

Sprawozdanie z pierwszej listy zadań na laboratorium z Technologii Sieciowych

Paweł Rubin

Marzec 2019

Spis treści

1	Wstęp	3
1.1	Opis zadania	3
1.2	Ping	3
1.2.1	Przykładowe użycie	3
1.2.2	Najważniejsze opcje	3
1.3	Traceroute	3
1.3.1	Przykładowe użycie	4
1.4	Wireshark	4
1.4.1	Przykładowe użycie	4
2	Testy	5
2.1	Odległości	5
2.2	Wielkość pakietu a ilość węzłów	5
2.3	Wielkość pakietu a czas propagacji	5
2.4	Fragmentacja	6
3	Podsumowanie	6

1 Wstęp

1.1 Opis zadania

Przetestuj działanie programów:

a) *Ping*: Sprawdź za jego pomocą ile jest węzłów na trasie do (i od) wybranego, odległego geograficznie, serwera. Uwaga: trasy tam i z powrotem mogą być różne. Zbadaj jaki wpływ ma na to wielkość pakietu. Zbadaj jak wielkość pakietu wpływa na obserwowane czasy propagacji. Zbadaj jaki wpływ na powyższe ma konieczność fragmentacji pakietów. Jaki największy niefragmentowany pakiet uda się przesłać. Przeanalizuj te same zagadnienia dla krótkich tras (do serwerów bliskich geograficznie). Określ "średnicę" internetu (najdłuższą ścieżkę którą uda się wyszukać). Czy potrafisz wyszukać trasy przebiegające przez sieci wirtualne (zdalne platformy "cloud computing"). Ile węzłów mają ścieżki w tym przypadku.

b) *Traceroute*,

c) *WireShark*.

1.2 Ping

Program ping to narzędzie służące do diagnozowania połączeń sieciowych. Pozwala na sprawdzenie istnienia połączenia pomiędzy hostami, umożliwia zmierzenie liczby zgubionych pakietów oraz opóźnień w ich transmisji.

1.2.1 Przykładowe użycie

```
$ ping google.com
```

```
Pinging google.com [2a00:1450:401b:807::200e] with 32 bytes of data:
```

```
Reply from 2a00:1450:401b:807::200e: time=66ms
```

```
Reply from 2a00:1450:401b:807::200e: time=70ms
```

```
Reply from 2a00:1450:401b:807::200e: time=67ms
```

```
Reply from 2a00:1450:401b:807::200e: time=66ms
```

```
Ping statistics for 2a00:1450:401b:807::200e:
```

```
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

```
Approximate round trip times in milli-seconds:
```

```
    Minimum = 66ms, Maximum = 70ms, Average = 67ms
```

1.2.2 Najważniejsze opcje

Poniżej przedstawiam opis najważniejszych opcji programu ping w systemie Windows.

- **-l wartość** - rozmiar wysyłanego pakietu.
- **-i wartość** - wartość pola TTL (Time To Live).
- **-f** - ustawienie flagi DF (Don't Fragment).
- **-4** - wymuszenie używania IPv4

1.3 Traceroute

Traceroute - program służący do badania trasy pakietów w sieci IP. Program traceroute jest szeroko dostępny we wszystkich uniksowych systemach operacyjnych.

1.3.1 Przykładowe użycie

```
tracert google.com
```

Tracing route to google.com [2a00:1450:401b:807::200e]
over a maximum of 30 hops:

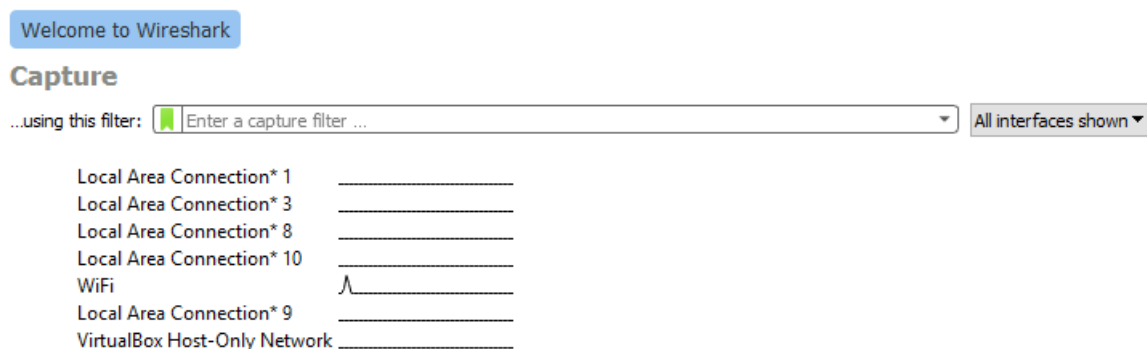
```
  1      5 ms      3 ms      3 ms  funbox.home [2a01:112f:563:e200:baee  
      :eff:fe85:778d]  
  2     32 ms     34 ms     33 ms  2a01:1000:0:56c::2  
  3     91 ms     93 ms     32 ms  2a01:1000:0:56c::1  
  4     33 ms     32 ms     32 ms  2a01:1000:0:26::2  
  5     34 ms     34 ms     34 ms  2a01:1000:0:1e::1  
  6     64 ms     59 ms     59 ms  bu16-10.ffttr4.Frankfurt.opentransit  
      .net [2001:688:0:3:8::e1]  
  7     67 ms     68 ms     68 ms  2001:67c:128c::5e  
  8     68 ms     68 ms     67 ms  2a00:1450:8097::1  
  9     68 ms     68 ms     68 ms  2001:4860:0:1::2046  
 10     67 ms     67 ms     68 ms  2001:4860:0:11e0::11  
 11     66 ms     65 ms     67 ms  2001:4860::8:0:cb95  
 12     67 ms     66 ms     68 ms  2001:4860::9:4000:d78c  
 13     67 ms     67 ms     68 ms  2001:4860:0:1::163d  
 14     66 ms     66 ms     67 ms  waw02s17-in-x0e.1e100.net [2a00  
      :1450:401b:807::200e]
```

Trace complete.

1.4 Wireshark

Wireshark to darmowe narzędzie opensource do analizy pakietów. Używany do rozwiązywania problemów i analizy sieci komputerowych oraz w procesie wytwarzania oprogramowania i projektowaniu protokołów komunikacji.

1.4.1 Przykładowe użycie



Rysunek 1: Wybór interfejsu

No.	Time	Source	Destination	Protocol	Length	Info
55	10.213626	192.168.1.46	162.254.196.68	TLSv1.2	117	Application Data
56	10.404230	162.254.196.68	192.168.1.46	TCP	54	27020 → 63130 [ACK]
57	10.624155	fe80::baee:eff:fe85...	fe80::8405:e54a:cdc...	ICMPv6	86	Neighbor Solicitat
58	10.624298	fe80::8405:e54a:cdc...	fe80::baee:eff:fe85...	ICMPv6	86	Neighbor Advertise
59	10.691725	2a03:2880:f02d:e:fa...	2a01:112f:563:e200:...	TLSv1.2	222	Application Data
60	10.708766	2a01:112f:563:e200:...	2a03:2880:f02d:e:fa...	TLSv1.2	349	Ignored Unknown Re
61	10.768322	2a03:2880:f02d:e:fa...	2a01:112f:563:e200:...	TCP	74	443 → 63002 [ACK]
62	10.768322	2a03:2880:f02d:e:fa...	2a01:112f:563:e200:...	TLSv1.2	109	Application Data
63	10.808851	2a01:112f:563:e200:...	2a03:2880:f02d:e:fa...	TCP	74	63002 → 443 [ACK]
64	10.922484	162.254.196.68	192.168.1.46	TLSv1.2	156	Application Data
65	10.962056	192.168.1.46	162.254.196.68	TCP	54	63130 → 27020 [ACK]

Rysunek 2: Przechwytywanie ruchu sieciowego

2 Testy

2.1 Odległości

Testy polegały na "pingowaniu" serwerów bliskich oraz odległych geograficznie. Obliczanie **skoków do** poprzez ustalanie początkowej wartości **ttl** aż dostaniemy odpowiedź, **skoki z** to różnica pomiędzy początkową wartością **ttl** (64, 128 lub 255) a wartością otrzymaną w odpowiedzi. Wyniki w postaci tabeli 1 poniżej.

Adres	Położenie geograficzne	skoki do	skoki z	Średni Czas [ms]
wcss.pl	Wrocław	8	7	48
edukacja.pwr.wroc.pl	Wrocław	11	9	49
southmiamifl.gov	Floryda, USA	18	14	177
kyoto-u.ac.jp	Japonia	19	17	314
ox.ac.uk	Anglia	21	16	88
sydney.edu.au	Australia	20	17	354
tourism.net.nz	Nowa Zelandia	25	15	354

Tabela 1: Odległości

2.2 Wielkość pakietu a ilość węzłów

Kolejny eksperyment celował w ustalenie - czy rozmiar pakietu wpływa na ilość węzłów od/do serwerów? Próby przeprowadzane na trzech różnych serwerach. Wielkość pakietu ustalana za pomocą opcji **-l wartość**. Wyniki w postaci tabeli 2 poniżej.

Adres	128	256	512	1024
google.com	55	55	55	55
wcss.pl	248	248	248	248
tourism.net.nz	49	49	49	49

Tabela 2: Wartości **ttl** odpowiedzi dla danych wielkości pakietów.

2.3 Wielkość pakietu a czas propagacji

W tym eksperymencie za cel stawiamy zbadanie czy rozmiar wysyłanych pakietów wpływa na czas propagacji. Próby przeprowadzane na trzech różnych serwerach. Wielkość pakietu ustalana za pomocą opcji **-l wartość**. Wyniki w postaci tabeli 4 poniżej.

Adres	128	256	512	1024	Możliwe maksimum
google.com	71	70	71	72	1472: 67
wcss.pl	49	49	49	51	1472: 51
tourism.net.nz	354	355	355	356	7392: 370

Tabela 3: Średni czas odpowiedzi dla danych wielkości pakietów.

2.4 Fragmentacja

Spróbujmy znaleźć maksymalny rozmiar niefragmentowanego pakietu. Flage DF (Don't Fragment) ustawimy za pomocą opcji **-f**, a rozmiar za pomocą opcji **-l rozmiar**.

Okazuje się, że największy możliwy do wysłania pakiet przy ustawionej fladze DF będzie miał rozmiar 1472 bajtów.

Wpływ fragmentacji na ilość węzłów i czas propagacji możemy zobaczyć w poniższej tabeli. Użyto maksymalnego rozmiaru pakietu dla flagi DF.

Adres	Średni czas	Średni czas DF	TTL	TTL DF
google.com	72	73	55	55
wcss.pl	51	51	248	248
google.com	357	358	49	49

Tabela 4: Średnie czasy odpowiedzi oraz wartości ttl dla fragmentowanych oraz niefragmentowanych pakietów.

3 Podsumowanie

Wyniki doświadczeń zgadzają się z intuicją - serwery dalsze geograficznie są również dalsze pod względem ilości skoków pomiędzy węzłami. Jak pokazały powyższe testy, wielkość pakietu nie wpływa znacznie ani na trasę pakietu ani na czas propagacji. Na czas odpowiedzi i trasę pakietu nie wpłynęło również fragmentowanie pakietów. Z analizy odległości można wywnioskować, że średnica internetu mieści się w przedziale [25,30].

Przydatność tych programów zależy od umiejętności użytkownika oraz od problemu, który chcemy rozwiązać. Bez wątpienia są to świetne narzędzia do diagnostyki i analizy ruchu sieciowego.