CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

FACULDADE DE TECNOLOGIA DE INDAIATUBA

DR. ARCHIMEDES LAMMOGLIA

ALEXANDRE LUIZ AMÂNCIO DE MORAIS

Análise do protocolo OSPF com ênfase nos seus subprotocolos

Indaiatuba

Dezembro/2023

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

FACULDADE DE TECNOLOGIA DE INDAIATUBA

DR. ARCHIMEDES LAMMOGLIA

Análise do protocolo OSPF com ênfase nos subprotocolos

Trabalho de Graduação apresentado por **(Alexandre Luiz Amâncio de Morais)** como pré-requisito para a conclusão do Curso Superior de Tecnologia em Redes de Computadores, da Faculdade de Tecnologia de Indaiatuba, elaborado sob a orientação do Prof**. (André Silva)**.

Indaiatuba

Dezembro/2023

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

FACULDADE DE TECNOLOGIA DE INDAIATUBA

DR. ARCHIMEDES LAMMOGLIA

ALEXANDRE LUIZ AMÂNCIO DE MORAIS

Banca avaliadora:

Nome:	**Orientador**
Nome:	**Avaliador(a) externo(a) – Fatec Indaiatuba**
Nome:	**Avaliador(a) interno(a)**

Data da defesa: --------/-------/-----

DEDICATÓRIA

Dedico este trabalho a todos envolvidos nesse projeto.

AGRADECIMENTOS

Agradeço a todos que participaram deste projeto, que me ajudaram a alcançar este importante objetivo.

A todos os professores envolvidos, e em especial aos professores Lincoln Peretto, André Silva, Wellington Roque, Michel Munhoz, Francisco Benedetti e Janaine Arantes, que me ensinaram tanto e me ajudaram a crescer como profissional e pessoa.

Aos meus colegas, que sempre me apoiaram e me incentivaram a seguir em frente, mesmo nos momentos difíceis.

Aos meus amigos, que sempre estiveram ao meu lado nos momentos difíceis e aos meus familiares, que me deram todo o apoio e amor que precisava para realizar este projeto.

Os levarei em minha memória com muita gratidão! Obrigado a todos.

EPÍGRAFE

“"A única maneira de aprender é fazendo."

(*Richard Branson*)

RESUMO

O estudo focou na implementação do protocolo OSPF em um ambiente virtual, destacando desafios encontrados e soluções bem-sucedidas para superá-los. Resultados experimentais foram compartilhados, incluindo as rotas dos roteadores, o algoritmo de roteamento do OSPF e as capturas de pacotes com o Wireshark.

Os dados obtidos apontam o OSPF como um protocolo eficaz e confiável, capaz de aprender rapidamente sobre redes e estabelecer rotas ideais para cada destino.

Além disso, o trabalho contribui para o entendimento do OSPF e seus subprotocolos, oferecendo uma visão detalhada de implementação e funcionamento. Pode servir como guia para futuras implementações do OSPF em ambientes virtuais.

Palavra-chave: OSPF, roteamento, redes virtuais.

Lista de Figuras

Lista de Quadros

Lista de Abreviaturas

ABRs - Area Border Routers

BDR - Backup Designated Router

BGP - Border Gateway Protocol

DR - Designated Router

EIGRP - Enhanced Interior Gateway Routing Protocol

IGP - Interior Gateway Protocol

IGRF - Interior Gateway Routing Protocol

IETF - Internet Engineering Task Force

IOS - Internetwork Operating System

IP - Internet Protocol

IS-IS - Intermediary system-Intermediary System

LSA - Link State Advertisement

LSDB - Link State Database

LSI - Link State ID

LSR - Link State Request

LSU - Link State Update

MPLS - Multiprotocol Label Switching

OS - Sistema Operacional

OSI - Open Systems Interconnection

OSPF - Open Shortest Path First

RID - Router Identification

RIB - Routing Information Base

RIP - Routing Information Protocol

SO - Sistema Operacional

SPF - Shortest Path First

VM - Virtual Machine

SUMÁRIO

INTRODUÇÃO

Desde a globalização a Internet se tornou o maior meio de difusão da informação no mundo, produzindo inúmeros benefícios à sociedade, tais como: fácil acesso ao conhecimento, colaboração entre organizações e pessoas, inclusão social, criação de valores, transações online, serviços, etc.

A cada dia, novos dispositivos e aplicações são conectados à rede, alterando o modo em que as pessoas interagem, vivem, consomem produtos, trabalham e se divertem. Muitas vezes, grandes e pequenas organizações utilizam as redes de computadores para troca de informações, consultas, análise de mercado, principalmente, pela facilidade e benefícios proporcionados pela Internet.

Sendo assim, a Internet torna-se essencial para a sobrevivência de uma empresa ou organização, independentemente de seu tamanho ou alcance de mercado. Mantê-la em funcionamento constante é sinônimo de sucesso em nível operacional, estratégico ou até mesmo de negócio de mercado.

O nosso trabalho é focado nos protocolos de roteamento que são essenciais para o funcionamento da Internet, pois permitem que os pacotes de dados sejam encaminhados até o destino, pelos roteadores de forma eficiente e confiável. Isso garante que os dados cheguem ao seu destino corretamente e em tempo hábil, o que é crucial para o bom desempenho da rede. Além disso, para as empresas, os protocolos de roteamento são importantes porque permitem que elas criem redes privadas que podem ser facilmente conectadas à Internet e a outras redes. Isso é fundamental para a comunicação e compartilhamento de dados entre diferentes departamentos, filiais e parceiros comerciais. Em suma, os protocolos de roteamento são indispensáveis para a operação das redes empresariais, permitindo uma comunicação ágil e eficiente entre diferentes departamentos e parceiros comerciais. Sem eles, o desempenho da rede seria comprometido, tornando difícil a transmissão e recebimento de dados essenciais para o sucesso das operações comerciais

No entanto, do ponto de vista dos desenvolvedores desses protocolos, criá-los e testá-los é tarefa árdua e, colocá-los em funcionamento em ambientes de

produção exige conhecimentos relevantes sobre a infraestrutura da rede e sobre o modo operacional desses protocolos. Considerando-se também as restrições de acesso e os custos elevados, especialmente quando se trata de equipamentos de alta qualidade, como os da família Cisco. Embora existam opções de simulação de infraestrutura, muitas vezes as ferramentas disponíveis não são tão completas ou fáceis de configurar quanto às soluções implementadas em hardware. Isso leva aos questionamentos sobre como configurar um ambiente de simulação semelhante aos equipamentos e IOS (Internetwork Operating System) da Cisco de maneira que auxilie administradores de rede às informações técnicas, testes, configurações relevantes e tanto de ambiente quanto do protocolo de roteamento OSPF (Open Shortest Path First) e suas mensagens.

Portanto, o objetivo deste trabalho é analisar o funcionamento do protocolo de roteamento OSPF através do serviço Quagga, suas mensagens de pacotes, as configurações, além de identificar os obstáculos durante os processos e como foram superados no desenvolvimento do trabalho. Especificamente, este trabalho tem como proposta auxiliar os administradores de rede na utilização e as configurações do protocolo de roteamento OSPF através do serviço Quagga em um ambiente virtualizado. Para alcançar esse objetivo, será realizada uma pesquisa experimental (Gil, 2017), seu objeto de estudo abrange uma topologia de rede que utilizam protocolos OSPF, através de um computador com software de virtualização, denominado Oracle VM VirtualBox. Mediante essa ferramenta será possível a instalação do sistema operacional (SO) Linux Centos 7, o serviço Quagga será instalado o que permitirá a implementação do protocolo de roteamento OSPF e as capturas de pacotes através da ferramenta WireShark.

Com a análise do funcionamento do protocolo OSPF através da ferramenta Quagga, os obstáculos e superações utilizadas para o desenvolvimento deste trabalho, espera-se auxiliar os administradores de rede a compreender melhor como utilizar e configurar o protocolo de roteamento OSPF através da ferramenta Quagga.

Quanto à estrutura, este trabalho está organizado da seguinte maneira:

Introdução, capítulo 1 que trata das referências teóricas, capítulo 2 metodologias utilizadas, capítulo 3 apresentação dos dados coletados, conclusão, referências bibliográficas.

CAPÍTULO 1

1. Fundamentação teórica

1.1 O que é roteamento?

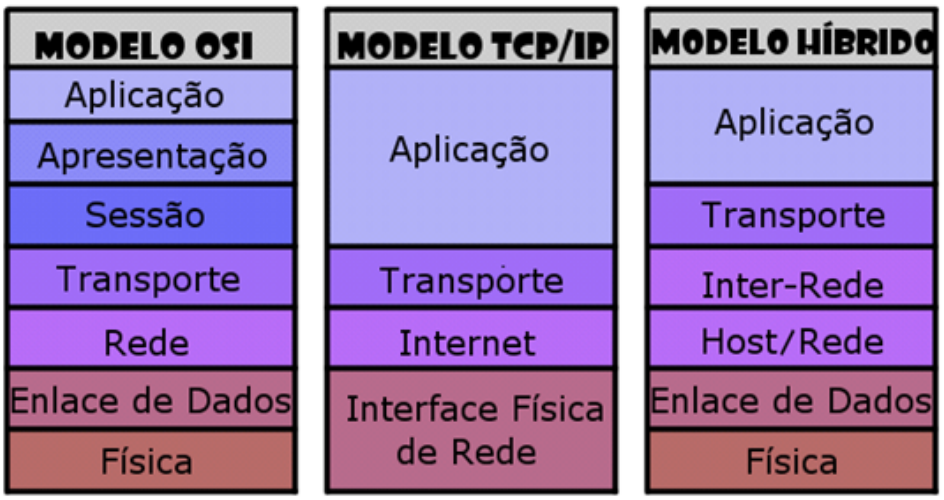
A internet faz parte do nosso cotidiano, revolucionando a forma como vivemos, nos divertimos, aprendemos e trabalhamos. Mas o que realmente é a internet? Como ela funciona? Para entendermos como a internet funciona e está estruturada, precisamos compreender o que é roteamento. Ou seja, como os pacotes de dados trafegam e localizam os seus destinos. Assim teremos uma visão da geografia global desta rede, ou seja, uma teia de enlaces conectados por roteadores possibilitando a troca e o encaminhamento de pacotes entre essas redes distintas, operacionalizada por intermédio de procedimentos padrões, os protocolos de comunicação. Em Filippetti (2016, p.237), ele utiliza o seguinte termo sintetizado em apenas uma frase, para determinar o que é roteamento: “Um conjunto de regras que definem como dados originados em uma determinada rede devem alcançar uma rede distinta”

O roteamento atua na camada de Rede do Modelo OSI (Open Systems Interconnection). A Camada de Rede provê duas funções essenciais, repasse e roteamento. Repasse é quando um pacote é transferido de uma interface de entrada para a interface de saída apropriada no mesmo roteador, ou seja, localmente. Já o roteamento é quando o pacote é transmitido de uma rede de origem diferente da rede de destino, saltando entre diferentes roteadores até chegar em seu destino final (Kurose, 2021). Ou seja, o roteamento é a determinação do caminho ideal entre hospedeiros residentes em diferentes enlaces.

Na prática, as redes funcionam através das pilhas de protocolos, que são conjuntos de pacotes e procedimentos (programas) que leem os campos do cabeçalho de cada pacote de dados e tomam decisões dependentes da camada na

qual o pacote se encontra podendo, por exemplo, encaminhá-lo para uma outra máquina ou roteá-lo para uma outra rede. Pacotes são encapsulados dentro de outros pacotes de camada inferior. Temos abaixo três exemplos de modelos de pilhas de protocolos mais utilizados nas redes.

Figura 1: Modelos de pilhas de protocolos



Fonte: Kurose (2021)

Cada camada cria o seu pacote e o encaminha para a camada seguinte que fará o mesmo, quando o pacote chegar até o destinatário então ocorre o processo de desencapsular o pacote no caminho inverso retornando camada por camada.

Cada roteador tem uma tabela de repasse, também denominada tabela de roteamento. Através dos algoritmos de roteamento esta tabela é populada com rotas. Segundo (Filippetti, 2016) cada roteador “aprende” sobre rotas remotas, não diretamente conectadas a eles, através da comunicação com os roteadores vizinhos (roteamento dinâmico), onde cada um vai ensinando aos adjacentes as suas rotas e aprendendo outras rotas a partir de outros roteadores; ou por intermédio do administrador de rede (roteamento estático). As rotas indicam a interface de saída que deve ser utilizada, dependendo do destino. Assim, pela comparação do endereço destino de cada pacote que entra no roteador, com cada entrada da tabela de rotas é que o caminho ideal é determinado.

Qual será o melhor caminho para um determinado pacote?

1.2 Os protocolos de roteamento

Existem dois tipos de roteamento: Dinâmico e estático. Na forma estática as tabelas de roteamento são construídas manualmente pelo administrador da rede que modificam e acrescentam ou modificam as rotas para cada rede na tabela. Essas listas possuem rotas pré-definidas e são mais utilizadas em rede de menor porte, com limitações de roteadores, tendo em vista menor consumo de recursos computacionais além de não utilizar largura de banda para a troca de tabelas de roteamento. As desvantagens são o requisito que depende de um conhecimento em redes como um todo, o trabalho manual de ter que adicionar cada rota manualmente e inviabiliza a implementação em redes de grande porte. Já as rotas dinâmicas possuem uma maneira diferente de descoberta de rede e de construir a tabela de roteamento, de adicionar novos dispositivos e além de poderem resolver questões complexas sendo mais rápidas e eficientes do que um administrador de sistema, as vantagens é a simplificação no processo de configuração de rede, viável e recomendado para redes de médio e grande porte, as desvantagens é o recurso computacional, a largura de banda utilizada para compartilhamento de informações de rotas, por ser um processo automatizado então deve-se tomar muito cuidado com o planejamento e compreensão, pois senão pode causar sérios problemas na rede.

Os protocolos de roteamento mais conhecidos da camada de rede são RIP (Routing Information Protocol), OSPF, EIGRP (Enhanced Interior Gateway Routing Protocol), BGP (Border Gateway Protocol), MPLS (Multiprotocol Label Switching) e IS-IS (Intermediary system-Intermediary System).

1.3 O protocolo OSPF.

OSPF é um protocolo de roteamento amplamente utilizado em redes IP, desenvolvido pelo grupo de trabalho IGP (*Interior Gateway Protocol*), da IETF (*Internet Engineering Task Force*). Em 1998, esse grupo foi criado para projetar um protocolo com base no algoritmo SPF (*Shortest Path First*) ou também conhecido como algoritmo de Edsger Wybe Dijkstra ou só Dijkstra, o nome do seu criador. O

OSPF é um protocolo aberto, ou seja, Open Source e de domínio público, cujas suas especificações podem ser encontradas nas RFC 's (*Request For Comments*)1247 e 2328.

Devido às alterações, crescimento e complexidade das redes, o protocolo RIP cada vez menos foi capaz de atender as demandas. O OSPF se difere do RIP e até do IGRP, protocolo proprietário da CISCO, que são baseados em roteamento por vetor de distância, ou seja, a cada atualização, enviam toda ou parte da tabela de roteamento para seus vizinhos.

O protocolo OSPF é um protocolo baseado no conceito de estado de link (link-state), que envia anúncio sobre o estado da conexão LSA (*link state advertisements*), para comunicar o estado das conexões de redes, informações como interfaces ligadas, métricas utilizadas, entre outras variáveis incluídas no LSAs.

Aqui estão algumas características essenciais do OSPF:

- **Roteamento link-state:** compartilha o estado de seus links (interfaces) com roteadores vizinhos imediatos, o que resulta em uma base de dados conhecida como LSDB (*Link State Database*).
- **Cálculo de rotas SPF:** com base nas informações contidas no LSDB, cada roteador pode calcular o caminho mais curto para um destino específico usando o algoritmo SPF. Isso garante que a rede encontre os caminhos mais eficientes.
- **Seleção de rotas:** quando uma rota é escolhida como a melhor, ela é inserida na tabela de rota, enquanto as outras permanecem em "*standby*" e podem ser usadas em caso de falha na rota primária.

Segundo Filippetti (2016) existem 4 modos de uma rota deixar de existir na tabela de roteamento:

- **Queda de Interface:** Caso ocorra uma queda de interface, ou seja, ela passa do estado *up* para o estado *Down*, a rota deixa de existir na tabela de roteamento para a interface diretamente conectada a ela.
- **Melhor distância administrativa:** Ocorre também quando tiver uma distância administrativa melhor, mais baixa, ou seja, se uma nova rota tiver a menor distância ela será inserida no lugar da menos eficiente.
- **Atualização de roteamento:** Caso uma rota aprendida por um roteador vizinho, pode ser removida da tabela deste vizinho por uma das razões

anteriormente mencionadas, caso ocorra uma atualização de roteamento será encaminhada pelo roteador vizinho informando que a rota em questão não deve ser mais utilizada.

- **Remoção manual:** Uma rota pode ser removida manualmente da tabela de roteamento através do administrador.

Mas caso não tenha ocorrido nenhuma das alternativas mencionadas anteriormente e suas interfaces estão em funcionamento, ou seja, estejam *UP,* então os roteadores começam a trocar pacotes *HELLO* entre si, isso ajuda a reconhecer seus vizinhos e informações de interfaces dos outros roteadores que estão em funcionamento. Quando existem mais de um roteador na rede, conhecida como redes multiacesso, o pacote *HELLO* designa um roteador DR (*designated router*) e um substituto BDR (*backup designated router*), o DR é responsável por distribuir as LSA para todos os roteadores do mesmo domínio, assim diminuindo o tráfego de rede e também o tamanho da base de dados. Uma analogia que se enquadra muito bem com esse caso é o de a escolha de um representante de classe e um vice, escolhidos para repassar todas as informações de classe e caso um não possa o seu vice assume o seu papel.

Além das características mencionadas o OSPF também oferece outros recursos adicionais, tornando uma escolha robusta para redes IP:

- **Segmentação de áreas:** O OSPF permite que as redes sejam divididas em áreas, reduzindo a complexidade do roteamento. Cada área tem sua LSDB e pode trocar informações de outras áreas por meio de roteadores de fronteira de área (ABRs(Area Border Routers)), melhorando a eficiência e a escalabilidade em redes maiores.
- **Autenticação:** O OSPF suporta autenticação para garantir a segurança das informações de roteamento trocados entre os roteadores, ajudando prevenir a inserção de informações falsas nas tabelas de roteamento.
- **Métricas Customizáveis:** Os administradores de rede podem personalizar métricas usadas pelo OSPF para calcular rotas. Isso permite ajustar o roteamento com base em requisitos específicos de desempenho ou políticas de redes.
- **Suporte IPV6:** Atualizado para suportar o IPV6.

- **Convergência rápida:** O OSPF é conhecido por se adaptar rapidamente às mudanças na topologia da rede, minimizando o impacto de falhas ou alterações.

Existem cinco tipos de mensagens e seus respectivos cabeçalhos, cada qual com sua função específica. São eles segundo imagem abaixo e a seguir descrito:

Quadro 1: Cinco mensagens OSPF

1	Hello
2	Database Description
3	Link State Request
4	Link State Update
5	Link State Acknowledgement

Fonte: https://www.gta.ufrj.br/grad/03_1/redes-industriais/ospf6.html.

- **DBD (*Data Base Descriptor*)** - Estes pacotes são responsáveis por verificar se as tabelas LSDB existentes entre dois routers vizinhos encontram-se sincronizadas. Os pacotes DBD enviam uma versão sumarizada dessa tabela para o router vizinho para certificar-se que nenhuma informação esteja faltando.
- **LSR (*Link State Request*)** - Quando um router perceber que não possui as informações de rotas que ele não possui então ele enviará um pacote LSR para o router vizinho solicitando as informações detalhadas sobre essas rotas.
- **LSU *(Link State Update*)** – Ao receber um pacote LSR então ele encaminha ao vizinho um pacote LSU com as informações solicitadas.
- **LSAck (*Link State Acknowledgment*)** – Este pacote é utilizado para confirmar se o recebimento de alguns pacotes anteriormente mencionados. Apenas os pacotes “*hello*” não são confirmados.

O OSPF envia pacotes para vizinhos para estabelecer e manter adjacências, enviar e receber solicitação, garantir entrega confiável de anúncio do link estado

entre vizinhos e para descrever o banco de dados do link-estado. O banco de dados do link-estado é gerado a partir de todos os LSAs que um roteador de área envia e recebe. O banco de dados de link-state é então usado para calcular a árvore *spanning* de caminho mais curto, usando o algoritmo SPF.

1.4 Protocolo Hello

O protocolo hello não é apenas uma saudação ou apresentação comum, mas um protocolo essencial para o mundo das redes de computadores. Desenvolvido inicialmente nas raízes da precursora da internet moderna, a ARPANET, seus engenheiros e pesquisadores da época desenvolveram essa ferramenta com a finalidade de verificar a disponibilidade de dispositivos em uma rede e garantir que ele pudesse se comunicar com os outros.

Na década de 80 o protocolo hello foi incorporado ao protocolo OSPF, com o propósito de permitir que roteadores descobrissem uns aos outros na rede e estabelecessem vizinhanças. As mensagens Hello desde então são utilizadas para verificar se os links estão funcionando e para eleger DR e o BDR. Essa eleição é relevante para ajudar a otimizar o tráfego na rede, evitando duplicação de esforço e melhorando a eficiência no geral. O protocolo funciona enviando periodicamente um pacote hello para todos os seus vizinhos, essas mensagens contêm informações como o próprio IP, IP de vizinhos e os estados de links. Caso algum roteador não receba a mensagem durante um período é determinado que o link se encontra indisponível então ele removerá o vizinho de sua tabela de roteamento.

A estrutura deste protocolo consiste em componentes essenciais, cada um desempenhando uma função específica no processo de comunicação, esses componentes incluem mensagem Hello, temporizadores, tabelas vizinhas, e gatilhos de eventos. A mensagem Hello são as mensagens de saudação para anunciar a sua presença como dispositivo, os temporizadores garantem trocas oportunas de mensagens hello, enquanto as tabelas vizinhas armazenam informações sobre dispositivos vizinhos e por fim o evento aciona ações imediatas com base nas alterações na topologia de rede.

Sem o protocolo de Hello a internet não seria como é atualmente, pois seria como se fosse uma festa sem que ninguém conheça ninguém, sendo assim caótico e ineficiente. O protocolo hello tem como objetivo facilitar a descoberta de redes, monitorar a conectividade entre eles e fornecer uma base para protocolos de roteamento e gerenciamento de redes.

A seguir serão apresentados os seguintes tipos de pacotes presentes em um ambiente OSPF.

1.5 Pacote Hello

Hello – Os pacotes “Hello” são responsáveis pela descoberta de routers vizinhos e manutenção das relações adjacentes de vizinhança entre eles. A cada 10 segundos são enviados pacotes em interfaces conectadas a redes do tipo broadcast e 30 segundos em redes do tipo non-broadcast, os intervalos são chamados de “*hello interval*”, sem receber um pacote “hello” do vizinho (4x “*hello interval*” = “*dead interval*”), o router considera esse vizinho inativo e terminará sua adjacência, eliminando as rotas aprendidas por ele e de sua tabela, informando outros vizinhos do ocorrido. Dois routers não estabelecem uma relação de vizinhança se houver incompatibilidade em qualquer um dos campos mencionados abaixo:

- **Area-id** = interfaces devem pertencer a mesma área OSPF, mesma sub-rede e a mesma máscara de rede;
- **Autenticação** = Existem três métodos de autenticação disponíveis: nenhum, ou seja, nenhuma autenticação, autenticação simples (senhas em texto transmitida na rede), e MD5 (senha não transmitida na rede, usando o algoritmo de criptografia MD5 para segurança adicional). O MD5 é um algoritmo message-digest, ou seja, é uma especificado na RFC 1321, ele é considerado o modo de autenticação OSPF mais seguro, mas ao configurar uma autenticação é necessário que toda uma área esteja configurada com o mesmo tipo de autenticação segundo descreve o documento da Cisco (https://www.cisco.com/c/pt_br/support/docs/ip/open-shortest-path-first-ospf/13697-25.html).
- **“*Hello e Dead Intervals*”** = O valor configurado do “Hello Interval e Dead Interval” devem ser consistentes em um mesmo segmento;
- **“Stub Area Flag”** = Para dois routers formarem uma conexão de relação de vizinhanças entre eles é necessário possuir o mesmo valor no campo “Stub Area Flag”.

1.6 Pacote OSPF

Apresentamos abaixo uma imagem do cabeçalho do pacote OSPF (presente em todos os pacotes), a seguir iremos descrever cada campo:

Quadro 2: Cabeçalho OSPF presente em todos os pacotes.

Version (1)	Type (1)	Packet length (2)
Router ID (4)		
Area ID (4)		
Checksum (2)		Autype (2)
Authentication (8)		

Fonte: Teleco (2023), https://www.gta.ufrj.br/grad/03_1/redes-industriais/ospf6.html

Version (Versão) – é o número da versão do OSPF.

Type (tipo) – é o tipo do pacote OSPF.

Packet Length (Tamanho do pacote) – em bytes.

Router ID (identificador do roteador) – é o identificador do roteador de origem de onde originou o pacote.

Area ID (identificador de área): é um número de 32 bits que identifica a área a que o pacote pertence. Este número será 0.0.0.0 em caso de links virtuais

Checksum (soma de verificação) – é uma verificação padrão do IP, sendo calculado o conteúdo inteiro do pacote, a fim de conferir a integridade do arquivo.

Autype – especifica a autenticação utilizada pelo pacote, podendo ser: 1. Sem autenticação – os bytes do pacote não são examinados. 2. Senha simples - os 64 bits passam sem encriptação, mas evitam que a máquina tente se juntar a uma área

que não deve. 3. Qualquer outro – sendo reservado para serem atribuídos pelo IANA.

Authentication (autenticação) – é um campo de 64 bits utilizado para autenticação.

Apresentamos na imagem abaixo o cabeçalho do pacote “hello”, descreveremos cada campo a seguir:

Quadro 3: Cabeçalho pacote HELLO

Cabeçalho OSPF (24 BYTES)		
Network Mask (4)		
Hello Interval (2)	Options (1)	Rtr Pri (1)
Router Dead Interval (4)		
Designated Router (4)		
Backup Designated Router (4)		
Neighbor 1 (4)		
Neighbor 2 (4)		

Fonte: Teleco (2023), https://www.gta.ufrj.br/grad/03_1/redes-industriais/ospf6.html

Network Mask: é a máscara associada às interfaces para as quais se deseja enviar o pacote.
Hello Interval: é o número de segundos entre os pacotes de Hello.

Options: Indica as capacidades opcionais suportadas pelo roteador.
RtrPri: é a prioridade do roteador, sendo utilizada na eleição do Designated Router e do Backup Designated Router.
Router Dead Interval: é o número de segundos antes de considerar que um roteador que está em silêncio saiu fora do ar
Designated Router: é o IP do Designated Router nesta rede, sendo 0.0.0.0 quando não houver um Designated Router.

Backup Designated Router: é o IP do Backup Designated Router nesta rede, sendo 0.0.0.0 quando não houver um Backup Designated Router.
Neighbor: são os identificadores de todos os roteadores que mandaram pacotes Hello válidos no último Router Dead Interval.

Esse tipo de pacote abaixo é enviado quando se deseja iniciar uma adjacência, tendo como conteúdo a descrição da base de dados topológica (database description), descrição esta que pode necessitar de mais de um pacote.

Quadro 4: Pacote Database Description

Cabeçalho OSPF (24 BYTES)						
0	0	Options	00000	I	M	MS
DD sequence number (4)						
LSA header (4)						

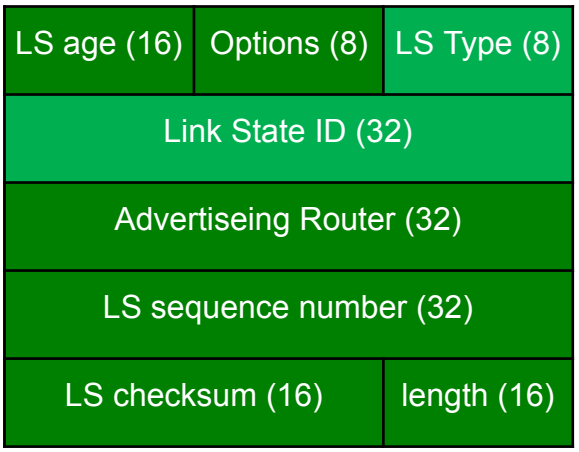
Fonte:https://www.gta.ufrj.br/grad/03_1/redes-industriais/ospf6.html

Options: este campo diz respeito às capacidades opcionais do OSPF que são diferenciação por Tipo de Serviço (*Type of Service*) e Capacidade de Roteamento Externo (*External Routing Capatibility*).
I: Quando contendo a valor 1, indica que este pacote é o primeiro de uma sequência de pacotes de descrição da base de dados
M: Quando contendo o valor 1, indica que há mais pacotes de descrição da base de dados por vir
MS: Este bit identifica quem é o *Master* e quem é o *Slave* durante o processo de troca de base de dados. O bit 1 identifica o *Master* e o 0, o *Slave*.
DD: identifica a sequências dos pacotes de descrição da base de dados

1.7 LSA header

É o cabeçalho que precede cada *link state advertisement*.

Quadro 5: Cabeçalho LSA



Fonte:https://www.gta.ufrj.br/grad/03_1/redes-industriais/ospf6.html.

LS *age*: é o tempo em segundos desde que o advertisement foi gerado

Options: este campo se refere às capacidades opcionais do OSPF

LS *Type*: é o tipo do LS. Há cinco tipos diferentes de link state advertisement:

Quadro 6: Tipos de LSA

Tipo	Descrição
Router link advertisement (tipo 1)	São anúncios originários de todos os roteadores de uma determinada área, sendo espalhados pela mesma
Network link advertisement (tipo 2)	São anúncios gerados pelo Designated Router em redes multiacesso, contendo a lista de roteadores numa determinada rede
Summary link advertisement (tipo 3, 4)	Gerado pelos área border routers e espalhados para a área associada, descrevendo rotas para destinos fora desta área, mas pertencente ao mesmo SA. O tipo 3 diz respeito a rotas para redes enquanto o tipo 4 diz respeito a rotas para roteadores de fronteira do SA (AS boundary routers).
AS external link advertisement (tipo 5)	Gerados por AS boundary routers e espalhados para toda o Sistema Autônomo, este tipo de anúncio descrever rotas para destinos fora do SA

Fonte: Filippetti (2016).

1.8 Link State ID

Este campo identifica que parte do domínio está sendo descrito pelo link state advertisement. Este campo assumirá valores dependendo do tipo do LS de acordo

com a tabela abaixo:

Quadro 7: Tipos de LSType e Link State ID

LS Type	Link State ID
1	Router ID do roteador que gerou o LSA
2	IP do DR da rede
3	ID da rede de destino
4	Router ID do ASBR
5	IP da rede de destino

Fonte: Filippetti (2016).

Advertising Router: é o Router ID do roteador que gerou o link state advertisement
LS *sequence*: este campo funciona como uma espécie de contador de advertisements, sendo utilizado na detecção de link state advertisement antigos ou duplicados
LS *checksum*: é o checksum do conteúdo completo (com exceção do LS age) do link state advertisement
Length: é o comprimento do link statement advertisement em bytes, incluindo o cabeçalho.

1.9 Link State Request

Este pacote é utilizado quando um roteador, tendo recebido as informações da base de dados topológica, envia uma mensagem requerendo informações por "perceber" a existência de informações obsoletas. Estes pacotes são enviados a roteadores que tem base de dados topológica mais atualizada. Pode ser que seja necessário o envio de pacotes desse tipo para mais de um roteador para se ter as informações

devidamente atualizadas.

Quadro 8: Cabeçalho LSR

Cabeçalho OSPF (24)
LS Type (4)
Link State ID (4)
Advertising Router (4)
.

Fonte: Teleco (2023), https://www.gta.ufrj.br/grad/03_1/redes-industriais/ospf6.html.

LS *Type*: é o tipo do LS, podendo ser um dos 5 tipos descritos anteriormente.
Link State ID: este campo identifica que parte do domínio está sendo descrito pelo link state advertisement, podendo assumir um dos valores descritos anteriormente.
Advertising Router: é o Router ID do roteador que gerou o link state advertisement.

1.10 Link State Update

Este tipo de pacote é utilizado no processo de atualização das bases de dados topológicas.

Quadro 9: Cabeçalho LSU

Cabeçalho OSPF (24)
#advertisements (4)
Link State advertisement 1 (4)
Link State advertisement 2 (4)

Fonte: https://www.gta.ufrj.br/grad/03_1/redes-industriais/ospf6.html.

advertisements: é o número de link state advertisements incluídos no pacote de update.
Link State advertisement 1,2,3, etc.: é uma lista de link state advertisements

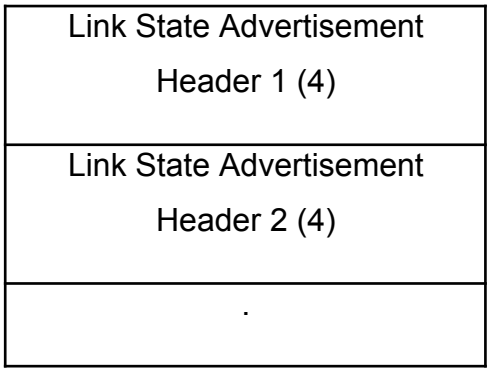
1.11 Link State Advertisement

Este tipo de pacote é utilizado no processo de atualização das bases de dados topológicas e é enviado como reconhecimento do recebimento dos Link State Updates.

Quadro 10: LSA

Cabeçalho OSPF (24)

Fonte:



https://www.gta.ufrj.br/grad/03_1/redes-industriais/ospf6.html.

Link State Advertisement Header 1,2,3, etc.: é uma lista com os cabeçalhos dos links state advertisements recebidos. Nota-se que nos dois últimos quadros os cabeçalhos são idênticos alterando somente o conteúdo do pacote.

1.12 Tipos de Áreas OSPF

O protocolo OSPF divide uma rede em diferentes áreas para melhorar a escalabilidade, a eficiência e a administração do roteamento. Veja a seguir os principais tipos de áreas OSPF:

- **Área Backbone (Área 0):** também conhecida como área 0, é o coração do OSPF, ela é responsável por conectar todas as outras áreas do OSPF, toda comunicação entre diferentes áreas deve passar pela área backbone. A backbone é conhecida como a espinha dorsal da rede OSPF, além de conectar outras partes da rede, ela contém todas as informações de roteamento e detalhes de toda parte da rede OSPF.
- **Área *Stub:*** são áreas que não possuem conexões diretas com outras áreas OSPF além da backbone, não tem função de roteamento entre as áreas OSPF. Em áreas stub os roteadores não precisam manter as informações de roteamento completas para outras áreas, o que reduz a carga de processamento e a complexibilidade. Os roteadores em áreas stub apenas conhecem rotas para redes dentro da sua própria área e a backbone. As rotas para redes externas são resolvidas pelo roteamento padrão para a área backbone. Só trocam LSAs do tipo 1 e 2, e por não trocarem LSAs do tipo 3 reduz o tráfego de roteamento nessas áreas.
- **Área Totally Stubby:** são uma variante das áreas stub e trocam somente LSAs do tipo 1, o que a torna simples em relação ao roteamento e reduz o tráfego de roteamento ao mínimo na área.

- **Área NSSA (Not-So-Stubby Áreas):** são áreas que têm permissão para injetar informações externas para dentro do OSPF por meio de um roteador especial chamado ASBR (Autonomous System Boundary Router). São úteis quando é necessário conectar o OSPF com um sistema externo, por exemplo, a internet, sem tornar a área em uma área regular que aceita todas as rotas externas. São áreas stub que podem receber LSAs do tipo 3 de áreas backbones, mas não podem propagar para outras áreas, porém podem utilizar para sumarizar rotas externas para redes stub.

  Em resumo, o OSPF utiliza diferentes tipos de áreas para segmentar e organizar redes de computadores, melhorando a eficiência do roteamento e a escalabilidade. A área backbone é o núcleo da rede, as áreas em trânsito conectam diferentes áreas e as áreas Stub são isoladas e simplificam o roteamento. A escolha do tipo de área depende das necessidades específicas da topologia e dos requisitos de roteamento de rede.

1.13 O algoritmo de Dijkstra

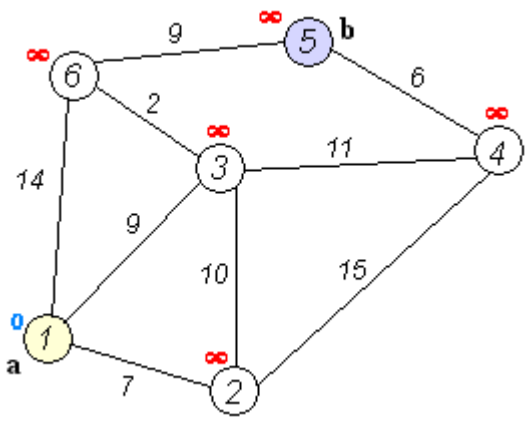
O algoritmo de Dijkstra, também conhecido como algoritmo SPF, foi desenvolvido em 1959 pelo renomado cientista da computação holandês Edsger Wybe Dijkstra. Este algoritmo é utilizado em teoria dos grafos, sendo especialmente adequado em grafos simples, ou seja, aqueles que não contém laços (arestas que conectam um vértice a ele mesmo) e nem múltiplas arestas ligando dois vértices, permitindo determinar o caminho de menor distância entre dois nós. Embora os grafos tenham diversas aplicações em áreas como análise de circuitos, planejamento de projetos, genética, transporte aéreo, web, estudos sociais, robótica, e outros campos, uma das suas aplicações mais notáveis é a otimização de percursos.

Por meio do algoritmo de Dijkstra, aliado à representação de problemas em formas de grafos , é possível calcular o caminho mais curto entre um ponto de origem a todos os outros pontos da rede, o que é crucial em diversos contextos, como roteamento de pacotes em redes de computadores.

O algoritmo SPF, que é derivado do algoritmo de Dijkstra, é amplamente utilizado em protocolos de roteamento, como o OSPF em redes internas. Seu funcionamento é detalhado da seguinte forma:

- **coleta de informações:** cada roteador coleta informações sobre suas conexões diretas e outros roteadores, inclusive detalhes de distâncias e custos associados a cada conexão. Esses dados são armazenados em uma tabela conhecida LSDB
- **seleção do ponto de origem:** o algoritmo inicia a partir do ponto de origem escolhido para começar a calcular as rotas, geralmente o próprio roteador em que o algoritmo está sendo executado.
- **iniciação:** todos os roteadores são considerados desconhecidos exceto o ponto de origem em relação aos caminhos mais curtos. O ponto de origem é configurado com a distância zero para si mesmo, uma vez que ele é o ponto de partida.
- **cálculo dos caminhos mais curtos:** o algoritmo começa a calcular o caminho mais curto para cada roteador desconhecido, isso é feito analisando as conexões diretas e escolhendo os caminhos com menor custo. À medida que o algoritmo avança, esses caminhos são continuamente atualizados para refletir as melhores rotas.
- **marcação de roteadores visitados:** a cada caminho mais curto encontrado ele marca como “visitados” evitando que sejam recalculados desnecessariamente.

Figura 2: Algoritmo de Dijkstra



Fonte: https://pt.wikipedia.org/wiki/Ficheiro:Dijkstra_Animation.gif

O ponto de partida é o nó correspondente ao número 1, e ele começa com uma contagem de 0, verificando todos os nós até o ponto b. Isso pode ser representado por uma lista que identifica os nós que já foram verificados, e na imagem, eles são mostrados em vermelho. Como iniciamos no nó 1, marcamos esse nó como visitado e após verificar a distância do nó 1 até seus adjacentes, que são os nós 2, 3 e 6, preenchendo com o peso das arestas que os conectam. Lembre-se de que ainda não foram preenchidas as rotas de melhor caminho. Em seguida, a contagem passa para o nó mais baixo diretamente conectado, do 2 ao 3 e do 2 ao 4. Depois, do 3 ao 4 e do 3 ao 6. Nota-se que o caminho do nó 1 ao nó 6 é mais eficiente através do nó 3, mudando a rota anterior para a atual. O processo finaliza com o nó de número 6 e seus correspondentes, que são do nó 6 ao 1, do 6 ao 3 e do 6 ao 5, preenchendo o peso de todos os nós e identificando a melhor rota até o destino final.

Em uma topologia de roteadores o algoritmo de Dijkstra é um algoritmo de roteamento de caminho mínimo que funciona da seguinte forma:

1. Cada roteador começa com uma tabela de rotas vazia.

2. O roteador seleciona a rede com o menor custo e adiciona essa rede a sua tabela de rotas.

3. O roteador então verifica todas as redes vizinhas da rede recém adicionada.

4. Para cada rede vizinha, o roteador calcula o custo total para chegar à rede usando o custo da rota para a rede recém adicionada e o custo da rota para a rede vizinha.

5. Se o custo total para chegar à rede vizinha for menor do que o custo total da rota existente para a rede vizinha, o roteador atualiza a tabela de rotas para a rede vizinha com o custo total menor.

6. O roteador repete os passos 3 e 5 até que todas as redes na topologia sejam adicionadas à tabela de rotas.

Em resumo, o algoritmo SPF desempenha um papel essencial na determinação dos caminhos mais eficientes em redes de computadores, garantindo que os dados sigam os trajetos mais curtos e eficazes. Isso é fundamental para

manter o desempenho e a confiabilidade das redes, especialmente em ambientes complexos e em constante mudança.

1.14 Operação do protocolo OSPF

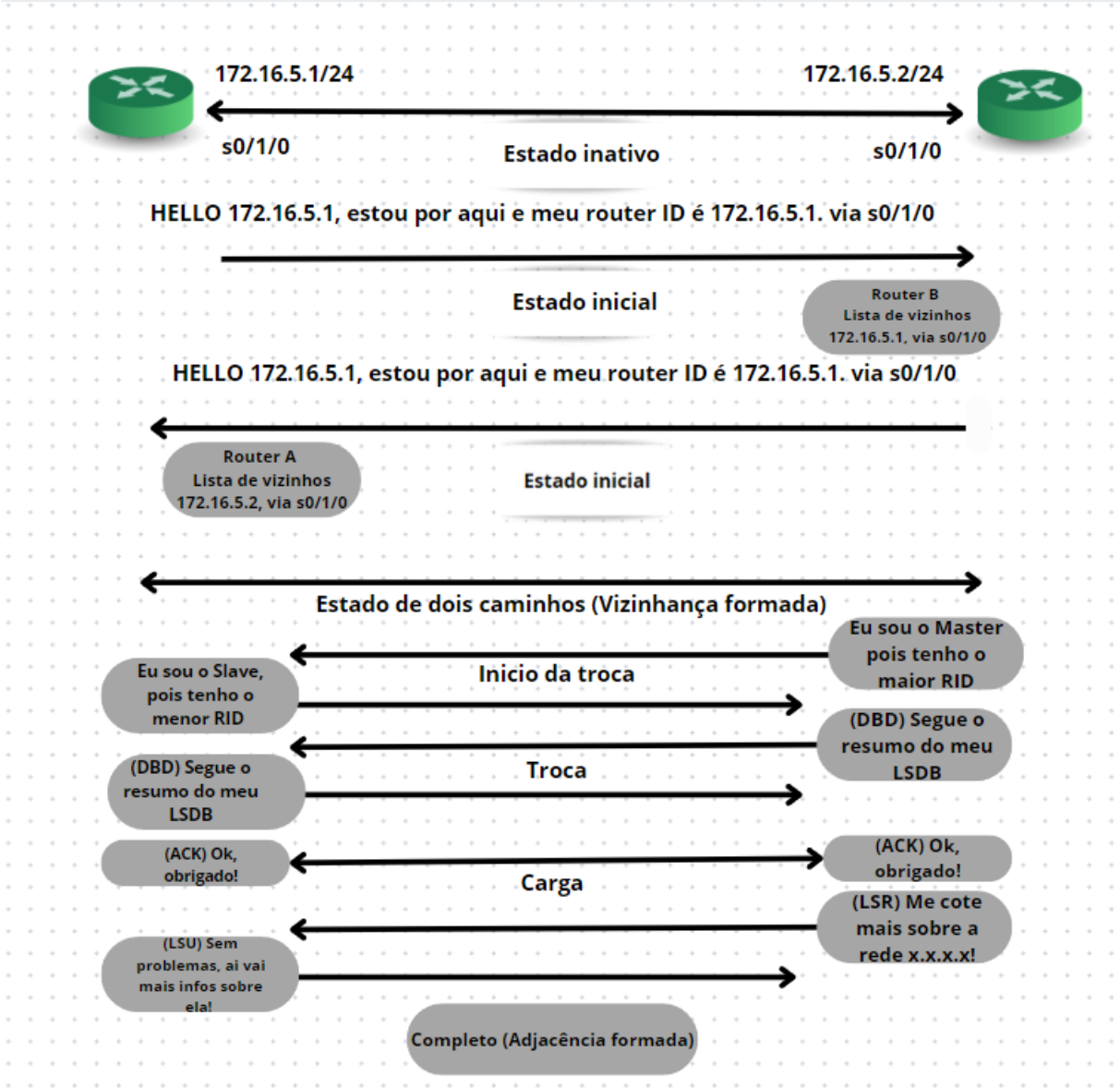
O protocolo OSPF precisa criar “vizinhança” entre os routers diretamente ligados antes de enviar informações de roteamento. Lembrando que somente os routers OSPF que compartilham do mesmo segmento podem formar uma relação de vizinhança, ou seja, os envolvidos devem pertencer a mesma área OSPF e possuir mesma sub-rede e máscara. Cada router cria vizinhança assim que recebe o RID (*router identification*) através do pacote HELLO encaminhado pelo seu router adjacente. Os ids estarão de acordo com o endereço primário, assim se havendo um endereço secundário, estes não serão utilizados, mas se caso configurados os mesmo devem pertencer a mesma área OSPF.

Em Felippetti página. 283, no caso de haver interfaces loopback configuradas, o endereço IP mais alto configurado em qualquer uma delas será adotado como o RID;

Se não houver interfaces loopback configuradas (interfaces lógicas), o endereço IP mais alto configurado em uma interface ATIVA (UP/UP) será usado como RID. Dessa forma, se tivermos um router com interface Serial configurada com o endereço IP 200.200.200.200 e uma interface *loopback* configurada com o endereço 1.1.1.1, o endereço que será o RID OSPF desse router será o da *loopback* (1.1.1.1), mesmo que o endereço 200.200.200.200 seja mais alto. O RID é fundamental que seja estável, pois já que qualquer alteração pode causar inconsistências na rede OSPF, levando isso em consideração então a interface loopback tem algumas vantagens, por serem lógicas não caem, não são conectadas a nada, o administrador pode criá-la quando bem entender e a associar a um endereço IP, reduzindo a chance de ter que alterar o endereço.

Abaixo uma imagem em relação ao processo de formação de vizinhança OSPF.

Figura 3: processo de formação de vizinhança OSPF



Fonte: Adaptação Felippetti, Autoria própria.

Em Felippetti pág. 284. “Estabelecida a adjacência OSPF, os pacotes de atualização chamados LSAs (Link State Advertisements) passam a ser trocados. Cada tipo de router gera um tipo distinto de LSA, e cada LSA possui informações e um comportamento distinto. Num primeiro momento, os routers geram LSAs do tipo 1 (chamados de “router LSA”), que contém uma descrição de todos os links (interfaces OSPF) que o router possui e seus respectivos estados. Após a sincronização da LSDB, o cálculo para determinar a melhor rota para cada rede é

executado aplicando-se o algoritmo de Dijkstra. Imagine que cada router constrói uma representação gráfica de uma árvore – como uma árvore genealógica – para cada área em que ele possua uma interface definida, colocando sua interface como a raiz e todas as outras redes arranjadas nos galhos remanescentes. Essa seria a "árvore SPF" (*Shortest Path Tree*), usada pelo OSPF para determinação de quais rotas irão para a tabela de roteamento (RIB). É importante frisar que rotas irão para a tabela de roteamento (RIB). É importante frisar que essa "árvore" contém apenas as redes originadas na mesma área OSPF na qual a interface colocada como "raiz" se encontra. Se um router possui interfaces em múltiplas áreas, então múltiplas árvores terão de ser formadas, uma para cada área."

1.15 Roteador designado e roteador designado backup.

Segundo Filippetti, o OSPF quando utilizado em uma rede ethernet, ou seja, multiacesso, o protocolo utiliza uma estratégia que reduz os números de adjacências necessárias para a operação da rede, chamado de eleição do DR (Designated router), que é responsável pela disseminação e recebimento das atualizações de roteamento, e o outro seu backup BDR(Backup designated router), caso o DR venha estar em estado DOWN então o BDR assume o seu papel. Para a eleição do DR e BDR, usa os pacotes "Hello" para examinar o valor da prioridade de cada router, o que tiver maior valor irá assumir o papel de DR e o segundo maior valor vai assumir o papel do BDR, em caso de mesmo valor de prioridade então o desempate será através do RID (Router ID), lembrando que o RID é o maior valor atribuído para um IP configurado no router ou loopback. Nota-se que a loopback possui suas vantagens em relação a estabilidade que ela proporciona (por serem lógicas não "caem") e além de que podem ser criadas a qualquer momento e ser associada a um endereço IP, devido a isso as chances de alterações são posteriormente bastante reduzidas.

1.16 A tabela master e a base de dados master.

A Base de dados master propriamente dita é conhecida como LSDB e a tabela master propriamente dita é conhecida como RIB (Routing Information Data base).

Em Filippetti, página 280, o protocolo OSPF mantém três diferentes tabelas: *neighbor table* - que contém informações dos routers OSPF vizinhos diretamente conectados*; link state database* (LSDB) ou tabela de topologia – trata-se do mapa da rede contendo uma relação de todos os routers OSPF, como eles se conectam entre si, quais redes cada um conhece e quais os custos para chegar até elas. Para cada área configurada, o router cria uma seção separada na tabela topológica listando os caminhos “aprendidos” por seus vizinhos para todas as redes pertencentes a aquela área; RIB (*Routing Information Base*) – é a tabela de roteamento, propriamente dita. As rotas com os melhores custos para cada uma das redes existentes na LSDB serão enviadas para a RIB após a execução do algoritmo SPF.

A LSDB é formada pelos roteadores do domínio OSPF e contém informações sobre o estado dos links e topologia da rede. Cada roteador envia informações sobre seus links e adjacências para seus vizinhos, que por sua vez propagam essas informações para seus próprios vizinhos. Essas informações são coletadas por cada roteador e armazenadas em sua LSDB. Dessa forma, cada roteador possui uma visão completa da topologia da rede.

Já a RIB é formada a partir da LSDB e contém as rotas escolhidas pelos algoritmos de roteamento do OSPF. Cada roteador usa as informações da LSDB para calcular as melhores rotas para cada destino. O processo de cálculo das rotas leva em consideração o custo dos links, a largura de banda, a carga e outros fatores que podem afetar o desempenho da rede.

Portanto, a LSDB é formada pelas informações de link state coletadas pelos roteadores, enquanto a RIB é formada a partir da análise dessas informações para escolher as rotas mais adequadas para cada destino.

1.16 Trabalhos correlatos

O trabalho de Thiele (2008), intitulado “ Estudo de caso implementação Interior Gateway Protocol em redes Wireless”, pela Universidade Federal do Rio Grande do Sul, apresenta um estudo e análise dos algoritmos de roteamento, definição do protocolo mais indicado para o cenário proposto, a implementação, execução e apuração dos resultados obtidos. O trabalho conclui que o protocolo OSPF foi o mais adequado para o estudo de caso, mostrando eficácia nos testes realizados, vantagens, mais estabilidade na rede.

O trabalho de Silva (2021), intitulado “Emulação de redes de computadores usando o GNS3”, pela Universidade Tecnológica Federal do Paraná, apresenta uma visão geral do software GNS3, um software de código aberto e gratuito para emulação de redes de computadores. Ele discute as seguintes questões: o que é emulação de redes de computadores, quais são as vantagens e desafios da emulação, o que é o GNS3, quais são suas principais funcionalidades e a implementação do protocolo OSPF. O trabalho conclui que o GNS3 é um software poderoso e versátil e recomenda a utilização da ferramenta para fins educacionais, de pesquisa e de desenvolvimento.

O trabalho de Guimarães (2021), intitulado “Estudo comparativo dos protocolos de roteamento RIP e OSPF usando o simulador Cisco Packet Tracer”, pela universidade Federal de Uberlândia, apresenta uma visão geral sobre o que é um protocolo de roteamento em redes de computadores e suas diferenças. Ele discute as seguintes questões: o que é roteamento, quais os principais protocolos, suas características e funcionalidades, vantagens e desvantagens, comparado os protocolos RIP, OSPF e EIGRP. com base nos critérios de seleção de rota, manutenção de tabela de roteamento, escalabilidade e performance. O trabalho conclui que o protocolo OSPF é o mais adequado para redes de grandes e complexas, enquanto o protocolo RIP é o mais simples e fácil de configurar. O protocolo EIGRP é uma boa opção para redes de médio porte.

CAPÍTULO 2

2. Percurso Metodológico

Neste capítulo será apresentado os procedimentos para implementação do protocolo OSPF através do serviço Quagga, demonstrando os componentes de hardware e os softwares utilizados neste projeto, por meio de uma descrição detalhada sobre os requisitos necessários para implementação dos componentes utilizando o ambiente virtual.

2.1. Caracterização de pesquisa

2.1.1 Quanto aos objetivos

A natureza da pesquisa escolhida é a exploratória. Segundo Gil (2002), estas pesquisas têm como objetivo proporcionar mais familiaridade com o problema, com vistas a torná-lo mais explícitos ou a constituir hipóteses.

O objeto do trabalho é realizar uma pesquisa experimental com a implementação do protocolo de roteamento OSPF, utilizando software de open source, de domínio público, chamado Quagga, para verificar a viabilidade de testar esta ferramenta em um ambiente virtual, utilizando a ferramenta de virtualização na versão gratuita, Oracle VM VirtualBox, e a ferramenta WireShark para análise das mensagens geradas pelo protocolo em estudo, assim como seu funcionamento na prática.

Será explicado detalhadamente cada linha de configuração, relacionada a instalação do serviço Quagga e as configurações do protocolo OSPF, quais são as suas funções, detectando os obstáculos durante o processo por meio de testes e como eles foram superados, documentando e explicitando a ferramenta, tendo como resultado um material a ser consultado pelos administradores de rede sempre que necessário.

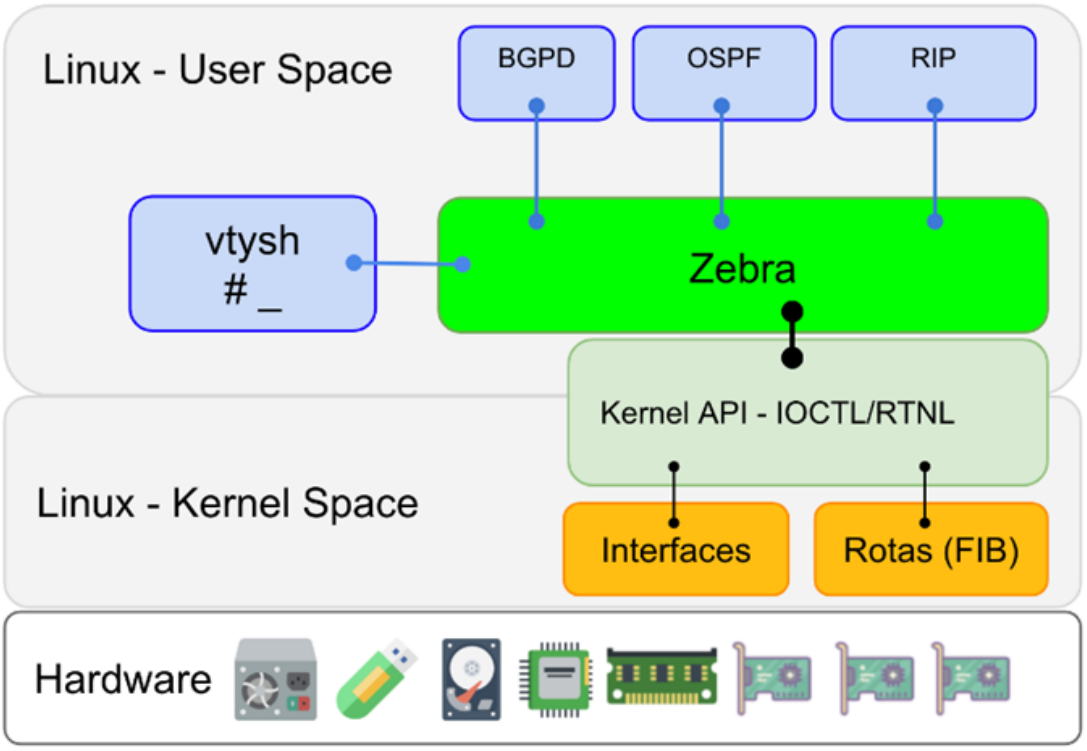
2.1.2 Caracterização do lugar e amostra de pesquisa

O delineamento dessa pesquisa será do tipo experimental, pois, será implementado um protocolo de roteamento denominado OSPF utilizando ambiente virtual Oracle VM VirtualBox e o serviço Quagga utilizado para emular os roteadores. Após as instalação e configurações, serão analisados o comportamento do protocolo OSPF e seus pacotes através da ferramenta de captura de pacotes WireShark.

2.2 Caracterização do lugar e amostra de pesquisas

O ambiente utilizado para a realização dessa pesquisa será virtual, sendo necessário a instalação do sistema operacional CentOS 7 em 5 V'M, chamadas r1, r2, r3, r4 e r5. Onde cada uma será utilizada como roteador através da ferramenta Quagga, logo abaixo uma figura que representa um pequeno resumo de como a ferramenta e seu principal framework chamado zebra atuam. Framework subjacente do Quagga que fornece uma API abstrata para implementar diferentes protocolos de roteamento. Ele separa a funcionalidade de roteamento específica do protocolo do código do SO subjacente, permitindo que o Quagga seja executado em várias plataformas. Assim então podemos realizar as coletas dos pacotes OSPF através da ferramenta Wireshark e realizar os experimentos.

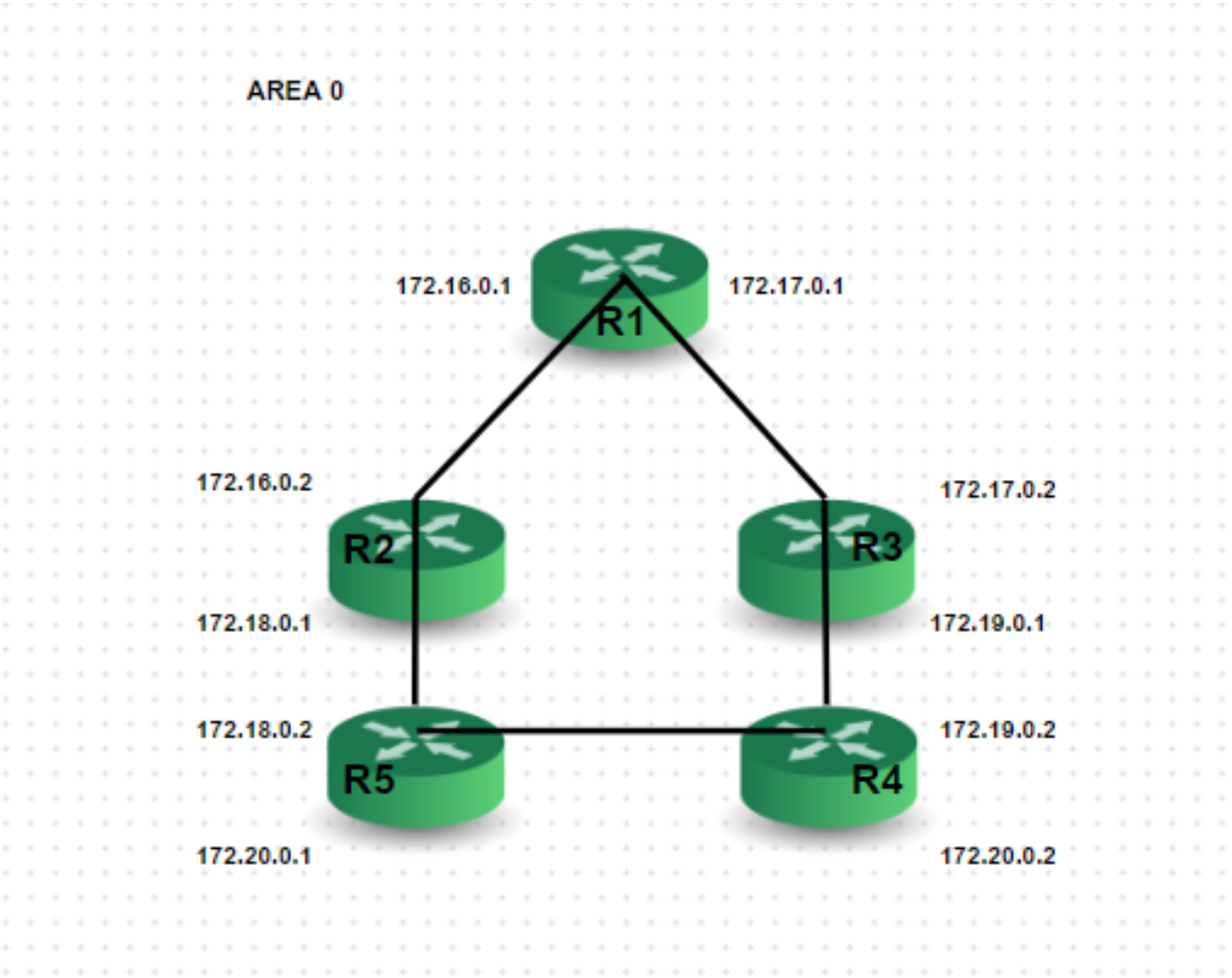
Figura 4: Arquitetura Quagga



Fonte: http://www.patrick.eti.br/?p=artigos&a=frr

A topologia de rede de roteadores utilizada neste trabalho estão todos estão na mesma área 0 ou área de backbone, cada roteador possui duas interfaces de rede que se conectam com outros roteadores.

Figura 5: Topologia OSPF



Fonte: Próprio Autor, 2023.

2.2.1 Componentes de Hardware

Quadro 11: Componentes de hardware

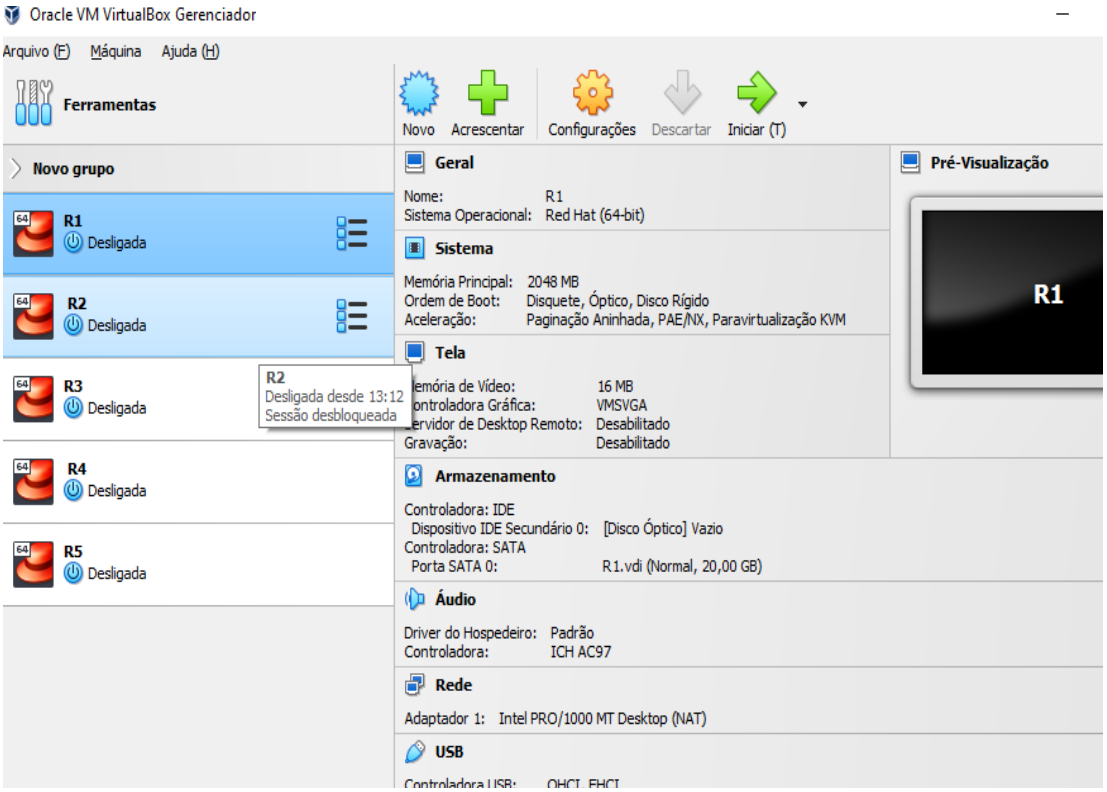
Sistema operacional	Windows 10 Home 64-bit 22h2 Windows Feature Experience Pack 1000.19044.1000.0
BIOS	R06ET39W (v1.13)
Processador	AMD Ryzen 5 5600G with Radeon Graphics 3.90 GHz
Memória	32 Gb
Placa de vídeo	AMD Radeon (TM) Graphics
Placa de som	AMD High Definition Audio Device
Espaço total do HD	280 GB
Espaço livre do HD	166 GB
Modelo do HD	Sata3 480GB SSD

Fonte: Próprio autor, 2023.

2.2.2 Componentes de Software

Nesta seção serão apresentados os softwares e aplicativos que serão utilizados na pesquisa. Para realizar a virtualização do sistema operacional será utilizado o software Oracle VM VirtualBox na sua versão mais recente 7.0.

Figura 6 : Imagem Virtual Machine.



Fonte: Próprio autor, 2023.

Cada máquina virtual representa o roteador de acordo com seu nome e ambas elas estão configuradas com os requisitos de sistema de acordo com o quadro abaixo:

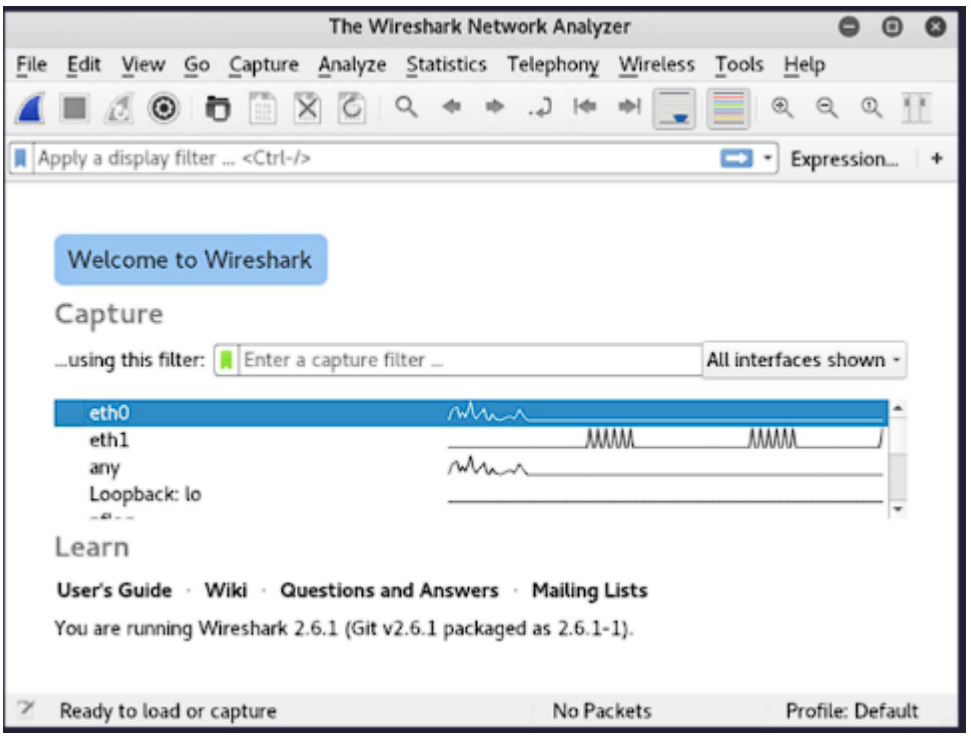
Quadro 12: Requisitos utilizados em Centos7 máquina virtual

CPU	1GB/logical CPU
Memória RAM	1024 MB
Disco Rígido	20GB

Fonte: Próprio autor, 2023.

Abaixo a figura da tela inicial da ferramenta de captura de pacotes utilizados para visualizar e entender o funcionamento e trocas dos pacotes OSPF:

Figura 7: Tela inicial Wireshark.



Fonte: Próprio autor, 2023.

CAPÍTULO 3

3. Experimentos e resultados

Neste capítulo serão apresentados os obstáculos e as dificuldades encontrados durante a implementação do protocolo OSPF, utilizando as ferramentas VM VirtualBox e quagga, como foram superadas e as recomendações bem-sucedidas de acordo com as ferramentas apresentadas neste projeto.

3.1 VirtualBox

A pesquisa se iniciou com a instalação da versão Versão 7.0.10 r158379 (Qt5.15.2) da Oracle, obtido no site oficial. Durante a instalação do Virtualbox surgiu o obstáculo referente a não apresentar suporte para sistemas operacionais arquitetura x64. Por meio de configurações da BIOS da placa mãe através de uma ativação de recursos virtuais conhecidos como VT-x (Virtualization Technology).

3.2 Sistema Operacional

Foi feita a instalação do sistema operacional para as máquinas virtuais R1, R2, R3, R4 e R5, cada uma com duas interfaces de redes, chamadas: enp0s3 e enp0s8. Sendo uma inicialmente configurada como placa em modo bridge e BOOTPROTO- DHCP, para instalação dos softwares necessários. A instalação do sistema CentOS 7 ocorreu sem nenhuma intercorrência

3.3 Quagga e seus componentes.

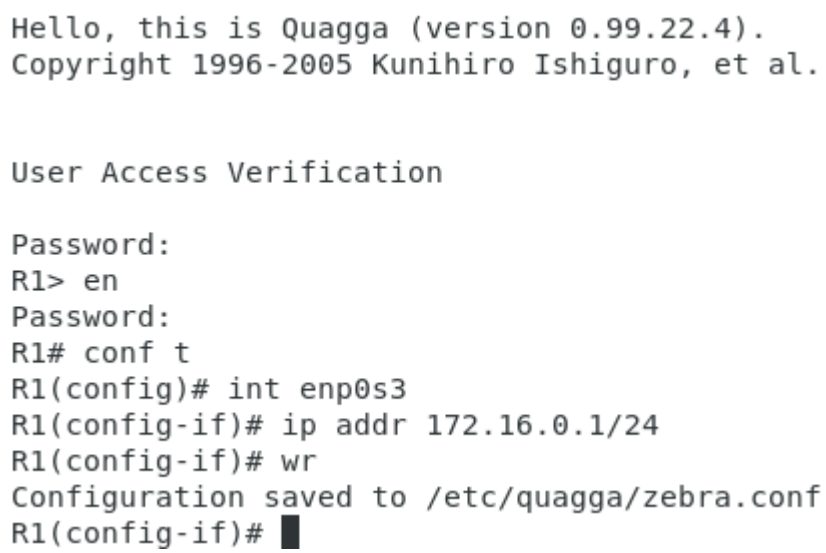
Para a realização deste projeto foram criadas cinco máquinas virtuais utilizando o sistema operacional CentOS 7. Cada uma delas com duas interfaces de rede, após as ferramentas instaladas será necessário configurar o arquivo etc/sysconfig/network-scripts/ifcfg-enp0s3 e ifcfg-enp0s8 inserindo a linha BOOTPROTO=none , para não receber nenhum endereço IP do sistema, pois o endereço será distribuído através do framework zebra.

3.3.1 Zebra.conf

Inicialmente se edita ou cria o arquivo acrescentando linha do hostname, a de senha de acesso com permissão de usuário e superusuário por último, para acesso quando conectar, também é necessário dar as permissões de grupo, acesso de arquivos e diretórios corretas no linux, além de reiniciar os serviços de rede e o serviço zebra no caso utilizado, é necessário isso sempre que houver alterações no arquivo.
A configuração do arquivo zebra.conf podem ser feitas via telnet ao acessar a porta 2601 e utilizando os comandos de configurações de acordo com a abaixo que foi utilizado para configurar

. Abaixo imagem dos comandos utilizados como exemplo para configurar as interfaces da máquina virtual r1:

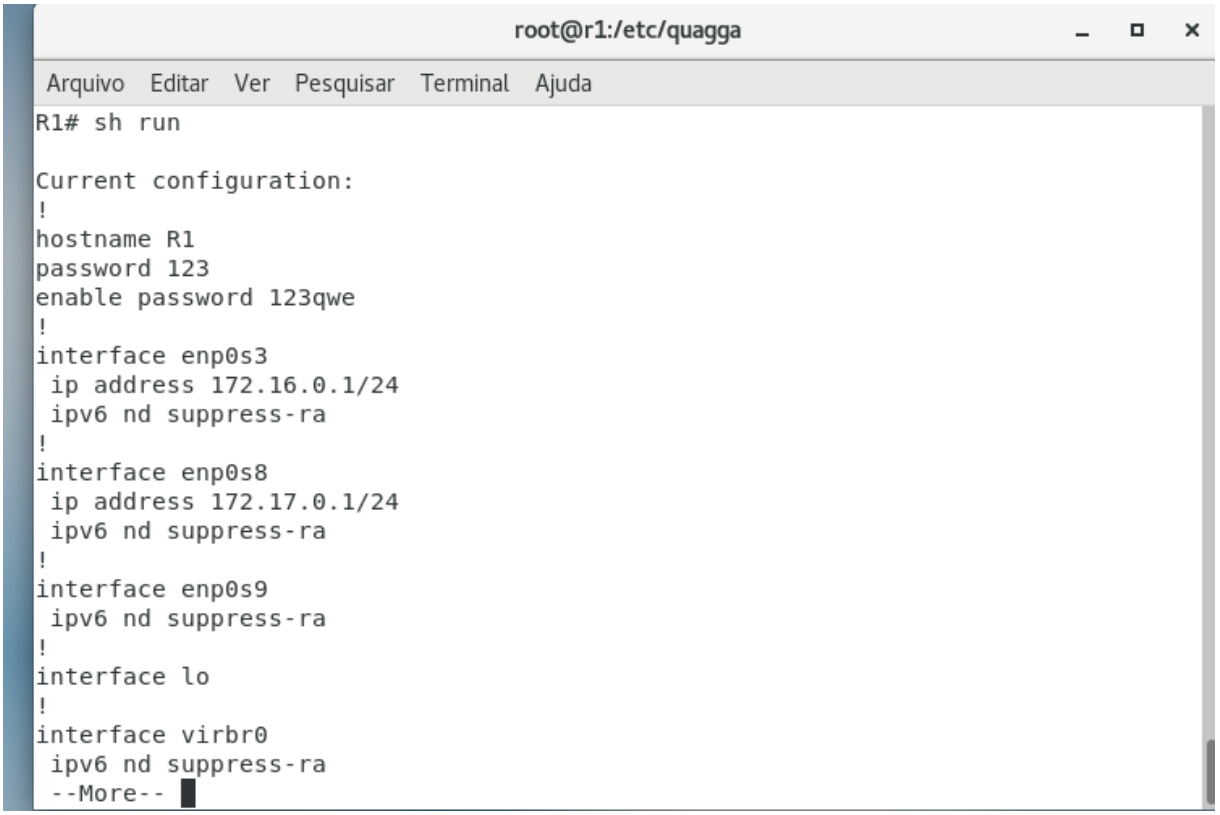
Figura 8: Comandos utilizados para configurar arquivo zebra.conf de r1.



Fonte: Próprio autor, 2023.

Ao iniciar e acessar o prompt de comando do router via telnet como usuário, o en ou enable é para acessar como super admin, conf t para configurar o terminal e int para apontar a interface que será configurada, por fim o wr de write para gravar as configurações no arquivo no linux /etc/quagga/zebra.conf.
Os comandos utilizados para realizar as configurações também são aplicados em cada máquina virtual e as interfaces de rede de acordo com a topologia proposta.
A outra forma é editar o arquivo que está localizado em /etc/quagga/zebra.conf , e fica igualmente ao arquivo editado via telnet, de acordo com a imagem abaixo:

Figura 9: Configuração arquivo zebra.conf de r1.



Fonte: Próprio autor, 2023.

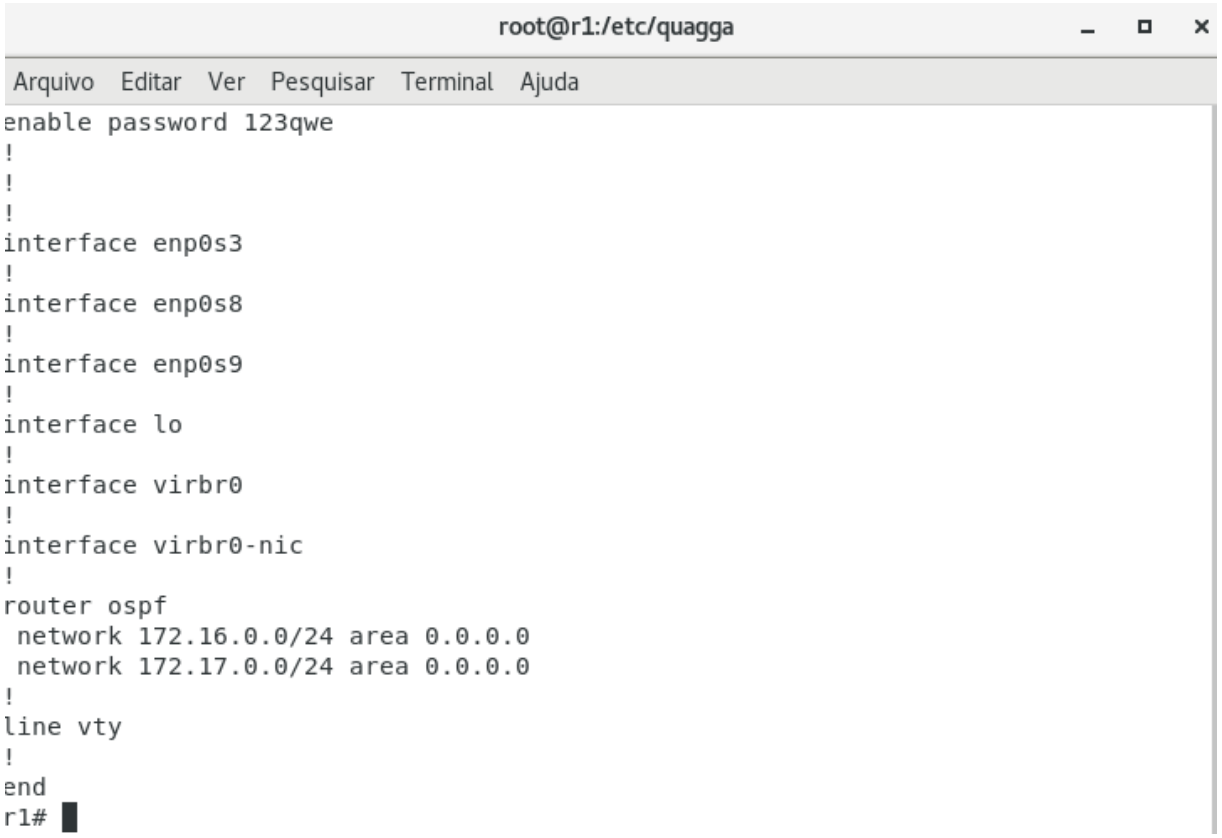
Com as configurações realizadas em todas as máquinas virtuais, agora é possível a troca de pacotes entre as interfaces diretamente conectadas, testadas via ICMP ou ping , mas ainda não aprenderam rotas, pois ainda não existe nenhum protocolo de roteamento em funcionamento.

3.3.2 Ospfd.conf

A configuração do arquivo ospfd.conf podem também serem feitas via telnet na porta

2604, apontando para cada nome das interfaces utilizadas, no caso enp0s3 e enp0s8, atribuir o endereço da rede de acordo com a topologia proposta, lembrando que por via telnet para configuração é mais seguro, pois não há risco de sintaxe errada. Também é possível configurar manualmente o arquivo que está localizado em /etc/quagga/ospfd.conf, e ficaria igualmente o arquivo editado via telnet, de acordo com a imagem abaixo:

Figura 10: Configuração arquivo ospfd.conf



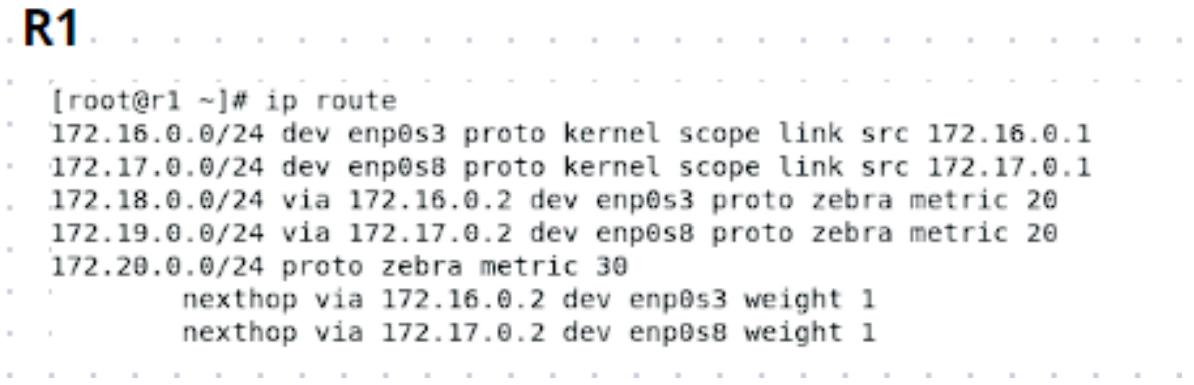
Fonte: Próprio autor, 2023.

Com as configurações adequadas do arquivo ospfd.conf em cada máquina virtual que então passam a funcionar como um routers e aplica os conceitos do protocolo de roteamento OSPF na prática que serão capturados via a ferramenta Wireshark, as rotas também podem ser consultados via comando ip route no prompt de comando ou via telnet com comando show ip ospf route.

Nesta topologia todos os roteadores são vizinhos diretos

Quando os roteadores inicializam o protocolo OSPF, eles enviam mensagens de estado de link para todos os seus vizinhos, essas mensagens contêm informações sobre as redes diretamente conectadas a cada roteador.

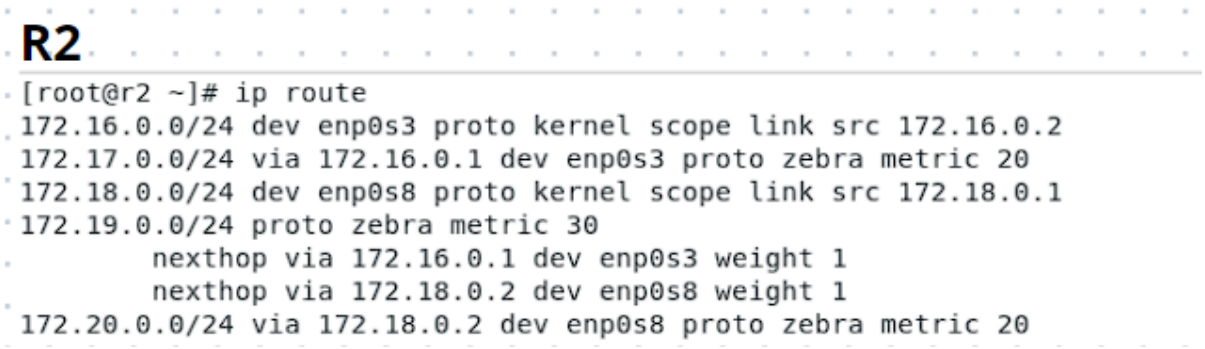
Figura 11: Rotas de R1



Fonte: Próprio autor, 2023.

No caso do R1, ele envia mensagens de estado de link para R2 e R3, essas mensagens informam que ele está conectado às redes 172.16.0.0/24 e 172.17.0.0/24. Tambem informa que para chegar a rede 172.18.0.0/24 ocorre via a interface 172.16.0.2 com métrica em 20, e para rede 172.19.0.0/24 via interface 172.17.0.2/24 com custo 20, por fim para chegar a interface 172.20.0.0/24 tem custo 30 e possui dois caminhos para o nexthop via 172.16.0.2 e via 172.17.0.2.

Figura 12: Rotas de R2

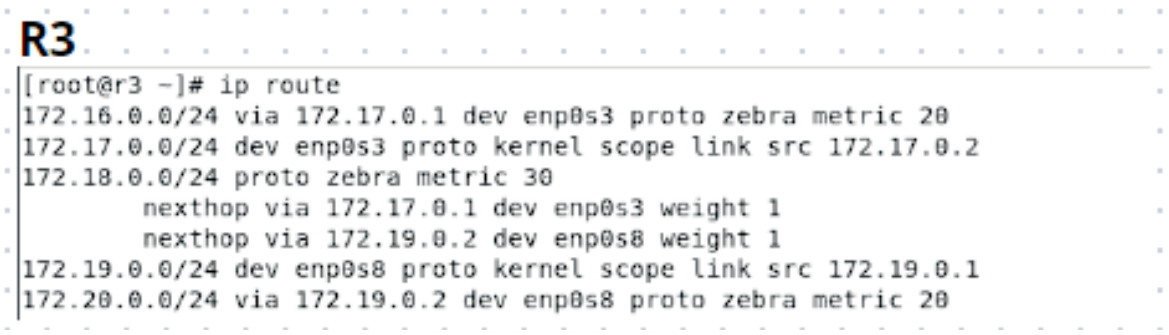


Fonte: Próprio autor, 2023.

No caso do R2, ele envia mensagens de estado de link para R1 e R5, essas mensagens informam que ele está conectado às redes 172.16.0.0/24 e 172.18.0.0/24. Também informa que para chegar à rede 172.17.0.0/24 ocorre via a interface 172.16.0.1 com métrica em 20, e para rede 172.20.0.0/24 via interface 172.18.0.2/24 com custo 20, por

fim para chegar à interface 172.19.0.0/24 tem custo 30 e possui dois caminhos para o nexthop via 172.16.0.1 e via 172.18.0.2

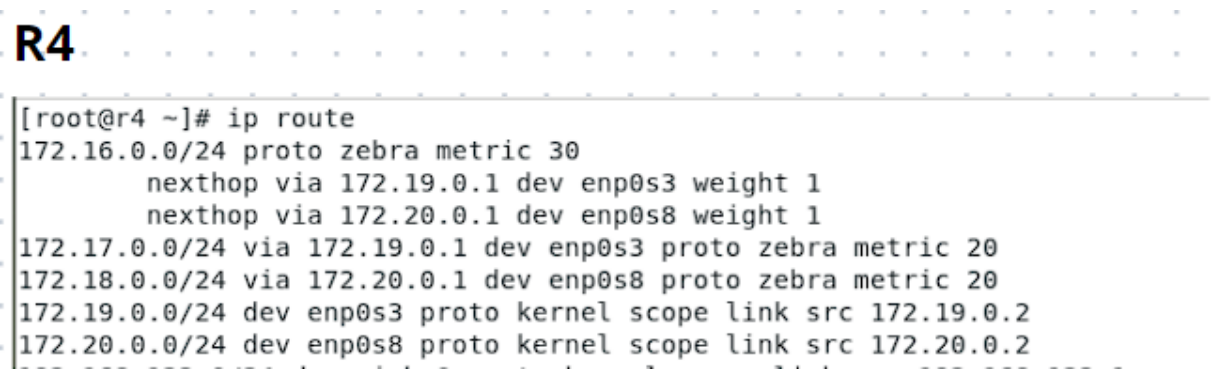
Figura 13: Rotas de R3



Fonte: Próprio autor, 2023.

No caso do R3, ele envia mensagens de estado de link para R1 e R5, essas mensagens informam que ele está conectado às redes 172.17.0.0/24 e 172.19.0.0/24. Também informa que para chegar à rede 172.16.0.0/24 ocorre via a interface 172.17.0.1 com métrica em 20, e para rede 172.20.0.0/24 via interface 172.19.0.2/24 com custo 20, por fim para chegar à interface 172.18.0.0/24 tem custo 30 e possui dois caminhos para o nexthop via 172.17.0.1 e via 172.19.0.2.

Figura 14: Rotas de R4



Fonte: Próprio autor, 2023.

No caso do R4, ele envia mensagens de estado de link para R3 e R5, essas mensagens informam que ele está conectado às redes 172.19.0.0/24 e 172.20.0.0/24. Também informa que para chegar à rede 172.17.0.0/24 ocorre via a interface 172.19.0.1 com métrica em 20, e para rede 172.18.0.0/24 via interface 172.20.0.1/24 com custo 20, por fim para chegar à interface 172.20.0.0/24 tem custo 30 e possui dois caminhos para o nexthop via 172.16.0.2 e via 172.17.0.2

Figura 15: Rotas de R5

Fonte: Próprio autor, 2023.

No caso do R5, ele envia mensagens de estado de link para R2 e R4, essas mensagens informam que ele está conectado às redes 172.18.0.0/24 e 172.20.0.0/24. Tambem informa que para chegar a rede 172.19.0.0/24 ocorre via a interface 172.18.20.2 com métrica em 20, e para rede 172.19.0.0/24 via interface 172.17.0.2/24 com custo 20, por fim para chegar a interface 172.17.0.0/24 tem custo 30 e possui dois caminhos para o nexthop via 172.18.0.1 e via 172.20.0.2.

Também é possível consultar via telnet na porta 2604 as rotas dos vizinhos diretamente conectados de acordo com a imagem abaixo de R5:

Figura 16: Rotas e vizinhos diretamente conectados de R5 via telnet

```
r5# sh ip ospf route
============ OSPF network routing table ============
N    172.16.0.0/24         [20] area: 0.0.0.0
                           via 172.18.0.1, enp0s3
N    172.17.0.0/24         [30] area: 0.0.0.0
                           via 172.18.0.1, enp0s3
                           via 172.20.0.2, enp0s8
N    172.18.0.0/24         [10] area: 0.0.0.0
                           directly attached to enp0s3
N    172.19.0.0/24         [20] area: 0.0.0.0
                           via 172.20.0.2, enp0s8
N    172.20.0.0/24         [10] area: 0.0.0.0
                           directly attached to enp0s8

============ OSPF router routing table =============

============ OSPF external routing table ===========

r5# sh ip os
r5# sh ip ospf nei
r5# sh ip ospf neighbor

    Neighbor ID Pri State           Dead Time Address         Interface            RXmtL RqstL DBsmL
172.18.0.1        1 Full/DR           32.380s 172.18.0.1      enp0s3:172.18.0.2        0     0     0
172.20.0.2        1 Full/DR           39.044s 172.20.0.2      enp0s8:172.20.0.1        0     0     0
```

Fonte: Próprio autor, 2023

.

Quando um roteador recebe uma mensagem de estado de link, ele verifica se a rede especificada na mensagem é desconhecida. Se for, o roteador adiciona a rota à sua tabela de rotas. O roteador também verifica o custo da rota. O custo da rota é um valor numérico que representa a distância entre um roteador atual e a rede especificada. O roteador também usa o custo da rota para determinar o melhor caminho para encaminhar pacotes para a rede especificada.

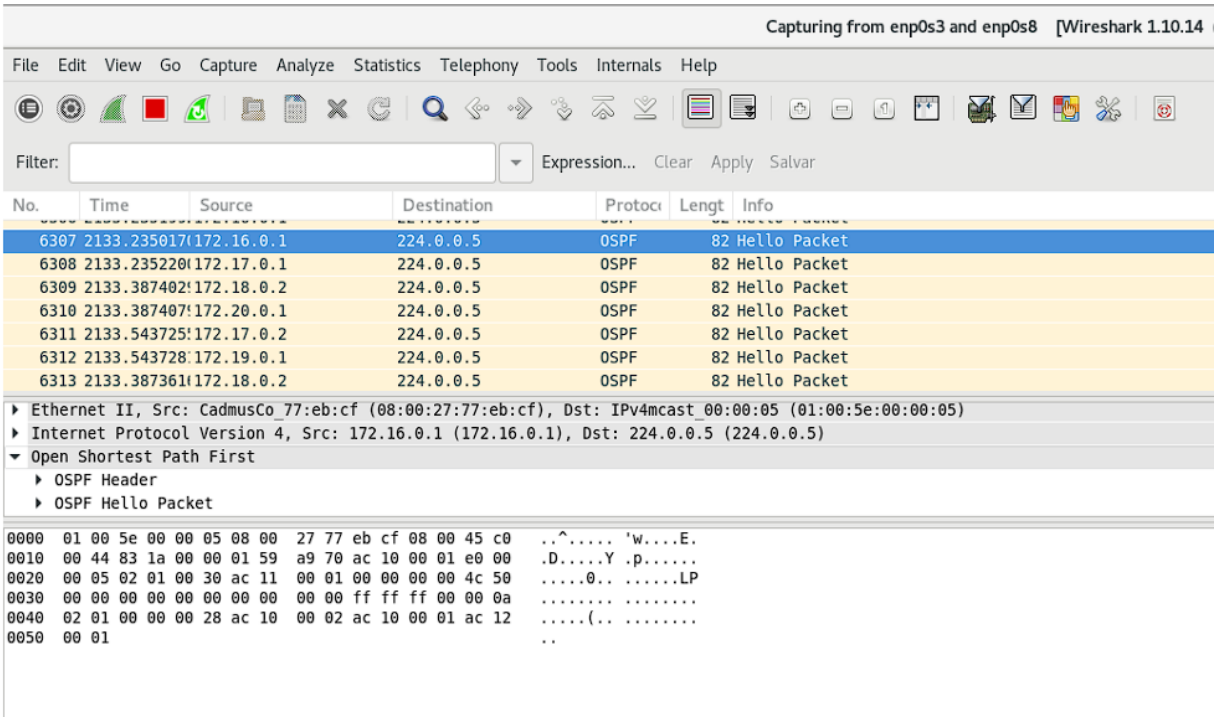
O algoritmo de roteamento de caminho mínimo do OSPF usa o custo da rota para determinar o melhor caminho para encaminhar pacotes.

3.4 Wireshark

Ao iniciar a ferramenta Wireshark selecionamos as duas interfaces que queremos realizar as capturas de pacotes, nesse caso as interfaces enp0s3 e enp0s8 e inicia no botão que parece um play.

A imagem abaixo mostra a troca de pacotes hello do OSPF entre as redes proposta pela topologia:

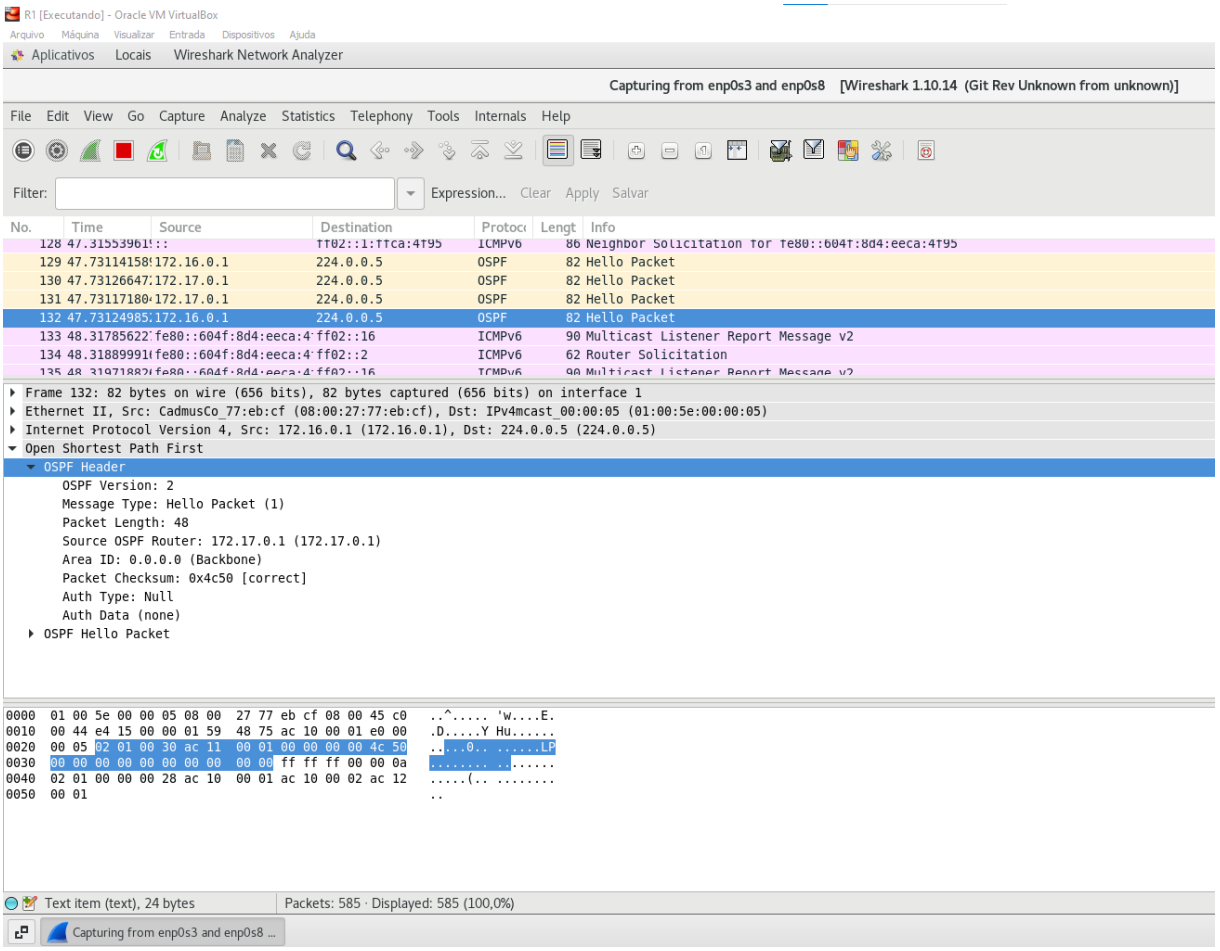
Figura 17: Captura Wireshark dos pacotes OSPF.



Fonte: Próprio autor, 2023.

Abaixo a imagem mostra o pacote hello do OSPF com detalhes e informações de cabeçalho do OSPF Header:

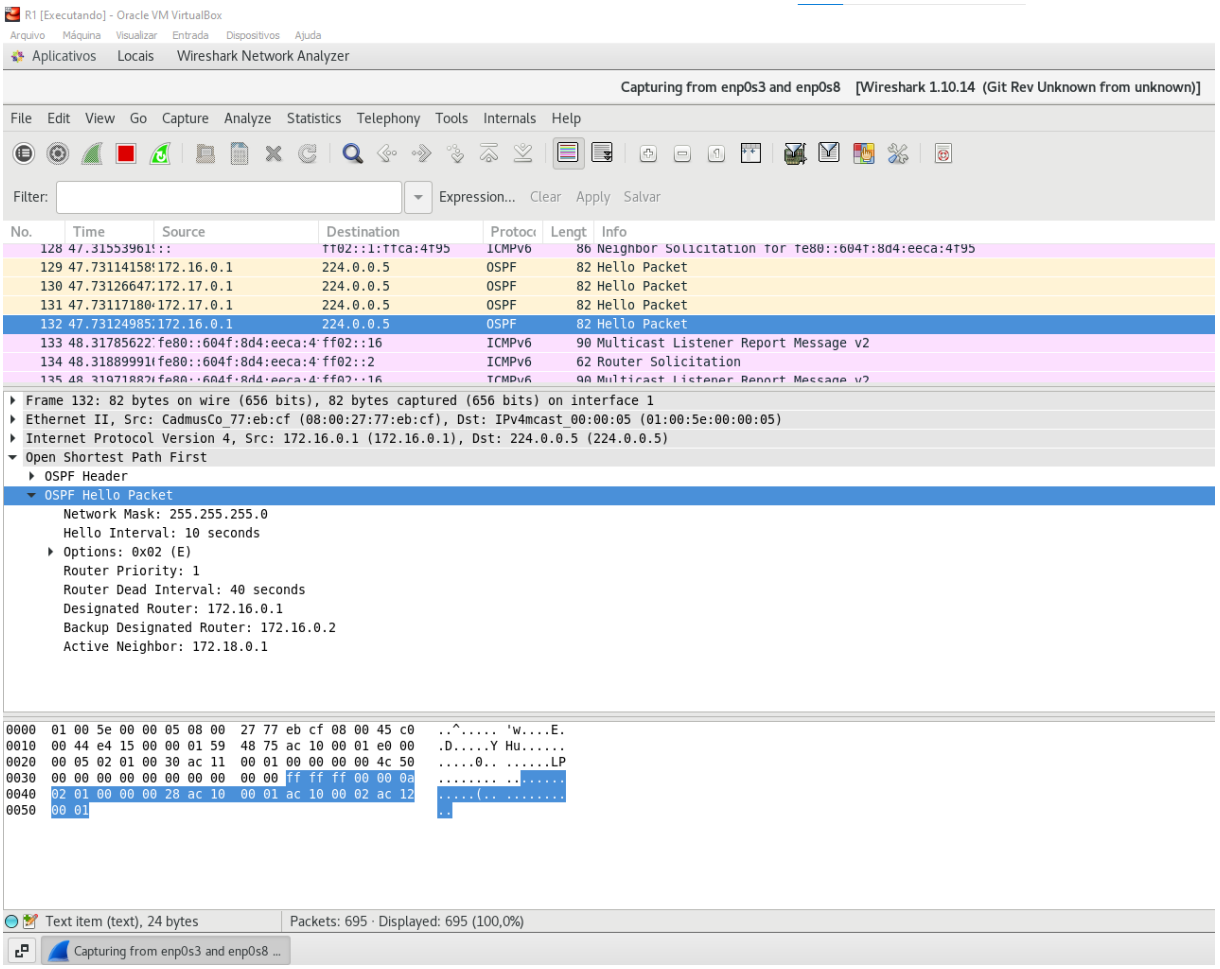
Figura 18: Captura Wireshark do OSPF Header.



Fonte: Próprio autor, 2023.

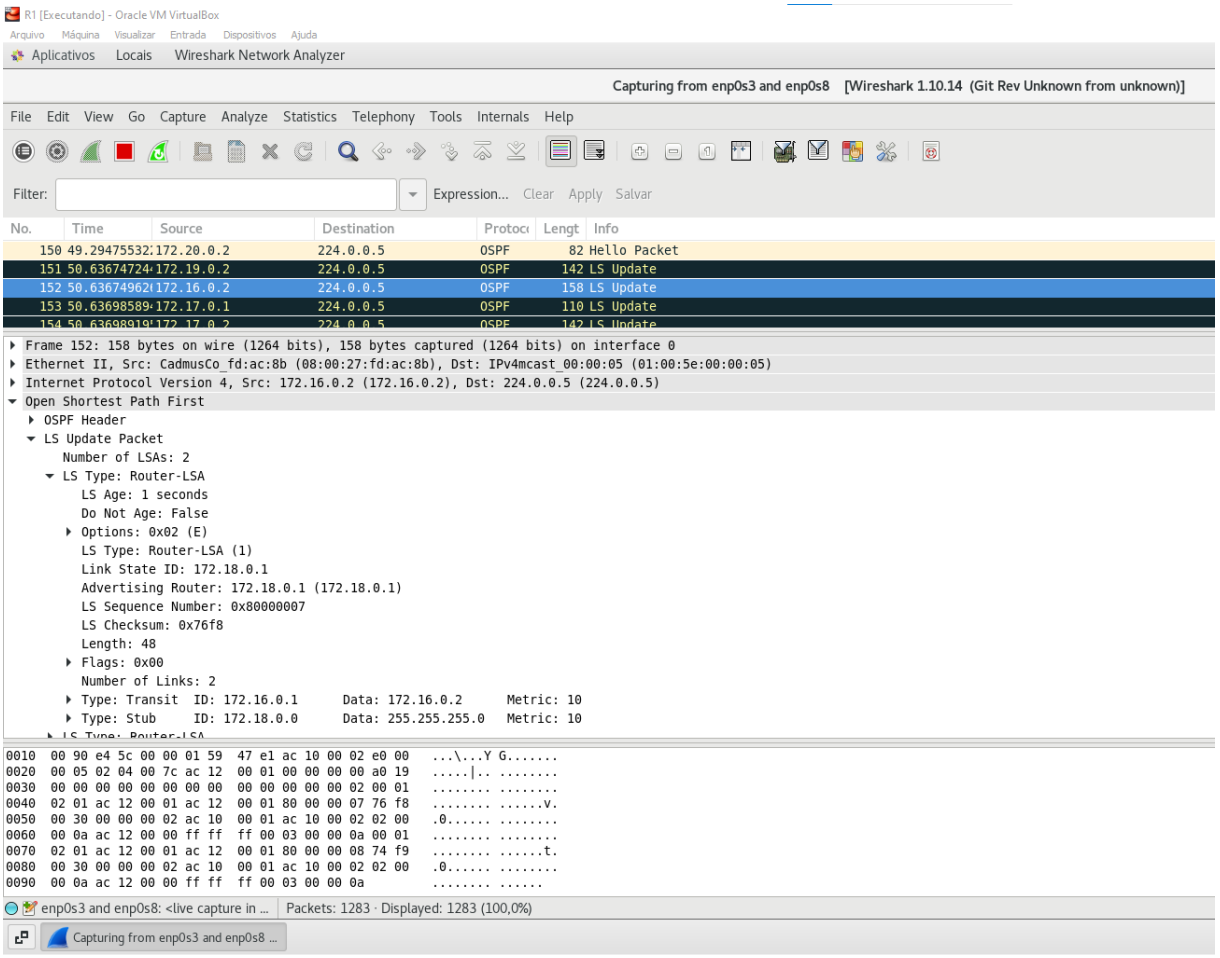
A seguir a imagem abaixo traz informações do pacote OSPF Hello:

Figura 19: Captura Wireshark do OSPF Hello Packet



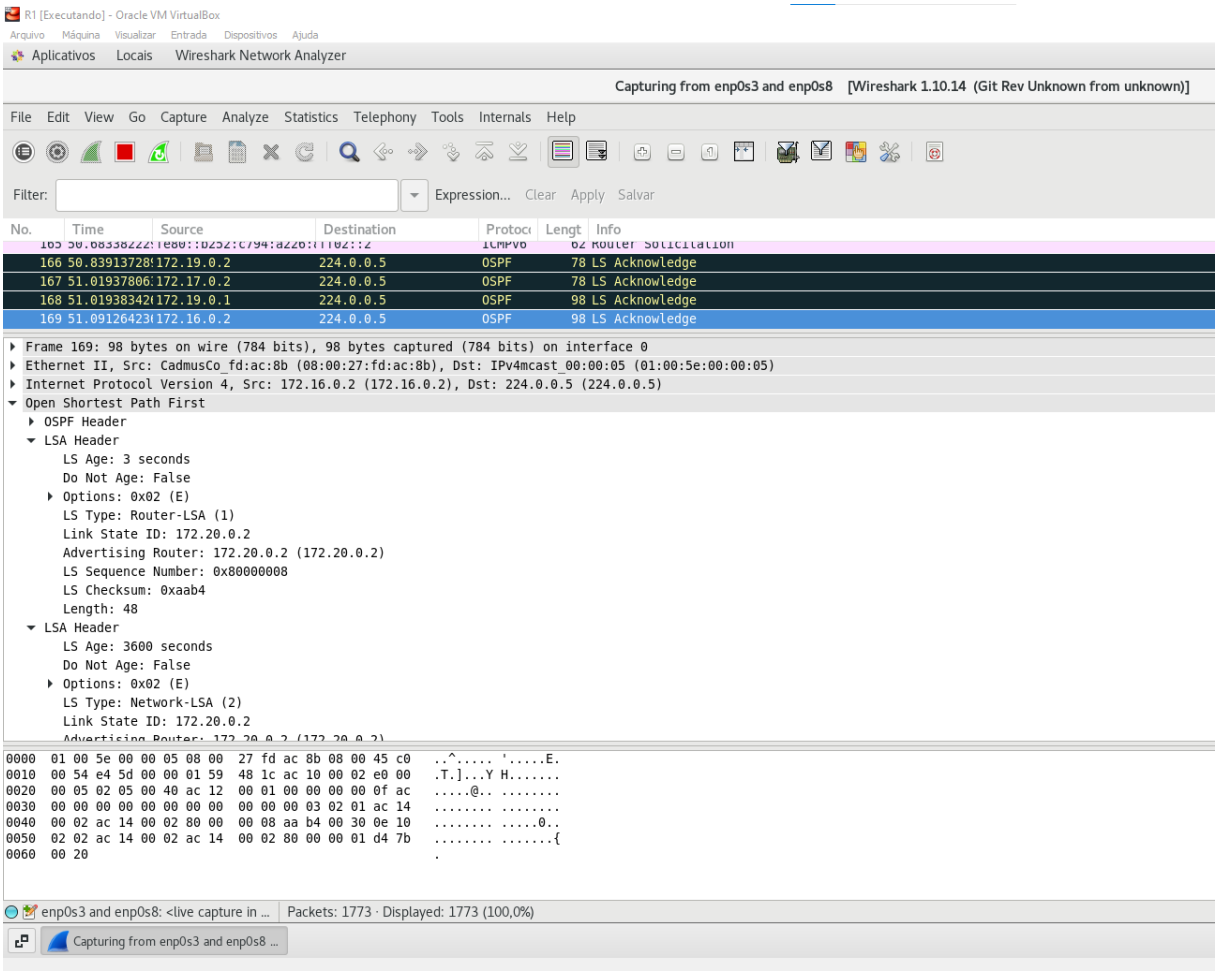
Fonte: Próprio autor, 2023.

Figura 20: Captura Wireshark do pacote LSU.



Fonte: Próprio autor, 2023.

Figura 21: Figura 21: Captura Wireshark dos pacotes LSA



Fonte: Próprio autor, 2023.

Após parar o serviço ospfd de R5, o protocolo entra em ação, ajustando-se de acordo com as configurações dos intervalos de Hello e Dead. Nessa topologia, os pacotes Hello são enviados a cada 10 segundos, e após quatro tentativas sem respostas, o protocolo OSPF considera que o vizinho está inativo e remove o endereço deste vizinho da lista de rotas. Isso ocorre devido a alteração na topologia de rede. O protocolo OSPF usa o LSU para propagar informações de roteamento entre os roteadores. Os pacotes LSU contém informações sobre as rotas que passam por um roteador. Quando um roteador detecta uma alteração na topologia de rede, ele começa a enviar pacotes LSU para todos os seus vizinhos. Os roteadores vizinhos recebem os pacotes LSU e atualizam suas tabelas de roteamento para refletir as alterações na topologia de rede. Para confirmar a entrega dos pacotes LSU, o protocolo OSPF usa o LSAck. Quando um roteador envia um

pacote LSU, ele espera receber um LSAck do destinatário para confirmar que o pacote foi recebido com sucesso. Na hipótese deste nó R5 estar conectado a uma outra área, a mesma iria realizar os mesmos passos mencionados e sua tabela de rota seria atualizada, buscando uma outra alternativa de caminho que não fosse por esse nó inativo. Esse processo é fundamental para garantir que todos os roteadores na rede tenham informações de roteamento atualizadas e estejam cientes das mudanças na topologia de rede.

CONSIDERAÇÕES FINAIS

Neste trabalho, foi apresentado um estudo sobre a implementação do protocolo OSPF em um ambiente virtual. Foram descritos os obstáculos e dificuldades encontrados durante a implementação, bem como as recomendações bem-sucedidas para superá-los. Também foram apresentados os resultados dos experimentos realizados, incluindo as rotas criadas em cada roteador, o algoritmo de roteamento de caminhos mínimos do OSPF e as capturas de pacotes realizadas com o Wireshark.

Com base nos resultados obtidos, pode-se concluir que o protocolo OSPF é um protocolo de roteamento eficaz e confiável. Ele é capaz de aprender rapidamente sobre as redes em uma área e construir uma tabela de rotas que fornece o melhor caminho para cada destino.

O trabalho contribui para o conhecimento sobre o protocolo OSPF, fornecendo uma visão detalhada de sua implementação e funcionamento. Ele também pode ser usado como um guia para quem deseja implementar o protocolo OSPF em um ambiente virtual.

Com base nos resultados obtidos, recomenda-se que os seguintes tópicos sejam considerados ao implementar o protocolo OSPF:

- **Configuração adequada dos roteadores:** Uma configuração adequada dos roteadores é essencial para o funcionamento correto do protocolo OSPF.
- **Monitoramento do protocolo OSPF:** O protocolo OSPF deve ser monitorado para garantir o funcionamento corretamente.
- **Segurança:** O protocolo OSPF pode ser vulnerável a ataques. Portanto medidas de segurança devem ser implementadas para proteger o protocolo.

Alguns tópicos que poderiam ser explorados em trabalhos futuros incluem:

- **Um estudo sobre a implementação do protocolo OSPF em redes móveis.**

- **Um estudo sobre a segurança do protocolo OSPF**
- **Um estudo sobre quais métodos de segurança são possíveis de implementar no OSPF**

Essas sugestões poderiam ajudar a fornecer uma visão mais completa sobre o protocolo OSPF, bem como a identificar áreas de pesquisas futuras para melhorar o protocolo e torná-lo mais seguro e confiável.

REFERÊNCIAS BIBLIOGRÁFICAS

FILIPPETTI, Marco Aurélio. CCNA 6.0. 3. ed. São Paulo: Novatec, 2016.

GIL, Antônio Carlos. Como elaborar projetos de pesquisa. 6. ed. São Paulo: Atlas, 2017.

KUROSE, James F.; ROSS, Keith W. Redes de computadores e a Internet: uma abordagem top-down. 8. ed. São Paulo: Pearson, 2021.

MOY, J. OSPF Version 2. RFC 2328. 1998. Disponível em: https://tools.ietf.org/html/rfc2328. Acesso em: 27 mar. 2023.

STALLINGS, William. Redes e sistemas de comunicação de dados. 10ª ed. São Paulo: Pearson, 2018.

TANENBAUM, Andrew S. Redes de computadores. 5. ed. São Paulo: Pearson, 2020.

REFERÊNCIAS ELETRÔNICAS

Disponível em: https://www.gta.ufrj.br/grad/03_1/redes-industriais/ospf6.html.

MIKROTIK. Disponível em: 13/11/2023

http://www.mikrotik.com/documentation/manual_2.5/Routing/OSPF.html

Acesso em: 13/11/2023

Cisco. Disponível em: https://www.cisco.com/c/pt_br/support/docs/ip/open-shortest-path-first-ospf/7039-1.html

Acesso em:

Linux Quagga. Disponível em: 13/11/2023

http://www.patrick.eti.br/?p=artigos&a=frr

Acesso em: 13/11/2023