



Instituto Federal de Educação, Ciência e Tecnologia da Paraíba
Campus Campina Grande
Coordenação do Curso Superior de Engenharia da Computação

TESTBED: AMBIENTE DE TESTES PARA SIMULAÇÃO E AVALIAÇÃO DE REDES DE COMPUTADORES

JOSÉ ROBERTO PEIXOTO FERREIRA FILHO

Orientador: Dr. ÉWERTON RÔMULO SILVA CASTRO

Campina Grande, Junho de 2026
® José Roberto Peixoto Ferreira Filho



Instituto Federal de Educação, Ciência e Tecnologia da Paraíba
Campus Campina Grande
Coordenação do Curso Superior em Engenharia da Computação

TESTBED: AMBIENTE DE TESTES PARA SIMULAÇÃO E AVALIAÇÃO DE REDES DE COMPUTADORES

JOSÉ ROBERTO PEIXOTO FERREIRA FILHO

Monografia apresentada à Coordenação do
Curso em Engenharia da Computação do
IFPB - Campus Campina Grande, como
requisito parcial para conclusão do curso de
Engenharia da Computação

Orientador: Dr. ÉWERTON RÔMULO SILVA CASTRO

Campina Grande, Junho de 2026

F383t

FERREIRA FILHO, José Roberto Peixoto

Testbed: Ambiente de testes para simulação e avaliação de redes de computadores - Campina grande. José Roberto Peixoto Ferreira Filho . 2026. 104f..

TCC (PDF)

Orientador: Éwerton Rômulo Silva Castro

1. Redes 2. Monitoramento . 3. Virtualização . I. Título II. José Roberto Peixoto Ferreira Filho

CDU: 004.7

***TESTBED*: AMBIENTE DE TESTES PARA SIMULAÇÃO E AVALIAÇÃO DE REDES DE COMPUTADORES**

JOSÉ ROBERTO PEIXOTO FERREIRA FILHO

Dr. Éwerton Rômulo Silva Castro
Orientador

Mikael Marinho Oliveira
Coorientador

Ma. Iana Daya Cavalcante Facundo Passos
Membro da Banca

Dra. Daniella Dias Cavalcante da Silva
Membro da Banca

Campina Grande, Paraíba, Brasil
Junho/2026

Dedico este trabalho, com todo amor e gratidão, aos meus pais, Ana Aparecida da Silva Felismino e José Roberto Peixoto Ferreira, que são a base de tudo em minha vida. Aos meus irmãos, José Augusto da Silva Ferreira e Ítalo Gustavo Ferreira da Silva, pelo companheirismo e por fazerem parte da minha caminhada. À minha fé e à minha religião, a Umbanda Nagô e Jurema Sagrada, que me fortalecem, me orientam e me dão equilíbrio para enfrentar os desafios da vida. Em especial, à minha mãe Iemanjá e ao meu pai Oxóssi, por sua proteção, sabedoria e por guiarem meus caminhos ao longo dessa trajetória.

“Exu matou um pássaro ontem, com uma pedra que ele só jogou hoje.”.

Autor Desconhecido

Agradecimentos

Agradeço, primeiramente, a meu pai Oxalá, pela força e pela oportunidade de concluir mais esta etapa.

Agradeço aos meus pais, Ana Aparecida da Silva Felismino e José Roberto Peixoto Ferreira, por todo amor, apoio, ensinamentos e por nunca desistirem de mim ao longo dessa caminhada. Aos meus irmãos, José Augusto da Silva Ferreira e Ítalo Gustavo Ferreira da Silva, pelo incentivo e companheirismo constantes.

À minha amiga Vanessa Candeia, pela amizade verdadeira, apoio constante e por tornar essa jornada mais leve. Aos meus amigos Lamarck Vinycius, Luan Silva, Marcus Lima e João Saraiva, pelo incentivo, amizade e por compartilharem momentos importantes ao longo dessa trajetória acadêmica.

À minha querida amiga Carla Rached, pela amizade sincera, pelo carinho, apoio incondicional, incentivo constante e por estar presente em momentos importantes da minha vida, contribuindo de forma especial para esta caminhada. À Hilda Galdino e à Angely Sales, pela amizade, apoio e contribuição durante essa jornada.

Ao meu sacerdote Babarolixá Luciano Sátiro, pelos ensinamentos, orientação espiritual e por fortalecer minha fé nos momentos mais desafiadores. À professora Marcia Gardênia Lustosa Pires, pelos ensinamentos, dedicação e orientação ao longo dos projetos acadêmicos.

Ao meu orientador, Ewerton, e ao coorientador, Mikael, pelo apoio, paciência e por todo o conhecimento compartilhado durante o desenvolvimento deste trabalho. A todos os professores que fizeram parte da minha formação, contribuindo com seus conhecimentos e experiências para o meu crescimento acadêmico e profissional.

À minha trajetória acadêmica e profissional, construída com esforço, dedicação e superação, que me trouxe até este momento tão importante.

E, de forma especial, à cantora Anitta — Larissa de Macedo Machado —, que vai muito além de uma inspiração artística, representando para mim um símbolo de força, coragem e superação. Sua trajetória, marcada por determinação, visão e conquistas, me motiva diariamente a sair da zona de conforto, acreditar no meu potencial e persistir, mesmo diante dos desafios. Sua história me lembra que sonhos são possíveis quando há foco, disciplina e fé no próprio caminho.

Por fim, agradeço a todos que contribuíram, direta ou indiretamente, para a realização deste trabalho.

Resumo

Este trabalho apresenta o desenvolvimento e a avaliação de uma plataforma laboratorial do tipo *testbed* voltada para experimentação, monitoramento e análise de desempenho em redes de computadores. A proposta surge da necessidade de ambientes experimentais controlados que permitam a realização de testes reproduzíveis, superando limitações associadas a laboratórios físicos tradicionais e complementando abordagens baseadas exclusivamente em simulação. A metodologia adotada integrou conceitos de virtualização e emulação, utilizando máquinas virtuais interconectadas em um ambiente de testes controlado. Para a geração de tráfego e avaliação do desempenho da rede foram empregadas as ferramentas *Iperf3*, *Zabbix* e *Grafana*, responsáveis pela coleta, monitoramento e visualização das métricas analisadas. Foram realizados testes utilizando os protocolos de transporte *TCP* e *UDP*, considerando cenários de comunicação cliente-servidor e servidor-cliente. As métricas avaliadas incluíram taxa de transferência (*throughput*), *bitrate*, *jitter*, perda de pacotes e utilização dos recursos computacionais dos hosts. Os resultados obtidos demonstraram estabilidade da infraestrutura implementada, com baixas taxas de utilização de *CPU* e memória, ausência de perdas de pacotes nos testes *UDP* e taxas de transmissão próximas ao limite teórico da rede utilizada nos testes *TCP*. Além disso, o monitoramento contínuo permitiu verificar o comportamento dos hosts e validar a capacidade do ambiente em suportar os experimentos realizados. Conclui-se que a utilização de um *testbed* virtual constitui uma alternativa viável para atividades de ensino, pesquisa e avaliação de desempenho de redes, proporcionando flexibilidade, baixo custo, reprodutibilidade e confiabilidade na execução dos experimentos.

Palavras-chave: *Testbed*; redes de computadores; virtualização; desempenho de rede; tcp; udp; monitoramento de redes.

Abstract

This work presents the development and evaluation of a *testbed* platform aimed at experimentation, monitoring, and performance analysis in computer networks. The proposal arises from the need for controlled experimental environments capable of providing reproducible tests, overcoming limitations associated with traditional physical laboratories and complementing approaches based exclusively on simulation. The adopted methodology integrated virtualization and emulation concepts through interconnected virtual machines deployed in a controlled testing environment. The tools *Iperf3*, *Zabbix*, and *Grafana* were employed for traffic generation, monitoring, data collection, and visualization of network performance metrics. Experiments were conducted using the *TCP* and *UDP* transport protocols under both client-server and server-client communication scenarios. The evaluated metrics included throughput, bitrate, jitter, packet loss, and host computational resource utilization. The obtained results demonstrated the stability of the implemented infrastructure, characterized by low CPU and memory utilization, absence of packet loss in *UDP* tests, and transmission rates close to the theoretical limit of the network used in *TCP* tests. Furthermore, continuous monitoring enabled the assessment of host behavior and validated the environment's capability to support the proposed experiments. It is concluded that the use of a virtual *testbed* represents a viable alternative for teaching, research, and network performance evaluation activities, providing flexibility, low cost, reproducibility, and reliability in experimental execution.

Keywords: *Testbed*; computer networks; virtualization; network performance; tcp; udp; network monitoring.

Sumário

Lista de Siglas	xii
Lista de Figuras	xiii
Lista de Tabelas	xvi
1 Introdução	1
1.1 Justificativa e Relevância do Trabalho	1
1.2 Objetivos	2
1.3 Organização do Documento	3
2 Fundamentação Teórica	6
2.1 Redes de Computadores	6
2.2 Redes Virtuais e Virtualização	8
2.3 Emulação e Simulação de Redes	10
2.4 Protocolos de Transporte: TCP e UDP	11
2.5 Métricas de Desempenho de Rede	12
2.6 <i>Testbeds</i> Virtuais em Redes de Computadores	14
2.7 Ferramentas para Geração de Tráfego e Monitoramento de Redes	15
3 Trabalhos Relacionados	17
3.1 Trabalhos Relacionados	17
4 Metodologia	18
4.1 Materiais e Métodos	18
4.2 Abordagem de Pesquisa Quantitativa	19
4.3 Natureza Aplicada da Pesquisa	20
5 Desenvolvimento do Ambiente Experimental	21
5.1 Infraestrutura Física do Laboratório	21
5.2 Topologia de Rede	23
5.3 Configuração do Ambiente Virtualizado	25
5.4 Ferramentas Utilizadas	30

5.5	Visão Geral do Monitoramento do Desempenho	34
6	Resultados Obtidos	38
6.1	Desempenho <i>TCP</i> Cliente-Servidor	38
6.2	Desempenho <i>TCP</i> Servidor-Cliente	39
6.3	Desempenho <i>UDP</i> Cliente-Servidor	41
6.4	Desempenho <i>UDP</i> Servidor-Cliente	42
6.5	Testes de Tráfego de Rede	43
6.6	Métricas de Desempenho dos Computadores	44
6.7	Comparação dos Resultados com Trabalhos Relacionados	45
7	Considerações Finais e Sugestões para Trabalhos Futuros	47
7.1	Sugestões para Trabalhos Futuros	47
A	Capturas de Tela dos Experimentos e Métricas Coletadas	50
A.1	Testes <i>TCP</i> Cliente - Servidor	50
A.2	Testes <i>TCP</i> Servidor - Cliente	55
A.3	Testes <i>UDP</i> Cliente - Servidor	60
A.4	Testes <i>UDP</i> Servidor - Cliente	65
A.5	Métricas de Desempenho Coletadas pelo Zabbix	70
A.6	Resultados do Monitoramento de Rede Obtidos com o Zabbix	76
	Referências Bibliográficas	86

Lista de Siglas

CPU	<i>Central Processing Unit</i> (Unidade Central de Processamento)
DDR4	<i>Double Data Rate 4</i> (Memória DDR de Quarta Geração)
EMULAB	<i>Emulation Laboratory</i> (Laboratório de Emulação)
GB	<i>Gigabyte</i> (Gigabyte)
Grafana	Plataforma de Visualização e Monitoramento de Dados
HDD	<i>Hard Disk Drive</i> (Disco Rígido)
IA	Inteligência Artificial
IP	<i>Internet Protocol</i> (Protocolo de Internet)
LAN	<i>Local Area Network</i> (Rede Local)
LTS	<i>Long Term Support</i> (Suporte de Longo Prazo)
MAN	<i>Metropolitan Area Network</i> (Rede Metropolitana)
NS-3	<i>Network Simulator 3</i> (Simulador de Rede)
OMNeT++	<i>Objective Modular Network Testbed in C++</i> (Plataforma Modular de Testes de Rede)
PCI	<i>Peripheral Component Interconnect</i> (Interconexão de Componentes Periféricos)
QoE	<i>Quality of Experience</i> (Qualidade de Experiência)
RAM	<i>Random Access Memory</i> (Memória de Acesso Aleatório)
SDN	<i>Software Defined Networking</i> (Redes Definidas por Software)
SNMP	<i>Simple Network Management Protocol</i> (Protocolo Simples de Gerenciamento de Rede)
TCP	<i>Transmission Control Protocol</i> (Protocolo de Controle de Transmissão)
TELECOM	<i>Telecommunications</i> (Telecomunicações)
TI	Tecnologia da Informação
UDP	<i>User Datagram Protocol</i> (Protocolo de Datagrama de Usuário)
VDI	<i>Virtual Disk Image</i> (Imagem de Disco Virtual)
VM	<i>Virtual Machine</i> (Máquina Virtual)
WAN	<i>Wide Area Network</i> (Rede de Longa Distância)
WiFi	<i>Wireless Fidelity</i> (Fidelidade sem Fio)

Lista de Figuras

2.1	Modelo TCP/IP	7
2.2	Virtualização vs sistema tradicional	9
2.3	Comparação entre TCP e UDP	12
5.1	Diferentes perspectivas do Laboratório de Telecomunicações (TELECOM), ambiente utilizado para a realização dos experimentos deste trabalho.	22
5.2	Topologia da rede utilizada nos testes.	24
5.3	Processo de criação de uma máquina virtual Ubuntu Desktop 24.04 LTS no VirtualBox.	27
5.4	Configuração de memória RAM e processadores virtuais das máquinas virtuais.	28
5.5	Configuração do armazenamento virtual utilizado pelas máquinas virtuais.	29
5.6	Configuração da interface conectada à rede interna do ambiente experimental.	30
5.7	Interface de monitoramento dos hosts no Zabbix.	31
5.8	Ferramenta Iperf3 instalada no servidor do ambiente experimental.	32
5.9	Monitoramento do tráfego geral da interface de rede enp0s3 por meio da plataforma Zabbix.	33
5.10	Dashboard de monitoramento por computador do tráfego de rede dos hosts do ambiente experimental utilizando o Zabbix.	34
5.11	Dashboard de monitoramento por computador do tráfego de rede dos hosts do ambiente experimental utilizando o Zabbix.	35
5.12	Monitoramento de desempenho do servidor por meio do Zabbix.	36
A.1	Interface de monitoramento do host PC01 durante o teste TCP no Iperf3.	50
A.2	Interface de monitoramento do host PC02 durante o teste TCP no Iperf3.	51
A.3	Interface de monitoramento do host PC03 durante o teste TCP no Iperf3.	51
A.4	Interface de monitoramento do host PC04 durante o teste TCP no Iperf3.	52
A.5	Interface de monitoramento do host PC05 durante o teste TCP no Iperf3.	52
A.6	Interface de monitoramento do host PC06 durante o teste TCP no Iperf3.	53
A.7	Interface de monitoramento do host PC07 durante o teste TCP no Iperf3.	53
A.8	Interface de monitoramento do host PC08 durante o teste TCP no Iperf3.	54
A.9	Interface de monitoramento do host PC09 durante o teste TCP no Iperf3.	54
A.10	Interface de monitoramento do host PC10 durante o teste TCP no Iperf3.	55

A.11 Interface de monitoramento do host PC01 observada pelo servidor Iperf3 durante o teste TCP.	55
A.12 Interface de monitoramento do host PC02 observada pelo servidor Iperf3 durante o teste TCP.	56
A.13 Interface de monitoramento do host PC03 observada pelo servidor Iperf3 durante o teste TCP.	56
A.14 Interface de monitoramento do host PC04 observada pelo servidor Iperf3 durante o teste TCP.	57
A.15 Interface de monitoramento do host PC05 observada pelo servidor Iperf3 durante o teste TCP.	57
A.16 Interface de monitoramento do host PC06 observada pelo servidor Iperf3 durante o teste TCP.	58
A.17 Interface de monitoramento do host PC07 observada pelo servidor Iperf3 durante o teste TCP.	58
A.18 Interface de monitoramento do host PC08 observada pelo servidor Iperf3 durante o teste TCP.	59
A.19 Interface de monitoramento do host PC09 observada pelo servidor Iperf3 durante o teste TCP.	59
A.20 Interface de monitoramento do host PC10 observada pelo servidor Iperf3 durante o teste TCP.	60
A.21 Interface de monitoramento do host PC01 durante o teste UDP.	60
A.22 Interface de monitoramento do host PC02 durante o teste UDP.	61
A.23 Interface de monitoramento do host PC03 durante o teste UDP.	61
A.24 Interface de monitoramento do host PC04 durante o teste UDP.	62
A.25 Interface de monitoramento do host PC05 durante o teste UDP.	62
A.26 Interface de monitoramento do host PC06 durante o teste UDP.	63
A.27 Interface de monitoramento do host PC07 durante o teste UDP.	63
A.28 Interface de monitoramento do host PC08 durante o teste UDP.	64
A.29 Interface de monitoramento do host PC09 durante o teste UDP.	64
A.30 Interface de monitoramento do host PC10 durante o teste UDP.	65
A.31 Interface de monitoramento do host PC01 observada pelo servidor Iperf3 durante o teste UDP.	65
A.32 Interface de monitoramento do host PC02 observada pelo servidor Iperf3 durante o teste UDP.	66
A.33 Interface de monitoramento do host PC03 observada pelo servidor Iperf3 durante o teste UDP.	66
A.34 Interface de monitoramento do host PC04 observada pelo servidor Iperf3 durante o teste UDP.	67
A.35 Interface de monitoramento do host PC05 observada pelo servidor Iperf3 durante o teste UDP.	67

A.36 Interface de monitoramento do host PC06 observada pelo servidor Iperf3 durante o teste UDP.	68
A.37 Interface de monitoramento do host PC07 observada pelo servidor Iperf3 durante o teste UDP.	68
A.38 Interface de monitoramento do host PC08 observada pelo servidor Iperf3 durante o teste UDP.	69
A.39 Interface de monitoramento do host PC09 observada pelo servidor Iperf3 durante o teste UDP.	69
A.40 Interface de monitoramento do host PC10 observada pelo servidor Iperf3 durante o teste UDP.	70
A.41 Captura de tela das métricas coletadas pelo Zabbix para o host PC01.	70
A.42 Captura de tela das métricas coletadas pelo Zabbix para o host PC02.	71
A.43 Captura de tela das métricas coletadas pelo Zabbix para o host PC03.	71
A.44 Captura de tela das métricas coletadas pelo Zabbix para o host PC04.	72
A.45 Captura de tela das métricas coletadas pelo Zabbix para o host PC05.	72
A.46 Captura de tela das métricas coletadas pelo Zabbix para o host PC06.	73
A.47 Captura de tela das métricas coletadas pelo Zabbix para o host PC07.	73
A.48 Captura de tela das métricas coletadas pelo Zabbix para o host PC08.	74
A.49 Captura de tela das métricas coletadas pelo Zabbix para o host PC09.	74
A.50 Captura de tela das métricas coletadas pelo Zabbix para o host PC10.	75
A.51 Monitoramento do desempenho da interface de rede do host PC01 por meio do Zabbix.	76
A.52 Monitoramento do desempenho da interface de rede do host PC02 por meio do Zabbix.	77
A.53 Monitoramento do desempenho da interface de rede do host PC03 por meio do Zabbix.	78
A.54 Monitoramento do desempenho da interface de rede do host PC04 por meio do Zabbix.	79
A.55 Monitoramento do desempenho da interface de rede do host PC05 por meio do Zabbix.	80
A.56 Monitoramento do desempenho da interface de rede do host PC06 por meio do Zabbix.	81
A.57 Monitoramento do desempenho da interface de rede do host PC07 por meio do Zabbix.	82
A.58 Monitoramento do desempenho da interface de rede do host PC08 por meio do Zabbix.	83
A.59 Monitoramento do desempenho da interface de rede do host PC09 por meio do Zabbix.	84
A.60 Monitoramento do desempenho da interface de rede do host PC10 por meio do Zabbix.	85

Lista de Tabelas

5.1	Distribuição das máquinas virtuais do ambiente experimental	24
5.2	Máquinas reais e virtuais no laboratório TELECOM	25
5.3	Especificações dos computadores físicos utilizados do laboratório	26
5.4	Especificações dos computadores virtuais utilizados no laboratório	26
6.1	Descrição das métricas analisadas nos testes TCP e UDP	38
6.2	Resultados dos testes de desempenho TCP no sentido cliente-servidor	39
6.3	Resultados dos testes de desempenho TCP no sentido servidor-cliente	40
6.4	Resultados dos testes de desempenho UDP no sentido cliente-servidor	41
6.5	Resultados dos testes de desempenho UDP no sentido servidor-cliente	43
6.6	Resumo do tráfego de rede monitorado pelo Zabbix	44
6.7	Resumo do monitoramento de desempenho dos hosts via Zabbix	45
6.8	Comparação dos resultados obtidos com trabalhos relacionados	46

Capítulo 1

Introdução

A crescente complexidade das redes de computadores e a constante evolução dos protocolos de comunicação demandam ambientes de teste que possibilitem a análise de desempenho, validação de soluções e experimentação controlada de diferentes cenários de rede. Entretanto, a utilização de laboratórios físicos tradicionais apresenta limitações significativas, como altos custos de aquisição e manutenção de equipamentos, restrições de espaço físico e dificuldade de reconfiguração das topologias. Além disso, tais ambientes exigem grande esforço operacional e apresentam baixa flexibilidade para experimentação em larga escala [Borges, Cardoso e Bussador 2023].

Por outro lado, simuladores e emuladores de rede amplamente utilizados no meio acadêmico e profissional, embora ofereçam flexibilidade e baixo custo, frequentemente apresentam limitações quanto à fidelidade dos resultados, à integração com equipamentos reais e à reprodutibilidade dos experimentos. Ainda assim, essas ferramentas desempenham papel fundamental na modelagem e análise de redes complexas, permitindo a avaliação de cenários sem interferir diretamente em ambientes reais [Borges, Cardoso e Bussador 2023]. Essas restrições tornam-se um obstáculo para avaliações mais precisas e para a experimentação de cenários próximos aos encontrados em ambientes reais de produção.

Diante desse contexto, surge a motivação para o desenvolvimento de uma plataforma laboratorial do tipo *testbed* (banco de Testes), que integre simulação, emulação e virtualização de redes de computadores em um ambiente único, modular e escalável. Essa abordagem permite a utilização de máquinas virtuais e recursos virtualizados para representar nós de rede reais, possibilitando a execução de experimentos com maior realismo e controle sobre parâmetros como latência, perda de pacotes e taxa de transmissão [Stockmayer 2023].

1.1 Justificativa e Relevância do Trabalho

O desenvolvimento e a expansão das redes de computadores têm desempenhado um papel fundamental na sociedade contemporânea, sustentando serviços essenciais como comunicação digital, computação em nuvem, sistemas distribuídos e aplicações baseadas na Internet.

Nesse contexto, a análise do desempenho, da confiabilidade e da segurança das redes torna-se uma atividade essencial tanto no âmbito acadêmico quanto no ambiente profissional da área de Tecnologia da Informação, conforme discutido em estudos recentes sobre sistemas distribuídos e redes avançadas [Silva *et al.* 2023].

Contudo, a realização de experimentos diretamente em redes operacionais apresenta limitações significativas, como riscos de interrupção de serviços, custos elevados de infraestrutura e dificuldades para reproduzir cenários específicos de testes. Dessa forma, torna-se necessário o uso de ambientes controlados que possibilitem a experimentação de forma segura, flexível e reproduzível, como evidenciado por Corujo *et al.* 2023.

Nesse cenário, os *testbeds* se destacam como plataformas laboratoriais voltadas para a experimentação em redes de computadores, permitindo a simulação e avaliação de diferentes topologias, protocolos e condições de tráfego. Essas plataformas possibilitam a execução de experimentos com maior fidelidade, uma vez que utilizam implementações reais de protocolos e sistemas operacionais, reduzindo a dependência de modelos teóricos simplificados [Silva *et al.* 2023]. Além disso, a simulação de redes permite a criação de cenários complexos e a análise detalhada do comportamento da rede, auxiliando na identificação de gargalos e na otimização de desempenho.

A utilização de um *testbed* para simulação e avaliação de redes de computadores torna possível a construção de um ambiente de experimentação que integra recursos de *hardware* e *software*, permitindo a reprodução de cenários realistas e a análise detalhada de métricas de desempenho, como latência, taxa de transferência, perda de pacotes e estabilidade da comunicação. Dessa forma, tais plataformas tornam-se ferramentas relevantes para o ensino, a pesquisa e o desenvolvimento tecnológico na área de redes, sendo amplamente utilizadas em estudos experimentais e validação de desempenho [Souza *et al.* 2023].

Diante desse contexto, este trabalho se justifica pela necessidade de investigar e apresentar uma plataforma laboratorial que permita a simulação e avaliação de redes de computadores em um ambiente controlado. A relevância desta pesquisa reside na contribuição para o fortalecimento das atividades experimentais no ensino de redes de computadores, bem como no apoio à realização de estudos que envolvam análise de desempenho, testes de protocolos e validação de arquiteturas de rede, promovendo uma maior integração entre teoria e prática no processo de formação acadêmica.

1.2 Objetivos

Esta seção apresenta os objetivos que guiaram a realização deste trabalho. Primeiro, é mostrado o objetivo geral, seguido dos objetivos específicos, que detalham os passos pensados para alcançar o que se propõe.

1.2.1 Objetivo Geral

Avaliar o desempenho de uma rede em um *testbed* virtual por meio da análise de métricas como *throughput*, latência, *jitter*, perda de pacotes e estabilidade da comunicação, visando identificar gargalos relacionados ao congestionamento da rede, ao aumento da latência, à perda de pacotes e à degradação do desempenho durante a transmissão de dados.

1.2.2 Objetivos Específicos

- a) Revisar conceitos de redes, simulação, emulação, protocolos TCP/UDP, métricas de desempenho e ferramentas utilizadas.
- b) Definir o *testbed*, incluindo hardware, software, topologia e máquinas virtuais.
- c) Configurar a infraestrutura virtual, endereçamento IP e cenários de teste.
- d) Coletar métricas de desempenho utilizando *Zabbix*, *iPerf3* e *Grafana*.
- e) Executar testes com os protocolos TCP e UDP em diferentes cenários.
- f) Analisar métricas de desempenho da rede e do servidor, como *throughput*, latência, *jitter*, perda de pacotes, carga do sistema, uso de CPU, memória RAM e memória *swap*.

1.3 Organização do Documento

Este trabalho está estruturado em quatro capítulos, além da seção de referências, organizados da seguinte forma:

- a) Capítulo 1 – Introdução: apresenta o tema da pesquisa, a justificativa e sua relevância, além dos objetivos geral e específicos.
- b) Capítulo 2 – Fundamentação teórica: descreve os principais conceitos relacionados ao tema, abordando redes de computadores, protocolos TCP/UDP e métricas de desempenho como *throughput*, *jitter* e perda de pacotes. Utiliza um *testbed* físico com computadores em rede cabeada e sistema *Ubuntu 24.04 LTS (Long Term Support - Suporte de Longo Prazo)* para simular diferentes níveis de tráfego e concorrência. Ferramentas como *Zabbix*, *Grafana* e *Iperf3* são empregadas para monitoramento e análise do desempenho da rede em condições reais.
- c) Capítulo 3 – Trabalhos Relacionados: O *Iperf3* e o *Zabbix* foram utilizados de forma complementar para a avaliação do desempenho da infraestrutura experimental. O *Iperf3* foi empregado na geração de tráfego TCP e UDP entre as máquinas virtuais do *testbed*, possibilitando a coleta de métricas como *throughput*, *jitter* e perda de pacotes.

Paralelamente, o Zabbix realizou o monitoramento contínuo dos recursos computacionais das máquinas virtuais, coletando informações sobre carga do sistema, utilização da CPU, consumo de memória RAM e uso da memória *swap*. A integração dessas ferramentas permitiu correlacionar o desempenho da rede com a utilização dos recursos computacionais, contribuindo para a validação da plataforma experimental e para a análise do comportamento dos protocolos TCP e UDP em um ambiente virtualizado.

- d) Capítulo 4 – Metodologia: caracteriza-se como experimental, de natureza aplicada e abordagem quantitativa, com foco na avaliação do desempenho de redes em um ambiente controlado por meio de um *testbed* (banco de testes). O estudo envolveu revisão bibliográfica, implementação de uma infraestrutura com recursos físicos e virtuais, configuração de cenários de rede e execução de testes utilizando ferramentas específicas para geração e monitoramento de tráfego.
- e) Capítulo 5 – Desenvolvimento do Ambiente Experimental: O ambiente experimental foi desenvolvido no TELECOM (Laboratório de Telecomunicações), utilizando os computadores disponíveis para a implementação do *testbed* (banco de testes). As máquinas executam o sistema operacional Windows 11 e utilizam o Oracle VM VirtualBox como plataforma de virtualização. Foram criadas máquinas virtuais com Ubuntu Desktop 24.04 *LTS* (*Long Term Support* - Suporte de Longo Prazo) para a execução dos serviços e ferramentas necessárias aos experimentos. Essa infraestrutura permitiu a realização de testes controlados e reproduzíveis, possibilitando a comparação do desempenho de rede entre ambientes físicos e virtualizados sob diferentes condições de operação.
- f) Capítulo 6 – Resultados Obtidos: Os resultados obtidos demonstram que a infraestrutura implementada apresentou desempenho estável e confiável durante a execução dos experimentos. Os testes realizados com os protocolos *TCP* e *UDP* evidenciaram baixas variações nas métricas analisadas, mantendo taxas de transmissão consistentes e ausência de perdas significativas de pacotes. O monitoramento por meio das ferramentas *Zabbix* e *Grafana* permitiu verificar a utilização adequada dos recursos computacionais e da rede, sem ocorrência de sobrecargas ou falhas operacionais. Dessa forma, o ambiente virtualizado mostrou-se adequado para a realização de testes de desempenho, fornecendo resultados reproduzíveis e compatíveis com os objetivos propostos neste trabalho.
- g) Capítulo 7 – Considerações Finais e Sugestões para Trabalhos Futuros: este trabalho demonstrou a eficácia da utilização de um *testbed* (banco de testes) virtual na avaliação de desempenho de redes de computadores, permitindo a criação de cenários controlados, flexíveis e reproduzíveis. A análise das métricas evidenciou diferenças relevantes entre os protocolos TCP/UDP (*Transmission Control Protocol* - Protocolo de Controle de Transmissão / *User Datagram Protocol* - Protocolo de Datagrama de Usuário), destacando a importância da escolha adequada conforme as necessidades das

aplicações. Além disso, o uso integrado de ferramentas de geração, monitoramento e visualização de dados contribuiu para uma análise mais precisa e estruturada.

- h) Capítulo 8 – A. Capturas de Tela dos Experimentos e Métricas Coletadas: Além das análises quantitativas, o trabalho apresenta capturas de tela dos experimentos realizados e das ferramentas empregadas no monitoramento do ambiente experimental. Essas imagens têm como objetivo ilustrar a configuração do *testbed*, a execução dos testes de desempenho e a coleta das métricas de rede e de recursos computacionais. As capturas obtidas por meio das plataformas *Zabbix* e *Grafana* permitem visualizar indicadores como tráfego de rede, utilização de CPU, consumo de memória e comportamento dos protocolos analisados, contribuindo para a compreensão dos resultados e para a validação dos experimentos realizados.

Capítulo 2

Fundamentação Teórica

Esta seção tem como finalidade apresentar os conceitos fundamentais de redes de computadores, responsáveis pela comunicação e transmissão de dados entre dispositivos por meio de protocolos como TCP/IP. O desempenho dessas redes é avaliado a partir de métricas como *throughput*, latência, *jitter* e perda de pacotes, sendo fatores essenciais para garantir confiabilidade, estabilidade e eficiência em aplicações digitais.

Nesse contexto, os *testbeds* de rede se destacam como ambientes experimentais voltados para a análise do comportamento de redes em condições controladas e reproduzíveis. Esses ambientes podem ser implementados por meio de abordagens físicas, simuladas ou virtualizadas, cada uma apresentando vantagens e limitações específicas.

A utilização de virtualização permite a criação de ambientes flexíveis, escaláveis e de menor custo, possibilitando a simulação de múltiplos dispositivos e cenários de rede em uma única infraestrutura física. Dessa forma, torna-se possível realizar experimentos controlados, avaliar protocolos como TCP e UDP sob diferentes condições de tráfego e analisar o desempenho da rede com maior facilidade de reconfiguração.

Além disso, a utilização de ferramentas de monitoramento e teste, como *Zabbix*, *Grafana* e *Iperf3*, permite a coleta e análise detalhada de métricas de desempenho em tempo real [Silva e Silva 2024]. Dessa forma, a adoção de uma infraestrutura baseada em virtualização contribui para a realização de experimentos reproduzíveis, mantendo um equilíbrio entre custo, flexibilidade e representatividade dos resultados.

2.1 Redes de Computadores

Segundo Kurose e Ross 2021, uma rede de computadores é composta por dispositivos interconectados que permitem a comunicação e o compartilhamento de recursos por meio de protocolos padronizados.

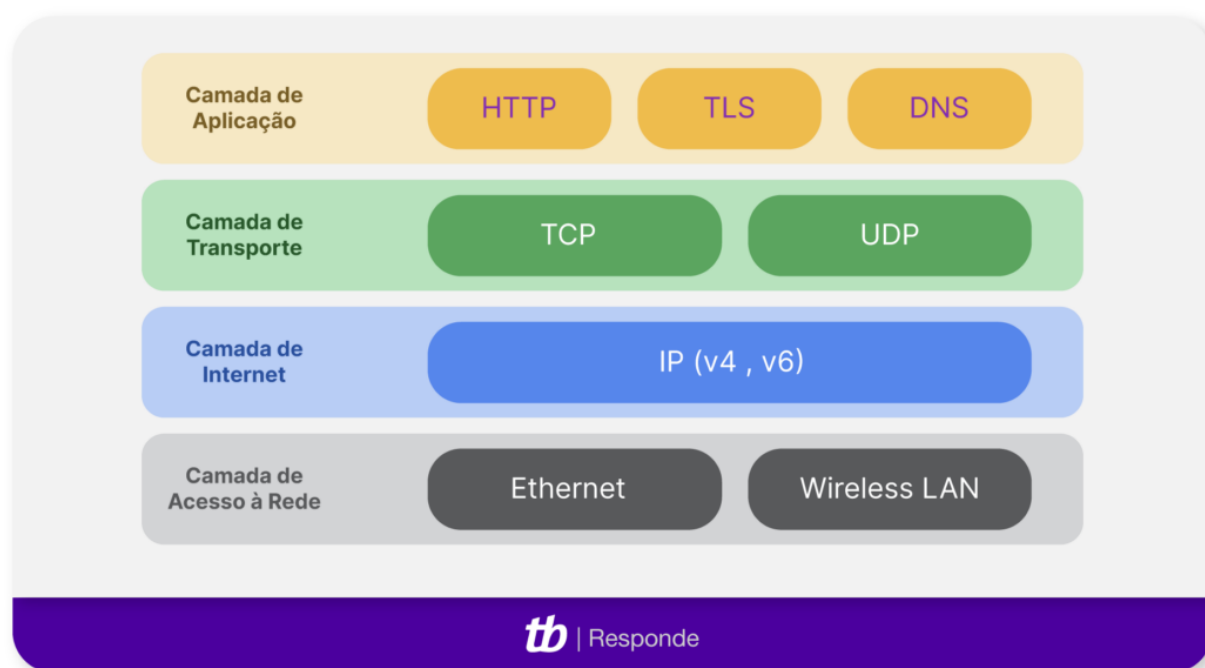
O avanço das redes de computadores tem sido impulsionado pelo crescimento das aplicações digitais e dos serviços em nuvem, consolidando sua importância para sistemas distribuídos e comunicação multimídia em larga escala [Zhang, Liu e Wang 2023]. Nesse contexto,

as redes desempenham um papel fundamental na garantia de conectividade, escalabilidade e desempenho, especialmente em cenários que demandam transmissão contínua de dados, como aplicações de vídeo e sistemas industriais [Moina-Rivera *et al.* 2024].

As redes podem ser classificadas de acordo com sua abrangência geográfica, destacando-se as LAN (*Local Area Network* - Redes Locais), MAN (*Metropolitan Area Network* - Redes Metropolitanas) e WAN (*Wide Area Network* - Redes de Longa Distância). que se diferenciam pela extensão da área coberta e pelos recursos de comunicação empregados [Stallings 2020]. Segundo Forouzan 2021, as redes locais são amplamente utilizadas em ambientes acadêmicos e corporativos devido às suas elevadas taxas de transmissão e baixa latência.

Além disso, a comunicação em redes de computadores é fundamentada em modelos de referência, como o modelo *TCP/IP* que organiza a comunicação em diferentes camadas, permitindo a interoperabilidade entre dispositivos e sistemas heterogêneos. Como ilustrado na Figura 2.1, esse modelo é estruturado em camadas, cada uma responsável por funções específicas no processo de comunicação [Huawei Technologies Co., Ltd. 2022].

Figura 2.1: *Divisão das camadas do modelo TCP/IP.*



Fonte: [Pádua 2023].

Do ponto de vista do meio de transmissão, as redes podem ser classificadas em cabeadas e sem fio. Redes cabeadas, geralmente baseadas em tecnologia *Ethernet*, oferecem maior estabilidade, menor interferência e maior confiabilidade na transmissão de dados. Por outro lado, redes sem fio proporcionam maior mobilidade e flexibilidade, porém podem apresentar limitações relacionadas à interferência e à variação de desempenho [Trabelsi, Fourati e Chen 2024].

Estudos recentes demonstram essas diferenças na prática. Em um ambiente experimental

controlado, foi realizada uma análise comparativa entre redes cabeadas e redes sem fio, considerando métricas como *throughput* e perda de pacotes. Os resultados indicaram que, embora tecnologias sem fio como *Wi-Fi 5* (*Wireless Fidelity - Fidelidade sem fio*) e *Wi-Fi 6* apresentem avanços significativos, as redes cabeadas ainda oferecem maior estabilidade e desempenho consistente, especialmente em cenários com múltiplos dispositivos e altas taxas de transmissão [Merces *et al.* 2026].

No contexto deste trabalho, a compreensão dos fundamentos de redes de computadores é essencial para a análise do desempenho em ambientes experimentais virtualizados, permitindo avaliar o comportamento da comunicação entre dispositivos e interpretar métricas fundamentais de desempenho.

2.2 Redes Virtuais e Virtualização

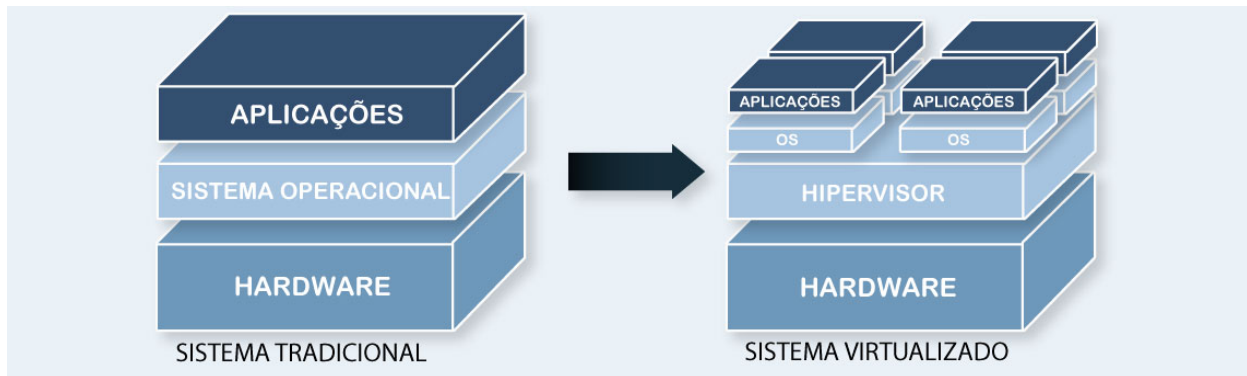
A virtualização é uma tecnologia amplamente utilizada na área de redes de computadores, permitindo a criação de múltiplos ambientes computacionais independentes a partir de um único *hardware* físico [Ahmad, Saeed e Khan 2023]. Essa abordagem possibilita a execução simultânea de diferentes sistemas operacionais e aplicações, promovendo maior flexibilidade, eficiência no uso de recursos e facilidade de gerenciamento [Yadav e Paul 2023].

No contexto de redes de computadores, a virtualização permite a construção de redes virtuais compostas por VMs (*Virtual Machines* - Máquinas Virtuais), que podem se comunicar entre si como se estivessem conectadas em uma infraestrutura física real. Essas redes virtuais são criadas por meio de *softwares* denominados hipervisores (*hypervisors*), responsáveis por abstrair os recursos físicos e distribuí-los entre as VMs [VMware Inc. 2024].

A abstração de recursos computacionais é um dos pilares fundamentais para a evolução das infraestruturas de rede, permitindo maior flexibilidade e independência entre *hardware* e *software* [Tanenbaum e Wetherall 2011]. Nesse contexto, a virtualização se consolida como uma tecnologia essencial para a construção de ambientes de experimentação controlados [Huawei Technologies Co., Ltd. 2022].

Como ilustrado na Figura 2.2, a virtualização permite a inserção de uma camada intermediária, denominada *hypervisor*, entre o *hardware* e os sistemas operacionais, possibilitando a execução simultânea de múltiplas máquinas virtuais de forma isolada [Oracle Corporation 2024].

Figura 2.2: Comparação entre um sistema tradicional e um sistema virtualizado, evidenciando o papel do hypervisor na abstração de recursos.



Fonte: [QNAP Brasil 2023].

Os hipervisores podem ser classificados em dois tipos principais: Tipo 1 (*bare-metal*), que executam diretamente sobre o *hardware*, e Tipo 2 (*hosted*), que operam sobre um sistema operacional hospedeiro. Exemplos de ferramentas amplamente utilizadas incluem o *Proxmox*, que atua como hipervisor do Tipo 1, e o *VirtualBox*, caracterizado como hipervisor do Tipo 2 [Oracle Corporation 2024; Proxmox Server Solutions GmbH 2024].

A utilização de ambientes virtualizados oferece diversas vantagens para experimentação em redes, como a redução de custos com infraestrutura física, facilidade de replicação de cenários, escalabilidade e rápida reconfiguração de topologias. Além disso, permite a simulação de ambientes complexos com múltiplos dispositivos, possibilitando a análise de diferentes condições de tráfego e concorrência [Yadav e Paul 2023].

A virtualização tem se tornado uma abordagem fundamental para experimentação em redes, uma vez que permite a criação de ambientes controlados e reproduzíveis, facilitando a análise de protocolos e o estudo do comportamento da rede em diferentes cenários [Kurose e Ross 2017].

Entretanto, embora a virtualização proporcione grande flexibilidade, ela apresenta limitações em relação a ambientes físicos reais. O desempenho das redes virtuais pode ser influenciado pelo compartilhamento de recursos do *hardware* hospedeiro, como *CPU* (*Central Processing Unit - Unidade Central de Processamento*), memória e interfaces de rede, podendo impactar métricas como *throughput*, latência e perda de pacotes [Morabito, Kjällman e Komu 2023].

Ainda assim, ambientes virtualizados são amplamente utilizados em pesquisas acadêmicas e testes de redes, devido à sua capacidade de oferecer um equilíbrio entre custo, controle experimental e reprodutibilidade dos resultados [Buyya e Srirama 2023].

No contexto deste trabalho, a virtualização é empregada como base para a construção do ambiente experimental, permitindo a criação de um *testbed* virtual no qual diferentes cenários de rede podem ser configurados e avaliados de forma controlada. Essa abordagem possibilita a análise do desempenho da rede sob diferentes condições de operação, contribuindo para a

validação dos experimentos propostos.

2.3 Emulação e Simulação de Redes

A análise e o desenvolvimento de redes de computadores frequentemente requerem o uso de ambientes controlados que permitam a avaliação de desempenho, validação de protocolos e reprodução de cenários específicos. Nesse contexto, destacam-se duas abordagens amplamente utilizadas: a simulação e a emulação de redes [Gomez *et al.* 2023].

A simulação de redes consiste na utilização de modelos matemáticos e computacionais para representar o comportamento de sistemas de rede. Nessa abordagem, não há execução real de protocolos em nível de sistema operacional, sendo o comportamento da rede abstraído por meio de eventos e parâmetros definidos pelo pesquisador. Ferramentas como *NS-3* (*Network Simulator 3* - Simulador de Redes) e *OMNeT* (*Objective Modular Network Testbed* - Plataforma de Testes de Rede Modular Objetiva) são exemplos de simuladores amplamente utilizados na literatura [Henderson *et al.* 2022]. De acordo com [Tanenbaum e Wetherall 2011], a simulação permite a análise de cenários complexos com baixo custo e alta flexibilidade, sendo especialmente útil para estudos teóricos e validação inicial de propostas.

Por outro lado, a emulação de redes busca reproduzir o comportamento de uma rede real de forma mais fiel, permitindo a execução de aplicações e protocolos reais sobre uma infraestrutura controlada. Nesse tipo de abordagem, os sistemas operacionais e aplicações funcionam como em um ambiente real, porém sob condições controladas de rede, como limitação de largura de banda, latência e perda de pacotes [Lantz, Heller e McKeown 2021]. Segundo [Kurose e Ross 2017], a emulação possibilita uma análise mais próxima da realidade, sendo adequada para testes de desempenho e validação de aplicações em condições específicas.

A principal diferença entre simulação e emulação reside no nível de abstração. Enquanto a simulação opera em um nível mais abstrato, focando em modelos e eventos, a emulação atua em um nível mais próximo do real, permitindo a execução de código e protocolos reais. Dessa forma, a emulação tende a oferecer maior fidelidade nos resultados, enquanto a simulação proporciona maior escalabilidade e menor custo computacional [Gomez *et al.* 2023].

Além dessas abordagens, a virtualização surge como uma alternativa intermediária, combinando características de emulação e execução real. Ambientes virtualizados permitem a criação de redes experimentais compostas por máquinas virtuais, nas quais aplicações e protocolos podem ser executados de forma semelhante a um ambiente físico, porém com maior flexibilidade de configuração [Buyya e Srirama 2023; Oracle Corporation 2024].

No contexto deste trabalho, a utilização de técnicas de virtualização se aproxima do conceito de emulação, uma vez que permite a execução de aplicações reais em um ambiente controlado, possibilitando a análise de métricas de desempenho da rede. Dessa forma, a compreensão das diferenças entre simulação, emulação e virtualização é fundamental para a

correta interpretação dos resultados obtidos nos experimentos realizados.

2.4 Protocolos de Transporte: TCP e UDP

Os protocolos de transporte desempenham um papel fundamental na arquitetura de redes de computadores, sendo responsáveis pela comunicação fim a fim entre aplicações distribuídas [Stallings 2020]. No modelo *TCP/IP*, a camada de transporte tem como principal função garantir a entrega de dados entre processos em diferentes dispositivos, oferecendo mecanismos de controle de fluxo, controle de congestionamento e confiabilidade na transmissão [Forouzan 2021].

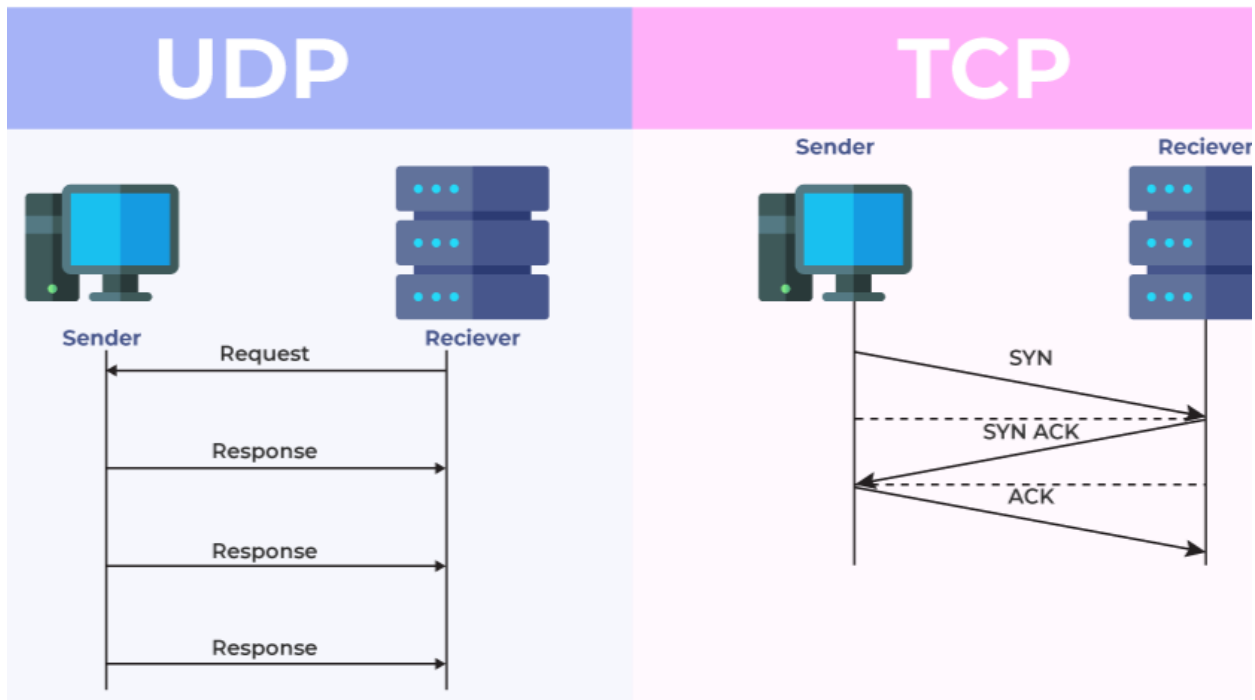
Dentre os principais protocolos dessa camada, destacam-se o (*TCP Transmission Control Protocol* - Protocolo de Controle de Transmissão) e o (*UDP User Datagram Protocol* - Protocolo de Datagrama do Usuário), que apresentam características distintas e são utilizados de acordo com os requisitos da aplicação [Tomar *et al.* 2022].

O *TCP* é um protocolo orientado à conexão, que oferece comunicação confiável entre os dispositivos. Para isso, utiliza mecanismos como estabelecimento de conexão, retransmissão de pacotes perdidos, controle de fluxo e controle de congestionamento. Essas características garantem a integridade e a ordem dos dados transmitidos, tornando o TCP adequado para aplicações que exigem alta confiabilidade, como transferência de arquivos e navegação [Tomar *et al.* 2022]. Segundo [Kurose e Ross 2017], o TCP ajusta dinamicamente a taxa de envio de dados com base nas condições da rede, buscando evitar congestionamentos e perdas excessivas.

Por outro lado, o *UDP* é um protocolo não orientado à conexão, caracterizado por sua simplicidade e baixo *overhead*. Diferentemente do TCP, o UDP não realiza controle de congestionamento nem retransmissão de pacotes, enviando os dados de forma direta e sem garantia de entrega. Essa abordagem resulta em menor latência, sendo adequada para aplicações sensíveis ao tempo, como transmissões de vídeo em tempo real, jogos online e sistemas de comunicação interativa [Hassan e Nabi 2022].

A escolha entre TCP e UDP envolve um compromisso entre confiabilidade e desempenho. Enquanto o TCP prioriza a entrega correta dos dados, mesmo que isso implique maior latência, o UDP prioriza a rapidez na transmissão, aceitando possíveis perdas de pacotes [Nguyen 2026]. De acordo com [Tanenbaum e Wetherall 2011], a utilização de um ou outro protocolo depende diretamente das exigências da aplicação em termos de confiabilidade, atraso e tolerância a perdas.

Como ilustrado na Figura 2.3, o protocolo UDP realiza a transmissão de dados de forma direta, sem a necessidade de estabelecer conexão prévia, enquanto o TCP utiliza um processo de estabelecimento de conexão conhecido como *three-way handshake*, envolvendo as etapas *SYN*, *SYN-ACK* e *ACK*. Esse mecanismo garante maior confiabilidade na comunicação, porém introduz um custo adicional em termos de tempo e overhead [Force 2022].

Figura 2.3: Comparação entre os protocolos UDP e TCP.

Fonte: [Antônio 2024].

No contexto de avaliação de desempenho de redes, a comparação entre TCP e UDP é amplamente utilizada para analisar o comportamento da rede sob diferentes condições de tráfego. Métricas como *throughput*, latência, *jitter* e perda de pacotes são influenciadas diretamente pelas características desses protocolos, sendo fundamentais para a compreensão do desempenho da infraestrutura de rede [Medhi e Ramasamy 2021].

No presente trabalho, ambos os protocolos são utilizados para a geração de tráfego e análise de desempenho no ambiente experimental. A utilização do TCP permite avaliar o comportamento da rede em cenários que exigem confiabilidade, enquanto o uso do UDP possibilita analisar situações em que a baixa latência e a alta taxa de transmissão são priorizadas. Dessa forma, a comparação entre esses protocolos contribui para uma análise mais abrangente do desempenho da rede no *testbed* proposto.

2.5 Métricas de Desempenho de Rede

A avaliação do desempenho de redes de computadores é fundamental para compreender o comportamento da comunicação entre dispositivos e identificar possíveis limitações da infraestrutura. Para isso, são utilizadas métricas específicas que permitem quantificar aspectos relacionados à eficiência, confiabilidade e qualidade da transmissão de dados.

Entre as principais métricas de desempenho, destacam-se o *throughput*, a latência, o *jitter* e a perda de pacotes, amplamente utilizadas em estudos de redes e experimentos de avaliação de desempenho [Kurose e Ross 2017].

O *throughput*, também conhecido como taxa de transferência efetiva, representa a quantidade de dados transmitidos com sucesso em um determinado intervalo de tempo, sendo geralmente expresso em bits por segundo (bps). Essa métrica é essencial para avaliar a capacidade da rede em suportar aplicações que demandam alta taxa de transmissão, como *streaming* de vídeo e transferência de grandes volumes de dados [Medhi e Ramasamy 2021].

A latência refere-se ao tempo necessário para que um pacote de dados seja transmitido de um ponto de origem até o destino. Essa métrica é influenciada por diversos fatores, como distância física, processamento nos dispositivos de rede e congestionamento. Baixos valores de latência são fundamentais para aplicações em tempo real, como videoconferência e jogos online [Olifer e Olifer 2022].

O *jitter* corresponde à variação da latência entre pacotes consecutivos. Em outras palavras, enquanto a latência mede o atraso médio, o *jitter* avalia a inconsistência desse atraso ao longo do tempo. Altos valores de *jitter* podem comprometer a qualidade de aplicações sensíveis ao tempo, como transmissões de áudio e vídeo em tempo real [Olifer e Olifer 2022].

A perda de pacotes ocorre quando um ou mais pacotes de dados não chegam ao destino, podendo ser causada por congestionamento, falhas na rede ou limitações de *hardware*. Essa métrica é particularmente importante em aplicações que exigem confiabilidade, uma vez que perdas frequentes podem degradar significativamente o desempenho da comunicação [Zhu, Chen e Li 2022].

De acordo com [Tanenbaum e Wetherall 2011], a análise conjunta dessas métricas é essencial para uma avaliação completa do desempenho de uma rede, uma vez que cada uma delas representa um aspecto distinto do comportamento da comunicação.

No contexto experimental, essas métricas são amplamente utilizadas para comparar diferentes tecnologias e configurações de rede. Estudos recentes demonstram que redes cabeadas tendem a apresentar maior estabilidade e menor variação de desempenho em comparação com redes sem fio, especialmente em cenários com múltiplos dispositivos e alta demanda de tráfego [Merces *et al.* 2026].

No presente trabalho, as métricas de *throughput*, latência, *jitter* e perda de pacotes são utilizadas como base para a avaliação do desempenho da rede no ambiente virtualizado. A coleta dessas informações é realizada por meio de ferramentas específicas, permitindo a análise detalhada do comportamento da rede sob diferentes condições de carga e concorrência [Ferreira, Silva e Santos 2024].

Durante a execução dos testes de desempenho da rede, foram monitoradas métricas relacionadas aos recursos computacionais dos hosts por meio da plataforma Zabbix. As métricas analisadas incluíram *CPU Usage*, *Memory Usage*, *Swap* e *Linux: System*, as quais permitiram avaliar o comportamento dos sistemas durante a geração e recepção do tráfego de dados.

A métrica *CPU Usage* foi utilizada para medir a porcentagem de utilização do processador em cada computador monitorado. O monitoramento da utilização da CPU é essencial para identificar gargalos de processamento e verificar se os recursos computacionais dis-

poníveis são capazes de suportar adequadamente as cargas de trabalho executadas pelo sistema [Oracle 2024]. Dessa forma, foi possível avaliar o impacto das atividades de rede sobre a capacidade de processamento dos hosts durante os experimentos.

A métrica *Memory Usage* permitiu acompanhar o consumo de memória *RAM* dos sistemas, fornecendo informações sobre a disponibilidade e a utilização dos recursos de memória durante os testes. O monitoramento contínuo da memória constitui um importante indicador do desempenho e da estabilidade dos sistemas Linux, permitindo identificar situações de consumo excessivo e possíveis limitações operacionais [Nagori 2024].

A análise da métrica *Swap* possibilitou identificar a utilização da área de troca do sistema operacional. O mecanismo de *swap* é empregado quando a memória física disponível se torna insuficiente, transferindo temporariamente páginas de memória para o disco. A utilização excessiva desse recurso pode resultar em degradação do desempenho devido ao aumento da latência no acesso aos dados [Tanaka 2005]. Assim, o monitoramento dessa métrica permitiu identificar possíveis cenários de sobrecarga de memória nos hosts monitorados.

Por sua vez, o conjunto de métricas fornecido pelo template *Linux: System* permitiu monitorar aspectos gerais do sistema operacional, incluindo disponibilidade, carga do sistema (*load average*), tempo de atividade (*uptime*) e estado dos serviços monitorados. A análise desses indicadores é fundamental para avaliar a saúde operacional dos sistemas e identificar condições que possam comprometer a disponibilidade e o desempenho dos serviços executados [Oracle 2024].

Dessa forma, o monitoramento dessas métricas por meio do *Zabbix* complementou a análise do desempenho da infraestrutura de rede, possibilitando correlacionar a utilização dos recursos computacionais com o comportamento do tráfego gerado durante os experimentos. Essa abordagem proporcionou uma avaliação mais abrangente da capacidade operacional, da estabilidade e da eficiência dos sistemas monitorados, contribuindo para uma análise mais detalhada do ambiente experimental.

2.6 *Testbeds* Virtuais em Redes de Computadores

Testbeds são ambientes experimentais utilizados para a análise, validação e avaliação de desempenho de sistemas em condições controladas. No contexto de redes de computadores, esses ambientes permitem a criação de cenários específicos para o estudo do comportamento de protocolos, análise de tráfego e identificação de gargalos em diferentes configurações de rede [Vandenberghe *et al.* 2021].

Tradicionalmente, *testbeds* podem ser implementados utilizando infraestrutura física dedicada, composta por dispositivos reais interconectados. No entanto, essa abordagem pode apresentar limitações relacionadas a custos, escalabilidade, manutenção e dificuldade de reconfiguração. Nesse contexto, surgem os *testbeds* virtuais como uma alternativa eficiente e amplamente adotada em ambientes acadêmicos e de pesquisa [Czajkowski *et al.* 2022].

Os *testbeds* virtuais são baseados em tecnologias de virtualização, permitindo a criação de redes experimentais compostas por VMs (*Virtual Machines* - Máquinas Virtuais), interconectadas por meio de redes virtuais configuráveis. Essa abordagem possibilita a simulação de diferentes topologias e condições de tráfego em um único ambiente físico, reduzindo significativamente os custos e aumentando a flexibilidade dos experimentos [Wang, Li e Zhang 2022].

De acordo com [Kurose e Ross 2017], ambientes experimentais controlados são essenciais para a análise de desempenho em redes, pois permitem a reprodução de cenários e a comparação de resultados de forma consistente. Nesse sentido, a virtualização contribui diretamente para a construção de *testbeds* mais acessíveis e reprodutíveis.

Além disso, estudos recentes demonstram a relevância do uso de *testbeds* para avaliação de redes em cenários reais e controlados. Em [Merces *et al.* 2026], por exemplo, foi desenvolvido um ambiente experimental para comparar o desempenho de redes cabeadas e sem fio, utilizando métricas como *throughput* e perda de pacotes. Os resultados evidenciaram a importância de ambientes controlados para a análise precisa do comportamento da rede.

Os *testbeds* virtuais oferecem diversas vantagens, como a facilidade de replicação de experimentos, rápida reconfiguração de cenários, escalabilidade e melhor aproveitamento dos recursos computacionais. Além disso, permitem a integração com ferramentas de monitoramento e análise, como *Zabbix*, *Grafana* e *Iperf3*, possibilitando a coleta de métricas em tempo real.

Entretanto, é importante destacar que, embora os *testbeds* virtuais proporcionem grande flexibilidade, eles podem apresentar limitações em relação à fidelidade dos resultados, uma vez que o desempenho pode ser influenciado pelo compartilhamento de recursos do hardware hospedeiro. Dessa forma, os resultados obtidos devem ser analisados considerando essas características.

No contexto deste trabalho, o *testbed* virtual é utilizado como base para a realização dos experimentos de avaliação de desempenho da rede. A partir da criação de múltiplas máquinas virtuais interconectadas, são simulados diferentes cenários de comunicação, permitindo a análise do comportamento dos protocolos *TCP* e *UDP* sob diferentes condições de tráfego e concorrência. Essa abordagem possibilita a obtenção de resultados consistentes e reprodutíveis, contribuindo para a validação dos objetivos propostos.

2.7 Ferramentas para Geração de Tráfego e Monitoramento de Redes

A avaliação do desempenho de redes de computadores requer a utilização de ferramentas capazes de gerar tráfego controlado e monitorar, de forma contínua, os recursos computacionais envolvidos nos experimentos. Nesse contexto, destacam-se o *Iperf3* e o *Zabbix*, amplamente utilizados em ambientes acadêmicos e profissionais para análise de desempenho de redes e

sistemas.

O Iperf3 é uma ferramenta de código aberto desenvolvida para medir o desempenho de redes por meio da geração de tráfego entre dispositivos conectados. Sua principal finalidade é avaliar parâmetros relacionados à comunicação de dados, como throughput, largura de banda disponível, jitter e perda de pacotes. A ferramenta suporta testes utilizando os protocolos TCP e UDP, permitindo analisar o comportamento da rede sob diferentes condições de carga e padrões de transmissão. Devido à sua precisão e flexibilidade, o Iperf3 tornou-se uma das soluções mais empregadas em pesquisas voltadas à avaliação de desempenho de redes de computadores [Zieliński 2023].

Segundo Zieliński 2023, o *Iperf3* destaca-se como uma ferramenta de referência para medições de throughput e análise de desempenho em redes de computadores, oferecendo suporte à avaliação de métricas relevantes para a caracterização da qualidade da comunicação entre dispositivos.

Enquanto o Iperf3 é responsável pela geração e medição do tráfego, o Zabbix atua como uma plataforma de monitoramento capaz de coletar, armazenar e analisar métricas de sistemas e redes em tempo real. A ferramenta permite acompanhar o estado operacional dos equipamentos e servidores, fornecendo informações detalhadas sobre a utilização de recursos computacionais e auxiliando na identificação de possíveis gargalos de desempenho.

Entre as principais métricas monitoradas em sistemas Linux destacam-se o *System Load*, que representa a carga de processamento do sistema; o *CPU Usage*, responsável por indicar o percentual de utilização do processador; o *Memory Usage*, que monitora o consumo de memória RAM; e o *Swap Usage*, utilizado para acompanhar a utilização da memória de troca. A análise conjunta dessas métricas possibilita compreender o impacto das aplicações e do tráfego de rede sobre os recursos computacionais disponíveis.

A utilização integrada de ferramentas de geração de tráfego e monitoramento permite uma análise mais abrangente do desempenho de ambientes computacionais. Enquanto o Iperf3 fornece informações relacionadas à comunicação de dados e à eficiência dos protocolos de transporte, o Zabbix possibilita a observação do comportamento dos recursos do sistema durante a execução dos testes. Dessa forma, torna-se possível correlacionar métricas de rede com métricas de infraestrutura, contribuindo para uma avaliação mais precisa e confiável do ambiente experimental.

Em estudos que envolvem ambientes virtualizados, essas ferramentas assumem papel fundamental, uma vez que permitem verificar não apenas o desempenho da comunicação entre máquinas virtuais, mas também os impactos da virtualização sobre o consumo de recursos computacionais. Assim, sua aplicação fornece subsídios importantes para a análise e validação de plataformas experimentais voltadas à pesquisa em redes de computadores.

Capítulo 3

Trabalhos Relacionados

3.1 Trabalhos Relacionados

A avaliação de desempenho em redes de computadores tem sido amplamente explorada na literatura, especialmente em estudos que utilizam ambientes virtualizados e ferramentas especializadas para geração de tráfego e monitoramento de recursos computacionais. De acordo com Zabbix LLC 2024, o uso de plataformas experimentais controladas possibilita a realização de testes reproduzíveis e a obtenção de métricas confiáveis para análise do comportamento da rede.

O avanço das tecnologias de virtualização permitiu o desenvolvimento de ambientes experimentais flexíveis e de baixo custo, conhecidos como *testbeds*, que vêm sendo amplamente empregados em pesquisas relacionadas ao desempenho de protocolos e serviços de rede. Conforme destacado por Silva e Silva 2024, a virtualização reduz custos de implementação e facilita a replicação de cenários experimentais complexos, tornando-se uma alternativa viável para instituições acadêmicas e centros de pesquisa.

Além disso, diversos estudos têm investigado o comportamento dos protocolos TCP e UDP em diferentes condições de tráfego, avaliando métricas como throughput, jitter, latência e perda de pacotes. Segundo Canonical Ltd. 2024, a análise dessas métricas é fundamental para compreender o impacto das características dos protocolos de transporte no desempenho das aplicações e serviços de rede.

Nesse contexto, os trabalhos relacionados apresentados nesta seção abordam diferentes estratégias para implementação de ambientes de testes, monitoramento de recursos computacionais e avaliação de desempenho de redes. A análise desses estudos permite identificar suas principais contribuições e limitações, servindo como base para o desenvolvimento da proposta apresentada neste trabalho.

Por fim, a comparação entre os trabalhos analisados possibilita evidenciar as contribuições desta pesquisa, que propõe a construção de um testbed virtual para avaliação dos protocolos TCP e UDP, utilizando ferramentas de geração de tráfego e monitoramento em um ambiente controlado e reproduzível.

Capítulo 4

Metodologia

A metodologia adotada neste trabalho caracteriza-se como experimental, de natureza aplicada e abordagem quantitativa, tendo como objetivo avaliar o desempenho de redes de computadores em um ambiente controlado por meio de um *testbed*. Inicialmente, foram estudados os conceitos relacionados a redes de computadores, virtualização e protocolos de transporte, com ênfase nos protocolos *TCP* e *UDP*, bem como as principais métricas de desempenho de rede, tais como *throughput*, latência, *jitter* e perda de pacotes.

4.1 Materiais e Métodos

Nos materiais e métodos deste trabalho, adicionalmente, foram utilizadas ferramentas de IA (*Artificial Intelligence - Inteligência Artificial*) como apoio na organização textual, revisão gramatical e estruturação do trabalho acadêmico, contribuindo para maior clareza, coesão e padronização da escrita, sem interferir na análise dos dados ou nos resultados obtidos.

Em seguida, foi projetado e implementado o ambiente experimental (*testbed*), composto por uma infraestrutura física integrada a recursos de virtualização. Para isso, foram utilizados computadores interconectados em rede cabeada, executando o sistema operacional *Ubuntu 24.04 LTS* (*Long Term Support* - Suporte de Longo Prazo) [Canonical Ltd. 2024], além da criação de máquinas virtuais por meio de ferramentas como *VirtualBox* permitindo a simulação de diferentes cenários de rede. Posteriormente, foi realizada a configuração da topologia de rede no ambiente virtual, contemplando a criação e interconexão das máquinas virtuais, a atribuição de endereços *IP* às interfaces de rede e a definição das rotas necessárias para o estabelecimento da comunicação entre os dispositivos. Também foram configurados os adaptadores de rede do ambiente de virtualização, de modo a reproduzir as condições necessárias para a execução dos experimentos. Após a configuração da infraestrutura, foram realizados testes de conectividade utilizando ferramentas de diagnóstico de rede, como o comando *ping*, com o objetivo de verificar a comunicação entre os nós da rede e validar o correto funcionamento da topologia implementada. Essa etapa foi essencial para assegurar a confiabilidade do ambiente experimental antes da realização dos testes de desempenho. Além

disso, foram elaborados diferentes cenários experimentais, variando a quantidade de tráfego gerado e o número de conexões simultâneas estabelecidas entre os hosts. Essas configurações tiveram como objetivo reproduzir condições de operação semelhantes às encontradas em ambientes reais, possibilitando a análise do comportamento da rede sob diferentes níveis de carga e demanda por recursos.

Para a execução dos experimentos, foi utilizada a ferramenta *Iperf3* para a geração e análise de tráfego de rede entre as máquinas virtuais do ambiente experimental. A ferramenta foi configurada em um modelo cliente-servidor, no qual uma máquina atuou como servidor, permanecendo em escuta para receber conexões, enquanto outra atuou como cliente, responsável pela geração do tráfego de dados. Os testes foram realizados utilizando os protocolos *TCP* e *UDP*, permitindo a comparação do desempenho da rede sob diferentes características de comunicação. Para cada cenário, foram definidas taxas de transmissão específicas e diferentes quantidades de conexões simultâneas, com o objetivo de simular condições reais de utilização da rede. Durante a execução dos testes, o *Iperf3* registrou métricas de desempenho como throughput, jitter e perda de pacotes, possibilitando uma análise detalhada do comportamento da infraestrutura de rede.

Os resultados obtidos foram coletados e posteriormente analisados, permitindo avaliar o impacto dos protocolos de transporte e das diferentes cargas de tráfego sobre o desempenho da rede virtual implementada. A coleta de métricas foi realizada com o auxílio do *Zabbix*, enquanto o *Grafana* foi empregado para visualização e análise dos dados obtidos.

Os experimentos foram conduzidos sob diferentes condições de carga e concorrência, permitindo a mensuração e análise das métricas de desempenho. Por fim, os dados coletados foram analisados de forma quantitativa, possibilitando a identificação de padrões de comportamento, variações de desempenho e possíveis gargalos na rede.

4.2 Abordagem de Pesquisa Quantitativa

Quanto à abordagem metodológica, a pesquisa classifica-se como quantitativa, pois baseia-se na coleta, mensuração e análise de dados numéricos obtidos durante os experimentos realizados no ambiente testbed. Essa abordagem permite avaliar de forma objetiva o desempenho da rede por meio de indicadores mensuráveis, fornecendo resultados técnicos mais precisos e confiáveis.

Durante a execução dos testes, foram analisadas métricas relevantes, como latência, *jitter*, *throughput*, perda de pacotes e consumo de recursos computacionais, tais como processamento e memória. A partir desses dados, tornou-se possível comparar diferentes cenários de rede, identificar variações de desempenho e verificar a estabilidade da infraestrutura proposta. A utilização da abordagem quantitativa mostra-se adequada ao tema deste trabalho, pois possibilita a interpretação dos resultados com base em valores concretos, facilitando a elaboração de conclusões fundamentadas e a validação técnica da plataforma desenvolvida.

4.3 Natureza Aplicada da Pesquisa

A presente pesquisa caracteriza-se como de natureza aplicada, uma vez que tem como objetivo utilizar conhecimentos científicos e tecnológicos para a solução de problemas práticos relacionados à área de redes de computadores. Diferentemente das pesquisas voltadas apenas à ampliação teórica do conhecimento, a pesquisa aplicada busca gerar resultados concretos e úteis para situações reais. Nesse sentido, este trabalho concentra-se no desenvolvimento e na utilização de uma plataforma *testbed* destinada à simulação, experimentação e avaliação de cenários de rede em ambiente controlado.

A aplicação prática do estudo permite analisar o comportamento de infraestruturas computacionais, validar configurações de rede, testar protocolos e identificar possíveis limitações operacionais. Além disso, a utilização de um *testbed* contribui para a redução de custos e riscos associados à implementação direta em ambientes produtivos, tornando-se uma alternativa relevante para pesquisas acadêmicas, capacitação técnica e aperfeiçoamento de soluções em redes de computadores.

Capítulo 5

Desenvolvimento do Ambiente Experimental

5.1 Infraestrutura Física do Laboratório

O ambiente experimental foi desenvolvido utilizando os computadores disponíveis no laboratório de sistemas de telecomunicações do IFPB, os quais serviram como plataforma física para a execução das máquinas virtuais que compõem o *testbed*. Cada computador executa o sistema operacional *Windows 11*, sobre o qual foi instalado o *software* de virtualização *Oracle VM VirtualBox*. Por meio dessa ferramenta, foram criadas máquinas virtuais com o sistema operacional *Ubuntu Desktop 24.04 LTS*, utilizadas para representar os diferentes nós da rede e executar os serviços necessários aos experimentos.

Os experimentos desenvolvidos neste trabalho foram realizados no Laboratório de Telecomunicações (TELECOM), ambiente destinado ao ensino, pesquisa e desenvolvimento de atividades práticas relacionadas às áreas de redes de computadores, telecomunicações e sistemas computacionais. O laboratório possui infraestrutura adequada para a execução de experimentos acadêmicos, disponibilizando recursos computacionais e de rede que permitem a realização de análises de desempenho e monitoramento de sistemas.

Durante a execução da pesquisa, o TELECOM foi utilizado para a configuração do ambiente experimental, implementação dos serviços necessários e coleta das métricas de desempenho analisadas neste estudo.

Na figura 5.1 é apresentada diferentes perspectivas do Laboratório de Telecomunicações (TELECOM). As imagens evidenciam a organização do ambiente, a disposição das bancadas de trabalho e a infraestrutura utilizada para a realização das atividades práticas e experimentais desenvolvidas ao longo desta pesquisa.

Figura 5.1: *Diferentes perspectivas do Laboratório de Telecomunicações (TELECOM), ambiente utilizado para a realização dos experimentos deste trabalho.*



(a) *Vista frontal do Laboratório de Telecomunicações (TELECOM).*



(b) *Vista lateral do Laboratório de Telecomunicações (TELECOM).*

Fonte: Elaborado pelo autor (2026).

A utilização do Laboratório de Telecomunicações (TELECOM) foi fundamental para o desenvolvimento da pesquisa, fornecendo a infraestrutura necessária para a execução dos experimentos, coleta de dados e validação dos resultados obtidos. O ambiente permitiu a realização das atividades de monitoramento e análise de desempenho em condições adequadas, contribuindo para a confiabilidade dos resultados apresentados neste trabalho. A utilização do laboratório foi fundamental para o desenvolvimento da pesquisa, fornecendo a infraestrutura necessária para a execução dos experimentos, coleta de dados e validação dos resultados obtidos. O ambiente permitiu a realização das atividades de monitoramento e análise de desempenho em condições adequadas, contribuindo para a confiabilidade dos resultados apresentados neste trabalho.

A utilização dessa infraestrutura permitiu o aproveitamento dos recursos computacionais já disponíveis no laboratório, reduzindo custos e facilitando a replicação do ambiente em outros contextos acadêmicos. Além disso, a combinação entre *Windows 11* como sistema operacional hospedeiro, *VirtualBox* como hipervisor do tipo 2 (*hosted*) e *Ubuntu Desktop 24.04 LTS* como sistema operacional convidado proporcionou um ambiente estável e flexível para a realização dos testes de desempenho.

As principais características dos computadores físicos utilizados no laboratório são apre-

sentadas a seguir:

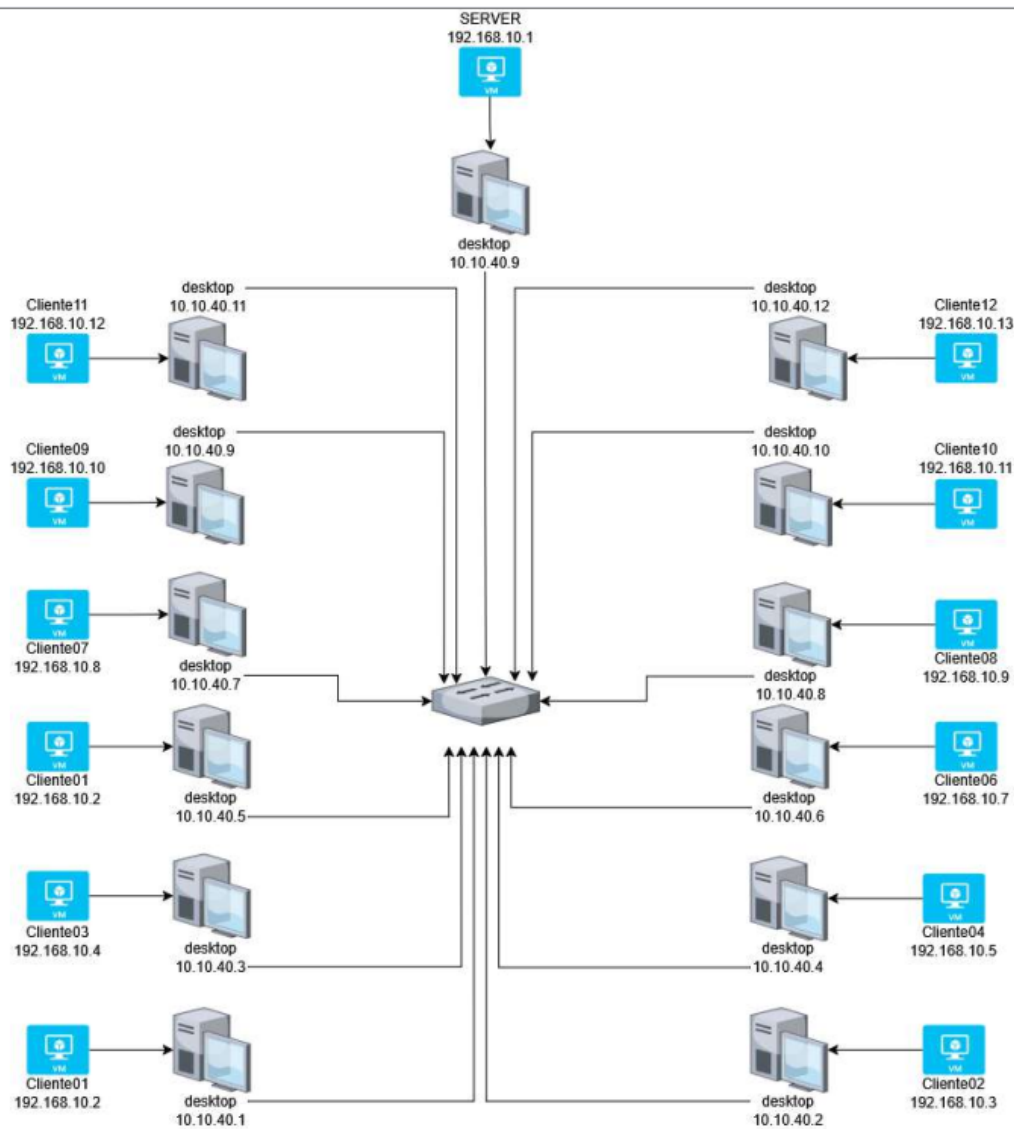
- a) **Fabricante:** *DATEN*;
- b) **Sistema Operacional Hospedeiro:** *Windows 11*;
- c) **Software de Virtualização:** *Oracle VM VirtualBox*;
- d) **Sistema Operacional das Máquinas Virtuais:** *Ubuntu Desktop 24.04 LTS*;
- e) **Processador:** *Intel Core i3* de 8^a geração, com frequência de 3,60 GHz;
- f) **Memória RAM:** 8 GB DDR4;
- g) **Armazenamento:** Disco rígido (*Hard Disk Drive* – HDD) Western Digital de 500 GB;
- h) **Interface de Rede:** Adaptador *Gigabit Ethernet TP-Link PCI Express*.

5.2 Topologia de Rede

As máquinas virtuais clientes foram configuradas com apenas uma interface conectada à rede interna do VirtualBox, permitindo a comunicação com o servidor e garantindo que todo o tráfego gerado durante os testes permanecesse isolado da infraestrutura física do laboratório. Essa abordagem contribui para a reprodutibilidade dos experimentos e reduz interferências externas durante a coleta das métricas de desempenho.

A Figura 5.2 é demonstrada a arquitetura lógica do ambiente experimental implementado, destacando a interligação entre a máquina virtual servidora, as máquinas virtuais clientes e as diferentes interfaces de rede utilizadas.

Figura 5.2: Topologia da rede utilizada nos testes.



Fonte: Elaborado pelo autor (2026).

No Quadro 5.1 é apresentada a distribuição das máquinas virtuais utilizadas no ambiente experimental.

Quadro 5.1: Distribuição das máquinas virtuais do ambiente experimental

Função	Quantidade	Sistema Operacional	Interfaces de Rede
Servidor	1	Ubuntu Desktop 24.04 LTS	2
Clientes	10	Ubuntu Desktop 24.04 LTS	1
Total	11	—	—

Fonte: Elaborado pelo autor (2026).

Com as máquinas virtuais configuradas e interconectadas, foram instaladas as ferramentas necessárias para a realização dos experimentos. O *Iperf3* foi utilizado para geração de

tráfego e medição de desempenho da rede, o *Zabbix* para monitoramento dos recursos computacionais e coleta de métricas, e o *Grafana* para visualização e análise dos dados obtidos durante os testes.

Ao final dessa etapa, obteve-se um ambiente experimental composto por um servidor virtual e dez clientes virtuais distribuídos em computadores físicos independentes, permitindo a realização de experimentos controlados para avaliação do desempenho da rede sob diferentes condições de tráfego e concorrência.

Conforme apresentado na 5.2, cada máquina do ambiente experimental foi associada a um endereço *IP* específico, permitindo sua identificação e monitoramento individual. Essa organização possibilitou o acompanhamento centralizado dos dispositivos participantes dos testes, facilitando a coleta e a análise das métricas de desempenho durante os experimentos.

Quadro 5.2: *Máquinas reais e virtuais no laboratório TELECOM*

Ambiente Real	Ambiente Virtualizado
Servidor	192.168.10.1
Computador 01	192.168.10.2
Computador 02	192.168.10.3
Computador 03	192.168.10.4
Computador 04	192.168.10.5
Computador 05	192.168.10.7
Computador 06	192.168.10.8
Computador 07	192.168.10.9
Computador 08	192.168.10.10
Computador 09	192.168.10.11
Computador 10	192.168.10.12

Fonte: Elaborado pelo autor (2026).

5.3 Configuração do Ambiente Virtualizado

Após a definição da infraestrutura física do laboratório, foi realizada a configuração do ambiente virtualizado responsável pela implementação do *testbed*. Para isso, utilizou-se o software Oracle VM *VirtualBox* [Oracle Corporation 2025], instalado nos computadores do laboratório e executado sobre o sistema operacional Windows 11. Essa ferramenta permitiu a criação e o gerenciamento das máquinas virtuais utilizadas nos experimentos.

As especificações dos computadores físicos e das máquinas virtuais utilizadas no ambiente experimental são apresentadas de forma resumida na Tabela 5.3.

Quadro 5.3: *Especificações dos computadores físicos utilizados do laboratório*

Componente	Especificações
Fabricante	DATEN
Sistema Operacional Hospedeiro	<i>Windows 11</i>
Software de Virtualização	<i>Oracle VM VirtualBox</i>
Memória Principal (RAM)	8 GB DDR4
Processador	<i>Intel Core i3</i> (8 ^a geração) 3,60 GHz
Armazenamento	<i>HDD Western Digital 500 GB</i>

Fonte: Elaborado pelo autor (2026).

Quadro 5.4: *Especificações dos computadores virtuais utilizados no laboratório*

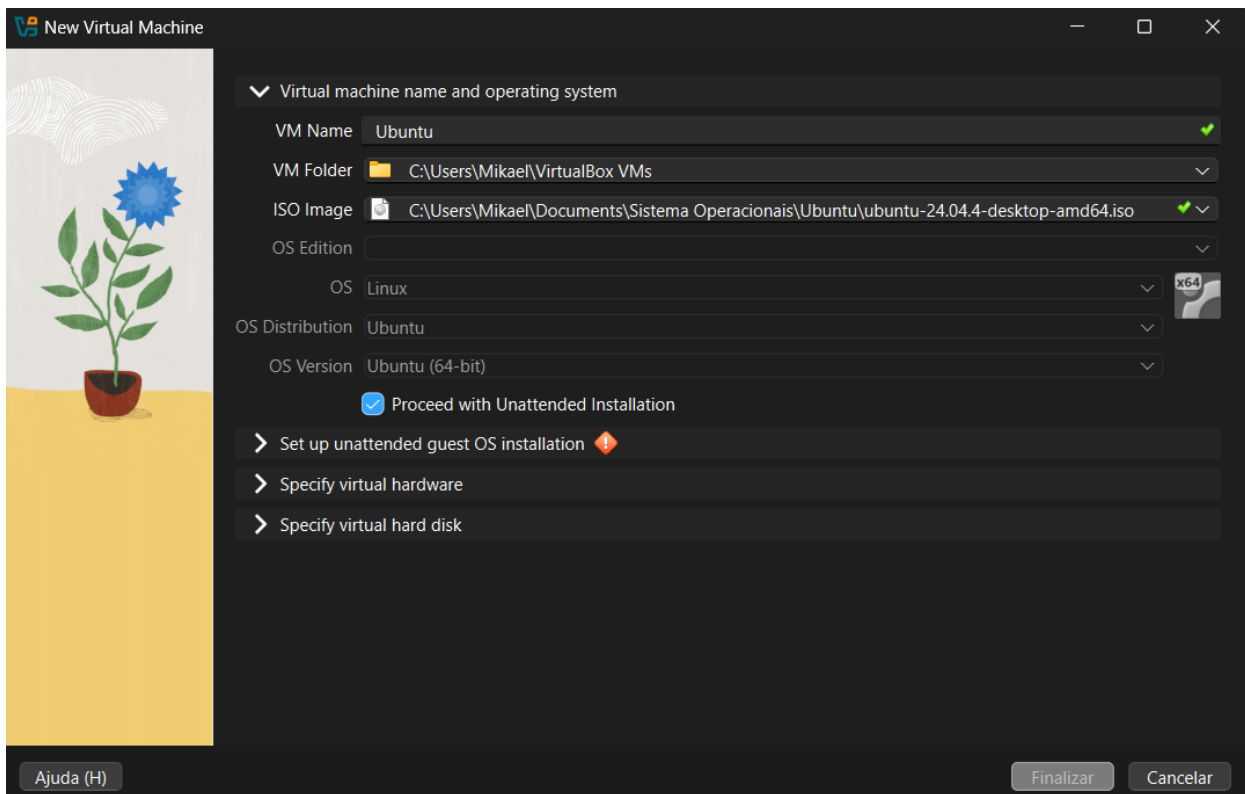
Componente	Especificação
Interface de Rede	Adaptador <i>TP-Link Gigabit PCI Express</i>
Sistema Operacional das VMs	<i>Ubuntu Desktop 24.04 LTS</i>
CPU	2 vCPUs (2 Cores)
Memória RAM	4 GB DDR4
Armazenamento	<i>HDD Western Digital 100 GB</i>
Modo de Rede	<i>Bridge</i>
Máquinas Virtuais	11 VMs

Fonte: Elaborado pelo autor (2026).

O ambiente experimental foi composto por 11 computadores físicos do laboratório, nos quais foram implantadas 11 máquinas virtuais executando o sistema operacional *Ubuntu Desktop 24.04 LTS*. Dentre essas máquinas virtuais, uma foi configurada para atuar como servidor central do ambiente experimental, enquanto as demais desempenharam o papel de clientes responsáveis pela geração de tráfego e execução dos testes de desempenho.

Inicialmente, foram criadas as máquinas virtuais por meio do *VirtualBox*, sendo definidos parâmetros como nome da máquina, sistema operacional, imagem ISO de instalação, quantidade de memória *RAM*, número de processadores virtuais e capacidade de armazenamento. A Figura 5.3 apresenta a interface utilizada durante o processo de criação das máquinas virtuais.

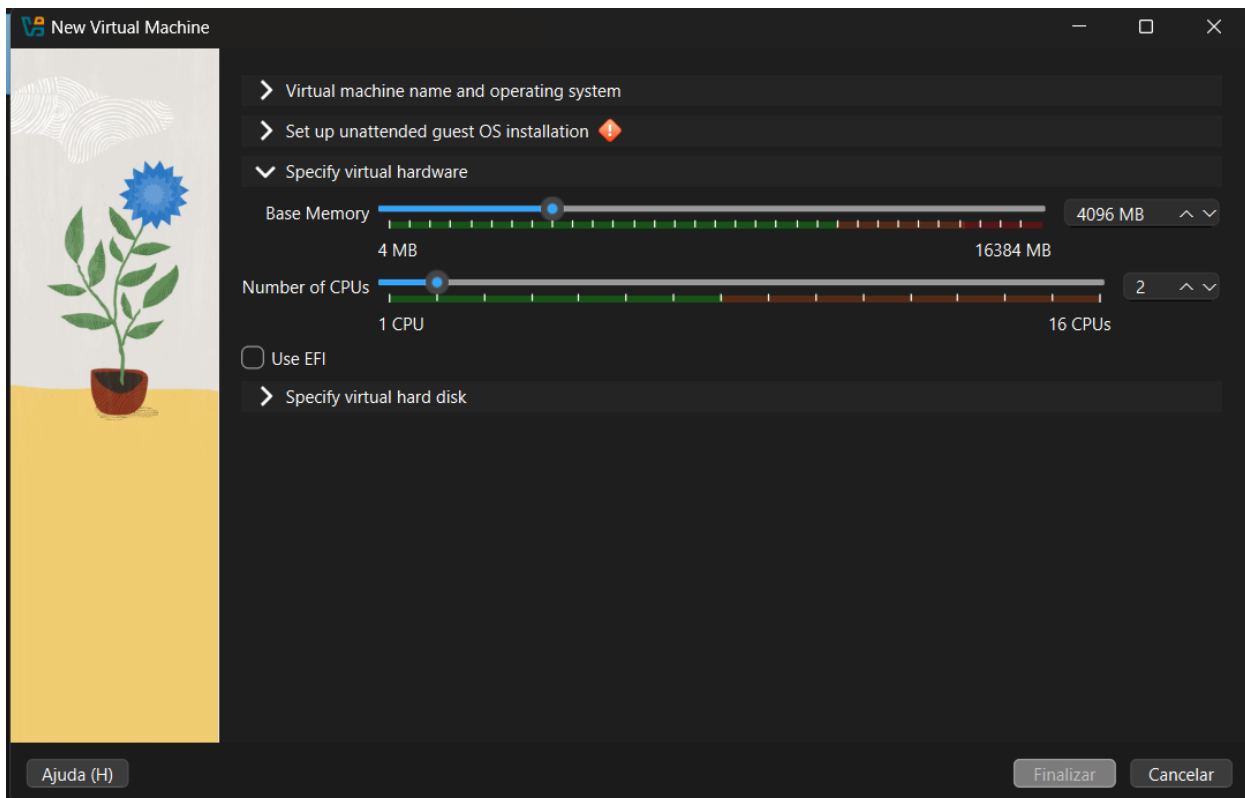
Figura 5.3: Processo de criação de uma máquina virtual Ubuntu Desktop 24.04 LTS no VirtualBox.



Fonte: Elaborado pelo autor (2026).

Após a criação das máquinas virtuais, foram configurados os recursos computacionais necessários para sua execução. Cada máquina virtual foi configurada com 4 GB de memória *RAM* e 2 processadores virtuais, valores considerados suficientes para a execução do sistema operacional e das ferramentas utilizadas nos experimentos. A Figura 5.4 apresenta a configuração da memória *RAM* e dos processadores virtuais atribuídos às máquinas virtuais.

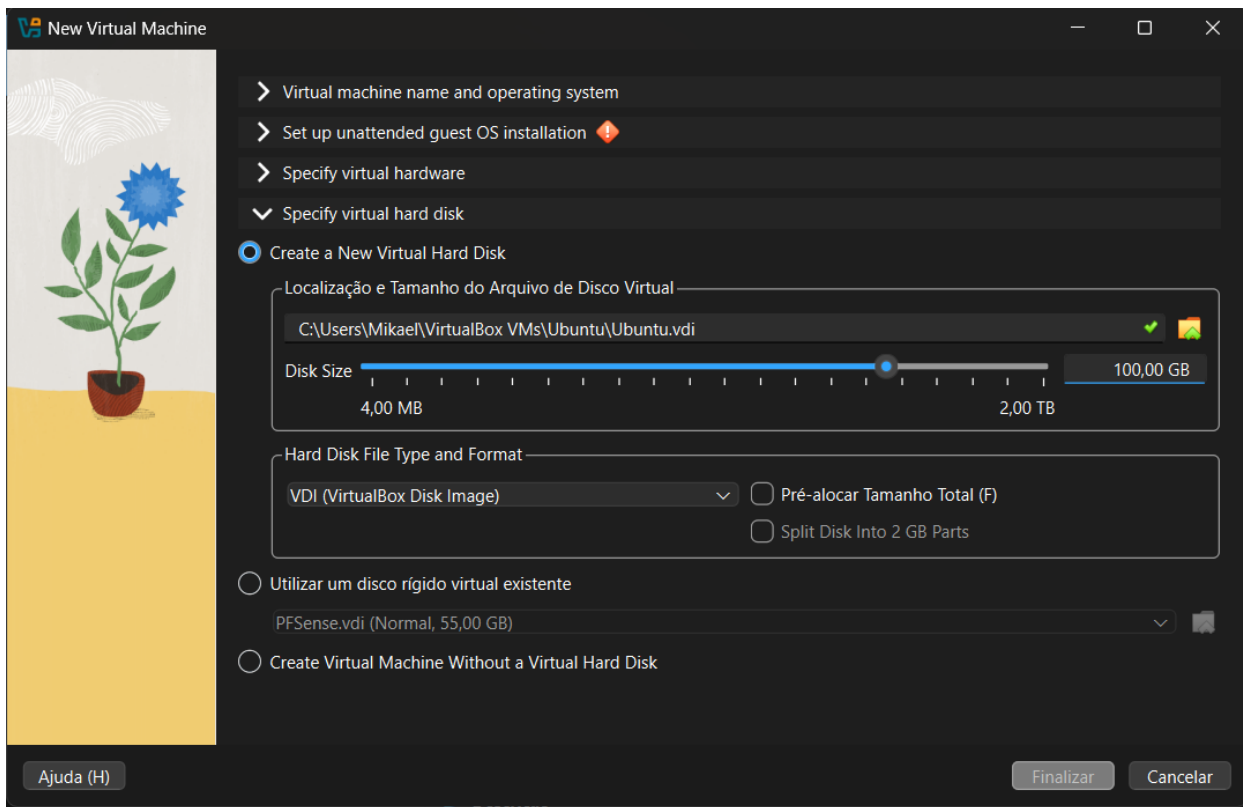
Figura 5.4: *Configuração de memória RAM e processadores virtuais das máquinas virtuais.*



Fonte: Elaborado pelo autor (2026).

Além dos recursos de processamento e memória, foi configurado o armazenamento virtual utilizado pelas máquinas virtuais. Para os experimentos realizados, foi adotado um disco virtual no formato VDI (*Virtual Disk Image*), conforme apresentado na Figura 5.5.

Figura 5.5: *Configuração do armazenamento virtual utilizado pelas máquinas virtuais.*

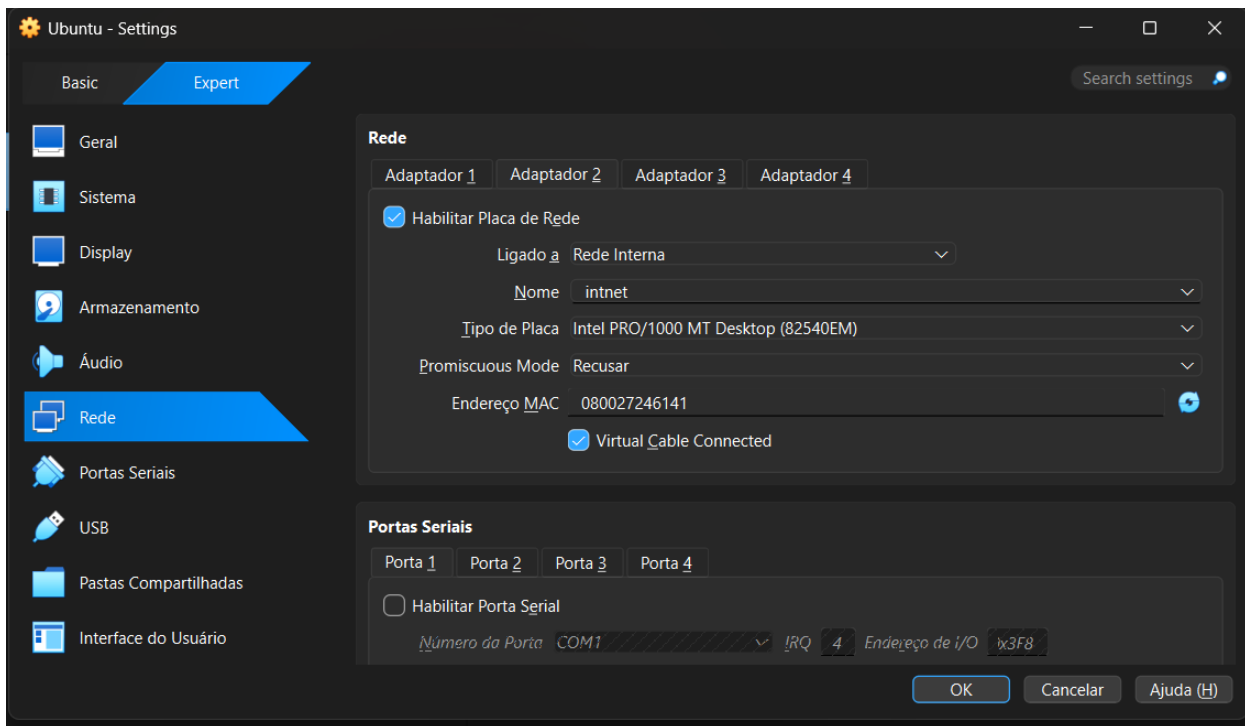


Fonte: Elaborado pelo autor (2026).

Em seguida, foram realizadas as configurações de rede necessárias para a comunicação entre os componentes do *testbed*. A máquina virtual responsável pelo servidor foi configurada com duas interfaces de rede. A primeira interface foi conectada à rede interna do *VirtualBox* (*Internal Network*), sendo utilizada para a comunicação com os clientes durante a execução dos experimentos. A segunda interface foi configurada em modo *Bridge*, permitindo o gerenciamento da máquina virtual a partir do sistema operacional hospedeiro sem interferir no tráfego experimental.

A Figura 5.6 apresenta a configuração da interface conectada à rede interna utilizada para a comunicação entre servidor e clientes.

Figura 5.6: Configuração da interface conectada à rede interna do ambiente experimental.



Fonte: Elaborado pelo autor (2026).

5.4 Ferramentas Utilizadas

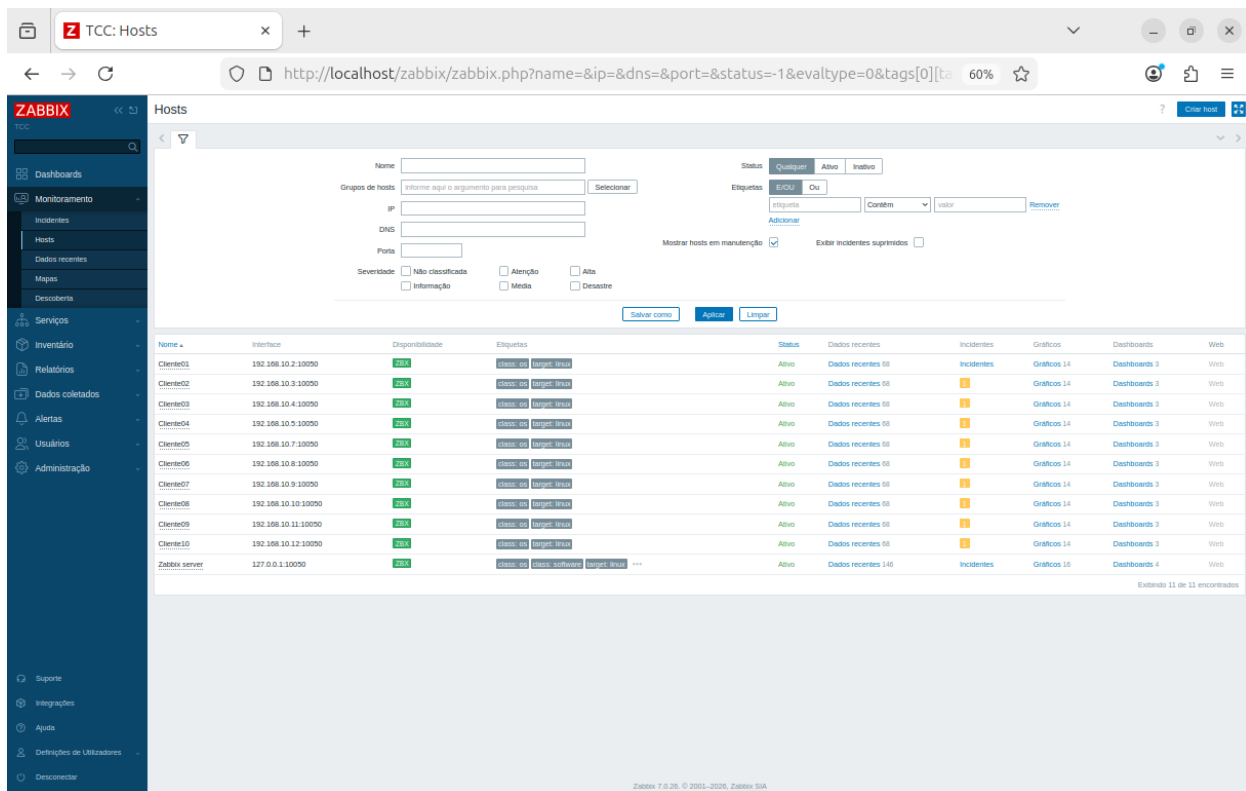
5.4.1 Zabbix: Monitoramento de Recursos

A Figura 5.7 é apresentada a interface de gerenciamento de hosts do *Zabbix*, ferramenta de monitoramento utilizada para acompanhar o desempenho e a disponibilidade dos dispositivos presentes na infraestrutura de rede. Por meio dessa interface, é possível visualizar informações detalhadas dos hosts monitorados, incluindo endereço *IP*, status de disponibilidade, incidentes registrados, gráficos de desempenho e dashboards associados.

No ambiente experimental desenvolvido, foram cadastradas dez máquinas virtuais clientes, além do próprio servidor *Zabbix*, permitindo o monitoramento centralizado de todos os dispositivos participantes dos testes. Os indicadores de disponibilidade exibidos em verde demonstram que os hosts estavam ativos e se comunicando corretamente com o servidor de monitoramento durante a realização dos experimentos.

Além da supervisão em tempo real, o *Zabbix* possibilitou a coleta e o armazenamento de métricas de desempenho da rede, fornecendo dados importantes para a análise dos resultados obtidos. Dessa forma, a ferramenta contribuiu para a validação do ambiente experimental, permitindo identificar falhas, acompanhar o comportamento dos hosts e garantir a confiabilidade das medições realizadas.

Figura 5.7: Interface de monitoramento dos hosts no Zabbix.



Fonte: Elaborado pelo autor (2026).

A Figura 5.7 é exibida a interface de gerenciamento de *hosts* do *Zabbix* utilizada no ambiente experimental. Nessa tela, é possível visualizar os dispositivos monitorados pelo servidor *Zabbix*, incluindo as dez máquinas virtuais clientes e o próprio servidor de monitoramento. Cada *host* é identificado por seu nome e endereço IP, permitindo o acompanhamento centralizado da infraestrutura virtual.

Conforme apresentado na Tabela 5.2, cada máquina do ambiente experimental foi associada a um endereço IP específico, permitindo sua identificação e monitoramento individual por meio do *Zabbix*. Essa organização possibilitou o acompanhamento centralizado dos dispositivos participantes dos testes, facilitando a coleta e a análise das métricas de desempenho durante os experimentos.

Os indicadores de disponibilidade exibidos na coluna *Disponibilidade* demonstram que os agentes *Zabbix* instalados nas máquinas virtuais estão se comunicando corretamente com o servidor de monitoramento. Além disso, a interface apresenta informações relacionadas ao *status* dos dispositivos, quantidade de gráficos disponíveis, incidentes registrados e acesso aos dados coletados em tempo real.

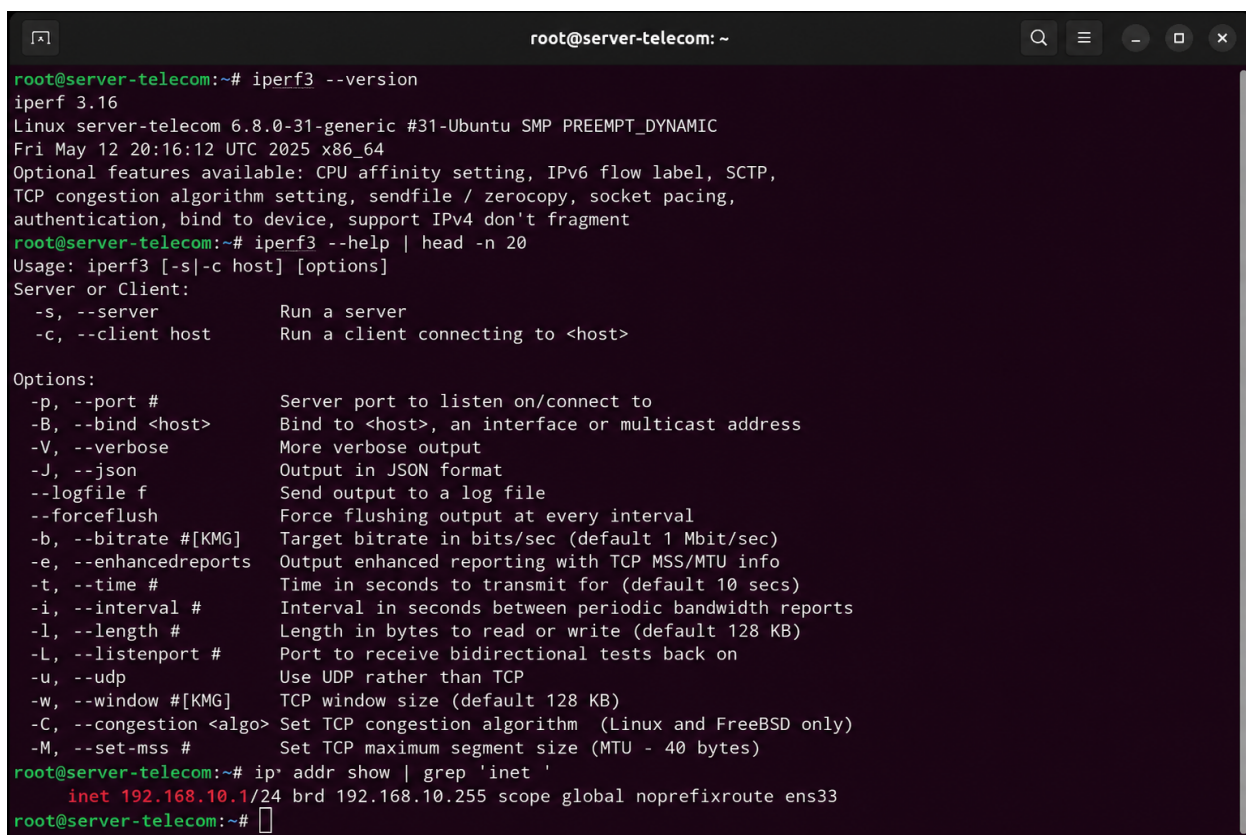
Através desse painel, foi possível monitorar continuamente o comportamento das máquinas virtuais durante a execução dos testes de desempenho realizados com o *Iperf3*. O *Zabbix* coletou métricas do sistema operacional *Ubuntu Server 24.04 LTS*, incluindo *Linux: System Load*, *Linux: CPU Usage*, *Linux: Memory Usage* e *Linux: Swap Usage*. Essas informações

permitiram analisar o impacto da geração de tráfego *TCP* e *UDP* sobre os recursos computacionais do ambiente virtualizado, contribuindo para a identificação de possíveis gargalos e para a validação dos resultados obtidos nos experimentos.

5.4.2 iPerf3: Avaliação de Desempenho da Rede

A Figura 5.8 apresenta a ferramenta *Iperf3* instalada no servidor do ambiente experimental. O *Iperf3* foi utilizado ao longo desta pesquisa para a geração de tráfego controlado entre as máquinas virtuais do *testbed*, permitindo a realização de testes de desempenho com os protocolos *TCP* e *UDP*.

Figura 5.8: Ferramenta *Iperf3* instalada no servidor do ambiente experimental.



```
root@server-telecom: ~  
root@server-telecom:~# iperf3 --version  
iperf 3.16  
Linux server-telecom 6.8.0-31-generic #31-Ubuntu SMP PREEMPT_DYNAMIC  
Fri May 12 20:16:12 UTC 2025 x86_64  
Optional features available: CPU affinity setting, IPv6 flow label, SCTP,  
TCP congestion algorithm setting, sendfile / zerocopy, socket pacing,  
authentication, bind to device, support IPv4 don't fragment  
root@server-telecom:~# iperf3 --help | head -n 20  
Usage: iperf3 [-s|-c host] [options]  
Server or Client:  
-s, --server          Run a server  
-c, --client host     Run a client connecting to <host>  
  
Options:  
-p, --port #         Server port to listen on/connect to  
-B, --bind <host>    Bind to <host>, an interface or multicast address  
-V, --verbose         More verbose output  
-J, --json           Output in JSON format  
--logfile f          Send output to a log file  
--forceflush         Force flushing output at every interval  
-b, --bitrate #[KMG] Target bitrate in bits/sec (default 1 Mbit/sec)  
-e, --enhancedreports Output enhanced reporting with TCP MSS/MTU info  
-t, --time #         Time in seconds to transmit for (default 10 secs)  
-i, --interval #     Interval in seconds between periodic bandwidth reports  
-l, --length #       Length in bytes to read or write (default 128 KB)  
-L, --listenport #   Port to receive bidirectional tests back on  
-u, --udp            Use UDP rather than TCP  
-w, --window #[KMG]  TCP window size (default 128 KB)  
-C, --congestion <algo> Set TCP congestion algorithm (Linux and FreeBSD only)  
-M, --set-mss #      Set TCP maximum segment size (MTU - 40 bytes)  
root@server-telecom:~# ip addr show | grep 'inet '  
    inet 192.168.10.1/24 brd 192.168.10.255 scope global noprefixroute ens33  
root@server-telecom:~#
```

Fonte: Elaborado pelo autor (2026).

Por meio dessa ferramenta, foram coletadas métricas fundamentais para a avaliação da rede, como *throughput*, *jitter* e perda de pacotes. Os resultados obtidos possibilitaram analisar o comportamento da infraestrutura virtualizada sob diferentes condições de carga, contribuindo para a comparação do desempenho dos protocolos estudados e para a validação do ambiente experimental proposto.

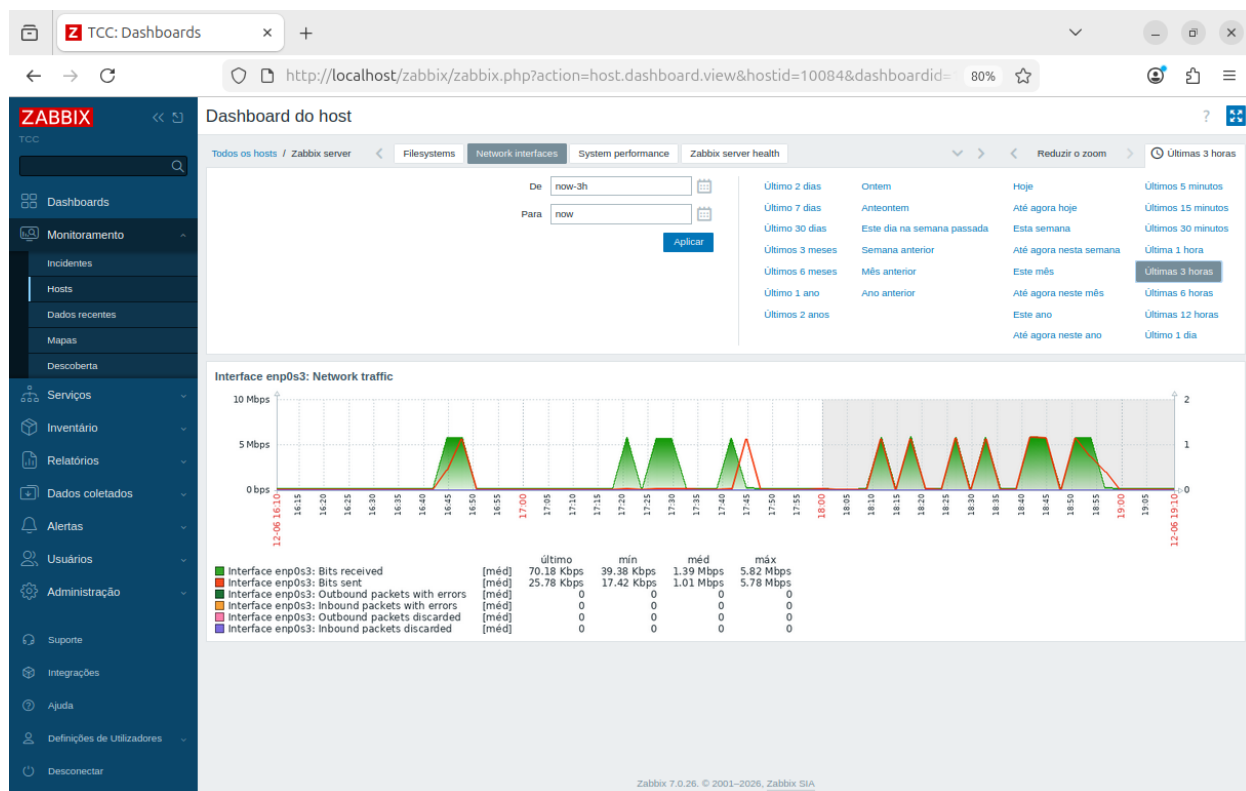
Por meio dessa ferramenta, foram coletadas métricas fundamentais para a avaliação da rede, como *throughput*, *jitter* e perda de pacotes. Os resultados obtidos possibilitaram analisar o comportamento da infraestrutura virtualizada sob diferentes condições de carga, con-

tribuinando para a comparação do desempenho dos protocolos estudados e para a validação do ambiente experimental proposto.

5.4.3 Monitoramento do Desempenho do Servidor da Rede

A Figura 5.9 é indicada o painel de monitoramento da interface de rede *enp0s3*, obtido por meio da plataforma *Zabbix*, contemplando o comportamento do tráfego de rede durante o intervalo analisado. O gráfico exibe as métricas de recepção (*Bits received*) e transmissão (*Bits sent*), permitindo uma avaliação detalhada do desempenho da comunicação de dados no ambiente monitorado.

Figura 5.9: Monitoramento do tráfego geral da interface de rede *enp0s3* por meio da plataforma *Zabbix*.



Fonte: Elaborado pelo autor (2026).

Os resultados evidenciam a ocorrência de picos de utilização da rede em instantes específicos, caracterizando um padrão de tráfego não contínuo. Esse comportamento é compatível com a execução de testes controlados de desempenho, nos quais a geração de tráfego ocorre apenas durante determinadas etapas do experimento. Durante o período monitorado, foram registrados valores máximos de aproximadamente **5,82 Mbps** para recepção de dados e **5,78 Mbps** para transmissão, demonstrando que a interface de rede suportou adequadamente a carga imposta pelos testes.

Em relação aos valores médios observados, verificou-se uma taxa de **1,39 Mbps** para tráfego de entrada e **1,01 Mbps** para tráfego de saída, indicando um nível moderado de

utilização dos recursos de rede ao longo da janela temporal considerada. Adicionalmente, os indicadores relacionados a erros de transmissão, erros de recepção e descarte de pacotes permaneceram com valor nulo durante todo o período analisado, evidenciando a inexistência de falhas ou perdas de dados na comunicação.

Dessa forma, os resultados obtidos demonstram que a infraestrutura de rede do ambiente virtualizado apresentou comportamento estável e confiável, assegurando a transmissão eficiente dos dados e mantendo a integridade da comunicação entre os sistemas monitorados. A ausência de erros e descartes de pacotes, aliada aos níveis consistentes de *throughput* observados, reforça a adequação do ambiente para a realização dos experimentos de avaliação de desempenho conduzidos neste trabalho.

5.5 Visão Geral do Monitoramento do Desempenho

A Figura 5.10 e 5.11 é exibida o painel de monitoramento configurado no Zabbix para acompanhar o tráfego de rede das máquinas participantes do ambiente experimental. O dashboard exibe, em tempo real, informações sobre a interface de rede *enp0s3* do servidor *Zabbix* e dos clientes monitorados.

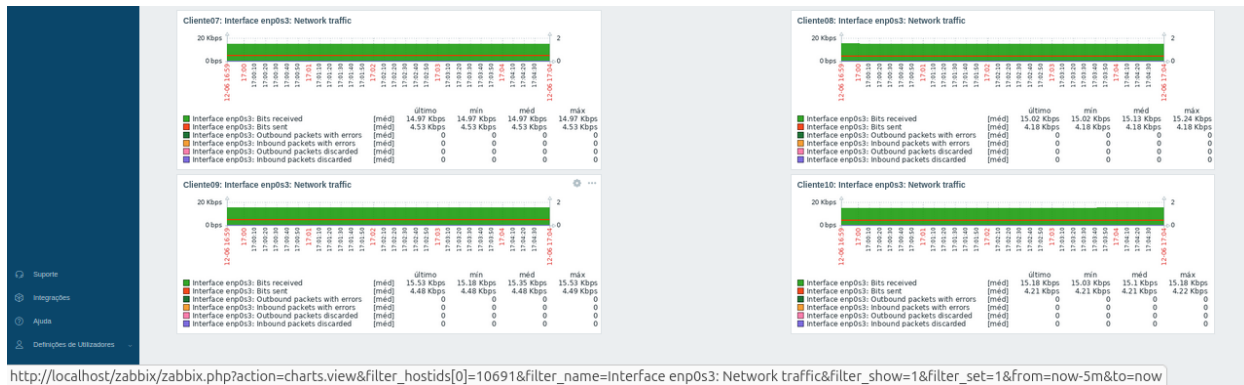
Os gráficos mostram as taxas de recepção (*Bits received*) e transmissão (*Bits sent*) de dados, medidas em kilobits por segundo (Kbps). Além disso, são disponibilizadas métricas estatísticas como valor mínimo, máximo, médio e último valor registrado para cada indicador monitorado.

Figura 5.10: *Dashboard de monitoramento por computador do tráfego de rede dos hosts do ambiente experimental utilizando o Zabbix.*



Fonte: Elaborado pelo autor (2026).

Figura 5.11: Dashboard de monitoramento por computador do tráfego de rede dos hosts do ambiente experimental utilizando o Zabbix.



Fonte: Elaborado pelo autor (2026).

Observa-se que o tráfego de rede permaneceu estável durante o período analisado, apresentando pequenas variações nas taxas de transmissão e recepção entre os diferentes clientes. Os valores registrados indicam uma utilização moderada da largura de banda disponível, sem ocorrência de picos significativos ou congestionamentos na rede.

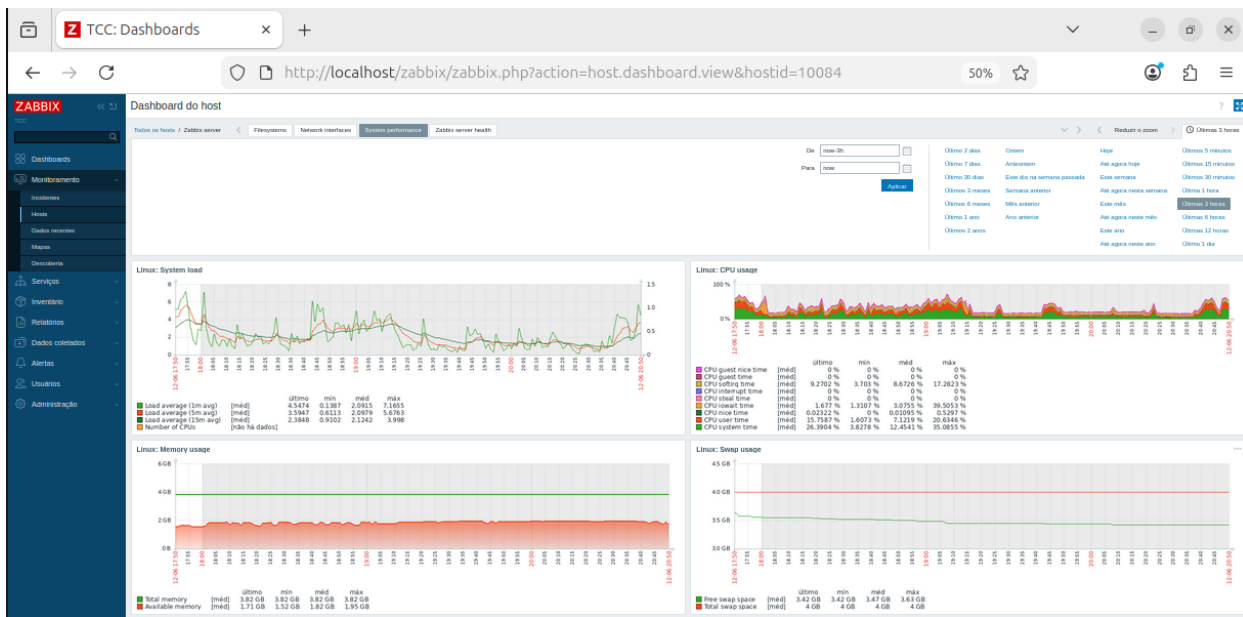
Outro aspecto relevante é a ausência de erros de transmissão, pacotes descartados ou falhas de comunicação, evidenciada pelos indicadores de *Outbound packets with errors*, *Inbound packets with errors*, *Outbound packets discarded* e *Inbound packets discarded*, todos com valor igual a zero. Esse comportamento demonstra que a infraestrutura de rede operou de forma estável e confiável durante os experimentos.

Dessa forma, o *dashboard* do Zabbix permitiu validar o correto funcionamento da comunicação entre as máquinas virtuais e fornecer evidências sobre a qualidade da rede utilizada nos testes de desempenho realizados com o *iPerf3*.

5.5.1 Monitoramento dos Recursos do Servidor de Métricas

A Figura 5.12 é demonstrada o painel de monitoramento do servidor obtido por meio da ferramenta Zabbix, contendo métricas relacionadas à carga do sistema, utilização da CPU, consumo de memória RAM e uso da memória de troca (*swap*). A análise desses indicadores é fundamental para avaliar o comportamento do ambiente computacional durante a execução dos experimentos e verificar a disponibilidade dos recursos de hardware.

Figura 5.12: Monitoramento de desempenho do servidor por meio do Zabbix.



Fonte: Elaborado pelo autor (2026).

No gráfico *Linux: System Load*, observa-se a evolução da carga média do sistema nos intervalos de 1, 5 e 15 minutos. Os resultados demonstram que, embora tenham ocorrido pequenas oscilações ao longo do período monitorado, os valores permaneceram dentro de uma faixa considerada adequada para a capacidade de processamento disponível. Os picos registrados ocorreram de forma pontual e não caracterizaram uma condição persistente de sobrecarga, indicando que o servidor foi capaz de atender satisfatoriamente às demandas de processamento geradas pelas aplicações em execução.

O gráfico *Linux: CPU Usage* apresenta a distribuição da utilização do processador entre os diferentes estados operacionais. Verifica-se que a maior parcela do tempo de processamento permaneceu em estado de inatividade (*idle*), evidenciando que a *CPU* operou com significativa margem de disponibilidade. Os percentuais associados aos processos de usuário (*user time*) e aos processos do sistema (*system time*) mantiveram-se em níveis reduzidos durante todo o período analisado, demonstrando que a carga computacional gerada pelas atividades executadas não comprometeu o desempenho do processador.

Em relação à utilização da memória principal, o gráfico *Linux: Memory Usage* indica que a capacidade total disponível manteve-se estável em aproximadamente 3,82 GB. O consumo de memória apresentou pequenas variações ao longo do monitoramento, permanecendo próximo de 1,8 GB durante a maior parte do período analisado. Dessa forma, observou-se a manutenção de uma quantidade significativa de memória livre, evidenciando que os recursos disponíveis foram suficientes para atender às necessidades do sistema sem a ocorrência de esgotamento da memória física.

Complementarmente, o gráfico *Linux: Swap Usage* demonstra o comportamento da memória de troca utilizada pelo sistema operacional. Os resultados revelam que a maior parte

da capacidade total de *swap* permaneceu livre durante todo o período analisado, indicando baixa dependência desse mecanismo. Esse comportamento é desejável, pois reduz a necessidade de transferência de dados entre a memória principal e o armazenamento secundário, contribuindo para a manutenção de níveis adequados de desempenho e responsividade do sistema.

De forma geral, os indicadores coletados evidenciam que o servidor operou em condições estáveis e dentro dos parâmetros esperados de desempenho. A baixa utilização dos recursos de processamento, a disponibilidade de memória *RAM* e a reduzida utilização da área de *swap* demonstram que a infraestrutura empregada possui capacidade suficiente para suportar as cargas de trabalho executadas durante os experimentos, garantindo confiabilidade e estabilidade ao ambiente de testes.

Capítulo 6

Resultados Obtidos

Durante os testes realizados com a ferramenta *iPerf3*, foram coletadas métricas específicas para os protocolos *TCP* e *UDP*. Essas métricas permitem avaliar aspectos relacionados à capacidade de transmissão, estabilidade da comunicação e qualidade da rede. A Tabela 6.1 apresenta a descrição dos principais indicadores analisados durante os experimentos.

Quadro 6.1: *Descrição das métricas analisadas nos testes TCP e UDP*

Métrica	Protocolo	Descrição
<i>Interval</i>	<i>TCP/UDP</i>	Intervalo de tempo utilizado para a medição do desempenho da transmissão, normalmente expresso em segundos.
<i>Transfer</i>	<i>TCP/UDP</i>	Quantidade total de dados transferidos durante o intervalo analisado, geralmente apresentada em Bytes, KBytes ou MBytes.
<i>Bitrate</i>	<i>TCP/UDP</i>	Taxa média de transmissão de dados durante o intervalo de medição, expressa em bits por segundo (bps), Kbps, Mbps ou Gbps.
<i>Jitter</i>	<i>UDP</i>	Variação no tempo de chegada dos pacotes à origem de destino. É uma métrica importante para aplicações em tempo real, como <i>VoIP</i> e videoconferência.
<i>Lost/Total Packets</i>	<i>UDP</i>	Relação entre o número de pacotes perdidos e o total de pacotes transmitidos durante o teste, indicando a qualidade da comunicação.

Fonte: Elaborado pelo autor (2026).

6.1 Desempenho *TCP* Cliente-Servidor

A Tabela 6.2 apresenta os resultados dos testes de desempenho do protocolo *TCP* no sentido cliente-servidor, obtidos a partir de dez execuções consecutivas. Observa-se que o PC 01 registrou a maior quantidade de dados transferidos, totalizando 115 MB durante um intervalo de 10,33 s, com taxa de transmissão de 93,7 Mbps. Os PCs 02, 03, 04, 06, 07, 08 e 09 apre-

sentaram comportamento bastante semelhante, transferindo predominantemente 114 MB de dados em intervalos compreendidos entre 10,24 s e 10,27 s, com taxas de transmissão variando entre 93,3 Mbps e 93,6 Mbps. O PC 05 foi a única execução a registrar uma transferência de 113 MB, alcançando taxa de transmissão de 92,8 Mbps, representando o menor desempenho observado entre os testes realizados. Por sua vez, o PC 10 obteve desempenho equivalente ao do PC 01 quanto ao volume de dados transferidos, registrando 115 MB em um intervalo de 10,31 s e taxa de transmissão de 93,6 Mbps.

A análise dos resultados demonstra que as variações entre as execuções foram reduzidas, evidenciando a estabilidade do protocolo *TCP* durante a comunicação no sentido cliente-servidor. O volume de dados transferidos oscilou entre 113 MB e 115 MB, enquanto as taxas de transmissão permaneceram compreendidas entre 92,8 Mbps e 93,7 Mbps, indicando um comportamento uniforme ao longo dos experimentos. Além disso, os tempos de execução mantiveram-se próximos de 10,3 s em todas as medições, reforçando a consistência do ambiente experimental. Considerando os valores médios apresentados na tabela, verificou-se um intervalo de execução de 10,27 s, transferência média de 114,1 MB e taxa média de transmissão de 93,44 Mbps. Dessa forma, os resultados confirmam que o protocolo *TCP* proporcionou um desempenho estável e confiável durante a transmissão de dados no sentido cliente-servidor, mantendo baixa variação entre as execuções e elevada repetibilidade dos experimentos.

Tabela 6.2: Resultados dos testes de desempenho *TCP* no sentido cliente-servidor

Computador	Intervalo (s)	Transfer (MBytes)	Bitrate (Mbps)
PC 01	0,00–10,33	115	93,7
PC 02	0,00–10,26	114	93,5
PC 03	0,00–10,27	114	93,5
PC 04	0,00–10,25	114	93,5
PC 05	0,00–10,25	113	92,8
PC 06	0,00–10,25	114	93,5
PC 07	0,00–10,27	114	93,6
PC 08	0,00–10,25	114	93,4
PC 09	0,00–10,24	114	93,3
PC 10	0,00–10,31	115	93,6
Média Geral	0,00–10,27	114,1	93,44

Fonte: Elaborado pelo autor (2026).

6.2 Desempenho *TCP* Servidor-Cliente

A Tabela 6.3 apresenta os resultados obtidos nos testes de desempenho do protocolo *TCP* no sentido servidor-cliente, realizados em dez execuções consecutivas. Verifica-se que, no PC 01, durante o intervalo de 0,00 a 10,33 s, foram transferidos 115 MB de dados, correspondendo a uma taxa de transmissão de 93,4 Mbps. No PC 02, considerando o intervalo de 0,00 a

10,00 s, registrou-se a transferência de 114 MB, com taxa de transmissão de 95,8 Mbps. O PC 03 apresentou, no mesmo intervalo, a transferência de 114 MB e taxa de transmissão de 95,7 Mbps. Na execução correspondente ao PC 04, foram transferidos 114 MB durante o intervalo de 0,00 a 10,01 s, alcançando taxa de transmissão de 95,5 Mbps. O PC 05 apresentou desempenho semelhante, com transferência de 114 MB, taxa de transmissão de 95,9 Mbps e intervalo de 0,00 a 10,01 s.

O PC 06 registrou a transferência de 113 MB durante o intervalo de 0,00 a 10,00 s, obtendo taxa de transmissão de 94,7 Mbps, sendo este o menor volume de dados transferidos entre as execuções. Os PCs 07, 09 e 10 destacaram-se por apresentarem a transferência de 115 MB de dados, com taxas de transmissão de 96,8 Mbps, 96,3 Mbps e 96,2 Mbps, respectivamente, configurando os melhores desempenhos observados. O PC 08, por sua vez, registrou a transferência de 114 MB durante o intervalo de 0,00 a 10,00 s, alcançando taxa de transmissão de 95,7 Mbps.

De maneira geral, os resultados demonstram elevada uniformidade entre as execuções, evidenciando a estabilidade do protocolo TCP durante os testes realizados. O volume de dados transferidos apresentou variação reduzida, oscilando entre 113 MB e 115 MB, enquanto a taxa de transmissão permaneceu compreendida entre 93,4 Mbps e 96,8 Mbps. Observa-se ainda que a maior parte das execuções foi concluída em aproximadamente 10 s, indicando consistência temporal na realização dos experimentos. Considerando os valores médios obtidos, verificou-se um intervalo de execução de 10,04 s, transferência média de 114,3 MB e taxa média de transmissão de 95,60 Mbps. Esses resultados evidenciam que o protocolo TCP apresentou desempenho consistente e confiável no ambiente experimental, mantendo elevada estabilidade na transferência de dados ao longo de todas as execuções.

Tabela 6.3: *Resultados dos testes de desempenho TCP no sentido servidor-cliente*

Computador	Intervalo (s)	Transfer (MBytes)	Bitrate (Mbps)
PC 01	0,00–10,33	115	93,4
PC 02	0,00–10,00	114	95,8
PC 03	0,00–10,00	114	95,7
PC 04	0,00–10,01	114	95,5
PC 05	0,00–10,01	114	95,9
PC 06	0,00–10,00	113	94,7
PC 07	0,00–10,00	115	96,8
PC 08	0,00–10,00	114	95,7
PC 09	0,00–10,00	115	96,3
PC 10	0,00–10,00	115	96,2
Média Geral	0,00–10,04	114,3	95,60

Fonte: Elaborado pelo autor (2026).

6.3 Desempenho *UDP* Cliente-Servidor

Os resultados apresentados na Tabela 6.4 correspondem aos testes de desempenho do protocolo *UDP* no sentido cliente-servidor, realizados com o auxílio da ferramenta *Iperf3*. Observa-se que os dez computadores apresentaram comportamento bastante homogêneo durante as medições, evidenciando a estabilidade da infraestrutura de rede utilizada. Os computadores PC 01, PC 02, PC 03, PC 04, PC 05, PC 06, PC 07, PC 08 e PC 10 registraram aproximadamente 1,25 KB de dados transferidos, com taxa de transmissão de 1,05 Kbit/s, enquanto o PC 09 apresentou uma pequena variação, registrando 1,22 KB e 1,02 Kbit/s. O tempo de execução dos testes permaneceu praticamente constante, variando entre 10,00 s e 10,09 s, demonstrando uniformidade nas condições experimentais.

Em relação ao *jitter*, os valores oscilaram entre 0,637 ms, observado no PC 07, e 2,055 ms, registrado no PC 04, com média geral de 1,31 ms, indicando baixa variação no atraso entre a chegada dos pacotes. Quanto aos datagramas transmitidos, a maioria dos computadores enviou 906 datagramas, enquanto pequenas diferenças foram observadas no PC 09, com 887 datagramas, e no PC 10, com 905 datagramas. Em todos os testes foi registrada ausência de perda de pacotes (0

De maneira geral, a média obtida foi de 10,01 segundos de duração, 1,25 KB de dados transferidos, 1,05 Kbit/s de taxa de transmissão, 1,31 ms de *jitter* e aproximadamente 904 datagramas transmitidos. Esses resultados evidenciam que a infraestrutura experimental apresentou desempenho estável e consistente para o protocolo *UDP*, com baixa variação entre os computadores avaliados e sem ocorrência de perdas de pacotes, confirmando a confiabilidade do ambiente utilizado para a realização dos experimentos.

Tabela 6.4: Resultados dos testes de desempenho *UDP* no sentido cliente-servidor

PC	Intervalo (s)	Transfer (KB)	Bitrate (Kbit/s)	Jitter (ms)	Datagramas
PC 01	0,00–10,01	1,25	1,05	0,808	0/906
PC 02	0,00–10,00	1,25	1,05	1,414	0/906
PC 03	0,00–10,00	1,25	1,05	0,796	0/906
PC 04	0,00–10,00	1,25	1,05	2,055	0/905
PC 05	0,00–10,00	1,25	1,05	1,473	0/906
PC 06	0,00–10,02	1,25	1,05	1,854	0/906
PC 07	0,00–10,00	1,25	1,05	0,637	0/906
PC 08	0,00–10,02	1,25	1,05	1,382	0/906
PC 09	0,00–10,09	1,22	1,02	1,667	0/887
PC 10	0,00–10,00	1,25	1,05	1,039	0/905
Média	0,00–10,01	1,25	1,05	1,31	0/904

Fonte: Elaborado pelo autor (2026).

6.4 Desempenho *UDP* Servidor-Cliente

A Tabela 6.5 apresenta os resultados obtidos nos testes de desempenho do protocolo *UDP* no sentido servidor-cliente, realizados em dez execuções consecutivas. Observa-se que, no PC 01, durante o intervalo de 0,00 a 10,00 s, foram transferidos 1,19 KB de dados, correspondendo a uma taxa de transmissão de 995 Kbit/s, com *jitter* igual a 0,000 ms e total de 859 datagramas transmitidos. No PC 02, considerando o intervalo de 0,00 a 10,06 s, registrou-se a transferência de 1,23 KB, taxa de transmissão de 1,02 Mbit/s, *jitter* de 0,000 ms e envio de 890 datagramas. O PC 03 apresentou, no intervalo de 0,00 a 10,00 s, transferência de 1,24 KB, taxa de 1,04 Mbit/s, *jitter* nulo e transmissão de 897 datagramas.

Na execução correspondente ao PC 04, foram transferidos 1,25 KB, com taxa de transmissão de 1,05 Mbit/s, *jitter* de 0,000 ms e 906 datagramas enviados. O PC 05 registrou a transferência de 1,24 KB durante o intervalo de 0,00 a 10,01 s, alcançando taxa de 1,04 Mbit/s, *jitter* de 0,000 ms e total de 898 datagramas. De forma semelhante, o PC 06 apresentou transferência de 1,24 KB, taxa de 1,04 Mbit/s, *jitter* igual a 0,000 ms e 902 datagramas transmitidos. Os PCs 07, 08 e 10 obtiveram resultados praticamente idênticos, com transferência de 1,25 KB, taxa de transmissão de 1,05 Mbit/s, *jitter* de 0,000 ms e 906 datagramas enviados em cada execução. Por sua vez, o PC 09 apresentou transferência de 1,22 KB, taxa de 1,02 Mbit/s, *jitter* de 0,000 ms e transmissão de 885 datagramas.

De maneira geral, os resultados evidenciam elevada uniformidade entre as execuções, indicando estabilidade na comunicação *UDP* durante todo o período experimental. A taxa de transmissão manteve-se próxima de 1 Mbit/s em todas as medições, enquanto o *jitter* permaneceu constante em 0,000 ms, demonstrando a inexistência de variações perceptíveis no atraso entre os pacotes transmitidos. Além disso, a quantidade de dados transferidos apresentou baixa variação, oscilando entre 1,19 KB e 1,25 KB, ao passo que o número de datagramas variou de 859 a 906. Considerando os valores médios, obteve-se um intervalo de teste de aproximadamente 10,01 s, transferência média de 1,24 KB, taxa média de transmissão de 1,03 Mbit/s, *jitter* de 0,000 ms e média de 895,5 datagramas transmitidos. Esses resultados demonstram a consistência do ambiente experimental e evidenciam que a comunicação baseada em *UDP* manteve desempenho estável e previsível ao longo de todas as execuções realizadas.

Tabela 6.5: *Resultados dos testes de desempenho UDP no sentido servidor-cliente*

PC	Intervalo (s)	Transfer. (KB)	Bitrate (Kbit/s)	Jitter (ms)	Datagramas
PC 01	0,00–10,00	1,19	995	0,000	859
PC 02	0,00–10,06	1,23	1,02	0,000	890
PC 03	0,00–10,00	1,24	1,04	0,000	897
PC 04	0,00–10,01	1,25	1,05	0,000	906
PC 05	0,00–10,01	1,24	1,04	0,000	898
PC 06	0,00–10,00	1,24	1,04	0,000	902
PC 07	0,00–10,00	1,25	1,05	0,000	906
PC 08	0,00–10,01	1,25	1,05	0,000	906
PC 09	0,00–10,01	1,22	1,02	0,000	885
PC 10	0,00–10,00	1,25	1,05	0,000	906
Média	0,00–10,01	1,24	1,03	0,000	895,5

Fonte: Elaborado pelo autor (2026).

6.5 Testes de Tráfego de Rede

A Tabela 6.6 apresenta o resumo do tráfego de rede monitorado pela ferramenta *Zabbix* durante a execução dos experimentos. Os resultados contemplam as taxas de bits recebidos e enviados por cada computador, bem como a ocorrência de erros e as observações registradas ao longo do monitoramento.

No PC 01, foram registrados 14,88 Kbps de bits recebidos e 4,96 Kbps de bits enviados, não sendo identificada qualquer ocorrência de erro. O comportamento observado foi caracterizado como tráfego estável. O PC 02 apresentou 14,14 Kbps de recepção e 4,44 Kbps de transmissão, também sem erros, sendo observado apenas um pequeno pico de envio durante o período de monitoramento. No PC 03, verificou-se uma taxa de 14,36 Kbps de bits recebidos e 5,19 Kbps de bits enviados, sem ocorrência de erros, sendo identificado um pico momentâneo de tráfego. De forma semelhante, o PC 04 registrou 14,90 Kbps de recepção e 5,81 Kbps de transmissão, apresentando igualmente um pico momentâneo de tráfego, porém sem comprometer a estabilidade da comunicação.

O PC 05 apresentou 14,50 Kbps de bits recebidos e 3,49 Kbps de bits enviados, mantendo tráfego uniforme durante toda a execução e sem registro de erros. No PC 06, foram observados 14,25 Kbps de recepção e 4,31 Kbps de transmissão, também sem erros, sendo identificada apenas uma redução temporária no tráfego ao final do monitoramento. Os PCs 07, 08, 09 e 10 apresentaram comportamento bastante semelhante, com taxas de recepção de 15,03 Kbps, 15,05 Kbps, 14,96 Kbps e 15,02 Kbps, respectivamente, enquanto as taxas de transmissão foram de 4,11 Kbps, 4,12 Kbps, 4,42 Kbps e 4,12 Kbps. Em todas essas execuções não foram registrados erros, sendo o comportamento da rede classificado como estável.

De maneira geral, os resultados demonstram elevada estabilidade do ambiente de rede monitorado pelo *Zabbix*. As taxas médias obtidas foram de 14,81 Kbps para os bits recebidos

e 4,50 Kbps para os bits enviados, não sendo registrada nenhuma ocorrência de erro durante todas as execuções realizadas. As pequenas variações observadas em alguns computadores, como picos momentâneos de tráfego ou redução temporária na transmissão, não comprometeram o desempenho da rede, sendo compatíveis com oscilações naturais do processo de comunicação. Dessa forma, os dados apresentados confirmam que o ambiente experimental permaneceu estável durante todo o período de monitoramento, evidenciando a confiabilidade da infraestrutura utilizada para a realização dos testes.

Tabela 6.6: *Resumo do tráfego de rede monitorado pelo Zabbix*

PC	Bits Recebidos	Bits Enviados	Erros	Observação
PC 01	14,88 Kbps	4,96 Kbps	0	Tráfego estável
PC 02	14,14 Kbps	4,44 Kbps	0	Pequeno pico de envio
PC 03	14,36 Kbps	5,19 Kbps	0	Pico momentâneo de tráfego
PC 04	14,90 Kbps	5,81 Kbps	0	Pico momentâneo de tráfego
PC 05	14,50 Kbps	3,49 Kbps	0	Tráfego uniforme
PC 06	14,25 Kbps	4,31 Kbps	0	Redução temporária ao final
PC 07	15,03 Kbps	4,11 Kbps	0	Tráfego estável
PC 08	15,05 Kbps	4,12 Kbps	0	Tráfego estável
PC 09	14,96 Kbps	4,42 Kbps	0	Tráfego estável
PC 10	15,02 Kbps	4,12 Kbps	0	Tráfego estável
Média	14,81 Kbps	4,50 Kbps	0	Rede estável

Fonte: Elaborado pelo autor (2026).

6.6 Métricas de Desempenho dos Computadores

A Tabela 6.7 apresenta o resumo do monitoramento de desempenho dos hosts realizado por meio da ferramenta *Zabbix*, considerando a carga de processamento, a utilização da *CPU*, o consumo de memória *RAM* e as observações registradas durante a execução dos experimentos. Os resultados permitem avaliar o comportamento individual de cada máquina virtual e verificar a estabilidade do ambiente experimental.

O PC 01 apresentou carga classificada como baixa, com utilização máxima de 17,54% e consumo de memória de 2,79 GB, sendo observado funcionamento estável durante todo o período de monitoramento. No PC 02, também classificado com carga baixa, registrou-se utilização máxima de 2,19% e consumo de 2,79 GB, sendo identificados apenas pequenos picos de carga, sem impacto significativo no desempenho. O PC 03 apresentou carga moderada, alcançando utilização máxima de 4,32% e consumo de 2,78 GB, registrando apenas um pico momentâneo de processamento. Da mesma forma, o PC 04 foi classificado com carga moderada, atingindo utilização máxima de 15,90% e consumo de 2,79 GB, sendo o host que apresentou a maior atividade entre todas as máquinas monitoradas.

O PC 05 apresentou carga baixa, utilização máxima de 11,87% e consumo de 2,79 GB, mantendo operação estável durante toda a execução dos testes. O PC 06 registrou carga

baixa, utilização máxima de 5,93% e consumo de 2,71 GB, evidenciando comportamento uniforme ao longo do monitoramento. O PC 07 também apresentou carga baixa, com utilização máxima de 11,15% e consumo de 2,78 GB, não sendo identificadas anomalias durante os experimentos. O PC 08 foi classificado com carga moderada, apresentando utilização máxima de 11,17% e consumo de 2,79 GB, sendo observados apenas picos ocasionais de carga. O PC 09 registrou carga baixa, com utilização máxima de 16,33% e consumo de 2,70 GB, apresentando apenas um pico isolado de processamento. Por fim, o PC 10 apresentou carga baixa, utilização máxima de 9,19% e consumo de 2,79 GB, mantendo comportamento estável durante todo o monitoramento.

De maneira geral, os resultados demonstram que o ambiente experimental apresentou elevado nível de estabilidade durante a execução dos testes. A carga média observada foi classificada como baixa, com utilização média de 10,96% e consumo médio de memória de 2,77 GB. Além disso, as variações verificadas em alguns hosts limitaram-se a pequenos picos temporários de processamento, sem comprometer a disponibilidade ou o desempenho das máquinas virtuais. Dessa forma, o monitoramento realizado pelo *Zabbix* confirma que os recursos computacionais permaneceram dentro de níveis adequados de utilização, evidenciando que o ambiente experimental manteve comportamento estável, confiável e compatível com a realização dos testes de desempenho da rede.

Tabela 6.7: *Resumo do monitoramento de desempenho dos hosts via Zabbix*

Host	Carga	Uso de CPU	Memória RAM	Observação
PC 01	Baixa	Até 17,54%	2,79 GB	Funcionamento estável
PC 02	Baixa	Até 2,19%	2,79 GB	Pequenos picos de carga
PC 03	Moderada	Até 4,32%	2,78 GB	Pico de processamento
PC 04	Moderada	Até 15,90%	2,79 GB	Maior atividade entre os hosts
PC 05	Baixa	Até 11,87%	2,79 GB	Operação estável
PC 06	Baixa	Até 5,93%	2,71 GB	Consumo uniforme
PC 07	Baixa	Até 11,15%	2,78 GB	Sem anomalias detectadas
PC 08	Moderada	Até 11,17%	2,79 GB	Picos ocasionais de carga
PC 09	Baixa	Até 16,33%	2,70 GB	Pico isolado de CPU
PC 10	Baixa	Até 9,19%	2,79 GB	Comportamento estável
Média	Baixa	10,96%	2,77 GB	Ambiente estável

Fonte: Elaborado pelo autor (2026).

6.7 Comparação dos Resultados com Trabalhos Relacionados

O Quadro 6.8 apresenta uma comparação entre os resultados de *throughput* obtidos neste trabalho e aqueles reportados em estudos relacionados, permitindo avaliar o desempenho da infraestrutura implementada em relação a diferentes cenários experimentais.

O estudo de [Zieliński 2023] foi realizado em um ambiente de rede *Ethernet Gigabit*, alcançando um *throughput* de 920 Mbps. Esse resultado é significativamente superior aos valores obtidos neste trabalho, uma vez que a infraestrutura utilizada dispõe de uma largura de banda nominal de 1 Gbps, permitindo elevadas taxas de transmissão e reduzida influência de gargalos de rede.

Por sua vez, [Rattanaopas e Boonchuay 2017] avaliaram um ambiente virtualizado utilizando três interfaces de rede, obtendo um *throughput* de 2,81 Gbps. O elevado desempenho observado está associado à utilização de múltiplas interfaces e à agregação de recursos de rede, ampliando significativamente a capacidade de transmissão disponível.

No trabalho proposto, os testes realizados com o protocolo *TCP* apresentaram resultados de 93,44 Mbps no sentido cliente-servidor e 95,60 Mbps no sentido servidor-cliente. Esses valores demonstram que a infraestrutura utilizada foi capaz de aproveitar grande parte da largura de banda disponível da rede *Fast Ethernet* (100 Mbps), atingindo aproximadamente 93% e 96% da capacidade nominal do enlace, respectivamente. Além disso, a pequena diferença entre os sentidos de transmissão evidencia a estabilidade e a consistência do ambiente experimental.

Em relação aos testes com o protocolo *UDP*, foram obtidos valores de *throughput* de 1,05 Mbps no sentido cliente-servidor e 1,03 Mbps no sentido servidor-cliente. Esses resultados são significativamente inferiores aos observados nos testes *TCP*, uma vez que a taxa de transmissão foi configurada especificamente para permitir a análise de métricas como perda de pacotes e *jitter*, não tendo como objetivo explorar a capacidade máxima da rede.

De modo geral, observa-se que os valores encontrados nos trabalhos relacionados são superiores aos obtidos neste estudo devido às diferenças de infraestrutura e configuração experimental. Enquanto os estudos de referência empregaram ambientes com enlaces Gigabit e soluções avançadas de virtualização, o presente trabalho foi desenvolvido em um ambiente laboratorial baseado em *Fast Ethernet*. Ainda assim, os resultados alcançados com o protocolo *TCP* demonstram elevado aproveitamento da capacidade disponível da rede, validando a eficiência da infraestrutura implementada para a realização dos experimentos.

Quadro 6.8: Comparação dos resultados obtidos com trabalhos relacionados

Referência	Ambiente	Throughput
Zieliński (2023)	Ethernet Gigabit	920 Mbps
Rattanaopas e Boonchuay (2017)	Virtualização com 3 NICs	2,81 Gbps
Trabalho Proposto	TCP Cliente–Servidor	93,44 Mbps
Trabalho Proposto	TCP Servidor–Cliente	95,60 Mbps
Trabalho Proposto	UDP Cliente–Servidor	1,05 Mbps
Trabalho Proposto	UDP Servidor–Cliente	1,03 Mbps

Fonte: Elaborado pelo autor (2026).

Capítulo 7

Considerações Finais e Sugestões para Trabalhos Futuros

O presente trabalho teve como objetivo principal avaliar o desempenho de redes de computadores por meio da utilização de um ambiente experimental do tipo *testbed*, integrando conceitos de virtualização, emulação e simulação. A proposta demonstrou-se eficaz ao possibilitar a criação de cenários controlados, reproduzíveis e flexíveis, fundamentais para a análise detalhada do comportamento da rede sob diferentes condições de tráfego e concorrência. A partir dos experimentos realizados, foi possível observar que a utilização de um *testbed* virtual permite uma análise consistente de métricas essenciais, como *throughput*, latência, *jitter* e perda de pacotes. Os resultados obtidos evidenciaram diferenças significativas no comportamento dos protocolos *TCP* e *UDP*, reforçando a importância da escolha adequada do protocolo de transporte conforme os requisitos da aplicação, seja priorizando confiabilidade ou baixa latência.

Além disso, a integração de ferramentas como *Iperf3*, *Zabbix* e *Grafana* mostrou-se fundamental para a geração de tráfego, coleta e visualização das métricas, contribuindo para uma análise mais precisa e estruturada dos dados. A abordagem adotada também evidenciou que ambientes virtualizados conseguem oferecer um equilíbrio adequado entre custo, flexibilidade e realismo, mesmo considerando as limitações inerentes ao compartilhamento de recursos físicos. Do ponto de vista acadêmico e prático, o trabalho contribui significativamente para o ensino e pesquisa em redes de computadores, ao disponibilizar uma metodologia acessível e replicável para experimentação. Dessa forma, o *testbed* proposto se consolida como uma ferramenta relevante para análise de desempenho, validação de soluções e desenvolvimento de competências técnicas na área.

7.1 Sugestões para Trabalhos Futuros

Os resultados obtidos neste trabalho demonstraram a viabilidade da utilização das ferramentas *Zabbix* e *Iperf3* para o monitoramento e a avaliação do desempenho de ambientes

computacionais e de rede. Entretanto, considerando a constante evolução das tecnologias de comunicação e das infraestruturas de TI, diversas possibilidades de expansão e aperfeiçoamento podem ser exploradas em trabalhos futuros.

Inicialmente, recomenda-se a ampliação do ambiente experimental por meio da inclusão de um número maior de hosts físicos e virtuais, permitindo a realização de análises em cenários mais complexos e representativos de ambientes corporativos. A utilização de diferentes sistemas operacionais e configurações de hardware também possibilitaria uma avaliação mais abrangente do impacto dos recursos computacionais sobre o desempenho da rede.

Outra proposta consiste na realização de testes em diferentes topologias de rede, incluindo arquiteturas distribuídas, ambientes com múltiplos segmentos de rede e infraestruturas geograficamente distribuídas. Essa abordagem permitiria avaliar o comportamento das métricas monitoradas em condições mais próximas das encontradas em organizações de médio e grande porte.

Além disso, sugere-se a inclusão de novas métricas de desempenho, tais como latência, perda de pacotes, tempo de resposta dos serviços, disponibilidade da infraestrutura e qualidade de experiência do usuário (*QoE*). A análise conjunta dessas métricas pode fornecer uma visão mais detalhada sobre o funcionamento da rede e auxiliar na identificação de possíveis gargalos ou pontos de falha.

Também é recomendada a integração do *Zabbix* com outras ferramentas de monitoramento, análise e visualização de dados, possibilitando a comparação dos resultados e a construção de painéis mais completos para apoio à tomada de decisão. A utilização de mecanismos automatizados para geração de alertas e relatórios pode contribuir para aumentar a eficiência do gerenciamento da infraestrutura monitorada.

Outra linha promissora de pesquisa envolve a aplicação de técnicas de inteligência artificial, aprendizado de máquina e análise preditiva sobre os dados coletados pelo sistema de monitoramento. Essas técnicas podem ser utilizadas para identificar padrões de comportamento, detectar anomalias, prever falhas e auxiliar na implementação de estratégias de manutenção preventiva.

Adicionalmente, futuros estudos podem investigar o desempenho das ferramentas em ambientes de computação em nuvem, virtualização em larga escala, contêineres e arquiteturas baseadas em microsserviços. Tais cenários são amplamente utilizados pelas organizações modernas e apresentam desafios específicos relacionados ao monitoramento e à gestão de recursos.

Outra possibilidade relevante consiste na avaliação do comportamento da infraestrutura em redes definidas por software (SDN) e em ambientes que utilizam tecnologias de virtualização de funções de rede (NFV). Essas abordagens representam tendências importantes na área de redes de computadores e podem beneficiar-se significativamente de soluções avançadas de monitoramento.

Por fim, recomenda-se a realização de experimentos envolvendo cenários de alta demanda de tráfego, ataques simulados, falhas controladas e eventos de congestionamento, com o

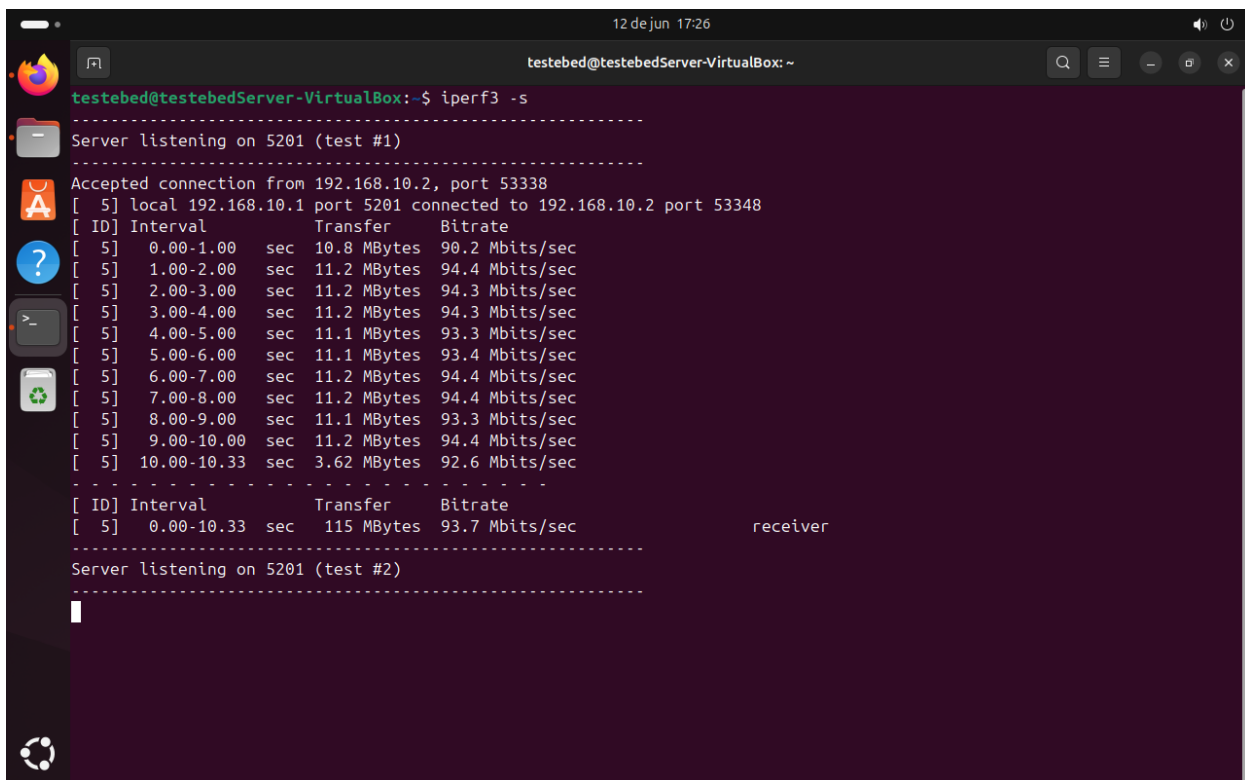
objetivo de analisar a capacidade das ferramentas em detectar alterações de desempenho e auxiliar na recuperação dos serviços. Dessa forma, será possível ampliar o conhecimento sobre a utilização de soluções de monitoramento em ambientes cada vez mais dinâmicos, complexos e críticos para as organizações.

Apêndice A

Capturas de Tela dos Experimentos e Métricas Coletadas

A.1 Testes TCP Cliente - Servidor

Figura A.1: Interface de monitoramento do host PC01 durante o teste TCP no Iperf3.



```
testebed@testebedServer-VirtualBox: ~$ iperf3 -s
Server listening on 5201 (test #1)
Accepted connection from 192.168.10.2, port 53338
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.2 port 53348
[ ID] Interval      Transfer    Bitrate
[ 5] 0.00-1.00 sec  10.8 MBytes 90.2 Mbits/sec
[ 5] 1.00-2.00 sec  11.2 MBytes 94.4 Mbits/sec
[ 5] 2.00-3.00 sec  11.2 MBytes 94.3 Mbits/sec
[ 5] 3.00-4.00 sec  11.2 MBytes 94.3 Mbits/sec
[ 5] 4.00-5.00 sec  11.1 MBytes 93.3 Mbits/sec
[ 5] 5.00-6.00 sec  11.1 MBytes 93.4 Mbits/sec
[ 5] 6.00-7.00 sec  11.2 MBytes 94.4 Mbits/sec
[ 5] 7.00-8.00 sec  11.2 MBytes 94.4 Mbits/sec
[ 5] 8.00-9.00 sec  11.1 MBytes 93.3 Mbits/sec
[ 5] 9.00-10.00 sec 11.2 MBytes 94.4 Mbits/sec
[ 5] 10.00-10.33 sec 3.62 MBytes 92.6 Mbits/sec
[ ID] Interval      Transfer    Bitrate
[ 5] 0.00-10.33 sec 115 MBytes 93.7 Mbits/sec
Server listening on 5201 (test #2)
```

Fonte: Elaborado pelo autor (2026).

Figura A.2: Interface de monitoramento do host PC02 durante o teste TCP no Iperf3.

```

-----
Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.10.3, port 52382
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.3 port 52384
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-1.03    sec  11.0 MBytes  89.2 Mb/s
[ 5]  1.03-2.00    sec  10.8 MBytes  93.3 Mb/s
[ 5]  2.00-3.00    sec  11.2 MBytes  94.4 Mb/s
[ 5]  3.00-4.00    sec  10.9 MBytes  91.2 Mb/s
[ 5]  4.00-5.00    sec  11.4 MBytes  95.4 Mb/s
[ 5]  5.00-6.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  6.00-7.00    sec  10.9 MBytes  91.2 Mb/s
[ 5]  7.00-8.00    sec  11.5 MBytes  96.5 Mb/s
[ 5]  8.00-9.00    sec  11.2 MBytes  94.4 Mb/s
[ 5]  9.00-10.00   sec  11.1 MBytes  93.2 Mb/s
[ 5] 10.00-10.26   sec   2.88 MBytes  92.8 Mb/s
-----
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-10.26   sec   114 MBytes  93.5 Mb/s
-----
Server listening on 5201 (test #2)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.3: Interface de monitoramento do host PC03 durante o teste TCP no Iperf3.

```

testebed@testebedServer-VirtualBox:~$ iperf3 -s
-----
Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.10.4, port 40362
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.4 port 40378
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-1.01    sec  11.0 MBytes  91.6 Mb/s
[ 5]  1.01-2.00    sec  11.1 MBytes  93.7 Mb/s
[ 5]  2.00-3.00    sec  11.0 MBytes  92.5 Mb/s
[ 5]  3.00-4.00    sec  11.2 MBytes  94.4 Mb/s
[ 5]  4.00-5.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  5.00-6.01    sec  11.1 MBytes  92.8 Mb/s
[ 5]  6.01-7.00    sec  11.2 MBytes  95.0 Mb/s
[ 5]  7.00-8.00    sec  11.1 MBytes  93.2 Mb/s
[ 5]  8.00-9.00    sec  11.1 MBytes  93.1 Mb/s
[ 5]  9.00-10.00   sec  11.2 MBytes  94.6 Mb/s
[ 5] 10.00-10.27   sec   2.88 MBytes  91.1 Mb/s
-----
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-10.27   sec   114 MBytes  93.5 Mb/s
-----
Server listening on 5201 (test #2)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.4: Interface de monitoramento do host PC04 durante o teste TCP no Iperf3.

```

Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.10.5, port 37714
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.5 port 37718
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-1.00      sec  11.0 MBytes  92.2 Mbits/sec
[ 5]  1.00-2.00      sec  11.1 MBytes  93.3 Mbits/sec
[ 5]  2.00-3.00      sec  11.2 MBytes  94.3 Mbits/sec
[ 5]  3.00-4.00      sec  11.1 MBytes  93.3 Mbits/sec
[ 5]  4.00-5.00      sec  11.1 MBytes  93.3 Mbits/sec
[ 5]  5.00-6.00      sec  11.2 MBytes  94.4 Mbits/sec
[ 5]  6.00-7.00      sec  11.1 MBytes  93.3 Mbits/sec
[ 5]  7.00-8.00      sec  11.1 MBytes  93.3 Mbits/sec
[ 5]  8.00-9.00      sec  11.2 MBytes  94.4 Mbits/sec
[ 5]  9.00-10.00     sec  11.1 MBytes  93.1 Mbits/sec
[ 5] 10.00-10.25     sec   2.75 MBytes 93.8 Mbits/sec
-----
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-10.25     sec   114 MBytes  93.5 Mbits/sec
-----
Server listening on 5201 (test #2)

```

receiver

Fonte: Elaborado pelo autor (2026).

Figura A.5: Interface de monitoramento do host PC05 durante o teste TCP no Iperf3.

```

Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.10.7, port 56398
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.7 port 56402
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-1.00      sec  10.1 MBytes  84.8 Mbits/sec
[ 5]  1.00-2.00      sec  11.1 MBytes  93.4 Mbits/sec
[ 5]  2.00-3.00      sec  11.0 MBytes  92.3 Mbits/sec
[ 5]  3.00-4.00      sec  11.2 MBytes  94.4 Mbits/sec
[ 5]  4.00-5.00      sec  11.1 MBytes  93.3 Mbits/sec
[ 5]  5.00-6.00      sec  11.1 MBytes  93.3 Mbits/sec
[ 5]  6.00-7.00      sec  11.2 MBytes  94.4 Mbits/sec
[ 5]  7.00-8.00      sec  11.1 MBytes  93.3 Mbits/sec
[ 5]  8.00-9.00      sec  11.1 MBytes  93.3 Mbits/sec
[ 5]  9.00-10.00     sec  11.2 MBytes  94.4 Mbits/sec
[ 5] 10.00-10.25     sec   2.75 MBytes 91.1 Mbits/sec
-----
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-10.25     sec   113 MBytes  92.8 Mbits/sec
-----
Server listening on 5201 (test #2)

```

receiver

Fonte: Elaborado pelo autor (2026).

Figura A.6: Interface de monitoramento do host PC06 durante o teste TCP no Iperf3.

```

-----
Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.10.8, port 42236
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.8 port 42242
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-1.00    sec  11.0 MBytes  92.2 Mb/s
[ 5]  1.00-2.00    sec  11.2 MBytes  94.4 Mb/s
[ 5]  2.00-3.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  3.00-4.00    sec  11.2 MBytes  94.4 Mb/s
[ 5]  4.00-5.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  5.00-6.03    sec  11.2 MBytes  92.0 Mb/s
[ 5]  6.03-7.00    sec  11.1 MBytes  95.7 Mb/s
[ 5]  7.00-8.00    sec  11.1 MBytes  93.2 Mb/s
[ 5]  8.00-9.00    sec  11.0 MBytes  92.3 Mb/s
[ 5]  9.00-10.00   sec  11.2 MBytes  94.5 Mb/s
[ 5] 10.00-10.25   sec   2.62 MBytes  87.9 Mb/s
-----
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-10.25   sec   114 MBytes  93.5 Mb/s
-----
Server listening on 5201 (test #2)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.7: Interface de monitoramento do host PC07 durante o teste TCP no Iperf3.

```

-----
Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.10.9, port 37842
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.9 port 37846
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-1.00    sec  11.1 MBytes  93.2 Mb/s
[ 5]  1.00-2.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  2.00-3.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  3.00-4.00    sec  11.2 MBytes  94.3 Mb/s
[ 5]  4.00-5.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  5.00-6.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  6.00-7.00    sec  11.2 MBytes  94.4 Mb/s
[ 5]  7.00-8.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  8.00-9.00    sec  11.1 MBytes  93.3 Mb/s
[ 5]  9.00-10.00   sec  11.2 MBytes  94.4 Mb/s
[ 5] 10.00-10.27   sec   2.88 MBytes  90.9 Mb/s
-----
[ ID] Interval      Transfer    Bitrate
[ 5]  0.00-10.27   sec   114 MBytes  93.6 Mb/s
-----
Server listening on 5201 (test #2)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.8: Interface de monitoramento do host PC08 durante o teste TCP no Iperf3.

```

-----
Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.10.10, port 56448
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.10 port 56450
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-1.00      sec   10.9 MBytes  91.2 Mbits/sec
[ 5]  1.00-2.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  2.00-3.00      sec   11.2 MBytes  94.4 Mbits/sec
[ 5]  3.00-4.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  4.00-5.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  5.00-6.00      sec   11.2 MBytes  94.4 Mbits/sec
[ 5]  6.00-7.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  7.00-8.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  8.00-9.00      sec   11.2 MBytes  94.3 Mbits/sec
[ 5]  9.00-10.00     sec   11.1 MBytes  93.4 Mbits/sec
[ 5] 10.00-10.25     sec    2.75 MBytes  90.8 Mbits/sec
-----
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-10.25     sec    114 MBytes  93.4 Mbits/sec
-----
Server listening on 5201 (test #2)

```

Fonte: Elaborado pelo autor (2026).

Figura A.9: Interface de monitoramento do host PC09 durante o teste TCP no Iperf3.

```

-----
Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.10.11, port 51492
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.11 port 51502
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-1.00      sec   10.8 MBytes  90.1 Mbits/sec
[ 5]  1.00-2.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  2.00-3.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  3.00-4.00      sec   11.2 MBytes  94.4 Mbits/sec
[ 5]  4.00-5.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  5.00-6.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  6.00-7.00      sec   11.2 MBytes  94.4 Mbits/sec
[ 5]  7.00-8.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  8.00-9.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  9.00-10.00     sec   11.2 MBytes  94.4 Mbits/sec
[ 5] 10.00-10.24     sec    2.62 MBytes  92.3 Mbits/sec
-----
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-10.24     sec    114 MBytes  93.3 Mbits/sec
-----
Server listening on 5201 (test #2)

```

Fonte: Elaborado pelo autor (2026).

Figura A.10: Interface de monitoramento do host PC10 durante o teste TCP no Iperf3.

```

Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.10.12, port 35224
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.12 port 35226
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-1.00      sec   11.0 MBytes  92.1 Mbits/sec
[ 5]  1.00-2.00      sec   11.2 MBytes  94.2 Mbits/sec
[ 5]  2.00-3.00      sec   11.1 MBytes  93.6 Mbits/sec
[ 5]  3.00-4.00      sec   11.1 MBytes  93.4 Mbits/sec
[ 5]  4.00-5.00      sec   11.1 MBytes  93.1 Mbits/sec
[ 5]  5.00-6.00      sec   11.2 MBytes  94.6 Mbits/sec
[ 5]  6.00-7.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  7.00-8.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  8.00-9.00      sec   11.2 MBytes  94.1 Mbits/sec
[ 5]  9.00-10.00     sec   11.1 MBytes  93.6 Mbits/sec
[ 5] 10.00-10.31     sec    3.50 MBytes  94.6 Mbits/sec
-----
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-10.31     sec   115 MBytes  93.6 Mbits/sec
-----
Server listening on 5201 (test #2)

```

Fonte: Elaborado pelo autor (2026).

A.2 Testes TCP Servidor - Cliente

Figura A.11: Interface de monitoramento do host PC01 observada pelo servidor Iperf3 durante o teste TCP.

```

Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.2, port 39918
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.2 port 39920
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-1.00      sec   10.5 MBytes  88.1 Mbits/sec
[ 5]  1.00-2.00      sec   11.1 MBytes  93.3 Mbits/sec
[ 5]  2.00-3.00      sec   11.2 MBytes  94.4 Mbits/sec
[ 5]  3.00-4.00      sec   11.2 MBytes  94.4 Mbits/sec
[ 5]  4.00-5.00      sec   11.2 MBytes  94.4 Mbits/sec
[ 5]  5.00-6.00      sec   11.2 MBytes  94.3 Mbits/sec
[ 5]  6.00-7.00      sec   11.1 MBytes  93.4 Mbits/sec
[ 5]  7.00-8.00      sec   11.2 MBytes  94.4 Mbits/sec
[ 5]  8.00-9.00      sec   11.2 MBytes  94.4 Mbits/sec
[ 5]  9.00-10.00     sec   11.2 MBytes  94.0 Mbits/sec
[ 5] 10.00-10.33     sec    3.62 MBytes  91.8 Mbits/sec
-----
[ ID] Interval          Transfer      Bitrate
[ 5]  0.00-10.33     sec   115 MBytes  93.4 Mbits/sec
-----
Server listening on 5201 (test #3)

```

Fonte: Elaborado pelo autor (2026).

Figura A.12: Interface de monitoramento do host PC02 observada pelo servidor Iperf3 durante o teste TCP.

```

Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.3, port 36508
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.3 port 36510
[ ID] Interval      Transfer    Bitrate      Retr  Cwnd
[ 5]  0.00-1.00    sec   13.6 MBytes  114 Mb/s      0    608 KBytes
[ 5]  1.00-2.00    sec   12.1 MBytes  102 Mb/s      0    1.15 MBytes
[ 5]  2.00-3.00    sec   10.9 MBytes  91.3 Mb/s     0    1.71 MBytes
[ 5]  3.00-4.00    sec   11.0 MBytes  92.0 Mb/s     0    2.27 MBytes
[ 5]  4.00-5.00    sec   10.9 MBytes  91.4 Mb/s     0    2.83 MBytes
[ 5]  5.00-6.00    sec   11.1 MBytes  93.5 Mb/s     0    3.38 MBytes
[ 5]  6.00-7.00    sec   12.2 MBytes  103 Mb/s     0    3.87 MBytes
[ 5]  7.00-8.00    sec   11.0 MBytes  92.2 Mb/s     0    3.87 MBytes
[ 5]  8.00-9.00    sec   10.5 MBytes  88.1 Mb/s     0    4.10 MBytes
[ 5]  9.00-10.00   sec   10.9 MBytes  91.2 Mb/s     0    4.10 MBytes
-----
[ ID] Interval      Transfer    Bitrate      Retr
[ 5]  0.00-10.00   sec   114 MBytes  95.8 Mb/s     0
                                           sender
-----
Server listening on 5201 (test #3)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.13: Interface de monitoramento do host PC03 observada pelo servidor Iperf3 durante o teste TCP.

```

Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.4, port 46966
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.4 port 46974
[ ID] Interval      Transfer    Bitrate      Retr  Cwnd
[ 5]  0.00-1.00    sec   13.4 MBytes  112 Mb/s      0    601 KBytes
[ 5]  1.00-2.00    sec   12.2 MBytes  103 Mb/s      0    1.14 MBytes
[ 5]  2.00-3.00    sec   11.0 MBytes  92.3 Mb/s     0    1.70 MBytes
[ 5]  3.00-4.00    sec   11.1 MBytes  93.3 Mb/s     0    2.26 MBytes
[ 5]  4.00-5.00    sec   11.0 MBytes  92.3 Mb/s     0    2.82 MBytes
[ 5]  5.00-6.01    sec   11.0 MBytes  91.6 Mb/s     0    3.38 MBytes
[ 5]  6.01-7.00    sec   11.0 MBytes  92.9 Mb/s     0    3.94 MBytes
[ 5]  7.00-8.00    sec   11.5 MBytes  96.3 Mb/s     0    4.02 MBytes
[ 5]  8.00-9.01    sec   11.2 MBytes  94.2 Mb/s     0    4.02 MBytes
[ 5]  9.01-10.00   sec   10.6 MBytes  89.5 Mb/s     0    4.32 MBytes
-----
[ ID] Interval      Transfer    Bitrate      Retr
[ 5]  0.00-10.00   sec   114 MBytes  95.7 Mb/s     0
                                           sender
-----
Server listening on 5201 (test #3)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.14: Interface de monitoramento do host PC04 observada pelo servidor Iperf3 durante o teste TCP.

```

-----
Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.5, port 43044
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.5 port 43060
[ ID] Interval          Transfer      Bitrate      Retr  Cwnd
[ 5]  0.00-1.00      sec   14.1 MBytes   118 Mbits/sec    0   587 KBytes
[ 5]  1.00-2.00      sec   10.6 MBytes   89.2 Mbits/sec    0   1.13 MBytes
[ 5]  2.00-3.00      sec   12.2 MBytes   103 Mbits/sec    0   1.69 MBytes
[ 5]  3.00-4.00      sec   10.5 MBytes   87.8 Mbits/sec    0   2.25 MBytes
[ 5]  4.00-5.00      sec   11.9 MBytes   100 Mbits/sec    0   2.81 MBytes
[ 5]  5.00-6.01      sec   10.2 MBytes   85.5 Mbits/sec    0   3.37 MBytes
[ 5]  6.01-7.01      sec   10.1 MBytes   84.4 Mbits/sec    0   3.91 MBytes
[ 5]  7.01-8.01      sec   11.5 MBytes   96.8 Mbits/sec    0   4.22 MBytes
[ 5]  8.01-9.00      sec   11.5 MBytes   97.2 Mbits/sec    0   4.32 MBytes
[ 5]  9.00-10.00     sec   10.9 MBytes   91.2 Mbits/sec    0   4.88 MBytes
-----
[ ID] Interval          Transfer      Bitrate      Retr
[ 5]  0.00-10.01     sec    114 MBytes   95.5 Mbits/sec    0
-----
Server listening on 5201 (test #3)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.15: Interface de monitoramento do host PC05 observada pelo servidor Iperf3 durante o teste TCP.

```

-----
Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.7, port 45674
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.7 port 45680
[ ID] Interval          Transfer      Bitrate      Retr  Cwnd
[ 5]  0.00-1.00      sec   13.8 MBytes   115 Mbits/sec    0   605 KBytes
[ 5]  1.00-2.00      sec   11.2 MBytes   94.4 Mbits/sec    0   1.15 MBytes
[ 5]  2.00-3.00      sec   11.0 MBytes   92.3 Mbits/sec    0   1.71 MBytes
[ 5]  3.00-4.00      sec   11.0 MBytes   92.2 Mbits/sec    0   2.27 MBytes
[ 5]  4.00-5.00      sec   12.2 MBytes   103 Mbits/sec    0   2.82 MBytes
[ 5]  5.00-6.00      sec   10.9 MBytes   91.3 Mbits/sec    0   3.38 MBytes
[ 5]  6.00-7.00      sec   11.8 MBytes   98.3 Mbits/sec    0   3.94 MBytes
[ 5]  7.00-8.00      sec   11.0 MBytes   92.5 Mbits/sec    0   4.01 MBytes
[ 5]  8.00-9.00      sec   10.2 MBytes   86.0 Mbits/sec    0   4.07 MBytes
[ 5]  9.00-10.00     sec   11.4 MBytes   95.4 Mbits/sec    0   4.43 MBytes
-----
[ ID] Interval          Transfer      Bitrate      Retr
[ 5]  0.00-10.01     sec    114 MBytes   95.9 Mbits/sec    0
-----
Server listening on 5201 (test #3)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.16: Interface de monitoramento do host PC06 observada pelo servidor Iperf3 durante o teste TCP.

```

Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.8, port 35264
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.8 port 35278
[ ID] Interval      Transfer    Bitrate      Retr  Cwnd
[ 5]  0.00-1.00    sec  12.1 MBytes  102 Mbits/sec  0    597 KBytes
[ 5]  1.00-2.00    sec  12.0 MBytes  101 Mbits/sec  0    1.14 MBytes
[ 5]  2.00-3.00    sec  11.0 MBytes  92.4 Mbits/sec  0    1.70 MBytes
[ 5]  3.00-4.00    sec  10.9 MBytes  91.1 Mbits/sec  0    2.26 MBytes
[ 5]  4.00-5.00    sec  11.0 MBytes  92.5 Mbits/sec  0    2.81 MBytes
[ 5]  5.00-6.00    sec  12.4 MBytes  103 Mbits/sec  0    3.37 MBytes
[ 5]  6.00-7.00    sec  11.0 MBytes  92.4 Mbits/sec  0    3.93 MBytes
[ 5]  7.00-8.00    sec  10.6 MBytes  89.2 Mbits/sec  0    4.32 MBytes
[ 5]  8.00-9.01    sec  11.5 MBytes  95.5 Mbits/sec  0    4.51 MBytes
[ 5]  9.01-10.00   sec  10.4 MBytes  88.0 Mbits/sec  0    4.51 MBytes
-----
[ ID] Interval      Transfer    Bitrate      Retr
[ 5]  0.00-10.00   sec  113 MBytes  94.7 Mbits/sec  0
-----
Server listening on 5201 (test #3)

```

Fonte: Elaborado pelo autor (2026).

Figura A.17: Interface de monitoramento do host PC07 observada pelo servidor Iperf3 durante o teste TCP.

```

Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.9, port 35738
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.9 port 35742
[ ID] Interval      Transfer    Bitrate      Retr  Cwnd
[ 5]  0.00-1.03    sec  13.8 MBytes  112 Mbits/sec  0    607 KBytes
[ 5]  1.03-2.00    sec  11.9 MBytes  102 Mbits/sec  0    1.14 MBytes
[ 5]  2.00-3.00    sec  11.2 MBytes  94.3 Mbits/sec  0    1.70 MBytes
[ 5]  3.00-4.00    sec  10.9 MBytes  91.2 Mbits/sec  0    2.26 MBytes
[ 5]  4.00-5.00    sec  10.9 MBytes  91.1 Mbits/sec  0    2.81 MBytes
[ 5]  5.00-6.00    sec  10.9 MBytes  91.3 Mbits/sec  0    3.37 MBytes
[ 5]  6.00-7.00    sec  11.5 MBytes  96.6 Mbits/sec  0    3.93 MBytes
[ 5]  7.00-8.00    sec  10.9 MBytes  91.2 Mbits/sec  0    4.07 MBytes
[ 5]  8.00-9.00    sec  10.5 MBytes  87.9 Mbits/sec  0    4.07 MBytes
[ 5]  9.00-10.00   sec  11.8 MBytes  98.7 Mbits/sec  0    4.07 MBytes
-----
[ ID] Interval      Transfer    Bitrate      Retr
[ 5]  0.00-10.00   sec  115 MBytes  96.8 Mbits/sec  0
-----
Server listening on 5201 (test #3)

```

Fonte: Elaborado pelo autor (2026).

Figura A.18: Interface de monitoramento do host PC08 observada pelo servidor Iperf3 durante o teste TCP.

```

Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.10, port 33050
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.10 port 33058
[ ID] Interval          Transfer      Bitrate      Retr  Cwnd
[ 5]  0.00-1.00    sec   13.1 MBytes   110 Mb/s      0    585 K
[ 5]  1.00-2.00    sec   10.8 MBytes   90.5 Mb/s     0    1.13 M
[ 5]  2.00-3.00    sec   11.0 MBytes   92.3 Mb/s     0    1.69 M
[ 5]  3.00-4.00    sec   10.9 MBytes   91.2 Mb/s     0    2.25 M
[ 5]  4.00-5.00    sec   11.2 MBytes   94.3 Mb/s     0    2.80 M
[ 5]  5.00-6.00    sec   11.2 MBytes   94.3 Mb/s     0    3.36 M
[ 5]  6.00-7.00    sec   11.2 MBytes   94.4 Mb/s     0    3.87 M
[ 5]  7.00-8.00    sec   11.1 MBytes   93.3 Mb/s     0    4.08 M
[ 5]  8.00-9.00    sec   11.4 MBytes   95.5 Mb/s     0    4.29 M
[ 5]  9.00-10.00   sec   10.9 MBytes   91.2 Mb/s     0    4.29 M
-----
[ ID] Interval          Transfer      Bitrate      Retr
[ 5]  0.00-10.00   sec   114 MBytes   95.7 Mb/s     0
-----
Server listening on 5201 (test #3)

```

Fonte: Elaborado pelo autor (2026).

Figura A.19: Interface de monitoramento do host PC09 observada pelo servidor Iperf3 durante o teste TCP.

```

Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.11, port 32918
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.11 port 32930
[ ID] Interval          Transfer      Bitrate      Retr  Cwnd
[ 5]  0.00-1.00    sec   13.2 MBytes   111 Mb/s      0    602 K
[ 5]  1.00-2.00    sec   12.2 MBytes   103 Mb/s     0    1.15 M
[ 5]  2.00-3.00    sec   10.9 MBytes   91.1 Mb/s     0    1.71 M
[ 5]  3.00-4.00    sec   11.0 MBytes   92.3 Mb/s     0    2.26 M
[ 5]  4.00-5.00    sec   10.9 MBytes   91.4 Mb/s     0    2.82 M
[ 5]  5.00-6.00    sec   11.1 MBytes   93.2 Mb/s     0    3.38 M
[ 5]  6.00-7.00    sec   12.5 MBytes   105 Mb/s     0    3.94 M
[ 5]  7.00-8.00    sec   10.9 MBytes   91.5 Mb/s     0    3.94 M
[ 5]  8.00-9.00    sec   11.1 MBytes   93.3 Mb/s     0    3.94 M
[ 5]  9.00-10.00   sec   11.0 MBytes   92.3 Mb/s     0    3.94 M
-----
[ ID] Interval          Transfer      Bitrate      Retr
[ 5]  0.00-10.00   sec   115 MBytes   96.3 Mb/s     0
-----
Server listening on 5201 (test #3)

```

Fonte: Elaborado pelo autor (2026).

Figura A.20: Interface de monitoramento do host PC10 observada pelo servidor Iperf3 durante o teste TCP.

```

Server listening on 5201 (test #2)
-----
Accepted connection from 192.168.10.12, port 47640
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.12 port 47644
[ ID] Interval          Transfer      Bitrate      Retr  Cwnd
[ 5]  0.00-1.00    sec   13.9 MBytes   116 Mbits/sec    0   605 KBytes
[ 5]  1.00-2.00    sec   12.2 MBytes   103 Mbits/sec    0   1.15 MBytes
[ 5]  2.00-3.00    sec   10.9 MBytes   91.2 Mbits/sec    0   1.71 MBytes
[ 5]  3.00-4.00    sec   11.0 MBytes   92.3 Mbits/sec    0   2.27 MBytes
[ 5]  4.00-5.00    sec   10.5 MBytes   88.1 Mbits/sec    0   2.82 MBytes
[ 5]  5.00-6.00    sec   12.2 MBytes   102 Mbits/sec    0   3.38 MBytes
[ 5]  6.00-7.00    sec   10.9 MBytes   91.5 Mbits/sec    0   3.94 MBytes
[ 5]  7.00-8.00    sec   11.1 MBytes   93.3 Mbits/sec    0   3.99 MBytes
[ 5]  8.00-9.00    sec   10.8 MBytes   90.2 Mbits/sec    0   4.04 MBytes
[ 5]  9.00-10.00   sec   11.2 MBytes   94.3 Mbits/sec    0   4.22 MBytes
-----
[ ID] Interval          Transfer      Bitrate      Retr
[ 5]  0.00-10.00   sec   115 MBytes   96.2 Mbits/sec    0
-----
Server listening on 5201 (test #3)

```

Fonte: Elaborado pelo autor (2026).

A.3 Testes UDP Cliente - Servidor

Figura A.21: Interface de monitoramento do host PC01 durante o teste UDP.

```

Server listening on 5201 (test #3)
-----
Accepted connection from 192.168.10.2, port 45488
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.2 port 46320
[ ID] Interval          Transfer      Bitrate      Jitter    Lost/Total Datagrams
[ 5]  0.00-1.01    sec    97.6 KBytes    789 Kbits/sec  1.904 ms    0/69 (0%)
[ 5]  1.01-2.00    sec   160 KBytes    1.32 Mbits/sec  3.551 ms    0/113 (0%)
[ 5]  2.00-3.00    sec   126 KBytes    1.03 Mbits/sec  1.395 ms    0/89 (0%)
[ 5]  3.00-4.00    sec   129 KBytes    1.05 Mbits/sec  1.825 ms    0/91 (0%)
[ 5]  4.00-5.00    sec   127 KBytes    1.04 Mbits/sec  1.119 ms    0/90 (0%)
[ 5]  5.00-6.00    sec   129 KBytes    1.05 Mbits/sec  1.748 ms    0/91 (0%)
[ 5]  6.00-7.00    sec   129 KBytes    1.05 Mbits/sec  1.572 ms    0/91 (0%)
[ 5]  7.00-8.00    sec   127 KBytes    1.04 Mbits/sec  1.003 ms    0/90 (0%)
[ 5]  8.00-9.00    sec   126 KBytes    1.03 Mbits/sec  2.241 ms    0/89 (0%)
[ 5]  9.00-10.00   sec   130 KBytes    1.06 Mbits/sec  0.817 ms    0/92 (0%)
[ 5] 10.00-10.01   sec    1.41 KBytes    2.56 Mbits/sec  0.808 ms    0/1 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter    Lost/Total Datagrams
[ 5]  0.00-10.01   sec   1.25 MBytes    1.05 Mbits/sec  0.808 ms    0/906 (0%)
-----
Server listening on 5201 (test #4)

```

Fonte: Elaborado pelo autor (2026).

Figura A.22: Interface de monitoramento do host PC02 durante o teste UDP.

```

Server listening on 5201 (test #3)
-----
Accepted connection from 192.168.10.3, port 43460
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.3 port 55096
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5]  0.00-1.00    sec      129 KBytes   1.05 Mb/s    1.853 ms    0/91 (0%)
[ 5]  1.00-2.00    sec      129 KBytes   1.05 Mb/s    1.043 ms    0/91 (0%)
[ 5]  2.00-3.00    sec      127 KBytes   1.04 Mb/s    0.702 ms    0/90 (0%)
[ 5]  3.00-4.00    sec      127 KBytes   1.04 Mb/s    7.319 ms    0/90 (0%)
[ 5]  4.00-5.00    sec      129 KBytes   1.05 Mb/s    0.691 ms    0/91 (0%)
[ 5]  5.00-6.00    sec      129 KBytes   1.06 Mb/s    0.678 ms    0/91 (0%)
[ 5]  6.00-7.00    sec      127 KBytes   1.04 Mb/s    2.087 ms    0/90 (0%)
[ 5]  7.00-8.00    sec      127 KBytes   1.04 Mb/s    0.872 ms    0/90 (0%)
[ 5]  8.00-9.00    sec      129 KBytes   1.05 Mb/s    1.574 ms    0/91 (0%)
[ 5]  9.00-10.00   sec      129 KBytes   1.06 Mb/s    1.414 ms    0/91 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5]  0.00-10.00   sec      1.25 MBytes  1.05 Mb/s    1.414 ms    0/906 (0%) receiver
-----
Server listening on 5201 (test #4)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.23: Interface de monitoramento do host PC03 durante o teste UDP.

```

-----
Server listening on 5201 (test #3)
-----
Accepted connection from 192.168.10.4, port 57454
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.4 port 58825
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5]  0.00-1.00    sec      115 KBytes   938 Kbits/sec  1.013 ms    0/81 (0%)
[ 5]  1.00-2.00    sec      140 KBytes   1.15 Mb/s     0.716 ms    0/99 (0%)
[ 5]  2.00-3.00    sec      112 KBytes   913 Kbits/sec  0.616 ms    0/79 (0%)
[ 5]  3.00-4.01    sec      146 KBytes   1.18 Mb/s     0.859 ms    0/103 (0%)
[ 5]  4.01-5.00    sec      127 KBytes   1.05 Mb/s     1.566 ms    0/90 (0%)
[ 5]  5.00-6.00    sec      130 KBytes   1.06 Mb/s     1.142 ms    0/92 (0%)
[ 5]  6.00-7.00    sec      127 KBytes   1.04 Mb/s     1.137 ms    0/90 (0%)
[ 5]  7.00-8.00    sec      127 KBytes   1.04 Mb/s     3.519 ms    0/90 (0%)
[ 5]  8.00-9.00    sec      129 KBytes   1.06 Mb/s     2.445 ms    0/91 (0%)
[ 5]  9.00-10.00   sec      129 KBytes   1.05 Mb/s     0.796 ms    0/91 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5]  0.00-10.00   sec      1.25 MBytes  1.05 Mb/s     0.796 ms    0/906 (0%) receiver
-----
Server listening on 5201 (test #4)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.24: Interface de monitoramento do host PC04 durante o teste UDP.

```

Server listening on 5201 (test #3)
-----
Accepted connection from 192.168.10.5, port 50238
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.5 port 49765
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-1.00    sec    84.8 KBytes    694 Kbits/sec  2.208 ms    0/60 (0%)
[ 5] 1.00-2.05    sec    117 KBytes    918 Kbits/sec  0.821 ms    0/83 (0%)
[ 5] 2.05-3.00    sec    134 KBytes    1.15 Mbits/sec  0.393 ms    0/95 (0%)
[ 5] 3.00-4.01    sec    117 KBytes    955 Kbits/sec  2.150 ms    0/83 (0%)
[ 5] 4.01-5.00    sec    164 KBytes    1.35 Mbits/sec  1.059 ms    0/116 (0%)
[ 5] 5.00-6.00    sec    126 KBytes    1.03 Mbits/sec  0.475 ms    0/89 (0%)
[ 5] 6.00-7.00    sec    153 KBytes    1.25 Mbits/sec  0.504 ms    0/108 (0%)
[ 5] 7.00-8.00    sec    129 KBytes    1.05 Mbits/sec  1.861 ms    0/91 (0%)
[ 5] 8.00-9.00    sec    127 KBytes    1.04 Mbits/sec  2.611 ms    0/90 (0%)
[ 5] 9.00-10.00   sec    127 KBytes    1.04 Mbits/sec  2.055 ms    0/90 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.00   sec    1.25 MBytes    1.05 Mbits/sec  2.055 ms    0/905 (0%) receiver
-----
Server listening on 5201 (test #4)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.25: Interface de monitoramento do host PC05 durante o teste UDP.

```

Server listening on 5201 (test #3)
-----
Accepted connection from 192.168.10.7, port 48576
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.7 port 32842
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-1.00    sec    129 KBytes    1.05 Mbits/sec  1.853 ms    0/91 (0%)
[ 5] 1.00-2.00    sec    127 KBytes    1.04 Mbits/sec  2.702 ms    0/90 (0%)
[ 5] 2.00-3.00    sec    127 KBytes    1.04 Mbits/sec  0.907 ms    0/90 (0%)
[ 5] 3.00-4.00    sec    129 KBytes    1.05 Mbits/sec  1.091 ms    0/91 (0%)
[ 5] 4.00-5.03    sec    132 KBytes    1.05 Mbits/sec  1.474 ms    0/93 (0%)
[ 5] 5.03-6.01    sec    126 KBytes    1.06 Mbits/sec  1.204 ms    0/89 (0%)
[ 5] 6.01-7.00    sec    127 KBytes    1.05 Mbits/sec  1.956 ms    0/90 (0%)
[ 5] 7.00-8.00    sec    129 KBytes    1.05 Mbits/sec  0.636 ms    0/91 (0%)
[ 5] 8.00-9.00    sec    127 KBytes    1.04 Mbits/sec  1.272 ms    0/90 (0%)
[ 5] 9.00-10.00   sec    129 KBytes    1.06 Mbits/sec  1.473 ms    0/91 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.00   sec    1.25 MBytes    1.05 Mbits/sec  1.473 ms    0/906 (0%) receiver
-----
Server listening on 5201 (test #4)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.26: Interface de monitoramento do host PC06 durante o teste UDP.

```

Server listening on 5201 (test #3)
-----
Accepted connection from 192.168.10.8, port 44124
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.8 port 37132
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-1.00 sec      99.0 KBytes   810 Kbits/sec 0.296 ms    0/70 (0%)
[ 5] 1.00-2.00 sec     124 KBytes   1.02 Mbits/sec 1.032 ms    0/88 (0%)
[ 5] 2.00-3.00 sec     76.4 KBytes   626 Kbits/sec 1.124 ms    0/54 (0%)
[ 5] 3.00-4.01 sec     174 KBytes   1.42 Mbits/sec 0.612 ms    0/123 (0%)
[ 5] 4.01-5.00 sec     141 KBytes   1.17 Mbits/sec 0.474 ms    0/100 (0%)
[ 5] 5.00-6.00 sec     126 KBytes   1.03 Mbits/sec 0.426 ms    0/89 (0%)
[ 5] 6.00-7.07 sec     140 KBytes   1.07 Mbits/sec 0.860 ms    0/99 (0%)
[ 5] 7.07-8.00 sec     123 KBytes   1.09 Mbits/sec 0.941 ms    0/87 (0%)
[ 5] 8.00-9.01 sec     148 KBytes   1.21 Mbits/sec 0.315 ms    0/105 (0%)
[ 5] 9.01-10.01 sec    127 KBytes   1.04 Mbits/sec 1.943 ms    0/90 (0%)
[ 5] 10.01-10.02 sec   1.41 KBytes   2.48 Mbits/sec 1.854 ms    0/1 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.02 sec    1.25 MBytes   1.05 Mbits/sec 1.854 ms    0/906 (0%) receiver
-----
Server listening on 5201 (test #4)

```

Fonte: Elaborado pelo autor (2026).

Figura A.27: Interface de monitoramento do host PC07 durante o teste UDP.

```

Server listening on 5201 (test #3)
-----
Accepted connection from 192.168.10.9, port 46352
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.9 port 50562
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-1.01 sec     129 KBytes   1.05 Mbits/sec 2.055 ms    0/91 (0%)
[ 5] 1.01-2.00 sec     127 KBytes   1.05 Mbits/sec 1.263 ms    0/90 (0%)
[ 5] 2.00-3.00 sec     129 KBytes   1.05 Mbits/sec 0.798 ms    0/91 (0%)
[ 5] 3.00-4.00 sec     127 KBytes   1.04 Mbits/sec 0.975 ms    0/90 (0%)
[ 5] 4.00-5.05 sec     136 KBytes   1.06 Mbits/sec 7.221 ms    0/96 (0%)
[ 5] 5.05-6.00 sec     122 KBytes   1.05 Mbits/sec 5.673 ms    0/86 (0%)
[ 5] 6.00-7.00 sec     127 KBytes   1.04 Mbits/sec 1.407 ms    0/90 (0%)
[ 5] 7.00-8.00 sec     129 KBytes   1.05 Mbits/sec 1.814 ms    0/91 (0%)
[ 5] 8.00-9.00 sec     127 KBytes   1.04 Mbits/sec 0.873 ms    0/90 (0%)
[ 5] 9.00-10.00 sec    129 KBytes   1.06 Mbits/sec 0.637 ms    0/91 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.00 sec    1.25 MBytes   1.05 Mbits/sec 0.637 ms    0/906 (0%) receiver
-----
Server listening on 5201 (test #4)

```

Fonte: Elaborado pelo autor (2026).

Figura A.28: Interface de monitoramento do host PC08 durante o teste UDP.

```

Server listening on 5201 (test #3)
-----
Accepted connection from 192.168.10.10, port 34388
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.10 port 39071
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5]  0.00-1.00    sec    107 KBytes    878 Kbits/sec  1.096 ms    0/76 (0%)
[ 5]  1.00-2.00    sec    141 KBytes    1.16 Mbits/sec  1.054 ms    0/100 (0%)
[ 5]  2.00-3.00    sec    122 KBytes    996 Kbits/sec  1.059 ms    0/86 (0%)
[ 5]  3.00-4.00    sec    124 KBytes    1.02 Mbits/sec  1.029 ms    0/88 (0%)
[ 5]  4.00-5.00    sec    124 KBytes    1.02 Mbits/sec  0.752 ms    0/88 (0%)
[ 5]  5.00-6.00    sec    147 KBytes    1.21 Mbits/sec  1.007 ms    0/104 (0%)
[ 5]  6.00-7.00    sec    129 KBytes    1.05 Mbits/sec  1.633 ms    0/91 (0%)
[ 5]  7.00-8.00    sec    127 KBytes    1.04 Mbits/sec  1.197 ms    0/90 (0%)
[ 5]  8.00-9.00    sec    70.7 KBytes    580 Kbits/sec  1.300 ms    0/50 (0%)
[ 5]  9.00-10.00   sec    187 KBytes    1.53 Mbits/sec  1.431 ms    0/132 (0%)
[ 5] 10.00-10.02   sec    1.41 KBytes    521 Kbits/sec  1.382 ms    0/1 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5]  0.00-10.02   sec    1.25 MBytes    1.05 Mbits/sec  1.382 ms    0/906 (0%) receiver
-----
Server listening on 5201 (test #4)

```

Fonte: Elaborado pelo autor (2026).

Figura A.29: Interface de monitoramento do host PC09 durante o teste UDP.

```

Server listening on 5201 (test #3)
-----
Accepted connection from 192.168.10.11, port 50958
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.11 port 55212
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5]  0.00-1.00    sec    107 KBytes    880 Kbits/sec  0.458 ms    0/76 (0%)
[ 5]  1.00-2.00    sec    148 KBytes    1.21 Mbits/sec  1.685 ms    0/105 (0%)
[ 5]  2.00-3.00    sec    127 KBytes    1.04 Mbits/sec  1.245 ms    0/90 (0%)
[ 5]  3.00-4.00    sec    129 KBytes    1.05 Mbits/sec  0.538 ms    0/91 (0%)
[ 5]  4.00-5.00    sec    127 KBytes    1.04 Mbits/sec  1.101 ms    0/90 (0%)
[ 5]  5.00-6.01    sec    130 KBytes    1.06 Mbits/sec  1.332 ms    0/92 (0%)
[ 5]  6.01-7.01    sec    109 KBytes    893 Kbits/sec  3.952 ms    0/77 (0%)
[ 5]  7.01-8.00    sec    91.9 KBytes    759 Kbits/sec  1.937 ms    0/65 (0%)
[ 5]  8.00-9.02    sec    185 KBytes    1.49 Mbits/sec  0.735 ms    0/131 (0%)
[ 5]  9.02-10.01   sec    97.6 KBytes    808 Kbits/sec  1.614 ms    0/69 (0%)
[ 5] 10.01-10.09   sec    1.41 KBytes    142 Kbits/sec  1.667 ms    0/1 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5]  0.00-10.09   sec    1.22 MBytes    1.02 Mbits/sec  1.667 ms    0/887 (0%) receiver
-----
Server listening on 5201 (test #4)

```

Fonte: Elaborado pelo autor (2026).

Figura A.30: Interface de monitoramento do host PC10 durante o teste UDP.

```

-----
Accepted connection from 192.168.10.12, port 40586
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.12 port 58790
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-1.00 sec      129 KBytes    1.05 Mbits/sec  1.350 ms    0/91 (0%)
[ 5] 1.00-2.00 sec      127 KBytes    1.04 Mbits/sec  0.719 ms    0/90 (0%)
[ 5] 2.00-3.00 sec      127 KBytes    1.04 Mbits/sec  2.897 ms    0/90 (0%)
[ 5] 3.00-4.00 sec      129 KBytes    1.05 Mbits/sec  2.656 ms    0/91 (0%)
[ 5] 4.00-5.00 sec      129 KBytes    1.05 Mbits/sec  0.756 ms    0/91 (0%)
[ 5] 5.00-6.00 sec      127 KBytes    1.04 Mbits/sec  3.671 ms    0/90 (0%)
[ 5] 6.00-7.00 sec      127 KBytes    1.05 Mbits/sec  2.502 ms    0/90 (0%)
[ 5] 7.00-8.00 sec      129 KBytes    1.05 Mbits/sec  1.534 ms    0/91 (0%)
[ 5] 8.00-9.00 sec      129 KBytes    1.05 Mbits/sec  1.242 ms    0/91 (0%)
[ 5] 9.00-10.00 sec     127 KBytes    1.04 Mbits/sec  1.039 ms    0/90 (0%)
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.00 sec     1.25 MBytes    1.05 Mbits/sec  1.039 ms    0/905 (0%) receiver
-----
Server listening on 5201 (test #4)
-----

```

Fonte: Elaborado pelo autor (2026).

A.4 Testes UDP Servidor - Cliente

Figura A.31: Interface de monitoramento do host PC01 observada pelo servidor Iperf3 durante o teste UDP.

```

Server listening on 5201 (test #4)
-----
Accepted connection from 192.168.10.2, port 36024
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.2 port 59316
[ ID] Interval          Transfer      Bitrate      Total Datagrams
[ 5] 0.00-1.13 sec      87.7 KBytes    634 Kbits/sec    63
[ 5] 1.13-2.17 sec      168 KBytes    1.32 Mbits/sec   118
[ 5] 2.17-3.06 sec      99.0 KBytes    912 Kbits/sec    70
[ 5] 3.06-4.06 sec      134 KBytes    1.10 Mbits/sec    96
[ 5] 4.06-5.02 sec      66.5 KBytes    568 Kbits/sec    46
[ 5] 5.02-6.23 sec      175 KBytes    1.19 Mbits/sec   124
[ 5] 6.23-7.00 sec      165 KBytes    1.76 Mbits/sec   117
[ 5] 7.00-8.00 sec      129 KBytes    1.05 Mbits/sec    91
[ 5] 8.00-9.00 sec      127 KBytes    1.04 Mbits/sec    90
[ 5] 9.00-10.00 sec     62.2 KBytes    508 Kbits/sec    44
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.00 sec     1.19 MBytes    995 Kbits/sec  0.000 ms    0/859 (0%) sender
-----
Server listening on 5201 (test #5)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.32: Interface de monitoramento do host PC02 observada pelo servidor Iperf3 durante o teste UDP.

```

Server listening on 5201 (test #4)
-----
Accepted connection from 192.168.10.3, port 48456
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.3 port 39282
[ ID] Interval          Transfer      Bitrate      Total Datagrams
[ 5] 0.00-1.00      sec    129 KBytes    1.05 Mbits/sec    91
[ 5] 1.00-2.10      sec    102 KBytes    760 Kbits/sec     72
[ 5] 2.10-3.00      sec    154 KBytes    1.40 Mbits/sec   109
[ 5] 3.00-4.45      sec    124 KBytes    704 Kbits/sec     88
[ 5] 4.45-5.11      sec    130 KBytes    1.60 Mbits/sec     92
[ 5] 5.11-6.01      sec    124 KBytes    1.14 Mbits/sec     88
[ 5] 6.01-7.00      sec    133 KBytes    1.09 Mbits/sec     94
[ 5] 7.00-8.01      sec    127 KBytes    1.04 Mbits/sec     90
[ 5] 8.01-9.05      sec    113 KBytes    890 Kbits/sec     80
[ 5] 9.05-10.06     sec    122 KBytes    984 Kbits/sec     86
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.06     sec    1.23 MBytes    1.02 Mbits/sec  0.000 ms    0/890 (0%) sender
-----
Server listening on 5201 (test #5)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.33: Interface de monitoramento do host PC03 observada pelo servidor Iperf3 durante o teste UDP.

```

-----
Accepted connection from 192.168.10.4, port 47912
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.4 port 47142
[ ID] Interval          Transfer      Bitrate      Total Datagrams
[ 5] 0.00-1.00      sec    129 KBytes    1.05 Mbits/sec    91
[ 5] 1.00-2.15      sec    112 KBytes    795 Kbits/sec     79
[ 5] 2.15-3.00      sec    144 KBytes    1.39 Mbits/sec   102
[ 5] 3.00-4.41      sec    120 KBytes    700 Kbits/sec     86
[ 5] 4.41-5.00      sec    124 KBytes    1.72 Mbits/sec     88
[ 5] 5.00-6.45      sec    116 KBytes    655 Kbits/sec     82
[ 5] 6.45-7.15      sec    129 KBytes    1.51 Mbits/sec     90
[ 5] 7.15-8.00      sec    151 KBytes    1.45 Mbits/sec   107
[ 5] 8.00-9.00      sec    127 KBytes    1.04 Mbits/sec     90
[ 5] 9.00-10.00     sec    115 KBytes    939 Kbits/sec     82
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.00     sec    1.24 MBytes    1.04 Mbits/sec  0.000 ms    0/897 (0%) sender
-----
Server listening on 5201 (test #5)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.34: Interface de monitoramento do host PC04 observada pelo servidor Iperf3 durante o teste UDP.

```

Server listening on 5201 (test #4)
-----
Accepted connection from 192.168.10.5, port 33156
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.5 port 34397
[ ID] Interval          Transfer      Bitrate      Total Datagrams
[ 5] 0.00-1.26 sec      127 KBytes    830 Kbits/sec  90
[ 5] 1.26-2.01 sec      120 KBytes    1.31 Mbits/sec  85
[ 5] 2.01-3.01 sec      124 KBytes    1.02 Mbits/sec  88
[ 5] 3.01-4.01 sec      119 KBytes    972 Kbits/sec  84
[ 5] 4.01-5.04 sec      150 KBytes    1.19 Mbits/sec  106
[ 5] 5.04-6.26 sec      105 KBytes    706 Kbits/sec  74
[ 5] 6.26-7.00 sec      151 KBytes    1.66 Mbits/sec  107
[ 5] 7.00-8.02 sec      123 KBytes    991 Kbits/sec  87
[ 5] 8.02-9.25 sec      97.6 KBytes    647 Kbits/sec  69
[ 5] 9.25-10.00 sec     164 KBytes    1.80 Mbits/sec  116
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.01 sec     1.25 MBytes    1.05 Mbits/sec  0.000 ms    0/906 (0%) sender
-----
Server listening on 5201 (test #5)

```

Fonte: Elaborado pelo autor (2026).

Figura A.35: Interface de monitoramento do host PC05 observada pelo servidor Iperf3 durante o teste UDP.

```

Server listening on 5201 (test #4)
-----
Accepted connection from 192.168.10.7, port 49228
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.7 port 47409
[ ID] Interval          Transfer      Bitrate      Total Datagrams
[ 5] 0.00-1.00 sec      129 KBytes    1.05 Mbits/sec  91
[ 5] 1.00-2.39 sec      120 KBytes    707 Kbits/sec  85
[ 5] 2.39-3.11 sec      113 KBytes    1.29 Mbits/sec  80
[ 5] 3.11-4.00 sec      151 KBytes    1.39 Mbits/sec  107
[ 5] 4.00-5.11 sec      123 KBytes    911 Kbits/sec  87
[ 5] 5.11-6.00 sec      132 KBytes    1.21 Mbits/sec  94
[ 5] 6.00-7.00 sec      117 KBytes    962 Kbits/sec  83
[ 5] 7.00-8.13 sec      127 KBytes    924 Kbits/sec  89
[ 5] 8.13-9.23 sec      130 KBytes    969 Kbits/sec  93
[ 5] 9.23-10.01 sec     127 KBytes    1.33 Mbits/sec  89
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.01 sec     1.24 MBytes    1.04 Mbits/sec  0.000 ms    0/898 (0%) sender
-----
Server listening on 5201 (test #5)

```

Fonte: Elaborado pelo autor (2026).

Figura A.36: Interface de monitoramento do host PC06 observada pelo servidor Iperf3 durante o teste UDP.

```

Server listening on 5201 (test #4)
-----
Accepted connection from 192.168.10.8, port 40504
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.8 port 57132
[ ID] Interval          Transfer      Bitrate      Total Datagrams
[ 5] 0.00-1.00 sec      129 KBytes   1.05 Mbits/sec  91
[ 5] 1.00-2.12 sec      123 KBytes   904 Kbits/sec  87
[ 5] 2.12-3.39 sec      130 KBytes   835 Kbits/sec  112
[ 5] 3.39-4.39 sec      77.8 KBytes  642 Kbits/sec  54
[ 5] 4.39-5.01 sec      132 KBytes   1.74 Mbits/sec  93
[ 5] 5.01-6.00 sec      132 KBytes   1.08 Mbits/sec  94
[ 5] 6.00-7.13 sec      123 KBytes   899 Kbits/sec  87
[ 5] 7.13-8.09 sec      127 KBytes   1.08 Mbits/sec  89
[ 5] 8.09-9.00 sec      153 KBytes   1.37 Mbits/sec  108
[ 5] 9.00-10.00 sec     122 KBytes   998 Kbits/sec  87
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.00 sec     1.24 MBytes   1.04 Mbits/sec  0.000 ms    0/902 (0%) sender
-----
Server listening on 5201 (test #5)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.37: Interface de monitoramento do host PC07 observada pelo servidor Iperf3 durante o teste UDP.

```

Server listening on 5201 (test #4)
-----
Accepted connection from 192.168.10.9, port 52348
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.9 port 56426
[ ID] Interval          Transfer      Bitrate      Total Datagrams
[ 5] 0.00-1.00 sec      129 KBytes   1.05 Mbits/sec  91
[ 5] 1.00-2.00 sec      127 KBytes   1.04 Mbits/sec  90
[ 5] 2.00-3.00 sec      129 KBytes   1.05 Mbits/sec  91
[ 5] 3.00-4.00 sec      129 KBytes   1.05 Mbits/sec  91
[ 5] 4.00-5.40 sec      116 KBytes   677 Kbits/sec  82
[ 5] 5.40-6.00 sec      139 KBytes   1.90 Mbits/sec  98
[ 5] 6.00-7.28 sec      102 KBytes   654 Kbits/sec  72
[ 5] 7.28-8.00 sec      156 KBytes   1.76 Mbits/sec  110
[ 5] 8.00-9.00 sec      127 KBytes   1.04 Mbits/sec  90
[ 5] 9.00-10.00 sec     129 KBytes   1.05 Mbits/sec  91
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.00 sec     1.25 MBytes   1.05 Mbits/sec  0.000 ms    0/906 (0%) sender
-----
Server listening on 5201 (test #5)
-----

```

Fonte: Elaborado pelo autor (2026).

Figura A.38: Interface de monitoramento do host PC08 observada pelo servidor Iperf3 durante o teste UDP.

```

Server listening on 5201 (test #4)
-----
Accepted connection from 192.168.10.10, port 51668
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.10 port 51684
[ ID] Interval          Transfer      Bitrate      Total Datagrams
[ 5] 0.00-1.09 sec      129 KBytes    963 Kbits/sec  91
[ 5] 1.09-2.11 sec      105 KBytes    848 Kbits/sec  75
[ 5] 2.11-3.00 sec      151 KBytes    1.39 Mbits/sec 106
[ 5] 3.00-4.36 sec      107 KBytes    648 Kbits/sec  76
[ 5] 4.36-5.81 sec      127 KBytes    717 Kbits/sec  89
[ 5] 5.81-6.00 sec      58.0 KBytes   2.53 Mbits/sec  42
[ 5] 6.00-7.11 sec      219 KBytes    1.62 Mbits/sec 154
[ 5] 7.11-8.32 sec      100 KBytes    681 Kbits/sec  71
[ 5] 8.32-9.00 sec      156 KBytes    1.86 Mbits/sec 110
[ 5] 9.00-10.00 sec     110 KBytes    905 Kbits/sec  79
[ 5] 10.00-10.01 sec     18.4 KBytes   14.4 Mbits/sec  12
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.01 sec     1.25 MBytes   1.05 Mbits/sec 0.000 ms    0/906 (0%) sender
-----
Server listening on 5201 (test #5)

```

Fonte: Elaborado pelo autor (2026).

Figura A.39: Interface de monitoramento do host PC09 observada pelo servidor Iperf3 durante o teste UDP.

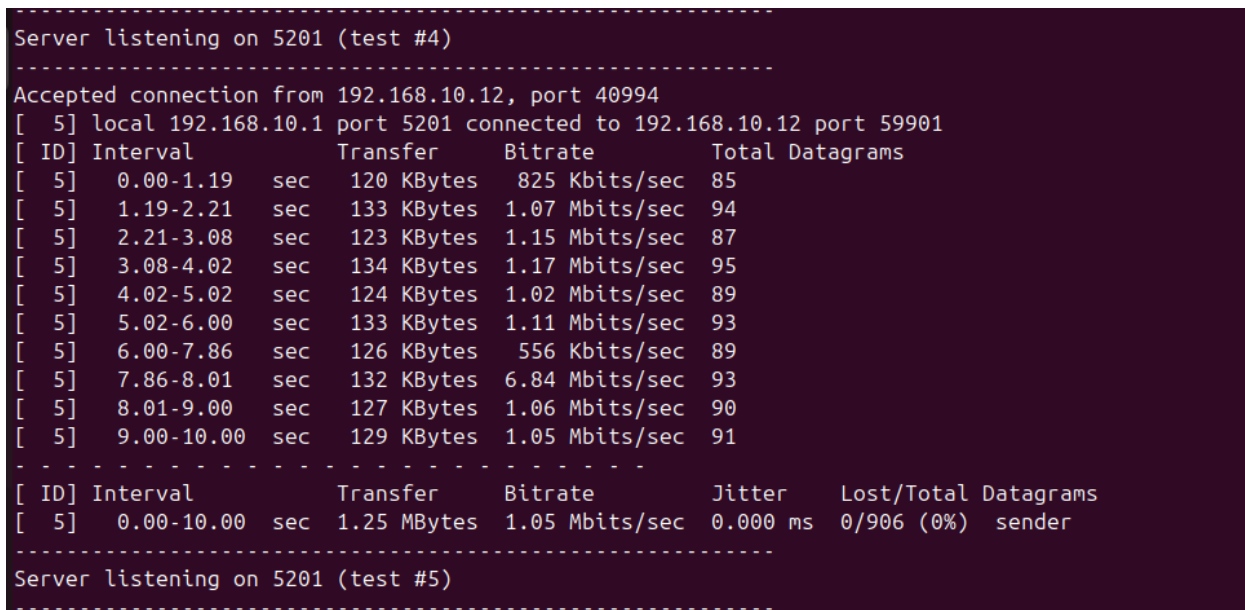
```

Server listening on 5201 (test #4)
-----
Accepted connection from 192.168.10.11, port 40508
[ 5] local 192.168.10.1 port 5201 connected to 192.168.10.11 port 39250
[ ID] Interval          Transfer      Bitrate      Total Datagrams
[ 5] 0.00-1.00 sec      129 KBytes    1.05 Mbits/sec  91
[ 5] 1.00-2.00 sec      127 KBytes    1.04 Mbits/sec  90
[ 5] 2.00-3.00 sec      129 KBytes    1.05 Mbits/sec  91
[ 5] 3.00-4.00 sec      129 KBytes    1.05 Mbits/sec  91
[ 5] 4.00-5.00 sec      127 KBytes    1.04 Mbits/sec  90
[ 5] 5.00-6.00 sec      129 KBytes    1.06 Mbits/sec  91
[ 5] 6.00-7.00 sec      127 KBytes    1.04 Mbits/sec  90
[ 5] 7.00-8.00 sec      129 KBytes    1.05 Mbits/sec  91
[ 5] 8.00-9.00 sec      127 KBytes    1.04 Mbits/sec  90
[ 5] 9.00-10.00 sec     99.0 KBytes    808 Kbits/sec  70
-----
[ ID] Interval          Transfer      Bitrate      Jitter      Lost/Total Datagrams
[ 5] 0.00-10.01 sec     1.22 MBytes   1.02 Mbits/sec 0.000 ms    0/885 (0%) sender
-----
Server listening on 5201 (test #5)

```

Fonte: Elaborado pelo autor (2026).

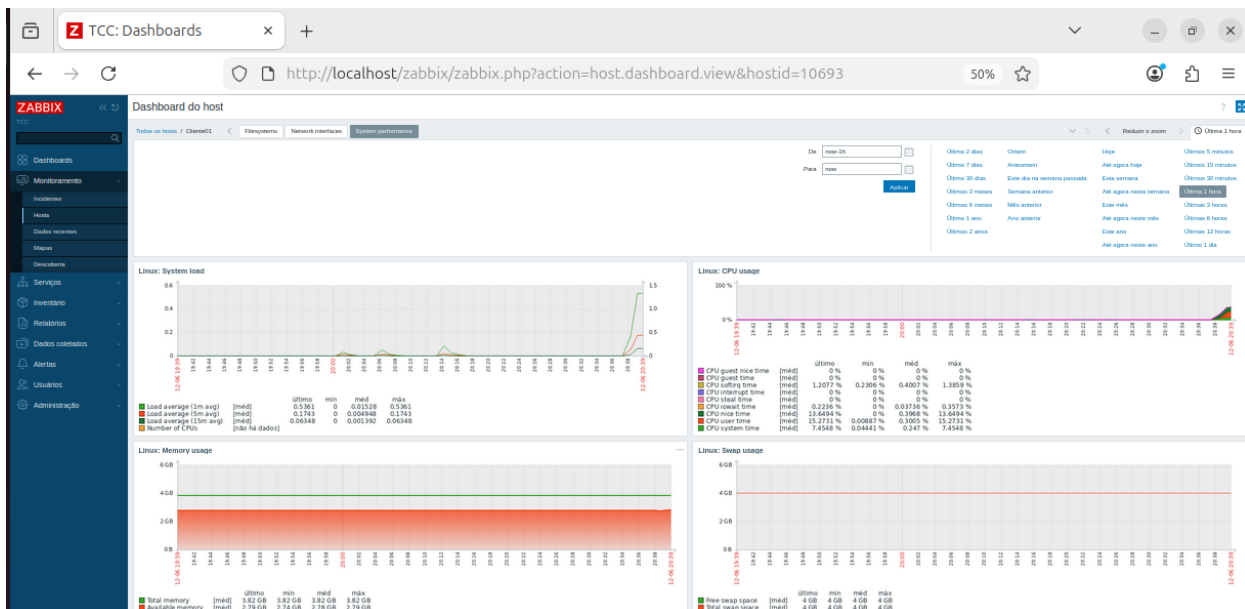
Figura A.40: Interface de monitoramento do host PC10 observada pelo servidor Iperf3 durante o teste UDP.



Fonte: Elaborado pelo autor (2026).

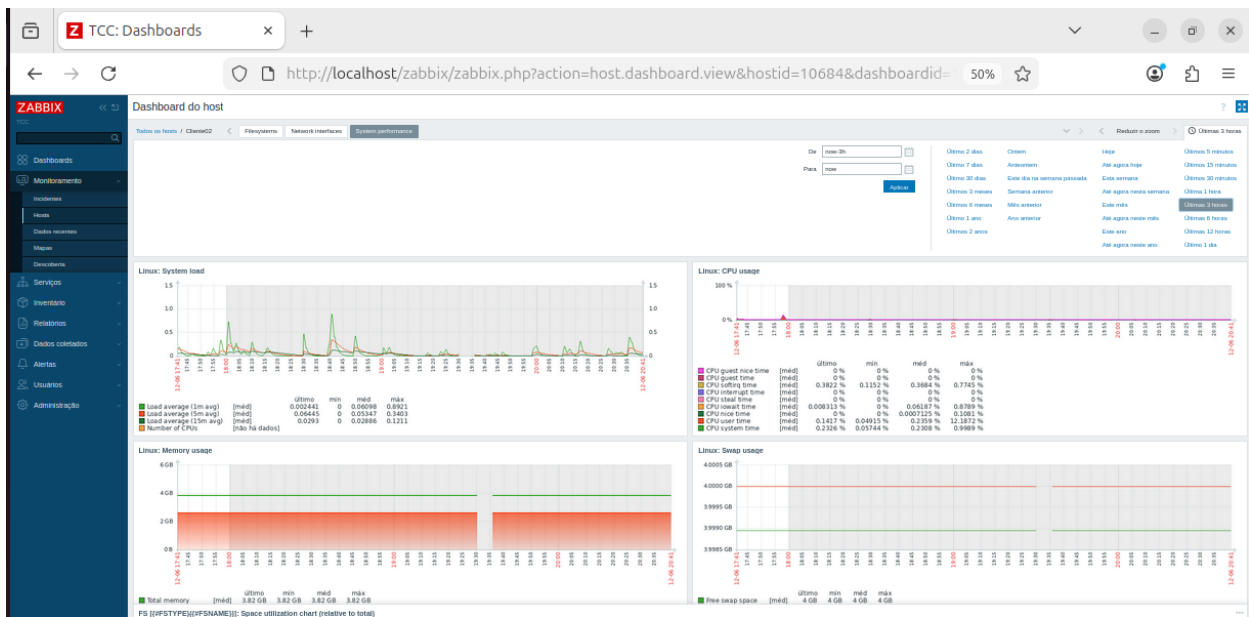
A.5 Métricas de Desempenho Coletadas pelo Zabbix

Figura A.41: Captura de tela das métricas coletadas pelo Zabbix para o host PC01.



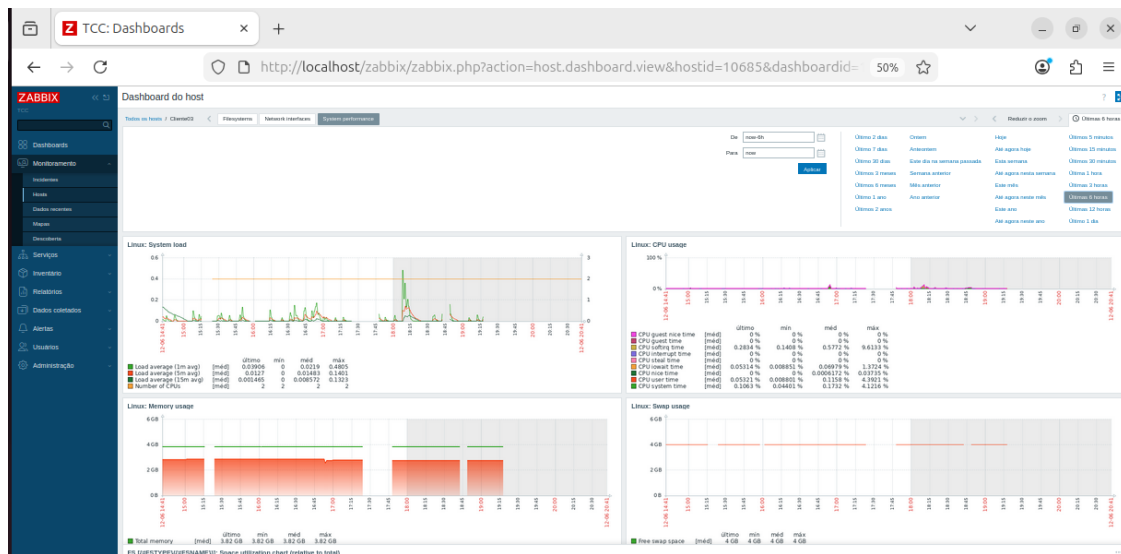
Fonte: Elaborado pelo autor (2026).

Figura A.42: Captura de tela das métricas coletadas pelo Zabbix para o host PC02.



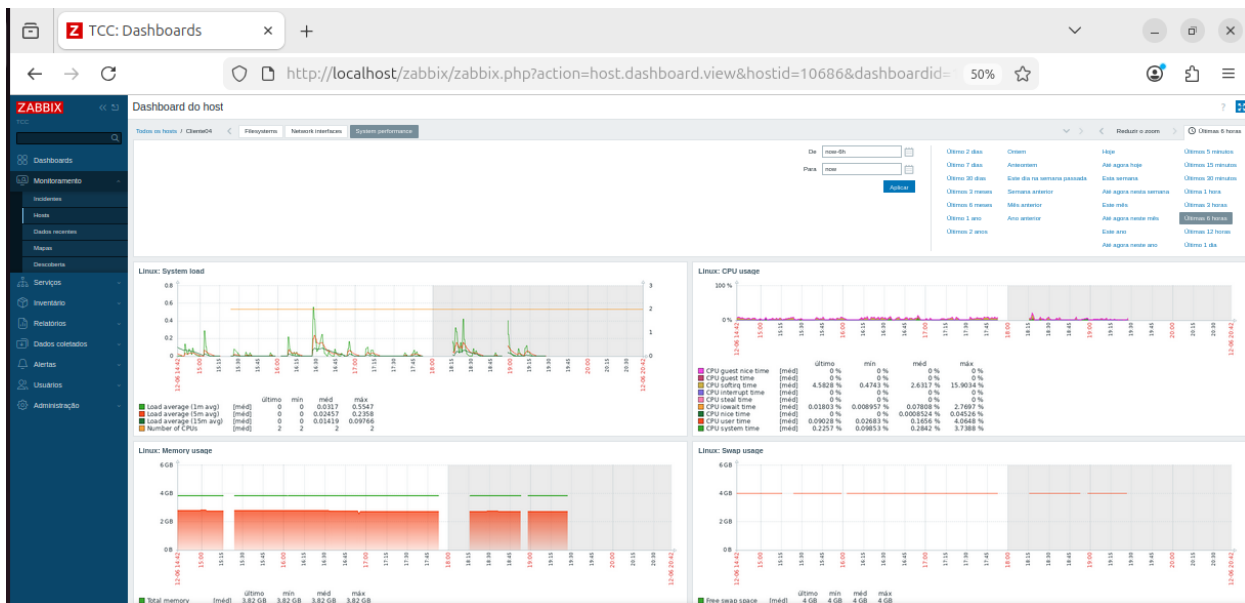
Fonte: Elaborado pelo autor (2026).

Figura A.43: Captura de tela das métricas coletadas pelo Zabbix para o host PC03.



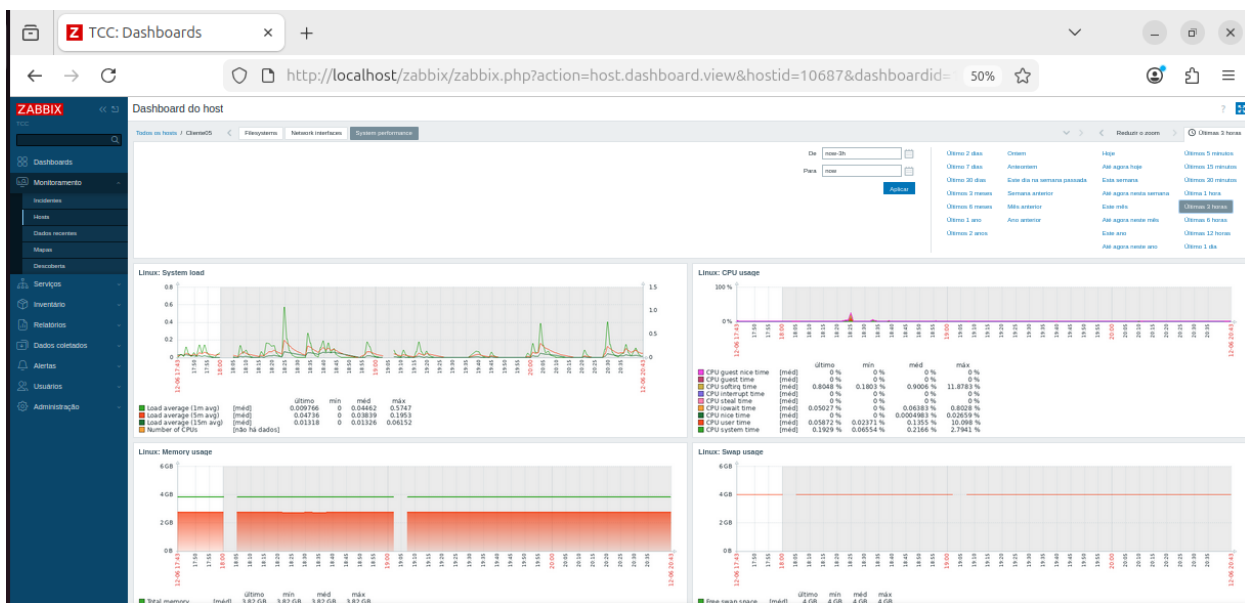
Fonte: Elaborado pelo autor (2026).

Figura A.44: Captura de tela das métricas coletadas pelo Zabbix para o host PC04.



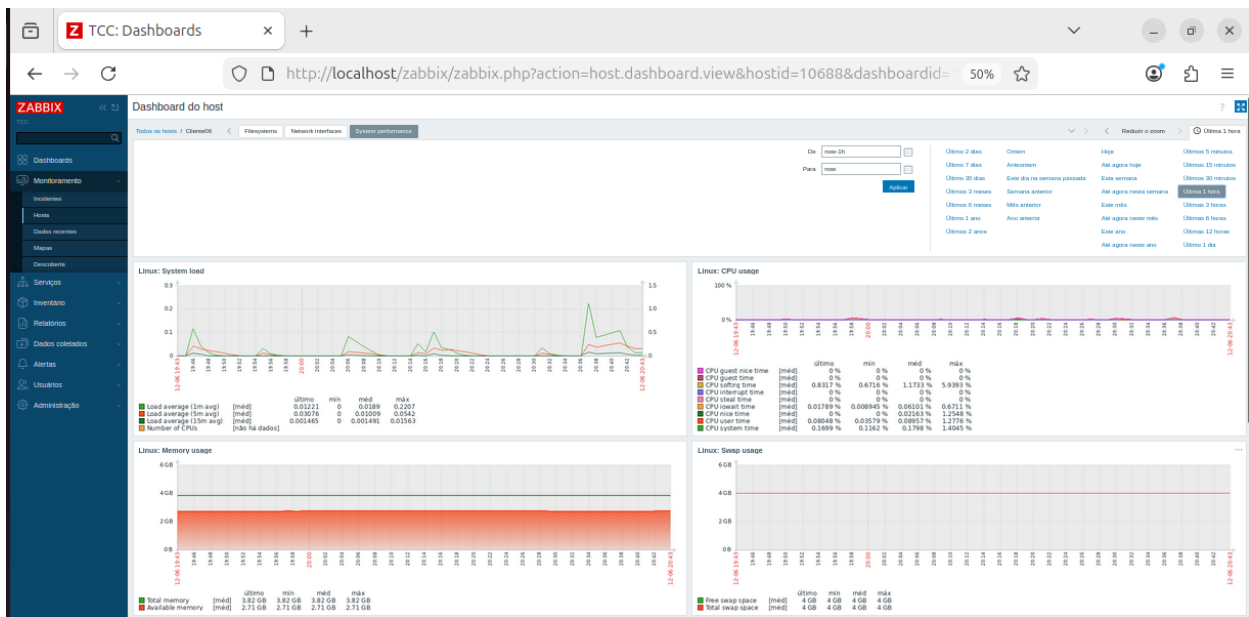
Fonte: Elaborado pelo autor (2026).

Figura A.45: Captura de tela das métricas coletadas pelo Zabbix para o host PC05.



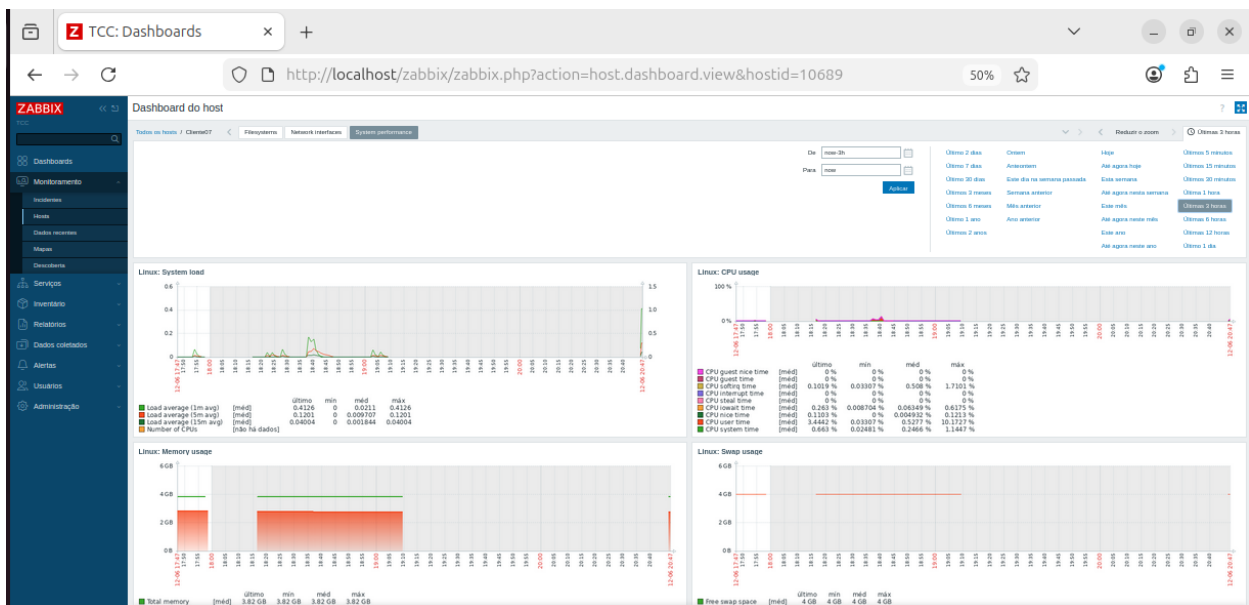
Fonte: Elaborado pelo autor (2026).

Figura A.46: Captura de tela das métricas coletadas pelo Zabbix para o host PC06.



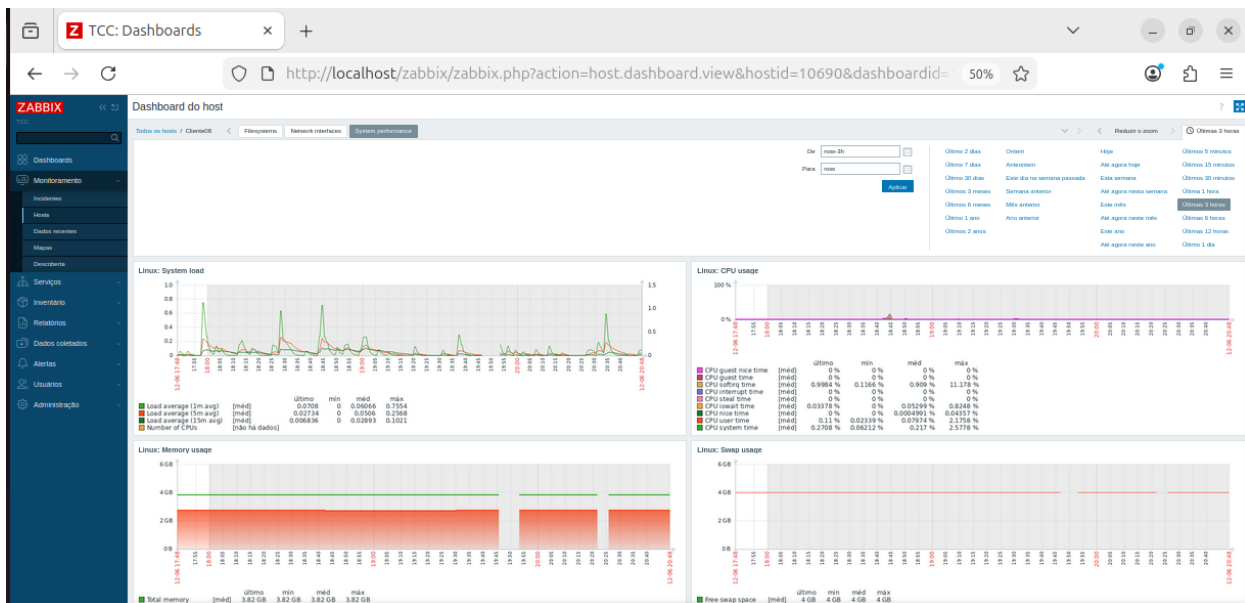
Fonte: Elaborado pelo autor (2026).

Figura A.47: Captura de tela das métricas coletadas pelo Zabbix para o host PC07.



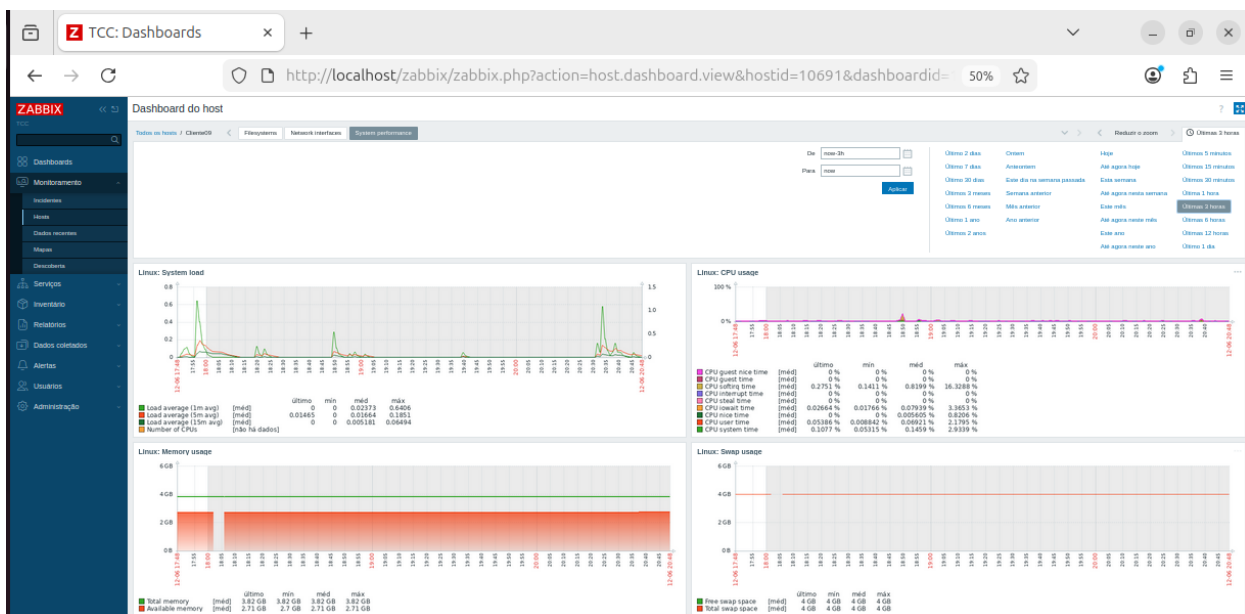
Fonte: Elaborado pelo autor (2026).

Figura A.48: Captura de tela das métricas coletadas pelo Zabbix para o host PC08.



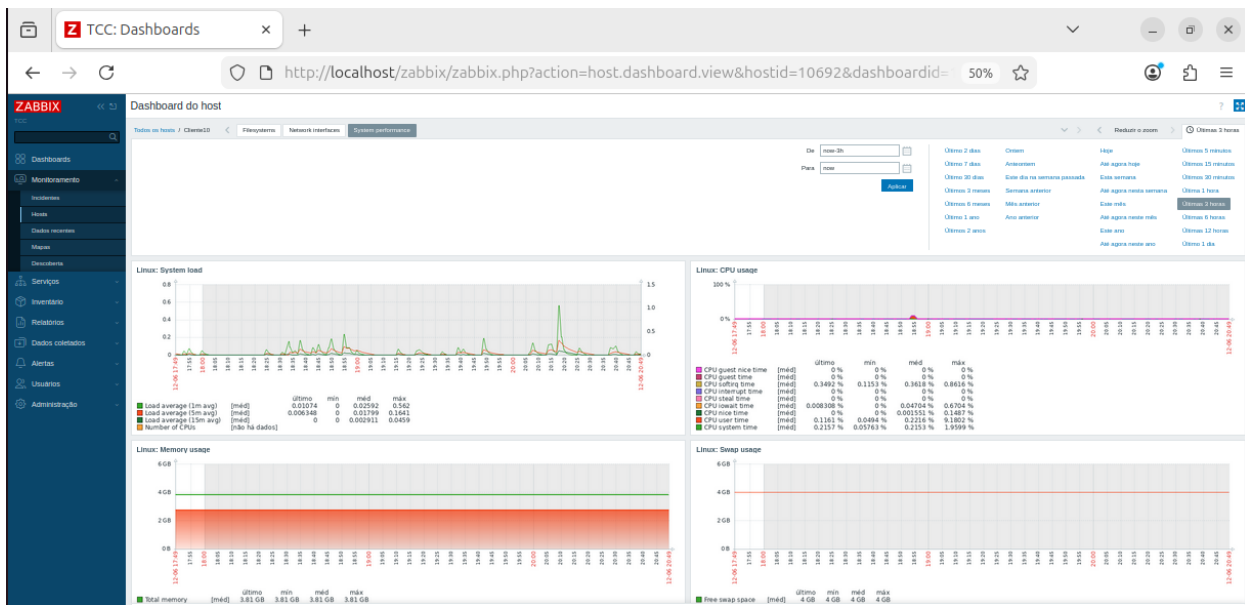
Fonte: Elaborado pelo autor (2026).

Figura A.49: Captura de tela das métricas coletadas pelo Zabbix para o host PC09.



Fonte: Elaborado pelo autor (2026).

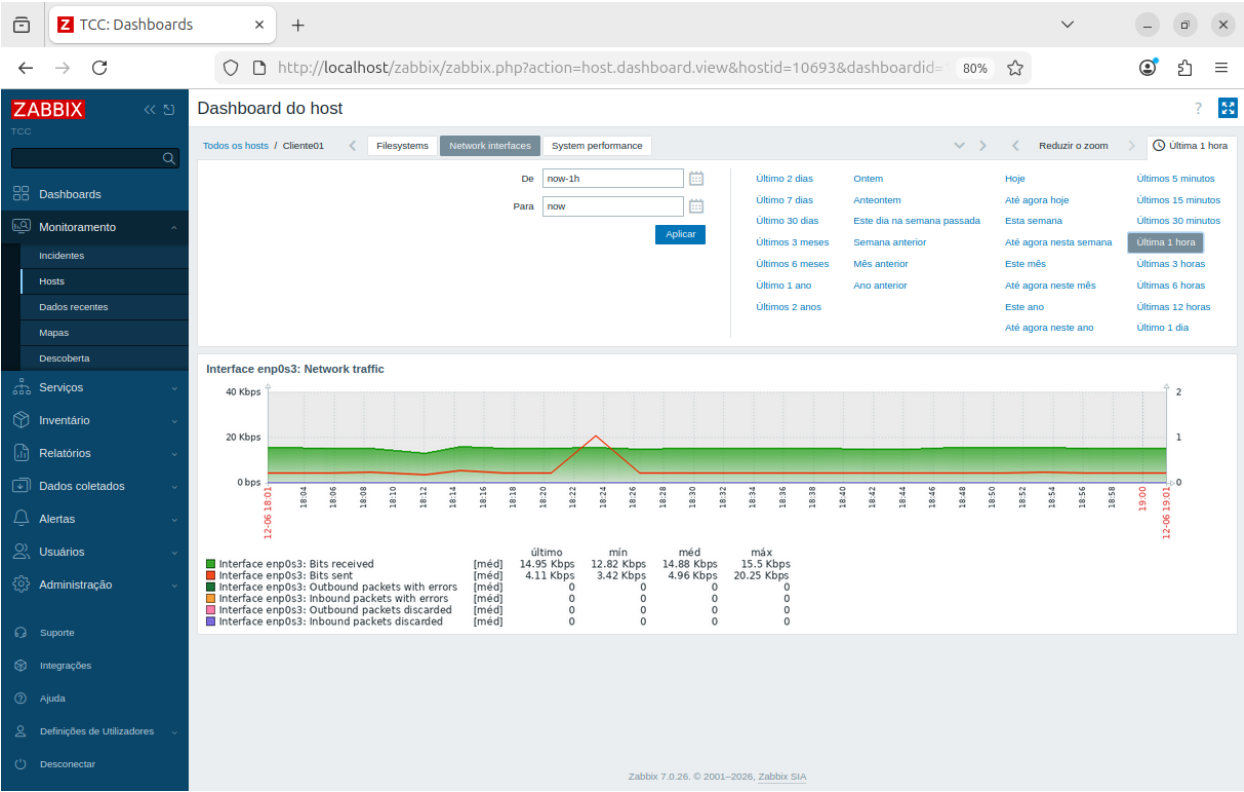
Figura A.50: *Captura de tela das métricas coletadas pelo Zabbix para o host PC10.*



Fonte: Elaborado pelo autor (2026).

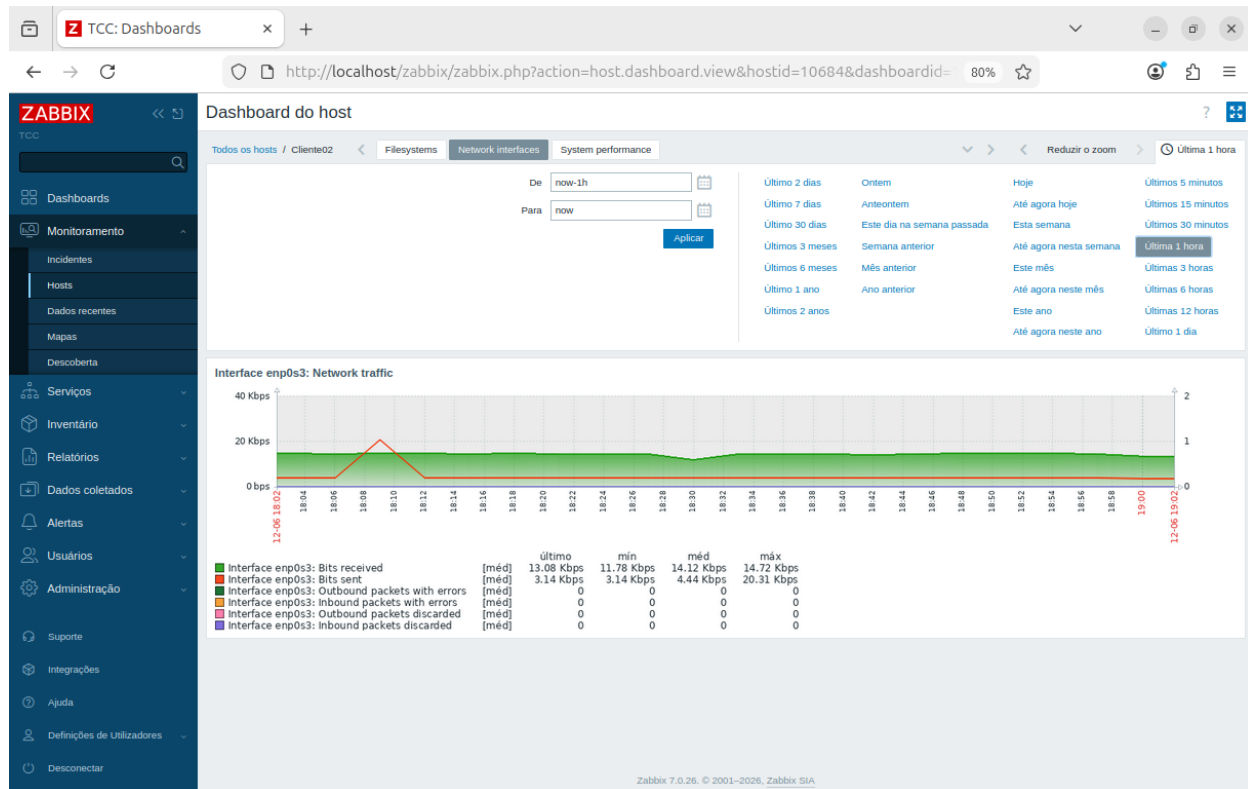
A.6 Resultados do Monitoramento de Rede Obtidos com o Zabbix

Figura A.51: Monitoramento do desempenho da interface de rede do host PC01 por meio do Zabbix.



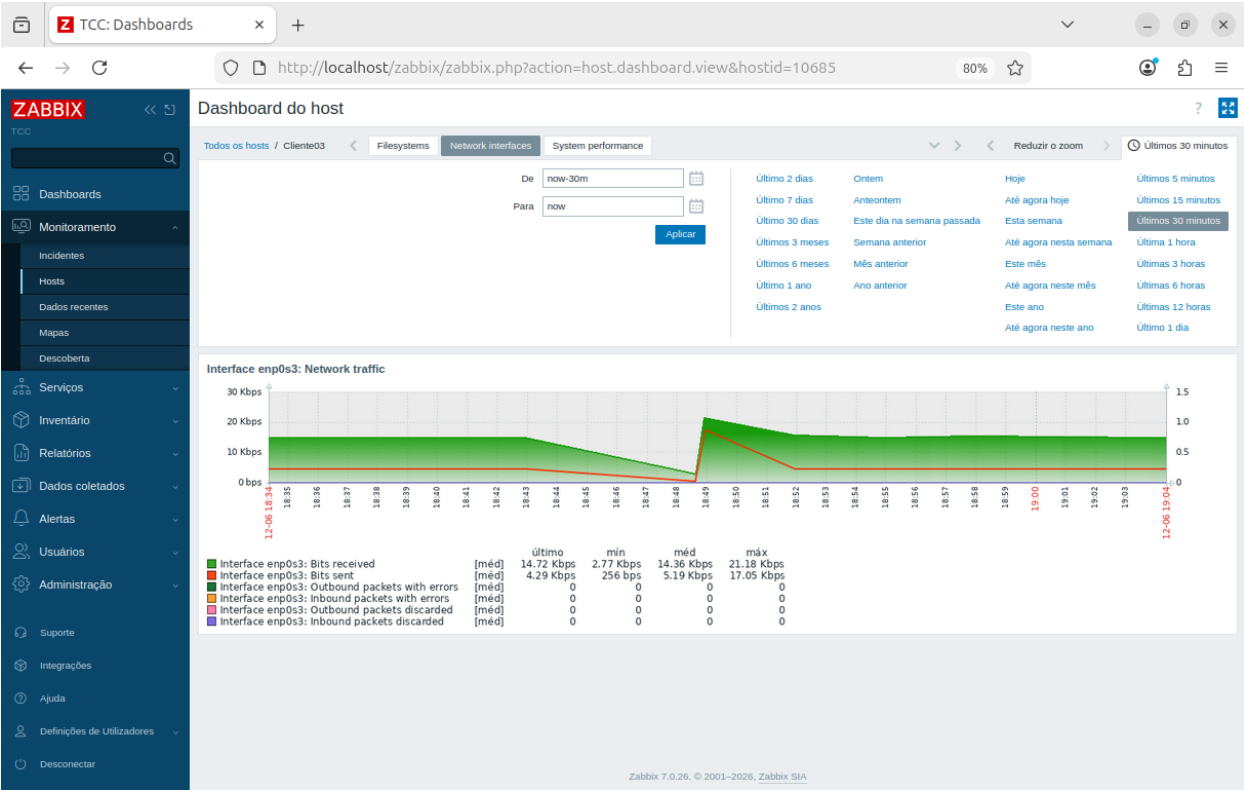
Fonte: Elaborado pelo autor (2026).

Figura A.52: Monitoramento do desempenho da interface de rede do host PC02 por meio do Zabbix.



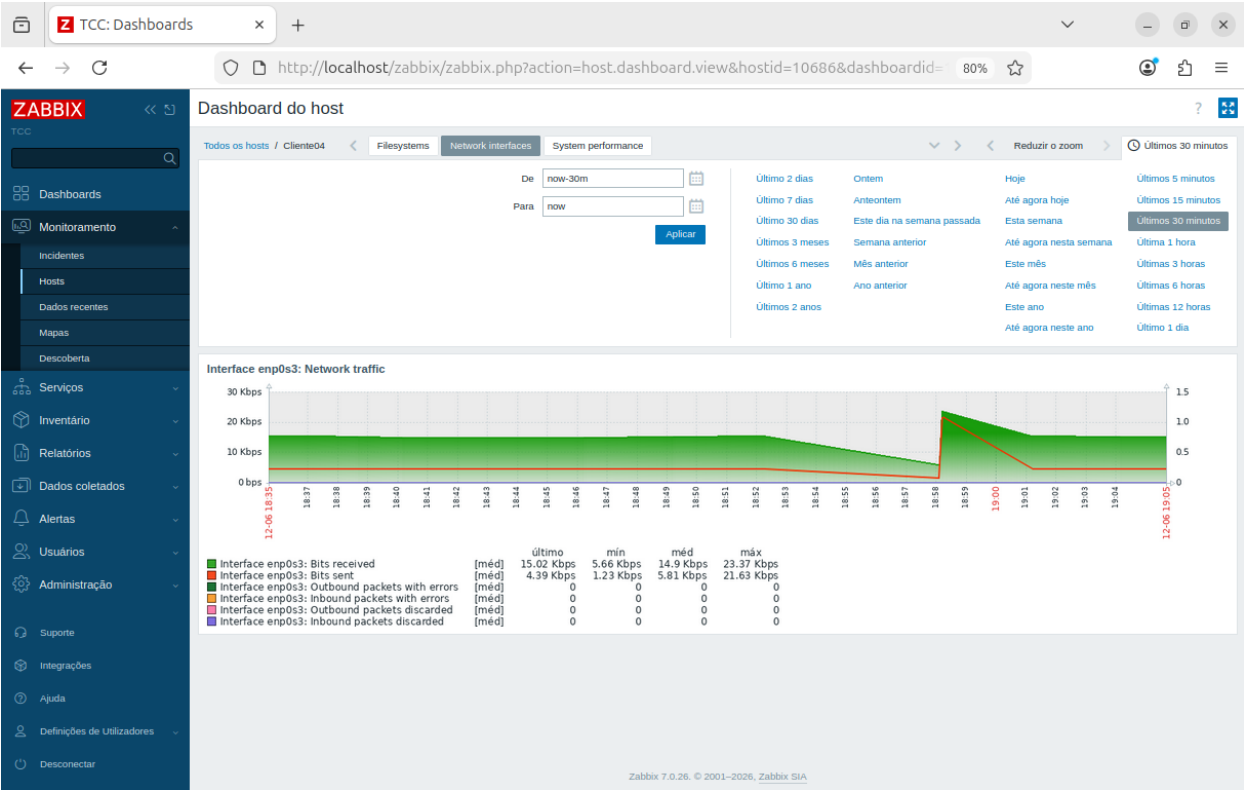
Fonte: Elaborado pelo autor (2026).

Figura A.53: Monitoramento do desempenho da interface de rede do host PC03 por meio do Zabbix.



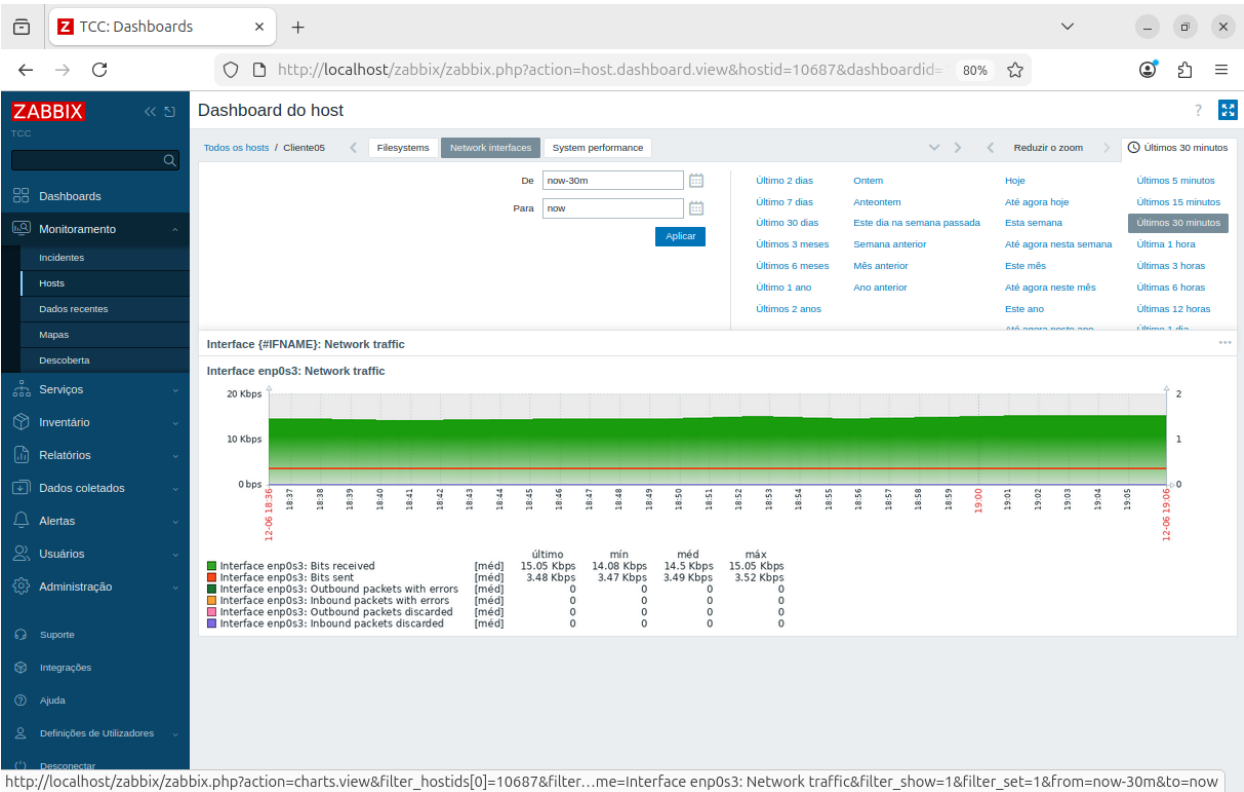
Fonte: Elaborado pelo autor (2026).

Figura A.54: *Monitoramento do desempenho da interface de rede do host PC04 por meio do Zabbix.*



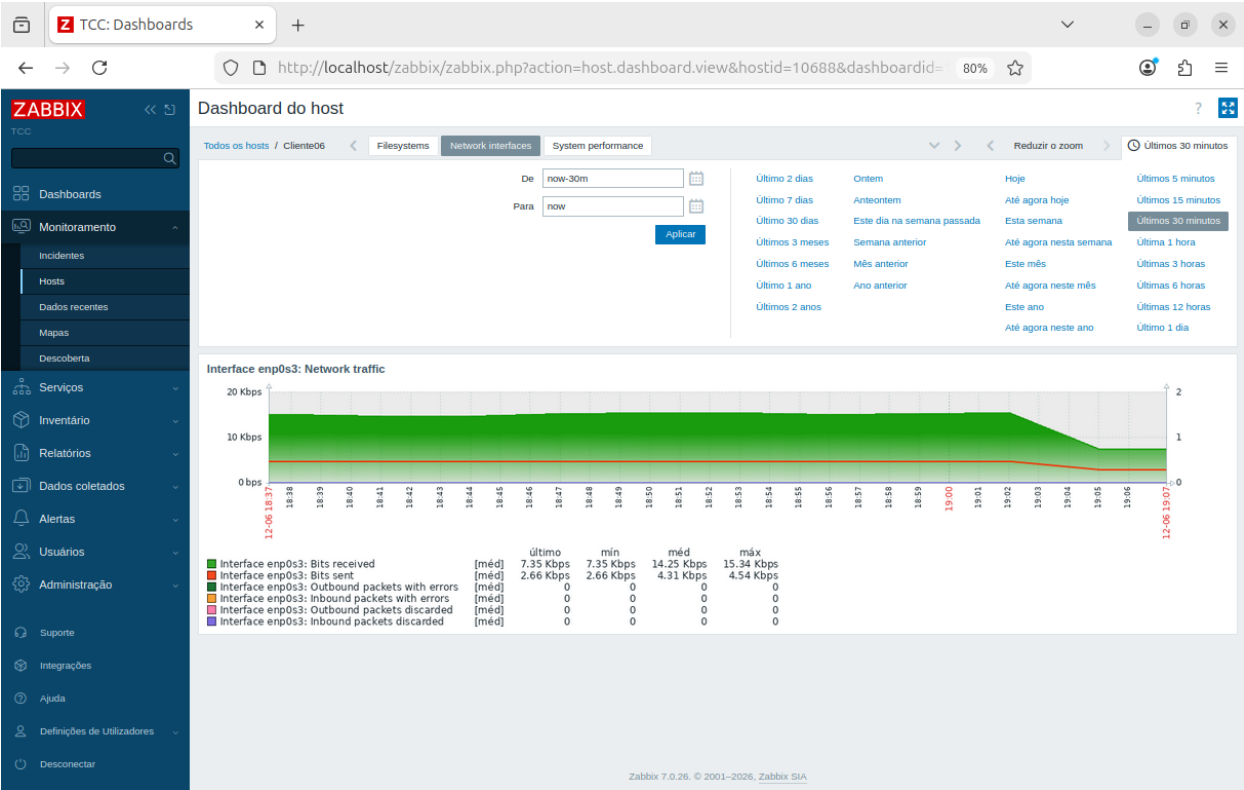
Fonte: Elaborado pelo autor (2026).

Figura A.55: Monitoramento do desempenho da interface de rede do host PC05 por meio do Zabbix.



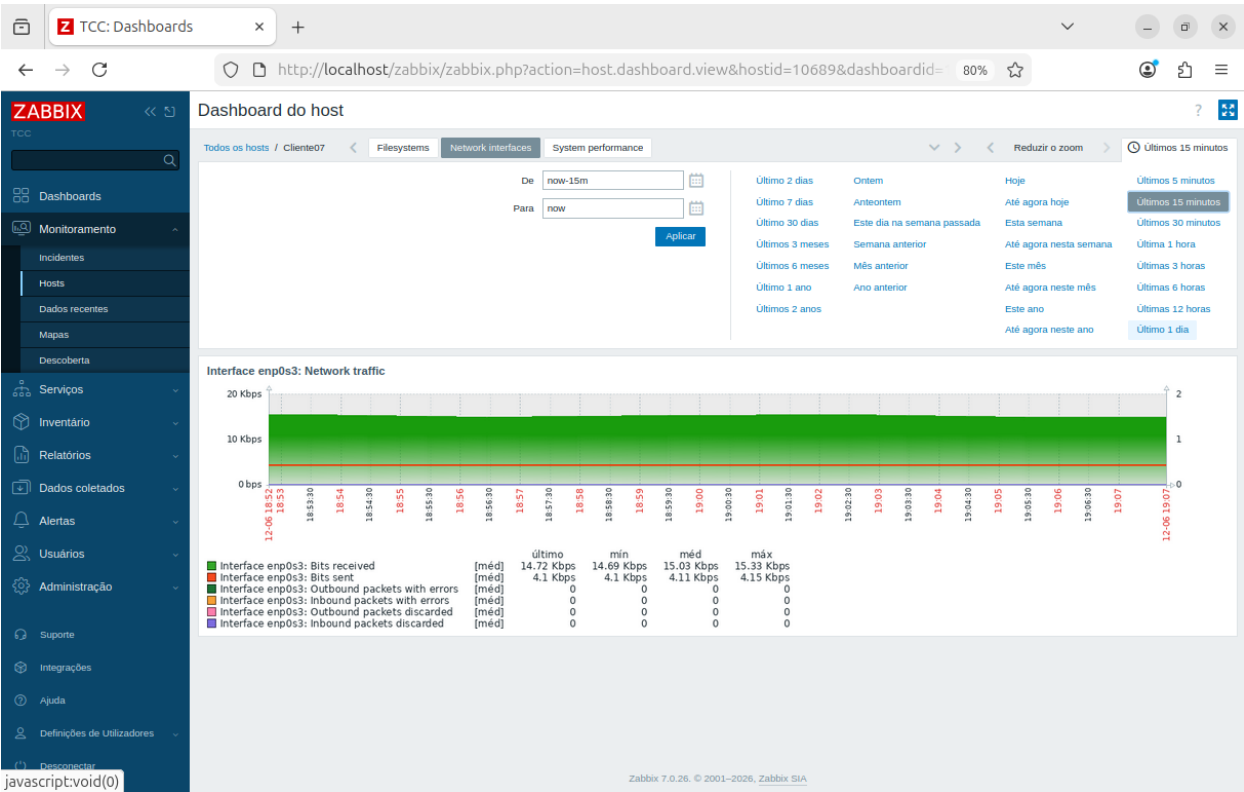
Fonte: Elaborado pelo autor (2026).

Figura A.56: Monitoramento do desempenho da interface de rede do host PC06 por meio do Zabbix.



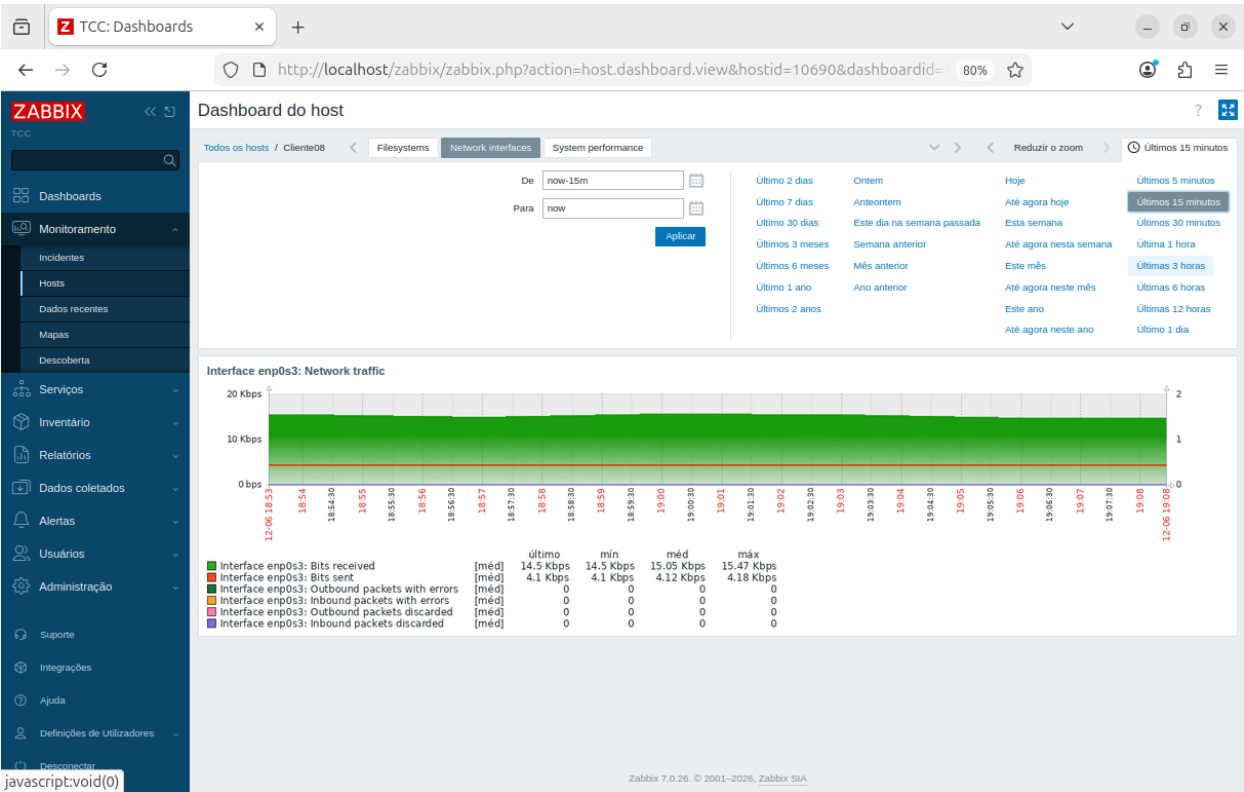
Fonte: Elaborado pelo autor (2026).

Figura A.57: Monitoramento do desempenho da interface de rede do host PC07 por meio do Zabbix.



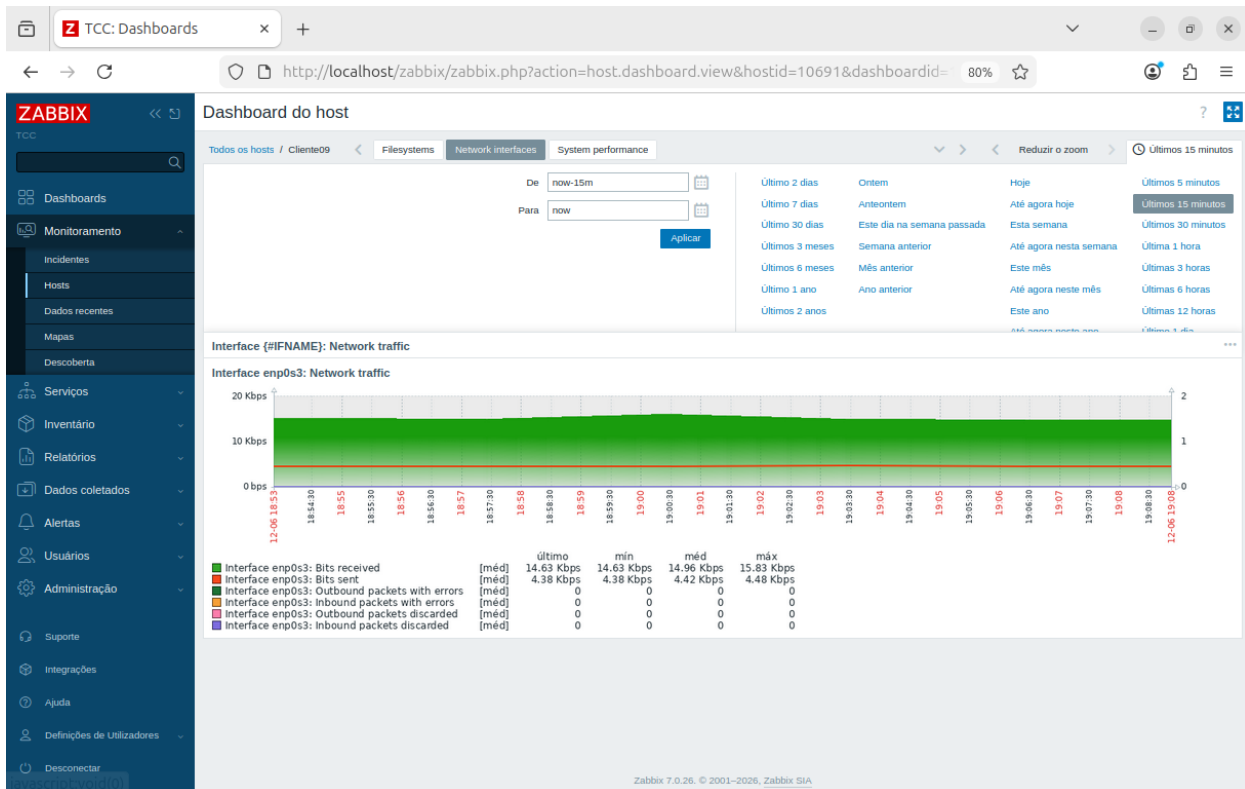
Fonte: Elaborado pelo autor (2026).

Figura A.58: Monitoramento do desempenho da interface de rede do host PC08 por meio do Zabbix.



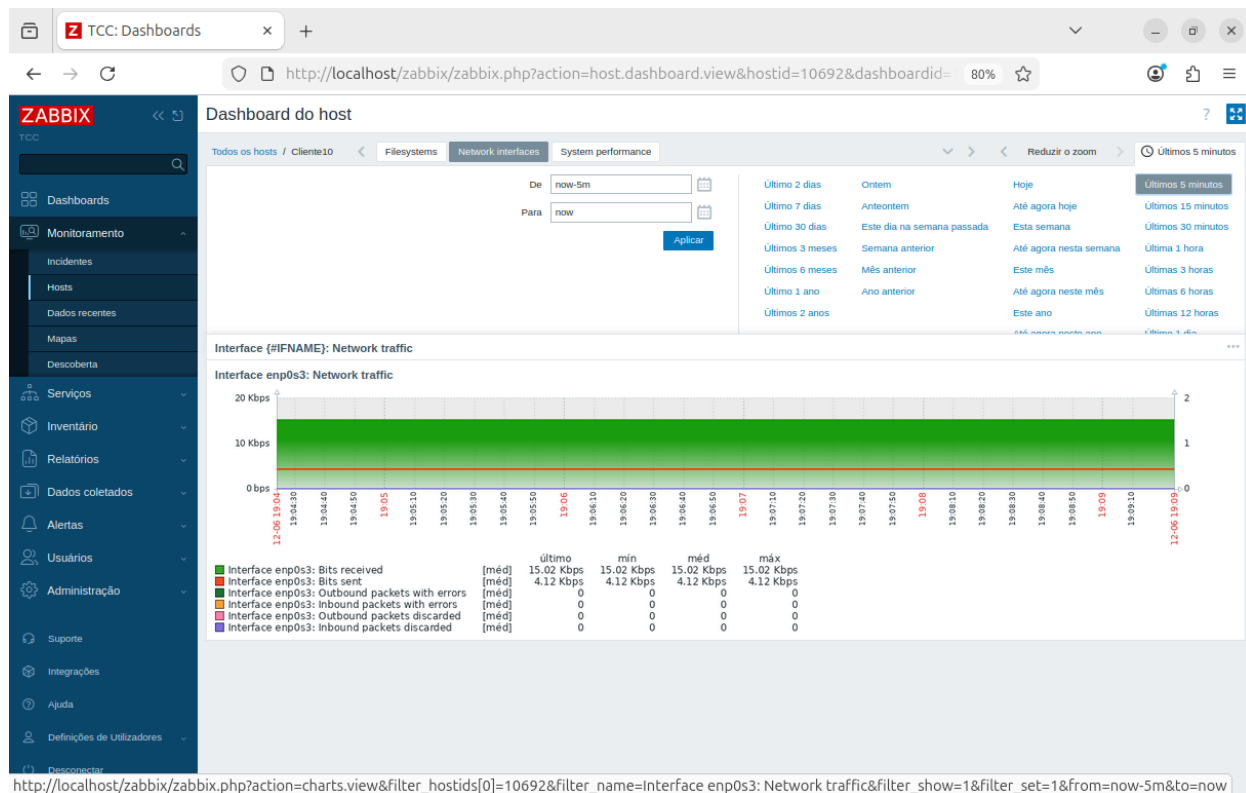
Fonte: Elaborado pelo autor (2026).

Figura A.59: Monitoramento do desempenho da interface de rede do host PC09 por meio do Zabbix.



Fonte: Elaborado pelo autor (2026).

Figura A.60: Monitoramento do desempenho da interface de rede do host PC10 por meio do Zabbix.



Fonte: Elaborado pelo autor (2026).

Referências Bibliográficas

[Ahmad, Saeed e Khan 2023] AHMAD, M.; SAEED, F.; KHAN, M. Virtualization technologies in modern computing environments: A review. *Journal of Cloud Computing*, v. 12, n. 1, 2023. 8

[Antônio 2024] ANTÔNIO, D. *Entendendo TCP e UDP de Forma Simples*. 2024. <<https://www.linkedin.com/pulse/entendendo-tcp-e-udp-de-forma-simples-domingos-antonio-0kxhf/>>. Acesso em: 2026. 12

[Borges, Cardoso e Bussador 2023] BORGES, J.; CARDOSO, L.; BUSSADOR, A. Redes de computadores e simulação: Um estudo comparativo entre softwares. In: *Anais do XX Congresso Latino-Americano de Software Livre e Tecnologias Abertas*. Porto Alegre, RS, Brasil: SBC, 2023. p. 198–201. ISSN 0000-0000. Acesso em: 2026. Disponível em: <<https://sol.sbc.org.br/index.php/latinoware/article/view/26109>>. 1

[Buyya e Srirama 2023] BUYYA, R.; SRIRAMA, S. N. *Cloud Computing: Principles and Paradigms*. 2. ed. [S.l.]: Wiley, 2023. 9, 10

[Canonical Ltd. 2024] Canonical Ltd. *Ubuntu 24.04 LTS Documentation*. 2024. Acesso em: 11 jun. 2026. Disponível em: <<https://ubuntu.com/server/docs>>. 17, 18

[Corujo *et al.* 2023] CORUJO, D. *et al.* An empirical assessment of the contribution of 5g in vertical industries: A case for the transportation sector. *IEEE Access*, 2023. 2

[Czajkowski *et al.* 2022] CZAJKOWSKI, K. *et al.* Network testbeds and experimental platforms for performance evaluation in modern networks. *Sensors*, MDPI, v. 22, n. 18, p. 1–25, 2022. 14

[Ferreira, Silva e Santos 2024] FERREIRA, J. R.; SILVA, M. A.; SANTOS, P. H. Performance analysis of network throughput in virtualized environments. *Journal of Network Systems*, v. 18, n. 2, p. 45–58, 2024. 13

[Force 2022] FORCE, I. E. T. *Transmission Control Protocol (TCP)*. 2022. Acesso em: 11 jun. 2026. Disponível em: <<https://www.ietf.org/>>. 11

[Forouzan 2021] FOROUZAN, B. A. *Data Communications and Networking*. 6. ed. [S.l.]: McGraw-Hill Education, 2021. 7, 11

[Gomez *et al.* 2023] GOMEZ, J. *et al.* A survey on network simulators, emulators, and testbeds used for research and education. *Computer Networks*, v. 237, p. 110054, 2023. 10

[Hassan e Nabi 2022] HASSAN, R.; NABI, M. *Computer Networks and Cyber Security*. [S.l.]: CRC Press, 2022. 11

- [Henderson *et al.* 2022] HENDERSON, T. R. *et al.* *The ns-3 Network Simulator*. 2022. Ns-3 Project Documentation. 10
- [Huawei Technologies Co., Ltd. 2022] Huawei Technologies Co., Ltd. *Data Communications and Network Technologies*. Singapore: Springer, 2022. 7, 8
- [Kurose e Ross 2017] KUROSE, J. F.; ROSS, K. W. *Computer Networking: A Top-Down Approach*. [S.l.]: Pearson, 2017. 9, 10, 11, 12, 15
- [Kurose e Ross 2021] KUROSE, J. F.; ROSS, K. W. *Redes de Computadores e a Internet: Uma Abordagem Top-Down*. 8. ed. São Paulo: Pearson, 2021. 6
- [Lantz, Heller e McKeown 2021] LANTZ, B.; HELLER, B.; MCKEOWN, N. A network in a laptop: Rapid prototyping for software-defined networks. In: *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*. [S.l.: s.n.], 2021. 10
- [Medhi e Ramasamy 2021] MEDHI, D.; RAMASAMY, K. *Network Routing: Algorithms, Protocols, and Architectures*. 3. ed. [S.l.]: Morgan Kaufmann, 2021. 12, 13
- [Merces *et al.* 2026] MERCES, J. M. O. *et al.* Evaluating the limits of wi-fi 5 and wi-fi 6 networks for industrial video applications. 2026. 8, 13, 15
- [Moina-Rivera *et al.* 2024] MOINA-RIVERA, W. *et al.* Cloud media video encoding: review and challenges. *Multimedia Tools and Applications*, v. 83, p. 81231–81278, 2024. 7
- [Morabito, Kjällman e Komu 2023] MORABITO, R.; KJÄLLMAN, J.; KOMU, M. Hypervisors vs. lightweight virtualization: A performance comparison. *IEEE Transactions on Cloud Computing*, 2023. 9
- [Nagori 2024] NAGORI, A. I. *How to Keep and Check Memory Usage History*. 2024. Disponível em: <<https://www.baeldung.com/linux/memory-track-usage-history>>. 14
- [Nguyen 2026] NGUYEN, D. *TCP vs UDP: Differences, Reliability vs Latency, and When to Use Each Transport Protocol*. 2026. Acesso em: 11 jun. 2026. Disponível em: <<https://netalith.com/blogs/networking-fundamentals/tcp-vs-udp-key-differences-when-each-is-used>>. 11
- [Olifer e Olifer 2022] OLIFER, N.; OLIFER, V. *Computer Networks: Principles, Technologies and Protocols for Network Design*. 2. ed. [S.l.]: Wiley, 2022. 13
- [Oracle 2024] ORACLE. *Monitoring CPU Usage*. 2024. Disponível em: <<https://docs.oracle.com/en/operating-systems/oracle-linux/6/admin/monitor-cpu-use.html>>. 14
- [Oracle 2024] ORACLE. *Monitoring the System and Optimizing Performance*. 2024. Disponível em: <<https://docs.oracle.com/en/operating-systems/oracle-linux/7/monitoring/monitoring-MonitoringtheSystemandOptimizingPerformance.html>>. 14
- [Oracle Corporation 2024] Oracle Corporation. *Oracle VM VirtualBox User Manual*. [S.l.], 2024. Version 7.x. 8, 9, 10
- [Oracle Corporation 2025] Oracle Corporation. *Oracle VM VirtualBox User Manual*. [S.l.], 2025. Acesso em: 27 maio 2026. Disponível em: <<https://www.virtualbox.org/manual/UserManual.html>>. 25

- [Proxmox Server Solutions GmbH 2024] Proxmox Server Solutions GmbH. *Proxmox VE Administration Guide*. [S.l.], 2024. 9
- [Pádua 2023] PáDUA, V. *Modelo TCP/IP: como funciona a pilha de protocolos da internet*. 2023. <<https://tecnoblog.net/responde/o-que-e-tcp-ip/>>. Acesso em: 2026. 7
- [QNAP Brasil 2023] QNAP Brasil. *O que é virtualização?* 2023. <https://www.gta.ufrj.br/ensino/eel879/trabalhos_vf_2023_2/Grupo_02/>. Acesso em: 2026. 9
- [Rattanaopas e Boonchuay 2017] RATTANAOPAS, A.; BOONCHUAY, C. A high performance network virtualization architecture with bandwidth guarantees. In: *International Conference on Information Networking*. [S.l.: s.n.], 2017. 46
- [Silva et al. 2023] SILVA, F. A. et al. Uma perspectiva analítica para avaliação de desempenho no aprendizado federado. In: SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. *Anais do Simpósio Brasileiro de Computação em Nuvem e Sistemas Distribuídos (SBCC)*. Brasil, 2023. 2
- [Silva e Silva 2024] SILVA, W.; SILVA, R. Implementação de monitoramento de rede de dados com zabbix e grafana: um estudo de caso. *P2P E INOVAÇÃO*, 04 2024. Disponível em: <https://www.researchgate.net/publication/380230043_IMPLEMENTACAO_DE_MONITORAMENTO_DE_REDE_DE_DADOS_COM_ZABBIX_E_GRAFANA_um_estudo_de_caso>. 6, 17
- [Souza et al. 2023] SOUZA, A. M. d. et al. Dispositivos, eu escolho vocês: Seleção de clientes adaptativa para comunicação eficiente em aprendizado federado. In: SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. *Anais do Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*. Brasil, 2023. 2
- [Stallings 2020] STALLINGS, W. *Data and Computer Communications*. 11. ed. [S.l.]: Pearson, 2020. 7, 11
- [Stockmayer 2023] STOCKMAYER, A. *A Virtual Testbed Orchestration (VITO) System for Education, Performance Evaluation, and Network Function Virtualization*. Tese (Doutorado) — Eberhard Karls Universität Tübingen, Tübingen, Germany, 2023. Acesso em: 2026. 1
- [Tanaka 2005] TANAKA, B. K. Monitoring virtual memory with vmstat. *Linux Journal*, n. 139, 2005. 14
- [Tanenbaum e Wetherall 2011] TANENBAUM, A. S.; WETHERALL, D. J. *Computer Networks*. [S.l.]: Pearson, 2011. 8, 10, 11, 13
- [Tomar et al. 2022] TOMAR, P. et al. Cmt-sctp and mptcp multipath transport protocols: A comprehensive review. *Electronics*, v. 11, n. 15, p. 2384, 2022. 11
- [Trabelsi, Fourati e Chen 2024] TRABELSI, N.; FOURATI, L. C.; CHEN, C. S. Interference management in 5g and beyond networks: A comprehensive survey. *Computer Networks*, v. 239, p. 110159, 2024. 7
- [Vandenberghe et al. 2021] VANDENBERGHE, W. et al. Experimental research in future internet testbeds: Challenges and opportunities. *Future Internet*, MDPI, v. 13, n. 5, p. 1–22, 2021. 14

[VMware Inc. 2024] VMware Inc. *What is a Hypervisor?* 2024. Disponível em: <https://www.vmware.com/topics/glossary/content/hypervisor>. 8

[Wang, Li e Zhang 2022] WANG, J.; LI, P.; ZHANG, W. Virtualized network testbeds: Design, deployment and performance evaluation. *Future Internet*, MDPI, v. 14, n. 7, p. 1–19, 2022. 15

[Yadav e Paul 2023] YADAV, A.; PAUL, S. K. Virtualization technologies and their impact on cloud computing: A review. *Journal of Cloud Computing*, v. 12, n. 1, 2023. 8, 9

[Zabbix LLC 2024] Zabbix LLC. *Zabbix Documentation*. 2024. Acesso em: 11 jun. 2026. Disponível em: <<https://www.zabbix.com/documentation/current/en/manual>>. 17

[Zhang, Liu e Wang 2023] ZHANG, Y.; LIU, H.; WANG, X. Recent advances in computer networks and cloud-based applications. *IEEE Access*, v. 11, p. 45872–45889, 2023. 6

[Zhu, Chen e Li 2022] ZHU, M.; CHEN, Y.; LI, X. Analysis of packet loss impact on network performance in modern communication systems. *IEEE Access*, IEEE, v. 10, p. 84567–84578, 2022. 13

[Zieliński 2023] ZIELIŃSKI, B. Assessment of iperf as a tool for lan throughput prediction. *International Journal of Electronics and Telecommunications*, v. 69, n. 3, p. 523–528, 2023. 16, 46