UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

LEONARDO TOMAS COSTA DA SILVA

EMULAÇÃO DE REDES DE COMPUTADORES USANDO O GNS3

PONTA GROSSA
2021

LEONARDO TOMAS COSTA DA SILVA

EMULAÇÃO DE REDES DE COMPUTADORES USANDO O GNS3

Emulation of Computer Networks Using GNS3

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Ciência da Computação, do Departamento Acadêmico de Informática, da Universidade Tecnológica Federal do Paraná (UTFPR).

Orientador: Prof. Dr. Augusto Foronda.

PONTA GROSSA

2021

LEONARDO TOMAS COSTA DA SILVA

EMULAÇÃO DE REDES DE COMPUTADORES USANDO O GNS3

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Ciência da Computação, do Departamento Acadêmico de Informática, da Universidade Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: 24 de novembro de 2021

_____

Prof. Dr. Augusto Foronda
Doutorado
Universidade Tecnológica Federal do Paraná

_____

Prof. MSc. Geraldo Ranthum
Mestrado
Universidade Tecnológica Federal do Paraná

_____

Prof. Dr. Lourival Aparecido de Gois
Doutorado
Universidade Tecnológica Federal do Paraná

PONTA GROSSA

2021

Dedico este trabalho a toda minha
família, amigos e professores que
me apoiaram nesta trajetória.0

AGRADECIMENTOS

Gostaria de agradecer a todos que me apoiaram nessa trajetória, principalmente aqueles que conviveram diretamente comigo, em especial meus pais que sempre me forneceram todo apoio que eu precisava, meu amigo Henry que sempre esteve comigo em todos os momentos, minha namorada Andressa que me manteve calmo durante os períodos de estresse e meu amigo Mauricio que compartilhou todas as dificuldades durante o período acadêmico. Acredito que sem o apoio e a paciência deles seria muito mais difícil vencer essa etapa da minha vida.

Agradeço também ao meu orientador Prof. Dr. Augusto Foronda, pela paciência e compreensão que me guiou nesta trajetória acadêmica, além disso foi um segundo pai que me deu conselhos e muita calma para seguir a conclusão do curso.

“Os sonhos não determinam o lugar
que você vai estar, mas produzem a
força necessária para o tirar do lugar
em que está.” (Augusto Cury)

RESUMO

Emulação de redes de computadores, tem como o objetivo de implementar, configurar e testar uma rede antes dela ser implementada em uma rede real, com isso é possível garantir uma maior qualidade de entrega para o ambiente real. Sendo assim, é possível evitar algumas inconsistências que poderiam ocorrer caso fosse realizada diretamente em um ambiente real, ocasionando alguns problemas críticos e quebras na rede. Este trabalho tem como objetivo emular uma rede de computadores utilizando o software GNS3, para configurar alguns protocolos como: *Etherchannel*, *Spanning-Tree*, *OSPF* e *DHCP*. Além disso configurar duas virtual LANs com o objetivo de conter duas redes logicamente independentes e também configurar todos os hosts ligados a rede para que se comuniquem entre si. Após o desenvolvimento, foi elaborado e construído uma rede onde foi aplicado todos os conhecimentos descritos nesse trabalho com o intuito de mostrar e testar a comunicação dos elementos pertencentes a rede.

Palavras-chave: Emulação. Redes. Protocolos. GNS3. VirtualBox.

ABSTRACT

Computer network emulation aims to implement, configure and test a network before it is implemented in a real network, with this it is possible to guarantee a higher quality of delivery for the real environment. Thus, it is possible to avoid some inconsistencies that could occur if performed directly in a real environment, causing some critical problems and network breakdowns. This work aims to emulate a computer network using GNS3 software, to configure some protocols such as Etherchannel, Spanning-Tree, OSPF, and DHCP. Also, configure two virtual LANs in order to contain two logically independent networks and also configure all hosts connected to the network to communicate with each other. After development, a network was designed and built where all the knowledge described in this work was applied in order to show and test the communication of elements belonging to the network.

Keywords: Emulation. Networks. Protocols. GNS3. VirtualBox.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Redes de computadores podem ser definidas como um conjunto de equipamentos, onde além de compartilhar os mesmos recursos, também podem trocar informações entre si, como por exemplo: dois computadores que têm a capacidade de se comunicar e trocar informações como documentos, e-mails, vídeo, áudio, entre outros. Essa interligação entre esses equipamentos dos usuários ocorre principalmente por meios de *switches* e roteadores (TANENBAUM, ANDREW S, 2001).

Os *switches* facilitam o compartilhamento de recursos, conectando todos os dispositivos na rede, sendo assim, esses dispositivos conectados podem compartilhar informações e conversar entre si, independentemente de onde estejam. Além disso, os *switches* são responsáveis por separar uma rede da outra, com um mecanismo implementado nele chamado Virtual LAN (VLAN). Essa função é útil para reduzir o tamanho dos domínios de transmissão ou para permitir que usuários sejam agrupados logicamente sem que precisem estar fisicamente localizados no mesmo local.

Em outras palavras, é possível definir VLANs como uma rede lógica onde pode-se agrupar várias máquinas de acordo com vários critérios. Existe também um protocolo implementado no *switch* que se chama spanning-tree (STP), que tem como objetivo controlar conexões redundantes entre *switches* garantindo o desempenho de troca de informações de uma rede e evitar loop's na rede. As máquinas conectadas nos *switches* terão a possibilidade de acessar uma rede física externa através de um roteador (KUROSE e ROSS, 2010).

Esse acesso a uma rede externa é criado através de rotas para ter o direcionamento a uma rede específica que uma máquina (host) tem permissão para acessar. Essas rotas podem ser estáticas ou dinâmicas. As rotas estáticas usam menos recursos de rede do que o roteamento dinâmico porque não tem que constantemente calcular a rota e atualizar as tabelas de roteamento porque elas já são pré-definidas pelo administrador da rede. Nas rotas dinâmicas, o roteador calcula o caminho mais eficiente para que os pacotes de dados viajem entre a origem e o destino (KUROSE e ROSS, 2010).

Para que essa comunicação ocorra entre os hosts, o mesmo deverá ter o seu endereço Internet *Protocol* (IP), que define uma identificação única de cada

host que pertence a uma determinada rede e é atribuído pelo administrador da rede para se comunicarem. O protocolo que permite as máquinas obterem um endereço IP automaticamente é chamado *Dynamic Host Configuration Protocol* (DHCP). Ele distribui esses endereços IP's quando cada máquina faz uma requisição para o servidor conectado, onde o mesmo retorna um endereço IP (TANENBAUM, ANDREW S, 2001).

Existem algumas opções para o aprendizado dos equipamentos e protocolos de rede citados acima. Uma primeira opção é através de livros e material na Internet, o que ajuda no aprendizado teórico. Outra opção é usar simuladores de rede, como o *Packet Tracer*, que ajuda no aprendizado prático, mas tem limitações por ser um simulador. Existe também a opção de ter um laboratório real, mas a sua principal desvantagem é o custo muito excessivo tornando assim inviável. Outra opção é emular redes de computadores, onde se cria um ambiente o mais próximo do real.

Um dos *softwares* mais utilizados para emulação de rede é o GNS3, que é um *software* de código aberto e gratuito, tendo como principal funcionalidade emular, configurar, testar e solucionar problemas de redes virtuais e reais. Ele permite que se execute uma topologia de rede que consiste em alguns dispositivos, como servidores, roteadores e *switches* em uma única máquina. Para a simulação dos hosts é utilizado o VirtualBox, um *software* livre, onde é possível instalar sistemas operacionais tanto com arquiteturas de X86 quanto X86_64. Uma de suas principais qualidades é poder criar várias máquinas virtuais sem danificar ou prejudicar o funcionamento do seu computador. Assim é possível analisar os principais protocolos utilizados atualmente, como o processo de configuração e o funcionamento deles (GNS3, 2021).

O objetivo desse trabalho é emular uma topologia de rede com o intuito de aprender a teoria e a configuração dos principais protocolos de um *switch*, roteador e servidor de rede.

1.1 Objetivos

Este capítulo descreve os objetivos do trabalho. A seção 1.1.1 descreve o objetivo geral e os objetivos específicos são apresentados na seção 1.1.2.

1.1.1 Objetivo Geral

Emular uma rede de computadores usando o *software* GNS3.

1.1.2 Objetivos Específicos

- Criar uma topologia de rede de computador para ser emulada;
- Emular protocolos de camada de enlace;
- Emular protocolos de camada de rede;
- Emular protocolos de camada de aplicação;
- Analisar os resultados.

1.2 Organização do trabalho

O trabalho foi dividido em 4 capítulos. O Capítulo 2 é referente ao referencial teórico do trabalho, abrangendo os conceitos e metodologias de redes de computadores. O capitulo 3 tem o intuito de exemplificar o processo de desenvolvimento e as configurações que foi preciso para o desenvolver esse trabalho. Por fim o capitulo 4 apresenta a conclusão do que foi realizado e possíveis trabalhos a partir desse.

2 REFERENCIAL TEÓRICO

Neste capítulo serão apresentados os conceitos básicos para a compreensão do trabalho e está divido da seguinte forma: Na sessão 2.1 foi apresentado toda a estrutura do modelo TCP/IP. A sessão 2.2 foi abordado a camada de enlace onde foi apresentado alguns protocolos de redes como *VLANs*, STP e *Etherchannel*. Na sessão 2.3 apresentou a camada de rede assim como alguns de seus protocolos como por exemplo os endereçamentos IPv4, NAT e tipos de roteamento sendo eles dinâmico e estático. A sessão 2.4 foi demonstrado a camada de aplicação e apresentar o protocolo DHCP. Na sessão 2.5 serão apresentados os *softwares* para resolução deste trabalho.

2.1 Modelo TCP/IP

Redes de computadores podem ser definidas como um conjunto de equipamentos, onde além de compartilhar os mesmos recursos, também podem trocar informações entre si, como por exemplo: dois computadores que têm a capacidade de se comunicar e trocar informações como documentos, e-mails, vídeo, áudio, entre outros. Essa comunicação geralmente é feita por cabos onde eles transferem impulsos elétricos, que é identificado pelo computador que recebe essa mensagem.

Essa comunicação ocorre utilizando protocolos de redes, que são um conjunto de regras utilizados por computadores interligados em uma rede para estabelecer a comunicação entre eles. O modelo utilizado para essa comunicação atualmente é o *Transmission Control Protocol/Internet Protocol* (TCP/IP). Além disso, foi criado um outro modelo para padronização geral onde iria ser adotado por todas as empresas quando tiverem a intenção de criar dispositivos para comunicação de redes, esse modelo é o *Open Systems Interconnection* (OSI). Mas como o modelo TCP/IP já era adotado pela maioria das empresas a algum tempo, o modelo OSI se tornou somente um modelo de referência. Uma das comparações entre um modelo e outro é que os dois modelos mencionados são divididos em camadas que serão descritas a seguir (TANENBAUM, ANDREW S, 2001).

O modelo OSI possui sete camadas sendo elas: aplicação, apresentação, sessão, transporte, rede, enlace e física. A Figura 1 mostra o modelo OSI com as sete camadas.

Figura 1 - Diagrama de camadas do modelo OSI

Fonte: Adaptado de Tanenbaum, Andrews (2011)

A camada de aplicação, que é a última camada do modelo OSI, consiste na camada para consumir dados. Esta camada possui programas onde eles garantem a interação homem-máquina, nela consegue-se enviar dados como por exemplo: e-mails, documentos, acessar websites, conectar remotamente outras máquinas, entre outras coisas. Um protocolo de aplicação amplamente utilizado é o *HyperText Transfer Protocol* (HTTP) que constitui a base para a *World Wide Web*. Quando um navegador deseja uma página da Web, ele envia o nome da página desejada ao servidor, utilizando o HTTP e então, o servidor transmite a página de volta. Outros protocolos de aplicação são usados para transferências de arquivos, correio eletrônico e transmissão de notícias pela rede (TANENBAUM, ANDREW S, 2001).

A camada de apresentação é responsável pela formatação dos dados, onde a sintaxe e a semântica das informações transmitidas são gerenciadas por essa camada. Nela ocorre a tradução desses dados para que a próxima camada possa utilizá-los. Esse processo é importante para que duas redes diferentes possam se comunicar entre elas, onde as estruturas de dados trocadas podem ser definidas de maneira abstrata (TANENBAUM, ANDREW S, 2001).

A camada de sessão permite que os usuários de diferentes máquinas estabeleçam sessões entre si. Após receber os dados da camada anterior, inicia-

se o processo de troca de dados e comunicação. Essa camada é responsável por gerenciar a comunicação entres os hosts, onde ocorre o controle de quem deve transmitir em cada momento. Uma sessão oferece vários serviços, dentre eles: o gerenciamento de símbolos, impedindo que duas partes tentem executar a mesma operação crítica ao mesmo tempo e a sincronização, onde é realizada a verificação periódica das transmissões para permitir que elas continuem de onde pararam. Com isso os dados ainda precisam ser tratados para serem usados, porque essa camada só é responsável por gerir a conexão entre esses hosts (TANENBAUM, ANDREW S, 2001).

A camada de transporte garante o envio e o recebimento dos dados recebidos da camada anterior. Ela é responsável por gerenciar o transporte desses dados para garantir o sucesso no envio e no recebimento. A principal função dessa camada é aceitar os dados da camada anterior, dividi-los em unidades menores quando necessário, repassar essas unidades a camada de rede e assegurar que todos os fragmentos cheguem corretamente à outra extremidade (TANENBAUM, ANDREW S, 2001).

Essa camada determina o tipo de serviço que deve ser fornecido à camada de sessão, que é determinado assim que ocorre a conexão. Os protocolos mais comuns utilizados por essa camada são: *Transmission Control Protocol* (TCP) e *User Datagram Protocol* (UDP). O primeiro tem como função garantir a entrega da mensagem, diferentemente do segundo, onde não ocorre a garantia da entrega dessa mensagem, porém é um pouco mais rápido que o TCP (KUROSE e ROSS, 2010).

Mas para ocorrer o transporte de um pacote entre os computadores, é necessário que os hosts consigam se comunicar e esse papel é feito pela camada de rede. Nesta camada tem-se o endereçamento IP de origem e de destino e ela também pode priorizar alguns pacotes e decidir qual caminho seguir (KUROSE e ROSS, 2010). O caminho que os pacotes percorrem são chamados de rotas onde são baseadas em tabelas estáticas ou dinâmicas (TANENBAUM, ANDREW S, 2001).

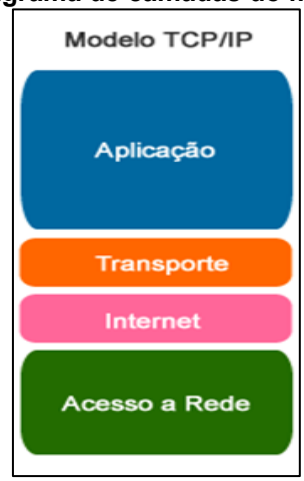
A camada de enlace de dados tem como função principal a verificação ou correção de erros no meio físico e caso possuírem, esse erro pode ser corrigido ou armazenado em um log para posterior análise. Dessa forma, as camadas superiores podem assumir uma transmissão praticamente sem erros.

Alguns exemplos de protocolos utilizados nesta camada, são *Ethernet* e *point-to-point-protocol* (PPP) (KUROSE e ROSS, 2010).

A primeira camada do modelo OSI é a camada física. Enquanto a tarefa da camada de enlace é movimentar quadros inteiros de um elemento da rede até um elemento adjacente, na camada física é movimentar os bits individuais que estão dentro do quadro de um nó para o seguinte. As regras de comunicação definidas nesta camada dependem do próprio meio de transmissão, como por exemplo, fios de cobre ou fibra óptica (KUROSE e ROSS, 2010).

Estas camadas do modelo OSI são simplificadas no modelo TCP/IP. Ele apresenta basicamente as mesmas camadas já descritas anteriormente no modelo OSI, com a diferença entre as quantidades de camadas, onde no modelo OSI possuí sete camadas e agora no TCP/IP possuí apenas quatro, que são: a camada de aplicação, a camada de transporte, a camada de internet e a camada de acesso à rede, como pode ser visto na Figura 2.

Figura 2 - Diagrama de camadas do modelo TCP/IP



Fonte: Adaptado de Tanenbaum, Andrews (2011)

O modelo TCP/IP não possui as camadas de sessão e de apresentação como já mencionado no modelo OSI. A camada de aplicação contém todos os protocolos de nível mais alto. Dentre eles o protocolo de terminal virtual (TELNET) que permite que um usuário de um computador qualquer se conecte a outro distante e consiga acesso sobre ele, o protocolo de transferência de arquivos (FTP) que permite mover dados com eficiência de uma máquina para outra e o protocolo de correio eletrônico (SMTP) que é utilizado para envio e recebimento de e-mail. Além disso muitos outros protocolos foram incluídos com o decorrer dos anos, como o *Domain Name Service* (DNS), que mapeia os

nomes de hosts para seus respectivos endereços de rede e o HTTP, o protocolo usado para buscar páginas na *World Wide Web*, entre muitos outros. A camada de transporte e internet do modelo TCP/IP apresentam as mesmas funções equivalentes já explicadas anteriormente no modelo OSI, assim como as camadas física e enlace equivalem a camada de acesso a rede no modelo TCP/IP (TANENBAUM, ANDREW S, 2001).

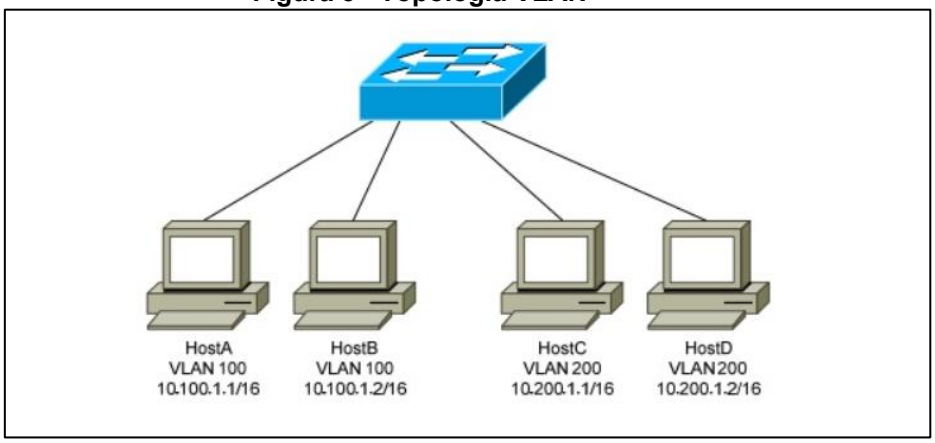
2.2 Camada de enlace

Nesta seção 2.2 foi abordado sobre a camada enlace onde nela foi exemplificado alguns protocolos sendo eles VLAN, *Spanning Tree Protocol* (STP) e *EtherChannel*.

2.2.1 VLAN

As Virtual LAN (*VLANs*) são um mecanismo implementado em um *switch* que permite que os administradores de rede criem os domínios de transmissão lógicos que podem ser distribuídos em um único *switch* ou em vários, independente da proximidade física. Essa função é útil para reduzir o tamanho dos domínios de transmissão ou para permitir que grupos ou usuários sejam agrupados logicamente sem que precisem estar fisicamente localizados no mesmo local. Em outras palavras, é possível definir *VLANs* como uma rede lógica onde pode-se agrupar várias máquinas de acordo com vários critérios, por exemplo, agrupá-las por departamento, como mostra a Figura 3, onde existem a *VLAN* 100 e *VLAN* 200, sendo que cada uma possui um grupo de hosts que pertence à mesma rede. Cada *VLAN* representa uma rede separada, embora esteja no mesmo *switch* e deve ter um endereçamento IP separado, como mostrado na Figura 3. A *VLAN* 100 tem o endereço de rede 10.100.0.0/16 e a *VLAN* 200 tem o endereço de rede 10.200.0.0/16 (PEREIRA, DIEGO, CESAR, 2018).

Figura 3 - Topologia VLAN



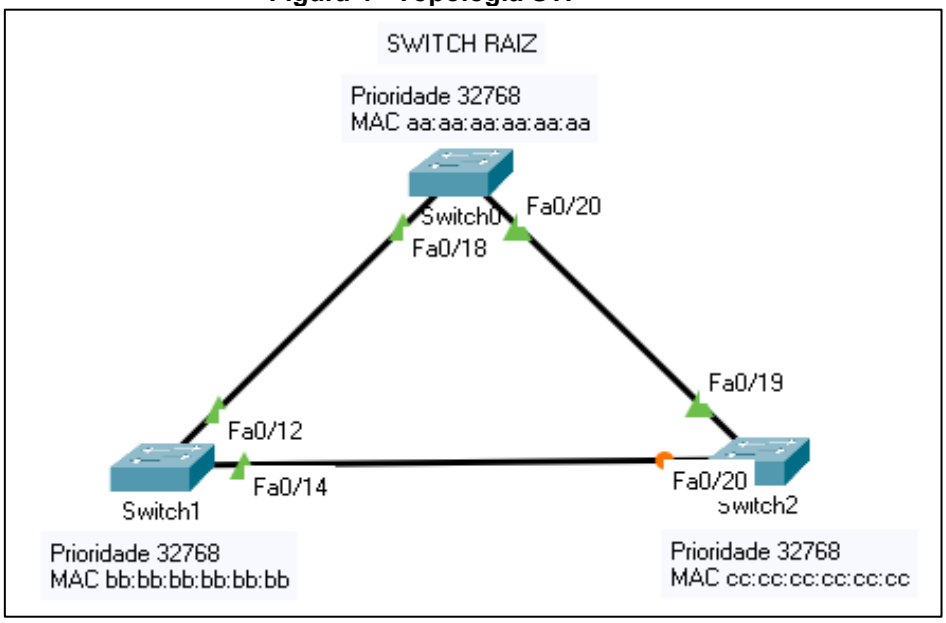
Fonte: Adaptado de Cisco CCNA (2021)

As principais vantagens de um *VLAN* são a segurança, a escalabilidade, a flexibilidade e redução de custos por ser uma rede lógica. Existem dois tipos de ligações para uma *VLAN*. O primeiro é uma Porta de Acesso (*access*), onde é possível associar uma porta do *switch* a uma *VLAN*. As portas do tipo acesso são usadas para ligar os hosts da rede. O outro tipo de ligação é a ligação partilhada (*trunk*), normalmente usada para interligação de *switches*, ela permite a passagem de tráfego de várias *VLANs*. Configurando uma porta como *trunk*, todo o tráfego de todas as *VLANs* criadas no *switch* pode passar por esta conexão.

2.2.2 STP

O protocolo *spanning-tree* (STP) é um protocolo que pertence a segunda camada do modelo OSI e tem como objetivo controlar conexões redundantes entre *switches* garantindo o desempenho de troca de informações de uma rede. O STP tem como estrutura o algoritmo de IEEE 802.1D, que tem como funcionalidade detectar loops na rede e removê-los, isso acontece quando ocorre trocas de mensagens com outros *switches* (NOGUEIRA; REIS; CALMON; SILVA; FORMIGONI, 2016).

Figura 4 - Topologia STP



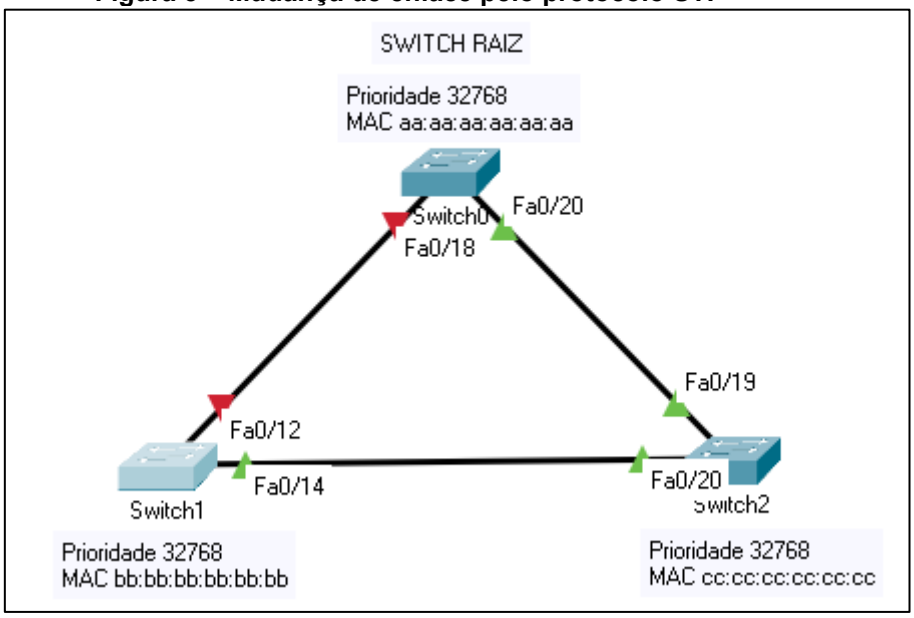
Fonte: Autoria própria (2021)

A Figura 4 mostra uma topologia de *switches* com enlaces redundantes, onde o protocolo STP deve ser usado para evitar loop na rede. A finalidade do protocolo STP é bloquear uma das seis portas no enlace entre os *switches*. A porta bloqueada na Figura 18 é a porta Fa0/20 do *switch*2.

Para escolher a porta que vai ser bloqueada, o STP usa dois critérios: a prioridade e o valor MAC do *switch*. Primeiro é usado o valor da prioridade e se for o mesmo valor, como no caso da Figura 4, onde o valor é 32768 para todos os *switches*, depois é usado o valor MAC de cada *switch*, onde o *switch* com o maior valor terá uma de suas portas bloqueadas. O *switch* com o menor valor, ou de prioridade ou de endereço MAC, foi o *switch* raiz e não teve suas portas bloqueadas (F0/18 e F0/20 do *switch* 0). As portas conectadas no *switch* raiz também não tem suas portas bloqueadas (Fa0/12 do *switch*1 e Fa0/19 do *switch*2). Como o *switch*1 tem endereço MAC menor do que o *switch*2, a porta F0/20 do *switch*2 é escolhida para ser bloqueada.

Se o enlace entre o *switch*0 e *switch*1 ou o enlace entre o *switch*0 e *switch*2 ficam desativados por algum problema no equipamento ou no meio físico, o enlace entre o *switch*1 e *switch*2 foi ativado, como pode ser visto na Figura 5.

Figura 5 – Mudança de enlace pelo protocolo STP



Fonte: Autoria própria (2021)

Além disso, o STP é um protocolo de árvores de abrangência rápida, com isso possui tempos de convergência mais rápidos. Para que ocorra o funcionamento do STP, os *switches* na mesma rede precisam ser habilitados para que possa executar o algoritmo de árvore de abrangência, assim é possível determinar com precisão qual *switch* deve ser eleito a “ponta raiz”. Esta ponte raiz foi responsável por enviar unidades de dados de protocolo de ponte de configuração, junto com outras informações para seus *switches* diretamente conectados que, por sua vez, encaminham para seus *switches* vizinhos. Cada *switch* tem um valor de prioridade, que é uma combinação de um valor de prioridade e o próprio endereço MAC do *switch*. O *switch* com valor de prioridade mais baixo se tornará a bridge raiz.

Existem cinco estados de *switchport* STP que são:

- Desabilitado - O resultado de um comando administrativo que desabilita a porta.
- Bloqueio - quando um dispositivo for conectado, a porta entra primeiro no estado de bloqueio.
- Ouvindo - O *switch* “ouve” e “envia” unidades de dados de protocolo de ponte de configuração.
- Aprendizagem - O *switch* recebe unidades de dados de protocolo de ponte de configuração superior e não envia os seus próprios.

- Encaminhamento - A porta encaminha o tráfego.

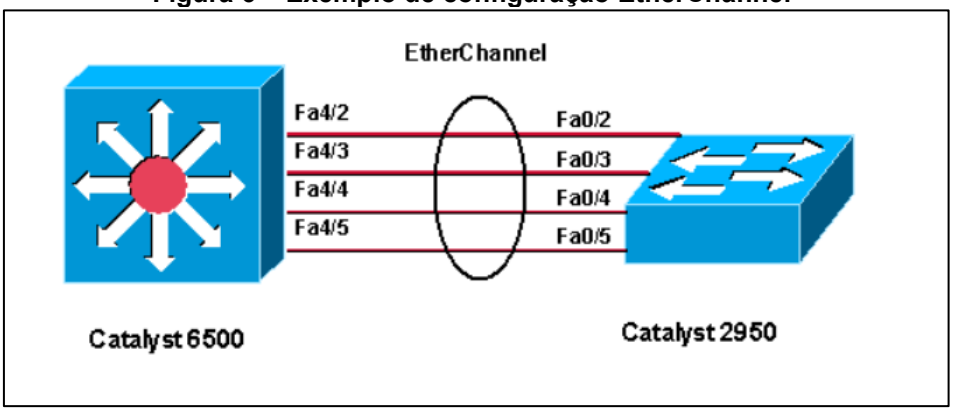
Além disso algumas regras para o STP:

- Root - são portas em *switches* não raiz com o melhor caminho de custo para a ponte raiz. Essas portas encaminham dados para a bridge raiz.
- Designado - são portas na raiz. Todas as portas na bridge raiz serão designadas.
- Bloqueado - todas as outras portas para pontes ou *switches* estão em um estado bloqueado.

2.2.3 EtherChannel

EtherChannel é uma tecnologia de agregação de link de porta. Ele permite agrupar vários links *Ethernet* físicos criando somente um link *Ethernet* lógico com o objetivo de fornecer tolerância a falhas e links de alta velocidade entre *switches*, roteadores e servidores.

Figura 6 – Exemplo de configuração EtherChannel



Fonte: Adaptado de Cisco CCNA (2021)

Portanto, *EtherChannel* é a união das portas físicas, como mostra na Figura 6. Por exemplo, as portas Fa4/ (2,3,4,5) e Fa0/ (2,3,4,5) são unidas através da criação de um link Ethernet lógico, passando a ser visto somente como uma única comunicação entre o *Catalyst* 6500 e o *Catalyst* 2950, assim, é possível evitar falhas referente a comunicação. Ao configurar um *EtherChannel*, cria-se um balanceamento de carga entre os links físicos envolvidos neste *channel (*VMWARE).

Existem dois protocolos utilizados na configuração do *EtherChannel*:

–**PagP** (*Port Aggregation Control Protocol*): Protocolo proprietário Cisco;

–**LacP** (*Link Aggregation Control Protocol*): Protocolo de padrão aberto.

2.3 Camada de rede

Esta seção apresenta alguns protocolos que pertencem a essa camada que serão utilizados neste trabalho, como por exemplo endereçamento IPv4, NAT e alguns tipo de roteamento como dinâmico e estático. Além disso em roteamento foi abordado sobre o protocolo OSPF.

2.3.1 Endereço IPv4

O protocolo Internet *Protocol* (IP) define uma identificação única de cada host que pertence a uma determinada rede e é atribuído pelo administrador da rede. O endereço IP está associado a uma máscara de rede que tem como função definir o número de hosts ou equipamentos que aquela rede pode ter, como mostra a Figura 7.

Figura 7 - Tabela de Máscara de Rede TCP/IP

HOSTS	REDES	CIDR	MÁSCARA DE SUB-REDE
1	256	/32	255.255.255.255
2	128	/31	255.255.255.254
4	64	/30	255.255.255.252
8	32	/29	255.255.255.248
16	16	/28	255.255.255.240
32	8	/27	255.255.255.224
64	4	/26	255.255.255.192
128	2	/25	255.255.255.128
256	1	/24	255.255.255.0

Fonte: Adaptado de Tanenbaum, Andrews (2011)

Na Figura 7 pode ser visto a quantidade de hosts que uma rede pode ter. Por exemplo, uma rede com a máscara 255.255.255.0/24 pode ter

interligados nela 256 hosts, onde dois IP's dessa rede são reservados, sendo assim um IP é definido configurado como endereço de rede e o outro como endereço de broadcast, que serão explicados posteriormente, portanto, possui 254 endereços de IP’s que podem ser atribuídos para os hosts ou equipamentos dessa rede.

Pode-se dividir todos esses IP's em dois grupos principais: endereços públicos e privados. Os IP's públicos são endereços projetados para serem usados nos hosts que são acessíveis publicamente a partir da Internet. Atualmente a vasta maioria dos endereços no intervalo de host unicast (endereçamento para um pacote feito a um único destino) IPv4 são endereços públicos.

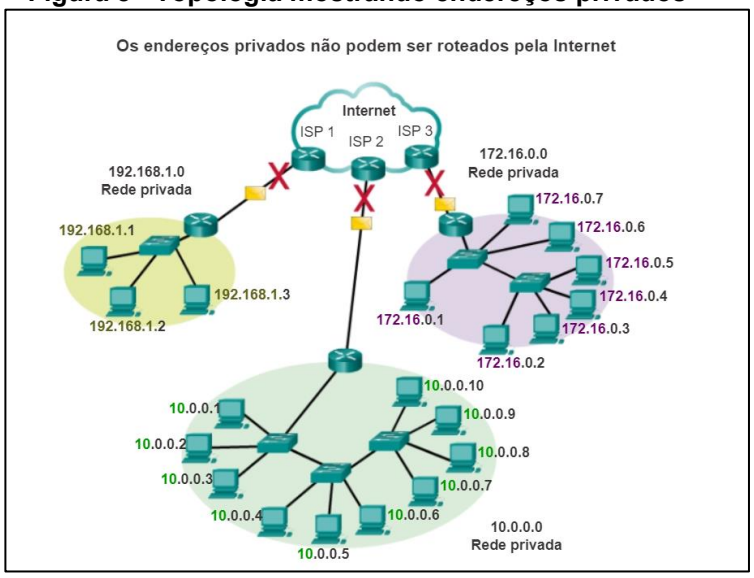
Embora a maioria dos endereços de host IPv4 sejam endereços públicos designados para uso em redes que são acessíveis pela Internet, existem blocos de endereços que são usados em redes que precisam de acesso limitado ou nenhum acesso à Internet. Esses endereços são chamados de endereços privativos.

Alguns exemplos de endereços privativos podem ser vistos na Figura 8 e são:

- 10.0.0.0 a 10.255.255.255 (10.0.0.0/8);
- 172.16.0.0 a 172.31.255.255 (172.16.0.0/12);
- 192.168.0.0 a 192.168.255.255 (192.168.0.0/16).

Esses endereços são destinados ao uso de uma rede local fechada e a alocação desses endereços não ocorre por uma rede externa, como por exemplo a Internet. Os hosts que não requerem acesso à Internet poderão usar endereços particulares, no entanto, dentro da rede privada, os hosts ainda exigem endereços IP exclusivos no espaço privado.

Figura 8 - Topologia mostrando endereços privados



Fonte: Adaptado de Cisco (2021)

2.3.2 NAT

O acesso direto à Internet usando um endereço IP privado não é possível. Nesse caso, a conexão com a Internet é via *Network Address Translation* (NAT), onde a tradução do endereço de rede substitui o endereço IP privado por um público. Os endereços IP privados na mesma rede local devem ser exclusivos, ou seja, não pode ocorrer um mesmo host ter o mesmo IP (AHMED, 2018).

Em relação à segurança na rede, o uso de um endereço privado se torna muito mais seguro do que um endereço público, porque os IP's privados não são diretamente visíveis na Internet e estão armazenados e tratados pelo NAT, o que também garante a segurança de uma rede doméstica, sendo que o oposto acontece com os IP’s públicos.

2.3.3 Roteamento

O roteamento é um processo que encaminha pacotes de dados com base nas políticas definidas pelos administradores da rede. O roteamento está dividido em dois tipos: roteamento estático e roteamento dinâmico.

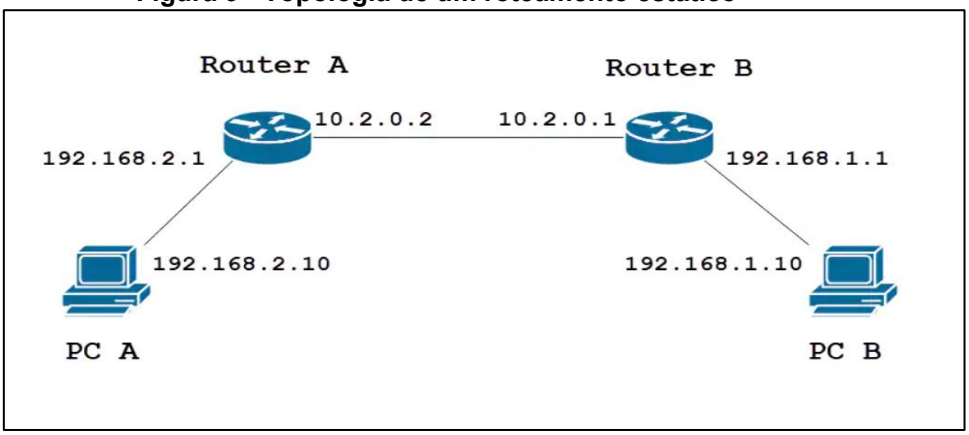
2.3.3.1 Roteamento estático

O roteamento estático é um caminho pré-definido pelo administrador da rede onde um pacote deve percorrer para chegar ao seu destino. Se há uma ausência de comunicação entre os roteadores em relação à topologia atual da rede, as rotas estáticas podem ser configuradas para estabelecer uma comunicação direta entre os roteadores.

As rotas estáticas usam menos recursos de rede do que o roteamento dinâmico porque não tem que constantemente calcular a rota seguinte para tomar, porque elas já são pré-definidas pelo administrador da rede. Uma rota estática cria um trajeto fixo onde um pacote deve viajar entre os roteadores, como pode ser visto na Figura 9, onde o roteador A deve ter uma rota estática para a rede do PC B e o roteador B deve ter uma rota estática para a rede do PC A.

A principal utilização de rotas estáticas acontece quando existem redes com poucos elementos de conexão e não existam caminhos redundantes, sendo assim a maior dificuldade que se pode ter é criação e manutenção dessas rotas caso a rede se torne muito grande.

Figura 9 - Topologia de um roteamento estático



Fonte: Adaptado de Cisco (2021)

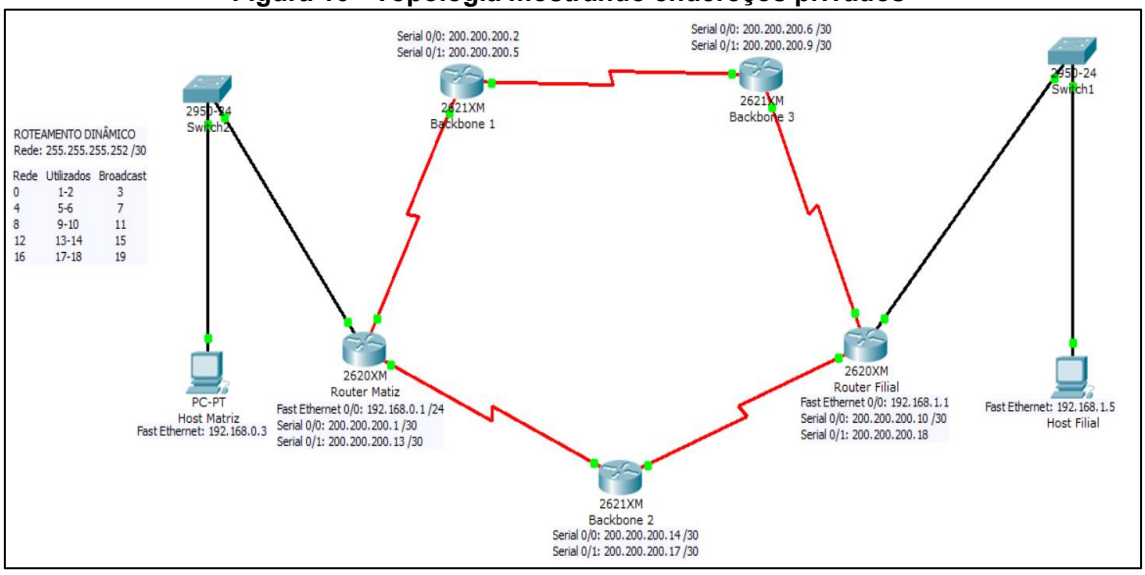
2.3.3.2 Roteamento dinâmico

O roteamento dinâmico permite que o roteador ajuste automaticamente as mudanças físicas na disposição de rede, onde o roteador calcula a rota mais

eficiente para que os pacotes de dados de rede viajem entre a origem e o destino. Para que essa comunicação ocorra, os roteadores compartilham informações através de protocolos de roteamento dinâmicos. A Figura 10 mostra uma topologia com caminhos redundantes, onde deve ser aplicado o roteamento dinâmico.

Esse protocolo de comunicação nada mais é que uma linguagem que o roteador se comunica com outros roteadores para troca de informações, como por exemplo sobre a distância entre as redes e o estado delas.

Figura 10 - Topologia mostrando endereços privados

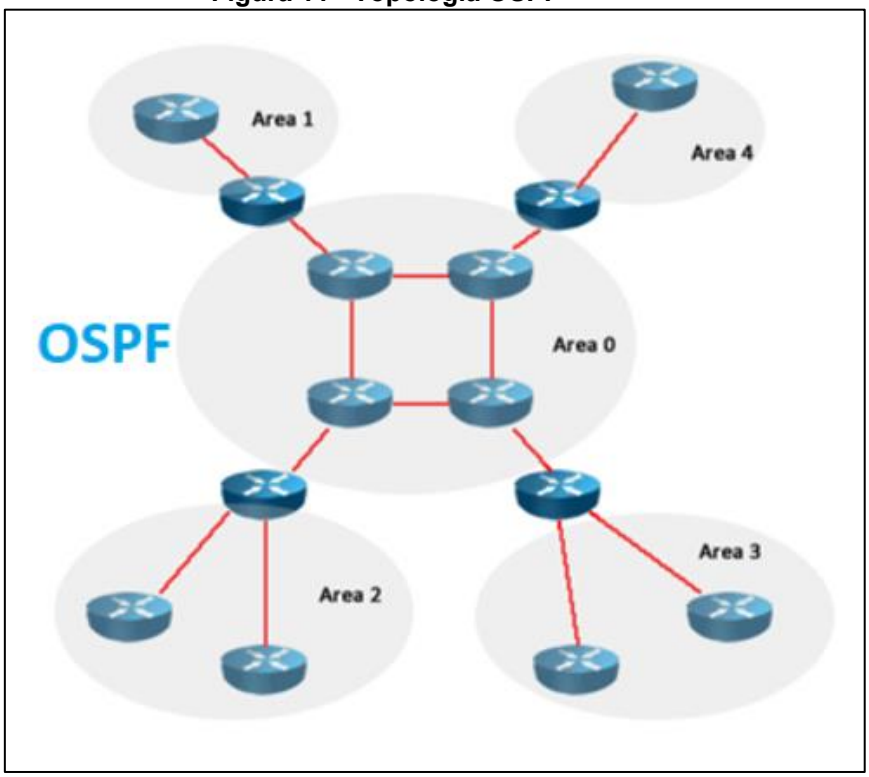


Fonte: Adaptado de Cisco (2021)

2.3.3.3 OSPF

O protocolo *Open Shortest Path First* (OSPF), definido no RFC 2328, é um protocolo de roteamento dinâmico *Internal Gateway Protocol* (IGP) utilizado para distribuir a informação de roteamento em um único Sistema Autônomo, como o representado na Figura 11. Ele foi desenvolvido devido a uma necessidade na comunidade da Internet de introduzir um IGP não proprietário de alta funcionalidade para a família de protocolos TCP/IP (CUNHA, 2018).

Figura 11 - Topologia OSPF



Fonte: Adaptado de Cisco (2021)

O protocolo tem por base a tecnologia *link-state*, que é o ponto de partida do vetor de *Bellman-Ford* com base em algoritmos utilizados nos protocolos de roteamento tradicionais da Internet. Ele distingue quatro classes de roteadores (TANENBAUM, ANDREW S, 2011):

- Os roteadores internos, que ficam inteiramente em uma área;
- Os roteadores de borda de área, que conectam duas ou mais áreas;
- Os roteadores de *backbone*, que ficam no *backbone*;
- Os roteadores de fronteira do SA, que interagem com roteadores de outros SAs.

No OSPF, para que os roteadores possam trocar informações entre eles, deve-se somente configurar as redes conectadas nas interfaces do roteador, com isso, a cada atualização, enviam toda ou parte de suas tabelas de roteamento para seus vizinhos e depois da convergência da rede, todos os roteadores têm nas suas tabelas todas as redes da topologia. Diferentemente do roteamento estático, onde as rotas devem ser configuradas manualmente em cada roteador.

2.4 Camada de aplicação

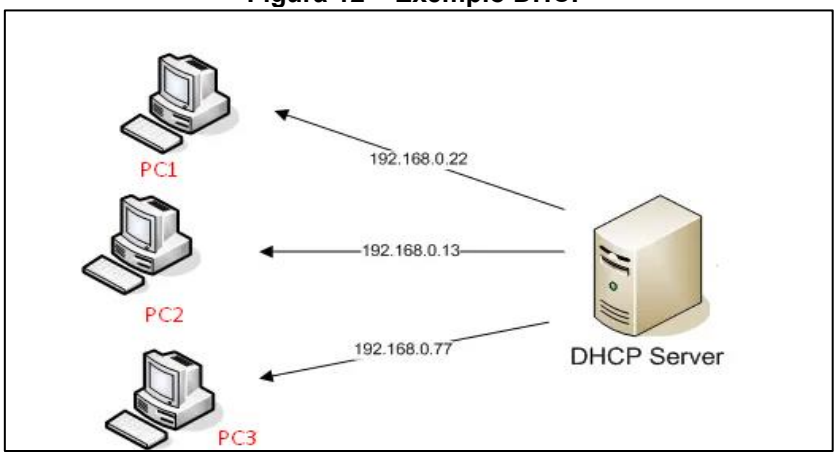
Nessa seção foi apresentado o protocolo DHCP onde ele é uns dos protocolos mais utilizados por essa camada.

2.4.1 DHCP

O *Dynamic Host Configuration Protocol* (DHCP) é um protocolo que permite às máquinas obterem um endereço IP automaticamente. Ele distribui esses endereços IP's quando cada máquina faz uma requisição para o servidor conectado, onde o mesmo retorna um endereço IP (DHCP).

Em uma topologia com poucos computadores, é fácil e rápido fazer a configuração manual dos IP's, agora quando existem muitas máquinas na rede, a aplicação do DHCP se torna uma necessidade. A distribuição de IP's ocorre quando as máquinas fazem a solicitação de conexão com a rede ou em um intervalo pré-definido pelo servidor, como pode ser visto na Figura 12. Os IP's fornecidos pelo servidor pertencem ao mesmo intervalo de rede e sempre que uma das máquinas for desconectada nessa rede, o IP ficará livre para o uso em outra máquina.

Figura 12 – Exemplo DHCP



Fonte: Adaptado de Cisco (2021)

O DHCP pode operar de três formas: automática, dinâmica e manual.

- Automática: uma quantidade de endereços de IP (dentro de uma faixa) é definida para ser utilizada na rede. Neste caso, sempre que

um dos computadores de uma rede solicitar a conexão com ela, um destes IP's foi designado para a máquina em questão;

- Dinâmica: o procedimento é bem parecido com o efetuado pela automática, porém a conexão do computador com determinado IP é limitada por um período de tempo pré-configurado que pode variar conforme desejado pelo administrador da rede;
- Manual: o DHCP aloca um endereço de IP conforme o valor *Medium Access Control* (MAC) de cada placa de rede de forma que cada computador utilizará apenas este endereço IP. Utiliza-se este recurso quando é necessário que uma máquina possua um endereço IP fixo.

2.5 GNS3

O GNS3 é um *software* de código aberto e gratuito, tendo como principal funcionalidade emular, configurar, testar e solucionar problemas de redes virtuais e reais. Ele permite que você execute uma topologia de rede que consiste em alguns dispositivos, como servidores, roteadores e *switches* em uma única máquina.

Além dessas características, o GNS3 pode emular dispositivos de vários fornecedores de rede, incluindo *switches* virtuais Cisco, Cisco ASAs, Brocade vRouters, *switches Cumulus Linux*, instâncias Docker, HPE VSRs, vários dispositivos Linux e muitos outros.

GNS3 possui alguns critérios recomendados para executar a simulação:

- SO: Windows 7 (64 bits) e posterior, Mavericks (10.9) e posterior, Any Linux Distro - Debian / Ubuntu são fornecidos e suportados.
- Processador: 4 ou mais núcleos lógicos - AMD-V / RVI Séries ou Intel VT-X / EPT - extensões de virtualização presentes e habilitadas no BIOS. Mais recursos permitem uma simulação maior.
- Memoria: 8 GB de RAM.
- Armazenamento: SSD - 35 GB de espaço disponível.

Algumas de suas principais vantagens são:

- *Software* livre;

- Sem limitação no número de dispositivos suportados (a única limitação é o seu *hardware*: CPU e memória);
- Suporta várias opções de equipamentos de camada 2 (módulo *Etherswitch* NM-ESW16, imagens IOU / IOL Camada 2, VIRL IOSvL2):
- Suporta todas as imagens VIRL (IOSv, IOSvL2, IOS-XRv, CSR1000v, NX-OSv, ASAv);
- Suporta ambientes de vários fornecedores;
- Pode ser executado com ou sem hipervisores;
- Suporta hipervisores gratuitos e pagos (Virtualbox, VMware workstation, VMware player, ESXi, Fusion);
- Dispositivos para download, gratuitos, pré-configurados e otimizados disponíveis para simplificar a implantação;
- Suporte nativo para Linux sem a necessidade de *software* de virtualização adicional;
- *Software* de vários fornecedores disponível gratuitamente;
- Comunidade grande e ativa (mais de 800.000 membros).

Contudo, o GNS3 também apresenta algumas desvantagens, sendo elas:

- As imagens da Cisco precisam ser fornecidas pelo usuário (faça download em Cisco.com, adquira a licença VIRL ou copie do dispositivo físico);
- Não é um pacote independente, mas requer uma instalação local do *software* (GUI);
- O GNS3 pode ser afetado pela configuração e limitações do seu PC devido à instalação local (*firewall* e configurações de segurança, políticas de laptop da empresa, etc.).

Para o processo de simulação foi utilizado uma imagem do roteador cisco 7200 e uma imagem do roteador cisco 3725, porém esse também pode se comportar como um *switch*. Ambos os roteadores têm suas imagens disponibilizadas pela Cisco gratuitamente (CISCO, 7200).

No processo de configuração e instalação de ambos no GNS3, serão usadas imagens iguais aos equipamentos reais onde todo o processo de instalação se encontra no Apêndice 1.

2.5.1 VirtualBox

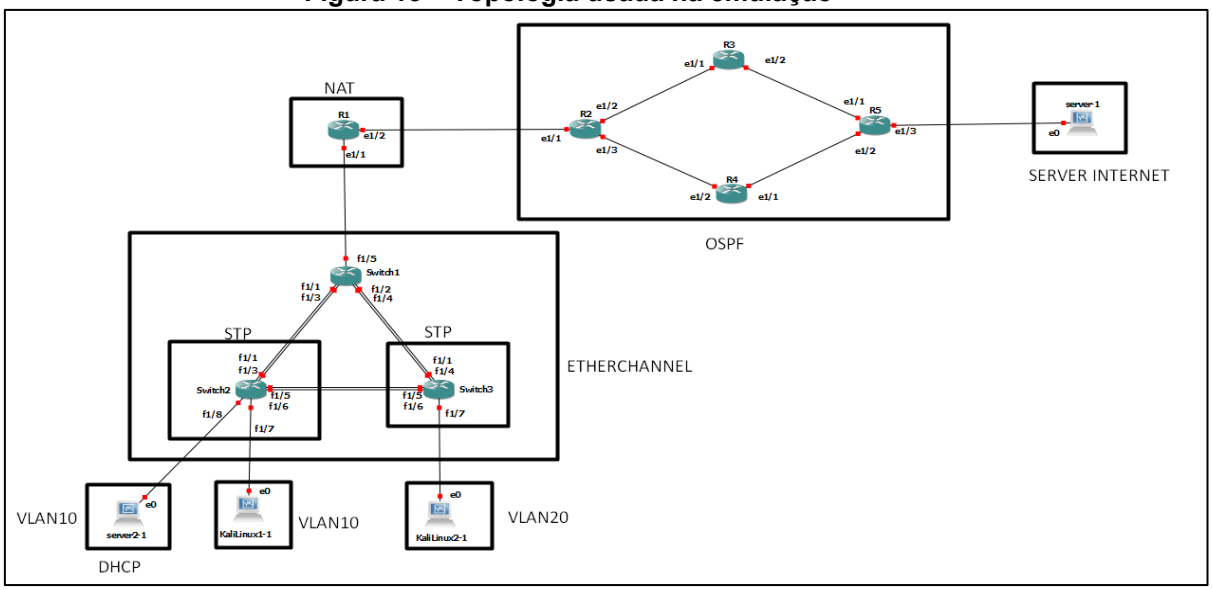
O VirtualBox é um *software* de virtualização criado e mantido pela Oracle. O VirtualBox pode ser usado tanto para a área acadêmica como para a área empresarial. Está disponível para computadores com arquiteturas de X86 e X86_64. É um *software* bem intuitivo e suporta diversos tipos de sistemas para instalação. Ele se destaca por ser de livre acesso, sem custos, e consiste na categoria de *softwares Open Sources* sobre os termos da versão GNU (*General Public Licence*).

Além disso, existem várias vantagens de usar virtualizadores como o VirtualBox, já que se pode criar várias máquinas virtuais sem danificar ou prejudicar o funcionamento do seu computador. Sendo assim, existem diversas opções que podem ser aplicadas com esse método, como simular redes, testes com outros sistemas operacionais e diversas outras opções.

3 Desenvolvimento

A Figura 13 apresenta a topologia que foi ser desenvolvida no trabalho, onde foram configurados alguns dos protocolos mais utilizados atualmente e que foram explicados anteriormente no capitulo 2. A topologia usada representa uma versão reduzida de como funciona a internet. A rede LAN é representada pelas máquinas nas VLANs 10 e 20, como se fossem 2 departamentos de uma empresa. As máquinas são conectadas em *switches* e estes são conectados no roteador de borda da empresa, onde tem o NAT. Este roteador é conectado em uma operadora, que é representada pelos 4 roteadores com OSPF e por fim é conectado em um servidor na internet, representado pela máquina Server 1. Ou seja, a topologia vai desde um host dentro de uma LAN até um servidor na internet passando por uma operadora.

Figura 13 – Topologia usada na emulação



Fonte: Autoria própria (2021)

A topologia da Figura 13 tem as seguintes configurações:

- VLANs: sendo VLAN 10 e VLAN 20, tendo como objetivo emular duas redes diferentes como se fossem dois departamentos de uma empresa onde cada departamento acessou somente a rede definida para ele. A VLAN 10 tem a máquina KaliLinux1-1 e o servidor DHCP e a VLAN 20 tem a máquina KaliLinux2-1;
- *Etherchannel*: onde tem como objetivo ampliar a conexão para aumentar a vazão da rede, por exemplo um enlace que tem 100Mbps

passará para 200Mbps se o *etherchannel* criar um canal virtual com duas portas físicas. Os 3 *switches* vão ter *etherchannel* configurado em 4 portas físicas, por exemplo no *switch*2 as portas f1/ (1,3,5,6) serão configuradas, no *switch*3 as portas f1/ (1,4,5,6), no *switch*1 f1/ (1,2,3,4);

- STP (Spanning-tree): que tem o objetivo de fornecer um caminho redundante caso ocorra uma queda de comunicação, por exemplo, na Figura 13 se as portas f1/1 e f1/3 do *switch* 2 caírem, o spanning-tree encaminhou para o *switch* 3 onde terá acesso ao *switch* 1 e a rede por completo, essa configuração foram feitos nos *switches* 2 e 3;
- OSPF: que tem o objetivo de emular uma rede redundante de um provedor de Internet. A configuração foi nos roteadores R2, R3, R4 e R5;
- Rota default: é uma rota estática configurada no roteador de saída da empresa, representado pelo roteador R1 para todas as máquinas poderem acessar a Internet, que nesta topologia é representado pelo servidor Server1. Essa rota foi configurada onde está sinalizado o NAT na Figura 13, somente o roteador R1 terá essa configuração;
- NAT: todo roteador de borda da rede tem o NAT configurado para converter endereço privado em endereço público e foi configurado no roteador R1;
- DHCP: é responsável por gerar IP's automaticamente para os hosts que se conectarem na rede, no caso os hosts Kali Linux 1 seria o cliente e o server2-1 seria o servidor DHCP como mostra a Figura 13.

3.1 Configuração do endereçamento IP

Nessa seção foi exemplificado como configurar os endereços IP's das máquinas presentes na topologia da Figura 13. A tabela 1 mostra a tabela de endereçamento IP dessas máquinas.

Tabela 1 – Endereçamento Ip

Equipamento	Endereço IP	Máscara		gateway
Server2-1	192.168.1.3	255.255.255.0	/24	192.168.1.1
KaliLinux1-1	192.168.1.4	255.255.255.0	/24	192.168.1.1
KaliLinux2-1	192.168.2.2	255.255.255.0	/24	192.168.2.1
Server 1	200.1.1.1	255.255.255.0	/24	200.1.1.2
Roteador R1 Ethernet 1/1.10	192.168.1.1	255.255.255.0	/24	
Roteador R1 Ethernet 1/1.20	192.168.2.1	255.255.255.0	/24	
Roteador R2 Ethernet 1/1	10.0.0.17	255.255.255.252	/30	
Roteador R2 Ethernet 1/2	10.0.0.6	255.255.255.252	/30	
Roteador R2 Ethernet 1/3	10.0.0.9	255.255.255.252	/30	
Roteador R3 Ethernet 1/1	10.0.0.5	255.255.255.252	/30	
Roteador R3 Ethernet 1/2	10.0.0.2	255.255.255.252	/30	
Roteador R4 Ethernet 1/1	10.0.0.13	255.255.255.252	/30	
Roteador R4 Ethernet 1/2	10.0.0.10	255.255.255.252	/30	
Roteador R5 Ethernet 1/1	10.0.0.1	255.255.255.252	/30	
Roteador R5 Ethernet 1/2	10.0.0.14	255.255.255.252	/30	
Roteador R5 Ethernet 1/3	200.1.1.2	255.255.255.0	/24	

Fonte: Autoria própria (2021)

A topologia desenvolvida, possui duas VLANs, portanto cada rede terá o seu endereço privado, a comunicação entre elas foi tratada pelo roteador R1. O server 1 representa um servidor na internet, assim ele tem um endereço verdadeiro.

A configuração do endereçamento IP foi feita manualmente, ou seja, configurado com IP’s estáticos. Depois, foi mostrado a configuração do DHCP no server 2 onde foi atribuído o IP automaticamente para o host Kali Linux 1. Além disso, foi configurado o server 1 como o servidor da internet com endereço verdadeiro, nesse caso ele terá o endereço IP público para que os hosts possam acessá-lo e o roteador R1 foi responsável por essa tradução de pacotes entre os hosts de dentro e fora da rede.

Figura 14 – Configuração server 2-1

```
iface enp03s3 inet static
  address 192.168.1.3
  netmask 255.255.255.0
  network 192.168.1.0
  broadcast 192.168.1.255
  gateway 192.168.1.1
```

Fonte: Autoria Própria (2021)

A Figura 14 mostra a configuração do host server 2. Essa configuração precisa ser feita no arquivo /etc/network/interfaces, lembrando que está sendo utilizado o sistema operacional Linux. Essa configuração tem o IP do host, o IP de rede onde o host está e o gateway, no caso da figura 14 o IP de rede é 192.168.1.0 onde foi configurado para a VLAN 10. A mesma configuração foi feita nos outros hosts da rede com os seus respectivos endereços IP's.

3.2 Configuração de VLANs

A configuração das VLANs é executada nos *switches* 1, 2 e 3 e é dividida nas seguintes etapas:

1) Criação das VLANs;
2) Configuração das portas dos *switches* como acesso ou tronco;
3) Configuração das sub interfaces nos roteadores.

3.2.1 Criação das VLANs

A Figura 15 mostra os comandos para a criação das VLANs 10 e 20 no *switch*1 onde terá as duas VLANs configuradas.

Figura 15 - Configuração da vlan no *switch* 1

```
Switch1#: vlan database
Switch1#: vlan 10
Switch1#: vlan 20
Switch1#: exit
```

Fonte: Autoria Própria (2021)

A Figura 15 mostra os comandos para a criação da VLAN 10 no *switch*2.

Figura 16 - Configuração da vlan no *switch* 2

```
Switch2#: vlan database
Switch2#: vlan 10
Switch2#: exit
```

Fonte: Autoria Própria (2021)

A Figura 16 mostra os comandos para a criação da VLAN 20 no *switch*3.

Figura 17 - Configuração da vlan no *switch* 3

```
Switch3#: vlan database
Switch3#: vlan 20
Switch3#: exit
```

Fonte: Autoria Própria (2021)

3.2.2 Configuração das portas dos *switches*

Após adicionada as VLANs em todos os *switches*, foi preciso configurar as portas de acesso e tronco. A Figura 18 mostra os comandos para a configuração das portas fastEthernet 1/ (1,2,3,4) do *switch* 1 como portas tronco.

Figura 18 - Configuração de modo tronco nas portas do *switch* 1

```
Switch1#: config t
Switch1#: interface fastEthernet 1/1
Switch1#: switchport mode trunk
Switch1#: exit
Switch1#: interface fastEthernet 1/3
Switch1#: switchport mode trunk
Switch1#: exit
Switch1#: interface fastEthernet 1/2
Switch1#: switchport mode trunk
Switch1#: exit
Switch1#: interface fastEthernet 1/4
Switch1#: switchport mode trunk
Switch1#: exit
Switch1#: interface fastEthernet 1/5
Switch1#: switchport mode trunk
Switch1#: exit
```

Fonte: Autoria Própria (2021)

A Figura 19 mostra os comandos para a configuração das portas fastEthernet 1/ (1,3,5,6) como portas tronco, que são as portas conectadas nos outros *switches*. E as portas fastEthernet 1/7 e 1/8 como portas de acesso, que estão ligadas nos hosts. E as portas de acesso devem indicar a VLAN que pertencem, neste caso, a VLAN 10.

Figura 19 - Configuração vlan nas portas do *switch* 2

```
Switch2#: config t
Switch2#: interface fastEthernet 1/1
Switch2#: switchport mode trunk
Switch2#: exit
Switch2#: interface fastEthernet 1/3
Switch2#: switchport mode trunk
Switch2#: exit
Switch2#: interface fastEthernet 1/5
Switch2#: switchport mode trunk
Switch2#: exit
Switch2#: interface fastEthernet 1/6
Switch2#: switchport mode trunk
Switch2#: exit
Switch2#: interface fastEthernet 1/7
Switch2#: switchport  access vlan 10
Switch2#: exit
Switch2#: interface fastEthernet 1/8
Switch2#: switchport  access vlan 10
Switch2#: exit
```

Fonte: Autoria Própria (2021)

A configuração do swtich 3 segue o mesmo padrão do *switch* 2. A Figura 20 mostra os comandos para a configuração das portas fastEthernet 1/ (1,4,5,6) como portas tronco e a porta fastEthernet 1/7 para o host Kali Linux 2-1 como porta de acesso.

Figura 20 - Configuração vlan nas portas do *switch* 3

```
Switch3#: config t
Switch3#: interface fastEthernet 1/1
Switch3#: switchport mode trunk
Switch3#: exit
Switch3#: interface fastEthernet 1/4
Switch3#: switchport mode trunk
Switch3#: exit
Switch3#: interface fastEthernet 1/5
Switch3#: switchport mode trunk
Switch3#: exit
Switch3#: interface range fastEthernet 1/6
Switch3#: switchport mode trunk
Switch3#: exit
Switch3#: interface fastEthernet 1/7
Switch3#: switchport  access vlan 20
Switch3#: exit
```

Fonte: Autoria Própria (2021)

3.2.3 Configuração das sub interfaces do roteador R1

Para a configuração de VLANs ficar completa é preciso configurar o roteador R1 na porta e1/1 com o endereçamento IP de cada VLAN nas suas sub interfaces. Como existem duas VLANs: VLAN 10 e VLAN 20, é necessário criar duas subinterfaces na porta do roteador, que no caso foi a sub interface e1/1.10 para a VLAN 10 e a sub interface e1/1.20 para a VLAN 20. Pode ser visto que a VLAN 10 pertence a rede 192.168.1.0/24 e a a VLAN 20 pertence a rede 192.168.2.0/24.

Figura 21 - Configuração das sub-interfaces no roteador R1

```
R1#: config t
R1#: interface ethernet 1/1
R1#: no ip add
R1#: exit
R1#: interface ethernet 1/1.10
R1#: encapsulation dot1q 10
R1#: ip add 192.168.1.1 255.255.255.0
R1#: exit
R1#: interface ethernet 1/1.20
R1#: encapsulation dot1q 20
R1#: ip add 192.168.2.1 255.255.255.0
R1#: exit
```

Fonte: Autoria Própria (2021)

3.2.4 Verificação do funcionamento das VLANs

A verificação do funcionamento das VLANs é feita em 2 etapas. A primeira etapa é através de comandos nos *switches* e roteadores para verificar a configuração feita.

Nas figuras abaixo é possível visualizar as configurações das VLANs em cada *switch* a partir do comando show *vlan-switch*, além disso, é possível ver as portas que foram configuradas, no caso essas portas não aparecerão com default ativo como por exemplo na Figura 22, que mostra a configuração das VLANs no swtich 1.

Figura 22 - Vlan configuradas no *switch* 1

```
Switch1#show vlan-switch

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa1/0, Fa1/6, Fa1/7, Fa1/8
                                                Fa1/9, Fa1/10, Fa1/11, Fa1/12
                                                Fa1/13, Fa1/14, Fa1/15
10   VLAN0010                         active
20   VLAN0020                         active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        1002   1003
10   enet  100010     1500  -      -      -        -    -        0      0
20   enet  100020     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        1      1003
1003 tr    101003     1500  1005   0      -        -    srb      1      1002
1004 fdnet 101004     1500  -      -      1        ibm  -        0      0
1005 trnet 101005     1500  -      -      1        ibm  -        0      0
```

Fonte: Autoria Própria (2021)

Além das configurações dos *switches*, foi realizado uma configuração inicial do roteador R1 da Figura 13, onde foi configurado as sub-interfaces para se comunicarem tanto com a VLAN 10 quanto com a VLAN 20. Abaixo segue o resultado da configuração no roteador R1, onde pode ser visto que a interface Ethernet1/1 possui 2 endereços IP, um para cada VLAN.

Figura 23 – Configuração de sub interfaces no Roteador R1

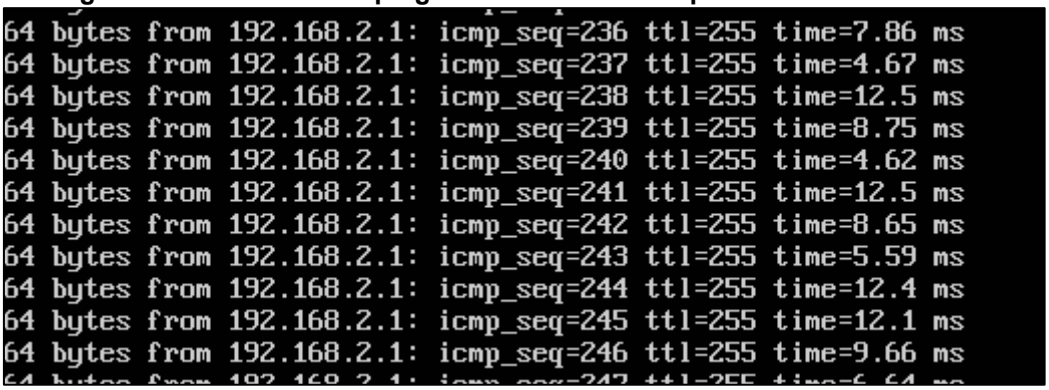
```
R1#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
FastEthernet0/0        unassigned      YES NVRAM  administratively down down
Ethernet1/0            unassigned      YES NVRAM  administratively down down
Ethernet1/1            unassigned      YES NVRAM  administratively down down
Ethernet1/1.10         192.168.1.1     YES NVRAM  administratively down down
Ethernet1/1.20         192.168.2.1     YES NVRAM  administratively down down
Ethernet1/2            unassigned      YES NVRAM  administratively down down
Ethernet1/3            unassigned      YES NVRAM  administratively down down
Ethernet1/4            unassigned      YES NVRAM  administratively down down
Ethernet1/5            unassigned      YES NVRAM  administratively down down
Ethernet1/6            unassigned      YES NVRAM  administratively down down
Ethernet1/7            unassigned      YES NVRAM  administratively down down
```

Fonte: Autoria Própria (2021)

A segunda etapa é testar a comunicação entre os hosts, que pode ser feito através do comando ping do protocolo ICMP. A Figura 24 mostra o resultado do ping do host 192.168.1.4 para o host 192.168.2.2. O resultado mostra o tempo de resposta da mensagem e, portanto, existe conectividade

entre estes 2 hosts, mostrando que a configuração das VLANs está correta, pois o pacote sai do host 192.168.1.4, passa pelo *switch* 2, *switch* 1, roteador, *switch* 1 e *switch* 2 até chegar no host 192.168.2.2. O mesmo ocorre para os outros hosts.

Figura 24 - Resultado do ping do host 192.168.1.1 para o host 192.168.2.1



Fonte: Autoria Própria (2021)

3.3 STP

As configurações do protocolo STP quanto do *Etherchannel* serão realizadas nos *switches* como mostra a Figura 13. A configuração do protocolo STP tem como função resolver o problema de loops que podem ocorrer na topologia em questão rede, por exemplo, caso as portas f1/1 e f1/3 do *switch* 2 parem de funcionar por algum motivo, o protocolo percorreu o melhor caminho possível, que no caso seria *switch* 2, *switch* 3 e *switch* 1 seguindo essa mesma ordem. Com isso a rede *VLAN* 10 não ficaria sem conexão com a rede externa por conta dessa eventualidade.

Admitindo que os passos anteriores foram configurados corretamente, pode-se configurar o protocolo STP como mostra a Figura 25 e 26. O primeiro ponto interessante é entender o comando "*spanning-tree vlan 10 root primary*", que tem como função configurar no *switch* 2, por exemplo, a rede principal ou primária, que no caso é a VLAN 10, tendo em vista que em ambos os *switches* contém as duas *VLANs* para que o STP consiga estipular o melhor caminho, com isso é possível a rede VLAN 10 que está conectada diretamente com o *switch* possa seguir os passos explicados anteriormente.

Figura 25 - Configuração STP *switch* 2

```
Switch2#: config t
Switch2#: spanning-tree vlan 10 root primary
```

Fonte: Autoria Própria (2021)

Figura 26 - Configuração STP *switch* 3

```
Switch3#: config t
Switch3#: spanning-tree vlan 20 root primary
```

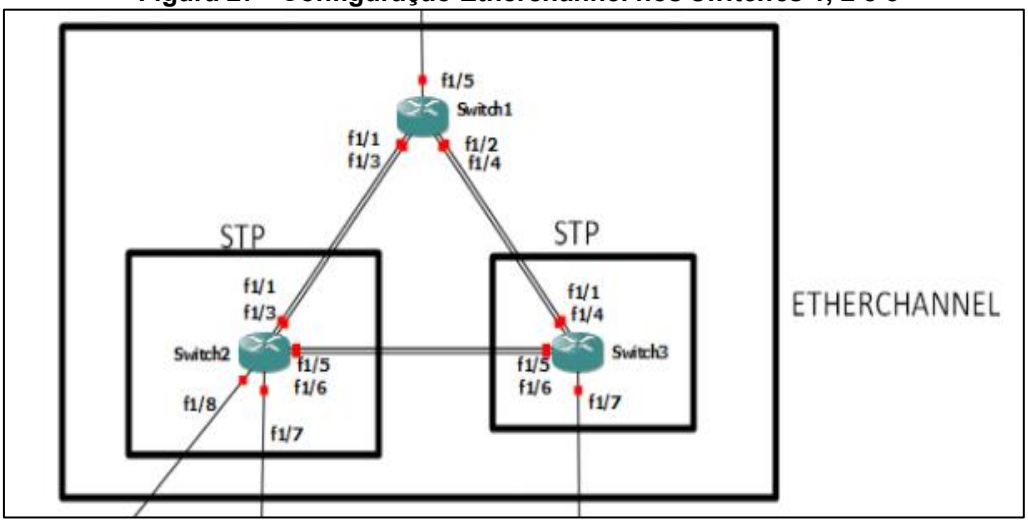
Fonte: Autoria Própria (2021)

A partir dessa configuração pode-se configurar o *Etherchannel*, mesmo um não dependendo do outro, mas sim sendo o complemento deles.

3.4 Etherchannel

Para ficar melhor exemplificado onde e como foi implementado o protocolo *Etherchannel*, usado como base a Figura 27

Figura 27 - Configuração Etherchannel nos *switches* 1, 2 e 3



Fonte: Autoria Própria (2021)

O *Etherchannel* tem como função fornecer o equilíbrio de carga para a rede, por exemplo, na Figura 27 pode-se ver que do *switch* 2 para o *switch* 1 possui dois cabos conectados entre eles, sendo f1/1 para f1/1 e f1/3 para f1/3, imagina-se que agora só existe um, ou seja essa ligação constitui em somente um ponto podendo assim ocorrer falhas ou oscilação de banda. Com isso o protocolo *etherchannel* controlou essa carga a partir dos cabos conectados, ou

seja, no caso do *switch* 2 para o *switch* 1 possuímos dois cabos conectados e ele tratou isso como somente uma ligação, sendo assim, unindo o conjunto de portas físicas em uma porta lógica. Para essa união ocorrer, deve-se criar uma área, por exemplo, *switch* 2: f1/1, f1/3 e *switch*1: f1/1, f1/3 precisam pertencer a mesma área como na Figura 28, para que o *Etherchannel* consiga realizar seus procedimentos.

Figura 28 - Configuração Etherchannel *switch* 1

```
Switch1#: config t
Switch1#: interface range fastEthernet 1/2 , fastEthernet 1/4
Switch1#: channel-group 2 mode on
Switch1#: exit
Switch1#: interface range fastEthernet 1/1 , fastEthernet 1/3
Switch1#: channel-group 1 mode on
Switch1#: exit
```

Fonte: Autoria Própria (2021)

Figura 29 - Configuração Etherchannel *switch* 2

```
Switch2#: config t
Switch2#: interface range fastEthernet 1/1 , fastEthernet 1/3
Switch2#: channel-group 1 mode on
Switch2#: exit
Switch2#: interface range fastEthernet 1/5 , fastEthernet 1/6
Switch2#: channel-group 3 mode on
Switch2#: exit
```

Fonte: Autoria Própria (2021)

Figura 30 - Configuração Etherchannel *switch* 3

```
Switch3#: config t
Switch3#: interface range fastEthernet 1/1 , fastEthernet 1/4
Switch3#: channel-group 2 mode on
Switch3#: exit
Switch3#: interface range fastEthernet 1/5 , fastEthernet 1/6
Switch3#: channel-group 3 mode on
Switch3#: exit
```

Fonte: Autoria Própria (2021)

3.5 NAT

Após a configuração dos endereços IP's e das VLANs, do protocolo STP e *Etherchannel*, o próximo passo foi configurar o NAT no roteador R1, que é o roteador da borda da rede. Ele tem a função de permitir o acesso a rede externa, por exemplo, quando o host 192.168.1.4 se comunicar com o roteador R1 192.168.1.1 foi transmitido para o ip 10.0.0.18 do roteador R1 porta e1/2 onde terá acesso a rede externa, ou seja, NAT dentro da rede (*inside*) foi configurado nas sub interfaces do roteador R1 onde terá como função traduzir o endereço IP de origem que viajam de dentro pra fora assim como o de destino que viajam de fora pra dentro, para que ocorra essa comunicação entre uma rede interna com a externa. O mesmo foi feito para a rede externa (*outside*) 10.0.0.18 do roteador R1 onde terá o mesmo objetivo que o *inside* de traduzir os pacotes que viajam de dentro para fora.

Para configurar o NAT é preciso configurar o roteador R1 na porta e1/ 2 com o endereçamento IP 10.0.0.18. Com isso configurar as subinterfaces como *inside* e a porta e1/ 2 como *outside* como mostra a Figura 31.

Figura 31 - Configuração do NAT

```
R1#: config t
R1#: interface ethernet 1/2
R1#: ip add 10.0.0.18 255.255.255.252
R1# ip nat outside
R1# no shutdown
R1#: exit
R1#: interface ethernet 1/1.10
R1#: ip nat inside
R1# no shutdown
R1#: exit
R1#: interface ethernet 1/1.20
R1#: ip nat inside
R1# no shutdown
R1#: exit
```

Fonte: Autoria Própria (2021)

Após essa configuração foi preciso criar uma lista de permissões apontando para a porta *outside*, no caso a interface e1/2 para que no momento da transição, verifique as permissões que estão disponíveis na rede, nesse caso a lista terá acesso total como mostra a Figura 32. Além disso, é preciso configurar a rota default estática que foi o IP do roteador R2 porta e1/1, essa configuração foi demonstrada mais à frente.

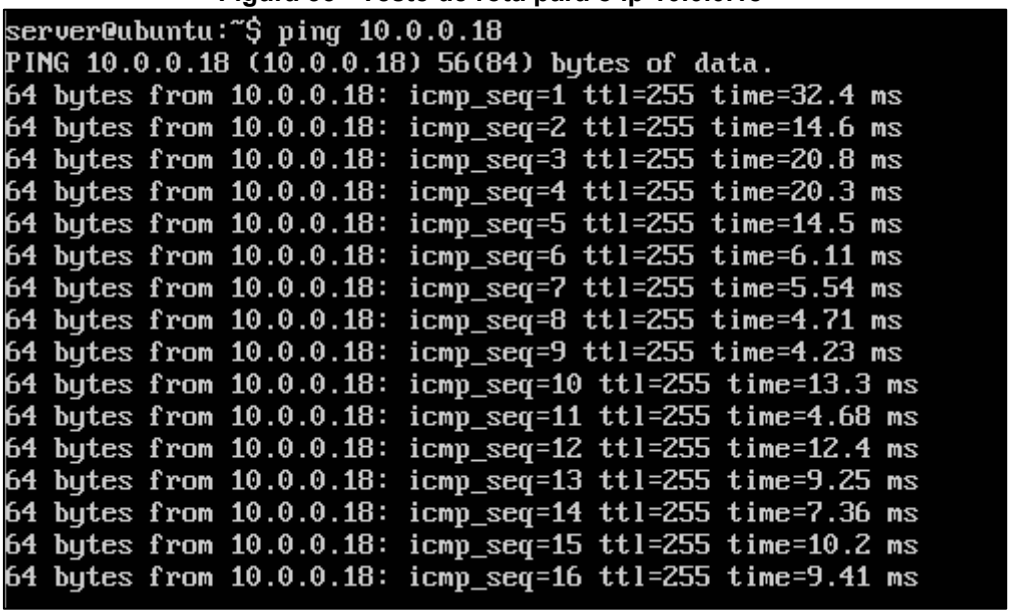
Figura 32 - Configuração da rota default e lista

```
R1#: config t
R1#: ip nat inside source list 1 int e1/2 overload
R1#: access-list 1 permit any
R1# ip route 0.0.0.0 0.0.0.0 10.0.0.17
```

Fonte: Autoria Própria (2021)

A partir dessas configurações é possível testar se a rota até o ip 10.0.0.18 está funcionando perfeitamente, através do comando ping, como mostrado na Figura 33 nos hosts server2-1, KaliLinux1-1 e KaliLinux2-1.

Figura 33 - Teste de rota para o ip 10.0.0.18



Fonte: Autoria Própria (2021)

Assim é possível ir para a próxima etapa, que foi realizado a configuração dos roteadores com o protocolo OSPF.

Para que seja possível a rede interna se comunicar com a rede externa foi preciso configurar uma rota default como foi demonstrado na Figura 32. Assim os hosts poderão acessar a rede externa com sucesso como foi demonstrado acima.

3.6 OSPF

O OSPF foi configurado nos roteadores R2, R3, R4 e R5 onde todos tem o protocolo configurado, que terá como função determinar a melhor rota, evitando tráfego congestionado ou alguma falha no roteador R3 ou R4.

Para a configuração do roteador R2, é preciso também configurar a porta e1/1 que se conecta ao roteador R1 onde ele é o intermediário para comunicação entre a rede interna e externa. Um ponto importante é a configuração das redes conectadas ao roteador, para isso foi preciso utilizar o comando network onde nele você deverá colocar o IP de rede e sua máscara invertida, por exemplo possuímos o ip 10.0.0.6 e 10.0.0.5, sedo que a máscara desses IP's é 255.255.255.252 possuindo quatro IP's no total dentre eles: 10.0.0.4 sendo o IP

de rede, 10.0.0.6 e 10.0.0.5 sendo os IP's dos dispositivos e 10.0.0.7 o IP de broadcast.

A máscara é configurada de outra forma, por exemplo, 255.255.255.240 /28 é a máscara de uma rede, para que ela se adapte ao comando network ela terá que ficar invertida. Toda máscara de rede é um número binário no caso da 255.255.255.240, ficaria 11111111.11111111.11111111.11110000 e invertendo esse valor ficaria 00000000.00000000.0000000.00001111 resultando em 0.0.0.15.

O parâmetro área tem como função delimitar a área onde o protocolo OSPF foi executado.

Figura 34 - Configuração OSPF roteador R2

```
R2#: config t
R2#: interface ethernet 1/1
R2#: ip add 10.0.0.17 255.255.255.252
R2# no shutdown
R2#: exit
R2#: interface ethernet 1/2
R2#: ip add 10.0.0.6 255.255.255.252
R2# no shutdown
R2#: exit
R2#: interface ethernet 1/3
R2#: ip add 10.0.0.9 255.255.255.252
R2# no shutdown
R2#: exit
R2# router ospf 10
R2# network 10.0.0.4 0.0.0.3 area 1
R2# network 10.0.0.8 0.0.0.3 area 1
R2# network 10.0.0.16 0.0.0.3 area 1
R2# end
```

Fonte: Autoria Própria (2021)

Após configurado o roteador R2 como mostra a Figura 34 foi preciso fazer basicamente a mesma configuração nos outros, porém respeitando as portas e os IP's de cada um. A configuração dos outros roteadores segue nas Figuras 35, 36 e 37.

Figura 35 - Configuração OSPF roteador R3

```
R3#: config t
R3#: interface ethernet 1/1
R3#: ip add 10.0.0.5 255.255.255.252
R3# no shutdown
R3#: exit
R3#: interface ethernet 1/2
R3#: ip add 10.0.0.2 255.255.255.252
R3# no shutdown
R3#: exit
R3# router ospf 10
R3# network 10.0.0.0 0.0.0.3 area 1
R3# network 10.0.0.4 0.0.0.3 area 1
R3# end
```

Fonte: Autoria Própria (2021)

Figura 36 - Configuração OSPF roteador R4

```
R4#: config t
R4#: interface ethernet 1/1
R4#: ip add 10.0.0.13 255.255.255.252
R4# no shutdown
R4#: exit
R4#: interface ethernet 1/2
R4#: ip add 10.0.0.10 255.255.255.252
R4# no shutdown
R4#: exit
R4# router ospf 10
R4# network 10.0.0.8 0.0.0.3 area 1
R4# network 10.0.0.12 0.0.0.3 area 1
R4# end
```

Fonte: Autoria Própria (2021)

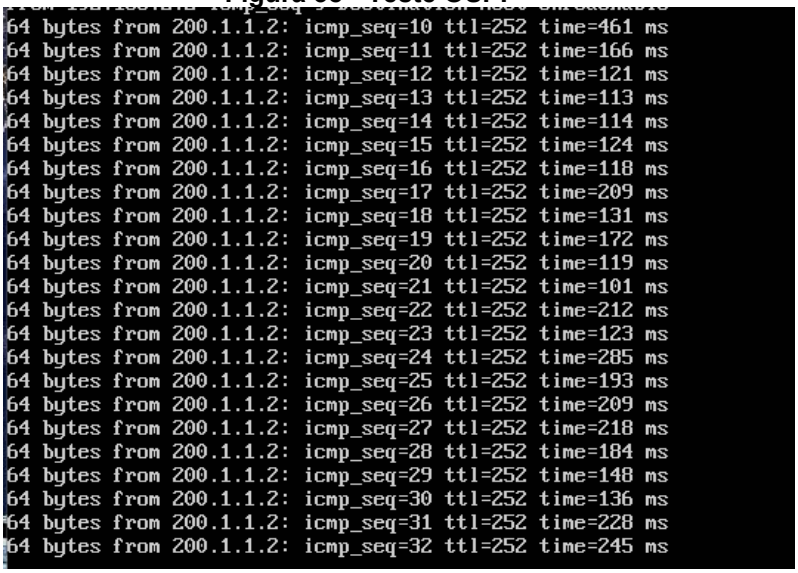
Figura 37 - Configuração OSPF roteador R5

```
R5#: config t
R5#: interface ethernet 1/1
R5#: ip add 10.0.0.1 255.255.255.252
R5# no shutdown
R5#: exit
R5#: interface ethernet 1/2
R5#: ip add 10.0.0.14 255.255.255.252
R5# no shutdown
R5#: exit
R5#: interface ethernet 1/3
R5#: ip add 200.1.1.2 255.255.255.0
R5# no shutdown
R5#: exit
R5# router ospf 10
R5# network 200.1.1.0 0.0.0.255 area 1
R5# network 10.0.0.0 0.0.0.3 area 1
R5# network 10.0.0.12 0.0.0.3 area 1
R5# end
```

Fonte: Autoria Própria (2021)

Após executada essa configuração, pode-se configurar a máquina server 1 que foi simulado o servidor da internet, como pode ser visto na Figura 37 a configuração do gateway para o servidor já foi realizada com o IP 200.1.1.2. Para verificar se a configuração ocorreu de maneira esperada, pode-se utilizar o comando ping no IP 200.1.1.2 como mostra a Figura 38. O próximo passo foi configurar o servidor 1 e testar a comunicação geral da rede, com todos os hosts se comunicando entre si.

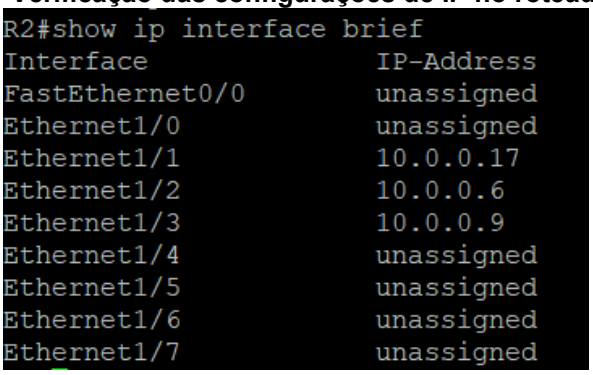
Figura 38 - Teste OSPF



Fonte: Autoria Própria (2021)

Também foi possível verificar os IP's configurados no roteador com o comando "show ip interface brief" Figura 39 e também verificar os IP's referente ao OSPF com o comando "show ip ospf neighbor" Figura 40.

Figura 39 - Verificação das configurações de IP no roteador R2



Fonte: Autoria Própria (2021)

Figura 40 - Verificação das configurações de OSPF no roteador R2

```
R2#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.0.0.13         1   FULL/BDR        00:00:38    10.0.0.10       Ethernet1/3
10.0.0.5          1   FULL/BDR        00:00:38    10.0.0.5        Ethernet1/2
```

Fonte: Autoria Própria (2021)

3.7 Configuração Servidor Internet

O servidor que foi representado como a internet, foi o servidor 1 da Figura 13, nele foi configurado o IP físico como descrito na Tabela de endereçamento IP. A configuração dele foi estática, assim como foi configurado anteriormente. Segue abaixo na Figura 41 a configuração detalhada.

Figura 41 - Configuração IP estático no server 1

```
iface enp03s3 inet static
  address 200.1.1.1
  netmask 255.255.255.0
  network 200.1.1.0
  broadcast 200.1.1.255
  gateway 200.1.1.2
```

Fonte: Autoria própria (2021)

A Figura 41 mostra a configuração do host server 1 assim como foi descrito anteriormente a configuração para o server 2. Essa configuração precisa ser realizada no arquivo /etc/network/interfaces essa configuração tem o IP do host, o IP de rede onde o host está e o gateway.

Assim que foi realizado essa configuração, todos os hosts terão acesso ao servidor de internet, para testar se essa comunicação está acontecendo de forma esperada basta executar o comando ping como mostra a Figura 42.

Figura 42 - Teste comunicação servidor de internet

Fonte: Autoria Própria (2021)

Após executado todas as configurações, é possível seguir para o último passo de configuração da topologia, que é a configuração do DHCP onde o servidor DHCP distribuiu os IP's para os hosts ligado à rede 192.168.1.0 que no caso é o host KaliLinux1-1 Figura 13.

3.8 Configuração DHCP

A configuração do servidor DHCP foi a última configuração dessa topologia. O objetivo do servidor foi fornecer os IP's para os clientes que se conectarem na rede, como por exemplo, na rede 192.168.1.0. Nesse caso teve somente um cliente que foi o host KaliLinux1-1.

Para realizar a configuração do DHCP foi preciso configurar o server2-1 como servidor e o host KaliLinux1-1 como cliente, seguindo os passos de configuração das Figuras 43 e 44.

Figura 43 - Configuração servidor DHCP

```
subnet 192.168.1.0 netmask 255.255.255.0{
   range 192.168.1.10 192.168.1.200;
   option routers 192.168.1.1;
}
```

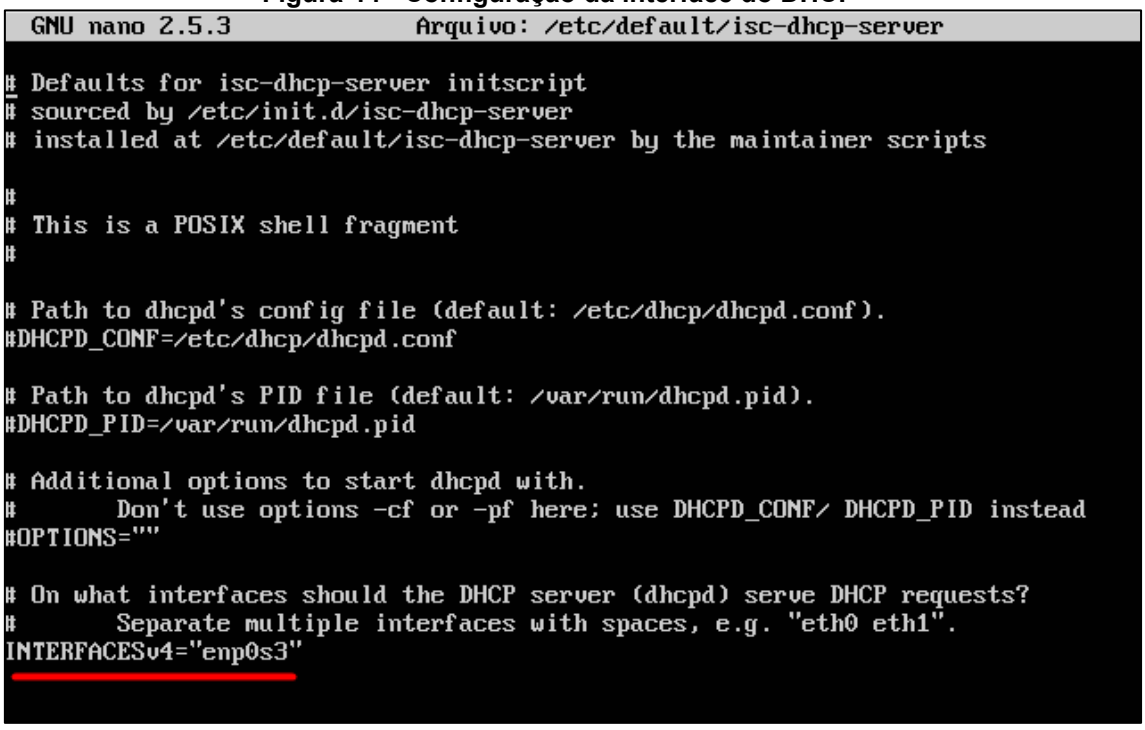
Fonte: Autoria Própria (2021)

Como mostra a Figura 43, foi preciso configurar no arquivo /etc/dhcp/dhcpd.conf uma subnet da rede que o servidor se encontra, que no

caso é a 192.168.1.0. Além disso, foi preciso definir o range de IP's que o DHCP forneceu, sendo IP's de 192.168.1.10 até 192.168.1.200 e configurar no arquivo /etc/default/isc-dhcp-server qual porta do servidor DHCP foi conectada, no caso a interface enp0s3 como mostra a Figura 44, assim o servidor DHCP poderá fornecer 190 IP's para novos hosts que se conectarem a essa rede.

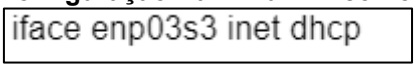
Após essa configuração pode-se iniciar o servidor DHCP com o seguinte comando: ***/etc/init.d/isc-dhcp-server start***.

Figura 44 - Configuração da interface do DHCP



Fonte: Autoria Própria (2021)

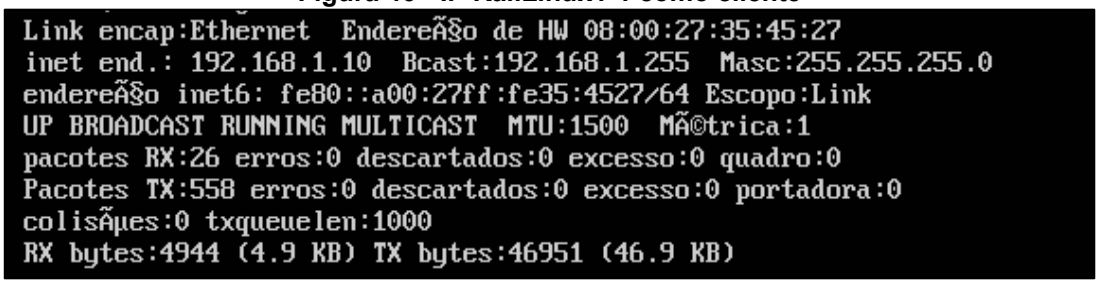
Figura 45 - Configuração KaliLinux1-1 como cliente



Fonte: Autoria Própria (2021)

Depois de configurar o host como cliente DHCP no arquivo /etc/network/interfaces seguindo a Figura 45 foi necessário reiniciá-lo, porque após a reinicialização ele receberá o IP fornecido pelo servidor, que foi 192.168.1.10 como mostra a Figura 46.

Figura 46 - IP KaliLinux1-1 como cliente



Fonte: Autoria Própria (2021)

4 CONCLUSÃO

Com a elaboração deste trabalho pode-se concluir que emulação de redes permite configurar e testar uma rede com dispositivos reais para ser implementada com maior facilidade em um ambiente real.

Após a finalização desse trabalho, foi verificado que existem muitos protocolos disponíveis para a configuração de uma rede, além disso eles executam em conjunto sendo que um complementa o outro, para maior redundância e eventualmente diminuindo possíveis gargalos na rede.

Desenvolveu-se neste trabalho uma rede de computadores com base nos objetivos específicos propostos. Primeiramente, realizou-se a pesquisa sobre Redes, *softwares* de emulação, máquinas virtuais e os protocolos mais utilizados. Após isso, foi criado uma topologia onde nela foi configurada todos os protocolos e IP's para que ocorresse o funcionamento esperado, como por exemplo a comunicação entre os hosts presente na rede.

Por fim, foi realizado testes de comunicação na rede, como por exemplo o comando "*ping*" onde foi possível visualizar a existência de uma comunicação entre os hosts.

As dificuldades encontradas na elaboração deste trabalho foram a falta de documentação para configurações especificas, como por exemplo, a exemplificação de cada parâmetro de um comando de configuração. Assim foi difícil definir algo mais especifico, sendo que alguns comandos possui uma configuração padrão onde é preciso respeitar os parâmetros propostos pela documentação.

A maior limitação deste trabalho se deve ao fato de que a inserção de mais hosts na topologia irá depender das configurações da sua máquina, sendo que quanto mais hosts sua topologia possuir, mais máquinas virtuais serão configuradas, sendo assim será preciso de mais recursos para a execução da topologia completa.

REFERÊNCIAS

AHMED, ATAB ABDUL MONEIM (A. A. M.). **Network Address Translation (NAT)**. Republic of Iraq Ministry of Higher Education and Scientific Research – University of Qadisiyah – Iraq, 2018. 26p.

CISCO, 7200. **Documentation router 7200**. 2021. *Disponível em: https://www.cisco.com/c/pt_br/support/routers/7200-series-routers/products-installation-and-configuration-guides-list.html*. Acesso em: 10 junho 2021.

CUNHA, Jaqueline de Souza. **Protocolos de roteamento dinâmico RIP e OSPF**. Fundação Educacional do Município de Assis – FEMA – Assis, 2018. 60p.

DHCP. **Dynamic Host Configuration Protocol**. *Disponível em: https://www.alliedtelesis.com/sites/default/files/documents/configuration-guides/dhcp_feature_overview_guide.pdf.* Acesso em: 21 maio 2021.

GNS3. **Getting Started with GNS3**. 2021. *Disponível em: https://docs.gns3.com/docs/.* Acesso em: 11 maio 2021.

KUROSE, James F. **Redes de Computadores E A Internet Quinta edição**. São Paulo, Pearson Universidades, 2010.

NOGUEIRA, Flavia da Silva; REIS, Guilherme Uliana; CALMON, Natalia Andrade; SILVA, Pedro Henrique; FORMIGONI, Vanessa Lourenço. **Spanning Tree Protocol**. 2016. *Disponível em: https://www.scribd.com/document/399868203/Artigo-Spanning-Tree*. Acesso em: 15 maio 2021.

PEREIRA, Diego Cesar. **Utilização de VLAN para segmentação de rede em universidade**. 5., 2018, Minas Gerais. *Disponível em: https://repositorio.uniube.br/bitstream/123456789/515/1/Diego%20C%C3%A9sar%20Pereira%20.pdf*. Acesso em 10 junho 2021.

TANENBAUM, Andrew S. **Redes de computadores 4ª edição**. Rio de Janeiro, Campus, 2001.

VMWARE. **Exemplo de configuração do EtherChannel/LACP (Protocolo de Controle de Agregação de Links) com o ESX/ESXi e os switches Cisco/HP**. 2019. *Disponível em: https://kb.vmware.com/s/article/1004048?lang=pt_PT*. Acesso em: 11 maio 2021.

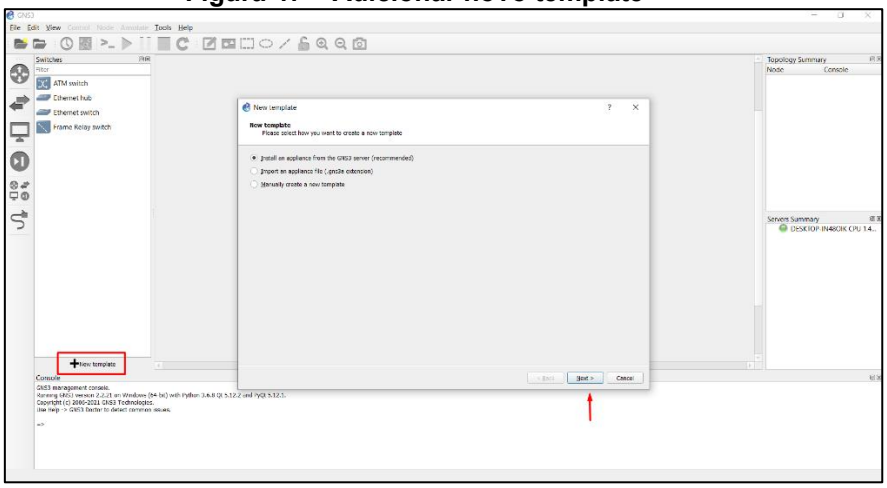
Apêndice A - Instalação e configuração dos *softwares*

Apêndice A - Instalação e configuração dos *softwares*

Processo de instalação cisco 3725:

Primeiramente é preciso adicionar um novo template.

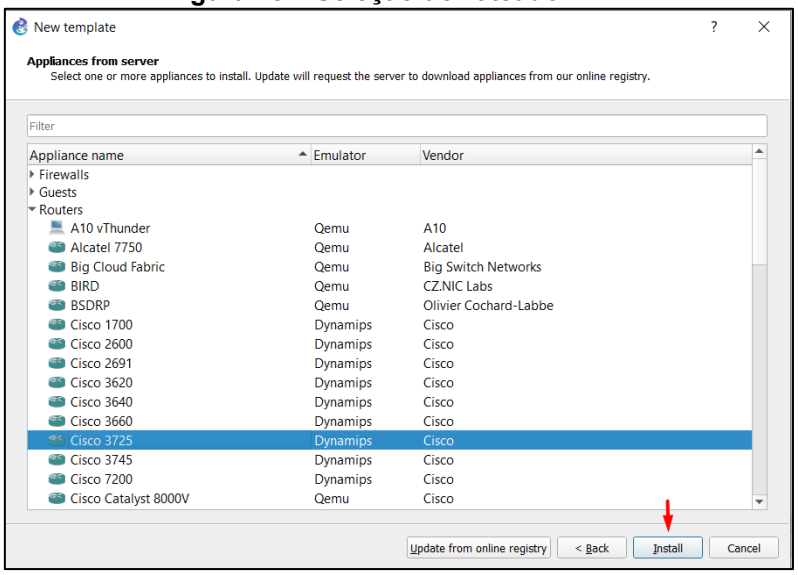
Figura 47 – Adicionar novo template



Fonte: Autoria própria (2021)

Após a execução do passo anterior é preciso selecionar o tipo de *appliance* que foi utilizado, que no caso foi um roteador Cisco 3725 (Figura 48).

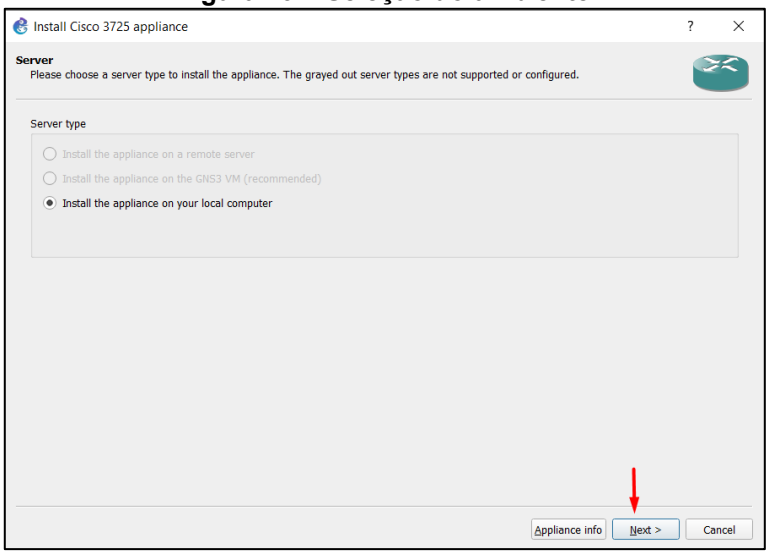
Figura 48 – Seleção do roteador



Fonte: Autoria própria (2021)

Nesse processo é preciso selecionar o ambiente em que o dispositivo foi executado, no caso o ambiente local como mostra a Figura 49.

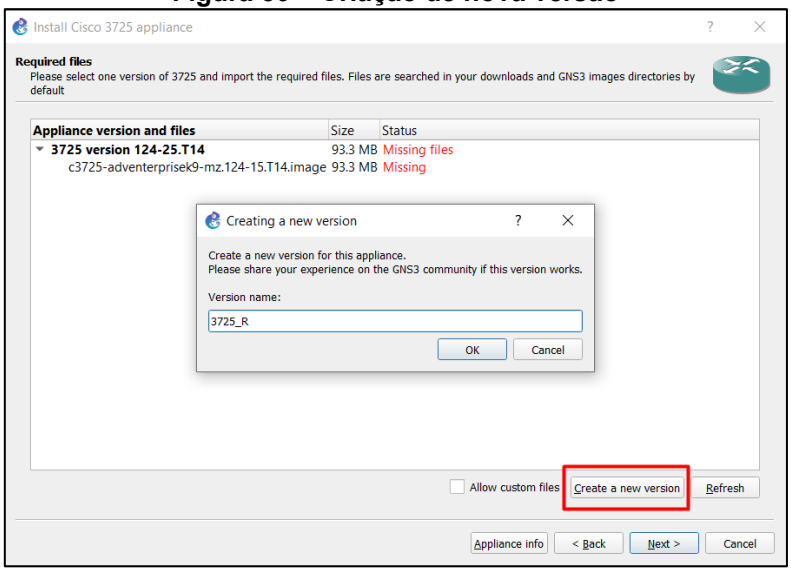
Figura 49 – Seleção do ambiente



Fonte: Autoria própria (2021)

Na etapa abaixo é possível visualizar o c3725 configurado por default, mas o aconselhável é instalar via imagem, com isso, basta selecionar nova versão e registrar um novo nome (Figura 50).

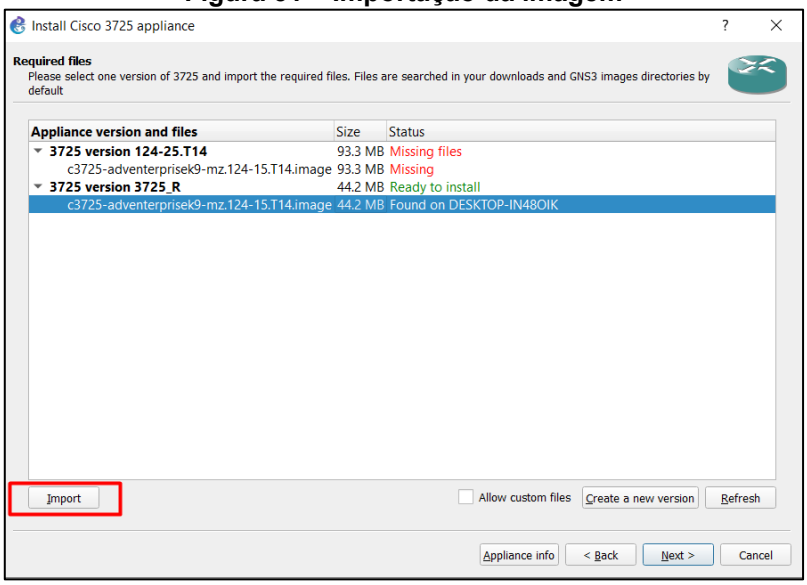
Figura 50 – Criação de nova versão



Fonte: Autoria própria (2021)

Após criada a nova versão, selecione ela e clique em importar, onde foi direcionado para selecionar o arquivo da imagem no seu computador.

Figura 51 – Importação da imagem

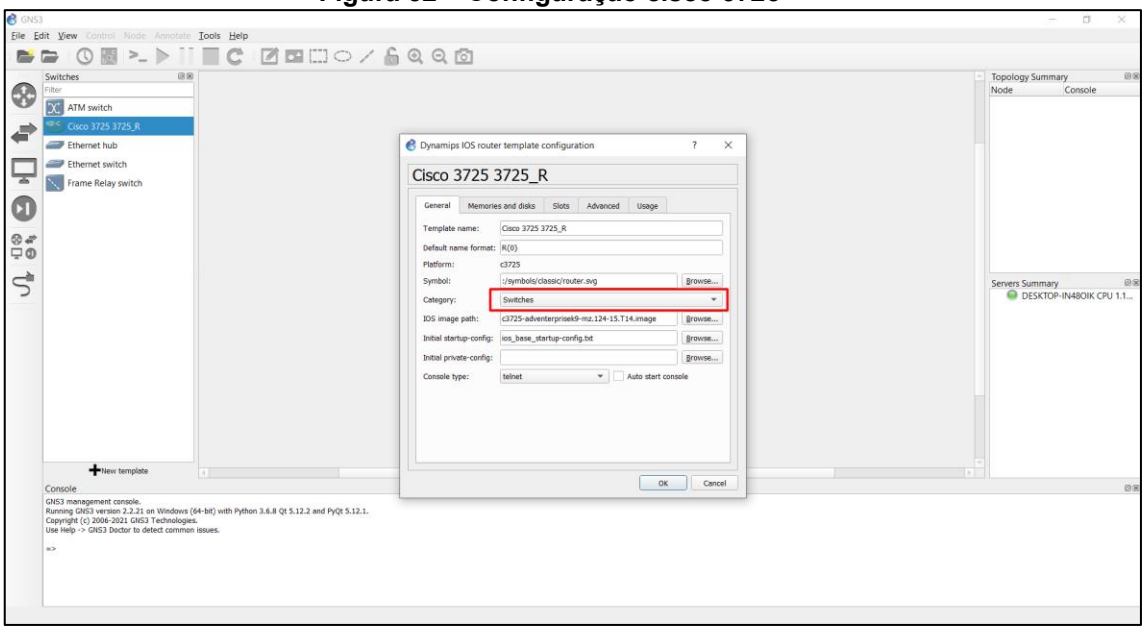


Fonte: Autoria própria (2021)

Para a configuração do roteador Cisco 7200 basta seguir esses mesmos passos.

Após a execução das etapas anteriores, precisa-se deixar pré-configurado o roteador cisco 3725 como um *switch*.

Figura 52 – Configuração cisco 3725



Fonte: Autoria própria (2021)

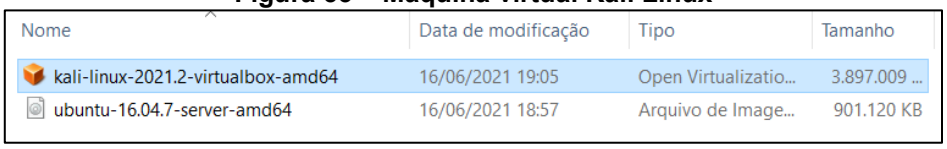
Apêndice B – Instalação Kali Linux

Apêndice B – Instalação Kali Linux

Processo de instalação Kali Linux:

Após ter a imagem compatível com a VirtualBox no Desktop basta seguir as etapas abaixo.

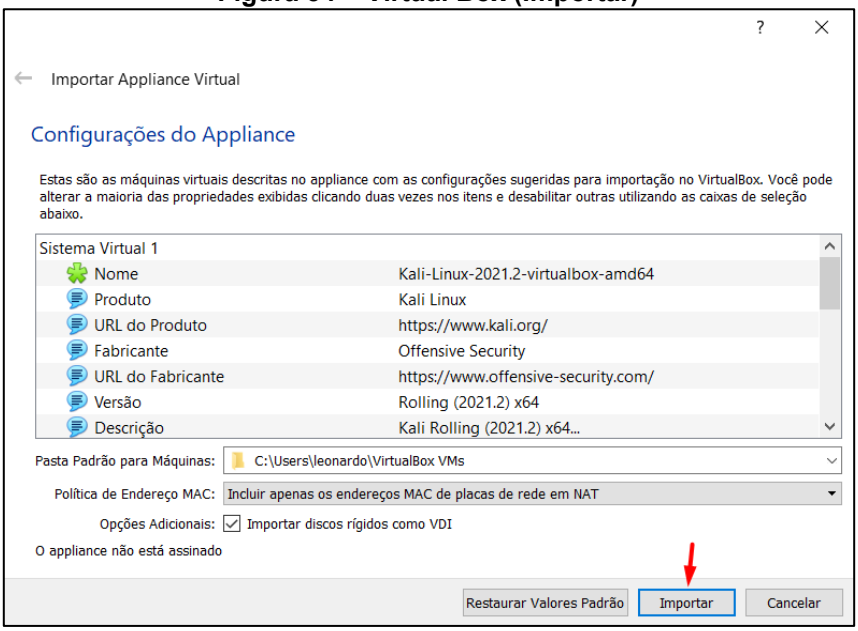
Figura 53 – Máquina virtual Kali Linux



Fonte: Autoria própria (2021)

Após selecionar a máquina virtual desejada, clique em importar e siga os procedimentos abaixo.

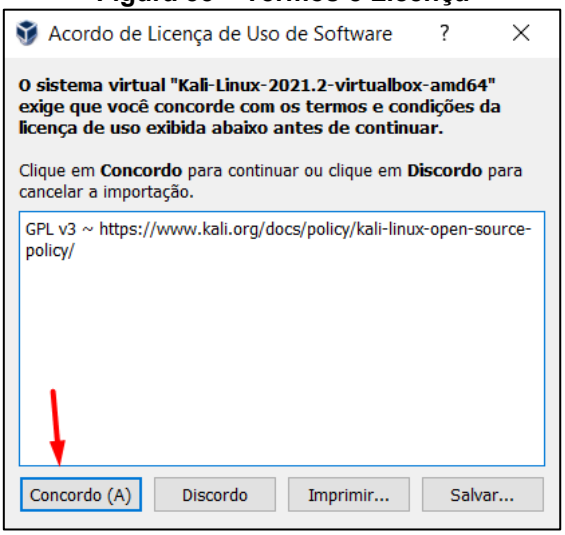
Figura 54 – Virtual Box (Importar)



Fonte: Autoria própria (2021)

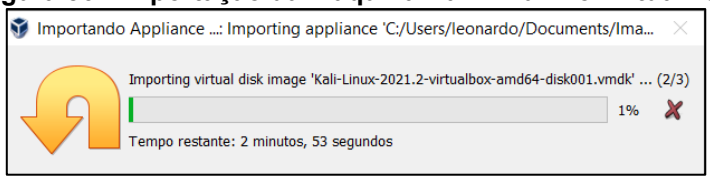
Confirme as condições da licença do sistema operacional instalado na máquina virtual.

Figura 55 – Termos e Licença



Fonte: Autoria própria (2021)

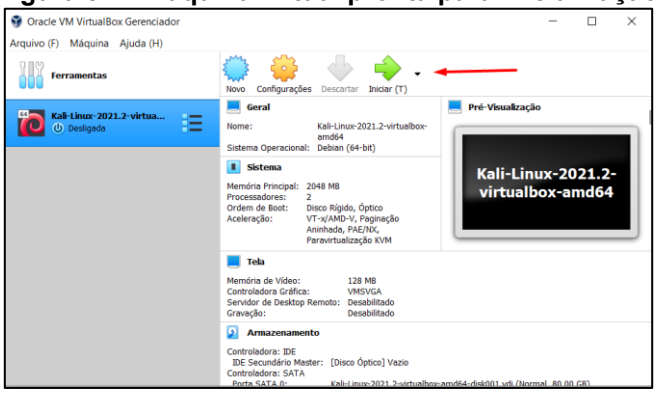
Figura 56 - Importação da máquina Kali Linux no VirtualBox



Fonte: Autoria própria (2021)

Após realizado a importação, a máquina virtual com o sistema Kali Linux já está configurada para uso.

Figura 57 – Máquina virtual pronta para inicialização

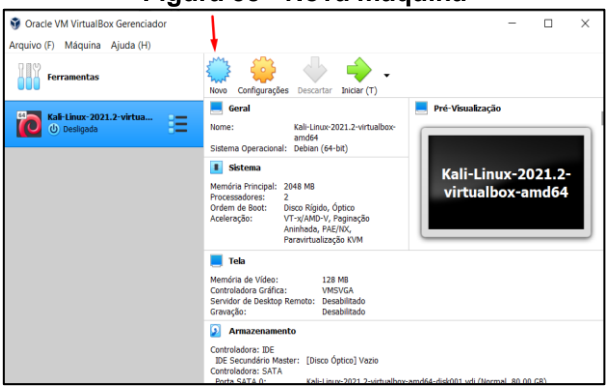


Fonte: Autoria própria (2021)

Processo de instalação Ubuntu Server 16.04 no VirtualBox.

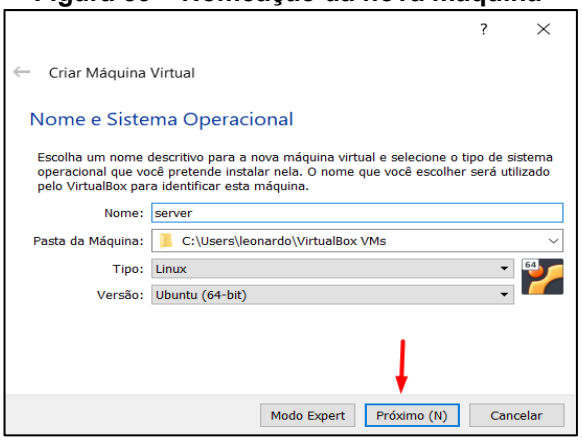
Agora é preciso criar uma nova máquina virtual, o Ubuntu Server. O processo de instalação dele é um pouco diferente da máquina anterior, nele foi utilizado a imagem real do sistema operacional.

Figura 58 - Nova máquina



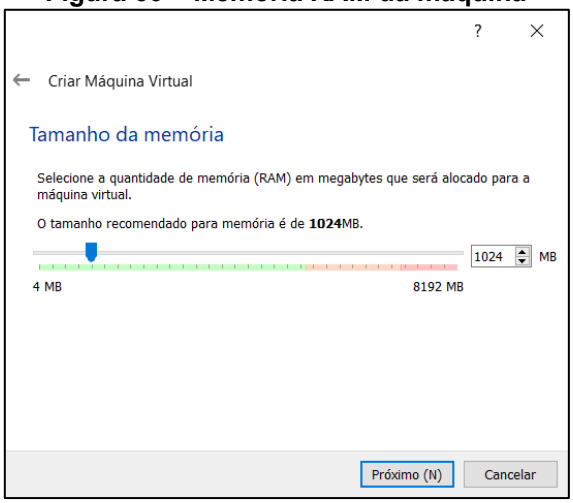
Fonte: Autoria própria (2021)

Figura 59 – Nomeação da nova máquina



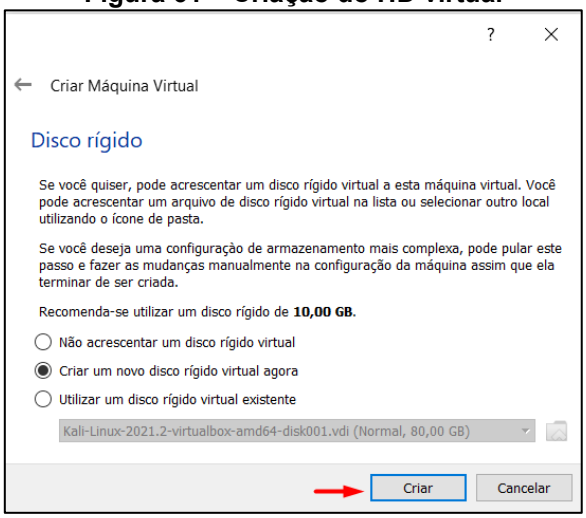
Fonte: Autoria própria (2021)

Figura 60 – Memória RAM da máquina



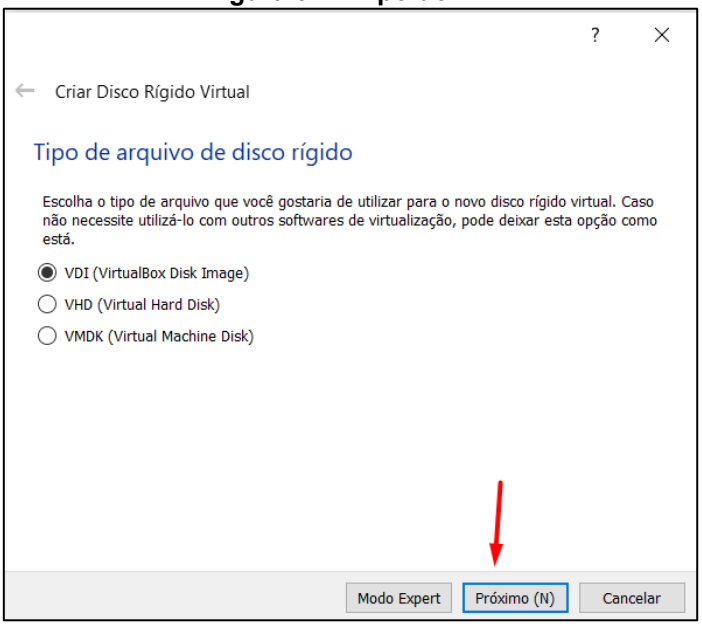
Fonte: Autoria própria (2021)

Figura 61 – Criação do HD virtual



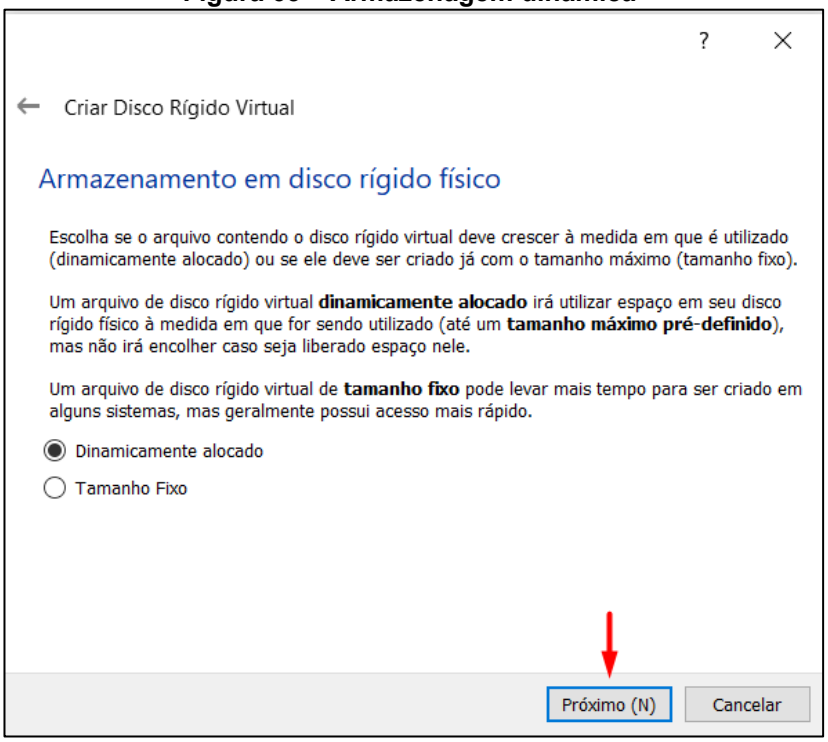
Fonte: Autoria própria (2021)

Figura 62 – Tipo do HD



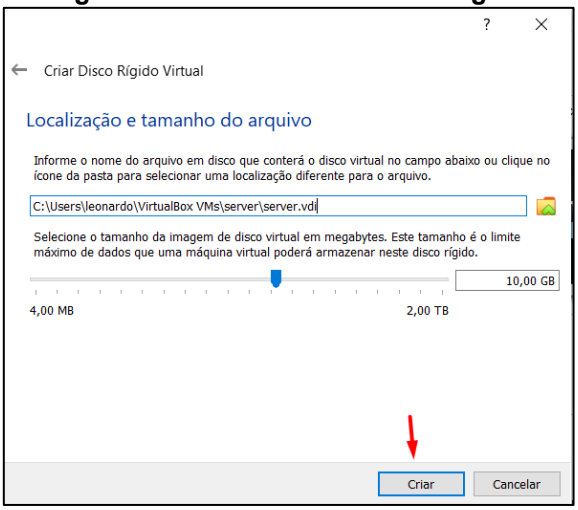
Fonte: Autoria própria (2021)

Figura 63 – Armazenagem dinâmica



Fonte: Autoria própria (2021)

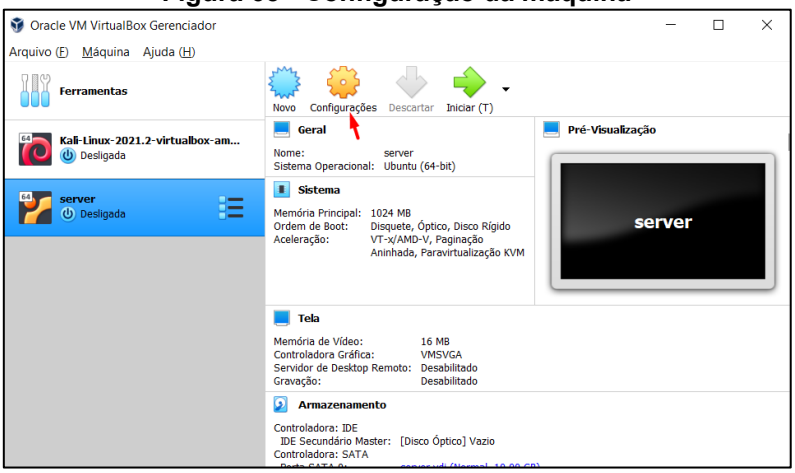
Figura 64 – Tamanho do Disco Rígido



Fonte: Autoria própria (2021)

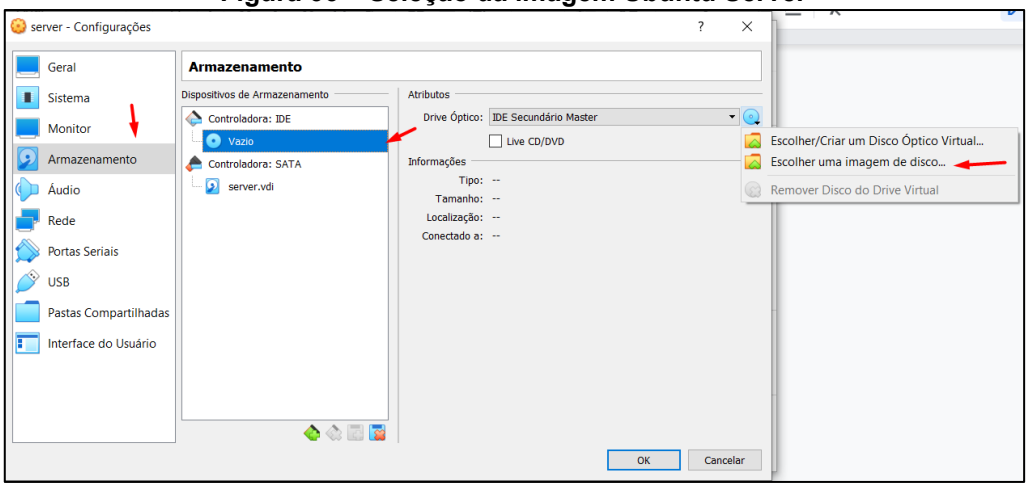
Após a conclusão das etapas anteriores, é preciso configurar a imagem do sistema operacional Ubuntu Server, assim é possível seguir as etapas de instalação do sistema operacional normalmente.

Figura 65 - Configuração da máquina



Fonte: Autoria própria (2021)

Figura 66 – Seleção da imagem Ubuntu Server

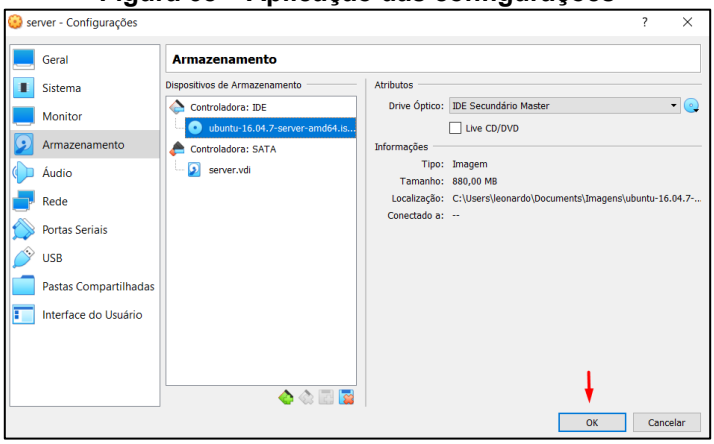


Fonte: Autoria própria (2021)

Figura 67 – Imagem Ubuntu Server

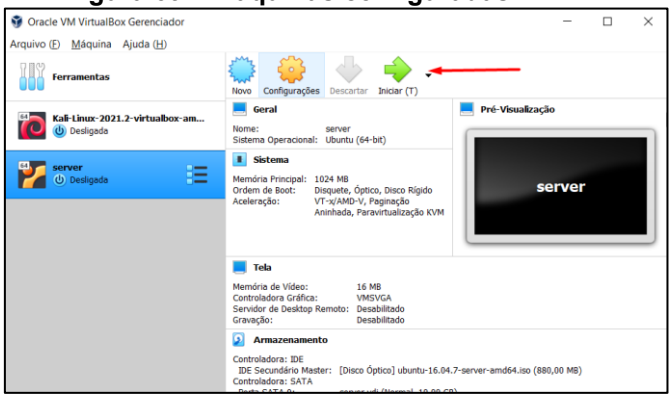
Fonte: Autoria própria (2021)

Figura 68 – Aplicação das configurações



Fonte: Autoria própria (2021)

Figura 69 – Máquinas configuradas



Fonte: Autoria própria (2021)

Concluindo todos os passos de configuração dos ambientes acima, podemos seguir para a implementação.