=65500 time=358ms TTL=48

Ping statistics for 110.34.55.6:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 358ms, Maximum = 360ms, Average = 358ms
```

2. Traceroute (Tracert)

2.1. Opis programu

Traceroute (Tracert w systemach Windows) – jest to program służący do badania drogi pakietów w sieci Internet Protocol (IP). Pozwala na wyświetlenie punktów na trasie pakietu wraz z opóźnieniami jakie zaszły na każdym z nich. Używając tego programu można stwierdzić np. przez jakie kraje szedł pakiet. Adresy IP kolejnych routerów ustalane są poprzez wysyłanie pakietów z odpowiednimi wartościami TTL. Przykładowo adres pierwszego routera uzyskiwany jest poprzez wysłanie pakietu z TTL równym 1. Wtedy na pierwszym routerze wartość TTL zostaje zmniejszona do 0, a to spowoduje nadanie informacji o odrzuceniu pakietu. W tej wiadomości znajduje się adres pierwszego routera. Adresy kolejnych routerów pozyskiwane są analogicznie. Jest to ta sama metoda, którą w poprzedniej części używałem ręcznie za pomocą programu ping. Ja w celu realizacji doświadczeń korzystałem z Tracert.

2.2. Badanie trasy

Na początku postanowiłem sprawdzić trasę serwera bardzo bliskiego mi geograficznie - jsos.pwr.edu.pl.

```
C:\Users\Kamil>tracert jsos.pwr.edu.pl

Tracing route to jsos.pwr.edu.pl [156.17.28.249]
over a maximum of 30 hops:

  0  <1 ms    1 ms     1 ms    156.17.240.254
  1  1 ms     1 ms     *       234.ds.pwr.wroc.pl [156.17.229.234]
  2  20 ms    20 ms    20 ms    ik194.ds.pwr.wroc.pl [156.17.229.194]
  3  21 ms    22 ms    93 ms    ik193.ds.pwr.wroc.pl [156.17.229.193]
  4  3 ms     1 ms     1 ms     156.17.229.255
  5  1 ms     1 ms     1 ms     pwr-zds-centrum3-vprn.wask.wroc.pl [156.17.254.41]
  6  1 ms     <1 ms    1 ms     rolnik2-centrum.wask.wroc.pl [156.17.254.65]
  7  2 ms     2 ms     1 ms     wazniak-rolnik.wask.wroc.pl [156.17.254.140]
  8  1 ms     1 ms     1 ms     z-wask2-do-pwr2.pwrnet.pwr.wroc.pl [156.17.18.244]
  9  3 ms     5 ms     1 ms     156.17.33.1
 10  2 ms     1 ms     1 ms     156.17.28.249

Trace complete.
```

Teraz sprawdźmy czy trasy wyznaczone przeze mnie ręcznie w poprzedniej części sprawozdania będą takie same przy użyciu Tracert. Porównam trasy dwóch do trzech serwerów tokyotimes.com, pw.edu.pl i msu.ru.

Adres hosta	Trasa do serwera zmierzona ręcznie przy użyciu programu Ping	Trasa do serwera zmierzona przez Tracert
tokyotimes.com	15	15
pw.edu.pl	11	11
msu.ru	15	15


```
C:\Users\Kamil>tracert tokyotimes.com
```

```
Tracing route to tokyotimes.com [106.185.26.82]  
over a maximum of 30 hops:
```

1	<1 ms	1 ms	<1 ms	156.17.240.254
2	<1 ms	1 ms	*	234.ds.pwr.wroc.pl [156.17.229.234]
3	14 ms	21 ms	20 ms	ik193.ds.pwr.wroc.pl [156.17.229.193]
4	2 ms	1 ms	1 ms	156.17.229.255
5	1 ms	1 ms	1 ms	pwr-zds-centrum3-vprn.wask.wroc.pl [156.17.254.41]
6	5 ms	5 ms	5 ms	z-wroclawia.poznan-gw3.10Gb.rtr.pionier.gov.pl [212.191.224.105]
7	10 ms	10 ms	10 ms	ae100.edge3.Berlin1.Level3.net [212.162.10.81]
8	*	*	165 ms	ae-1-2.ear1.SanJose3.Level3.net [4.69.209.149]
9	166 ms	165 ms	165 ms	4.53.212.94
10	165 ms	165 ms	166 ms	pajbb001.int-gw.kddi.ne.jp [111.87.3.125]
11	276 ms	276 ms	275 ms	106.187.13.9
12	269 ms	265 ms	265 ms	27.85.134.174
13	274 ms	272 ms	282 ms	cm-fcu204.kddnet.ad.jp [124.215.194.181]
14	273 ms	271 ms	272 ms	124.215.199.170
15	266 ms	266 ms	266 ms	li721-82.members.linode.com [106.185.26.82]

```
Trace complete.
```

```
C:\Users\Kamil>tracert pw.edu.pl
```

```
Tracing route to pw.edu.pl [194.29.151.5]  
over a maximum of 30 hops:
```

1	1 ms	1 ms	1 ms	156.17.240.254
2	<1 ms	2 ms	*	234.ds.pwr.wroc.pl [156.17.229.234]
3	50 ms	20 ms	21 ms	ik194.ds.pwr.wroc.pl [156.17.229.194]
4	16 ms	18 ms	47 ms	ik193.ds.pwr.wroc.pl [156.17.229.193]
5	1 ms	1 ms	1 ms	156.17.229.255
6	2 ms	1 ms	<1 ms	pwr-zds-centrum3-vprn.wask.wroc.pl [156.17.254.41]
7	13 ms	5 ms	5 ms	z-wroclawia.poznan-gw3.10Gb.rtr.pionier.gov.pl [212.191.224.105]
8	9 ms	9 ms	10 ms	z-poznan-gw3.nask.10Gb.rtr.pionier.gov.pl [212.191.224.74]
9	10 ms	9 ms	10 ms	148.81.253.70
10	10 ms	21 ms	9 ms	194.29.132.162
11	10 ms	10 ms	10 ms	www5.coi.pw.edu.pl [194.29.151.5]

```
Trace complete.
```

```
C:\Users\Kamil>tracert msu.ru
```

```
Tracing route to msu.ru [188.44.50.103]  
over a maximum of 30 hops:
```

1	1 ms	1 ms	1 ms	156.17.240.254
2	<1 ms	1 ms	*	234.ds.pwr.wroc.pl [156.17.229.234]
3	19 ms	21 ms	15 ms	ik194.ds.pwr.wroc.pl [156.17.229.194]
4	6 ms	16 ms	20 ms	ik193.ds.pwr.wroc.pl [156.17.229.193]
5	2 ms	2 ms	1 ms	156.17.229.255
6	1 ms	2 ms	<1 ms	pwr-zds-centrum3-vprn.wask.wroc.pl [156.17.254.41]
7	5 ms	5 ms	5 ms	z-wroclawia.poznan-gw3.10Gb.rtr.pionier.gov.pl [212.191.224.105]
8	15 ms	15 ms	15 ms	de-hmb.nordu.net [109.105.98.124]
9	32 ms	31 ms	31 ms	fi-csc2.nordu.net [109.105.97.76]
10	37 ms	36 ms	36 ms	ndn-gw2.runnet.ru [109.105.102.58]
11	37 ms	38 ms	37 ms	spb-bm18-1-gw.runnet.ru [185.141.124.140]
12	45 ms	47 ms	45 ms	msk-m9-1-gw.runnet.ru [185.141.124.144]
13	45 ms	46 ms	45 ms	msu.msk.runnet.ru [194.190.254.118]
14	46 ms	45 ms	46 ms	93.180.0.191
15	47 ms	46 ms	46 ms	188.44.50.103

Widać, że te serwery znajdowały się w Rosji,
a ten ostatni konkretniej w Moskwie

```
Trace complete.
```

Wnioskiem jest to, że długość tras była taka sama niezależnie od tego czy mierzyłem ją ręcznie przy użyciu Pinga czy mierzył to Tracert. Trasa do tokyotimes.com biegła przez Berlin, a następnie przez San Jose w Kalifornii lub na Kostaryce*. Docelowy serwer znajduje się w Japonii. *miasto o takiej nazwie występuje w obu tych lokalizacjach.

2.3. Próba znalezienia nietypowych zachowań lub anomalii

Na koniec zabawy Tracert postanowiłem sprawdzić jak program zachowa się z serwerem w Chinach (114.114.115.119), który za każdym razem zwracał inną wartość TTL w komunikacie zwrotnym. Jednak trasa do wyznaczona przez Tracert okazała się taka sama. Poźniej sprawdziłem jak Tracert zachowa się z adresem biedronka.pl. Trasa okazała się niemożliwa do wyznaczenia.

```
C:\Users\Kamil>tracert -h 150 biedronka.pl

Tracing route to biedronka.pl [176.31.131.204]
over a maximum of 150 hops:

  1  <1 ms    1 ms    1 ms    156.17.240.254
  2  1 ms     <1 ms   *       234.ds.pwr.wroc.pl [156.17.229.234]
  3  21 ms    15 ms   18 ms   ik193.ds.pwr.wroc.pl [156.17.229.193]
  4  2 ms     4 ms    1 ms    156.17.229.255
  5  1 ms     <1 ms  1 ms    pwr-zds-centrum3-vprn.wask.wroc.pl [156.17.254.41]
  6  1 ms     1 ms    1 ms    karkonosz-centrum-rtr.wask.wroc.pl [156.17.254.111]
  7  1 ms     1 ms    3 ms    sniezka-karkonosz.wask.wroc.pl [156.17.250.222]
  8  5 ms     5 ms    5 ms    z-Wroclaw-COM.poznan-gw2-amsix.rtr.pionier.gov.pl [212.191.237.121]
  9  *        *        *       Request timed out.
 10  *        *        *       Request timed out.
 11  *        *        *       Request timed out.
 12  32 ms    33 ms   34 ms   po7.rbx-s6-6k.fr.eu [178.33.100.110]
 13  38 ms    33 ms   32 ms   176.31.131.201
 14  *        *        *       Request timed out.
 15  *        *        *       Request timed out.
 16  *        *        *       Request timed out.
 17  *        *        *       Request timed out.
 18  *        *        *       Request timed out.

 71  *        *        *       Request timed out.
 72  *        *        *       Request timed out.
 73  *        *        *       Request timed out.
 74  *        *        *       Request timed out.
 75  *        *        *       Request timed out.
 76  *        *        *       Request timed out.

139  *        *        *       Request timed out.
140  *        *        *       Request timed out.
141  *        *        *       Request timed out.
142  *        *        *       Request timed out.
143  *        *        *       Request timed out.
144  *        *        *       Request timed out.
145  *        *        *       Request timed out.
146  *        *        *       Request timed out.
147  *        *        *       Request timed out.
148  *        *        *       Request timed out.
149  *        *        *       Request timed out.
150  *        *        *       Request timed out.

Trace complete.
```

Podaję, że jest przyczyną np. umyślnego zapętlenia. Udało mi się znaleźć inny serwer, na którym wyraźnie było widać zapętlenie trasy.

```

C:\Users\Kamil>tracert 66.249.234.114

Tracing route to addr-66.249.234.114.nptpop-cmts02-dial-sub.rdns-bnin.net [66.249.234.114]
over a maximum of 30 hops:

  1    4 ms    3 ms    3 ms  192.168.0.1
  2    *      *      *      Request timed out.
  3   33 ms   33 ms   33 ms  pl-ktw01a-rc1-ae18-0.aorta.net [84.116.253.129]
  4   38 ms   49 ms   34 ms  de-fra04d-rc1-ae30-0.aorta.net [84.116.137.41]
  5   34 ms   34 ms   32 ms  de-fra01b-ri2-ae32-0.aorta.net [84.116.134.190]
  6   35 ms   32 ms   34 ms  213.46.177.138
  7   46 ms   48 ms   48 ms  be2814.ccr42.ams03.atlas.cogentco.com [130.117.0.141]
  8   63 ms   54 ms   57 ms  be2183.ccr22.lpl01.atlas.cogentco.com [154.54.58.69]
  9  356 ms  146 ms  140 ms  be3043.ccr22.ymq01.atlas.cogentco.com [154.54.44.166]
 10  353 ms  155 ms  351 ms  be3260.ccr32.yyz02.atlas.cogentco.com [154.54.42.89]
 11  300 ms  154 ms  352 ms  be2994.ccr22.cle04.atlas.cogentco.com [154.54.31.233]
 12  313 ms  154 ms  353 ms  be3745.rcr51.tol01.atlas.cogentco.com [154.54.30.130]
 13  337 ms  154 ms  353 ms  be3743.rcr21.sbn01.atlas.cogentco.com [66.28.4.229]
 14  328 ms  158 ms  348 ms  38.104.216.226
 15  358 ms  156 ms  350 ms  npt-edg-rt3.bnin.net [66.170.44.23]
 16  329 ms  195 ms  312 ms  npt-edg-rt1.bnin.net [66.170.44.1]
 17  356 ms  161 ms  346 ms  npt-edg-rt3.bnin.net [66.170.44.23]
 18  340 ms  162 ms  345 ms  npt-edg-rt1.bnin.net [66.170.44.1]
 19  345 ms  158 ms  349 ms  npt-edg-rt3.bnin.net [66.170.44.23]
 20  358 ms  156 ms  350 ms  npt-edg-rt1.bnin.net [66.170.44.1]
 21  356 ms  163 ms  344 ms  npt-edg-rt3.bnin.net [66.170.44.23]
 22  334 ms  162 ms  344 ms  npt-edg-rt1.bnin.net [66.170.44.1]
 23  358 ms  174 ms  333 ms  npt-edg-rt3.bnin.net [66.170.44.23]
 24  333 ms  166 ms  341 ms  npt-edg-rt1.bnin.net [66.170.44.1]
 25  332 ms  168 ms  159 ms  npt-edg-rt3.bnin.net [66.170.44.23]
 26  352 ms  171 ms  336 ms  npt-edg-rt1.bnin.net [66.170.44.1]
 27  351 ms  158 ms  161 ms  npt-edg-rt3.bnin.net [66.170.44.23]
 28  355 ms  164 ms  159 ms  npt-edg-rt1.bnin.net [66.170.44.1]
 29  351 ms  158 ms  349 ms  npt-edg-rt3.bnin.net [66.170.44.23]
 30  346 ms  157 ms  349 ms  npt-edg-rt1.bnin.net [66.170.44.1]

```

Na poniższym zrzucie ekranu widać, że pakiet nie może dotrzeć do celu nawet przy wartości początkowej TTL równej 255 (maksymalnej).

```

C:\Users\Kamil>ping -i 255 66.249.234.114

Pinging 66.249.234.114 with 32 bytes of data:
Reply from 66.170.44.23: TTL expired in transit.
Reply from 66.170.44.23: TTL expired in transit.
Reply from 66.170.44.23: TTL expired in transit.
Reply from 66.170.44.23: TTL expired in transit.

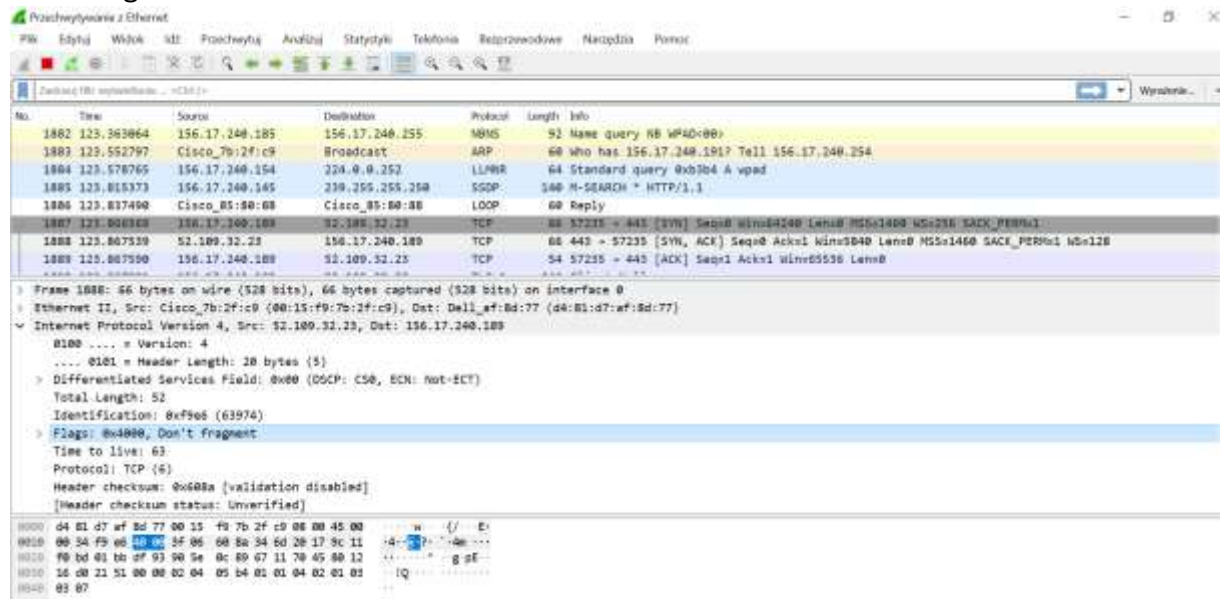
Ping statistics for 66.249.234.114:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

```

3. Wireshark

3.1. Opis programu

Wireshark – jest to program pozwalający analizować wszystkie pakiety wchodzące i wychodzące. Jest w stanie rozpoznać i dekodować wiele różnych protokołów komunikacyjnych. Jedną z jego głównych zalet jest obecność interfejsu graficznego. Pozwala na edytowanie i zapisywanie przechwyconych pakietów. Sam Wireshark nie przechwytytuje pakietów, używa do tego innego narzędzia np. nmap (w moim przypadku). Wireshark jest silnym narzędziem wykorzystywanym między innymi do zarządzania sieciami, śledzenia pakietów przez hakerów czy różne służby bezpieczeństwa oraz do prac nad protokołami komunikacyjnymi. Poniżej zrzut ekranu, na którym widać, że przechwycony pakiet TCP został wysłany z flagą mówiącą aby go nie fragmentować.



3.2. Przechwycenie pakietów ICMP

Spróbowałem przechwycić pakiety wysyłane przez program ping.

```
C:\Users\Kamil>ping -n 1 spotify.com

Pinging spotify.com [104.199.64.136] with 32 bytes of data:
Reply from 104.199.64.136: bytes=32 time=28ms TTL=43

Ping statistics for 104.199.64.136:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 28ms, Maximum = 28ms, Average = 28ms
```

Na poniższym zrzucie ekranu widać wysłany pakiet 'echo request' i 'echo reply'. Widać też, że IP docelowego serwera: 104.199.64.136 na obu zrzutach jest takie same stąd wiem, że przechwyciłem właściwy pakiet.

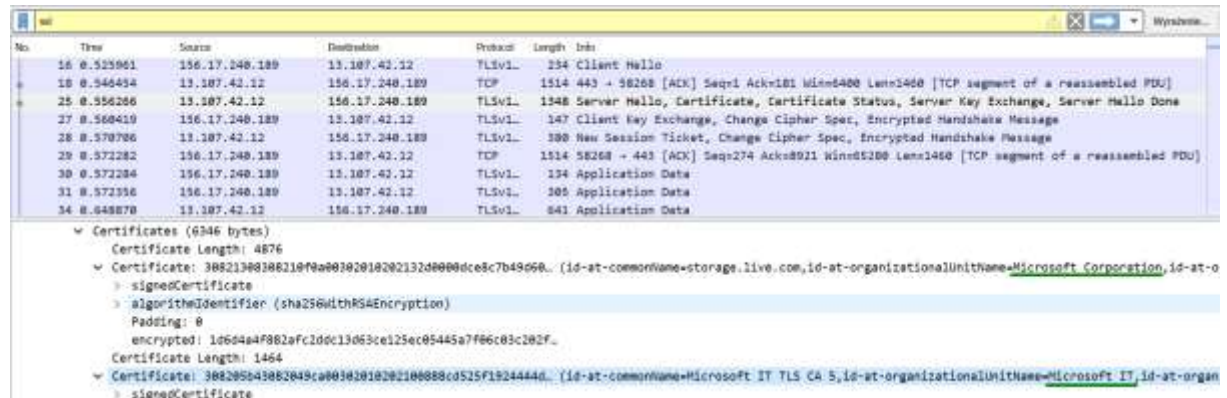
No.	Time	Source	Destination	Protocol	Length	Info
263	10.954051	31.13.72.8	156.17.240.189	TLSv1.	82	Application Data
264	10.994872	156.17.240.189	31.13.72.8	TCP	54	57097 → 443 [ACK] Seq=65 Ack=57 Win=2052 Len=0
265	11.608316	156.17.240.159	239.255.255.250	SSDP	216	M-SEARCH * HTTP/1.1
266	11.675372	Cisco_7b:2f:c9	Broadcast	ARP	60	Who has 156.17.240.189? Tell 156.17.240.254
267	11.832512	156.17.240.184	230.0.0.1	UDP	92	64999 → 6666 Len=50
268	11.970567	156.17.240.189	104.199.64.136	ICMP	74	Echo (ping) request id=0x0001, seq=4013/44303, ttl=128 (reply in 269)
269	11.999297	104.199.64.136	156.17.240.189	ICMP	74	Echo (ping) reply id=0x0001, seq=4013/44303, ttl=43 (request in 268)
270	12.609306	156.17.240.159	239.255.255.250	SSDP	216	M-SEARCH * HTTP/1.1
271	12.833173	156.17.240.184	230.0.0.1	UDP	92	64999 → 6666 Len=50
Frame 268: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0						
Ethernet II, Src: Dell_af:8d:77 (d4:81:d7:af:8d:77), Dst: Cisco_7b:2f:c9 (00:15:f9:7b:2f:c9)						
Internet Protocol Version 4, Src: 156.17.240.189, Dst: 104.199.64.136						
0100 = Version: 4 0101 = Header Length: 20 bytes (5) Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT) Total Length: 60 Identification: 0x03f7 (1015) Flags: 0x0000 Time to live: 128 Protocol: ICMP (1) Header checksum: 0x00ac [validation disabled] [Header checksum status: Unverified]						
0000	00 15 f9 7b 2f c9 d4 81 d7 af 8d 77 06 00 65 00U..				
0010	00 3c 03 f7 00 00 00 01 00 ac 9c 11 f0 bd 68 c7h.				
0020	40 5c 00 00 3d ae 00 01 0f ad 61 62 63 64 65 66abcde				
0030	67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76	ghijklmnopqrstuv				
0040	77 61 62 63 64 65 66 67 68 69	wxyzefgh				
268	11.970567	156.17.240.189	104.199.64.136	ICMP	74	Echo (ping) request id=0x0001, seq=4013/44303, ttl=128 (reply in 269)
269	11.999297	104.199.64.136	156.17.240.189	ICMP	74	Echo (ping) reply id=0x0001, seq=4013/44303, ttl=43 (request in 268)
270	12.609306	156.17.240.159	239.255.255.250	SSDP	216	M-SEARCH * HTTP/1.1
271	12.833173	156.17.240.184	230.0.0.1	UDP	92	64999 → 6666 Len=50
Frame 269: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0						
Ethernet II, Src: Cisco_7b:2f:c9 (00:15:f9:7b:2f:c9), Dst: Dell_af:8d:77 (d4:81:d7:af:8d:77)						
Internet Protocol Version 4, Src: 104.199.64.136, Dst: 156.17.240.189						
0100 = Version: 4 0101 = Header Length: 20 bytes (5)						

3.3. Obserwowanie logowania do poczty studenckiej

Przed logowaniem włączyłem przechwytywanie pakietów, a następnie użyłem filtrowania wpisując jako kryterium 'ssl'. Udało mi się w ten sposób uchwycić pakiet z wymianą kluczy. Na zrzucie ekranu widać też klucz publiczny.

No.	Time	Source	Destination	Protocol	Length	Info
2642	8.770858	156.17.240.189	156.17.193.220	TLSv1.	772	Application Data
2643	8.783109	156.17.193.220	156.17.240.189	TLSv1.	728	Application Data
2651	8.796060	13.107.42.12	156.17.240.189	TLSv1.	1348	Server Hello, Certificate, Certificate Status, Server Key Exchange, Server Hello Done
2653	8.798028	156.17.240.189	13.107.42.12	TLSv1.	147	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
2654	8.800579	13.107.42.12	156.17.240.189	TLSv1.	380	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
2656	8.809137	156.17.240.189	13.107.42.12	TLSv1.	134	Application Data
2657	8.809164	156.17.240.189	13.107.42.12	TLSv1.	305	Application Data
2661	8.975720	31.13.72.8	156.17.240.189	TLSv1.	671	Application Data
2665	9.376937	156.17.240.189	156.17.193.220	TLSv1.	648	Application Data
2671	9.381390	156.17.193.220	156.17.240.189	TLSv1.	1514	Application Data [TCP segment of a reassembled PDU]
2681	9.383354	156.17.193.220	156.17.240.189	TLSv1.	1514	Application Data [TCP segment of a reassembled PDU]
2687	9.384295	156.17.193.220	156.17.240.189	TLSv1.	1514	Application Data [TCP segment of a reassembled PDU]
TLSv1.2 Record Layer: Handshake Protocol: Client Key Exchange Content Type: Handshake (22) Version: TLS 1.2 (0x0303) Length: 57 Handshake Protocol: Client Key Exchange Handshake Type: Client Key Exchange (16) Length: 33 EC Diffie-Hellman Client Params Pubkey Length: 32 Pubkey: 78ec331ed6d1d0e90ad23f57ad5485ed32d9f71d70e8a134_						
TLSv1.2 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec Content Type: Change Cipher Spec (20) Version: TLS 1.2 (0x0303) Length: 1 Change Cipher Spec Protocol: Change Cipher Spec Change Cipher Spec: 0x00000000						
0040	78 ec 33 1e d6 d1 d0 e9 0a d2 3f 57 ad 54 85 ed78ec331e				
0050	32 d9 f7 1d 70 a6 a1 34 0b 5b 8c 79 25 a0 a1 2532d9f71d				
0060	14 03 03 00 01 01 16 03 03 00 20 00 00 00 00 0014030300				
0070	00 00 00 0c de d5 31 ce d1 79 0c c7 c7 cb 95 c40000000c				

Na kolejnym rzucie ekranu widać 2 certyfikaty Microsoftu.

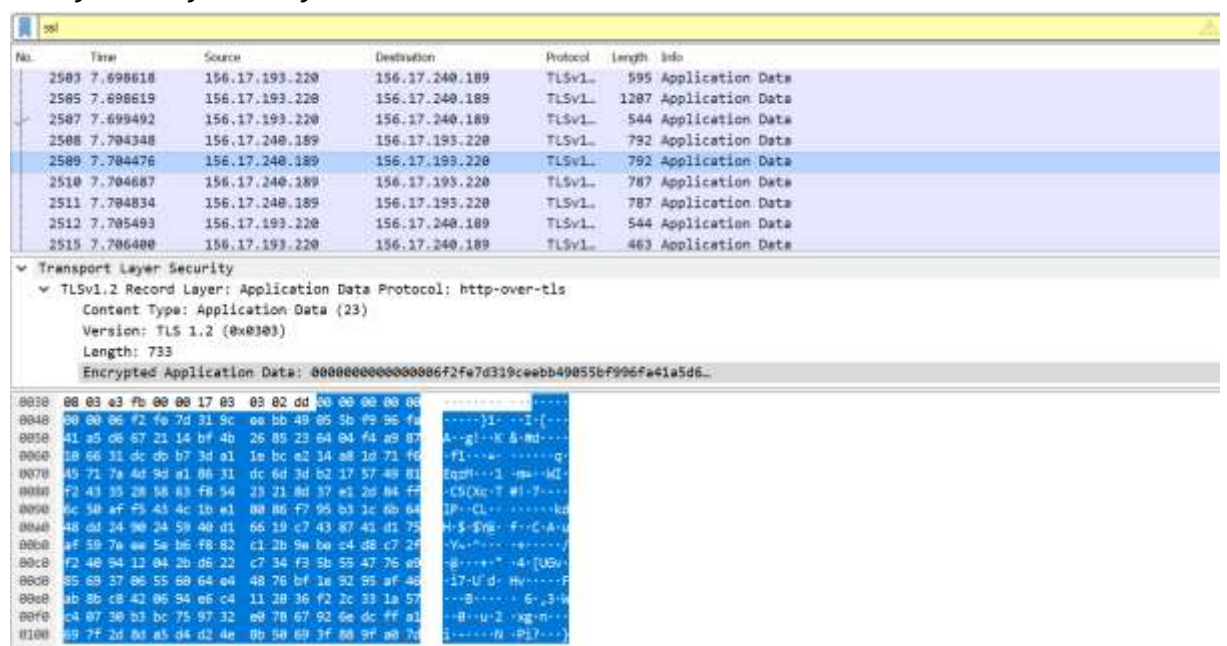


The image shows a Wireshark packet capture of a TLS handshake. The top pane displays a list of packets, with packets 25 and 26 selected. Packet 25 is a TLSv1.1 Server Hello, Certificate, Certificate Status, Server Key Exchange, and Server Hello Done. Packet 26 is a TLSv1.1 Client Key Exchange, Change Cipher Spec, and Encrypted Handshake Message. The bottom pane shows the details of the selected certificates. The first certificate is for 'storage.live.com' and the second is for 'Microsoft IT'. Both are signed by Microsoft Corporation.

No.	Time	Source	Destination	Protocol	Length	Info
16	0.525961	156.17.240.189	13.107.42.12	TLSv1..	134	Client Hello
18	0.546454	13.107.42.12	156.17.240.189	TCP	1514	443 → 58268 [ACK] Seq=1 Ack=181 Win=6400 Len=1460 [TCP segment of a reassembled PDU]
25	0.556256	13.107.42.12	156.17.240.189	TLSv1..	1348	Server Hello, Certificate, Certificate Status, Server Key Exchange, Server Hello Done
27	0.560419	156.17.240.189	13.107.42.12	TLSv1..	147	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
28	0.570706	13.107.42.12	156.17.240.189	TLSv1..	380	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
29	0.572282	156.17.240.189	13.107.42.12	TCP	1514	58268 → 443 [ACK] Seq=274 Ack=8031 Win=65280 Len=1460 [TCP segment of a reassembled PDU]
30	0.572284	156.17.240.189	13.107.42.12	TLSv1..	134	Application Data
31	0.572356	156.17.240.189	13.107.42.12	TLSv1..	305	Application Data
34	0.648870	13.107.42.12	156.17.240.189	TLSv1..	841	Application Data

Certificates (6346 bytes)
Certificate Length: 4876
Certificate: 30821308108210f0a00302010202132d000dce8c7b49d50... (id-at-commonName=storage.live.com,id-at-organizationalUnitName=Microsoft Corporation,id-at-o...
signedCertificate
algorithmIdentifier (sha256withRSAEncryption)
padding: 0
encrypted: 106d4a4f882afcd0c13d653e125ec05445a7f66c03c202f...
Certificate Length: 1464
Certificate: 308205043082049ca00302010202100888cd525f1924444d... (id-at-commonName=Microsoft IT TLS CA 5,id-at-organizationalUnitName=Microsoft IT,id-at-organ...
signedCertificate

Na kolejnym rzucie ekranu znajduje się przykładowy pakiet z zaszyfrowanymi danymi.



The image shows a Wireshark packet capture of a TLS session. The top pane displays a list of packets, with packets 2503 through 2515 selected. These packets are all 'Application Data' of various lengths. The bottom pane shows the details of the selected packets, which are all 'Application Data' of various lengths. The 'Encrypted Application Data' field is highlighted, showing a long string of hexadecimal characters.

No.	Time	Source	Destination	Protocol	Length	Info
2503	7.698618	156.17.193.220	156.17.240.189	TLSv1..	595	Application Data
2505	7.698619	156.17.193.220	156.17.240.189	TLSv1..	1207	Application Data
2507	7.699492	156.17.193.220	156.17.240.189	TLSv1..	544	Application Data
2508	7.704348	156.17.240.189	156.17.193.220	TLSv1..	792	Application Data
2509	7.704476	156.17.240.189	156.17.193.220	TLSv1..	792	Application Data
2510	7.704687	156.17.240.189	156.17.193.220	TLSv1..	787	Application Data
2511	7.704834	156.17.240.189	156.17.193.220	TLSv1..	787	Application Data
2512	7.705493	156.17.193.220	156.17.240.189	TLSv1..	544	Application Data
2515	7.706400	156.17.193.220	156.17.240.189	TLSv1..	463	Application Data

Transport Layer Security
TLSv1.2 Record Layer: Application Data Protocol: http-over-tls
Content Type: Application Data (23)
Version: TLS 1.2 (0x0303)
Length: 733
Encrypted Application Data: 0000000000000000f2fe7d339cwebb40855bf906fa41a5d6...

3.4. Wyszukiwanie na niezasyfrowanej stronie

Sprawdziłem czy tekst wpisany w wyszukiwarkę na jakiejś stronie nie zabezpieczonej przez https będzie się dało zobaczyć w pakietach. Wybrałem stronę skt.pwr.edu.pl.



Pakiet z danymi wyszukiwania udało się znaleźć. Widać bezpośrednio treść wpisaną na stronie.

Info pakietów
Wykres i szczeg.
☐ Rozdzielaj wielkość znaków
String
raw
Zapiś

Time	Source	Destination	Protocol	Length	Info
6551	8.838069	31.15.72.12	156.17.240.189	TCP	66 443 → 58331 [ACK] Seq=1 Ack=2 Win=67 Len=0 SL=1 SR=2
6552	8.950277	156.17.240.189	216.58.215.110	QUIC	1392 Client Hello, PKN: 1, CID: 6809144113227953005
6553	8.952231	156.17.240.189	156.17.1.107	TCP	54 58336 → 88 [FIN, ACK] Seq=892 Ack=9323 Win=65288 Len=0
6554	8.952288	156.17.240.189	216.58.215.110	TCP	54 58332 → 443 [RST, ACK] Seq=1 Ack=2 Win=256 Len=0
6555	8.952590	156.17.240.189	156.17.1.107	TCP	66 58337 → 88 [SYN] Seq=8 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
6556	8.953487	156.17.1.107	156.17.240.189	TCP	66 88 → 58337 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1 WS=128
6557	8.953572	156.17.240.189	156.17.1.107	TCP	54 58337 → 88 [ACK] Seq=1 Ack=1 Win=52568 Len=0
6558	8.953935	156.17.240.189	156.17.1.107	HTTP	499 GET /?s=rzczz HTTP/1.1

GET /?s=rzczz HTTP/1.1\r\n

- [Expert Info (Chat/Sequence): GET /?s=rzczz HTTP/1.1\r\n]
 - [GET /?s=rzczz HTTP/1.1\r\n]
 - [Severity level: Chat]
 - [Group: Sequence]
 - Request Method: GET
- Request URI: /?s=rzczz
 - Request URI Path: /
 - Request URI Query: s=rzczz
 - Request URI Query Parameter: s=rzczz
 - Request Version: HTTP/1.1

```

08 05 3d 0c 00 00 47 45 34 20 2f 3f 73 3d 72 7a  u: GET /?s=r
55 63 72 28 48 54 54 50 2f 31 2e 31 0d 0a 48 6f  acc HTTP /1.1--Mo
73 74 3a 20 73 60 74 2a 70 77 72 2a 65 64 75 2a  st: sct. par.edu.
70 6c 0d 0a 43 6f 6e 6a 65 63 74 09 6f 6a 3a 20  pl: Conn action:
6b 65 65 70 2d 61 6c 69 76 65 0d 0a 55 70 67 72  keep-alive: Upgr
61 64 65 2d 49 6a 73 65 63 75 72 65 2d 52 65 71  ode-Inse cure-Req
75 65 73 74 73 3a 20 31 0d 0a 55 73 65 72 2d 41  uests: 1 --User-A
67 65 6a 74 3a 20 4d 6f 7a 09 6c 6c 31 2f 35 2e  gent: Mozilla/5.

```



No.	Time	Source	Destination	Protocol	Length	Info
47	5.374536	Cisco_7b:2f:c8	Broadcast	ARP	60	Who has 156.17.248.238? Tell 156.17.248.254
48	5.613321	Cisco_85:80:88	Cisco_85:80:88	LOOP	60	Reply
49	5.617313	156.17.248.189	52.59.182.144	TCP	54	58533 → 80 [FIN, ACK] Seq=1 Ack=1 Win=2896 Len=0
50	5.617982	156.17.248.189	52.59.182.144	TCP	66	58547 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WGS=256 SACK_PERM=1
51	5.619583	52.59.182.144	156.17.248.189	TCP	60	80 → 58547 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1 W=128
52	5.619676	156.17.248.189	52.59.182.144	TCP	54	58547 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
53	5.619888	156.17.248.189	52.59.182.144	HTTP	1124	POST /schedule HTTP/1.1 (application/x-www-form-urlencoded)
54	5.621364	52.59.182.144	156.17.248.189	TCP	60	80 → 58547 [ACK] Seq=1 Ack=1861 Win=7296 Len=0
55	5.737323	Cisco_7b:2f:c8	Broadcast	ARP	60	Who has 156.17.248.175? Tell 156.17.248.254

```

HTML Form URL Encoded: application/x-www-form-urlencoded
  Form item: "utf8" = "✓"
    Key: utf8
    Value: \342\154\213
  Form item: "login" = "kam11"
    Key: login
    Value: kam11
  Form item: "password" = "superhaslo"
    Key: password
    Value: superhaslo
  
```

```

0000 62 39 58 63 36 60 31 37 30 39 64 39 66 32 61 61 008cf17 90d9f2be
0100 33 65 38 35 38 64 31 31 66 63 39 63 32 37 35 39 36060112 fc9c2759
0200 38 9c 0a 0a 0a 75 74 66 38 3d 25 45 32 25 39 43 8- utf 8uN250C
0300 25 39 35 26 6c 6f 67 69 6e 3d 66 61 66 69 6c 26 7058logi rnkam11
0400 70 61 73 73 77 6f 72 64 3d 71 75 38 65 72 64 01 password =super
0500 79 6c 6f 26 73 63 68 65 84 75 6c 65 25 35 42 72 8uache n1u1k58r
0600 61 77 25 35 44 3d 26 63 6f 6d 69 74 3d 58 6f 4uRSD=Sc om1tFo
  
```