

Sprawozdanie

Lista1, technologie sieciowe, Gabriel Wechta

(„Jedyny sposób, żeby ludzie byli razem, to zesłać im dżumę.”)

- Opis działania programów:

1. Ping:

Komenda ping przyjmuje jako argumenty opcje wywołania oraz miejsce docelowe (DNS lub IP). Ping wysyła jeden lub więcej pakietów ICMP (Internet Control Message Protocol) i czeka na odpowiedź. Ping ponadto wyświetla informacje o czasie transportu pakietu oraz informuje o zgubionych pakietach. Ping pozwala zmieniać ilość bajtów w wiadomości, ustawiać ttl (Time to Live) itd.

```
gabriel@gabriel-HP-Laptop-15-bs1xx:~/Desktop/studia/rok 2/technologie_sieciowe/List_1$ ping -c 10 -s 42 cs.pwr.edu.pl
PING cs.pwr.edu.pl (156.17.7.22) 42(70) bytes of data.
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=1 ttl=54 time=64.2 ms
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=2 ttl=54 time=32.7 ms
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=3 ttl=54 time=100 ms
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=4 ttl=54 time=28.7 ms
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=5 ttl=54 time=64.5 ms
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=6 ttl=54 time=24.0 ms
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=7 ttl=54 time=32.6 ms
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=8 ttl=54 time=31.3 ms
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=9 ttl=54 time=48.5 ms
50 bytes from informatyka.im.pwr.wroc.pl (156.17.7.22): icmp_seq=10 ttl=54 time=26.7 ms
--- cs.pwr.edu.pl ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9017ms
rtt min/avg/max/mdev = 24.083/45.390/100.296/23.123 ms
```

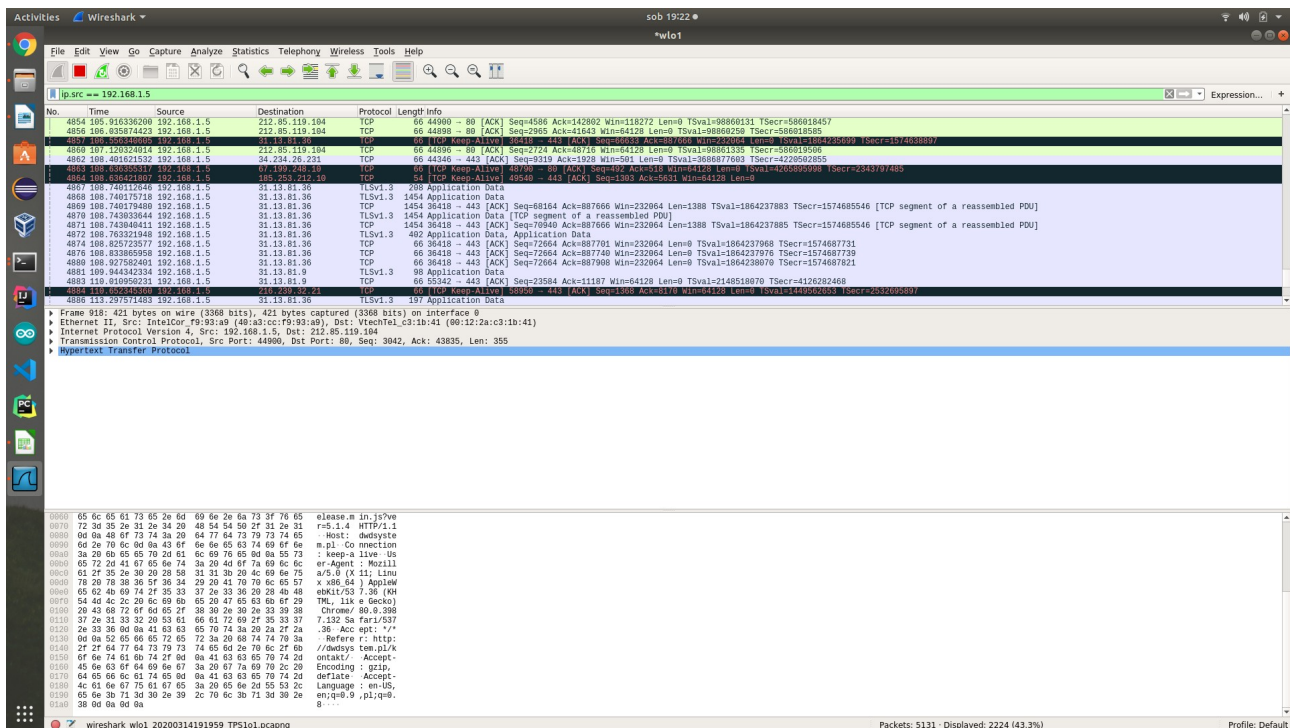
2. Traceroute:

Komenda traceroute mapuje drogę jaką pakiet przechodzi aby dotrzeć do celu. Robi to poprzez wysyłanie datagramów UDP z niskim ttl, inkrementacyjnie zwiększając ttl. Następnie nasłuchuje na odpowiedź – TIME_EXCEEDED od poszczególnych ruterów.

```
gabriel@gabriel-HP-Laptop-15-bs1xx:~/Desktop/studia/rok 2/technologie_sieciowe/List_1$ traceroute cs.pwr.edu.pl
traceroute to cs.pwr.edu.pl (156.17.7.22), 30 hops max, 60 byte packets
 1  netiaspot.home (192.168.1.254)  5.923 ms  7.533 ms  8.775 ms
 2  83.238.252.80 (83.238.252.80)  38.495 ms  38.491 ms  40.450 ms
 3  wrocc002rt09.inetia.pl (83.238.113.18)  40.439 ms  40.414 ms  40.385 ms
 4  WroCC002RT05-WASK.inetia.pl (83.238.249.150)  40.319 ms  40.294 ms  40.266 ms
 5  156.17.250.215 (156.17.250.215)  40.229 ms  40.193 ms  40.156 ms
 6  centrum-rtr-karkonosz.wask.wroc.pl (156.17.254.110)  41.299 ms  34.756 ms  34.177 ms
 7  rolnik2-centrum.wask.wroc.pl (156.17.254.65)  32.899 ms  90.949 ms  93.117 ms
 8  wazniak-rolnik.wask.wroc.pl (156.17.254.140)  54.590 ms  54.517 ms  54.480 ms
 9  z-wask2-do-pwr2.pwrnet.pwr.wroc.pl (156.17.18.244)  53.274 ms  54.281 ms  54.301 ms
10  156.17.33.1 (156.17.33.1)  54.285 ms  54.278 ms  54.238 ms
11  informatyka.im.pwr.wroc.pl (156.17.7.22)  54.209 ms  54.932 ms  54.930 ms
```

3. Wireshark:

Jest to sniffer, z interfacem służącym do monitorowania ruchu pakietów. Ma wbudowaną obsługę filtrów do monitorowania pakietów o konkretnych cechach, z konkretnego źródła itd. Obsługuje wiele protokołów, pozwala na wyświetlenie wartości konkretnych pól pakietu.



- Liczenie węzłów:
Do wykonania zadania użyłem, skryptu z komendą ping.

#!/bin/bash

timeOut=\$2

while :

do

if ping -W \$3 -Mdo -c1 -t \$timeOut \$1 >respond

then

break

else

((timeOut++))

fi

done

echo "it takes: \$timeOut"

Skrypt w zasadzie stosuje tę samą technikę co traceroute.

					10 packets in bytes, avg [ms]		
	DNS	Localization	number of nodes „to”	ttl value from ping response	108	1008	2008
1	charlie.pl	Łódź, Poland	10	55	52.906	56.974	57.803
2	royal.uk	London, UK	7	58	48.521	63.177	56.069
3	fbcscycamore.com	Sycamore, USA	16	236			
4	ameblo.jp	?, Japan	9	54			
5	heide.com.au	Melbourne, Australia	21	40	497.012	479.731	502.514

Trasy tam i z powrotem rzeczywiście mogą być różne natomiast niesposób ocenić. Jedyny feedback jaki mamy to ttl pakietu wysłanego przez cel.

Przy wysyłaniu, na 5 adresów wymienionych wyżej, pakietów 64 bytowych i 512 bytowych otrzymałem taki sam rezultat w ilości węzłów użytych do komunikacji.

Następnie wybrałam trzy adresy i wysłałam do nich po 100 pakietów, o rozmiarach 108, 1008 i 2008 bajtów. Na załączonej tabelce widać, że wyniki wahają i odchodzą od spodziewanego wyniku to jest od rosnącego czasu propagacji, ale na szybkość propagacji wiele rzeczy ma wpływ chociażby to, że do mojego domowego routera jest podłączonych wiele urządzeń. Natomiast widać tendencje wzrostową, wraz ze zwiększeniem pakietu zwiększa się czas propagacji.

W testach jakie przeprowadziłem fragmentacja pakietu nie miała znaczącego wpływu na czas propagacji, ponownie wiele czynników mogło mieć na to wpływ.

Największy niefragmentowany pakiet jaki udało mi się wysłać miał rozmiar 1488 bajtów.

```
gabriel@gabriel-HP-Laptop-15-bs1xx:~$ ping -s $((1500 - 28)) -D 8.8.8.8 -c 1
PING 8.8.8.8 (8.8.8.8) 1472(1500) bytes of data.
[1584212819.399783] From 192.168.1.254 icmp_seq=1 Frag needed and DF set (mtu = 1488)

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

gabriel@gabriel-HP-Laptop-15-bs1xx:~$ ping -s $((1488 - 28)) -D 8.8.8.8 -c 1
PING 8.8.8.8 (8.8.8.8) 1460(1488) bytes of data.
[1584212859.175914] 76 bytes from 8.8.8.8: icmp_seq=1 ttl=55 (truncated)

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 45.907/45.907/45.907/0.000 ms
gabriel@gabriel-HP-Laptop-15-bs1xx:~$ ping -s $((1489 - 28)) -D 8.8.8.8 -c 1
PING 8.8.8.8 (8.8.8.8) 1461(1489) bytes of data.

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

Bazując na wyniku połączenia z Australią – 21 węzłów, z całą pewnością średnica internetu jest większa niż 19 jak sugerował R.Albert z University of Notre Dame w 1999. Natomiast udało mi się również znaleźć informacje o połączeniach o 30 i więcej węzłach.

Mam problem z odpowiedzeniem na ostatnie pytanie. Cloud computing jako taki opiera się na wewnętrznych protokołach zapewniających bezstratną transmisję danych pomiędzy urządzeniami/komputerami. Ciężko powiedzieć cokolwiek na temat tych protokołów. Przez to ciężko ocenić ile węzłów będą miały połączenia w sieci wirtualnej. Natomiast bazując na oczekiwaniu, że wiele urządzeń będzie obliczało nasze zapytanie (również ICMP) można się spodziewać, że węzłów będzie dużo.

Udało mi się jeszcze znaleźć informacje o sieciach wirtualnych chociażby w nieszczęsnych Chinach, ale nie udało mi się dostać response'a od żądanej adresu który testowałem.