Análise Comparativa de Controladores SDN: Uma Avaliação com Base em Desempenho, Escalabilidade e Facilidade de Uso

Yooh Bezerra de Brito

CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

JOÃO PESSOA - PB
2024

Yooh Bezerra de Brito

Análise Comparativa de Controladores SDN
Uma Avaliação com Base em Desempenho, Escalabilidade e Facilidade de Uso

Monografia apresentada ao curso Ciência da Computação do Centro de Informática, da Universidade Federal da Paraíba como requisito para a obtenção do grau de Bacharel.

Orientador: Ed Porto Bezerra

JOÃO PESSOA - PB

2024

Catalogação na publicação
Seção de Catalogação e Classificação

B862a Brito, Yooh Bezerra de.
Análise comparativa de controladores SDN: uma avaliação com base em desempenho, escalabilidade e facilidade de uso / Yooh Bezerra de Brito. - João Pessoa, 2025.
45 f. : il.

Orientação: Ed Porto Bezerra.
TCC (Graduação) - UFPB/CI.

1. Software (SDN). 2. Dispositivos de rede. 3. Controladores SDN. 4. Desempenho de redes. 5. Gestão de redes. I. Bezerra, Ed Porto. II. Título.

UFPB/CI CDU 004.4

Elaborado por ROGERIO FERREIRA MARQUES - CRB-15/690

CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

Trabalho de Conclusão de Curso de Ciência da Computação intitulado **Análise Comparativa de Controladores SDN Uma Avaliação com Base em Desempenho, Escalabilidade e Facilidade de Uso** de autoria de Yooh Bezerra Brito, aprovada pela banca examinadora constituída pelos seguintes professores:

Professora Dra. Giorgia de Oliveira Mattos

CI - UFPB

Prof. Dr. Ewerton Monteiro Salvador

CI - UFPB

Prof. Dr. Ed Porto Bezerra

CI - UFPB

Coordenadora do Departamento Centro de Informática

Giorgia De Oliveira Mattos

CI/UFPB

João Pessoa, 9 de novembro de 2024

Centro de Informática, Universidade Federal da Paraíba
Rua dos Escoteiros, Mangabeira VII, João Pessoa, Paraíba, Brasil CEP:5058-600
Fone: +55 (83) 3216 7093 / Fax: +55 (83) 3216 7117

RESUMO

Este Trabalho de Conclusão de Curso (TCC) objetiva compreender os impactos da escolha do controlador SDN nas Redes Definidas por Software. O estudo é fundamentado no uso dessas redes, uma abordagem inovadora na gestão de redes, destacando a importância da escolha de um controlador SDN para uso numa rede. Os controladores de SDN envolvidos neste TCC são os seguintes: OpenDaylight, Floodlight e Ryu. A metodologia possui três etapas: a configuração de um ambiente de teste com Mininet, a avaliação de desempenho, escalabilidade e facilidade de uso deste ambiente e, por fim, a análise comparativa, com visualização de dados, dos resultados desta avaliação. O resultado deste TCC contribuirá para a compreensão prática e teórica da escolha de controladores SDN, fornecendo assim insights para que profissionais e pesquisadores possam ter mais assertividade na escolha de controladores de SDN. Este TCC traz resultados relevantes à área por causa da demanda crescente por ambientes de rede mais inteligentes e ágeis.

Palavras-chave: Redes Definidas por Software (SDN); Controladores SDN; Desempenho de Redes.

ABSTRACT

The proposed undergraduate thesis, titled "Comparative Analysis of SDN Controllers: An Assessment Based on Performance, Scalability, and Ease of Use," aims to explore and compare three prominent SDN controllers: OpenDaylight, Floodlight, and Ryu. Grounded in Software Defined Networking (SDN), an innovative approach to network management, the study underscores the critical importance of selecting the appropriate SDN controller. The methodology involves configuring a test environment with Mininet, evaluating performance, scalability, and ease of use, and conducting a comparative analysis of the results. The research contributes to both practical and theoretical understanding of SDN controller selection, providing valuable insights for professionals and researchers. The proposal highlights the relevance of the research in the current SDN landscape, outlining objectives, methodology, expected results, and ethical considerations, solidifying its significance in the field.

Key-words: Software-Defined Networking (SDN); SDN Controllers; Network Performance.

LISTA DE GRÁFICOS

LISTA DE FIGURAS

SUMÁRIO

1. INTRODUÇÃO

1.1 Redes Definidas por Software (SDN) no Contexto Atual

Redes Definidas por Software (SDN) visam otimizar redes, logo, representam uma mudança paradigmática nas arquiteturas tradicionais de rede. Conforme explica Kreutz et al. (2014), a SDN permite um gerenciamento mais flexível e programável das redes, separando o plano de controle do plano de dados. Usando SDN, o plano de controle é desacoplado do plano de dados, permitindo o controle centralizado e a orquestração dos recursos de uma rede em questão. Essa mudança de abordagem em relação às configurações de rede convencionais permite uma grande flexibilidade e adaptabilidade.

A importância da SDN reside em sua capacidade de aprimorar a escalabilidade da rede, simplificar a configuração e otimizar a alocação de recursos. Em sistemas convencionais, o ajuste a novas demandas de forma rápida e eficiente é uma tarefa complexa, o que pode gerar problemas de desempenho (Kim & Feamster, 2013). Redes tradicionais frequentemente enfrentam desafios na adaptação a demandas em constante evolução, resultando em ineficiências na utilização de recursos. A SDN aborda esses problemas ao fornecer um controlador centralizado capaz de alocar e gerenciar dinamicamente os recursos com base em requisitos em tempo real.

Além disso, a SDN facilita a implementação de serviços e aplicações de rede por meio de interfaces programáveis. Essa programabilidade capacita os administradores de rede a adaptarem o comportamento da rede às necessidades específicas, promovendo a inovação e a rápida implementação de novos serviços. À medida que as organizações adotam cada vez mais a transformação digital, a SDN surge como uma tecnologia fundamental para atender às demandas em constante evolução das redes modernas. Compreender a importância contextual da SDN prepara o terreno para explorar as nuances dos controladores SDN, ponto central desta pesquisa.

1.2 Significado da Escolha do Controlador SDN

A escolha do controlador possui um papel crucial na eficácia e desempenho de uma rede definida por software. O controlador funciona como o cérebro da SDN, sendo responsável para executar decisões centralizadas e consegue coordenar as operações na rede toda. Segundo Kreutz et al. (2014), essa centralização proporciona um maior controle sobre os fluxos de dados e permite uma orquestração mais eficiente dos recursos, o que é essencial para garantir uma gestão otimizada da rede e adaptabilidade às demandas específicas de cada organização.

O controlador SDN influencia diretamente a agilidade operacional da rede, determinando como os dispositivos de rede respondem às mudanças nas condições de tráfego e demandas de serviço. Sua capacidade de processamento, eficiência na comunicação com switches e suporte a protocolos específicos impactam diretamente na latência, na largura de banda e na capacidade de adaptação da rede. Ao escolher um controlador afeta a facilidade de implementação de novos serviços e aplicativos. A escolha certa pode simplificar a integração de soluções personalizadas e acelerar o desenvolvimento de novas funcionalidades.

A escolha do controlador leva em consideração a parte técnica e a de planejamento. Como observam Nunes et al. (2014), ela define a capacidade da rede de ser operacionalmente eficiente e adaptável em um ambiente de rápida evolução tecnológica. Assim, ao moldar a capacidade de resposta e o potencial de crescimento da rede, a escolha do controlador torna-se essencial para maximizar os benefícios oferecidos pela arquitetura SDN.

1.3 Motivação para a Pesquisa

A motivação para esta pesquisa reside na necessidade de compreender profundamente os impactos da escolha do controlador SDN nas Redes Definidas por Software (SDN). À medida que as organizações buscam redes mais flexíveis, eficientes e adaptáveis, a seleção do controlador SDN adequado torna-se um fator crítico para alcançar esses objetivos. A dinâmica inerente à SDN, com a separação do plano de controle e do plano de dados, oferece uma oportunidade para otimizar o gerenciamento de redes de forma centralizada. Contudo, a variedade de controladores disponíveis introduz complexidades na escolha, uma vez que diferentes

tipos podem impactar significativamente o desempenho, escalabilidade e facilidade de uso da rede.

A motivação também se fundamenta na crescente demanda por ambientes de rede mais inteligentes e ágeis. Com a rápida evolução das tecnologias e das necessidades organizacionais, compreender como diferentes controladores SDN se comportam em cenários práticos torna-se crucial para orientar escolhas informadas. Ao abordar esse tópico, a pesquisa visa fornecer insights para profissionais, pesquisadores e tomadores de decisão, contribuindo para a eficiência operacional das redes no contexto das SDN.

1.4 Objetivo Principal e Objetivos Específicos

1.4.1 Objetivo Principal:

Explorar e comparar o impacto de diferentes controladores, como OpenDaylight, Floodlight e Ryu, nas Redes Definidas por Software (SDN). Escolhidos pelo motivo para acrescentar o material para pesquisas, pesquisas anteriores envolvem as de Sharma et al. (2023), Li et al. (2020), Chouhan et al. (2019) e Mamushiane et al. (2018). A ideia é entender o desempenho, a escalabilidade e a facilidade de uso de cada um, trazendo informações práticas que ajudem na escolha de controladores SDN para uso em uma rede, comparando resultados e propondo recomendações práticas.

1.4.2 Objetivos Específicos:

Os objetivos específicos deste TCC são os seguintes:

1. Avaliar o Desempenho de controladores: observar como cada controlador se comporta em termos de tempo de resposta, velocidade de dados e uso de recursos em diferentes situações de carga e tráfego, oferecendo uma visão detalhada do que cada controlador é capaz de entregar.
2. Analisar a Escalabilidade de controladores: investigar como cada controlador lida com cenários de diferentes tamanhos, com mais ou menos dispositivos conectados, para identificar até onde eles conseguem crescer sem problemas.

3. Avaliar a Facilidade de Uso de controladores: verificar o quão fácil é configurar e operar cada controlador, considerando a interface, a documentação e o processo de instalação e uso, para ter uma visão completa da usabilidade de cada um.

4. Comparar Resultados: fazer uma comparação dos resultados para destacar padrões, diferenças importantes e pontos em que cada controlador se destaca ou apresenta dificuldades.

5. Propor Recomendações Práticas: sugerir recomendações que ajudem profissionais e organizações a escolher o controlador SDN mais adequado para suas necessidades.

1.5 Justificativa do Estudo

A justificativa para essa pesquisa passa por vários pontos: a escolha do controlador certo afeta diretamente o funcionamento da rede. A relevância deste estudo fundamenta-se na crescente adoção de Redes Definidas por Software (SDN) pelas empresas e na necessidade crítica de compreender o impacto da escolha do controlador SDN. A justificativa para esta pesquisa reside em diversos aspectos como o impacto na eficiência operacional, adaptação de rede, tomadas de decisão, contribuição para conhecimento em SDN e incentivo ao desenvolvimento tecnológico.

A pesquisa adicionará ao conhecimento sobre SDN, com uma análise comparativa de controladores amplamente usados, entender as particularidades dos controladores pode incentivar a inovação e o avanço tecnológico. Os resultados deste estudo podem não só ajudar a melhorar os controladores existentes como também inspirar a criação de novas soluções. Em suma, a justificativa deste estudo reside na sua capacidade de fornecer informações para aprimorar a eficiência, adaptabilidade e escolha informada de controladores SDN, contribuindo para o avanço das Redes Definidas por Software.

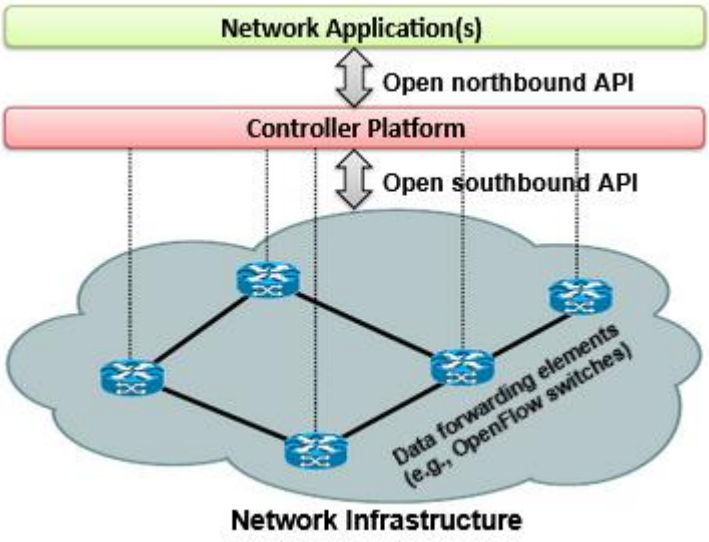
2. FUNDAMENTAÇÃO TEÓRICA

2.1 Redes Definidas por Software (SDN)

As Redes Definidas por Software (SDN) representam uma evolução na engenharia de redes, quebrando paradigmas e permitindo que as redes sejam administradas de maneira mais fluida e intuitiva. Diferente dos sistemas tradicionais, em que o controle e o encaminhamento de dados são mesclados nos próprios dispositivos físicos, a SDN separa essas funções (Kreutz et al., 2014). Nessa nova abordagem, o controle da rede fica centralizado em um controlador SDN, enquanto switches e roteadores cuidam apenas do trânsito dos dados. Esse modelo traz uma leveza estrutural e uma flexibilidade que antes não eram possíveis, facilitando a aplicação de políticas e ajustes de forma centralizada (Nunes et al., 2014).

Para os administradores de rede, essa mudança é importante. A SDN permite configurar e otimizar a rede com uma agilidade que elimina a necessidade de ajustes manuais em cada dispositivo. Em cenários complexos, como data centers ou grandes redes de telecomunicações, essa possibilidade representa um avanço considerável, simplificando a gestão de sistemas que estão sempre em transformação (Feamster, Rexford & Zegura, 2014). A SDN também abre portas para o desenvolvimento de novos serviços de rede. Com Application Programming Interface (API) que permitem programação e personalização da rede conforme as necessidades específicas de cada aplicação, a SDN oferece um controle direto sobre aspectos como segurança, qualidade de serviço (QoS) e balanceamento de carga. As APIs Northbound em SDN são interfaces que permitem a comunicação entre o controlador SDN e as aplicações ou serviços de rede de mais alto nível, facilitando a gestão e a automação de políticas de rede. As APIs Southbound, por outro lado, conectam o controlador SDN aos dispositivos de rede, como switches e roteadores, permitindo o controle direto do fluxo de tráfego e a configuração da infraestrutura de rede. Na figura 1 temos uma exemplificação da arquitetura SDN.

Figura 1: Arquitetura SDN vista de uma forma simplificada

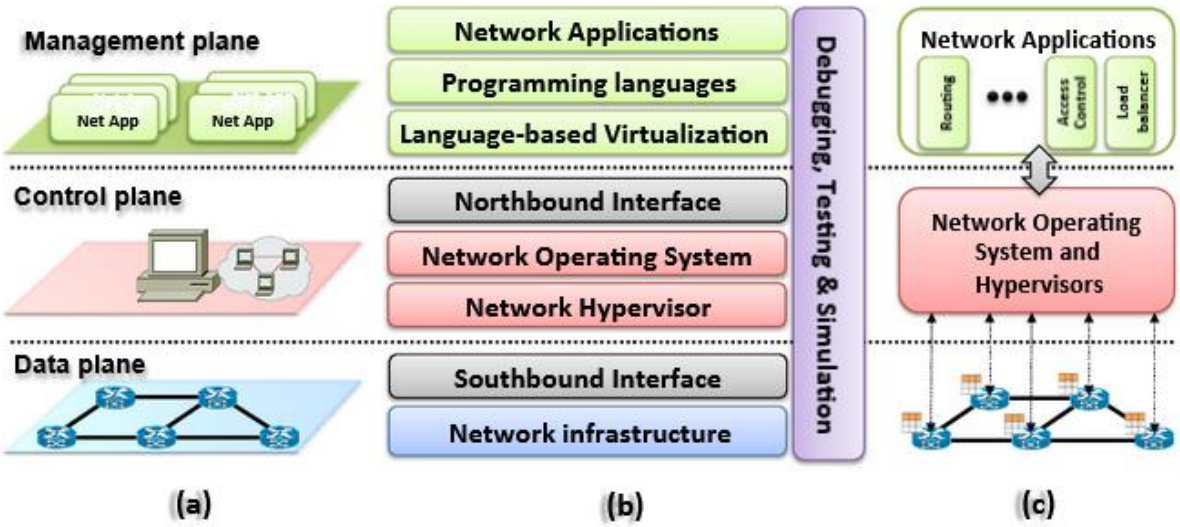


Fonte: (Kreutz et al., 2014)

2.2 Arquitetura SDN

A arquitetura das Redes Definidas por Software (SDN) se estrutura em três camadas essenciais: o plano de dados (data plane), o plano de controle (control plane) e o plano de aplicação (application plane). A separação das funções em cada plano proporciona a flexibilidade e a programabilidade que marcam a estrutura das SDNs (Kreutz et al., 2014). A figura 2 mostra uma divisão de uma arquitetura SDN.

Figura 2: SDN definido em (a) planos, (b) camadas, (c) arquitetura do sistema



Fonte: (Kreutz et al., 2014)

1. Plano de Dados (Data Plane): O plano de dados é o núcleo prático da rede, encarregado do tráfego dos pacotes. Aqui estão switches e roteadores, sejam físicos ou virtuais, que, em vez de "pensarem" sozinhos, recebem as direções do controlador SDN para encaminhar o tráfego de acordo com as regras definidas. Essa "simplicidade" de funções nos dispositivos torna o plano de dados direto e eficiente, sem margem para escolhas autônomas, permitindo que a rede funcione com agilidade e precisão (Nunes et al., 2014).

2. Plano de Controle (Control Plane): No plano de controle, encontramos o centro da SDN. Este plano é onde as decisões sobre o tráfego e as políticas de rede são tomadas. O controlador SDN, que habita esta camada, age como a mente por trás da rede, se comunicando com o plano de dados por meio de protocolos, como o OpenFlow (McKeown et al., 2008).

3. Plano de Aplicação (Application Plane): Já o plano de aplicação é o meio onde os aplicativos e serviços de rede se comunicam com a SDN. Aplicações como sistemas de segurança, ferramentas de balanceamento de carga, gestão de tráfego e qualidade de serviço (QoS) são executadas sobre a infraestrutura de controle, provendo as instruções que o controlador SDN deverá implementar na rede. A interação entre esses aplicativos e o controlador é facilitada por APIs, que permitem o desenvolvimento de soluções sob medida, ajustadas às demandas específicas de cada ambiente (Kim & Feamster, 2013).

Comunicação entre os Planos: A colaboração entre esses três planos é o que confere a funcionalidade harmoniosa e flexível às SDNs. O controlador SDN traduz as orientações dos aplicativos do plano de aplicação em regras práticas que são aplicadas no plano de dados. Esse procedimento permite que a rede responda de forma rápida e adaptável às alterações nas políticas, mantendo-se sempre sob o monitoramento e controle centralizado do controlador.

2.3 Papel do Controlador SDN

O controlador SDN desempenha um papel central na arquitetura das Redes Definidas por Software, atuando como a entidade responsável por gerenciar e orquestrar as operações da rede. Conforme descrevem Kreutz et al. (2014), o controlador é o responsável pela divisão entre o plano de controle e o plano de dados, permitindo que as decisões de gerenciamento estejam centralizadas em um único ponto.

Um dos principais papéis do controlador é tomar decisões sobre o roteamento e o encaminhamento de pacotes. Ao contrário das redes tradicionais, onde essas decisões são distribuídas entre os dispositivos de rede, em uma SDN o controlador centraliza essa função. Isso significa que ele tem uma visão global da rede e pode aplicar políticas de encaminhamento de maneira mais eficiente, com base em informações atualizadas sobre o estado da rede. Além disso, conforme discute Tootoochian et al. (2012), o controlador SDN atua como elo de comunicação entre o plano de controle e o plano de dados, utilizando protocolos como o OpenFlow para instruir switches e roteadores sobre como direcionar o tráfego.

Ele atua como intermediário entre o plano de aplicação e o plano de dados, recebendo instruções dos aplicativos de rede e traduzindo essas políticas em regras operacionais que são implementadas pelos dispositivos de rede. Isso permite que serviços de rede, como sistemas de segurança, balanceamento de carga e qualidade de serviço (QoS), sejam aplicados de forma eficiente e centralizada (Kreutz et al., 2014).

Além de tomar decisões em tempo real sobre o tráfego, o controlador coleta e armazena dados sobre o desempenho da rede. Esses dados são fundamentais para o monitoramento e otimização da rede, fornecendo informações que podem ser

utilizadas para ajustar políticas de controle, identificar problemas e garantir o desempenho ideal da infraestrutura.

2.4 Importância da Escolha do Controlador SDN

A escolha do controlador SDN é uma decisão crítica que influencia diretamente o desempenho e a eficiência de uma rede definida por software. Cada controlador SDN oferece características específicas, e a seleção adequada pode ser determinante para alcançar os objetivos da rede, seja em termos de escalabilidade, desempenho, ou capacidade de integração com diferentes aplicações e dispositivos.

O controlador atua como o cérebro da rede SDN, responsável por gerenciar todo o tráfego e garantir que as políticas de rede sejam aplicadas corretamente. Por isso, sua capacidade de lidar com grandes volumes de dados e de se adaptar a mudanças na infraestrutura é fundamental. Um controlador que não consiga processar adequadamente as informações ou que tenha limitações na comunicação com o plano de dados pode comprometer a eficiência da rede e, em alguns casos, provocar falhas ou gargalos (Kreutz et al., 2014).

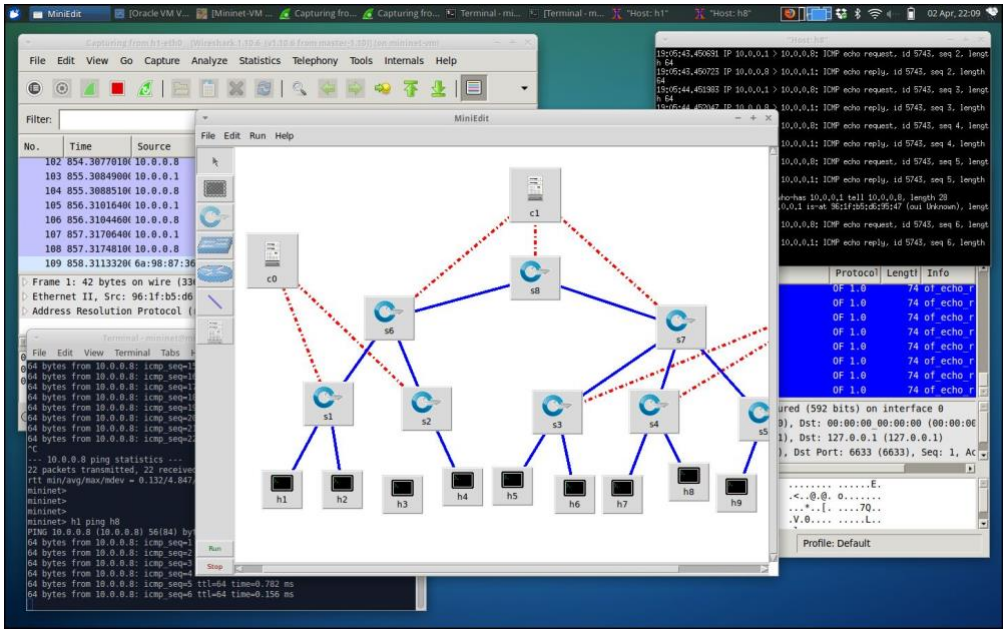
Outro aspecto crucial é a interoperabilidade. Segundo Nunes et al. (2014), os controladores diferem quanto à compatibilidade com protocolos e dispositivos, o que afeta diretamente a flexibilidade e a capacidade de expansão da rede. Em redes que precisam integrar tecnologias variadas ou evoluir de forma contínua, a compatibilidade com múltiplos padrões é essencial para suportar diferentes dispositivos e protocolos de forma harmoniosa.

A facilidade de gerenciamento e a curva de aprendizado também variam entre os controladores. Conforme destaca Marsico et al. (2017), alguns controladores oferecem configurações mais intuitivas e ferramentas de automação mais avançadas, o que facilita a operação e a manutenção da rede, reduzindo o tempo e os recursos dedicados a essas atividades. Em cenários onde a eficiência operacional é prioridade, essas facilidades representam um diferencial importante para garantir a sustentabilidade do gerenciamento de redes SDN.

2.5 Mininet, OpenDaylight, Floodlight e Ryu: Visão Geral

Mininet[1]: É uma ferramenta essencial para a emulação de redes, especialmente útil para desenvolvedores e pesquisadores em Redes Definidas por Software (SDN). Ela permite a criação de uma rede virtual completa em um único computador, simulando hosts, switches, roteadores e conexões de rede, que podem ser gerenciados e testados como se fossem uma rede real. Isso proporciona uma plataforma valiosa para experimentação e teste de configurações de rede, protocolos e aplicações em um ambiente controlado. Mininet suporta uma variedade de comandos que facilitam a modificação dinâmica da rede durante a execução, permitindo aos usuários observar o comportamento da rede sob diferentes condições quase instantaneamente. A figura 3 mostra a interface do Mininet.

Figura 3: Interface Mininet



Fonte: (Brian Linkletter et al. 2015).

OpenDaylight[2] é uma plataforma de código aberto feita em Java e modular desenvolvida pela Linux Foundation, conhecida por sua flexibilidade e arquitetura expansível. Ele oferece suporte a uma ampla gama de protocolos, incluindo

[1] **Mininet: An Instant Virtual Network on Your Laptop (or Other PC) - Mininet**. Disponível em: https://mininet.org. Acesso em: 29 nov. 2024.

[2] Empresa “fundada em 2013, a OpenDaylight faz parte da LF Networking (LFN), uma entidade que integra a governança de projetos participantes, a fim de melhorar a excelência operacional, simplificar o engajamento dos membros e aumentar a colaboração entre projetos de redes de código aberto e órgãos de padrões. Cada projeto técnico mantém sua independência técnica e roteiros de projetos”. OpenDaylight. **The Linux Foundation Projects,** 2024. Disponível em: https://www.opendaylight.org/. Acesso em: 29 nov. 2024.

OpenFlow, BGP e PCEP, tornando-se adequado para redes heterogêneas e complexas. Sua modularidade permite que os usuários customizem a plataforma de acordo com as necessidades específicas da rede, adicionando ou removendo funcionalidades conforme necessário. O OpenDaylight tem forte integração com grandes infraestruturas, sendo amplamente utilizado em redes de data centers e por grandes provedores de serviços.

Floodlight[3] é um controlador SDN leve e de código aberto, baseado em Java, que foi projetado para ser simples de usar e de fácil integração com outros sistemas. Sua simplicidade o torna uma escolha popular para pesquisadores e desenvolvedores que buscam implementar SDN sem a complexidade de plataformas mais robustas. Floodlight oferece suporte ao protocolo OpenFlow e, embora tenha menos recursos nativos que o OpenDaylight, é uma opção eficiente para redes menores ou experimentais, onde o controle rápido e direto é mais valorizado do que a extensibilidade.

Ryu[4], também de código aberto, é amplamente utilizado em ambientes que exigem alta customização e controle detalhado. Escrito em Python, o controlador é especialmente atraente para desenvolvedores devido à facilidade de integração com scripts e aplicativos personalizados. Ryu oferece suporte nativo ao OpenFlow e a outros protocolos, mas sua força reside na capacidade de ser rapidamente adaptado para soluções específicas. Com uma comunidade ativa e documentação acessível, Ryu é frequentemente escolhido em ambientes acadêmicos e de pesquisa, onde a flexibilidade e a capacidade de programação são essenciais.

2.6 Critérios de Avaliação

Para a análise comparativa dos controladores OpenDaylight, Floodlight e Ryu, serão utilizados critérios de avaliação que visam medir o desempenho, escalabilidade e usabilidade de cada controlador no contexto de Redes Definidas por Software (SDN). Esses critérios são fundamentais para garantir uma análise abrangente e

[3] Floodlight Documentation. **Floodlight Controller - Confluence,** 2003-2020. Disponível em: https://floodlight.atlassian.net/wiki/spaces/floodlightcontroller/overview. Acesso em: 29 nov. 2024.
[4] Ryu SDN framework. **Ryu-sdn.org.** [Online]. Disponível em: https://ryu-sdn.org/. Acesso em: Acesso em: 29 nov. 2024.

objetiva, permitindo uma compreensão clara das capacidades e limitações de cada solução.

1. Desempenho: Este critério envolve a avaliação da capacidade de cada controlador em lidar com o tráfego de rede em diferentes condições. Serão analisados parâmetros como latência, throughput e perda de pacotes. O desempenho é um aspecto crucial em redes SDN, pois afeta diretamente a qualidade do serviço e a experiência dos usuários finais. Controladores que conseguem manter baixa latência e altos níveis de throughput são preferíveis para ambientes com alta demanda de tráfego.

2. Escalabilidade: A escalabilidade refere-se à capacidade do controlador de gerenciar um número crescente de dispositivos e fluxos de dados sem degradação significativa de desempenho. Esse critério será avaliado através de testes com diferentes tamanhos de topologias, simulando redes de pequena, média e grande escala. Controladores que mantêm um desempenho consistente à medida que o número de dispositivos, ou nós, aumenta são mais adequados para ambientes que necessitam de crescimento ou expansão contínuos.

3. Facilidade de Uso: Este critério avalia a simplicidade de configuração, operação e gerenciamento de cada controlador. Envolve a análise da interface de usuário, documentação disponível, e as ferramentas de automação e monitoramento que cada plataforma oferece. Controladores com interfaces mais intuitivas e maior facilidade de integração com outros sistemas podem reduzir a complexidade operacional e o tempo necessário para implementar e gerenciar redes SDN.

4. Interoperabilidade: A interoperabilidade será analisada para medir a capacidade de cada controlador de funcionar com diferentes dispositivos, testando com várias. Este critério é importante para garantir que o controlador possa ser integrado a uma variedade de equipamentos e infraestruturas de rede já existentes.

5. Documentação e Extensibilidade: Avalia a documentação disponível e extensibilidade dos controladores será avaliada com base nas ferramentas disponíveis, na facilidade de desenvolvimento de novos módulos ou serviços, e na flexibilidade da arquitetura de cada controlador. Controladores que permitem maior customização podem ser mais adequados para redes que exigem soluções específicas e que precisam de alta adaptabilidade, como o teste com vários hosts.

6. Consumo de Recursos: Por fim, o consumo de recursos, como CPU e memória, será avaliado para verificar a eficiência de cada controlador. Em redes de

grande escala, controladores que consomem muitos recursos podem exigir infraestrutura mais robusta, o que impacta os custos operacionais. Controladores que mantêm bom desempenho com baixo consumo de recursos são preferidos, pois contribuem para uma maior eficiência energética e operacional.

3. Experimentos realizados com os controladores SDN OpenDaylight, Floodlight e Ryu

A metodologia empregada neste TCC possui cinco etapas: a configuração do ambiente de teste, o desenvolvimento de cenários de teste, a definição dos critérios de avaliação, a execução dos testes e, por fim, a coleta, análise e visualização de dados. Cada etapa será detalhada nas seguintes subseções.

3.1 Configuração do Ambiente de Teste

A configuração do ambiente de teste é uma etapa fundamental para garantir que os experimentos realizados com os controladores SDN OpenDaylight, Floodlight e Ryu sejam conduzidos de forma padronizada e replicável. Para esta pesquisa, utilizou-se o Mininet como ferramenta de simulação, permitindo a criação de topologias de rede personalizadas e a interação com os controladores de maneira controlada.

1. Ferramentas Utilizadas

Mininet: Ferramenta de emulação de redes SDN que permite criar e testar diferentes topologias de rede. O Mininet foi escolhido devido à sua capacidade de simular cenários realistas com baixo custo computacional e por sua compatibilidade com os controladores OpenDaylight, Floodlight e Ryu. Apenas os controladores Ryu e floodlight foram utilizados, não foi possível fazer a instalação do l2switch para utilizar o Opendaylight.

2. Infraestrutura do Teste

Os testes foram realizados utilizando uma máquina virtuai rodando em um servidor local, com as seguintes especificações:

Servidor Local:

- Processador: Ryzen 5 5500
- Memória RAM: 16 GB DDR4
- Sistema Operacional: Ubuntu 20.04 LTS

- Ferramentas Instaladas: Mininet, OpenFlow

3. Instalação do Mininet

O Mininet foi instalado utilizando o seguinte procedimento:

```
$ sudo apt-get update
$ sudo apt-get install mininet
```

Configuração básica: Após a instalação, foram realizadas configurações básicas para permitir a comunicação com os controladores. Foi configurada uma topologia em árvore para simular uma rede com um certo número de switches e hosts.

4. Topologia de Rede Utilizada

Para garantir a comparabilidade entre os testes dos diferentes controladores, a mesma topologia de rede foi usada em todos os experimentos. A topologia escolhida foi uma topologia em árvore como base com 3 switches e 4 hosts, conectando todos os switches ao respectivo controlador.

A topologia de profundidade 2 e 2 filhos foi configurada no Mininet da seguinte forma:

```
$  sudo mn --topo=tree, depth=2,fanout =2
--mac --controller remote --switch ovsk
```

Para redes de média escala nessas configurações, foi escolhida uma árvore de profundidade 3 e 3 filhos.

```
$  sudo mn --topo=tree, depth=3,fanout =3
--mac --controller remote --switch ovsk
```

Para redes de grande escala com as configurações existentes de servidor, foi escolhida uma árvore de profundidade 4 e 4 filhos.

$ sudo mn --topo=tree, depth=4,fanout =4
--mac --controller remote --switch ovsk

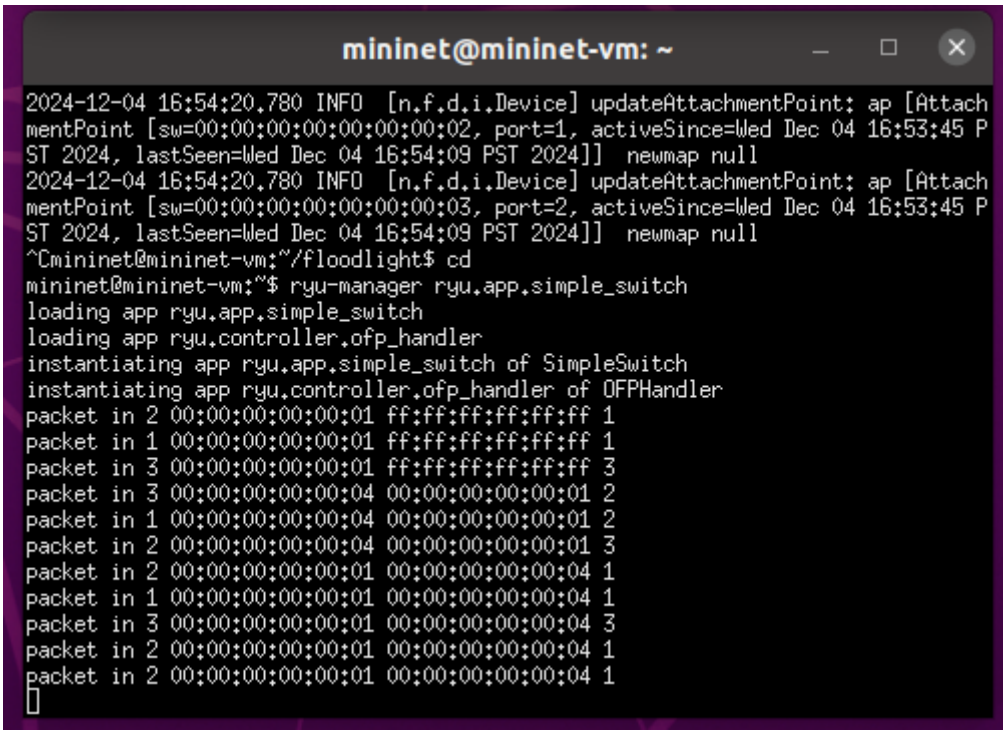
Essa topologia foi escolhida por ser simples o suficiente para ser gerenciada por controladores de diferentes níveis de complexidade, ao mesmo tempo em que permite observar as características de desempenho e escalabilidade dos controladores SDN, testando a escalabilidade com 2, 3 e 4 níveis.

5. Configuração dos Controladores

Ryu: Configurado utilizando a seguinte linha de comando que inicializa o controlador para o uso com o mininet (Figura 4):

$ ryu-manager ryu.app.simple_switch

Figura 4: Ryu em funcionamento no terminal.

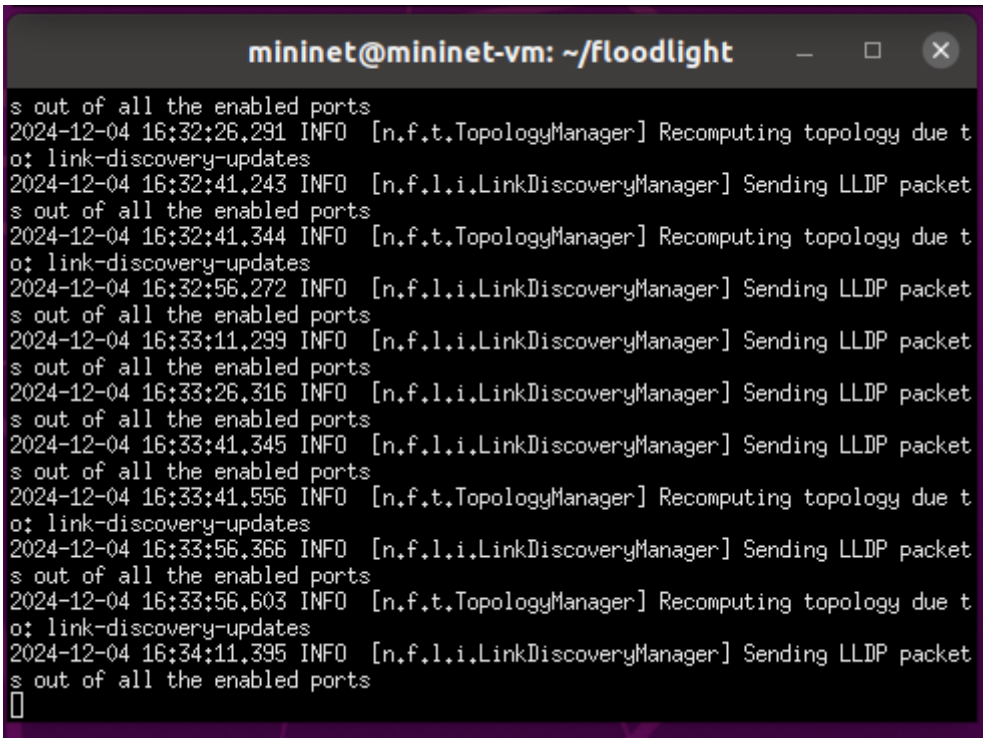


Fonte: O autor

Floodlight: Após a instalação, o controlador foi inicializado para utilizar junto com o mininet com o comando (Figura 5):

$ sudo java -jar target/floodlight.jar

Figura 5: Floodlight em funcionamento no terminal.



Fonte: O autor

6. Geração de Tráfego

Para testar o desempenho dos controladores, foram utilizados geradores de tráfego, como o iperf e o ping, para simular diferentes níveis de carga na rede. Esses testes permitiram avaliar o comportamento dos controladores sob diferentes condições de tráfego.

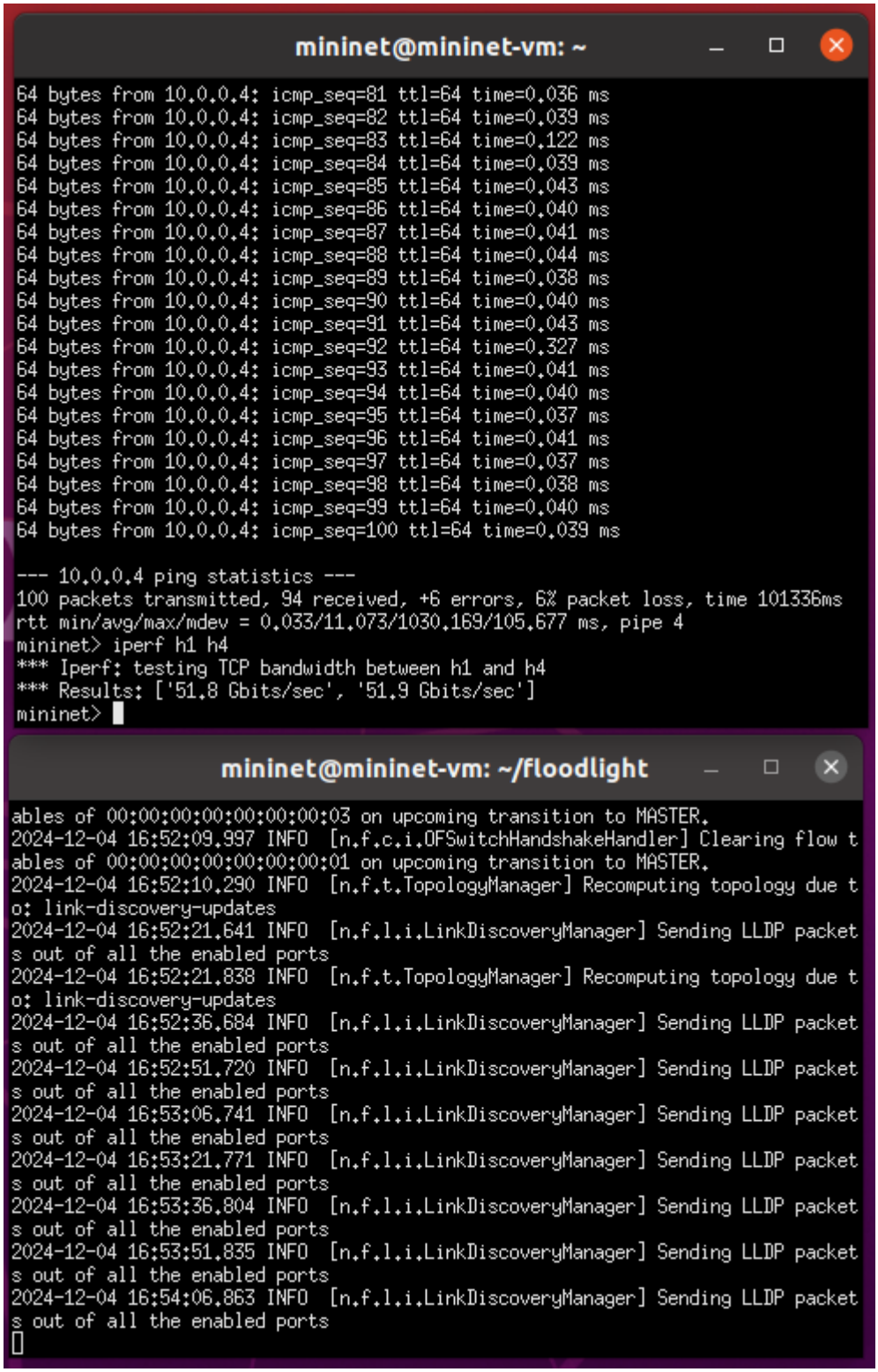
Exemplo de geração de tráfego com iperf:

$ iperf -c 100

7. Monitoramento e Análise

O desempenho dos controladores foi monitorado em tempo real utilizando o mininet (Figura 6).

Figura 6: Floodlight em teste com erro de pacotes.



Fonte: O autor

3.2 Desenvolvimento de Cenários de Teste

As topologias em árvore no Mininet são definidas por dois parâmetros: *depth* (profundidade) e *fanout* (número de filhos por nó). Uma descrição correta de como essas topologias são configuradas é a seguinte:

Topologia com Depth=2, Fanout=2

- Esta configuração resulta em uma árvore onde cada nó (incluindo a raiz) tem dois filhos, e a árvore tem duas camadas de profundidade (excluindo a raiz). Isso cria um total de 4 hosts na última camada. A configuração completa incluiria o nó raiz, 2 nós no primeiro nível abaixo da raiz, e 4 nós no segundo nível abaixo da raiz, resultando em 4 hosts conectados.

Topologia com Depth=3, Fanout=3

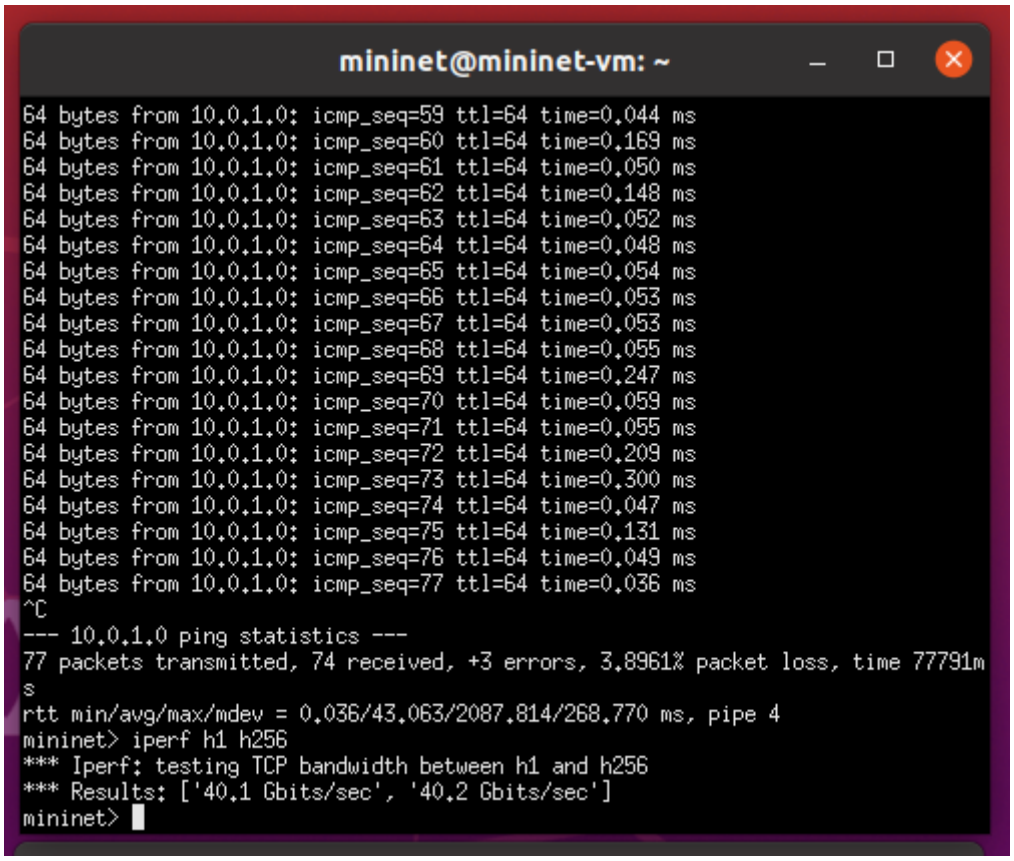
- Neste cenário, a árvore é construída com três camadas de profundidade (excluindo a raiz), e cada nó se expande para três filhos. Isso resulta em 27 hosts no último nível, pois a árvore expande-se exponencialmente com cada camada adicionada.

Topologia com Depth=4, Fanout=4

- A configuração mais complexa, com quatro camadas de profundidade (excluindo a raiz). Cada nó na árvore se expande para quatro filhos, resultando em 256 hosts no último nível (Figura 7).

Essas topologias são essenciais para avaliar como os controladores SDN gerenciam redes de diferentes escalas e complexidades, especialmente em termos de escalabilidade e desempenho sob cargas crescentes de rede. A precisão na configuração dessas topologias é crucial para garantir a validade dos testes e a relevância dos resultados obtidos.

Figura 7: Floodlight sendo testado para 256 hosts.



Fonte: O autor

3.3 Definição dos Critérios de Avaliação

Para garantir uma análise precisa e relevante dos controladores SDN Ryu e Floodlight, os critérios de avaliação foram estabelecidos com base nos objetivos principais do estudo. Esses critérios são essenciais para medir de forma objetiva o desempenho e a eficácia dos controladores em diferentes cenários e condições de rede. Cada critério é cuidadosamente selecionado para alinhar-se com as metas específicas da pesquisa, proporcionando uma base sólida para comparação e análise.

Principais Critérios de Avaliação:

1. **Desempenho:** Inclui a análise de latência e throughput para avaliar quão eficientemente cada controlador processa e transmite dados. Este critério é crucial para determinar a capacidade dos controladores de manter a integridade e a rapidez da comunicação de dados sob diversas cargas de rede.

Medido com o comando ping entre os hosts para testar a taxa de transferência de pacotes e iperf para throughput.

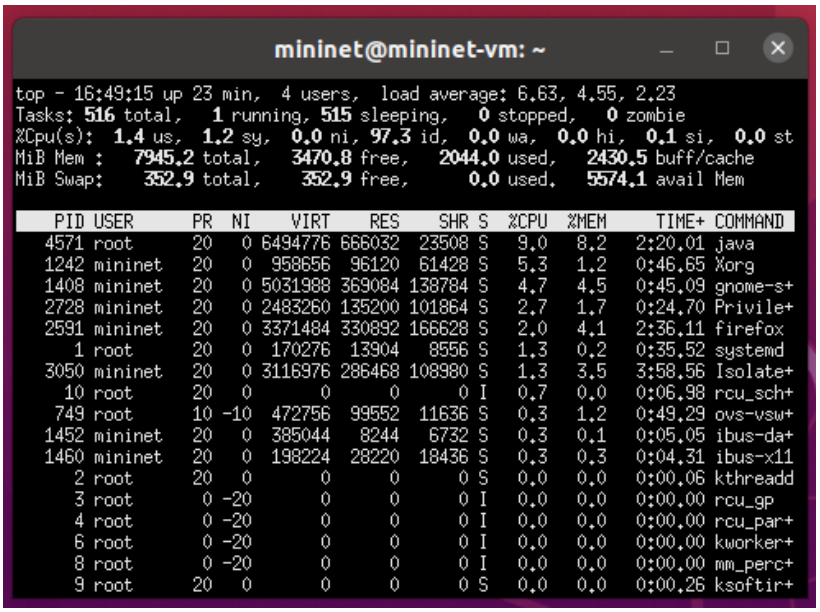
2. **Escalabilidade:** Avalia a habilidade dos controladores de expandir sua capacidade de gerenciamento à medida que o número de hosts ou dispositivos na rede aumenta. Esse critério é vital para entender como cada controlador se comporta em redes em expansão, mantendo a estabilidade e o desempenho. Configurado com 4, 27 e 256 hosts.
3. **Facilidade de Uso:** Considera a simplicidade da instalação e configuração inicial dos controladores, um fator importante para usuários que podem ter diferentes níveis de habilidade técnica. Facilidade de uso também influencia a rapidez com que um sistema pode ser implementado e operacionalizado em um ambiente real. Critério visa a facilidade de instalação e uso após iniciado.
4. **Consumo de Recursos:** Mede o uso de recursos de sistema, como CPU e memória, pelos controladores, especialmente sob condições de alta carga. Este critério é essencial para avaliar a eficiência dos controladores e sua adequação para operar em ambientes com recursos limitados ou restritos. Usando o comando htop para monitorar o uso.
5. **Interoperabilidade:** Examina a capacidade dos controladores de operar harmoniosamente com diferentes dispositivos e protocolos de rede. A interoperabilidade é fundamental para assegurar que os controladores possam ser integrados em diversos ambientes de rede sem incompatibilidades ou problemas de comunicação. A análise é feita com diferentes números de hosts.
6. **Customização e Extensibilidade:** Avalia a documentação disponível e a capacidade de customização dos controladores. Este critério é crucial para usuários que precisam adaptar ou estender as funcionalidades dos controladores para atender a requisitos específicos de rede ou para incorporar novas tecnologias.

3.4 Execução dos Testes

A execução dos testes para avaliar os controladores SDN Ryu e Floodlight foi realizada utilizando um ambiente controlado no Mininet, com topologias variadas para testar sua eficácia sob diferentes condições de rede (Figura 8). As topologias foram

estruturadas com variações nos parâmetros de *depth* e *fanout*, ambos variando de 2 a 4, para criar uma gama de cenários de rede com complexidades crescentes.

Figura 8: Uso de recursos pelo Floodlight.



Fonte: O autor

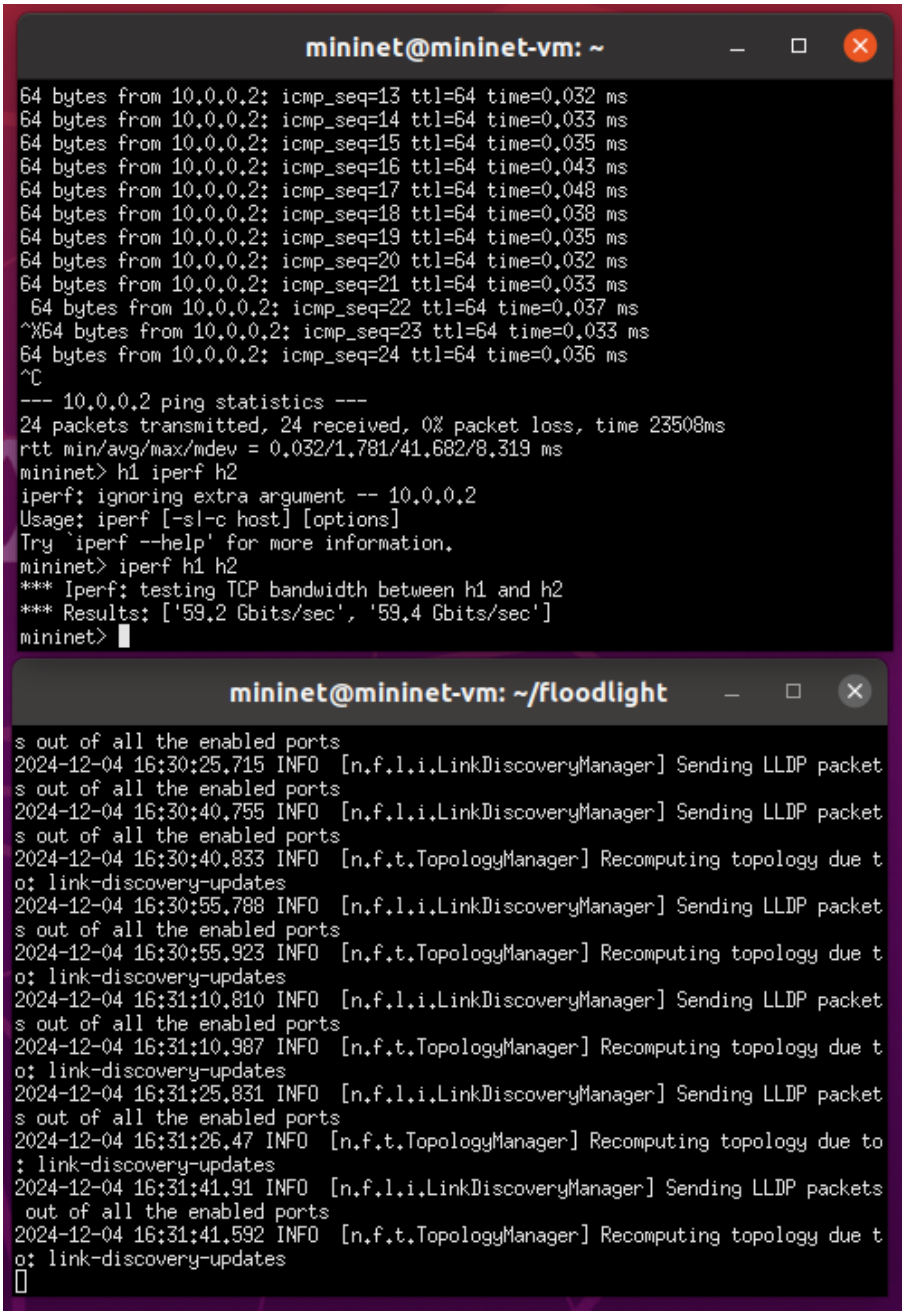
Procedimento de Teste:

1. **Configuração Inicial:** O Mininet é configurado para cada teste com topologias específicas que variam o *depth* (profundidade da rede) e o *fanout* (número de conexões por nó). Esta configuração é fundamental para simular redes de diferentes tamanhos e complexidades.
2. **Execução de Cenários:** Os testes são executados para cada configuração de topologia, ativando o Ryu ou o Floodlight como o controlador da rede. Os cenários são desenhados para avaliar como os controladores gerenciam o aumento progressivo no número de dispositivos e no volume de tráfego.
3. **Monitoramento e Coleta de Dados:** As métricas de desempenho, como latência e throughput, são coletadas diretamente das saídas do terminal do Mininet. A coleta de dados se foca em extrair informações usando as saídas do mininet sobre o desempenho dos controladores.
4. **Documentação de Resultados:** Todos os resultados são cuidadosamente documentados, incluindo detalhes das topologias utilizadas, as métricas de desempenho registradas e observações sobre o comportamento dos

controladores durante os testes. Essa documentação fornece a base para a análise posterior e suporta as conclusões do estudo.

A Figura 9 mostra o controlador Floodlight em uso com o Mininet.

Figura 9: O Floodlight em uso conjunto com o Mininet.



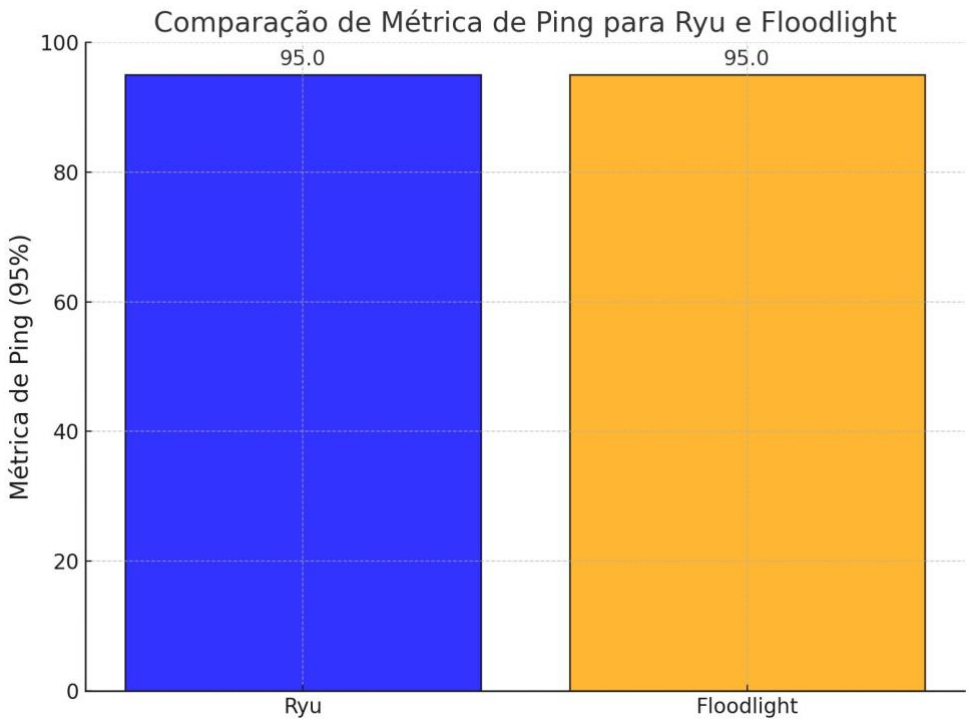
Fonte: O autor

4. RESULTADOS E DISCUSSÃO

4.1 Desempenho dos Controladores SDN

A avaliação de desempenho dos controladores SDN Ryu e Floodlight revelou resultados importantes em termos de latência e throughput. Ambos os controladores apresentaram taxas de transferência de pacotes entre 95% e 99,9%, indicando um desempenho de latência bastante semelhante em diversas configurações de rede, no Gráfico 1 temos o comparativo de ping entre os dois controladores.

Gráfico 1: Taxa de transferência de pacotes

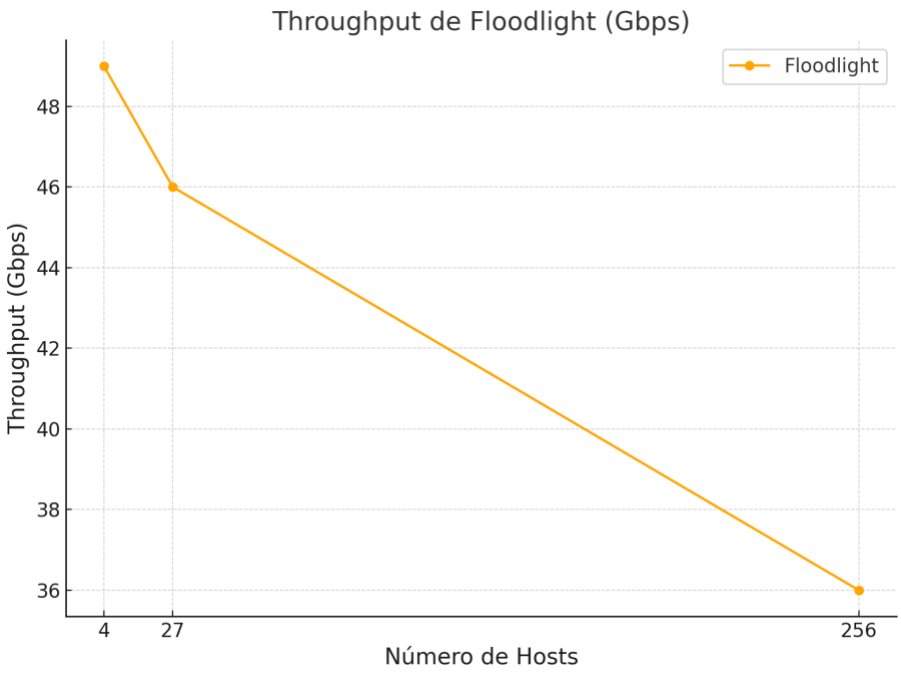


Fonte: O autor

No que se refere ao throughput, o Ryu demonstrou uma capacidade superior, mantendo altas taxas de transferência em diferentes cenários de teste: 52 Gbps com 256 hosts, 57 Gbps com 27 hosts, e 54 Gbps com 4 hosts. Em contraste, o Floodlight apresentou um throughput de 36 Gbps com 256 hosts, 46 Gbps com 27 hosts, e 49 Gbps com 4 hosts, sugerindo um desempenho levemente inferior ao Ryu,

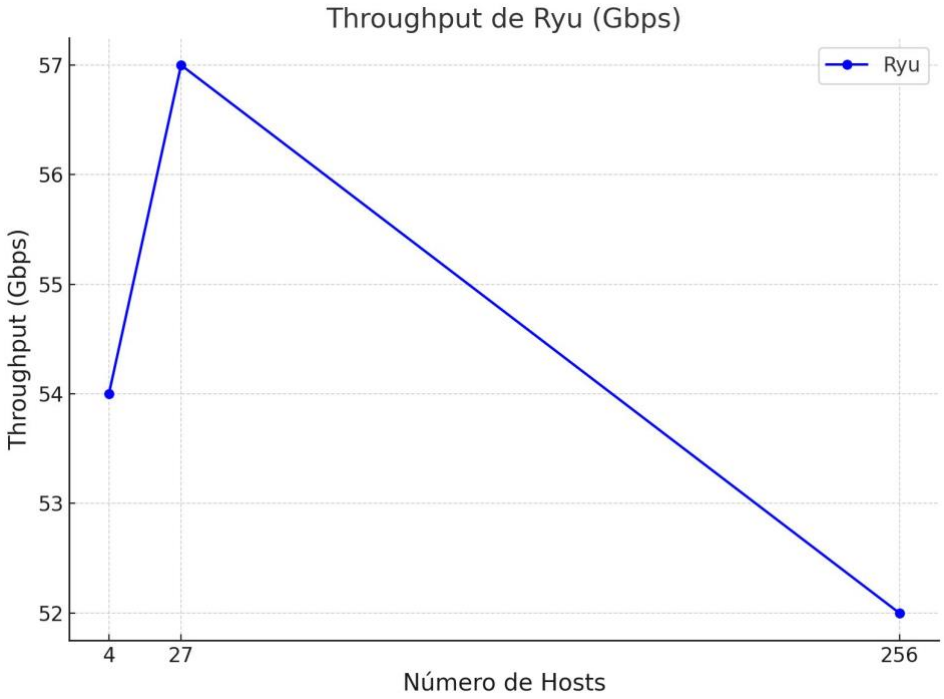
especialmente em redes maiores, nos Gráficos 2 e 3 temos o comparativo de uso de banda para os controladores.

Gráfico 2: Throughput Floodlight



Fonte: O autor

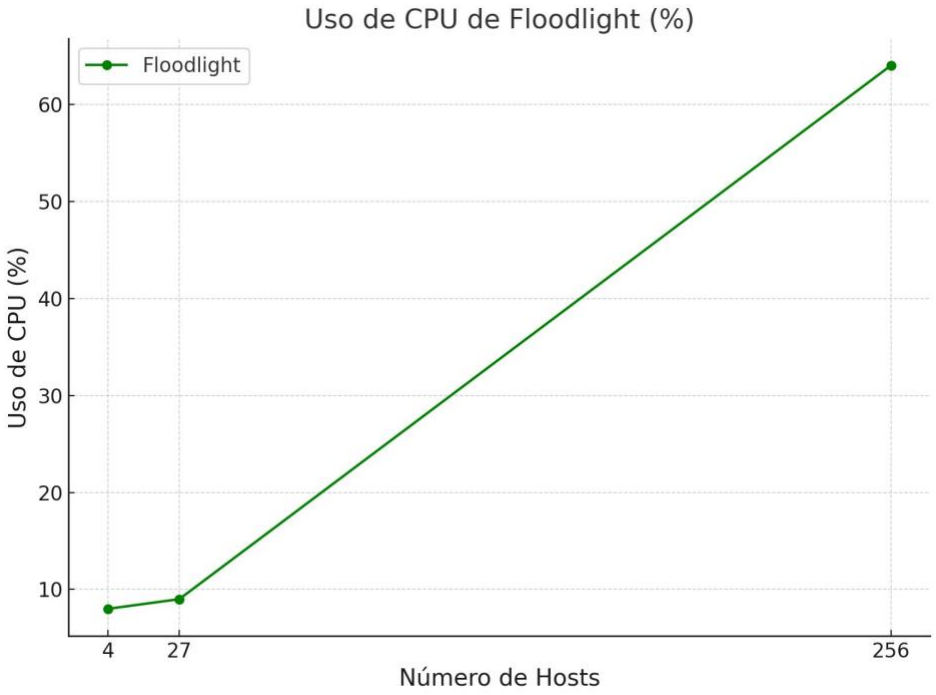
Gráfico 3: Throughput Ryu



Fonte: O autor

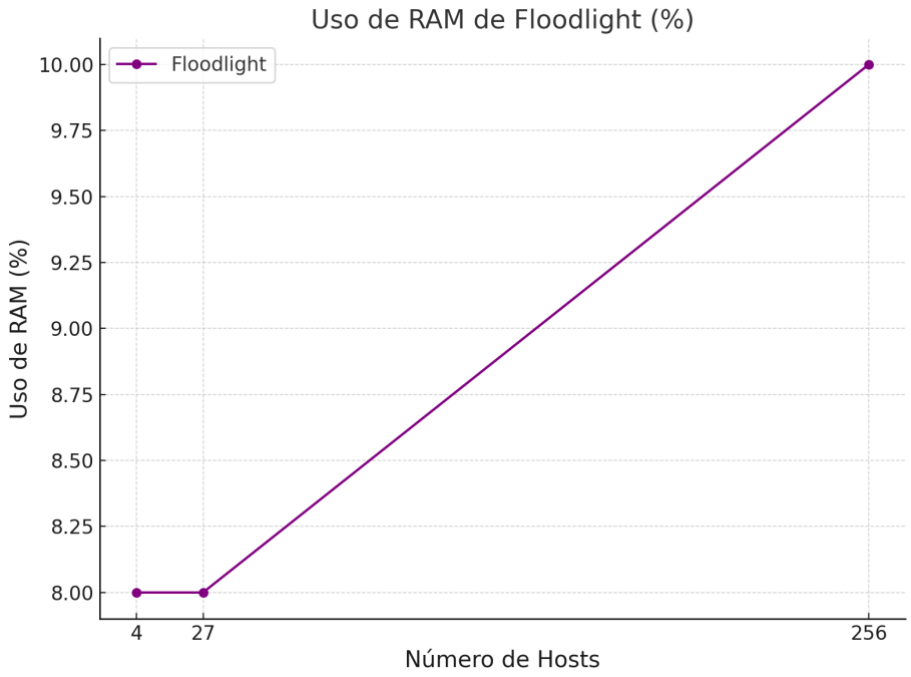
Além disso, observou-se que o Floodlight consumiu mais recursos de processador e memória à medida que o número de hosts aumentava, enquanto o Ryu manteve um consumo consistente de recursos, destacando sua eficiência em manter a performance mesmo sob carga crescente, nos Gráficos 4, 5, 6 e 7 temos o uso de CPU e RAM para os dois controladores.

Gráfico 4: Uso de CPU Floodlight



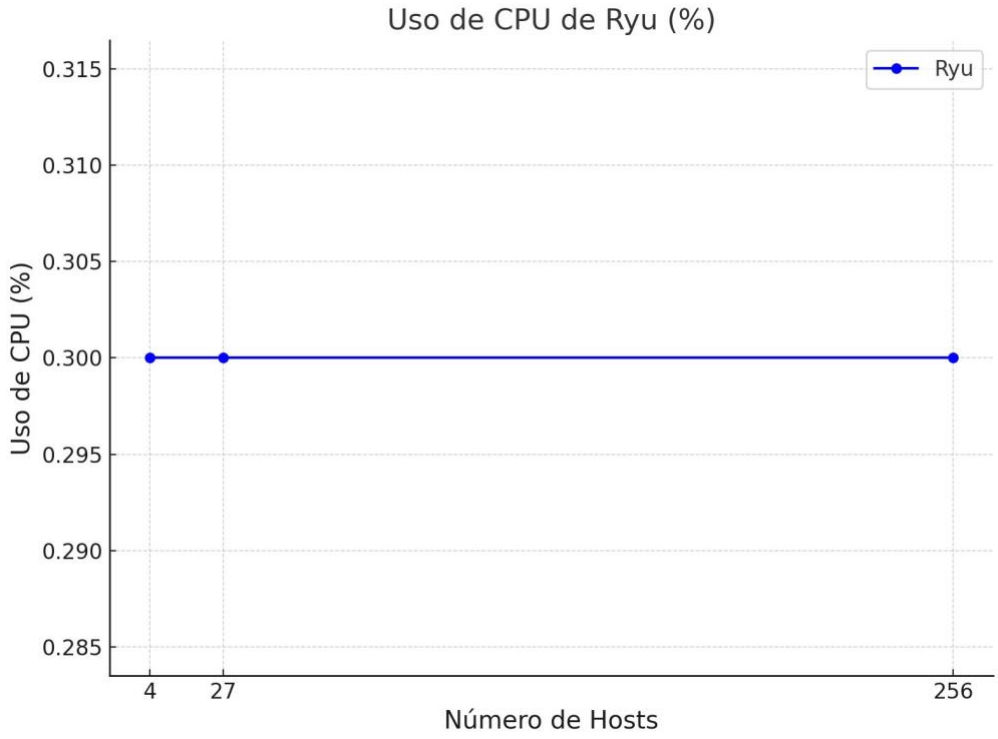
Fonte: O autor

Gráfico 5: Uso de RAM Floodlight



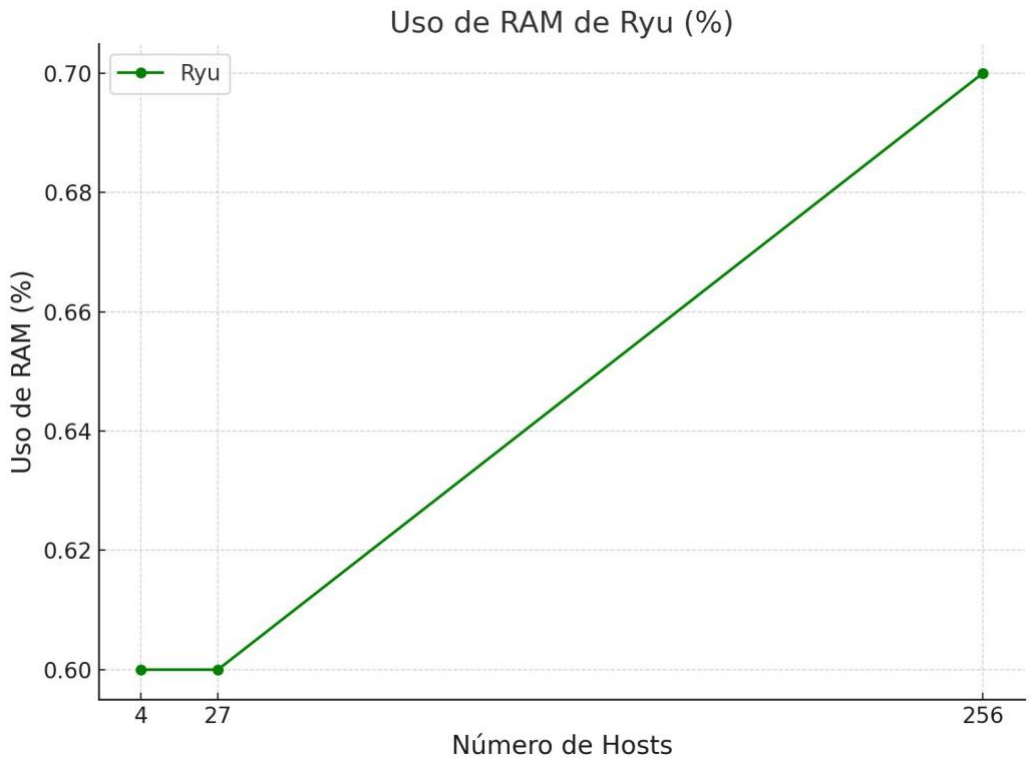
Fonte: O autor

Gráfico 6: Uso de CPU Ryu



Fonte: O autor

Gráfico 7: Uso de RAM Ryu



Fonte: O autor

Portanto, esses resultados indicam que, embora ambos os controladores ofereçam desempenho de latência semelhante e sejam bem documentados, proporcionando facilidade de customização e extensibilidade, o Ryu se sobressai em termos de throughput e eficiência no consumo de recursos, especialmente em cenários de maior escala.

4.2 Escalabilidade

A análise de escalabilidade dos controladores SDN Ryu e Floodlight mostrou resultados distintos quando submetidos a um aumento no número de hosts. Enquanto ambos os controladores foram capazes de funcionar adequadamente em todos os testes de rede, o Floodlight enfrentou dificuldades com o aumento de hosts, o que impactou seu desempenho. Especificamente, problemas relacionados ao consumo de recursos foram notados, com um aumento significativo no uso de processador e memória conforme o número de hosts crescia.

Por outro lado, o Ryu manteve um desempenho consistente e estável, sem variações significativas no consumo de recursos, mesmo com o aumento do número de hosts. Esta consistência se reflete em sua capacidade de escalabilidade, demonstrando ser uma opção robusta para ambientes de rede que esperam crescimento ou variações no número de dispositivos conectados.

Portanto, enquanto ambos os controladores são capazes de operar em redes em expansão, o Ryu mostrou-se mais eficaz ao lidar com redes de grande escala sem comprometer o desempenho, fazendo dele a escolha preferencial para cenários que requerem alta escalabilidade com eficiência no consumo de recursos.

4.3 Facilidade de Uso

A facilidade de uso dos controladores Floodlight e Ryu foi avaliada considerando os processos de instalação, configuração inicial e gerenciamento durante os testes.

O Ryu destacou-se pela simplicidade no processo de instalação e configuração, que pôde ser realizado de forma rápida e direta, sem a necessidade de ajustes complexos ou dependências adicionais. Sua documentação clara e interface de linha de comando intuitiva também facilitaram o início dos testes e o gerenciamento das funções do controlador.

Em contrapartida, o Floodlight apresentou um processo de instalação mais complexo, exigindo a resolução de algumas dependências e configurações específicas antes de ser totalmente funcional. Apesar dessa dificuldade inicial, o uso e o gerenciamento dos dois controladores durante os testes foram semelhantes, com ambos oferecendo interfaces de linha de comando eficientes e respostas estáveis aos comandos de rede.

Portanto, embora o Ryu tenha se mostrado mais fácil de instalar e configurar, ambos os controladores apresentaram uma experiência de uso semelhante após a etapa inicial, proporcionando estabilidade e eficiência no gerenciamento da rede simulada.

4.4 Comparação entre os Controladores quanto aos critérios de avaliação

Ao comparar os controladores SDN Ryu e Floodlight, observou-se que:

1. **Desempenho de Latência:** Ambos os controladores mostraram desempenho semelhante com latências entre 95% e 99,9% nos testes de ping, indicando uma resposta eficaz em condições normais de rede.
2. **Throughput:** O Ryu exibiu throughput superior em todos os cenários testados (52 Gbps com 256 hosts, 57 Gbps com 27 hosts, e 54 Gbps com 4 hosts) em comparação ao Floodlight (36 Gbps com 256 hosts, 46 Gbps com 27 hosts, e 49 Gbps com 4 hosts), destacando sua maior eficiência em gerenciar dados em diversas configurações de rede.
3. **Escalabilidade:** Enquanto o Floodlight enfrentou dificuldades com o aumento de hosts, refletido por um maior consumo de recursos, o Ryu manteve sua performance estável, demonstrando melhor escalabilidade em ambientes de rede crescentes.
4. **Facilidade de Uso:** Ryu provou ser mais fácil de configurar, ideal para implementações rápidas e usuários menos experientes, enquanto o Floodlight, apesar de mais complexo na instalação, oferece funcionalidades comparáveis após a configuração.
5. **Consumo de Recursos:** O Ryu manteve um consumo de recursos constante mesmo com o aumento de hosts, ao contrário do Floodlight que mostrou aumento no consumo de processador e memória.
6. **Documentação e Extensibilidade:** Ambos são bem documentados com ampla disponibilidade de recursos online para customização e extensão, o que facilita a integração e adaptação em diferentes ambientes de rede, temos exemplos como os quatro artigos mencionados e os tutoriais de cada ferramenta nos seus próprios sites e o de Roy et al. (2019) para o Ryu.

Essas observações indicam que, enquanto o Ryu oferece vantagens em throughput, facilidade de configuração e manutenção de performance sob carga, o Floodlight ainda se mantém competitivo devido à sua robustez após a instalação inicial.

5. CONSIDERAÇÕES FINAIS

5.1 Recapitulação dos Objetivos

Este trabalho teve como principal objetivo realizar uma comparação detalhada entre os controladores SDN Ryu, Opendaylight e Floodlight, avaliando-os em termos de desempenho, escalabilidade, facilidade de uso, consumo de recursos e interoperabilidade em diferentes cenários de rede. Devido a problemas de execução o estudo buscou identificar qual controlador entre Ryu e Floodlight oferece a melhor performance em ambientes de rede simulados.

Objetivos Alcançados:

1. **Desempenho:** Foi constatado que ambos os controladores têm performance semelhante em latência, mas o Ryu apresentou throughput superior em todos os cenários testados, indicando uma melhor gestão de largura de banda.
2. **Escalabilidade:** Enquanto o Ryu manteve a performance estável independentemente do aumento no número de hosts, o Floodlight mostrou dificuldades em cenários com maior número de hosts, especialmente no que tange ao consumo de recursos.
3. **Facilidade de Uso:** O Ryu destacou-se por sua facilidade de configuração, sendo mais acessível para instalação e ajustes iniciais, o que é vantajoso para ambientes que exigem rápida configuração e para usuários menos experientes.
4. **Consumo de Recursos:** Foi observado que o Floodlight consumia mais recursos com o aumento do número de hosts, ao contrário do Ryu, que manteve um consumo consistente, evidenciando uma melhor eficiência.
5. **Interoperabilidade:** Ambos os controladores funcionaram bem em todos os testes de rede realizados, mas o Floodlight apresentou problemas ao escalar com o aumento de hosts, enquanto o Ryu não mostrou declínio na performance.
6. **Documentação e Extensibilidade:** Os dois controladores possuem boas pesquisas e integrações mencionadas, logo, são bem documentados.

Os resultados deste estudo proporcionam insights valiosos sobre as capacidades e limitações de cada controlador, com o Ryu emergindo como a opção mais robusta e eficiente para a maioria dos cenários de rede. Este trabalho não

apenas atingiu seus objetivos, mas também abriu caminho para futuras investigações, sugerindo áreas que requerem mais exploração, especialmente em relação ao Floodlight e sua gestão de recursos em redes grandes.

5.2 Contribuições para o Conhecimento

Esta pesquisa gerou contribuições relevantes tanto para a área acadêmica quanto para aplicações práticas em Redes Definidas por Software (SDN), ao analisar e comparar os controladores Floodlight e Ryu com base em critérios objetivos e replicáveis.

Análise Comparativa Detalhada

A pesquisa forneceu uma análise comparativa entre dois dos controladores SDN mais utilizados, Floodlight e Ryu, considerando métricas como desempenho, escalabilidade e facilidade de uso.

Este estudo contribui para preencher uma lacuna na literatura ao realizar uma avaliação prática e sistemática das capacidades de ambos os controladores em ambientes simulados com topologias escaláveis.

Destaque para Facilidades e Limitações

Os resultados destacaram que o **Ryu** é mais acessível para implementação devido à sua instalação simples e configuração direta, tornando-o uma opção ideal para ambientes experimentais e acadêmicos.

O **Floodlight**, por outro lado, demonstrou robustez e eficiência em escalabilidade, apesar de exigir maior complexidade durante a instalação inicial.

Avaliação de Escalabilidade em Cenários Controlados

A análise de topologias com *fanout* e *depth* variando de 2 a 4 revelou que ambos os controladores podem gerenciar redes crescentes sem degradação significativa de desempenho, o que contribui para a aplicação prática em redes maiores e mais complexas.

Aplicações para Redes e Ambientes Acadêmicos

A pesquisa reforça a relevância de critérios como facilidade de instalação e uso para instituições acadêmicas, enquanto identifica robustez e escalabilidade como fatores críticos para redes corporativas.

Os resultados também fornecem uma base sólida para futuros estudos em ambientes híbridos ou que envolvam tecnologias emergentes como IoT e 5G.

Validação e Novas Perspectivas

Os achados desta pesquisa validam conclusões de estudos anteriores sobre a eficiência de Floodlight e Ryu em redes escaláveis, ao mesmo tempo que destacam novas nuances, como a equivalência de desempenho em cenários de maior complexidade. A ausência de resultados para o OpenDaylight aponta para desafios de configuração e estabilidade que podem ser explorados em estudos futuros.

Ao fornecer uma análise detalhada e prática, esta pesquisa contribui significativamente para o campo das SDN, auxiliando pesquisadores, desenvolvedores e profissionais a entenderem melhor as capacidades e limitações de Floodlight e Ryu, e a tomarem decisões informadas sobre a escolha do controlador para diferentes cenários de rede.

5.3 Relevância do Estudo

Este estudo sobre os controladores SDN Ryu e Floodlight tem uma relevância significativa tanto para a academia quanto para a indústria, contribuindo de maneira substancial para o campo das Redes Definidas por Software (SDN). Aqui estão os principais aspectos que destacam a importância deste trabalho:

Avanço no Conhecimento Técnico de SDN:

Ao fornecer uma análise comparativa detalhada do desempenho, escalabilidade, facilidade de uso e consumo de recursos entre os controladores Ryu e Floodlight, o estudo enriquece o corpo de conhecimento técnico em SDN. Ele oferece dados valiosos e insights que ajudam a compreender as nuances operacionais desses controladores em diferentes cenários de rede.

Suporte à Tomada de Decisão em Redes:

Os resultados deste estudo são cruciais para profissionais envolvidos na seleção e implementação de soluções SDN. Com uma compreensão clara dos pontos fortes e limitações de cada controlador, os administradores de rede podem tomar decisões mais informadas que alinham a tecnologia SDN com as necessidades específicas de suas organizações, otimizando a performance e a eficiência das redes.

Impulso para Inovação e Aplicações Práticas:

A investigação estimula a inovação ao identificar áreas onde melhorias são necessárias, como o consumo de recursos pelo Floodlight em redes densas. Esses insights podem inspirar o desenvolvimento de novas funcionalidades ou ajustes nos controladores existentes, impulsionando a evolução de tecnologias de redes mais adaptativas e sustentáveis.

Formação e Educação em Redes:

Este estudo também tem implicações educacionais ao evidenciar a facilidade de uso do Ryu, tornando-o uma opção preferencial para contextos educacionais onde a simplicidade e a acessibilidade são cruciais. Instituições educacionais podem se beneficiar desses achados ao configurar laboratórios de ensino e treinamento em redes, proporcionando aos estudantes experiências de aprendizado mais eficazes e engajadoras.

Contribuição para Redes Sustentáveis:

Ao destacar o desempenho eficiente do Ryu em termos de consumo de recursos, o estudo alinha-se com as crescentes demandas por tecnologias que suportem a sustentabilidade ambiental. Organizações que buscam reduzir o consumo de energia e a pegada de carbono podem considerar essas informações ao planejar e implementar suas infraestruturas de rede.

REFERÊNCIAS

CHOUHAN, R. K.; ATULKAR, M.; NAGWANI, N. K. **Performance comparison of Ryu and Floodlight controllers in different SDN topologies**. 2019 1st International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE), Bangalore, Índia, 2019. p. 188-191. DOI: 10.1109/ICATIECE45860.2019.9063806. Disponível em: https://ieeexplore.ieee.org/document/9063806. Acesso em: 18 nov. 2024.

FEAMSTER, N.; REXFORD, J.; ZEGURA, E. **The road to SDN: An intellectual history of programmable networks**. ACM SIGCOMM Computer Communication Review, v. 44, n. 2, p. 87-98, 2014.

FLOODLIGHT DOCUMENTATION. Floodlight Controller - Confluence, [s.d.]. Disponível em: https://floodlight.atlassian.net/wiki/spaces/floodlightcontroller/overview. Acesso em: 29 nov. 2024.

Getting Started: Please Read: Ryu not Currently Maintained. GitHub, [s.d.]. Disponível em: https://ryu.readthedocs.io/en/latest/getting_started.html. Acesso em: 29 nov. 2024.

Installation Guide. Floodlight Controller - Confluence, [s.d.]. Disponível em: https://floodlight.atlassian.net/wiki/spaces/floodlightcontroller/pages/1343544/Installation+Guide. Acesso em: 29 nov. 2024.

Installing Opendaylight - Opendaylight Documentation Nitrogen Documentation. Disponível em: https://test-odl-docs.readthedocs.io/en/latest/getting-started-guide/installing_opendaylight.html. Acesso em: 29 nov. 2024.

Installing Opendaylight - Opendaylight Documentation Titanium Documentation. Disponível em: https://docs.opendaylight.org/en/latest/getting-started-guide/installing_opendaylight.html. Acesso em: 29 nov. 2024.

Installing Opendaylight. OpenDaylight Documentation, 2016-2024. Disponível em: https://docs.opendaylight.org/en/latest/getting-started-guide/installing_opendaylight.html. Acesso em: 29 nov. 2024.

KIM, H.; FEAMSTER, N. **Improving network management with software defined networking.** IEEE Communications Magazine, v. 51, n. 2, p. 114-119, 2013.

KREUTZ, D. et al. **Software-defined networking: A comprehensive survey**. Proceedings of the IEEE, v. 103, n. 1, p. 14-76, 2014. DOI: 10.1109/JPROC.2014.2371999.

LI, Y.; GUO, X.; PANG, X.; PENG, B.; LI, X.; ZHANG, P. **Performance analysis of Floodlight and Ryu SDN controllers under Mininet simulator**. 2020 IEEE/CIC International Conference on Communications in China (ICCC Workshops), Chongqing, China, 2020. p. 85-90. DOI: 10.1109/ICCCWorkshops49972.2020.9209935. Disponível em: https://ieeexplore.ieee.org/document/9209935. Acesso em: 18 nov. 2024.

LINKLETTER, B. **How to use MiniEdit, Mininet's graphical user interface**. Disponível em: https://brianlinkletter.com/2015/04/how-to-use-miniedit-mininets-graphical-user-interface/. Acesso em: 29 nov. 2024.

MAMUSHIANE, L.; LYSKO, A.; DLAMINI, S. **A comparative evaluation of the performance of popular SDN controllers**. 2018 Wireless Days (WD), Dubai, Emirados Árabes Unidos, 2018. p. 54-59. DOI: 10.1109/WD.2018.8361694. Disponível em: https://ieeexplore.ieee.org/document/8361694. Acesso em: 29 nov. 2024.

MARSICO, Antonio; DORIGUZZI CORIN, Roberto; SIRACUSA, Domenico. **Overcoming the memory limits of network devices in SDN-enabled data centers**. In: IFIP/IEEE International Symposium on Integrated Network Management (IM), 2017. DOI: 10.23919/INM.2017.7987402. Disponível em: https://www.researchgate.net/publication/316877954_Overcoming_the_Memory_Limits_of_Network_Devices_in_SDN-enabled_Data_Centers. Acesso em: 29 nov. 2024.

MCKEOWN, N. et al**. OpenFlow: enabling innovation in campus networks**. ACM SIGCOMM Computer Communication Review, v. 38, n. 2, p. 69-74, 2008.

Mininet: an Instant Virtual Network on Your Laptop (Or Other Pc) - Mininet. Disponível em: https://mininet.org. Acesso em: 29 nov. 2024.

NUNES, B. A. A. et al. **A survey of software-defined networking: Past, present, and future of programmable networks**. IEEE Communications Surveys & Tutorials, v. 16, n. 3, p. 1617-1634, 2014.

Opendaylight. The Linux Foundation Projects, 2024. Disponível em: https://www.opendaylight.org/. Acesso em: 29 nov. 2024.

ROWSHANRAD, Shiva; PARSAEI, Mohamad Reza; KESHTGARI, Manijeh. **Implementing NDN using SDN: a review on methods and applications**. IIUM Engineering Journal, v. 17, n. 2, 2016. Disponível em: https://www.academia.edu/30193139/PERFORMANCE_EVALUATION_OF_SDN_CONTROLLERS_FLOODLIGHT_AND_OPENDAYLIGHT. Acesso em: 29 nov. 2024.

ROY, Shanto. **Software defined network: Mininet with Ryu controller**. Roy's Blog, 2019. Disponível em: https://shantoroy.com/sdn/sdn-mininet-ryu/. Acesso em: 29 nov. 2024.

RYU SDN Framework. Disponível em: https://ryu-sdn.org/. Acesso em: 29 nov. 2024.

SHARMA, Amrita; VERMA, Gargi Shankar. **Performance comparison of Ryu and floodlight SDN controller**. AIP Conference Proceedings, v. 2705, n. 1, 2023. DOI: 10.1063/5.0133684. Disponível em: https://pubs.aip.org/aip/acp/article-abstract/2705/1/040017/2897049/Performance-comparison-of-Ryu-and-floodlight-SDN?redirectedFrom=fulltext. Acesso em: 18 nov. 2024.

TOOTOONCHIAN, Amin; GORBUNOV, Sergey; GANJALI, Yashar; CASADO, Martin; SHERWOOD, Rob. **On controller performance in software-defined networks**. *In:* USENIX Workshop on Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services (Hot-ICE), 2012. p. 54. Disponível em: https://www.researchgate.net/publication/262423689_On_Controller_Performance_in_Software-defined_Networks. Acesso em: 18 nov. 2024.