A black background with a black square

Description automatically generated with medium confidence

Redes de Computadores

*2º Trabalho Laboratorial*

*Licenciatura em Engenharia Informática e Computação*

*2023*

João Alves **up202108670**@fe.up.pt

Eduardo Sousa **up202103342**@fe.up.pt

**Índice**

[**Sumário** 3](#_Toc152622312)

[**Introdução** 3](#_Toc152622313)

[**Aplicação de Download** 3](#_Toc152622314)

[**Arquitetura** 4](#_Toc152622315)

[**Testes** 4](#_Toc152622316)

[**Configuração de Redes e Análise** 4](#_Toc152622317)

[**Experiência 1 – Configuração de IP de rede** 4](#_Toc152622318)

[**Experiência 2 – Implementar *bridges* num switch** 5](#_Toc152622319)

[**Experiência 3 – Configurar um router em Linux** 5](#_Toc152622320)

[**Experiência 4 – Configurar router comercial e implementar o NAT** 5](#_Toc152622321)

[**Experiência 5 – DNS** 5](#_Toc152622322)

[**Experiência 6 – Conexão TCP** 6](#_Toc152622323)

[**Conclusão** 6](#_Toc152622324)

[**Referências** 6](#_Toc152622325)

[**Anexos** 6](#_Toc152622326)

# **Sumário**

Este projeto, desenvolvido no contexto da disciplina de Redes de Computadores, tem como objetivo a implementação de um programa de download utilizando o protocolo FTP, juntamente com a configuração e operação de uma rede de computadores. Ao aplicar os conceitos aprendidos durante as aulas teóricas, conseguimos criar o programa mencionado e realizar a configuração da rede.

# **Introdução**

# O propósito deste projeto consistiu no desenvolvimento e teste de um programa de download, utilizando o protocolo FTP, além da configuração de uma rede de computadores conforme as diretrizes do guião. O relatório subsequente está estruturado em seções que abrangem diversos aspetos do projeto, como demonstrado um pouco acima pelo índice.

# **Aplicação de Download**

## **Arquitetura**

## **Testes**

# **Configuração de Redes e Análise**

## **Experiência 1 – Configuração de IP de rede**

**1) What are the commands required to configure this experience?**

Para a nossa experiência foram necessários os seguintes comandos (excluindo o subentendido system config reset):

$ ifconfig eth0 up

$ ifconfig eth0 172.16.Y0.1/24, para a máquina 3

$ ifconfig eth0 172.16.Y0.254/24, para a máquina 4

$ ping 172.16.Y0.1 -c 5, para a máquina 4

$ ping 172.16.Y0.254 -c 5, para a máquina 3

$ arp -a

$ arp -d 172.16.Y0.254/24

$ arp -a

$ ping 172.16.Y0.254 -c 5, para a máquina 3 com wireshark a captar

**2) What are the ARP packets and what are they used for?**

ARP (Address Resolution Protocol) é um protocolo ou procedimento que tem como objetivo estabelecer e efetuar a ligação entre um endereço IP e um endereço MAC.

**3) What are the MAC and IP addresses of ARP packets and why?**

Inicialmente são enviados 2 endereços de IP em Broadcast, que correspondem ao IP da máquina de destino e ao IP da máquina de origem, tendo como IPs, respetivamente, 172.16.50.254 (tux54) e 172.16.50.1 (tux53), para a experiência efetuada na bancada 5. Estes IPs são enviados no mesmo pacote do tipo MAC. Depois como resposta a máquina tux54 envia através de um pacote do tipo ARP o seu endereço MAC, que no caso é: 00:21:5a:c3:78:70.

**4) What packets does the ping command generate?**

Enquanto não for obtido o endereço MAC da máquina de destino, o comando ping gera pacotes do tipo ARP. Depois de obter o endereço MAC da máquina de destino, o comando ping irá passar a gerar pacotes do tipo ICMP (Internet Control Message Protocol).

**5) What are the MAC and IP addresses of the ping packets?**

Os endereços IP e MAC usados em pacotes são os das máquinas em comunicação, ou seja, das máquinas tuxY3 e tuxY4.

**6) How to determine if a receiving Ethernet frame is ARP, IP, ICMP?**

Para determinar qual o tipo de protocolo que foi recebido basta olhar para a coluna “Protocol” do Wireshark e visualizar, ou ao inspecionar o header de ethernet de um pacote (primeiros 2 bytes), conseguimos determinar qual foi o protocolo usado pelo pacote.

**7) How to determine the length of a receiving frame?**

O Wireshark apresenta uma coluna que nos indica o tamanho do frame recebido, em bytes.

**8) What is the loopback interface and why is it important?**

Uma interface loopback é uma interface virtual que permite à máquina receber respostas dela mesma. Esta interface é usada para verificar periodicamente se as ligações da rede estão configuradas como pretendemos.

## **Experiência 2 – Implementar *bridges* num switch**

**1) How to configure bridgeY0?**

Para configurar uma bridge, primeiro temos de a criar, de modo a pudermos então depois configurá-la de modo a criar uma subrede que irá conter o tuxY3 e tuxY4. Após a sua criação eliminamos as configurações default atribuídas pelo switch aquando da criação da bridge, de modo a pudermos configurá-la como é suposto. Os comandos para fazer a configuração são os seguintes (M e N representam o número da porta do switch aonde os tuxs estão ligados):

> /interface bridge add name=bridgeY0

> /interface bridge port remove [find interface=etherM]

> /interface bridge port remove [find interface=etherN]

> /interface bridge port add bridge=bridgeY0 interface=etherM

> /interface bridge port add bridge=bridgeY0 interface=etherN

**2) How many broadcast domains are there? How can you conclude it from the logs?**

No final das

Existem dois domínios de broadcast pois existem duas bridges implementadas. Analisando os logs, reparamos que o ping do Tux53 conseguiu resposta do Tux54 mas não do Tux52, pois estão inseridos em duas bridges diferentes.

## **Experiência 3 – Configurar um router em Linux**

**1) What are the commands required to configure this experience?**

Para além dos comandos usados para configurar as experiências anteriores, são usados os seguintes comandos para configurar esta experiência:

$ ifconfig eth1 up

$ ifconfig eth1 172.16.51.253/24

> /interface bridge port remove [find interface=ether4]

> /interface bridge port add bridge=bridge51 interface=ether4

$ sysctlnet.ipv4.ip\_forward=1

$ sysctlnet.ipv4.icmp\_echo\_ignore\_broadcasts=0

$ route add -net 172.16.50.0/24 gw 172.16.Y1.253

$ route add -net 172.16.51.0/24 gw 172.16.Y0.254

$ arp -d 172.16.Y0.1

$ arp -d 172.16.Y1.1

$ arp -d 172.16.Y1.253

$ arp -d 172.16.Y0.254

**2) What routes are there in the tuxes? What is their meaning?**

**3) What information does an entry of the forwarding table contain?**

**4) What ARP messages, and associated MAC addresses, are observed and why?**

**5) What ICMP packets are observed and why?**

**6) What are the IP and MAC addresses associated to ICMP packets and why?**

## **Experiência 4 – Configurar router comercial e implementar o NAT**

**1) How to configure a static route in a commercial router?**

**2) What are the paths followed by the packets in the experiments carried out and why?**

**3) How to configure NAT in a commercial router?**

**4) What does NAT do?**

## **Experiência 5 – DNS**

**1) How to configure the DNS service in a host?**

**2) What packets are exchanged by DNS and what information is transported**

## **Experiência 6 – Conexão TCP**

**1) How many TCP connections are opened by your FTP application?**

**2) In what connection is transported the FTP control information?**

**3) What are the phases of a TCP connection?**

**4) How does the ARQ TCP mechanism work? What are the relevant TCP fields? What relevant information can be observed in the logs?**

**5) How does the TCP congestion control mechanism work? What are the relevant fields. How did the throughput of the data connection evolve over time? Is it according to the TCP congestion control mechanism?**

**6) Is the throughput of a TCP data connection disturbed by the appearance of a second TCP connection? How?**

# **Conclusão**

# **Referências**

# **Anexos**