Instituto Superior Técnico

Redes e Serviços Internet

Laboratório 1 - Acesso à Internet

Autores:
Diogo Moura - nº 86976
Diogo Alves - nº 86980
Tomás Malcata - nº 87130

Professor:
Paulo Pereira

19 de Outubro de 2019

3-b)

```
guest-zaly03@adm68:~$ ifconfig
enp1s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.10.10.2  netmask 255.255.255.248  broadcast 10.10.10.7
        inet6 fe80::3142:fb73:7d90:15d4  prefixlen 64  scopeid 0x20<link>
        ether d8:cb:8a:ca:a9:1c  txqueuelen 1000  (Ethernet)
        RX packets 13115  bytes 9809217 (9.8 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3080  bytes 309264 (309.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 693  bytes 53232 (53.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 693  bytes 53232 (53.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 192.168.122.1  netmask 255.255.255.0  broadcast 192.168.122.255
        ether 52:54:00:01:22:53  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

guest-zaly03@adm68:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.10.10.1      0.0.0.0         UG    20100  0        0 enp1s0
10.10.10.0      0.0.0.0         255.255.255.248 U     100    0        0 enp1s0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 virbr0
192.168.122.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
guest-zaly03@adm68:~$
```

Figura 1: Resultado dos comandos *ifconfig* e *route -n*

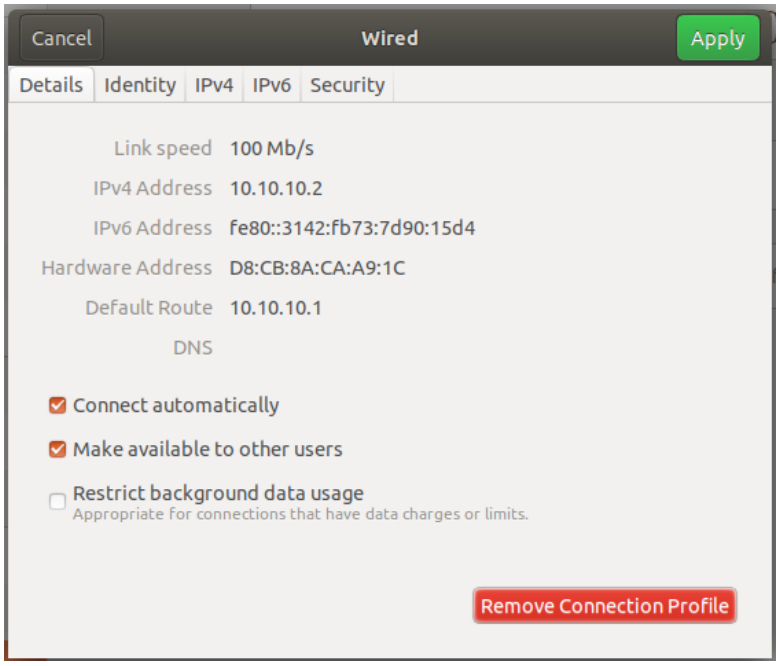

Figura 2: Configuração da rede

Tanto na configuração de rede do PC como através do comando *ifconfig*, obtemos o endereço IP do PC (10.10.10.2) e o endereço MAC da placa de rede (D8:CB:8A:CA:A9:1C). O endereço do router é obtido recorrendo ao comando *route -n* ou verificando nas definições de rede (10.10.10.1).

3-d)

```
guest-zaly03@adm68:~$ ssh admin@10.10.10.1
The authenticity of host '10.10.10.1 (10.10.10.1)' can't be established.
RSA key fingerprint is SHA256:0HhbZR7olcaFIw8vGHktkO9Isy160tAfDdnNLaNGXCE.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.10.10.1' (RSA) to the list of known hosts.
 IST/UTL Area Cientifica de Computadores Password:

yourname#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
yourname(config)#interface fastEthernet 4
yourname(config-if)#ip address 193.136.143.68 255.255.255.0
yourname(config-if)#no shutdown
yourname(config-if)#end
yourname#
```

Figura 3: Configuração do router, com indicação do endereço IP utilizado

O endereço IP utilizado foi o 193.136.143.68.

4-a)

```
yourname#show version
Cisco IOS Software, C870 Software (C870-ADVSECURITYK9-M), Version 12.4(24)T8, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Sun 09-Sep-12 09:09 by prod_rel_team

ROM: System Bootstrap, Version 12.3(8r)YI4, RELEASE SOFTWARE

yourname uptime is 21 minutes
System returned to ROM by power-on
System image file is "flash:c870-advsecurityk9-mz.124-24.T8.bin"


This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

Cisco 871W (MPC8272) processor (revision 0x300) with 118784K/12288K bytes of memory.
Processor board ID FCZ142891X7
MPC8272 CPU Rev: Part Number 0xC, Mask Number 0x10
5 FastEthernet interfaces
1 802.11 Radio
128K bytes of non-volatile configuration memory.
28672K bytes of processor board System flash (Intel Strataflash)

Configuration register is 0x2102

yourname#show flash
28672K bytes of processor board System flash (Intel Strataflash)

Directory of flash:/

    2  -rwx    18839652   Mar 1 2002 05:54:13 +00:00  c870-advsecurityk9-mz.124-24.T8.bin
    3  -rwx        3331  Jun 30 2014 15:16:31 +00:00  cpconfig-8xx.cfg
    4  -rwx     1697952  Jun 30 2014 15:15:56 +00:00  securedesktop-ios-3.1.1.45-k9.pkg
    5  -rwx      415956  Jun 30 2014 15:16:22 +00:00  sslclient-win-1.1.4.176.pkg
    6  -rwx        1038  Jun 30 2014 15:16:31 +00:00  home.shtml
    7  -rwx      115712  Jun 30 2014 15:16:36 +00:00  home.tar
    8  -rwx      527849   Mar 1 2002 00:06:03 +00:00  128MB.sdf
    9  -rwx     2941440  Jun 30 2014 15:19:22 +00:00  cpexpress.tar
   10  -rwx     2246144  Jun 30 2014 15:22:43 +00:00  wlanui.tar

27611136 bytes total (806912 bytes free)
```

Figura 4: Detalhes do router (versão e memória)

O router corre a versão 12.4(24)T8 do Cisco IOS, tem 118784K/11288K bytes de memória RAM, 128Kbytes de memória não-volátil para armazenar a configuração do router e 28672Kbytes de memória Flash, dos quais 806912 bytes estão livres.

4-c)

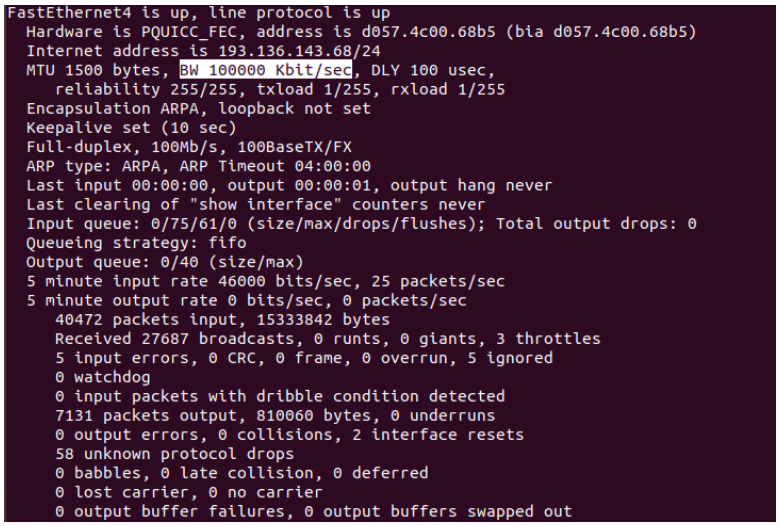

Figura 5: Indicação do ritmo máximo do router

A bandwidth/ritmo máximo da interface FastEthernet4 do router é 100 000 Kbit/s

5-a)

Em sistemas Unix, o traceroute envia, por defeito, uma sequência de pacotes User Datagram Protocol (UDP), com porto destino entre 33434 a 33534.
Em Windows, o traceroute envia pacotes ICMP Echo Request, em vez de pacotes UDP por defeito.
O campo "TTL"(Time to live) do cabeçalho IP determina o número de segundos durante o qual o pacote é válido, tempo após o qual o mesmo deve ser descartado. No Traceroute é utilizado como "hop count"(contador de saltos), em que cada router que entrega o pacote reduz o "TTL"em 1 unidade. Pacotes de IPv6, em vez do campo "TTL", têm o campo "hop limit"no cabeçalho, que serve o mesmo propósito.
O traceroute envia pacotes com valores de TTL que gradualmente aumentam de pacote para pacote, começando com o valor de TTL de 1. Os routers decrementam o valor do TTL em um e descartam os pacotes cujo valor de TTL atinja 0, returnando a mensagem ICMP de erro "ICMP Time Exceeded".
Para os primeiros pacotes, o primeiro router recebe o pacote e decrementa o valor do TTL e descarta-o , dado que o valor do TTL atinje o valor de 0. O router envia uma mensagem "ICMP Time Exceeded"de volta para a fonte. Aos pacotes seguintes é atribuído um valor de TTL de 2, pelo que o primeiro router entrega os pacotes, mas o segundo router descarta-os e responde com um mensagem "ICMP Time Exceeded".
Procedendo deste modo, o "traceroute"utiliza as mensagens "ICMP Time Exceeded"para construir uma lista de routers que os pacotes atravessam, até que o endereço IP destino é alcançado e retorna uma mensagem "ICMP Destination Unrecheable"se estiverem a ser utilizados pacotes UDP (caso do Linux, por defeito) ou "ICMP Echo Reply"se estiverem a ser utilizadas mensagens "ICMP Echo".

```
guest-zaly03@adm68:~$ wireshark&
[1] 3985
guest-zaly03@adm68:~$ ping www.cmu.edu
PING WWW.R53.cmu.edu (128.2.42.52) 56(84) bytes of data.
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=1 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=2 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=3 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=4 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=5 ttl=238 time=132 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=6 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=7 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=8 ttl=238 time=129 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=9 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=10 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=11 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=12 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=13 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=14 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=15 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=16 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=17 ttl=238 time=128 ms
64 bytes from WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52): icmp_seq=18 ttl=238 time=128 ms
^C
--- WWW.R53.cmu.edu ping statistics ---
18 packets transmitted, 18 received, 0% packet loss, time 17021ms
rtt min/avg/max/mdev = 128.087/128.542/132.151/0.923 ms
guest-zaly03@adm68:~$ wireshark&
[2] 4585
[1]   Done                    wireshark
guest-zaly03@adm68:~$ traceroute www.cmu.edu
traceroute to www.cmu.edu (128.2.42.52), 30 hops max, 60 byte packets
 1  _gateway (10.10.10.1)  16.907 ms  17.400 ms  18.178 ms
 2  b0-130.gwtnorte.ist.utl.pt (193.136.143.253)  6.247 ms  6.946 ms  7.651 ms
 3  e0.gatekeeper2.tecnico.ulisboa.pt (193.136.134.158)  8.586 ms  9.508 ms  10.190 ms
 4  194.117.12.129 (194.117.12.129)  11.295 ms  12.156 ms  12.884 ms
 5  194.117.12.249 (194.117.12.249)  13.784 ms  14.649 ms  15.352 ms
 6  Router13.Lisboa.fccn.pt (193.136.1.93)  18.078 ms  12.468 ms  12.129 ms
 7  ROUTER1.10GE.CR2.Lisboa.fccn.pt (193.137.0.17)  11.997 ms ROUTER1.10GE.CR1.Lisboa.fccn.pt (193.137.0.1)  10.705 ms ROUTER1.10GE.CR2.Lisboa.fccn.pt (193.137.0.17)  10.436 ms
 8  fccn.mx2.lis.pt.geant.net (62.40.124.97)  10.432 ms  8.897 ms  8.684 ms
 9  ae0.mx1.mad.es.geant.net (62.40.98.107)  19.112 ms  18.728 ms  18.693 ms
10  ae3.mx1.par.fr.geant.net (62.40.98.65)  36.619 ms  35.205 ms  35.358 ms
11  et-2-1-5.102.rtsw.newy32aoa.net.internet2.edu (198.71.45.236)  119.768 ms  122.072 ms  122.383 ms
12  et-4-0-0.4079.rtsw.phil.net.internet2.edu (162.252.70.103)  122.688 ms  123.330 ms  124.042 ms
13  204.238.76.33 (204.238.76.33)  122.915 ms  125.011 ms  124.621 ms
14  204.238.76.46 (204.238.76.46)  124.234 ms  118.866 ms  119.616 ms
15  162.223.17.79 (162.223.17.79)  144.433 ms  145.090 ms  144.704 ms
16  CORE0-POD-I-DCNS.GW.CMU.NET (128.2.0.193)  141.015 ms  129.405 ms  129.688 ms
17  POD-D-CYH-CORE0.GW.CMU.NET (128.2.0.202)  133.624 ms  133.934 ms  132.604 ms
18  WWW-CMU-PROD-VIP.ANDREW.CMU.EDU (128.2.42.52)  134.211 ms  135.335 ms  135.988 ms
guest-zaly03@adm68:~$
```

Figura 6: Resultados do Ping e Traceroute realizados a www.cmu.edu

```
47 14.498478637  193.136.134.158  10.10.10.2      ICMP  102 Time-to-live exceeded (Time to live exceeded in transit)
48 14.498632675  10.10.10.2       128.2.42.52     UDP    74 32804 → 33454 Len=32
49 14.499197201  193.136.134.158  10.10.10.2      ICMP  102 Time-to-live exceeded (Time to live exceeded in transit)
50 14.499350839  10.10.10.2       128.2.42.52     UDP    74 53940 → 33455 Len=32
51 14.500340188  194.117.12.129   10.10.10.2      ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
52 14.500517597  10.10.10.2       128.2.42.52     UDP    74 51112 → 33456 Len=32
53 14.501239562  194.117.12.129   10.10.10.2      ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
54 14.501442406  10.10.10.2       128.2.42.52     UDP    74 36545 → 33457 Len=32
55 14.502005643  194.117.12.129   10.10.10.2      ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)

Internet Protocol Version 4, Src: 10.10.10.2, Dst: 128.2.42.52
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 60
    Identification: 0xe747 (59207)
  > Flags: 0x0000
    Time to live: 7
    Protocol: UDP (17)
    Header checksum: 0x0e28 [validation disabled]
    [Header checksum status: Unverified]
```

Figura 7: Resultado captura de tráfego do Wireshark da rota obtida para www.cmu.edu, neste caso já com TTL de 7.

Como podemos observar, a partir da captura de tráfego, para efetuar o traceroute são utilizadas mensagens UDP, dado que estamos a usar o traceroute defeito do linux. O TTL (IPv4) vai aumentando e as respostas de cada um dos routers é feita utilizando ICMP, "ICMP Time exceeded", no caso dos routers intermédios e "ICMP Destination Unreachable", no caso do router destino. Dado que estamos a usar UDP, a resposta do endereço IP final é realizada com "ICMP Destination Unreachable". Caso estivéssemos a utlizar "ICMP Echo", a mensagem de resposta seria "ICMP Echo Reply".

```
177 15.490065579  128.2.0.202    10.10.10.2      ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
179 15.491006490  128.2.0.202    10.10.10.2      ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
180 15.491357220  128.2.0.202    10.10.10.2      ICMP   70 Time-to-live exceeded (Time to live exceeded in transit)
182 15.491708842  128.2.42.52    10.10.10.2      ICMP   70 Destination unreachable (Port unreachable)
183 15.492576484  128.2.42.52    10.10.10.2      ICMP   70 Destination unreachable (Port unreachable)
184 15.492870906  128.2.42.52    10.10.10.2      ICMP   70 Destination unreachable (Port unreachable)

> Frame 39: 102 bytes on wire (816 bits), 102 bytes captured (816 bits) on interface 0
> Ethernet II, Src: Cisco_00:68:ab (d0:57:4c:00:68:ab), Dst: Micro-St_ca:a9:1c (d8:cb:8a:ca:a9:1c)
> Internet Protocol Version 4, Src: 193.136.143.253, Dst: 10.10.10.2
> Internet Control Message Protocol
```

Figura 8: Resposta ICMP Destination Unreachable do router final, com IP 128.2.42.52, e ICMP Time-to-live Exceeded, no caso dos routers intermédios

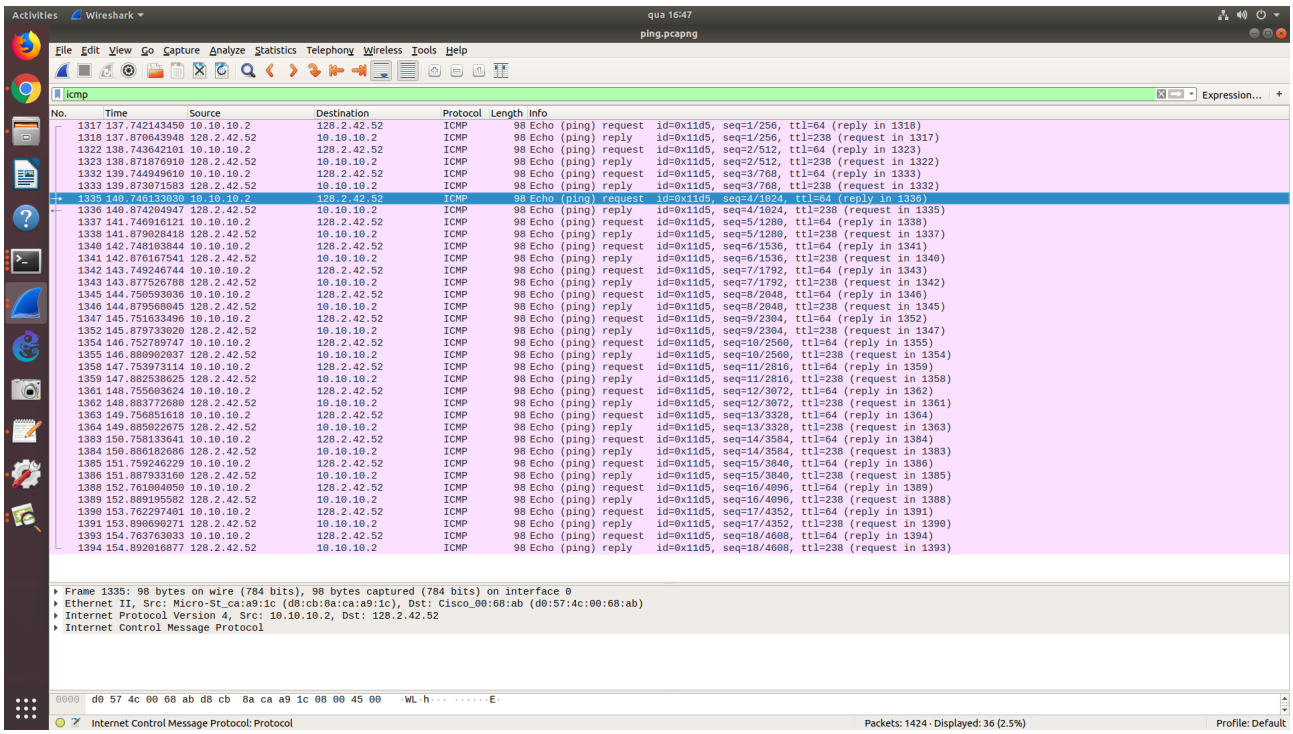

Figura 9: Captura de tráfego do ping a www.cmu.edu obtido no Wireshark com o filtro para icmp

O ping usa o ICMP para tentar obter uma resposta do destino, mais especificamente envia mensages "ICMP Echo".
Neste caso, como existe uma rota para o IP destino, o mesmo responde com uma mensagem "ICMP Echo Reply", caso contrário o resultado seria "Destination Host Unreachable", devido os pacotes "ICMP Time Exceeded Recebidos".

O ping funciona da seguinte maneira:
1) A fonte gera uma unidade de dados do protocolo (PDU) ICMP.
2) A PDU ICMP é encapsulada num datagrama IP, com o ip fonte e o ip destino especificados no cabeçalho ip. Neste momento, o datagrama é referida como uma mensagem "ICMP Echo", mas do ponto de vista da rede é vista como um datagrama IP.
3) A fonte anota a tempo local do seu relógio quando transmite o datagrama IP para o destino. Cada "host"verifica o datagrama IP para ver se o endereço destino correponde ao seu.
4) Se o endereço IP não não corresponder ao do "local host", o datagrama IP é encaminhado para a rede onde esse endereço IP reside.
5) O "host"destino recebe o datagrama IP e encontra uma correspondência entre o seu endereço IP e o do datagrama ou ao do "all hosts adress"(tudo 1's no campo "host"do endereço IP).
6) O "host"destino repara que a informação "ICMP ECHO"consta no datagrama IP e discarta o datagrama IP/ICMP ECHO original.
7) O "host"destino cria uma resposta "ICMP ECHO REPLY", encapsula-a num datagrama IP colocando o seu próprio endereço IP com fonte e o da fonte original como destino do datagrama.
8) O novo datagrama IP é roteado de volta para a origem do PING. O "host"fonte recebe-o, repara no tempo da receção do pacote e compara com o tempo de envio e, por fim, imprime para o ecrã a informação do PING, incluindo o tempo decorrido.

5-b)

No teste *iperf* utilizando TCP numa só direção obtivemos um *throughput* de 69.2Mbit/s, valor este que é bastante semelhante ao *throughput* obtido para download no *Speedtest*. Isto porque também a aplicação *Speedtest* utiliza o protocolo TCP.

Relativamente ao teste *iperf* que utiliza TCP nas duas direções, obtivemos valores de *throughput* 29.2Mbit/s e 36.8Mbit/s. Neste caso a soma destes valores é semelhante ao *throughput* obtido para a situação anterior, uma vez que como os testes nas duas direções estão a ser executados simultâneamente, o *throughput* total tem que ser dividido pelos dois testes.

Quanto ao teste *iperf* que utiliza UDP numa direção, verificamos um *throughput* de 95.8Mbit/s, ou seja, um valor que é praticamente a largura de banda da interface (100Mbit/s). Este valor é possivel para UDP porque, ao contrário de TCP, o protocolo UDP não realiza verificações de erros nem qualquer tipo de controlo de congestão de tráfego.

Para o teste *iperf* que utiliza UDP nas duas direções, verificamos mais uma vez que a soma dos valores de *throughput* obtidos (23.3Mbit/s e 71.7Mbit/s) é semelhante ao valor obtido para o teste em UDP numa só direção.

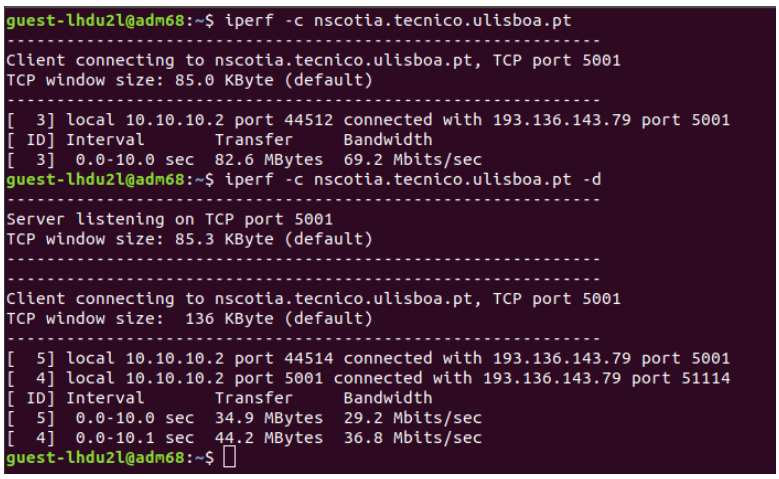

Figura 10: Teste da largura de banda com o comando *iperf* usando TCP, numa só direção e nas duas dircções em simultâneo

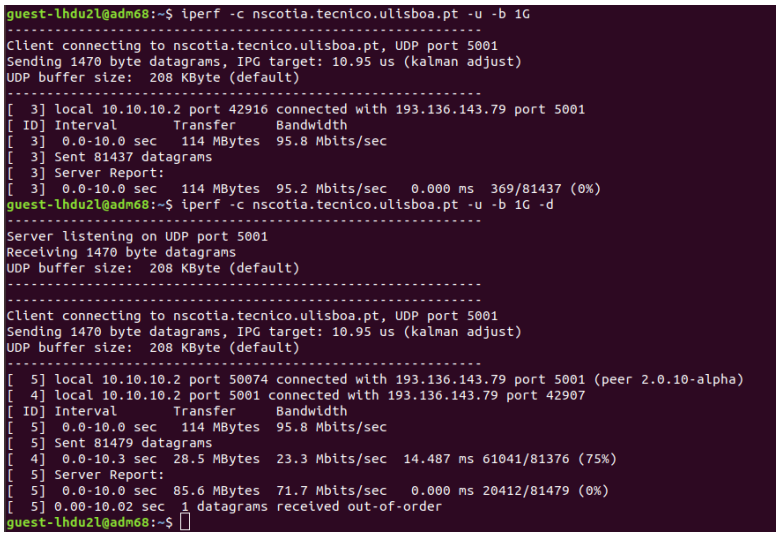

Figura 11: Teste da largura de banda com o comando *iperf* usando UDP, numa só direção e nas duas dircções em simultâneo

Figura 12: Teste da largura de banda com o Speedtest

6-b)

A aplicação "iperf"gera um tráfico UDP que pode ser alterado. O valor escolhido para este teste foi o de 1 Gbps, o que é muito superior aos 2 Mbps da rede. Deste modo, a aplicação satura a rede sem que a aplicação deixe de funcionar. Esta saturação dá-se pois o protocolo UDP não realiza verificações de erros nem qualquer tipo de controlo de congestão de tráfego. Por consequência, temos que no SpeedTest não há qualquer tipo de tráfego.

Sobre a utilização da "iperf"em TCP, este protocolo realiza controlo de congestão e de erros, pelo que se existir um pacote de informação perdido o processo para até que haja uma reenvio do pacote. Deste modo, os erros provocados pela saturação da rede fazem com que a transmissão TCP reduza o ritmo, permitindo assim que no SpeedTest se obtenha transmissão de tráfego, embora seja mais reduzido que os 2Mbps obtidos sem a aplicação "iperf"a correr em simultâneo.

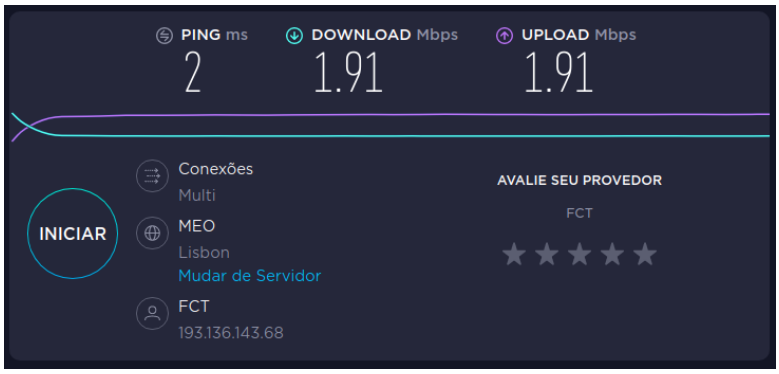

Figura 13: Teste de largura de banda realizado no SpeedTest sem "iperf"a correr.

Figura 14: Teste de largura de banda realizado no SpeedTest com "iperf"a correr em simultânio em ambas as direções usando TCP.

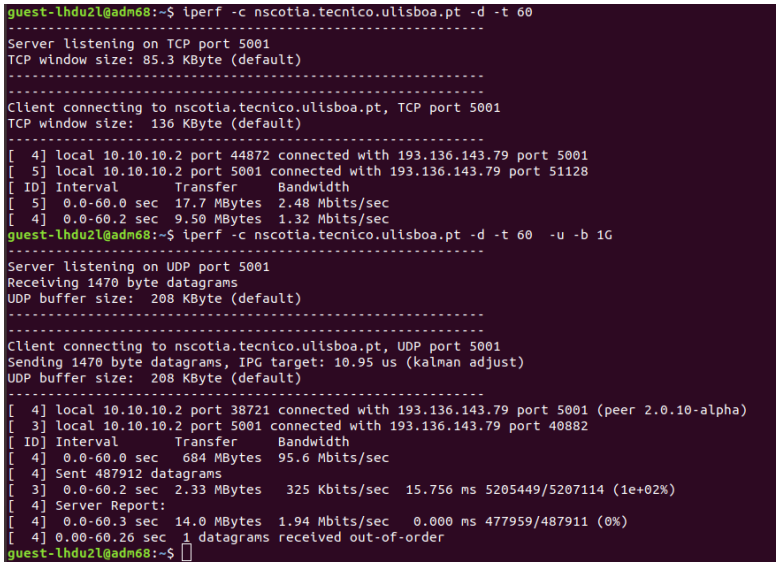

Figura 15: Terminal com a iperf a correr em UDP.

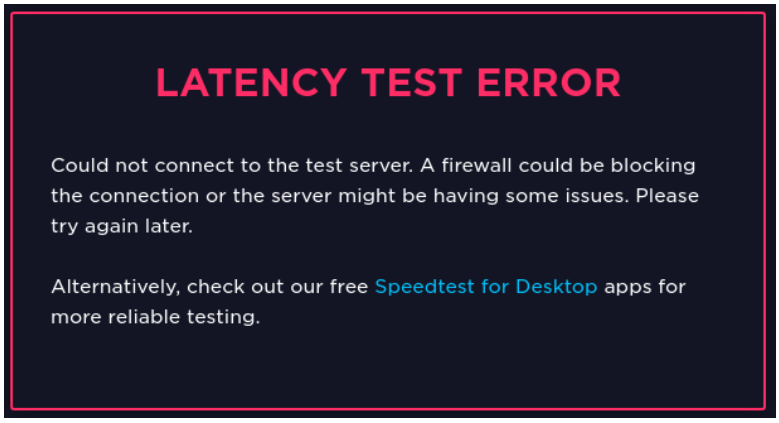

Figura 16: Teste de largura de banda realizado no SpeedTest com "iperf"a correr em simultânio em ambas as direções usando UDP (com um elevado débito).

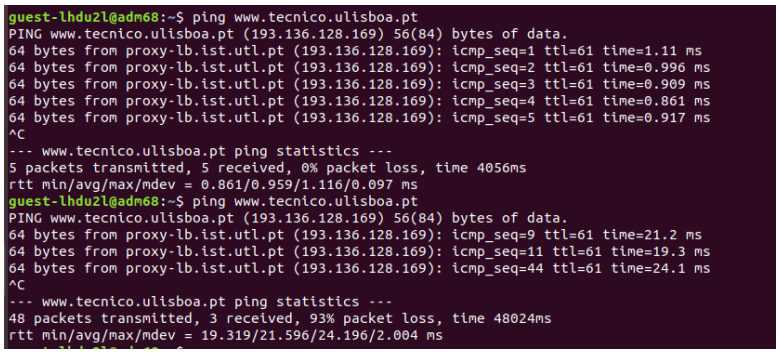

Figura 17: Ping realizado a www.tecnico.ulisboa.pt sem e com "iperf"a correr utilizando UDP nas duas direcções (com um elevado débito).