Uma imagem com Tipo de letra, Gráficos, captura de ecrã, preto

Descrição gerada automaticamente

Faculdade de Engenharia da Universidade do Porto

2º Project

Instructor:

Eduardo Nuno Moreira Soares de Almeida

Redes de Computadores

Turma 15 – Grupo 2

- Clarisse Maria Teixeira de Carvalho ([up202008444@fc.up.pt](mailto:up202008444@fc.up.pt))

- Maria Eduarda Pacheco Mendes Araújo ([up202004473@fc.up.pt](mailto:up202004473@fc.up.pt))

Porto, 23 de dezembro de 2024

**Index:**

Index

Summary

Introduction

Download Application

Application Architeture

Configuration and Network Analysis

Experiment 1 – Configure an IP Network

Experiment 2 – Implement two bridges in a switch

Experiment 3 – Configure a router in Linux

Experiment 4 – Configure a Commercial Router and Implement NAT

Experiment 5 – DNS

Experiment 6 – TCP connections

Conclusions

Annexes

1 – Código da aplicação de download

2- Comandos de configuração

3 – Logs Capturados

**Summary:**

This project was carried out as part of the Computer Networks course and aimed to implement a download program using FTP and to configure and use a computer network.

**Introduction:**

The goal of the project was to develop and test a download program using FTP and set up a computer network, according to the specifications given to us by the script, to download a file from the internet using the network we had set up.

**Download Application:**

**Application Architecture:**

The application developed downloads a file via the FTP protocol.

Initially, the URL given as an argument is processed using regular expressions in order to obtain a data structure with the information needed to make the connection:

* host: name of the server where the communication will be created
* resource: path to the file
* file: name and extension of the file to be transferred
* user and password: to register on the server
* ip: obtained from the host

**Configuration and Network Analysis:**

**Experiment 1: Configure an IP Network (Bancada 6)**

In this experiment we want to configure two IP addresses for two computers, tuxY3 and tuxY4, connected to a switch

We start by restarting both GTK and the terminal using the following commands:

GTK: /system reset-configuration

Terminal: systemctl restart networking

Next, we configure the IPs:

TuxY3 (terminal) : ifconfig eth1 172.16.60.1/24

TuxY4 (terminal) : ifconfig eth1 172.16.60.254/24

TuxY4 (terminal) : ifconfig eth2 172.16.61.253/24

**Questions:**

1. **What are the ARP packets and what are they used for?**

ARP (Address Resolution Protocol) packets are network-layer messages used to map IP addresses to MAC (Media Access Control) addresses within a local network.

1. **What are the MAC and IP addresses of ARP packets and why?**

The ARP protocol uses a specific addressing structure where:

The ARP Request packet contains the MAC and IP of the sending device, a broadcast MAC as the destination and the IP being sought.

In the ARP Reply, the device that owns the IP being sought replies with its own MAC and IP as the source, and places the MAC and IP of the requester as the destination.

This structure allows ARP to fulfill its main function: to find out which MAC corresponds to a given IP on the local network.

1. **What packets does the ping command generate?**

The ping command generates ICMP (Internet Control Message Protocol) packets. Specifically, it uses two types of ICMP messages to test network connectivity between two devices.

1. **What are the MAC and IP addresses of the ping packets?**

Ping (ICMP) packets work as follows:

Source:

- MAC address of the sending device

- IP address of the sending device

Destination:

- MAC address of the target device

- IP address of the target device

In other words, each ping packet contains the real MAC and IP addresses of both the source and the destination.

1. **How to determine if a receiving Ethernet frame is ARP, IP, ICMP?**

To determine the type of protocol encapsulated in an Ethernet frame, we need to examine the EtherType field, which is a 16-bit field that indicates the type of payload carried in the Ethernet frame.

1. **How to determine the length of a receiving frame?**

We can see the Frame length field in the Ethernet frame header, which indicates the total length of the Ethernet frame, including the frame header and the frame payload.

1. **What is the loopback interface and why is it important?**

This is a fundamental component in networks for local testing, debugging and ensuring consistent behavior in network applications, providing a controlled environment for communication within the same device.

**Experiment 2: Implement two bridges in a switch (Bancada 6)**

In this experiment, we configured bridges for tuxY3 and tuxY4 and another for tuxY2 only, using two bridges on the switch.

First we configured the tuxY2 network:

TuxY2 (terminal) : ifconfig eth1 172.16.61.1/24

Next, we create the necessary bridges:

/interface bridge add name=bridge60

/interface bridge add name=bridge61

We delete the ports from the GTK's default bridge and add the ports to the correct bridges:

/interface bridge port remove [find interface =ether1]

/interface bridge port remove [find interface =ether3]

/interface bridge port remove [find interface =ether5]

/interface bridge port add interface=ether1 bridge=bridge60

  /interface bridge port add interface=ether3 bridge=bridge60

  /interface bridge port add interface=ether5 bridge=bridge61

**Questions:**

1. **How to configure bridgeY0?**

To configure the bridgeY0 we need to:

- Create the bridge: */interface bridge add name=bridgeY0*

- Removal of the port from default: */interface bridge port remove [find interface=etherX]*

- Add port to bridgeY0: */interface bridge port add interface=etherX bridge=bridgeY0*

1. **How many Broadcast domains are there? How can you conclude it from the logs?**

There ate two Broadcast domains: 172.16.60.255 and 172.16.61.255.

Running the command *ping -b 172.16.50.255* shows that the tuxes that received requests were tuxY3 and tuxY4, both connected to bridge60, confirming the validity of the broadcast domain associated with this bridge.

When we used the command *ping -b 172.16.61.255*, we found that the requests were only received by tuxY2, which was connected to brige61.

This means that there are two distinct and valid broadcast domains.

**Experiment 3: Configure a Router in Linux (Bancada 6)**

First, we activate IP\_Forwarding and deactivate ICMP echo-ignore-broadcast in the tux4 terminal:

sysctl net.ipv4.ip\_forward=1

sysctl net.ipv4.icmp\_echo\_ignore\_broadcasts=0

In the tux3 terminal, we define the following route:

route add -net 172.16.Y1.0/24 gw 172.16.Y0.254

In the tux2 terminal, we define the following route:

route add -net 172.16.Y0.0/24 gw 172.16.Y1.253

**Questions:**

1. **What routes are there in the tuxes? What are their meaning?**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **dest** | **gateway** | **interface** |
| tux62 | 172.16.60.0 | 172.16.61.253 | eth1 |
|  | 172.16.61.0 | 0.0.0.0 | eth1 |
| tux63 | 172.16.60.0 | 0.0.0.0 | eth1 |
|  | 172.16.61.0 | 172.16.60.254 | eth1 |
| tux64 | 172.16.60.0 | 0.0.0.0 | eth1 |
|  | 172.16.61.0 | 0.0.0.0 | eth2 |

1. **What information does an entry of the forwarding table contain?**

Contains information on how to forward packets to a particular destination. In the case shown in the previous answer, it contains the destination address, the gateway and the interface used.

1. **What ARP messages, and associated MAC addresses, are observed and why?**
2. **What ICMP packets are observed and why?**
3. **What are the IP and MAC addresses associated to ICMP packets and why?**

**Experiment 4: Configure a Commercial Router and Implement NAT (Bancada 4)**

This experiment began with the network that resulted from the previous experiment, which aimed to turn tux64 into a router for tux63 and tux62.

First we switched the cable connected to the switch console to the MT Router port (or MTIK) and reset it.

GTK: /system reset-configuration

Next, we had to configure the router's IPs and add the following routes to tuxY3, tuxY4 and tuY2:

GTK:

/ip address add address=172.16.1.49/24 interface=ether1

/ip address add address=172.16.41.254/24 interface=ether2

- tuxY3:

route add -net 172.16.41.0/24 gw 172.16.40.254

route add -net 172.16.1.0/24 gw 172.16.40.254

- tuxY4:

route add -net 172.16.1.0/24 gw 172.16.41.254

- tuxY2:

route add -net 172.16.40.0/24 gw 172.16.41.253

route add -net 172.16.1.0/24 gw 172.16.41.254

We set up a static route on the GTK device:

 /ip route add dst-address=172.16.40.0/24 gateway=172.16.41.253

**Questions:**

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

First we have to reset the router (*/system reset-configuration*). Then we add the corresponding bridges (*/ip address add address=172.16.1.49/24 interface=ether1*) and match internal and external IPs.

1. **What are the paths followed by the packets, with and without ICMP redirect enabled, in the experiments carried out and why?**

With tux62 and tux64 switched off: the packets used ICMP redirection via the router

However, here there was no such redirection because the shortest path was available for use

1. **How to configure NAT in a commercial router?**

To disable default NAT: */ip firewall nat desable 0*

To add NAT rules: */ip firewall nat add chain=srcnat action=masquerade out-interface=ether1*

1. **What does NAT do?**

NAT (Network Address Translation) translates private IP addresses to public IP addresses, enabling multiple devices to access external networks while conserving public IPs and providing a layer of security.

1. **What happens when tuxY3 pings the FTP server with the NAT disabled? Why?**

NAT disabled: when tux63 tries to ping the FTP server, it fails. This is because the router can't translate a private IP into a public one, so tux63's packets can't be routed on a public network.

That said, any possible response from the FTP server won't be reached by tux63

**Experiment 5: DNS (Bancada 4)**

Here we configure the DNS so that we can access websites on the Internet, within the network created, using your domain name. This must be done on tuxY3, tuxY4 and tuxY2.

We use **10.227.20.3** because it's the DNS of the *netlab.fe.up.pt* server.

nano /etc/resolv.conf

nameserver 10.227.20.3

**Questions:**

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

To configure the DNS service we have to do: *nano /etc/resolv.conf*

Next, we add the IP address of the DNS server: *nameserver 10.227.20.3*

1. **What packets are exchanged by DNS and what information is transported?**

For the router to be able to identify and resolve the destination IPs, DNS packets must be exchanged at an early stage, translating the domain names into the corresponding IPs

**Experiment 6: TCP connections (Bancada 4)**

In this experiment we used the network we configured throughout the experiments and we also used the download application we made.

**Questions:**

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 relevante 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 envolve along the time? Is it according to the TCP congestion control mechanism?**
6. **Is the troughput of a TCP data connections disturbed by the appearance of the second TCP connection? Why?**