

**Redes de Computadores**

*Desenvolvimento de uma Aplicação Download e Configuração e Estudo de uma Rede de Computadores*

3MIEIC04 - Grupo 7

**Diogo Almeida –** [**up201806630@fe.up.pt**](mailto:up201806630@fe.up.pt)

**Pedro Queirós –** [**up201806329@fe.up.pt**](mailto:up201806329@fe.up.pt)

Conteúdo

[Sumário 4](#_Toc59626488)

[Introdução 4](#_Toc59626489)

[Parte 1 – Desenvolvimento de uma Aplicação *Download* 4](#_Toc59626490)

[Arquitetura 4](#_Toc59626491)

[Resultados 4](#_Toc59626492)

[Parte 2 – Configuração e Estudo de uma Rede de Computadores 5](#_Toc59626493)

[**Experiência 1 – Configurar uma rede IP** 5](#_Toc59626494)

[O que são pacotes ARP e para que é que são usados? 5](#_Toc59626495)

[Quais são os endereços MAC e IP dos pacotes ARP e porquê? 5](#_Toc59626496)

[Que pacotes são gerados pelo comando ping? 5](#_Toc59626497)

[Quais são os endereços MAC e IP nos pacotes do comando ping? 5](#_Toc59626498)

[Como determinar se uma trama de Ethernet recebida é ARP, IP, ICMP? 5](#_Toc59626499)

[Como determinar o tamanho de uma trama recebida? 6](#_Toc59626500)

[O que é a interface *loopback* e porque é que é importante? 6](#_Toc59626501)

[**Experiência 2 – Implementar duas LANs virtuais num *switch*** 6](#_Toc59626502)

[Como configurar a vlany0? 6](#_Toc59626503)

[Quantos domínios de transmissão existem? Como se pode concluir a partir dos *logs*? 6](#_Toc59626504)

[**Experiência 3 – Configuração de um *Router* em Linux** 6](#_Toc59626505)

[Que rotas existem nos *tux’s*? Quais são os seus significados? (talvez incompleta) 6](#_Toc59626506)

[Que informação está contida numa entrada de uma tabela de encaminhamento? 7](#_Toc59626507)

[Quais as mensagens ARP e endereços MAC observados e porquê? 7](#_Toc59626508)

[Que pacotes ICMP são observados e porquê? 8](#_Toc59626509)

[Quais os endereços IP e MAC associados aos pacotes ICMP e porquê? 8](#_Toc59626510)

[**Experiência 4 – Configuração de um *router* comercial e implementação de NAT** 8](#_Toc59626511)

[Como configurar uma rota estática num *router* comercial? (incompleta) 8](#_Toc59626512)

[Quais são os caminhos seguidos pelos pacotes nas experiências realizadas e porquê? 8](#_Toc59626513)

[Como configurar NAT num *router* comercial? 8](#_Toc59626514)

[O que é que faz o NAT? 9](#_Toc59626515)

[**Experiência 5 – DNS** 9](#_Toc59626516)

[Como configurar um serviço DNS num *host*? 9](#_Toc59626517)

[Que pacotes são trocados pelo DNS e que informação transportam? 9](#_Toc59626518)

[Experiência 6 – Conexões TCP 9](#_Toc59626519)

[Quantas conexões TCP são abertas pela aplicação FTP desenvolvida? 9](#_Toc59626520)

[Em que conexão é transportada a informação de controlo FTP? 9](#_Toc59626521)

[Quais são as fases de uma conexão FTP? 10](#_Toc59626522)

[Como funciona o mecanismo ARQ FTP? Quais são os parâmetros TCP relevantes? Que informação importante pode ser observada nos *logs*? 10](#_Toc59626523)

[Como funciona o mecanismo de controlo de congestionamento TCP? Quais são os parâmetros relevantes? Como evoluiu o rendimento da ligação de dados ao longo do tempo? Está de acordo com o mecanismo de controlo de congestionamento TCP? (incompleto) 10](#_Toc59626524)

[A conexão de dados TCP é afetada pelo aparecimento de uma segunda conexão TCP? De que maneira é afetada? 10](#_Toc59626525)

[Conclusões 10](#_Toc59626526)

[Anexo 11](#_Toc59626527)

[Imagens 11](#_Toc59626528)

# Sumário

Este trabalho foi desenvolvido no âmbito da unidade curricular de Redes de Computadores com o objetivo de configurar e estudar uma rede de computadores e ainda desenvolver uma aplicação de transferência de ficheiros aplicando um protocolo FTP *(File Transfer Protocol*).

O trabalho foi concluído com sucesso, visto que a rede de computadores foi corretamente estabelecida e a aplicação desenvolvida permite a transferência correta de ficheiros através de um protocolo FTP.

# Introdução

O objetivo do trabalho consiste na criação de software que permita a transferência de ficheiros aplicando um protocolo FTP e criação de uma rede de computadores e estudo da mesma, através de uma série de experiências fornecidas no guião do trabalho.

# Parte 1 – Desenvolvimento de uma Aplicação *Download*

Na primeira parte deste trabalho foi desenvolvida uma aplicação *download* na linguagem de programação C. Esta aplicação recebe, como argumento, o seguinte formato: *ftp://[<user>:<password>@]<host>/<url-path>*. Esta aplicação é capaz de fazer a transferência de qualquer tipo de ficheiro de um servidor FTP. Com esse objetivo, foi estudado o RFC959, respetivo a FTP, e o RFC1738, respetivo ao tratamento de informação contida num URL.

## Arquitetura

Primeiramente, é efetuado o processamento do URL através da função *parseUrlInfo*, que guarda o *username*, a *password*, o *host*, o *path* e o *filename* na *struct urlInfo*. Caso o *username* e a password não sejam fornecidos, estes são colocados com os valores *anonymous* e *anypassword* respetivamente. Caso apenas seja fornecido o *username*, a aplicação pergunta ao utilizador a *password* respetiva. De seguida, obtém-se o endereço IP através da função *getIpAddressFromHost*. Nesta aplicação é sempre usada a porta 21. Depois, é aberto o *socket* pelo qual o cliente e o servidor vão comunicar, através da função *ftpStartConnection*, e efetuado o login com os comandos *User user* e *Pass password*, utilizando a função *ftpLoginIn*. Após efetuado o *login*, é efetuada a entrada em modo passivo, através do comando *Pasv*, e processada a resposta na qual é fornecido o endereço IP e os valores necessários para o cálculo da porta a utilizar para abrir o *socket* responsável pela troca de dados. Para tal, é utilizada a função *ftpPassiveMode*. Após a entrada no modo passivo, a função *ftpRetrieveFile*, que envia o comando RETR filename para pedir a transferência do ficheiro respetivo e o *download* é efetuado pela função *ftpDownloadAndCreateFile*. Finalmente, são encerradas todas as ligações. Ao longo da execução do programa, todas as respostas do servidor são processadas e em caso de erro, o programa termina com a mensagem respetiva à situação ocorrida.

## Resultados

A aplicação desenvolvida foi testada em diversas situações: com username e password fornecidos, apenas username fornecido, sem username nem password, fornecidos, com diferentes tipos de ficheiros, com um ficheiro que não existia entre outros. Todos os testes a que o programa foi submetido foram superados com sucesso. A execução e funcionamento da aplicação podem ser consultados Figura 1.

# Parte 2 – Configuração e Estudo de uma Rede de Computadores

## **Experiência 1 – Configurar uma rede IP**

### O que são pacotes ARP e para que é que são usados?

ARP (*Address Resolution Protocol*) é um protocolo da camada de rede utilizado para converter um endereço IP num endereço físico designado endereço MAC (*Media Access Control*). Este endereço de *hardware* é único e identifica um específico nó numa rede.

Os pacotes ARP são utilizados para descobrir o endereço físico de um *host*. Quando um *host* pretende obter o endereço MAC de outro *host* na mesma rede, tendo apenas o endereço IP, envia em *broadcast* um pacote para a rede TCP/IP. Assim sendo, o *host* com o respetivo endereço IP envia uma resposta com o seu endereço físico.

### Quais são os endereços MAC e IP dos pacotes ARP e porquê?

Os pacotes ARP contêm o endereço MAC e IP do transmissor e do recetor. No pacote enviado pelo *host* que pretende obter o endereço físico de outro *host*, o endereço MAC do recetor é ignorado, uma vez que não se sabe o valor. Deverão ser verificados os *logs* registados nas Figuras 2 e 3 para consultar estes endereços.

### Que pacotes são gerados pelo comando ping?

O comando ping gera pacotes ICMP (*Internet Control Message Protocol*). Inicialmente, se o endereço físico de destino não estiver contido na tabela ARP, são enviados pacotes ARP para obter esse endereço.

Este comando é utilizado para identificar o alcance de um *host*, indicando a existência de erros na rede estabelecida, perda de pacotes e ainda estatísticas dos resultados.

### Quais são os endereços MAC e IP nos pacotes do comando ping?

Os pacotes ICMP, gerados pelo comando *ping*, contêm os endereços MAC e IP do transmissor e do recetor. Deverão ser verificados os *logs* registados nas Figuras 4 e 5 para consultar estes endereços.

### Como determinar se uma trama de Ethernet recebida é ARP, IP, ICMP?

Analisando o Ethernet *header* da trama recebida, obtém-se o *EtherType*, composto por 2 octetos. Se o seu valor for 0x0800, a trama é do tipo IP. Sendo uma trama deste tipo, se o campo *Protocol* do IP header tiver o valor de 1, o protocolo aplicado é ICMP. Se o seu valor for 0x0806, a trama é do tipo ARP. Deverão ser verificados os *logs* registados nas Figuras 6 e 7.

### Como determinar o tamanho de uma trama recebida?

No Ethernet *header* de uma trama recebida existem 2 bytes que contêm o valor relativo ao tamanho total da trama. Deverá ser verificado o *log* registado na Figura 8.

### O que é a interface *loopback* e porque é que é importante?

O termo *loopback* refere-se ao encaminhamento de sinais digitais ou fluxo de dados para a fonte de origem do envio, sem qualquer processamento ou modificação. Este método é utilizado para testar a rede estabelecida para comunicação.

A interface *loopback* é uma interface virtual da rede utilizada por um computador para comunicar com ele mesmo. Esta interface é utilizada principalmente para diagnóstico e resolução de problemas da rede estabelecida. Para além disso, é também utilizada para estabelecer a ligação a servidores na máquina local. Deverá ser verificado o *log* registado na Figura 9.

## **Experiência 2 – Implementar duas LANs virtuais num *switch***

### Como configurar a vlany0?

Para se configurar a vlany0 deverá conectar-se a porta T4 à porta *switch* *console*. De seguida, conecta-se a porta T3 à entrada S0 do *tux* a que se pretende que conectar a consola do *switch*. Após as ligações estarem corretamente estabelecidas, terá de se abrir o *GTKTerm* do *tux* escolhido para criar a vlan com os seguintes comandos:

>configure terminal

>vlan y0

>end

>configure terminal

>interface fastethernet 0/(número da porta escolhida)

>switchport mode access

>switchport access vlan y0

>end

### Quantos domínios de transmissão existem? Como se pode concluir a partir dos *logs*?

Nesta experiência, existem dois domínios de transmissão. A partir da análise dos *logs*, é possível concluir que ao efetuar ping *broadcast* a partir do *tux* 3, este recebe resposta do *tux* 4, mas não do *tux* 2. Ao efetuar ping *broadcast* a partir do *tux* 2, este não recebe nenhuma resposta por parte do *tux* 3 e do *tux* 4, concluindo-se a existência de dois domínios de transmissão.

## **Experiência 3 – Configuração de um *Router* em Linux**

## Que rotas existem nos *tux’s*? Quais são os seus significados? (talvez incompleta)

**Rotas do *tux* 2:**

- vlan 60 (192.16.60.0) pela *gateway* 192.16.61.253

- vlan 61 (192.16.61.0) pela *gateway* 192.16.61.1

**Rotas do *tux* 3:**

- vlan 60 (192.16.60.0) pela *gateway* 192.16.60.1

- vlan 61 (192.16.61.0) pela *gateway* 192.16.60.254

**Rotas do *tux* 4:**

- vlan 60 (192.16.60.0) pela *gateway* 192.16.60.254

- vlan 61 (192.16.61.0) pela *gateway* 192.16.61.253

### Que informação está contida numa entrada de uma tabela de encaminhamento?

Uma entrada da tabela de encaminhamento tem os seguintes parâmetros:

***Destination***: valor com o qual o endereço IP de destino é correspondido.

***Gateway***: rota através da qual o destino especificado pode ser alcançado.

***Genmask***: máscara de endereço utilizada para determinar o endereço IP corresponde ao parâmetro *Destination*.

***Flags***: contém informação da rota.

***Metric***: custo associado à rota.

***Ref***: número de vezes que a rota foi referenciada para estabelecer uma conexão. Este valor não é utilizado por sistemas Linux.

**Use**: número de vezes que a rota foi procurada por IP.

**Interface**: nome da interface de rede utilizada por esta rede.

### Quais as mensagens ARP e endereços MAC observados e porquê?

Após fazer *ping* de um *tux* para o outro, os computadores não conhecem o endereço MAC um do outro. Desta maneira, o *tux* recetor “pergunta” qual o endereço MAC do *tux* emissor que tem aquele endereço IP através do envio de uma mensagem ARP. Deverá ser verificada a Figura 10.

Esta mensagem terá o endereço MAC do *tux* de origem associado e o endereço 00:00:00:00:00:00, indicando que esta foi enviada em modo *Broadcast*, uma vez que ainda não se sabe qual o *tux* de destino. De seguida, o *tux* de destino responde com outra mensagem ARP indicando o seu endereço MAC.

No final do processo, são obtidos tanto o endereço MAC do *tux* de destino como o de origem.

### Que pacotes ICMP são observados e porquê?

São observados pacotes ICMP de request e reply caso a rede tenha sido corretamente estabelecida. Caso contrário, em vez de pacotes ICMP de reply seriam pacotes ICMP de *Destination* Unreachable indicando um problema na rede.

### Quais os endereços IP e MAC associados aos pacotes ICMP e porquê?

Os pacotes ICMP contêm os endereços IP e MAC dos *hosts* de origem e de destino, neste caso os *tux*’s de origem e de destino. Por exemplo, ao efetuar ping do *tux* 3 para o *tux* 4, os endereços de origem serão o IP e MAC do *tux* 3 e os endereços de destino serão o IP e MAC do *tux* 4.

## **Experiência 4 – Configuração de um *router* comercial e implementação de NAT**

### Como configurar uma rota estática num *router* comercial? (incompleta)

Para configurar o *router* comercial é necessário ligar a porta T4 à porta *Router* *Console*. De seguida liga-se a porta T3 à porta S0 do *tux* que se pretende que esteja ligado à consola do *router*, neste caso foi usado o *tux* 3. Após estabelecer esta ligação, basta invocar os seguintes comandos no *GTKTerm* no *tux* conectado à consola do *router*:

> configure terminal

> ip route 0.0.0.0 0.0.0.0 172.16.1.254

> end

### Quais são os caminhos seguidos pelos pacotes nas experiências realizadas e porquê?

Quando um pacote é enviado, se a rota estiver estabelecida, segue essa mesma rota. Caso a rota não esteja estabelecida, o pacote é enviado para o *router*, uma vez que é a rota *default*, e de seguida é reencaminhado para o destino da rota.

### Como configurar NAT num *router* comercial?

Para configurar a NAT num *router* comercial, foi necessário configurar a interface interna no processo de NAT de acordo com o guião fornecido. Foi ainda necessário reconfigurar as rotas IP. Assim sendo, a partir do terminal GTK foram inseridos os seguintes comandos:

> conf t

> interface gigabitethernet 0/0

> ip address 172.16.61.254 255.255.255.0

> no shutdown

> ip nat inside

> exit

> interface gigabitethernet 0/1

> ip address 172.16.1.69 255.255.255.0

> no shutdown

> ip nat outsider

> exit

> ip nat pool ovrld 172.16.1.69 172.16.1.69 prefix 24

> ip nat inside source list 1 pool ovrld overload

> access-list 1 permit 172.16.60.0 0.0.0.7

> access-list 1 permit 172.16.61.0 0.0.0.7

> end

### O que é que faz o NAT?

O NAT (*Network Address Translation*) é uma técnica utilizada para a conservação de endereços IP. Esta técnica permite que redes IP privadas que utilizam endereços IP não registados se possam conectar à Internet. O NAT é aplicado num *router* que conecta, geralmente, duas redes e traduz os endereços privados (não globalmente únicos) na rede interna em endereços legais, antes de os pacotes serem encaminhados para outra rede. O NAT pode ser configurado para anunciar um endereço para toda a rede para a rede externa. Deste modo, fornece segurança adicional ao ocultar toda a rede por de trás desse endereço. O NAT oferece uma função dupla de segurança e de conservação de endereços e é, geralmente, utilizado em ambientes de acesso remoto.

## **Experiência 5 – DNS**

Para realizar esta experiência foi necessário configurar o DNS (*Domain Name System*) nos *tux*’s 2, 3 e 4. Um servidor DNS contém uma base de dados de endereços IP públicos e dos seus respetivos *hostnames*. É utilizado para traduzir os *hostnames* nos respetivos endereços IP. Nesta experiência, foi utilizado o servidor DNS services.netlab.fe.up.pt.

### Como configurar um serviço DNS num *host*?

Para configurar o DNS num *host* é necessário aceder ao ficheiro no diretório /dev/resolv.conf, responsável pela configuração de name servers de DNS. Com o comando *“echo $'search netlab.fe.up.pt\nnameserver 172.16.1.1' > /etc/resolv.conf”*. Deste modo, o nome do servidor DNS será netlab.fe.up.pt com o endereço IP 172.16.1.1. Depois de corretamente configurado, os *tux*’s terão acesso à internet.

### Que pacotes são trocados pelo DNS e que informação transportam?

Primeiramente, o *host* envia um pacote para o servidor que contém o *host*name desejado, como por exemplo google.com, pedindo o seu endereço IP. De seguida, o servidor responde enviando um pacote contendo o IP do *host*name.

## Experiência 6 – Conexões TCP

### Quantas conexões TCP são abertas pela aplicação FTP desenvolvida?

A aplicação FTP desenvolvida abre duas conexões TCP, uma para controlo, na qual são enviados os comandos FTP ao servidor e recebidas as respostas correspondentes, e outra para receber os dados enviados pelo servidor e enviar as respostas ao cliente.

### Em que conexão é transportada a informação de controlo FTP?

A informação de controlo FTP é transportada na conexão FTP responsável pela troca de comandos.

### Quais são as fases de uma conexão FTP?

Uma conexão FTP é constituída por três fases: o estabelecimento da ligação, a troca de dados e a terminação da ligação.

### Como funciona o mecanismo ARQ FTP? Quais são os parâmetros TCP relevantes? Que informação importante pode ser observada nos *logs*?

O TCP (*Transmission Control Protocol*) utiliza o mecanismo ARQ (*Automatic Repeat Request*) com o método *Go Back N ARQ* (*Sliding Window*). Este método consiste no controlo de erros na transmissão de dados ao utilizar mensagens *acknowledgments* e *timeouts* de forma a conseguir uma transmissão segura de dados através de um serviço não seguro. De forma a obter este objetivo, são utilizados *acknowledgment numbers*, localizados no campo de mensagens enviadas pelo recetor que indicam se a trama foi recebida corretamente, *window size*, que indica o número máximo de tramas de dados que o emissor pode ter pendentes, e o *sequence number*, que indica o número da trama enviada.

### Como funciona o mecanismo de controlo de congestionamento TCP? Quais são os parâmetros relevantes? Como evoluiu o rendimento da ligação de dados ao longo do tempo? Está de acordo com o mecanismo de controlo de congestionamento TCP? (incompleto)

O mecanismo de controlo de congestionamento TCP utiliza uma janela de congestão que é um dos fatores que determina o número de bytes que podem ser enviados em qualquer momento. A janela de congestionamento é mantida pelo emissor, sendo um meio de impedir que a ligação entre o emissor e recetor fique sobrecarregada com muito tráfego. Outro fator que determina o número de bytes que podem ser enviados é a janela de deslizamento (sliding window) do recetor, referido anteriormente.

### A conexão de dados TCP é afetada pelo aparecimento de uma segunda conexão TCP? De que maneira é afetada?

Com o aparecimento de uma segunda conexão FTP, ao existir uma transferência de dados em simultâneo, pode levar a uma diminuição da taxa de transferência, uma vez que a taxa de transferência é distribuída de igual forma para cada ligação.

# Conclusões

Este trabalho teve como objetivo o desenvolvimento de uma aplicação *download* e a configuração de uma rede de computadores. Esta aplicação *download* foi, posteriormente, testada nesta rede configurada no laboratório, tendo superado com sucesso todos os testes a que esteve sujeita.

Em suma, as experiências na rede estabelecida foram bem-sucedidas assim como o programa desenvolvido para a transferência de dados. Ao longo da realização do projeto, foram adquiridos importantes conhecimentos teórico-práticos em relação ao tema abordado, aprofundando a aprendizagem relativos a redes de computadores, algo presente no nosso quotidiano de forma relevante.

# Anexo

## Imagens

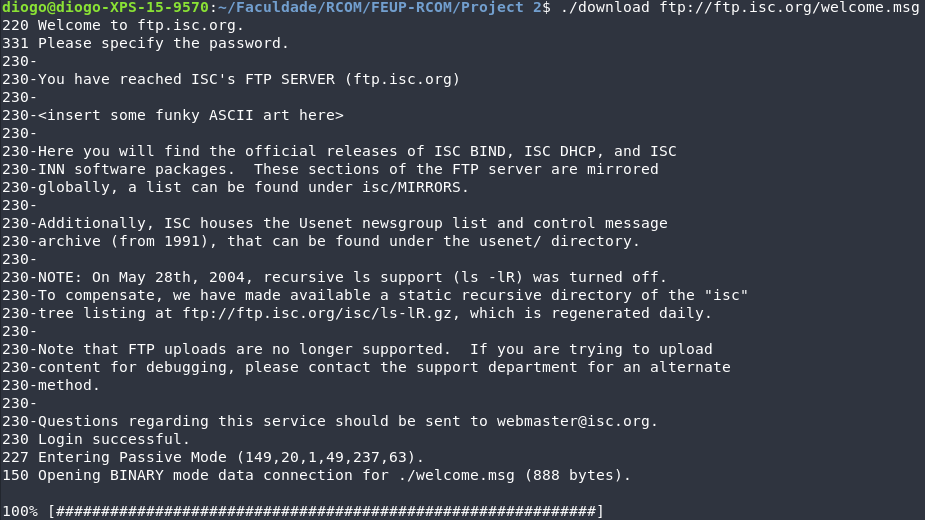


Figura 1 – Aplicação Download

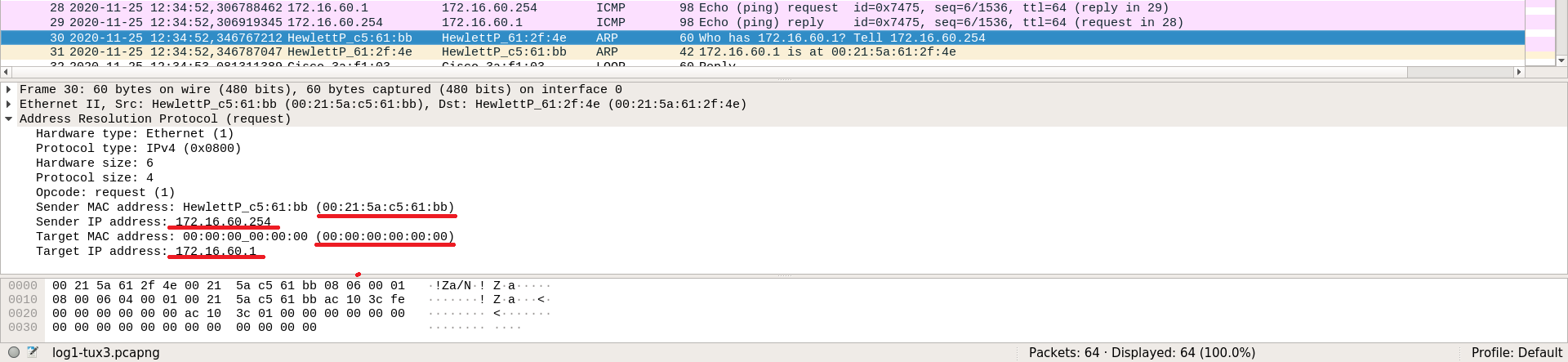


Figura 2 – Pacote ARP 1

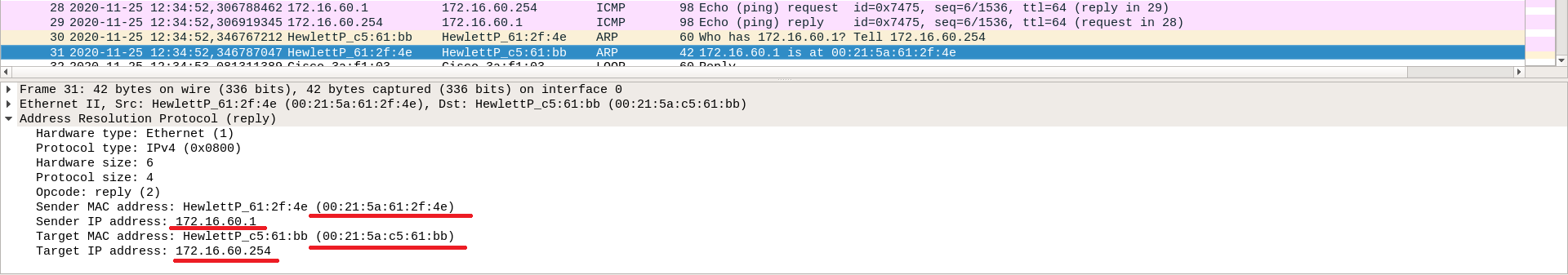


Figura 3 – Pacote ARP 2

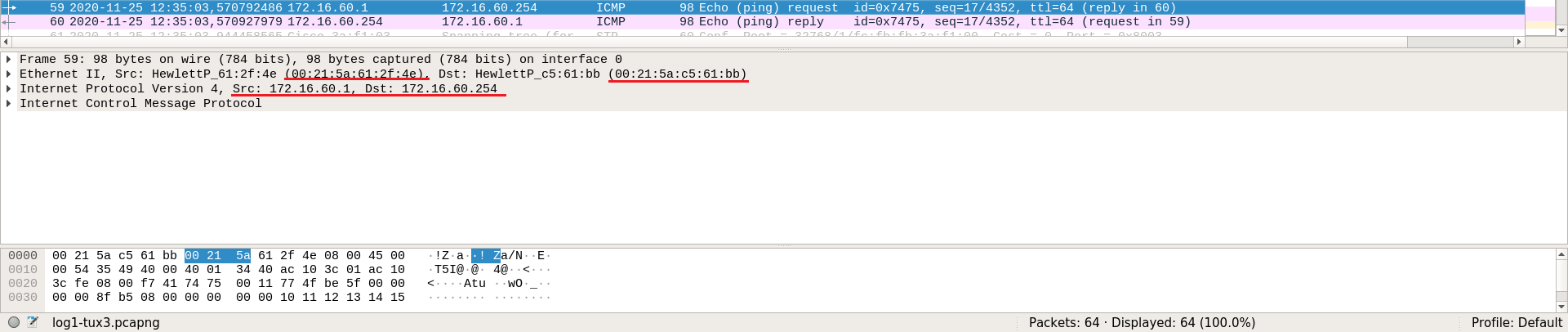


Figura 4 – Pacote Ping Emissor

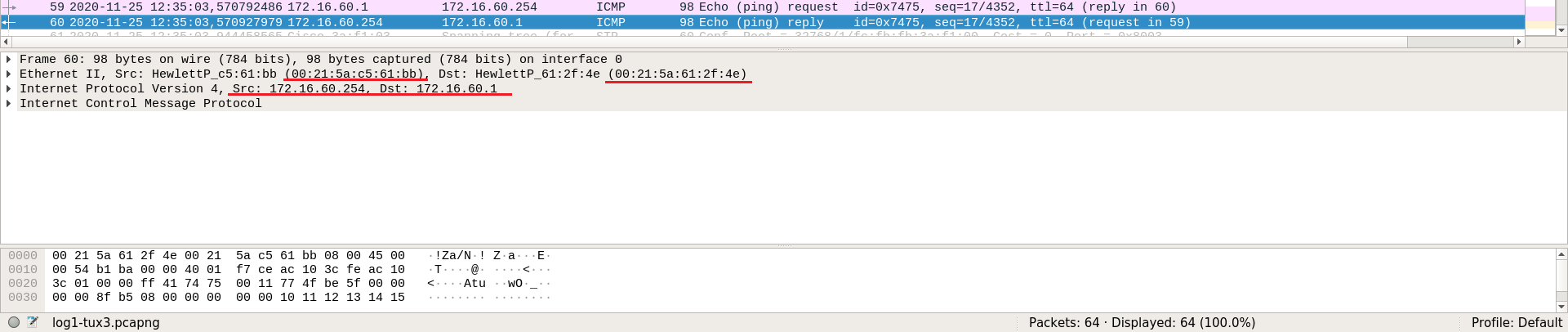


Figura 5 – Pacote Ping Recetor

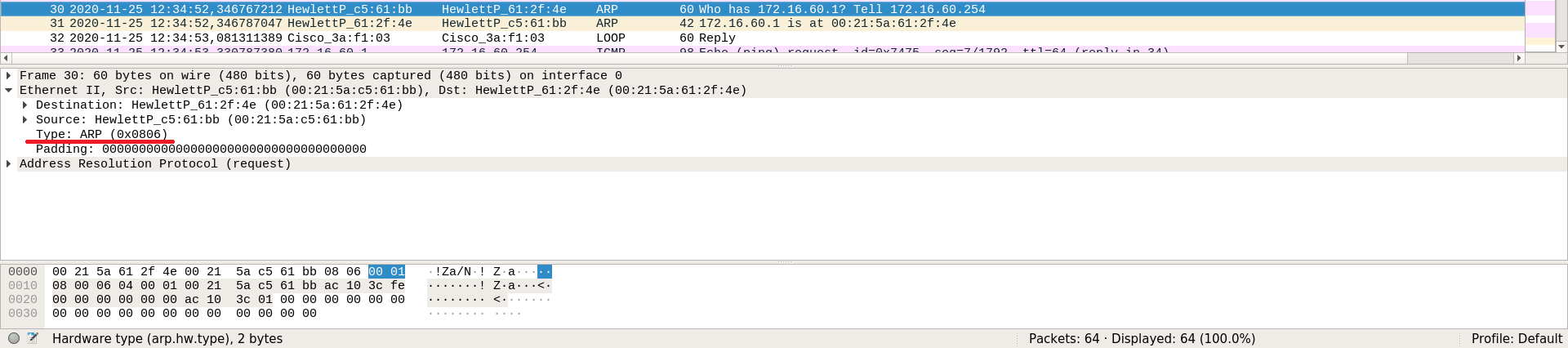


Figura 6 – Tipo da Trama



Figura 7 – Tipo da Trama

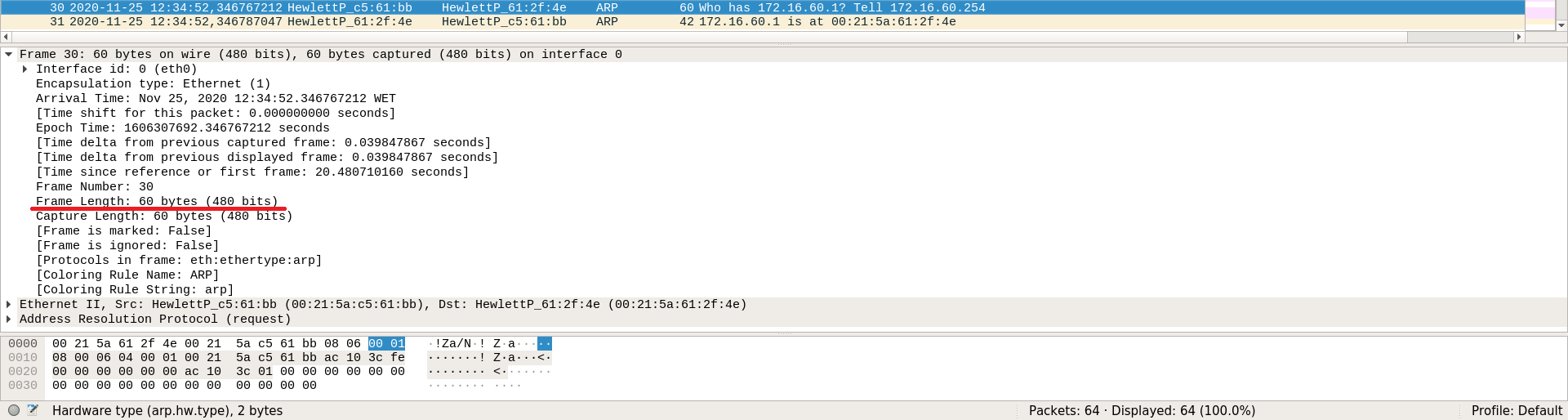


Figura 8 – Tamanho da Trama

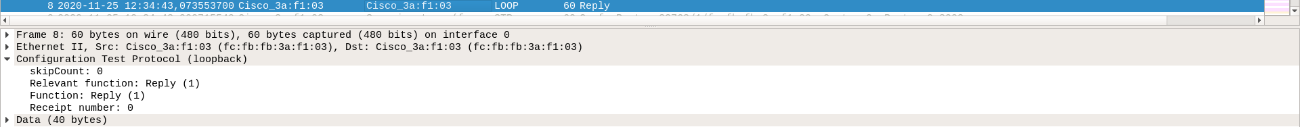


Figura 9 – Interface Loopback

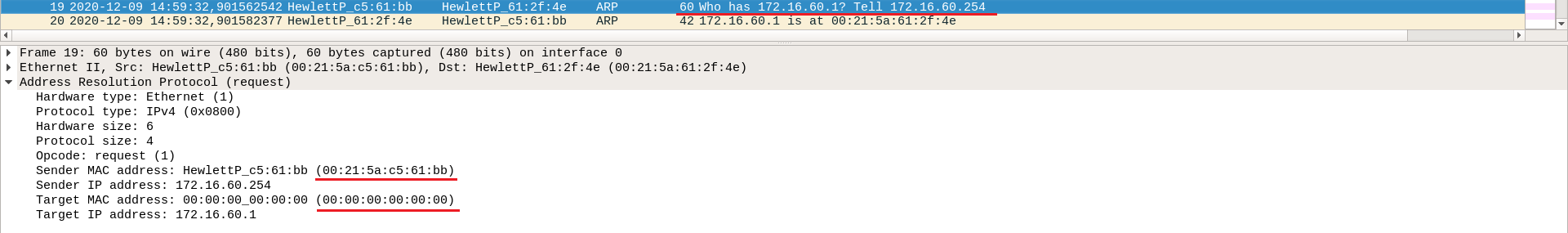


Figura 10 – Mensagem ARP 1

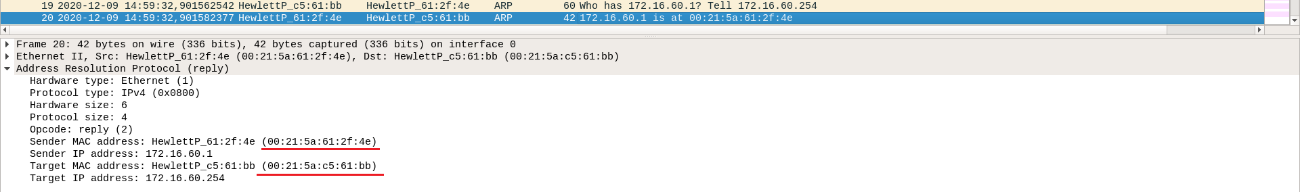


Figura 11 – Mensagem ARP 2