

# Protocolo IPv4

André Gonçalves (A80368), Diogo Gonçalves (A81860), Maria Pires(A

September 9, 2019

## 1 Questões e Respostas

### 1.1 Parte I

#### 1.1.1 Pergunta 1

Active o wireshark ou o tcpdump no pc h1. Numa shell de h1, execute o comando traceroute -I para o endereço IP do host s4.

ALINEA 1 - A

Registe e analise o tráfego ICMP enviado por h1 e o tráfego ICMP recebido como resposta. Comente os resultados face ao comportamento esperado.

ALINEA 1 - b

Qual deve ser o valor inicial mínimo do campo TTL para alcançar o destino s4? Verifique na prática que a sua resposta está correta.

ALINEA 1 - c

Qual o valor médio do tempo de ida-e-volta (Round-Trip Time) obtido?

ALINEA 1 - D

### 1.1.2 Pergunta 2

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

O endereço IP da interface ativa do computador em causa é 196.168.100.167

```
Internet Protocol Version 4, Src: 192.168.100.167, Dst: 193.136.9.254
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 512
    Identification: 0x613f (24895)
  ▶ Flags: 0x0000
  ▶ Time to live: 1
    Protocol: ICMP (1)
    Header checksum: 0x65e8 [validation disabled]
    [Header checksum status: Unverified]
  Source: 192.168.100.167
  Destination: 193.136.9.254
```

Figure 1: Legenda

**Qual é o valor do campo protocolo? O que identifica?**

O protocolo usado foi o ICMP ???

```
▼ Internet Protocol Version 4, Src: 192.168.100.167, Dst: 193.136.9.254
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 512
    Identification: 0x613f (24895)
  ▶ Flags: 0x0000
  ▶ Time to live: 1
    Protocol: ICMP (1)
    Header checksum: 0x65e8 [validation disabled]
    [Header checksum status: Unverified]
  Source: 192.168.100.167
  Destination: 193.136.9.254
```

Figure 2: Legenda

**Quantos bytes tem o cabeçalho IP(v4)? Quantos bytes tem o campo de dados (payload) do datagrama? Como se calcula o tamanho do payload?**

O cabeçalho do IPv4 tem um total de 20 bytes, o payload tem 506 bytes. O cabeçalho é calculado retirando ao número total de bytes transferidos o número de bytes dos headers.

```

Frame Number: 21
Frame Length: 526 bytes (4208 bits)
Capture Length: 526 bytes (4208 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: eth:ethertype:ip:icmp:data]
[Coloring Rule Name: ICMP]
[Coloring Rule String: icmp || icmpv6]
▼ Ethernet II, Src: AsustekC_1d:bc:23 (9c:5c:8e:1d:bc:23), Dst: Vmware_d2:19:f0 (
  ► Destination: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
  ► Source: AsustekC_1d:bc:23 (9c:5c:8e:1d:bc:23)
  Type: IPv4 (0x0800)
▼ Internet Protocol Version 4, Src: 192.168.100.167, Dst: 193.136.9.254
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ► Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 512
  Identification: 0x6140 (24896)
▼ Flags: 0x0000
  0... .. = Reserved bit: Not set
  .0... .. = Don't fragment: Not set
  ..0. .... = More fragments: Not set
  ...0 0000 0000 0000 = Fragment offset: 0
  ► Time to live: 1

```

Figure 3: Legenda

### O datagrama IP foi fragmentado? Justifique

Não, pois a flag "Fragment offset" tem o valor 0.

```

▼ Internet Protocol Version 4, Src: 192.168.100.167, Dst:
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ► Differentiated Services Field: 0x00 (DSCP: CS0, ECN:
  Total Length: 512
  Identification: 0x613f (24895)
▼ Flags: 0x0000
  0... .. = Reserved bit: Not set
  .0... .. = Don't fragment: Not set
  ..0. .... = More fragments: Not set
  ...0 0000 0000 0000 = Fragment offset: 0
  ► Time to live: 1

```

Figure 4: Legenda

Ordene os pacotes capturados de acordo com o endereço IP fonte (e.g., selecionando o cabeçalho da coluna Source), e analise a sequência de tráfego ICMP gerado a partir do endereço IP atribuído à interface da sua máquina. Para a sequência de mensagens ICMP enviadas pelo seu computador, indique que campos do cabeçalho IP variam de pacote para pacote.

Os campos do cabeçalho IP que variam de pacote para pacote são os TTL (Time to live) e a Identification.

```

Wireshark · Packet 23 · enp3s0f1
▶ Frame 23: 526 bytes on wire (4208 bits), 526 bytes captured (4208 bits) on interface 0
▶ Ethernet II, Src: AsustekC_1d:bc:23 (9c:5c:8e:1d:bc:23), Dst: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
▼ Internet Protocol Version 4, Src: 192.168.100.167, Dst: 193.136.9.254
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 512
      Identification: 0x6142 (24898)
      ▼ Flags: 0x0000
        0... .. = Reserved bit: Not set
        .0.. .. = Don't fragment: Not set
        ..0. ... = More fragments: Not set
        ...0 0000 0000 0000 = Fragment offset: 0
      ▶ Time to live: 2
        Protocol: ICMP (1)
        Header checksum: 0x64e5 [validation disabled]
        [Header checksum status: Unverified]
        Source: 192.168.100.167
        Destination: 193.136.9.254
▶ Internet Control Message Protocol

```

Figure 5: Legenda

```

Wireshark · Packet 30 · enp3s0f1
▶ Frame 30: 526 bytes on wire (4208 bits), 526 bytes captured (4208 bits) on interface 0
▶ Ethernet II, Src: AsustekC_1d:bc:23 (9c:5c:8e:1d:bc:23), Dst: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
▼ Internet Protocol Version 4, Src: 192.168.100.167, Dst: 193.136.9.254
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 512
      Identification: 0x6149 (24905)
      ▼ Flags: 0x0000
        0... .. = Reserved bit: Not set
        .0.. .. = Don't fragment: Not set
        ..0. ... = More fragments: Not set
        ...0 0000 0000 0000 = Fragment offset: 0
      ▶ Time to live: 4
        Protocol: ICMP (1)
        Header checksum: 0x62de [validation disabled]
        [Header checksum status: Unverified]
        Source: 192.168.100.167
        Destination: 193.136.9.254
▶ Internet Control Message Protocol

```

Figure 6: Legenda

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

O campo de Identificação do datagrama aumenta 1 por pacote enquanto que o TTTL aumenta 1 de 3 em 3 pacotes.

```

Wireshark - Packet 37 - enp3s0f1

Frame 37: 554 bytes on wire (4432 bits), 554 bytes captured (4432 bits) on interface 0
Ethernet II, Src: Vmware_d2:19:f0 (00:0c:29:d2:19:f0), Dst: AsustekC_1d:bc:23 (9c:5c:8e:1d:bc:23)
  Destination: AsustekC_1d:bc:23 (9c:5c:8e:1d:bc:23)
  Source: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
  Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.100.254, Dst: 192.168.100.167
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
  Total Length: 540
  Identification: 0xd3bc (54204)
  Flags: 0x0000
    0... .. = Reserved bit: Not set
    .0... .. = Don't fragment: Not set
    ..0... .. = More fragments: Not set
    ...0 0000 0000 0000 = Fragment offset: 0
  Time to live: 64
  Protocol: ICMP (1)
  Header checksum: 0x596e [validation disabled]
  [Header checksum status: Unverified]
  Source: 192.168.100.254
  Destination: 192.168.100.167
Internet Control Message Protocol

```

Figure 7: Legenda

```

Wireshark - Packet 38 - enp3s0f1

Frame 38: 554 bytes on wire (4432 bits), 554 bytes captured (4432 bits) on interface 0
Ethernet II, Src: Vmware_d2:19:f0 (00:0c:29:d2:19:f0), Dst: AsustekC_1d:bc:23 (9c:5c:8e:1d:bc:23)
  Destination: AsustekC_1d:bc:23 (9c:5c:8e:1d:bc:23)
  Source: Vmware_d2:19:f0 (00:0c:29:d2:19:f0)
  Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.100.254, Dst: 192.168.100.167
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
  Total Length: 540
  Identification: 0xd3bd (54205)
  Flags: 0x0000
    0... .. = Reserved bit: Not set
    .0... .. = Don't fragment: Not set
    ..0... .. = More fragments: Not set
    ...0 0000 0000 0000 = Fragment offset: 0
  Time to live: 64
  Protocol: ICMP (1)
  Header checksum: 0x596d [validation disabled]
  [Header checksum status: Unverified]
  Source: 192.168.100.254
  Destination: 192.168.100.167
Internet Control Message Protocol

```

Figure 8: Legenda

Ordene o tráfego capturado por endereço destino e encontre a série de respostas ICMP TTL exceeded enviadas ao seu computador. Qual é o valor do campo TTL? Esse valor permanece constante para todas

**as mensagens de resposta ICMP TTL exceeded enviados ao seu host?  
Porquê?**

O TTL é constante a 64 unidades o que significa que o pacote encontra-se em routing loop. (figuras 7 e 8)

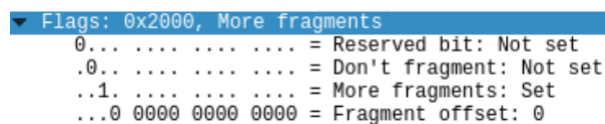
### 1.1.3 Pergunta 3

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

Houve necessidade de fragmentar o pacote inicial porque o MTU(Maximum Transmission Unit) size para ethernet é no máximo 1500 bytes e o pacote enviado foi de 3125 bytes. Isto significa que foi necessário a fragmentação do pacote em 3 pacotes mais pequenos.

**Imprima o primeiro fragmento do datagrama IP segmentado. Que informação no cabeçalho indica que o datagrama foi fragmentado? Que informação no cabeçalho IP indica que se trata do primeiro fragmento? Qual é o tamanho deste datagrama IP?**

O que indica que o datagrama foi fragmentado é a flag "More Fragments" encontrar-se a 1. O que indica que se trata do primeiro fragmento é a flag "Fragment offset" encontrar-se a 0. O tamanho deste datagrama IP é de 1500 bytes, que corresponde ao máximo MTU na conexão ethernet.



```
▼ Flags: 0x2000, More fragments
0... .. = Reserved bit: Not set
.0.. .. = Don't fragment: Not set
..1. .... = More fragments: Set
...0 0000 0000 0000 = Fragment offset: 0
```

Figure 9: Legenda

**Imprima o segundo fragmento do datagrama IP original. Que informação do cabeçalho IP indica que não se trata do 1º fragmento? Há mais fragmentos? O que nos permite afirmar isso?**

É possível concluir que existem mais datagramas porque a flag "More Fragments" encontra-se a 1. Podemos também concluir que não se trata do primeiro fragmento quando o "Fragment offset" é diferente de 0.

```

▼ Flags: 0x20b9, More fragments
  0... .... = Reserved bit: Not set
  .0... .... = Don't fragment: Not set
  ..1. .... = More fragments: Set
  ...0 0000 1011 1001 = Fragment offset: 185

```

Figure 10: Legenda

**Quantos fragmentos foram criados a partir do datagrama original?  
Como se detecta o último fragmento correspondente ao datagrama original?**

Foram criados 3 fragmentos a partir do datagrama original. Sabemos que é o último fragmento quando a flag "More fragments" encontra-se a 0.

```

▼ Flags: 0x0172
  0... .... = Reserved bit: Not set
  .0... .... = Don't fragment: Not set
  ..0. .... = More fragments: Not set
  ...0 0001 0111 0010 = Fragment offset: 370

```

Figure 11: Legenda

**Indique, resumindo, os campos que mudam no cabeçalho IP entre os diferentes fragmentos, e explique a forma como essa informação permite reconstruir o datagrama original.**

A identificação mantém-se constante nos diferentes fragmentos. A partir da identificação conseguimos reconhecer quais são os fragmentos correspondentes ao datagrama original o que nos irá permitir reconstruí-lo. A ordem crescente da flag "Fragment offset" permite ordenar os fragmentos pela ordem original de forma a reconstruir o datagrama.

## 1.2 Parte II

### 1.2.1 Pergunta 1

**Indique que endereços IP e máscaras de rede foram atribuídos pelo CORE a cada equipamento. Para simplificar, pode incluir uma imagem que ilustre de forma clara a topologia definida e o endereçamento usado.**

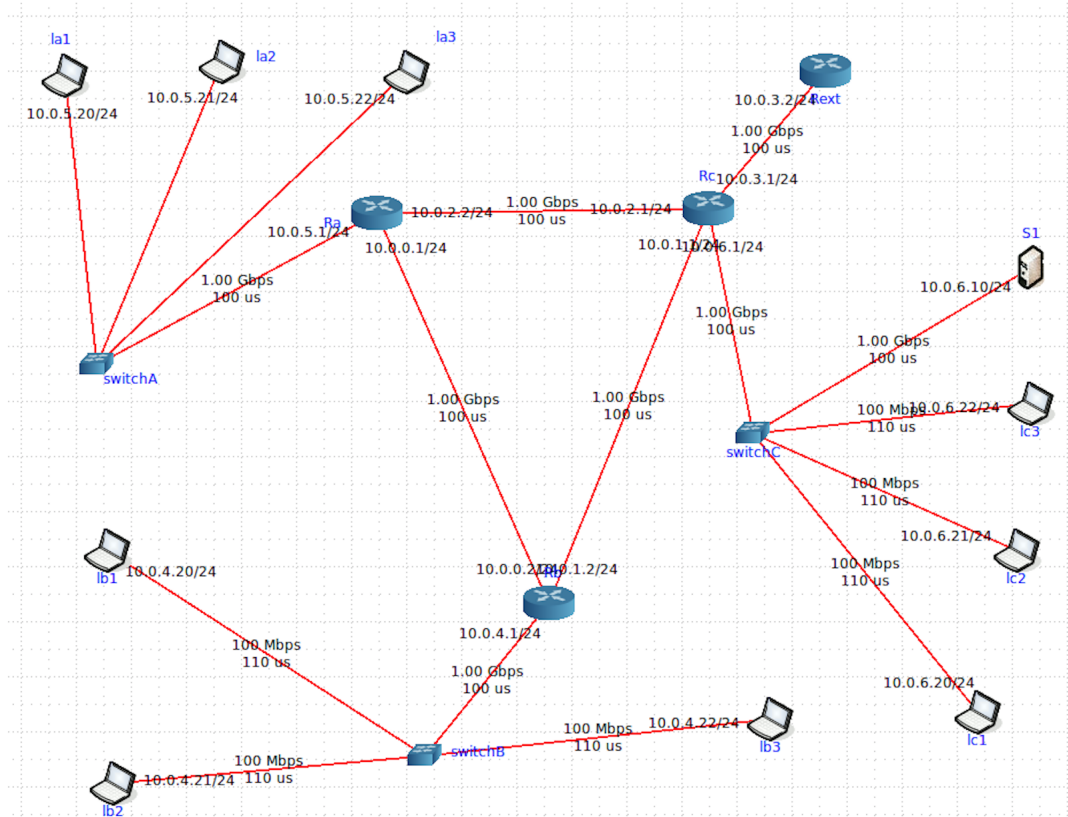


Figure 12: Topologia de Rede

Utilizando o laptop la1 com o endereço IP de 10.0.5.20/24 , conseguimos concluir que a máscara terá um total de 24 bytes, ou seja a máscara de rede será 255.255.255.0

### Tratam-se de endereços públicos ou privados? Porquê?

Os endereços atribuídos para internets privadas (sem ligação ao IP global) são:

Start	End	CIDR
192.168.0.0	192.168.255.255	16
172.16.0.0	172.31.255.255	12
10.0.0.0	10.255.255.255	8

Neste caso trata-se de um endereço privado porque o endereço 10.5.10.0 -



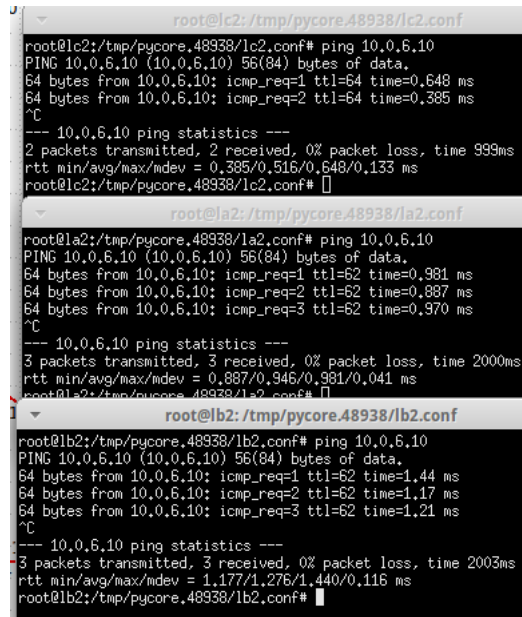
10.5.10.255 faz parte do endereço 10.0.0.0 - 10.255.255.255

### Porque razão não é atribuído um endereço IP aos switches?

O switch não recebe qualquer IP porque este opera no nível 2 do TCP/IP enquanto o IPv4 é de nível 3.

Usando o comando ping certifique-se que existe conectividade IP entre os laptops dos vários departamentos e o servidor do departamento C (basta certificar-se da conectividade de um laptop por departamento).

Sim, existe como podemos verificar pela figura abaixo:



```
root@lc2:/tmp/pycore.48938/lc2.conf
root@lc2:/tmp/pycore.48938/lc2.conf# ping 10.0.6.10
PING 10.0.6.10 (10.0.6.10) 56(84) bytes of data:
64 bytes from 10.0.6.10: icmp_req=1 ttl=64 time=0.648 ms
64 bytes from 10.0.6.10: icmp_req=2 ttl=64 time=0.385 ms
^C
--- 10.0.6.10 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.385/0.516/0.648/0.133 ms
root@lc2:/tmp/pycore.48938/lc2.conf#

root@la2:/tmp/pycore.48938/la2.conf
root@la2:/tmp/pycore.48938/la2.conf# ping 10.0.6.10
PING 10.0.6.10 (10.0.6.10) 56(84) bytes of data:
64 bytes from 10.0.6.10: icmp_req=1 ttl=62 time=0.981 ms
64 bytes from 10.0.6.10: icmp_req=2 ttl=62 time=0.887 ms
64 bytes from 10.0.6.10: icmp_req=3 ttl=62 time=0.970 ms
^C
--- 10.0.6.10 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.887/0.946/0.981/0.041 ms
root@la2:/tmp/pycore.48938/la2.conf#

root@lb2:/tmp/pycore.48938/lb2.conf
root@lb2:/tmp/pycore.48938/lb2.conf# ping 10.0.6.10
PING 10.0.6.10 (10.0.6.10) 56(84) bytes of data:
64 bytes from 10.0.6.10: icmp_req=1 ttl=62 time=1.44 ms
64 bytes from 10.0.6.10: icmp_req=2 ttl=62 time=1.17 ms
64 bytes from 10.0.6.10: icmp_req=3 ttl=62 time=1.21 ms
^C
--- 10.0.6.10 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.177/1.276/1.440/0.116 ms
root@lb2:/tmp/pycore.48938/lb2.conf#
```

Figure 13: Topologia de Rede

Verifique se existe conectividade IP do router de acesso Rext para o servidor S1

Sim, existe como podemos verificar pela figura abaixo:

```
root@Rext: /tmp/pycore.48938/Rext.conf
root@Rext:/tmp/pycore.48938/Rext.conf# ping 10.0.6.10
PING 10.0.6.10 (10.0.6.10) 56(84) bytes of data:
64 bytes from 10.0.6.10: icmp_req=1 ttl=63 time=1.28 ms
64 bytes from 10.0.6.10: icmp_req=2 ttl=63 time=0.889 ms
64 bytes from 10.0.6.10: icmp_req=3 ttl=63 time=0.698 ms
64 bytes from 10.0.6.10: icmp_req=4 ttl=63 time=1.01 ms
^C
--- 10.0.6.10 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3001ms
rtt min/avg/max/mdev = 0.698/0.970/1.280/0.214 ms
root@Rext:/tmp/pycore.48938/Rext.conf#
```

Figure 14: Topologia de Rede

### 1.2.2 Pergunta 2

Execute o comando `netstat -rn` por forma a poder consultar a tabela de encaminhamento unicast (IPv4). Inclua no seu relatório as tabelas de encaminhamento obtidas; interprete as várias entradas de cada tabela. Se necessário, consulte o manual respetivo (`man netstat`).

ALINEA 2 - A

Diga, justificando, se está a ser usado encaminhamento estático ou dinâmico (sugestão: analise que processos estão a correr em cada sistema).

ALINEA 2 - B

Admita que, por questões administrativas, a rota por defeito (0.0.0.0 ou default) deve ser retirada definitivamente da tabela de encaminhamento do servidor S1 localizado no departamento C. Use o comando `route delete` para o efeito. Que implicações tem esta medida para os utilizadores da empresa que acedem ao servidor. Justifique.

ALINEA 2 - C

Adicione as rotas estáticas necessárias para restaurar a conectividade para o servidor S1, por forma a contornar a restrição imposta na alínea c). Utilize para o efeito o comando `route add` e registe os comandos que usou

ALLINEA 2 - D

Teste a nova política de encaminhamento garantindo que o servidor está novamente acessível, utilizando para o efeito o comando `ping`. Registe a nova tabela de encaminhamento do servidor.

ALINEA 2 - E

### 1.2.3 Pergunta 3

Considere que dispõe apenas do endereço de rede IP 172.XX.48.0/20, em que XX é o decimal correspondendo ao seu número de grupo (PLXX). Defina um novo esquema de endereçamento para as redes dos departamentos (mantendo a rede de acesso e core inalteradas) e atribua endereços às interfaces dos vários sistemas envolvidos. Deve justificar as opções usadas

ALINEA 3 - A

Qual a máscara de rede que usou (em formato decimal)? Quantos hosts IP pode interligar em cada departamento? Justifique

ALINEA 3 - B

Garanta e verifique que conectividade IP entre as várias redes locais da organização MIEI-RC é mantida. Explique como procedeu

ALINEA 3 - C

## 2 Conclusões

O *internet Protocol (IP)*, que designa a interface da máquina numa rede, define um datagrama ou pacote independente da ligação de dados. Este datagrama é composto por um cabeçalho que nos indica a versão do protocolo, o tipo de dados transportados, o TTL (time to live) que designa o número de saltos na rede, entre outras informações que permite, a transferência do pacote sem erros. Com a aplicação dos conceitos teóricos na resolução deste trabalho obtemos uma perspetiva do funcionamento e aplicação destes.

Para qualquer protocolo IP, de modo a poder transmitir o datagrama para o nível 2 este é encapsulado consoante o seu tamanho, este encapsulamento é feito através da fragmentação dos pacotes de IP em vários pacotes de menores dimensões como pudemos constatar com a análise dos resultados obtidos pelo *Wireshark*, e estes são reconstruídos quando chegam ao destino através dos bits de identificação no cabeçalho, os quais permitem ordenar o datagrama de forma correta no destino. Também observamos na prática esta ordenação através do campo "More Fragments" que indica qual é o último fragmento do pacote. Através do ICMP conseguimos efetuar a deteção dos poucos erros causados pela fragmentação.

O endereço IP é associado a cada interface, observou-se no trabalho a existência de endereços privados através da interpretação da dos bits no endereço.

Os endereços são compostos por duas partes sendo a primeira definida pelo endereço de rede e a segunda pelo endereço do host. O endereçamento de uma rede sem classes é feito através de uma máscara de 32 bits que permite obter tabelas de encaminhamento menores face ao endereçamento com classes.

Quanto ao encaminhamento, as tabelas de encaminhamento incluem o endereço de rede do destino e a máscara, e caso se aplique, o endereço IP do próximo salto. Constatamos na prática os dois tipos de encaminhamento existentes, estático e dinâmico e o funcionamento da transmissão de tráfego e a conectividade entre os departamentos.