# **Trabalho Prático Nº2**

Universidade do Minho

Ana Inês Leite

[a96159@uminho.pt](mailto:a96159@uminho.pt)

Diana Filipa Ferreira Malheiro Teixeira

[a97516@uminho.pt](mailto:a97516@uminho.pt)

Pedro Marcelo Bogas Oliveira

[a95076@uminho.pt](mailto:a95076@uminho.pt)

**PL41**

Índice

[**Trabalho Prático Nº2** 1](#_Toc100347050)

[Objetivo 3](#_Toc100347051)

[Parte I 3](#_Toc100347052)

[1. Prepare uma topologia CORE para verificar o comportamento do *traceroute*. 3](#_Toc100347053)

[2. Pretende-se agora usar o traceroute na sua máquina nativa, e gerar datagramas IP de diferentes tamanhos. 5](#_Toc100347054)

[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 (4000 + X) bytes. 7](#_Toc100347055)

[Parte II 10](#_Toc100347056)

[1. Atenda aos endereços IP atribuídos automaticamente pelo CORE aos diversos equipamentos da topologia. 10](#_Toc100347057)

[2. Para o router RA e o portátil Bela: 14](#_Toc100347058)

[3. Definição de Sub-redes. 17](#_Toc100347059)

[Conclusão 20](#_Toc100347060)

# Objetivo

O principal objetivo deste trabalho é o estudo do Internet Protocol (IP) nas suas principais vertentes, nomeadamente:

* Estudo do formato de um pacote ou datagrama IP
* Fragmentação de pacotes IP
* Endereçamento IP
* Encaminhamento IP

Numa primeira fase, é realizado o registo de datagramas IP enviados e recebidos através da execução do programa *traceroute*. São analisados os vários campos de um datagrama IP e detalhado o processo de fragmentação realizado pelo protocolo IP.

# Parte I

## Prepare uma topologia CORE para verificar o comportamento do *traceroute*.

Seguindo as instruções dadas no enunciado, preparamos uma topologia CORE, com um host (PC), com o nome Bela, ligado ao router de acesso R2. Este router está ligado, simultaneamente, aos routers R3 e R4, ligados a R5. Por fim, criamos uma ligação entre R5 e o host Monstro (servidor).

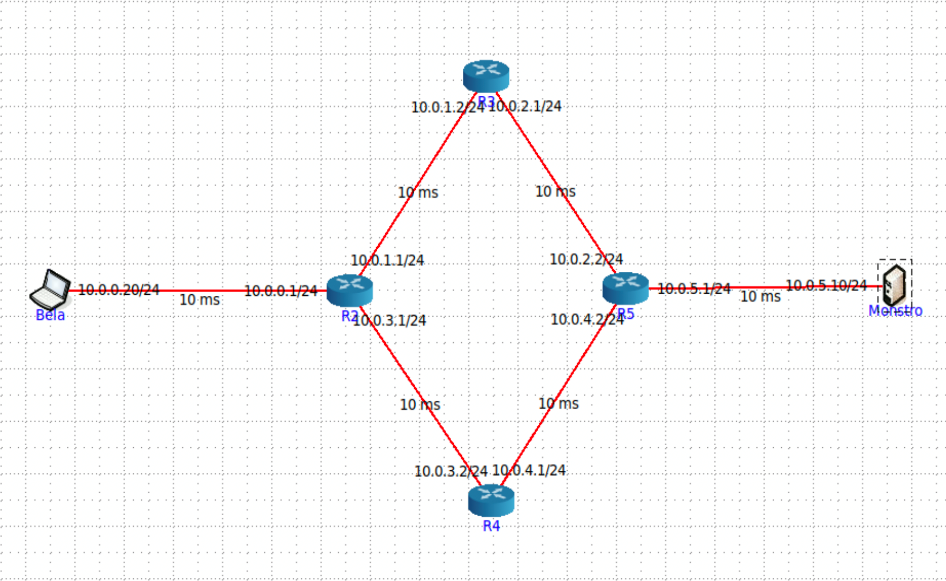


Figura - Topologia CORE para verificar o comportamento do traceroute.

1. **Ative o wireshark ou o tcpdump no host Bela. Numa Shell de Bela, execute o comando *traceroute -l* para o endereço IP do Monstro.**

Uma imagem com texto

Descrição gerada automaticamente

Figura - Execução do comando traceroute -l para o endereço IP do Monstro

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

Uma imagem com mesa

Descrição gerada automaticamente

Figura - Tráfego ICMP enviado e recebido como resposta pelo sistema Bela

Uma imagem com mesa

Descrição gerada automaticamente

Figura - Tráfego ICMP enviado e recebido como resposta pelo sistema Bela

Esperávamos que o TTL começasse em 1 e fosse incrementando de 1 em 1, até chegar ao TTL necessário.

No entanto, verificamos que o TTL ultrapassou o valor necessário, porque depois de o ter atingido, continuou a executar com os pacotes restantes. Verificamos, ainda, que a impressão é feita à medida que os TTL vão terminando a execução, ao invés de ser feita por ordem de início.

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

O valor inicial mínimo do campo TTL para alcançar o servidor Monstro deve ser 4, pois é a partir deste que o servidor recebe uma resposta, como podemos ver na Figura 4.

1. **Calcule o valor médio do tempo de ida-e-volta (RTT - Round-Trip Time) obtido no acesso ao servidor. Para melhorar a média, poderá alterar o número pacotes de prova com a opção -q.**

Uma imagem com texto

Descrição gerada automaticamente

Figura - Tempo médio inicial de ida-e-volta obtido no acesso ao servidor

Valor médio do tempo de ida-e-volta = (165.772+165.772+165.771)/3 = 165.7717 ms.

Valor médio melhorado = 84.165 ms.

**Uma imagem com texto

Descrição gerada automaticamente**

Figura - Tempo médio melhorado (-q) de ida-e-volta obtido no acesso ao servidor

1. **O valor médio do atraso num sentido (One-Way Delay) poderia ser calculado com precisão dividindo o RTT por dois? O que torna difícil o cálculo desta métrica?**

Não, uma vez que isto não é uma métrica precisa e o que é enviado pode não ser o que é recebido. Havendo também diferenças nos atrasos de cada sentido, o que acaba por influenciar o delay, tornando este cálculo mais difícil.

## Pretende-se agora usar o traceroute na sua máquina nativa, e gerar datagramas IP de diferentes tamanhos.

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

172.26.97.144

Uma imagem com texto

Descrição gerada automaticamente

Figura - endereço IP da interface ativa do computador

1. **Qual é o valor do campo protocolo? O que permite identificar?**

O valor do campo protocolo é ICMP(1) e permite identificar o protocolo utilizado no envio da mensagem.

1. Uma imagem com texto

   Descrição gerada automaticamente**Quantos bytes tem o cabeçalho IPv4? Quantos bytes tem o campo de dados (payload) do datagrama? Como se calcula o tamanho do payload?**

Figura -bytes de cabeçalho e total

* Tamanho do cabeçalho IPv4 (bytes) = 20;
* Tamanho total (bytes) = 56;
* Tamanho payload = 56 – 20 = 36.

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

Não. Pois como se pode ver na Figura 8, o “Fragment Offset” é 0 e o campo “More Fragments” está “Not Set”, o que nos leva à conclusão de que não foi fragmentado.

1. **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.**

Uma imagem com texto, captura de ecrã, eletrónica, computador

Descrição gerada automaticamenteOs campos que variam de pacote para pacote são Identification, TTL e o o Header Checksum.

Figura 9 - Pacotes ordenados de acordo com o endereço IP fonte

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

Como podemos ver na Figura 8, o valor da “identification” decrementa 1 a 1, bem como o TTL.

Quando o TTL chega a 1, tem de seguida um valor igual a 225 e repete-se o processo de decrementação 1 a 1.

1. **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?**

O valor do TTL está no intervalo [253,255], uma vez que excede o seu limite.

Figura - Tráfego ordenado por endereço de destino

## 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 (4000 + X) bytes.

**Uma imagem com texto

Descrição gerada automaticamente**

Figura 10 - Alteração do package size para 4000 + 41

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

Houve necessidade de fragmentar o pacote inicial uma vez que era demasiado grande para ser transmitido de uma só vez, porque a mensagem tinha 4021 bytes, sendo que só é possível transmitirem-se 1500 de cada vez.

**Uma imagem com texto

Descrição gerada automaticamente**

Figura 11 - Primeira mensagem ICMP

1. **Uma imagem com texto

   Descrição gerada automaticamenteImprima 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 12 - Primeiro fragmento do datagrama IP segmentado

Como a flag “More Fragments” tem valor 1, concluímos que existem mais fragmentos. Como o campo “fragment offset” tem valor 0, concluímos que estamos perante o primeiro fragmento.

O tamanho deste datagrama IP é 1500 bytes.

1. **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?**

**Uma imagem com texto

Descrição gerada automaticamente**

Figura 13 - Segundo fragmento do datagrama IP original

Como podemos ver na Figura 13, o “Fragment Offset” tem valor 1480. Sendo este valor diferente de 0, sabemos que não é o primeiro fragmento.

Como “more fragments” tem valor 1, podemos concluir que há mais fragmentos.

1. **Quantos fragmentos foram criados a partir do datagrama original?**

**Uma imagem com texto

Descrição gerada automaticamente**

Figura 14- Terceiro fragmento do datagrama IP segmentado

Como podemos ver na Figura 14, há mais dois fragmentos com identification igual à do primeiro fragmento. Assim, podemos concluir que foram criados 3 fragmentos a partir do datagrama original.

Sabemos, ainda, que a flag “more fragments” do terceiro fragmento tem valor 0, permitindo-nos concluir que se trata do último.

1. **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 diferentes fragmentos são header checksum, fragment offset e total length.

Essa informação permite reconstruir o datagrama original, uma vez que todos os fragmentos pertencem ao mesmo datagrama (informação obtida através do campo identification).

O campo fragment offset permite-nos saber onde começada cada fragmento.

E, por fim, o campo total length fornece-nos informação sobre o tamanho de cada fragmento.

1. **Verifique o processo de fragmentação através de um processo de cálculo.**

Dados:

* Tamanho da mensagem = 4041 bytes
* Cabeçalho = 20 bytes
* Tamanho do datagrama = 1500 bytes
* Fragment Offset (do primeiro fragmento) = 0

Assim, fragment offset(do segundo fragmento) será 0+(1500-20)= 1480 bytes e o do último fragmento é 1480+(1500-20)=2960 bytes.

1. **Escreva uma expressão lógica que permita detetar o último fragmento correspondente ao datagrama original.**

Último fragmento = Valor no campo Fragment Offset + Tamanho da mensagem.

# Parte II

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

1. **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.**

**Uma imagem com texto, captura de ecrã, computador

Descrição gerada automaticamente**

Figura 15 - Topologia e endereçamento usado

Os endereços IP atribuídos podem ser observados na figura 15.

1. **Tratam-se de endereços públicos ou privados? Porquê?**

Endereços privados, porque estão no range de endereços privados (endereços no intervalo [10.0.0.0; 10.255.255.255]).

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

Sendo que um switch opera com endereços físicos, servindo de ligação para a transmissão de dados, não necessita de um endereço lógico (endereço IP).

1. **Usando o comando ping certifique-se que existe conectividade IP interna a cada departamento (e.g. entre um laptop e o servidor respetivo).**

Após efetuar o comando ping para um PC de cada departamento, verificou-se existência de conectividade entre os dois dispositivos.

**Uma imagem com texto, captura de ecrã, computador

Descrição gerada automaticamente**

Figura 16- Conectividade IP entre SA e Bela

**Uma imagem com texto, captura de ecrã, computador

Descrição gerada automaticamente**

Figura 17- Conectividade IP entre SB e Jasmine

**Uma imagem com texto, captura de ecrã, computador

Descrição gerada automaticamente**

Figura 18 - Conectividade IP entre SC e Ariel

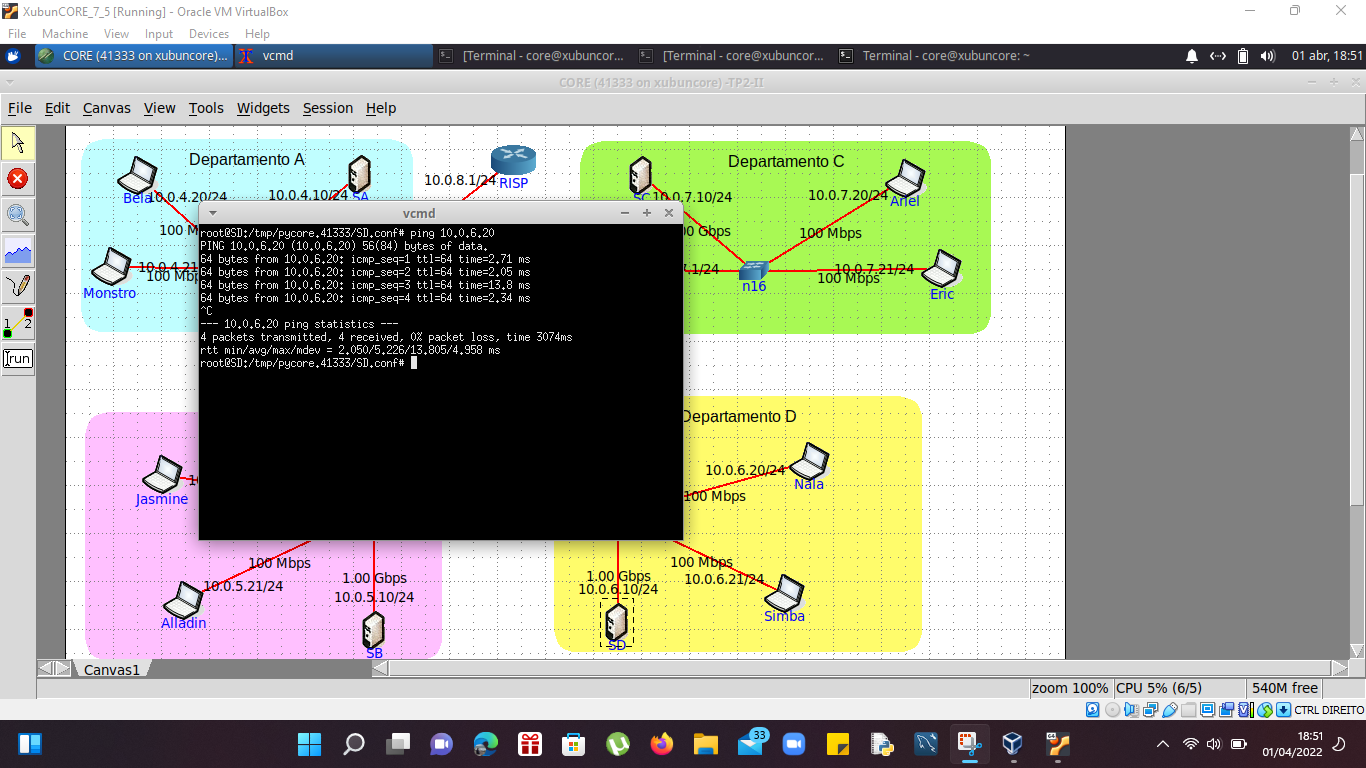
****

Figura 19- Conectividade IP entre SD e Nala

1. **Execute o número mínimo de comandos ping que lhe permite verificar a existência de conetividade IP entre departamentos.**

**Uma imagem com texto, captura de ecrã, computador, apresentação

Descrição gerada automaticamente**

Figura 20- Conectividade entre o departamento A e B,C e D

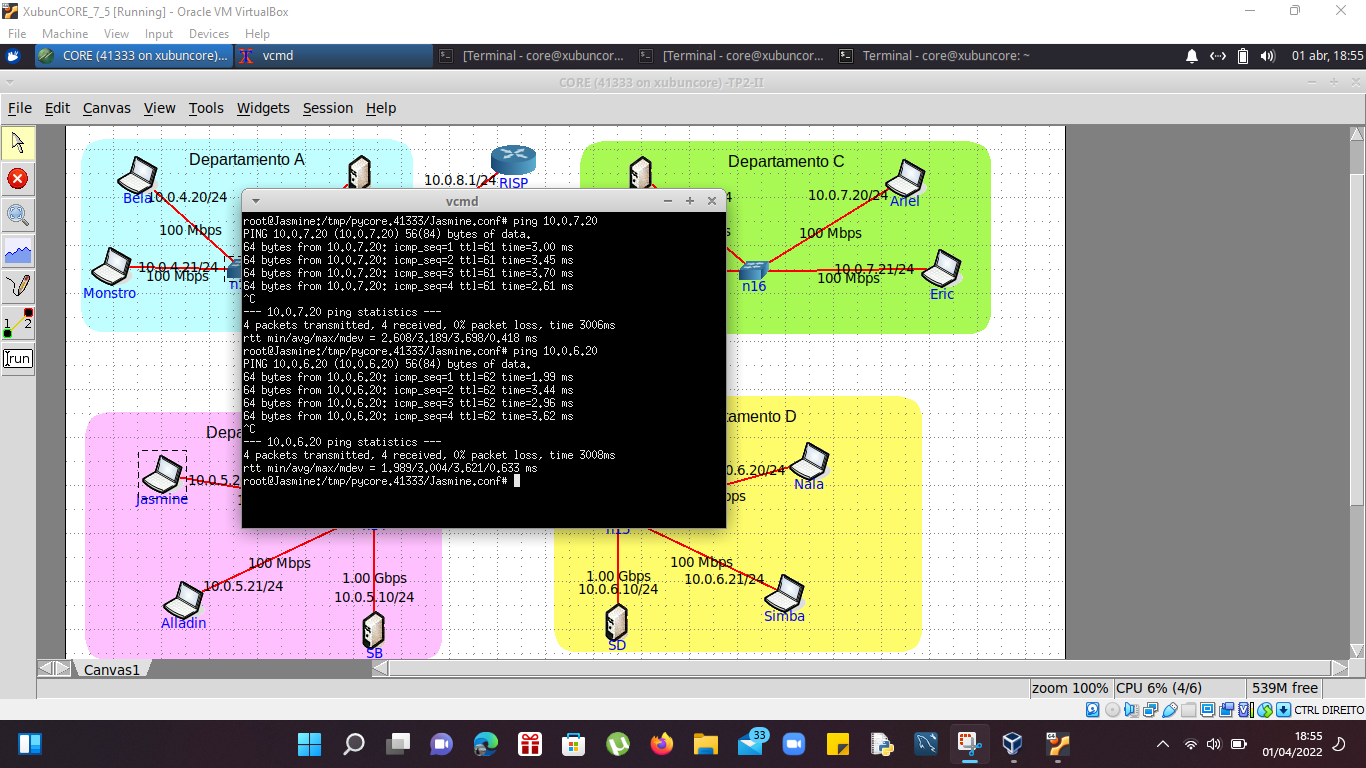
****

Figura 21 - Conectividade entre o departamento B e C,D

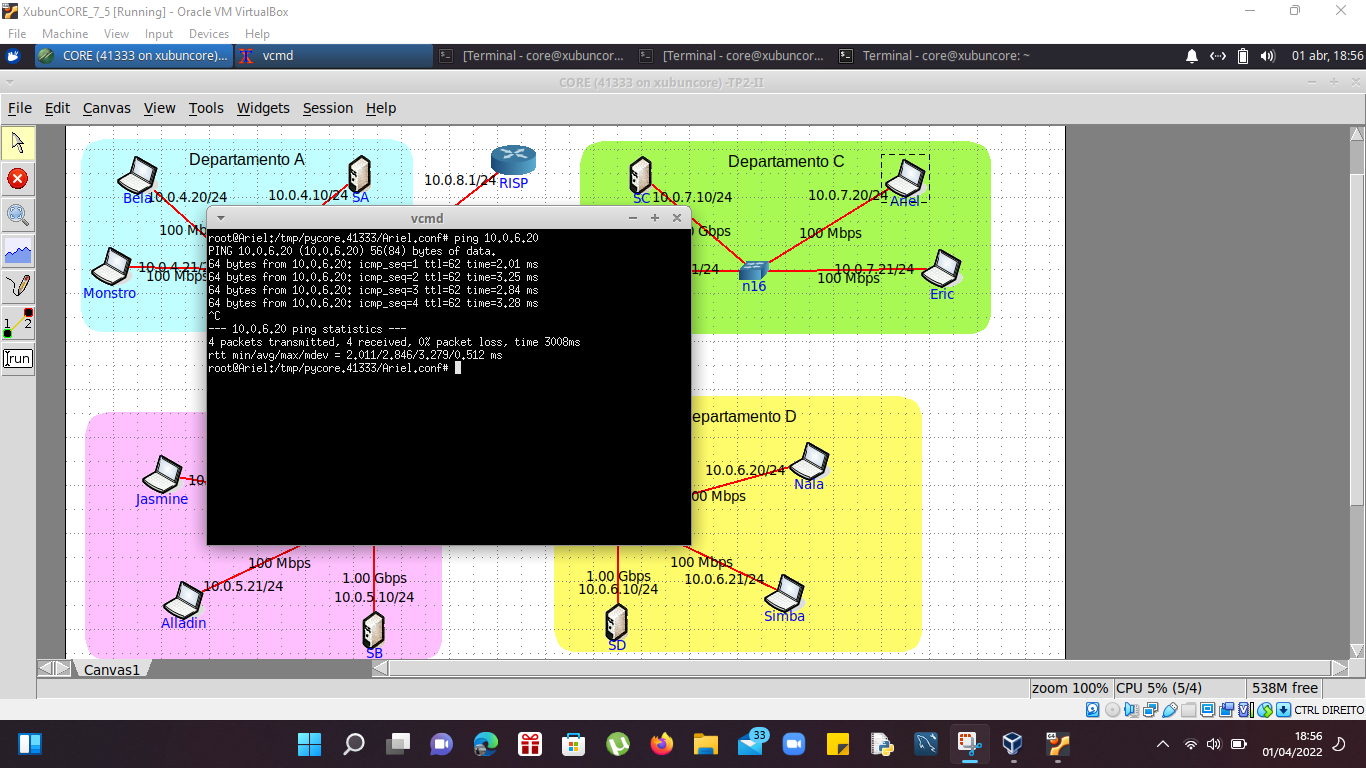
****

Figura 22 - Conectividade entre os departamentos C e D

1. **Verifique se existe conectividade IP do portátil Bela para o router de acesso RISP.**

Como podemos ver na Figura 23, existe conectividade IP entre o portátil Bela e o router RISP.

**Uma imagem com texto, captura de ecrã, computador, interior

Descrição gerada automaticamente**

Figura 23 - Conectividade IP entre Bela e RISP

## Para o router RA e o portátil Bela:

1. **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).**

Interpretação das entradas de cada tabela:

* Destination – IP destino;
* Gateway – Porta de saída do router;
* Gensmask – Máscara 255.255.255.0 (24 bits);
* Flag – Para distinguir redes locais(U) de globais(UG);
* MSS – Tamanho máximo de pacote que o destinatário suporta num único segmento de TCP;
* Window – Quantidade de dados que podem ser enviados antes do último;
* Irtt lface – Interface destino

**Uma imagem com texto, captura de ecrã, computador, interior

Descrição gerada automaticamente**

Figura 24 - Tabela de encaminhamento unicast (IPv4)

1. **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 ou equivalente).**

Como podemos ver na Figura 27, está a ser usado o OSPF6, que, sendo um protocolo de encaminhamento dinâmico, nos indica que está a ser utilizado um encaminhamento dinâmico.

**Uma imagem com texto, captura de ecrã, computador, interior

Descrição gerada automaticamente**

Figura 25 - Processos que decorrem em cada sistema

1. **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 SA. Use o comando route delete para o efeito. Que implicações tem esta medida para os utilizadores da LEI-RC que acedem ao servidor. Justifique.**

**Uma imagem com texto, captura de ecrã, computador, interior

Descrição gerada automaticamente**

Figura 26 - Remoção da rota por defeito da tabela de encaminhamento do SA

Ao eliminar definitivamente a rota por defeito, a ligação entre SA e as restantes sub-redes é eliminada, pelo que o SA apenas encaminha tráfego de pacotes para a sua rede.

1. **Não volte a repor a rota por defeito. Adicione todas as rotas estáticas necessárias para restaurar a conectividade para o servidor SA, por forma a contornar a restrição imposta na alínea c). Utilize para o efeito o comando route add e registe os comandos que usou.**

Comandos utilizados:

* route add -net 10.0.6.0 netmask 255.255.255.0 gw 10.0.4.1
* route add -net 10.0.7.0 netmask 255.255.255.0 gw 10.0.4.1

**Uma imagem com texto, captura de ecrã, computador, interior

Descrição gerada automaticamente**

Figura27– Adição das rotas estáticas

1. **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.**

**Uma imagem com texto

Descrição gerada automaticamente**

Figura 28- Testes com o comando ping

## Definição de Sub-redes.

1. **Considere que dispõe apenas do endereço de rede IP 192.168.XXX.128/25, em que XXX é o decimal correspondendo ao seu número de grupo (PLXX). Defina um novo esquema de endereçamento para as redes dos departamentos (mantendo as redes de acesso externo e backbone inalteradas), sabendo que o número de departamentos pode vir a aumentar no curto prazo. Atribua endereços às interfaces dos vários sistemas envolvidos. Assuma que todos os endereços de sub-redes são usáveis. Justifique as opções tomadas no planeamento.**

Para o nosso grupo, o endereço de rede IP que possuímos será o 192.168.41.128/25, onde o /25 significa:

* 25 bits para a rede, que passará a 28 (25 para a rede + 3 para as sub-redes)
* 7 bits para o host, que dará 3 para as sub-redes e passará a ser 4 para os hosts

Fazendo estas mudanças porque, da forma original, temos 0000000 bits para os hosts, que dá, no total (1111111) 147 hosts diferentes, ou seja, mais do que é pedido, possuindo, por outro lado, menos do que os necessários para as sub-redes.

Tendo passado 3 bits para as sub-redes porque, como com 2 bits, é apenas possível obter 4 das mesmas, o que não permite a adição de mais no futuro e com 4 bits, é possível obter 16, o que, para além de ser muito, limita bastante o número de hosts possíveis, concluímos que 3 é o número ideal, uma vez que, permite uma boa margem para o aumento das sub-redes (resulta em 8 sub-redes) e um bom número de hosts (16).

Face a estas mudanças, terá de ser mudada também a máscara de sub-rede, passando esta a ser 255.255.255.224 (11111111.11111111.11111111.11100000).

Com estas em mente, verificamos também que dos 16 hosts possíveis em cada sub-rede, vamos apenas ter 14, pois o endereço zero é inválido (é utilizado para especificar uma rede sem especificar um host) e o último endereço é usado para transmitir uma mensagem para cada host numa rede, o que os faz não poderem ser atribuídos a um host individual.

Como tal, e com estes endereços de host binários com todos os bits a um ou zeros são inválidos, sabemos então que vamos andar sempre de 14 em 14 nos endereços.

O que os faz com que fiquemos apenas com estes endereços disponíveis:

* + 1. do 192.168.41.129 -> 192.168.41.142
    2. do 192.168.41.145 -> 192.168.41.158
    3. do 192.168.41.161 -> 192.168.41.174
    4. do 192.168.41.177 -> 192.168.41.190
    5. do 192.168.41.193 -> 192.168.41.206
    6. do 192.168.41.209 -> 192.168.41.222
    7. do 192.168.41.225 -> 192.168.41.238
    8. do 192.168.41.241 -> 192.168.41.254

Fazendo o nosso grupo as seguintes atribuições:

*Departamento A:*

* RA -> 192.168.41.129/28
* SA -> 192.168.41.130/28
* Bela -> 192.168.41.131/28
* Monstro -> 192.168.41.132/28

*Departamento B:*

* RB -> 192.168.41.145/28
* SB -> 192.168.41.146/28
* Jasmine -> 192.168.41.147/28
* Alladin -> 192.168.41.148/28

*Departamento C:*

* RC -> 192.168.41.161/28
* SC -> 192.168.41.162/28
* Eric -> 192.168.41.163/28
* Ariel -> 192.168.41.164/28

*Departamento D:*

* RD -> 192.168.41.177/28
* SD -> 192.168.41.178/28
* Nala -> 192.168.41.179/28
* Simba -> 192.168.41.180/28

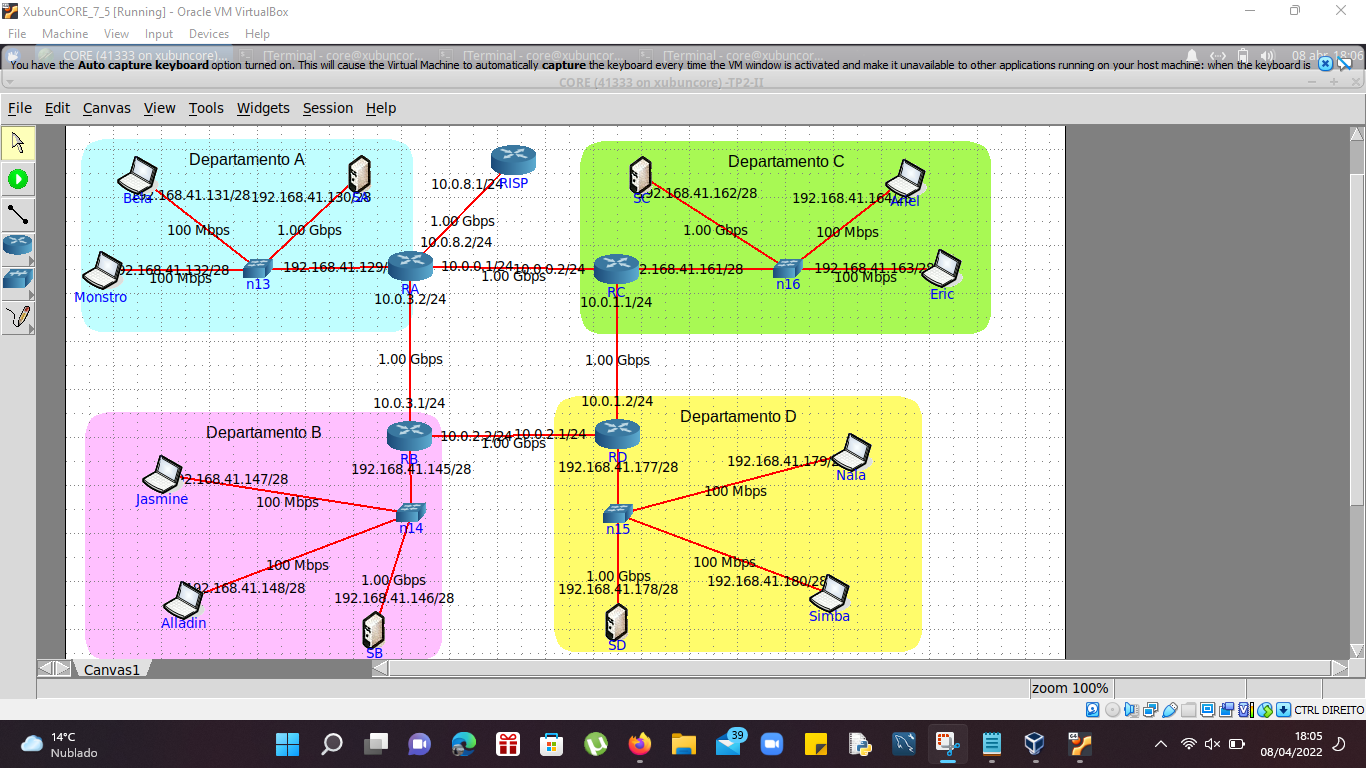
****

Figura 29- Atribuição dos novos endereços

1. **Qual a máscara de rede que usou (em formato decimal)? Quantos hosts IP pode interligar em cada departamento? Quantos prefixos de sub-rede ficam disponíveis para uso futuro? Justifique.**

Tal como visto na alínea anterior, a máscara de sub-rede teve que ser: 255.255.255.224, visto que alteramos o número de bits para os hosts e para as sub-redes, passando a ter 192.168.41.128/28 e não192.168.41.128 /25.

Há 14 hosts IP que se podem interligar em cada departamento, pelo simples facto de que, tendo 4 bits para os hosts, podíamos representar 16 hosts para cada, só que, tal como vimos anteriormente, o primeiro (tudo a 0’s) e o último (tudo a 1’s) não podem ser usados.

Ficando a sobrar, das 8 sub-redes possíveis, 4, pois estaremos agora a usar 4 para implementar os 4 departamentos.

1. **Verifique e garanta que a conectividade IP interna na rede local LEI-RC é mantida. No caso de não existência de conetividade, reveja a atribuição de endereços efetuada e eventuais erros de encaminhamento por forma a realizar as correções necessárias. Explique como procedeu.**

**Uma imagem com texto

Descrição gerada automaticamente**

Figura 30- Garantia que a conectividade IP interna na rede local é mantida

# Conclusão

Para finalizar, graças a este trabalho prático, foi-nos possível a consolidação dos conhecimentos e da matéria abordada nas aulas teóricas, ficando, assim, com uma melhor compreensão de como funciona a subneting, o endereçamento, a fragmentação e o encaminhamento de IPs. Tendo este ajudado também a adquirir experiência com ferramentas importantes como é o cado do CORE, o Wireshark e o PingPlotter, que nos auxiliaram no desenvolvimento deste trabalho.