U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Redes de Computadores 2024/2025

Redes de Computadores

Relatório do 2º trabalho laboratorial

Maria João Vieira (up202204802@up.pt)
Rodrigo Martins (up202008868@up.pt)
Turma 5

23 de dezembro de 2024

1. Sumário

No âmbito da unidade curricular de Redes de Computadores, lecionada no primeiro semestre do terceiro ano da Licenciatura em Engenharia Informática e Computação, desenvolvemos um projeto com o objetivo de criar uma aplicação de download utilizando o protocolo FTP sobre TCP/IP. Complementarmente, projetámos e implementámos uma rede de computadores para testar e validar o funcionamento da aplicação.

Este projeto permitiu consolidar os conhecimentos adquiridos ao longo da unidade curricular, proporcionando uma aplicação prática dos conteúdos lecionados. Além disso, a execução das tarefas propostas possibilitou o aprofundamento do nosso entendimento sobre o funcionamento de redes de comunicação e protocolos associados.

2. Introdução

Este projeto tem dois grandes objetivos: configurar uma rede de computadores em ambiente laboratorial, composta por várias máquinas clientes, um router e um switch; e analisar o tráfego gerado na comunicação entre os clientes e o servidor FTP. Essa análise permitiu validar tanto as tarefas propostas quanto o funcionamento da aplicação desenvolvida na primeira parte do projeto. A aplicação foi implementada utilizando o protocolo FTP (File Transfer Protocol), estabelecendo conexões através do protocolo TCP (Transmission Control Protocol) com o uso de sockets.

Resumindo, o projeto está dividido em duas partes. Na primeira, abordamos a arquitetura e os resultados esperados da aplicação de download. Na segunda, realizamos uma análise detalhada das experiências propostas e respondemos às questões associadas.

Este relatório está organizado nas seguintes secções:

3. Parte 1: Aplicação de Download

Como referido anteriormente, o objetivo da primeira parte do projeto foi desenvolver uma aplicação de download que aceite links no formato **ftp://:@/**. Essa aplicação tem como finalidade realizar o download de qualquer tipo de ficheiro disponibilizado por um servidor FTP, garantindo a correta interação com o protocolo FTP para estabelecer a conexão, autenticar o utilizador e transferir o ficheiro pretendido.

3.1. Arquitetura

A aplicação desenvolvida é composta por três ficheiros interligados, cada um com uma função específica no processo de comunicação com o servidor FTP e na realização do download do ficheiro.

O módulo **clientTCP.c** é responsável pela criação do cliente TCP, estabelecendo a ligação com o servidor FTP na porta padrão (21) através de um socket TCP/IP.

O **getip.c** implementa a função `gethostbyname`, que resolve o endereço IP de um hostname fornecido, permitindo que o cliente obtenha o IP do servidor FTP a partir do seu nome de domínio, facilitando assim a conexão com o servidor.

Por fim, o **download.c** é o núcleo da aplicação, sendo responsável pelas operações específicas do protocolo FTP para realizar o download do ficheiro. Inicialmente, a aplicação usa a função **parse_ftp_url** para extrair informações da URL fornecida, como o hostname, credenciais de autenticação (utilizador e senha) e o caminho do ficheiro a ser transferido. A aplicação tenta então conectar-se ao servidor FTP na porta 21 utilizando a função **connect_to_ftp_server**. Uma vez estabelecida a ligação, o cliente envia os comandos FTP essenciais: **USER e PASS** para autenticação, **PASV** para colocar o cliente em modo passivo e abrir um canal de dados separado para a transferência, e **RETR** para solicitar a transferência do ficheiro. O ficheiro transferido é então armazenado no sistema de ficheiros local, e as conexões de controlo e dados são encerradas.

Este conjunto de módulos permite que a aplicação realize a transferência de ficheiros através do protocolo FTP de forma eficiente, respeitando os comandos e procedimentos necessários para autenticação e transferência de dados.

3.2 Resultados

Para testar a nossa aplicação, utilizamos vários comandos e ficheiros associados para verificar se a tarefa estava a ser executada corretamente. Inicialmente, realizámos testes com servidores FTP públicos, e todos os testes foram bem-sucedidos, como pode ser visto nas Figura 1, Figura 2 e Figura 3 anexas a este relatório.

No entanto, não conseguimos realizar todas as experiências. Quando finalmente testámos a aplicação no ambiente laboratorial, identificámos um erro no comando PASV. Este erro não foi detectado durante os testes realizados em casa, o que impediu a realização desta tarefa com 100% de sucesso.

4. Parte 2: Configuração de Rede

4.1. Experiência 1 - Configurar uma rede IP

O objetivo desta experiência foi configurar uma rede IP entre dois computadores (TUXY3 e TUXY4) e analisar a comunicação entre eles, observando os pacotes ARP e ICMP gerados durante o processo. Esta experiência também proporcionou a exploração da configuração das interfaces de rede e a inspeção das tabelas de encaminhamento e ARP.

A experiência começou com a conexão das portas E1 dos computadores TUXY3 e TUXY4 ao switch, seguida da configuração dos endereços IP nas interfaces eth1 de cada máquina. A configuração foi feita com os seguintes comandos no terminal:

- TUXY3: *ifconfig eth1 172.16.Y0.1/24*
- TUXY4: *ifconfig eth1 172.16.Y0.254/24*

Além disso, o switch foi configurado para as experiências seguintes, conectando a sua consola à porta série do TUXY3. Para testar a conectividade entre os dispositivos, foi executado o comando **ping** de TUXY3 para TUXY4, gerando pacotes ICMP de Solicitação de Eco (Echo Request) e Resposta de Eco (Echo Reply), usados para verificar a conectividade na rede. Em seguida, foram consultadas as tabelas de rotas e ARP de ambos os dispositivos:

- TUXY3: IP 172.16.30.1/24, MAC 00:50:fc:ee:0e:93
- TUXY4: IP 172.16.30.254/24, MAC 00:08:54:50:3f:2c

Posteriormente, capturamos os pacotes no Wireshark (Figura 4) enquanto executamos novamente o comando **ping**, após limpar as entradas da tabela ARP. Isso permitiu observar de forma mais clara a função do protocolo ARP. Na captura, foi possível verificar que o campo “Protocol” identifica se os pacotes são ARP, ICMP ou IP, e o campo “Length” mostra o comprimento do quadro recebido.

Além disso, a análise dos pacotes ICMP mostrou que o endereço MAC do remetente corresponde ao MAC de origem, enquanto o endereço MAC do dispositivo de destino é o MAC de destino. Os endereços IP registados eram os dos dispositivos de origem e destino, confirmando que ambos os dispositivos estavam a comunicar corretamente na rede.

A captura também evidenciou o funcionamento do protocolo ARP (Address Resolution Protocol), que é utilizado para mapear endereços IP em endereços MAC, essencial para a comunicação em redes Ethernet. Antes de enviar pacotes IP entre TUXY3 e TUXY4, o dispositivo de origem precisa conhecer o endereço MAC do destino. Como o mapeamento ARP foi apagado, pacotes ARP de solicitação ("request") foram enviados para descobrir o endereço MAC

correspondente ao IP de destino. Estes pacotes ARP apareceram nas primeiras linhas da captura, evidenciando a sua função no processo de comunicação.

A “loopback interface” é uma interface de rede virtual que está sempre ativa e conecta o dispositivo a si próprio. O endereço padrão que esta utiliza, é o **127.0.0.1(IPv4).** Esta tem uma grande importância, visto que fornece testes de conectividade, isto é, por exemplo, ao executar um **ping** apenas é testado o funcionamento interno do protocolo de rede. Também o isolamento que esta interface fornece, é essencial, não necessitando de expormos os serviços de rede à rede externa.

4.2. Experiência 2 - Implementar duas bridges num switch

Esta experiência tem como objetivo a configuração de duas bridges num switch para segmentar a rede em dois domínios de broadcast, e analisar o seu impacto na comunicação entre os dispositivos e o tráfego de pacotes, como os pings e os broadcasts.

Primeiro, repetimos a configuração da experiência anterior com os computadores TUXY3 e TUXY4. Em seguida, conectamos a interface E1 do TUXY2 ao switch e configuramos com o comando *ifconfig eth1 172.16.Y1.1/24*.

Na consola do switch, criámos as bridges Y0 e Y1, utilizando o comando */interface bridge add name=<>*. Após isso, removemos as portas às quais os TUX estavam conectados (portas 2, 3 e 4) com o comando */interface bridge port remove [find interface=ether<>]*. Em seguida, adicionamos as portas dos TUXs às bridges correspondentes: o TUXY2 foi adicionado à bridgeY1 e o TUXY3 e TUXY4 foram adicionados à bridgeY0, utilizando o comando */interface bridge port add bridge=<> [find interface=ether<>].*

Para verificar a conectividade entre os dispositivos, realizámos um teste com ping do TUXY3 para o TUXY4 e para o TUXY2. A análise dos pacotes mostrou que a comunicação entre o TUXY3 e o TUXY4 funcionava corretamente, com pacotes ICMP de Solicitação de Eco (Echo Request) e Resposta de Eco (Echo Reply) alternados. No entanto, os resultados do ping entre o TUXY3 e o TUXY2 indicaram que não havia conexão entre eles (Figura 5).

Em seguida, realizámos um ping broadcast no TUXY3. Neste teste, o TUXY2 (Figura 6) não recebeu nenhum pacote, enquanto o TUXY3 (Figura 7) e o TUXY4 (Figura 8) receberam os pacotes, mas não responderam. Executamos o ping broadcast também no TUXY2, o que confirmou que não havia conexão entre o TUXY2 e os outros dispositivos. A captura de pacotes no TUXY2 (Figura 9) mostrou pacotes enviados, mas o TUXY3 (Figura 10) e o TUXY4 (Figura 11) não receberam nada.

A partir destes resultados, concluímos que o TUXY2, por fazer parte de outra bridge, não consegue comunicar com os outros dispositivos. Isso mostra que o tráfego de broadcast é isolado entre as bridges, garantindo que as comunicações dentro de cada segmento de rede não afetem o outro. Este isolamento de tráfego é essencial para melhorar o desempenho da rede e evitar a propagação desnecessária de broadcasts entre diferentes segmentos.

4.3. Experiência 3 - Configurar um router em Linux

Esta experiência tem como objetivo compreender o funcionamento do envio de pacotes em uma rede composta por sub-redes, possibilitando a comunicação entre os dispositivos TUXY2 e TUXY3.

Para iniciar a experiência, configuramos a interface Ethernet 2 do TUXY4 e conectamos o E2 do TUXY4 ao switch. Também foi necessário permitir o IP Forwarding para que o TUXY4 funcionasse como um router, ou seja, para encaminhar pacotes entre duas sub-redes. Além disso, desativamos a opção de ignorar ICMP Echo Request, permitindo que o TUXY4 respondesse aos pings de broadcast.

Para garantir a comunicação entre os dispositivos, configuramos a rota 172.16.Y0.0/24 no TUXY2 com o gateway 172.16.Y1.253, e no TUXY3 configuramos a rota para o gateway 172.16.Y0.254. Ambas as máquinas possuem uma rota com o mesmo gateway, pois o TUXY4 é o nó comum entre as duas sub-redes.

Abordando um pouco as tabelas, conseguimos consultar informação variada (com o comando *route -n*) como:

- O endereço de IP do destino da rota;
- A gateway, ou seja, o IP por onde a rota passará;
- A Genmask, a máscara de endereço que é usada de forma a corresponder um endereço de IP ao valor mostrado no campo de destino;
- A interface de rede que deve ser usada para encaminhar o pacote.;
- A flag, onde mostra as características da rota.

Depois de observarmos as rotas, passamos a fazer uma captura no TUXY3, para podermos fazer **ping** de outras interfaces, tal como 172.16.Y0.254 (Figura 12), 172.16.Y1.253 (Figura 13), 172.16.Y1.1 (Figura 14).

Apesar do ping ser realizado entre o TUXY3 e o TUXY2, os endereços MAC associados são consultados no TUXY4, pois os pacotes são primeiro enviados para o TUXY4 e, em seguida, encaminhados para o TUXY2. Isso ocorre porque não há comunicação direta entre o TUXY2 e o TUXY3. Pelas imagens das capturas fornecidas nos anexos, é possível observar pacotes ICMP de request e reply, visto que todos os TUX's conseguem comunicar entre si. Isto só é possível devido à configuração das rotas nesta experiência.

De seguida, iniciou-se a captura no TuxY4 com uma instância para cada interface de rede. Apagamos as entradas das tabelas ARP dos 3 dispositivos, para assegurar a reconstrução do mapeamento de endereços. Após parar as capturas, os resultados confirmaram o encaminhamento dos pacotes, com a comunicação e tabela restabelecidas. (Figura 15, Figura 16)

4.4. Experiência 4 - Configurar um router comercial com NAT

O objetivo desta experiência é configurar um router comercial com funcionalidade NAT (Network Address Translation) e entender o seu impacto na comunicação entre redes distintas. Inicialmente, conectámos os interfaces ether1 e ether2 do router aos dispositivos PY.12 e ao switch, respetivamente. Depois, removemos a porta conectada ao ether2 do router e adicionámo-la à bridgeY1, utilizando o comando: */interface bridge port add bridge=bridgeY1 interface=ether<>.*

Em seguida, alterámos o cabo da consola do switch para o router para iniciar a configuração. A primeira etapa da configuração do router foi definir os endereços IP nas interfaces do router, utilizando os seguintes comandos:

- *ip adress add address=172.16.Y1.254/24 interface=ether2*
- *ip adress add address=172.16.1.Y1/24 interface=ether1*

Na configuração das rotas, se ainda não existissem, foi necessário adicionar rotas nos dispositivos TUX para garantir que todos tivessem acesso ao router. As rotas configuradas foram as seguintes:

TUXY3:

- Rota para router com gateway pelo TUXY4: *route add -net 172.16.Y1.0/24 gw 172.16.Y0.254*
- Rota para servidor FTP com gateway pelo TUXY4: *route add -net 172.16.1.0/24 gw 172.16.Y0.254*

TUXY4:

- Rota para servidor FTP com gateway pelo router: *route add -net 172.16.1.0/24 gw 172.16.Y1.254/24*

TUXY2:

- Rota para TUX3 com gateway pelo TUXY4: *route add -net 172.16.Y0.0/24 gw 172.16.Y1.253*
- Rota para servidor FTP com gateway pelo router: *route add -net 172.16.1.0/24 gw 172.16.Y1.254*

Router:

- Rota para TUX3 pelo TUXY4: *ip route add dst-address=172.16.30.0/24 gateway=172.16.31.253*

Após a configuração das rotas, foi testada a conectividade entre os TUX's utilizando o comando **ping**. No TUX3, foram feitos pings para o TUXY2 (Figura 17), TUXY4 (Figura 18) e o Router (Figura 19). Todos os testes de conectividade foram bem-sucedidos, com pacotes ICMP de "request" e "reply" trocados entre os dispositivos, confirmando que a configuração das rotas foi realizada corretamente.

Após desativarmos o redirecionamento ICMP no TUXY2, utilizámos os seguintes comandos para garantir que o TUXY2 não aceitasse mais redirecionamentos de ICMP: *sysctl net.ipv4.conf.eth1.accept_redirects=0* e *sysctl net.ipv4.conf.all.accept_redirects=0.*

Em seguida, alteramos as rotas no TUXY2 para utilizar o router como gateway para a sub-rede 172.16.Y0.0/24, em vez de utilizar o TUXY4. A nova configuração da rota foi realizada para garantir que os pacotes destinados à sub-rede fossem encaminhados através do router.

Após as alterações, fizemos testes de conectividade **ping** (Figura 20) e confirmámos que a alteração no caminho de encaminhamento foi bem-sucedida, com os pacotes sendo

redirecionados pelo router. Isso confirmou que o comportamento do redirecionamento ICMP foi alterado, e a comunicação foi devidamente ajustada para seguir o novo caminho.

O comando **traceroute** para o TUXY3 confirmou a mudança no percurso após as alterações nas rotas. Quando o TUXY2 utilizava o **TUXY4** como gateway para a sub-rede 172.16.Y0.0/24, o **traceroute** mostrou que a comunicação não passava pelo router, mas seguia um caminho mais direto através do TUXY4. Isso evidenciou que, sem a utilização do router como gateway, a comunicação seguia um percurso mais simples.

Após a ativação do redirecionamento ICMP, o **traceroute** para o TUXY3 mostrou que o percurso foi alterado novamente. Inicialmente, os pacotes eram encaminhados via router, mas, após o redirecionamento ICMP, o caminho mudou para uma ligação direta ao TUXY4. Isso aconteceu devido à mensagem **"next hop"**, indicando que o tráfego estava a ser redirecionado, evitando caminhos desnecessários.

Testámos a conectividade do TUXY3 com o servidor FTP através de **ping**, com e sem o **NAT** ativado. Quando o **NAT** estava ativado, a comunicação foi bem-sucedida, pois o NAT permite a tradução dos endereços privados (da sub-rede local) para endereços públicos, possibilitando a comunicação entre as sub-redes e a rede externa, como o servidor FTP.

Por outro lado, ao desativar o **NAT** com o comando /ip firewall nat disable 0, a comunicação falhou. Isso ocorreu porque, sem o **NAT**, os endereços privados não são traduzidos para endereços públicos, o que impossibilita a comunicação entre as redes privadas (como a do TUXY3) e redes externas (como a do servidor FTP). Os pacotes enviados pela rede privada não conseguem alcançar o destino e, da mesma forma, as respostas também não são retornadas, pois os IPs privados não são roteáveis na Internet sem a devida tradução de endereços.

Isto ilustra a importância do **NAT** para permitir a comunicação entre redes privadas e externas, garantindo que os endereços privados sejam traduzidos corretamente e a comunicação seja viável.

4.5. Experiência 5 - DNS

Nesta experiência, configurámos o serviço **DNS (Domain Name System)** nos três dispositivos TUX, com o objetivo de permitir que o domínio **ftp.netlab.fe.up.pt** fosse traduzido para um endereço IP numérico, facilitando a comunicação entre os dispositivos na rede.

Para configurar o **DNS** em cada TUX, editámos o arquivo de configuração **/etc/resolv.conf** com o comando: *nano /etc/resolv.conf.* Este arquivo foi configurado para incluir o endereço do servidor DNS (**nameserver 10.227.20.3**), garantindo que todos os TUX estivessem a utilizar o mesmo servidor para resolução de nomes.

Na análise da captura (Figura 21), referente ao **ping** para **www.google.com**, podemos observar os pacotes trocados durante a comunicação **DNS**. A comunicação DNS é realizada em duas fases principais:

1. **Solicitação DNS (DNS Query)**: O dispositivo (TUX) envia uma solicitação para resolver o nome de domínio www.google.com. O pacote contém o nome de domínio que o dispositivo deseja resolver, e a consulta será direcionada para o servidor DNS configurado (neste caso, **10.227.20.3**).

2. **Resposta DNS (DNS Response)**: O servidor DNS recebe a consulta e responde com o endereço IP associado ao domínio www.google.com. A resposta pode incluir um ou mais registos de **A** (endereço IPv4) que correspondem ao domínio consultado.

SEndo assim, ao analisar os pacotes trocados entre o dispositivo e o servidor DNS, podemos ver a solicitação e a resposta DNS no tráfego da rede. O processo de resolução de DNS permite que o nome www.google.com seja convertido em um endereço IP numérico, essencial para estabelecer a comunicação entre o dispositivo e o servidor.

4.6. Experiência 6 - Conexões TCP

Nesta experiência, o objetivo é explorar as conexões TCP (Transmission Control Protocol) e compreender como o TCP gere a transmissão de dados. Além disso, procura-se analisar o tráfego gerado por uma aplicação de download utilizando o Wireshark.

Para realizar a experiência, foi necessário desenvolver uma aplicação de download, conforme indicado nos materiais de apoio. O passo seguinte consistia em compilar e executar essa aplicação no TUXY3, a fim de verificar a transmissão e receção correta de um arquivo. Durante esse processo, também seria observado o comportamento das conexões TCP através do tráfego capturado pelo Wireshark, permitindo uma análise detalhada do protocolo.

Contudo, devido à obrigatoriedade de desenvolver a aplicação fora do espaço laboratorial, não foi possível testar o código como previsto. Isso resultou na não receção do arquivo, impedindo a análise do tráfego no Wireshark e a captura das informações necessárias. Dessa forma, a experiência não foi concluída da forma esperada.

5. Conclusão

Em suma, podemos concluir que adquirimos os conceitos essenciais para a implementação de quase todas as tarefas descritas no guião. Além de absorvermos diversos detalhes importantes, conseguimos configurar uma rede de computadores, o que nos proporcionou uma compreensão mais aprofundada sobre o funcionamento das redes e como trabalhar com elas na prática. O desenvolvimento da aplicação de download também foi bastante proveitoso, pois permitiu compreender melhor o funcionamento do protocolo FTP, além de nos oferecer uma visão mais clara sobre a transmissão de dados e a gestão das conexões numa rede.

Esta experiência foi fundamental para consolidar os nossos conhecimentos na área de redes e protocolos de comunicação.

6. ANEXOS

ANEXO 1 - Código da implementação do download e resultados :

1.1. download.c

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <netdb.h>
#include <errno.h>

#define BUFFER_SIZE 1024
#define FTP_PORT 21

// sair do programa em caso de erro
void handle_error(const char *message) {
    perror(message);
    exit(EXIT_FAILURE);
}

// infos modo passivo
void extract_pasv_info(const char *response, char *ip, int *port) {
    int a, b, c, d, p1, p2;
    // IP e porta da resposta PASV
    if (sscanf(response, "227 Entering Passive Mode (%d,%d,%d,%d,%d,%d)", &a, &b, &c, &d, &p1, &p2) == 6) {
        sprintf(ip, "%d.%d.%d.%d", a, b, c, d);
        *port = p1 * 256 + p2;
    } else {
        fprintf(stderr, "Invalid PASV response format.\n");
        exit(EXIT_FAILURE);
    }
}

 //  analisar a URL FTP e extrair a senha,user,host e path
void parse_ftp_url(const char *url, char *host, char *user, char *pass, char *path) {
    // utilizador e senha
    if (sscanf(url, "ftp://%99[^:]:%99[^@]@%99[^/]/%s", user, pass, host, path) == 4) return;
    // utilizador anónimo sem senha
    if (sscanf(url, "ftp://anonymous@%99[^/]/%s", host, path) == 2) {
        strcpy(user, "anonymous");
        strcpy(pass, "anonymous");
        return;
    }
    // apenas o hostname e path - sem utilizador e senha
    if (sscanf(url, "ftp://%99[^/]/%s", host, path) == 2) {
        strcpy(user, "anonymous");
        strcpy(pass, "anonymous");
        return;
    }
    fprintf(stderr, "URL format error\n");
    exit(EXIT_FAILURE);
 }
```

```c
// conectar ao servidor FTP
int connect_to_ftp_server(const char *hostname) {
    struct hostent *server;
    struct sockaddr_in server_addr;
    int sockfd;

    // checka hostname para obter o endereço IP do servidor
    if ((server = gethostbyname(hostname)) == NULL) {
        herror("gethostbyname");
        exit(EXIT_FAILURE);
    }

    // Cria o socket para a conexão de controle
    if ((sockfd = socket(AF_INET, SOCK_STREAM, 0)) < 0) {
        handle_error("socket");
    }

    // Configura a estrutura de endereço do servidor
    memset(&server_addr, 0, sizeof(server_addr));
    server_addr.sin_family = AF_INET;
    server_addr.sin_port = htons(FTP_PORT);
    memcpy(&server_addr.sin_addr.s_addr, server->h_addr, server->h_length);

    // Conexão ao FTP
    if (connect(sockfd, (struct sockaddr *)&server_addr, sizeof(server_addr)) < 0) {
        handle_error("connect");
    }

    return sockfd;
}

// enviar comandos FTP
void send_ftp_command(int sockfd, const char *command, const char *param) {
    char buffer[BUFFER_SIZE];
    snprintf(buffer, sizeof(buffer), "%s %s\r\n", command, param ? param : "");
    if (write(sockfd, buffer, strlen(buffer)) < 0) {
        handle_error("write");
    }
}

//  ler a resposta do servidor FTP
void read_ftp_response(int sockfd, char *buffer) {
    ssize_t bytes_read = read(sockfd, buffer, BUFFER_SIZE - 1);
    if (bytes_read < 0) {
        handle_error("read");
    }
    buffer[bytes_read] = '\0';
}
```

```c
int main(int argc, char *argv[]) {
    if (argc != 2) {
        fprintf(stderr, "Usage: %s <ftp_url>\n", argv[0]);
        exit(EXIT_FAILURE);
    }

    char host[BUFFER_SIZE], user[BUFFER_SIZE], pass[BUFFER_SIZE], path[BUFFER_SIZE];
    char ip[BUFFER_SIZE], buffer[BUFFER_SIZE];
    int control_sock, data_sock, port;

    parse_ftp_url(argv[1], host, user, pass, path);
    printf("FTP server: %s\n", host);

    // Conexão ao FTP
    control_sock = connect_to_ftp_server(host);

    // Lê a resposta inicial do servidor
    read_ftp_response(control_sock, buffer);
    printf("Server: %s", buffer);

    // manda user
    send_ftp_command(control_sock, "USER", user);
    read_ftp_response(control_sock, buffer);
    printf("USER Response: %s", buffer);

    // manda pass
    send_ftp_command(control_sock, "PASS", pass);
    read_ftp_response(control_sock, buffer);
    printf("PASS Response: %s", buffer);

    // Entra no modo passivo (PASV)
    send_ftp_command(control_sock, "PASV", NULL);
    read_ftp_response(control_sock, buffer);
    printf("PASV Response: %s", buffer);

    // info do modo passivo
    extract_pasv_info(buffer, ip, &port);
    printf("Passive Mode IP: %s, Port: %d\n", ip, port);

    // Configura a conexão de dados
    struct sockaddr_in data_addr;
    data_sock = socket(AF_INET, SOCK_STREAM, 0);
    if (data_sock < 0) {
        handle_error("data socket");
    }
```

```c
    memset(&data_addr, 0, sizeof(data_addr));
    data_addr.sin_family = AF_INET;
    data_addr.sin_port = htons(port);
    if (inet_pton(AF_INET, ip, &data_addr.sin_addr) <= 0) {
        handle_error("inet_pton");
    }

    // Conexão da socket de dados
    if (connect(data_sock, (struct sockaddr *)&data_addr, sizeof(data_addr)) < 0) {
        handle_error("data connect");
    }

    // Envia o comando RETR para baixar o arquivo
    send_ftp_command(control_sock, "RETR", path);
    read_ftp_response(control_sock, buffer);
    printf("RETR Response: %s", buffer);

    // Ajusta o nome do arquivo para salvar
    const char *output_filename = strrchr(path, '/');
    output_filename = output_filename ? output_filename + 1 : path;

    FILE *file = fopen(output_filename, "wb");
    if (!file) {
        fprintf(stderr, "Error opening file '%s': %s\n", output_filename, strerror(errno));
        exit(EXIT_FAILURE);
    }

    // ler socket e escrever no arquivo
    int bytes_read;
    while ((bytes_read = read(data_sock, buffer, BUFFER_SIZE)) > 0) {
        fwrite(buffer, 1, bytes_read, file);
    }

    fclose(file);
    close(data_sock);

    read_ftp_response(control_sock, buffer);
    printf("Final Response: %s", buffer);

    close(control_sock);
    printf("Download complete!\n");

    return 0;
}
```

1.2. Resultados do download

```
roger@RogerDesktop:~/universidade/RCOM2$ ./download ftp://ftp.up.pt/pub
/gnu/emacs/elisp-manual-21-2.8.tar.gz
FTP server: ftp.up.pt
Server: 220-Welcome to the University of Porto's mirror archive (mirrors.up.pt)
220----------------------------------------------------------------
220-
220-All connections and transfers are logged. The max number of connections is 200.
220-
220-For more information please visit our website: http://mirrors.up.pt/
220-Questions and comments can be sent to mirrors@uporto.pt
220-
220-
220
USER Response: 331 Please specify the password.
PASS Response: 230 Login successful.
PASV Response: 227 Entering Passive Mode (193,137,29,15,206,157).
Passive Mode IP: 193.137.29.15, Port: 52893
RETR Response: 150 Opening BINARY mode data connection for pub/gnu/emacs/elisp-manual-21-2.8.tar.gz (2455995 bytes).
Final Response: 226 Transfer complete.
Download complete!
```

Figura 1: Resultado URL1

```
roger@RogerDesktop:~/universidade/RCOM2$ ./download ftp://demo:password@test.rebex.net/readme.txt
FTP server: test.rebex.net
Server: 220-Welcome to test.rebex.net!
    See https://test.rebex.net/ for more information and terms of use.
220 If you don't have an account, log in as 'anonymous' or 'ftp'.
USER Response: 331 Anonymous login OK, send your complete email address as your password.
PASS Response: 230 User 'demo' logged in.
PASV Response: 227 Entering Passive Mode (194,108,117,16,4,20)
Passive Mode IP: 194.108.117.16, Port: 1044
RETR Response: 125 Data connection already open; starting 'BINARY' transfer.
Final Response: 226 Transfer complete.
Download complete!
```

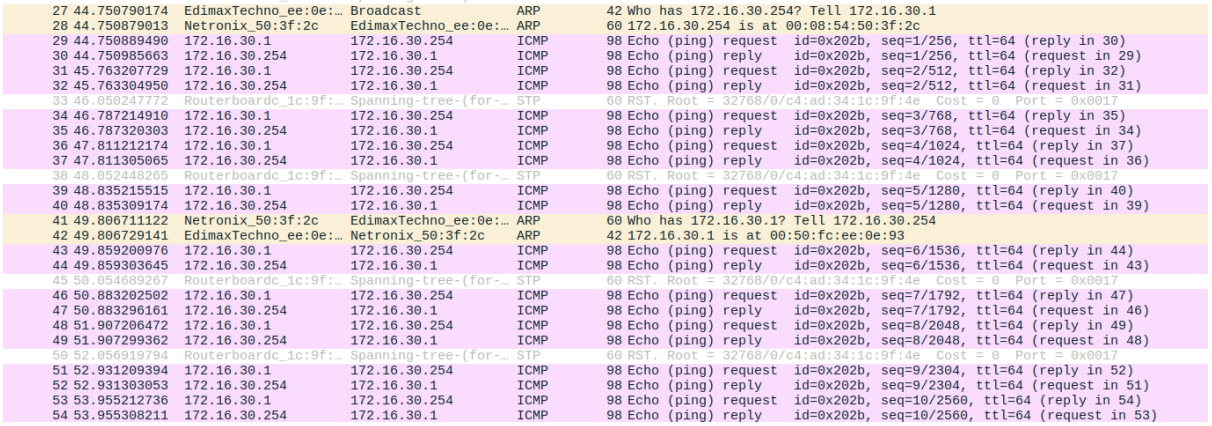
Figura 2: Resultado URL2

```
roger@RogerDesktop:~/universidade/RCOM2$ ./download ftp://anonymous:anonymous@ftp.bit.nl/speedtest/100mb.bin
FTP server: ftp.bit.nl
Server: 220 Welcome to ftp.bit.nl
USER Response: 331 Anonymous login ok, send your complete email address as your password
PASS Response: 230-
230-
230-Dear "Security Researchers",
230-
230-Welcome to our *PUBLIC* OPEN SOURCE SOFTWARE MIRROR SERVER.
230-Please DO NOT report this under our responsible disclosure policy.
230-This is a PUBLIC service, with OPEN SOURCE SOFTWARE, and NOT a security threat to our company.
230-There is NO SENSITIVE INFORMATION on this server.
230-
230-Thanks.
230-
230-
230 Anonymous access granted, restrictions apply
PASV Response: 227 Entering Passive Mode (213,136,12,213,168,132).
Passive Mode IP: 213.136.12.213, Port: 43140
RETR Response: 150 Opening ASCII mode data connection for speedtest/100mb.bin (104857600 bytes)
Final Response: 226 Transfer complete
Download complete!
```

Figura 3: Resultado URL 3

ANEXO 2 - Resultados das capturas da parte 2

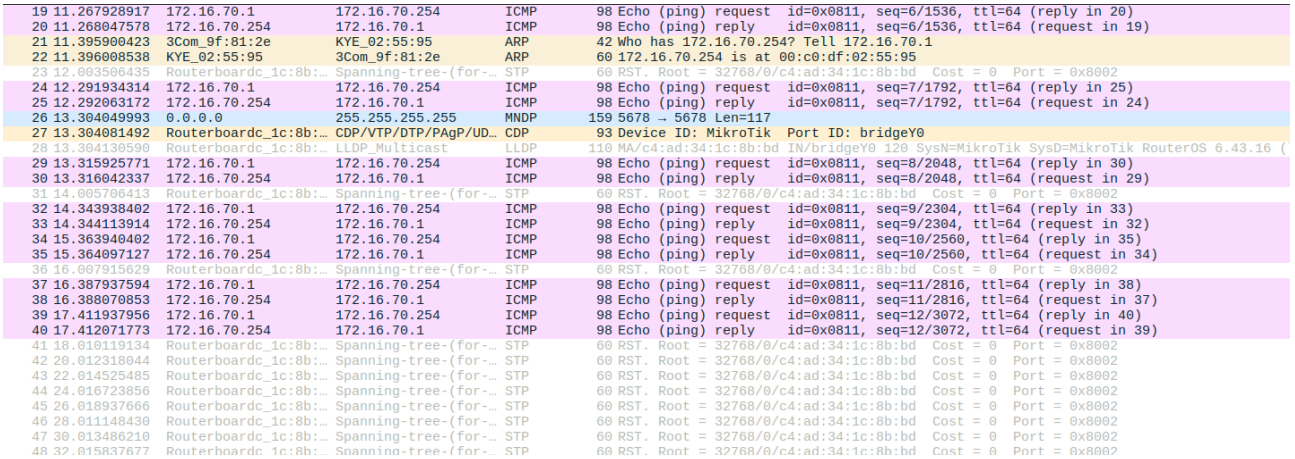
2.1. Experiência 1:



27	44.750790174	EdimaxTechno_ee:0e:…	Broadcast	ARP	42	Who has 172.16.30.254? Tell 172.16.30.1
28	44.750879013	Netronix_50:3f:2c	EdimaxTechno_ee:0e:…	ARP	60	172.16.30.254 is at 00:08:54:50:3f:2c
29	44.750889490	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=1/256, ttl=64 (reply in 30)
30	44.750985663	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=1/256, ttl=64 (request in 29)
31	45.763207729	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=2/512, ttl=64 (reply in 32)
32	45.763304950	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=2/512, ttl=64 (request in 31)
33	46.050247772	Routerboardc_1c:9f:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:9f:4e Cost = 0 Port = 0x0017
34	46.787214910	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=3/768, ttl=64 (reply in 35)
35	46.787320303	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=3/768, ttl=64 (request in 34)
36	47.811212174	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=4/1024, ttl=64 (reply in 37)
37	47.811305065	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=4/1024, ttl=64 (request in 36)
38	48.052448265	Routerboardc_1c:9f:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:9f:4e Cost = 0 Port = 0x0017
39	48.835215515	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=5/1280, ttl=64 (reply in 40)
40	48.835309174	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=5/1280, ttl=64 (request in 39)
41	49.806711122	Netronix_50:3f:2c	EdimaxTechno_ee:0e:…	ARP	60	Who has 172.16.30.1? Tell 172.16.30.254
42	49.806729141	EdimaxTechno_ee:0e:…	Netronix_50:3f:2c	ARP	42	172.16.30.1 is at 00:50:fc:ee:0e:93
43	49.859200976	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=6/1536, ttl=64 (reply in 44)
44	49.859303645	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=6/1536, ttl=64 (request in 43)
45	50.054689267	Routerboardc_1c:9f:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:9f:4e Cost = 0 Port = 0x0017
46	50.883202502	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=7/1792, ttl=64 (reply in 47)
47	50.883296161	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=7/1792, ttl=64 (request in 46)
48	51.907206472	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=8/2048, ttl=64 (reply in 49)
49	51.907299362	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=8/2048, ttl=64 (request in 48)
50	52.056919794	Routerboardc_1c:9f:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:9f:4e Cost = 0 Port = 0x0017
51	52.931209394	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=9/2304, ttl=64 (reply in 52)
52	52.931303053	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=9/2304, ttl=64 (request in 51)
53	53.955212736	172.16.30.1	172.16.30.254	ICMP	98	Echo (ping) request id=0x202b, seq=10/2560, ttl=64 (reply in 54)
54	53.955308211	172.16.30.254	172.16.30.1	ICMP	98	Echo (ping) reply id=0x202b, seq=10/2560, ttl=64 (request in 53)

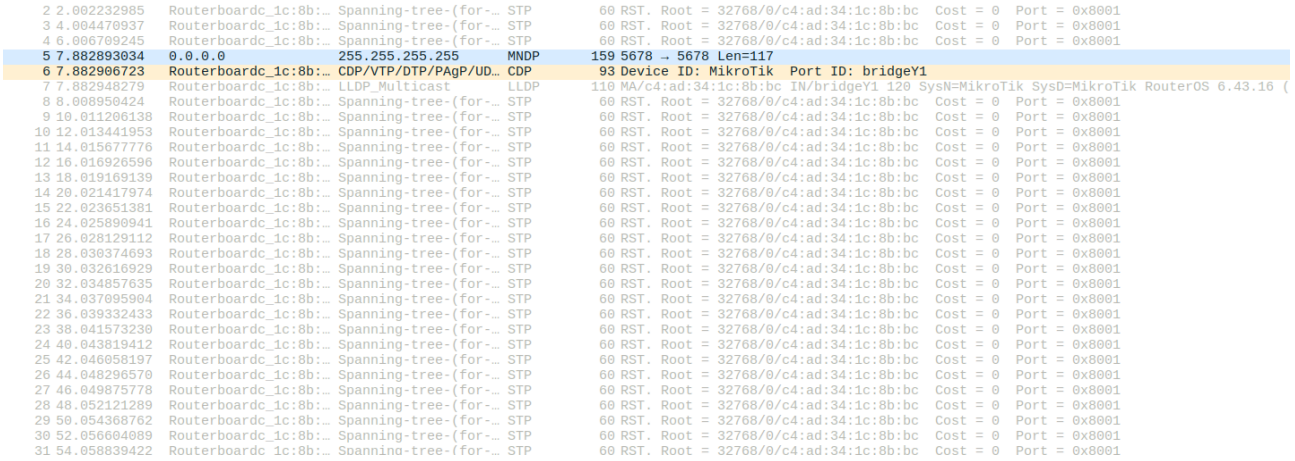
Figura 4: Captura no TUXY3.

2.2. Experiência 2:



19	11.267928917	172.16.70.1	172.16.70.254	ICMP	98	Echo (ping) request id=0x0811, seq=6/1536, ttl=64 (reply in 20)
20	11.268047578	172.16.70.254	172.16.70.1	ICMP	98	Echo (ping) reply id=0x0811, seq=6/1536, ttl=64 (request in 19)
21	11.395900423	3Com_9f:81:2e	KYE_02:55:95	ARP	42	Who has 172.16.70.254? Tell 172.16.70.1
22	11.396008538	KYE_02:55:95	3Com_9f:81:2e	ARP	60	172.16.70.254 is at 00:c0:df:02:55:95
23	12.003506435	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
24	12.291934314	172.16.70.1	172.16.70.254	ICMP	98	Echo (ping) request id=0x0811, seq=7/1792, ttl=64 (reply in 25)
25	12.292063172	172.16.70.254	172.16.70.1	ICMP	98	Echo (ping) reply id=0x0811, seq=7/1792, ttl=64 (request in 24)
26	13.304049993	0.0.0.0	255.255.255.255	MNDP	159	5678 → 5678 Len=117
27	13.304081492	Routerboardc_1c:8b:…	CDP/VTP/DTP/PAgP/UD…	CDP	93	Device ID: MikroTik Port ID: bridgeY0
28	13.304130590	Routerboardc_1c:8b:…	LLDP_Multicast	LLDP	110	MA/c4:ad:34:1c:8b:bd IN/bridgeY0 120 SysN=MikroTik SysD=MikroTik RouterOS 6.43.16 (
29	13.315925771	172.16.70.1	172.16.70.254	ICMP	98	Echo (ping) request id=0x0811, seq=8/2048, ttl=64 (reply in 30)
30	13.316042337	172.16.70.254	172.16.70.1	ICMP	98	Echo (ping) reply id=0x0811, seq=8/2048, ttl=64 (request in 29)
31	14.005706413	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
32	14.343938402	172.16.70.1	172.16.70.254	ICMP	98	Echo (ping) request id=0x0811, seq=9/2304, ttl=64 (reply in 33)
33	14.344113914	172.16.70.254	172.16.70.1	ICMP	98	Echo (ping) reply id=0x0811, seq=9/2304, ttl=64 (request in 32)
34	15.363940402	172.16.70.1	172.16.70.254	ICMP	98	Echo (ping) request id=0x0811, seq=10/2560, ttl=64 (reply in 35)
35	15.364097127	172.16.70.254	172.16.70.1	ICMP	98	Echo (ping) reply id=0x0811, seq=10/2560, ttl=64 (request in 34)
36	16.007915629	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
37	16.387937594	172.16.70.1	172.16.70.254	ICMP	98	Echo (ping) request id=0x0811, seq=11/2816, ttl=64 (reply in 38)
38	16.388070853	172.16.70.254	172.16.70.1	ICMP	98	Echo (ping) reply id=0x0811, seq=11/2816, ttl=64 (request in 37)
39	17.411937956	172.16.70.1	172.16.70.254	ICMP	98	Echo (ping) request id=0x0811, seq=12/3072, ttl=64 (reply in 40)
40	17.412071773	172.16.70.254	172.16.70.1	ICMP	98	Echo (ping) reply id=0x0811, seq=12/3072, ttl=64 (request in 39)
41	18.010119134	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
42	20.012318044	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
43	22.014525485	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
44	24.016723856	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
45	26.018937666	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
46	28.011148430	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
47	30.013486210	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
48	32.015837677	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002

Figura 5: Captura no TUX3 (resultado do ping TUXY4 e TUXY2)



2	2.002232985	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
3	4.004470937	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
4	6.006709245	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
5	7.882893034	0.0.0.0	255.255.255.255	MNDP	159	5678 → 5678 Len=117
6	7.882906723	Routerboardc_1c:8b:…	CDP/VTP/DTP/PAgP/UD…	CDP	93	Device ID: MikroTik Port ID: bridgeY1
7	7.882948279	Routerboardc_1c:8b:…	LLDP_Multicast	LLDP	110	MA/c4:ad:34:1c:8b:bc IN/bridgeY1 120 SysN=MikroTik SysD=MikroTik RouterOS 6.43.16 (
8	8.008950424	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
9	10.011206138	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
10	12.013441953	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
11	14.015677776	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
12	16.016926596	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
13	18.019169139	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
14	20.021417974	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
15	22.023651381	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
16	24.025890941	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
17	26.028129112	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
18	28.030374693	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
19	30.032616929	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
20	32.034857635	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
21	34.037095904	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
22	36.039332433	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
23	38.041573230	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
24	40.043819412	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
25	42.046058197	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
26	44.048296570	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
27	46.049875778	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
28	48.052121289	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
29	50.054368762	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
30	52.056604089	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001
31	54.058839422	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/c4:ad:34:1c:8b:bc Cost = 0 Port = 0x8001

Figura 6: Captura no TUXY2 (resultado do ping broadcast no TUXY3)

```
25 42.858968653  Routerboardc_1c:8b:… LLDP_Multicast       LLDP      110 MA/c4:ad:34:1c:8b:bd IN/bridgeY0 120 SysN=MikroTik SysD=MikroTik RouterOS 6.43.16 (
26 44.048731949  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
27 46.050975925  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
28 48.053214160  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
29 50.055456502  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
30 52.057706441  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
31 54.059955108  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
32 54.524988743  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=1/256, ttl=64 (no response found!)
33 55.529657756  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=2/512, ttl=64 (no response found!)
34 56.062201175  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
35 56.553658083  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=3/768, ttl=64 (no response found!)
36 57.577657865  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=4/1024, ttl=64 (no response found!)
37 58.064443314  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
38 58.601656264  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=5/1280, ttl=64 (no response found!)
39 59.625654607  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=6/1536, ttl=64 (no response found!)
40 60.066678243  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
41 60.649660158  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=7/1792, ttl=64 (no response found!)
42 61.673657830  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=8/2048, ttl=64 (no response found!)
43 62.068783246  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
44 62.697648113  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=9/2304, ttl=64 (no response found!)
45 63.721657616  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=10/2560, ttl=64 (no response found!)
46 64.071022052  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
47 64.745657355  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=11/2816, ttl=64 (no response found!)
48 65.769647888  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=12/3072, ttl=64 (no response found!)
49 66.073252039  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
50 66.793658620  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=13/3328, ttl=64 (no response found!)
51 67.817660774  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=14/3584, ttl=64 (no response found!)
52 68.075479913  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
53 70.077714404  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
54 72.079987080  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8002
```

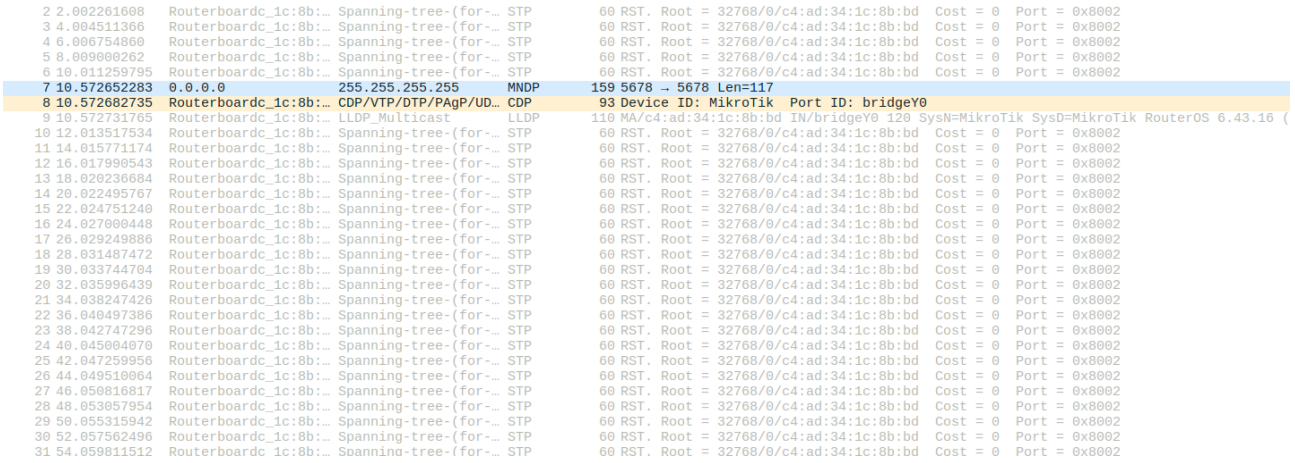
Figura 7: Captura no TUXY3 (resultado do ping broadcast no TUXY3)

```
20 32.034739615  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
21 34.036952180  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
22 36.039169560  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
23 36.504320014  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=1/256, ttl=64 (no response found!)
24 37.508967405  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=2/512, ttl=64 (no response found!)
25 38.041379675  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
26 38.532950339  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=3/768, ttl=64 (no response found!)
27 39.556934529  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=4/1024, ttl=64 (no response found!)
28 40.043587690  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
29 40.580912433  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=5/1280, ttl=64 (no response found!)
30 41.604895924  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=6/1536, ttl=64 (no response found!)
31 42.045788091  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
32 42.628883743  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=7/1792, ttl=64 (no response found!)
33 43.652868769  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=8/2048, ttl=64 (no response found!)
34 44.047859771  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
35 44.676835006  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=9/2304, ttl=64 (no response found!)
36 45.700831065  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=10/2560, ttl=64 (no response found!)
37 46.050061353  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
38 46.724813992  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=11/2816, ttl=64 (no response found!)
39 47.748785675  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=12/3072, ttl=64 (no response found!)
40 48.052260976  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
41 48.772781592  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=13/3328, ttl=64 (no response found!)
42 49.796761933  172.16.70.1          172.16.70.255        ICMP       98 Echo (ping) request  id=0x0a81, seq=14/3584, ttl=64 (no response found!)
43 50.054457315  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
44 52.056664406  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
45 54.058895310  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
46 56.061146605  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
47 58.063359694  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
48 60.065576343  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
49 62.067779090  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd  Cost = 0  Port = 0x8001
```

Figura 8: Captura no TUXY4 (resultado do ping broadcast no TUXY3)

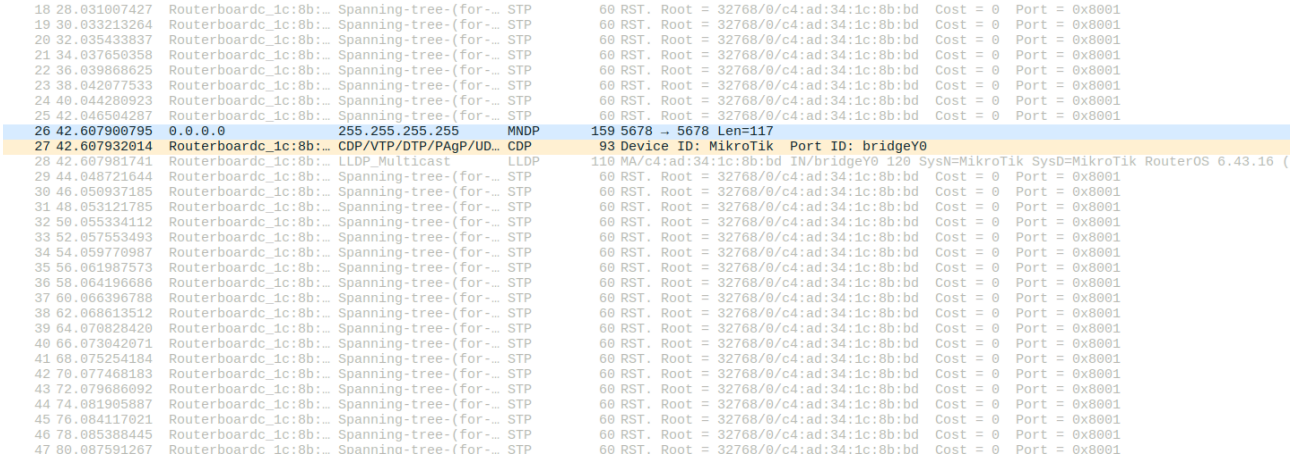
```
22 32.034243033  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
23 34.036387846  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
24 36.038541666  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
25 38.040682214  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
26 40.042809906  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
27 42.044957007  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
28 44.047098862  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
29 45.316907315  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=1/256, ttl=64 (no response found!)
30 46.049255025  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
31 46.332132334  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=2/512, ttl=64 (no response found!)
32 47.356135487  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=3/768, ttl=64 (no response found!)
33 48.051393302  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
34 48.380134469  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=4/1024, ttl=64 (no response found!)
35 49.404137588  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=5/1280, ttl=64 (no response found!)
36 50.053542811  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
37 50.428135559  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=6/1536, ttl=64 (no response found!)
38 51.452137528  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=7/1792, ttl=64 (no response found!)
39 52.055692307  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
40 52.476132951  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=8/2048, ttl=64 (no response found!)
41 53.500139706  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=9/2304, ttl=64 (no response found!)
42 54.057840743  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
43 54.524144385  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=10/2560, ttl=64 (no response found!)
44 55.548134345  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=11/2816, ttl=64 (no response found!)
45 56.059993563  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
46 56.576133844  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=12/3072, ttl=64 (no response found!)
47 57.596132544  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=13/3328, ttl=64 (no response found!)
48 58.062136659  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
49 58.620130587  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=14/3584, ttl=64 (no response found!)
50 59.644134373  172.16.71.1          172.16.71.255        ICMP       98 Echo (ping) request  id=0x10ea, seq=15/3840, ttl=64 (no response found!)
51 60.063337051  Routerboardc_1c:8b:… Spanning-tree-(for-… STP        60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8001
```

Figura 9: Captura no TUX2 (resultado do ping broadcast no TUXY2)



2	2.002261608	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
3	4.004511366	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
4	6.006754860	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
5	8.009000262	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
6	10.011259795	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
7	10.572652283	0.0.0.0	255.255.255.255	MNDP	159 5678 → 5678 Len=117
8	10.572682735	Routerboardc_1c:8b:…	CDP/VTP/DTP/PAgP/UD…	CDP	93 Device ID: MikroTik Port ID: bridgeY0
9	10.572731765	Routerboardc_1c:8b:…	LLDP_Multicast	LLDP	110 MA/c4:ad:34:1c:8b:bd IN/bridgeY0 120 SysN=MikroTik SysD=MikroTik RouterOS 6.43.16 (
10	12.013517534	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
11	14.015771174	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
12	16.017990543	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
13	18.020236684	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
14	20.022495767	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
15	22.024751240	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
16	24.027000448	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
17	26.029249886	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
18	28.031487472	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
19	30.033744704	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
20	32.035996439	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
21	34.038247426	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
22	36.040497386	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
23	38.042747296	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
24	40.045004070	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
25	42.047259956	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
26	44.049510064	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
27	46.050816817	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
28	48.053057954	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
29	50.055315942	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
30	52.057562496	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002
31	54.059811512	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8002

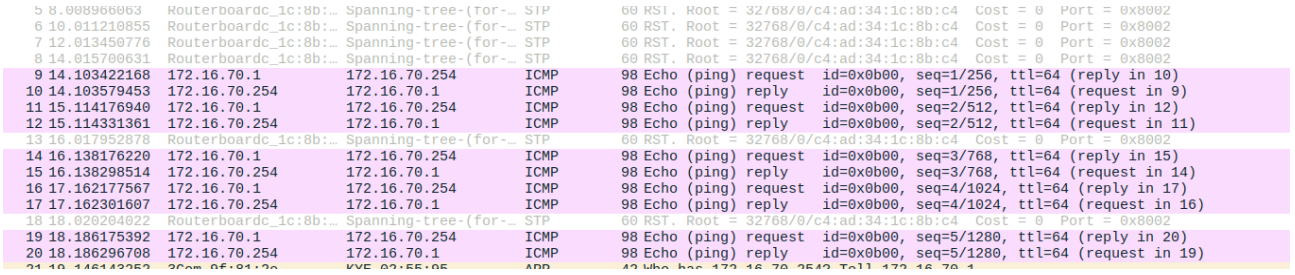
Figura 10: Captura no TUXY3 (resultado do ping broadcast no TUXY2)



18	28.031007427	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
19	30.033213264	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
20	32.035433837	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
21	34.037650358	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
22	36.039868625	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
23	38.042077533	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
24	40.044280923	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
25	42.046504287	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
26	42.607900795	0.0.0.0	255.255.255.255	MNDP	159 5678 → 5678 Len=117
27	42.607932014	Routerboardc_1c:8b:…	CDP/VTP/DTP/PAgP/UD…	CDP	93 Device ID: MikroTik Port ID: bridgeY0
28	42.607981741	Routerboardc_1c:8b:…	LLDP_Multicast	LLDP	110 MA/c4:ad:34:1c:8b:bd IN/bridgeY0 120 SysN=MikroTik SysD=MikroTik RouterOS 6.43.16 (
29	44.048721644	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
30	46.050937185	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
31	48.053121785	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
32	50.055334112	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
33	52.057553493	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
34	54.059770987	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
35	56.061987573	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
36	58.064196686	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
37	60.066396788	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
38	62.068613512	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
39	64.070828420	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
40	66.073042071	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
41	68.075254184	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
42	70.077468183	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
43	72.079686092	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
44	74.081905887	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
45	76.084117021	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
46	78.085388445	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001
47	80.087591267	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:bd Cost = 0 Port = 0x8001

Figura 11: Captura no TUXY4 (resultado do ping broadcast no TUXY2)

2.3. Experiência 3:



5	8.008966063	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
6	10.011210855	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
7	12.013450776	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
8	14.015700631	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
9	14.103422168	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x0b00, seq=1/256, ttl=64 (reply in 10)
10	14.103579453	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x0b00, seq=1/256, ttl=64 (request in 9)
11	15.114176940	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x0b00, seq=2/512, ttl=64 (reply in 12)
12	15.114331361	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x0b00, seq=2/512, ttl=64 (request in 11)
13	16.017952878	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
14	16.138176220	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x0b00, seq=3/768, ttl=64 (reply in 15)
15	16.138298514	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x0b00, seq=3/768, ttl=64 (request in 14)
16	17.162177567	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x0b00, seq=4/1024, ttl=64 (reply in 17)
17	17.162301607	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x0b00, seq=4/1024, ttl=64 (request in 16)
18	18.020204022	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
19	18.186175392	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x0b00, seq=5/1280, ttl=64 (reply in 20)
20	18.186296708	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x0b00, seq=5/1280, ttl=64 (request in 19)

Figura 12: Captura no TUXY3 (resultado do ping 172.16.Y0.254)

```
 26 22.024688964  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
 27 24.026937916  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
 28 25.559312980  172.16.70.1           172.16.71.253         ICMP    98 Echo (ping) request  id=0x0b07, seq=1/256, ttl=64 (reply in 29)
 29 25.559468798  172.16.71.253         172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b07, seq=1/256, ttl=64 (request in 28)
 30 26.029190444  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
 31 26.570177080  172.16.70.1           172.16.71.253         ICMP    98 Echo (ping) request  id=0x0b07, seq=2/512, ttl=64 (reply in 32)
 32 26.570302168  172.16.71.253         172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b07, seq=2/512, ttl=64 (request in 31)
 33 27.594177718  172.16.70.1           172.16.71.253         ICMP    98 Echo (ping) request  id=0x0b07, seq=3/768, ttl=64 (reply in 34)
 34 27.594302805  172.16.71.253         172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b07, seq=3/768, ttl=64 (request in 33)
 35 28.031444380  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
 36 28.618178325  172.16.70.1           172.16.71.253         ICMP    98 Echo (ping) request  id=0x0b07, seq=4/1024, ttl=64 (reply in 37)
 37 28.618302784  172.16.71.253         172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b07, seq=4/1024, ttl=64 (request in 36)
 38 29.642176598  172.16.70.1           172.16.71.253         ICMP    98 Echo (ping) request  id=0x0b07, seq=5/1280, ttl=64 (reply in 39)
 39 29.642301196  172.16.71.253         172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b07, seq=5/1280, ttl=64 (request in 38)
 40 30.033691065  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
 41 32.035932174  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
```

Figura 13: Captura no TUXY3 (resultado do ping 172.16.Y1.253)

```
 44 38.042674910  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
 45 39.863658718  172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0b11, seq=1/256, ttl=64 (reply in 46)
 46 39.864099423  172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b11, seq=1/256, ttl=63 (request in 45)
 47 40.044924161  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
 48 40.874186860  172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0b11, seq=2/512, ttl=64 (reply in 49)
 49 40.874438432  172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b11, seq=2/512, ttl=63 (request in 48)
 50 41.898176731  172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0b11, seq=3/768, ttl=64 (reply in 51)
 51 41.898414754  172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b11, seq=3/768, ttl=63 (request in 50)
 52 42.047176214  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
 53 42.922184800  172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0b11, seq=4/1024, ttl=64 (reply in 54)
 54 42.922420937  172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b11, seq=4/1024, ttl=63 (request in 53)
 55 43.946180476  172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0b11, seq=5/1280, ttl=64 (reply in 56)
 56 43.946418638  172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b11, seq=5/1280, ttl=63 (request in 55)
 57 44.048676701  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
 58 44.970179612  172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0b11, seq=6/1536, ttl=64 (reply in 59)
 59 44.970447318  172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0b11, seq=6/1536, ttl=63 (request in 58)
 60 46.050921366  Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8002
```

Figura 14: Captura no TUXY3 (resultado do ping 172.16.Y1.1)

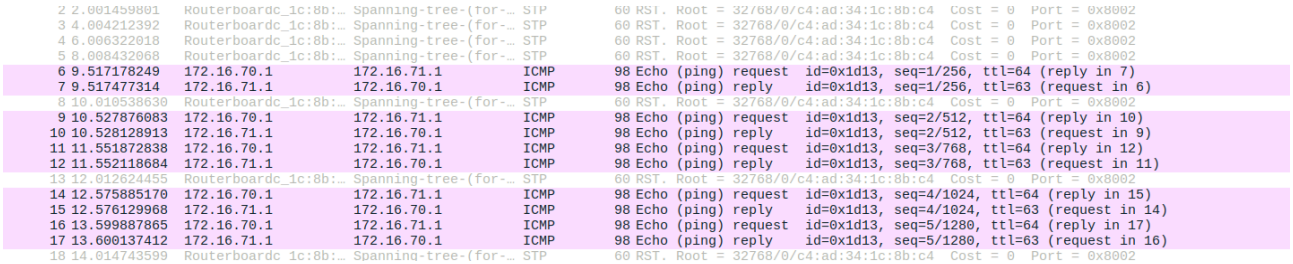
```
316 564.844461031 3Com_9f:81:2e         Broadcast             ARP     60 Who has 172.16.70.254? Tell 172.16.70.1
317 564.844480866 KYE_02:55:95          3Com_9f:81:2e         ARP     42 172.16.70.254 is at 00:c0:df:02:55:95
318 564.844588492 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=1/256, ttl=64 (reply in 319)
319 564.844874282 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=1/256, ttl=63 (request in 318)
320 565.867067258 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=2/512, ttl=64 (reply in 321)
321 565.867243817 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=2/512, ttl=63 (request in 320)
322 566.622103076 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8001
323 566.891048380 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=3/768, ttl=64 (reply in 324)
324 566.891194768 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=3/768, ttl=63 (request in 323)
325 567.915045496 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=4/1024, ttl=64 (reply in 326)
326 567.915189090 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=4/1024, ttl=63 (request in 325)
327 568.624361325 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8001
328 568.939021660 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=5/1280, ttl=64 (reply in 329)
329 568.939177475 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=5/1280, ttl=63 (request in 328)
330 569.963012140 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=6/1536, ttl=64 (reply in 331)
331 569.963167677 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=6/1536, ttl=63 (request in 330)
332 570.028456426 KYE_02:55:95          3Com_9f:81:2e         ARP     42 Who has 172.16.70.1? Tell 172.16.70.254
333 570.028586122 3Com_9f:81:2e         KYE_02:55:95          ARP     60 172.16.70.1 is at 00:01:02:9f:81:2e
334 570.626616779 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8001
335 570.987014355 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=7/1792, ttl=64 (reply in 336)
336 570.987197548 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=7/1792, ttl=63 (request in 335)
337 572.628860710 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8001
338 574.631113720 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4  Cost = 0  Port = 0x8001
```

Figura 15: Captura no eth1 TUXY4 (resultado do ping TUXY2 no TUXY3)

```
303 548.601859720 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8002
304 548.826658627 3Com_a0:ad:91         Broadcast             ARP     42 Who has 172.16.71.1? Tell 172.16.71.253
305 548.826781129 Netronix_b5:8c:8f     3Com_a0:ad:91         ARP     60 172.16.71.1 is at 00:e0:7d:b5:8c:8f
306 548.826795027 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=1/256, ttl=63 (reply in 307)
307 548.826916481 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=1/256, ttl=64 (request in 306)
308 549.849144378 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=2/512, ttl=63 (reply in 309)
309 549.849280918 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=2/512, ttl=64 (request in 308)
310 550.604112311 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8002
311 550.873121310 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=3/768, ttl=63 (reply in 312)
312 550.873234243 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=3/768, ttl=64 (request in 311)
313 551.897122127 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=4/1024, ttl=63 (reply in 314)
314 551.897235479 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=4/1024, ttl=64 (request in 313)
315 552.606368044 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8002
316 552.921099198 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=5/1280, ttl=63 (reply in 317)
317 552.921218068 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=5/1280, ttl=64 (request in 316)
318 553.945087584 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=6/1536, ttl=63 (reply in 319)
319 553.945201146 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=6/1536, ttl=64 (request in 318)
320 554.067286154 Netronix_b5:8c:8f     3Com_a0:ad:91         ARP     60 Who has 172.16.71.253? Tell 172.16.71.1
321 554.067304522 3Com_a0:ad:91         Netronix_b5:8c:8f     ARP     42 172.16.71.253 is at 00:01:02:a0:ad:91
322 554.608628737 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8002
323 554.969090706 172.16.70.1           172.16.71.1           ICMP    98 Echo (ping) request  id=0x0ecb, seq=7/1792, ttl=63 (reply in 324)
324 554.969227456 172.16.71.1           172.16.70.1           ICMP    98 Echo (ping) reply    id=0x0ecb, seq=7/1792, ttl=64 (request in 323)
325 556.610872737 Routerboardc_1c:8b:…  Spanning-tree-(for-…  STP     60 RST. Root = 32768/0/c4:ad:34:1c:8b:bc  Cost = 0  Port = 0x8002
```

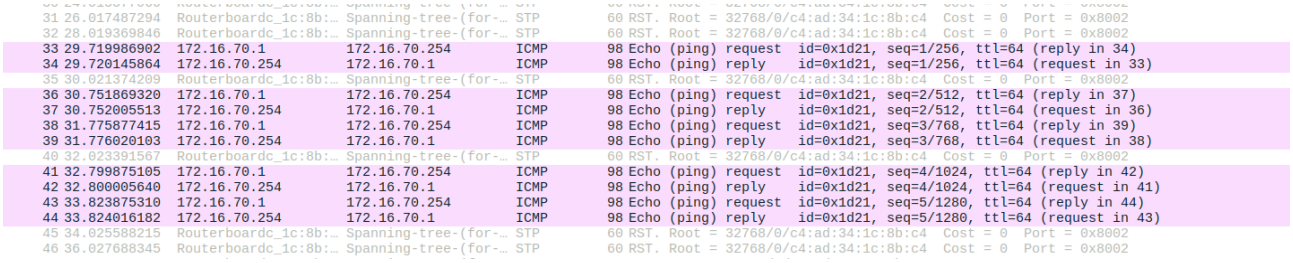
Figura 16: Captura no eth2 TUXY4 (resultado do ping TUXY2 no TUXY3)

2.4. Experiência 4:



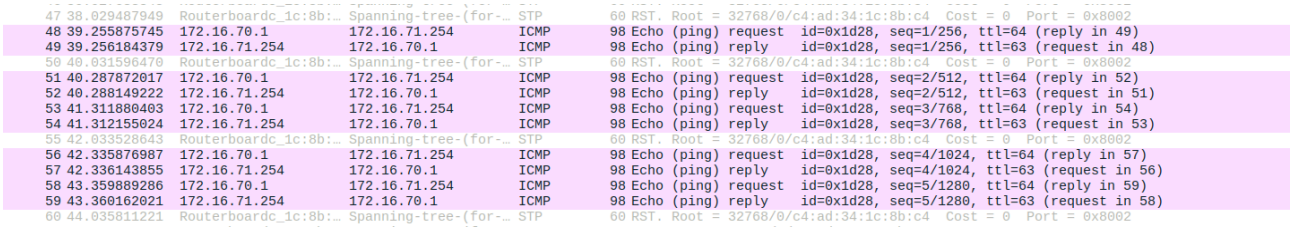
2	2.001459801	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
3	4.004212392	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
4	6.006322018	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
5	8.008432068	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
6	9.517178249	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) request id=0x1d13, seq=1/256, ttl=64 (reply in 7)
7	9.517477314	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d13, seq=1/256, ttl=63 (request in 6)
8	10.010538630	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
9	10.527876083	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) request id=0x1d13, seq=2/512, ttl=64 (reply in 10)
10	10.528128913	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d13, seq=2/512, ttl=63 (request in 9)
11	11.551872838	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) request id=0x1d13, seq=3/768, ttl=64 (reply in 12)
12	11.552118684	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d13, seq=3/768, ttl=63 (request in 11)
13	12.012624455	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
14	12.575885170	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) request id=0x1d13, seq=4/1024, ttl=64 (reply in 15)
15	12.576129968	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d13, seq=4/1024, ttl=63 (request in 14)
16	13.599887865	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) request id=0x1d13, seq=5/1280, ttl=64 (reply in 17)
17	13.600137412	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d13, seq=5/1280, ttl=63 (request in 16)
18	14.014743599	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002

Figura 17 : Captura no TUXY3 (ping TUXY2)



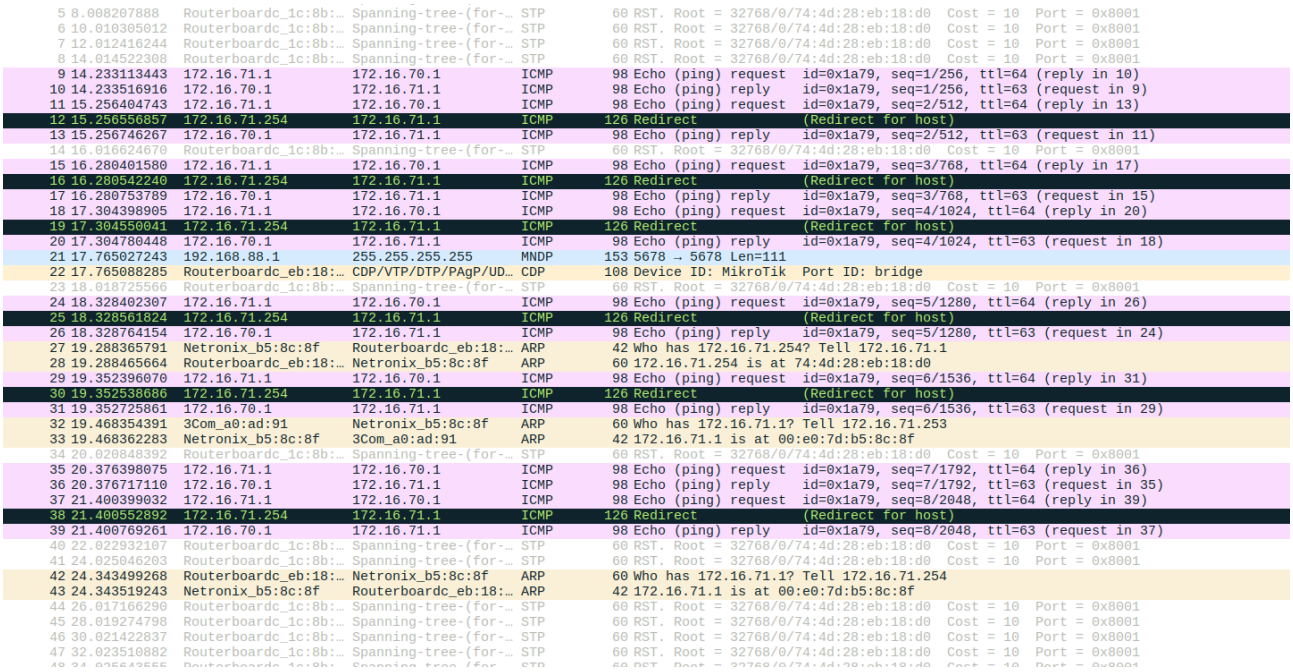
31	26.017487294	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
32	28.019369846	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
33	29.719986902	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x1d21, seq=1/256, ttl=64 (reply in 34)
34	29.720145864	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d21, seq=1/256, ttl=64 (request in 33)
35	30.021374209	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
36	30.751869320	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x1d21, seq=2/512, ttl=64 (reply in 37)
37	30.752005513	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d21, seq=2/512, ttl=64 (request in 36)
38	31.775877415	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x1d21, seq=3/768, ttl=64 (reply in 39)
39	31.776020103	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d21, seq=3/768, ttl=64 (request in 38)
40	32.023391567	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
41	32.799875105	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x1d21, seq=4/1024, ttl=64 (reply in 42)
42	32.800005640	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d21, seq=4/1024, ttl=64 (request in 41)
43	33.823875310	172.16.70.1	172.16.70.254	ICMP	98 Echo (ping) request id=0x1d21, seq=5/1280, ttl=64 (reply in 44)
44	33.824016182	172.16.70.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d21, seq=5/1280, ttl=64 (request in 43)
45	34.025588215	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
46	36.027688345	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002

Figura 18 : Captura no TUXY3 (ping TUXY4)



47	38.029487949	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
48	39.255875745	172.16.70.1	172.16.71.254	ICMP	98 Echo (ping) request id=0x1d28, seq=1/256, ttl=64 (reply in 49)
49	39.256184379	172.16.71.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d28, seq=1/256, ttl=63 (request in 48)
50	40.031596470	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
51	40.287872017	172.16.70.1	172.16.71.254	ICMP	98 Echo (ping) request id=0x1d28, seq=2/512, ttl=64 (reply in 52)
52	40.288149222	172.16.71.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d28, seq=2/512, ttl=63 (request in 51)
53	41.311880403	172.16.70.1	172.16.71.254	ICMP	98 Echo (ping) request id=0x1d28, seq=3/768, ttl=64 (reply in 54)
54	41.312155024	172.16.71.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d28, seq=3/768, ttl=63 (request in 53)
55	42.033528643	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002
56	42.335876987	172.16.70.1	172.16.71.254	ICMP	98 Echo (ping) request id=0x1d28, seq=4/1024, ttl=64 (reply in 57)
57	42.336143855	172.16.71.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d28, seq=4/1024, ttl=63 (request in 56)
58	43.359889286	172.16.70.1	172.16.71.254	ICMP	98 Echo (ping) request id=0x1d28, seq=5/1280, ttl=64 (reply in 59)
59	43.360162021	172.16.71.254	172.16.70.1	ICMP	98 Echo (ping) reply id=0x1d28, seq=5/1280, ttl=63 (request in 58)
60	44.035811221	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/c4:ad:34:1c:8b:c4 Cost = 0 Port = 0x8002

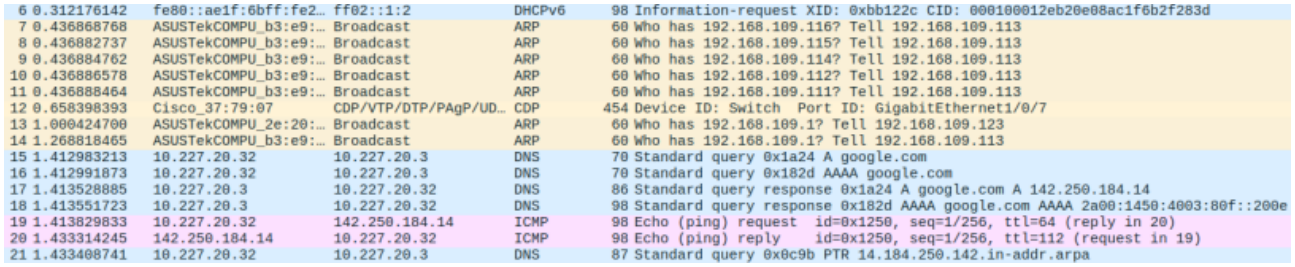
Figura 19 : Captura no TUXY3 (ping router)



5	8.008207888	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
6	10.010305012	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
7	12.012416244	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
8	14.014522308	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
9	14.233113443	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) request id=0x1a79, seq=1/256, ttl=64 (reply in 10)
10	14.233516916	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) reply id=0x1a79, seq=1/256, ttl=63 (request in 9)
11	15.256404743	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) request id=0x1a79, seq=2/512, ttl=64 (reply in 13)
12	15.256556857	172.16.71.254	172.16.71.1	ICMP	126 Redirect (Redirect for host)
13	15.256746267	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) reply id=0x1a79, seq=2/512, ttl=63 (request in 11)
14	16.016624670	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
15	16.280401580	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) request id=0x1a79, seq=3/768, ttl=64 (reply in 17)
16	16.280542240	172.16.71.254	172.16.71.1	ICMP	126 Redirect (Redirect for host)
17	16.280753789	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) reply id=0x1a79, seq=3/768, ttl=63 (request in 15)
18	17.304398905	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) request id=0x1a79, seq=4/1024, ttl=64 (reply in 20)
19	17.304550041	172.16.71.254	172.16.71.1	ICMP	126 Redirect (Redirect for host)
20	17.304780448	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) reply id=0x1a79, seq=4/1024, ttl=63 (request in 18)
21	17.765027243	192.168.88.1	255.255.255.255	MNDP	153 5678 → 5678 Len=111
22	17.765088285	Routerboardc_eb:18:…	CDP/VTP/DTP/PAgP/UD…	CDP	108 Device ID: MikroTik Port ID: bridge
23	18.018725566	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
24	18.328402307	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) request id=0x1a79, seq=5/1280, ttl=64 (reply in 26)
25	18.328561824	172.16.71.254	172.16.71.1	ICMP	126 Redirect (Redirect for host)
26	18.328764154	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) reply id=0x1a79, seq=5/1280, ttl=63 (request in 24)
27	19.288365791	Netronix_b5:8c:8f	Routerboardc_eb:18:…	ARP	42 Who has 172.16.71.254? Tell 172.16.71.1
28	19.288465664	Routerboardc_eb:18:…	Netronix_b5:8c:8f	ARP	60 172.16.71.254 is at 74:4d:28:eb:18:d0
29	19.352396070	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) request id=0x1a79, seq=6/1536, ttl=64 (reply in 31)
30	19.352538686	172.16.71.254	172.16.71.1	ICMP	126 Redirect (Redirect for host)
31	19.352725861	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) reply id=0x1a79, seq=6/1536, ttl=63 (request in 29)
32	19.468354391	3Com_a0:ad:91	Netronix_b5:8c:8f	ARP	60 Who has 172.16.71.1? Tell 172.16.71.253
33	19.468362283	Netronix_b5:8c:8f	3Com_a0:ad:91	ARP	42 172.16.71.1 is at 00:e0:7d:b5:8c:8f
34	20.020848392	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
35	20.376398075	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) request id=0x1a79, seq=7/1792, ttl=64 (reply in 36)
36	20.376717110	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) reply id=0x1a79, seq=7/1792, ttl=63 (request in 35)
37	21.400399032	172.16.71.1	172.16.70.1	ICMP	98 Echo (ping) request id=0x1a79, seq=8/2048, ttl=64 (reply in 39)
38	21.400552892	172.16.71.254	172.16.71.1	ICMP	126 Redirect (Redirect for host)
39	21.400769261	172.16.70.1	172.16.71.1	ICMP	98 Echo (ping) reply id=0x1a79, seq=8/2048, ttl=63 (request in 37)
40	22.022932107	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
41	24.025046203	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
42	24.343499268	Routerboardc_eb:18:…	Netronix_b5:8c:8f	ARP	60 Who has 172.16.71.1? Tell 172.16.71.254
43	24.343519243	Netronix_b5:8c:8f	Routerboardc_eb:18:…	ARP	42 172.16.71.1 is at 00:e0:7d:b5:8c:8f
44	26.017166290	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
45	28.019274798	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
46	30.021422837	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001
47	32.023510882	Routerboardc_1c:8b:…	Spanning-tree-(for-…	STP	60 RST. Root = 32768/0/74:4d:28:eb:18:d0 Cost = 10 Port = 0x8001

Figura 20 : Captura no TUXY2 (ping TUXY3 sem ICMP)

2.5. Experiência 5:



6	0.312176142	fe80::ae1f:6bff:fe2…	ff02::1:2	DHCPv6	98	Information-request XID: 0xbb122c CID: 000100012eb20e08ac1f6b2f283d
7	0.436868768	ASUSTekCOMPU_b3:e9:…	Broadcast	ARP	60	Who has 192.168.109.116? Tell 192.168.109.113
8	0.436882737	ASUSTekCOMPU_b3:e9:…	Broadcast	ARP	60	Who has 192.168.109.115? Tell 192.168.109.113
9	0.436884762	ASUSTekCOMPU_b3:e9:…	Broadcast	ARP	60	Who has 192.168.109.114? Tell 192.168.109.113
10	0.436886578	ASUSTekCOMPU_b3:e9:…	Broadcast	ARP	60	Who has 192.168.109.112? Tell 192.168.109.113
11	0.436888464	ASUSTekCOMPU_b3:e9:…	Broadcast	ARP	60	Who has 192.168.109.111? Tell 192.168.109.113
12	0.658398393	Cisco_37:79:07	CDP/VTP/DTP/PAgP/UD…	CDP	454	Device ID: Switch Port ID: GigabitEthernet1/0/7
13	1.000424700	ASUSTekCOMPU_2e:20:…	Broadcast	ARP	60	Who has 192.168.109.1? Tell 192.168.109.123
14	1.268818465	ASUSTekCOMPU_b3:e9:…	Broadcast	ARP	60	Who has 192.168.109.1? Tell 192.168.109.113
15	1.412983213	10.227.20.32	10.227.20.3	DNS	70	Standard query 0x1a24 A google.com
16	1.412991873	10.227.20.32	10.227.20.3	DNS	70	Standard query 0x182d AAAA google.com
17	1.413528885	10.227.20.3	10.227.20.32	DNS	86	Standard query response 0x1a24 A google.com A 142.250.184.14
18	1.413551723	10.227.20.3	10.227.20.32	DNS	98	Standard query response 0x182d AAAA google.com AAAA 2a00:1450:4003:80f::200e
19	1.413829833	10.227.20.32	142.250.184.14	ICMP	98	Echo (ping) request id=0x1250, seq=1/256, ttl=64 (reply in 20)
20	1.433314245	142.250.184.14	10.227.20.32	ICMP	98	Echo (ping) reply id=0x1250, seq=1/256, ttl=112 (request in 19)
21	1.433408741	10.227.20.32	10.227.20.3	DNS	87	Standard query 0x0c9b PTR 14.184.250.142.in-addr.arpa

Figura 21 : Captura do ping www.google.com

ANEXO 3 - Comandos de configuração da parte 2

TUX2, TUX3, TUX4: systemctl restart networking
GTK: system reset-configuration

Experiência 1:

TUX 3:
ifconfig eth1 up
ifconfig eth1 172.16.Y0.1/24

TUX 4:
ifconfig eth1 up
ifconfig eth1 172.16.Y0.254/24

TUX 3:
ping 172.16.Y0.254
arp -d <ip>

Experiência 2:

TUX 2:
ifconfig eth1 up
ifconfig eth1 172.16.Y1.1/24

/interface bridge add name=bridgeY0
/interface bridge add name=bridgeY1
/interface bridge port remove [find interface=ether2]
/interface bridge port remove [find interface=ether3]
/interface bridge port remove [find interface=ether4]
/interface bridge port add bridge=bridgeY0 interface=ether3
/interface bridge port add bridge=bridgeY0 interface=ether4

/interface bridge port add bridge=bridgeY1 interface=ether2

Experiência 3:

TUX4:
ifconfig eth2 up
ifconfig eth2 172.16.Y1.253/24
/interface bridge port remove [find interface=ether10]
/interface bridge port add bridge=bridgeY1 interface=ether10
sysctl net.ipv4.ip_forward=1
sysctl net.ipv4.icmp_echo_ignore_broadcasts=0

TUX 3: route add -net 172.16.Y1.0/24 gw 172.16.Y0.254

TUX 2: route add -net 172.16.Y0.0/24 gw 172.16.Y1.253

Experiência 4:
/interface bridge port remove [find interface=ether9]
/interface bridge port add bridge=bridgeY1 interface=ether9

/ip address add address=172.16.Y1.254/24 interface=ether2
/ip address add address=172.16.1.Y1/24 interface=ether1

TUX 3:
route add -net 172.16.1.0/24 gw 172.16.Y0.254
route add -net 172.16.Y1.0/24 gw 172.16.Y0.254

TUX 4:
route add -net 172.16.1.0/24 gw 172.16.Y1.254/24

TUX 2:
route add -net 172.16.Y0.0/24 gw 172.16.Y1.253
route add -net 172.16.1.0/24 gw 172.16.Y1.254

Router: ip route add dst-address=172.16.30.0/24 gateway=172.16.31.253

TUX 2:
sysctl net.ipv4.conf.eth1.accept_redirects=0
sysctl net.ipv4.conf.all.accept_redirects=0

route del -net 172.16.Y0.0/24 gw 172.16.Y1.253
ping 172.16.Y0.1
route add -net 172.16.Y0.0/24 gw 172.16.Y1.254
ping 172.16.Y0.1
traceroute 172.16.Y0.1
route del -net 172.16.Y0.0/24 gw 172.16.Y1.254
route add -net 172.16.Y0.0/24 gw 172.16.Y1.253

traceroute 172.16.Y0.1

sysctl net.ipv4.conf.eth1.accept_redirects=1
sysctl net.ipv4.conf.all.accept_redirects=1
/ip firewall nat disable 0
TUX 3 : ping 172.16.1.10

/ ip firewall nat enable 0
TUX 3 : ping 172.16.1.10

Experiência 5:

TUX2,TUX3,TUX4: nano /etc/resolv.conf - adicionar nameserver 10.227.20.3
ping www.google.com