

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL

FACULDADE DE ENGENHARIA

LABORATÓRIO DE REDES DE COMPUTADORES

e

REDES E COMUNICAÇÃO II

T3 - Implementação de um Túnel ICMP

Felipe da Silva Angnes

Henrique Correa

Rafael Sperb

Professor: Sérgio Johann Filho

Professora: Cristina Moreira Nunes

Porto Alegre, 20 de Novembro de 2018

Introdução

Este trabalho tem como objetivo desenvolver uma aplicação que Implementa um túnel ICMP para transporte de pacotes TCP sobre IPv4, implementado um Proxy para conexões TCP.

Referencial teórico

Sockets Raw

É um mecanismo que permite o recebimento de pacotes de rede juntamente com seus cabeçalhos. Geralmente o SO entrega somente os dados dos pacotes as aplicações específicas, portanto com Sockets Raw é possível analisar todo o tráfego recebido pela rede.

ICMP

ICMP, sigla para o inglês Internet Control Message Protocol, é um protocolo integrante do Protocolo IP, definido pelo RFC 792, é utilizado para fornecer relatórios de erros à fonte original. Qualquer computador que utilize IP precisa aceitar as mensagens ICMP e alterar o seu comportamento de acordo com o erro relatado. Os gateways devem estar programados para enviar mensagens ICMP quando receberem datagramas que provoquem algum erro.

TCP

O TCP (acrônimo para o inglês Transmission Control Protocol, que significa "Protocolo de Controle de Transmissão") é um dos protocolos sob os quais assenta a Internet. Ele é complementado pelo Protocolo da Internet, sendo normalmente chamado de TCP/IP. A versatilidade e robustez do TCP tornou-o adequado a redes globais, já que este verifica se os dados são enviados de forma correta, na sequência apropriada e sem erros, pela rede.

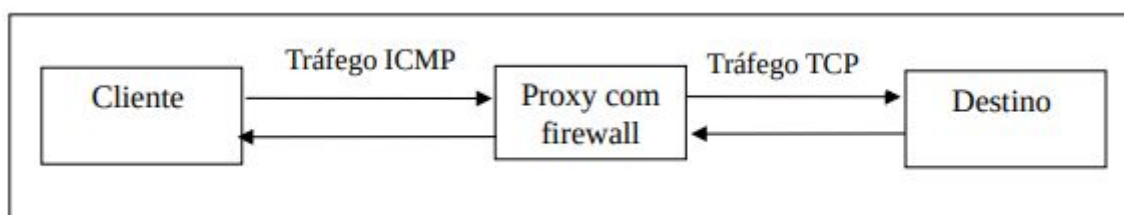
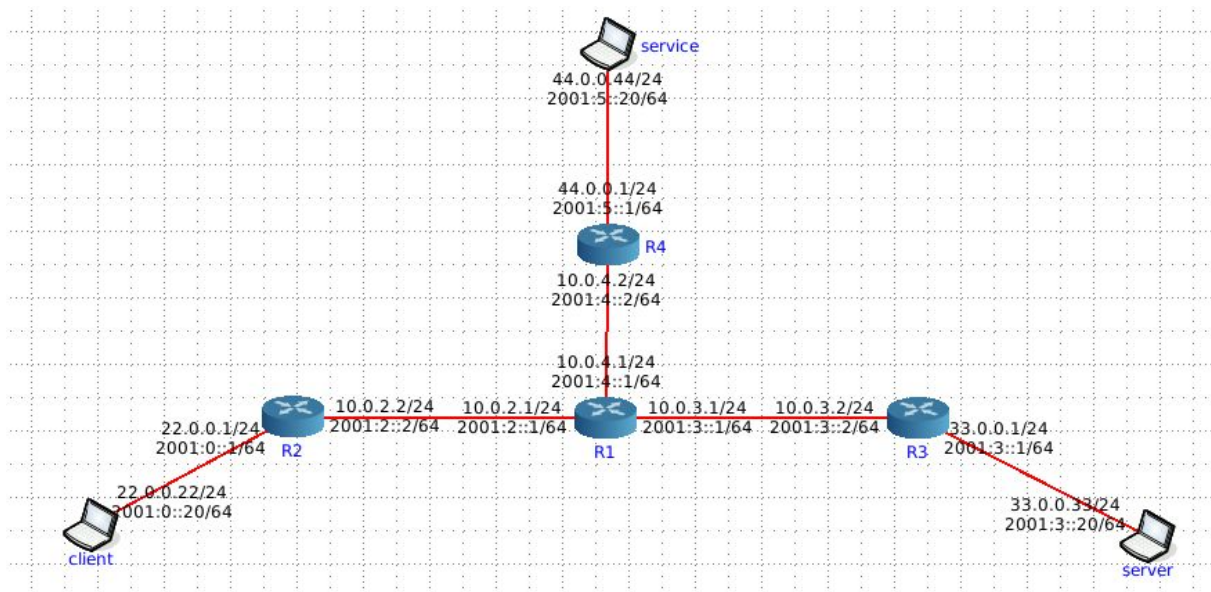
Proxy

Um proxy é um servidor (um sistema de computador ou uma aplicação) que age como um intermediário para requisições de clientes solicitando recursos de outros servidores. Um cliente conecta-se ao servidor proxy, solicitando algum serviço, como um arquivo, conexão, página web ou outros recursos disponíveis de um servidor diferente, e o proxy avalia a solicitação como um meio de simplificar e controlar sua complexidade. Os proxies foram inventados para adicionar estrutura e encapsulamento aos sistemas distribuídos.

Firewall

Um firewall é um dispositivo de uma rede de computadores que tem por objetivo aplicar uma política de segurança a um determinado ponto da rede. O firewall pode ser do tipo filtros de pacotes, proxy de aplicações, etc. Os firewalls são geralmente associados a redes TCP/IP.

Topologia



Desenvolvimento

Implementação da aplicação tem o fluxo da seguinte forma: O cliente configura a uma conexão ao proxy para um determinado IP passado por parâmetros.

```
//check if program has been initialized as a client
else if ((argc == 3) && (strcmp(argv[1], "--client") == 0) && (strlen(argv[2]) <= 15))
{
    isClient = 1;
    //Scan server IP
    sscanf(argv[2], "%hhu.%hhu.%hhu.%hhu", &ip_arg[0], &ip_arg[1], &ip_arg[2], &ip_arg[3]);
}
```

E então é iniciado um túnel entre cliente e proxy.

```
run_tunnel(ip_arg, isServer, isClient);
```

O túnel fica aguardando até que alguma aplicação tente enviar um pacote na rede. Quando a flag sinalizar a presença de dados no túnel, a aplicação inicia a montagem do pacote ICMP.

```
// Preparing ICMP packet to be sent
clean_data_buffer(&packet); //Clean packet buffer

//mount (init) packet
initPacket(&packet, socketInfo.this_mac, gateway_mac, isClient, isServer);
```

E em seguida faz a leitura dos dados enviados pelo cliente no túnel. Neste momento é passado o ponteiro para onde o pacote TCP ficará encapsulado.

```
//Get data from tunnel
int payload_size = tun_read(tun_fd, packet.raw_data + FRAME_HEADER_SIZE /*Pointer to packet data*/,
    PACKET_DATA_BUFFER_SIZE /*packet data available length*/);
```

O envio do pacote ICMP é feito através de socket raw com a mensagem TCP encapsulada no ICMP.

```
// Sending ICMP packet
proxy_sendRawPacket(sock_fd, &packet, FRAME_HEADER_SIZE + payload_size, &socketInfo);
```

O proxy recebe o pacote ICMP pelo socket raw.

```
// Getting ICMP packet
clean_data_buffer(&packet); //Clean packet buffer
int payload_size = proxy_receivePacket(sock_fd, &packet); /* CHANGE TO MY FUNCTION */
```

Verifica se é um pacote ICMP válido e envia apenas o payload TCP no túnel para o serviço.

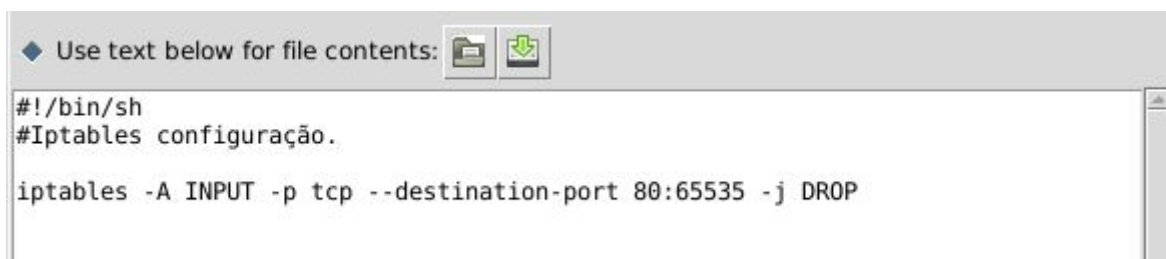
```
if (validateICMPPacket(&packet))
{
    printf("[DEBUG] Read ICMP packet with payload_size: %d\n", payload_size);
    // Writing out to tun device
    tun_write(tun_fd, packet.raw_data + FRAME_HEADER_SIZE, getPacketDataLength(&packet));

    //Overwrite destination address (server)
    memcpy(dest, packet.cooked_data.payload.ip.src, 4);
}
```

O proxy fica aguardando a resposta do serviço. O serviço responde ao proxy enviando um pacote TCP.

O proxy faz o mesmo procedimento como descrito anteriormente, mas agora com o cliente como destino. Ele recebe pelo mesmo túnel a resposta do serviço. Inicia um pacote ICMP Reply, recebe os dados TCP no túnel, encapsula o TCP dentro do ICMP e envia por socket raw para o cliente.

O Firewall é configurado utilizando IpTables entre Cliente e Proxy.



The screenshot shows a terminal window with a title bar that says "Use text below for file contents:". The terminal content is as follows:

```
#!/bin/sh
#Iptables configuração.

iptables -A INPUT -p tcp --destination-port 80:65535 -j DROP
```

Resultados

Conclusão

Com este trabalho podemos verificar o quão fácil pode ser burlar troca de mensagens utilizando o encapsulamento de mensagens TCP.