	<p>ATIVIDADE ACADÊMICA: Redes de Computadores: Internetworking, Roteamento e Transmissão</p> <p>TRABALHO: Atividade 2</p> <p>ALUNOS: Rafael Hansen Klauck</p>
---	---

Para fazer essa atividade, utilizei o Docker para criar um container Linux. Isso ajudou a ter um ambiente mais isolado e controlado, facilitando a captura e análise dos pacotes de rede. Além disso, rodar a análise dentro do container reduziu interferências de outros processos do sistema, deixando as informações dos pacotes mais fáceis de visualizar.

**1 - Descubra qual IP está configurado na sua máquina?**

**2 - Qual tamanho do MTU está sendo utilizado?**

```
root@97c2c871fd27:/# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: tunl0@NONE: <NOARP> mtu 1480 qdisc noop state DOWN group default qlen 1000
    link/ipip 0.0.0.0 brd 0.0.0.0
3: ip6tnl0@NONE: <NOARP> mtu 1452 qdisc noop state DOWN group default qlen 1000
    link/tunnel6 :: brd :: permaddr ee08:b2d5:d7e4::
18402: eth0@if18403: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:ac:11:00:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 172.17.0.2/16 brd 172.17.255.255 scope global eth0
        valid_lft forever preferred_lft forever
root@97c2c871fd27:/#
```

Para descobrir o IP e o tamanho do MTU foi utilizado o comando `ip a`. Como é possível visualizar na imagem, o IP da máquina é `172.17.0.2` e tamanho do MTU é `1500 bytes`.

**3 - Inicie uma coleta de pacotes, como o Wireshark ou tcpdump**

Para iniciar uma coleta de pacotes, foi utilizado o `tcpdump`. Primeiro foi necessário fazer a instalação `apt install -y tcpdump`. Após isso foi utilizado o comando `tcpdump -i eth0 -w captura.pcap` para iniciar a captura. No comando o `-i eth0` significa captura pacotes na interface `eth0` e o `-w captura.pcap` é o arquivo em que vai ser salva essa

captura. Após utilizar o comando, foi necessário abrir outro terminal e rodar a mesma instância.

## 4 - Envie usando o programa traceroute um datagrama de 2000 bytes

Para enviar um datagrama de 2000 bytes foi utilizado o comando *traceroute 8.8.8.8 2000*. Esse comando foi executado em um terminal separado do terminal que estava rodar o comando para capturar pacotes. Após executar o comando *traceroute*, foi necessário finalizar a captura no outro terminal.

Após esses passos, foi necessário transferir o arquivo *capturar.pcap* do container para meu computador e abrir o arquivo com o Wireshark para visualizar seu conteúdo e responder as demais perguntas.

## 5 - Dentro do cabeçalho do pacote IP, qual é o valor no campo de protocolo da camada superior?

## 6 - Quantos bytes há no cabeçalho IP?

The image shows a Wireshark packet capture of a network traffic. The packet list on the left shows a fragmented IP packet (protocol=UDP, offset=0, ID=88a3) reassembled in #2. The packet details pane on the right shows the IP header (version 4, length 1500, flags: 0x0, more fragments) and the UDP header (source address 172.17.0.2, destination address 8.8.8.8, stream index 0). The packet bytes pane on the right shows the raw data of the packet, including the IP header and the UDP payload.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=88a3) [Reassembled in #2]
2	0.000010	172.17.0.2	8.8.8.8	UDP	534	50348 - 33434 Len=1972
3	0.000078	172.17.0.1	172.17.0.2	ICMP	590	Time-to-live exceeded (Time to live exceeded in transit)
4	0.000122	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=87fa) [Reassembled in #5]
5	0.000127	172.17.0.2	8.8.8.8	UDP	534	40366 - 33435 Len=1972
6	0.000138	172.17.0.1	172.17.0.2	ICMP	590	Time-to-live exceeded (Time to live exceeded in transit)
7	0.000152	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=89a6) [Reassembled in #8]
8	0.000154	172.17.0.2	8.8.8.8	UDP	534	56959 - 33436 Len=1972
9	0.000160	172.17.0.1	172.17.0.2	ICMP	590	Time-to-live exceeded (Time to live exceeded in transit)
10	0.000172	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=82b7) [Reassembled in #11]
11	0.000174	172.17.0.2	8.8.8.8	UDP	534	37687 - 33437 Len=1972
12	0.000395	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=8786) [Reassembled in #13]
13	0.000310	172.17.0.2	8.8.8.8	UDP	534	55148 - 33438 Len=1972
14	0.000363	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=84d0) [Reassembled in #15]
15	0.000368	172.17.0.2	8.8.8.8	UDP	534	57759 - 33439 Len=1972
16	0.000450	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=8966) [Reassembled in #17]
17	0.000454	172.17.0.2	8.8.8.8	UDP	534	68742 - 33440 Len=1972
18	0.000627	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=83bc) [Reassembled in #19]
19	0.000631	172.17.0.2	8.8.8.8	UDP	534	44587 - 33441 Len=1972
20	0.000749	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=18bc) [Reassembled in #21]
21	0.000753	172.17.0.2	8.8.8.8	UDP	534	53334 - 33442 Len=1972
22	0.000833	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=18d2) [Reassembled in #23]
23	0.000837	172.17.0.2	8.8.8.8	UDP	534	30851 - 33443 Len=1972

Frame 11: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on Ethernet II, Src: 02:42:ac:11:00:02 (02:42:ac:11:00:02), Dst: 02:42:46:7b:e7:d0 (02:42:46:7b:e7:d0)

Internet Protocol Version 4, Src: 172.17.0.2, Dst: 8.8.8.8

0100 .... = Version: 4

.... 0101 = Header Length: 20 bytes (5)

0 Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

Total Length: 1500

Identification: 0x00a3 (163)

001 .... = Flags: 0x0, More fragments

... 0000 0000 0000 = Fragment Offset: 0

Time to Live: 1

Protocol: UDP (17)

Header Checksum: 0x0740 [validation disabled]

[Header checksum status: Unverified]

Source Address: 172.17.0.2

Destination Address: 8.8.8.8

[Reassembled IPv4 in frame: 2]

[Stream index: 0]

[Data (1488 bytes)]

Como podemos ver na imagem, o valor no campo de protocolo da camada superior é 17, que é UDP e o tamanho do cabeçalho IP é 20 bytes.

**7 - Quantos bytes estão no payload do datagrama IP? Explique como você determinou o número de bytes de carga útil.**

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.17.0.2	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=UDP 17, off=0, ID=00a3) [Reassembled in #2]
2	0.000018	172.17.0.2	8.8.8.8	UDP	534	56348 → 33434 Len=1972

O datagrama IP original tinha 1972 bytes antes da fragmentação. Como o cabeçalho IP ocupa 20 bytes, o payload total do datagrama é 1952 bytes (1972 - 20). Como esse valor excede o MTU da Ethernet (1500 bytes), ele foi fragmentado em dois pacotes menores: o primeiro fragmento contém 1480 bytes de payload e o segundo 472 bytes (1480 + 472 = 1952).

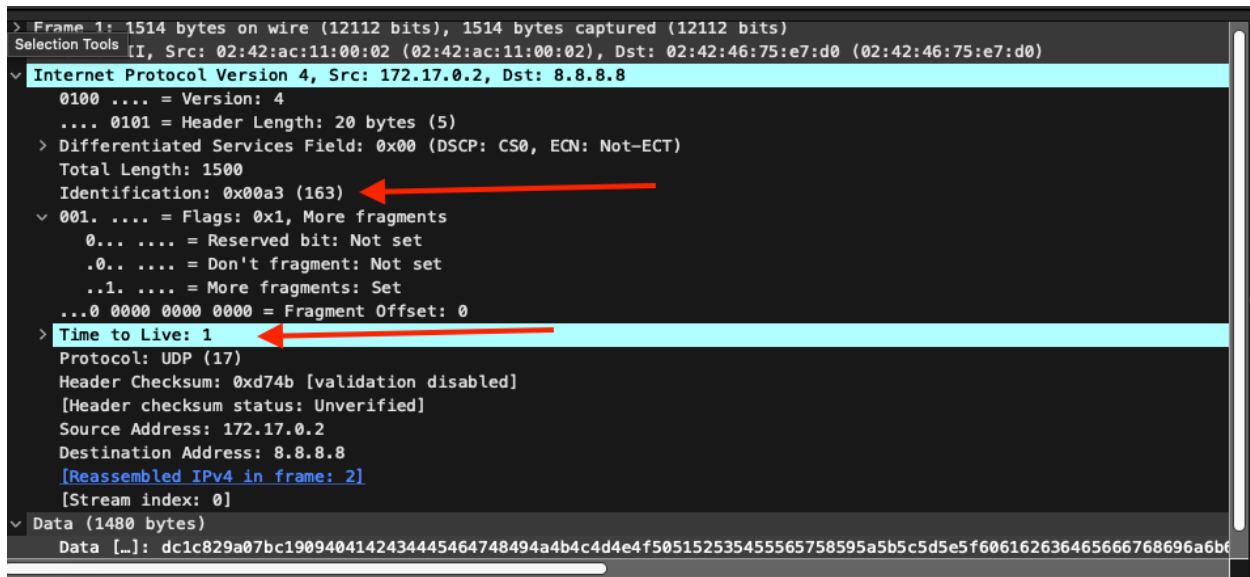
**8 - Este datagrama IP foi fragmentado? Explique como você determinou se o datagrama foi ou não fragmentado. Quais os valores dos fragmentos?**

Sim, o datagrama foi fragmentado. Para determinar isso, foi analisado as flags do cabeçalho IP, mais precisamente a flag MF(More Fragments). A imagem a seguir, mostra que essa flag está setada com 1, informando que há mais fragmentos após esse.

```
> Frame 1: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits)
> Ethernet II, Src: 02:42:ac:11:00:02 (02:42:ac:11:00:02), Dst: 02:42:46:75:e7:d0 (02:42:46:75:e7:d0)
v Internet Protocol Version 4, Src: 172.17.0.2, Dst: 8.8.8.8
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1500
    Identification: 0x00a3 (163)
  v 001. .... = Flags: 0x1, More fragments
    0... .... = Reserved bit: Not set
    .0.. .... = Don't fragment: Not set
    ..1. .... = More fragments: Set
    ...0 0000 0000 0000 = Fragment Offset: 0
  > Time to Live: 1
  Protocol: UDP (17)
  Header Checksum: 0xd74b [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 172.17.0.2
  Destination Address: 8.8.8.8
  [Reassembled IPv4 in frame: 2]
  [Stream index: 0]
> Data (1480 bytes)
```

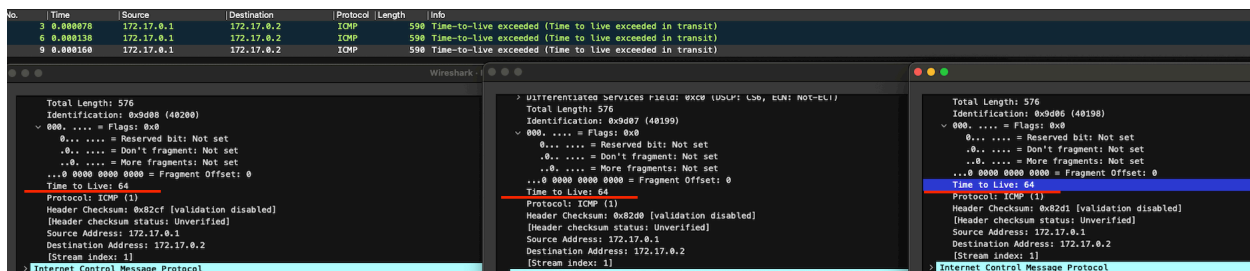
Assim como já mencionado na resposta da pergunta 7, o tamanho do primeiro fragmento é 1500 bytes (1480 bytes de payload + 20 bytes de cabeçalho) e do segundo fragmento 512 bytes (492 bytes de payload + 20 bytes de cabeçalho).

### 9 - Qual é o valor do campo Identification e do campo TTL?



Como podemos ver na imagem, o TTL é 1 e o Identification é 0x00a3 (163 em decimal)

**10 - Esses valores permanecem inalterados para todas as respostas ICMP TTL excedidas enviadas ao seu computador pelo roteador mais próximo (primeiro salto)? Por que?**



As respostas ICMP TTL Exceeded enviadas pelo roteador mais próximo ao meu computador não mantêm o mesmo valor no campo Identification. O campo Identification é um identificador único para cada datagrama IP e muda na resposta ICMP, porque essa resposta é um novo pacote gerado pelo roteador. Já o campo TTL também pode mudar, porque o roteador define um novo valor inicial para a resposta ICMP. Além disso, esse TTL diminui a cada roteador pelo qual a resposta passa antes de chegar ao meu

computador. Portanto, o valor de Identification na resposta ICMP não permanece inalterado, pois a resposta é um novo datagrama IP gerado pelo roteador. Já o TTL geralmente muda, pois cada roteador define um TTL inicial diferente para a resposta ICMP. No meu caso, o TTL se manteve igual, mas isso pode variar dependendo da rede e dos roteadores intermediários.