**Protocolo IP**

Ficha TP2

**Alberto Faria, César Augusto e Diogo Nogueira (Grupo 47)**

Universidade do Minho, Departamento de Informática, 4710-057 Braga, Portugal

e-mail: {a79077, a79014, a78957}@alunos.uminho.pt

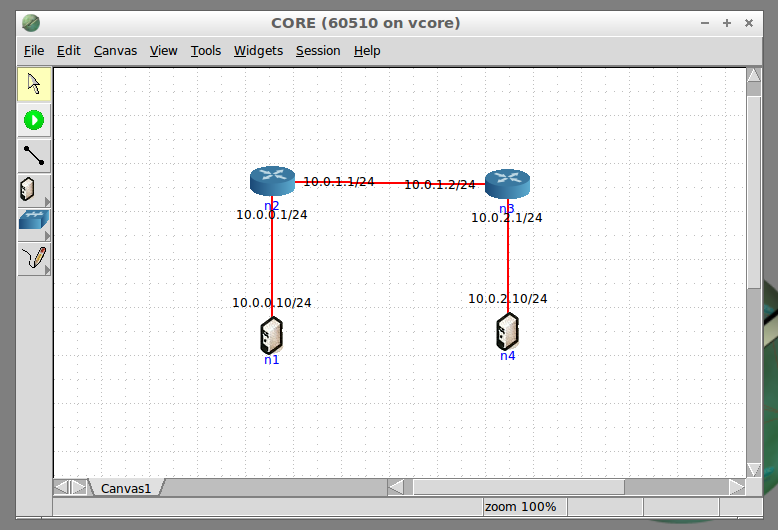
------------------------------------

**Questões e Respostas**

Parte I

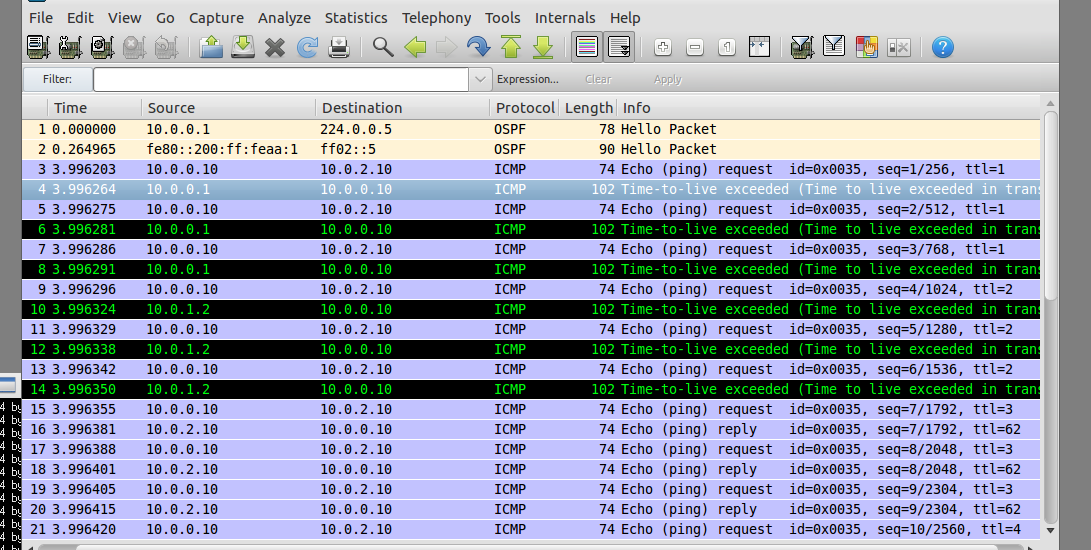
Exercício 1.

**a.**



**b.** Pela observação de todo o tráfego ICMP que consta no Wireshark (figura abaixo anexada) verificamos que inicialmente é enviada pelo host n1 uma request que possuí, tal como seria de esperar, um TTL = 1. Sendo este TTL insuficiente para a obtenção de uma resposta por parte do host n4, ao chegar ao router n2, este gera uma mensagem de erro do tipo “Time to live exceded”. Desta forma, é feita uma nova request por parte do host n1, mas, desta vez, com um TTL = 2. Por continuar a ser um TTL escasso é novamente produzida a mesma mensagem de erro, sendo que desta vez, por parte do router n3, já que este foi o limite máximo que se consegui alcançar com este TTL. Verificamos assim que apenas quando o TTL passa a ser igual a 3 é que o host n4 consegue finalmente conceder uma resposta.

Todas estas conclusões podem ser facilmente suportadas pela figura abaixo anexada.



**c.** O valor mínimo do campo TTL para pode alcançar o destino n4 terá de ser igual a 3 conforme concluído na alínea (b). Em termos práticos este valor é facilmente deduzível. Basta pensar que, como um router deve decrementar o TTL de pelo menos 1 unidade **(informação que consta no RFC 791)** de cada vez que acontece routing, então, para que seja enviada uma mensagem do host n1 para o host n4, o TTL terá de ter no mínimo o valor 3. Ao ser igual a 3, o TTL chega ao host n4 com o valor de 0, conseguindo-se assim obter uma resposta por parte do host n4.

**d.** Pela observação dos diversos valores, podemos concluir que o valor médio de tempo ida-e-volta é aproximadamente na ordem dos 0.130 ms.

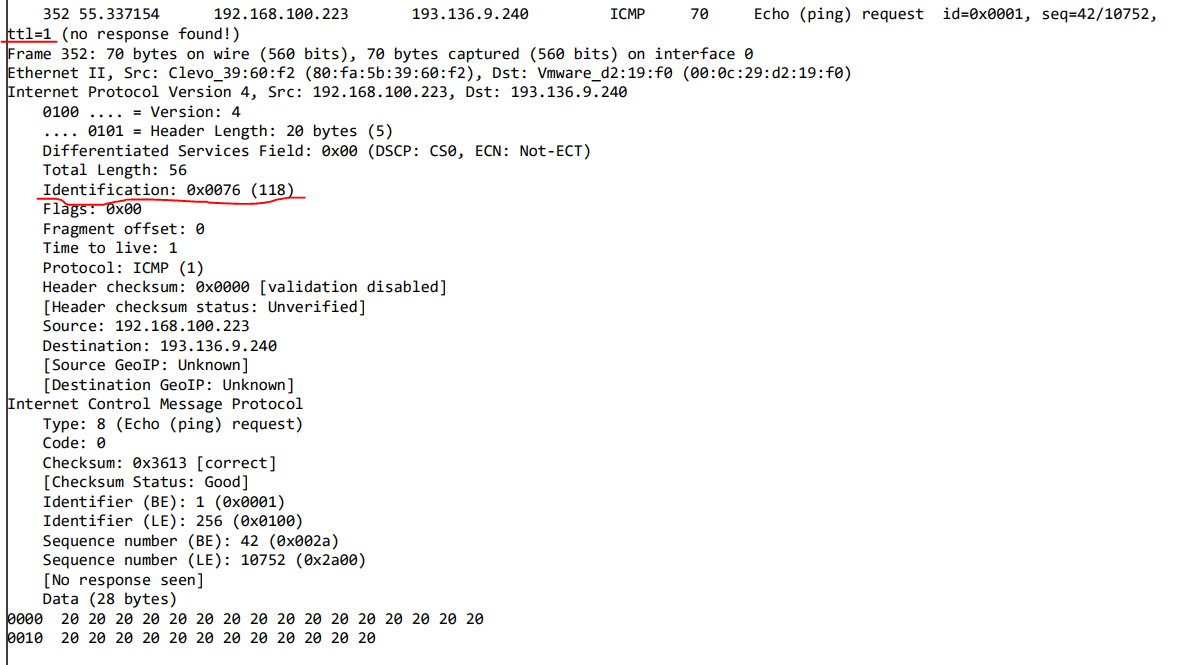
Exercício 2.

**a.** O endereço IP da interface ativa é 192.168.100.223. Este valor pode ser confirmado pela **Fig. 3.** Este endereço corresponde ao endereço IP Source (Src).

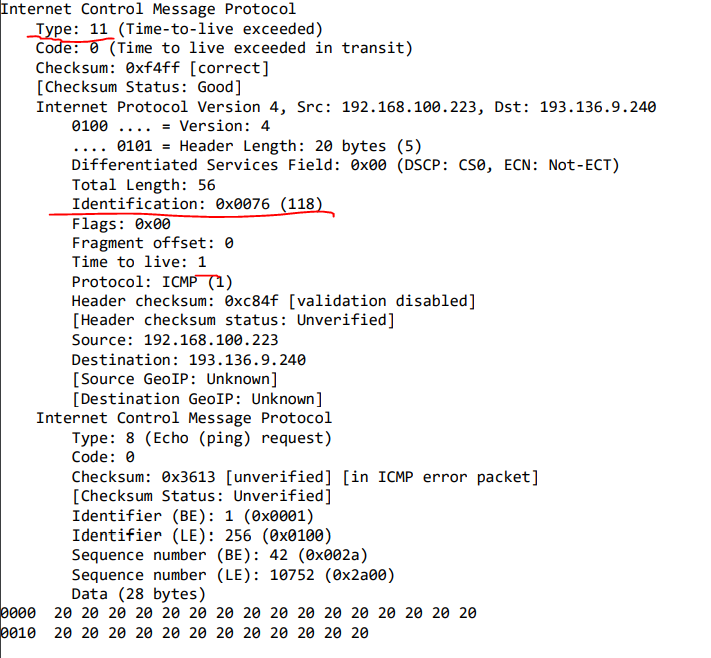
**b.** O valor do campo do protocolo ICMP é igual a 1 **(definido pelo RFC 792)** como consta nas figuras relativas a este exercício**.** Este protocolo identifica uma mensagem, ou seja, informação relativa ao que se sucede no decorrer do envio dos pacotes na rede. Note-se que esta mensagem é identificada por um tipo e por um código e ainda que estes dois alteram-se quando se consegue finalmente obter uma reposta por parte do destino do pacote.

**c.** O cabeçalho IP(v4) tem 20 bytes de tamanho. O campo de dados do datagrama corresponde ao tamanho total do pacote subtraído ao tamanho do seu cabeçalho. Sendo assim, como o pacote tem um tamanho total de 56 bytes, o campo de dados (payload) é igual a 36 bytes.

**d.** O datagrama IP não foi fragmentado uma vez que o Fragment Offset é igual 0 e o bit que indica a existência de More Fragments também **(RFC 791).**

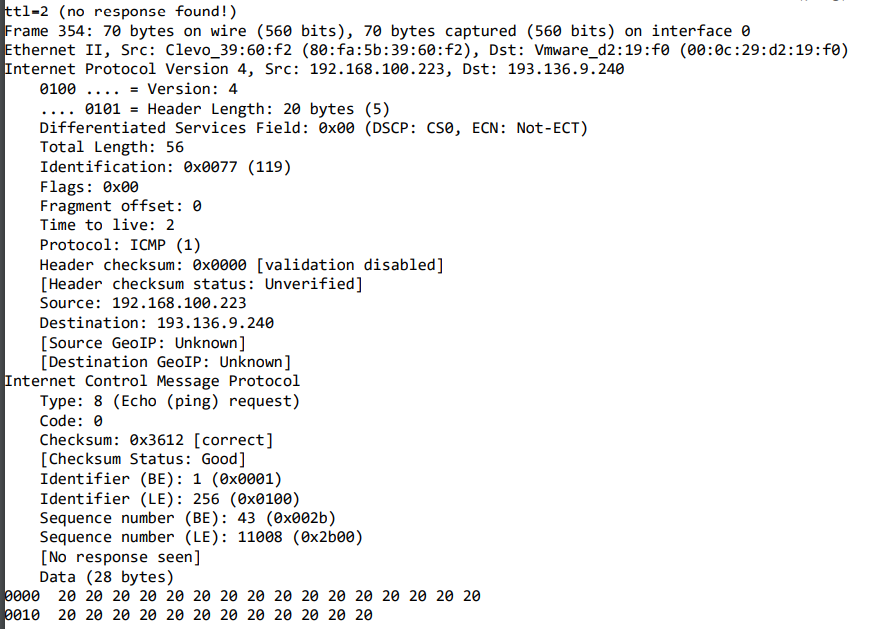
 **e.** O primeiro pacote é enviado com um TTL = 1 sendo a sua identificação = 118. O tipo da mensagem ICMP deste primeiro pacote é 8, o que significa que é um Echo Request **(RFC 792),** conforme indicado na figura abaixo.

Em resposta ao envio deste primeiro pacote obtemos uma nova mensagem ICMP. Desta vez, esta mensagem é do tipo 11 que nos informa que o pacote excedeu o tempo de vida antes mesmo de alcançar o seu destino **(RFC 792).** Como a identificação desta mensagem é igual à identificação da mensagem do pacote inicial, apuramos que se trata de uma resposta a este envio.

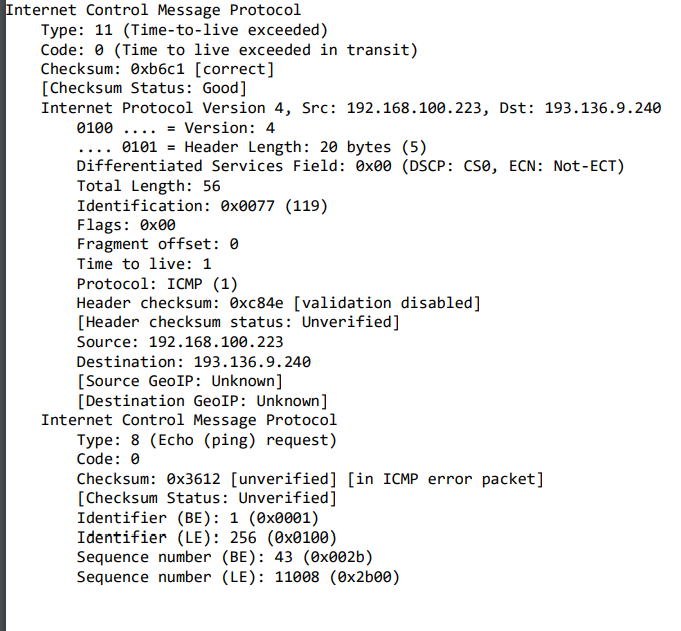


Como aprendemos e constatamos anteriormente, quando o TTL não é suficiente para que um determinado pacote chegue ao seu destino, então, o mesmo será aumentado até que consigamos obter uma resposta. Assim sendo, o TTL passa agora a ser = 2, continuando a ser insuficiente.

Notamos que a identificação do pacote mudou de 118 para 119, mas, o tipo da mensagem ICMP manteve-se o mesmo - Echo Request **(RFC 792).**

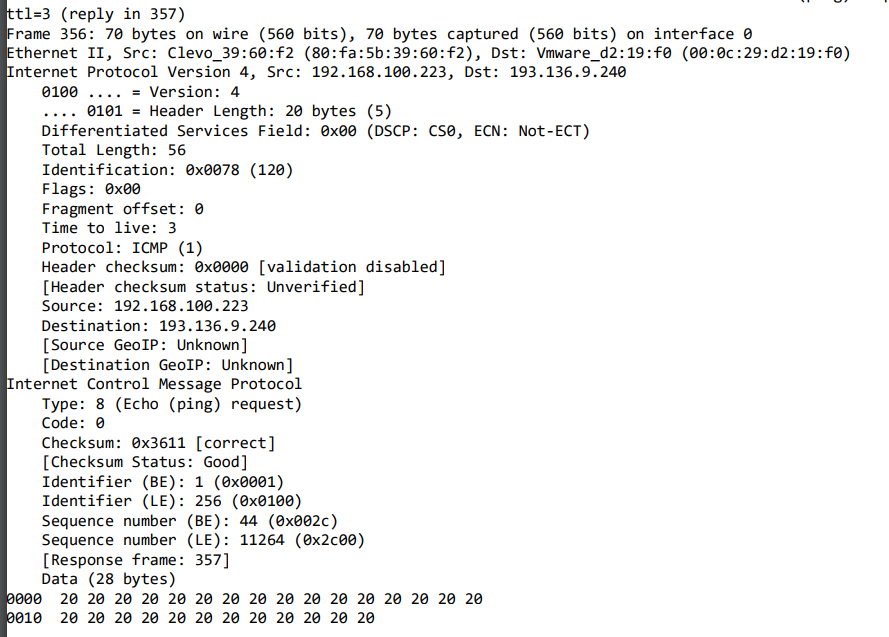


Em resposta ao envio deste pacote obtemos novamente uma mensagem ICMP do tipo 11, com aconteceu anteriormente – Time to live exceded **(RFC 792).**



Com TTL = 3 conseguimos finalmente obter uma resposta. O tipo de mensagem ICMP continua a ser 8, ou seja, Echo Request. O único parâmetro que variou foi novamente a identificação do pacote, que passou a ser 120.

Note-se que, logo na primeira linha vemos a seguinte frase “reply in 357”.

****

**f.** A identificação no request e no ICMP do reply é igual. Deduzimos por isso que, esta é uma forma de saber qual reply corresponde a cada request.

**g.** Não configuramos os TTL dos routers. Os TTLs podem ser diferentes de router para router.

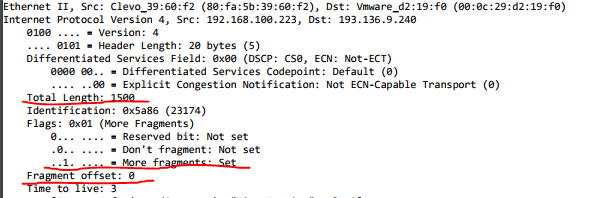
Exercício 3.

**Nota – o número do nosso grupo é 47, mas por engano fizemos ping com o número 27, tamanho do pacote 3027.**

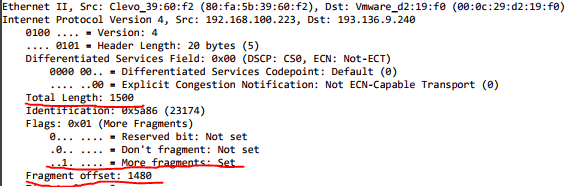
**a.** O tamanho do nosso pacote é de 3027 bytes. Como o tamanho total do pacote é de 1500 bytes **(MTU – RFC 791)**, é necessário existir a fragmentação do nosso pacote inicial em vários pacotes de forma a que o router de destino seja capaz de os receber, e, posteriormente reconstruir.

**b.** A informação no cabeçalho que nos indica que o datagrama foi de facto fragmentado são as Flags. Nelas, constatamos que o bit que indica a existência de ***More fragments*** se encontra a 1. Já a informação que indica que se trata do primeiro fragmento é o facto do ***Fragment offset*** se encontrar a 0 **(RFC 791).**

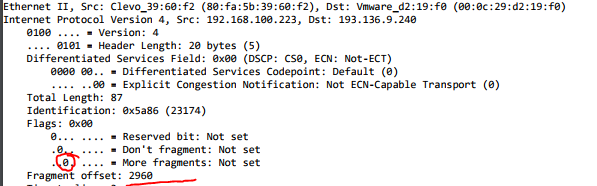
O tamanho deste datagrama IP é de 1500 bytes, sendo 20 deles reservados ao cabeçalho.



**c.** A informação que nos indica que não se trata do primeiro fragmento, e como averiguamos acima, é o ***Fragment offset***. Como o ***Fragment offset*** deixou de ser 0 então, este não é o primeiro fragmento. Como o bit ***More fragments*** se encontra a 1 então, ainda há mais fragmentos para alem destes.

****

**d.** Foram criados 3 fragmentos a partir do datagrama original. Sabemos que o fragmento representado na imagem abaixo anexada é o último, uma vez que, o bit que indica a possibilidade de existência de ***More fragments*** se encontra a 0.



**e.** Os campos que mudam no cabeçalho IP entre os diferentes fragmentos são:

1. O comprimento de cada um dos pacotes;
2. O bit ***More fragments*** que indica a existência ou não, de mais pacotes a seguir ao pacote em questão;
3. O ***Fragment offset*** que é 0 quando se trata do pacote inicial e vai variando de pacote para pacote, indicando onde cada fragmento pertence no datagrama **(RFC 791)**.

A identificação mantém-se igual em todos os pacotes, tal como seria de esperar, dando-nos assim a indicação de que estamos a lidar com “sub-pacotes” referentes ao pacote inicialmente repartido.

**Combinando esta informação é possível saber quando e em que ordem se reconstrói o pacote.**

------------------------------------

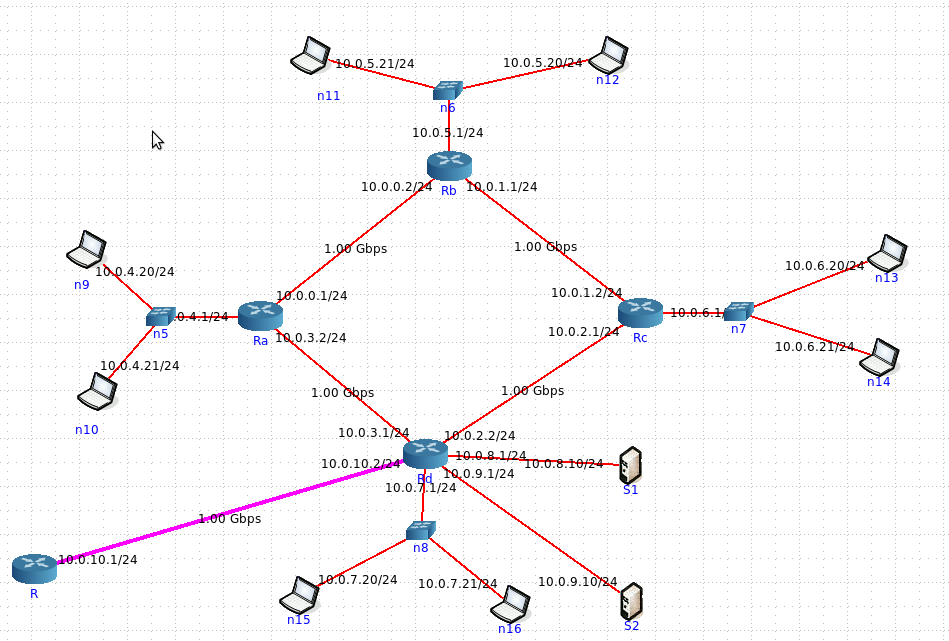
**Questões e Respostas**

Parte II

Exercício 1.

**a.** Os primeiros 24 bits de cada endereço IP são identificadores de rede, não podendo por isso ser alterados. Isto verifica-se porque o prefixo é 24, ou seja, a máscara de rede atribuída pelo core é 255.255.255.0.

Os outros restantes 8 bits são os bits disponíveis para os hosts, ou seja, de 0 a 255 (excetuando estes 2 dado que estes dois são reservados para os endereços de rede e broadcast, respetivamente).



**b.** Tratam-se de endereços privados porque estão dentro da gama de endereços privados definida segundo o standard de rede **RFC1918**.

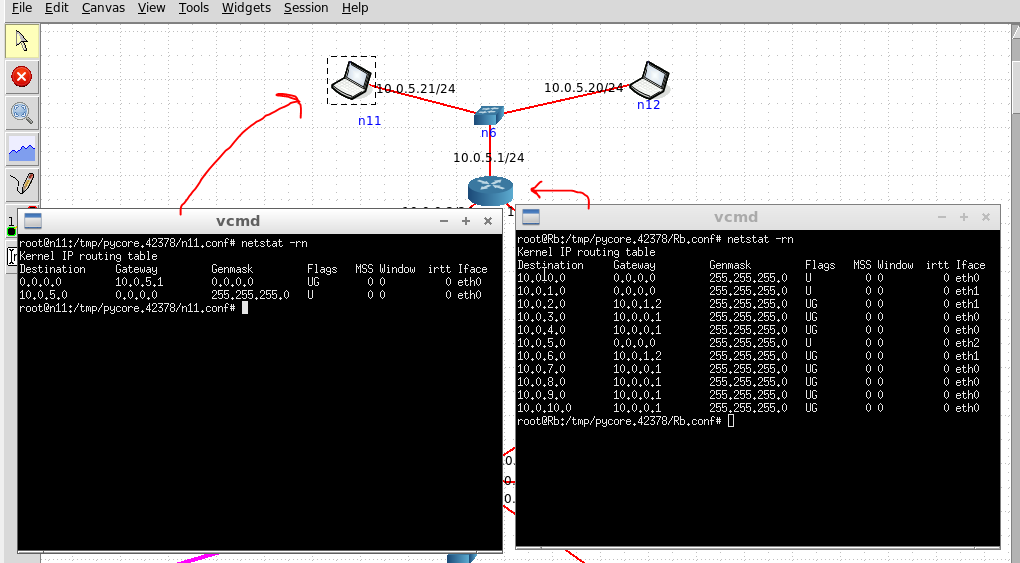
**c.** O comutador (switch) não necessita de ter um endereço IP para fazer o seu trabalho que nada mais é que redirecionar tráfego na rede.

**d,e.** Verificamos que existe conectividade entre os equipamentos da nossa rede simulada no CORE.

Exercício 2.

**a.** Para o laptop n11, a tabela de endereçamento mostra apenas duas entradas. A primeira, com Destination 0.0.0.0 significa que para contactar qualquer equipamento, envia a informação para o router com endereço 10.0.5.1.

A segunda entrada tem Gateway 0.0.0.0, o que indica que o laptop se encontra fisicamente ligado a esse equipamento. A Destination é 10.0.5.0, o que significa que para contactar qualquer equipamento da rede local (10.0.5.X), o laptop envia informação para a Gateway com endereço 0.0.0.0, através da interface eth0.



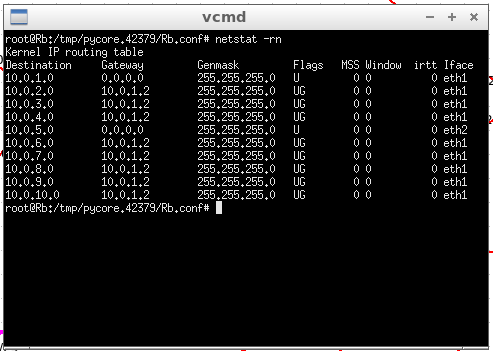
Para a tabela de endereçamento do router, sabemos que está ligado fisicamente a 3 redes, com Gateway 0.0.0.0.

A título de exemplo, para contactar um equipamento da rede 10.0.4.0/24, o router tem na tabela na coluna Gateway o endereço 10.0.0.1, que é o router da rede em questão. Então para contactar o laptop n9, por exemplo, o router apenas envia a informação para o router da rede 10.0.4.0/24 e este tem na tabela de endereçamento o endereço do laptop de destino.

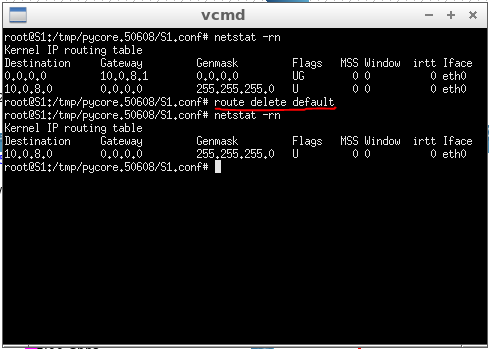
Para contactar um laptop da rede local o router apenas envia informação para o Gateway com endereço 0.0.0.0 através da interface eth2, que é um dos equipamentos ligados fisicamente, e pertencente à rede local.

**b.** Apagamos o link entre o router Rb e o router Ra, que tem rede local 10.0.4.0/24. Sendo assim, a ligação através do Gateway 10.0.0.1 foi perdida e para contactar o laptop n9 tem agora de ser feita uma ligação através do router Rc – efetivamente, para contactar a Destination com endereço 10.0.4.0 o endereço do Gateway é o do router Rc, 10.0.1.2. A partir daí o Rc terá a sua própria tabela de endereçamento, que envia o pacote para Rd, que envia para Ra.

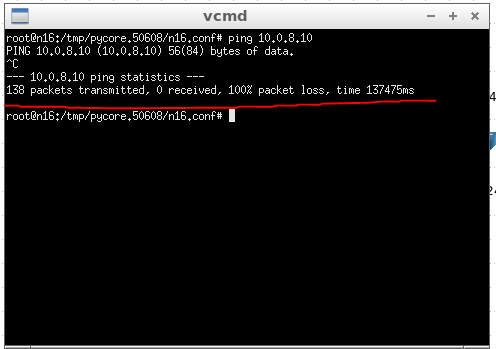
Concluímos assim que a tabela de endereçamento foi automaticamente ajustada à mudança e, portanto, sabemos que existe endereçamento dinâmico.



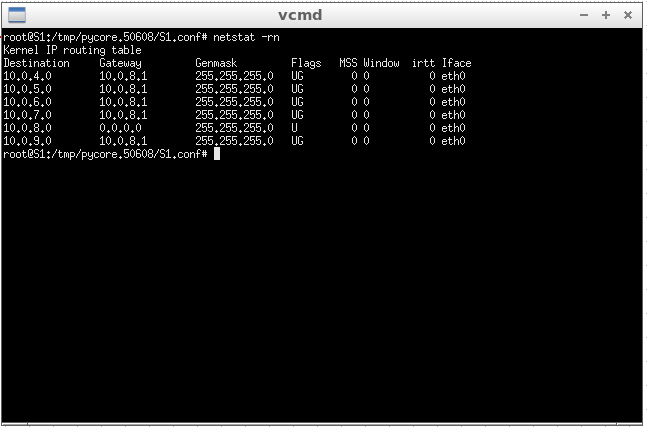
**c.** Eliminando a route default o servidor S1 só “sabe” contactar o router a que está fisicamente ligado. Tentativas de contactar outros laptops ou routers retornaram a mensagem “Network is unreachable”.



Para os outros equipamentos, utilizando o comando para fazer ping ao servidor S1 observamos que pudemos enviar informação, mas S1 não consegui responder – obtivemos 100% de perda de pacotes.



**d,e.** Após a configuração das novas rotas estáticas a tabela de endereçamento de S1 é a seguinte:



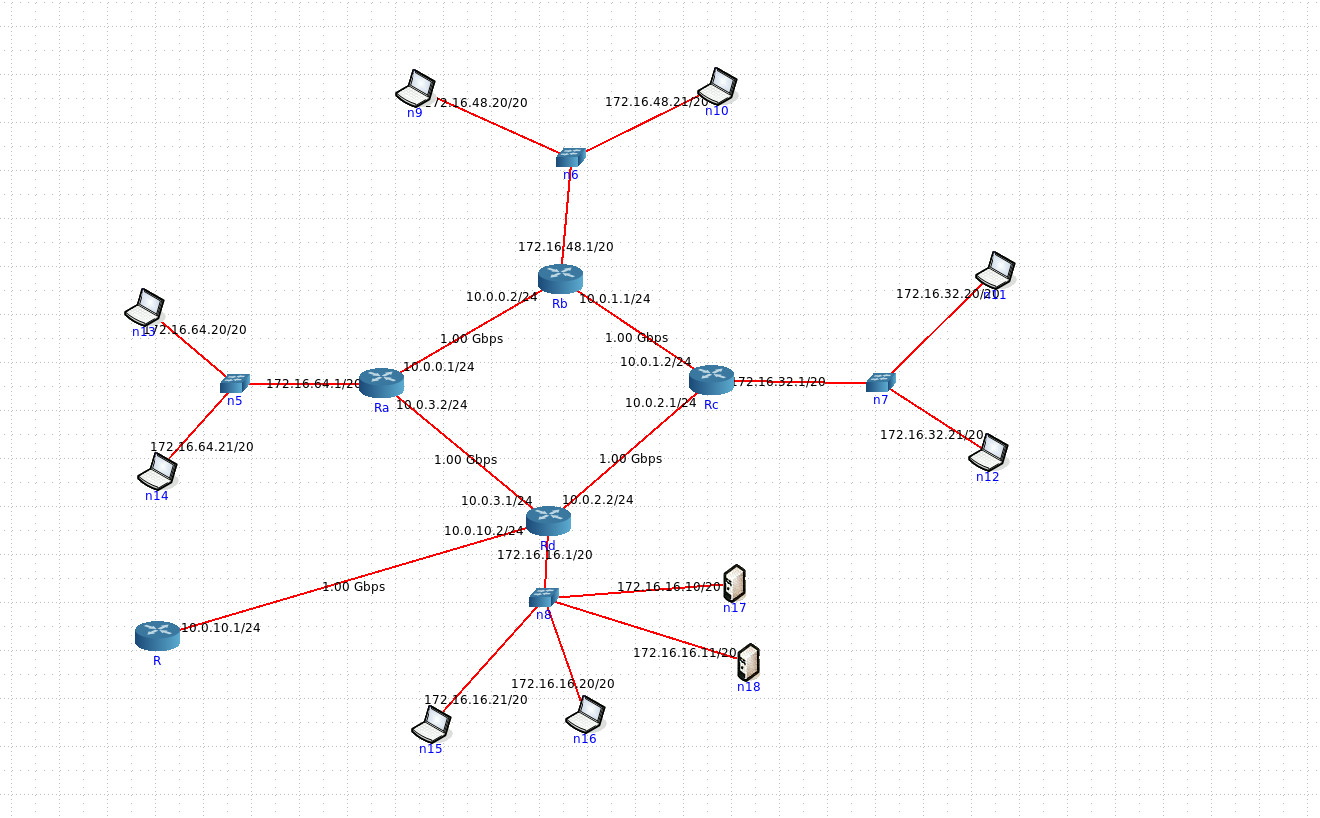
Exercício 3.

**1.** Decidimos usar 4 bits dos 16 bits disponíveis para subredes. Assim, acabamos por ter a possibilidade de suportar subredes e consequentemente, hosts por cada subrede. Como tipicamente uma Universidade possui mais dispositivos conectados à rede do que redes propriamente ditas, achamos que oferecer mais espaço para hosts era uma escolha acertada.

Neste caso possuímos 4 subredes e usamos os endereços 172.16.16.0/20, 172.16.32.0/20, 172.16.48.0/20 e 172.16.64.0/20, sendo cada um para uma subrede diferente.

De referir que, inicialmente atribuímos manualmente um endereço IP para cada laptop conectado à subrede. Por exemplo: na subrede 172.16.48.0/20 atribuímos os endereços 172.16.48.2/20 e 172.16.48.3/20 para os dois laptops conectados a essa rede e 172.16.48.1/20 para a interface do router.

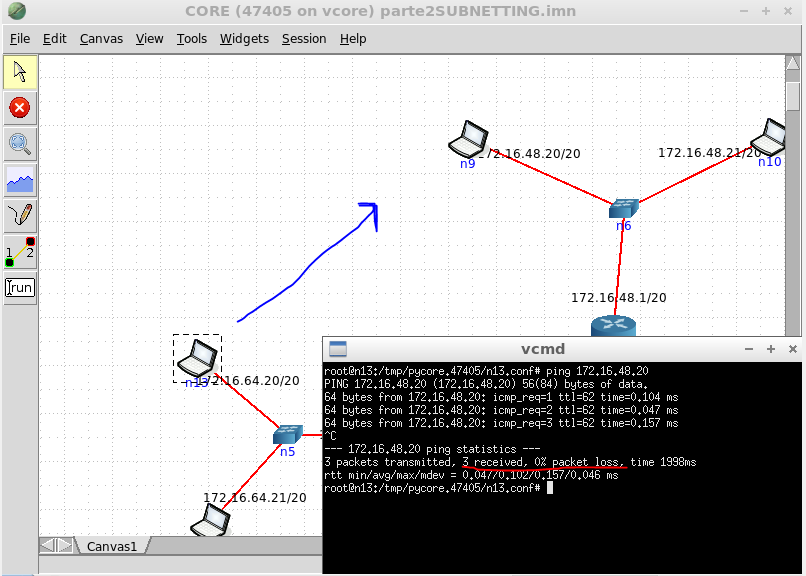
No entanto, acabamos por achar mais fácil deixar o CORE atribuir os endereços para cada host automaticamente.



**2.** A netmask que usamos foi 255.255.240.0.

Podemos interligar em cada departamento 4094 hosts IP. Note-se que, temos 12 bits por cada subrede reservados para hosts. Assim, excluindo os endereços de broadcast e de identificação, ficamos com endereços disponíveis para hosts, por cada subrede.

**3.** Somos capazes de verificar que a conectividade IP entre as várias redes locais da organização MIEI-RC é mantida, tal como a imagem abaixo anexada o comprova.



**Conclusões**

Após concluirmos a realização desta ficha prática que teve como objetivo base fortificar os nossos conhecimentos acerca de todo o Protocolo IP podemos tecer alguns comentários que nos parecem ser importantes.

Relativamente à primeira parte, o grupo não sentiu grande dificuldade. Consideramos assim que, a compreensão de como funciona o TTL, as mensagens ICMP e os seus diferentes tipos e por fim, a informação do cabeçalho no envio dos diversos pacotes ficou perfeitamente compreendida.

Onde sofremos mais dificuldade foi mesmo na segunda parte, especificamente nas sub-redes (subnetting). Ainda de referir que, nesta parte, no exercício 3 de subredes, começamos por atribuir um endereço IP a cada laptop manualmente acabando por nos aperceber que o próprio CORE atribuía endereços automaticamente caso os do router estivessem devidamente configurados, o que acabou por nos simplificar o trabalho.

Achamos ainda importante referenciar que a realização desta ficha prática foi suportada pela bibliografia que consideramos ter sido uma mais valia não só para a correta confirmação dos nossos palpites de resposta, mas também para uma aperfeiçoada consolidação da matéria.