Redes de Computadores
Trabalho Prático Nº2
Protocolo IP

Alexandra Candeias, Pedro Araújo, and Tiago Ribeiro

University of Minho, Department of Informatics, 4710-057 Braga, Portugal
e-mail: {a89521,a70699,a76420}@alunos.uminho.pt

Introdução

A elaboração do presente relatório tem o intuito de responder às questões colocadas em ambas as partes do *Trabalho Prático 2* (TP2), cujo o foco é o estudo do Internet Protocol (*IP*). As questões realizadas tem como objectivo primordial o estudo das principais vertentes deste protocolo, nomeadamente: *Datagramas, Fragmentação de Pacotes, Endereçamento e Encaminhamento*.

Assim sendo, optou-se por dividir o relatório em três secções principais. Nas secções 1 e 2 são expostas as questões presentes no TP2 e apresentadas respostas às mesmas. Na secção 3, são apresentadas as conclusões.

1 Parte I - *IPv4*: Datagramas e Fragmentação

Esta secção tem o intuito de responder às questões colocadas na *Parte I* do TP2. O foco desta primeira parte passou pelo registo de datagramas, tanto enviados como recebidos, através da execução do comando *Traceroute*, assim como após a análise dos mesmos, um processo detalhado de fragmentação. Todo este processo foi realizado utilizando o programa *CORE*.

Questões e Respostas

Q1. Nesta primeira questão é pedido que seja criada uma topologia *CORE* de modo a que se possa verificar o comportamento do *Traceroute*. Como pode ser observado na Figura 1, esta topologia é composta por dois *Hosts* (Cliente 1 e Servidor 1) e dois *Routers* (R2 e R3).

a. Active o *WireShark* ou o *TCPdump* no Cliente1. Numa *shell* do Cliente1, execute o comando *Traceroute -I* para o endereço *IP* do Servidor1.

R:

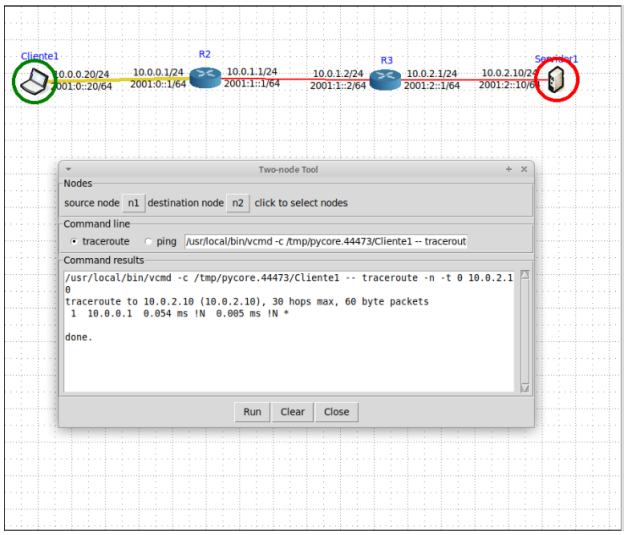

Fig. 1. Topologia *CORE*

b. Registe e analise o tráfego *ICMP* enviado pelo Cliente1 e o tráfego *ICMP* recebido como resposta. Comente os resultados face ao comportamento esperado.

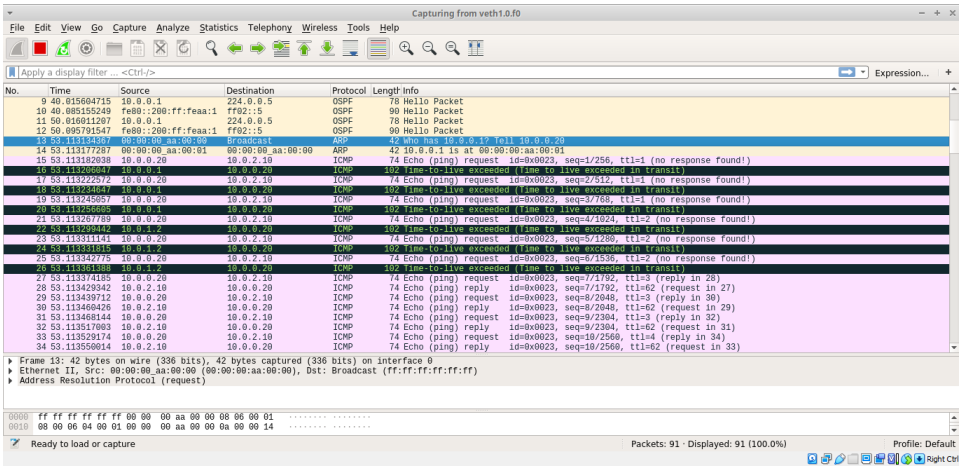

Fig. 2. Tráfego

R: São enviados três datagramas *ICMP* inicialmente com o *TTL*=1 e os seguintes com o *TTL*=2, 3, . . . , 6. Verifica-se assim que é realizado o esperado, isto é, são enviadas uma ou mais tramas *ICMP* com o *TTL* (de modo crescente) até os datagramas chegarem ao destino final.

c. Qual deve ser o valor inicial mínimo do campo *TTL* para alcançar o Servidor1?

R: Como demonstrado na Figura 2, em concreto na linha 27 e 28, o valor mínimo de *TTL* de modo a chegar ao destino é igual 3.

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

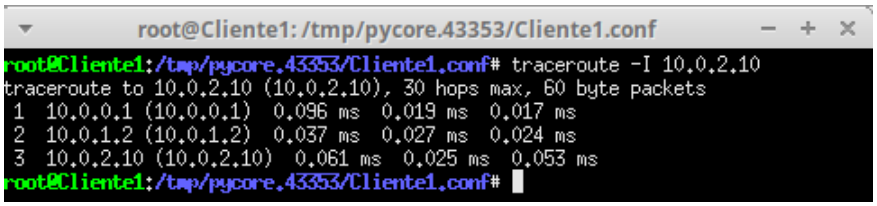

Fig. 3. *Traceroute*

R: A média do RTT (Round-Trip Time), é obtida usando o *Traceroute*, sendo a mesma calculada através dos valores fornecidos pelo último nodo. Assim, e pela 3 onde esta indica que o último nodo é o terceiro, a média pode ser calculada da seguinte forma: (0.061 + 0.025 + 0.053) / 3 = 0.046 ms

Q2. Nesta questão é pedido que seja feita a utilização da máquina nativa de modo a usar o *Traceroute* para gerar datagramas de *IP* de diferentes tamanhos. De seguida é utilizada a ferramenta *WireShark* de modo a que o tráfego gerado pelo *Traceroute* possa ser capturado para o tamanho por defeito assim como para o tamanho de 3271bytes.

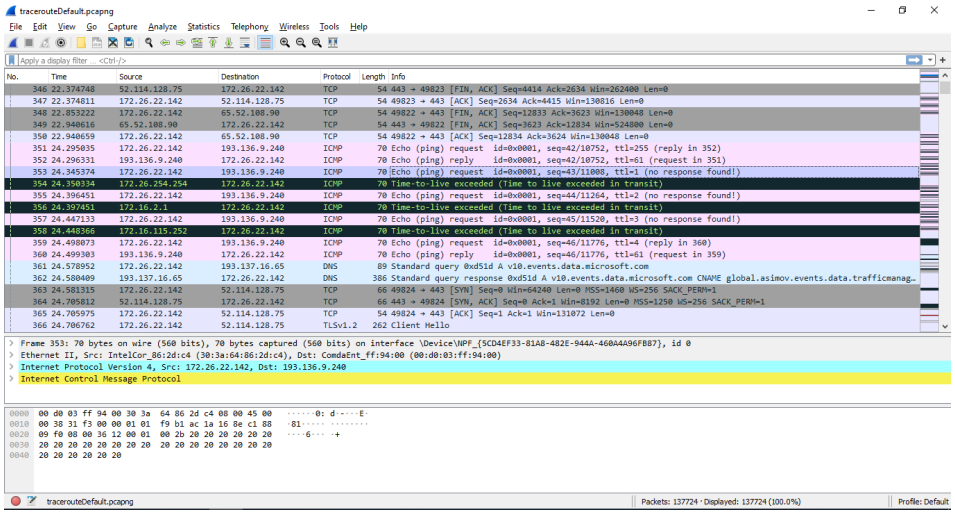

Fig. 4. Captura realizada pelo *WireShark* na máquina nativa

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

R: Como pode ser observado na Figura 4, o endereço da interface ativa da máquina em questão é o 172.26.22.142.

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

R: O valor do campo do protocolo é presentado por *ICMP* (1), sendo possível visualizar na trama esse mesmo valor.

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*?

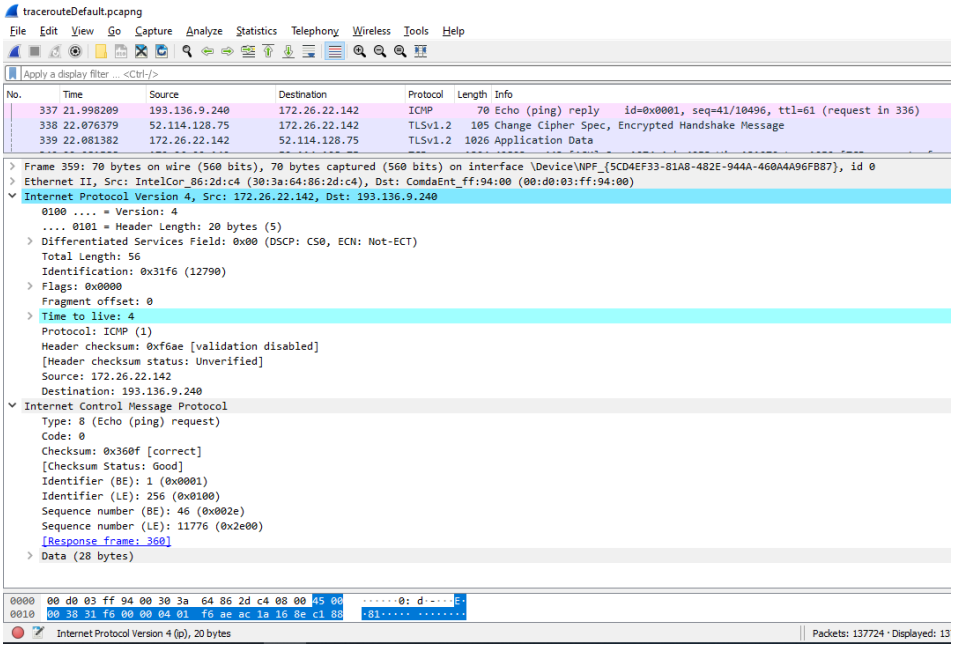

Fig. 5. Valores do *IPv4*

R: Na Figura 5, pode-se observar que o cabeçalho de *IP* tem 56 Bytes sendo que destes, 20 Bytes são cabeçalho e os restantes 36 Bytes de dados para a mensagem *ICMP*. Da mensagem *ICMP*, 28 Bytes são dados e 8 Bytes cabeçalho da mesma. O tamanho do *Payload* é calculado subtraindo o tamanho total dos dados da mensagem *IP* (56 Bytes) ao cabeçalho da mesma (20 Bytes) e ainda o cabeçalho da mensagem *ICMP* (8 Bytes). O valor do *Payload* = 56 - 20 - 8 = 28 Bytes.

d. O datagrama *IP* foi fragmentado? Justifique.

R: Não, uma vez que datagrama é pequeno, este possui poucos Bytes, apenas é necessária uma trama *ICMP* para realizar a operação.

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 interface da sua máquina. Para a sequência de mensagens *ICMP* enviadas pelo seu computador, indique que campos do cabeçalho *IP* variam de pacote para pacote.

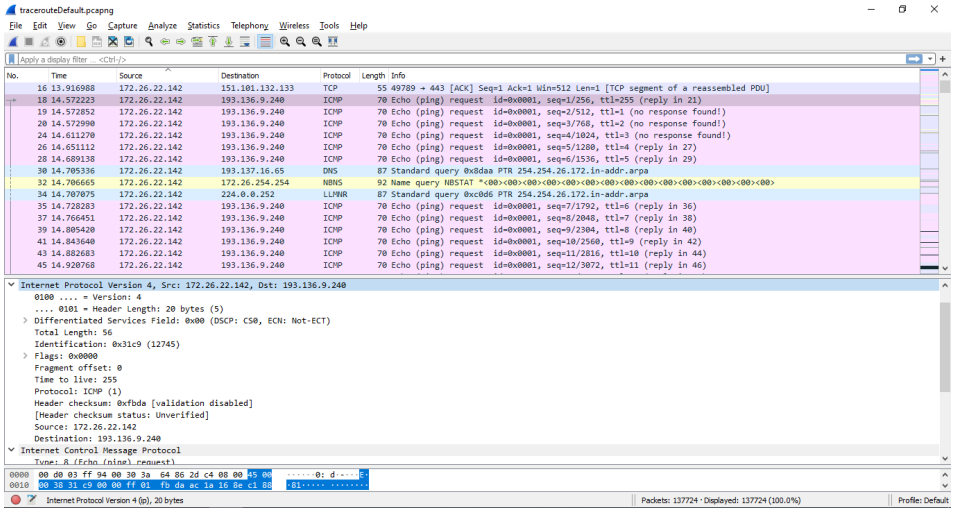

Fig. 6. Pacotes capturados ordenados

R: A cada envio de um *Payload*, apenas os campos *Identification*, *Time to Live* e *Header Checksum* é que variam ao longo da sequência de tramas.

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

R: A identificação das tramas são sempre um incremento da identificação da trama anterior. O *TTL* também é incrementado. No entanto quando é pedida uma nova mensagem *ICMP*, este *TTL* volta a ter o valor 1 enquanto que a identificação se mantém.

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?

R: O valor do campo *TTL* é de 255 sendo que este valor diminui ao longo das respostas *ICMP TTL Exceeded* porque como a mensagem que o *Router* obteve tinha o *TTL* baixo e não alcançou o seu destino, o *Router* como não sabe de onde e quão longe está da origem desta mensagem, é necessário inserir um *TTL* alto para que consiga definitivamente chegar ao destino.

Q3. Após a resolução da questão anterior é pedido que seja feita uma nova análise, agora com o foco na fragmentação de pacotes *IP*. Essa análise começa pela reposição da ordem do tráfego capturado com recurso à coluna que apresenta o tempo. Dá-se especial foco ao tráfego depois do tamanho de pacote ter sido definido para 3271 Bytes.

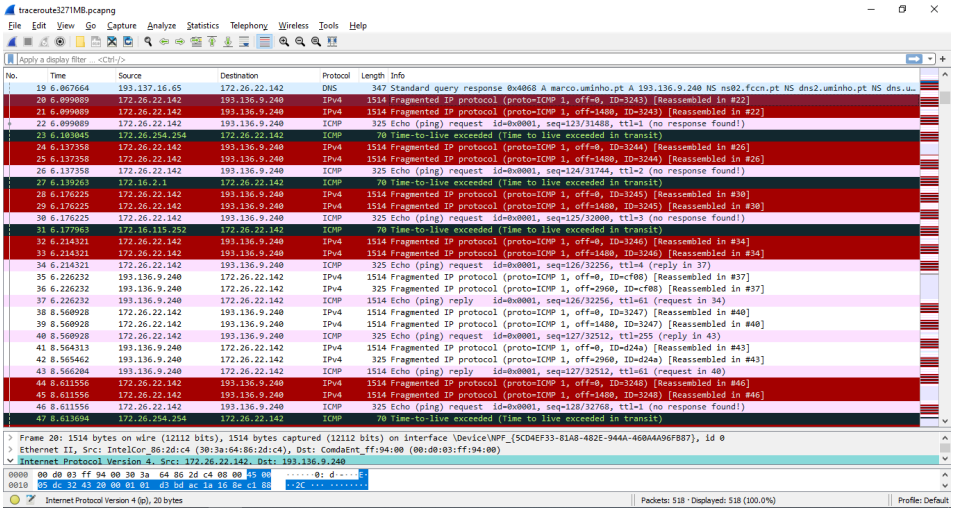

Fig. 7. Tráfego ordenado de forma temporal

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

R: Uma vez que a mensagem de dados a enviar tem elevado comprimento, foi necessário fragmentar a trama em secções mais pequenas de modo a que a transmissão fosse bem sucedida.

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*?

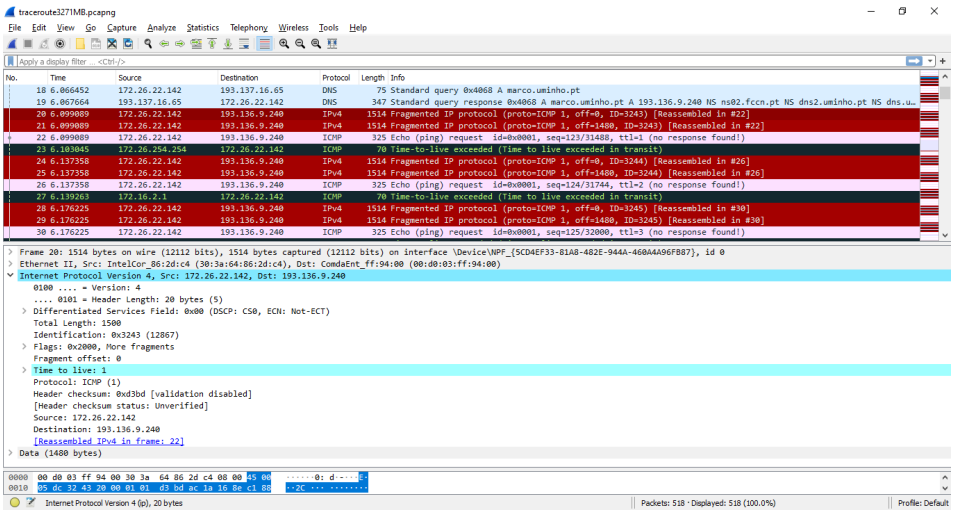

Fig. 8. Primeiro fragmento

R: No primeiro fragmento, o campo *Fragment Offset* e o campo das *Flags*, indicam que o datagrama foi fragmentado. O campo *Fragment Offset* é responsável por indicar em qual dos fragmentos o utilizador se encontra, uma vez que este se encontra igual a 0, isto

indica que este é o primeiro fragmento. O tamanho deste datagrama fragmentado será igual a 1500 Bytes. Isto é visível nas Figuras 8 e 9 na linha Total Length.

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?

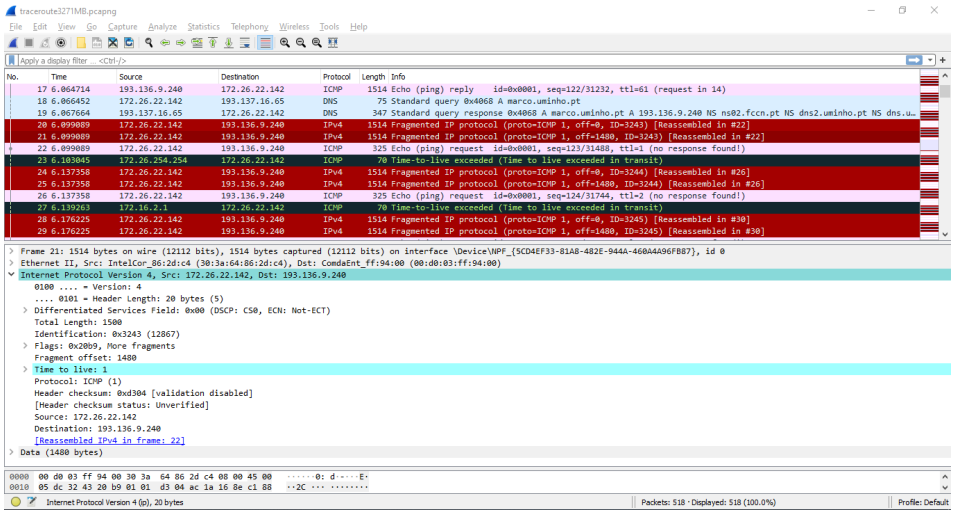

Fig. 9. Segundo fragmento

R: Tal como explicado na alínea anterior o Fragment Offset é responsável por indicar me que fragmento o utilizador se encontra. Como podemos ver na Figura 9 o valor deste é diferente de 0, logo isto indica que não é o primeiro fragmento. O campo das Flags indica-nos se há ou não mais fragmentos a transmitir.

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

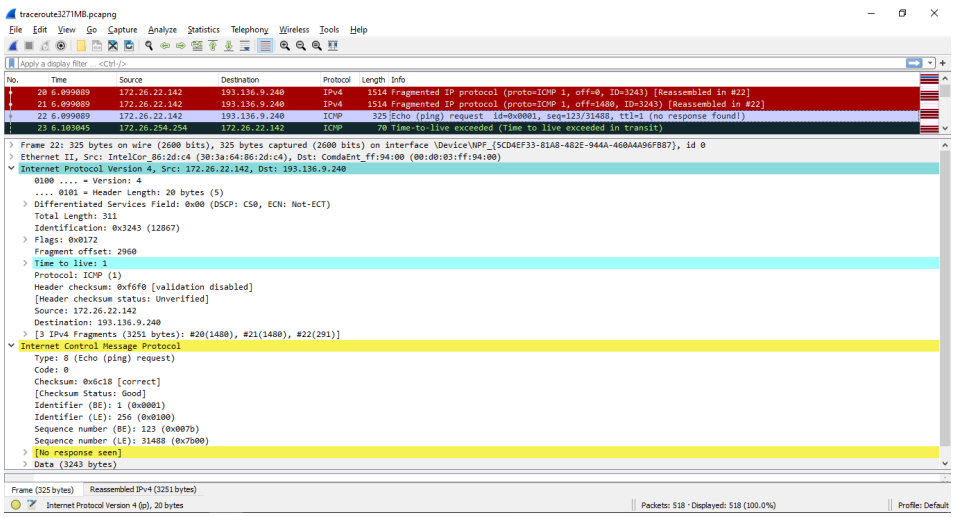

Fig. 10. Número de fragmentos

R: Como pode ser observado na Figura 10, foram criados três fragmentos. O modo como o último fragmento é detectado, foi explicado na alínea anterior, passa por observar o que o campo das Flags apresenta. Deste modo conseguimos perceber que este é o último 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.

R: Como sabemos os campos Flags e o Fragment Offset são responsáveis por essas mudanças entre diferentes fragmentos. Através do campo Fragment Offset é possível reconstruir o datagrama original uma vez que este apresenta a informação da ordem dos fragmentos e em que offset é que foram fragmentados.

2 Parte II - Endereçamento e Encaminhamento

Tal como na secção 1, o intuito é responder às questões colocadas a Parte II do TP2. O foco desta segunda parte é no endereçamento e encaminhamento.

2.1 Questões e Respostas

Q1. Nesta primeira questão é novamente pedido que seja criada uma topologia *CORE*, como pode ser observado na Figura 11, esta topologia é composta por quatro departamentos (A, B, C e D) onde cada departamento possui um *Router* de acesso à sua rede local. Estes *Router*s de acesso (RA, RB, RC e RD) estão interligados entre si formando um anel. Por sua vez, existe um servidor (S1) na rede do departamento A e dois laptops por departamento, interligados ao *Router* respetivo através de um switch. A conectividade *IP* externa da organização é assegurada através de um *Router* de acesso RISP conectado a RA.

a. Indique que endereços *IP* e máscaras de rede foram atribuídos pelo *CORE* a cada equipamento. Para simplificar, pode incluir uma imagem que ilustre de forma clara a topologia definida e o endereçamento usado.

R:

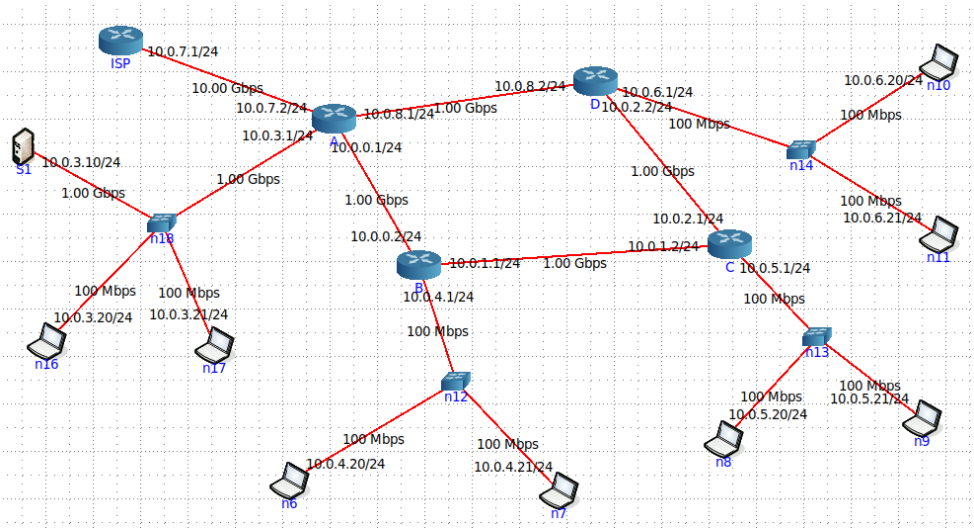

Fig. 11. Topologia *CORE* para o caso de estudo

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

R: Tratam-se de endereços privados, uma vez que a máscara sendo /24 apenas é possível endereçar as redes para poucos *Host*s, isto é, são denominados como endereços privados locais.

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

R: Os switches não tem endereço *IP* uma vez que a sua função é a comutação de ligações. Uma vez que nunca existirá um pacote endereçado para o switch, este não necessita de um endereço de *IP*.

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

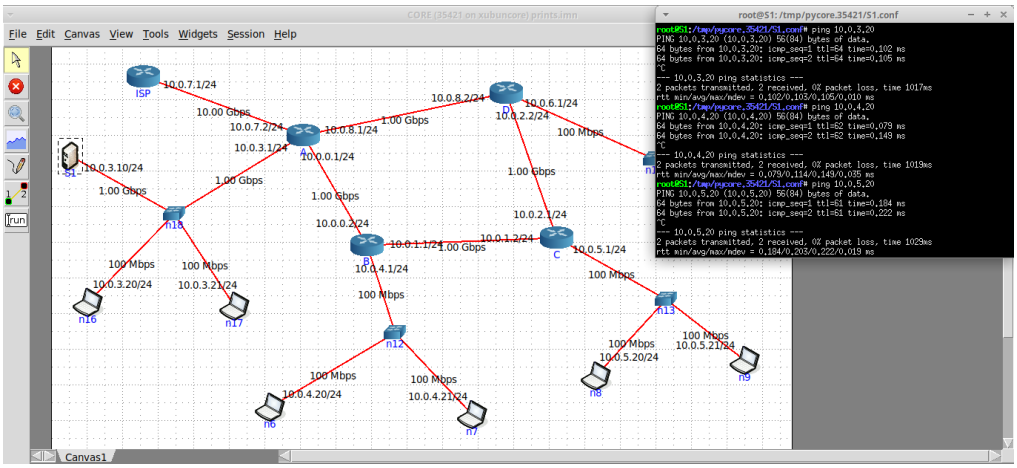

Fig. 12. Conetividade apartir do S1

Q2. Esta questão foca a sua atenção para o *Router* e um laptop do departamento C.

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.

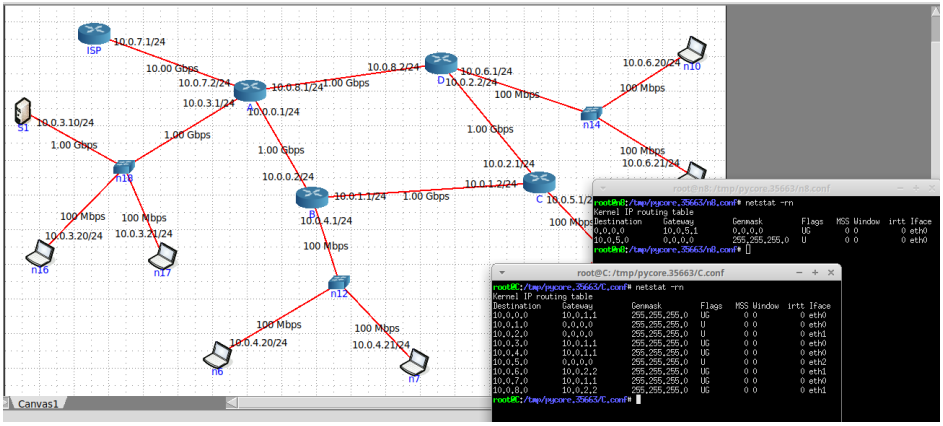

Fig. 13. Tabela de encaminhamento

R: Como observável na Figura 13 a tabela de encaminhamento do *Router C* tem presente todas as redes que se encontram na topologia. Constate-se ainda que quando uma rede está directamente ligada ao *Router* C, o identificador Gateway será igual a 0.0.0.0, isto é, o próximo nodo indica-se a si mesmo. Assim apenas necessita-se de referir para que interface o *Router C* deve enviar o pacote. Recorrendo de novo à Figura 13, o *IP* da interface:

- eth0 corresponderá ao 10.0.1.2/24; - eth1 corresponderá ao 10.0.2.1/24: - eth2 corresponderá ao 10.0.5.1/24.

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

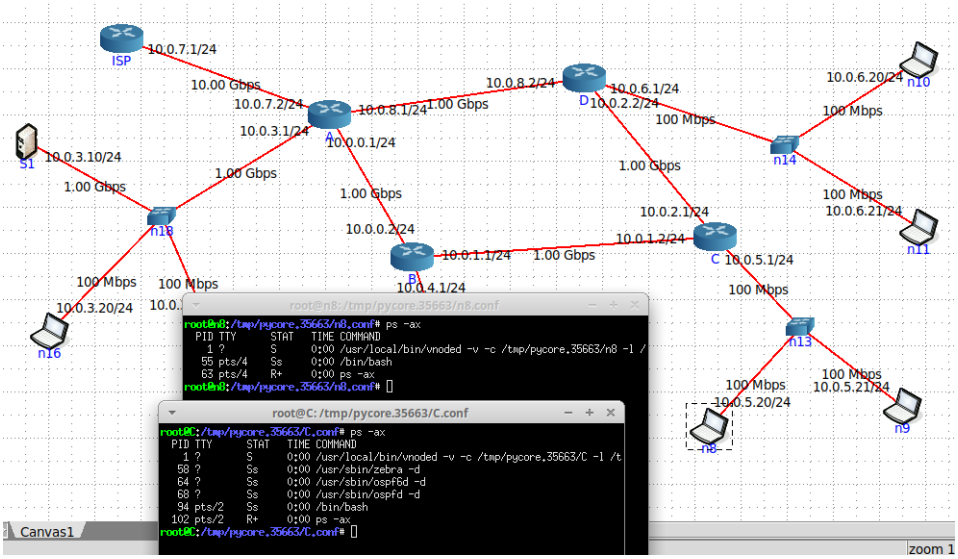

Fig. 14. Resultados do uso do comando ps -ax

R: Está a ser usado o encaminhamento dinâmico, isto é observável na Figura 13 uma vez que o *Router C* está a usar o protocolo Open Shortest Path First (OSPF). Este protocolo anuncia automaticamente as rotas a que o *Router C* tem acesso. Recorrendo à Figura 14 observamos que "n8" não tem nenhum processo, provando que está a usar este protocolo.

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 S 1 localizado no departamento A. Use o comando route delete para o efeito. Que implicações tem esta medida para os utilizadores da organização MIEI-RC que acedem ao servidor. Justifique.

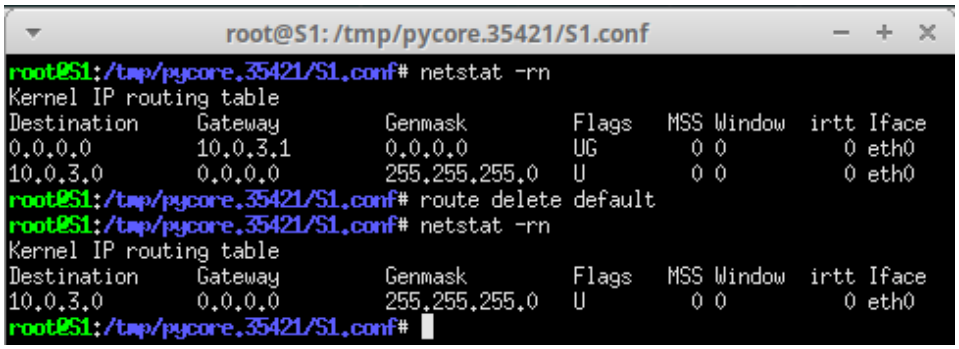

Fig. 15. Aplicação do comando *route delete*

R: Uma vez eliminada a linha default da tabela de encaminhamento do servidor S1, todos os utilizadores do servidor fora do departamento A ficam sem o acesso ao mesmo. Ora aquando da chega um datagrama ao servidor, este não sabe para onde o tem de o encaminhar, logo o pacote é perdido. No caso de ser um *Host* dentro do departamento A, como o servidor ainda possui uma linha de encaminhamento para a rede deste departamento, o pacote é encaminhado de acordo com a linha em questão.

d. Adicione as rotas estáticas necessárias para restaurar a conectividade para o servidor S1, por forma a contornar a restrição imposta na alínea c). Utilize para o efeito o comando route add e registe os comandos que usou.

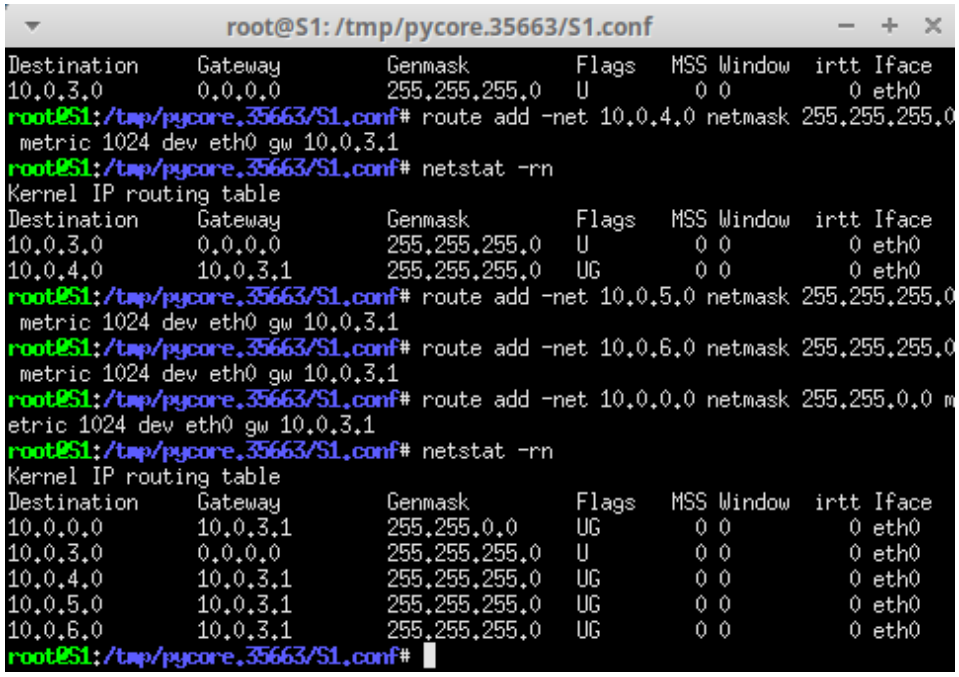

Fig. 16. Adição de rotas estáticas

R: Foram adicionadas toas as rotas estáticas das redes da organização MIEI-RC, usando os comandos demonstrados na Figura 15.

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.

R: Todos os *Host*s das suas respectivas redes tem agora conectividade ao servidor S1 e a tabela de encaminhamento resultante encontra-se na Figura 15. Optou-se pela utilização o endereço *IP* 10.0.0.0 e da máscara 255.255.0.0 (/16) de modo a funcionar como rede default, isto é, qualquer datagrama com um endereço *IP* desconhecido irá dar ao escolhido.

Q3. Nesta questão, e utilizando a mesma topologia *CORE* que foi utilizada até agora, pretende-se criar uma definição de sub-rede.

1) Considere que dispõe apenas do endereço de rede *IP* 130.71.96.0/19. Defina um novo esquema de endereçamento para as redes dos departamentos (mantendo a rede de acesso e *CORE* inalterados) e atribua endereços às interfaces dos vários sistemas envolvidos. Assuma que todos os endereços de sub-redes são usáveis. Deve justificar as opções usadas.

R: Para 4 redes, 2 bits serão suficientes para endereçar as redes. Isto é, teremos as seguintes redes para endereçamento:

130.71.96.0/19
10000010 01000111 011|00000 00000000

Usando 2 bits:

130.71.011|XX|000.0

00 A
01 B
10 C
11 D

	IP	IP-Inicial	IP-Final
A	130.71.96.0/21	130.71.96.0	130.71.103.255
B	130.71.104.0/21	130.71.104.0	130.71.111.255
C	130.71.112.0/21	130.71.112.0	130.71.119.255
D	130.71.120.0/21	130.71.120.0	130.71.127.255

A	IP	B		C		D	
PC 1	130.71.96.3	PC 1	130.71.104.2	PC 1	130.71.112.2	PC 1	130.71.120.2
PC 2	130.71.96.4	PC 2	130.71.104.3	PC 2	130.71.112.3	PC 2	130.71.120.3
SA	130.71.96.2	RB	130.71.104.1	RA	130.71.112.1	RD	130.71.120.1
RA	130.71.96.1						

Fig. 17. Sub-rede

2) Qual a máscara de rede que usou (em formato decimal)? Quantos *Host*s *IP* pode interligar em cada departamento? Justifique.

R: A máscara usada foi a 255.255.248.0 (/21). Logo, em cada departamento poderemos usar: 130.71.(01100000).0 até 130.71.(01100111).255

Isto é, existem 11 bits possíveis para endereçamento, logo: 2^{11} = 2048 endereços *IP*s disponíveis para endereçamento. Retira-se o endereço de rede e o de Broadcast, ficando assim 2046 endereços *IP*s disponíveis para endereçar *Hosts*.

3) Garanta e verifique que conectividade *IP* entre as várias redes locais da organização MIEI-RC é mantida. Explique como procedeu.

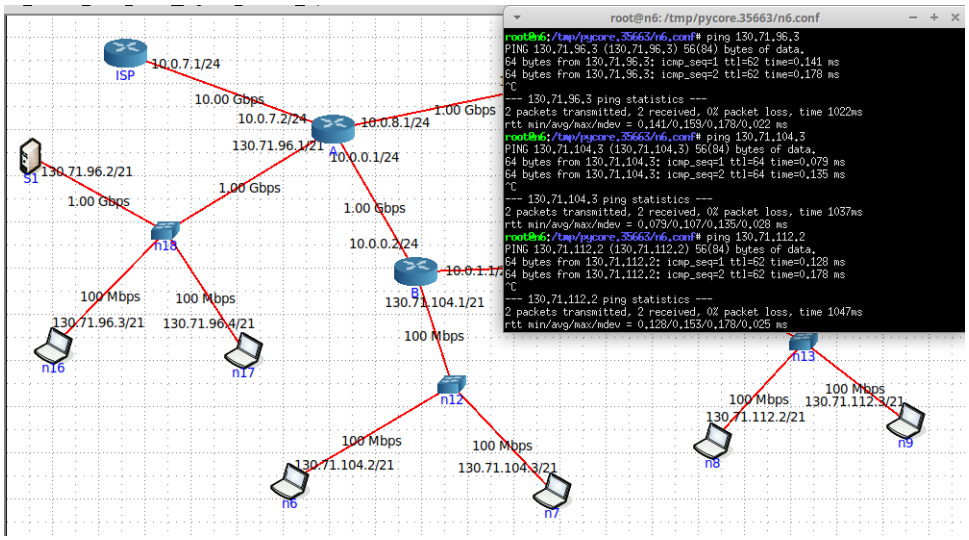

Fig. 18. Verificação de conectividade na sub-rede criada

R: Como é possível verificar na Figura 18, a ligação à organização MIEI-RC manteve-se. Pegando, por exemplo, no *Host* n6 (130.71.104.2), e correndo o comando ping para as restantes redes, incluindo a rede local, verifica-se que este tem resposta ao mesmo em todas as redes.

3 Conclusões

Neste trabalho prático foi possível aprofundar os conhecimentos adquiridos nas aulas teóricas sobre *Internet Protocol*, em duas etapas de projeto. O foco da primeira parte passou pela análise do protocolo *IPv4* utilizando uma topologia *CORE*, onde foi possível estudar o seu comportamento e o tráfego IMCP recebido. Alguns dos casos particulares deste mesmo protocolo também foram abordados, nomeadamente a fragmentação de pacotes. A segunda etapa deu ênfase ao endereçamento e encaminhamento *IP*, através da inclusão de um estudo do funcionamento e impacto de diferentes encaminhamentos, assim como a manipulação de endereços *IP* para efeitos de sub-redes.

References