

Universidade do Minho

Departamento de Informática

TP2 - Protocolo IP (1ª parte)

Redes de Computadores

Grupo 37

Catarina Pais Vieira (a89524)

José Duarte Pereira de Castro Alves (a89563)

Leonardo de Freitas Marreiros (a89537)

**Conteúdo**

[**Parte 1** 4](#_Toc57221619)

[**1.** **Exercício 1** 4](#_Toc57221620)

[**1.1** **Alínea a** 4](#_Toc57221621)

[**1.2** **Alínea b** 5](#_Toc57221622)

[**1.3** **Alínea c** 5](#_Toc57221623)

[**1.4** **Alínea d** 5](#_Toc57221624)

[**2.** **Exercício 2** 6](#_Toc57221625)

[**2.1** **Alínea a** 6](#_Toc57221626)

[**2.2** **Alínea b** 6](#_Toc57221627)

[**2.3** **Alínea c** 6](#_Toc57221628)

[**2.4** **Alínea d** 7](#_Toc57221629)

[**2.5** **Alínea e** 7](#_Toc57221630)

[**2.6** **Alínea f** 7](#_Toc57221631)

[**2.7** **Alínea g** 8](#_Toc57221632)

[**3.** **Exercício 3** 8](#_Toc57221633)

[**3.1** **Alínea a** 9](#_Toc57221634)

[**3.2** **Alínea b** 9](#_Toc57221635)

[**3.3** **Alínea c** 10](#_Toc57221636)

[**3.4** **Alínea d** 10](#_Toc57221637)

[**3.5** **Alínea e** 11](#_Toc57221638)

[**Parte 2** 12](#_Toc57221639)

[**1.** **Exercício 1** 12](#_Toc57221640)

[**1.1** **Alínea a** 12](#_Toc57221641)

[**1.2** **Alínea b** 12](#_Toc57221642)

[**1.3** **Alínea c** 12](#_Toc57221643)

[**1.4** **Alínea d** 13](#_Toc57221644)

[**1.5** **Alínea e** 13](#_Toc57221645)

[**2.** **Exercício 2** 14](#_Toc57221646)

[**2.1** **Alínea a** 14](#_Toc57221647)

[**2.2** **Alínea b** 15](#_Toc57221648)

[**2.3** **Alínea c** 16](#_Toc57221649)

[**2.4** **Alínea d** 16](#_Toc57221650)

[**2.5** **Alínea e** 17](#_Toc57221651)

[**3.** **Exercício 3** 18](#_Toc57221652)

[**3.1** **Alínea a** 18](#_Toc57221653)

[**3.2** **Alínea b** 19](#_Toc57221654)

[**3.3** **Alínea c** 19](#_Toc57221655)

[**Conclusão** 22](#_Toc57221656)

# **Parte 1**

## **Exercício 1**

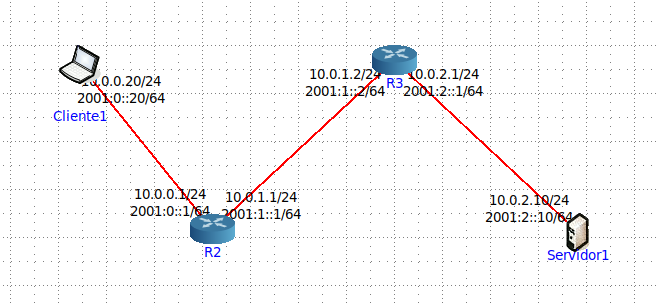


Figura 1- Topologia CORE

### **Alínea a**

Active o wireshark ou o tcpdump no Cliente1. Numa shell do Cliente1, execute o comando traceroute -I para o endereço IP do Servidor1.

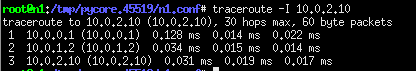


Figura 2- Traceroute

### **Alínea b**

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

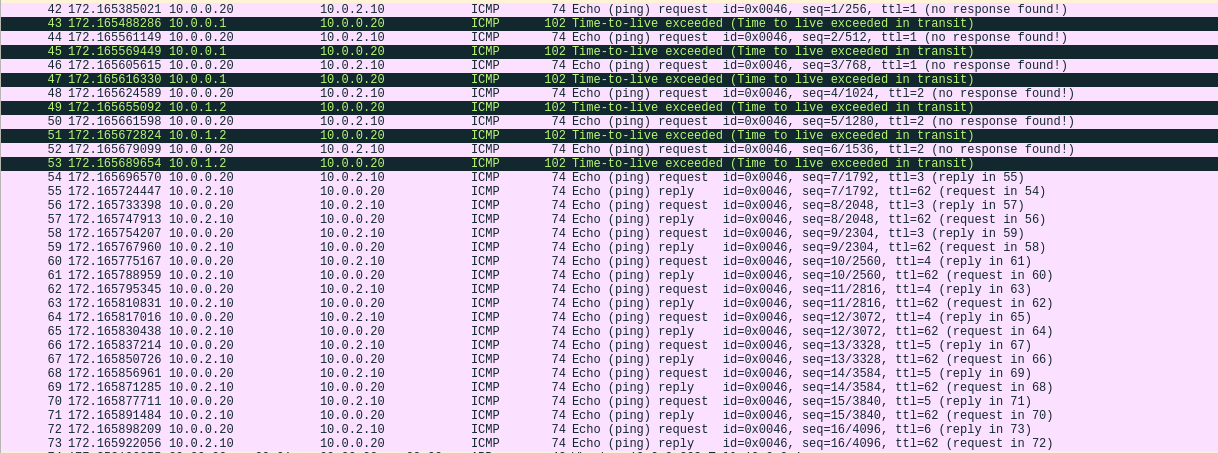


Figura 3- Wireshark

Em relação aos resultados obtidos é importante referir os seguintes aspetos: primeiramente, os pacotes enviados com TTL=1 foram rejeitados pelo router r2; da mesma forma, os pacotes enviados para o router r3 com TTL=2 foram descartados. Nesta fase, foi recebido um pacote “Time to live exceeded” como resposta. Finalmente, os pacotes enviados com TTL=3 chegaram ao seu destino, Servidor1, e foi obtida como resposta pacotes “Echo (ping) reply”.

### **Alínea c**

Qual deve ser o valor inicial mínimo do campo TTL para alcançar o Servidor1? Verifique na prática que a sua resposta está correta.

Tal como foi referido na questão anterior, TTL=3.

### **Alínea d**

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

RTT =2\*((0.128 + 0.014 + 0.022)/3 + (0.034 + 0.015 + 0.014)/3 + (0.031 + 0.019 + 0.017)/3)ms

= 0.196 ms.

## **Exercício 2**

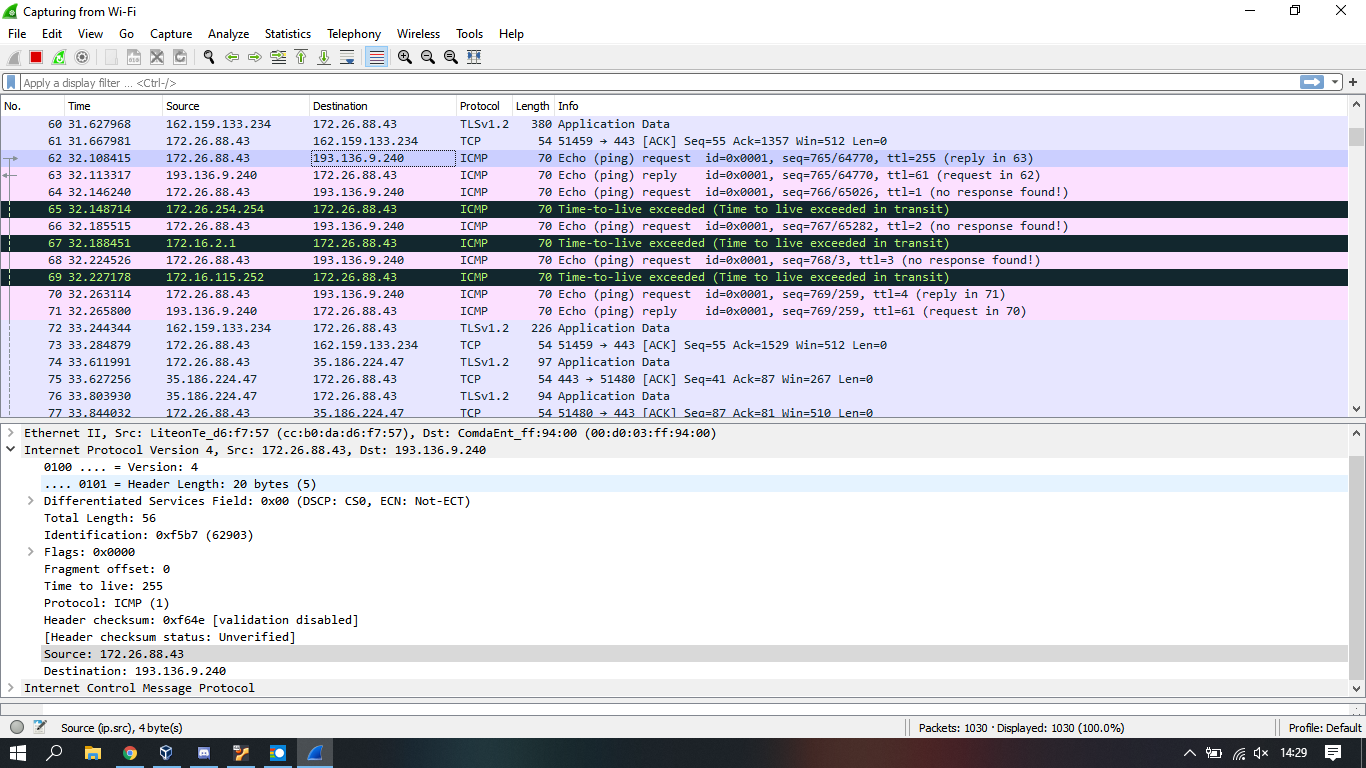


Figura 4- Cabeçalho IP

### **Alínea a**

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

172.26.88.43

### **Alínea b**

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

ICMP (1). Identifica Internet Protocol.

### **Alínea 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 playload calcula-se subtraindo o total length do pacote pelo header length (56-20) que resulta um tamanho de 36 bytes.

### **Alínea d**

O datagrama IP foi fragmentado? Justifique.

Ao analisarmos a figura 4 podemos verificar que o campo “Fragmet offset” está com o valor 0. Além disso, a Flag “More fragments” estava definida como “Not Set” o que significa que não existem mais fragmentos. Logo, concluímos que o datagrama não foi fragmentado.

### **Alínea 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.

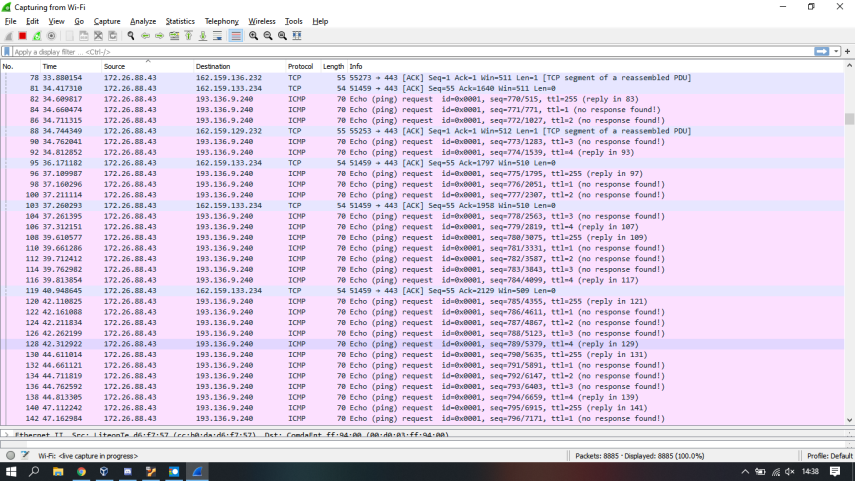


Figura 5- Pacotes capturados no Wireshark ordenados pelo endereço IP fonte

Os campos do cabeçalho IP que variam de pacote para pacote são: TTL, Header checksum e Identification.

### **Alínea f**

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

Identificação do datagrama IP: os primeiros 8 bits são iguais e os restantes aumentam sequencialmente.

TTL: aumenta sequencialmente.

### **Alínea 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?

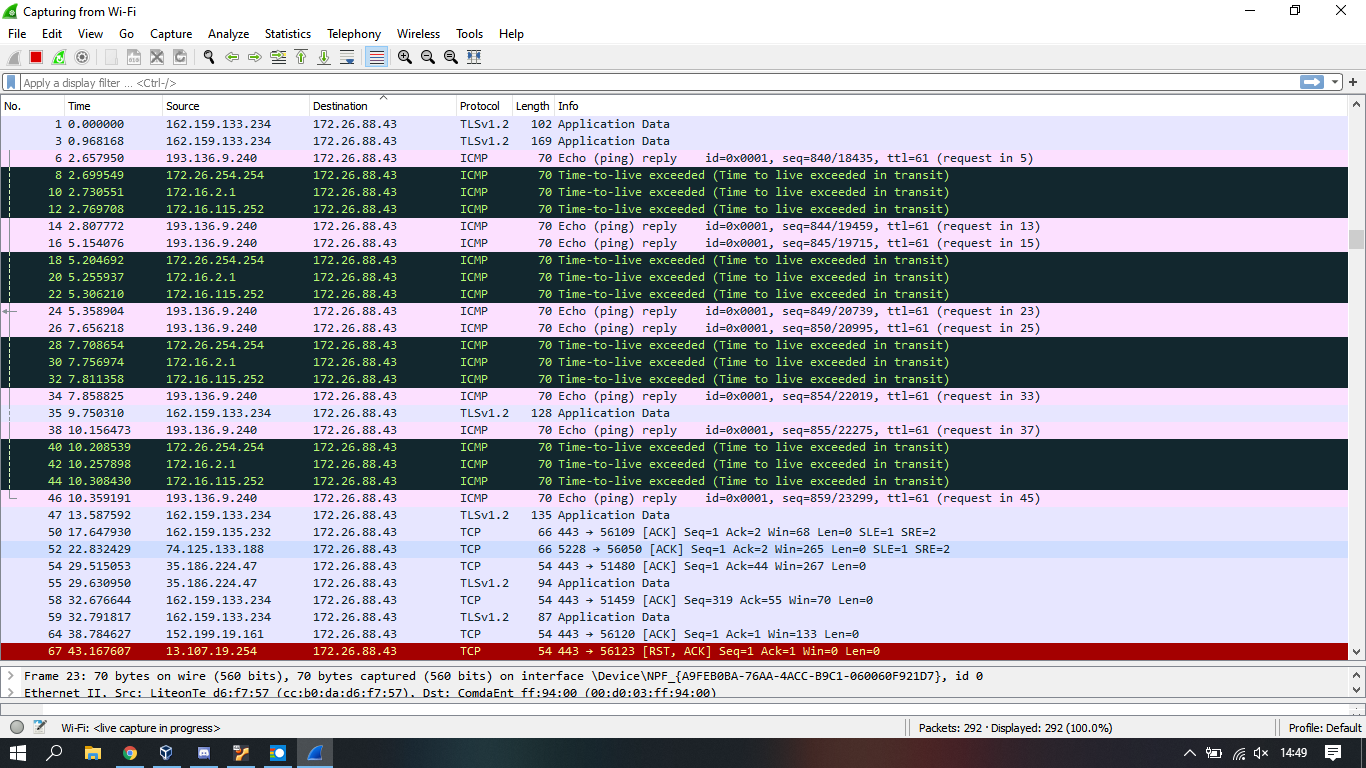


Figura 6- Pacotes capturados no Wireshark ordenados pelo endereço destino

O valor do TTL permanece constante com um valor de 61 para todas as mensagens de resposta ICMP TTL exceeced. Este valor advém do facto do TTL predefinido pelo destino ser 64; quando o pacote chega ao seu destino, como passou por 3 routers intermédios,e sabendo que por cada salto diminui em uma unidade, o TTL foi decrementado 3 vezes: 64 – 3 = 61.

## **Exercício 3**

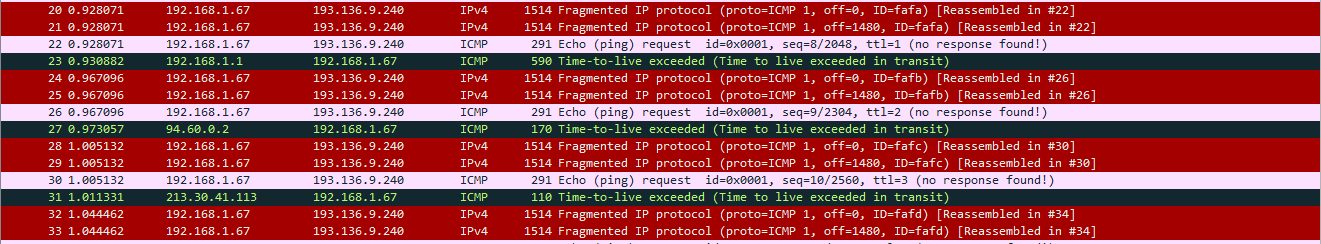


Figura 7- Fragmentos do datagrama IP

### **Alínea a**

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

A primeira mensagem ICMP é a 20. Há necessidade de fragmentar os pacotes quando o seu tamanho ultrapassa o MTU (Maximum Transmission Unit) que pela análise dos dados dos pacotes capturados é 1500 bytes. Ora, como o pacote enviado foi de 3237 bytes então foi fragmentado em 3 pacotes mais pequenos.

### **Alínea b**

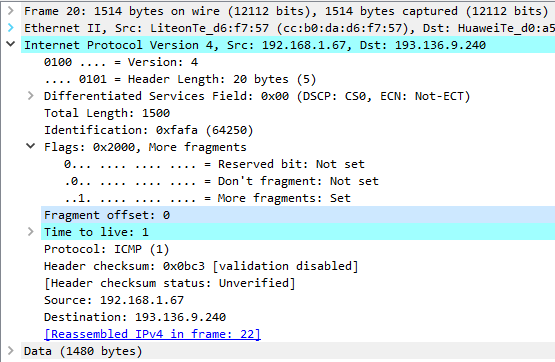
Imprima o primeiro fragmento do datagrama IP segmentado. Que informação no cabeçalho indica que o datagrama foi fragmentado? Que informação no cabeçalho IP indica que se trata do primeiro fragmento? Qual é o tamanho deste datagrama IP?

Figura 8- Cabeçalho do primeiro fragmento do datagrama IP segmentado

É possível verificar que o datagrama foi fragmentado pois a Flag “More Fragments” tem o valor 1. Trata-se do primeiro fragmento porque tem “Fragment offset “ 0. O tamanho deste datagrama IP é 1500 bytes, informação que se encontra no campo “Total Length”.

### **Alínea 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?

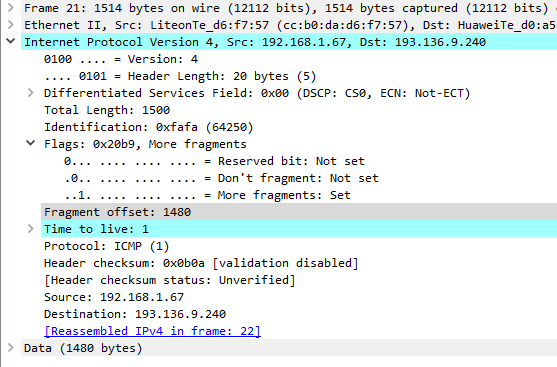


Figura 9- Cabeçalho do segundo fragmento do datagrama IP segmentado

Não se trata do primeiro fragmento do pacote pois o “Fragment offset” é diferente de 0, neste caso, 1480. Podemos concluir que existem mais fragmentos porque a flag “More Fragments” encontra-se a 1.

### **Alínea d**

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

Através da análise da figura 10, podemos verificar que a flag “More Fragments” se encontra agora a 0, o que significa que se trata do último fragmento. Assim, e de acordo com o que está indicado na figura, foram criados 3 fragmentos: 20, 21 e 22.

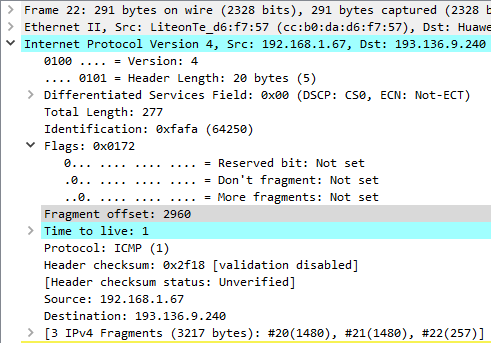


Figura 10- - Cabeçalho do último fragmento do datagrama IP segmentado

### **Alínea 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.

Os campos que mudam no cabeçalho IP entre os fragmentos são: o “Fragment offset” (que indica onde informação vai ser agregada), a flag “More Fragments” e o “Header checksum”. A flag “More Fragments” em conjunto com o offset permitem reconstruir o pacote na medida em que se ordenam por ordem crescente de offset até que a flag “More Fragments” seja 0. O campo header checksum serve apenas para verificar a integridade dos fragmentos.

# **Parte 2**

## **Exercício 1**

Figura 11 . Topologia CORE

Figura 12 - Topologia CORE

### **Alínea 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.

Representado na figura 12.

### **Alínea b**

Tratam-se de endereços públicos ou privados? Porquê?

Tratam-se de endereços privados pois os endereços de IP são valores entre 10.0.0.0 e 10.255.255.255 /8.

### **Alínea c**

Porque razão não é atribuído um endereço IP aos switches?

Não lhe são atribuídos IP porque servem para interligar dispositivos e passar informação entre eles; Fazem parte da camada de ligação 2 pelo que não são visíveis na camada de ligação 3. O switch aprende com a rede e depois apenas encaminha para os endereços MAC conhecidos.

### **Alínea 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).

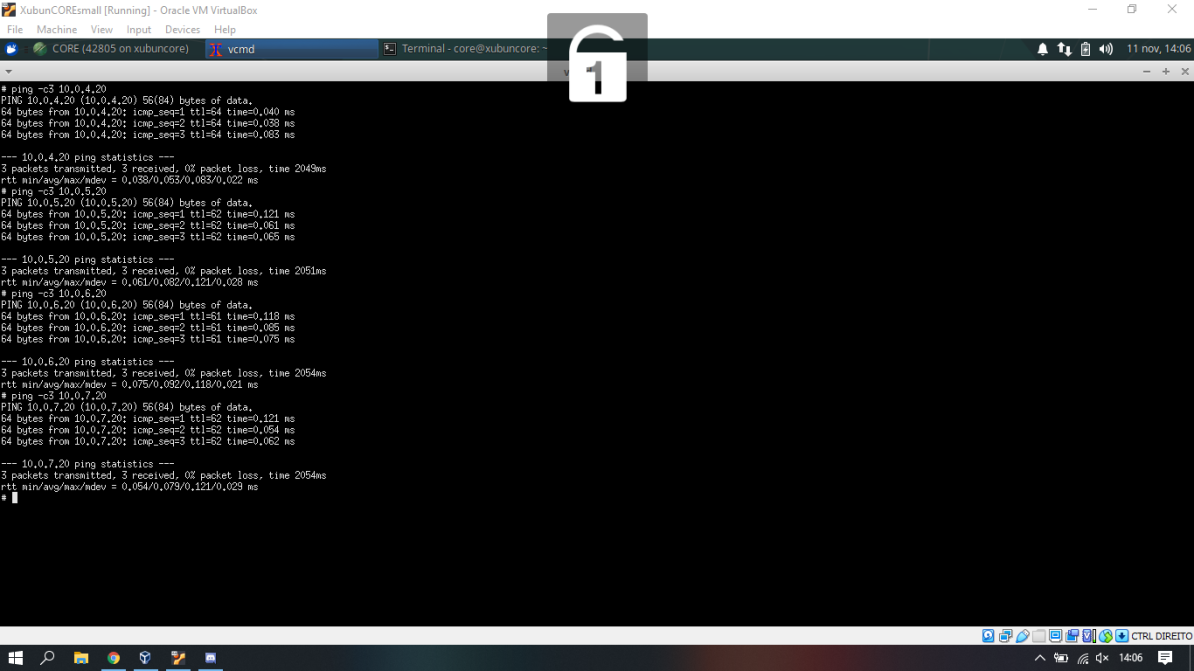


Figura 13- Conetividade entre o servidor e laptops de cada departamento

Como podemos observar na figura 13, existe conetividade IP entre os laptops dos vários departamentos e o servidor do departamento A. O comando ping teve resposta do servidor em cada um dos laptops.

### **Alínea e**

Verifique se existe conectividade IP do router de acesso RISP para o servidor S1.

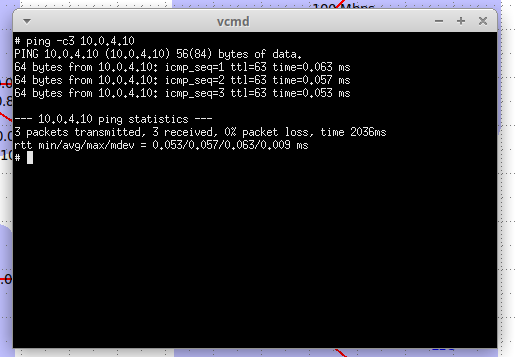


Figura 14- Ping para S1 a partir de RISP

Observando a figura 14, verifica-se que existe conetividade IP do router RISP para o servidor S1.

## **Exercício 2**

### **Alínea 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).

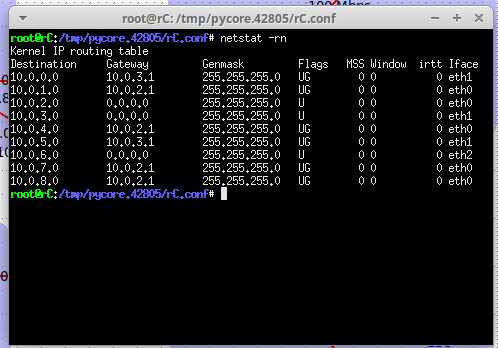


Figura 15- Tabela de encaminhamento do router do Departamento C

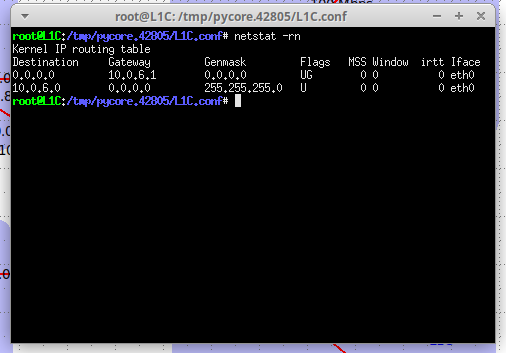


Figura 16 - Tabela de endereçamento de um laptop do Departamento C

As tabelas das imagens 15 e 16 são interpretadas da seguinte forma: um datagrama tem destino “Destination”, e é entrege na interface de endereço “Gateway” saindo pela interface “Iface”. A “Genmask” representa o tipo da máscara.

Mudando o foco de análise para as primeiras linhas da tabela da figura 15, reparamos que diferem no Gateway. Quando os endereços estão ligados diretamente, o Gateway não precisa de estar definido. Por sua vez, se não estiverem ligados diretamente, é necessário saber qual é o próximo salto.

Finalmente, em relação ao campo Flags, estas adicionam informação adicional. No caso, a flag “U” - rota válida; e “G” - a rota é para um router de gateway, e não para uma rede ou host diretamente conectado.

### **Alínea 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).

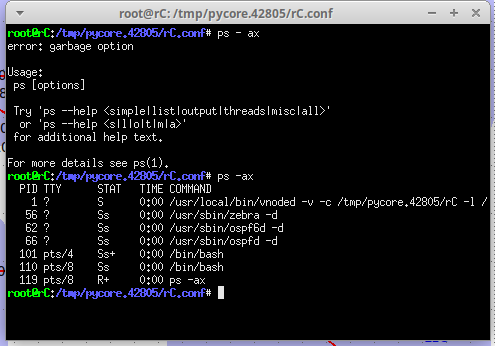


Figura 17- Comando ps -ax no Router C

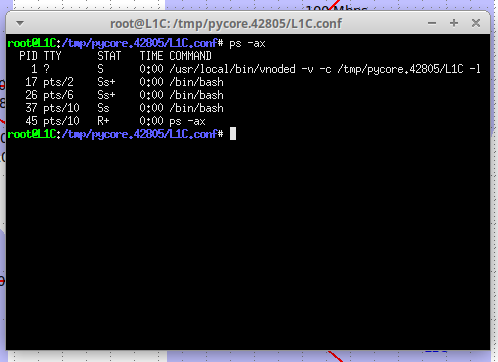


Figura 18 - Comando ps -ax num laptop do Departamento C

Ao analisar a figura 17, vemos que o router usa o protocolo OSPF (Open Shortest Path First) logo é usado encaminhamento dinâmico. As rotas são atualizadas ao longo do tempo.

### **Alínea 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 S1 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.

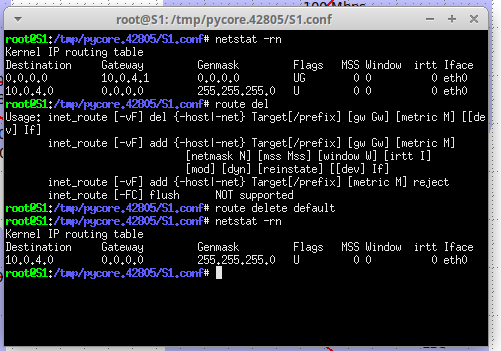


Figura 19 - Tabela de encaminhamento antes e depois da remoção da rota por defeito

Ao remover a rota por defeito é perdida a conetividade entre o servidor S1 e os restantes hosts que não pertencem à sua rede local. Isto acontece porque, ao eliminar a rota default, o servidor S1 passa a não saber para onde enviar de volta o que recebeu, isto é, não tem definida a rota de envio de tráfego para redes não locais. É possível enviar dados, no entanto, não é possível recebê-los de volta.

### **Alínea 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.

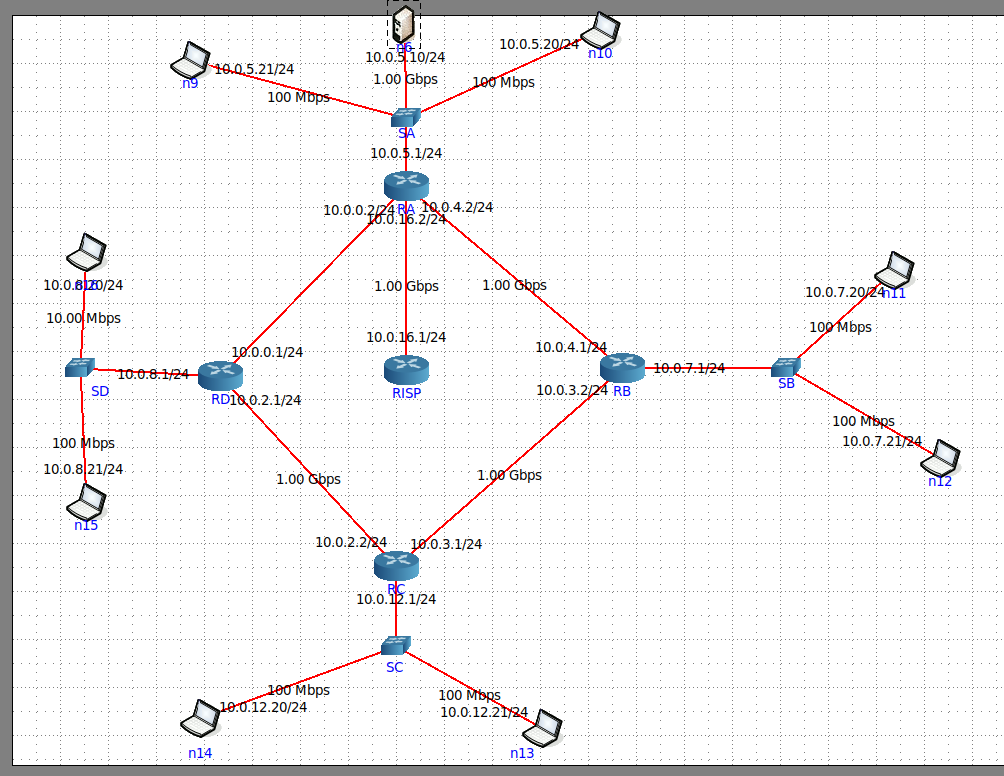
**Nota:** Devido a um problema na máquina virtual, a topologia a considerar a partir deste ponto é a apresentada acima.

Figura 20 - Topologia CORE

As rotas adicionadas, a partir de S1, foram:

* route add -net 10.0.8.0 netmask 255.255.255.0 gw 10.0.5.1
* route add -net 10.0.12.0netmask 255.255.255.0 gw 10.0.5.1
* route add -net 10.0.7.0 netmask 255.255.255.0 gw 10.0.5.1

### **Alínea 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

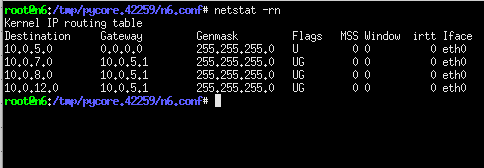


Figura 21- Tabela de endereçamento de S1 após serem repostas as rotas

Através da figura 21 verificamos que o servidor S1 está novamente acessível.

## **Exercício 3**

### **Alínea a**

Considere que dispõe apenas do endereço de rede IP 130.XX.96.0/19, em que XX é o decimal correspondendo ao seu número de grupo (PLXX). Defina um novo esquema de endereçamento para as redes dos departamentos (mantendo a rede de acesso e core inalteradas) e atribua endereços às interfaces dos vários sistemas envolvidos. Assuma que todos os endereços de sub-redes são usáveis. Deve justificar as opções usadas.

É dado o endereço 130.39.96.0/19 para o *subnetting* dos departamentos.

     - Assim para quatro sub-redes e tendo em conta que todos os endereços da sub-rede são usáveis, são necessários 2 bits para diferenciar os quatro departamentos (130.39.011**|00|**000.0/21).

    Desta maneira, os endereços para as sub-redes foram atribuídos da seguinte forma:

     Subrede 1(Departamento A) : 130.39.011|00|000(96).0/21

     Subrede 2(Departamento B) : 130.39.011|01|000(104).0/21

     Subrede 3(Departamento C) : 130.39.011|10|000(112).0/21

     Subrede 4(Departamento D) : 130.39.011|11|000(120).0/21

   - Assumindo que os primeiros e últimos endereços estão reservados, cada departamento dispõe da respetiva gama de endereçamento:

     Departamento A: 130.39.96.1/21 - 130.39.103.254/21

     Departamento B: 130.39.104.1/21 - 130.39.111.254/21

     Departamento C: 130.39.112.1/21 - 130.39.119.254/21

     Departamento D: 130.39.120.1/21 - 130.39.127.254/21

   - Tendo em mente a gama de cada departamento, foi escolhido o primeiro endereço de cada gama para a interface do router do respetivo departamento. Quanto aos hosts e ao servidor estes podem levar qualquer endereço na gama e, assim sendo, a cada host foi atribuído o endereço menor disponível e ao servidor foi atribuído o maior endereço do seu departamento (Departamento A) para fácil identificação.

    A) Interface do router 130.39.96.1/21

         HostA1………...130.39.96.2/21

         HostA2………...130.39.96.3/21

         Server………...130.39.96.254/21

    B) Interface do router 130.39.104.1/21

         HostB1………...130.39.104.2/21

         HostB2………...130.39.104.3/21

    C) Interface do router 130.39.112.1/21

         HostC1………...130.39.112.2/21

         HostC2………...130.39.112.3/21

    D) Interface do router 130.39.120.1/21

         HostD1………...130.39.120.2/21

         HostD2………...130.39.120.3/21

### **Alínea b**

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

A máscara usada foi 255.255.248.0.

A quantidade de endereços IP disponíveis para interligação em cada departamento, tendo em conta que sobram 11 bits para a identificação dos hosts e que o primeiro e último endereço são reservados, é endereços.

### **Alínea c**

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

Alterar a tabela de routing IP de cada host de forma a:

         - Remover as rotas antigas. (1)

        -Ter o ip novo da respetiva interface do router do respetivo departamento como a rota default (0.0.0.0). (2)

        -Ter a rota da subrede do respetivo departamento. (3)

    Alterar a tabela de routing IP de cada router de forma a:

        - Remover as rotas antigas para os departamentos. (4)

        -Ter as novas rotas para os outros departamentos. (5)

(1)  Every Host: route del default

       Hosts A: route del -net 10.0.5.0/24

       Hosts B: route del -net 10.0.7.0/24

       Hosts C: route del -net 10.0.12.0/24

       Hosts D: route del -net 10.0.8.0/24

(2)  Hosts A: route add -net default gw 130.39.96.1

       Hosts B: route add -net default gw 130.39.104.1

       Hosts C: route add -net default gw 130.39.112.1

       Hosts D: route add -net default gw 130.39.120.1

(3)  Hosts A: route add -net 130.39.96.0/21 dev eth0

       Hosts B: route add -net 130.39.104.0/21 dev eth0

       Hosts C: route add -net 130.39.112.0/21 dev eth0

       Hosts D: route add -net 130.39.120.0/21 dev eth0

(4)  Every Router:  route del -net 10.0.7.0/24

                               route del -net 10.0.12.0/24

                               route del -net 10.0.8.0/24

                            route del -net 10.0.5.0/24

(5)  Router A:  route add -net 130.39.96.0/21 dev eth2

                         route add -net 130.39.104.0/21 dev eth1

                         route add -net 130.39.112.0/21 dev eth1

                         route add -net 130.39.120.0/21 dev eth0

       Router B:  route add -net 130.39.96.0/21 dev eth1

                         route add -net 130.39.104.0/21 dev eth2

                         route add -net 130.39.112.0/21 dev eth0

                         route add -net 130.39.120.0/21 dev eth0

      Router C:  route add -net 130.39.96.0/21 dev eth0

                        route add -net 130.39.104.0/21 dev eth1

                        route add -net 130.39.112.0/21 dev eth2

                        route add -net 130.39.120.0/21 dev eth0

      Router D:  route add -net 130.39.96.0/21 dev eth0

                        route add -net 130.39.104.0/21 dev eth0

                        route add -net 130.39.112.0/21 dev eth1

                        route add -net 130.39.120.0/21 dev eth2

O resultado desta operação foi:

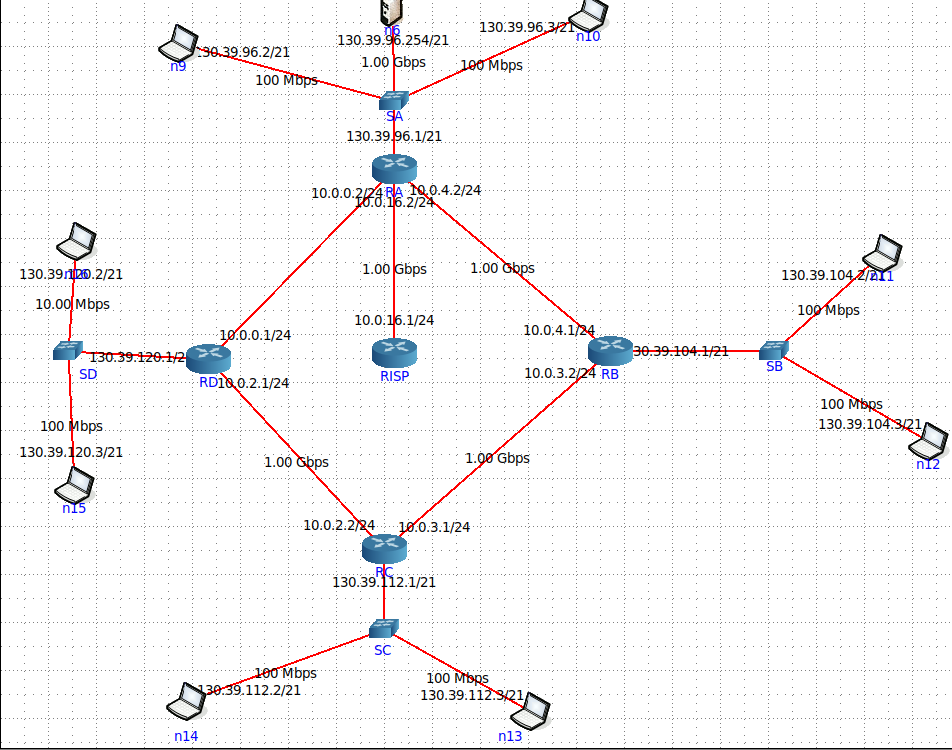
****

Figura 22-Nova Topologia CORE

**Nota:** Um dos alunos neste grupo estava inserido erroneamente no grupo 39 da Blackboard, que não corresponde ao nosso grupo. Apenas reparamos neste pormenor após o exercício estar concluído pelo que o valor de XX utilizado foi 39 em vez de 37 que seria o número correto.

# **Conclusão**