

Laboratório 2

MIEIC – Mestrado Integrado em Engenharia Informática e Computação

RCOM – Redes de Computadores

Grupo:

João Paulo Moreira Barbosa – up201406241

Miguel Lira Barbeitos Luís – up201405324

Miriam Cristiana Meireles Campos Gonçalves – up201403441

Ano letivo 2016/2017

Sumário

No âmbito da unidade curricular de Redes de Computadores, do primeiro semestre do terceiro ano, foi proposto o desenvolvimento de uma aplicação FTP(*File Transfer Protocol*) e a configuração e estudo de uma rede de um computador, através de várias experiências laboratoriais.

No total, realizou-se seis experiências laboratoriais ao longo das aulas práticas. Começou-se por configurar uma rede IP. De seguida, implementou-se duas LANs (*Local Area Network*) virtuais num *switch*, configurou-se um router em Linux e, por fim, configurou-se um router comercial com implementação NAT (*Network Address Translation*). Após a realização das experiências anteriormente referidas, desenvolveu-se uma aplicação de download FTP, usando as configurações das experiências.

Índice

1. Introdução	2
2. Aplicação download FTP	2
3. Experiências Laboratoriais	3
3.1. Configuração de uma rede IP	3
3.2. Implementar duas LANs virtuais num switch	4
3.3. Configurar um router em Linux	4
3.4. Configurar um router comercial e implementar NAT	5
3.5. DNS	6
3.6. Conexões TCP	7
4. Conclusão	8
5. Anexos	9

1. Introdução

No segundo e último trabalho da unidade curricular de Redes de Computadores (RCOM), foi proposto pelo docente a criação de uma aplicação de download utilizando o protocolo FTP e um conjunto de experiências laboratoriais, a serem realizadas durante as aulas práticas.

A aplicação tinha como principais objetivos a implementação de um cliente simples FTP na linguagem de programação C, o uso de *sockets* e TCP, e, por último, aprender o serviço providenciado pelo DNS (*Domain Name System*). Este cliente FTP é de fácil uso e funciona corretamente desde que o URL passado como argumento esteja correto:

ftp://[<user>:<password>@]<host>/<url-path>.

As experiências laboratoriais tinham como principais objetivos a configuração e o estudo de uma rede, abordando os conteúdos lecionados da unidade curricular e, permitindo, uma melhor compreensão destes. No final das experiências, era possível aos alunos desenvolver e testar a aplicação de forma bem-sucedida.

Este relatório visa explicar a maneira como se realizou a aplicação e responder às perguntas propostas durante o guião de trabalho das diferentes experiências, assim como, explicá-las melhor.

2. Aplicação Download FTP

Após a realização das experiências laboratoriais, desenvolveu-se uma aplicação que tinha como objetivo ligar-se a um servidor e fazer download de um ficheiro escolhido pelo utilizador através do uso do protocolo FTP (File Transfer Protocol). Este protocolo permite transferir ficheiros de uma forma bastante rápida e simples.

Antes de se começar o processo de ligação ao servidor até ao download do ficheiro, é necessário processar-se o argumento dado pelo utilizador, ou seja, o URL passado que contém o nome do utilizador, a sua password, o host e o caminho url do ficheiro que se quer transferir. Para se decompor esta informação utiliza-se a função auxiliar `parseURL`, esta função receberá o argumento que terá como estrutura padrão a seguinte: **ftp://[<user>:<password>@]<host>/<url-path>**, caso o utilizador queira entrar em modo anónimo então deve passar o argumento com a seguinte estrutura: **ftp://<host>/<url-path>**, neste último caso, o código assumirá que o utilizador terá como conteúdo a string "anonymous" e a password o conteúdo "mail@domain.com".

Inicia-se o processo pela chamada à função auxiliar `initTCP` que realizará a conexão TCP para a porta 21 do servidor. Após a conexão ser feita, verifica-se a resposta dada através de três dígitos de código de estado em ASCII, caso esta resposta comece com o dígito 5 verifica-se que o comando enviado ou a ação pedida não foi aceite.

Após a conexão feita, procede-se ao login, escrevendo-se para o socket dado aquando da conexão os devidos valores do utilizador com os seus respetivos comandos.

Caso o login seja bem-sucedido, envia-se o comando para que o corra em modo passivo e o servidor responderá com a porta e o endereço que deverão ser utilizados para tratar o resto dos dados. É criada uma nova conexão com o endereço e a porta dados como resposta anteriormente e é nesta porta que é enviado o comando `RETR` juntamente com o caminho url dado pelo utilizador, este comando irá pedir ao servidor para enviar o conteúdo do ficheiro através da conexão já estabelecida

pelo utilizador. No final de todos estes comandos e passos anteriores serem bem-sucedidos, é criado o novo ficheiro com o mesmo nome e conteúdo do servidor.

É de realçar que a aplicação está preparada para mostrar ao utilizador, caso ocorra algum erro, a fase em que este ocorreu. Após várias análises de código e testes feitos à aplicação acredita-se que o resultado final é robusto, simples e de fácil compreensão.

3. Experiências Laboratoriais

3.1. Configuração de uma rede IP

Esta experiência liga dois computadores, via *switch*, e tem como objetivo identificar e distinguir os diferentes pacotes enviados, assim como a sua finalidade.

Um dos tipos de pacotes de dados são os ARP. Este protocolo destina-se a mapear um IP ao endereço físico (MAC), isto é, encontrar a máquina a qual o IP se encontra associado. Neste processo, há dois pacotes que são transmitidos.

O primeiro pacote é enviado para todos os computadores, contendo o IP para o qual é pretendido saber o MAC. Para obter esta informação, o pacote é enviado em *broadcast*, a partir da máquina que deseja receber os dados.

No segundo pacote, a máquina cujo endereço MAC é o endereço alvo do primeiro pacote, vai enviar o seu endereço MAC à máquina que pretendia esta informação.

Outros pacotes enviados são os pacotes ICMP, gerados pelo comando *ping*. Também são agrupados em dois pacotes, de pergunta e resposta, tendo como IPs (e MACs) as máquinas de fonte e de destino da comunicação. Na experiência efetuada, o IP fonte é 172.16.10.1 e o IP destino é 172.16.10.254, tendo estes os endereços MACs 00:0f:fe:8b:e4:ef e 00:22:64:a6:a4:f8, respetivamente.

A distinção entre os pacotes é feita no cabeçalho das tramas enviadas - cada protocolo tem o seu identificador e as suas particularidades quanto à informação que é enviada nos pacotes.

Por exemplo, para distinguir os pacotes ARP e dos pacotes ICMP, deve-se olhar para o identificador do tipo de trama, que está guardado após os MACs das duas máquinas. No caso do ARP, o identificador é 0x0806, e no caso do ICMP, o identificador é 0x0800.

```
00 0f fe 8b e4 ef 00 22 64 a6 a4 f8 08 06 00 01
08 00 06 04 00 02 00 22 64 a6 a4 f8 ac 10 0a fe
00 0f fe 8b e4 ef ac 10 0a 01 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Figura 5: Trama ARP

```
00 22 64 a6 a4 f8 00 0f fe 8b e4 ef 08 00 45 00
00 54 3e 81 40 00 40 01 8f 08 ac 10 0a 01 ac 10
0a fe 08 00 4c d0 09 89 00 01 d0 45 4d 58 8c 04
0d 00 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15
16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25
26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35
36 37
```

Figura 6: Trama ICMP

Por vezes, no envio das tramas é ativado o mecanismo de *loopback*. Esta interface tem um canal de comunicação com apenas um ponto final, sendo, neste caso, o computador que enviou a trama. Este mecanismo permite detetar erros na transmissão das tramas, e também testar a qualidade dos elementos intermédios (como cabos e modems).

3.2. Implementar duas LANs virtuais num switch

Nesta experiência foi possível configurar o switch, e criar duas LANs virtuais.

Para configurar uma *vlan* é necessário inserir os seguintes comandos na consola do switch:

1. configure terminal
2. vlan x (x = número da VLAN)
3. interface fastethernet 0/y (y = número da porta no switch)
4. switchport mode access
5. switchport access vlan x
6. end

O primeiro passo serve para abrir o modo de configuração, enquanto que o segundo cria uma VLAN. Os passos seguintes servem para adicionar portas à VLAN. Note-se que os passos 3 a 5 devem ser repetidos, por cada porta do switch que queremos que seja adicionada.

Existe um canal de broadcast para cada subrede, i.e., para cada VLAN. Analisando os logs, é possível verificar que só IPs dentro da mesma VLAN respondem aos broadcasts (por exemplo, fazendo ping em broadcast a partir do tux1, só o tux4 respondeu).

3.3. Configurar um router em Linux

Esta experiência permite configurar uma máquina para trabalhar como um router, assim como definir corretamente as rotas da rede.

Utilizando o comando "route -n", é possível verificar as rotas de cada máquina.

No tux1 as rotas são:

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
172.16.10.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
172.16.11.0	172.16.10.254	255.255.255.0	UG	0	0	0	eth0

A primeira rota é a rota automaticamente criada quando é configurado o IP da interface eth0 da máquina, e permite comunicar com as máquinas dentro da mesma subrede 172.16.10.0/24.

A segunda rota permite comunicar com as máquinas da subrede 172.16.11.0/24, utilizando a gateway 172.16.10.254 - ou seja, as tramas de informação que se pretende enviar para a outra subrede, devem passar por este IP.

As rotas no tux2 são:

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
172.16.10.0	172.16.11.253	255.255.255.0	UG	0	0	0	eth0
172.16.11.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0

Note-se que as rotas são, de certa forma, o espelho das rotas do tux1: é permitido ao tux2 comunicar com as máquinas dentro da subrede 172.16.11.0/24, e é possível comunicar com a subrede 172.16.10.0/24 utilizando a gateway 172.16.10.253.

Finalmente, no tux4 temos as seguintes rotas:

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
172.16.10.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
172.16.11.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1

Esta máquina pode comunicar com as duas sub-redes, utilizando as interfaces eth0 e eth1, funcionando, portanto, como a ponte de comunicação entre as sub-redes - ou seja, tem a funcionalidade de um router.

A tabela de forwarding (visíveis com o comando "route - n"), contém informação acerca da rede de destino, a gateway por onde é feito o acesso, a máscara da rede, e a interface associada. São ainda dadas informações acerca das flags, da distância ao destino (Metric), número de referências à rota (Ref), e também a contagem de lookups pela rota (Use).

Nos passos 10 e 11, é possível verificar a comunicação entre os computadores tux1 e tux2, servindo-se do tux4 como intermediário. Primeiramente, o tux1 vai obter o endereço MAC da gateway 172.16.10.254. O tux4, por sua vez, vai obter o endereço MAC do tux2 (IP 172.16.11.1).

A utilização deste protocolo também ocorre em sentido inverso, ou seja, o tux2 obtém o endereço MAC da gateway, e o tux4 obtém o endereço MAC do tux1.

Os pacotes ICMP têm sempre os mesmos endereços IP e MAC (fonte e destino), pois já têm a informação necessária para a transferência de dados.

3.4. Configurar um router comercial e implementar NAT

Esta experiência consiste em configurar um router comercial, implementando e percebendo melhor esta funcionalidade.

Para adicionar uma rota estática, é necessário, na consola do router, utilizar o comando "**ip route {rede de destino} {máscara de rede} {gateway}**". No caso da experiência, foram utilizadas duas rotas estáticas:

```
> ip route 0.0.0.0 0.0.0.0 172.16.1.254
```

```
> ip route 172.16.10.0 255.255.255.0 172.16.11.253
```

O caminho dos pacotes, quando se faz "ping" do tux2 para o tux1, dependem da configuração a ser utilizada.

Quando o tux2 tem uma rota para o tux1, então os pacotes vão passar pelo tux4, chegando depois ao tux1. Utilizando o comando "traceroute", é possível verificar esse fenómeno:

```
traceroute to 172.16.10.1 (172.16.10.1), 30 hops max, 60 byte packets
 1  172.16.11.253 (172.16.11.253)  0.298 ms  0.281 ms  0.274 ms
 2  172.16.10.1 (172.16.10.1)  0.747 ms  0.839 ms  0.837 ms
```

Quando tux2 não tem uma rota para o tux1 definida, então os pacotes vão passar primeiramente pelo router comercial:

```
traceroute to 172.16.10.1 (172.16.10.1), 30 hops max, 60 byte packets
 1  172.16.11.254 (172.16.11.254)  0.474 ms  0.528 ms  0.587 ms
 2  172.16.11.253 (172.16.11.253)  0.783 ms  0.338 ms  0.344 ms
 3  172.16.10.1 (172.16.10.1)  0.716 ms  0.788 ms  0.784 ms
```

No entanto, se os redirects estiverem ativados, então os pacotes só vão passar uma vez pelo router comercial, seguindo, depois, o caminho ilustrado no primeiro caso.

```
traceroute to 172.16.10.1 (172.16.10.1), 30 hops max, 60 byte packets
 1  172.16.11.254 (172.16.11.254)  0.486 ms  0.546 ms  0.598 ms
 2  172.16.11.253 (172.16.11.253)  0.798 ms  0.339 ms  0.346 ms
 3  172.16.10.1 (172.16.10.1)    0.677 ms  0.673 ms  0.739 ms
tux12:~/Desktop/scripts# traceroute 172.16.10.1
traceroute to 172.16.10.1 (172.16.10.1), 30 hops max, 60 byte packets
 1  172.16.11.253 (172.16.11.253)  0.197 ms  0.192 ms  0.182 ms
 2  172.16.10.1 (172.16.10.1)    0.421 ms  *  *
```

Para configurar o NAT num router comercial, é necessário proceder aos seguintes passos:

- na consola do router, deve-se utilizar o comando "interface gigabitethernet 0/X", para aceder à interface do router de número X;
- escrever o comando "ip nat inside", se a interface corresponde à que está ligada aos tux's, ou "ip nat outside", se está ligada ao router da sala;

É ainda necessário configurar os endereços a serem utilizados pelo router. No caso da experiência os comandos a utilizar são:

```
> ip nat pool ovrlid 172.16.1.19 172.16.1.19 prefix 24
> ip nat inside source list 1 pool ovrlid overload
> access-list 1 permit 172.16.10.0 0.0.0.7
> access-list 1 permit 172.16.11.0 0.0.0.7
```

NAT é uma técnica que consiste em reescrever os endereços IP de origem de um pacote, que passam por um router, de maneira que um computador de uma rede privada tenha acesso à rede pública. Para tal, o IP de origem é traduzido, em tempo real, para o endereço público, associado ao router que faz a ligação com a Internet.

Para separar e identificar os pacotes que devem ser enviados da rede pública para a privada, o router utiliza uma hashing table.

3.5. DNS

Esta experiência permite configurar o servidor DNS, assim como perceber qual é o seu papel nas redes de computadores.

Para configurar o DNS, é necessário alterar o conteúdo do ficheiro /etc/resolv.conf, modificando o search (lixa.netlab.fe.up.pt) e o nameserver (172.16.1.1).

Para modificar, pode-se abrir, normalmente, o ficheiro e alterar os valores pretendidos, ou então utilizar o comando "echo "search lixa.fe.up.pt\nnameserver 172.16.1.1" > /etc/resolv.conf".

O DNS é responsável por traduzir um nome num IP. Para tal, deve transmitir e receber pacotes.

Utilizando o comando "ping www.google.pt", é possível verificar a troca de informação necessária para obter os IPs correspondentes ao nome do host: o primeiro pacote, que tem como fonte o tux1 e destino o DNS da sala, vai perguntar qual o IP associado ao nome "www.google.pt". Por sua vez, será enviada uma resposta com a lista de todos os servidores com aquele nome associado.

3.6. Conexões TCP

Esta experiência consiste em transferir um ficheiro através do protocolo FTP, utilizando a aplicação de download desenvolvida pelo grupo.

Nesta aplicação, são abertas duas ligações TCP. Na primeira é enviada a informação de controlo FTP, para poder enviar pedidos ao servidor - como enviar username e password, e qual o ficheiro que queremos receber. A segunda ligação destina-se a transferir a informação do ficheiro propriamente dita.

A conexão TCP passa por várias fases, estando nelas incluídas as fases de sincronização das duas máquinas, aceitação e rejeição de pacotes, e a finalização de ligação.

O protocolo TCP utiliza um mecanismo ARQ similar ao Go-Back-N, utilizando uma Sliding Window. Este mecanismo é otimista, no sentido em que, em caso de erro na transmissão, reenvia um pacote de cada vez, pois assume que só um foi perdido. Campos relevantes para este método são o ACK, o tamanho da janela, e o número de sequência. É também importante saber qual o último byte que foi escrito, para atualizar o tamanho da Sliding Window.

Para controlar o congestionamento da rede, o protocolo TCP calcula se é possível enviar a informação sem o ultrapassar o buffer. Se tal acontecer, a aplicação bloqueia até haver mais espaço disponível.

Foi possível também observar, quanto à Sliding Window, o fenómeno de slow-start do TCP, mantendo depois o tamanho da janela.

Quando a aplicação foi corrida, ao mesmo tempo, em duas máquinas diferentes, foi notada mais interferência no crescimento da Sliding Window do TCP. Por esta razão, entre outros fatores, a velocidade de transferência foi reduzida, ainda que ligeiramente.

4. Conclusão

Durante a realização da aplicação de download utilizando o protocolo FTP e das experiências laboratoriais, a leitura e compreensão do guião tornou-se essencial para que fossem cumpridos todos os objetivos propostos. Foi necessário, também, o apontamento de várias informações e comandos, devido à quantidade substancial de detalhes e pormenores que eram necessários para que estas se realizassem com sucesso, que permitiram concluir as experiências sem erros e, desta forma, sem comprometer a realização das experiências seguintes.

Após a realização das várias experiências, partiu-se para o desenvolvimento da aplicação de download, sendo a solução final bastante estruturada, simples, robusta e bem-sucedida.

No final, a realização das duas partes do guião permitiu aos alunos ter uma breve ideia de como é o funcionamento de uma rede, neste caso de baixa complexidade, e, de forma descontraída e simples, praticar os conteúdos aprendidos nas aulas, assim como, assimilá-los.

Anexo I

O código fonte encontra-se anexado juntamente com o relatório na pasta code.zip.

Anexo II

Experiência 1

1 0.000000	G-ProCom_8b:e4:ef	Broadcast	ARP	42 Who has 172.16.10.254? Tell 172.16.10.1
2 0.000219	HewlettP_a6:a4:f8	G-ProCom_8b:e4:ef	ARP	60 172.16.10.254 is at 00:22:64:a6:a4:f8
3 0.000239	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request id=0x5385, seq=1/256, ttl=64 (reply in 4)
4 0.000495	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5385, seq=1/256, ttl=64 (request in 3)
5 0.108959	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8003
6 0.999006	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request id=0x5385, seq=2/512, ttl=64 (reply in 7)
7 0.999367	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5385, seq=2/512, ttl=64 (request in 6)
8 1.999002	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request id=0x5385, seq=3/768, ttl=64 (reply in 9)
9 1.999239	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5385, seq=3/768, ttl=64 (request in 8)
10 2.118713	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8003
11 2.999001	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request id=0x5385, seq=4/1024, ttl=64 (reply in 12)
12 2.999241	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5385, seq=4/1024, ttl=64 (request in 11)
13 3.330078	CiscoInc_3a:fc:03	CDP/VTP/DTP/PagP/UD...	CDP	453 Device ID: tux-sw1 Port ID: FastEthernet0/1
14 3.999007	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request id=0x5385, seq=5/1280, ttl=64 (reply in 15)
15 3.999145	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5385, seq=5/1280, ttl=64 (request in 14)
16 4.118512	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8003
17 4.895385	CiscoInc_3a:fc:03	CiscoInc_3a:fc:03	LOOP	60 Reply

Experiência 2

20 10.970286	HewlettP_a6:a4:f8	G-ProCom_8b:e4:ef	ARP	60 Who has 172.16.10.1? Tell 172.16.10.254
21 10.970303	G-ProCom_8b:e4:ef	HewlettP_a6:a4:f8	ARP	42 172.16.10.1 is at 00:0f:fe:8b:e4:ef
22 11.954360	172.16.10.1	172.16.10.255	ICMP	98 Echo (ping) request id=0x5ba9, seq=7/1792, ttl=64 (no response found...
23 11.954567	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5ba9, seq=7/1792, ttl=64
24 12.029125	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8003
25 12.954358	172.16.10.1	172.16.10.255	ICMP	98 Echo (ping) request id=0x5ba9, seq=8/2048, ttl=64 (no response found...
26 12.954744	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5ba9, seq=8/2048, ttl=64
27 13.276129	CiscoInc_3a:fc:03	CiscoInc_3a:fc:03	LOOP	60 Reply
28 13.954362	172.16.10.1	172.16.10.255	ICMP	98 Echo (ping) request id=0x5ba9, seq=9/2304, ttl=64 (no response found...
29 13.954567	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5ba9, seq=9/2304, ttl=64
30 14.034925	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8003
31 14.954361	172.16.10.1	172.16.10.255	ICMP	98 Echo (ping) request id=0x5ba9, seq=10/2560, ttl=64 (no response found...
32 14.954591	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5ba9, seq=10/2560, ttl=64
33 15.954364	172.16.10.1	172.16.10.255	ICMP	98 Echo (ping) request id=0x5ba9, seq=11/2816, ttl=64 (no response found...
34 15.954569	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x5ba9, seq=11/2816, ttl=64
35 16.039117	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8003

2 0.242945	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
3 2.247771	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
4 4.252633	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
5 6.257640	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
6 8.262705	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
7 10.007631	CiscoInc_3a:fc:04	CiscoInc_3a:fc:04	LOOP	60 Reply
8 10.267351	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
9 12.272251	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
10 14.277169	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
11 16.282035	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
12 18.286929	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
13 20.012361	CiscoInc_3a:fc:04	CiscoInc_3a:fc:04	LOOP	60 Reply
14 20.291822	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
15 22.296721	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
16 24.301640	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004
17 26.306536	CiscoInc_3a:fc:04	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8004

19 18.978737	172.16.10.1	172.16.10.255	ICMP	98 Echo (ping) request	id=0x5ba9, seq=4/1024, ttl=64 (no response found...
20 18.978779	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x5ba9, seq=4/1024, ttl=64
21 19.978758	172.16.10.1	172.16.10.255	ICMP	98 Echo (ping) request	id=0x5ba9, seq=5/1280, ttl=64 (no response found...
22 19.978797	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x5ba9, seq=5/1280, ttl=64
23 20.049601	CiscoInc_3a:fc:06	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00	Cost = 0 Port = 0x8006
24 20.978766	172.16.10.1	172.16.10.255	ICMP	98 Echo (ping) request	id=0x5ba9, seq=6/1536, ttl=64 (no response found...
25 20.978784	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x5ba9, seq=6/1536, ttl=64
26 20.994356	HewlettP_a6:a4:f8	G-ProCom_8b:e4:ef	ARP	42 Who has 172.16.10.1? Tell 172.16.10.254	
27 20.994700	G-ProCom_8b:e4:ef	HewlettP_a6:a4:f8	ARP	60 172.16.10.1 is at 00:0f:fe:8b:e4:ef	
28 21.978792	172.16.10.1	172.16.10.255	ICMP	98 Echo (ping) request	id=0x5ba9, seq=7/1792, ttl=64 (no response found...
29 21.978830	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x5ba9, seq=7/1792, ttl=64

Experiência 3

2 0.000199	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x64fd, seq=1/256, ttl=64 (request in 1)
3 0.428166	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00	Cost = 0 Port = 0x8003
4 0.998991	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request	id=0x64fd, seq=2/512, ttl=64 (reply in 5)
5 0.999185	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x64fd, seq=2/512, ttl=64 (request in 4)
6 2.000263	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request	id=0x64fd, seq=3/768, ttl=64 (reply in 7)
7 2.000434	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x64fd, seq=3/768, ttl=64 (request in 6)
8 2.341069	CiscoInc_3a:fc:03	CDP/VTP/DTP/PagP/UD...	CDP	435 Device ID: tux-sw1	Port ID: FastEthernet0/1
9 2.433106	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00	Cost = 0 Port = 0x8003
10 2.999264	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request	id=0x64fd, seq=4/1024, ttl=64 (reply in 11)
11 2.999457	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x64fd, seq=4/1024, ttl=64 (request in 10)
12 3.998266	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request	id=0x64fd, seq=5/1280, ttl=64 (reply in 13)
13 3.998411	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x64fd, seq=5/1280, ttl=64 (request in 12)
14 4.437806	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00	Cost = 0 Port = 0x8003
15 4.997872	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request	id=0x64fd, seq=6/1536, ttl=64 (reply in 16)
16 4.998119	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x64fd, seq=6/1536, ttl=64 (request in 15)
17 5.989887	CiscoInc_3a:fc:03	CiscoInc_3a:fc:03	LOOP	60 Reply	

10 14.178899	G-ProCom_8b:e4:ef	Broadcast	ARP	60 Who has 172.16.10.254? Tell 172.16.10.1	
11 14.178922	HewlettP_a6:a4:f8	G-ProCom_8b:e4:ef	ARP	42 172.16.10.254 is at 00:22:64:a6:a4:f8	
12 14.179255	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=1/256, ttl=64 (reply in 13)
13 14.179560	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=1/256, ttl=63 (request in 12)
14 15.180046	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=2/512, ttl=64 (reply in 15)
15 15.180227	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=2/512, ttl=63 (request in 14)
16 16.039029	CiscoInc_3a:fc:06	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00	Cost = 0 Port = 0x8006
17 16.179055	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=3/768, ttl=64 (reply in 18)
18 16.179249	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=3/768, ttl=63 (request in 17)
19 17.178083	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=4/1024, ttl=64 (reply in 20)
20 17.178292	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=4/1024, ttl=63 (request in 19)
21 18.049059	CiscoInc_3a:fc:06	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00	Cost = 0 Port = 0x8006
22 18.177094	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=5/1280, ttl=64 (reply in 23)
23 18.177261	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=5/1280, ttl=63 (request in 22)

8 10.412714	Kye_04:20:99	Broadcast	ARP	42 Who has 172.16.11.1? Tell 172.16.11.253	
9 10.412831	HewlettP_61:2e:c3	Kye_04:20:99	ARP	60 172.16.11.1 is at 00:21:5a:61:2e:c3	
10 10.412842	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=1/256, ttl=63 (reply in 11)
11 10.412966	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=1/256, ttl=64 (request in 10)
12 11.413494	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=2/512, ttl=63 (reply in 13)
13 11.413622	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=2/512, ttl=64 (request in 12)
14 12.029223	CiscoInc_3a:fc:08	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00	Cost = 0 Port = 0x8008
15 12.412494	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=3/768, ttl=63 (reply in 16)
16 12.412632	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=3/768, ttl=64 (request in 15)
17 13.411529	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=4/1024, ttl=63 (reply in 18)
18 13.411681	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=4/1024, ttl=64 (request in 17)
19 14.034041	CiscoInc_3a:fc:08	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00	Cost = 0 Port = 0x8008
20 14.410524	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=5/1280, ttl=63 (reply in 21)
21 14.410648	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply	id=0x68a3, seq=5/1280, ttl=64 (request in 20)
22 14.982356	CiscoInc_3a:fc:08	CiscoInc_3a:fc:08	LOOP	60 Reply	
23 15.410197	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request	id=0x68a3, seq=6/1536, ttl=63 (reply in 24)

Experiência 4

8	10.412714	Kye_04:20:99	Broadcast	ARP	42 Who has 172.16.11.1? Tell 172.16.11.253
9	10.412831	HewlettP_61:2e:c3	Kye_04:20:99	ARP	60 172.16.11.1 is at 00:21:5a:61:2e:c3
10	10.412842	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request id=0x68a3, seq=1/256, ttl=63 (reply in 11)
11	10.412966	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply id=0x68a3, seq=1/256, ttl=64 (request in 10)
12	11.413494	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request id=0x68a3, seq=2/512, ttl=63 (reply in 13)
13	11.413622	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply id=0x68a3, seq=2/512, ttl=64 (request in 12)
14	12.029223	CiscoInc_3a:fc:08	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8008
15	12.412494	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request id=0x68a3, seq=3/768, ttl=63 (reply in 16)
16	12.412632	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply id=0x68a3, seq=3/768, ttl=64 (request in 15)
17	13.411529	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request id=0x68a3, seq=4/1024, ttl=63 (reply in 18)
18	13.411681	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply id=0x68a3, seq=4/1024, ttl=64 (request in 17)
19	14.034041	CiscoInc_3a:fc:08	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/11/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8008
20	14.410524	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request id=0x68a3, seq=5/1280, ttl=63 (reply in 21)
21	14.410648	172.16.11.1	172.16.10.1	ICMP	98 Echo (ping) reply id=0x68a3, seq=5/1280, ttl=64 (request in 20)
22	14.982356	CiscoInc_3a:fc:08	CiscoInc_3a:fc:08	LOOP	60 Reply
23	15.410197	172.16.10.1	172.16.11.1	ICMP	98 Echo (ping) request id=0x68a3, seq=6/1536, ttl=63 (reply in 24)

8	6.926675	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request id=0x4f6f, seq=1/256, ttl=64 (reply in 9)
9	6.926827	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x4f6f, seq=1/256, ttl=64 (request in 8)
10	7.927943	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request id=0x4f6f, seq=2/512, ttl=64 (reply in 11)
11	7.928287	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x4f6f, seq=2/512, ttl=64 (request in 10)
12	8.019468	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8003
13	8.928220	172.16.10.1	172.16.10.254	ICMP	98 Echo (ping) request id=0x4f6f, seq=3/768, ttl=64 (reply in 14)
14	8.928584	172.16.10.254	172.16.10.1	ICMP	98 Echo (ping) reply id=0x4f6f, seq=3/768, ttl=64 (request in 13)

48	30.246805	172.16.10.1	172.16.1.19	ICMP	98 Echo (ping) request id=0x4f7f, seq=1/256, ttl=64 (reply in 49)
49	30.247473	172.16.1.19	172.16.10.1	ICMP	98 Echo (ping) reply id=0x4f7f, seq=1/256, ttl=254 (request in 48)
50	31.245801	172.16.10.1	172.16.1.19	ICMP	98 Echo (ping) request id=0x4f7f, seq=2/512, ttl=64 (reply in 51)
51	31.246435	172.16.1.19	172.16.10.1	ICMP	98 Echo (ping) reply id=0x4f7f, seq=2/512, ttl=254 (request in 50)
52	32.077415	CiscoInc_3a:fc:03	Spanning-tree-(for-...	STP	60 Conf. Root = 32768/10/fc:fb:fb:3a:fc:00 Cost = 0 Port = 0x8003
53	32.244799	172.16.10.1	172.16.1.19	ICMP	98 Echo (ping) request id=0x4f7f, seq=3/768, ttl=64 (reply in 54)
54	32.245446	172.16.1.19	172.16.10.1	ICMP	98 Echo (ping) reply id=0x4f7f, seq=3/768, ttl=254 (request in 53)

Experiência 5

2	1.042077000	G-ProCom_8c:af:af	Broadcast	ARP	42 Who has 172.16.40.254? Tell 172.16.40.1
3	1.042363000	Hewlett-_5a:7b:ea	G-ProCom_8c:af:af	ARP	60 172.16.40.254 is at 00:21:5a:5a:7b:ea
4	1.042374000	172.16.40.1	172.16.1.1	DNS	73 Standard query 0x7232 A www.google.pt
5	1.044961000	172.16.1.1	172.16.40.1	DNS	475 Standard query response 0x7232 A 194.210.238.148 A 194.210.238.152 A 194.210.238.154
6	1.045292000	172.16.40.1	194.210.238.148	ICMP	98 Echo (ping) request id=0x46a2, seq=1/256, ttl=64 (reply in 7)
7	1.051866000	194.210.238.148	172.16.40.1	ICMP	98 Echo (ping) reply id=0x46a2, seq=1/256, ttl=55 (request in 6)
8	1.053810000	172.16.40.1	172.16.1.1	DNS	88 Standard query 0x6652 PTR 148.238.210.194.in-addr.arpa
9	1.055451000	172.16.1.1	172.16.40.1	DNS	147 Standard query response 0x6652 No such name
10	2.004635000	Cisco_d4:1c:03	Spanning-tree-(for-br	STP	60 Conf. TC + Root = 32768/40/30:37:a6:d4:1c:00 Cost = 0 Port = 0x8003
11	2.046707000	172.16.40.1	194.210.238.148	ICMP	98 Echo (ping) request id=0x46a2, seq=2/512, ttl=64 (reply in 12)
12	2.052771000	194.210.238.148	172.16.40.1	ICMP	98 Echo (ping) reply id=0x46a2, seq=2/512, ttl=55 (request in 11)
13	2.977542000	Cisco_d4:1c:03	Cisco_d4:1c:03	LOOP	60 Reply
14	3.047866000	172.16.40.1	194.210.238.148	ICMP	98 Echo (ping) request id=0x46a2, seq=3/768, ttl=64 (reply in 15)
15	3.053659000	194.210.238.148	172.16.40.1	ICMP	98 Echo (ping) reply id=0x46a2, seq=3/768, ttl=55 (request in 14)

Experiência 6

5	1.409895000	172.16.40.1	172.16.1.1	DNS	69	Standard query 0xd615 A ftp.up.pt
6	1.411176000	172.16.1.1	172.16.40.1	DNS	333	Standard query response 0xd615 A 193.136.37.8
7	1.411343000	172.16.40.1	193.136.37.8	TCP	74	48262->21 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=4595909 TSecr=0 WS=128
8	1.413740000	193.136.37.8	172.16.40.1	TCP	70	21->48262 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1380 SACK_PERM=1 TSval=734821635 TSecr=4
9	1.413763000	172.16.40.1	193.136.37.8	TCP	66	48262->21 [ACK] Seq=1 Ack=1 Win=3737600 Len=0 TSval=4595910 TSecr=734821635
10	1.418775000	193.136.37.8	172.16.40.1	FTP	106	Response: 220 Bem-vindo \303\240 Universidade do Porto
11	1.418810000	172.16.40.1	193.136.37.8	TCP	66	48262->21 [ACK] Seq=1 Ack=41 Win=3737600 Len=0 TSval=4595911 TSecr=734821636
12	1.418905000	172.16.40.1	193.136.37.8	FTP	81	Request: USER anonymous
13	1.420516000	193.136.37.8	172.16.40.1	TCP	66	21->48262 [ACK] Seq=41 Ack=16 Win=5792 Len=0 TSval=734821637 TSecr=4595911
14	1.420526000	193.136.37.8	172.16.40.1	FTP	100	Response: 331 Please specify the password.
15	1.420590000	172.16.40.1	193.136.37.8	FTP	83	Request: PASS mail@domain
16	1.423977000	193.136.37.8	172.16.40.1	FTP	89	Response: 230 Login successful.
17	1.424226000	172.16.40.1	193.136.37.8	FTP	71	Request: PASV
18	1.426295000	193.136.37.8	172.16.40.1	FTP	116	Response: 227 Entering Passive Mode (193,136,37,8,238,186)
19	1.426465000	172.16.40.1	193.136.37.8	TCP	74	40696->61114 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=4595913 TSecr=0 WS=128
20	1.428080000	193.136.37.8	172.16.40.1	TCP	70	61114->40696 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1380 SACK_PERM=1 TSval=734821638 TSecr=4
21	1.428098000	172.16.40.1	193.136.37.8	TCP	66	40696->61114 [ACK] Seq=1 Ack=1 Win=3737600 Len=0 TSval=4595913 TSecr=734821638
22	1.428194000	172.16.40.1	193.136.37.8	FTP	95	Request: RETR pub/CPAN/RECENT-1Q.json
23	1.431068000	193.136.37.8	172.16.40.1	FTP	152	Response: 150 Opening BINARY mode data connection for pub/CPAN/RECENT-1Q.json (4519410 byte

3078	1.822663000	193.136.37.8	172.16.40.1	FTP-DATA	2802	FTP Data: 2736 bytes
3079	1.822678000	172.16.40.1	193.136.37.8	TCP	66	40696->61114 [ACK] Seq=1 Ack=4507561 Win=8388480 Len=0 TSval=4596012 TSecr=734821735
3080	1.822913000	193.136.37.8	172.16.40.1	FTP-DATA	2802	FTP Data: 2736 bytes
3081	1.822929000	172.16.40.1	193.136.37.8	TCP	66	40696->61114 [ACK] Seq=1 Ack=4510297 Win=8388480 Len=0 TSval=4596012 TSecr=734821735
3082	1.823163000	193.136.37.8	172.16.40.1	FTP-DATA	2802	FTP Data: 2736 bytes
3083	1.823180000	172.16.40.1	193.136.37.8	TCP	66	40696->61114 [ACK] Seq=1 Ack=4513033 Win=8388480 Len=0 TSval=4596012 TSecr=734821735
3084	1.823414000	193.136.37.8	172.16.40.1	FTP-DATA	4170	FTP Data: 4104 bytes
3085	1.823430000	172.16.40.1	193.136.37.8	TCP	66	40696->61114 [ACK] Seq=1 Ack=4517137 Win=8388480 Len=0 TSval=4596012 TSecr=734821735
3086	1.823662000	193.136.37.8	172.16.40.1	FTP-DATA	2340	FTP Data: 2274 bytes
3087	1.823686000	172.16.40.1	193.136.37.8	TCP	66	40696->61114 [ACK] Seq=1 Ack=4519412 Win=8388480 Len=0 TSval=4596012 TSecr=734821735
3088	1.823925000	172.16.40.1	193.136.37.8	TCP	66	40696->61114 [FIN, ACK] Seq=1 Ack=4519412 Win=8388480 Len=0 TSval=4596012 TSecr=734821735
3089	1.823952000	172.16.40.1	193.136.37.8	TCP	66	48262->21 [FIN, ACK] Seq=67 Ack=234 Win=3737600 Len=0 TSval=4596012 TSecr=734821639



