UNIVERSIDADE FEDERAL DA FRONTEIRA SUL
CAMPUS CHAPECÓ
CURSO DE CIÊNCIA DA COMPUTAÇÃO

WAGNER OSÓRIO BENDER

CLASSIFICAÇÃO DOS PROTOCOLOS DE VPN QUANTO À LARGURA DE BANDA NO SISTEMA ROUTEROS

CHAPECÓ
2022

WAGNER OSÓRIO BENDER

CLASSIFICAÇÃO DOS PROTOCOLOS DE VPN QUANTO À LARGURA DE BANDA NO SISTEMA ROUTEROS

Trabalho de conclusão de curso de graduação apresentado como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação da Universidade Federal da Fronteira Sul.
Orientador: Prof. Dr. Marco Aurélio Spohn

CHAPECÓ
2022

Bibliotecas da Universidade Federal da Fronteira Sul - UFFS

Bender, Wagner Osório
Classificação dos protocolos de VPN quanto à largura de banda no sistema RouterOS / Wagner Osório Bender. -- 2022.
40 f.:il.

Orientador: Dr Marco Aurélio Spohn

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal da Fronteira Sul, Curso de Bacharelado em Ciência da Computação, Chapecó,SC, 2022.

1. VPN. 2. Mikrotik. 3. RouterOS. 4. TCP/UDP. 5. iPerf. I. Spohn, Marco Aurélio, orient. II. Universidade Federal da Fronteira Sul. III. Título.

Elaborada pelo sistema de Geração Automática de Ficha de Identificação da Obra pela UFFS com os dados fornecidos pelo(a) autor(a).

WAGNER OSÓRIO BENDER

CLASSIFICAÇÃO DOS PROTOCOLOS DE VPN QUANTO À LARGURA DE BANDA NO SISTEMA ROUTEROS

Trabalho de conclusão de curso de graduação apresentado como requisito para obtenção do grau de Bacharel em Ciência da Computação da Universidade Federal da Fronteira Sul.

Orientador: Prof. Dr. Marco Aurélio Spohn

Este trabalho de conclusão de curso foi defendido e aprovado pela banca avaliadora em: 16/08/2022

BANCA AVALIADORA:

Dr. Marco Aurélio Spohn - UFFS

Dr. Braulio Adriano de Mello - UFFS

Dr. Luciano Lores Caimi - UFFS

RESUMO

Em redes de computadores é imprescindível o uso de túneis virtuais para conexão de dois ou mais pontos, de modo a permitir tráfego de dados entre si de forma segura e confiável. Fundada na Letônia em 1996, a fabricante de equipamentos para telecomunicações Mikrotik possui um vasto portfólio de roteadores de custo acessível que executam o sistema operacional proprietário denominado RouterOS e é uma das principais escolhas em pequenos e médios negócios do mercado brasileiro. O sistema operacional Mikrotik RouterOS possui alguns protocolos de VPN dos quais pode operar como cliente e/ou servidor. Com eles é possível estabelecer um túnel de VPN entre equipamentos do mesmo ou de fabricantes distintos. Dos protocolos existentes, destacam-se os seguintes: PPTP, SSTP, L2TP e OpenVPN. Neste trabalho, foram testados os protocolos de VPN disponíveis no RouterOS utilizando o gerador de tráfego por IP iPerf através da VPN estabelecida em um ambiente fechado com equipamentos fixos, de modo a classificar o desempenho dos mesmos quanto à máxima largura de banda alcançada. Após uma análise comparativa, foi possível aferir que o protocolo L2TP possui o melhor desempenho em largura de banda total. Outros fatores podem ser relevantes como criptografia de pacotes, então é necessário classificar qual protocolo de VPN melhor atende cada cenário de rede, levando em consideração os requisitos da rede.

Palavras-chave: VPN; Mikrotik; RouterOS; TCP/UDP; IPv4; iPerf.

ABSTRACT

In order to allow data traffic between computer networks in a safe and reliable way, the usage of virtual tunnels is indispensable. Founded in 1996, the Latvian network equipment manufacturer Mikrotik has a vast portfolio of affordable routers that run their proprietary operating system RouterOS, being a prominent choice for small and medium businesses in the Brazilian market. The Mikrotik RouterOS operating system supports several VPN protocols which it can operate as a client and/or server. Those protocols allow establishing a VPN tunnel between equipments from the same or different manufacturers. Among the existing protocols, the following were selected: PPTP, SSTP, L2TP and OpenVPN. In this paper, the VPN protocols which are available in RouterOS were tested using the iPerf IP traffic generator utility through the established VPN connection in a closed environment with fixed equipment, in order to classify their performance by the maximum bandwidth achieved. After a comparative analysis, it was possible to verify that the L2TP protocol has the best performance based on total bandwidth. Other factors may be relevant such as packet encryption, so it is necessary to classify which VPN protocol best suits each network scenario, taking into account the network requirements.

Keywords: VPN; Mikrotik; RouterOS; TCP/UDP; IPv4; iPerf.

LISTA DE FIGURAS

LISTA DE ABREVIATURAS E SIGLAS

ATM	*Asynchronous Transfer Mode*
CGNAT	*Carrier-Grade Network Address Translation*
CHAP	*Challenge Handshake Authentication Protocol*
CPU	*Central Processing Unit*
FTP	*File Transfer Protocol*
GRE	*Generic Routing Encapsulation*
HMAC	*Hash-based Message Authentication Code*
HTTP	*Hyper Text Transfer Protocol*
HTTPS	*Hyper Text Transfer Protocol Secure*
ICMP	*Internet Control Message Protocol*
IKE	*Internet Key Exchange*
IP	*Internet Protocol*
IPsec	*Internet Protocol Security*
IPv4	*Internet Protocol version 4*
IPv6	*Internet Protocol version 6*
ISAKMP	*Internet Security Association and Key Management Protocol*
L2F	*Layer Two Forwarding*
L2TP	*Layer Two Tunneling Protocol*
MK	*MikroTik*
MRU	*Maximum Receive Unit*
MSCHAP	*Microsoft Challenge Handshake Authentication Protocol*
MSS	*Maximum Segment Size*
MTU	*Maximum Transmission Unit*
NAT	*Network Address Translation*
P2P	*Peer to Peer*
PAP	*Password Authentication Protocol*
PPP	*Point-to-Point Protocol*
PPTP	*Point-to-Point Tunneling Protocol*
PSK	*Pre-Shared Key*
RFC	*Request For Comments*
RTT	*Round-Trip Time*
SSH	*Secure Shell*

SSL *Secure Sockets Layer*

TAP *Test Access Points*

TCP *Transmission Control Protocol*

TLS *Transport Layer Security*

TUN *Tunneling*

UDP *User Datagram Protocol*

VPN *Virtual Private Network*

SUMÁRIO

1 INTRODUÇÃO

As conexões de VPN *(Virtual Private Network)* têm como objetivo principal encapsular pacotes de rede IP através de uma conexão única de ponto-a-ponto entre o cliente originador e o servidor de destino, fazendo com que o cliente passe a participar de uma rede remota no servidor e acesse pacotes na camada 2, usando exclusivamente tráfego IP de camada 3 que pode ser a internet ou uma rede de qualquer tamanho.

Deste modo, as conexões VPN são largamente utilizadas para os mais diversos fins, devido ao fato de que os clientes podem escolher encaminhar parcialmente ou todo o seu tráfego através desta conexão, muitos serviços usam a conexão VPN para mascarar seu endereço IP real, de forma com que sejam identificados na internet com o endereço de IP do servidor da VPN na qual estão conectados. Isso permite, por exemplo, ver conteúdos exclusivos para um certo país, ou acessar dados que são bloqueados no seu local de origem, como por exemplo uma rede corporativa que possui firewall para bloquear acesso a conteúdo multimídia para seus funcionários. (CHIN, 1998)

Entretanto, a conexão VPN possui mais funções, como ela opera nas camadas 2 e 3 como dito por BORGES; FAGUNDES; CUNHA (2014), ela também possui um endereço IP que pode ser tanto um endereço fictício criado apenas para a conexão, como também um endereço de IP global válido na internet, que permita que o cliente se comunique com o servidor, e também que o servidor consiga se comunicar com o cliente, de forma bidirecional.

No mundo todo já está em uso um novo padrão de endereçamento que é o IPv6, este padrão veio com a proposta de eliminar a falta de endereços IPv4 públicos roteáveis na internet, já que ele possui capacidade para 340 undecilhões (340,000,000,000,000,000,000,000,000,000,000,000,000) de endereços contra os 4,3 bilhões de endereços possíveis pelo IPv4 (SANTOS, 2016), porém, nem todos os dispositivos suportam o IPv6, e muitos dispositivos fabricados até mesmo nos dias atuais (como sistemas embarcados mais simples e equipamentos de IoT), ainda saem de fábrica com suporte apenas ao IPv4. Se não bastasse, uma consulta rápida realizada em diversos locais da cidade de Chapecó e cidades vizinhas, mostrou que um número considerável de provedores de internet da cidade apenas disponibiliza acesso via IPv4 aos seus clientes, nem sequer ofertando endereços IPv6 aos mesmos. SANTOS (2016) realizou um estudo sobre a implementação de IPv6 em um provedor no estado do Paraná e encontrou dados similares.

1.1 O problema

Diante da situação onde diversos protocolos de VPN estão disponíveis para uso com custo de implantação similar e sem maiores explicações sobre suas diferenças, não fica evidente qual deles escolher, uma vez que apenas um é necessário para efetuar um túnel de VPN, e que qualquer um deles consegue atingir o mesmo objetivo. Diferentes requisitos de segurança ou até mesmo diferentes limitações de firewall podem dar preferência ou até mesmo tornar inviável a utilização de alguns dos protocolos de VPN disponíveis. Em um cenário onde todos os protocolos conseguem operar de igual para igual, ainda assim é necessário aferir qual deles entregará o melhor desempenho, usando a menor quantidade de recursos do sistema.

1.2 Delimitação do Problema

Este trabalho considerará apenas o uso de IPv4 para as conexões efetuadas. Os protocolos de VPN a serem aferidos serão os existentes no sistema operacional proprietário para roteadores RouterOS, da Mikrotik. São eles: PPTP *(Point-to-Point Tunneling Protocol)*, SSTP *(Secure Socket Tunneling Protocol)*, L2TP *(Layer 2 Tunneling Protocol)* e OpenVPN. Em ambiente de produção, é muito utilizado juntamente com o L2TP o encapsulamento IPsec *(Internet Protocol Security)* de modo a oferecer uma camada de criptografia, uma vez que o protocolo L2TP não fornece criptografia para os dados, apenas para suas próprias mensagens de controle. Neste trabalho será testado o protocolo L2TP puro e o protocolo L2TP com criptografia IPsec.

1.3 Justificativa

Os protocolos de VPN possuem dentre suas características o encapsulamento de pacotes, fazendo com que todos os pacotes passem por uma única conexão, a qual também pode ser criptografada dependendo do protocolo escolhido, isso demanda poder computacional nas duas pontas da conexão, e como qualquer outro algoritmo, é passível de falhas e degradações de desempenho mediante os cenários adversos de implementação e equipamentos utilizados no processo. (BORGES; FAGUNDES; CUNHA, 2014)

A escolha do Mikrotik RouterOS como sistema de testes se deu pelo fato dos equipamentos desta fabricante terem larga adoção no mercado corporativo local de pequenas e médias empresas além de provedores de serviço de internet. São equipamentos robustos e completos

com boa relação de custo x benefício e alta confiabilidade. O sistema operacional RouterOS também pode ser instalado em plataforma x86 mediante aquisição de licença avulsa, já os equipamentos físicos possuem a licença pré-instalada. A fabricante também fornece garantia de atualização de sistema por no mínimo 5 anos para todos os equipamentos vendidos. As versões do RouterOS atuais possuem suporte nativo em ambos os modos cliente e servidor aos protocolos de VPN: PPTP, SSTP, L2TP e OpenVPN. Isto gera uma incerteza sobre qual protocolo escolher no momento de implantação de um serviço de VPN, especialmente quando ambos os participantes da conexão (cliente e servidor) possuem suporte a todos os protocolos descritos.

É necessário então, avaliar o funcionamento dos diversos protocolos existentes no ambiente escolhido. A maioria dos sistemas operacionais modernos suportam nativamente os protocolos PPTP e L2TP. Adicionalmente, o protocolo OpenVPN pode ser usado através de softwares cliente gratuitos, e o protocolo SSTP é suportado nativamente nos sistemas operacionais Microsoft Windows.

Também é importante classificar os protocolos de VPN quanto a vazão de tráfego, como dito por COMER (2015), já que na abstinência de testes reais, fica imprevisível estimar a capacidade de largura de banda real. Num ambiente onde a rede externa possui largura de banda maior que a largura de banda da qual os serviços através da VPN vão utilizar, a largura de banda disponível para a VPN fica limitada a capacidade de processamento nos equipamentos cliente e servidor.

1.4 Objetivos

1.4.1 Objetivo Geral

O objetivo deste trabalho é gerar tráfego de rede IP com a ferramenta iPerf entre dois hosts que se comunicam através do túnel VPN, a fim de classificar os existentes protocolos de VPN disponíveis no RouterOS quanto à capacidade real de vazão de tráfego IP apenas alterando os protocolos de VPN, sem alteração física na rede de transporte entre os equipamentos.

1.4.2 Objetivos Específicos

- Montagem do ambiente físico e equipamentos Mikrotik dedicados ao trabalho;
- Configuração das redes IPv4 locais separadas para cada um dos dois roteadores;

- Estabelecer a conexão de VPN entre os roteadores, um protocolo de cada vez;
- Definir uma rota estática entre os roteadores para que os hosts acessem um ao outro de forma direta;
- Realizar os experimentos por simulação de tráfego IP entre os hosts com a ferramenta iPerf;
- Avaliar os resultados obtidos e classificar os protocolos.

2 FUNDAMENTOS

2.1 Modelo OSI

A proposta de separar a rede em camadas se padronizou pelo modelo *Open Systems Interconection* (OSI), pertencente a International Organization for Standardization (ISO). Apesar de que o modelo OSI sugere que sejam utilizados sete camadas lógicas para modularidade de implementação de uma arquitetura de protocolo, a arquitetura TCP/IP normalmente consiste de quatro delas, também sendo base para o modelo ARPANET que precedeu-a durante a década de 1970. (FALL; STEVENS, 2011)

2.2 Arquitetura TCP/IP

O modelo de arquitetura TCP/IP permite que diversos dispositivos como computadores, smartphones, impressoras, roteadores e quaisquer que sejam identificados como hospedeiros *(hosts)*, se comuniquem entre si, independente de versão ou de sistema operacional. A partir do século XXI, as telecomunicações em grande escala se tornaram indispensáveis para o mundo dos negócios, pesquisa, entretenimento, e cada dia novas soluções para internet surgem e tornam a rede cada vez mais utilizada. (FALL; STEVENS, 2011)

A figura 2.1 demonstra o nome e uma breve descrição de cada uma das sete camadas do modelo OSI juntamente com as camadas equivalentes do modelo TCP/IP.

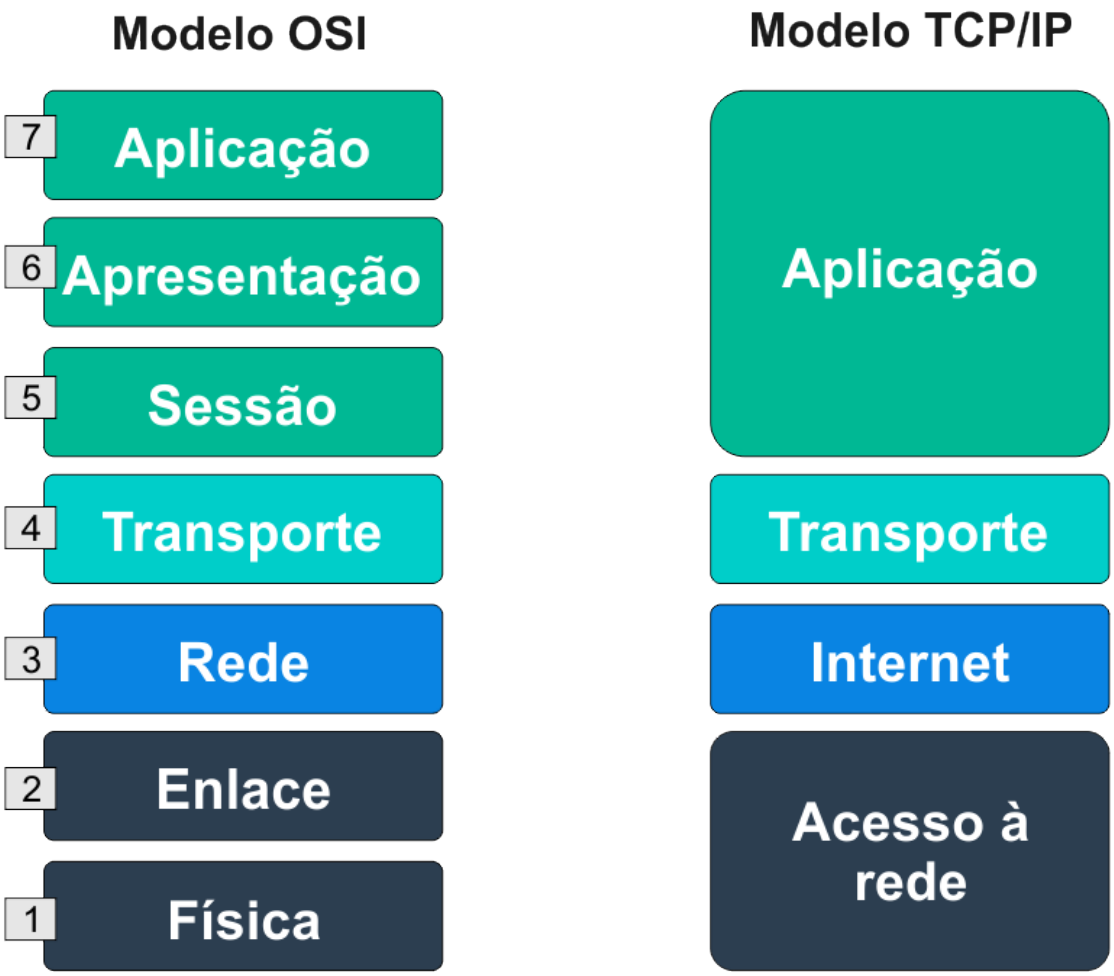

Figura 2.1 – Modelo OSI lado a lado com o modelo TCP/IP.

2.3 RouterOS

O sistema operacional para roteadores RouterOS possui a implementação de todos os protocolos de VPN utilizados neste trabalho, assim como uma interface gráfica de gerenciamento através do console Winbox ou webserver. Ambos podem ser encontrados no site oficial da fabricante MikroTik.[1]

A figura 2.2 apresenta o console Winbox, o qual foi utilizado para configurar os roteadores e do qual foram obtidas as capturas de tela apresentadas ao longo do trabalho.

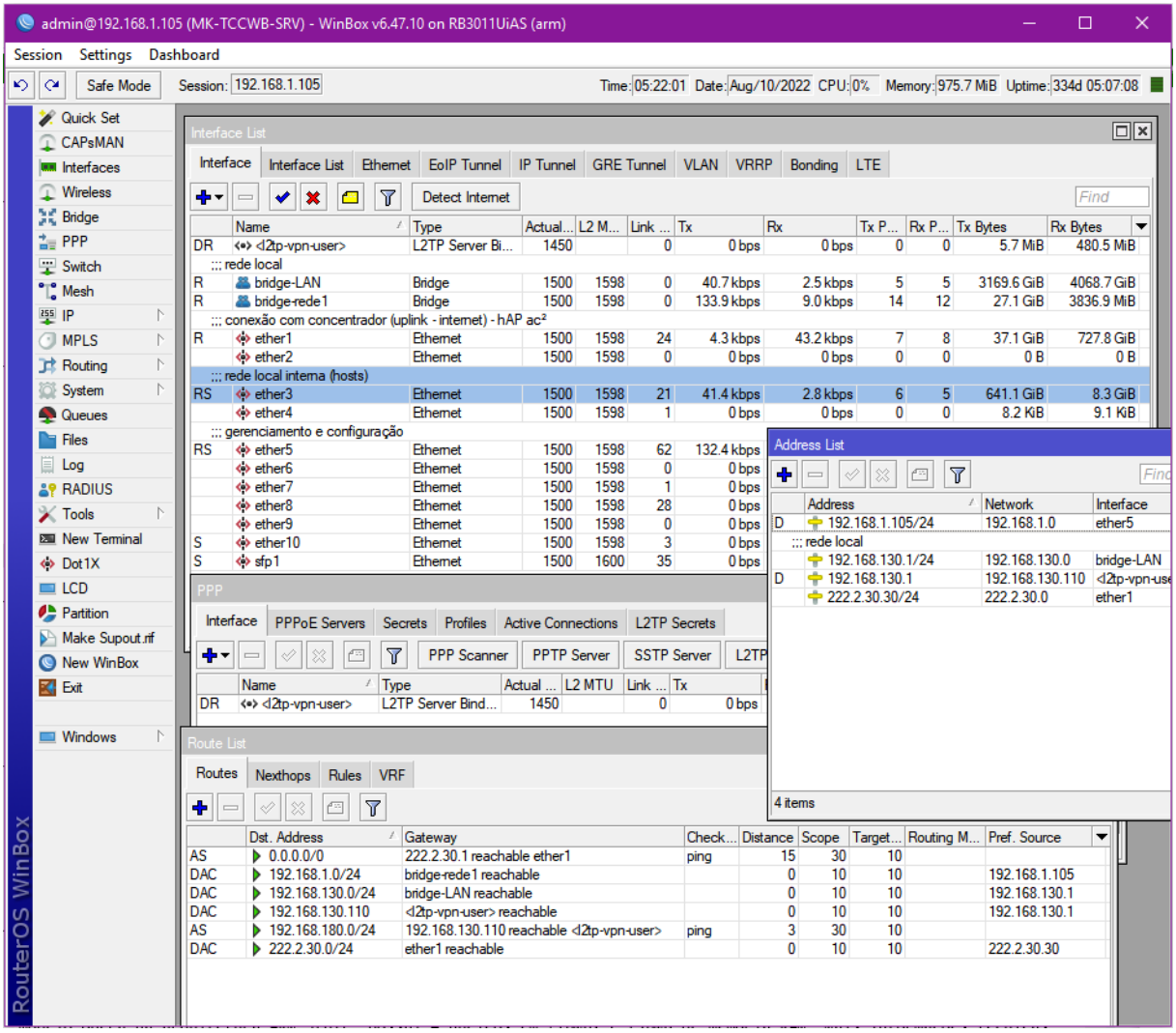

Figura 2.2 – Console de configuração Winbox.

[1] https://mikrotik.com/software

2.3.1 Roteador Servidor de VPN

Como servidor de VPN, foi destinado um equipamento da marca MikroTik modelo RouterBOARD RB3011UiAS-RM (ilustrado na figura 2.3), do qual foi possível aferir a operação de todos os protocolos de VPN destacados em modo servidor. O modelo opera na arquitetura ARM 32bit, possui 2 núcleos em 1.4GHz e 1GB de memória RAM. Mais informações técnicas sobre o equipamento podem ser consultadas na página[2] oficial do fabricante.

Figura 2.3 – MikroTik modelo RouterBOARD RB3011UiAS-RM.

2.3.2 Roteador Cliente de VPN

Como cliente de VPN, foi destinado um equipamento da marca MikroTik modelo RouterBOARD RBD53iG-5HacD2HnD hAP ac3 (ilustrado na figura 2.4), do qual foi possível aferir a operação de todos os protocolos de VPN destacados em modo cliente. O modelo opera na arquitetura ARM 32bit, possui 4 núcleos em 716MHz e 256MB de memória RAM. Mais informações técnicas sobre o equipamento podem ser consultadas na página[3] oficial do fabricante.

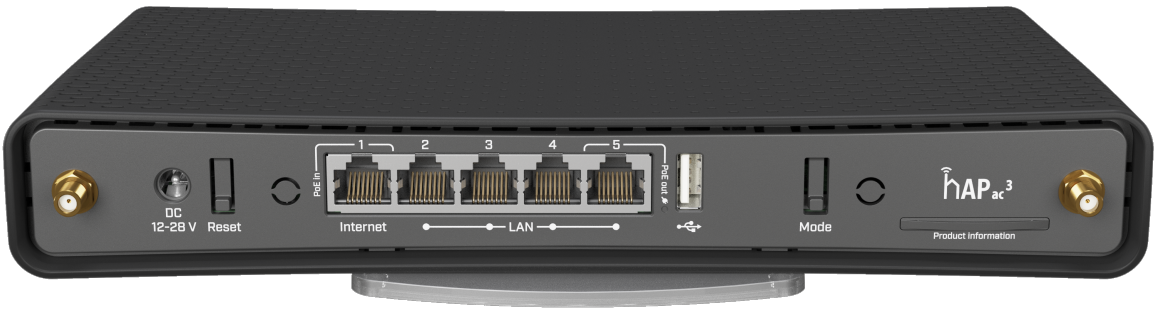

Figura 2.4 – MikroTik modelo RouterBOARD RBD53iG-5HacD2HnD.

[2] https://mikrotik.com/product/RB3011UiAS-RM
[3] https://mikrotik.com/product/hap_ac3

2.4 Tipos de autenticação: PPP, IPsec e IKE

Os tipos de autenticação são usados para autorizar o acesso da conexão por meio de credenciais, tendo particularidades entre si.

A IKE *Internet Key Exchange*, definida na [RFC2409] (THE INTERNET KEY EXCHANGE , IKE), provê uma estrutura de autenticação e troca de chaves mas não as define por completo. A ISAKMP é um protocolo definido pela [RFC2408] (INTERNET SECURITY ASSOCIATION AND KEY MANAGEMENT PROTOCOL , ISAKMP) para definição de segurança das chaves de criptografia.

Enquanto o PPP *Point-to-Point Protocol* provê a autenticação inicial, ele não provê autenticação por pacote, nem integridade ou proteção contra retransmissões. Isso implica que a identidade verificada na inicialização da autenticação PPP não é subsequentemente verificada na recepção de cada pacote.

Com o IPsec, quando a identidade firmada no IKE é autenticada, as chaves resultantes são usadas para prover autenticação por pacote, assim como proteção contra repetições e integridade. Como resultado, a identidade verificada na inicialização do IKE é subsequentemente verificada na recepção de cada pacote.

2.5 Protocolo de VPN PPTP

O protocolo PPTP é definido pela [RFC2637] (POINT-TO-POINT TUNNELING PROTOCOL , PPTP) e não especifica nenhuma mudança ao protocolo PPP mas sim uma descrição para um novo método de transporte para o PPP sobre IP.

Uma arquitetura cliente-servidor é definida para implantar o modelo de VPN PPTP. A porta de escuta do servidor para estabelecer a sessão inicial de controle da conexão é a TCP 1723, da qual também é usada para os fins de autenticação. O cliente atribui-se qualquer porta livre no seu escopo. Adicionalmente, é estabelecida uma conexão de transporte do protocolo 47 *Generic Routing Encapsulation* (GRE) diretamente entre o servidor e o cliente. Isso faz com que o PPTP não funcione em ambientes onde protocolos diferentes dos mais usados para navegação *Internet Control Message Protocol* (ICMP), *Transmission Control Protocol* (TCP) e *User Datagram Protocol* (UDP) sejam bloqueados por firewall.

A figura 2.5 mostra a captura de tráfego na interface de rede externa durante uma conexão pelo protocolo PPTP, onde a linha selecionada em azul representa a conexão na qual todo o tráfego encapsulado é transportado.

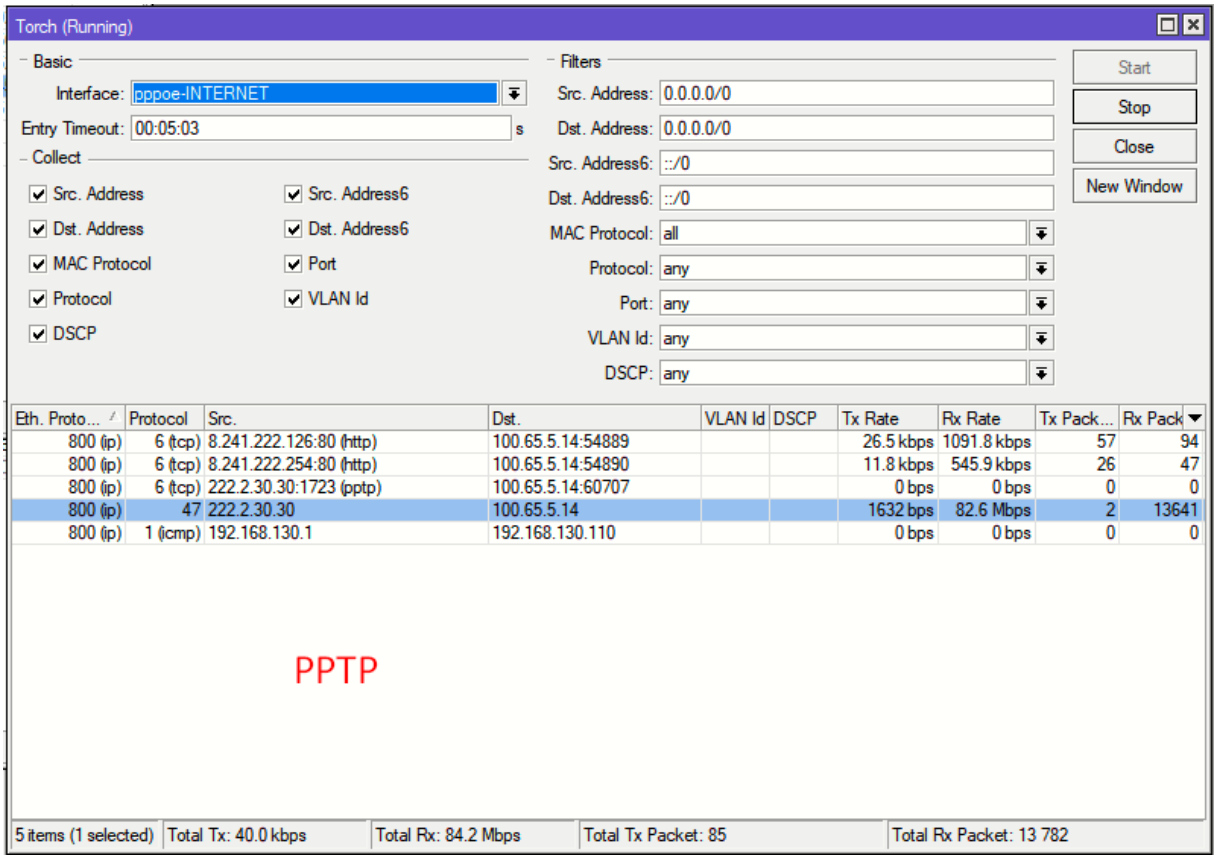

Figura 2.5 – Captura de tráfego pelo Winbox através de conexão de VPN PPTP.

2.6 Protocolo de VPN SSTP

O protocolo SSTP é proprietário da empresa Microsoft, desenvolvido primeiramente para Windows apesar de já estar disponível para outros sistemas operacionais.

SSTP é um mecanismo para encapsular tráfego do protocolo ponto-a-ponto (PPP) através do protocolo HTTPS, como especificado em [RFC1945] (HYPERTEXT TRANSFER PROTOCOL – HTTP 1.0, 1996), [RFC2616] (HYPERTEXT TRANSFER PROTOCOL – HTTP 1.1, 1999), e [RFC2818] (HTTP OVER TLS, 2000). Este protocolo permite que usuários acessem uma rede de VPN usando o HTTPS. O uso do HTTPS permite que esta VPN seja acessível mesmo com uso de firewalls e proxies, como especificado na documentação oficial.[4]

O servidor opera por padrão na porta 443, que pode ser alterada para qualquer porta TCP livre no servidor. O cliente por padrão atribui-se uma porta livre no seu escopo.

A figura 2.6 mostra a captura de tráfego na interface de rede externa durante uma conexão pelo protocolo SSTP, onde a linha selecionada em azul representa a conexão única na qual todo o tráfego encapsulado é transportado.

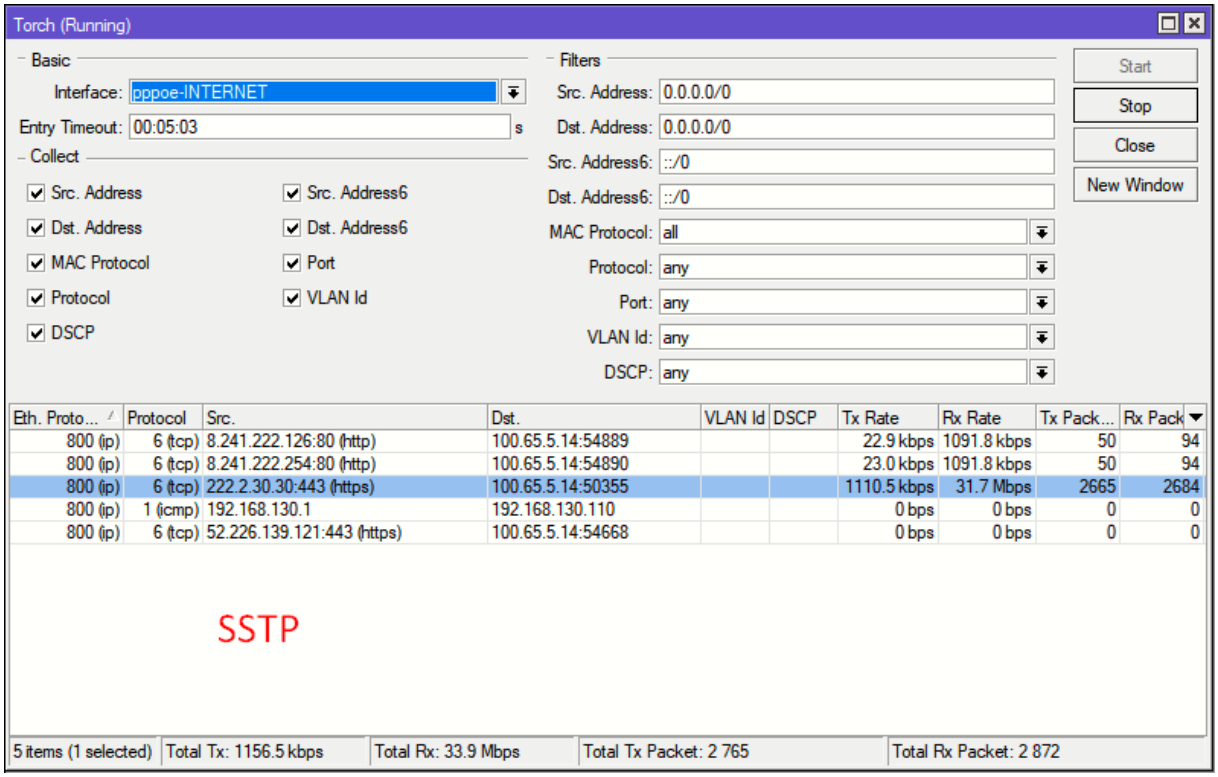

Figura 2.6 – Captura de tráfego pelo Winbox através de conexão de VPN SSTP.

[4] https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-sstp/70adc1df-c4fe-4b02-8872-f1d8b9ad806a

2.7 Protocolo de VPN L2TP

Definido pela [RFC2661] (LAYER TWO TUNNELING PROTOCOL L2TP, 1999), o L2TP facilita o encapsulamento de pacotes PPP em uma rede intermediária de maneira o mais transparente possível para usuários finais e aplicativos.

O LT2P usa a porta UDP registrada 1701 estabelecida na [RFC1700] (ASSIGNED NUMBERS, 1994). A negociação do L2TP é enviada dentro de um datagrama UDP. O inicializador (cliente) do túnel escolhe uma porta UDP livre em seu sistema (que pode ou não ser a 1701) e utiliza-a para enviar o pacote até o servidor de destino na porta de escuta 1701. O servidor escolhe uma porta livre no seu escopo (que também pode ou não ser a 1701) e envia-a de volta para a porta UDP do cliente originador, definindo sua própria porta para esta conexão. Depois que ambos cliente e servidor tiverem negociado os endereços e portas respectivos, estes itens deverão permanecer estáticos durante toda a duração da conexão da VPN.

A figura 2.7 mostra a captura de tráfego na interface de rede externa durante uma conexão pelo protocolo L2TP, onde a linha selecionada em azul representa a conexão única na qual todo o tráfego encapsulado é transportado.

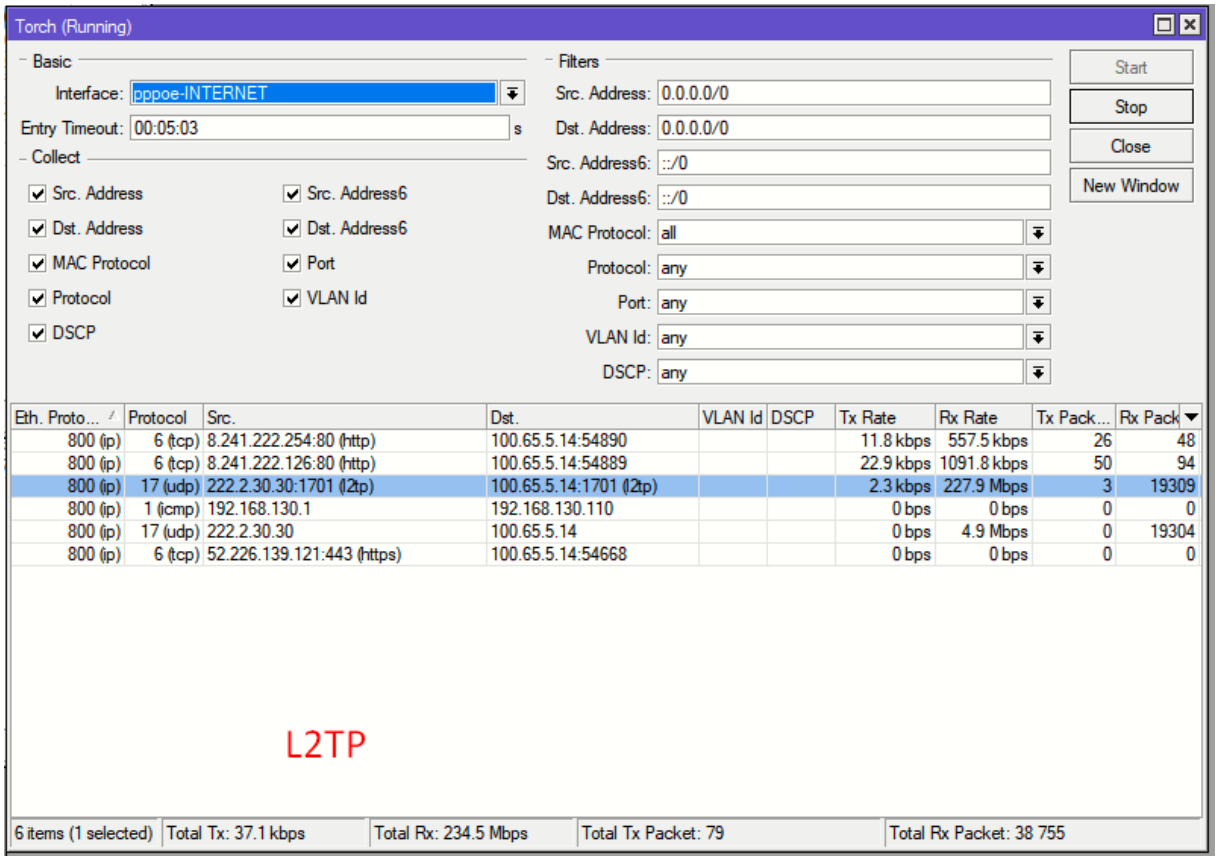

Figura 2.7 – Captura de tráfego pelo Winbox através de conexão de VPN L2TP.

2.8 Protocolo de VPN L2TP com IPsec

Descrito pela [RFC3193] (SECURING L2TP USING IPSEC, 2001), os túneis L2TP podem utilizar IPsec para prover autenticação do túnel, proteção de privacidade, verificação de integridade e proteção contra repetições. A sessão é iniciada através do protocolo ISAKMP (que inicia uma conexão UDP na porta 500), e posteriormente o tráfego é encaminhado pelo túnel IPsec estabelecido diretamente do servidor para o cliente.

A figura 2.8 mostra a captura de tráfego na interface de rede externa durante uma conexão pelo protocolo L2TP com IPsec, onde a linha selecionada em azul representa a conexão na qual todo o tráfego encapsulado é transportado.

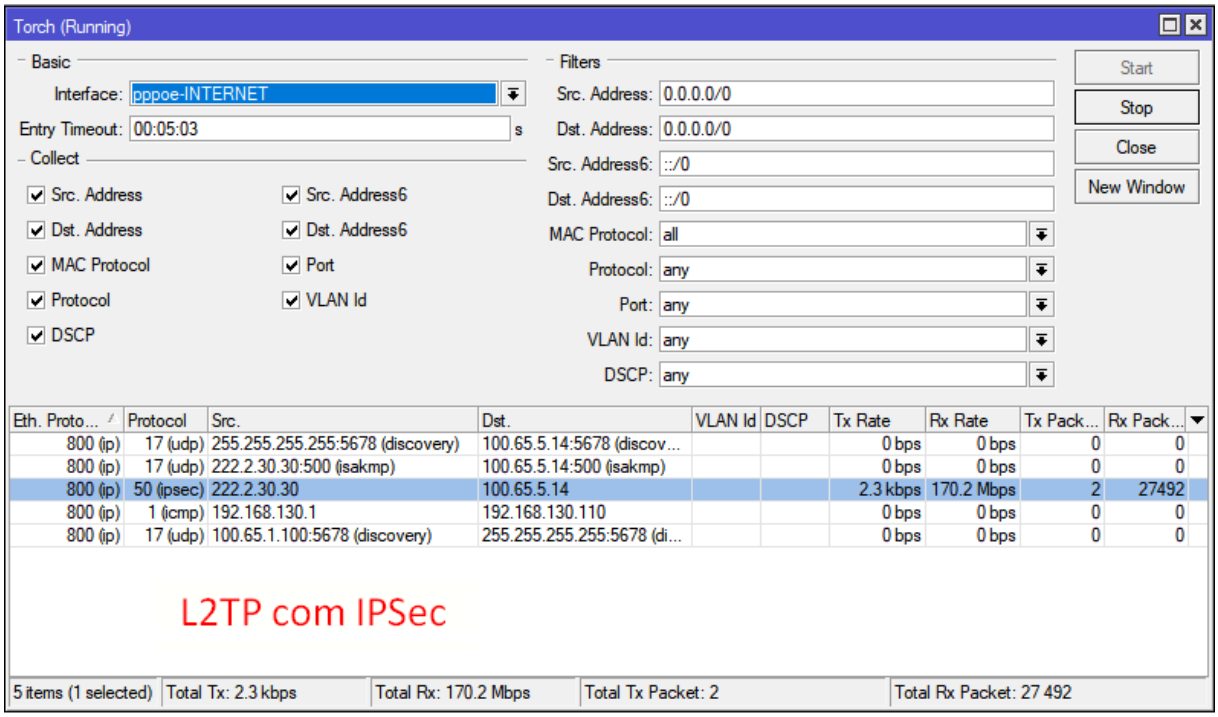

Figura 2.8 – Captura de tráfego pelo Winbox através de conexão de VPN L2TP com IPsec.

2.9 Protocolo de VPN OpenVPN

OpenVPN é um sistema de VPN que implementa técnicas para criar conexões do tipo ponto-a-ponto seguras através de redes roteadas ou diretas, tendo sua implementação e operação por meio de aplicações cliente e servidor.

Este protocolo permite que os clientes se autentiquem por meio de chaves secretas pré-compartilhadas (PSK), certificados ou nome de usuário e senha. O servidor pode também expe-

dir um certificado para cada cliente usando assinaturas e cadeias de certificação. Ele se baseia tanto na biblioteca de criptografia OpenSSL como no protocolo *Transport Layer Security* (TLS), que são utilizados para a troca de chaves.

O servidor opera por padrão na porta TCP 1194, que assim como no SSTP, esta porta pode ser alterada para qualquer porta TCP livre no servidor. O cliente por padrão atribui-se uma porta livre no seu escopo. Esta versatilidade permite que ele opere através da maioria dos firewalls, por não necessitar de outro protocolo ou outra conexão adicional para funcionamento.

O OpenVPN está disponível no modelo cliente e no modelo servidor para diversos sistemas operacionais através de aplicativo e vários outros já possuem o suporte ao OpenVPN implementado de fábrica.

O protocolo OpenVPN foi escrito por James Yonan e é um software livre, liberado sob os termos da GNU General Public License versão 2 (GPLv2).

Mais informações possem ser obtidas no site oficial do protocolo.[5]

A figura 2.9 mostra a captura de tráfego na interface de rede externa durante uma conexão pelo protocolo OpenVPN, onde a linha selecionada em azul representa a conexão única na qual todo o tráfego encapsulado é transportado.

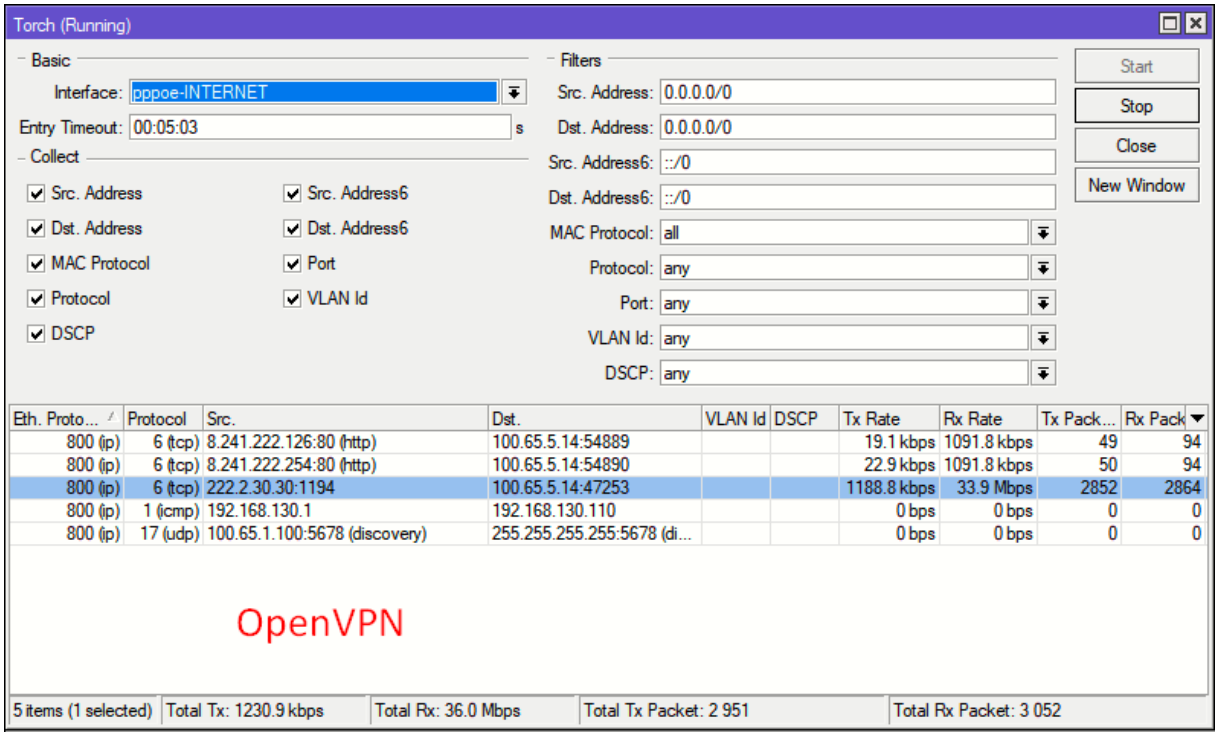

Figura 2.9 – Captura de tráfego pelo Winbox através de conexão de VPN OpenVPN.

[5] https://openvpn.net/community-resources/openvpn-protocol/

2.10 iPerf3

O software iPerf3 é uma ferramenta para medição ativa da máxima largura de banda alcançável em redes IP. Ele suporta o ajuste de diversos parâmetros relacionados a temporização, buffer e opera nos protocolos TCP e UDP através de IPv4 e IPv6. Para cada teste, ele registra a largura de banda, perda de pacotes e outros parâmetros. Ele não possui interface gráfica e opera através de linha de comando. O software é liberado sob uma licença de três cláusulas BSD.

Funcionalidades destacadas

- Em modo TCP
  - Mede a largura de banda
  - Relata o tamanho do *Maximum Segment Size* (MSS) / *Maximum Transmission Unit* (MTU)
- Em modo UDP
  - Permite ao cliente criar transmissões UDP de tamanho especificado
  - Mede a perda de pacotes
  - Mede o jitter de atraso
- Multiplataforma: Windows, Linux, Android, MacOS X, FreeBSD, OpenBSD, NetBSD, VxWorks, Solaris,...
- Cliente e servidor podem operar com múltiplas conexões simultâneas
- Pode operar por um período pré especificado pelo parâmetro (-t) ou por uma quantia de dados (opções -n ou -k)

Mais informações podem ser obtidas no site oficial da ferramenta.[6]

2.10.1 Hosts com iPerf

Fazendo o papel de hosts rodando o software iPerf, foram destinadas duas máquinas com conexão de rede local estabelecida diretamente aos equipamentos roteadores por padrão 1000BASE-T (1Gbps).

[6] https://iperf.fr/

	PPTP	SSTP	L2TP	L2TP com IPsec	OpenVPN
Autenticação segura	Sim	Sim	Sim	Sim	Sim
Criptografia por pacote	Não	Sim	Não	Sim	Sim
Flexibilidade de portas	Não	Sim	Não	Não	Sim
Código aberto	Sim	Não	Sim	Sim	Sim

Tabela 2.1 – Tabela comparativa de funcionalidades

2.11 Comparativo de funcionalidades

Na tabela 2.1 é possível comparar as funcionalidades de cada protocolo, quanto à segurança de autenticação, criptografia por cada pacote trafegado ou apenas durante o período inicial da conexão, flexibilidade na alocação de portas permitindo utilização de portas diferentes da padrão, e quanto a disponibilidade de código aberto para modificações. Estas características podem limitar ou inviabilizar a implantação de alguns dos protocolos dependendo do ambiente de rede externa ao qual eles forem submetidos. Dependendo da sensibilidade das informações a serem trafegadas, pode ser obrigatória a escolha de protocolos que reforcem este requisito.

3 TRABALHOS RELACIONADOS

Em 2020, foi realizado um estudo para interligação de diversos campus de uma escola que estavam separados em diversos prédios. Para tal, o autor de (RYANSYAH; MUSYAFFA, 2020) utilizou equipamentos MikroTik modelo RB2011L mas apenas executou a configuração pelo PPTP sem explorar os outros protocolos. O PPTP é um dos protocolos mais antigos e simples de configurar, no caso deste trabalho, o autor cita que após configuração, foi possível acessar todas as redes LAN de todos os locais.

Um dos principais fatores que impedem o sucesso de uma conexão de VPN são os firewalls configurados para permitir apenas serviços básicos de navegação, dos quais permitem apenas pacotes do tipo TCP ou UDP em portas conhecidas. Outro problema é a tradução de endereços NAT, que numa rede com muitas conexões simultâneas pode apresentar indisponibilidade de portas padrão das quais os protocolos diferentes do TCP e UDP precisam. Protocolos de VPN como PPTP ou L2TP com IPSec necessitam de protocolos adicionais como o GRE e o IPsec respectivamente, e podem não operar nestes cenários. No artigo de 2004, (ABOBA; DIXON, 2004) já era abordada a questão de que o NAT estava sendo implantado na maioria das redes, e o autor trata sobre os requisitos de compatibilidade de NAT para funcionamento do IPsec, assim como os desafios encontrados para operar tal protocolo diante do cenário proposto.

No trabalho de 2021 publicado pela IEEE (WAHANANI; IDHOM; MANDYARTHA, 2021), foi realizada uma simulação de transmissão de vídeo em diferentes qualidades e resoluções através de VPN por LT2P com IPsec e OpenVPN. Foi utilizado o software de captura de pacotes Wireshark para coleta e análise dos dados. Os autores calcularam um padrão de qualidade de serviço, baseado em atraso, largura de bandas, jitter e perda de pacotes, que são importantes para este tipo de aplicação. Apesar do L2TP levar vantagem em alguns cenários, na conclusão os autores decidiram que o OpenVPN é o melhor protocolo para esta tarefa específica.

Os autores do trabalho de 2016 (LAWAS; VIVERO; SHARMA, 2016), também publicado pela IEEE, explicam o funcionamento da tecnologia da VPN e suas aplicações, tal como as implicações de segurança. Como o foco deste trabalho eram VPNs que possuem criptografia em todos os pacotes, foram escolhidos os protocolos SSTP e IKEv2 para testes comparativos medindo largura de banda, atraso e jitter. Neste trabalho, foi encontrado que o protocolo IKEv2 possuía desempenho significativamente melhor que o SSTP em todos os parâmetros aferidos.

4 AVALIAÇÃO DE DESEMPENHO

Os testes foram executados em um ambiente físico composto de três equipamentos de rede Mikrotik e dois computadores, denominados Host A e Host B. Ambos os hosts físicos são conectados através de interface Ethernet em velocidade 1000BASE-T (1Gbps) para cada um dos dois roteadores Mikrotik de forma respectiva. O terceiro equipamento Mikrotik atua como uma interconexão entre os dois roteadores que executam o serviço de VPN, de forma a representar uma conexão externa, que pode ser a internet ou uma rede de backbone de qualquer porte. A figura 4.1 ilustra a topologia da rede montada contendo todos os equipamentos físicos e caminhos de conexão física (representados por linhas contínuas) e dos enlaces virtuais (representados por linha tracejada). Os enlaces na cor preta representam conexões situadas na rede local (LAN) dos roteadores, e as linhas azuis são conexões situadas na rede de transporte externa (internet). A linha entre os dois roteadores centrais na cor laranja representa a conexão de VPN estabelecida entre os mesmos. Esta conexão é virtual e não possui enlace físico, após os pacotes serem encapsulados pelo protocolo de VPN, eles são transportados pelas linhas em azul.

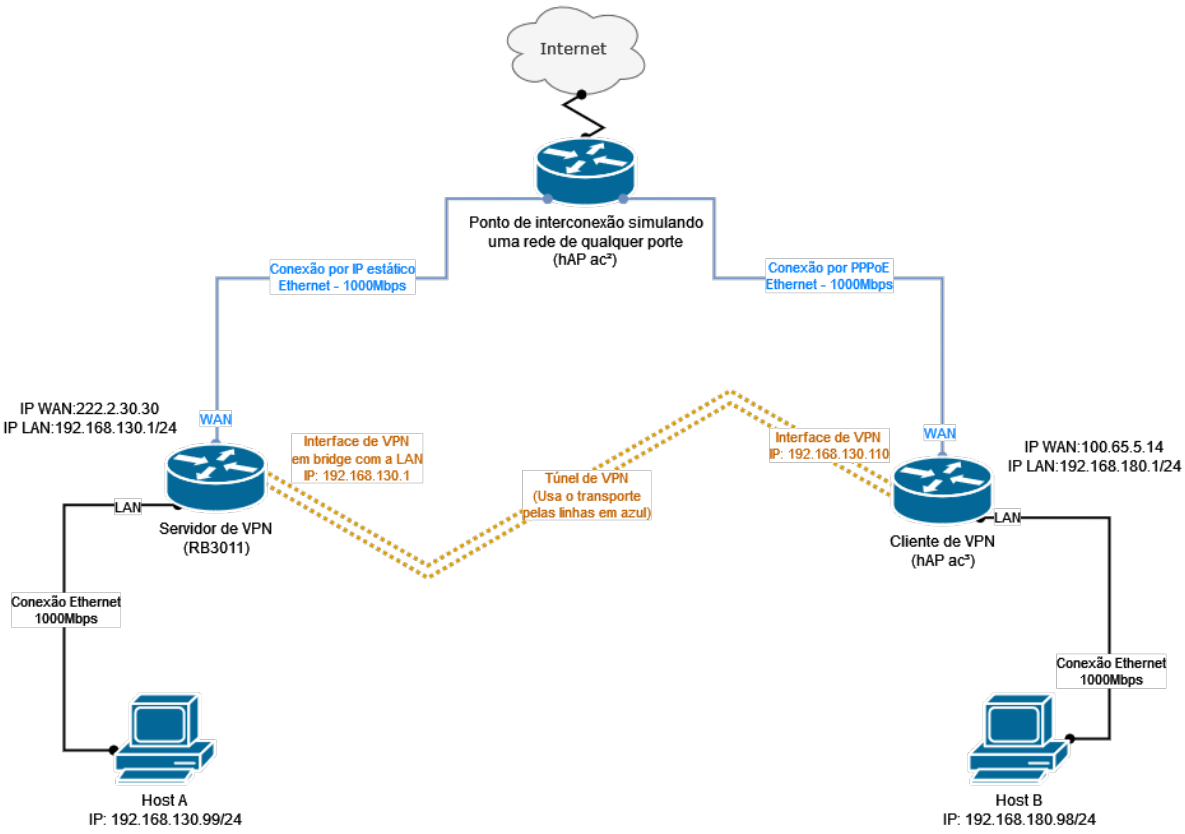

Figura 4.1 – Diagrama de conexão da rede.

Os dois roteadores Mikrotik que fazem as conexões de VPN (cliente e servidor) foram conectados entre si passando por um terceiro equipamento que serve como referência a uma rede de qualquer distância, neste caso, simulando uma rede maior, que pode ser vista como a Internet.

Após a conexão entre os hosts ser efetuada e acessível exclusivamente pelo túnel de VPN de quaisquer um dos protocolos de VPN a ser testado, foi utilizado o software iPerf3 para gerar tráfego TCP/UDP entre os hosts.

A figura 4.2 ilustra o software iPerf3 inicializado como servidor na porta 27005 no sistema operacional Windows, através da linha de comando deste exemplo:

```
iperf3.exe --server --port 27005 --verbose
```

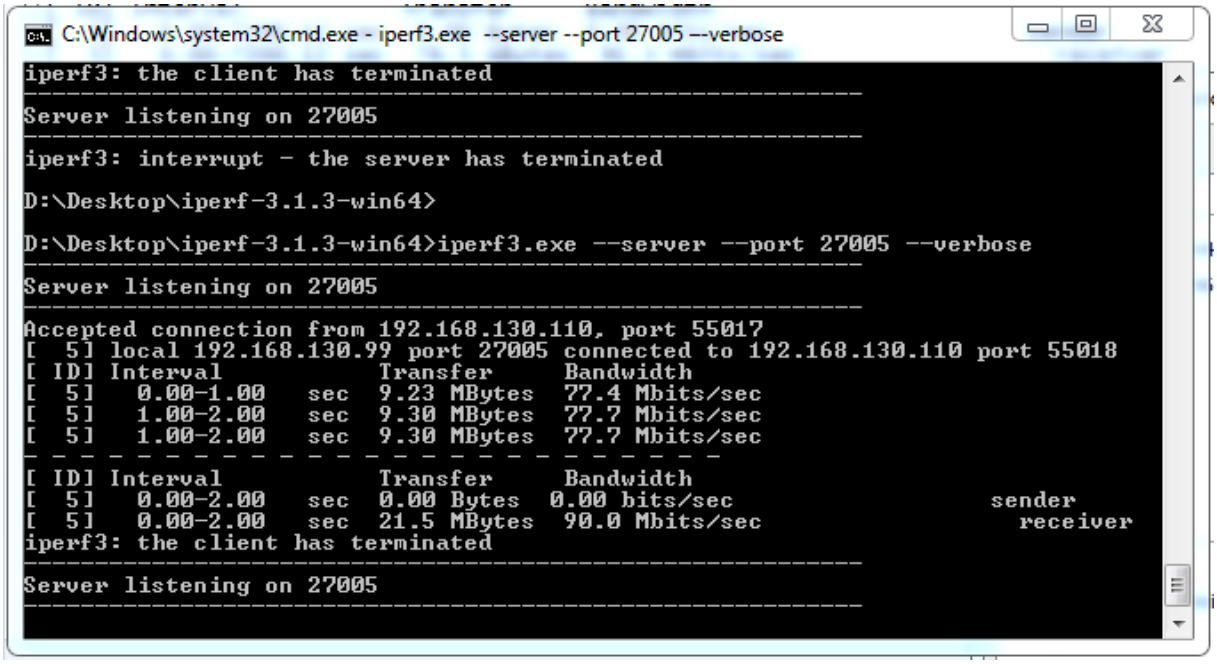

Figura 4.2 – Captura de tela do iPerf em modo servidor

Em modo cliente, no sistema operacional Windows é possível iniciar a conexão por TCP e por UDP. Para este trabalho, foi estipulado um tempo de 7200 segundos (2 horas) para cada ensaio. A figura 4.3 ilustra o console do Windows com o iPerf sendo executado em modo cliente.

Para executar o iPerf3 em modo cliente, foram utilizados os seguintes parâmetros: (–verbose ou -V) para que o software exiba um resultado mais completo, (-c) onde é informado o endereço IP do servidor do iPerf, (-p) onde é informada a porta do servidor do iPerf, (-b 0) para definir a largura de banda máxima como ilimitada, (-t 7200) para definir o tempo limite da

execução como 7200 segundos. Por padrão, o cliente efetua a conexão por TCP, entretanto se adicionado o parâmetro (–udp) a conexão é efetuada por UDP em vez do TCP.

Para os testes em TCP foi executada a linha de comando:

```
iperf3.exe --verbose -c 192.168.130.99 -p 27005 -b 0 -t 7200
```

Para os testes em UDP foi executada a linha de comando:

```
iperf3.exe -V -c 192.168.130.99 -p 27005 --udp -b 0 -t 7200
```

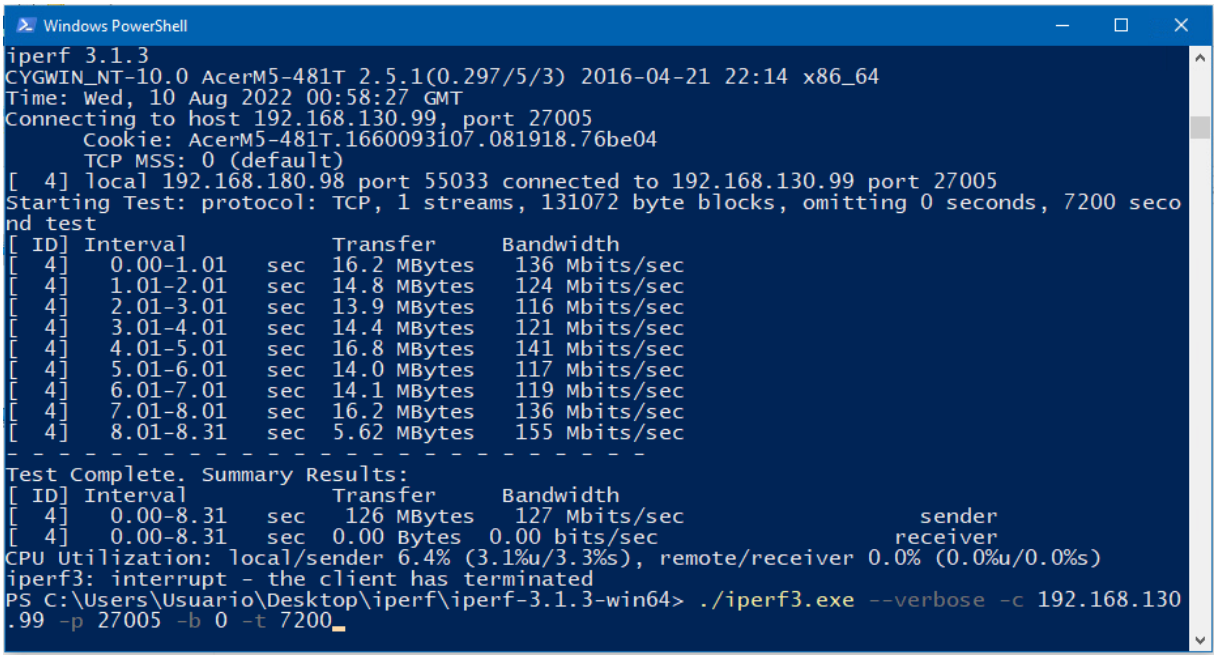

Figura 4.3 – Captura de tela do iPerf em modo cliente

Para mais detalhes sobre os parâmetros da linha de comando, a documentação completa do iPerf pode ser consultada no site oficial da ferramenta.[7]

O tráfego gerado pelo software iPerf será encaminhado do Host A para o Host B passando pelo túnel de VPN estabelecido entre os roteadores Mikrotik conforme figura 4.1. Apenas um túnel de VPN estará ativo por vez, de modo que o tráfego gerado seja exclusivo do túnel que está sendo testado naquele momento. Ao término do ensaio de cada protocolo, a conexão de VPN é desativada e um nova conexão por outro protocolo é estabelecida. Após estabelecida, um novo ensaio do iPerf é executado pela nova conexão VPN.

[7] https://iperf.fr/iperf-doc.php

4.1 Teste do PPTP

No ensaio efetuado através do protocolo PPTP através do TCP, foi obtida uma largura de banda de 91.3Mbps, informada pela saída do console do iPerf do lado cliente. Pelo UDP observa-se uma taxa aproximada através do gráfico do RouterOS (figura 4.5), já que o iPerf não informa este valor.

Os gráficos do RouterOS, representados pelas figuras 4.4 e 4.5, demonstram a variação da largura de banda durante o ensaio de duas horas com tráfego sendo gerado pelo iPerf via TCP e UDP respectivamente, passando pela interface dinâmica da conexão de VPN PPTP.

Segue abaixo a saída do console do iPerf para TCP:

Lado cliente:

```
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-7200.00 sec  76.5 GBytes  91.3 Mbits/sec                  sender
[  4]   0.00-7200.00 sec  76.5 GBytes  91.3 Mbits/sec                  receiver
CPU Utilization: local/sender 4.5% (1.7%u/2.8%s), remote/receiver 8.9% (3.0%u/6.0%s)
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-7200.12 sec  0.00 Bytes  0.00 bits/sec                 sender
[  5]   0.00-7200.12 sec  76.5 GBytes  91.3 Mbits/sec                  receiver
```

Saída do console do iPerf para UDP:

Lado cliente:

```
Test Complete. Summary Results:
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  4]   0.00-7200.00 sec   748 GBytes   892 Mbits/sec  0.920 ms  92850767/98052183 (95%)
[  4] Sent 98052183 datagrams
CPU Utilization: local/sender 50.9% (9.6%u/41.3%s), remote/receiver 5.2% (2.0%u/3.2%s)
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  5]   0.00-7200.12 sec  0.00 Bytes  0.00 bits/sec  0.920 ms  92850767/98052183 (95%)
```

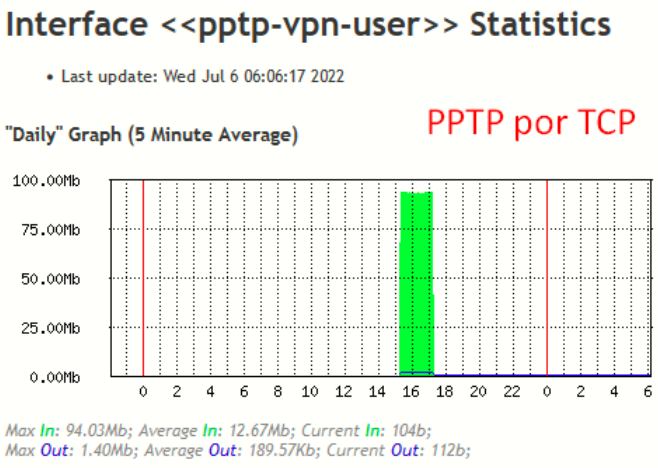

Figura 4.4 – Gráfico da interface de VPN durante o ensaio via PPTP por TCP

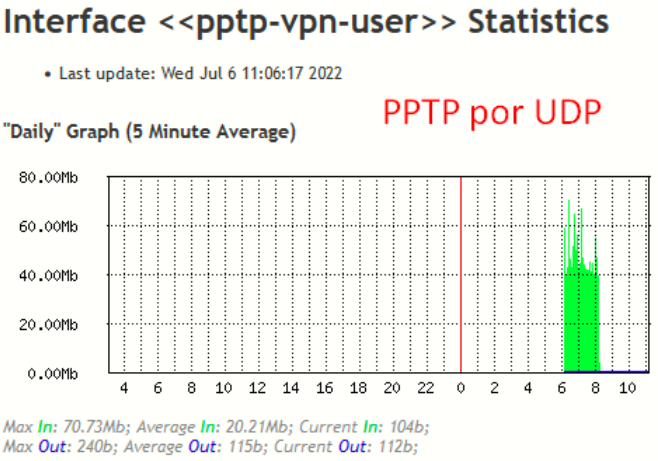

Figura 4.5 – Gráfico da interface de VPN durante o ensaio via PPTP por UDP

4.2 Teste do SSTP

No ensaio efetuado através do protocolo SSTP através do TCP, foi obtida uma largura de banda de 30.1Mbps, informada pela saída do console do iPerf do lado cliente. Pelo UDP observa-se uma taxa aproximada através do gráfico do RouterOS (figura 4.7), já que o iPerf não informa este valor.

Os gráficos do RouterOS, representados pelas figuras 4.6 e 4.7, demonstram a variação da largura de banda durante o ensaio de duas horas com tráfego sendo gerado pelo iPerf via TCP e UDP respectivamente, passando pela interface dinâmica da conexão de VPN SSTP.

Segue abaixo a saída do console do iPerf para TCP:

Lado cliente:

```
Test Complete. Summary Results:
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-7200.00 sec  25.3 GBytes  30.1 Mbits/sec                  sender
[  4]   0.00-7200.00 sec  25.3 GBytes  30.1 Mbits/sec                  receiver
CPU Utilization: local/sender 1.5% (0.6%u/0.8%s), remote/receiver 4.0% (1.5%u/2.5%s)

iperf Done.
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-7200.08 sec  0.00 Bytes  0.00 bits/sec                  sender
[  5]   0.00-7200.08 sec  25.3 GBytes  30.1 Mbits/sec                  receiver
```

Saída do console do iPerf para UDP:

Lado cliente:

```
Test Complete. Summary Results:
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  4]   0.00-7200.00 sec   747 GBytes   891 Mbits/sec  0.568 ms  97083602/97866854 (99%)
[  4] Sent 97866854 datagrams
CPU Utilization: local/sender 50.7% (9.5%u/41.2%s), remote/receiver 0.6% (0.3%u/0.3%s)
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  5]   0.00-7200.47 sec  0.00 Bytes  0.00 bits/sec  0.568 ms  97083602/97866854 (99%)
```

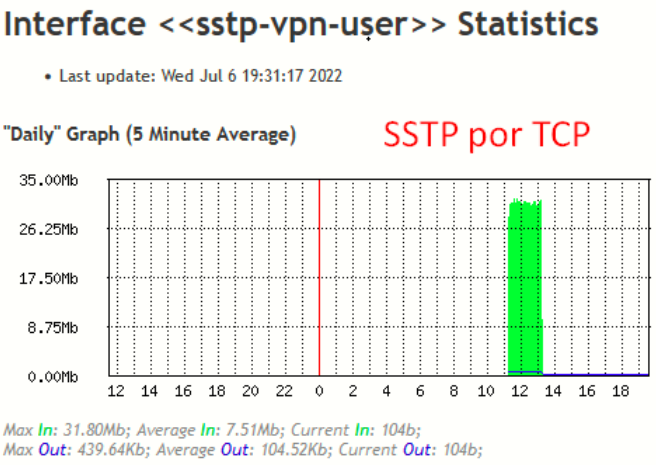

Figura 4.6 – Gráfico da interface de VPN durante o ensaio via SSTP por TCP

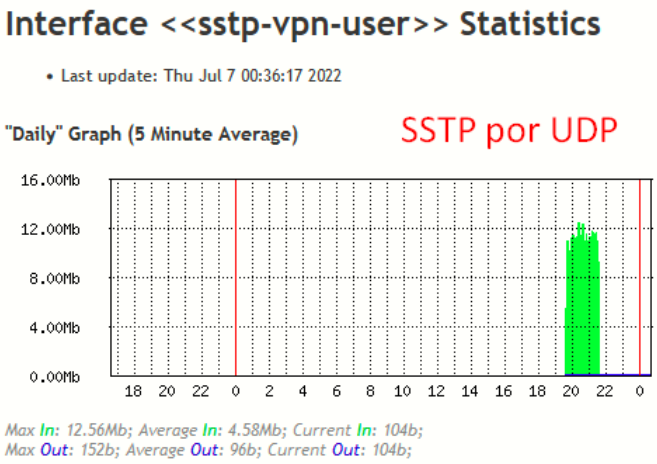

Figura 4.7 – Gráfico da interface de VPN durante o ensaio via SSTP por UDP

4.3 Teste do L2TP

No ensaio efetuado através do protocolo L2TP através do TCP, foi obtida uma largura de banda de 207Mbps, informada pela saída do console do iPerf do lado cliente. Pelo UDP observa-se uma taxa aproximada através do gráfico do RouterOS (figura 4.9), já que o iPerf não informa este valor.

Os gráficos do RouterOS, representados pelas figuras 4.8 e 4.9, demonstram a variação da largura de banda durante o ensaio de duas horas com tráfego sendo gerado pelo iPerf via TCP e UDP respectivamente, passando pela interface dinâmica da conexão de VPN L2TP.

Segue abaixo a saída do console do iPerf para TCP:

Lado cliente:

```
Test Complete. Summary Results:
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-7200.00 sec   173 GBytes   207 Mbits/sec                  sender
[  4]   0.00-7200.00 sec   173 GBytes   207 Mbits/sec                  receiver
CPU Utilization: local/sender 4.6% (1.7%u/2.9%s), remote/receiver 15.8% (4.8%u/11.0%s)
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-7200.12 sec  0.00 Bytes  0.00 bits/sec                  sender
[  5]   0.00-7200.12 sec   173 GBytes   207 Mbits/sec                  receiver
```

Saída do console do iPerf para UDP:

Lado cliente:

```
Test Complete. Summary Results:
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  4]   0.00-7200.00 sec   748 GBytes   893 Mbits/sec  0.488 ms  84860980/98070982 (87%)
[  4] Sent 98070982 datagrams
CPU Utilization: local/sender 50.7% (9.6%u/41.0%s), remote/receiver 5.9% (2.1%u/3.8%s)
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  5]   0.00-7200.12 sec  0.00 Bytes  0.00 bits/sec  0.488 ms  84860980/98070982 (87%)
```

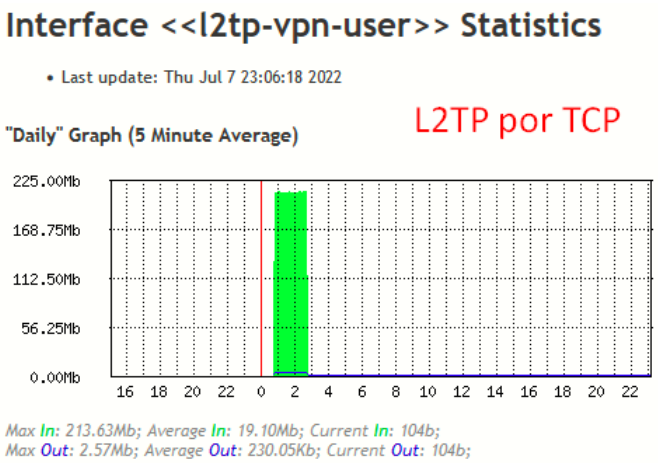

Figura 4.8 – Gráfico da interface de VPN durante o ensaio via L2TP por TCP

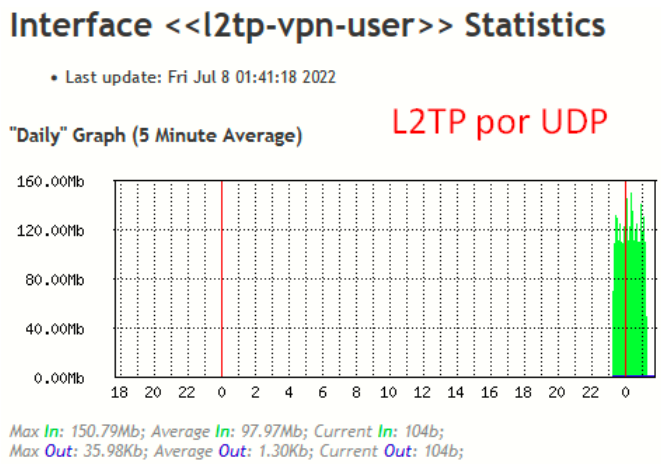

Figura 4.9 – Gráfico da interface de VPN durante o ensaio via L2TP por UDP

4.4 Teste do L2TP com IPsec

No ensaio efetuado através do protocolo L2TP com IPsec através do TCP, foi obtida uma largura de banda de 88.7Mbps, informada pela saída do console do iPerf do lado cliente. Pelo UDP observa-se uma taxa aproximada através do gráfico do RouterOS (figura 4.11), já que o iPerf não informa este valor.

Os gráficos do RouterOS, representados pelas figuras 4.10 e 4.11, demonstram a variação da largura de banda durante o ensaio de duas horas com tráfego sendo gerado pelo iPerf via TCP e UDP respectivamente, passando pela interface dinâmica da conexão de VPN L2TP com IPsec.

Segue abaixo a saída do console do iPerf para TCP:

Lado cliente:

```
Test Complete. Summary Results:
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-7200.00 sec  74.4 GBytes  88.7 Mbits/sec                  sender
[  4]   0.00-7200.00 sec  74.4 GBytes  88.7 Mbits/sec                  receiver
CPU Utilization: local/sender 4.4% (1.8%u/2.7%s), remote/receiver 5.7% (2.1%u/3.6%s)
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-7200.12 sec  0.00 Bytes  0.00 bits/sec                 sender
[  5]   0.00-7200.12 sec  74.4 GBytes  88.7 Mbits/sec                  receiver
```

Saída do console do iPerf para UDP:

Lado cliente:

```
Test Complete. Summary Results:
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  4]   0.00-7200.00 sec   746 GBytes   890 Mbits/sec  0.499 ms  88007812/97764309 (90%)
[  4] Sent 97764309 datagrams
CPU Utilization: local/sender 51.3% (9.7%u/41.6%s), remote/receiver 4.9% (1.9%u/3.0%s)
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  5]   0.00-7200.13 sec  0.00 Bytes  0.00 bits/sec  0.499 ms  88007812/97764309 (90%)
```

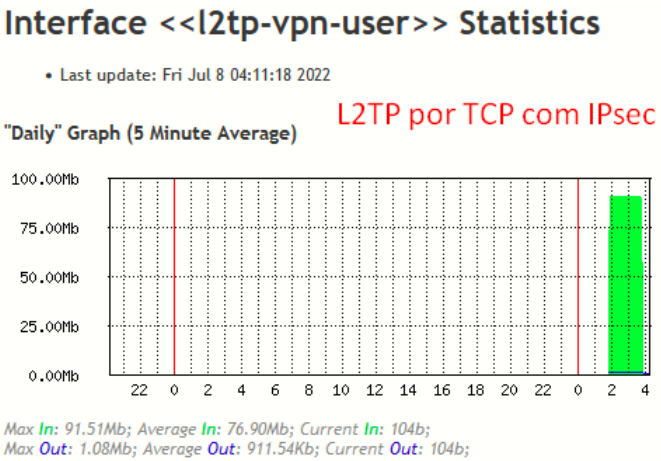

Figura 4.10 – Gráfico da interface de VPN durante o ensaio via L2TP com IPsec por TCP

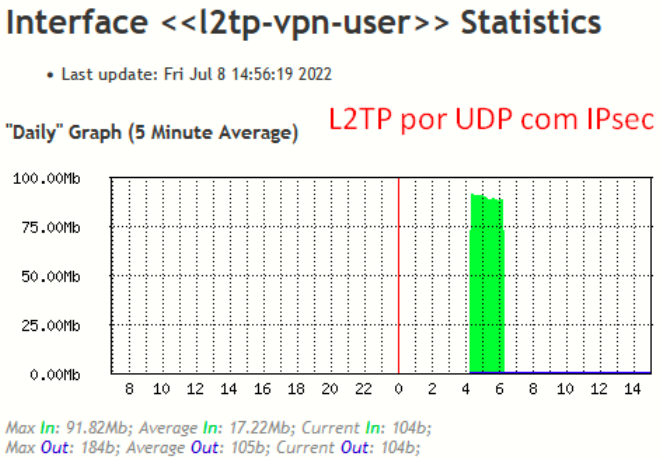

Figura 4.11 – Gráfico da interface de VPN durante o ensaio via L2TP com IPsec por UDP

4.5 Teste do OpenVPN

No ensaio efetuado através do protocolo OpenVPN através do TCP, foi obtida uma largura de banda de 32.1Mbps, informada pela saída do console do iPerf do lado cliente. Pelo UDP observa-se uma taxa aproximada através do gráfico do RouterOS (figura 4.13), já que o iPerf não informa este valor.

Os gráficos do RouterOS, representados pelas figuras 4.12 e 4.13, demonstram a variação da largura de banda durante o ensaio de duas horas com tráfego sendo gerado pelo iPerf via TCP e UDP respectivamente, passando pela interface dinâmica da conexão de VPN OpenVPN.

Segue abaixo a saída do console do iPerf para TCP:

Lado cliente:

```
Test Complete. Summary Results:
[ ID] Interval           Transfer     Bandwidth
[  4]   0.00-7200.00 sec  26.9 GBytes  32.1 Mbits/sec                  sender
[  4]   0.00-7200.00 sec  26.9 GBytes  32.1 Mbits/sec                  receiver
CPU Utilization: local/sender 1.3% (0.6%u/0.7%s), remote/receiver 4.1% (1.5%u/2.6%s)
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth
[  5]   0.00-7200.08 sec  0.00 Bytes  0.00 bits/sec                  sender
[  5]   0.00-7200.08 sec  26.9 GBytes  32.1 Mbits/sec                  receiver
```

Saída do console do iPerf para UDP:

Lado cliente:

```
Test Complete. Summary Results:
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  4]   0.00-7200.00 sec   754 GBytes   900 Mbits/sec  0.927 ms  98838000/98842667 (1e+02%)
[  4] Sent 98842667 datagrams
CPU Utilization: local/sender 48.7% (9.1%u/39.7%s), remote/receiver 0.0% (0.0%u/0.0%s)
```

Lado servidor:

```
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[  5]   0.00-7200.42 sec  0.00 Bytes  0.00 bits/sec  0.927 ms  98838000/98842667 (1e+02%)
```

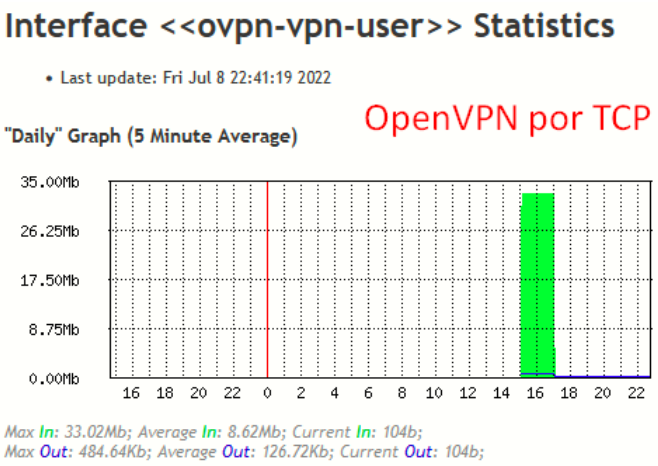

Figura 4.12 – Gráfico da interface de VPN durante o ensaio via OpenVPN por TCP

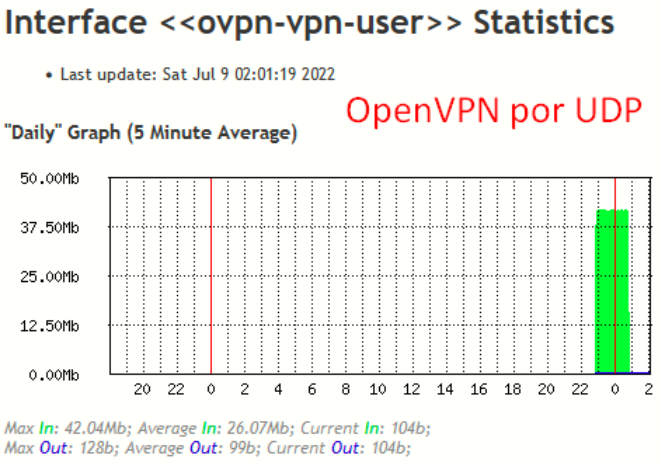

Figura 4.13 – Gráfico da interface de VPN durante o ensaio via OpenVPN por UDP

4.6 Análise comparativa de largura de banda

Diante dos testes executados gerando tráfego TCP pelo iPerf entre o Host A e o Host B passando por cada um dos protocolos de VPN existentes no RouterOS um de cada vez, foi possível identificar que a vazão de 207Mbps atingida pelo L2TP foi a maior, seguido pelo protocolo PPTP com 91.3Mbps, em terceiro lugar o L2TP com IPsec atingindo 88.7Mbps, em quarto lugar o OpenVPN com 32.1Mbps, e em quinto e último lugar o SSTP com 30.1Mbps. Na figura 4.14 foi elaborado um gráfico comparativo na plataforma online plotly[8], onde é possível visualizar a diferença de largura de banda em Mbps de todos os protocolos testados.

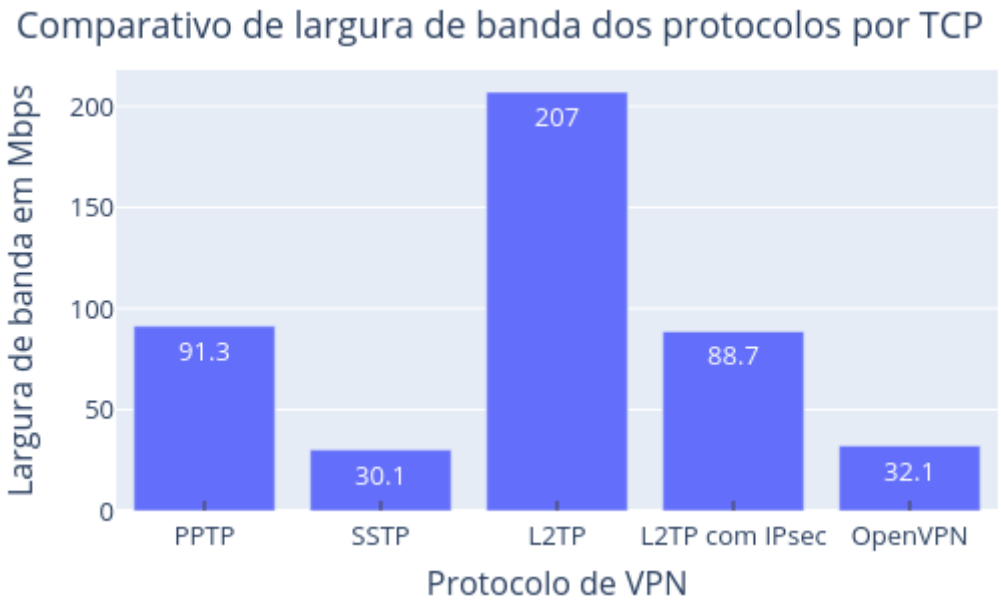

Figura 4.14 – Gráfico comparativo de largura de banda alcançada pelos protocolos em TCP

[8] https://chart-studio.plotly.com/create/

5 CONCLUSÕES

Todos os protocolos testados cumpriram com o objetivo de realizar o encapsulamento e conectar as duas redes distintas, entretanto fica evidente o melhor aproveitamento de recursos pelo L2TP, mesmo quando utilizado com o IPsec que garante a criptografia dos dados. Salienta-se também as condições de rede necessárias para operação de alguns dos protocolos, onde os mais versáteis são o SSTP e o OpenVPN que possuem flexibilidade na escolha da porta de escuta do servidor e não requerem o uso de conexões adicionais. Também é necessário avaliar a necessidade de segurança na transmissão dos dados, que pode ser por cada pacote, ou apenas da negociação inicial. Dos protocolos apresentados, o SSTP, L2TP com IPsec e OpenVPN possuem criptografia por pacote.

5.1 Trabalhos futuros

Durante a execução deste trabalho, percebi algumas ocorrências que gostaria de deixar como sugestões para trabalhos futuros. Aqui foi possível analisar o desempenho dos protocolos quanto à restrição de banda imposta mediante geração de tráfego ilimitado pelo software iPerf, entretanto, em cenários de rede distintos, outros parâmetros possuem maior relevância, como fragmentação de pacotes, tempo de ida e volta (RTT), utilização de CPU em hardware sem aceleração de criptografia, etc. Estes podem ser quantificados e então os protocolos podem ser reclassificados por cada parâmetro medido. A definição de um método para medir o tráfego efetivo ao ser gerado pelo iPerf em UDP, já que o marcador integrado do iPerf não retorna este dado de forma precisa. Os gráficos gerados pela interface do RouterOS apenas mostram a média de uso durante todo o tempo desde o momento de inicialização até o desligamento da interface, e não apenas durante o período dos testes. Uma comparação entre os mesmos protocolos em equipamentos de fabricantes distintos também é válida, uma vez que cada implementação pode ter otimizações de acordo com o sistema operacional, e isso pode gerar resultados distintos.

REFERÊNCIAS

ABOBA, B.; DIXON, W. **IPsec-Network Address Translation (NAT) Compatibility Requirements", RFC 3715**. 2004.

ASSIGNED NUMBERS. [S.l.]: RFC Editor, 1994. RFC. (1700).

BORGES, F.; FAGUNDES, B. A.; CUNHA, G. N. d. VPN: protocolos e segurança. **S/D**, [S.l.], v.10, 2014.

CHIN, L. K. Rede privada virtual–VPN. **Rede Nacional de Ensino e Pesquisa (RNP)**, [S.l.], 1998.

COMER, D. **Interligação de Redes com TCP/IP–**: princípios, protocolos e arquitetura. [S.l.]: Elsevier Brasil, 2015. v.1.

FALL, K. R.; STEVENS, W. R. **TCP/IP illustrated, volume 1**: the protocols. [S.l.]: addison-Wesley, 2011.

HTTP Over TLS. [S.l.]: RFC Editor, 2000. RFC. (2818).

Hypertext Transfer Protocol – HTTP 1.0. [S.l.]: RFC Editor, 1996. RFC. (1945).

Hypertext Transfer Protocol – HTTP 1.1. [S.l.]: RFC Editor, 1999. RFC. (2616).

Internet Security Association and Key Management Protocol (ISAKMP). [S.l.]: RFC Editor, 1998. RFC. (2408).

LAWAS, J. B. R.; VIVERO, A. C.; SHARMA, A. Network performance evaluation of VPN protocols (SSTP and IKEv2). **2016 Thirteenth International Conference on Wireless and Optical Communications Networks (WOCN)**, [S.l.], p.1–5, 2016.

Layer Two Tunneling Protocol L2TP. [S.l.]: RFC Editor, 1999. RFC. (2661).

Point-to-Point Tunneling Protocol (PPTP). [S.l.]: RFC Editor, 1999. RFC. (2637).

RYANSYAH, M.; MUSYAFFA, N. Implementation of VPN Using Router MikroTik at Al-Basyariah Education Foundation Bogor. **Jurnal Teknik Informatika CIT Medicom**, [S.l.], v.12, n.2, p.49–55, 2020.

SANTOS, L. S. d. **Implementação de IPv6 em um provedor de internet**. 2016. B.S. thesis — Universidade Tecnológica Federal do Paraná.

Securing L2TP using IPsec. [S.l.]: RFC Editor, 2001. RFC. (3193).

The Internet Key Exchange (IKE). [S.l.]: RFC Editor, 1998. RFC. (2409).

WAHANANI, H. E.; IDHOM, M.; MANDYARTHA, E. P. Analysis of Streaming Video on VPN Networks Between OpenVPN and L2TP/IPSec. **2021 IEEE 7th Information Technology International Seminar (ITIS)**, [S.l.], p.1–5, 2021.

6 APÊNDICE A: CÓDIGOS

Estes códigos foram exportados diretamente dos equipamentos destinados ao trabalho através do comando executado pelo terminal do Winbox:

`/export`

6.1 Código de configuração do RouterOS no equipamento servidor

```
/interface bridge
add arp=proxy-arp auto-mac=no comment="rede local " name=bridge-LAN
add name=bridge-rede1
/interface ethernet
set [ find default-name=ether1 ] comment="conex\E3o com concentrador (uplink - internet) - hAP ac\B2"
set [ find default-name=ether3 ] comment="rede local interna (hosts)"
set [ find default-name=ether5 ] comment="gerenciamento e configura\E7\E3o"
/interface list
add comment=defconf name=WAN
add comment=defconf name=LAN
/interface wireless security-profiles
set [ find default=yes ] supplicant-identity=MikroTik
/ip pool
add name=default-dhcp ranges=192.168.130.10-192.168.130.99
/ip dhcp-server
add address-pool=default-dhcp disabled=no interface=bridge-LAN lease-time=2w1d10m name=defconf
/ppp profile
add bridge=bridge-LAN change-tcp-mss=yes local-address=192.168.130.1 name=profile-VPN-IN only-one=yes remote-address=default-dhcp
/interface bridge port
add bridge=bridge-LAN comment=defconf disabled=yes interface=ether2
add bridge=bridge-LAN comment="PC na LAN do equipamento" interface=ether3
add bridge=bridge-LAN interface=sfp1
add bridge=bridge-rede1 interface=ether5
add bridge=bridge-rede1 interface=ether10
/ip neighbor discovery-settings
set discover-interface-list=all
/interface l2tp-server server
set default-profile=profile-VPN-IN enabled=yes ipsec-secret=ipsec-secret use-ipsec=yes
/interface list member
add comment=defconf interface=bridge-LAN list=LAN
add comment=defconf interface=ether1 list=WAN
/interface ovpn-server server
set certificate=Server cipher=blowfish128,aes128,aes192,aes256 default-profile=profile-VPN-IN enabled=yes
/interface pptp-server server
set enabled=yes
/interface sstp-server server
set default-profile=profile-VPN-IN enabled=yes max-mru=1600 max-mtu=1600
/ip address
add address=192.168.130.1/24 comment="rede local" interface=bridge-LAN network=192.168.130.0
add address=222.2.30.30/24 interface=ether1 network=222.2.30.0
/ip dhcp-client
add comment=defconf default-route-distance=10 disabled=no interface=ether1
add add-default-route=no disabled=no interface=ether5
/ip dhcp-server network
add address=192.168.130.0/24 gateway=192.168.130.1
/ip firewall filter
add action=accept chain=input connection-state=established,related,untracked
add action=drop chain=input comment="defconf: drop invalid" connection-state=invalid
add action=accept chain=input comment="defconf: accept ICMP" protocol=icmp
add action=accept chain=input comment="defconf: accept to local loopback (for CAPsMAN)" dst-address=127.0.0.1
add action=drop chain=input comment="defconf: drop all not coming from LAN" disabled=yes in-interface-list=!LAN
add action=accept chain=forward comment="defconf: accept in ipsec policy" ipsec-policy=in,ipsec
add action=accept chain=forward comment="defconf: accept out ipsec policy" ipsec-policy=out,ipsec
add action=fasttrack-connection chain=forward connection-state=established,related disabled=yes
add action=accept chain=forward connection-state=established,related,untracked
add action=drop chain=forward comment="defconf: drop invalid" connection-state=invalid
add action=drop chain=forward connection-nat-state=!dstnat connection-state=new disabled=yes in-interface-list=WAN
/ip firewall nat
add action=masquerade chain=srcnat comment="defconf: masquerade" ipsec-policy=out,none out-interface-list=WAN
/ip route
add check-gateway=ping distance=15 gateway=222.2.30.1
add check-gateway=ping distance=3 dst-address=192.168.180.0/24 gateway=192.168.130.110
/ppp secret
add name=vpn-user password=vpn-pass profile=profile-VPN-IN remote-address=192.168.130.110
/system clock
set time-zone-name=America/Sao_Paulo
/system identity
set name=MK-TCCWB-SRV
/tool graphing interface
add
/tool graphing queue
add
/tool graphing resource
add
/tool mac-server
```

```
set allowed-interface-list=LAN
/tool mac-server mac-winbox
set allowed-interface-list=LAN
/tool romon
set enabled=yes
```

6.2 Código de configuração do RouterOS no equipamento cliente

```
/interface bridge
add arp=proxy-arp auto-mac=no name=bridgeLAN
/interface wireless
set [ find default-name=wlan1 ] band=2ghz-b/g/n mode=ap-bridge name=wlan2G rx-chains=0 ssid=MikroTik-AB991E
set [ find default-name=wlan2 ] band=5ghz-a/n/ac mode=ap-bridge name=wlan5G rx-chains=1 ssid=MikroTik-AB991F
/interface ethernet
set [ find default-name=ether1 ] comment="acesso \"internet\" por PPPoE - hAP ac\B2"
set [ find default-name=ether3 ] comment="rede local interna (hosts)"
set [ find default-name=ether5 ] comment="gerenciamento e configura\E7\E3o"
/interface pppoe-client
add add-default-route=yes default-route-distance=10 interface=ether1 name=pppoe-INTERNET password=hapac3 user=hapac3
/interface l2tp-client
add connect-to=222.2.30.30 name=vpn-L2TP password=vpn-pass user=vpn-user
add connect-to=222.2.30.30 disabled=no ipsec-secret=ipsec-secret name=vpn-L2TP-IPSec password=vpn-pass use-ipsec=yes user=vpn-user
/interface sstp-client
add authentication=mschap1,mschap2 connect-to=222.2.30.30 name=vpn-SSTP password=vpn-pass user=vpn-user
/interface ovpn-client
add connect-to=222.2.30.30 disabled=yes mac-address=02:22:5B:4E:37:E9 name=vpn-OVPN password=vpn-pass user=vpn-user
/interface pptp-client
add connect-to=222.2.30.30 name=vpn-PPTP password=vpn-pass user=vpn-user
/interface list
add comment=defconf name=WAN
add comment=defconf name=LAN
/interface wireless security-profiles
set [ find default=yes ] authentication-types=wpa2-psk eap-methods="" mode=dynamic-keys wpa2-pre-shared-key=88887777
/ip pool
add name=default-dhcp ranges=192.168.180.10-192.168.180.99
/ip dhcp-server
add address-pool=default-dhcp disabled=no interface=bridgeLAN lease-time=2w1d10m name=defconf
/interface bridge port
add bridge=bridgeLAN comment=defconf interface=ether3
add bridge=bridgeLAN comment=defconf disabled=yes interface=ether4
add bridge=bridgeLAN comment=defconf disabled=yes interface=ether5
/ip neighbor discovery-settings
set discover-interface-list=all
/interface list member
add comment=defconf interface=bridgeLAN list=LAN
add comment=defconf disabled=yes interface=ether1 list=WAN
add interface=pppoe-INTERNET list=WAN
/ip address
add address=192.168.180.1/24 comment="rede local" interface=bridgeLAN network=192.168.180.0
/ip dhcp-client
add comment=defconf default-route-distance=20 disabled=no interface=ether1
add add-default-route=no disabled=no interface=ether5
/ip dhcp-server network
add address=192.168.180.0/24 comment=defconf gateway=192.168.180.1
/ip firewall filter
add action=accept chain=input connection-state=established,related,untracked
add action=drop chain=input comment="defconf: drop invalid" connection-state=invalid
add action=accept chain=input comment="defconf: accept ICMP" protocol=icmp
add action=accept chain=input comment="defconf: accept to local loopback (for CAPsMAN)" dst-address=127.0.0.1
add action=drop chain=input comment="defconf: drop all not coming from LAN" disabled=yes in-interface-list=!LAN
add action=accept chain=forward comment="defconf: accept in ipsec policy" ipsec-policy=in,ipsec
add action=accept chain=forward comment="defconf: accept out ipsec policy" ipsec-policy=out,ipsec
add action=fasttrack-connection chain=forward connection-state=established,related disabled=yes
add action=accept chain=forward connection-state=established,related,untracked
add action=drop chain=forward comment="defconf: drop invalid" connection-state=invalid
add action=drop chain=forward connection-nat-state=!dstnat connection-state=new disabled=yes in-interface-list=WAN
/ip firewall nat
add action=masquerade chain=srcnat comment="defconf: masquerade" ipsec-policy=out,none out-interface-list=WAN
add action=masquerade chain=srcnat comment="defconf: masquerade" ipsec-policy=out,none
/ip route
add check-gateway=ping distance=3 dst-address=192.168.130.0/24 gateway=192.168.130.1
/system clock
set time-zone-name=America/Sao_Paulo
/system identity
set name=MK-TCCWB-CLI
/tool graphing interface
add
/tool graphing queue
add
/tool graphing resource
add
/tool mac-server
set allowed-interface-list=LAN
/tool mac-server mac-winbox
set allowed-interface-list=LAN
```