TP4: Protocolo IPv4

Universidade do Minho

Departamento de Informática, 4710-057 Braga, Portugal

Redes de Computadores

Autores:

Paulo Guedes, a74411@alunos.uminho.pt

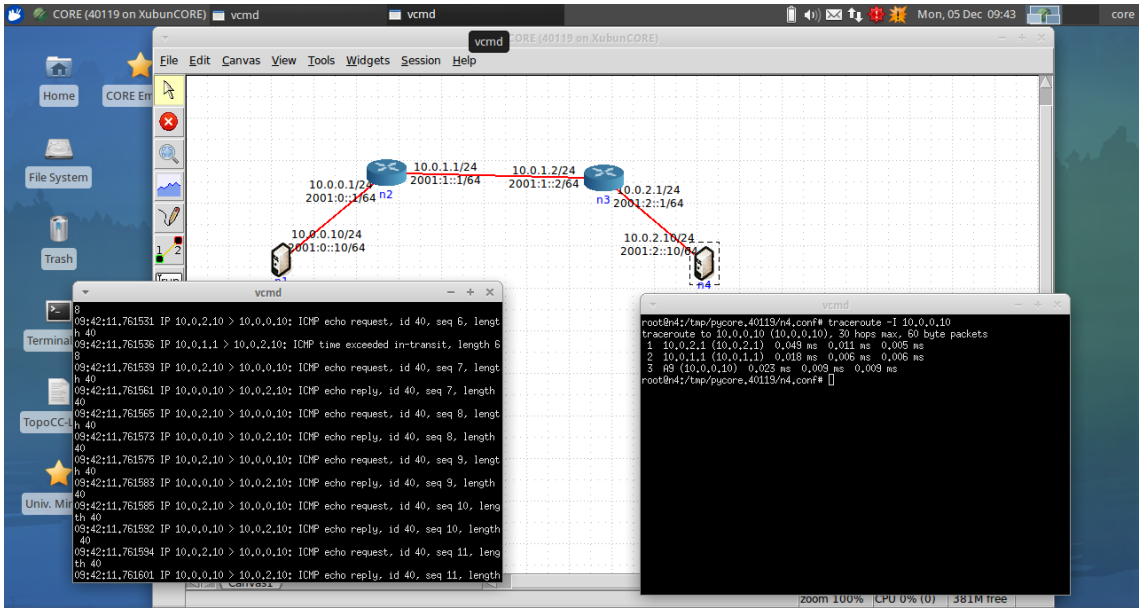
Diogo Gomes, a73825@alunos.uminho.pt

Carlos Campos, a74745@alunos.uminho.pt

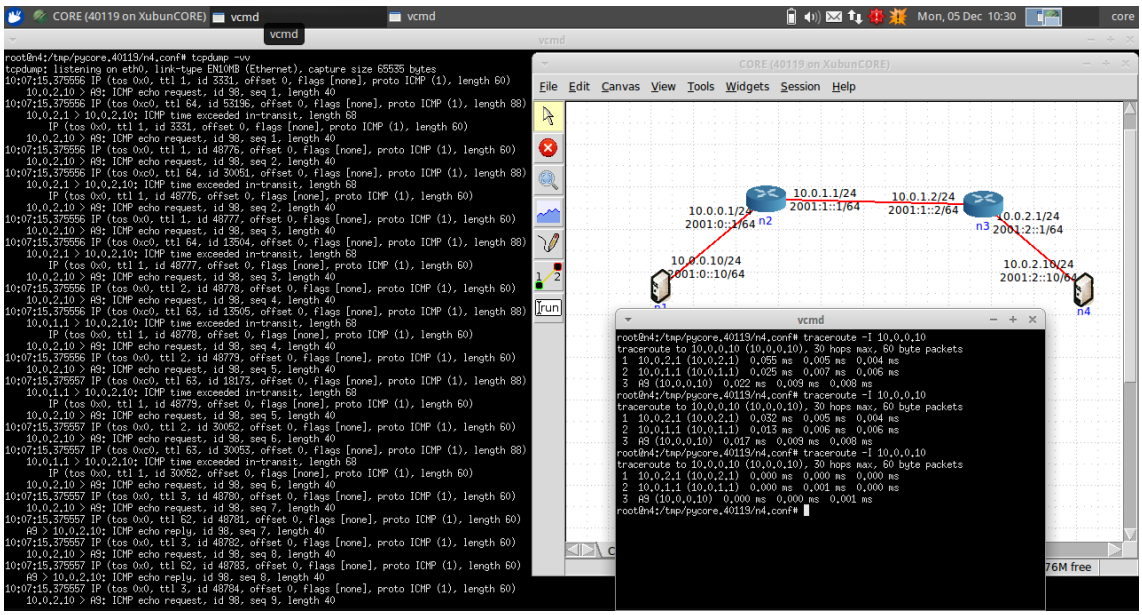
Parte I

1. Prepare uma **topologia CORE** para verificar o comportamento do *traceroute*. Ligue um *host* n1 a um *router* n2; o *router* n2 a um *router* n3 que, por sua vez, se liga a um *host* n4. (*Note que pode não existir conectividade IP imediata entre n1 e n4 até que o routing estabilize*).

a) Active o *wireshark* ou o *tcpdump* no *host* n4. Numa *shell* de n4, execute o comando *traceroute -I* para o endereço IP do *host* n1.

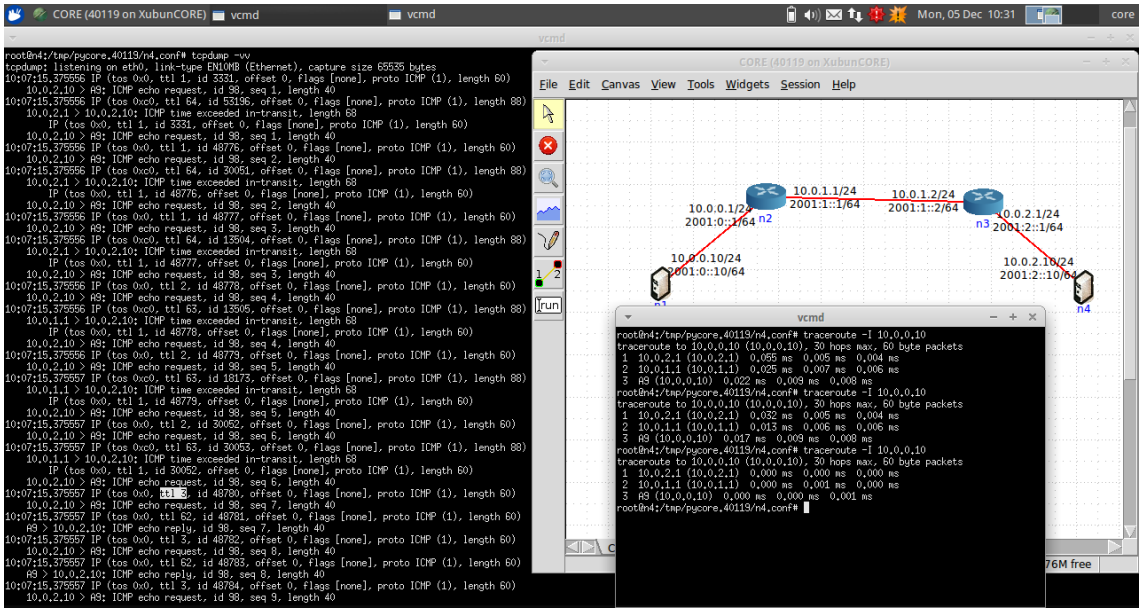


b) Registe e analise o tráfego ICMP enviado por n4 e o tráfego ICMP recebido como resposta. Comente os resultados face ao comportamento esperado.



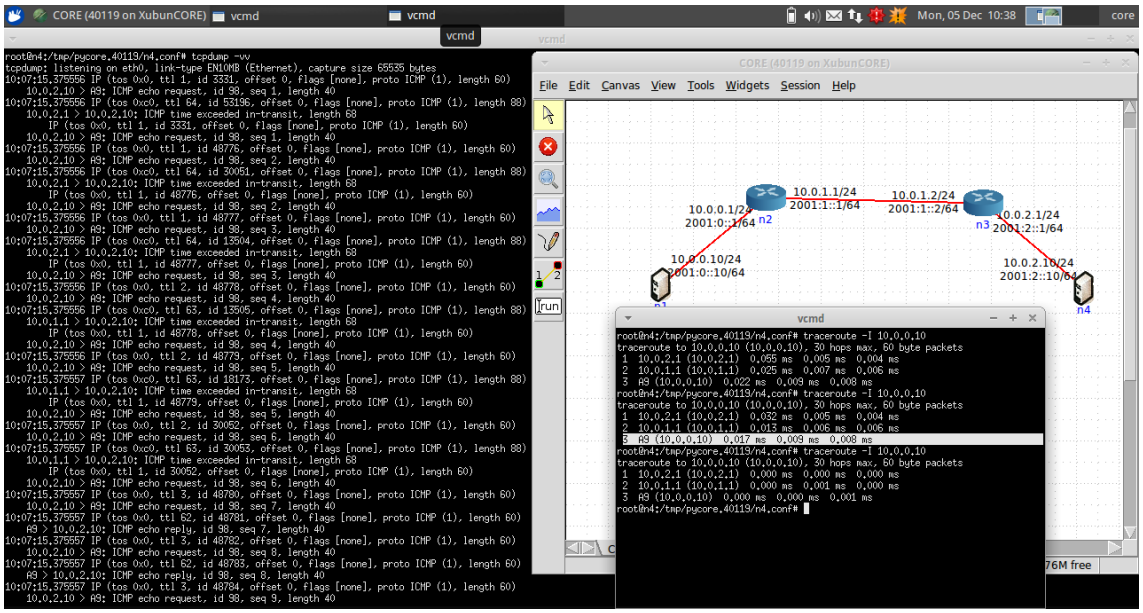
Podemos verificar a cada vez são feitos 3 eccho requests, de modo a verificar que este chega ao respetivo TTL. Para além disso também verificamos que antes de o request chegar ao destino este passa pelos intermediários.

c) Qual deve ser o valor inicial mínimo do campo TTL para alcançar o destino n1? Verifique na prática que a sua resposta está correta.



O que seria esperado era o TTL 3, e como podemos observar este foi obtido.

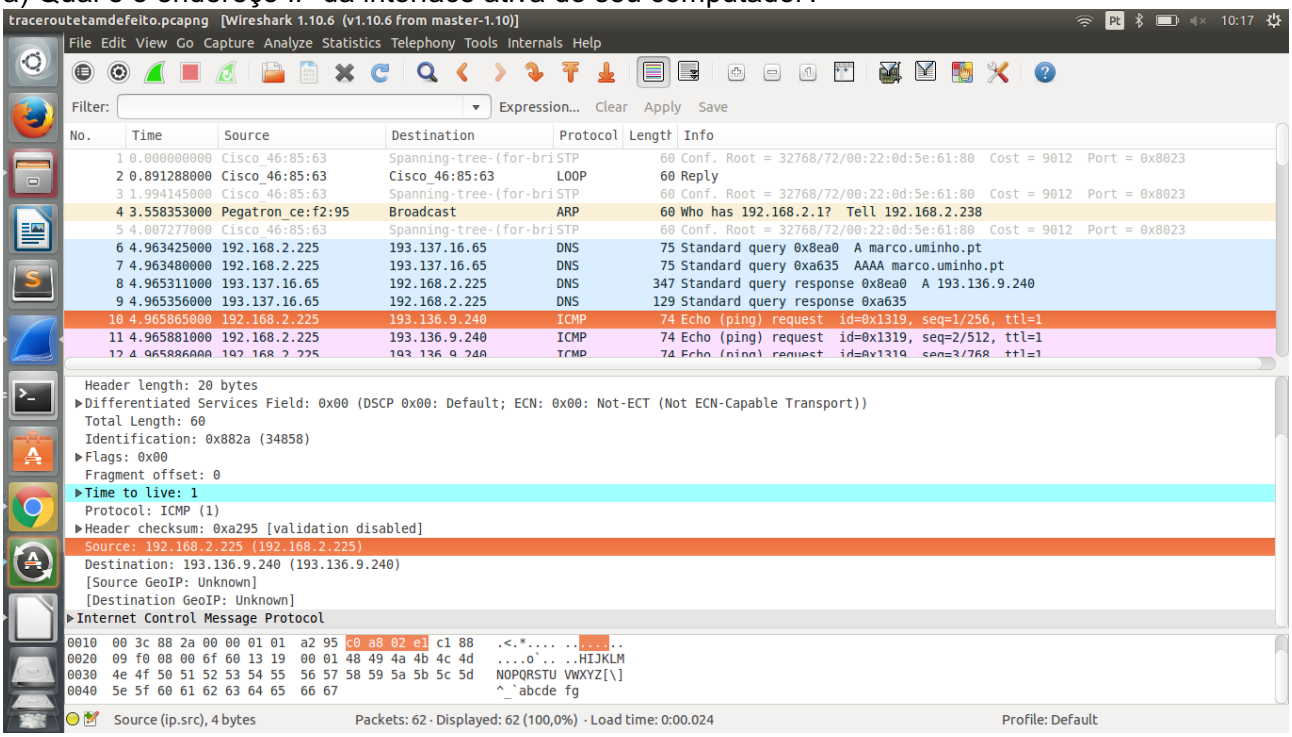
d) Qual o valor médio do tempo de ida-e-volta (Round-Trip Time) obtido?



RTT≈0,011(3) ms, utilizando os valores marcados na imagem acima.

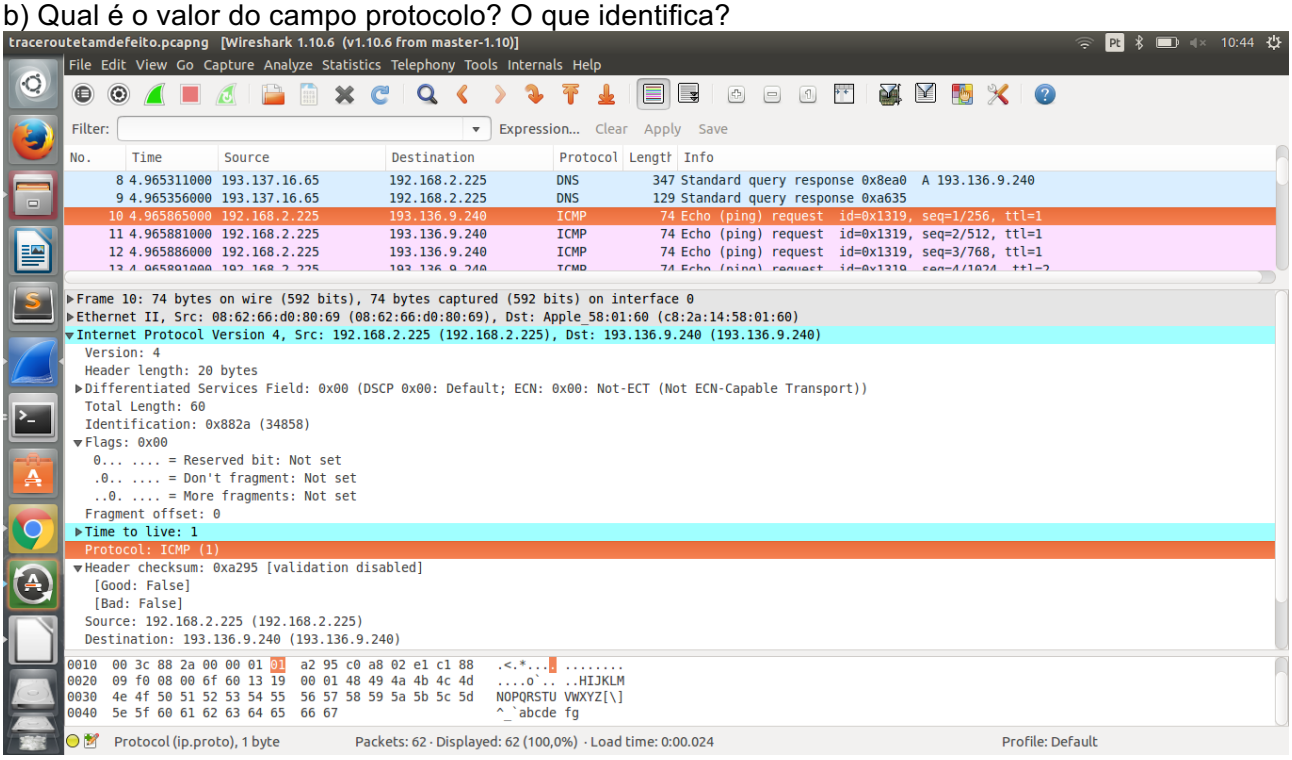
2)

a) Qual é o endereço IP da interface ativa do seu computador?



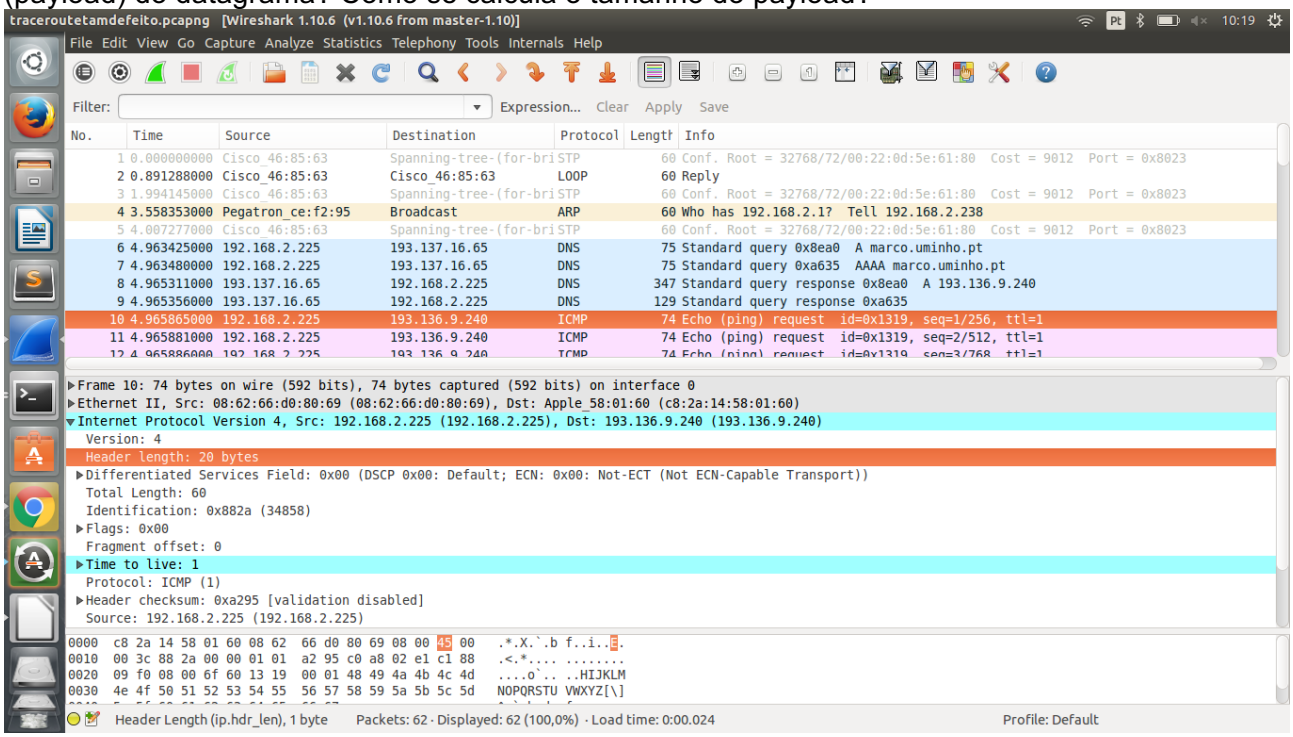
O endereço IP é 192.168.2.225.

b) Qual é o valor do campo protocolo? O que identifica?



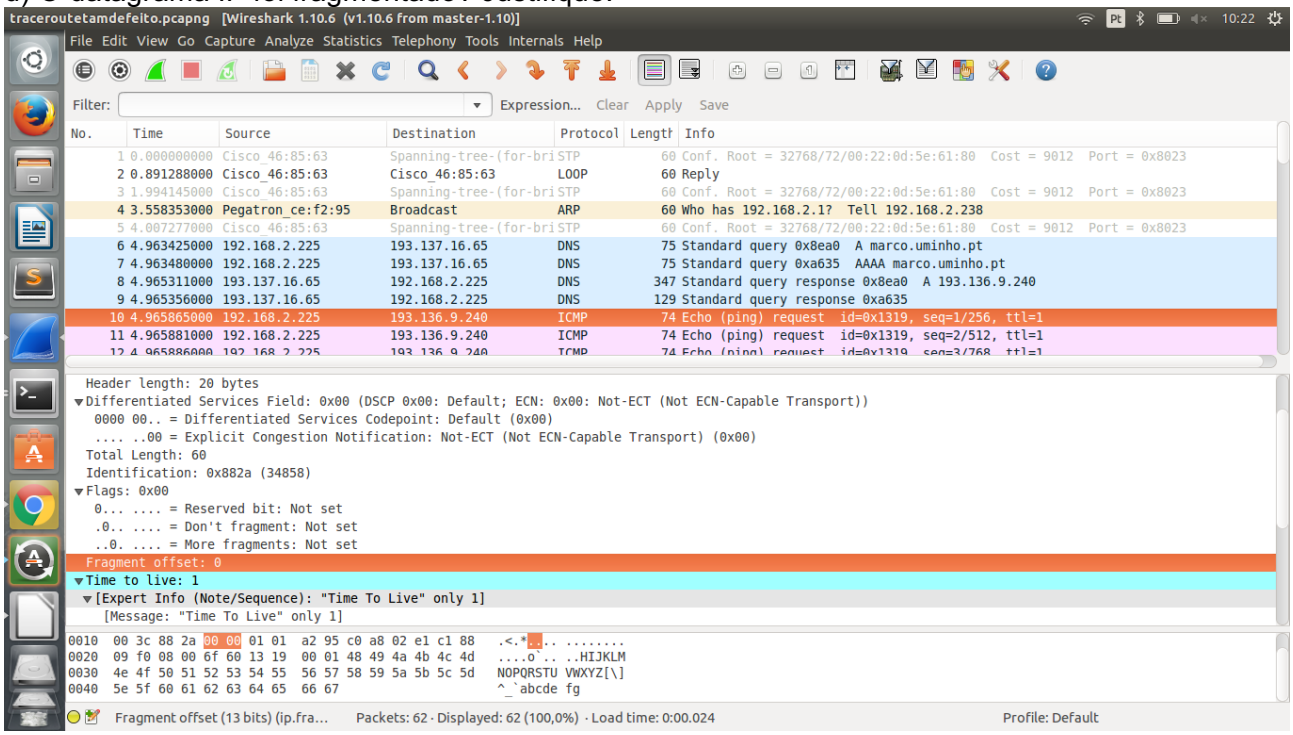
O valor do campo protocolo é 1 e identifica que está a ser usado o protocolo ICMP.

c) Quantos bytes tem o cabeçalho IP(v4)? Quantos bytes tem o campo de dados (payload) do datagrama? Como se calcula o tamanho do payload?



O cabeçalho IP(v4) tem 20 bytes. O campo de dados do datagrama tem 40 bytes. Calcula-se subtraindo o cabeçalho pelo total comprimento do IP(v4): 60-20=40 bytes.

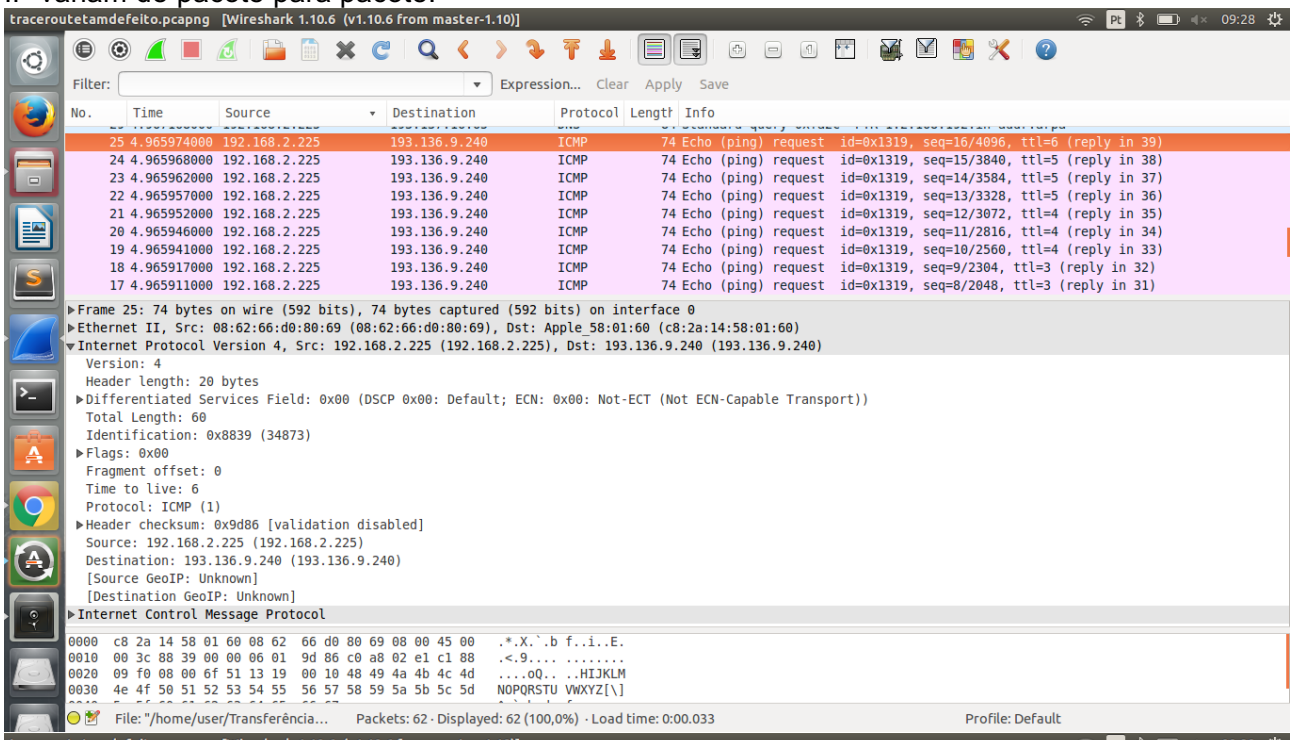
d) O datagrama IP foi fragmentado? Justifique.

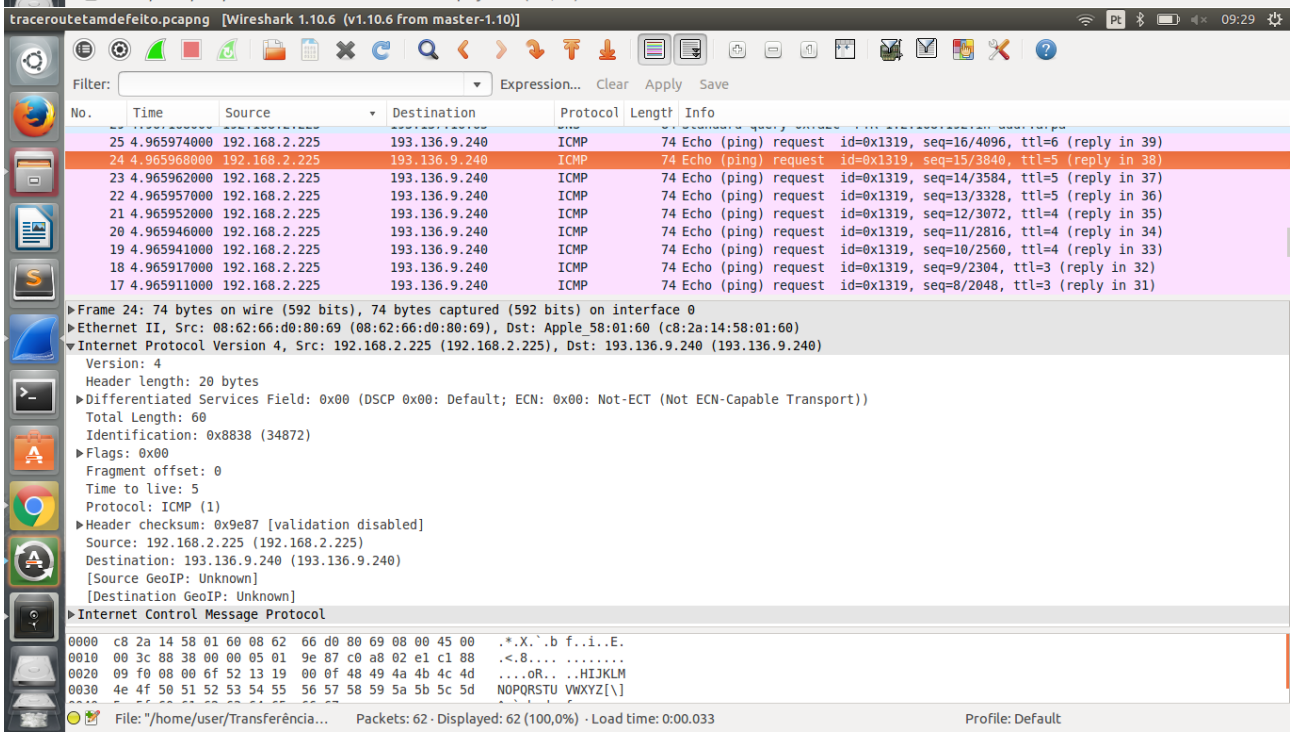


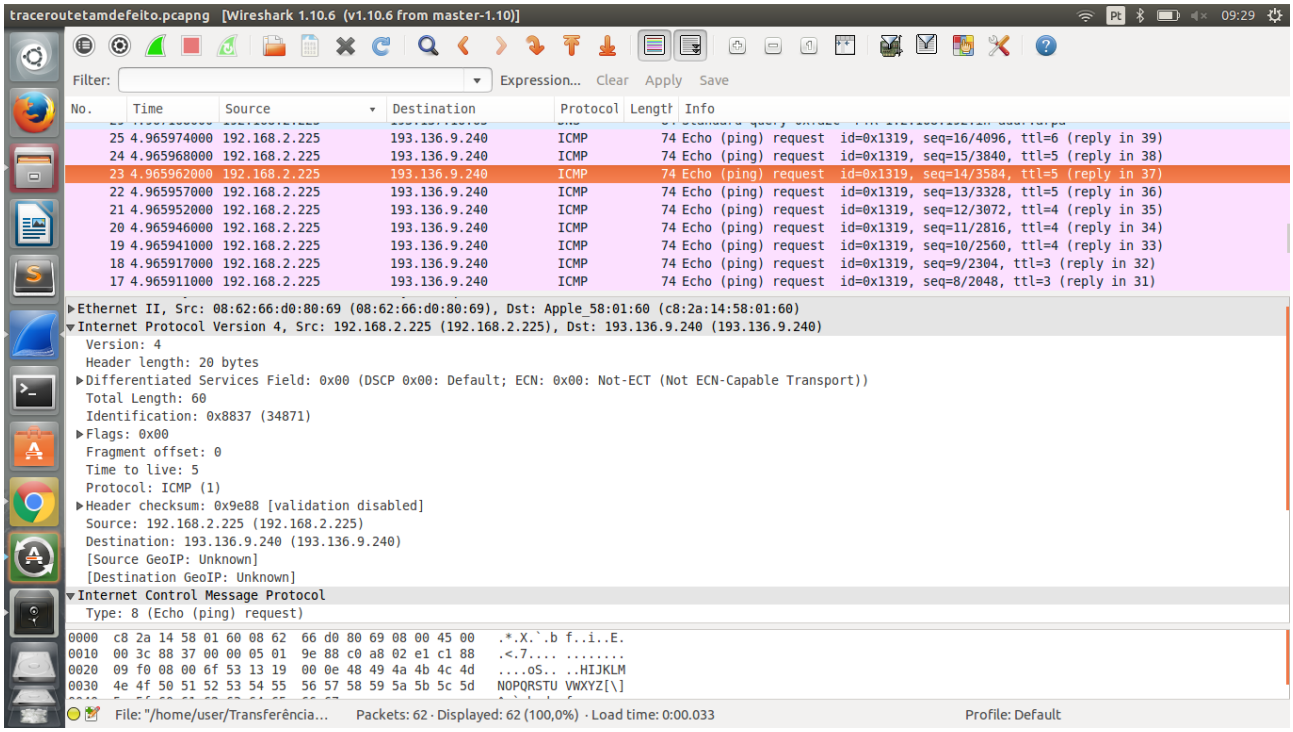
Não, porque o fragment offset é 0.

e) Ordene os pacotes capturados de acordo com o endereço IP fonte (e.g., selecionando o cabeçalho da coluna Source), e analise a sequência de tráfego ICMP

gerado a partir do endereço IP atribuído à sua máquina. Para a sequência de mensagens ICMP enviadas pelo seu computador, indique que campos do cabeçalho IP variam de pacote para pacote.

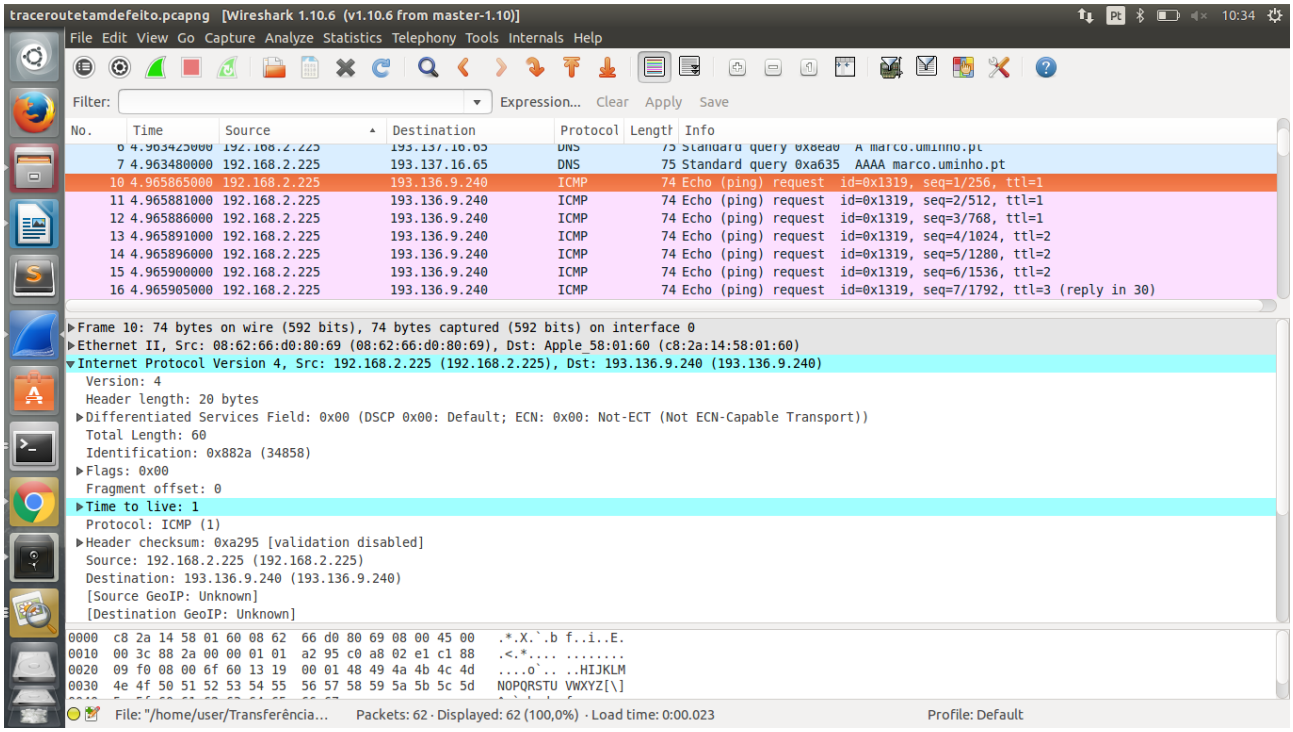


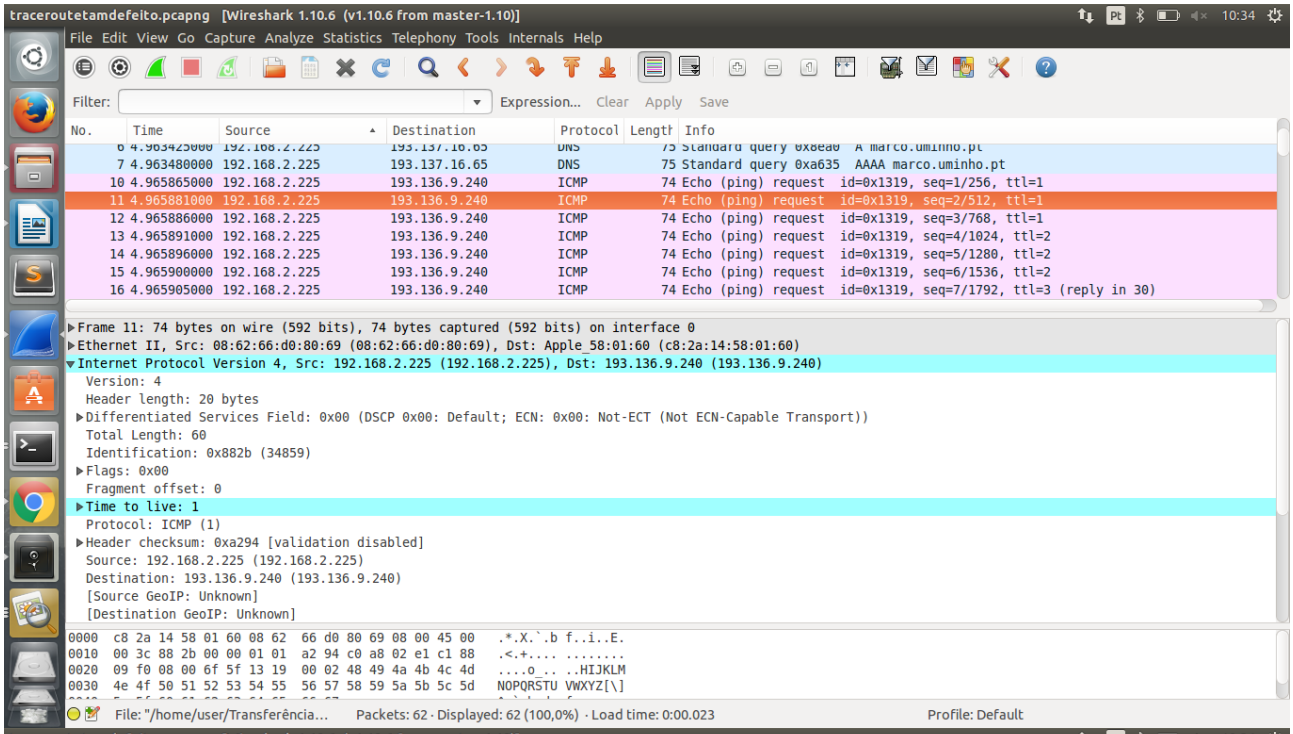


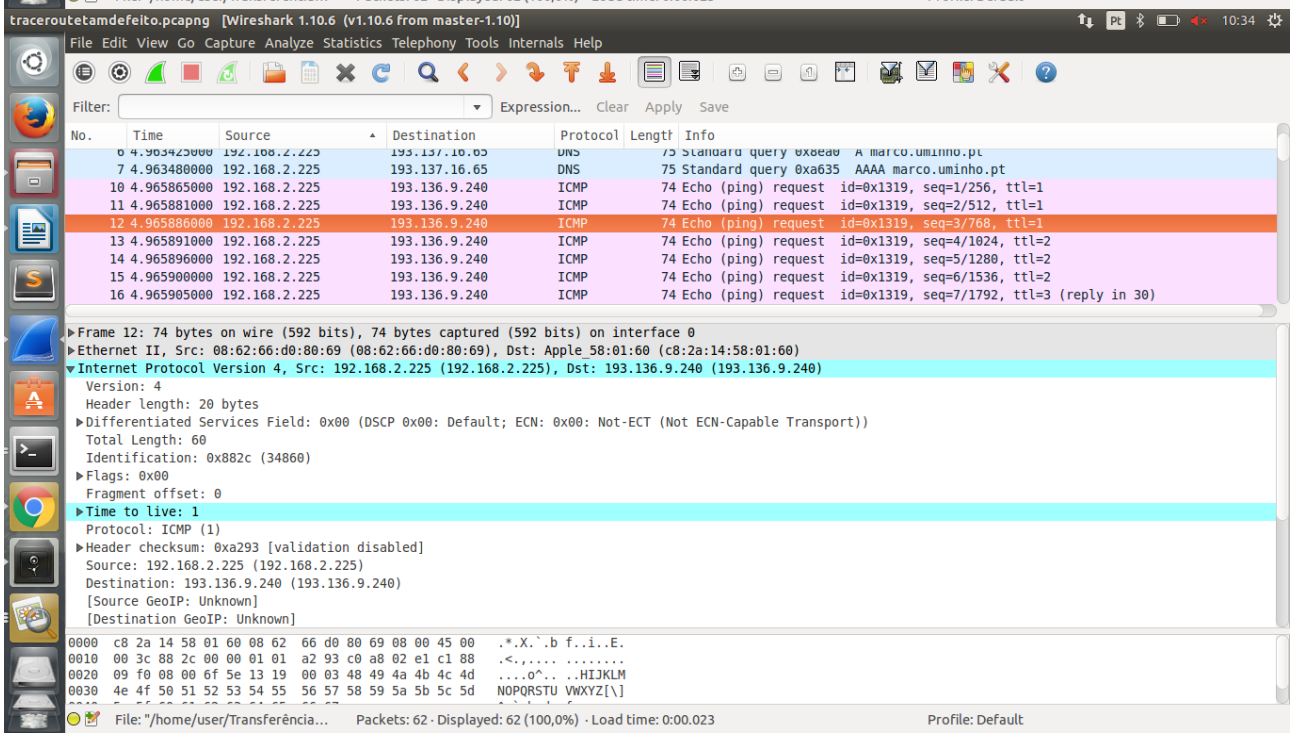


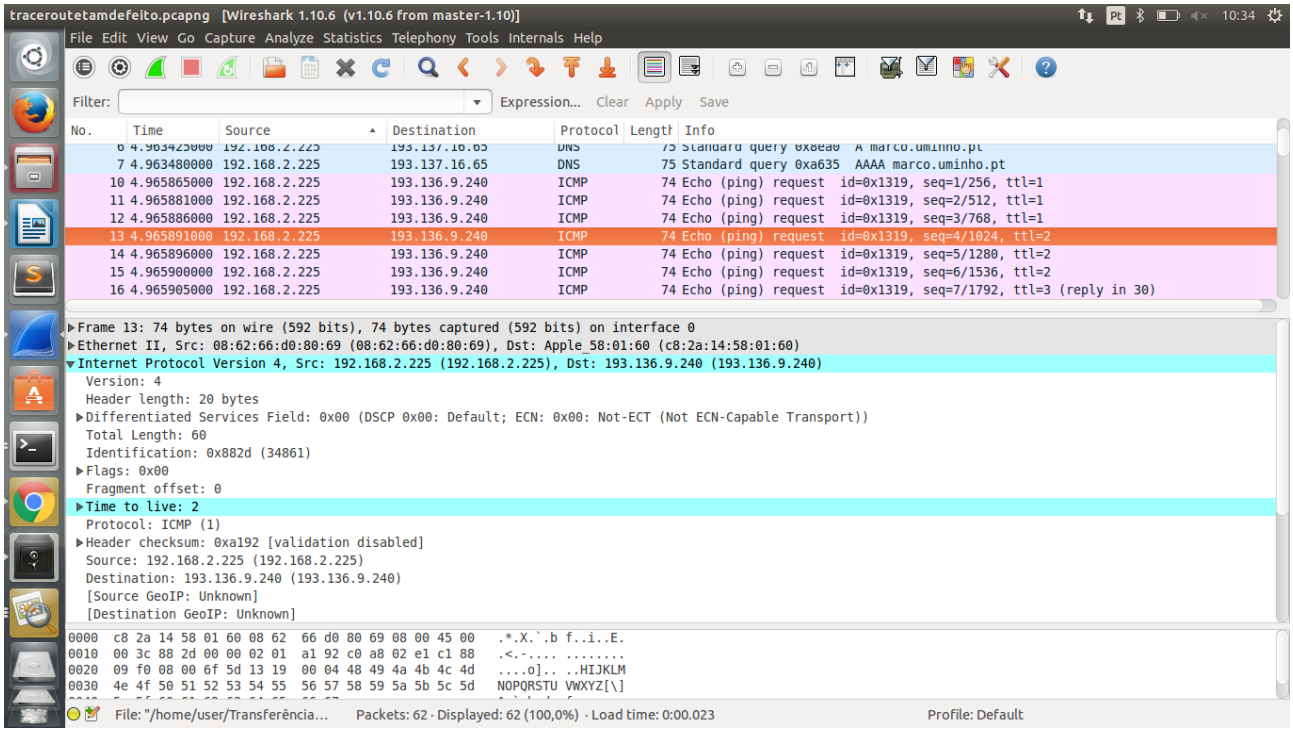
O campos do cabeçalho IP que variam de pacote para pacote são : Identification, TTL, header checksum.

f) Observa algum padrão nos valores do campo de Identificação do datagrama IP e TTL?



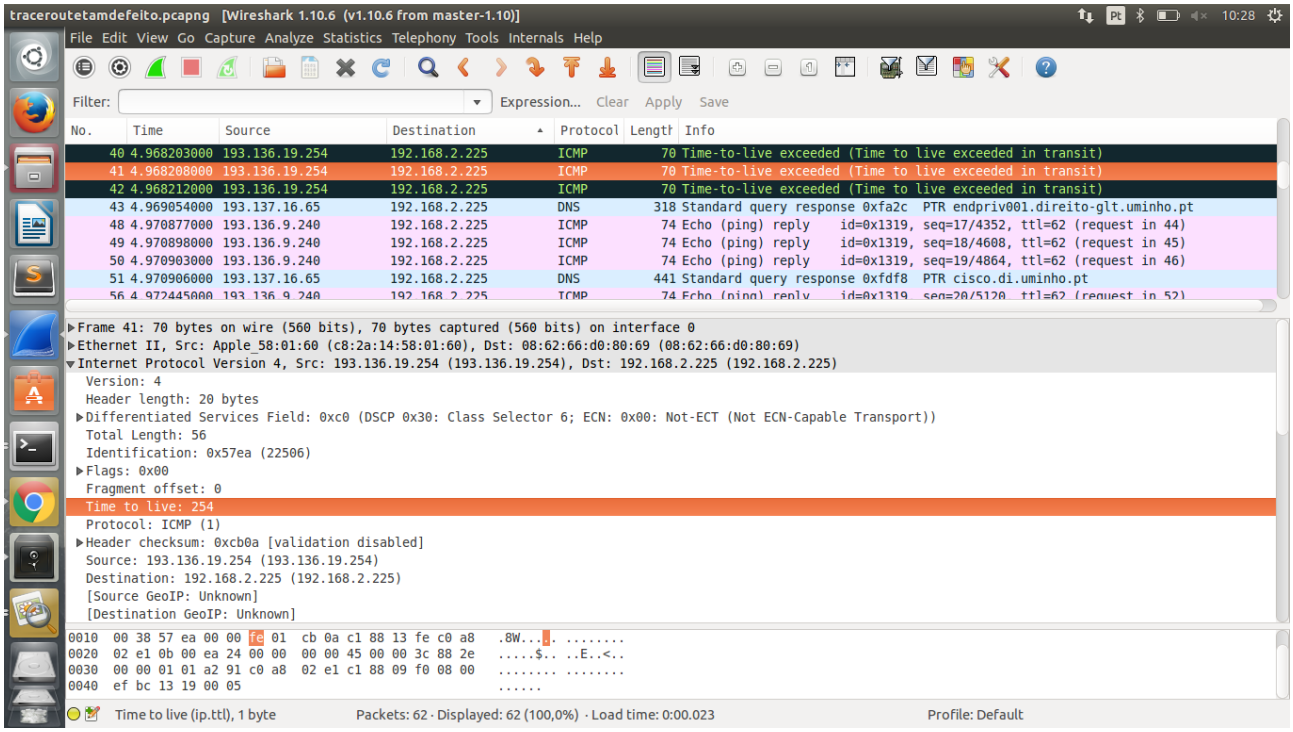

traceroutetamdefeito.pcapng [Wireshark 1.10.6 (v1.10.6 from master-1.10)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: Expression... Clear Apply Save
No. Time Source Destination Protocol Length Info
7 4.963480000 192.168.2.225 193.137.16.65 DNS 75 Standard query 0xa635 AAAA marco.uminho.pt
10 4.965865000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=1/256, ttl=1
11 4.965881000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=2/512, ttl=1
12 4.965886000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=3/768, ttl=1
13 4.965891000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=4/1024, ttl=2
14 4.965896000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=5/1280, ttl=2
15 4.965900000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=6/1536, ttl=2
16 4.965905000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=7/1792, ttl=3 (reply in 30)
Frame 11: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
Ethernet II, Src: 08:62:66:d0:80:69 (08:62:66:d0:80:69), Dst: Apple_58:01:60 (c8:2a:14:58:01:60)
Internet Protocol Version 4, Src: 192.168.2.225 (192.168.2.225), Dst: 193.136.9.240 (193.136.9.240)
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
Total Length: 60
Identification: 0x882b (34859)
Flags: 0x00
Fragment offset: 0
Time to live: 1
Protocol: ICMP (1)
Header checksum: 0xa294 [validation disabled]
Source: 192.168.2.225 (192.168.2.225)
Destination: 193.136.9.240 (193.136.9.240)
[Source GeoIP: Unknown]
[Destination GeoIP: Unknown]
0000 c8 2a 14 58 01 60 08 62 66 d0 80 69 08 00 45 00 .*.X.`.b f..i..E.
0010 00 3c 88 2b 00 00 01 01 a2 94 c0 a8 02 e1 c1 88 .<.+.... ....
0020 09 f0 08 00 6f 5f 13 19 00 02 48 49 4a 4b 4c 4d ....o_.. ..HIJKLM
0030 4e 4f 50 51 52 53 54 55 56 57 58 59 5a 5b 5c 5d NOPQRSTU VWXYZ[\]
File: "/home/user/Transferência... Packets: 62 · Displayed: 62 (100,0%) · Load time: 0:00.023 Profile: Default

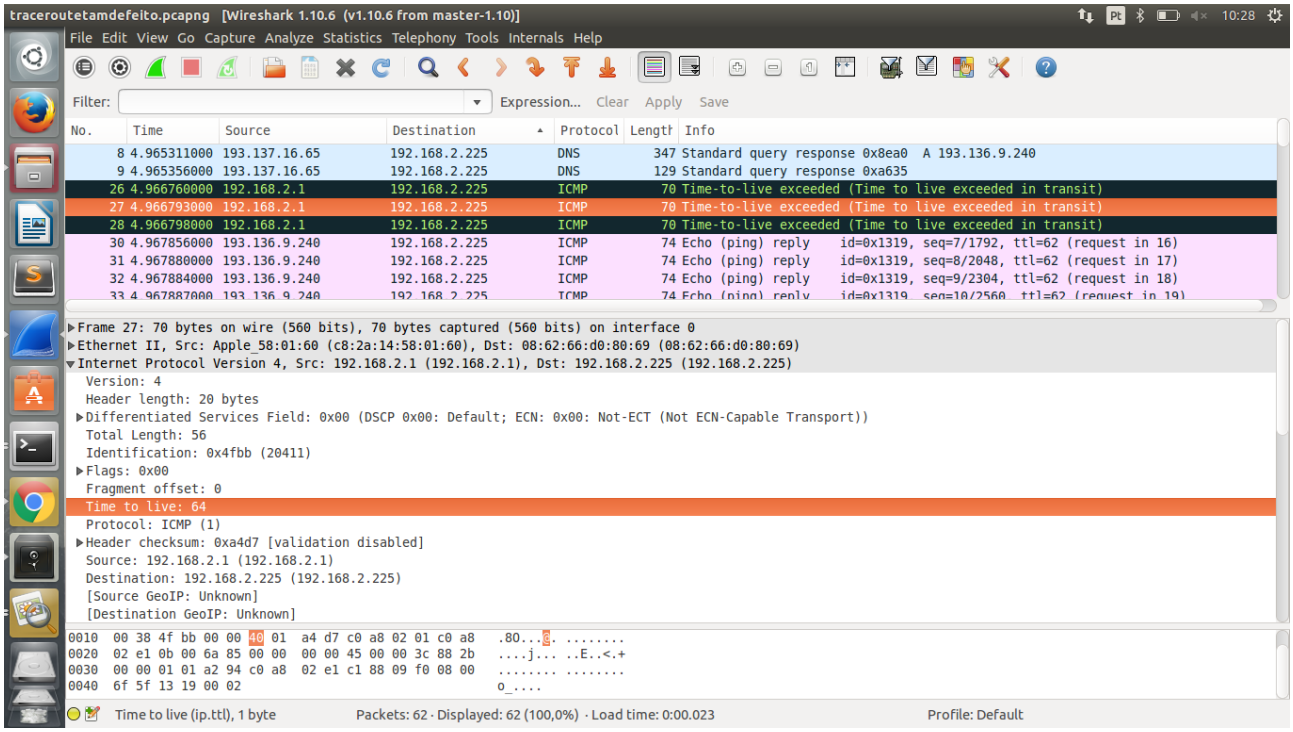

traceroutetamdefeito.pcapng [Wireshark 1.10.6 (v1.10.6 from master-1.10)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: Expression... Clear Apply Save
No. Time Source Destination Protocol Length Info
7 4.963480000 192.168.2.225 193.137.16.65 DNS 75 Standard query 0xa635 AAAA marco.uminho.pt
10 4.965865000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=1/256, ttl=1
11 4.965881000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=2/512, ttl=1
12 4.965886000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=3/768, ttl=1
13 4.965891000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=4/1024, ttl=2
14 4.965896000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=5/1280, ttl=2
15 4.965900000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=6/1536, ttl=2
16 4.965905000 192.168.2.225 193.136.9.240 ICMP 74 Echo (ping) request id=0x1319, seq=7/1792, ttl=3 (reply in 30)
Frame 12: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
Ethernet II, Src: 08:62:66:d0:80:69 (08:62:66:d0:80:69), Dst: Apple_58:01:60 (c8:2a:14:58:01:60)
Internet Protocol Version 4, Src: 192.168.2.225 (192.168.2.225), Dst: 193.136.9.240 (193.136.9.240)
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
Total Length: 60
Identification: 0x882c (34860)
Flags: 0x00
Fragment offset: 0
Time to live: 1
Protocol: ICMP (1)
Header checksum: 0xa293 [validation disabled]
Source: 192.168.2.225 (192.168.2.225)
Destination: 193.136.9.240 (193.136.9.240)
[Source GeoIP: Unknown]
[Destination GeoIP: Unknown]
0000 c8 2a 14 58 01 60 08 62 66 d0 80 69 08 00 45 00 .*.X.`.b f..i..E.
0010 00 3c 88 2c 00 00 01 01 a2 93 c0 a8 02 e1 c1 88 .<.,.... ....
0020 09 f0 08 00 6f 5e 13 19 00 03 48 49 4a 4b 4c 4d ....o^.. ..HIJKLM
0030 4e 4f 50 51 52 53 54 55 56 57 58 59 5a 5b 5c 5d NOPQRSTU VWXYZ[\]
File: "/home/user/Transferência... Packets: 62 · Displayed: 62 (100,0%) · Load time: 0:00.023 Profile: Default



Os valores da identificação incrementam 1 valor a cada ICMP, e o valor TTL incrementa 1 valor a cada 3 tramas enviadas.

g) Ordene o tráfego capturado por endereço destino e encontre a série de respostas ICMP TTL exceeded enviadas ao seu computador. Qual é o valor do campo TTL? Esse valor permanece constante para todas as mensagens de resposta ICMP TTL exceeded enviados ao seu host? Porquê?

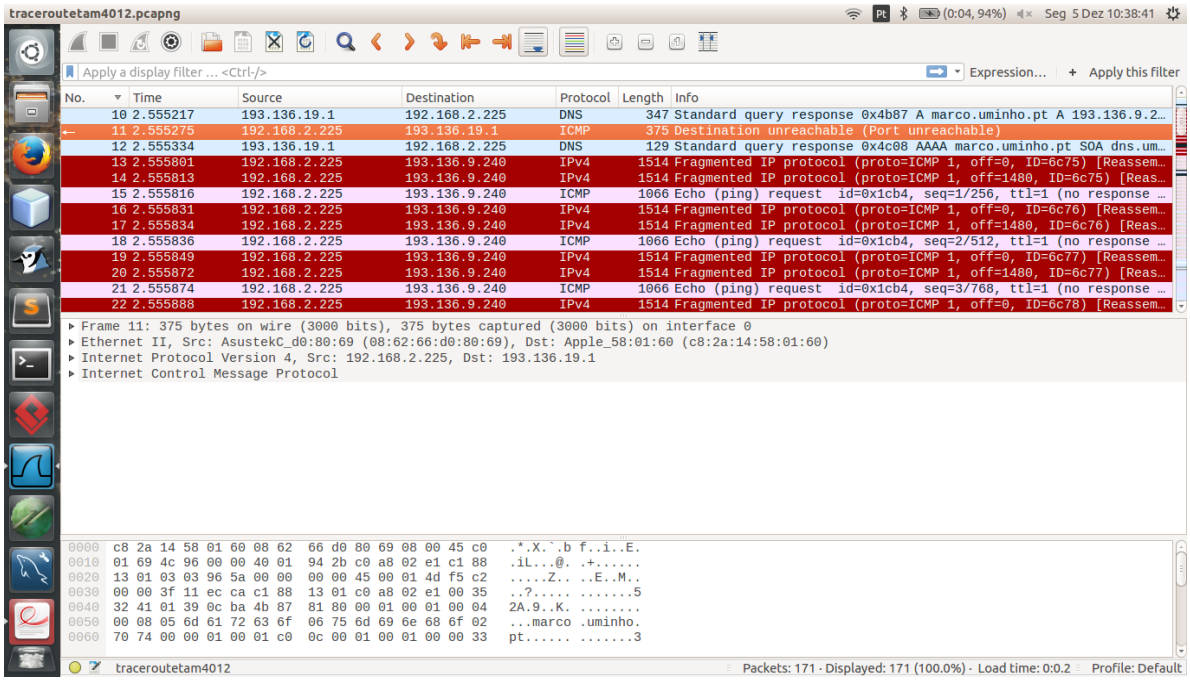




No TTL exceeded do primeiro salto, o campo TTL é 64, que permanece constante para todas as mensagens de resposta ICMP TTL do primeiro salto. No TTL exceeded do segundo salto, o campo TTL é 255, mas o valor apresentado é 254 por ter decrementado 1 salto, visto que este TTL exceeded é relativo ao segundo salto. É maior que o primeiro salto porque a dimensão da rede é maior no segundo equipamento.

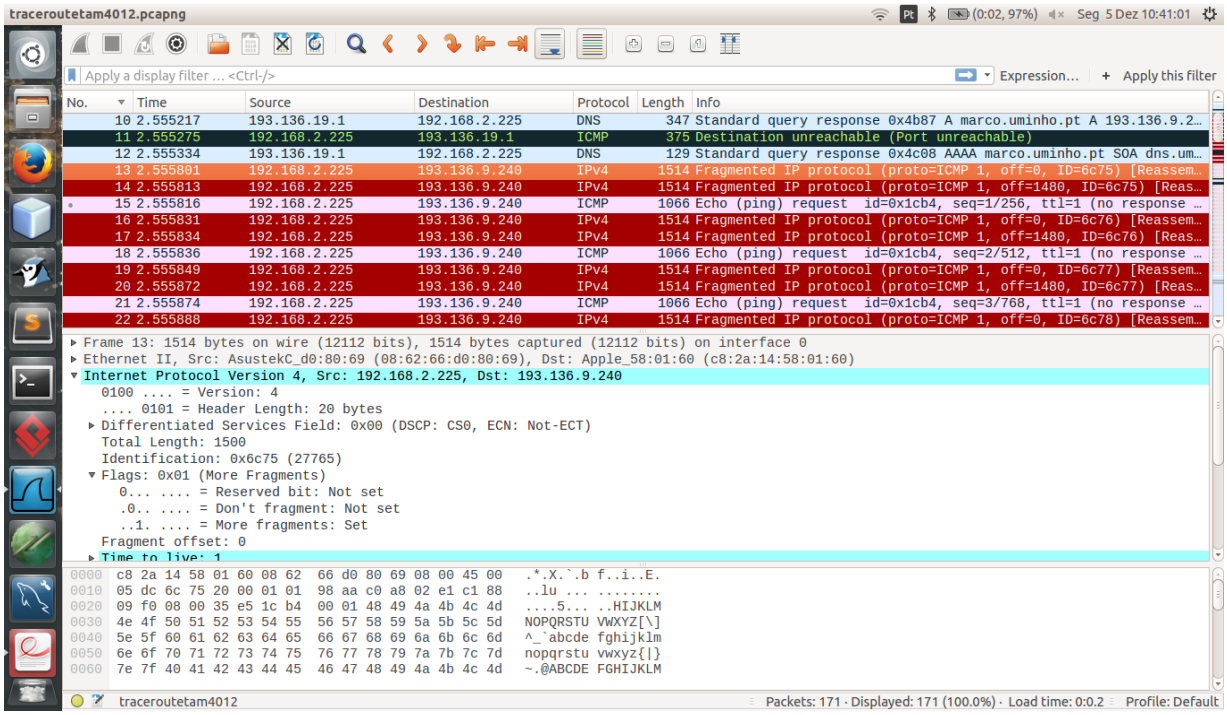
3) Pretende-se agora analisar a fragmentação de pacotes IP. Reponha a ordem do tráfego capturado usando a coluna do tempo de captura. Observe o tráfego depois do tamanho de pacote ter sido definido para 40XX bytes.

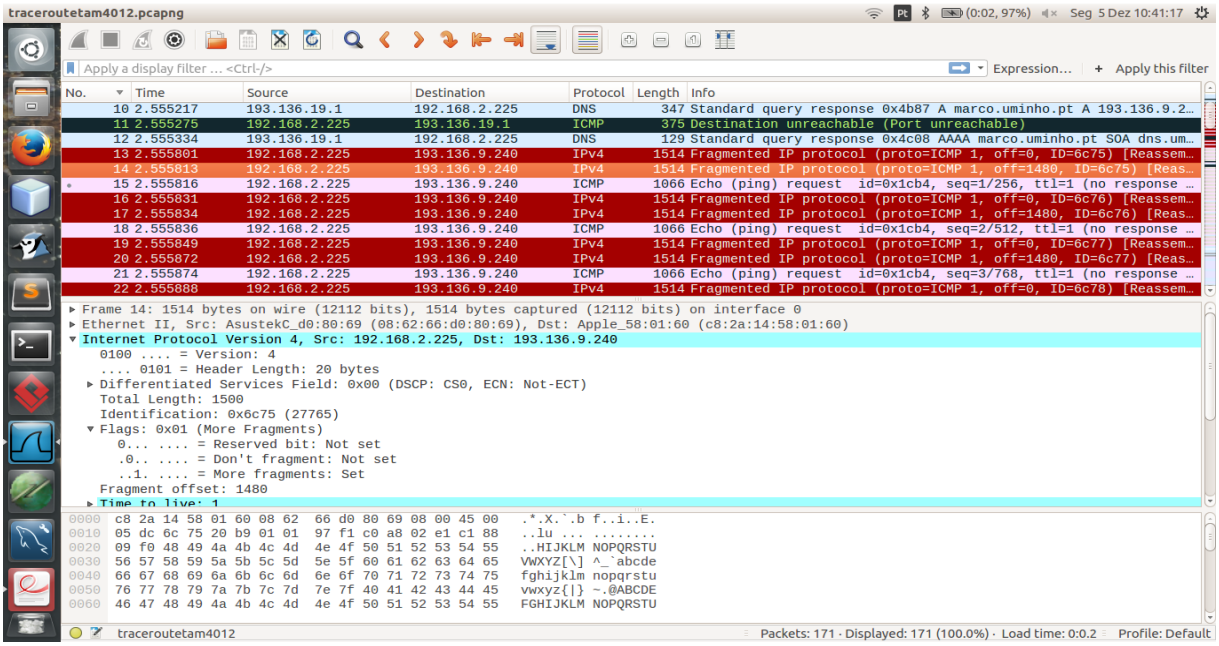
a) Localize a primeira mensagem ICMP. Porque é que houve necessidade de fragmentar o pacote inicial?



A primeira mensagem ICMP é o pacote número 11. Na camada 3, que é a camada de rede, segundo o protocolo da Ethernet, existe por defeito, um limite de transporte de 1500 bytes por pacote, ou seja, se um pacote ocupar mais espaço do que esse valor, existe a necessidade de fragmenta-lo de modo a que seja possível transmiti-lo.

b) Imprima o primeiro fragmento do datagrama IP segmentado. Que informação no cabeçalho indica que o datagrama foi fragmentado? Que informação no cabeçalho IP indica que se trata do primeiro fragmento? Qual é o tamanho deste datagrama IP?



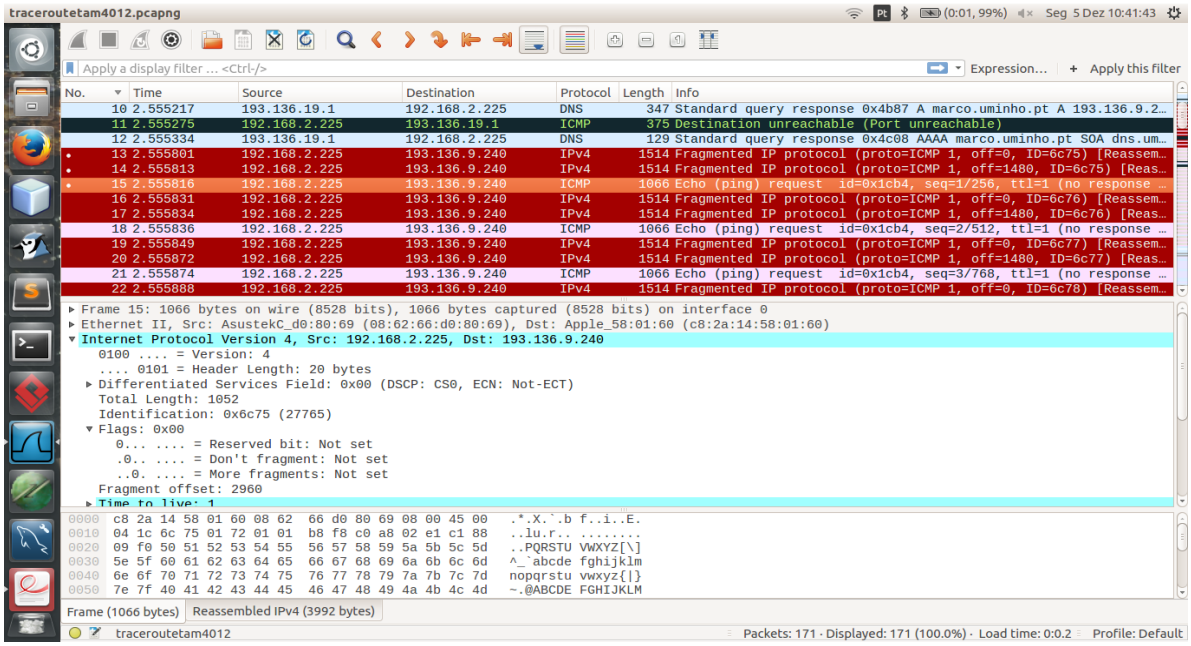


O primeiro fragmento do datagrama IP segmentado é o pacote número 13. No cabeçalho, as flags 0x01, mais especificamente, o 1, refere que existe mais fragmentos deste datagrama, logo isto indica que o pacote original foi fragmentado. No cabeçalho, o valor do campo offset é 0, portanto este é o primeiro fragmento do pacote fragmentado. Este datagrama IP tem o tamanho de 1500 bytes, que é o limite máximo referido na alínea anterior.

c) Imprima o segundo fragmento do datagrama IP original. Que informação do cabeçalho IP indica que não se trata do 1º fragmento? Há mais fragmentos? O que nos permite afirmar isso?

O campo offset, neste datagrama, tem o valor de 1480, isto indica que este mesmo, não é o primeiro fragmento do pacote original, que foi fragmentado. No campo das flags, mais especificamente, no campo antes do offset, a flag é 1, o que indica que existem mais fragmentos posteriores a este datagrama.

d) Quantos fragmentos foram criados a partir do datagrama original? Como se detecta o último fragmento correspondente ao datagrama original?



A partir do datagrama original, foram criados 3 fragmentos, que para este exemplo são os pacotes número 13, 14 e 15. No campo flag, do pacote número 15, o ultimo número é 0, o que indica que este é o ultimo fragmento do datagrama original.

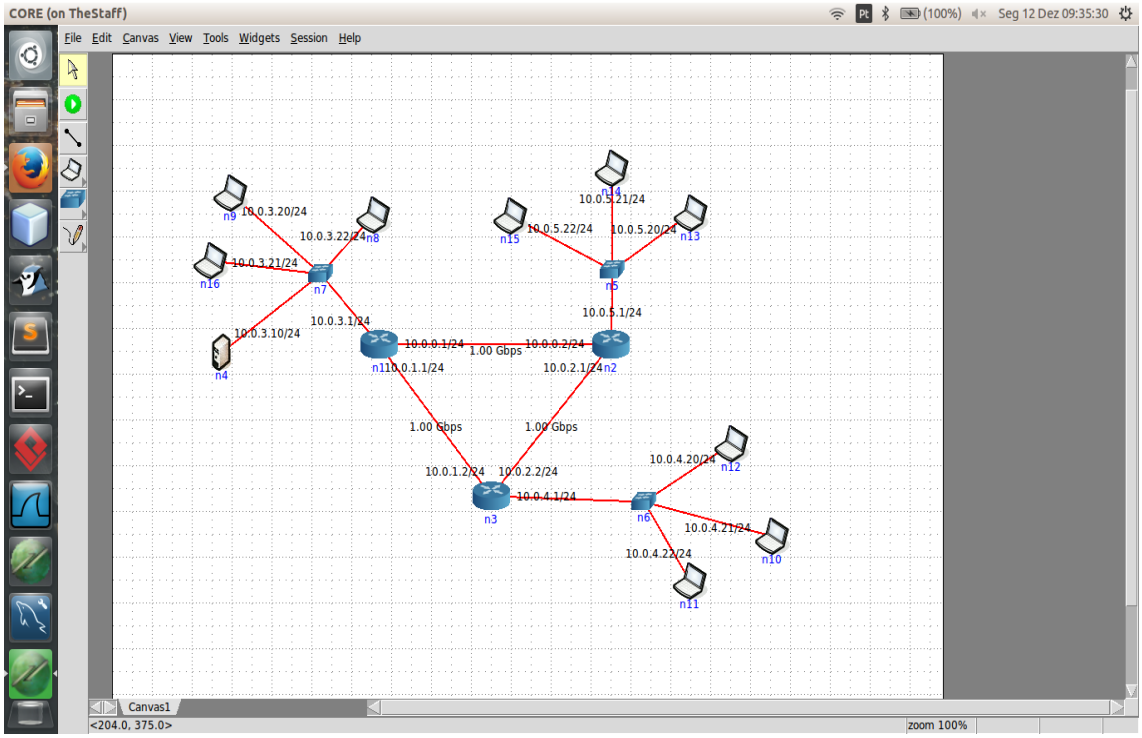
e) Indique, resumindo, os campos que mudam no cabeçalho IP entre os diferentes fragmentos, e explique a forma como essa informação permite reconstruir o datagrama original.

Entre fragmentos do mesmo pacote, os únicos campos que mudam são as Flags e o Offset. Antes de cada datagrama ser fragmentado, é lhe atribuído um identificador, para que todos os fragmentos desse datagrama, possuam no cabeçalho esse mesmo identificador. Assim o equipamento final sabe que fragmentos juntar, de modo a não misturar informação. Como os vários fragmentos podem chegar ao equipamento final de forma desordenada, porque podem por exemplo, fazer percursos diferentes, então esse mesmo equipamento, guarda num buffer os vários fragmentos, e assim através do campo offset de cada, sabe a sua ordem. Através do campo das Flags, ele sabe também quando termina a sequencia dos fragmentos de um determinado pacote que fora fragmentado.

Parte II

1) Atenda aos endereços IP atribuídos automaticamente pelo CORE aos diversos equipamentos da topologia.

a) Indique que endereços IP e máscaras de rede foram atribuídos pelo CORE a cada equipamento. Se preferir, pode incluir uma imagem que ilustre de forma clara a topologia e o endereçamento.



O endereço da rede do Departamento A é 10.0.3.0/24. O endereço da rede do Departamento B é 10.0.5.0/24. E o endereço da rede do Departamento C é 10.0.4.0/24.

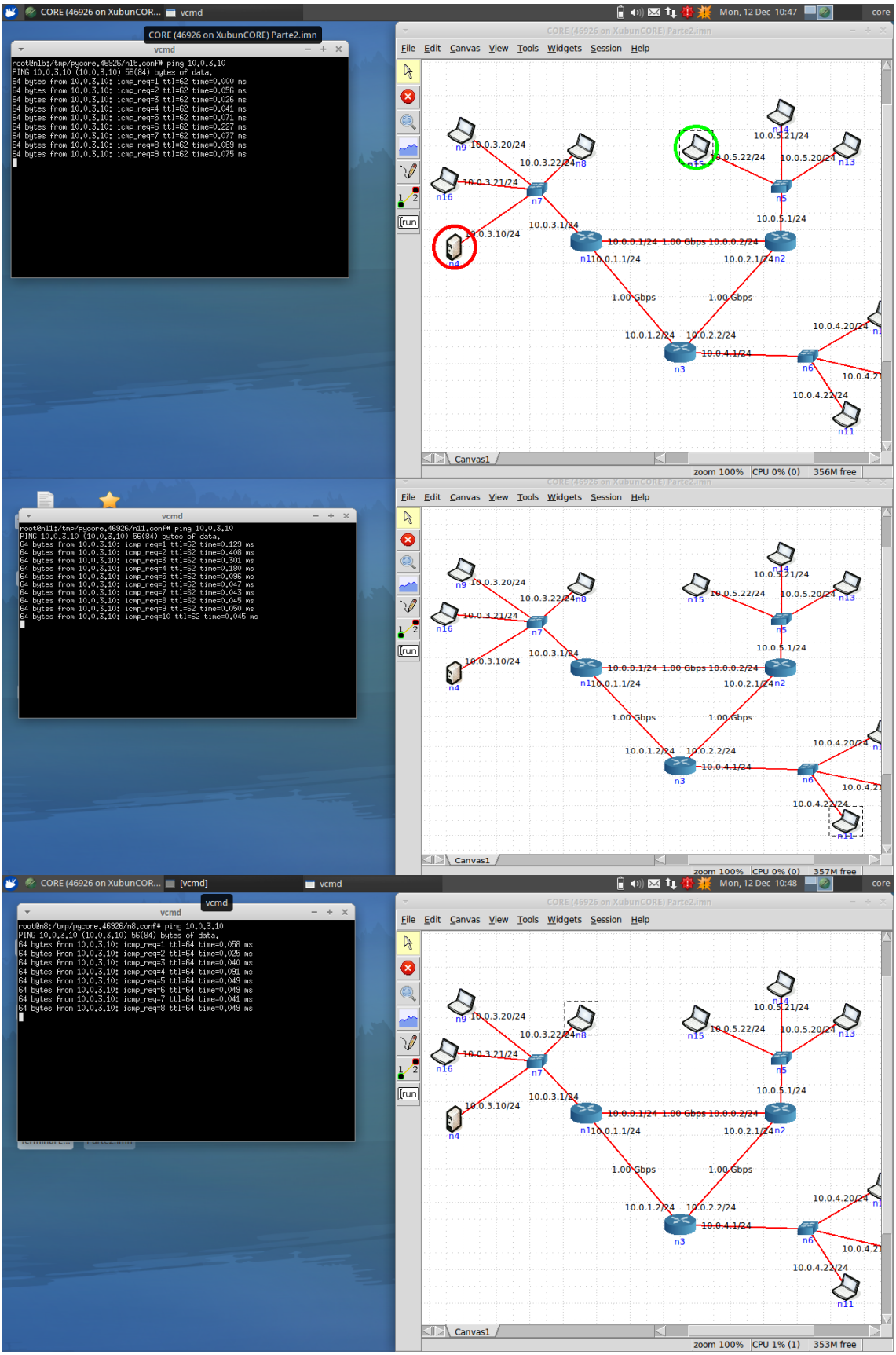
b) Tratam-se de endereços públicos ou privados? Porquê?

São privados, comporta-se como públicos, porque são poucos equipamentos, mas não são privados, para que quando se faça a ligação desta topologia CORE ao Internet, não haja conflitos de endereços IP.

c) Porque razão não é atribuído um endereço IP aos switches?

Não é atribuído endereço IP ao switch porque, este equipamento só opera até a camada 2, ou seja, a camada de ligação logica, logo não lhe é atribuído endereço IP.

d) Usando o comando ping certifique-se que existe conectividade IP entre os laptops dos utilizadores e o servidor do departamento A (basta certificar a conectividade de um laptop por departamento).



2) Para o *router* e um *laptop* do departamento A:

a)Execute o comando netstat –rn por forma a poder consultar a tabela de encaminhamento unicast (IPv4). Inclua no seu relatório as tabelas de encaminhamento obtidas; interprete as várias entradas de cada tabela. Se necessário, consulte o manual respetivo (man netstat).

```
root@n1:/tmp/pycore.48265/n1.conf# netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
10.0.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
10.0.1.0        0.0.0.0         255.255.255.0   U         0 0          0 eth1
10.0.2.0        10.0.0.2        255.255.255.0   UG        0 0          0 eth0
10.0.3.0        0.0.0.0         255.255.255.0   U         0 0          0 eth2
10.0.4.0        10.0.1.2        255.255.255.0   UG        0 0          0 eth1
10.0.5.0        10.0.0.2        255.255.255.0   UG        0 0          0 eth0
```

b) Diga, justificando, se está a ser usado encaminhamento estático ou dinâmico (sugestão: analise que processos estão a correr em cada sistema).

```
root@n1:/tmp/pycore.48287/n1.conf# ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   2304   572 ?        S    10:17   0:00 /usr/sbin/vnoded -v -c /tmp/pycore.48287/n1 -l
root        42  0.0  0.1   3584  1124 ?        Ss   10:17   0:00 /usr/lib/quagga/zebra -u root -g root -d
root        48  0.0  0.1   4072  1384 ?        Ss   10:17   0:00 /usr/lib/quagga/ospfd -u root -g root -d
root        54  0.0  0.1   4000  1360 ?        Ss   10:17   0:00 /usr/lib/quagga/ospf6d -u root -g root -d
root        71  0.0  0.1   5540  1780 pts/5    Ss   10:33   0:00 /bin/bash
root        81  0.0  0.1   4948  1132 pts/5    R+   10:33   0:00 ps aux
```

```
root@n9:/tmp/pycore.48290/n9.conf# ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   2304   568 ?        S    10:47   0:00 /usr/sbin/vnoded -v -c /tmp/pycore.48290/n9 -l
root        16  0.0  0.1   5540  1768 pts/5    Ss   10:48   0:00 /bin/bash
root        27  0.0  0.1   4948  1132 pts/5    R+   10:48   0:00 ps aux
```

```
root@n15:/tmp/pycore.48290/n15.conf# ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   2304   564 ?        S    10:47   0:00 /usr/sbin/vnoded -v -c /tmp/pycore.48290/n15 -l
root        16  0.0  0.1   5540  1772 pts/5    Ss   10:49   0:00 /bin/bash
root        26  0.0  0.1   4948  1132 pts/5    R+   10:49   0:00 ps aux
```

```
root@n12:/tmp/pycore.48290/n12.conf# ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   2304   564 ?        S    10:47   0:00 /usr/sbin/vnoded -v -c /tmp/pycore.48290/n12 -l
root        16  0.0  0.1   5540  1772 pts/5    Ss   10:49   0:00 /bin/bash
root        26  0.0  0.1   4948  1132 pts/5    R+   10:49   0:00 ps aux
```

Entre os routers, está a ser usado encaminhamento dinâmico pois está a usar o protocolo ospf, que é um protocolo dinâmico. Nas sub-redes está a ser usado um encaminhamento estático visto que não está a usar um protocolo dinâmico.

c) Admita que, por questões administrativas, a rota por defeito (0.0.0.0 ou default) deve ser retirada definitivamente da tabela de encaminhamento do servidor localizado no departamento A. Use o comando route delete para o efeito. Que implicações tem esta medida para os utilizadores da empresa que acedem ao servidor. Justifique.

```
root@n4:/tmp/pycore.48287/n4.conf# route delete default
root@n4:/tmp/pycore.48287/n4.conf# netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
10.0.3.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
root@n4:/tmp/pycore.48287/n4.conf#
```

Isto implica que o router não sabia se conectar a redes fora da rede local.

d) Adicione as rotas estáticas necessárias para restaurar a conectividade para o servidor, por forma a contornar a restrição imposta em c). Utilize para o efeito o comando route add e registe os comandos que usou.

```
root@n4:/tmp/pycore.48287/n4.conf# netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
10.0.3.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
<add -net 10.0.4.0 gw 10.0.3.1 netmask 255.255.255.0
<add -net 10.0.5.0 gw 10.0.3.1 netmask 255.255.255.0
root@n4:/tmp/pycore.48287/n4.conf# netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
10.0.3.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
10.0.4.0        10.0.3.1        255.255.255.0   UG        0 0          0 eth0
10.0.5.0        10.0.3.1        255.255.255.0   UG        0 0          0 eth0
```

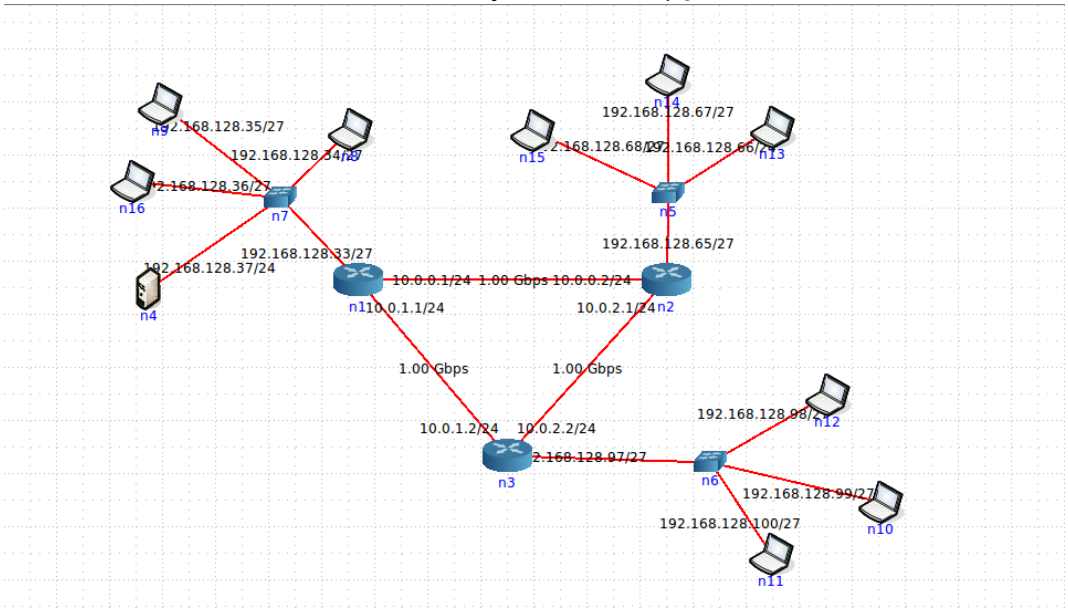
e) Teste a nova política de encaminhamento garantindo que o servidor está novamente acessível, utilizando para o efeito o comando ping. Registe a nova tabela de encaminhamento do servidor.

```
root@n12:/tmp/pycore.48287/n12.conf# ping 10.0.3.10
PING 10.0.3.10 (10.0.3.10) 56(84) bytes of data.
64 bytes from 10.0.3.10: icmp_req=1 ttl=62 time=0.038 ms
64 bytes from 10.0.3.10: icmp_req=2 ttl=62 time=0.037 ms
64 bytes from 10.0.3.10: icmp_req=3 ttl=62 time=0.037 ms
64 bytes from 10.0.3.10: icmp_req=4 ttl=62 time=0.037 ms
```

```
root@n15:/tmp/pycore.48287/n15.conf# ping 10.0.3.10
PING 10.0.3.10 (10.0.3.10) 56(84) bytes of data.
64 bytes from 10.0.3.10: icmp_req=1 ttl=62 time=0.136 ms
64 bytes from 10.0.3.10: icmp_req=2 ttl=62 time=0.000 ms
64 bytes from 10.0.3.10: icmp_req=3 ttl=62 time=0.040 ms
```

3) Definição de Sub-redes

1)Assumindo que dispõe apenas de um único endereço de rede IP classe C 192.168.128.0/24, defina um novo esquema de endereçamento para as redes dos departamentos (mantendo a rede de core inalterada) e atribua endereços às interfaces dos vários sistemas envolvidos. Devejustificar as opções usadas.



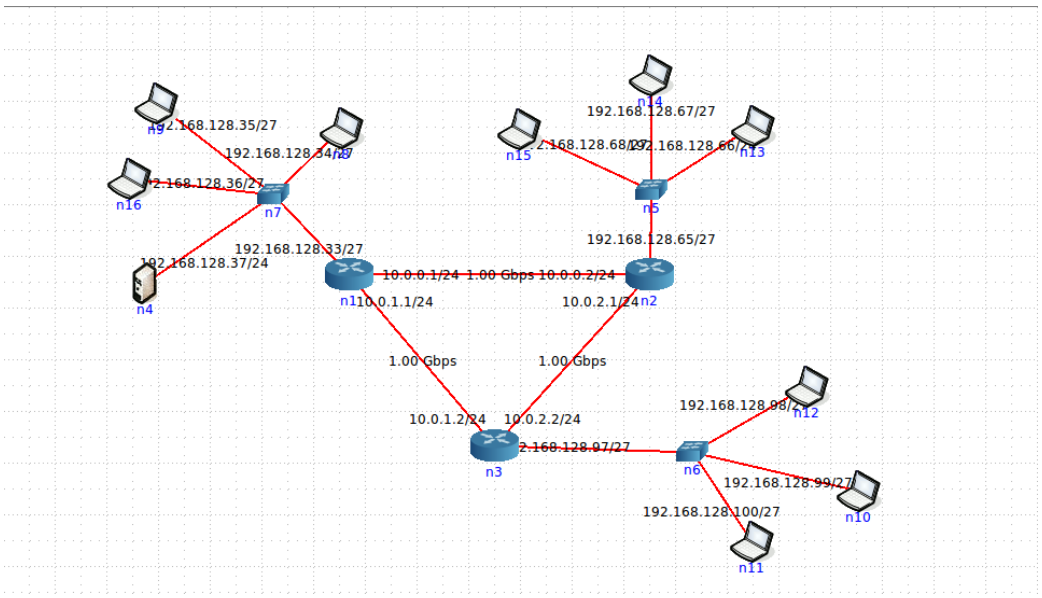
2)Qual a máscara de rede que usou (em formato decimal)? Justifique.

224. 2^7+2^6+2^5=224, porque se usou 3 bits.

3) Quantos hosts IP pode interligar em cada departamento? Justifique.

2^8-224-2=30. Estão disponíveis 32 hosts IP em cada departamento, mas como 2 IP estão reservados (tudo a 1 e a 0), só estão disponíveis 30 hosts IP.

4) Garanta que conectividade IP entre as várias redes locais da empresa MIEInet é mantida.



Nesta rede é garantida a conectividade IP.

Conclusões

Quanto ao protocolo ICMP permitiu descobrir todos os caminhos desde o host até ao destino, através de “saltos”, optando geralmente pelo mais pequeno, levando consigo informações sobre a mensagem a ser transmitida, de maneira eficiente e fiável, visto que envia três tramas de cada vez que tenta chegar ao destino. Dependendo da dimensão da rede o ICMP envia a mensagem com o campo TTL (“saltos”) maior, de maneira a conseguir percorrer a rede toda, para conseguir chegar ao destino.

Concluindo a primeira parte, quanto à fragmentação, acompanhamos o processo da divisão dos datagramas em sub-datagramas, de forma eficiente, e aprendemos o modo como os cabeçalhos são gerados, de forma que, os equipamentos que vão receber esse conjunto de pacotes, consigam fazer a organização e ordenação de modo a obter a mensagem original.

Já em relação a topologia CORE que nos foi estipulada, conseguimos perceber o modo de como o programa gera automaticamente, os endereços IP, de modo que à posteriori, quando existir uma ligação Ethernet à maquina virtual, e consequentemente a essa mesma topologia, não exista conflitos de endereços IP, com outras maquinas ligadas á rede.

Podemos também verificar que todas as sub-redes têm uma tabela de encaminhamento que lhe permite armazenar todos os caminhos de maneira a poder chegar a qualquer lado da rede. Essa tabela de encaminhamento é atualizada automaticamente se for utilizada encaminhamento dinâmico, ou é fixa se for utilizado encaminhamento estático. Esta tabela contêm o endereço IP de todos os hosts da rede, usando uma netmask, de maneira a aumentar o número de redes ou hosts, dependendo da necessidade da rede. Esta netmask identifica os bits usados para identificar a rede e os bits usados para identificar o host. Esta netmask pode ser alterada de maneira a satisfazer o aumento ou diminuição de hosts na rede, aumentando a eficiência. Esta tabela de encaminhamento tem 2 endereços IP reservados, os endereços IP com os bits do host tudo a 1 e a 0. Tudo a 0 está reservado para representar o endereço IP default dessa rede, o IP com os bits todos a 1 está reservado para representar a broadcast.

Podemos desta forma concluir que foi possível, por um lado, entender este protocolo melhor, tal como o processo de divisão de datagramas em sub-datagrams, e aprofundamos o conhecimento no protocolo ICMP. Por outro lado, melhoramos a nossa habilidade de utilizar o CORE e outras ferramentas úteis nesta área de Redes de Computadores.