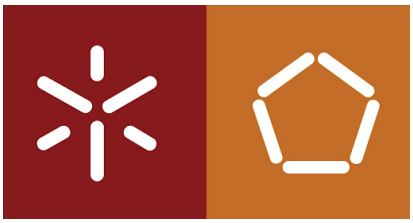

Universidade do Minho

Licenciatura/Mestrado Integrado em Engenharia Informática

Redes de Computadores - TP3

Grupo 94

Rui Guilherme Monteiro (A93179) Rui Moreira (A93232)
José Pereira (A89596)

Ano Lectivo 2021/2022

Conteúdo

Capítulo 1

Questões e Respostas

1.1 Captura e análise de Tramas Ethernet

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.26.26.189	66.102.1.188	TCP	54	50403 → 5228 [ACK] Seq=1 Ack=1 Win=2048 Len=0
2	0.042255	66.102.1.188	172.26.26.189	TCP	54	5228 → 50403 [RST] Seq=1 Win=0 Len=0
3	1.027374	172.26.26.189	193.137.16.65	DNS	79	Standard query 0x912d A elearning.uminho.pt
4	1.027543	172.26.26.189	193.137.16.65	DNS	79	Standard query 0xb6e0 Unknown (65521) elearning.uminho.pt
5	1.035207	193.137.16.65	172.26.26.189	DNS	351	Standard query response 0x912d A elearning.uminho.pt A 193.137.9.150 NS ns02.fccn.pt NS dns.uminho.pt NS
6	1.035452	193.137.16.65	172.26.26.189	DNS	133	Standard query response 0xb6e0 Unknown (65521) elearning.uminho.pt SOA dns.uminho.pt
7	1.035643	172.26.26.189	193.137.9.150	TCP	78	50464 → 80 [SYN, ECN, CWR] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=3360594319 TSecr=0 SACK_PERM=1
8	1.035843	172.26.26.189	193.137.9.150	TCP	78	50465 → 80 [SYN, ECN, CWR] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=903191062 TSecr=0 SACK_PERM=1
9	1.037428	193.137.9.150	172.26.26.189	TCP	74	80 → 50464 [SYN, ACK, ECN] Seq=0 Ack=1 Win=12500 Len=0 MSS=1250 WS=4 SACK_PERM=1 TSval=1822463849 TSecr=3
10	1.037464	172.26.26.189	193.137.9.150	TCP	66	50464 → 80 [ACK] Seq=1 Ack=1 Win=131200 Len=0 TSval=3360594321 TSecr=1822463849
11	1.037577	172.26.26.189	193.137.9.150	HTTP	532	GET / HTTP/1.1
12	1.038068	193.137.9.150	172.26.26.189	TCP	74	80 → 50465 [SYN, ACK, ECN] Seq=0 Ack=1 Win=12500 Len=0 MSS=1250 WS=4 SACK_PERM=1 TSval=1822463850 TSecr=9
13	1.038099	172.26.26.189	193.137.9.150	TCP	66	50465 → 80 [ACK] Seq=1 Ack=1 Win=131200 Len=0 TSval=903191064 TSecr=1822463850
14	1.040518	193.137.9.150	172.26.26.189	HTTP	198	HTTP/1.0 302 Moved Temporarily
15	1.040572	172.26.26.189	193.137.9.150	TCP	66	50464 → 80 [ACK] Seq=467 Ack=133 Win=131072 Len=0 TSval=3360594324 TSecr=1822463852
16	1.042916	172.26.26.189	193.137.9.150	TCP	78	50466 → 443 [SYN, ECN, CWR] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=3671981760 TSecr=0 SACK_PERM=1
17	1.044852	193.137.9.150	172.26.26.189	TCP	78	443 → 50466 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1250 SACK_PERM=1 WS=8 TSval=749552263 TSecr=367198
18	1.044888	172.26.26.189	193.137.9.150	TCP	66	50466 → 443 [ACK] Seq=1 Ack=1 Win=131200 Len=0 TSval=3671981762 TSecr=749552263
19	1.045035	172.26.26.189	193.137.9.150	TLSv1.2	583	Client Hello
20	1.046370	193.137.9.150	172.26.26.189	TCP	66	[TCP Window Update] 443 → 50466 [ACK] Seq=1 Ack=1 Win=262144 Len=0 TSval=749552263 TSecr=3671981762
21	1.062177	193.137.9.150	172.26.26.189	TCP	66	443 → 50466 [ACK] Seq=1 Ack=518 Win=262144 Len=0 TSval=749552263 TSecr=3671981762
22	1.079975	193.137.9.150	172.26.26.189	TLSv1.2	152	Server Hello
23	1.080057	172.26.26.189	193.137.9.150	TCP	66	50466 → 443 [ACK] Seq=518 Ack=87 Win=131136 Len=0 TSval=3671981797 TSecr=749552263
24	1.080172	193.137.9.150	172.26.26.189	TLSv1.2	72	Change Cipher Spec
25	1.080172	193.137.9.150	172.26.26.189	TLSv1.2	111	Encrypted Handshake Message
26	1.080206	172.26.26.189	193.137.9.150	TCP	66	50466 → 443 [ACK] Seq=518 Ack=138 Win=131072 Len=0 TSval=3671981797 TSecr=749552263
27	1.080337	172.26.26.189	193.137.9.150	TLSv1.2	117	Change Cipher Spec, Encrypted Handshake Message
28	1.080461	172.26.26.189	193.137.9.150	TLSv1.2	789	Application Data
29	1.082685	193.137.9.150	172.26.26.189	TCP	66	443 → 50466 [ACK] Seq=138 Ack=1292 Win=262144 Len=0 TSval=749552263 TSecr=3671981797
30	1.117695	193.137.9.150	172.26.26.189	TLSv1.2	922	Application Data
31	1.117696	193.137.9.150	172.26.26.189	TLSv1.2	1252	Application Data
32	1.117797	172.26.26.189	193.137.9.150	TCP	66	50466 → 443 [ACK] Seq=1292 Ack=2180 Win=129024 Len=0 TSval=3671981835 TSecr=749552263
33	1.118099	193.137.9.150	172.26.26.189	TCP	1304	443 → 50466 [ACK] Seq=2180 Ack=1292 Win=262144 Len=1238 TSval=749552263 TSecr=3671981797 [TCP segment of
34	1.118101	193.137.9.150	172.26.26.189	TLSv1.2	841	Application Data

Figura 1.1: Pacotes capturados pelo *wireshark*

1.1.1 Exercício 1

Anote os endereços MAC de origem e de destino da trama capturada

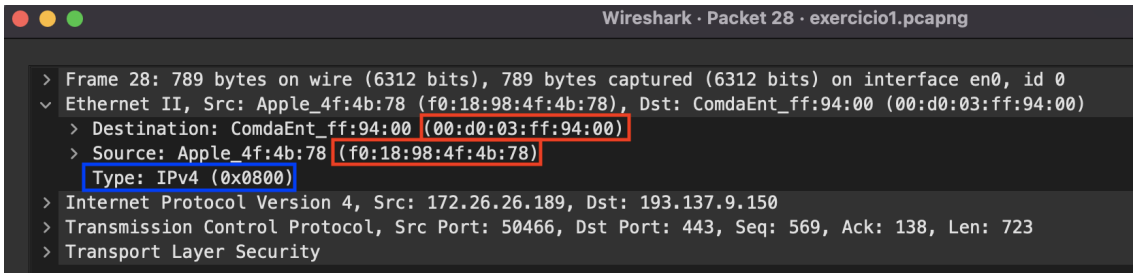

Figura 1.2: Trama Ethernet que contém a mensagem de acesso ao servidor (HTTP GET encriptada)

Os endereços MAC de origem e de destino da trama capturada são, respetivamente, **f0:18:98:4f:4b:78** e **00:d0:03:ff:94:00**.

1.1.2 Exercício 2

Identifique a que sistemas se referem. Justifique.

O endereço MAC de origem refere-se à máquina nativa que foi utilizado para aceder e o endereço MAC de destino refere-se ao router da rede local a qual a máquina nativa está ligado.

1.1.3 Exercício 3

Qual o valor hexadecimal do campo *Type* da trama *Ethernet*? O que significa?

Observando a Figura 1.2, o valor hexadecimal do campo *Type* é **0x0800**, que significa que o protocolo de camada superior utilizado é IPv4.

1.1.4 Exercício 4

Quantos *bytes* são usados no encapsulamento protocolar, i.e. desde o início da trama até ao início dos dados do nível aplicacional (*Application Data Protocol*: *http-over-tls*)? Calcule e indique, em percentagem, a sobrecarga (*overhead*) introduzida pela pilha protocolar.

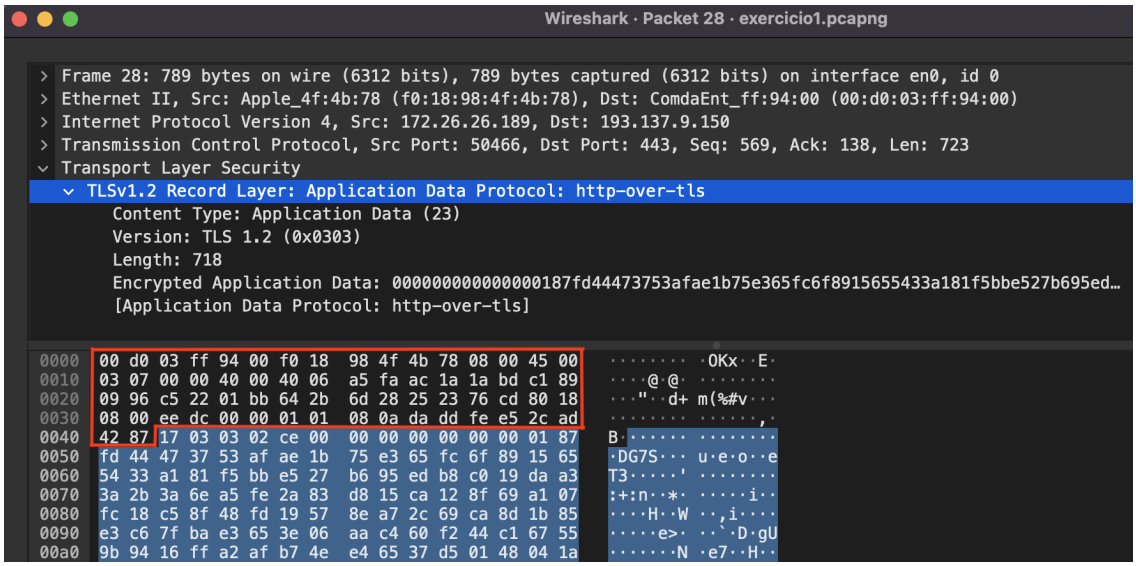

Figura 1.3: Trama

Como é possível observar na Figura 1.3, os dados do nível aplicacional são os que estão sublinhados a azul, logo são usados 66 *bytes* no encapsulamento protocolar (rodeado a vermelho).

Como a trama tem um comprimento total de 789 *bytes* , temos então um *overhead* de (66/789) * 100 = 8.4%.

1.1.5 Exercício 5

Qual é o endereço *Ethernet* da fonte? A que sistema de rede corresponde? Justifique.

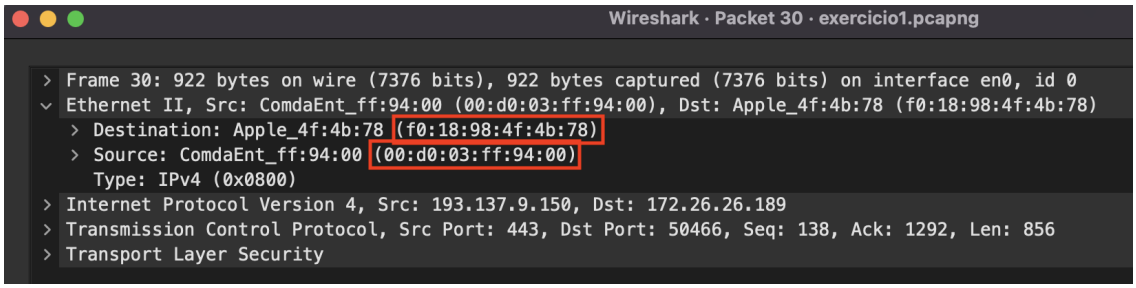

Figura 1.4: Trama Ethernet que contém o primeiro byte da resposta HTTP proveniente do servidor.

Como podemos verificar pelo campo *Source* da Figura 1.5, o endereço *Ethernet* da fonte é **00:d0:03:ff:94:00**, que corresponde ao *default gateway* da rede local à qual estamos conectados. Isto porque, visto que o servidor *elearning.uminho.pt* não se encontra na rede local, não sendo alcançável pelo nosso computador, então as mensagens são trocadas com o *default gateway*, ao invés de ser diretamente com o servidor.

1.1.6 Exercício 6

Qual é o endereço MAC do destino? A que sistema corresponde?

Observando o campo *Destination* da imagem 1.5, verifica-se que o endereço MAC destino é **f0:18:98:4f:4b:78**, que se refere à máquina nativa.

1.1.7 Exercício 7

Atendendo ao conceito de desencapsulamento protocolar, identifique os vários protocolos contidos na trama recebida.

Observado a figura 1.5, verificámos que os protocolos usados são: *Ethernet* II, *Internet Protocol Version* 4 (**IPv4**), *Transmission Control Protocol* (**TCP**) e *Hyper Text Transfer Protocol Secure* (**HTTPS**).

1.2 Protocolo ARP

1.2.1 Exercício 8

Observe o conteúdo da tabela ARP. Diga o que significa cada uma das colunas.

```
ruimoreira@MacBook-Pro-de-Rui ~ % arp -a
? (172.26.254.254) at 0:d0:3:ff:94:0 on en0 ifscope [ethernet]
? (172.26.255.255) at ff:ff:ff:ff:ff:ff on en0 ifscope [ethernet]
? (224.0.0.251) at 1:0:5e:0:0:fb on en0 ifscope permanent [ethernet]
? (239.255.255.250) at 1:0:5e:7f:ff:fa on en0 ifscope permanent [ethernet]
```

Figura 1.5: Tabela ARP

A primeira coluna apresenta do endereço IP do *host*, e a segunda coluna, entre o *at* e o *on*, indica os endereços MAC correspondentes. Já a última coluna indica a interface relativa ao envio do pacote e se a entrada na tabela é permanente.

1.2.2 Exercício 9

Qual é o valor hexadecimal dos endereços origem e destino na trama *Ethernet* que contém a mensagem com o pedido ARP (ARP *Request*)? Como interpreta e justifica o endereço destino usado?

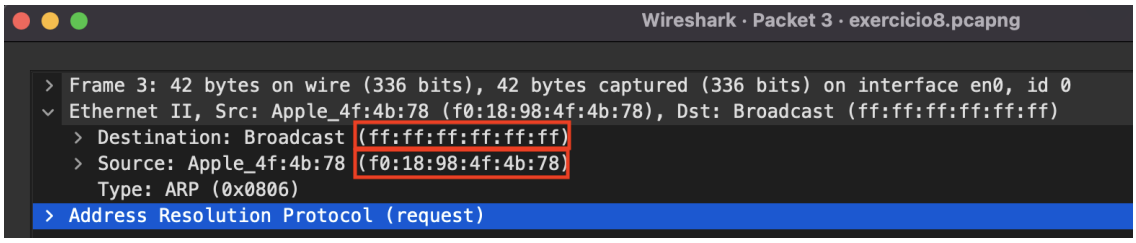

Figura 1.6: Trama *Ethernet* que contém o ARP *Request*

O valor hexadecimal do endereço origem é **f0:18:98:4f:4b:78** e do endereço destino é **ff:ff:ff:ff:ff:ff**.

Visto que a tabela de endereçamento ainda não tinha uma entrada para o endereço MAC correspondente ao endereço IP ao qual era destinado, utilizou como endereço destino o endereço de *broadcast*. Isto significa que vai enviar para todas as interfaces e esperar uma resposta da máquina destino com o seu endereço MAC. Assim que receber a resposta, adiciona este valor à tabela ARP.

1.2.3 Exercício 10

Qual o valor hexadecimal do campo tipo da trama *Ethernet*? O que indica?

Observando a Figura 1.6, o valor hexadecimal do campo tipo da trama *Ethernet* é **0x0806** que indica que se trata de *Address Resolution Protocol* (ARP).

1.2.4 Exercício 11

Como pode confirmar que se trata efetivamente de um pedido ARP? Identifique que tipo de endereços estão contidos na mensagem ARP? Que conclui?

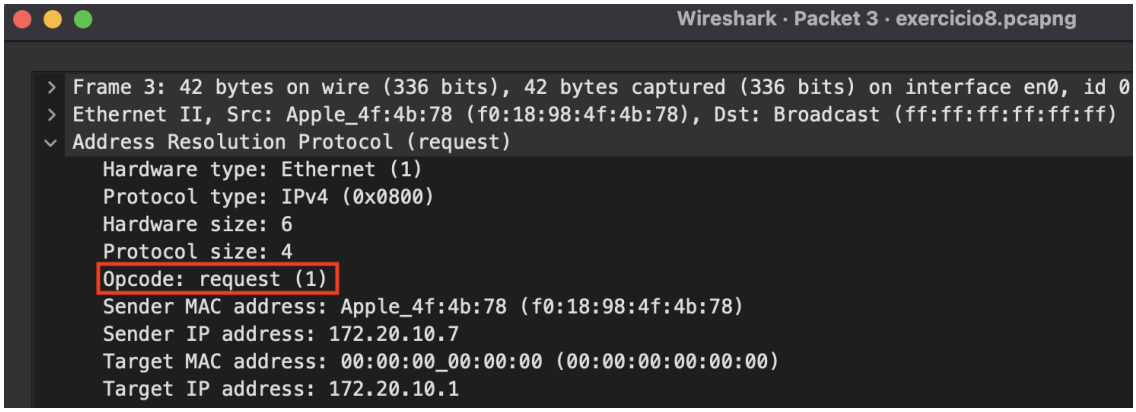

Figura 1.7: Mensagem ARP

Como observamos na figura 1.7, o campo *Opcode* tem o valor 1, logo trata-se de um pedido

(*request*).

Na mensagem ARP estão contidos os endereços do tipo IP e MAC, tanto da origem como do destino.

1.2.5 Exercício 12

Explicite que tipo de pedido ou pergunta é feita pelo *host* de origem.

```
  3 0.741072  Apple_4f:4b:78  Broadcast       ARP             42 Who has 172.20.10.1? Tell 172.20.10.7
```

Figura 1.8: Pergunta feita pelo *host* de origem

"Quem tem 172.20.10.1? Diga a 172.20.10.7"

A máquina origem pretende saber quem tem o endereço 172.20.10.1, então pergunta a todos os *hosts* qual deles tem esse endereço IP. Se um desses *hosts* o tiver, pede para que este envie uma resposta para o endereço IP 172.20.10.7, de onde iremos obter o endereço MAC.

1.2.6 Exercício 13

Localize a mensagem ARP que é a resposta ao pedido ARP efetuado.

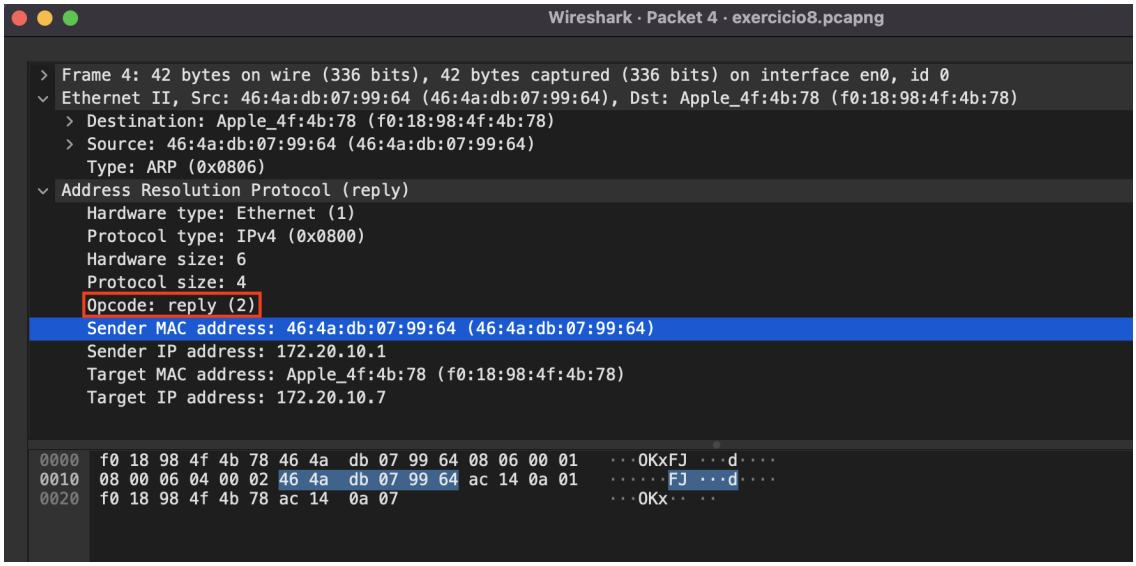

Figura 1.9: Mensagem ARP de resposta ao pedido efectuado

Exercício 13.a)

Qual o valor do campo ARP *opcode*? O que especifica?

Observando a figura 1.9, o valor do campo ARP *opcode* é ***reply*(2)**.

Exercício 13.b)

Em que campo da mensagem ARP está a resposta ao pedido ARP?

Como podemos ver pela Figura 1.9, a resposta ao pedido ARP está no campo *Sender MAC address*.

1.2.7 Exercício 14

Na situação em que efetua um *ping* a outro *host*, assuma que este está diretamente ligado ao mesmo *router*, mas noutra subrede, e que todas as tabelas ARP se encontram inicialmente vazias. Esboce um diagrama em que indique claramente, e de forma cronológica, todas as mensagens ARP e ICMP trocadas, até à recepção da resposta ICMP do *host* destino

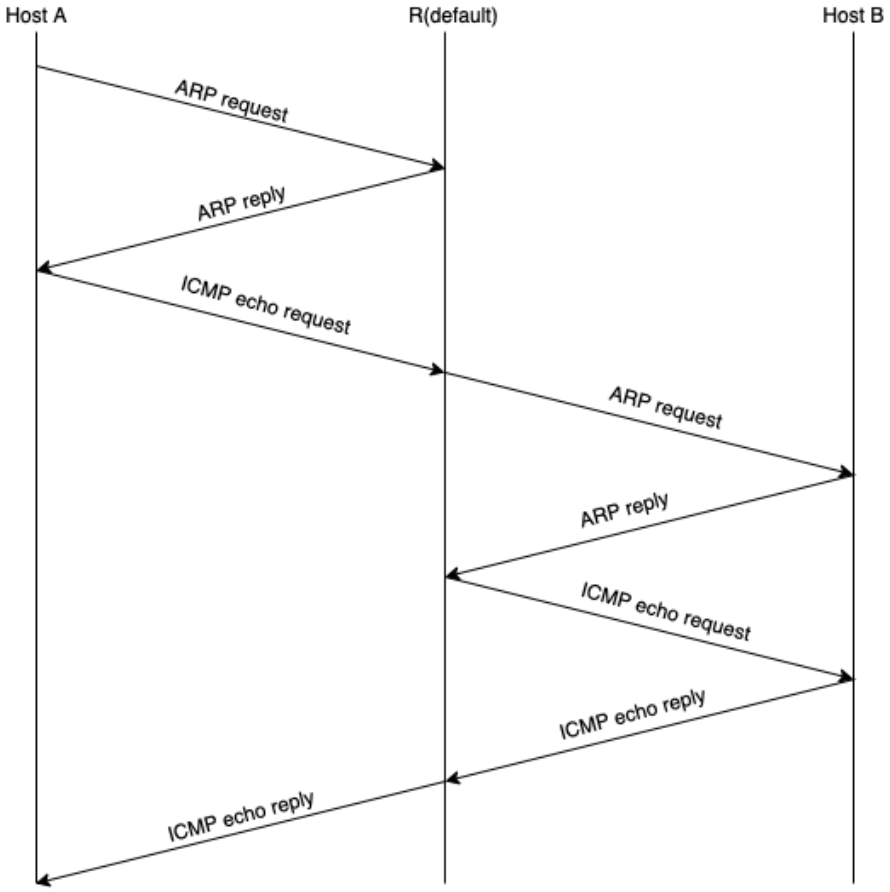

Figura 1.10: Diagrama temporal com todas as mensagens ARP e ICMP trocadas, até à recepção da resposta ICMP do *host* destino

1.3 Domínios de colisão

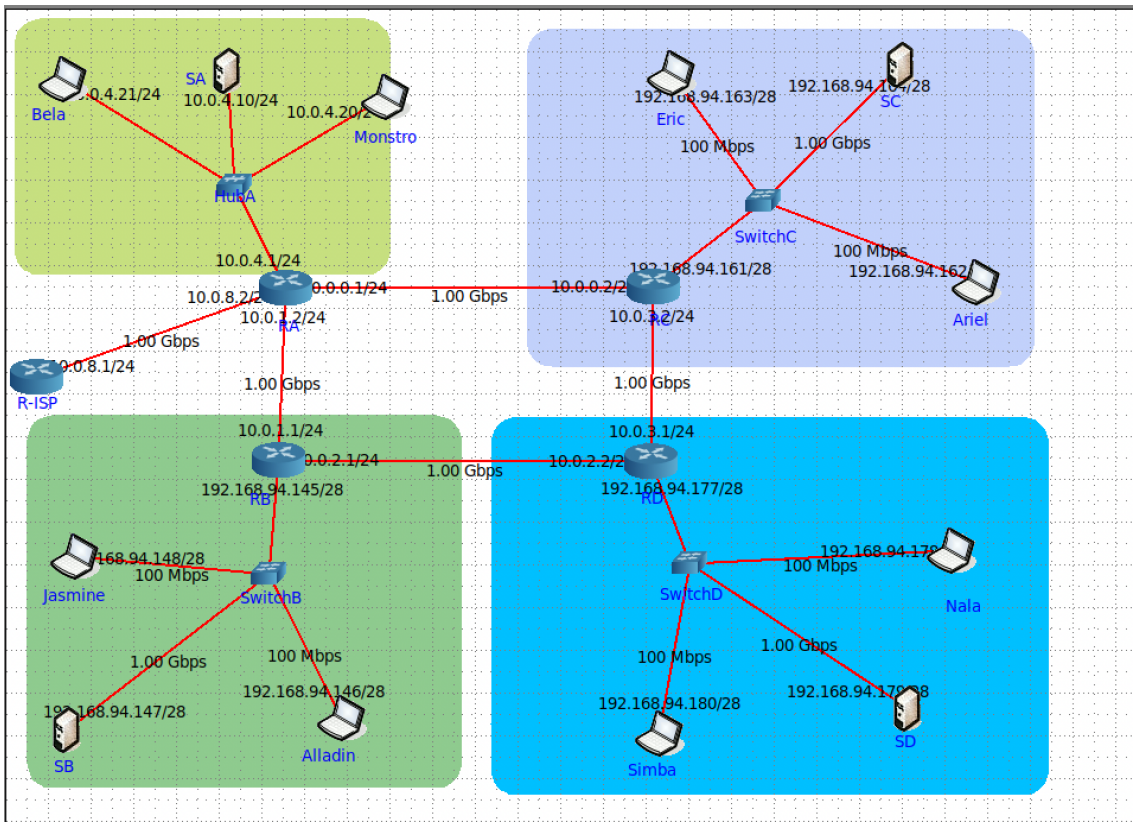

Figura 1.11: Topologia do TP2 com a substituição do *switch* do Dep.A por um *hub* (repetidor)

1.3.1 Exercício 15

Através da opção *tcpdump* verifique e compare como flui o tráfego nas diversas interfaces do dispositivo de interligação no departamento A (LAN partilhada) e no departamento B (LAN comutada) quando se gera tráfego intra-departamento (por exemplo, fazendo *ping* IPaddr da Bela para Monstro, da Jasmine para o Alladin, etc.) Que conclui?

De maneira a comparar o funcionamento dos *switches* e dos *hubs*, substituiu-se no Departamento A o *switch* por um *hub*. Neste departamento (LAN partilhada), executou-se o comando *ping* do computador Bela para o computador Monstro estando o servidor A(SA) a executar o comando *tcpdump*. Como se pode observar na seguinte figura 1.12, tanto Monstro como SA receberam os pacotes enviados por Bela.

Isto comprova o funcionamento dos *hubs*, em que qualquer pacote recebido numa porta é distribuído por todas as portas, o que neste caso, resultou na distribuição dos pacotes enviados por Bela pelas portas de Monstro e SA.

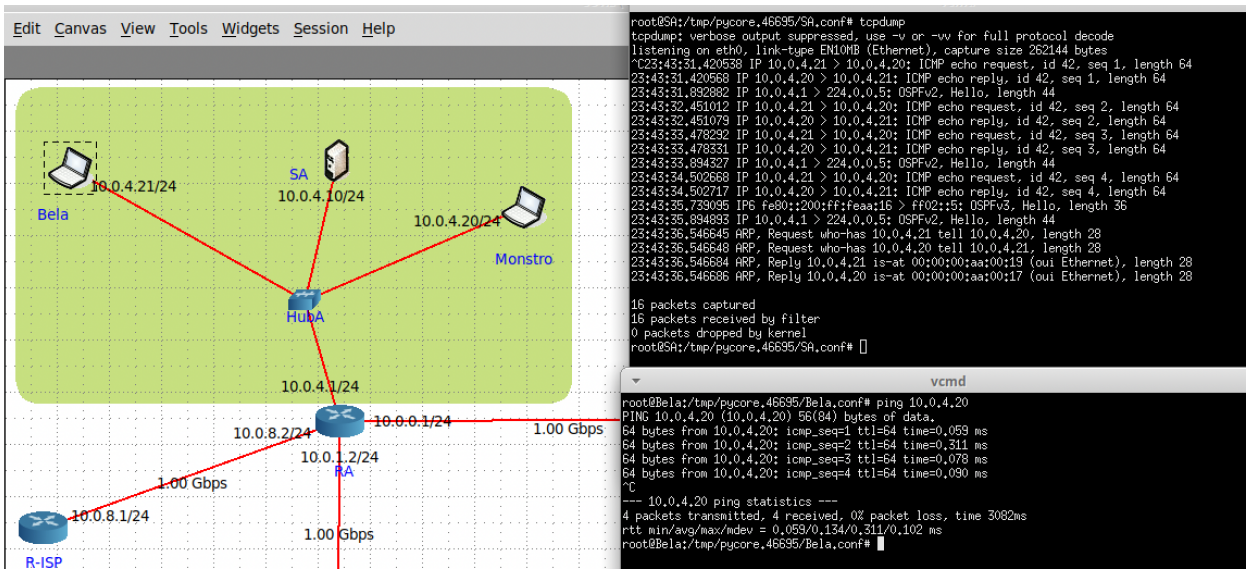

Figura 1.12: *Ping* de Bela para o Monstro e *tcpdump* em SA

Já no Departamento B, foi utilizado um ***switch*** (LAN comutada). Executou-se um comando *ping* do computador Jasmine para o computador Alladin enquanto o servidor B estava a correr o comando *tcpdump*. Como se pode observar na figura 1.13, Alladin recebeu os pacotes enviados por Jasmine mas SB não recebeu nenhum pacote.

Isto deve-se ao comportamento de um *switch* que, ao contrário de um *hub*, envia o pacote apenas para o *host* indicado em vez de o distribuir por todos os *hosts* a si ligados. Isto é possível devido ao facto de serem estabelecidos vários canais de comunicação, ao contrário dos *hubs*, onde não existem canais separados. Isto resulta numa redução do número de colisões, sendo assim os *switches* uma melhor opção para reduzir colisões.

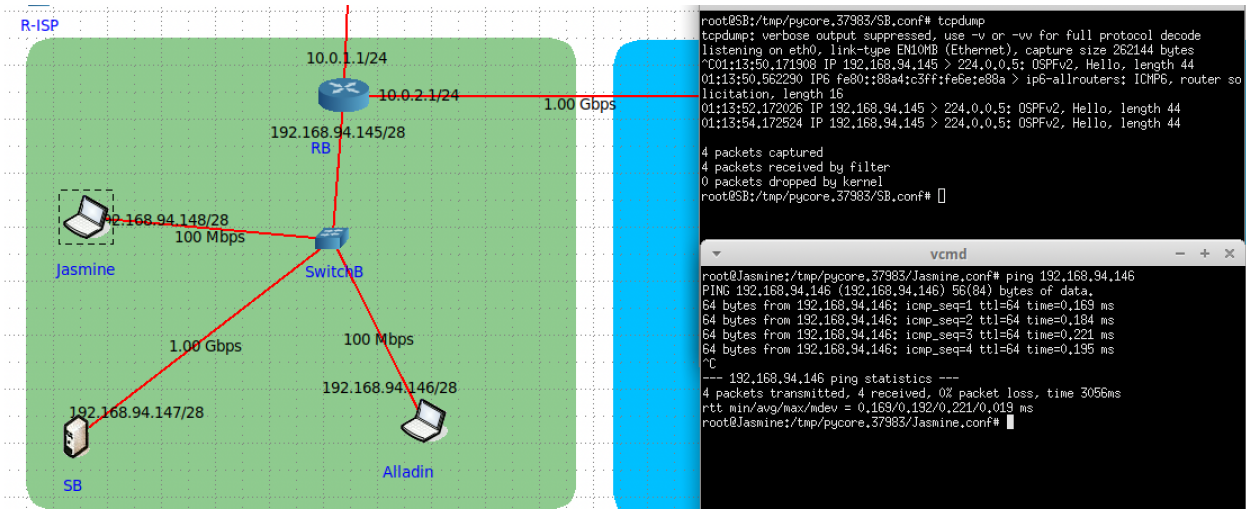

Figura 1.13: *Ping* de Jasmine para o Alladin e *tcpdump* em SB

1.3.2 Exercicio 16

Construa manualmente a tabela de comutação do *switch* do Departamento B, atribuindo números de porta à sua escolha.

A seguir, apresenta-se uma representação da rede do Departamento B:

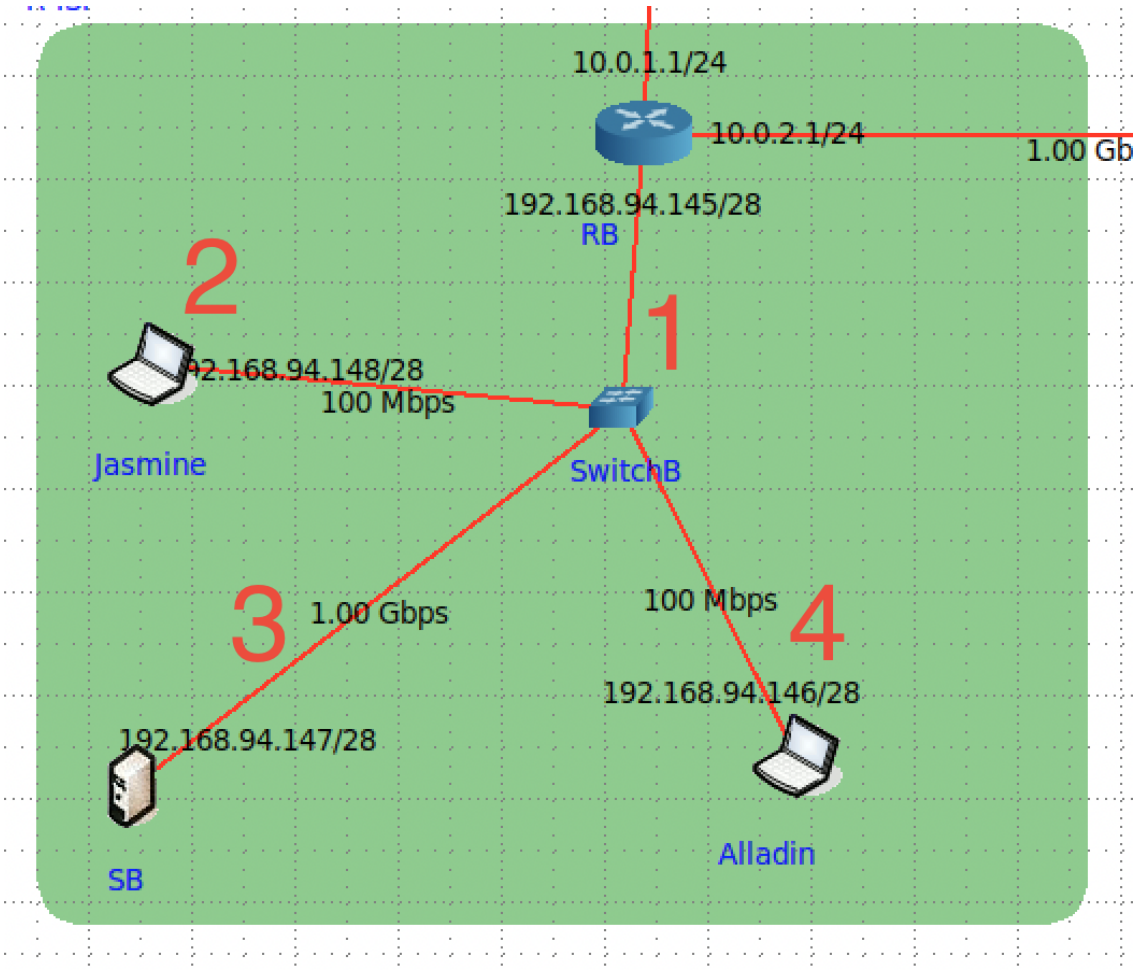

Figura 1.14: Rede do Departamento B com indicação das interfaces

A tabela de comutação do *switch* obtida pelo grupo, tendo em conta a numeração das interfaces na Figura 1.14, foi a seguinte:

Máquina	Endereço MAC	Porta
Router	00:00:00:aa:00:08	1
Jasmine	00:00:00:aa:00:12	2
Servidor B	00:00:00:aa:00:13	3
Alladin	00:00:00:aa:00:14	4

Tabela 1.1: Tabela de comutação do *switch*

Capítulo 2

Conclusão

Este trabalho permitiu a consolidação de conhecimentos sobre a temática de **Ligação Lógica**. Assim, foram estudados conceitos associados a tramas *Ethernet* e ao seu endereçamento através de endereços MAC e o protocolo de endereçamento ARP.

De modo a atingir os objetivos propostos, utilizamos o *Wireshark*, que permitiu capturar pacotes e para inspecionar o conteúdo relevante ao protocolo Ethernet, e a ferramenta de emulação CORE, que permitiu emular as topologias propostas, de modo a estudar, a diferença entre *switches* e *hubs Ethernet*.

Em suma, pensamos ter cumprido os objetivos a que nos propusemos, aprofundando o conhecimento nas componentes exploradas durante o trabalho prático.