MONOGRAFIA DE PROJETO FINAL DE GRADUAÇÃO

ANÁLISE DE SOLUÇÃO *OPEN SOURCE* PARA A IMPLEMENTAÇÃO DE UMA REDE SD-WAN EM AMBIENTE CONTROLADO

João Paulo da Costa e Silva Garcia

Curso Superior de Engenharia de Redes de Comunicação

DEPARTAMENTO DE ENGENHARIA ELÉTRICA
FACULDADE DE TECNOLOGIA
UNIVERSIDADE DE BRASÍLIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

MONOGRAFIA DE PROJETO FINAL DE GRADUAÇÃO

ANÁLISE DE SOLUÇÃO *OPEN SOURCE* PARA A IMPLEMENTAÇÃO DE UMA REDE SD-WAN EM AMBIENTE CONTROLADO

João Paulo da Costa e Silva Garcia

Monografia de Projeto Final de Graduação submetida ao Departamento de Engenharia Elétrica como requisito parcial para obtenção do grau de Bacharel em Engenharia de Redes de Comunicação

Banca Examinadora

Dr. Georges Daniel Amvame Nze, EnE/UnB _____
Orientador

Dr. Fábio Lúcio Lopes de Mendonça, EnE/UnB _____
Examinador Interno

Esp. Welber Santos de Oliveira, Estácio/Brasília _____
Examinador Externo

Agradeço à todos que, de alguma forma, contribuiram para a realização deste trabalho.

AGRADECIMENTOS

Gostaria de agradecer primeiramente a Deus, pois sem ele, nada disso seria possível. Agradeço à minha família, que sempre estiveram comigo, em todos os momentos. Agradeço ao meus colegas de trabalho, onde sempre que precisei, se prontificaram a me ajudar. Agradeço ao meu orientador, Prof. Dr. Georges Daniel, por me orientar durante todo o curso.

RESUMO

Com o avanço exponencial da tecnologia, as organizações precisam criar mecanismos que facilitem a sua gestão de redes, atrelando a segurança, disponibilidade e confidencialidade. As *Wide Area Networks* tradicionais foram criadas baseadas em roteadores convencionais, exigindo-se o *backhaul* de todo tráfego de rede, tornando o processo mais oneroso. O modelo *Software-Defined Wide Area Network (SD-WAN)* foi projetado para diminuir tempo, gastos indesejados e melhorar o gerenciamento de rede de um determinado campus como um todo. Desta forma, oferece suporte a ativos de rede, melhorando o desempenho da aplicação, conferindo maior agilidade, otimização da experiência do usuário, simplificação operações, como automação e gerenciamento baseado em nuvem, assim como a otimização e autonomia de transporte *Multiprotocol Label Switching*, 4G/5G/6G, Wi-Fi, entre outros. Neste trabalho, é proposto um modelo de gerenciamento de rede *SD-WAN*, composto por dois campus, gerenciados pela ferramenta *Flexiwan*. Para tal, foi utilizado o emulador de redes chamado de *Graphical Network Simulator-3 (GNS3)* , além de softwares de código aberto, como *Pfsense, VyOS* e *Exos Switch*. Os resultados mostram uma série de análises de métricas propostas controladas pela interface de gerenciamento, o *flexiwan management*.

Palavras-chave: *SD-WAN, Flexiwan, GNS3, Exos Switch*.

ABSTRACT

With the exponential advancement of technology, organizations need to create a revolution that facilitates their network management, linking security, availability and confidentiality. Traditional Wide Area Networks were created on conventional routers, conducting the backhaul of all network traffic, making the process more costly. The Software-Defined Wide Area Network (SD-WAN) model was designed to reduce time, unnecessary expenses and improve network management for a given campus as a whole. In this way, it supports network assets, efficient application performance, providing greater agility, optimization of the user experience, simplification of operations, such as automation and cloud-based management, as well as the optimization and autonomy of transport Multiprotocol Label Switching, 4G/5G/6G, Wi-Fi, among others. In this work, an SD-WAN network management model is proposed, composed of two campuses, managed by the Flexiwan tool. For this, a network emulator called Graphical Network Simulator-3 (GNS3) was used, in addition to open source software such as Pfsense, VyOS and Exos Switch. The results show a series of monitoring analyzes controlled by the management interface, the flexiwan management.

Keywords: SD-WAN, Flexiwan, Exos Switch, GNS3.

SUMÁRIO

1 INTRODUÇÃO 1
1.1 OBJETIVOS 1
1.1.1 OBJETIVO GERAL 1
1.1.2 OBJETIVOS ESPECÍFICOS 1
1.2 JUSTIFICATIVA 2

2 FUNDAMENTAÇÃO TEÓRICA 3
2.1 *SDN e SD-WAN* 3
2.1.1 *OpenFlow* 4
2.1.2 *Open vSwitch* 5
2.2 *Secure Access Service Edge* 5
2.3 *Switch* 6
2.4 *Firewall* 7
2.5 TRABALHOS RELACIONADOS 8

3 ARQUITETURA PROPOSTA 10
3.1 FERRAMENTAS UTILIZADAS 10
3.1.1 *GRAPHICAL NETWORK SIMULATOR 3* 10
3.1.2 ORACLE VIRTUALBOX 10
3.1.3 PFSENSE 11
3.1.4 EXTREME SWITCH EXOS 11
3.1.5 FLEXIWAN 11
3.2 MOTIVAÇÃO PARA FERRAMENTAS ESCOLHIDAS 13
3.3 *Software* PARA EMULAÇÃO DA INFRAESTRUTURA 13
3.4 IMPLEMENTAÇÃO DA INFRAESTRUTURA 14
3.4.1 DEFINIÇÃO DA TOPOLOGIA 14
3.4.2 DEFINIÇÃO DE ÁREAS 15
3.5 DESCRIÇÃO DOS DISPOSITIVOS 16
3.6 PROTOCOLO DE ROTEAMENTO IMPLEMENTADO 17
3.7 DEFINIÇÃO DO ENDEREÇAMENTO 17
3.8 CONFIGURAÇÕES 18
3.8.1 FIREWALL 18
3.8.2 SWITCH EXOS 19
3.8.3 *Flexiwan Edge* 22
3.8.4 *Flexiwan Management* 23

4 RESULTADOS E ANÁLISES 28

5 CONCLUSÃO 39

5.1 TRABALHOS FUTUROS 39

REFERÊNCIAS BIBLIOGRÁFICAS 41

ANEXOS 43

I INSTALAÇÃO DA FLEXIWAN EDGE 44

LISTA DE FIGURAS

2.1 Visão da topologia do *OpenFlow*; Fonte: SDN - UFRJ . 2019 4
2.2 Visão geral do SASE; Fonte: SASE *Documentation*. 2022 5
2.3 Camadas do modelo OSI; Fonte: OSI *Documentation* 2019 6
2.4 Esquemático. Fonte: [Troia et al. 2020] 8
2.5 Esquemático. Fonte: [Mora-Huiracocha et al. 2019] 9

3.1 Visão geral da solução SD-WAN Flexiwan; Fonte: Flexiwan *Documentation* 12
3.2 Apresentação da topologia. Fonte: Autor 14
3.3 Apresentação detalhada do campus 1. Fonte: Autor 15
3.4 Apresentação detalhada do campus 2. Fonte: Autor 15
3.5 Apresentação detalhada do *edge*. Fonte: Autor 16
3.6 Configuração *EXOS1-CORE1* via *promt* de comando. Fonte: Autor 19
3.7 Configuração *EXOS1-CORE1* via interface gráfica. Fonte: Autor 19
3.8 Configuração *EXOS3-CORE3* via interface gráfica. Fonte: Autor 20
3.9 Configuração *EXOS3-CORE3* via interface gráfica. Fonte: Autor 20
3.10 OSPF habilitado no *EXOS1-CORE1*. Fonte: Autor 21
3.11 OSPF habilitado no *EXOS3-CORE3*. Fonte: Autor 21
3.12 *DHCP* configurado no *switch EXOS1-CORE1*. Fonte: Autor 22
3.13 *DHCP* configurado no *switch EXOS3-CORE3*. Fonte: Autor 22
3.14 Exemplo de interface gráfica do *FlexiEdge (R-F1)*. Fonte: Autor 22
3.15 Interface do *Flexiwan Management* para a criação de *tokens*. Fonte: Autor 23
3.16 Interface do *Flexiwan Management* para a criação de *tokens*. Fonte: Autor 23
3.17 Criação do *path label*. Fonte: Autor 24
3.18 Gerenciamento do campus 1. Fonte: Autor 25
3.19 Gerenciamento do campus 2. Fonte: Autor 26
3.20 Parte da tabela de roteamento do campus 1. Fonte: Autor 27
3.21 Para da tabela de roteamento do campus 2. Fonte: Autor 27

4.1 Exemplo de criação de regra de firewall de bloqueio no campus 1. Fonte: Autor 28
4.2 Exemplo de lista de IPs da plataforma de mídia social. Fonte: Autor 29
4.3 Exemplo de lista de IPs da plataforma de mídia social. Fonte: Autor 30
4.4 Criação de regra de bloqueio pelo *Firewall Policies*. Fonte: Autor 31
4.5 Criação de regra de bloqueio pelo *Firewall Policies* nos campus 1 e 2. Fonte: Autor 31
4.6 Primeira etapa, sem bloqueio. Fonte: Autor 31
4.7 Segunda etapa, com bloqueio. Fonte: Autor 32
4.8 Primeira etapa, com bloqueio. Fonte: Autor 32
4.9 Segunda etapa, sem bloqueio. Fonte: Autor 32
4.10 Instalação do *remote worker VPN*. Fonte: Autor 33
4.11 Página de configuração do *Remote Worker VPN configuration*. Fonte: Autor 33

4.12 Página de acesso ao *link* para *download* do cliente. Fonte: Autor 33
4.13 *status* da aplicação *remote worker VPN*. Fonte: Autor 34
4.14 Tabela de roteamento do campus 2 em funcionamento com a VPN. Fonte: Autor 34
4.15 Regra criada automaticamente pela solução. Fonte: Autor 34
4.16 Aplicação VPN em execução. Fonte: Autor 35
4.17 Alteração do método de encriptação. Fonte: Autor 35
4.18 Adição do *path label*. Fonte: Autor 36
4.19 Visão geográfica dos *sites*. Fonte: Autor 36
4.20 Visão via *wireshark* dos pacotes trocados. Fonte: Autor 37
4.21 Visão via *wireshark* do *flow graph*. Fonte: Autor 38

I.1 Configuração de rede. 44

LISTA DE TABELAS

3.1 Detalhamento das versões dos sistemas operacionais utilizados nesse trabalho 17
3.2 Detalhamento do endereçamento *IP* adotado nesse trabalho. Fonte: Autor 18

LISTA DE ABREVIATURAS E SÍMBOLOS

Siglas

API	*Application Programming Interface*
WAN	*Wide Area Networks*
SD-WAN	*Software Defined Wide Area Network*
MPLS	*Multiprotocol Label Switching*
Wi-Fi	*Wireless Fidelity*
GNS3	*Graphical Network Simulator-3*
VoiIP	*Voice Over Internet Protocol*
VPN	*Virtual Private Network*
SASE	*Secure Access Service Edge*
SSE	*Security Service Edge*
OSI	*Open Systems Interconnection*
TCP/IP	*Transmission Control Protocol/Internet Protocol*
MAC	*Media Access Control*
VLAN	*Virtual Local Area Network*
UTM	*Unified threat management*
NGFW	*Next-generation firewall*
URL	*Uniform Resource Locator* sobre TLS
GUI	*Graphical User Interface*
VM	*Virtual Machine*
IP	*Internet Protocol*
QoS	*Quality of Service*
NAT	*Network Address Translation*
CLI	*command-line interface*
DHCP	*Dynamic Host Configuration Protocol*
RAM	*Random Access Memory*
SO	Sistema Operacional
TCP	*Transport Control Protocol*
TLS	*Transport Layer Security*
UDP	*User Datagram Protocol*
VM	*Virtual machine*

1 INTRODUÇÃO

A rede *Wide Area Network (WAN)* é responsável por interligar múltiplos dispositivos de acesso distribuído em diferentes locais geográficos. É fato que, a tecnologia avança de uma forma rápida, aumentando consideravelmente a demanda por *WANs* de maior capacidade e qualidade. Desta forma, os espaços corporativos e domésticos exigem cada vez mais, uma melhor performance de rede, demandando serviços críticos, como videoconferências, chamadas *Voice Over Internet Protocol (VoIP)* e acesso a serviços privados e públicos em nuvem.

Diante da mudança exponencial que vem acontecendo no mercado de tecnologia, o sistema de redes de comunicação precisou ser adaptado para suprir as demandas desse contexto moderno. É seguro dizer, que hoje a tecnologia de comunicação dentro das empresas faz parte da alma de qualquer negócio. A continuidade e alta disponibilidade necessárias na transmissão de dados para clientes, até mesmo no serviço interno das organizações, é de extrema importância e está diretamente relacionada com a qualidade do serviço prestado pela mesma. Para que se possa garantir tal padrão de conectividade, é rotineiro que redes privadas implantadas e geridas por prestadores de serviços externos sejam utilizadas pela maioria das empresas. No entanto, uma nova tecnologia de integração chamada *Software-Defined Wide Area Network (SD-WAN)*, vem sendo explorada no mercado.

A *SD-WAN* é uma tecnologia relativamente nova, que oferece importante mudança na rede, pois traz inovação no modo de operação e retorno do investimento feito a médio prazo, além de aportar outros diversos benefícios para a empresa investidora. Com isso, ao adaptar soluções que usem o contexto de *WANs*, criou-se a *SD-WAN*. Nela, existe um suporte de serviço diferenciado, fornecendo uma estrutura de rede que move as camadas de controle e gerenciamento para a nuvem, por meio de um controlador centralizado. Ao fazer essa transição de solução *WAN* para uma forma de *software* definido, a corporação tem inúmeras vantagens, como melhor desempenho da rede, automação na implantação de rede, redução de certos custos e melhora na prestação de serviço. Hoje, a *SD-WAN* está ganhando força, com diversos fornecedores no mercado, com diferentes níveis de maturidade de seus produtos, além de soluções *open source* capazes de suprir demandas básicas.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Emular, em ambiente controlado, uma arquitetura de rede baseada em *SD-WAN*, para teste e análise de autenticação de segurança de sistemas de código aberto interconectados.

1.1.2 Objetivos Específicos

- Realizar e identificar a comunicação entre os ativos propostos através de protocolo dinâmico;

- Realizar a configuração do *router VyOS* através de protocolo dinâmico para a comunicação com o sistema proposto;
- Validar e adaptar os *Firewalls* para o ambiente proposto através de configurações;
- Validar cenários que possam vir a ser utilizados em ambientes de produção;
- Emular cenários que viabilizam parâmetros de redes trocados entre dispositivos;
- Detectar se as informações capturadas entre os dispositivos são relevantes para a autenticação contínua na ferramenta *Flexiwan*.

1.2 JUSTIFICATIVA

Com a sofisticação das redes de comunicação, o uso de soluções *SD-WAN*, torna-se necessário para otimizar e melhorar o gerenciamento de uma rede complexa. É de fácil implementação, permitindo o gerenciamento centralizado ao mesmo tempo que visa a segurança, além de ser uma solução mais barata para a implementação em um ambiente que busque alta disponibilidade.

Dessa forma, este trabalho se propõe a emular uma rede *SD-WAN* com soluções de código aberto, ainda não trabalhadas em conjunto no cenário acadêmico e científico, buscando emular possíveis cenários de controle, conseguindo extrair métricas, dados e informações que visem as boas práticas e um gerenciamento otimizado compatível com soluções de mercado privado.

Espera-se que este trabalho possa contribuir para o processo acadêmico de estudos e pesquisa, para profissionais que queiram fazer o uso de soluções de código aberto. Os resultados e os arquivos de configuração, aqui disponibilizados, os arquivos de configuração, bem como o projeto estará disponível e acessível a todos.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 *SDN E SD-WAN*

A rede definida por *software* (SDN) , é uma arquitetura dinâmica, gerenciável e adaptável, entregando uma flexibilidade desejada pelo administrador, com baixo custo. Sua principal característica se dá pela separação entre o chamado plano de controle e dados. O plano de dados consiste nos elementos responsáveis por fazer o encaminhamento na rede, sendo seu papel principal de encaminhar os dados e apresenta responsabilidades como o monitoramento de informações e acúmulo de estatísticas. Já no plano de controle, fica responsável pelo gerenciamento do plano de dados, possuindo uma visão geral da rede, usando informações adquiridas pelo plano de dados para definir como as operações na rede serão feitas [UFRJ-SDN 2019].

A *SD-WAN* é uma implemenação específica de *Software Defined Network (SDN)*, que direciona tráfego de forma dinâmica entre possíveis filiais, campus, nuvens ou centro de dados, obtendo cobertura *WAN*. Devido a isso, a *SD-WAN* apresenta benefícios comerciais para diversos negócios *multi-site*, incluindo expansão simplificada, gestão centralizada, redução de possíveis custos e uma maior rapidez, com tráfego direcionado [Yalda, Hamad e Ţăpuş 2022].

Com isso, o mercado continua a migração de roteadores tradicionais, os chamados *edge routers* de clientes, em uma implementação de *Multiprotocol Label Switching (MPLS)*, que são usados para conectar filiais a recursos físicos de *sites*, como *Data Centers* para a *SD-WAN*, com uma arquitetura mais descentralizada com cargas de trabalho em nuvem. Desta forma, a *SD-WAN* está, aos poucos, substituindo os roteadores de borda e adicionando seleção de caminho com reconhecimento de aplicativo em vários *links*, com orquestração centralizada e segurança nativa, entre outras funções de otimização de desempenho de aplicativo [Cisco 2023].

Geralmente, os fornecedores devem ter suporte a vários recursos, como:

- Operar como *router*, suportando protocolos como *Border Gateway Protocol (BGP)*, *Open Shortest Path First (OSPF)*, *hubs, spokes* e *mesh*;
- Gerenciamento centralizado para dispositivos com interface gráfica, com configuração e *upgrade* de *softwares*;
- *Virtual Private Network (VPN)*;
- Segurança em nuvem nativa;
- Solução de trabalho remoto.

As soluções estão em contínua evolução, portanto, há uma grande expectativa para novas funcionalidades, que possam, cada vez mais, integrar diversos cenários de redes.

2.1.1 *OpenFlow*

O *OpenFlow* permite que controladores de rede determinem o caminho dos pacotes de rede em uma determinada rede de *switches*. Para o entendimento do *OpenFlow*, é necessário entender os elementos que são compostos por ele. Desta forma, temos:

- Controlador: É responsável pelas regras e ações que gerenciam o encaminhamento de pacotes. Se comunica com os comutadores *OpenFlow* e é configurado conforme a aplicação, podendo garantir flexibilidade. É programável, desta forma, o desenvolvimento de melhorias associadas ao plano de controle e de dados não dependem estritamente de novos dispositivos de redes, podendo ser considerado como um servidor ou até mesmo uma única máquina, se comunicando via *TCP* [UFRJ-SDN 2019];
- Tabela de Fluxos: Ocorre a caracterização dos fluxos recebidos no roteador *OpenFlow*, contendo três campos, cabeçalho, os contadores, e ações [UFRJ-SDN 2019];
- Canal Seguro: É onde o controlador distribui as regras de encaminhamento, visando a proteção da comunicação entre controlador e roteador através de protocolos, como o *Secure Socket Layer (SSL)* [UFRJ-SDN 2019].

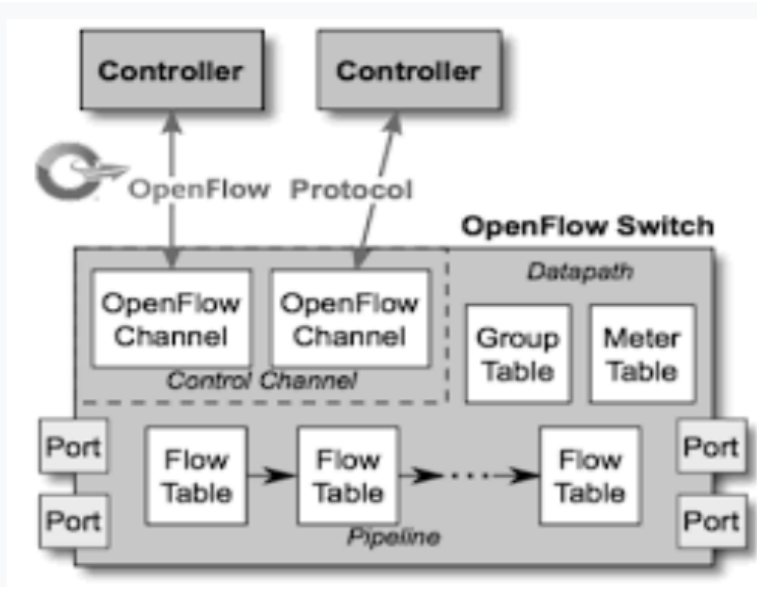

Figura 2.1: Visão da topologia do *OpenFlow*; Fonte: SDN - UFRJ . 2019

Desta forma, SDNs permitem experimentos em rede de forma rápida e independente da tecnologia proprietária dos dispositivos, possuindo baixo custo, interessando diversas fabricantes de *switch*. Existem algumas soluções de código aberto que possuem o *OpenFlow*, entre elas estão o *Opev vSwitch e Exos switch*. O *Exos Switch* será abordado posteriormente.

2.1.2 *Open vSwitch*

O *Open vSwitch* é um *switch* de *software* multicamada de código aberto. Foi projetado para oferecer suporte à distribuição em vários setores físicos, além de oferecer a várias tecnologias de virtualização baseadas em Linux, incluindo KVM e VirtualBox. Além disso, opera inteiramente no espaço do usuário sem assistência de um módulo do kernel. O *Open vSwitch* oferece suporte a uma gama de recursos que permitem que um sistema de controle de rede responda e se adapte à medida que o ambiente muda. Podemos citar, suporte ao *Netflow, IPFIX e sFlow, OpenFlow*, banco de dados de estado de redes (OVSDB) e ao GRE, que é uma ferramenta que lida com milhares de túneis simultaneamente oferecendo suporte à configuração remota para criação, configuração e desmontagem, que pode ser usado para conectar redes privadas em diferentes *data centers* [OpenvSwitch 2022].

2.2 *SECURE ACCESS SERVICE EDGE*

Secure Access Service Edge (SASE) é uma implementação, que combina capacidade de conectividade de borda de rede como, por exemplo, a *SD-WAN*, com um conjunto de *Security Service Edge (SSE)* centrados em nuvem, como o acesso de confiança zero. Desta forma, essa solução convergente, permite uma maior segurança entregue de acesso à rede, bem como recursos para usuários de qualquer lugar.

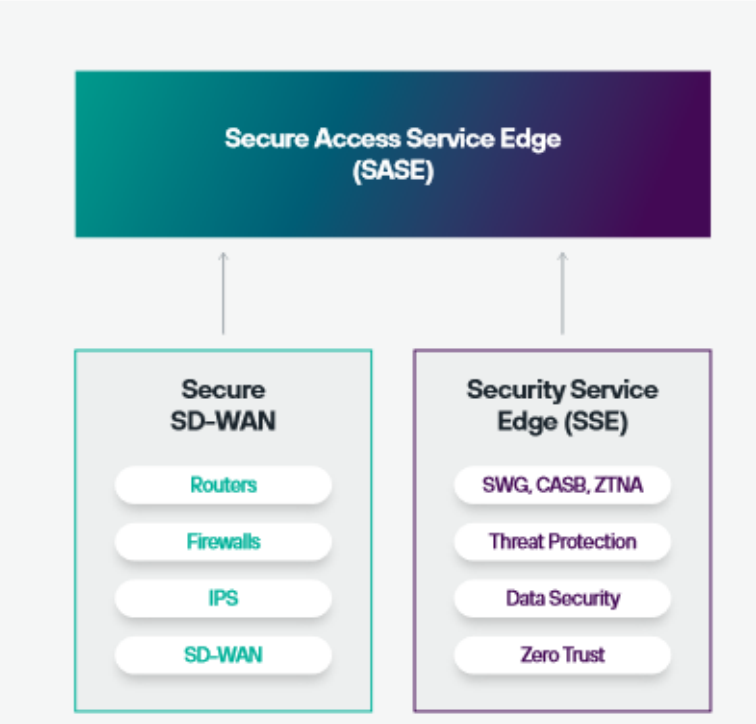

Figura 2.2: Visão geral do SASE; Fonte: SASE *Documentation*. 2022

Com isso, o objetivo do *SASE* é fornecer segurança simplificada e consistente para usuários, que façam o uso de acesso de dados de todos os lugares, além de fornecer estratégias e recursos *Zero Trust* para combinar diferentes tecnologias de rede e segurança como serviços convergentes entregues a partir da

nuvem [SASE 2022].

2.3 *SWITCH*

Switch é um dispositivo que conecta todos os elementos de uma rede, atuando como uma ligação entre *endpoints* em rede. Existem dois *modus operandi* de um *switch*. Ele pode atuar como um *switch* de camada 2 ou um *swtich* de camada 3. Ao se falar em camadas, em *hardware*, por convenção de mercado, é referenciado pelo protocolo *Open Systems Interconnection (OSI)*, e não o *Transmission Control Protocol/Internet Protocol (TCP/IP)*. A figura 2.2, as camadas compostas pelo protocolo OSI [F5 2021].

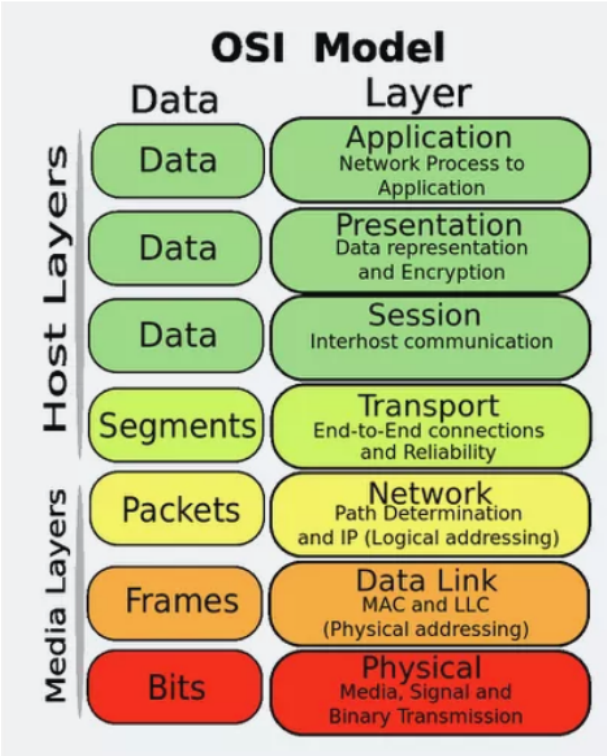

Figura 2.3: Camadas do modelo OSI; Fonte: OSI *Documentation* 2019

De forma geral, um *switch* de camada 2 é um dos equipamentos de menor complexidade, usado para conectar dispositivos de redes e clientes. Por outro lado, o *switch* de camada 3, opera em lugares mais complexos, como *data centers*, redes convergentes, entre outras. Com isso, eles se diferem basicamente em função de roteamento. Um *switch* de camada 2, utiliza endereços *Media Access Control (MAC)*, não se preocupando com endereços *Internet Protocol (IP)* ou quaisquer itens de camadas superiores. O *switch* de camada 3, possui uma tabela de endereços *MAC* e uma tabela de roteamento *IP*, além de lidar com a comunicação *intra-Virtual Local Area Network (VLAN)* e o roteamento de pacotes entre diferentes *VLANs* [F5 2021].

Há outras designações de nomes para esses *switches*, como:

- *Switch Core*: É associado aos comutadores centrais de grandes infraestruturas, centralizando o trá-

fego de dados de outros *switches*, desta forma, operam como *switches* de camada 3, possuindo alta disponibilidade [Controle 2023];

- *Switch* de distribuição: É um elemento intermediário entre os *switches core e switches* de borda. São utilizados normalmente em redes locais de empresas com grande demanda de dados, possuindo como papel principal limitar a quantidade de conexões de um *switch core*. Também são considerados de camada 3 [Controle 2023];

- *Switch* de borda: Interagem diretamente com dispositivos operados por usuários finais. Responsável pela conexão de todos os *endpoints* linkados ao *backbone* da rede. Geralmente, são dispositivos de camada 2 [Controle 2023].

2.4 *FIREWALL*

Firewall é um dispositivo de segurança da informação, podendo ser baseado em *hardware*, *software* ou ambos. Além disso, é possível, a partir de um conjunto de regras, analisar o tráfego de uma rede determinada, para delimitar operações de transmissão ou recepção de dados, que serão executadas ou não. Desta forma, pode ser entendido como, uma barreira que ajuda a bloquear algum tipo de conteúdo malicioso, possibilitando manter o tráfego necessário de rede ao mesmo tempo.

Esse bloqueio, pode ser entendido como um conjunto de políticas, que se baseiam basicamente em dois princípios, o primeiro é que todo tráfego é bloqueado, exceto o que está explicitamente autorizado, e, todo tráfego é permitido, exceto o que está explicitamente bloqueado. Com isso, os *firewalls* trabalham delimitando regras de segurança preestabelecidas, onde os pacotes de dados aprovados que entram na rede, estejam dentro das regras, enquanto o restante não consiga penetrar na rede, até o dispositivo final [Firewall 2023].

Conforme a tecnologia avança, e os novos ataques são criados, novos tipos de *firewalls* são criados e/ou melhorados. Existem alguns tipos de *firewalls* no mercado, como [Cisco 2023] :

- *Proxy Firewall*: Esse tipo de *firewall*, serve como *gataway* de uma rede para outra, direcionado a um aplicativo específico. Servidores *proxy* podem fornecer funcionalidades adicionais, como por exemplo, *cache* de segurança, impedindo possíveis conexões fora da rede;

- *Stateful firewall*: Permite ou bloqueia o tráfego, baseando-se no estado, porta e protocolo. Ademais, monitora todas as atividades, desde a abertura até seu fechamento. As decisões de filtragem são tomadas com base nas regras, que são definidas pelo administrador da rede;

- *Unified threat management (UTM) firewall*: Combina as funções de um *firewall* de inspeção de estado com prevenção de intrusão e antivírus. Além disso, é possível incluir possíveis serviços adicionais e gerenciamento em nuvem, focando em simplicidade e facilidade de uso;

- *Next-generation firewall (NGFW)*: Como os *firewalls* estão em constante evolução, há uma preferência no mercado pelos *NGFW*, para o bloqueio de ameaças modernas, como *malwares* e ataques de

camada de aplicativos. Para que um *Firewall* seja considerado um *next-generation firewall*, segundo o *gartner*, deverá incluir controles de acesso baseados em inspeção de estado, sistema integrado de prevenção de intrusão (IPS), controle de aplicativos para ver e bloquear aplicativos arriscados, caminhos de atualização para inclusão de futuros *feeds* de informações, técnicas de filtragem de *Uniform Resource Locator (URL)* baseada em geolocalização e reputação;

- *Cloud Native Firewall*: Estão em constante atualização para proteger aplicativos e a infraestrutura de carga de trabalho em escala. Por meio de recursos de dimensionamento automatizado, os *firewalls* nativos da nuvem, permitem operações de redes e que suas equipes de segurança sejam executadas de forma mais rápida, bem como o balanceamento de carga inteligente.

2.5 TRABALHOS RELACIONADOS

Neste estudo, busca-se a implementação de uma metodologia que emule o funcionamento de uma arquitetura *SD-WAN*, através de um ambiente controlado, com o objetivo de extrair métricas para a autenticação de segurança, com um sistema de componentes de código aberto interconectados.

Em [Troia et al. 2020], foi mostrado uma implementação de *SD-WAN* baseada em componentes de código aberto utilizando-se o contrador *SDN OpenDaylight* e o *Open vSwitch* com um conjunto de serviços para o monitoramento de rede e caminho baseado em políticas. Através do *OpenFlow* e um *backbone* de roteadores *VyOS*, o autor buscou escanear múltiplas redes em localizações diferentes através módulos do *Open vSwitch*. Neste aspecto, o estudo proposto trás uma implementação com uma controladora de código aberto *SD-WAN* que cumpre com os requisitos da implementação de uma rede *SD-WAN* de forma prática, contemplando a centralização dos serviços, de análises e extração de métricas na própria ferramenta de gerenciamento.

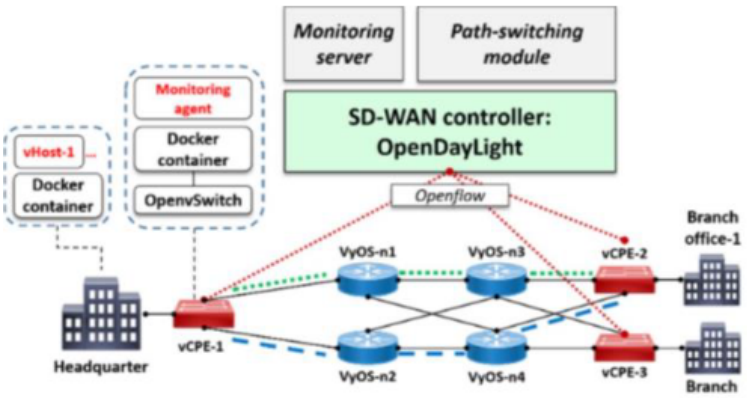

Figura 2.4: Esquemático. Fonte: [Troia et al. 2020]

Com o constante desenvolvimento da informação e tecnologias, há uma maior demanda por aplicativos e serviços que requerem a automação e implantação individualizada de recursos de rede. Em

[Mora-Huiracocha et al. 2019], foi proposto uma implementação *SD-WAN* de interconexão entre *data centers* definidos por *software* que garante uma qualidade de serviço e tráfego de priorização. Nele, o autor sugere que, com a implementação do Data Center definido por Área (SDDC), o plano de controle é dissociado do plano de dados para ser localizado em uma entidade centralizada que gerencia todos os dispositivos, surgindo novas demandas da rede, permitindo a implementação de serviços de rede baseados em aplicativos sob demanda, otimizando seu tempo na rede, dando ao administrador, plena capacidade de visualização e gerenciamento de interfaces de configuração e seus componentes de qualquer lugar. Além disso, fornece automação de serviços, facilitando o controlador para agendar eventos que requerem diferentes tipos de QoS, utilizando um gerenciamento variável de largura de banda. Para implementação da arquitetura proposta, o autor utilizou da otimização de tráfego e serviços de segurança contra falhas. Foi utilizado ferramentas como *VMWare, OpenStack Kilo, Docker* e o controlador *FloodLight*.A controladora Flexiwan possui diversas funcionalidades funcionais, como o *load balancing, QoS, VPN, IPSec*, entre outras. Desta forma, através deste estudo, optou-se pela demonstração de algumas dessas funcionalidades, sendo possível a extração de métricas completas em um ambiente proposto, através do controlador *Flexiwan Management*.

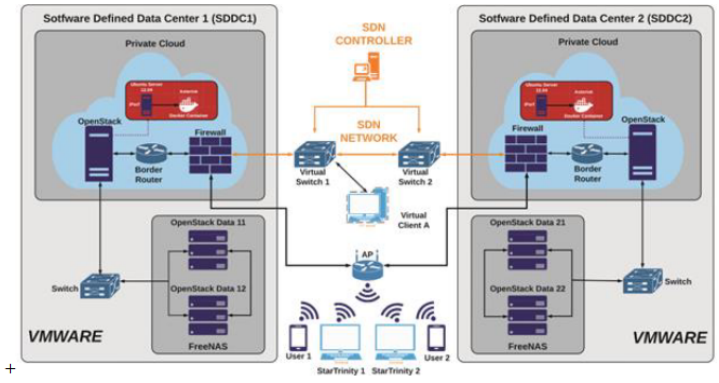

Figura 2.5: Esquemático. Fonte: [Mora-Huiracocha et al. 2019]

Por fim, assim como descreve [Bustamante e Avila-Pesantez 2021], a *SD-WAN* trouxe inúmeras soluções de diversos fornecedores, elevando o número de ameaças e vulnerabilidades desta tecnologia. Desta forma, os autores desenvolveram uma metodologia *The Grinder Framework*, que fornece diretrizes para a execução de tarefas automatizadas, sendo capazes de concluir que os ataques mais comuns estão focados em nível de gestão e vulnerabilidades *Zero Day*. Ao comparar a controladora *Flexiwan* com soluções privadas, chega-se a conclusão que soluções de código aberto, são soluções robustas contra ataques, como *XSS, CSRF, API Northbound Rest*, sendo tão competitivas quanto as soluções privadas.

3 ARQUITETURA PROPOSTA

Este capítulo busca descrever a arquitetura proposta, bem como a metodologia da infraestrutura utilizada para execução da emulação realizada. Neste capítulo, será possível verificar a topologia lógica, bem como o ambiente e as condições em que foram feitas essa emulação.

3.1 FERRAMENTAS UTILIZADAS

3.1.1 *GRAPHICAL NETWORK SIMULATOR 3*

O *software GNS3* é *open source*, mundialmente conhecido e usado por milhares de engenheiros de redes, para emular, configurar, testar e solucionar problemas de redes reais e virtuais, sendo seu uso de código aberto [GNS3 2023].

A arquitetura do *GNS3* consiste em dois componentes:

- ***GNS3-all-in-one (GUI)*:**

  Usado como a interface gráfica do usuário, sendo possível a instalação do *software* multifuncional no computador local, o que permite criar topologias. Estas precisam ser hospedadas e executadas por um processo de servidor;

- ***GNS3 virtual machine*:** O *GNS3 VM*, utilizado nesse trabalho, é recomendado pela própria fabricante, em interfaces *Windows*, para ser usado em conjunto com o *GNS3*, sendo possível fazer a execução localmente usando um *software* de virtualização, como o *Oracle VirtualBox*.

Além disso, utilizou-se o *Market Place* da própria fabricante para fazer o download das *Appliances* do *Pfsense* e do *Switch Exos*, que serão abordados posteriormente.

3.1.2 ORACLE VIRTUALBOX

O *Oracle VM VirtualBox* é uma aplicação de virtualização multiplataforma, que permite estender os recursos da máquina existente, para que ela possa executar vários sistemas operacionais, utilizando-se de uma ou várias máquinas virtuais [VirtualBox 2023].

No contexto deste trabalho e de acordo com o manual do *Oracle VM VirtualBox*, foram usadas as seguintes funcionalidades:

- **Execução e emulação de diversos sistemas operacionais de forma concorrente:** Através do virtualizador, foi possível a execução de vários sistemas operacionais, simultaneamente, no contexto *SD-WAN*. A instalação de todas as máquinas utilizadas no *VM VirtualBox* deste trabalho, foi pela extensão *.VDI*, que é suportada pela ferramenta.

As duas instâncias das máquinas virtuais baixadas da ferramenta *Flexiwan*, podem ser encontradas no site da própria ferramenta, Flexiwan. Além disso, foram utilizadas máquinas, como *Ubuntu e Ubuntu Server*, encontrados no site do Osboxes.

Desta forma, o *Oracle VM VirtualBox*, torna-se uma ferramenta essencial na extensão do emulador *GNS3*, que apesar de possuir um *Market Place* próprio, ainda é limitado em termos de usabilidade e soluções integradas.

3.1.3 Pfsense

O *Pfsense* é uma distrbuição customizada de código aberto da empresa *FreeBSD*, adaptada para ser usada como *Firewall* e roteador, sendo gerenciado por uma interface *web* gráfica. Ele provou ser bem sucedido em inúmeras instalações, desde a proteção de um único computador em pequenas redes, até milhares de dispositivos de rede em grandes corporações, universidades e outras organizações. Pode executar muitas tarefas básicas de filtragem de pacotes e *firewall* de *QoS* que o *software pfsense* fornece, facilitando o gerenciamento, o monitoramento e a manutenção[Docs 2023].

3.1.4 Extreme Switch Exos

Os *Switches* utilizados neste trabalho são da empresa *Extreme*, chamados de *Switch Exos*. Esse *switch* possui suporte no modo de funcionamento como *switch* de camada L2 e L3. Através do *Market Place* do *GNS3*, foi possível baixar a *Appliance*. É um *software* de código aberto, e possui diversas funcionalidades [Extreme 2023], como:

- **Segurança:** Utiliza o recurso chamado *MACsec*, que é um recurso de segurança chamado de *hop-by-hop*, que criptografa/descriptografa pacotes entre *switches* ou dispositivos conectados;
- **Nuvem:** É possível a combinação de vários *switches* com um gerenciamento de nuvem para agilizar e simplificar os aspectos das operações de rede, desde a implatação até o suporte;
- **Automação:** É possível a permissão do provisionamento dinâmico de serviços de rede na medida em que os usuários e dispositivos se conectam à rede.

Como dito anteriormente, o *Switch Exos* possui suporte a tecnologia do *OpenFlow*, porém, para poder usufruir desse recurso, é necessário comprar uma licença junto a fabricante.

3.1.5 Flexiwan

A *Flexiwan* é uma solução *SD-WAN e SASE*, de modelo de código aberto, que oferece uma estrutura, que contém um dispositivo de borda baseado em *software*, o *flexiEdge*, e um sistema de gerenciamento central, chamado de *flexiManage* [Flexiwan 2023]. O diagrama a seguir, apresenta sua estrutura:

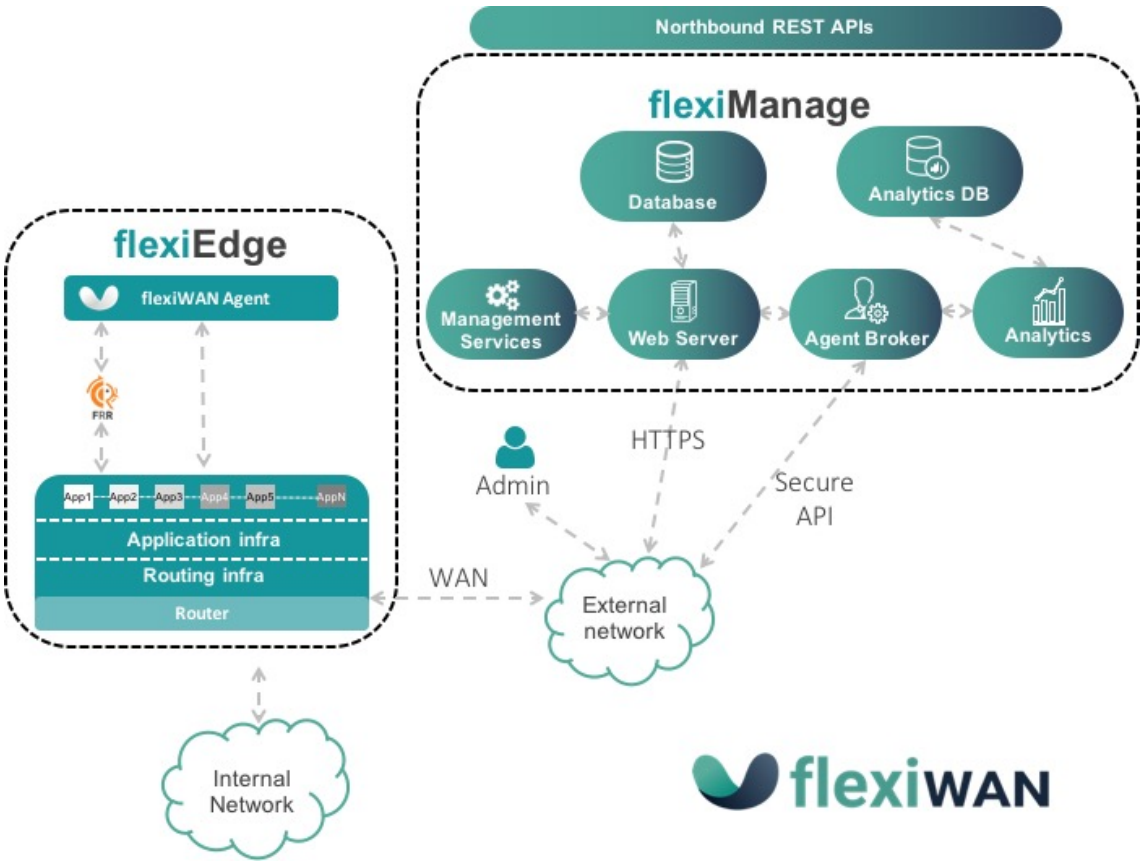

Figura 3.1: Visão geral da solução SD-WAN Flexiwan; Fonte: Flexiwan *Documentation*

A sua estrutura é composta por:

- **flexiEdge:** Compreende três componentes principais:
  - Infraestrutura do roteador, uma versão modificada do FD.io *Vector Packet Processor (VPP)*;
  - Plano de controle de roteamento, *Free Range Rounting (FRR)*;
  - *FlexiWAN Agent*, que é o elemento de *software* que conecta o *flexiEdge* ao corretor do sistema *flexiManage* por meio de *APIs* seguras *on-the-wire*. Conecta-se ao gerenciamento *FlexiWAN* usando uma conexão de soquete da *web* bidirecional segura para configurações e estatísticas, suportando recursos, como *API JSON* simplificado, separação e tradução de *APIs* em comandos internos provisionados no Linux e no roteador, armazenamento de configuração de chave e valor, comandos *CLI* para soluções de problemas, entre outros [Flexiwan 2023].
- **flexiManage:** É executado por um servidor *web* escalável, fornecendo gerenciamento de toda a rede. Por meio dele, é possível que os administradores de rede possam gerenciar dispositivos e redes, sem o mediador da rede responsável pela comunicação entre o servidor da *web* e os dispositivos *flexiEdge*, fornecendo *status* de monitoramento e atualizações para administradores de redes.

  Existe uma infinidade de *Features* suportadas e que, futuramente, serão suportadas pela ferramenta *Flexiwan*, porém, neste trabalho, as que foram usadas são:
  - *IPSec over VxLAN tunnels*

- *Flexible tunnel configuration: Full-Mesh, Hub Spoke, Partial-Mesh*
- *Tunnel quality metrics*
- *Application Identification (L3/L4)*
- *Multiple WAN/LAN interfaces*
- *DHCP server*
- *Static routes configuration*
- *Dynamic flexiEdge configuration changes*
- *Monitoring Dashboards*
- *WAN side DHCP*
- *More NAT Traversal options, including single-side symmetric NAT traversal*
- *Report Interfaces' Link Status in flexiManage*

3.2 MOTIVAÇÃO PARA FERRAMENTAS ESCOLHIDAS

SD-WAN é uma implementação específica de *SDN*, que direciona o tráfego dinamicamente entre filiais, nuvens e centro de dados para obter uma cobertura *WAN* máxima, além de apresentar consideráveis benefícios comerciais para negócios, incluindo expansão simplificada e gestão, redução de custos e uma maior rapidez no gerenciamento da infraestrutura [Yalda, Hamad e Ţăpuş 2022]. Ao se utilizar softwares de código aberto, pode-se classificar como uma inovação no processo de produção, baseado no acesso irrestrito ao código fonte em oposição à abordagem tradicional fechada, caracterizada por uma propriedade comercial [Bonaccorsi e Rossi 2003]. Com isso, a licença de código aberto permite aos usuários a liberdade de executar o programa para qualquer finalidade, seja ela para estudo, ou alguma solução em específico, modificando o programa livremente [Coppola e Neelley 2004].

Ao utilizar essas ferramentas, que são referências em suas respectativas áreas, buscou-se o estudo na implementação de uma rede *SD-WAN* totalmente *open source*, com o objetivo de inovar e contribuir para a sociedade, ao juntar ferramentas conhecidas no meio acadêmico, integrando-as em um ambiente complexo e controlado, extraindo informações úteis para possíveis cenários de redes e infraestrutura.

Portanto, utilizando-se um cenário com soluções totalmente em código aberto, foi possível unir ferramentas de diversas fabricantes, até então não exploradas, conjuntamente, em trabalhos acadêmicos, e extrair diversas informações relevantes em um ambiente emulado.

3.3 *SOFTWARE* PARA EMULAÇÃO DA INFRAESTRUTURA

Visando emular uma topologia de rede mais realística possível, que possibilitasse explorar diversos dispositivos e cenários de configuração para as análises às quais esse trabalho se propôs, foi necessário a escolha de um *software* que fosse capaz de trabalhar com emulação e gerenciamento de dispositivos virtualizados.

Desse modo, a escolha utilizada foi o *software GNS3*, devido à sua estrutura ser de código aberto e gratuita, sendo sua comunidade ativa em todo o mundo, além de compor ferramentas que possam facilitar o trabalho conjunto em um mesmo projeto, utilizando o sistema de *VPN*, por exemplo.

Neste trabalho, o *software GNS3* foi utilizado para emular a rede, por meio de ferramentas de integração de virtualização, como o *VirtualBox, VMWare, Docker*, entre outros, sendo possível a virtualização de diversos sistemas operacionais que possuem suporte para tal.

O *GNS3* possui como atribuição, a orquestração dessas máquinas virtuais, gerenciando os enlaces virtuais de interligação da topologia proposta, fazendo o controle da infraestrutura física, virtual, entre outros.

3.4 IMPLEMENTAÇÃO DA INFRAESTRUTURA

A infraestrutura física disponível para a utilização pelo autor é um *laptop* composto por um processador core i7 de décima primeira geração, 32GB de memória RAM DDR4 3200MHz e 200GB de memória SSD M.2 NVMe. Essa estrutura inviabiliza a construção de topologias maiores e mais complexas, além da implementação de *softwares* com maiores requisitos de processamento. Como dito anteriormente, o acesso restrito a *softwares* não foi impeditivo, pois o estudo propôs a emular soluções de código aberto e gratuitos.

3.4.1 Definição da topologia

O estudo propôs uma topologia inspirada em uma infraestrutura de um campus, composto por duas unidades. Foi proposta uma topologia, como pode ser visualizada no diagrama da figura 3.2, composta por três áreas principais, identificadas como *Campus 1*, *Campus 2* e *Edge*.

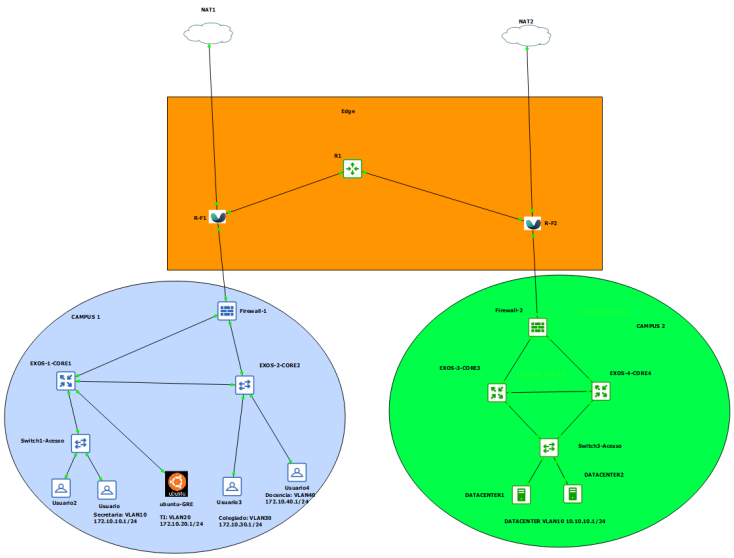

Figura 3.2: Apresentação da topologia. Fonte: Autor

3.4.2 Definição de áreas

No campus 1, representado pela figura 3.3, são apresentadas três partes principais: as zonas *Vlans*, o *backbone* interno, representado pelos *switche core e switch* de acesso e um *firewall*, possibilitando mais uma camada de proteção contra possíveis ataques. Além disso, há redundância em caso de falha de algum *link*. O cenário é semelhante ao apresentado no campus 2, composto pelo *Datacenter*, contendo as mesmas três partes.

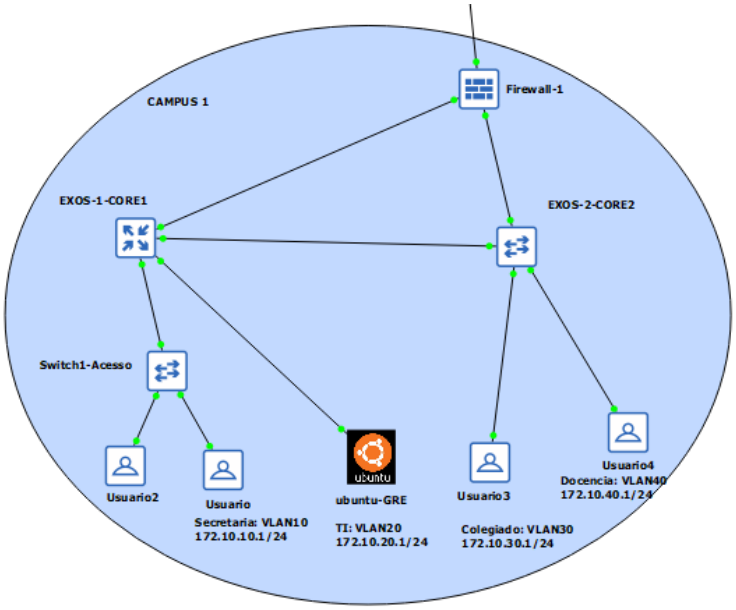

Figura 3.3: Apresentação detalhada do campus 1. Fonte: Autor

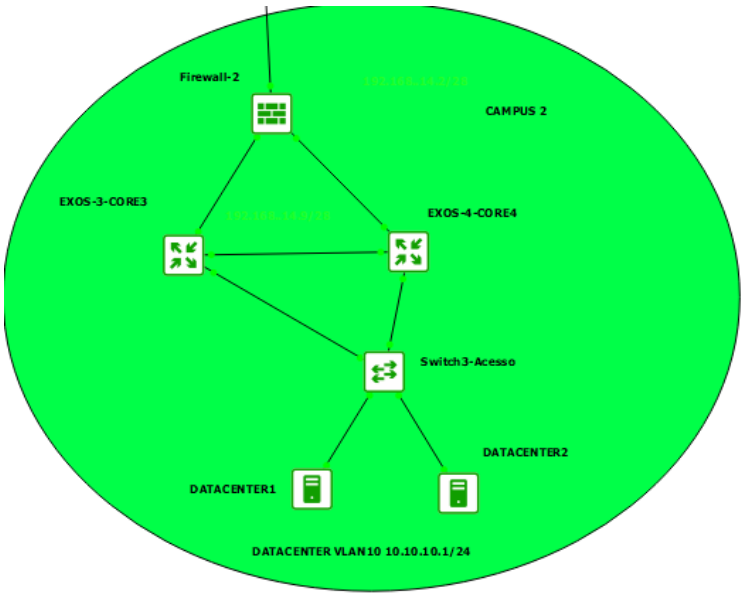

Figura 3.4: Apresentação detalhada do campus 2. Fonte: Autor

Já na terceira parte, *Edge*, composta por duas instâncias *Flexiwan Edges* e por um roteador *Vyos*, como

apresentado na figura 3.5:

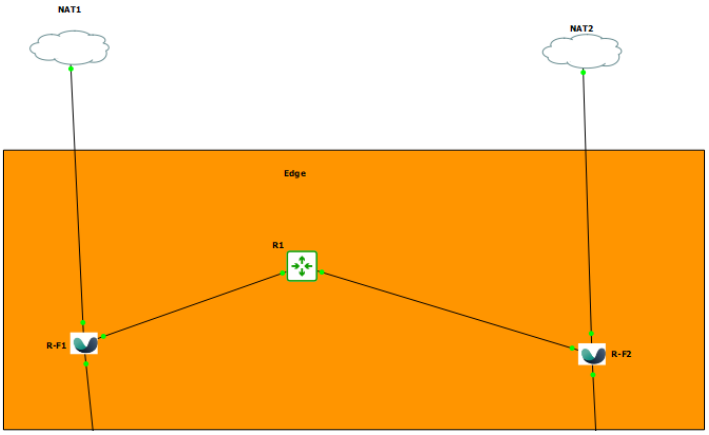

Figura 3.5: Apresentação detalhada do *edge*. Fonte: Autor

Ressalta-se que a gerência de todas as máquinas, no projeto, foi feita através da *Vlan* TI, localizada no campus 1, composta por um *Ubuntu* de versão 22.04.

3.5 DESCRIÇÃO DOS DISPOSITIVOS

No ambiente virtualizado em questão, para que a implementação do projeto fosse bem sucedida, fez-se necessário o uso de diversas ferramentas de código aberto, que serão abordadas nesta seção, bem como seus sistemas operacionais utilizados.

Optou-se pela utilização para o *Router* R1, o sistema *VyOS* como roteador, considerado uma das principais soluções *open source*, executado em uma ampla gama de *hardware*, desde roteadores de pequenos a grandes ambientes, como servidores ou até mesmo máquinas virtuais e vários provedores de nuvem [Vyos 2023]. No contexto em questão, o roteador foi utilizado como protocolo de redundância de internet, caso o provedor de algum *FlexiEdge* falhasse, ele poderia fornecer internet para ambos os *Edges*, constituindo parte da implementação de boas práticas.

Assim como o *VyOS*, os *Firewalls* do projeto, sendo compostos por sua totalidade de *Pfsenses*, são baseados em FreeBSD, e são uma das soluções mais utilizadas no mercado de código aberto. Apesar de contar com um sistema mais completo do que apenas *Firewalls*, neste projeto, ele foi utilizado com protocolos de roteamento dinâmico e bloqueio de alguns fluxos de pacote, que serão explicados posteriormente.Para os servidores de *Data Centers*, foi utilizada a distribuição *Ubuntu Server* .

Para a administração do *backbone*, como dito anteriormente, foi utilizado os *Switches EXOs*, que são de código aberto, sendo uma ferramenta poderosa, que, no âmbito deste projeto, foi configurado como *switch* de camada dois e três, também possuindo redundância, caso algum *link* falhasse.

Na utilização da ferramenta *SD-WAN*, foi utilizada a *Flexiwan*, solução de código aberto reconhecida no mercado. É uma ferramenta poderosa, onde são lançadas constantes atualizações de performance e novidades. Diferentemente da *Flexiwan Router*, o *Flexiwan Management* é utilizado através de uma plata-

forma *Web*, sendo acessível em qualquer local, incluindo dispositivos *mobiles*. A tabela 3.1 evidencia, em detalhes, as versões utilizadas para cada um dos sistemas implementados.

Tabela 3.1: Detalhamento das versões dos sistemas operacionais utilizados nesse trabalho

Sistema	**Fabricante**	**Versão**
VyOS	VyOS	1.3.0
Switch EXOs	Extreme	32.1.1.6
Pfsense	*Pfsense*	2.6.0
Ubuntu	Canonical Ltd	20.04
Ubuntu Server	Canonical Ltd	18.04
Flexiwan Edge	*Flexiwan*	5.3.20
Flexiwan Management	*Flexiwan*	5.3.14

3.6 PROTOCOLO DE ROTEAMENTO IMPLEMENTADO

O protocolo OSPF é feito para redes que possuem como base o protocolo IP. Há duas características principais no OSPF. A primeira, é ser um protocolo aberto, ou seja, suas especificações são de domínio público e podem ser encontradas na *Request for comments (RFC)*, número 1247. A segunda característica é ser um protocolo baseado no algoritmo *Short Path First (SPF)*, conhecido como algoritmo de Dijkstra [OSPF 2023].

Além disso, o OSPF é um protocolo de roteamento do tipo *link-state*, ou seja, envia avisos sobre o estado da conexão a todos os outros roteadores que estejam na mesma rede hierárquica. As informações sobre interfaces ligadas, métricas e outras variáveis são incluídas nas *link-state advertisements (LSAs)*. Desta forma, ao mesmo tempo que o roteador *OSPF* acumula informações sobre o estado do *link*, ele utiliza o algoritmo SPF para calcular a menor rota para cada nó.

Na proposta dessa implementação, os roteadores, os *firewalls* e os *switches* farão a comunicação entre si via, *OSPF*. O *Flexiwan Edge* fará o uso dessas informações, condensando em tabelas de roteamento. É válido ressaltar que essa mesma configuração poderia ter sido realizada via rotas estáticas, contudo, para a promoção de um ambiente automatizado, fez-se o uso de um protocolo de roteamento dinâmico.

3.7 DEFINIÇÃO DO ENDEREÇAMENTO

A tabela 3.2 descreve todo o endereçamento *IP* adotado no trabalho. Na topologia em questão, em relação à um ambiente real, são adotados poucos dispositivos, devido à quantidade limitada de recursos computacionais. Contudo, o endereçamento adotado, visa uma possível expansão de todas as áreas, seja para estudo ou para uma possível implementação física.

Tabela 3.2: Detalhamento do endereçamento *IP* adotado nesse trabalho. Fonte: Autor

Dispositivo	**Interface**	***IP***	***Vlan***
R1	eth1	192.168.18.20/24	-
	eth2	192.168.19.30/24	-
R-F1	eth0	WAN - DHCP	-
	eth1	192.168.1.1/24	-
	eth2	192.168.17.10/24	-
	eth3	192.168.18.10/24	-
R-F2	eth0	WAN - DHCP	-
	eth1	192.168.2.1/24	-
	eth2	192.168.19.10/24	-
	eth3	192.168.20.10/24	-
Firewall-1	eth0	192.168.17.20/24	-
	eth1	192.168.3.1/24	-
	eth2	192.168.16.2/24	-
	eth3	192.168.11.2/24	-
Firewall-2	eth0	192.168.19.20/24	-
	eth1	192.168.4.1/24	-
	eth2	192.168.14.2/24	-
	eth3	192.168.13.2/24	-
EXOS-1-CORE1	Secretaria	172.10.10.1/24	10
	TI	172.10.20.1/24	20
	Colegiado	172.10.30.1/24	30
	Docencia	172.10.40.1/24	40
	Default	192.168.16.4/24	1
EXOS-1-CORE2	Secretaria	172.10.10.1/24	10
	TI	172.10.20.1/24	20
	Colegiado	172.10.30.1/24	30
	Docencia	172.10.40.1/24	40
	Default	192.168.16.5/24	1
EXOS-3-CORE3	Datacenter	10.10.10.1/24	50
	Default	192.168.14.9/24	1
EXOS-3-CORE4	Datacenter	10.10.10.1/24	50
	Default	192.168.14.10/24	1

3.8 CONFIGURAÇÕES

3.8.1 Firewall

Nesse trabalho, foram instalados e gerenciados dois *firewalls* na organização, estabelecendo políticas e regras semelhantes. O *FlexiwanEdge*, o *Firewall-1* e *Firewall-2*, além de uma relação entre as *VLANs* e o *FlexiwanEdge* dos seus respectivos campus. Essas configurações, serão especificadas nas seções a seguir.

Para que o campus 1 e 2 funcionassem da maneira necessária para que a emulação fosse feita, algumas configurações no *pfsense* foram necessárias. Primeiramente, fez-se necessária a instalação do pacote *Free Range Routing (FRR)*, disponível nativamente no próprio *pfsense*. Através dessa ferramenta, é possível fazer a configuração do protocolo de roteamento dinâmico, o OSPF. No próprio pacote FRR, foi feita a

configuração das *subnets* da rede. Além dessa configuração, em *Rules*, foi liberado o protocolo IPv4 TCP na interface dos campus 1 e 2. Para concluir, foi configurado em *Gateways*, os *gateways* necessários de saída, bem como as rotas estáticas, utilizadas para as *VLANs*. Como o intuito desse trabalho é mostrar o funcionamento de uma rede *SD-WAN*, bem como as funcionalidades que o *SASE* pode nos fornecer, não foram criadas regras nos *firewalls* internos, que pudessem impedir algum tipo de pacote.

3.8.2 Switch EXOS

Como mostrado na seção anterior, o *switch* utilizado nesse projeto, foi o *switch EXOs*. No campus 1, o *switch EXOS1-CORE1* foi configurado como *switch* de camada três e o *switch EXOS2-CORE2* foi configurado como camada dois. Ainda foram definidos quatro *VLANs*, Secretaria, TI, Colegiado e Docência, com VLANs definidas, 10, 20, 30 e 40, respectivamente. Por padrão, o *EXOs* cria uma *VLAN Default* que cria um *link* de comunicação direto com o *pfsense*. Na figura 3.7, é possível ver essa configuração no *switch EXOS1-CORE1*, através da *prompt* de comando. Além disso, o *switch EXOs* possui uma interface gráfica, de fácil configuração, como podemos ver na figura 3.8, acessado através do IP da VLAN *Default*.

```
* Campus1.1 # show vlan
Untagged ports auto-move: Inform
-------------------------------------------------------------------------------
Name            VID  Protocol Addr       Flags                     Proto  Ports  Virtual
                                                                           Active router
                                                                           /Total
-------------------------------------------------------------------------------
Colegiado       30   172.10.30.1    /24  -f-------o----------------  ANY    1 /1   VR-Default
Default         1    192.168.16.4   /28  ------------T-------------  ANY    2 /10  VR-Default
Docencia        40   172.10.40.1    /24  -f-------o----------------  ANY    1 /1   VR-Default
Mgmt            4095 ----------------------------------------------  ANY    0 /1   VR-Mgmt
Secretaria      10   172.10.10.1    /24  -f-------o----------------  ANY    2 /2   VR-Default
TI              20   172.10.20.1    /24  -f-------o----------------  ANY    2 /2   VR-Default
-------------------------------------------------------------------------------
```

Figura 3.6: Configuração *EXOS1-CORE1* via *promt* de comando. Fonte: Autor

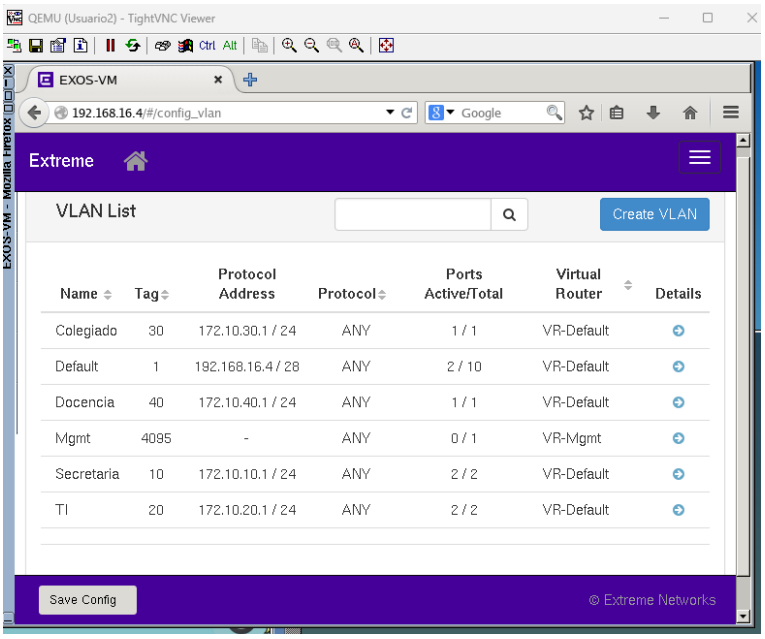

Figura 3.7: Configuração *EXOS1-CORE1* via interface gráfica. Fonte: Autor

No campus 2, foi definida a *VLAN* do *DATACENTER*. Na figura 3.9, é possível ver a instalação via *prompt* de comando do *switch EXOS3-CORE3* de camada três. Já na figura 3.10, é possível ver a instalação da interface gráfica.

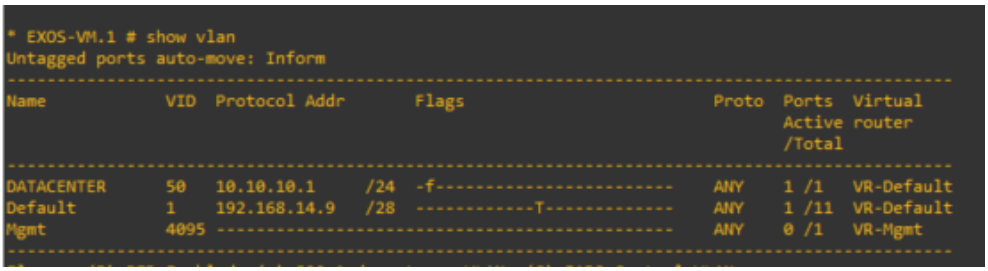

Figura 3.8: Configuração *EXOS3-CORE3* via interface gráfica. Fonte: Autor

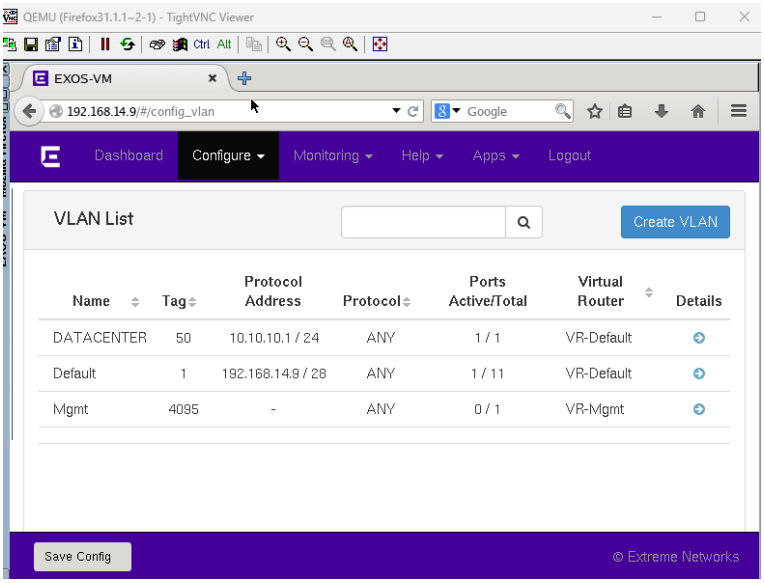

Figura 3.9: Configuração *EXOS3-CORE3* via interface gráfica. Fonte: Autor

Os *switches* de camada 2 funcionam como expansores de rede e redundância, desta forma, a informação é passada via *trunk* entre as portas dos *switches*. Para que houvesse a comunicação e identificação com a *SD-WAN*, foi habilitado o *OSPF*. Na figura 3.11 e 3.12, é possível observar o *OSPF* ativo nos *switches* de camada três, bem como o *RouterID*, que foram designados no *firewall-1* e *firewall-2*.

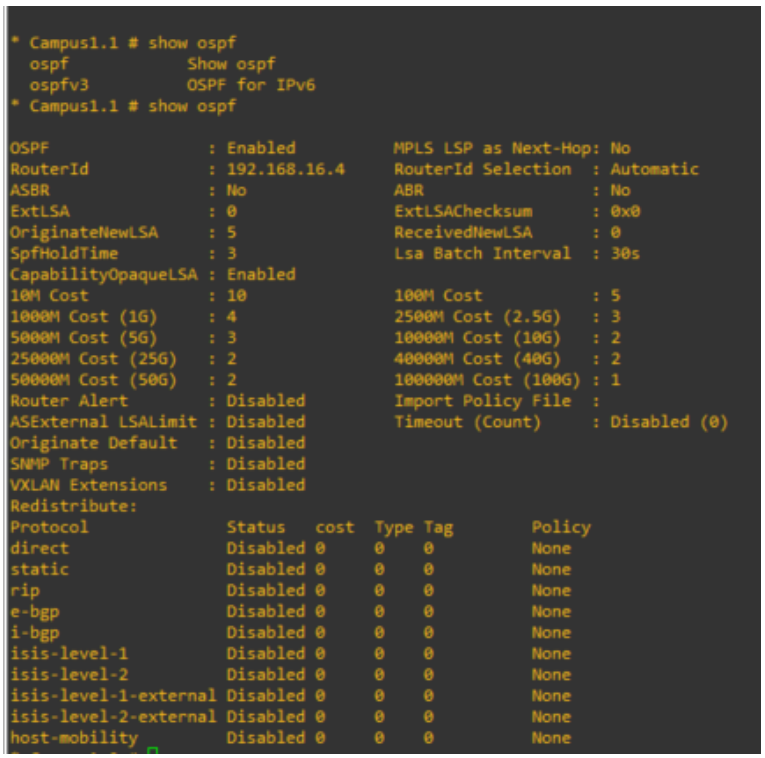

Figura 3.10: OSPF habilitado no *EXOS1-CORE1*. Fonte: Autor

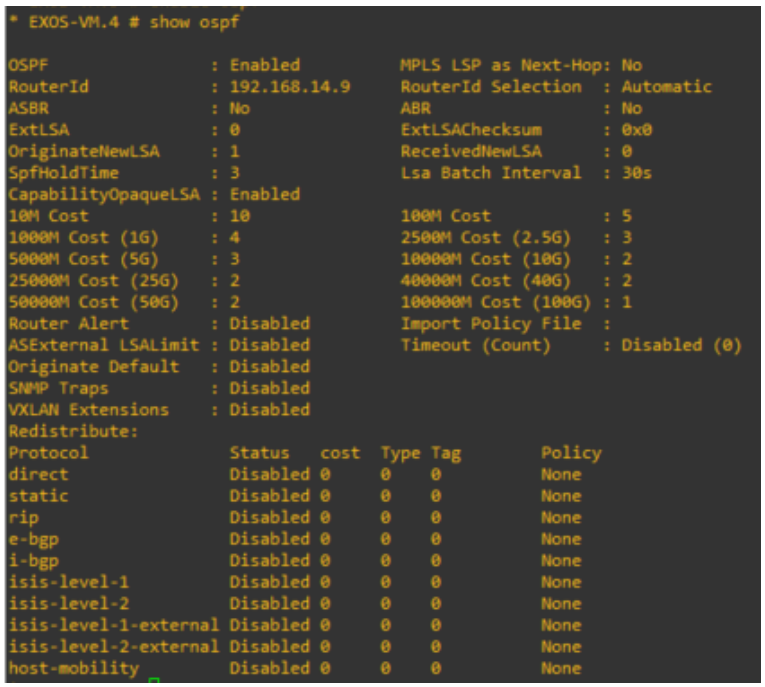

Figura 3.11: OSPF habilitado no *EXOS3-CORE3*. Fonte: Autor

Ambos os *switches* de camada três, possuem o *dhcp* ativado nas *VLANs*, ou seja, ao conectar um novo *endpoint*, ele já terá um IP criado automaticamente, alocado conforme o *range* estabelecido no momento da configuração da *VLAN*. Na figura 3.12 e 3.13, é possível observar a configuração feita para o *dhcp* em ambos os *switches*.

```
enable dhcp vlan Default
enable dhcp vlan Mgmt
configure vlan Colegiado dhcp-address-range 172.10.30.2 - 172.10.30.200
configure vlan Colegiado dhcp-options default-gateway 172.10.30.1
configure vlan Colegiado dhcp-options dns-server 8.8.8.8
enable dhcp ports 12 vlan Colegiado
configure vlan Docencia dhcp-address-range 172.10.40.2 - 172.10.40.200
configure vlan Docencia dhcp-options default-gateway 172.10.40.1
configure vlan Docencia dhcp-options dns-server 8.8.8.8
enable dhcp ports 12 vlan Docencia
configure vlan Secretaria dhcp-address-range 172.10.10.2 - 172.10.10.254
configure vlan Secretaria dhcp-options default-gateway 172.10.10.1
configure vlan Secretaria dhcp-options dns-server 8.8.8.8
enable dhcp ports 1,12 vlan Secretaria
configure vlan TI dhcp-address-range 172.10.20.2 - 172.10.20.200
configure vlan TI dhcp-options default-gateway 172.10.20.1
configure vlan TI dhcp-options dns-server 8.8.8.8
enable dhcp ports 2,12 vlan TI
```

Figura 3.12: *DHCP* configurado no *switch EXOS1-CORE1*. Fonte: Autor

```
enable dhcp vlan Default
configure vlan DATACENTER dhcp-address-range 10.10.10.2 - 10.10.10.200
configure vlan DATACENTER dhcp-options default-gateway 10.10.10.1
configure vlan DATACENTER dhcp-options dns-server 8.8.8.8
enable dhcp ports 3 vlan DATACENTER
```

Figura 3.13: *DHCP* configurado no *switch EXOS3-CORE3*. Fonte: Autor

3.8.3 *Flexiwan Edge*

Como mostrado na seção anterior, o *FlexiEdge* trabalha como *Flexiwan Agent*, que é o elemento de *software* que conecta o *flexiEdge* ao *FlexiManagement*. Foi feita a instalação de duas instâncias do *flexiEdge*, responsáveis pelos campus 1 e 2. Ao conectar na rede, é necessário rodar o *System Checker* para observar se foi instalada de forma correta. Feito isso, ao ir em *interfaces*, é possível observar quais portas estão ativas e se é em formato *DHCP* ou estático. A figura 3.15, apresenta um exemplo de interface ativa. Vale ressaltar que o IPv4 estático, é criado e gerenciado diretamente no *Flexiwan Management*, que será abordado posteriormente.

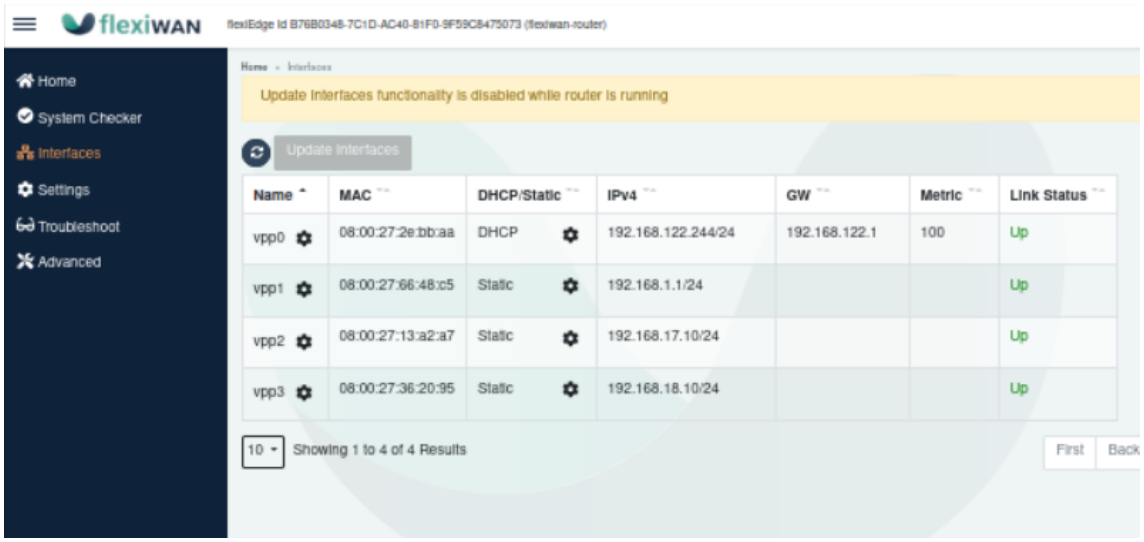

Figura 3.14: Exemplo de interface gráfica do *FlexiEdge (R-F1)*. Fonte: Autor

3.8.4 *Flexiwan Management*

Para que ocorra a sincronização entre o *FlexiwanEdge* e o *Flexiwan Management*, é necessário a criação de *tokens* para a vinculação do *FlexiwanEdge* na topologia *SD-WAN*. A criação é feita diretamente com a conta vinculada no *Flexiwan Management*, como pode-se observar na figura 3.16. É valido ressaltar que a quantidade de *tokens* criados de forma gratuita é limitada, sendo necessário fazer uma assinatura para cenários de redes maiores.

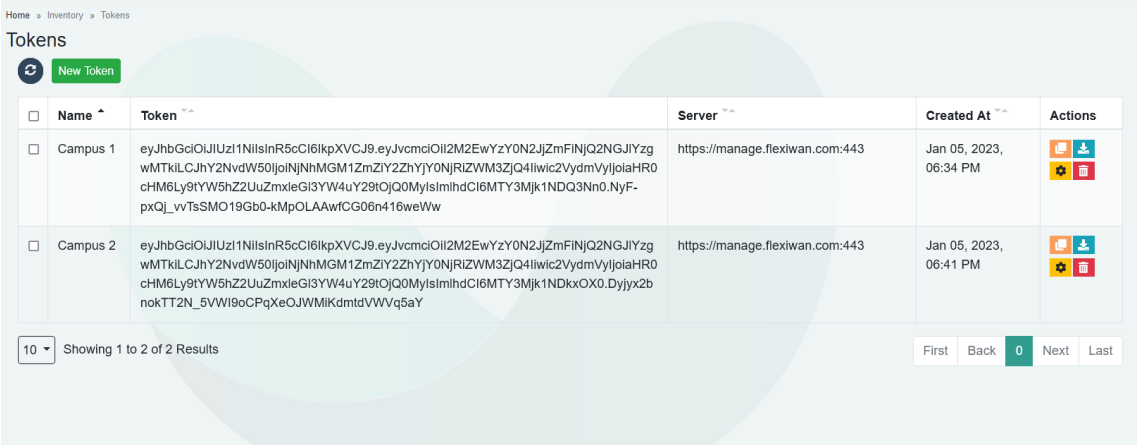

Figura 3.15: Interface do *Flexiwan Management* para a criação de *tokens*. Fonte: Autor

Após a instalação do *FlexiwanEdge*, o controle da *Flexiwan* é feito diretamente na interface gráfica do gerenciador *Flexiwan Management*. Nele, é possível a visualização e gerenciamento de toda a solução. A figura 3.16, permite a visualização de algumas informações relevantes, como, *status, hostname, WAN IP, ID*, PPS e BPS.

Figura 3.16: Interface do *Flexiwan Management* para a criação de *tokens*. Fonte: Autor

- *Status*: Mostra que a solução está aprovada, conectada e sincronizada em ambos os campus;
- *Hostname*: Designa o nome de *flexiwan-router*, que é o *flexiwan edge*;
- WAN IPs:É o endereço designado pela NAT;
- ID: Cada dispositivo *flexiwanEdge* possui um ID único, que serve para a identificação;
- PPS e BPS: Através do *Flexiwan Management*, é possível visualizar em tempo real a quantidade de bits por segundo e pacotes por segundo trafegando na rede, além da quantidade de pacotes sendo

transmitidos (TX) e recebidos (RX).

Existe o *path label*, que é responsável pelas chamadas etiquetas. Através das etiquetas, é oferecido uma maneira poderosa de organizar túneis do *Flexiwan*. Através delas, é possível que os usuários definam capacidades únicas de redes subjacentes, com políticas definidas com rótulos (nome lógicos) e não com interfaces, àquelas especificadas por dispositivo.

Com isso, a *Flexiwan* permite a criação de túneis, conectando todos os sites da organização. Quando um túnel é criado, os sites são conectados, criando automaticamente um *loopback* para cada site. Como nessa emulação é trabalhado com dois sites, criou-se automaticamente o *loopback* 10.100.0.4 para o campus 1, e o 10.100.0.5, para o campus 2. Além do loopback, outras métricas que são criadas, como o avg latency, drop rate, o tipo de encriptação, bem como o status da rede no momento. Ressalta-se que, a encriptação padrão no Flexiwan é a Pre-Shared Key (PSK).

A criação do *path label* é condicionado a interface WAN, desta forma, foi criado uma etiqueta chamada *IKEv2*, atrelado a interface eth0 de ambos os campus, que será usada para criar um túnel *IPSec* mais adiante, conforme mostrado na figura 3.17.

Figura 3.17: Criação do *path label*. Fonte: Autor

A plataforma de gerenciamento da *Flexiwan* é fácil e intuitiva, composta por algumas informações, entre elas, está a interface georeferencial das aplicações SD-WAN, sendo inteiramente responsivo. Como o projeto foi feito através de um simulador, a localização não está mostrada de forma correta, podendo ser modificada posteriormente. Na figura 3.18 e 3.19, é mostrado como a aplicação se apresenta.

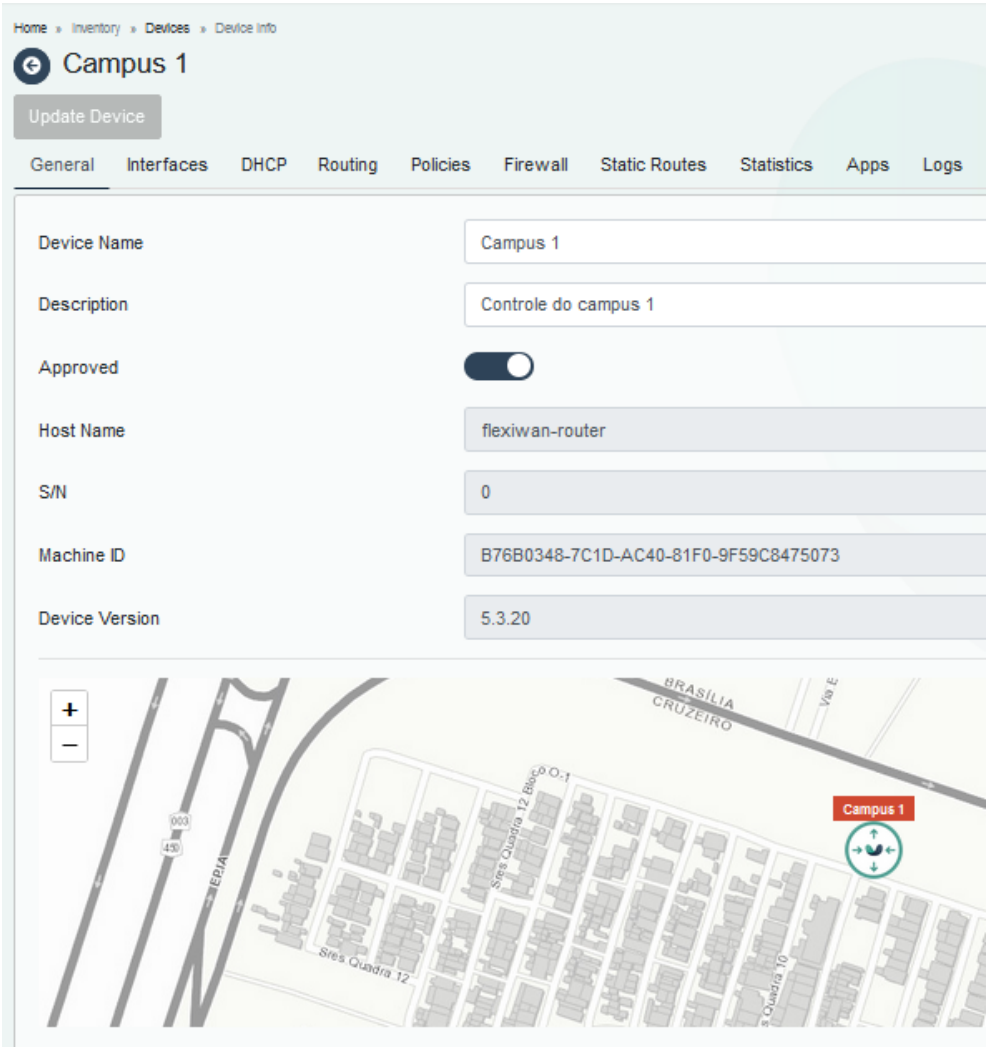

Figura 3.18: Gerenciamento do campus 1. Fonte: Autor

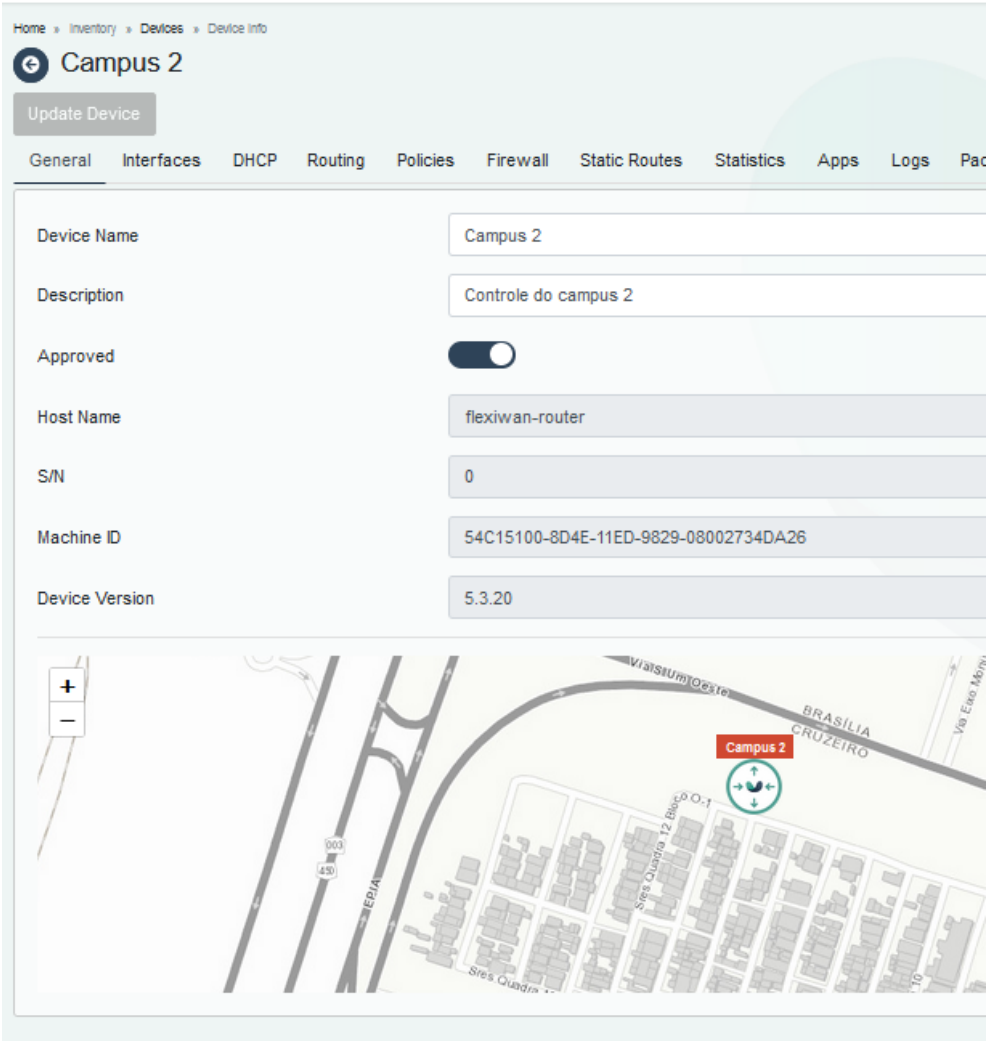

Figura 3.19: Gerenciamento do campus 2. Fonte: Autor

Através da tabela de roteamento de cada *flexiEdge*, é possível observar quais protocolos estão sendo utilizados, bem como seus *gateways*. Além disso, é possível ver o *loopback* em cada tabela de roteamento, bem como seu comportamento, como mostrado na figura 3.21 e 3.22.

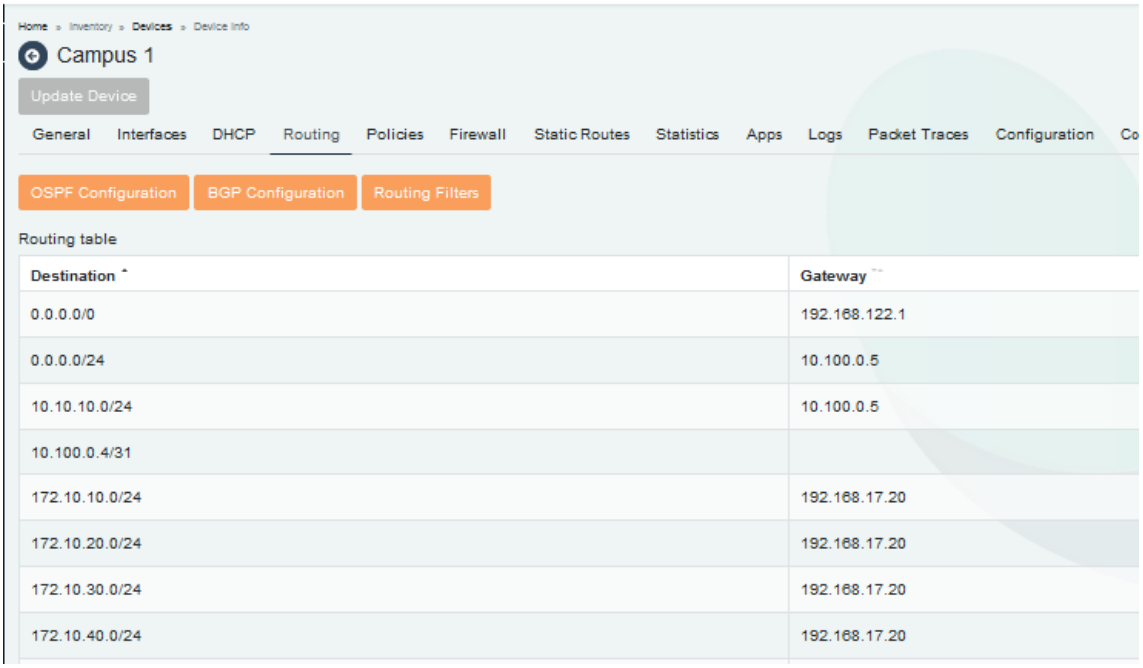

Figura 3.20: Parte da tabela de roteamento do campus 1. Fonte: Autor

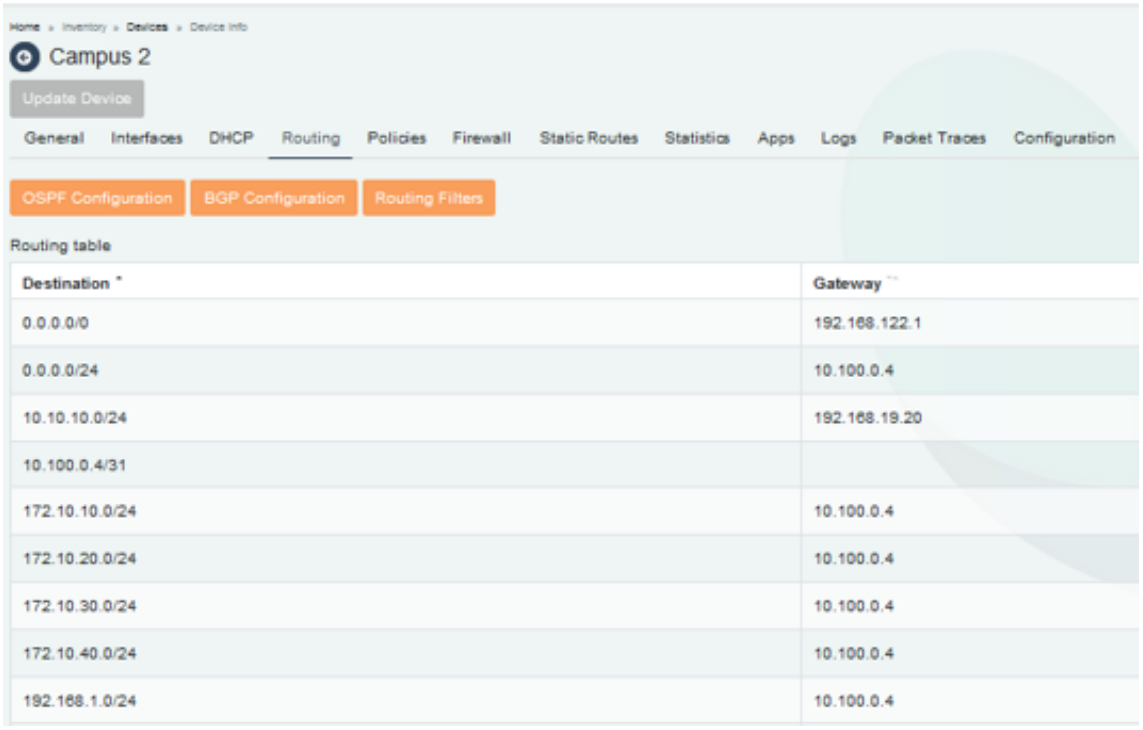

Figura 3.21: Para da tabela de roteamento do campus 2. Fonte: Autor

4 RESULTADOS E ANÁLISES

Como *Flexiwan* está voltada para segurança em *SD-WAN* e *SASE*, o *firewall* da *Flexiwan* oferece controles flexíveis para filtrar determinados ou todo o tráfego de rede, por vários critérios diferentes. Desta forma, o *firewall* da *Flexiwan*, combina três componentes principais:

- Políticas: Configura e implementa em vários sites *flexiwaEdges* de uma só vez;
- Regras específicas do dispositivo: Configura regras de *firewall* específicas de um *site* determinado;
- Identificação de tráfego e aplicativos: Banco de dados com faixas de endereços IP de serviços conhecidos e portas de aplicativos populares, sendo possível a adição de suas próprias identificações.

Neste trabalho, a primeira emulação feita, foi criar uma regra de *firewall* que bloqueia, no campus 1, dois aplicativos de redes sociais, conhecidos e utilizados mundialmente. No *Flexiwan*, existe duas formas de configurar o *firewall*. A primeira, ao ir diretamente no *site SD-WAN* que deseja fazer o bloqueio e executá-lo manualmente. Na figura 4.1, é possível observar a regra de *firewall* que bloqueia o uso desses aplicativos, de forma individual, no campus 1.

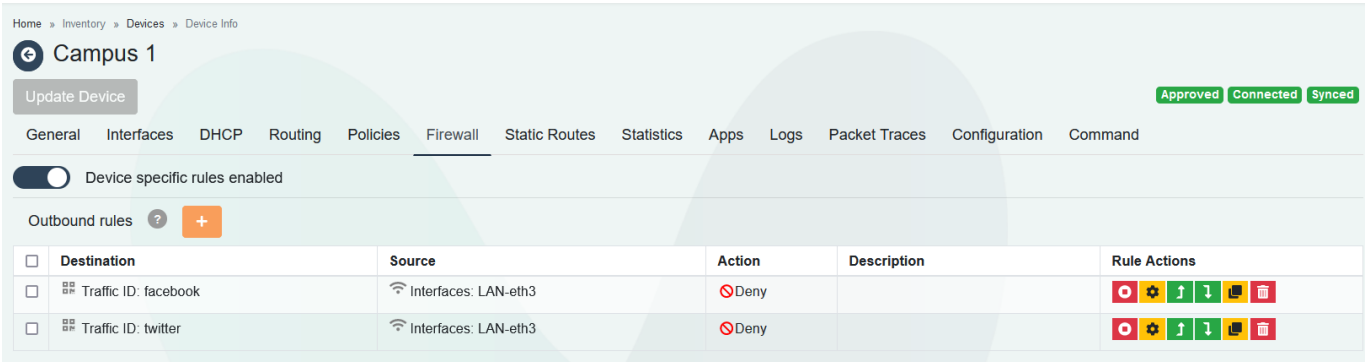

Figura 4.1: Exemplo de criação de regra de firewall de bloqueio no campus 1. Fonte: Autor

A *flexiwan*, trás por padrão, diversos IPs pré configurados com o objetivo de automatizar certas ações para o administrador de redes. A regra de bloqueio no exemplo, foi feita, através da opção *traffic name*. Nesta opção, a própria ferramenta *flexiwan* trás criado por padrão, uma lista de possíveis IPs ao fazer a comunicação com os servidores da aplicação e já deixa configurado para o administrador de rede, como pode ser visto na figura 4.2 e 4.3.

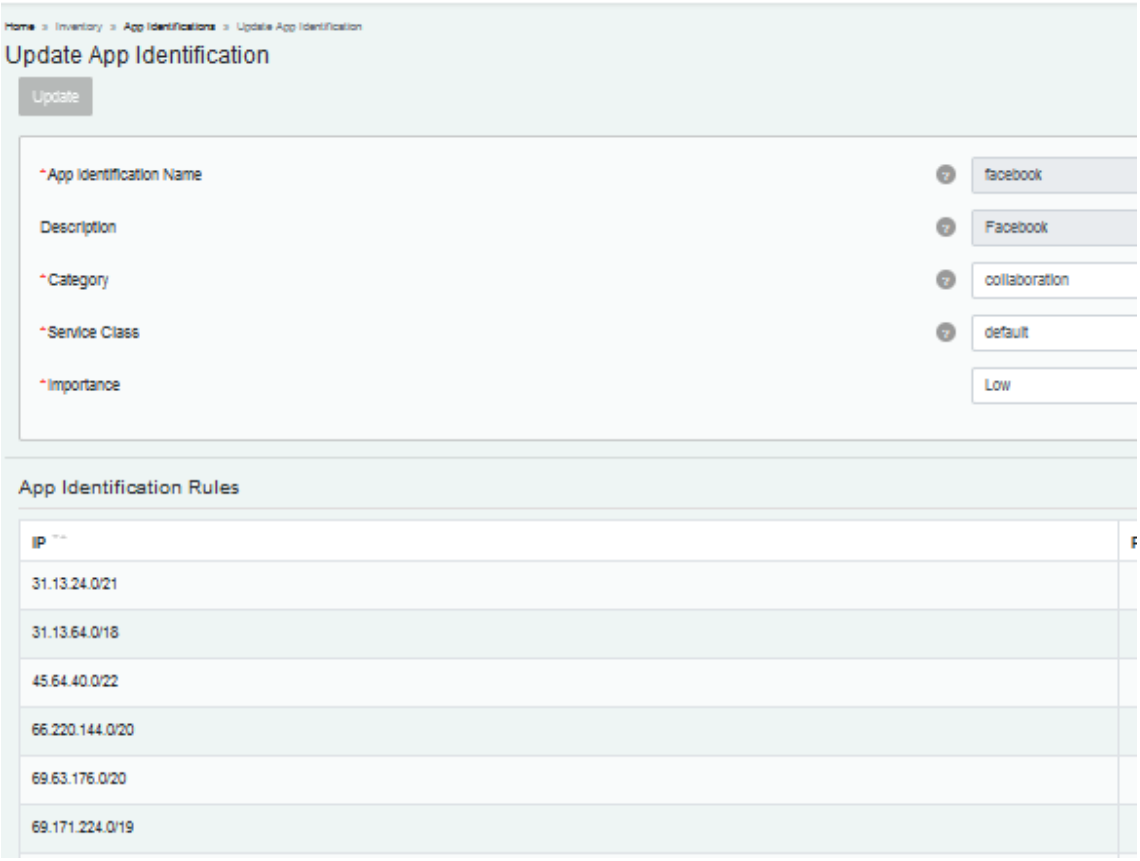

Figura 4.2: Exemplo de lista de IPs da plataforma de mídia social. Fonte: Autor

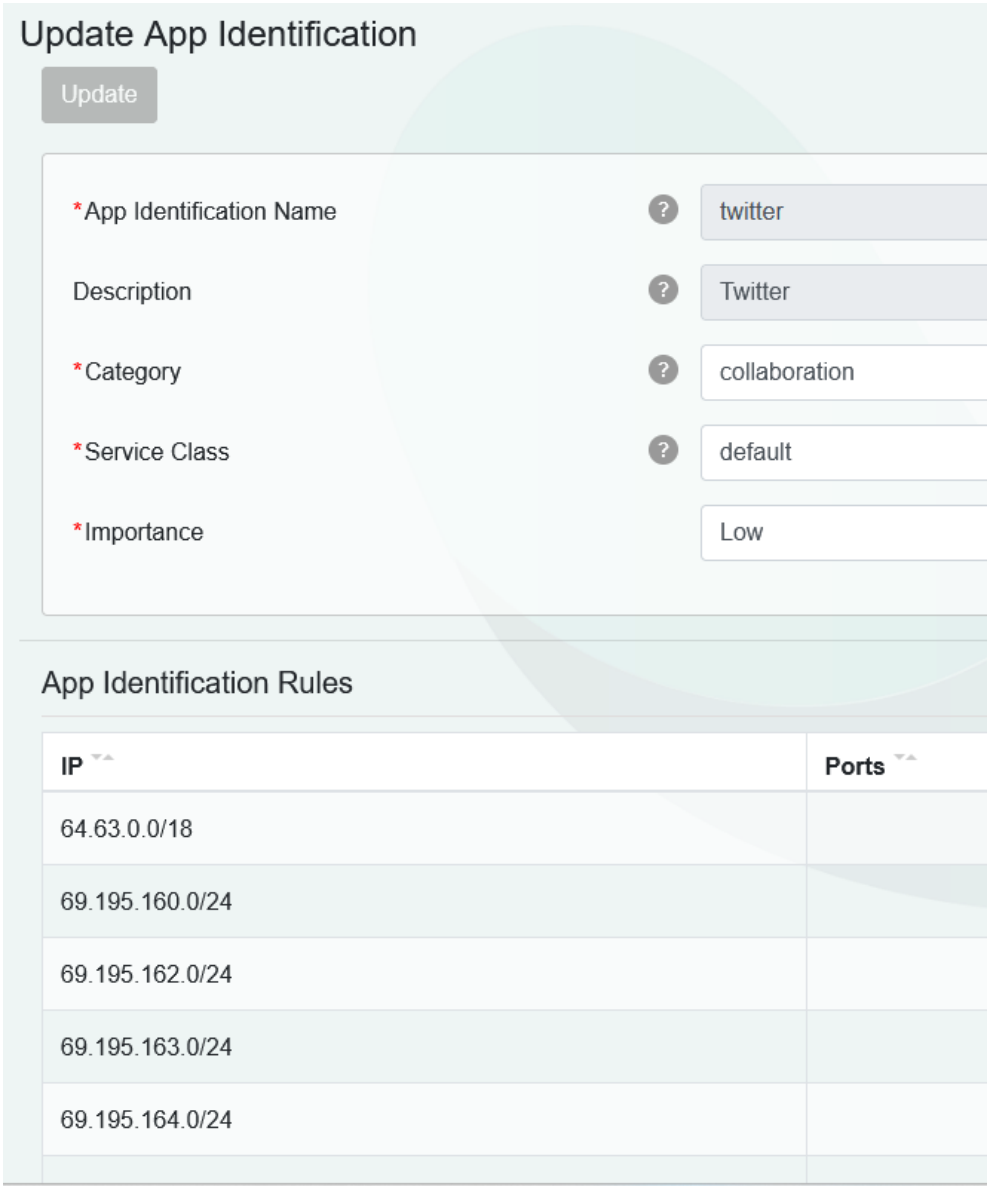

Figura 4.3: Exemplo de lista de IPs da plataforma de mídia social. Fonte: Autor

Além de possuir uma plataforma repleta de aplicações já pré instaladas, a *flexiwan* permite segregar pela importância, categoria e o tipo de classe de serviço, que essa aplicação possui para a organização.

A outra forma de conseguir criar regras, é criar uma regra geral, podendo ser referenciada para os outros *sites* que precisem utilizar a mesma regra futuramente. Essa forma é feita, através da opção *Firewall Policies*. Nela, você cria a política que vai ser utilizada para todos ou alguns *sites* apenas uma vez, e referencia essa regra, indo diretamente no *firewall* desses *sites*, referenciado pela figura 4.4

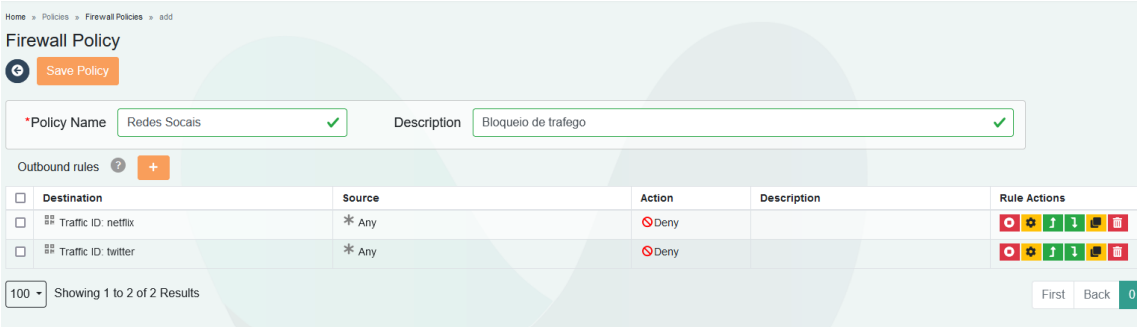

Figura 4.4: Criação de regra de bloqueio pelo *Firewall Policies*. Fonte: Autor

Ao ir em *devices*, é possível atribuir essa regra para o *site* necessário. Na figura 4.5, foi criado uma regra de bloqueio e colocado nos campus 1 e 2.

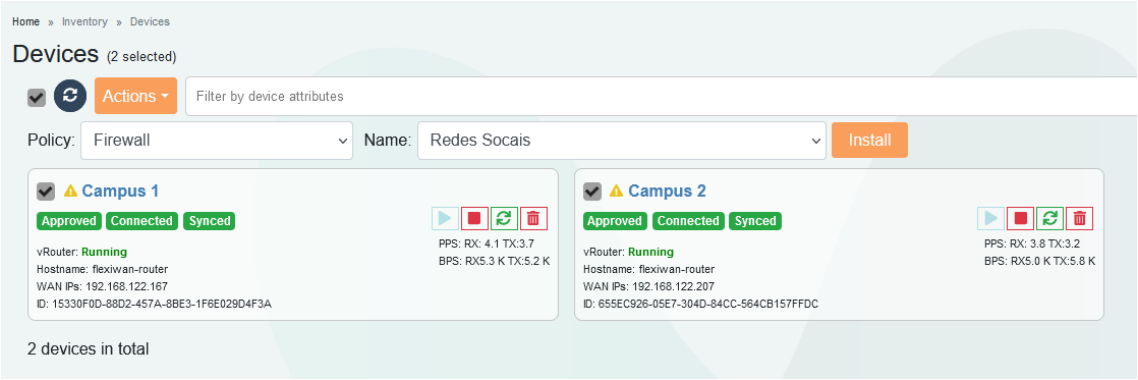

Figura 4.5: Criação de regra de bloqueio pelo *Firewall Policies* nos campus 1 e 2. Fonte: Autor

Para comprovar que a política de bloqueio surtiu efeito, foi usado o comando *tcpdump*. Esse comando é um farejador de pacotes de linha de comando, usado para capturar ou filtrar pacotes *tcp/ip* que são recebidos ou transferidos por uma rede em uma interface específica, estando disponível na maioria dos sistemas operacionais baseados em *linux / unix*. [tecmint 2023]. Para que fosse comprovado que a regra de bloqueio no *Flexiwan* estava funcionando, foi dividido em duas etapas. Para a primeira etapa, foi liberado o tráfego de uma das mídias sociais, com isso, é possível observar, através do comando *tcpdump*, a troca de sinalização entre o ambiente e o aplicativo em questão, mostrado na figura 4.6.

```
20:13:22.201494 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 94, length 64
20:13:22.233695 IP edge-star-mini-shv-01-gig2.facebook.com > 172-10-20-2.lightspeed.clmasc.sbcglobal.net: ICMP echo reply, id 1, seq 94, length 64
20:13:23.202984 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 95, length 64
20:13:23.235795 IP edge-star-mini-shv-01-gig2.facebook.com > 172-10-20-2.lightspeed.clmasc.sbcglobal.net: ICMP echo reply, id 1, seq 95, length 64
20:13:24.206654 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 96, length 64
20:13:24.238955 IP edge-star-mini-shv-01-gig2.facebook.com > 172-10-20-2.lightspeed.clmasc.sbcglobal.net: ICMP echo reply, id 1, seq 96, length 64
20:13:25.207924 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 97, length 64
20:13:25.238904 IP edge-star-mini-shv-01-gig2.facebook.com > 172-10-20-2.lightspeed.clmasc.sbcglobal.net: ICMP echo reply, id 1, seq 97, length 64
20:13:26.209459 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 98, length 64
20:13:26.243053 IP edge-star-mini-shv-01-gig2.facebook.com > 172-10-20-2.lightspeed.clmasc.sbcglobal.net: ICMP echo reply, id 1, seq 98, length 64
20:13:27.210720 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 99, length 64
20:13:27.241642 IP edge-star-mini-shv-01-gig2.facebook.com > 172-10-20-2.lightspeed.clmasc.sbcglobal.net: ICMP echo reply, id 1, seq 99, length 64
20:13:28.211802 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 100, length 64
20:13:28.239939 IP edge-star-mini-shv-01-gig2.facebook.com > 172-10-20-2.lightspeed.clmasc.sbcglobal.net: ICMP echo reply, id 1, seq 100, length 64
20:13:28.355706 ARP, Request who-has 172-10-20-1.lightspeed.clmasc.sbcglobal.net tell 172-10-20-2.lightspeed.clmasc.sbcglobal.net, length 28
```

Figura 4.6: Primeira etapa, sem bloqueio. Fonte: Autor

Na figura 4.6, é possível observar algumas informações. Primeiramente, ela mostra em ordem, o horário em que está ocorrendo a sinalização das requisições, logo em seguida, mostra qual dispositivo está

tentando encaminhar essas informações, que neste caso, é o endereço IP 172.10.20.2, que é o endereço da *VLAN*, que usa um registro *PTR Records*, sinalizado pelo *lightspeed.clmasc.sbcglobal.net*, que tem como objetivo de verificar se o remetente corresponde ao endereço IP que afirma estar usando, utilizado como *DNS reverso*. A mensagem é encaminhada para o endereço *edge-star-mini-shv-01.gig2.facebook.com*, que também funciona como o endereço público, que se utilizada do protocolo *ICMP*. Desta forma, é possível observar a troca de mensagens, sinalizando que o tráfego está ocorrendo normalmente.

Na segunda etapa, o bloqueio é feito no *flexiwan manage*, devido a isso, há uma quebra na comunicação entre o ambiente emulado e o aplicativo. Conforme mostrado na figura 4.7, há diversas requisições por parte do ambiente interno, sinalizado pelo 172.10.20.2, mas sem resposta, por parte do *site*, devido ao bloqueio feito.

```
20:14:07.290331 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net.41110 > dns.google.domain: 53718+ PTR? b.f.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.0.f.f.ip6.arpa. (90)
20:14:07.523765 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 139, length 64
20:14:08.547662 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 140, length 64
20:14:09.572304 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 141, length 64
20:14:10.595881 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 142, length 64
20:14:11.322309 IP 172-10-20-1.lightspeed.clmasc.sbcglobal.net > ospf-all.mcast.net: OSPFv2, Hello, length 44
20:14:11.619707 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 143, length 64
20:14:11.994535 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net.42229 > dns.google.domain: 57471+ [1au] PTR? 251.0.0.224.in-addr.arpa. (53)
20:14:12.035943 IP dns.google.domain > 172-10-20-2.lightspeed.clmasc.sbcglobal.net.42229: 57471 NXDomain 0/1/1 (110)
20:14:12.036015 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net.42229 > dns.google.domain: 57471+ PTR? 251.0.0.224.in-addr.arpa. (42)
20:14:12.494387 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net.50058 > dns.google.domain: 53718+ [1au] PTR? b.f.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.0.f.f.ip6.arpa. (101)
20:14:12.536181 IP dns.google.domain > 172-10-20-2.lightspeed.clmasc.sbcglobal.net.50058: 53718 NXDomain 0/1/1 (165)
20:14:12.536275 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net.50058 > dns.google.domain: 53718+ PTR? b.f.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.0.f.f.ip6.arpa. (90)
20:14:12.644023 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 144, length 64
20:14:13.668031 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 145, length 64
20:14:14.691765 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 146, length 64
20:14:15.715990 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > edge-star-mini-shv-01-gig2.facebook.com: ICMP echo request, id 1, seq 147, length 64
```

Figura 4.7: Segunda etapa, com bloqueio. Fonte: Autor

A mesma lógica é usada para confirmar que a solução está em pleno funcionamento, entretanto, para essa segunda aplicação, iniciou-se primeiramente com o bloqueio da aplicação, e posteriormente, a liberação do tráfego demonstrado pelas figuras 4.8 e 4.9.

```
21:16:43.907665 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 116, length 64
21:16:44.931841 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 117, length 64
21:16:45.955898 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 118, length 64
21:16:46.979824 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 119, length 64
21:16:48.003843 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 120, length 64
21:16:49.028014 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 121, length 64
21:16:50.051867 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 122, length 64
21:16:51.075901 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 123, length 64
21:16:52.101765 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 124, length 64
21:16:52.562329 IP 172-10-20-1.lightspeed.clmasc.sbcglobal.net > ospf-all.mcast.net: OSPFv2, Hello, length 44
21:16:53.123772 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 125, length 64
```

Figura 4.8: Primeira etapa, com bloqueio. Fonte: Autor

Na imagem é possível observar diversas requisições feitas pelo endereço 172.10.20.2, porém sem resposta, mostrando que a política feita na controladora *Flexiwan* está ativa. Ao liberar o tráfego, é possível observar a troca de mensagens entre os IPs 172.10.20.2 e o 104.244.42.1, sinalizado pelo *site* em questão.

```
21:17:13.108956 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 127, length 64
21:17:13.255280 IP 104.244.42.1 > 172-10-20-2.lightspeed.clmasc.sbcglobal.net: ICMP echo reply, id 7, seq 127, length 64
21:17:13.494357 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net.40114 > dns.google.domain: 45527+ [1au] PTR? 1.42.244.104.in-addr.arpa. (54)
21:17:16.603743 IP dns.google.domain > 172-10-20-2.lightspeed.clmasc.sbcglobal.net.40114: 45527 ServFail 0/0/1 (54)
21:17:16.603831 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net.40114 > dns.google.domain: 45527+ [1au] PTR? 1.42.244.104.in-addr.arpa. (54)
21:17:18.115627 ARP, Request who-has 172-10-20-1.lightspeed.clmasc.sbcglobal.net tell 172-10-20-2.lightspeed.clmasc.sbcglobal.net, length 28
21:17:18.116459 ARP, Reply 172-10-20-1.lightspeed.clmasc.sbcglobal.net is-at 08:00:27:8c:10:77 (oui Unknown), length 46
21:17:21.636233 IP dns.google.domain > 172-10-20-2.lightspeed.clmasc.sbcglobal.net.40114: 45527 ServFail 0/0/1 (54)
21:17:21.636382 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net.40114 > dns.google.domain: 45527+ PTR? 1.42.244.104.in-addr.arpa. (43)
21:17:21.672704 IP dns.google.domain > 172-10-20-2.lightspeed.clmasc.sbcglobal.net.40114: 45527 ServFail 0/0/0 (43)
21:17:21.672962 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net > 104.244.42.1: ICMP echo request, id 7, seq 128, length 64
21:17:21.817909 IP 104.244.42.1 > 172-10-20-2.lightspeed.clmasc.sbcglobal.net: ICMP echo reply, id 7, seq 128, length 64
21:17:21.818213 IP 172-10-20-2.lightspeed.clmasc.sbcglobal.net.34314 > dns.google.domain: 55865+ [1au] PTR? 1.42.244.104.in-addr.arpa. (54)
```

Figura 4.9: Segunda etapa, sem bloqueio. Fonte: Autor

Desta forma, ficou constatado que, as regras impostas pela controladora *Flexiwan* estão em pleno funcionamento, podendo essas regras, serem aplicadas para diversas outras aplicações por meio do *traffic name*, ou outra que o administrador de redes desejar.

A segunda emulação feita, foi a criação da VPN da *flexiwan* para o campus 2, na *VLAN* Datacenter. A criação da VPN se dá de forma nativa. Em *App Store*, no *flexiwan management*, é possível fazer a instalação do *remote worker VPN configuration*, como é possível observar na figura 4.10.

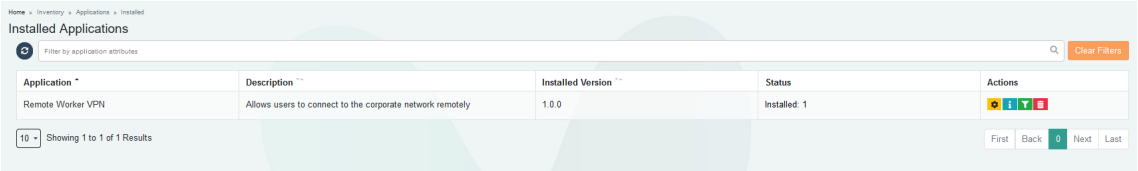

Figura 4.10: Instalação do *remote worker VPN*. Fonte: Autor

Ao fazer a sua configuração, o *remote worker VPN* trás alguns campos a serem preenchidos, como o nome do *workspace*, o *link* de acesso, a quantidade máxima de acessos que aquela VPN terá direito, a porta de acesso e o tipo de autenticação, como é observado na figura 4.13.

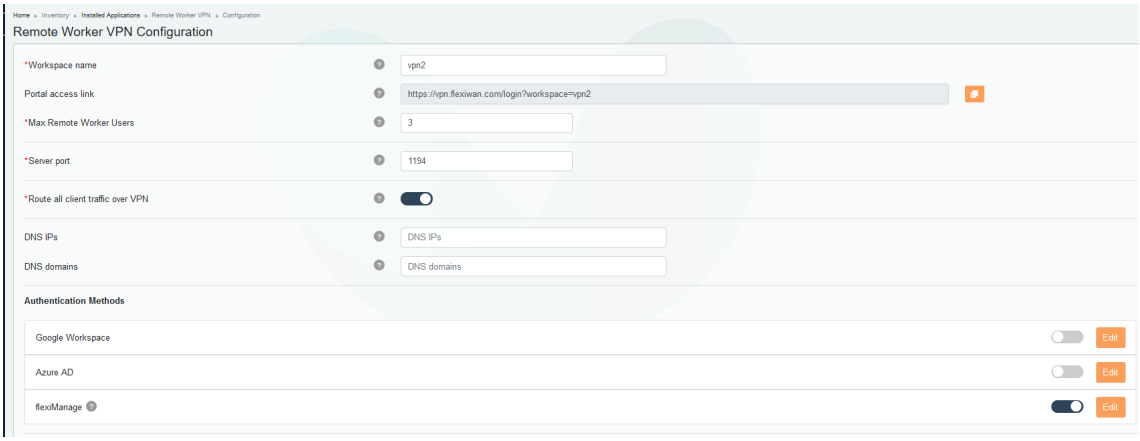

Figura 4.11: Página de configuração do *Remote Worker VPN configuration*. Fonte: Autor

No *authentication methods*, é possível obter o *link* para o *download* do cliente, conforme mostrado na figura 4.8. É valido ressaltar que, ainda não há suporte de *download* para sistemas operacionais *Linux*, sendo suportados apenas para *Windows* e *Mac OS*.

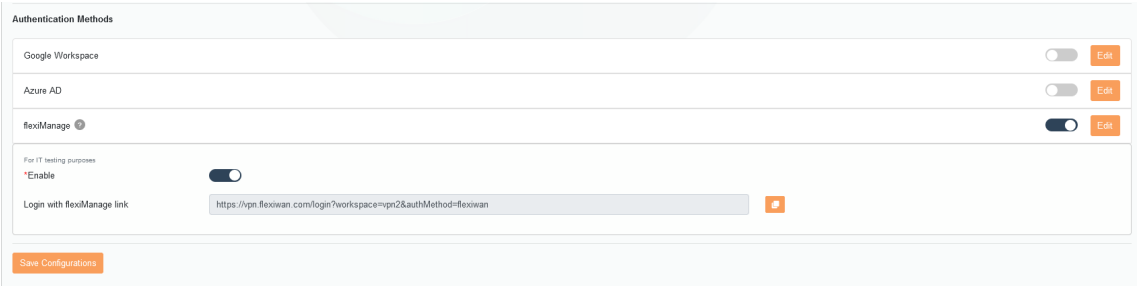

Figura 4.12: Página de acesso ao *link* para *download* do cliente. Fonte: Autor

Ao colocar o *link* no navegador, ele será direcionado para uma página de autenticação *flexiwan*, onde apenas aquele usuário autorizado e autenticado dentro da ferramenta poderá fazer o uso de acesso a VPN. Ainda na ferramenta do *remote worker VPN* é possível ver o seu *status*, para ver se a aplicação está funcionando de forma correta, como observado na figura 4.13.

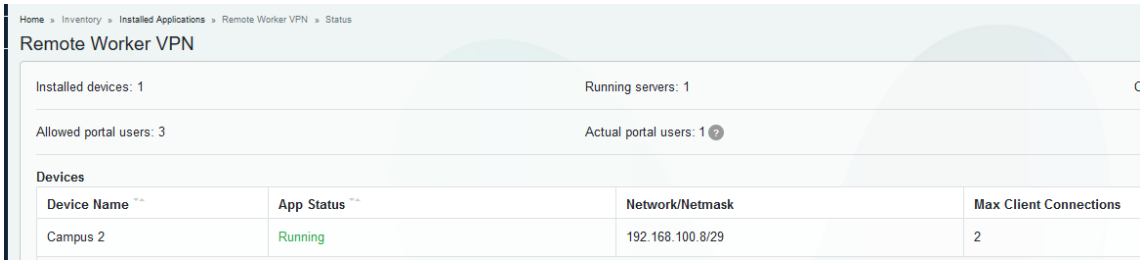

Figura 4.13: *status* da aplicação *remote worker VPN*. Fonte: Autor

Além dessa visualização, é possível observar na tabela de roteamento do campus 2, a *network/netmask* criada, e também, a ferramenta cria automaticamente uma regra de *firewall inbound* para acesso da aplicação, como mostrado na figura 4.14 e 4.15

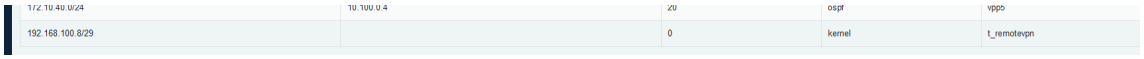

Figura 4.14: Tabela de roteamento do campus 2 em funcionamento com a VPN. Fonte: Autor

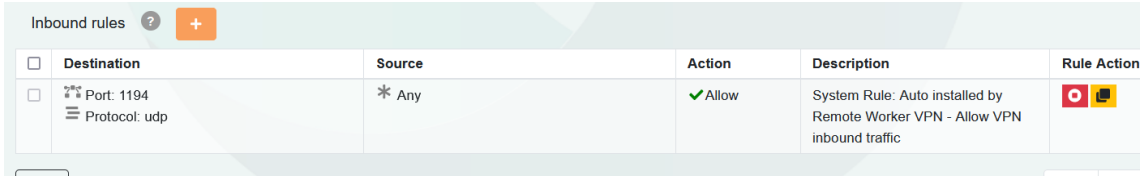

Figura 4.15: Regra criada automaticamente pela solução. Fonte: Autor

Para simular o funcionamento da *VPN* entre os campus, foi instalado uma máquina virtual *Windows* 10 no campus 1, com o objetivo de baixar e executar o cliente *VPN*. Na imagem 4.16, é possível ver a instalação executada com sucesso entre os campus 1 e 2, através de uma plataforma muito utilizada, chamada de *OpenVPN*.

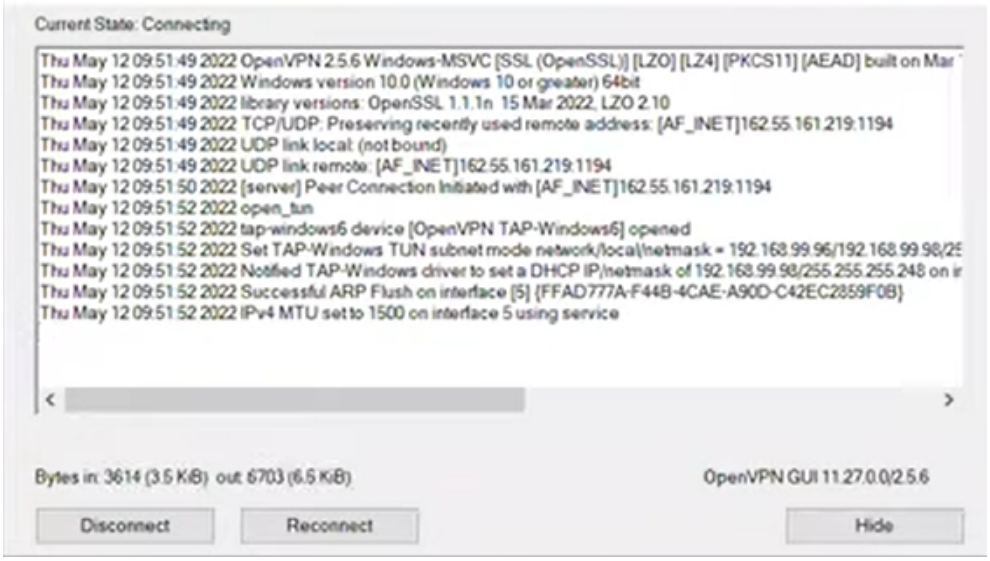

Figura 4.16: Aplicação VPN em execução. Fonte: Autor

Uma outra ferramenta muito comum, é a implementação do *IPSec* em *SD-WAN*. O *IPSec* implementa uma forma de tunelamento na camada de rede, além de fornecer autenticação em nível de rede, verificação da integridade dos dados e transmissão com criptografia e chaves de 128 bits, melhorando a segurança na transmissão das informações [UFRJ 2023].

Na ferramenta *Flexiwan*, o *IPSec* é instalado seguindo alguns passos. Primeiramente, ao ir em *organizations*, é necessário mudar o campo *Tunnels Key Exchange Method* para o padrão de encriptação *IPSec*, chamado de *IKEV2*. O *Internet Key Exchange* (*IKEV2*), é um protocolo baseado no *IPSec*, fornecendo um canal de comunicação *VPN* seguro entre dispositivos, definindo a negociação e autenticação para associações de segurança de maneira protegida [Juniper 2023]. Através da figura 4.17, é mostrado a alteração do método.

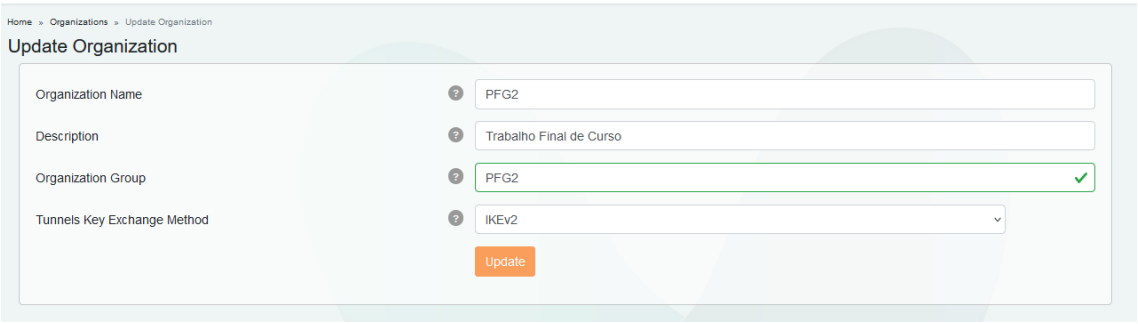

Figura 4.17: Alteração do método de encriptação. Fonte: Autor

Em *path labels*, como dito anteriormente, será adicionado mais uma etiqueta chamada *IKEv2*, em laranja, sinalizada em sua descrição como *IPSec*, designadas para a vinculação das interfaces *WANs*.

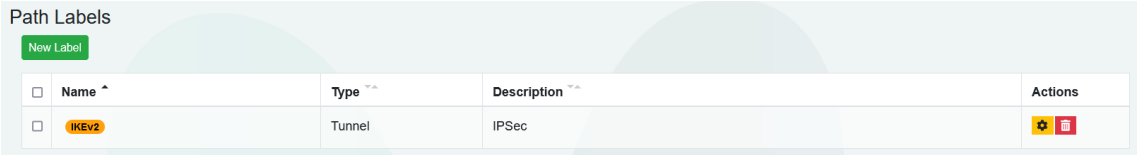

Figura 4.18: Adição do *path label*. Fonte: Autor

Essa etiqueta foi vinculada com as interfaces *WANs* de ambos os sites, campus 1 e campus 2. Ao selecionar ambos os *sites*, ir na opção *actions*, e *create tunnels*, o túnel *IPSec* será criado, como mostrado na figura 3.17.

Em *words maps*, temos a visão geográfica dos *sites*, como é observado na figura 4.19

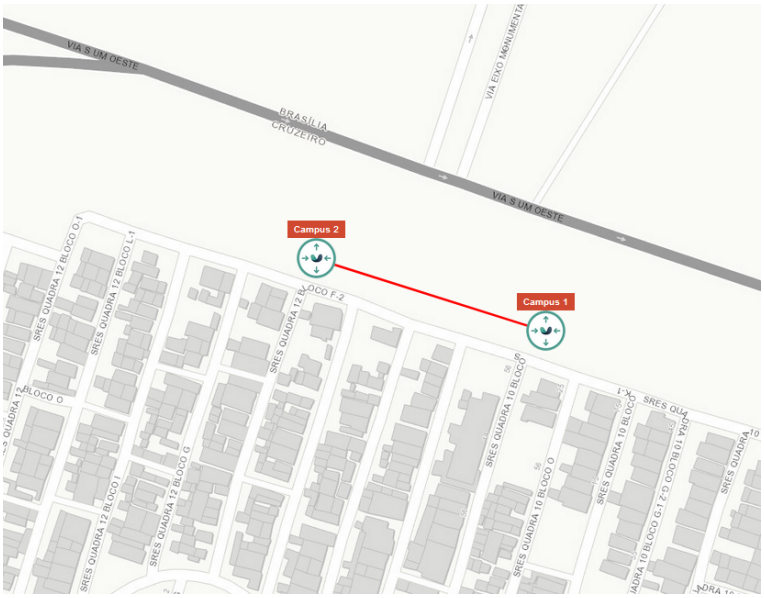

Figura 4.19: Visão geográfica dos *sites*. Fonte: Autor

Para que seja possível comprovar que o túnel *IPSec* está em pleno funcionamento, foi utilizado o *Wireshark* para visualizar a distribuição dos pacotes entre o túnel. O *Wireshark* é um analisador de protocolo de redes, que permite a visualização do que está acontecendo na rede analisada. Desta forma, ao analisar as trocas nos túneis entre os campus, é possível observar os protocolos *ESP (Encapsulating Security Payload* e o *ISAKMP (Internet Security Association e Key Management Protocol)*. O protocolo *ESP*, é projetado para fornecer uma combinação de serviços de segurança em IPv4 e IPv6, oferecendo confidencialidade e integridade para os dados. O *ESP*, é um membro do protocolo *IPSec*, que criptografa e autentica os pacotes de dados entre computadores. O *SPI (Security Parameter Index)*, corresponde em um número utilizado na identificação de uma associação de segurança (AS). Já o *ISAKMP*, define procedimentos e formatos de pacotes para estabelecer, negociar, modificar e excluir associações de segurança (AS), desta forma, o *ISAKMP*, destina-se a apoiar a negociação de SAs para segurança em todas as camadas da pilha de rede, por exemplo, o *IPSec*.

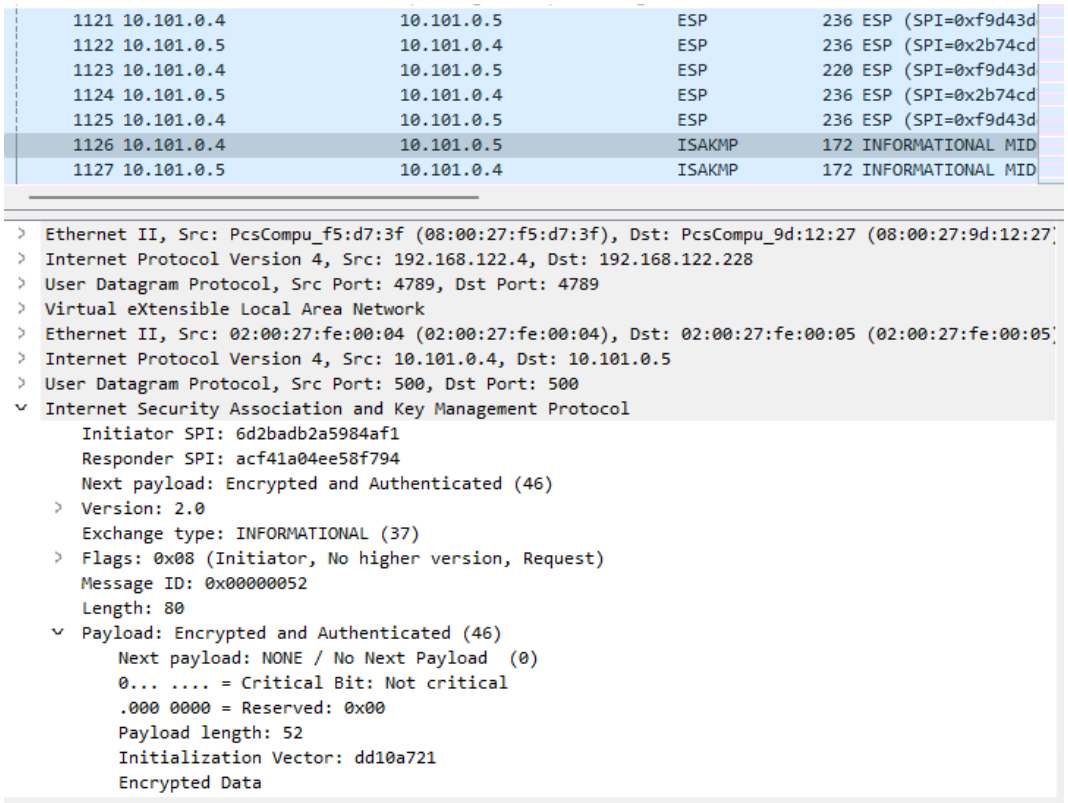

Figura 4.20: Visão via *wireshark* dos pacotes trocados. Fonte: Autor

Através do *Flow Graph*, é possível mostrar as conexões entre hospedeiros. Ele exibe o tempo do pacote, direção, portas e comentários para cada conexão capturada. Na figura 4.19, é possível observar o *Flow Graph* da conexão *IPSec* entre os campus.

Figura 4.21: Visão via *wireshark* do *flow graph*. Fonte: Autor

5 CONCLUSÃO

O projeto implementado nessa proposta, adiciona sistemas, ferramentas e aplicações capazes de emular o ambiente de uma possível organização real em um laboratório local. Através de diversas configurações, foi possível configurar um ambiente composto por alta disponibilidade, mantendo o funcionamento das aplicações, assim como o funcionamento de ferramentas de código aberto, ainda não trabalhadas em conjunto, no cenário acadêmico.

Através do *GNS3* e sua integração com o *VirtualBox*, foi possível a orquestração de múltiplos dispositivos e sistemas operacionais, em um mesmo ambiente virtualizado. O foco deste estudo, foi a implementação de uma rede *SD-WAN*, utilizando-se em sua totalidade, ferramentas de código aberto, objetivando o estudo de métricas, funcionalidades e comportamentos da solução *Flexiwan* em um cenário composto por ferramentas como *Switch EXOs* e *Pfsense*.

Por fim, através das análises da solução realizada, foi possível mostrar e comprovar, a perfeita integração entre as ferramentas de código aberto, bem como algumas funcionalidades presentes em soluções *SD-WAN* de código aberto e de soluções privadas. É possível observar, que para grandes redes, que necessitem de uma grande variedades de *sites*, a solução *SD-WAN* se faz necessário, trazendo diversos benefícios, como redução de custos, gerenciamento centralizado e uma maior segurança. Conclui-se que o estudo proposto foi capaz de cumprir com seus objetivos inicialmente propostos, oferecendo uma fonte de estudo para profissionais e estudantes da área.

5.1 TRABALHOS FUTUROS

Devido ao tempo limitado e a falta de recursos computacionais para o desenvolvimento dessa emulação, tornou-se possível explorar a evolução desse estudo de diversas maneiras. A evolução desse estudo baseado em uma maior exploração de segurança dos *pfsenses* e uma integração mais aprofundada com outros possíveis *campus*, utilizando outro tipos de protocolos de roteamento dinâmico, como o *BGP*, também suportado pela solução *Flexiwan*, é viável e plenamente possível.

Desta forma sugere-se como trabalhos futuros:

- Maior integração da ferramenta *pfsense*, criando regras de *firewalls* mais aprofundadas, podendo utilizar ferramentas de *IDS* e/ou *IPS*, como o suricato, que possui integração com a ferramenta *pfsense*;
- A *Flexiwan* possui integração com algumas *clouds*, como a *Google Cloud* e *AWS*. Desta forma, é possível demonstrar sua integração com as ferramentas utilizadas nesse trabalho;
- Uma maior integração com outros *sites*, fazendo o uso de outros protocolos de roteamento dinâmico, como por exemplo, o *BGP*.
- Integração com outras ferramentas para agregar a proposta desse trabalho, como o *openVswitch*,

Zabbix, Grafana, entre outros.

REFERÊNCIAS BIBLIOGRÁFICAS

Bonaccorsi e Rossi 2003 BONACCORSI, A.; ROSSI, C. Why open source software can succeed. *Research Policy*, v. 32, n. 7, p. 1243–1258, 2003. ISSN 0048-7333. Open Source Software Development. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0048733303000519>.

Bustamante e Avila-Pesantez 2021 BUSTAMANTE, J. R.; AVILA-PESANTEZ, D. Comparative analysis of cybersecurity mechanisms in sd-wan architectures: A preliminary results. In: *2021 IEEE Engineering International Research Conference (EIRCON)*. [S.l.: s.n.], 2021. p. 1–4.

Cisco 2023 CISCO. *Cisco*. 2023. <https://www.cisco.com/c/en/us/solutions/enterprise-networks/sd-wan/gartner-2021-wan-edge-infrastructure.html>. (Accessed on 01/ 26/2023).

Controle 2023 CONTROLE. *Controle*. 2023. <https://www.controle.net/faq/switch-core-switches-de-distribuicao-e-switches-de-borda>. (Accessed on 01/ 26/2023).

Coppola e Neelley 2004 COPPOLA, C.; NEELLEY, E. Open source-opens learning: Why open source makes sense for education. 2004.

Docs 2023 DOCS, N. *Pfsense*. 2023. <https://docs.netgate.com/pfsense/en/latest/preface/index.html>. (Accessed on 01/ 12/2023).

Extreme 2023 EXTREME. *Extreme*. 2023. <https://www.extremenetworks.com/products/extremeswitching/>. (Accessed on 01/ 12/2023).

F5 2021 F5. *F5*. 2021. <https://community.fs.com/blog/layer-2-switch-vs-layer-3-switch-which-one-do-you-need.html>. (Accessed on 01/ 26/2023).

Firewall 2023 FIREWALL. *Firewall*. 2023. <https://slideplayer.com.br/slide/5718490/>. (Accessed on 01/ 27/2023).

Flexiwan 2023 FLEXIWAN. *Flexiwan*. 2023. <https://docs.flexiwan.com/>. (Accessed on 01/ 12/2023).

GNS3 2023 GNS3. *GNS3*. 2023. <https://docs.gns3.com/docs/>. (Accessed on 01/ 12/2023).

Juniper 2023 JUNIPER. *Juniper IKEV2*. 2023. <https://www.juniper.net/documentation/br/pt/software/junos/vpn-ipsec/topics/topic-map/security-vpns-for-ikev2.html>. (Accessed on 02/ 06/2023).

Mora-Huiracocha et al. 2019 MORA-HUIRACOCHA, R. E.; GALLEGOS-SEGOVIA, P. L.; VINTIMILLA-TAPIA, P. E.; BRAVO-TORRES, J. F.; CEDILLO-ELIAS, E. J.; LARIOS-ROSILLO, V. M. Implementation of a sd-wan for the interconnection of two software defined data centers. In: *2019 IEEE Colombian Conference on Communications and Computing (COLCOM)*. [S.l.: s.n.], 2019. p. 1–6.

OpenvSwitch 2022 OPENVSWITCH. *ovs*. 2022. <https://docs.openvswitch.org/en/latest/intro/what-is-ovs/>. (Accessed on 02/ 07/2023).

OSPF 2023 OSPF. *OSPF*. 2023. <https://www.gta.ufrj.br/grad/02_2/ospf/ospf.html>. (Accessed on 01/ 17/2023).

SASE 2022 SASE. *SASE*. 2022. <https://www.forcepoint.com/resources/whitepapers/practical-executives-guide-to-sase?_bt=583168487258&_bk=sase&_bm=b&_bn=g&_bg=133293930413&sf_src_cmpid=77015f0000001Qpw&hsa_acc=9947757997&hsa_cam=16285775201&hsa_grp=133293930413&hsa_ad=583168487258&hsa_src=g&hsa_

tgt=kwd-307778569&hsa_kw=sase&hsa_mt=b&hsa_net=adwords&hsa_ver=3&gclid=
CjwKCAiA5sieBhBnEiwAR9oh2rHW9thJJYwhFuaIopiSjayLHYMFaygdDQC5UcNa9fUYNH1mwIEUARoCx1gQAvD
BwE&utm_term=sase&utm_campaign=WW.DEP.PS.Worldwide_search_campaigns_for_SASE.Ever&
utm_source=google&utm_medium=search_pd&utm_content=sase_search>. (Accessed on 01/ 27/2023).

tecmint 2023 TECMINT. *tcpdump*. 2023. <https://www.tecmint.com/
12-tcpdump-commands-a-network-sniffer-tool/>. (Accessed on 02/ 15/2023).

Troia et al. 2020 TROIA, S.; ZORELLO, L. M. M.; MARALIT, A. J.; MAIER, G. Sd-wan: An open-source implementation for enterprise networking services. In: *2020 22nd International Conference on Transparent Optical Networks (ICTON)*. [S.l.: s.n.], 2020. p. 1–4.

UFRJ 2023 UFRJ. *IPsec*. 2023. <https://www.gta.ufrj.br/grad/04_1/vpn/Script/RDIIPSec.html>. (Accessed on 02/ 06/2023).

UFRJ-SDN 2019 UFRJ-SDN. *SDN*. 2019. <https://www.gta.ufrj.br/ensino/eel879/v1/openflow/>. (Accessed on 02/ 07/2023).

VirtualBox 2023 VIRTUALBOX, O. V. *Virtual Box*. 2023. <https://www.virtualbox.org/manual/ch01.html#virtintro>. (Accessed on 01/ 12/2023).

Vyos 2023 VYOS. *Vyos*. 2023. <https://vyos.io/vyos-router>. (Accessed on 01/ 17/2023).

Yalda, Hamad e Ţăpuş 2022 YALDA, K. G.; HAMAD, D. J.; ŢăPUş, N. A survey on software-defined wide area network (sd- wan) architectures. In: *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*. [S.l.: s.n.], 2022. p. 1–5.

ANEXOS

I. INSTALAÇÃO DA FLEXIWAN EDGE

A instalação da ferramenta *Flexiwan Edge* foi feita através do site da *flexiwan*, com a *appliance* da própria fabricante, encontrada no site https://docs.flexiwan.com/installation/download.html. Foi utilizado a versão para o VirtualBox, com a extensão *.VDI*.

Ao fazer o *download*, a imagem já vem montada. Foi configurado alguns parâmetros. O primeiro foi sobre os adaptadores de rede. Foi habilitado o modo *Driver* Genérico, chamado de *UDPTunnel* para as quatro interfaces, para poder trabalhar em conjunto com o *GNS3*, como mostrado na figura 1.1:

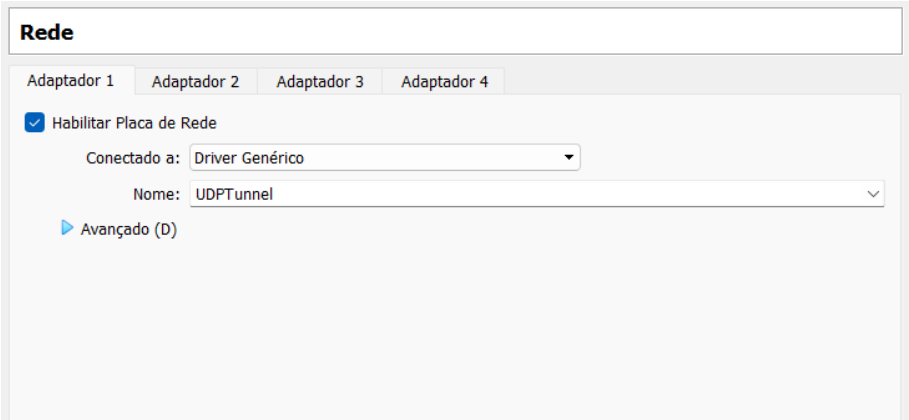

Figura I.1: Configuração de rede.

Além disso, foi habilitado 2 *CPUs* em sistema e 4096 *MB* como memória base, sendo estes pré requisitos para o bom funcionamento de cada *flexi edge*.

A importação para a ferramenta *GNS3*, foi através das opções *Edit - Preferences - VirtualBox - New*. Com isso, o *link* é feito diretamente com o *VirtualBox* e a importação é concluída com sucesso. A máquina fica disponível na opção *end devices* para fazer o seu uso. Esse procedimento foi feito duas vezes, para os campus 1 e campus 2.