



Universidade do Minho
Escola de Engenharia

16 de Maio de 2025

Universidade do Minho - Escola de Engenharia

Redes de Computadores - TP3 e TP4

Afonso Martins
a106931

Luis Felício
a106913

Gonçalo Castro
a107337

Índice

1 Parte 1	4
1.1 Exercício 1 - Captura e análise de Tramas Ethernet	4
1.1.1 alínea 1	4
1.1.2 alínea 2	4
1.1.3 alínea 3	5
1.1.4 alínea 4	5
1.1.5 alínea 5	6
1.2 Exercício 2 - Protocolo ARP e Domínios de Colisão	6
1.2.1 alínea 1	6
1.2.2 alínea 2	7
1.2.2.1 alínea a)	7
1.2.2.2 alínea b)	7
1.2.2.3 alínea c)	7
1.2.3 alínea 3	8
1.2.3.1 alínea a)	8
1.2.3.2 alínea b)	8
1.2.3.3 alínea c)	9
1.2.4 alínea d)	9
1.2.5 alínea 4	10
1.2.6 alínea 5	10
1.2.7 alínea 6	12
1.2.8 alínea 7	12
1.2.9 alínea 8	14
1.3 Exercício 3 - Serviço de NAT/PAT	15
1.3.1 alínea 1	15
2 Parte 2	16
2.1 Exercício 1 - Acesso Rádio	16
2.1.1 alínea 1	16
2.1.2 alínea 2	16
2.1.3 alínea 3	16
2.2 Exercício 2 - Scanning Passivo e Scanning Ativo	17
2.2.1 alínea 4	17
2.2.2 alínea 5	18
2.2.3 alínea 6	19
2.2.4 alínea 7	19
2.2.5 alínea 8	20
2.2.6 alínea 9	21
2.2.7 alínea 10	22
2.2.8 alínea 11	22
2.3 Exercício 3 - Processo de Associação	22
2.3.1 alínea 12	22
2.3.2 alínea 13	23
2.4 Exercício 4 - Transferência de Dados	24
2.4.1 alínea 14	24
2.4.2 alínea 15	24
2.4.3 alínea 16	25

Lista de Figuras

Figura 1	Captura do WireShark Jasmin	4
Figura 2	Execução do comando do ssh core@10.0.2.90 e Endereços MAC	4
Figura 3	Wireshark SSH Protocol	5
Figura 4	SSH Protocol	5
Figura 5	Identificação da Fonte de acordo com execução do comando do ssh core@10.0.2.90	5
Figura 6	Execução do comando ifconfig em R1	5
Figura 7	Execução do comando ifconfig em Jasmin	6
Figura 8	arp -a do Aladdin	6
Figura 9	Trama Ethernet que contém o ARP Request	7
Figura 10	Trama Ethernet que contém o ARP Reply	8
Figura 11	ifconfig aladdin MAC de origem	9
Figura 12	ifconfig R1 MAC de destino	9
Figura 13	Captura do WireShark do host Jasmin	10
Figura 14	Captura do WireShark do host Beauty	10
Figura 15	Tabelas ARP dos hosts Aladdin e Beast	12
Figura 16	Conexão ARP entre Aladdin e R1	13
Figura 17	Conexão entre Aladdin e o DServer: conexão ARP entre R1 e DServer	13
Figura 18	Envio de uma Trama IP do Aladdin para o DServer	14
Figura 19	Resposta do DServer para o Aladdin	14
Figura 20	Definição de portas no SW1	14
Figura 21	Cabeçalho da trama nº90	16
Figura 22	Cabeçalho da trama nº90	16
Figura 23	Cabeçalho da trama nº90	17
Figura 24	Cabeçalho da trama nº190	17
Figura 25	Frame control field format in S1G PPDUs	18
Figura 26	Cabeçalho da trama nº190	18
Figura 27	Ativar opção 'Validate the FCS checksum if possible'	18
Figura 28	Cabeçalho da trama nº190	19
Figura 29	Cabeçalho da trama nº190	20
Figura 30	Resposta ao filtro por tramas beacon	20
Figura 31	Cabeçalho da trama nº190	21
Figura 32	Exemplo de um trama com o rádio signal -25dBm	22
Figura 33	Exemplo de um trama com o rádio signal -25dBm	23
Figura 34	Diagramas das sequências de todas as tramas	24
Figura 35	Trama de dados em estudo	24
Figura 36	Cabeçalho IEEE 802.11 em estudo	25
Figura 37	Exemplo de uma transferência em que não é usado RTS/CTS	25
Figura 38	Exemplo de uma transferência de dados RTC/CTS	25

Resumo

Este relatório documenta a resolução do Trabalho Prático 3 (TP3) e Trabalho Prático 4 (TP4) da Unidade Curricular de Redes de Computadores. A execução das atividades propostas ocorreu integralmente durante o período das aulas, sem enfrentarmos quaisquer dificuldades ou obstáculos significativos. A nossa máquina nativa principal tem o sistema operativo *Windows*.

1 Parte 1

1.1 Exercício 1 - Captura e análise de Tramas Ethernet

Enunciado: A topologia de rede representada na figura abaixo é constituída por: (i) uma LAN comutada que interliga os hosts Beauty, Beast e o servidor DServer (Disney Server) através de um switch (SW1) ao router de acesso Rxy; (ii) uma LAN partilhada que interliga os hosts Jasmine, Aladdin através de um hub ao router de acesso (R1); e (iii) uma rede IP ponto-a-ponto que interliga as duas LANs. Construa a topologia indicada e particularize o router Rxy com o seu número de grupo (e.g., R27 para o grupo 7 do turno PL2). De igual forma, o endereço IP do servidor DServer deve ser alterado para incluir o seu número de grupo no identificador da host interface (4º octeto), e.g. 10.0.2.27, bem como o seu endereço MAC, e.g., 00:00:00:AA:BB:27. Ative a topologia de rede e ative o Wireshark na interface de saída do host Jasmine. Antes de ver a sua série favorita, a Jasmine começa por abrir um terminal e estabelecer um acesso seguro ao servidor DServer usando o comando `ssh core@10.0.2.xy`

1.1.1 alínea 1

Enunciado: Anote os endereços MAC de origem e MAC destino da trama capturada. Identifique a que hosts se referem. Justifique.

20	14.893821841	10.0.0.20	10.0.2.90	TCP	66 52/18 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3335233406...
21	14.894550049	10.0.0.20	10.0.2.90	SSHv2	107 Client: Protocol (SSH-2.0-OpenSSH 8.2p1 Ubuntu-4ubuntu0.3)
22	14.894584620	10.0.2.90	10.0.0.20	TCP	66 22 → 52718 [ACK] Seq=1 Ack=42 Win=65152 Len=0 TSval=237661888...
23	14.910745142	10.0.2.90	10.0.0.20	SSHv2	107 Server: Protocol (SSH-2.0-OpenSSH 8.2p1 Ubuntu-4ubuntu0.3)
24	14.910763017	10.0.0.20	10.0.2.90	TCP	66 52718 → 22 [ACK] Seq=42 Ack=42 Win=64256 Len=0 TSval=33352334...
25	14.913166748	10.0.0.20	10.0.2.90	TCP	1514 52718 → 22 [ACK] Seq=42 Ack=42 Win=64256 Len=1448 TSval=33352...
26	14.913167428	10.0.0.20	10.0.2.90	SSHv2	130 Client: Key Exchange Init
27	14.913210437	10.0.2.90	10.0.0.20	TCP	66 22 → 52718 [ACK] Seq=42 Ack=1490 Win=64128 Len=0 TSval=237661...
28	14.913211684	10.0.2.90	10.0.0.20	TCP	66 22 → 52718 [ACK] Seq=42 Ack=1554 Win=64128 Len=0 TSval=237661...
29	14.916482480	10.0.2.90	10.0.0.20	SSHv2	1090 Server: Key Exchange Init
30	14.916495027	10.0.0.20	10.0.2.90	TCP	66 52718 → 22 [ACK] Seq=1554 Ack=1066 Win=64128 Len=0 TSval=3335...
31	14.918420941	10.0.0.20	10.0.2.90	SSHv2	114 Client: Diffie-Hellman Key Exchange Init
32	14.918744999	10.0.2.90	10.0.0.20	TCP	66 22 → 52718 [ACK] Seq=1066 Ack=1602 Win=64128 Len=0 TSval=2376...
33	14.925497532	10.0.2.90	10.0.0.20	SSHv2	1182 Server: Diffie-Hellman Key Exchange Reply, New Keys, Encrypte...
34	14.925506706	10.0.0.20	10.0.2.90	TCP	66 52718 → 22 [ACK] Seq=1602 Ack=3192 Win=64128 Len=0 TSval=2376...

Figura 1: Captura do WireShark Jasmin

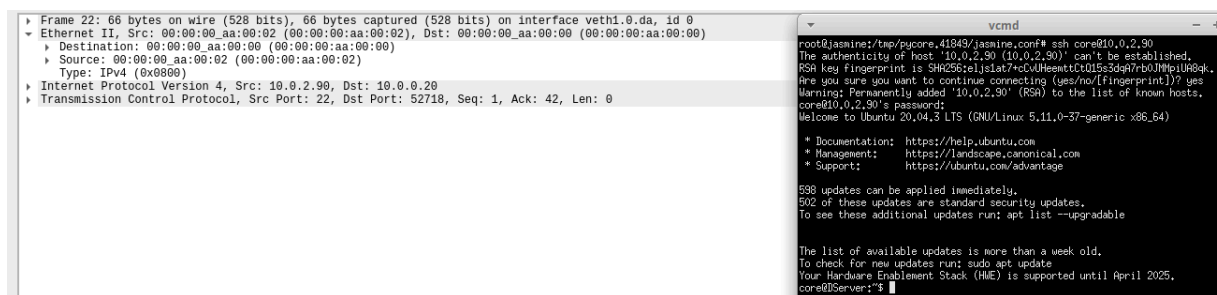


Figura 2: Execução do comando do `ssh core@10.0.2.90` e Endereços MAC

Origem: 00:00:aa:00:02 (corresponde ao host Jasmine)

Destino: 00:00:aa:00:00 (corresponde ao DServer)

1.1.2 alínea 2

Enunciado: Qual o valor hexadecimal do campo Type contido no header da trama Ethernet? O que significa? Qual o campo do header IP que tem semântica idêntica?

Type: 0x0800 , corresponde ao protocolo da camada superior a transportar, neste caso corresponde ao protocolo IPv4 como podemos ver na Figura 2.

O campo do header IP que tem semântica idêntica ao campo 'Type' da trama Ethernet é o campo 'Protocol' (ou 'Protocol Number') do cabeçalho IP. Este campo indica qual protocolo da camada de transporte (por exemplo, TCP, UDP, ICMP) deve ser utilizado para interpretar os dados encapsulados dentro do pacote IP. Por exemplo, um valor de 6 indica TCP, 17 indica UDP e 1 indica ICMP.

1.1.3 alínea 3

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

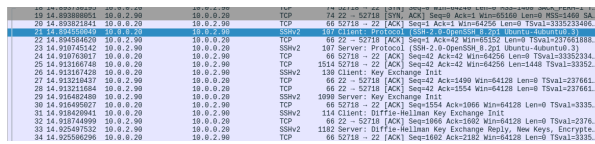


Figura 3: Wireshark SSH Protocol

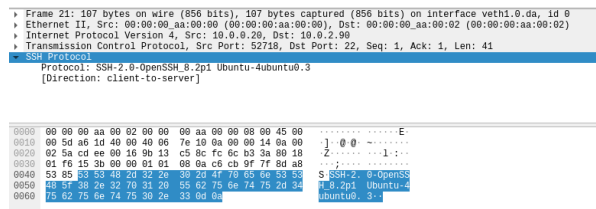


Figura 4: SSH Protocol

Neste pacote, são utilizados 107 bytes no total, dos quais 39 bytes são destinados ao encapsulamento protocolar. Portanto, a sobrecarga introduzida pela pilha protocolar é de 36,4% da memória total do pacote. Esta percentagem indica a quantidade de bytes que são utilizados para fins de controlo e manutenção da comunicação, em contraste com os dados efetivamente transmitidos.

$$\text{Overhead Protocolar} = \frac{39}{107} * 100 = 36,4485981\%$$

1.1.4 alínea 4

Enunciado: Qual é o endereço MAC da fonte? A que host e interface corresponde? Justifique.

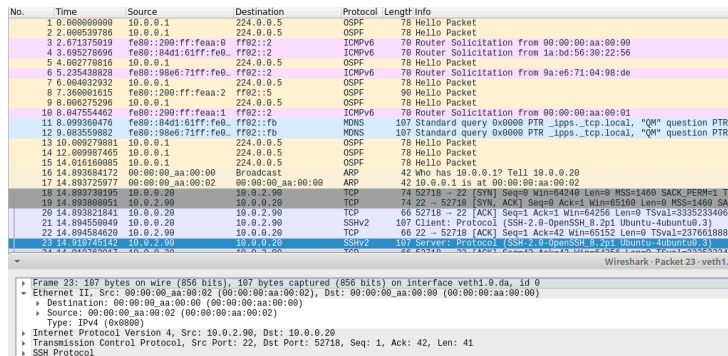


Figura 5: Identificação da Fonte de acordo com execução do comando do ssh core@10.0.2.90

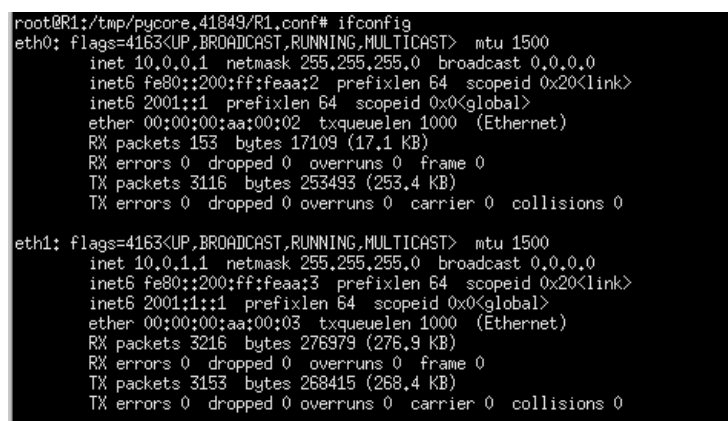


Figura 6: Execução do comando ifconfig em R1

Através da execução do comando `ssh core@10.0.2.90` demonstrado na Figura 5, é possível verificar a existência de uma fonte com o endereço `00:00:00:aa:00:02`. Pela análise da topologia proposta, procede-se à realização do comando `ifconfig` para perceber qual o endereço de cada um dos dispositivos naquele que é o contexto efetivo dentro da rede, conclui-se, como suportado na Figura 6, pela observação da linha

determinada por 'ether', correspondente ao campo *eth0*, a interface pertence ao **host de fonte é o router R1**.

1.1.5 alínea 5

Enunciado: Qual é o endereço MAC do destino? A que host e interface corresponde?

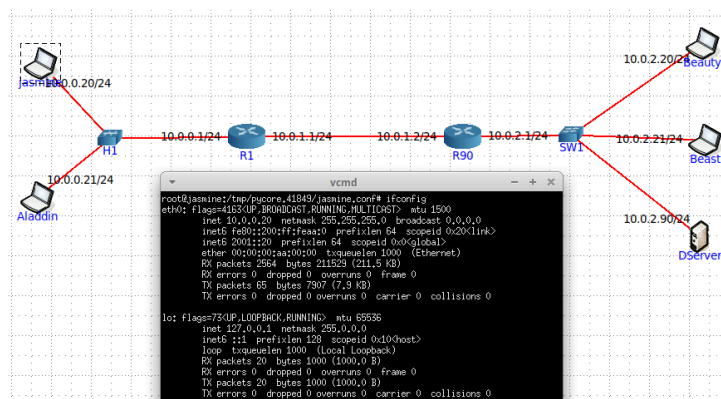


Figura 7: Execução do comando ifconfig em Jasmine

Como visto na Figura 5 é possível verificar a existência de um destino com o endereço **00:00:00:aa:00:00**. Pela análise da topologia proposta, procede-se à realização do comando `ifconfig` para perceber qual o endereço de cada um dos dispositivos naquele que é o contexto efetivo dentro da rede, conclui-se, como suportado na Figura 7, pela observação da linha determinada por 'ether', dentro do campo *eth0*, a interface pertence ao **host de destino é a Jasmine**.

1.2 Exercício 2 - Protocolo ARP e Domínios de Colisão

Enunciado: Deverá ter a cache ARP completamente vazia antes de iniciar esta secção: reinicie a topologia, ou utilize o comando `arp -d`. Comece a capturar tráfego com o Wireshark na interface dos hosts Jasmine, Aladdin, Beauty e Beast. Não sabendo que a Jasmine e a Beauty estavam a capturar tráfego, o Aladdin e o Beast fazem um acesso secreto por ssh para o servidor DServer. Efetue esse acesso e depois pare as várias capturas de tráfego.

1.2.1 alínea 1

Enunciado: Observe o conteúdo da tabela ARP de Aladdin com o comando `arp -a`. Com a ajuda do manual ARP (*man arp*), interprete o significado de cada uma das colunas da tabela.

```
root@Aladdin:/tmp/pycore.38855/Aladdin.conf# arp -a
CondadOnline (10.0.0.1) at 00:00:00:aa:00:02 [ether] on eth0
```

Figura 8: `arp -a` do Aladdin

Ao executar o comando `arp -a` no Linux, é possível obter uma lista de entradas na tabela de cache ARP (Address Resolution Protocol) do sistema. Essa tabela armazena o mapeamento entre endereços IP e endereços MAC (Media Access Control) dos dispositivos na rede local. Neste caso, cada entrada na tabela é constituída por 4 colunas, entre elas:

- Endereço IP: Endereço IP do dispositivo dentro da rede local;
- Endereço MAC: Endereço MAC do dispositivo dentro da rede local;
- Tipo: Tipo de hardware do dispositivo (Ex: Wi-Fi, Ethernet);
- Interface: Interface da rede em que o dispositivo está conectado.

Neste caso, por observação da Figura 8, o *host* Aladdin tem conhecimento que para o próximo salto terá de enviar para o endereço IP 10.0.0.1, endereço MAC 00:00:00:aa:00:02, tipo "ether" e interface *eth0*.

1.2.2 alínea 2

Enunciado: Observe a trama Ethernet que contém a mensagem com o pedido ARP (ARP Request).

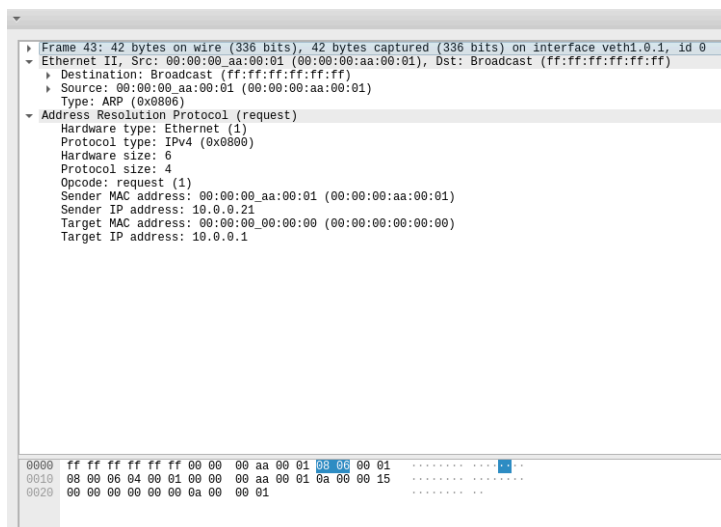


Figura 9: Trama Ethernet que contém o ARP Request

1.2.2.1 alínea a)

Enunciado: Qual é o valor hexadecimal dos endereços MAC origem e destino? Como interpreta e justifica o endereço destino usado?

O endereço da fonte é dado pelo campo **Source: 00:00:00:aa:00:01**, que denota o endereço MAC do Aladdin, que é a origem deste *ARP Request*.

O endereço MAC de destino, por sua vez, é **Destination: ff:ff:ff:ff:ff:ff**, que é um endereço reservado para *broadcast* em redes *Ethernet*. Esse endereço não é atribuído a um dispositivo específico, mas sim a todos os dispositivos na mesma rede local. Ou seja, quando um pacote é enviado para este endereço MAC de destino, todos os dispositivos na rede recebem e processam o pacote. Esse mecanismo de *broadcast* é utilizado quando a origem não sabe o endereço MAC de destino.

No caso do *ARP Request*, o *host* Aladdin está a tentar descobrir o endereço MAC correspondente a um determinado endereço IP (neste caso o do DServer), mas ele não sabe de antemão qual é esse endereço MAC. Portanto, o pacote ARP é enviado com o endereço de *broadcast* **ff:ff:ff:ff:ff:ff**, o que garante que todos os dispositivos na rede local irão ouvir a solicitação. O dispositivo que possui o endereço IP solicitado irá, então, responder com um *ARP Reply*, indicando o seu endereço MAC (situação que iremos explorar mais à frente).

1.2.2.2 alínea b)

Enunciado: Qual o valor hexadecimal do campo Type da trama Ethernet? O que indica?

O campo *Type* da trama em questão contém o valor **ARP (0x0806)**, que identifica o protocolo encapsulado na carga útil da trama. Este campo é fundamental para que a camada *link* (neste caso, *Ethernet*) consiga indicar à camada de rede superior qual protocolo deve processar os dados recebidos.

O valor 0x0806 corresponde ao protocolo ARP (Address Resolution Protocol). Isso significa que a informação transportada na carga útil desta trama *Ethernet* é uma mensagem ARP, usada para mapear endereços IP (de nível de rede) para endereços MAC (de *link layer*).

1.2.2.3 alínea c)

Enunciado: Observando a mensagem ARP, como pode saber que se trata efetivamente de um pedido ARP? Refira duas formas distintas de obter essa informação.

Para determinar que se trata de um pedido ARP (*ARP Request*), podemos recorrer a duas formas distintas de análise da mensagem ARP:

Verificação do campo Opcode: O campo *Opcode* (ou *Operation*) da mensagem ARP indica o tipo de operação que está a ser realizada - um valor de 1 neste campo identifica a mensagem como um *ARP Request* (pedido ARP). Assim, ao observar que o Opcode é 1, conclui-se que se trata de um pedido ARP.

Análise dos campos “Target MAC Address” e “Target IP Address”: Numa mensagem de ARP Request, o campo *Target MAC Address* (endereço MAC de destino) encontra-se tipicamente com o valor 00:00:00:00:00:00, o que indica que o emissor ainda não conhece o endereço MAC associado ao IP de destino. Por sua vez, o campo *Target IP Address* está preenchido com o endereço IP do próximo salto para o router R1. Esta combinação – MAC de destino nulo e IP de destino válido – é característica de um pedido ARP.

Desta forma, tanto pelo valor do *Opcode* como pela análise dos campos de destino, é possível identificar inequivocamente que se trata de um *ARP Request*.

1.2.3 alínea 3

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

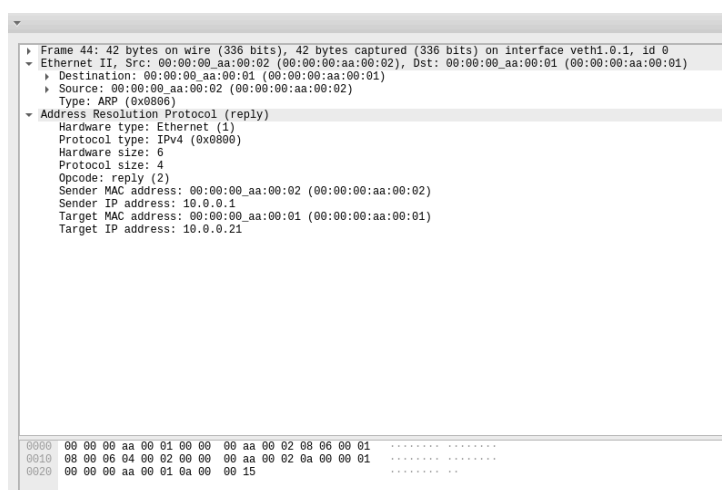


Figura 10: Trama Ethernet que contém o ARP Reply

1.2.3.1 alínea a)

Enuciado: Qual o valor do campo ARP opcode? O que especifica?

O valor do campo *Opcode* é **reply (2)**, o que indica que esta mensagem ARP corresponde a uma resposta ARP (*ARP Reply*).

Este campo faz parte do cabeçalho do protocolo ARP e é utilizado para distinguir o tipo de operação realizada:

- Um valor de 1 indica um *ARP Request*, ou seja, um pedido para obter o endereço MAC correspondente a um determinado endereço IP (como vimos nas alíneas anteriores).
- Um valor de 2, como neste caso, indica um *ARP Reply*, ou seja, uma resposta a um pedido anterior.

Neste tipo de mensagem, o dispositivo que recebeu o pedido ARP está a responder com o seu endereço MAC associado ao endereço IP solicitado, completando assim o processo de resolução de endereços. Esta resposta é enviada diretamente ao dispositivo que efetuou o pedido, permitindo-lhe atualizar a sua cache ARP com a correspondência correta entre o IP e o MAC requisitado em *broadcast* anteriormente.

1.2.3.2 alínea b)

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

A resposta ao pedido ARP efetuado encontra-se nos campos *Sender MAC Address* e *Sender IP Address* da mensagem ARP de resposta (*ARP Reply*). O campo *Sender IP Address* indica o endereço IP que foi alvo do pedido inicial - **10.0.0.1** -, enquanto o campo *Sender MAC Address* fornece o endereço MAC correspondente a esse IP - **00:00:00:aa:00:02**. Estes campos permitem ao dispositivo que originou o pedido ARP conhecer o endereço físico (MAC) associado ao endereço IP pretendido, completando assim o processo de resolução de endereços.

1.2.3.3 alínea c)

Enuciado: Identifique a que sistemas correspondem os endereços MAC de origem e de destino da trama em causa, recorrendo aos comandos *ifconfig*, *netstat -rn* e *arp* executados no host selecionado (*Aladdin*).

```
root@Aladdin:/tmp/pycore.43069/Aladdin.conf# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.21 netmask 255.255.255.0 broadcast 0.0.0.0
    inet6 fe80::200:ff:feaa:1 prefixlen 64 scopeid 0x20<link>
    inet6 2001::21 prefixlen 64 scopeid 0x0<global>
    ether 00:00:00:aa:00:01 txqueuelen 1000 (Ethernet)
    RX packets 415 bytes 35473 (35,4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 14 bytes 1172 (1,1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0,0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0,0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figura 11: ifconfig aladdin MAC de origem

```
root@R1:/tmp/pycore.43069/R1.conf# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.1 netmask 255.255.255.0 broadcast 0.0.0.0
    inet6 fe80::200:ff:feaa:2 prefixlen 64 scopeid 0x20<link>
    inet6 2001::1 prefixlen 64 scopeid 0x0<global>
    ether 00:00:00:aa:00:02 txqueuelen 1000 (Ethernet)
    RX packets 66 bytes 7225 (7,2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 87 bytes 7210 (7,2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.1.1 netmask 255.255.255.0 broadcast 0.0.0.0
    inet6 fe80::200:ff:feaa:3 prefixlen 64 scopeid 0x20<link>
    inet6 2001:1::1 prefixlen 64 scopeid 0x0<global>
    ether 00:00:00:aa:00:03 txqueuelen 1000 (Ethernet)
    RX packets 150 bytes 15082 (15,0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 102 bytes 9212 (9,2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figura 12: ifconfig R1 MAC de destino

Através da Figura 11, podemos verificar o endereço MAC de origem, **00:00:00:aa:00:02** (através da interface *eth0*) e pela Figura 12 podemos concluir que esse endereço MAC pertence ao router R1. O endereço MAC de destino **00:00:00:aa:00:01** (pela interface *eth0*) através da Figura 12 podemos deduzir que pertence ao *host* Aladdin.

1.2.4 alínea d)

Enuciado: Discuta, justificando, o modo de comunicação (*unicast* vs. *broadcast*) usado no envio da resposta ARP (*ARP Reply*).

Nesta resposta, *ARP Reply*, quem dá “match” afirmativo à ação de *broadcast* (isto é, o endereço IP dessa máquina é igual ao endereço IP de Destino do pacote) sabe o endereço MAC proveniente desse pedido, consegue enviar a resposta diretamente a quem a pediu, ou seja, a resposta é enviada em *Unicast* apenas para o endereço MAC que fez o pedido inicialmente. Neste caso em específico, o *host* Aladdin recebeu o *broadcast* e verificou que o endereço MAC que procuravam era exatamente o correspondente ao seu

próprio IP. Desta forma ele reconhece isto, e em *unicast* envia a resposta de volta ao *router* R1 que a pediu inicialmente.

1.2.5 alínea 4

Enuciado: *Verifique se a Jasmine teve conhecimento ou não de todo o tráfego gerado pelo acesso secreto do Aladdin? Qual será a razão para tal?*

No.	Time	Source	Destination	Protocol	Length	Info
41	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
42	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
43	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
44	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
45	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
46	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
47	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
48	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
49	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
50	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
51	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
52	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
53	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
54	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
55	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
56	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
57	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
58	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
59	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
60	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
61	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
62	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
63	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
64	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
65	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
66	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
67	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
68	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
69	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
70	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
71	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
72	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
73	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
74	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
75	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
76	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
77	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
78	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
79	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
80	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
81	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
82	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
83	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
84	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
85	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
86	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
87	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
88	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
89	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
90	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
91	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
92	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
93	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
94	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
95	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
96	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
97	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
98	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
99	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
100	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1

Figura 13: Captura do WireShark do host Jasmin

Através da captura feita pelo Wireshark, como explicito na Figura 13, podemos verificar que o *host* Jasmine tem acesso a toda a informação da interação, uma vez que recebe tanto o *ARP Request* (em *broadcast*) como o *ARP Reply*(apenas destinado ao Aladdin). Isto acontece porque a ligação entre o *router* R1, o *host* Jasmine e Aladdin é um **Hub**, que é um dispositivo da camada 1 (transporte - camada física) que simplesmente replica todos os sinais recebidos para todas as portas. Portanto, quando o DServer responde ao Aladdin, os pacotes chegam ao R1, e depois são enviados para todas as portas do *hub* ligado a R1. Isso significa que Jasmine também recebe uma cópia desses pacotes, mesmo que não sejam destinados a ela.

1.2.6 alínea 5

Enuciado: *De igual modo, verifique se a Beauty teve conhecimento ou não de todo o tráfego gerado pelo acesso secreto do Beast? Qual será a razão para tal?*

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
2	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
3	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
4	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
5	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
6	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
7	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
8	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
9	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
10	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
11	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
12	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
13	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
14	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
15	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
16	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
17	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
18	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
19	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
20	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
21	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
22	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
23	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
24	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
25	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
26	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
27	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
28	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
29	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
30	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
31	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
32	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
33	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
34	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
35	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
36	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
37	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
38	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1
39	0.000000	192.168.1.1	192.168.1.1	ARP	60	Request to resolve local address 192.168.1.1
40	0.000000	192.168.1.1	192.168.1.1	ARP	60	Reply to resolve local address 192.168.1.1

Figura 14: Captura do WireShark do host Beauty

De acordo com a imagem o *host* Beauty não recebe todos os pacote desta interação, isto é o *host* Beauty apenas vai receber o pedido de *broadcast* associado ao pedido de procura (feito pelo DServer) do endereço MAC de destino (o Aladdin), mas não a resposta em *unicast* destinada apenas ao Aladdin.

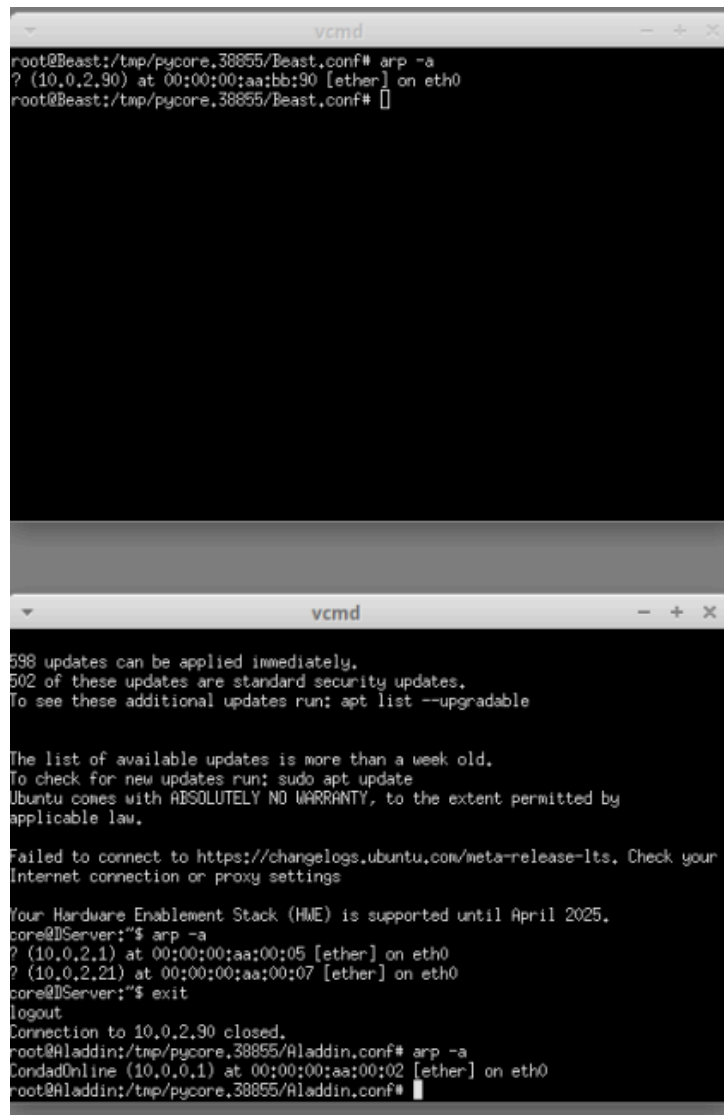
Este fenómeno acontece porque o *router* R1, *host* Beauty e o servidor do DServer estão ligados por um *Switch*, que é um dispositivo de rede que atua na camada *link* (Camada 2) e tem inteligência para aprender os endereços MAC associados a cada porta, ou seja:

Aladdin envia dados para o DServer -> o switch regista o MAC do Aladdin na porta onde ele está ligado -> Encaminha a trama apenas para a porta que liga ao próximo salto (R1)
-> A resposta do DServer regressa para o Aladdin.

Quando o pacote volta pela rota inversa, o *switch* que liga Beauty ao DServer envia o tráfego só para a porta onde está o Aladdin. Beauty não vê esse tráfego, nem de ida nem de volta, porque: o tráfego é *unicast*; o switch sabe onde está cada endereço MAC; só envia a trama para a porta correta (a do Aladdin).

1.2.7 alínea 6

Enunciado: Consulte a tabela ARP do Aladdin e do Beast. Que principal diferença entre as tabelas obtidas e que impacto tem no funcionamento da rede?



```
vcmd
root@Beast:/tmp/pycore.38855/Beast.conf# arp -a
? (10.0.2.90) at 00:00:00:aa:bb:90 [ether] on eth0
root@Beast:/tmp/pycore.38855/Beast.conf#

vcmd
598 updates can be applied immediately.
502 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings

Your Hardware Enablement Stack (HWE) is supported until April 2025.
core@Server:~$ arp -a
? (10.0.2.1) at 00:00:00:aa:00:05 [ether] on eth0
? (10.0.2.21) at 00:00:00:aa:00:07 [ether] on eth0
core@Server:~$ exit
logout
Connection to 10.0.2.90 closed.
root@Aladdin:/tmp/pycore.38855/Aladdin.conf# arp -a
CondadOnline (10.0.0.1) at 00:00:00:aa:00:02 [ether] on eth0
root@Aladdin:/tmp/pycore.38855/Aladdin.conf#
```

Figura 15: Tabelas ARP dos hosts Aladdin e Beast

1.2.8 alínea 7

Enunciado: Esboce um diagrama em que ilustre claramente, e de forma cronológica, todo o tráfego layer 2 (tramas) entre o Aladdin e os hosts com os quais comunica, até à receção do primeiro pacote que contém dados do acesso remoto.

1. ARP Request (Aladdin → Broadcast): Aladdin envia um pedido ARP em *broadcast* para descobrir o endereço MAC do gateway R1. Todos os dispositivos ligados ao *hub* (incluindo Jasmine) recebem esta trama.

2. ARP Reply (R1 → Aladdin): R1 responde diretamente a Aladdin com o seu endereço MAC numa trama *unicast*. Como estão ligados a um *hub*, Jasmine também observa esta resposta.

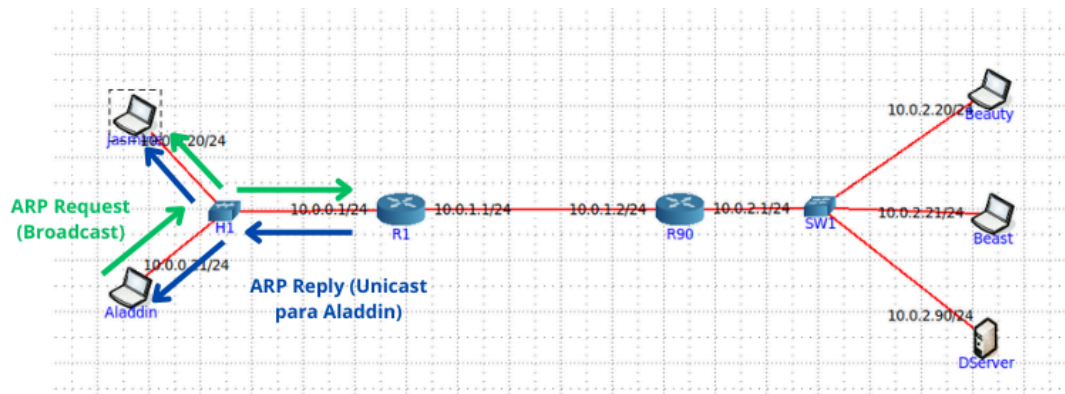


Figura 16: Conexão ARP entre Aladdin e R1

3. Envio do primeiro pacote (Aladdin → R1): Aladdin envia o primeiro pacote IP (ex. TCP SYN para iniciar a sessão SSH), encapsulado numa trama *Ethernet* com destino *unicast* ao MAC de R1. Jasmine, partilhando o *hub*, também a recebe.

4. ARP Request (R1 → Broadcast na LAN 10.0.2.0/24): R1 envia um *ARP Request* para saber o MAC do DServer. Como é um *broadcast*, todos os dispositivos ligados ao switch (Beauty, Beast e DServer) o recebem.

5. ARP Reply (DServer → R1): DServer responde a R1 com uma trama *unicast* contendo o seu MAC. O switch envia esta resposta apenas para a porta onde está ligado R1.

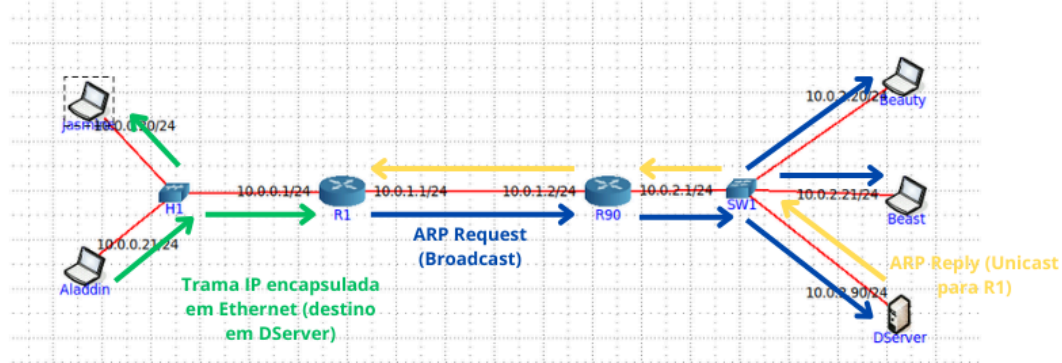


Figura 17: Conexão entre Aladdin e o DServer: conexão ARP entre R1 e DServer

6. Encaminhamento do pacote para o DServer: R1 reencaminha o pacote de dados do Aladdin para o DServer, encapsulando-o numa trama *unicast* com o MAC de destino do DServer. O switch envia a trama apenas ao DServer.

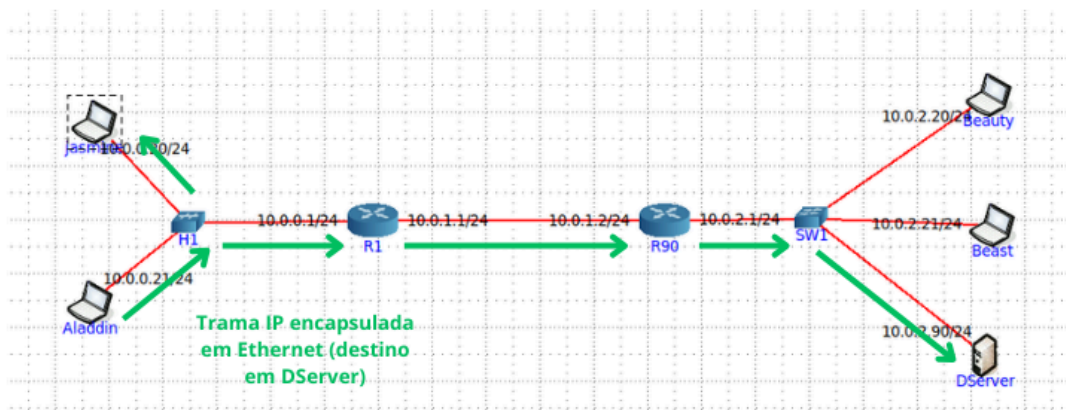


Figura 18: Envio de uma Trama IP do Aladdin para o DServer

7. Resposta do DServer (para Aladdin): DServer responde ao pedido (ex. com um TCP SYN-ACK), enviando a trama ao R1. Este reencaminha para Aladdin via o *hub*, permitindo que Jasmine intercepte a resposta.

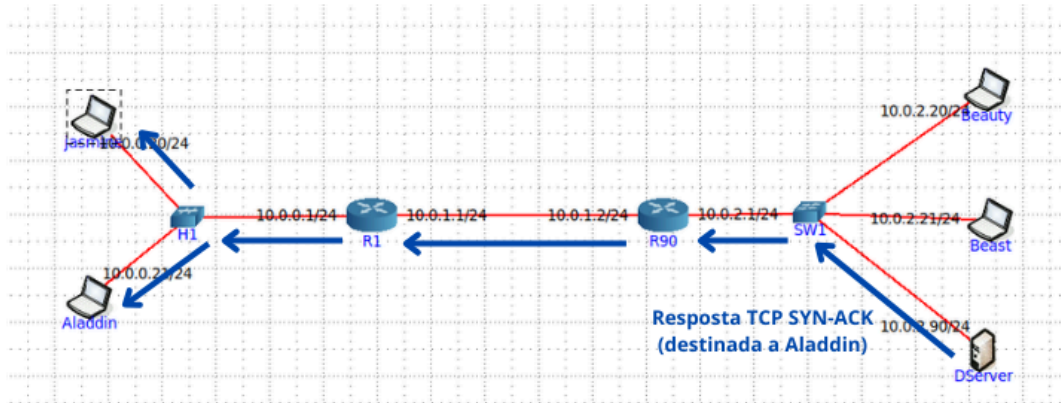


Figura 19: Resposta do DServer para o Aladdin

Neste momento a ligação está pronta a ser estabelecida e pronta a enviar os primeiros *requests* HTTP associados a esta comunicação.

1.2.9 alínea 8

Enuciado: Construa manualmente a tabela de comutação completa do switch da casa da Beauty e do Beast, (SW1) atribuindo números de porta à sua escolha.

Endereço MAC	Porta	Dispositivo	TTL (s)
00:00:00:aa:00:05	1	R90	300
00:00:00:aa:00:06	2	Beauty	300
00:00:00:aa:00:07	3	Beast	300
00:00:00:aa:00:08	4	DServer	300

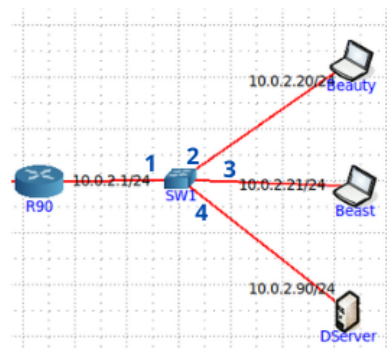


Figura 20: Definição de portas no SW1

Nota: O *TTL (Time to Live)* é tipicamente 300 segundos por defeito em *switches Ethernet*, mas pode variar consoante o equipamento.

1.3 Exercício 3 - Serviço de NAT/PAT

1.3.1 alínea 1

Enunciado: *Como proteção, a Jasmine e o Aladdin, juntamente com a Beauty e o Beast, decidiram conectar R1 e Rxy a uma rede de um ISP com endereços IP públicos, mantendo todo o endereçamento privado das suas LANs. Sabe-se que o ISP não encaminha tráfego para redes privadas, portanto, R1 e Rxy não conseguem encaminhar tráfego para endereços privados remotos, i.e., não fisicamente adjacentes. Discuta que solução implementaria em R1 e em Rxy de modo a manter todas as funcionalidades anteriormente existentes (conectividade IP, acesso ssh ao servidor, etc.).*

Esta topologia apresenta duas redes privadas com endereços IP na gama **10.0.x.x**. Pretende-se agora ligar os *routers* R1 e R90 a uma rede pública de um ISP (ou seja, à *Internet*). No entanto, existe uma limitação: por padrão, os ISP descartam qualquer tráfego destinado a endereços privados, como os das redes 10.0.0.0/8, 192.168.0.0/16, entre outros. Assim, se R1 ou R90 tentarem enviar pacotes diretamente com endereços IP privados, o ISP irá rejeitá-los, impossibilitando a comunicação entre as duas LANs.

Para manter a conectividade entre as redes privadas de Jasmine/Aladdin e Beauty/Beast após a ligação dos *routers* R1 e Rxy a uma rede de um ISP com endereços IP públicos, é necessário ultrapassar a limitação imposta pelo ISP, que não encaminha tráfego destinado a endereços privados. Como as LANs internas mantêm endereçamento privado (ex.: 10.0.0.0/24, 10.0.2.0/24), os *routers* deixam de conseguir comunicar diretamente com redes privadas remotas.

Para contornar esta limitação e manter todas as funcionalidades anteriormente existentes, incluindo o acesso SSH ao DServer e a comunicação entre *hosts* remotos em redes privadas, foi adotada uma solução combinada envolvendo **NAT/PAT e túneis GRE**:

- **NAT/PAT nos routers R1 e R90:**

Cada *router* recebe um endereço IP público atribuído pelo ISP. O tráfego gerado nas LANs privadas é traduzido para esse IP público através de NAT (*Network Address Translation*) ou PAT (*Port Address Translation*). Esta tradução permite que os *routers* comuniquem com a rede do ISP de forma válida, mantendo o acesso externo (por exemplo, ao SSH do DServer) através de regras de *port forwarding* que redirecionam tráfego da porta pública para o IP e porta interna correspondente.

- **Túnel GRE entre R1 e R90:**

Para restabelecer a comunicação direta entre as duas redes privadas (por exemplo, entre Aladdin e DServer), é estabelecido um túnel GRE (*Generic Routing Encapsulation*) entre os endereços IP públicos de R1 e R90. O túnel encapsula os pacotes IP privados dentro de pacotes IP públicos, permitindo o transporte transparente do tráfego privado entre os dois domínios. Assim, apesar do ISP não encaminhar tráfego para endereços privados, o túnel GRE permite essa interligação lógica por cima da infraestrutura pública.

Esta abordagem garante:

- A manutenção de endereçamento privado nas redes locais.
- A continuidade dos serviços anteriormente disponíveis (como *pings*, SSH, etc.).
- A transparência do encaminhamento entre as LANs privadas por via do túnel GRE.
- O acesso ao DServer a partir do exterior, via NAT/PAT e redirecionamento de portas.

Em suma, a combinação de NAT/PAT com GRE permite contornar as restrições do ISP, manter a conectividade entre redes privadas e assegurar todos os serviços essenciais ao funcionamento da rede.

2 Parte 2

Enunciado: A Jasmine, como não gosta de ver os cabos da rede Ethernet espalhados pelo palácio, convenceu o Aladdin a substituir a infraestrutura Ethernet por uma rede sem fios. O Aladdin decidiu então comprar equipamento Wi-Fi e fazer uma captura de tráfego para perceber melhor o funcionamento da rede. Descarregue da plataforma de ensino a captura WLAN-traffic-20250407.pcapng.zip e abra o ficheiro .pcapng no Wireshark. Não se esqueça que deve ser incluída evidência prática que sustente a resposta às questões.

2.1 Exercício 1 - Acesso Rádio

Enunciado: Como pode ser observado, a sequência de bytes capturada inclui meta-informação do nível físico (radiotap header, radio information) obtida do firmware da interface Wi-Fi, para além dos bytes correspondentes a tramas 802.11. Selecione a trama de ordem xy correspondente ao seu identificador de grupo (TurnoGrupo, e.g., 27).

2.1.1 alínea 1

Enunciado: Identifique em que frequência do espectro está a operar a rede sem fios, e o canal que corresponde a essa frequência.

Ao analisar o cabeçalho da trama nº90, denotam-se os campos **Channel: 1** e **Frequency: 2412MHz**. Podemos concluir que a rede sem fios opera numa frequência do espectro 2412Mhz e corresponde ao canal 1.

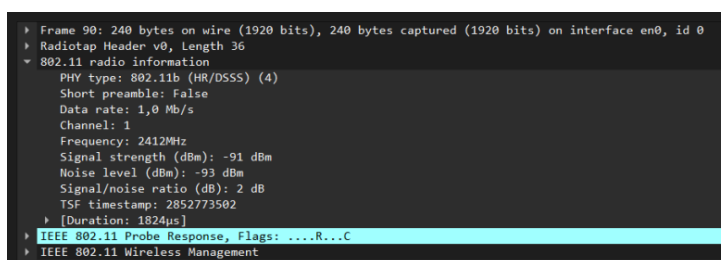


Figura 21: Cabeçalho da trama nº90

2.1.2 alínea 2

Enunciado: Identifique a versão da norma IEEE 802.11 que está a ser usada.

Ao vermos no cabeçalho na secção IEEE 802.11 podemos descobrir a versão utilizada pela norma é a 0 (**Version: 0**).

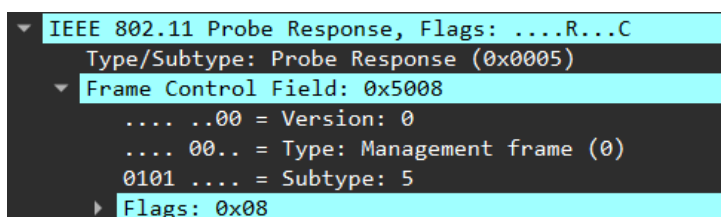


Figura 22: Cabeçalho da trama nº90

2.1.3 alínea 3

Enunciado: Qual a taxa de transmissão a que foi enviada a trama escolhida? Será que essa taxa de transmissão corresponde à máxima que a interface Wi-Fi pode operar? Justifique.

```

▼ Radiotap Header v0, Length 36
  Header revision: 0
  Header pad: 0
  Header length: 36
  ▶ Present flags
    MAC timestamp: 2852773502
  ▶ Flags: 0x10
    Data Rate: 1,0 Mb/s
    Channel frequency: 2412 [2.4 GHz 1]
  ▶ Channel flags: 0x0480, 2 GHz spectrum, Dynamic CCK-OFDM
    Antenna signal: -91 dBm
    Antenna noise: -93 dBm
    Antenna: 0
  ▶ Vendor namespace: Broadcom-3
▼ 802.11 radio information
  PHY type: 802.11b (HR/DSSS) (4)
  Short preamble: False
  Data rate: 1,0 Mb/s
  Channel: 1
  Frequency: 2412MHz
  Signal strength (dBm): -91 dBm
  Noise level (dBm): -93 dBm
  Signal/noise ratio (dB): 2 dB
  TSF timestamp: 2852773502
  ▶ [Duration: 1824µs]

```

Figura 23: Cabeçalho da trama nº90

De acordo com a Figura 23, a trama nº90 foi transmitida a uma taxa de transmissão de 1,0 Mb/s (**802.11 Radio Information** -> **Data Rate: 1,0 Mb/s**) e a taxa máxima de transmissão que a interface Wi-Fi pode operar é de 1,0 Mb/s (**Radio Tap Header** -> **Data Rate: 1,0 Mb/s**), ou seja a trama foi transmitida à taxa máxima.

2.2 Exercício 2 - Scanning Passivo e Scanning Ativo

Enunciado: Como referido, as tramas beacon permitem efetuar scanning passivo em redes IEEE 802.11 (Wi-Fi). Para a captura de tramas disponibilizada, e considerando xy o seu nº de TurnoGrupo (PLxy), responda às seguintes questões:

2.2.1 alínea 4

Enunciado: Selecione uma trama beacon cuja ordem (ou terminação) corresponda ao seu ID de grupo. Esta trama pertence a que tipo de tramas 802.11? Identifique o valor dos identificadores de tipo e de subtipo da trama. Em que parte concreta do cabeçalho da trama estão especificados (ver Anexo I)?

```

▼ IEEE 802.11 Beacon frame, Flags: .....C
  Type/Subtype: Beacon frame (0x0008)
  ▼ Frame Control Field: 0x8000
    .... ..00 = Version: 0
    .... 00.. = Type: Management frame (0)
    1000 .... = Subtype: 8

```

Figura 24: Cabeçalho da trama nº190

A trama selecionada é uma trama *beacon*, utilizada para a divulgação periódica de informação sobre a rede Wi-Fi por parte de um ponto de acesso (AP), sendo fundamental para o processo de *scanning* passivo. Esta trama pertence ao tipo 0, que corresponde às tramas de controlo de gestão (*Management frames*) no padrão *IEEE 802.11*. O subtipo da trama é 8, que identifica especificamente uma **Beacon frame** dentro das tramas de gestão.

Type value b3 b2	Type description	Subtype value b7 b6 b5 b4	Subtype description
00	Management	0000	Association request
00	Management	0001	Association response
00	Management	0010	Reassociation request
00	Management	0011	Reassociation response
00	Management	0100	Probe request
00	Management	0101	Probe response
00	Management	0110-0111	Reserved
00	Management	1000	Beacon
00	Management	1001	Announcement traffic indication message (ATIM)
00	Management	1010	Disassociation
00	Management	1011	Authentication
00	Management	1100	Deauthentication
00	Management	1101-1111	Reserved

Figura 25: Frame control field format in S1G PPDU

Os valores de tipo (0) e subtipo (8) estão localizados no campo 'Frame Control', mais concretamente nos bits correspondentes ao "Type" e "Subtype", conforme apresentado na imagem:

- **Campo "Type":** 00 (binário) → 0 (decimal)
- **Campo "Subtype":** 1000 (binário) → 8 (decimal)

Portanto, a informação relativa ao tipo e subtipo da trama está codificada no 'Frame Control Field' do cabeçalho da trama, cujo valor hexadecimal é 0x8000.

2.2.2 alínea 5

Enuciado: *Verifique se está a ser usado o método de deteção de erros (CRC). Justifique. (Poderá ter de ativar a verificação no Wireshark, em Edit -> Preferences -> Protocols -> IPv4 -> "Validate Checksum if Possible")*

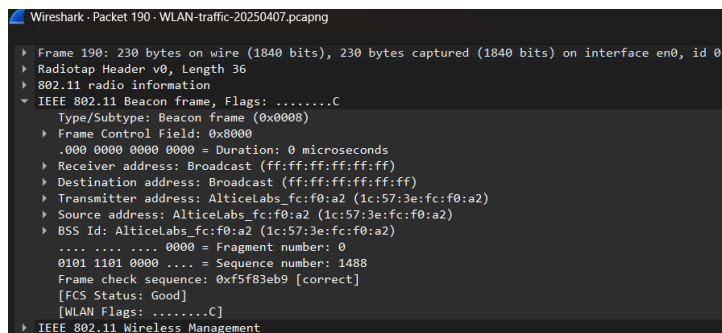


Figura 26: Cabeçalho da trama nº190

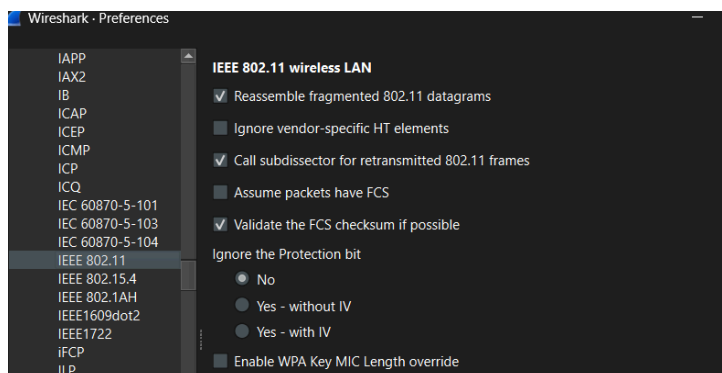


Figura 27: Ativar opção 'Validate the FCS checksum if possible'

```

▶ 802.11 radio information
▼ IEEE 802.11 Beacon frame, Flags: .....C
  Type/Subtype: Beacon frame (0x0008)
  ▶ Frame Control Field: 0x8000
    .000 0000 0000 0000 = Duration: 0 microseconds
  ▶ Receiver address: Broadcast (ff:ff:ff:ff:ff:ff)
  ▶ Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
  ▶ Transmitter address: AlticeLabs_fc:f0:a2 (1c:57:3e:fc:f0:a2)
  ▶ Source address: AlticeLabs_fc:f0:a2 (1c:57:3e:fc:f0:a2)
  ▶ BSS Id: AlticeLabs_fc:f0:a2 (1c:57:3e:fc:f0:a2)
    .... .... 0000 = Fragment number: 0
    0101 1101 0000 .... = Sequence number: 1488
  Frame check sequence: 0xf5f83eb9 [unverified]
  [FCS Status: Unverified]
  [WLAN Flags: .....C]

```

Figura 28: Cabeçalho da trama nº190

O CRC, acrónimo de **Cyclic Redundancy Check**, é um algoritmo matemático que verifica erros nos dados transmitidos. Funciona através da geração de um valor de 32 bits, conhecido como CRC-32, a partir dos dados a serem enviados. Este valor é anexado aos dados como um campo adicional no quadro.

No destino, o mesmo algoritmo CRC-32 é aplicado aos dados recebidos. Se o valor calculado coincidir com o valor anexado, significa que os dados não sofreram alterações durante a transmissão.

Como descrito na Figura 28, o Cyclic Redundancy Check (CRC) não está a ser verificado de acordo com o *Frame Check Sequence* (FCS) que demonstra o status *Unverified*. Deste modo verifica-se que o sistema de deteção de erros CRC não foi efetuado neste pacote.

2.2.3 alínea 6

Enuciado: *Justifique o porquê de ser necessário usar deteção de erros em redes sem fios.*

As redes sem fios operam num ambiente mais suscetível a instabilidades e erros de transmissão do que as redes com fios. Isto deve-se a fatores como interferência de outros dispositivos que usam a mesma frequência, ruído no meio físico ou até obstáculos entre os equipamentos. Por serem redes dinâmicas e muitas vezes associadas à mobilidade, podem ocorrer quebras de sinal e perdas de pacotes. Se esses erros não forem detetados, os pacotes podem chegar corrompidos ao destino, comprometendo a integridade dos dados. Para corrigir esses problemas, torna-se necessário reenviar os pacotes perdidos, o que consome largura de banda adicional e reduz o desempenho da rede, tornando-a mais lenta e instável. Para mitigar estes efeitos, utilizam-se mecanismos de deteção de erros, por exemplo o que vimos na alínea anterior, como o CRC (Cyclic Redundancy Check), que calcula um valor de verificação (*checksum*) para cada pacote. O recetor volta a calcular esse valor e compara com o recebido; se houver diferença, assume-se que ocorreu um erro. Este tipo de verificação contribui para uma comunicação mais fiável e eficiente, melhorando tanto a velocidade como a disponibilidade da rede.

2.2.4 alínea 7

Enuciado: *Uma trama beacon anuncia o intervalo entre beacons às várias taxas de transmissão (B) que o AP suporta, assim como várias taxas de transmissão adicionais (extended supported rates). Indique qual a periodicidade e as taxas de transmissão suportadas pelo AP da trama beacon selecionada.*

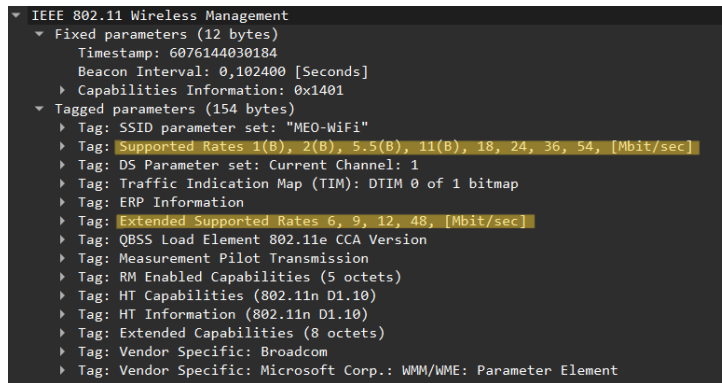


Figura 29: Cabeçalho da trama nº190

De acordo com a Figura 29, a trama *beacon*, com uma periodicidade de 0,102400 segundos (**Beacon Interval: 0,102400 [Seconds]**), é crucial para a sincronização em redes Wi-Fi. As taxas de transmissão básicas suportadas pelo AP são **1, 2, 5.5 e 11 Mbit/sec**, essenciais para a comunicação entre dispositivos clientes e o AP. Adicionalmente, as taxas de transmissão estendidas são **6, 9, 12 e 48 Mbit/sec**, permitindo uma transferência de dados mais rápida sob condições ideais. Essas taxas indicam a velocidade de transmissão que o ponto de acesso (AP) suporta para a comunicação com os dispositivos conectados à rede.

Estas informações da trama *beacon* ajudam os dispositivos clientes a ajustar suas configurações para uma conexão otimizada, garantindo uma comunicação eficiente e estável com o ponto de acesso.

2.2.5 alínea 8

Enunciado: Identifique e liste os SSIDs dos APs que estão a operar na vizinhança da STA de captura. Explícite o modo como obteve essa informação (por exemplo, se usou algum filtro para o efeito).

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1429, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
2	0.002792	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1430, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
3	0.004732	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2528, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
4	0.032178	PTInovacao, 9b:f2:a0	Broadcast	802.11	337	Beacon Frame, Sh=1319, Fw=0, Flags=.....C, BI=100, SSID="MEO-9B2A0"
5	0.032182	PTInovacao, 9b:f2:a2	Broadcast	802.11	230	Beacon Frame, Sh=1320, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
6	0.100364	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1431, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
8	0.105100	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1432, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
11	0.111769	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2529, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
12	0.132458	PTInovacao, 9b:f2:a0	Broadcast	802.11	337	Beacon Frame, Sh=1321, Fw=0, Flags=.....C, BI=100, SSID="MEO-9B2A0"
13	0.207046	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1433, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
14	0.222308	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2530, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
15	0.309521	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1435, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
16	0.309533	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1436, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
17	0.336744	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2531, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
18	0.411983	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1437, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
19	0.411926	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1438, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
20	0.432353	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2532, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
21	0.514371	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1439, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
22	0.514404	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1440, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
23	0.535943	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2533, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
24	0.545715	PTInovacao, 9b:f2:a0	Broadcast	802.11	337	Beacon Frame, Sh=1330, Fw=0, Flags=.....C, BI=100, SSID="MEO-9B2A0"
25	0.545723	PTInovacao, 9b:f2:a2	Broadcast	802.11	230	Beacon Frame, Sh=1331, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
28	0.616817	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1441, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
29	0.616903	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1442, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
32	0.641922	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2534, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
33	0.648860	PTInovacao, 9b:f2:a0	Broadcast	802.11	337	Beacon Frame, Sh=1333, Fw=0, Flags=.....C, BI=100, SSID="MEO-9B2A0"
34	0.728126	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1443, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
35	0.728330	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1444, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
36	0.744311	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2535, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
37	0.750436	PTInovacao, 9b:f2:a0	Broadcast	802.11	337	Beacon Frame, Sh=1335, Fw=0, Flags=.....C, BI=100, SSID="MEO-9B2A0"
38	0.821493	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1445, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
39	0.821520	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1446, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
40	0.846720	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2536, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
41	0.854009	PTInovacao, 9b:f2:a2	Broadcast	802.11	230	Beacon Frame, Sh=1338, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
45	0.923034	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1448, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
48	0.940617	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2537, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
58	1.051597	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2538, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
63	1.131863	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1453, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
64	1.131866	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1454, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
65	1.153978	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2539, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
66	1.160059	PTInovacao, 9b:f2:a0	Broadcast	802.11	337	Beacon Frame, Sh=1340, Fw=0, Flags=.....C, BI=100, SSID="MEO-9B2A0"
67	1.160063	PTInovacao, 9b:f2:a2	Broadcast	802.11	230	Beacon Frame, Sh=1347, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
68	1.256435	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2540, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
70	1.333929	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1459, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
71	1.358705	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2541, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
72	1.364900	PTInovacao, 9b:f2:a0	Broadcast	802.11	337	Beacon Frame, Sh=1350, Fw=0, Flags=.....C, BI=100, SSID="MEO-9B2A0"
86	1.432720	HitronTechno, f3:9a:9a	Broadcast	802.11	222	Beacon Frame, Sh=2573, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
87	1.433442	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1461, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
88	1.435899	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1462, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"
89	1.461213	HitronTechno, f3:9a:9a	Broadcast	802.11	362	Beacon Frame, Sh=2542, Fw=0, Flags=.....C, BI=100, SSID="FlyingNet"
101	1.530806	AliceLab, fc:f0:a0	Broadcast	802.11	305	Beacon Frame, Sh=1465, Fw=0, Flags=.....C, BI=100, SSID="MEO-FCBA0"
102	1.532200	AliceLab, fc:f0:a2	Broadcast	802.11	230	Beacon Frame, Sh=1466, Fw=0, Flags=.....C, BI=100, SSID="MEO-HIFI1"

Figura 30: Resposta ao filtro por tramas beacon

Primeiramente utilizou-se o filtro `wlan.fc.type_subtype == 0x0008` para encontrar todas as tramas *beacon*. De seguida com o resultado foram se aplicando filtros sucessivos para excluir as tramas referentes a cada SSID encontrado (`!wlan.ssid == <SSID>`). Repetimos este processo até não restarem tramas *beacon* para procurar. Aqui está a listagem de todos os SSID encontrados:

- MEO-WiFi

- MEO-FCF0A0
- FlyingNet
- MEO-828830
- MEO-9BF2A0
- Masmorra do Sexo
- phi_F41927C3C600
- GVBRAGA_quarto
- GVBRAGA_EXT
- NOS-26F6
- NOS-C8B6
- NOS-9946_EXT
- GVBRAGA
- Vodafone-D0ED8A
- MEO-66DB70
- NOS-52C6
- MEO-854C80
- NOS-FD24
- MEO-F17570

E este é o filtro que foi sendo incrementado até obtermos todos os SSID:

```
wlan.fc.type_subtype == 0x08 && !(wlan.ssid == "MEO-WiFi") && !(wlan.ssid == "MEO-FCF0A0") && !(wlan.ssid == "FlyingNet") && !(wlan.ssid == "MEO-828830") && !(wlan.ssid == "MEO-9BF2A0") && !(wlan.ssid == "Masmorra do Sexo") && !(wlan.ssid == "phi_F41927C3C600") && !(wlan.ssid == "GVBRAGA_quarto") && !(wlan.ssid == "GVBRAGA_EXT") && !(wlan.ssid == "NOS-26F6") && !(wlan.ssid == "NOS-C8B6") && !(wlan.ssid == "NOS-9946_EXT") && !(wlan.ssid == "GVBRAGA") && !(wlan.ssid == "Vodafone-D0ED8A") && !(wlan.ssid == "MEO-66DB70") && !(wlan.ssid == "NOS-52C6") && !(wlan.ssid == "MEO-854C80") && !(wlan.ssid == "NOS-FD24") && !(wlan.ssid == "MEO-F17570")
```

2.2.6 alínea 9

Enuciado: Estabeleça um filtro Wireshark apropriado que lhe permita visualizar todas as tramas probing request e probing response, simultaneamente.

Realizamos um filtro pelo subtipo das tramas que sejam igual ao valor 4 ou 5, restringindo a captura a todos. *Probing request e responses* presentes:

```
wlan.fc.type_subtype == 0x04 || wlan.fc.type_subtype == 0x05
```

No.	Time	Source	Destination	Protocol	Length	Info
461	6.150862	PTInovacao_29:a9:c2	XiaomiMobile_0a:80:11	802.11	434	Probe Response, SN=3977, FN=0, Flags=.....C, BI=100, SSID="Masmorra do Sexo"
462	6.151368	PTInovacao_29:a9:c2	XiaomiMobile_0a:80:11	802.11	240	Probe Response, SN=3980, FN=0, Flags=.....C, BI=100, SSID="MEO-WiFi"
463	6.156382	PTInovacao_29:a9:c2	XiaomiMobile_0a:80:11	802.11	240	Probe Response, SN=3980, FN=0, Flags=.....C, BI=100, SSID="MEO-WiFi"
464	6.156486	PTInovacao_29:a9:c2	XiaomiMobile_0a:80:11	802.11	240	Probe Response, SN=3980, FN=0, Flags=.....C, BI=100, SSID="MEO-WiFi"
465	6.160268	PTInovacao_29:a9:c2	XiaomiMobile_0a:80:11	802.11	240	Probe Response, SN=3980, FN=0, Flags=.....C, BI=100, SSID="MEO-WiFi"
489	6.550237	HitronTechno_f3:9a:11	Commscope_aa:9c:66	802.11	486	Probe Response, SN=3839, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
490	6.550983	HitronTechno_f3:9a:11	Commscope_aa:9c:66	802.11	486	Probe Response, SN=3839, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
493	6.550719	HitronTechno_f3:9a:11	Commscope_aa:9c:66	802.11	486	Probe Response, SN=3839, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
495	6.592221	HitronTechno_f3:9a:11	Commscope_aa:9c:66	802.11	486	Probe Response, SN=3840, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
496	6.592979	HitronTechno_f3:9a:11	Commscope_aa:9c:66	802.11	486	Probe Response, SN=3840, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
497	6.601673	HitronTechno_f3:9a:11	Commscope_aa:9c:66	802.11	486	Probe Response, SN=3840, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
500	6.604755	PTInovacao_29:a9:c2	Commscope_aa:9c:66	802.11	240	Probe Response, SN=3990, FN=0, Flags=.....C, BI=100, SSID="MEO-WiFi"
501	6.605511	PTInovacao_29:a9:c2	Commscope_aa:9c:66	802.11	240	Probe Response, SN=3990, FN=0, Flags=.....C, BI=100, SSID="MEO-WiFi"
503	6.615776	PTInovacao_29:a9:c2	Commscope_aa:9c:66	802.11	240	Probe Response, SN=3990, FN=0, Flags=.....C, BI=100, SSID="MEO-WiFi"
510	6.669892	Commscope_aa:9c:66	Broadcast	802.11	260	Probe Request, SN=1128, FN=0, Flags=.....C, SSID=Wildcard (Broadcast)
592	7.707615	a6:e4:60:55:5b:bf	Broadcast	802.11	125	Probe Request, SN=2130, FN=0, Flags=.....C, SSID=Wildcard (Broadcast)
593	7.713951	HitronTechno_f3:9a:11	a6:e4:60:55:5b:bf	802.11	486	Probe Response, SN=3841, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
598	7.749443	82:fc:70:79:b7:c3	Broadcast	802.11	125	Probe Request, SN=473, FN=0, Flags=.....C, SSID=Wildcard (Broadcast)
599	7.755530	HitronTechno_f3:9a:11	82:fc:70:79:b7:c3	802.11	486	Probe Response, SN=3842, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
600	7.755557	HitronTechno_f3:9a:11	82:fc:70:79:b7:c3	802.11	486	Probe Response, SN=3842, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
601	7.758628	HitronTechno_f3:9a:11	82:fc:70:79:b7:c3	802.11	486	Probe Response, SN=3842, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
613	7.986085	a6:e4:60:55:5b:bf	Broadcast	802.11	125	Probe Request, SN=1802, FN=0, Flags=.....C, SSID=Wildcard (Broadcast)
649	8.578459	HitronTechno_e5:26:11	Commscope_93:9d:b1	802.11	453	Probe Response, SN=3084, FN=0, Flags=.....C, BI=100, SSID="NOS-26F6"
655	8.619230	HitronTechno_e5:26:11	Commscope_93:9d:b1	802.11	453	Probe Response, SN=3085, FN=0, Flags=.....C, BI=100, SSID="NOS-26F6"
663	8.715846	AltoBeam_08:32:99	Broadcast	802.11	110	Probe Request, SN=3487, FN=0, Flags=.....C, SSID=Wildcard (Broadcast)
670	8.768157	PTInovacao_e5:26:11	AltoBeam_08:32:99	802.11	224	Probe Response, SN=1802, FN=0, Flags=.....C, BI=100, SSID="MEO-WiFi"
721	9.237493	a6:ef:15:08:32:99	7a:01:a1:20:d9:b2	802.11	216	Probe Response, SN=2455, FN=0, Flags=.....C, BI=100, SSID="phi_F41927C3C600"

Figura 31: Cabeçalho da trama nº190

- wlan.fc.type_subtype == 0x04 — filtra as **Probe Request**;
- wlan.fc.type_subtype == 0x05 — filtra as **Probe Response**;

2.2.7 alínea 10

Enuciado: Assuma que a STA de captura consegue-se associar a qualquer AP na vizinhança. Dadas as tramas recebidas através do scanning ativo e passivo, observe os valores da força do sinal (Signal Strength) nas meta-informações de nível físico e indique a qual AP a STA de captura se deve associar para obter a melhor qualidade de ligação possível. Indique como chegou a esta resposta.

O AP ao qual a STA deve se associar para obter a melhor qualidade de ligação possível é o com o endereço MAC **ca:8c:cf:d6:28:fb**, porque possui uma força do sinal de **-25 dBm**. Para obter esse valor, construímos um filtro que compara a força do sinal com um valor negativo inicial (-1) e decrementamos esse valor até chegar a **-25 dBm**. O filtro final e os resultados correspondentes foram os seguintes:

wlan_radio_signal_dbm == -25

No.	Time	Source	Destination	Protocol	Length	Info
1518	18.829692	ca:8c:cf:d6:28:fb	Broadcast	802.11	208	Probe Request, SN=3730, FN=0, Flags=.....C, SSID=Wildcard (Broadcast)
1520	18.834760	HitronTechno_f3:9a:...	Broadcast	802.11	48	Acknowledgement, Flags=.....C
1620	19.853562	MS-NLB-PhysServer-3...	Broadcast	802.11	208	Probe Request, SN=3368, FN=0, Flags=.....C, SSID=Wildcard (Broadcast)
1622	19.859806	HitronTechno_f3:9a:...	Broadcast	802.11	48	Acknowledgement, Flags=.....C
1623	19.859889	MS-NLB-PhysServer-3...	Broadcast	802.11	208	Probe Request, SN=3369, FN=0, Flags=.....C, SSID=Wildcard (Broadcast)
44013	228.547073	fe:bd:a5:05:6c:84	76:9b:e8:f3:9a:43	802.11	140	QoS Data, SN=192, FN=0, Flags=p.P...TC

Figura 32: Exemplo de um trama com o rádio signal -25dBm

2.2.8 alínea 11

Enuciado: Os valores de taxa de transmissão do Wi-Fi estão diretamente associados à qualidade da receção do sinal. Considerando os valores de sensibilidade mínima (Minimum Sensivity) e taxa de transmissão (Data Rate) que constam nas tabelas de referência (ver Anexo II), e a força do sinal recebido nas tramas do AP identificado na resposta anterior, estime o débito que a STA obterá nessa ligação.

O sinal mais forte possui uma força de **-25 dBm**, mas este não se enquadra em nenhum dos intervalos especificados nos Anexos II. No entanto, se os Anexos incluíssem informações sobre esta situação, poderíamos aplicar uma abordagem de interpolação. Isso significaria calcular uma taxa de transmissão estimada multiplicando a fração da taxa pelo valor máximo que a tecnologia permite. Ao realizar esse procedimento, poderíamos obter uma estimativa mais precisa da taxa de transmissão que a estação (STA) poderia alcançar nessa conexão Wi-Fi específica. No entanto, é crucial ter em mente que mesmo essa abordagem não garante uma precisão absoluta, uma vez que a qualidade da receção do sinal é influenciada por uma variedade de fatores além da força do sinal, como interferências, obstáculos físicos e distância do ponto de acesso (AP). Assim, enquanto podemos usar esses dados como uma referência inicial para estimar o débito da conexão, é importante reconhecer que o débito real pode variar em condições reais devido a esses fatores externos. Portanto, é sempre recomendável realizar testes práticos para avaliar o desempenho real da conexão WiFi num ambiente específico.

2.3 Exercício 3 - Processo de Associação

Enuciado: Numa rede Wi-Fi estruturada, um nodo ou STA deve associar-se a um ponto de acesso antes de enviar dados. O processo de associação nas redes IEEE 802.11 é executada enviando a trama association request da STA para o AP e a trama association response enviada pelo AP para a STA, em resposta ao pedido de associação recebido. Este processo é antecedido por uma fase de autenticação. Para a sequência de tramas capturada:

2.3.1 alínea 12

Enuciado: Identifique uma sequência de tramas que corresponda a um processo de associação realizado com sucesso entre a STA e o AP, incluindo a fase de autenticação.

Um processo de associação bem-sucedido entre uma Estação (STA) e um Ponto de Acesso (AP) segue uma sequência de trocas de tramas definidas pelo padrão de comunicação Wi-Fi. Tudo começa quando o AP envia uma trama do tipo *Management* e subtipo *Authentication*, identificada pelo número de ordem 2042.

Essa trama é um convite para que a STA prove sua identidade na rede. Em resposta, a STA envia uma trama de autenticação, confirmando a sua presença e disposição para se associar à rede No 2044. Após essa fase de autenticação, o AP então prossegue a enviar uma trama do subtipo *Association Request*. Esta trama contém informações sobre as capacidades da STA e suas preferências de comunicação. A STA, por sua vez, responde a este convite com uma trama *Association Response*, indicando a sua aceitação.

No.	Time	Source	Destination	Protocol	Length	Info
2042	23.707373	fe:bd:a5:05:6c:84	HitronTechno_f3:9a:...	802.11	106	Authentication, SN=3343, FN=0, Flags=.....C
2044	23.707398	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	70	Authentication, SN=3852, FN=0, Flags=.....C
2046	23.710405	fe:bd:a5:05:6c:84	HitronTechno_f3:9a:...	802.11	202	Association Request, SN=3344, FN=0, Flags=.....C, SSID="FlyingNet"
2048	23.716772	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	210	Association Response, SN=3853, FN=0, Flags=.....C
10356	56.657756	Apple_71:41:a1	HitronTechno_f3:9a:...	802.11	81	Authentication, SN=1387, FN=0, Flags=.....C
10368	56.659702	HitronTechno_f3:9a:...	Apple_71:41:a1	802.11	70	Authentication, SN=3889, FN=0, Flags=.....C
10372	56.661907	Apple_71:41:a1	HitronTechno_f3:9a:...	802.11	205	Association Request, SN=1388, FN=0, Flags=.....C, SSID="FlyingNet"
10376	56.669795	HitronTechno_f3:9a:...	Apple_71:41:a1	802.11	210	Association Response, SN=3890, FN=0, Flags=.....C
10530	57.303645	AzureWaveTec_0f:0e:...	HitronTechno_f3:9a:...	802.11	70	Authentication, SN=257, FN=0, Flags=.....C
10532	57.303655	HitronTechno_f3:9a:...	AzureWaveTec_0f:0e:...	802.11	70	Authentication, SN=3891, FN=0, Flags=.....C
10534	57.304688	HitronTechno_f3:9a:...	AzureWaveTec_0f:0e:...	802.11	70	Authentication, SN=3891, FN=0, Flags=.....C
10536	57.306944	AzureWaveTec_0f:0e:...	HitronTechno_f3:9a:...	802.11	164	Association Request, SN=258, FN=0, Flags=.....C, SSID="FlyingNet"
10538	57.309944	HitronTechno_f3:9a:...	AzureWaveTec_0f:0e:...	802.11	210	Association Response, SN=3892, FN=0, Flags=.....C
31459	144.971369	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	66	Deauthentication, SN=3925, FN=0, Flags=.....C
31460	144.971372	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	66	Deauthentication, SN=3925, FN=0, Flags=.....C
31461	144.971380	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	66	Deauthentication, SN=3925, FN=0, Flags=.....C
31462	144.971383	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	66	Deauthentication, SN=3925, FN=0, Flags=.....C
32460	153.869966	fe:bd:a5:05:6c:84	HitronTechno_f3:9a:...	802.11	106	Authentication, SN=1556, FN=0, Flags=.....C
32462	153.870102	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	70	Authentication, SN=3939, FN=0, Flags=.....C
32464	153.874096	fe:bd:a5:05:6c:84	HitronTechno_f3:9a:...	802.11	202	Association Request, SN=1557, FN=0, Flags=.....C, SSID="FlyingNet"
32466	153.878503	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	210	Association Response, SN=3940, FN=0, Flags=.....C
36369	182.163376	fe:bd:a5:05:6c:84	HitronTechno_f3:9a:...	802.11	106	Authentication, SN=3823, FN=0, Flags=.....C
36371	182.165154	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	70	Authentication, SN=3956, FN=0, Flags=.....C
36373	182.167893	fe:bd:a5:05:6c:84	HitronTechno_f3:9a:...	802.11	202	Association Request, SN=3824, FN=0, Flags=.....C, SSID="FlyingNet"
36375	182.170791	HitronTechno_f3:9a:...	fe:bd:a5:05:6c:84	802.11	210	Association Response, SN=3957, FN=0, Flags=.....C
38955	190.577959	PTInovacao5_66:db:...	92:8d:33:af:7f:13	802.11	81	Authentication, SN=2053, FN=0, Flags=.....C
44080	229.030351	AzureWaveTec_0f:0e:...	HitronTechno_f3:9a:...	802.11	66	Deauthentication, SN=259, FN=0, Flags=.....C
50813	271.988597	PTInovacao_77:f9:60	TuyaSmart_0d:5f:c4	802.11	81	Authentication, SN=860, FN=0, Flags=.....C
52812	279.325067	PTInovacao_77:f9:60	TuyaSmart_0d:5f:c4	802.11	81	Authentication, SN=1020, FN=0, Flags=.....C
52814	279.333938	PTInovacao_77:f9:60	TuyaSmart_0d:5f:c4	802.11	81	Authentication, SN=1021, FN=0, Flags=.....C
57557	284.586003	HitronTechno_e5:26:...	96:c4:1d:35:d2:22	802.11	70	Authentication, SN=3177, FN=0, Flags=.....C
61476	290.322291	PTInovacao_77:f9:60	TuyaSmart_0d:5f:c4	802.11	81	Authentication, SN=1252, FN=0, Flags=.....C
61477	290.324753	PTInovacao_77:f9:60	TuyaSmart_0d:5f:c4	802.11	81	Authentication, SN=1252, FN=0, Flags=.....C
61478	290.324757	PTInovacao_77:f9:60	TuyaSmart_0d:5f:c4	802.11	81	Authentication, SN=1252, FN=0, Flags=.....C
61480	290.334736	PTInovacao_77:f9:60	TuyaSmart_0d:5f:c4	802.11	81	Authentication, SN=1253, FN=0, Flags=.....C

Figura 33: Exemplo de um trama com o rádio signal -25dBm

A captura foi realizada aplicando os seguintes filtros no Wireshark:

wlan.fc.type_subtype == 0x0b || wlan.fc.type_subtype == 0x0c || wlan.fc.type_subtype == 0x00 || wlan.fc.type_subtype == 0x01

Estes filtros foram selecionados para capturar especificamente:

- **0x00: Association Request**
- **0x01: Association Response**
- **0x0b: Authentication frames**
- **0x0c: Deauthentication frames**

Esta combinação de filtros permite visualizar claramente todas as etapas do processo de associação sem o ruído de outros tipos de tramas, capturando tanto o estabelecimento da conexão quanto sua eventual conclusão (de *authentication*).

2.3.2 alínea 13

Enuciado: Efetue um diagrama que ilustre a sequência de todas as tramas trocadas no processo.

São trocadas num total de 4 tramas durante todo o processo de associação entre o STA e o AP:

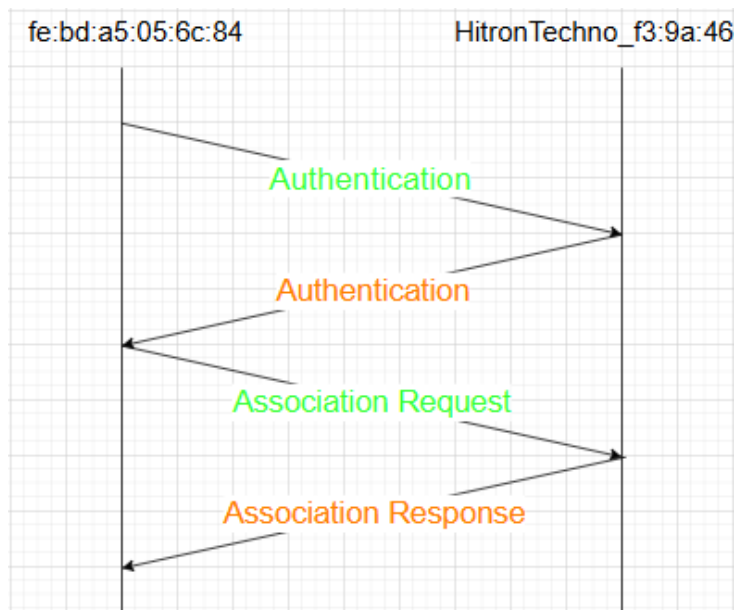


Figura 34: Diagramas das sequências de todas as tramas

2.4 Exercício 4 - Transferência de Dados

Enuciado: O trace disponibilizado, para além de tramas de gestão da ligação de dados, inclui tramas de dados e tramas de controlo da transferência desses mesmos dados.

2.4.1 alínea 14

Enuciado: Estabeleça um filtro apropriado e selecione uma trama de dados (Data ou QoS Data), cujo número de ordem inclua o seu identificador de grupo (terminação xy, ou y caso não exista xy). Sabendo que o campo Frame Control contido no cabeçalho das tramas 802.11 permite especificar a direccionalidade das tramas, o que pode concluir face à direccionalidade dessa trama, será local à WLAN?

Ao definir um filtro que não restringe as tramas com o subtipo 'Data' ou 'QoS Data', e que o número de ordem contém o nosso identificador de grupo, 90, deparamos com a trama de número de ordem 290. Ao analisarmos o cabeçalho da mesma, na secção das *flags*, deparamos com as *flags* "To DS 1" e "From DS 0", o que indica que a trama está a sair da estação (cliente) em direção ao ponto de acesso (AP), ou seja, a trama está a ser transmitida do ambiente sem fio para o sistema com fio.

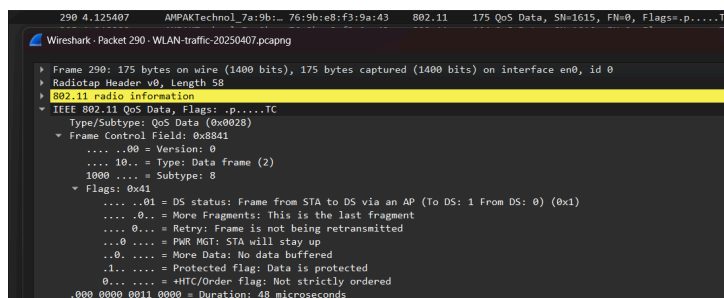


Figura 35: Trama de dados em estudo

2.4.2 alínea 15

Enuciado: Para a trama de dados selecionada, transcreva os endereços MAC em uso, identificando quais os endereços correspondentes à estação sem fios (STA), ao AP e ao router de acesso ao sistema de distribuição (DS)?

Ao analisar o cabeçalho deparamos com 3 endereços MAC:

- **HitronTechno_f3:9a:46** (Endereço MAC: 74:9b:e8:f3:9a:46),
- **AMPAKTechnol_7a:9b:68** (Endereço MAC: b8:2d:28:7a:9b:68),
- **76:9b:e8:f3:9a:43** (Endereço MAC: 76:9b:e8:f3:9a:43)

De acordo com a alínea anterior sabemos que a trama em questão está a entrar no ambiente com fio ou seja o cliente corresponde ao endereço destino, **76:9b:e8:f3:9a:43** (MAC: 76:9b:e8:f3:9a:43), o STA corresponde ao endereço transmissor, **AMPAKTechnol_7a:9b:68** (MAC: b8:2d:28:7a:9b:68), e o AP corresponde ao endereço *source*, **HitronTechno_f3:9a:46** (MAC: 74:9b:e8:f3:9a:46).

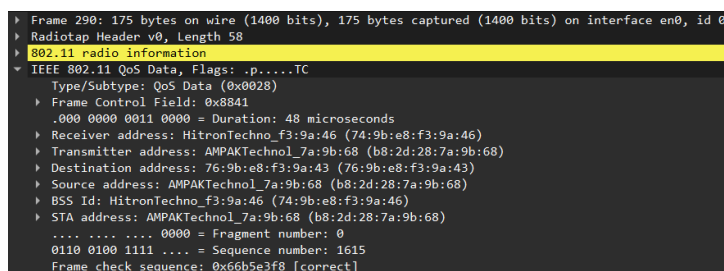


Figura 36: Cabeçalho IEEE 802.11 em estudo

2.4.3 alínea 16

Enunciado: *O uso de tramas Request To Send e Clear To Send, apesar de opcional, é comum para efetuar “pré-reserva” do acesso ao meio quando se pretende enviar tramas de dados, com o intuito de reduzir o número de colisões resultante maioritariamente de STAs escondidas. Para o envio de dados selecionado acima, verifique se está a ser usada a opção RTS/CTS na troca de dados entre a STA e o AP/Router da WLAN, identificando a direccionalidade das tramas e os sistemas envolvidos. Dê um exemplo de uma transferência de dados em que é usada a opção RTC/CTS e um outro em que não é usada.*

760 9.617947	HitronTechno_e5:26:: Broadcast	802.11	329 Beacon frame, SN=3846, FN=0, Flags=.....C, BI=100, SSID="NOS-26F6"
761 9.629835	AlticeLabs_fc:f0:a2 Broadcast	802.11	230 Beacon frame, SN=1630, FN=0, Flags=.....C, BI=100, SSID="MEO-WiFi"
762 9.629842	AMPAKTechnol_7a:9b:: HitronTechno_f3:9a::	802.11	64 Null function (No data), SN=662, FN=0, Flags=...R..TC
763 9.653272	HitronTechno_f3:9a:: Broadcast	802.11	362 Beacon frame, SN=2626, FN=0, Flags=.....C, BI=100, SSID="FlyingNet"
764 9.674131	de:62:79:01:e2:39 Broadcast	802.11	116 QoS Data, SN=482, FN=0, Flags=.p..R..TC

Figura 37: Exemplo de uma transferência em que não é usado RTS/CTS

Na figura observa-se a ausência de frames RTS (Request to Send) e CTS (Clear to Send). A transferência de dados é feita diretamente com uma frame do tipo “QoS Data”, indicando que o mecanismo RTS/CTS não foi utilizado.

Usou-se o seguinte filtro para localizar **tramas RTS/CTS**:

wlan.fc.type_subtype == 0x1b || wlan.fc.type_subtype == 0x1c

992 12.632940	PTinovacao_29:a9:c0 XiaomiMobile_0a:80::	802.11	434 Probe Response, SN=25, FN=0, Flags=...R...C, BI=100, SSID="Masmorra do Sexo"
993 12.632944	AMPAKTechnol_7a:9b:: HitronTechno_f3:9a::	802.11	76 Request-to-send, Flags=.....C
994 12.632947	AMPAKTechnol_7a:9b:: AMPAKTechnol_7a:9b::	802.11	68 Clear-to-send, Flags=.....C
995 12.632949	AMPAKTechnol_7a:9b:: Broadcast	802.11	141 QoS Data, SN=1621, FN=0, Flags=.p.....TC
996 12.632952	HitronTechno_f3:9a:: AMPAKTechnol_7a:9b::	802.11	68 802.11 Block Ack, Flags=.....C

Figura 38: Exemplo de uma transferência de dados RTC/CTS