PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA POLITÉCNICA E DE ARTES
GRADUAÇÃO EM ENGENHARIA DA COMPUTAÇÃO

Implantação de *firewall* em *Raspberry Pi*

HEBERT CORDEIRO DE SOUZA

GOIÂNIA
2023

HEBERT CORDEIRO DE SOUZA

Implantação de *firewall* em *Raspberry Pi*

Trabalho de Conclusão de Curso apresentado à Escola Politécnica e de Artes da Pontifícia Universidade Católica de Goiás, como parte dos requisitos para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador(a): Prof. M.E.E. Marcelo Antonio Adad de Araújo.

GOIÂNIA
2023

HEBERT CORDEIRO DE SOUZA

Implantação de *firewall* em *Raspberry Pi*

Este trabalho de Conclusão de Curso julgado adequado para obtenção o título de Bacharel em Engenharia de Computação, e aprovado em sua forma final pela Escola Politécnica, da Pontifícia Universidade Católica de Goiás, em /02/2023.

_____

Prof. MSc. Ludmilla Reis Pinheiro dos Santos

Coordenadora de Trabalho de Conclusão de Curso

Banca examinadora:

_____

Orientador: Prof. Marcelo Antonio Adad de Araújo, M.E.E.

_____

Prof. Carlos Alexandre Ferreira de Lima, M.E.E.

_____

Dr. Phd. Prof. Nilson Cardoso Amaral

GOIÂNIA

2021

RESUMO

Este trabalho apresenta a implantação de um *firewall* utilizando um *Raspberry Pi* 4 como plataforma, com o objetivo de explorar suas funcionalidades e avaliar sua efetividade como uma solução de segurança de rede em um ambiente de pequeno porte. São abordados os conceitos teóricos de *firewall*, os tipos disponíveis e considerações para implantação. O processo de configuração do *Raspberry Pi* como *firewall* é descrito, incluindo a instalação do sistema operacional, configuração da interface de rede e do *software* de *firewall*. São discutidos os prós e contras da utilização do *Raspberry Pi* como plataforma de *firewall* em relação a outras soluções, bem como considerações de segurança. Os resultados da implementação são apresentados, juntamente com as conclusões sobre a efetividade do *Raspberry Pi* como plataforma de segurança de rede em um ambiente de pequeno porte. Este trabalho contribui para a compreensão das funcionalidades de um *firewall* e das possibilidades de utilização do *Raspberry Pi* como plataforma de segurança de rede.

Palavras-chave: Segurança de rede, *firewall*, *Raspberry Pi*, implantação, efetividade.

ABSTRACT

This work presents the implementation of a firewall using a Raspberry Pi 4 as the platform, aiming to explore its functionalities and assess its effectiveness as a network security solution in a small-scale environment. The theoretical concepts of firewalls, available types, and deployment considerations are discussed. The process of configuring the Raspberry Pi as a firewall is described, including the installation of the operating system, configuration of the network interface, and firewall software. The pros and cons of using the Raspberry Pi as a firewall platform are discussed in comparison to other solutions, as well as security considerations. The implementation results are presented, along with conclusions regarding the effectiveness of the Raspberry Pi as a network security platform in a small-scale environment. This work contributes to understanding firewall functionalities and the possibilities of using the Raspberry Pi as a network security platform.

Keywords: Network security, firewall, Raspberry Pi, deployment, effectiveness.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

Prof(a).	Professor(a)
M.E.E	Mestre em Engenharia Elétrica
Ma.	Mestra
Me	Mestre
Dr.	Doutor
PHD	*Doctor of Philosophy*

LISTA DE SIGLAS

ARM Advanced RISC Machine (Máquina de Conjunto de Instruções Reduzido Avançada)

DoS Denial of Service (Negação de Serviço)

HDMI High-Definition Multimedia Interface (Interface Multimídia de Alta Definição)

HTTP Hypertext Transfer Protocol (Protocolo de Transferência de Hipertexto)

HTTPS Hypertext Transfer Protocol Secure (Protocolo de Transferência de Hipertexto Seguro)

IoT Internet of Things (Internet das Coisas)

ISO International Organization for Standardization (Organização Internacional de Normalização)

OSI Open Systems Interconnection (Interconexão de Sistemas Abertos)

RAM Random Access Memory (Memória de Acesso Aleatório)

SBC Random Access Memory (Memória de Acesso Aleatório)

SD Secure Digital (Digital Seguro)

SSL *Secure* Sockets Layer (Camada de Soquetes Segura)

TLS Transport Layer Security (Segurança da Camada de Transporte)

USB Transport Layer Security (Segurança da Camada de Transporte)

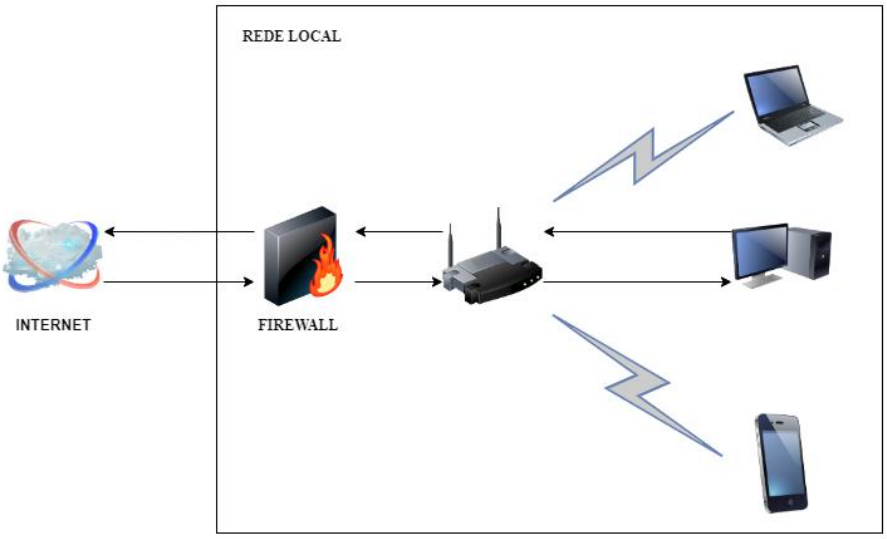
Wi-Fi Wireless Fidelity (Fidelidade Sem Fio)

Sumário

1 Introdução

A segurança de rede é uma preocupação crítica para qualquer organização que faça uso de tecnologia da informação. As redes estão expostas a uma variedade de ameaças, como ataques de *hackers*, *malware* e *phishing*, que podem comprometer a confidencialidade, integridade e disponibilidade dos dados. Nesse contexto, a implementação de um *firewall* é uma estratégia essencial para garantir a segurança da rede e proteger os recursos da organização contra ataques, conforme exemplificado na figura 1.

Figura 1 - *Firewall*



Fonte: Elaborado pelo autor.

De acordo com Tanenbaum e Wetherall (2011), *firewall* é uma tecnologia de segurança de rede que monitora e controla o tráfego de rede, filtrando os pacotes de dados que entram e saem da rede. Ele é capaz de identificar e bloquear ameaças, como *malware* e ataques de negação de serviço, além de permitir a criação de políticas de acesso para restringir o tráfego indesejado ou não autorizado. Existem diversos tipos de *firewall*s, desde os baseados em *software*, como os que rodam em sistemas operacionais, até os baseados em hardware, que são dispositivos especializados para essa finalidade.

Neste trabalho, será abordada a implantação de um *firewall* em um *single board computer* (SBC), sendo representado por um *Raspberry Pi* 4, um dispositivo de baixo

custo e consumo de energia que tem se tornado popular devido à sua versatilidade e facilidade de uso. O objetivo é explorar as funcionalidades do *Raspberry Pi* como plataforma de *firewall* e avaliar sua capacidade de fornecer segurança de rede efetiva em um ambiente de pequeno porte.

Conforme Kurose e Ross (2013), a escolha da plataforma de *firewall* é crucial para a segurança da rede. A implantação de um *firewall* em um SBC como o *Raspberry Pi*, apresenta diversas vantagens, como baixo custo e consumo de energia, além de oferecer flexibilidade na configuração e personalização do *firewall*.

Neste trabalho são apresentados os conceitos teóricos de *firewall*, os diferentes tipos de *firewall* disponíveis e as considerações importantes para a implantação de um *firewall* em uma rede. Em seguida, é descrito o processo de configuração do *Raspberry Pi* como plataforma de *firewall*, abrangendo a instalação do sistema operacional, a configuração da interface de rede e a instalação e configuração do *software* de *firewall*.

Parker (2023) ressalta a relevância da configuração apropriada do *firewall* para assegurar a eficácia da segurança da rede. Nesta pesquisa, são abordados os prós e contras da utilização de um *Raspberry Pi* como plataforma de *firewall* em relação a outras soluções de *firewall*, além das considerações cruciais para garantir a segurança tanto do dispositivo quanto da rede.

Por fim, são apresentados os resultados da implementação do *firewall* no *Raspberry Pi* e as conclusões sobre a efetividade do dispositivo como plataforma de *firewall* em um ambiente de pequeno porte. Espera-se que este trabalho contribua para a compreensão das funcionalidades de um *firewall* e das possibilidades de utilização de um *Raspberry Pi* como plataforma de segurança de rede.

1.1 Objetivos

Esta seção visa apresentar e descrever os objetivos gerais e específicos deste trabalho.

1.1.1 Objetivos Gerais

O objetivo geral deste trabalho é avaliar a efetividade da utilização de um *Raspberry Pi* como plataforma de *firewall* para fornecer segurança de rede em um ambiente de pequeno porte.

1.1.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- Identificar as características técnicas e funcionais do *Raspberry Pi* que o tornam uma plataforma viável para atuar como *firewall* em um ambiente de pequeno porte.
- Analisar os principais desafios e riscos de segurança enfrentados por redes de pequeno porte e como um *firewall* baseado em *Raspberry Pi* pode mitigá-los.
- Projetar e implementar um sistema de *firewall* baseado em *Raspberry Pi* em um ambiente de teste simulando um ambiente de pequeno porte.
- Realizar testes de desempenho e segurança do sistema de *firewall* implementado e avaliar sua efetividade em relação à segurança da rede e sua capacidade de suportar a carga de tráfego.
- Avaliar os custos e benefícios da utilização do *Raspberry Pi* como plataforma de *firewall* em comparação com outras soluções de segurança de rede disponíveis no mercado.

1.2 Justificativa

O presente trabalho tem como objetivo a implementação de um *firewall* em um *Single Board Computer* (SBC), utilizando o *Raspberry Pi* como base. A relevância desse estudo se dá pelo aumento da utilização de SBCs em projetos de *Internet of Things* (IoT), e consequentemente, a necessidade de segurança desses dispositivos.

Com o crescente número de dispositivos conectados à Internet, a segurança se torna uma questão crítica, e a falta de proteção pode resultar em perda de dados e danos financeiros para as empresas. A implementação de um *firewall* no SBC, além

de garantir a segurança do dispositivo, também pode aumentar a proteção dos dados trafegados na rede.

Além disso, a implementação de um *firewall* em um SBC com o *Raspberry Pi* pode ser uma alternativa mais econômica e acessível para pequenas e médias empresas que desejam aumentar a segurança de suas redes. Dessa forma, o estudo pode contribuir para disseminar a importância da segurança em projetos de IoT e ajudar na disseminação de soluções de segurança mais acessíveis e eficientes para empresas de menor porte.

1.3 Procedimentos metodológicos

Para realizar a implementação do *Raspberry Pi* como plataforma de *firewall*, serão realizados os seguintes procedimentos metodológicos:

- Revisão bibliográfica: será realizada uma pesquisa bibliográfica sobre as principais soluções de *firewall* disponíveis, incluindo o *Raspberry Pi*, a fim de avaliar suas vantagens e desvantagens e determinar quais são as melhores práticas de configuração e segurança.
- Seleção de equipamentos: serão selecionados os equipamentos necessários para a implementação, como o próprio *Raspberry Pi,* adaptador de Ethernet, entre outros.
- Instalação do sistema operacional: será realizado o download e instalação do sistema operacional adequado para o *Raspberry Pi*, considerando as opções disponíveis e as recomendações da literatura.
- Configuração do *firewall*: será realizada a configuração do *firewall* no *Raspberry Pi*, seguindo as melhores práticas de segurança identificadas na revisão bibliográfica.
- Testes e avaliação: serão realizados testes para avaliar a efetividade do *Raspberry Pi* como plataforma de *firewall*. Serão avaliados também aspectos como desempenho e segurança.
- Análise dos resultados: serão analisados os resultados obtidos nos testes e avaliações, comparando a efetividade do *Raspberry Pi* como plataforma

de *firewall*. Serão identificadas também possíveis limitações e desafios encontrados durante o processo de implementação.

1.4 Resultados Esperados

Espera-se que os resultados deste trabalho possam AUXILIAR:

- Análise da efetividade da utilização do *Raspberry Pi* como plataforma de *firewall* em um ambiente de pequeno porte;
- Demonstrar a capacidade do dispositivo em fornecer segurança de rede adequada e efetiva, comparando-o com outras soluções de *firewall* disponíveis no mercado;
- Identificação das vantagens e desvantagens do uso do *Raspberry Pi* como plataforma de *firewall*;
- Fornece orientações importantes para garantir a segurança do dispositivo e da rede;
- Contribuir para a melhoria da segurança de rede em ambientes de pequeno porte;
- Oferecer *insights* para futuras pesquisas nessa área.

2 Referencial Teórico

Este capítulo contém o referencial teórico necessário para a elaboração do presente trabalho, com conceitos e definições importantes para uma compreensão desse projeto, de forma a incluir pesquisas bibliográficas relacionadas ao tema do presente trabalho de conclusão de curso

2.1 Informação

De acordo com a norma ISO/IEC 13335, informação é "um ativo que tem valor para a organização e que precisa ser protegido contra ameaças para garantir a continuidade dos negócios". Por sua vez, dado é definido como "uma representação simbólica de fatos, conceitos ou instruções em um formato adequado para comunicação, interpretação ou processamento por seres humanos ou por máquinas" (ABNT NBR ISO/IEC 27000:2018).

Conforme o livro "Fundamentos de Segurança da Informação: Com base na ISO 27001 e na ISO 27002", informação pode ser definida como um dado que possui significado em um contexto específico para quem o recebe, sendo que, após processamento, o dado de saída pode ser novamente percebido como informação (Hintzbergen et al., 2018).

2.2 Segurança da informação

A segurança da informação conhecida também pelo termo "cibersegurança" ou em inglês “*Cybersecurity*” teve origem nos Estados Unidos da América na década de 1980, quando a Internet começou a se tornar uma presença cada vez mais importante no mundo digital. Com o aumento da atividade *online*, a necessidade de proteger os sistemas e as informações dos usuários contra ameaças cibernéticas se tornou uma preocupação cada vez mais urgente.

De acordo com a ABNT NBR ISO/IEC 27001:2013, a segurança da informação é definida como "a preservação da confidencialidade, integridade e disponibilidade da informação por meio da aplicação de um conjunto de controles adequados, incluindo processos, tecnologias e práticas de gestão de riscos". Já a ABNT NBR ISO/IEC

27032:2012 define a cibersegurança como "a proteção da privacidade, propriedade intelectual, ativos financeiros e dados pessoais, bem como a proteção contra interrupção de negócios e o seu impacto, relacionados ao uso, armazenamento e transmissão de informações por meio de redes de computadores ou outros meios eletrônicos".

A cibersegurança abrange a proteção das informações e dos sistemas de informação em um contexto de ameaças cibernéticas, enquanto a segurança da informação tem como objetivo a proteção geral das informações contra qualquer tipo de ameaça.

Além das definições já apresentadas, existem diversas normas e padrões internacionais que abordam a segurança da informação. A ABNT NBR ISO/IEC 27001:2013 é uma das mais conhecidas e utilizadas em todo o mundo. Ela estabelece os requisitos para um sistema de gestão de segurança da informação e fornece diretrizes para sua implementação.

Já a ABNT NBR ISO/IEC 27032:2012 fornece orientações para a cibersegurança em nível organizacional e técnico, com o objetivo de melhorar a resiliência cibernética de uma organização.

Outro padrão relevante inclui o NIST *Cybersecurity Framework*, desenvolvido pelo *National Institute of Standards and Technology* dos Estados Unidos. Esse *framework* fornece orientações para a gestão de riscos cibernéticos, incluindo a proteção de dados e informações sensíveis (NIST, 2018), composto por cinco etapas: identificação, proteção, detecção, resposta e recuperação. Cada uma dessas etapas é dividida em diversas categorias, que apresentam as melhores práticas e controles de segurança que devem ser adotados pelas organizações para garantir a proteção adequada de seus ativos de informação.

No Brasil, além de se ter a normas criadas pela ISO/IEC e traduzidas pela ABNT, existem algumas leis que definem as questões de segurança de informação:

- Lei Geral de Proteção de Dados (LGPD) - BRASIL. Lei nº 13.709, de 14 de agosto de 2018.
- Marco Civil da Internet - BRASIL. Lei nº 12.965, de 23 de abril de 2014.

2.3 LGPD e Marco Civil da Internet

Segundo o Marco Civil da Internet, também conhecido como "Constituição da Internet", criado em 2014, as empresas e organizações que fornecem serviços na Internet têm a responsabilidade de proteger a privacidade dos usuários e não podem monitorar os conteúdos acessados, exceto em casos de ordem judicial (BRASIL, 2014). O objetivo dessa lei é estabelecer princípios, garantias, direitos e deveres para o uso da Internet no país.

A LGPD (Lei Geral de Proteção de Dados) é uma legislação brasileira que entrou em vigor em 2018 com o objetivo de proteger a privacidade e os direitos dos titulares de dados pessoais. A lei estabelece regras claras para a coleta, armazenamento, tratamento e compartilhamento das informações, buscando garantir maior transparência e segurança no seu uso (BRASIL, 2018).

Já que grande parte das informações é gerada e processada digitalmente, a LGPD se torna ainda mais relevante, pois visa proteger os dados pessoais coletados nesse ambiente. Dessa forma, a lei exige que as empresas e organizações que coletam dados pessoais obtenham o consentimento dos titulares e adotem medidas para garantir a privacidade e segurança dessas informações (BRASIL, 2018).

O Marco Civil da Internet e a LGPD são leis que, juntas, estabelecem um conjunto de diretrizes e regras para garantir a proteção da privacidade e segurança da informação no ambiente digital. De acordo com Gouveia e Silva (2020), essas leis possuem uma relação de complementariedade, em que cada uma reforça e aprimora as disposições da outra, assegurando a proteção dos direitos dos usuários e a responsabilidade das empresas e organizações em relação à proteção de dados pessoais.

Outra regulamentação importante é o GDPR (*General Data Protection Regulation*), que estabelece regras para a proteção de dados pessoais na União Europeia. O GDPR se assemelha à LGPD em vários aspectos, estabelecendo regras para a coleta, armazenamento, tratamento e compartilhamento de dados pessoais, bem como penalidades para o descumprimento das normas (EU, 2016).

2.4 Redes de computadores

De acordo com Kurose e Ross (2013), uma rede de computadores pode ser definida como um conjunto de dispositivos eletrônicos interconectados, que são capazes de trocar informações entre si. Essa troca de informações ocorre por meio de uma linguagem de comunicação de computador comum, como o Protocolo de Internet (IP).

Também é importante destacar que as redes de computadores podem variar em tamanho e em estrutura, dependendo da sua finalidade. Segundo Tanenbaum e Wetherall (2011), as redes podem ser classificadas em dois tipos principais: pública e privada. A rede pública mais conhecida é a Internet, que é uma rede massiva e global, conectando milhões de computadores ao redor do mundo.

Contudo, é importante ter em mente que as redes de computadores podem apresentar vulnerabilidades e riscos de segurança, como destaca Forouzan (2006). Por isso, é necessário adotar medidas de segurança para garantir a proteção das informações que são transmitidas na rede. Como menciona Parker (2023), uma das medidas de segurança que pode ser adotada é a utilização de *firewall*s para controlar o tráfego de dados que entra e sai da rede.

As redes de computadores são essenciais para a comunicação entre dispositivos e o compartilhamento de recursos. Segundo Tanenbaum e Wetherall (2011), uma rede de computadores é um conjunto de computadores autônomos interconectados por uma única tecnologia de comunicação para comunicar e compartilhar recursos. A comunicação em rede permite que os dispositivos compartilhem recursos, como arquivos, impressoras e conexões com a Internet.

Em suma, as redes de computadores são fundamentais para a comunicação e troca de informações entre dispositivos eletrônicos. Porém, é importante estar atento aos riscos de segurança que podem ser associados a essa troca de informações, adotando medidas de proteção para garantir a segurança dos dados transmitidos na rede.

As redes de computadores são classificadas de acordo com seu tamanho e alcance geográfico. De acordo com Kurose e Ross (2013), as redes locais (LANs) são redes de pequeno porte que conectam dispositivos em uma área limitada, como uma

empresa ou uma casa. Já as redes de longa distância (WANs) são redes maiores que se estendem por grandes áreas geográficas, como países ou continentes.

Além disso, as redes de computadores também podem ser classificadas de acordo com sua arquitetura de comunicação. Segundo Forouzan (2006), as redes ponto-a-ponto (*peer-to-peer*) são aquelas em que todos os dispositivos têm o mesmo papel e podem compartilhar recursos igualmente. Já as redes cliente-servidor são aquelas em que alguns dispositivos são servidores que fornecem recursos e serviços para outros dispositivos, que atuam como clientes.

Em suma, as redes de computadores são um conjunto de dispositivos interconectados por uma única tecnologia de comunicação, que permitem a comunicação e o compartilhamento de recursos. Elas variam em tamanho e alcance geográfico, e podem ser classificadas de acordo com sua arquitetura de comunicação. O entendimento dessas definições é fundamental para o desenvolvimento e gerenciamento de redes de computadores eficientes e seguras.

2.5 Segurança de redes de computadores

A origem do termo "segurança de redes de computadores" está relacionada com o crescimento exponencial das redes de computadores e a necessidade de proteger essas redes de ameaças externas. Conforme explica Stallings (2005), a segurança é um requisito fundamental para a comunicação em rede, e a preocupação com a segurança de redes de computadores cresceu significativamente com o aumento da utilização de redes para comunicação confidencial.

Com o passar do tempo e o avanço da tecnologia, surgiram cada vez mais desafios na área da segurança de redes de computadores, como ataques cibernéticos. Segundo Parker (2023), "a segurança de redes de computadores é um tema complexo que evolui constantemente, exigindo uma abordagem proativa e multidisciplinar para garantir a proteção dos recursos de rede".

Segurança de redes de computadores é o conjunto de medidas técnicas, políticas e procedimentos que visam proteger os recursos de uma rede de computadores contra ameaças. Segundo Kurose e Ross (2013), a segurança de redes é uma das maiores preocupações em redes de computadores, visto que as redes

estão cada vez mais presentes na vida das pessoas e das empresas, e os dados trafegados podem conter informações sigilosas e valiosas.

Para Gabriel Torres (2016), a segurança de redes deve ser tratada como um processo contínuo, que envolve diversas etapas, desde a identificação das ameaças até a implementação de controles de segurança. Essas etapas incluem análise de riscos, políticas de segurança, controles de acesso, criptografia, prevenção e detecção de intrusos, entre outros.

Douglas Rocha Mendes (2015) destaca que a segurança de redes de computadores envolve a proteção de diversos recursos, como hardware, *software*, dados e informações. De acordo com Stallings (2005), a segurança de redes é um processo complexo, que envolve a utilização de diversas tecnologias e técnicas, como *firewall*s, criptografia, detecção e prevenção de intrusos, entre outros. Essas tecnologias devem ser combinadas de forma a garantir a segurança da rede como um todo.

Tanenbaum e Wetherall (2011) ressaltam que a segurança de redes de computadores é um desafio constante, já que as ameaças estão sempre evoluindo e se adaptando. Por isso, é importante que as empresas invistam em políticas de segurança robustas e atualizadas, além de capacitarem seus funcionários e utilizarem tecnologias de ponta para proteger seus recursos.

A segurança de redes de computadores é um conjunto de medidas essenciais para garantir a proteção e a integridade das informações e recursos de uma rede. Essas medidas envolvem a utilização de tecnologias e técnicas de segurança, além de políticas e procedimentos que visam prevenir e detectar ameaças internas e externas. Também é um conjunto de medidas que visa proteger as informações e recursos contidos em uma rede contra ameaças internas e externas. Segundo Gabriel Torres (2016), a segurança de redes de computadores envolve a adoção de políticas de segurança, o uso de tecnologias de criptografia, autenticação e autorização, além da implementação de *firewall*s, antivírus e outras ferramentas de segurança.

De acordo com Parker (2023), a segurança de redes de computadores é importante para garantir a privacidade, integridade e disponibilidade dos dados, evitando que informações sensíveis caiam em mãos erradas e prejudiquem os usuários e a empresa. A segurança de redes de computadores também envolve a prevenção de ataques cibernéticos e a identificação de vulnerabilidades.

Para Tanenbaum e Wetherall (2011), a segurança de redes de computadores é um desafio constante, pois novas ameaças surgem constantemente, exigindo a atualização e melhoria das medidas de segurança adotadas. A comunicação segura entre dispositivos e a proteção contra invasões, malwares e outras ameaças são aspectos críticos da segurança de redes de computadores.

Forouzan (2006) destaca que a segurança de redes de computadores não se limita apenas a questões técnicas, mas também envolve aspectos humanos, como a conscientização dos usuários sobre a importância da segurança da informação e a adoção de boas práticas de segurança, como o uso de senhas fortes e a atualização regular de *software*s.

A segurança de redes de computadores é fundamental para garantir a proteção das informações e recursos contidos em uma rede, bem como para a manutenção da privacidade e confidencialidade dos usuários. A implementação de políticas de segurança, tecnologias e boas práticas, aliada à conscientização dos usuários, são elementos-chave para uma rede segura e confiável.

2.6 World Wide Web (WWW)

A *World Wide Web* (WWW), ou simplesmente Web, é uma plataforma de distribuição global de informação na Internet, criada em 1989 por Tim Berners-Lee, com o objetivo de facilitar o compartilhamento e o acesso a documentos e informações em formato digital. Conforme Forouzan (2006), a Web é uma aplicação da Internet que utiliza o protocolo HTTP para transferência de dados.

A Web teve um papel fundamental na democratização da informação, uma vez que possibilitou que qualquer pessoa pudesse criar e compartilhar conteúdo na rede, sem a necessidade de conhecimentos técnicos avançados. Berners-Lee e Fischetti (2000) afirmam que a Web foi criada com o intuito de ser um espaço aberto e livre, onde todas as pessoas tivessem acesso à informação, independentemente de sua localização geográfica, cultura, idioma ou condição socioeconômica.

De acordo com Kurose e Ross (2013), a Web é composta por uma coleção de documentos interligados, conhecidos como páginas da Web, que podem conter textos, imagens, áudio, vídeo e outros elementos multimídia. A navegação na Web é realizada por meio de hipertextos, que permitem ao usuário acessar diferentes

páginas da Web por meio de links, formando uma teia de conexões entre os documentos.

A Web tem como base três tecnologias principais: o protocolo HTTP (*Hypertext Transfer Protocol*), que permite a transferência de informações na forma de hipertexto; o HTML (*Hypertext Markup Language*), que é a linguagem utilizada para criar e formatar as páginas da Web; e o URL (*Uniform Resource Locator*), que é o endereço que identifica cada recurso disponível na Web. A combinação dessas tecnologias permitiu a criação de uma rede global de informação que conecta milhões de pessoas e dispositivos em todo o mundo. Por definição: "A Web é construída sobre o modelo cliente-servidor, em que um cliente envia uma solicitação a um servidor, que processa a solicitação e envia a resposta ao cliente." (Forouzan, 2006)

A Web tornou-se uma das principais ferramentas de comunicação e informação da atualidade, permitindo o acesso a informações de diferentes áreas do conhecimento, o compartilhamento de ideias e o desenvolvimento de negócios e serviços online. (*World Wide Web Foundation*)

Kurose e Ross (2013) destacam que a Web tem um papel fundamental na chamada "revolução da informação". A sua evolução tem sido constante, com o surgimento de novas tecnologias e linguagens de programação, como o CSS (*Cascading Style Sheets*), o JavaScript e o XML (*Extensible Markup Language*), que permitem a criação de páginas mais sofisticadas e interativas.

É importante lembrar que a Web também apresenta desafios, como a segurança e a privacidade dos usuários, a disseminação de informações falsas e a exclusão digital. Como destaca Mendes (2015), "a Web trouxe grandes benefícios para a sociedade, mas também apresenta riscos e desafios que precisam ser enfrentados de forma responsável e consciente".

Portanto, a *World Wide Web* é uma rede de informações interconectadas que permite o acesso e compartilhamento de conhecimento em escala global. Como afirma Torres (2016), a Web é uma das principais conquistas da era digital, tendo um impacto profundo e duradouro em praticamente todos os aspectos da sociedade contemporânea.

Este impacto na sociedade, permite o compartilhamento de informações em tempo real, a criação de redes sociais, e-commerce, marketing digital, entre outros. No entanto, a expansão da Web também trouxe desafios relacionados à privacidade e à segurança dos dados. Como ressalta Parker (2023), a Web é vulnerável a diversos

tipos de ameaças, como ataques cibernéticos, *phishing* e roubo de identidade. o que motivou a criação de leis e normas de segurança, como a Lei Geral de Proteção de Dados (LGPD) no Brasil e o Regulamento Geral de Proteção de Dados (GDPR) na União Europeia. Por isso, é fundamental adotar medidas de proteção, como o uso de *firewalls* e a criptografia de dados, conforme as normas estabelecidas pelas ISO (*International Organization for Standardization*) 27001 e 27002, conforme mencionado por Hintzbergen et al. (2018).

2.7 Protocolo de redes

Os protocolos de redes tiveram origem com a necessidade de interconectar sistemas computacionais em redes, permitindo a troca de informações entre eles. Segundo Tanenbaum e Wetherall (2011), "um protocolo é um conjunto de regras que governam a comunicação entre os computadores em uma rede". Essas regras estabelecem como os dados são transmitidos, formatados e interpretados pelos dispositivos de rede.

Segundo Kurose e Ross (2013), protocolos de redes de computadores são divididos em camadas, cada uma com suas funções específicas, que juntas possibilitam a troca de informações entre os dispositivos conectados. Essas camadas são definidas pelo modelo OSI (Open Systems Interconnection), desenvolvido na década de 1980 pela ISO (*International Organization for Standardization*), é uma referência para a padronização dos protocolos de redes. Ele é composto por sete camadas, cada uma com suas funções específicas, e permite que diferentes dispositivos de rede se comuniquem de forma padronizada. Segundo Kurose e Ross (2013), "o modelo OSI não é usado diretamente em nenhuma rede, mas serve como referência para o desenvolvimento de protocolos específicos".

O modelo OSI (Open Systems Interconnection) é um modelo de referência para protocolos de rede, que divide as funções de comunicação em camadas, cada uma responsável por tarefas específicas. Existem sete camadas no modelo OSI, cada uma com sua própria função. conforme exemplificado na figura 2.

Figura 2 - Modelo OSI

Fonte: Elaborado pelo autor.

A camada de enlace de dados (*layer* 2) é responsável por transmitir quadros de dados em um canal de comunicação. Torres (2016) destaca que "a camada de enlace de dados é responsável pelo endereçamento físico, pela detecção e correção de erros, controle de fluxo e sequência, e o acesso compartilhado ao meio físico".

A camada de rede (*layer* 3) é responsável pela transmissão de pacotes de dados de origem para destino através de uma rede interconectada. Tanenbaum e Wetherall (2011) afirmam que "a camada de rede realiza a entrega de pacotes entre sistemas finais por meio de uma ou mais redes intermediárias".

A camada de transporte (*layer* 4) é responsável pela entrega de dados confiável e ordenada de um processo para outro. Segundo Forouzan (2006), "a camada de transporte fornece comunicação de extremidade a extremidade confiável e orientada à conexão. Essa camada pode controlar o fluxo de dados e garantir que os dados sejam entregues sem erros e na ordem correta".

A camada de sessão (*layer* 5) é responsável pela comunicação entre aplicativos em diferentes dispositivos. Mendes (2015) explica que "a camada de sessão é responsável por estabelecer, manter e encerrar conexões entre aplicativos em diferentes dispositivos. Essa camada também pode lidar com sincronização e recuperação de sessão".

A camada de apresentação (*layer* 6) é responsável pela representação de dados, codificação e compressão. De acordo com Stallings (2005), "a camada de apresentação é responsável por formatar, comprimir e criptografar os dados a serem transmitidos. Essa camada também pode realizar a conversão entre diferentes formatos de dados".

A camada de aplicação (*layer* 7) é responsável por fornecer serviços para aplicativos de usuário final. Parker (2023) destaca que "a camada de aplicação fornece interfaces para aplicativos de usuário final, como e-mail, navegador da Web e protocolos de transferência de arquivos"

Mendes (2015) destaca que, além do modelo OSI, existem outros modelos de referência, como o TCP/IP (*Transmission Control Protocol*/*Internet Protocol*), que é amplamente utilizado na Internet. O TCP/IP é composto por quatro camadas: aplicação, transporte, rede e link de dados.

Tanenbaum e Wetherall (2011) ressaltam que o protocolo de rede mais utilizado atualmente é o TCP/IP, que é a base da Internet. Ele é composto por várias camadas, cada uma com seus próprios protocolos específicos, e permite que diferentes dispositivos em diferentes redes possam se comunicar de forma confiável. Stallings (2005) ressalta a importância dos protocolos de segurança nas redes de computadores, como o SSL (*Secure Sockets Layer*) e o TLS (*Transport Layer Security*), que garantem a privacidade e integridade dos dados transmitidos.

Tanenbaum e Wetherall (2011) afirmam que a escolha do protocolo mais adequado para uma determinada rede depende de diversos fatores, como o tipo de aplicação, o tamanho da rede e as características dos dispositivos.

Torres (2015) destaca a importância da padronização dos protocolos, para que diferentes dispositivos e redes possam se comunicar entre si, independentemente do fabricante ou tecnologia utilizada.

Forouzan (2006) reforça a importância dos protocolos de rede para a eficiência e segurança das comunicações entre dispositivos, além de enfatizar a necessidade de atualização constante desses protocolos, para acompanhar as mudanças e evoluções tecnológicas.

Parker (2023) destaca a importância dos protocolos de segurança, como *firewall*s, para proteger as redes de computadores contra-ataques maliciosos e garantir a segurança dos dados transmitidos.

Os protocolos de redes de computadores são essenciais para a comunicação entre dispositivos em uma rede, garantindo a transmissão eficiente e segura de dados. A escolha do protocolo adequado depende de diversos fatores, e é importante manter os protocolos atualizados para acompanhar as evoluções tecnológicas e garantir a segurança das comunicações.

2.8 TCP/IP

O TCP/IP é um protocolo de rede que se tornou o principal padrão para comunicação em redes de computadores em todo o mundo. Sua origem remonta à década de 1960, quando o Departamento de Defesa dos Estados Unidos iniciou o desenvolvimento de uma rede de comunicações robusta e confiável, conhecida como ARPANET. Segundo Tanenbaum e Wetherall (2011), "o TCP/IP foi originalmente desenvolvido para a ARPANET como um conjunto de protocolos padrão para conectar computadores em rede".

Durante a década de 1970, o TCP/IP evoluiu para se tornar o protocolo padrão para a comunicação em redes de computadores, sendo adotado por empresas e governos em todo o mundo. Conforme Kurose e Ross (2013) apontam, "a aceitação do TCP/IP foi acelerada pelo surgimento da Internet, uma rede global de computadores interconectados que usam o TCP/IP como protocolo padrão de comunicação".

O modelo TCP/IP é uma família de protocolos de comunicação utilizados para interconexão de redes de computadores. Segundo Tanenbaum e Wetherall (2011), "o TCP/IP é uma combinação de dois protocolos diferentes: o TCP (Protocolo de Controle de Transmissão) e o IP (Protocolo da Internet)". O TCP é responsável pela entrega confiável e ordenada dos dados entre processos, enquanto o IP é responsável pela entrega de pacotes de dados entre dispositivos em diferentes redes. De acordo com Kurose e Ross (2013), "o TCP/IP é o conjunto de protocolos de rede mais utilizado na Internet e em outras redes de computadores". Esse modelo é dividido em quatro camadas, que são: camada de aplicação, camada de transporte, camada de rede e camada de enlace de dados, conforme exemplificado na figura 3.

Fonte: Elaborado pelo autor.

A camada de aplicação (*application layer*) é responsável por fornecer serviços de rede aos aplicativos que utilizam a rede. De acordo com Tanenbaum e Wetherall (2011), essa camada "oferece interfaces de programação de aplicativos (APIs) para os aplicativos de usuário final, como e-mail, navegador da Web e protocolos de transferência de arquivos".

A camada de transporte (*transport layer*) é responsável pela comunicação confiável entre processos de aplicação em diferentes dispositivos. Kurose e Ross (2013) afirmam que "a camada de transporte é responsável pela entrega de dados confiável e ordenada de um processo para outro. Essa camada pode controlar o fluxo de dados e garantir que os dados sejam entregues sem erros e na ordem correta".

A camada de Internet (*Internet layer*) é responsável pelo roteamento de pacotes de dados através de uma rede interconectada. Segundo Torres (2016), "a camada de rede é responsável pelo roteamento de pacotes de dados entre diferentes redes. Essa camada pode fornecer serviços como endereçamento lógico, fragmentação e montagem de pacotes".

A camada de acesso à rede (*network access layer*) é responsável por transmitir quadros de dados através de um meio físico de comunicação. Forouzan (2006) explica que "a camada de enlace de dados é responsável pelo endereçamento físico dos

dispositivos de rede, detecção e correção de erros e controle de acesso ao meio físico".

2.9 HTTP

O protocolo HTTP (*Hypertext Transfer Protocol*), sigla em inglês para Protocolo de Transferência de Hipertexto, é um protocolo de rede utilizado para a comunicação entre um cliente e um servidor na *World Wide Web* (WWW). Criado por Tim Berners-Lee em 1989, o HTTP permite que os usuários acessem, visualizem e interajam com os recursos disponíveis na Internet.

De acordo com Kurose e Ross (2013), o HTTP funciona de forma cliente-servidor, onde o cliente faz uma requisição ao servidor para acessar um determinado recurso, como uma página da Web, e o servidor responde com a informação solicitada. Esse modelo de troca de informações entre cliente e servidor é a base do funcionamento do HTTP e permite que os usuários da Web acessem diversos tipos de informações disponíveis em servidores remotos.

O HTTP possui uma estrutura simples, composta por uma linha de requisição (*request-line*), um cabeçalho (*header*) e uma mensagem de corpo (*body*). A linha de requisição indica o método utilizado (GET, POST, PUT, DELETE etc.), o recurso solicitado (URL) e a versão do protocolo. O cabeçalho contém informações adicionais sobre a requisição, como as opções de cache, autenticação, tipo de conteúdo, entre outras. Já a mensagem de corpo contém os dados enviados pelo cliente, como no caso de uma requisição POST.

O HTTP tem passado por diferentes versões, cada uma com suas particularidades e melhorias em relação à versão anterior. Segundo Kurose e Ross (2013), na primeira versão do HTTP, o HTTP/0.9, somente era possível a transferência de arquivos de hipertexto, não havendo suporte para cabeçalhos ou outros tipos de dados. Já a versão HTTP/1.0, introduzida em 1996, incluiu suporte a cabeçalhos e a outros tipos de arquivos além do HTML, como imagens e vídeos. Além disso, o HTTP/1.0 também permitiu a conexão persistente, o que melhorou significativamente o desempenho.

A versão HTTP/1.1, lançada em 1999, trouxe uma série de melhorias em relação à sua antecessora. Segundo Forouzan (2006), O HTTP/1.1 é uma versão

aprimorada do HTTP/1.0, com melhorias significativas em vários aspectos, tais como a redução do número de conexões TCP necessárias para carregar uma página Web e o suporte a cabeçalhos de resposta mais robustos, como o cabeçalho 'Cache-Control', entre outros. O HTTP/1.1 também introduziu o conceito de 'pipelining', que permitiu o envio de várias solicitações em uma única conexão, reduzindo ainda mais o tempo de carregamento da página.

A versão mais utilizada do HTTP é a HTTP/2, lançada em 2015. Stallings (2005) aponta que o HTTP/2 apresenta uma série de melhorias, como o uso da multiplexação, que permite o envio de várias solicitações e respostas simultaneamente em uma única conexão, e o uso de compressão de cabeçalhos, que reduz a quantidade de dados a serem transferidos. Além disso, o HTTP/2 também introduziu o conceito de 'push', que permite ao servidor enviar recursos adicionais para o cliente antes mesmo de serem solicitados, melhorando ainda mais o desempenho.

De acordo com Ghedini e Lalkaka (2020) no artigo 'HTTP/3: the past, the present, and the future', a principal vantagem do HTTP/3 em relação ao HTTP/2 é a utilização do protocolo QUIC, que foi projetado para reduzir a latência na transmissão de dados, principalmente em redes móveis e com alta perda de pacotes. Isso é especialmente importante para a entrega de conteúdo em tempo real, como streaming de vídeo e áudio.

O principal desafio atual do HTTP é lidar com o aumento constante no volume de tráfego na Web, o que pode levar a congestionamentos na rede e redução no desempenho. Para enfrentar esse desafio, novas tecnologias, como a CDN (*Content Delivery Network*), têm sido desenvolvidas para ajudar a distribuir conteúdo de forma mais eficiente e reduzir a carga nos servidores.

O protocolo HTTP é uma parte fundamental da infraestrutura da Web, permitindo que os usuários acessem e interajam com os recursos disponíveis na Internet. Desde sua criação por Tim Berners-Lee em 1989, o HTTP tem evoluído constantemente, passando por diferentes versões e introduzindo melhorias significativas em cada uma delas. Embora ainda enfrente desafios, o HTTP continua sendo um dos protocolos de rede mais importantes e amplamente utilizados na atualidade.

2.10 HTTPS

O HTTPS (HTTP Seguro) é uma evolução do protocolo HTTP que acrescenta uma camada de segurança à comunicação entre o cliente e o servidor. Segundo Kurose e Ross (2013), o HTTPS utiliza criptografia para proteger os dados em trânsito e garantir a autenticidade do servidor. O SSL (*Secure Socket Layer*) foi o primeiro protocolo de segurança utilizado no HTTPS, mas foi substituído pelo TLS (*Transport Layer Security*) por ser mais seguro e eficiente.

Segundo Forouzan (2006), HTTPS é uma extensão do protocolo HTTP, que foi desenvolvida para prover segurança na transferência de informações sensíveis pela Internet, como senhas e dados financeiros. A sua origem está relacionada com a necessidade de garantir a privacidade e a integridade dos dados, diante das ameaças presentes na rede.

De acordo com Kurose e Ross (2013), o HTTPS funciona utilizando um certificado digital, que é emitido por uma autoridade de certificação confiável e contém informações sobre a identidade do proprietário do site. Quando o cliente acessa um site com HTTPS, o navegador verifica a validade do certificado e estabelece uma conexão criptografada com o servidor. Todas as informações trocadas entre o cliente e o servidor são criptografadas e somente podem ser decodificadas pelo destinatário.

O HTTPS é utilizado para proteger a privacidade dos usuários, impedindo que informações sensíveis sejam interceptadas por terceiros mal-intencionados. Como afirma Tanenbaum e Wetherall (2011), o HTTPS também ajuda a prevenir ataques de *phishing* e a garantir a autenticidade do site, já que o certificado digital garante a identidade do proprietário do site. Além disso, o HTTPS é um requisito para o SEO (*Search Engine Optimization*), pois os mecanismos de busca dão preferência aos sites seguros na hora de apresentar os resultados de pesquisa.

De acordo com a *World Wide Web Foundation* (2023), o HTTPS tem se tornado cada vez mais importante na medida em que as pessoas passaram a compartilhar informações pessoais e financeiras pela Internet. Parker (2023) destaca que o HTTPS é uma das principais medidas de segurança utilizadas para proteger a privacidade dos usuários e evitar ataques cibernéticos. Por isso, é fundamental que os desenvolvedores de sites e os usuários finais estejam conscientes da importância do HTTPS e de como utilizá-lo corretamente.

2.11 SSL (*Secure Socket Layer*) e TLS (*Transport Layer Security*)

SSL (*Secure Socket Layer*) e TLS (*Transport Layer Security*) são protocolos de segurança utilizados no HTTPS para garantir a confidencialidade, integridade e autenticidade das informações trocadas entre o cliente e o servidor. Ambos os protocolos utilizam criptografia para proteger os dados em trânsito e garantir a autenticidade do servidor.

O SSL e o TLS funcionam utilizando criptografia para proteger os dados em trânsito. De acordo com Mendes (2015), o SSL e o TLS utilizam uma combinação de criptografia simétrica e assimétrica para garantir a segurança das informações trocadas. Quando o cliente acessa um site com HTTPS, o navegador verifica a validade do certificado e estabelece uma conexão criptografada com o servidor utilizando um algoritmo de criptografia simétrica para criptografar os dados em trânsito. Em seguida, é utilizado um algoritmo de criptografia assimétrica para trocar as chaves de criptografia simétrica, garantindo a confidencialidade e a integridade dos dados.

De acordo com Berners-Lee e Fischetti (2000), o SSL e o TLS são protocolos que permitem a comunicação segura entre um cliente e um servidor por meio da criptografia dos dados transmitidos. Quando um cliente acessa um site com HTTPS, o navegador verifica a validade do certificado e estabelece uma conexão criptografada com o servidor apenas se o certificado digital for válido.

É importante destacar que a segurança do SSL e do TLS depende da força dos algoritmos de criptografia utilizados e da proteção das chaves de criptografia. Segundo Stallings (2005), o SSL e o TLS têm sido alvo de ataques cibernéticos, especialmente nos últimos anos, o que tem levado a evolução constante desses protocolos para garantir a segurança das informações trocadas pela Internet.

Em uma conexão HTTP não segura, as informações trocadas entre o cliente e o servidor podem ser interceptadas e manipuladas por terceiros mal-intencionados. Conforme Mendes (2015), existem vários métodos de ataques que podem ocorrer em HTTP, como ataques de *sniffing*, ataques de injeção de SQL e ataques de *phishing*. O HTTPS tenta solucionar esses problemas por meio da criptografia e da autenticidade do servidor, que impedem a interceptação e a manipulação das informações, bem como evitam que o usuário seja direcionado para sites falsos.

De acordo com Parker (2023), os ataques mais comuns contra o HTTP são os ataques *man-in-the-middle* (homem-do-meio), que consistem na interceptação e na modificação de dados trocados entre o cliente e o servidor. O HTTPS tenta solucionar esse problema por meio da autenticidade do servidor e da criptografia das informações trocadas, o que impede a modificação e a interceptação dos dados.

É importante ressaltar que a segurança do HTTPS depende do uso correto e da configuração adequada do protocolo. Segundo Torres (2015), é importante que os desenvolvedores de sites e os usuários finais estejam conscientes da importância do HTTPS e de como utilizá-lo corretamente para garantir a segurança das informações trocadas pela Internet.

2.12 Interface de redes de computadores

As interfaces de rede são uma parte essencial das redes de computadores, permitindo a comunicação entre dispositivos através da transmissão de dados. De acordo com Forouzan (2006), uma interface de rede é uma conexão física ou lógica entre um dispositivo de rede e um meio de transmissão. As interfaces de rede permitem que os dispositivos se comuniquem em diferentes níveis, como físico, de enlace de dados e de rede.

Segundo Kurose e Ross (2013), as interfaces de rede têm origem nas primeiras redes de computadores, quando as máquinas eram conectadas diretamente através de cabos de par trançado. Com o tempo, as interfaces de rede evoluíram para suportar diferentes tecnologias, como Ethernet, Wi-Fi e Bluetooth.

Existem diferentes tipos de interfaces de rede, como placas de rede, adaptadores de rede sem fio e modems. De acordo com Mendes (2015), as placas de rede são dispositivos que se conectam diretamente a um cabo de rede e fornecem uma interface de rede física para um computador ou outro dispositivo. Os adaptadores de rede sem fio, por sua vez, permitem a conexão sem fio entre dispositivos em uma rede. Já os modems são dispositivos que permitem a conexão com a Internet através de uma linha telefônica, cabo ou fibra ótica.

Além disso, Stallings (2005) ressalta que as interfaces de rede podem ser classificadas em diferentes categorias, como interface de rede ponto a ponto e interface de rede cliente-servidor. A interface de rede ponto a ponto é utilizada em

redes *peer-to-peer*, onde todos os dispositivos têm as mesmas capacidades e podem se comunicar diretamente uns com os outros. Já a interface de rede cliente-servidor é utilizada em redes onde os dispositivos têm funções distintas, como servidores e clientes, e a comunicação ocorre através de um servidor central.

Stallings (2005) também destaca que as interfaces de rede podem ser vulneráveis a ataques de *spoofing*, em que um invasor falsifica o endereço MAC ou IP para fingir ser outro dispositivo na rede. Isso pode permitir que o invasor intercepte e manipule o tráfego de rede, causando problemas de segurança e de integridade dos dados.

Para mitigar esses riscos, é importante implementar medidas de segurança adequadas, como o uso de autenticação forte para acesso à rede, o monitoramento contínuo do tráfego de rede e a aplicação de atualizações de segurança regulares em drivers e protocolos de rede.

De acordo com Forouzan (2006), um dos principais problemas de segurança associados às interfaces de rede é a possibilidade de interceptação de tráfego por usuários mal-intencionados. Isso pode ocorrer por meio de ataques como o "*Man-in-the-Middle*" (MITM), em que o invasor se posiciona entre o cliente e o servidor para interceptar e manipular as informações trocadas entre eles.

Stallings (2005) destaca que outras vulnerabilidades que podem afetar as interfaces de rede incluem ataque de Negação de Serviço (*Denial of Service* - DoS) é uma tentativa deliberada de tornar-se um serviço, recurso ou sistema indisponível para seus usuários legítimos. No caso dos ataques DoS, Segundo Forouzan (2006) o objetivo é sobrecarregar a interface de rede com um grande volume de tráfego, impedindo que ela atenda a outras requisições legítimas. Já nos ataques de elevação de privilégios, o invasor busca obter acesso a recursos e informações que não estariam disponíveis para um usuário comum.

Para mitigar esses problemas, as interfaces de rede devem ser configuradas corretamente e possuir mecanismos de segurança, como *firewall*s e sistemas de detecção e prevenção de intrusos. Stallings (2005) destaca que a combinação dessas medidas é importante para garantir a segurança e proteção dos dados em redes de computadores.

2.13 Forma de segurança de redes de computadores

A segurança em redes de computadores é um conjunto de técnicas e medidas para proteger a integridade, confidencialidade e disponibilidade de informações transmitidas em uma rede. Segundo Tanenbaum e Wetherall (2011), a segurança em redes de computadores é um tema muito importante e atual, especialmente devido ao aumento dos ataques cibernéticos e à necessidade de garantir a privacidade e a segurança das informações.

Entre as técnicas de segurança em redes de computadores, destacam-se os *firewall*s, que são sistemas de segurança que controlam o tráfego de rede com base em um conjunto de regras predefinidas. Segundo Parker (2023), os *firewall*s são uma das principais medidas de segurança em redes de computadores e são amplamente utilizados para proteger redes empresariais e domésticas. Forouzan (2006) define *firewall* como sendo um dispositivo ou um conjunto de dispositivos que controlam o acesso entre redes, filtrando o tráfego de rede e bloqueando ou permitindo a passagem de dados com base em um conjunto de regras pré-definidas.

Outra técnica de segurança comum é a criptografia, que consiste em codificar as informações para torná-las ilegíveis para qualquer pessoa que não possua a chave de decodificação. Segundo Forouzan (2006), a criptografia tem sido utilizada desde a antiguidade, mas com a popularização da Internet e das redes de computadores, tornou-se uma técnica essencial para garantir a segurança das comunicações. Tanenbaum e Wetherall (2011) explicam que a criptografia pode ser realizada por meio de diversas técnicas, tais como a criptografia simétrica, que utiliza uma mesma chave para cifrar e decifrar a informação, e a criptografia assimétrica, que utiliza um par de chaves - uma pública e outra privada - para realizar a codificação e decodificação dos dados.

Além desses métodos, há também a utilização de protocolos de segurança, como o *Secure Sockets Layer* (SSL) e o *Transport Layer Security* (TLS), que garantem a segurança na transmissão de dados pela Internet. De acordo com Stallings (2005), o SSL e o TLS são protocolos que permitem a criptografia dos dados durante a transmissão, garantindo a privacidade e a integridade dos dados.

Existem outras técnicas de segurança, como autenticação, controle de acesso, detecção de intrusos e monitoramento de tráfego. Mendes (2015) destaca a

importância de implementar várias camadas de segurança em redes de computadores, a fim de minimizar o risco de violação de segurança.

Todos esses métodos são importantes para garantir a segurança em redes de computadores, mas é importante ressaltar que nenhum deles é completamente infalível. É necessário sempre estar atualizado sobre as novas técnicas e ameaças que surgem constantemente e implementar medidas de segurança cada vez mais robustas.

2.14 *Firewall*

O termo "*firewall*", em português "parede de fogo", teve sua origem no campo militar, sendo utilizado para se referir a barreiras físicas que impediam a propagação de incêndios e explosões entre diferentes áreas de uma instalação. Como afirma Kurose e Ross (2013), o termo "*firewall*" foi utilizado pela primeira vez em 1984 por um engenheiro da Digital Equipment Corporation, que buscava uma metáfora para explicar o funcionamento de um sistema de segurança em rede.

Firewall é um sistema de segurança que tem como objetivo proteger os recursos de uma rede de computadores contra ameaças externas. De acordo com Forouzan (2006), um *firewall* é um dispositivo ou *software* que controla o tráfego entre a rede local e a Internet, permitindo que apenas o tráfego autorizado passe por ele.

Segundo Kurose e Ross (2013), um *firewall* é capaz de monitorar e filtrar o tráfego de rede com base em regras predefinidas, que podem ser configuradas pelo administrador da rede. Essas regras determinam quais pacotes de dados serão permitidos ou bloqueados, com base em critérios como endereço IP de origem e destino, porta de origem e destino, protocolo utilizado, entre outros.

O *firewall* também pode ser utilizado para implementar políticas de segurança, como por exemplo, restringir o acesso a determinados sites ou serviços da Internet, ou ainda, bloquear o acesso de usuários não autorizados à rede. De acordo com Parker (2023), o *firewall* é um componente essencial da segurança de rede, mas não é uma solução completa. Ele deve ser utilizado em conjunto com outras medidas de segurança, como antivírus, criptografia e autenticação de usuários.

Conforme Tanenbaum e Wetherall (2011), existem diversos tipos de *firewall*, como o *firewall* de pacotes, o *firewall* de aplicação e o *firewall* de estado. Cada tipo

possui suas próprias características e vantagens, e a escolha do tipo adequado deve ser feita de acordo com as necessidades específicas da rede e dos usuários.

Além disso, é importante destacar que os *firewall*s são uma peça fundamental na proteção contra-ataques cibernéticos. Como mencionado por Hintzbergen et al. (2018), as normas ISO 27001 e ISO 27002 reconhecem a importância dos *firewall*s para a segurança da informação, e recomendam a sua implementação em qualquer organização que lide com dados sensíveis.

Vale ressaltar que, apesar de ser uma ferramenta muito útil na segurança de rede, um *firewall* não é capaz de proteger completamente contra todos os tipos de ataques. Como aponta Parker (2023), os *firewall*s podem ser contornados por meio de técnicas como a engenharia social ou a exploração de vulnerabilidades em sistemas não atualizados. Por isso, é fundamental que as organizações adotem outras medidas de segurança, como a implementação de atualizações de *software*, o treinamento de usuários e a realização de testes de segurança regulares.

Os *firewall*s são um importante mecanismo de segurança em redes de computadores, atuando como uma barreira contra ameaças externas e permitindo o controle de acesso a recursos de rede. No entanto, é importante lembrar que o *firewall* deve ser utilizado em conjunto com outras medidas de segurança para garantir a proteção efetiva dos recursos de rede.

2.15 Tipos de *firewall*

Os *firewall*s são ferramentas essenciais para garantir a segurança da informação em redes de computadores. Eles têm como principal função monitorar o tráfego de dados entre redes distintas, permitindo ou bloqueando o acesso de acordo com regras predefinidas. Existem diferentes tipos de *firewall*s, cada um com suas particularidades e características.

Os *firewall*s de filtro de pacotes de acordo com Tanenbaum e Wetherall (2011), foram os primeiros a serem desenvolvidos e ainda são os mais utilizados. Eles operam na camada de rede do modelo OSI e filtram os pacotes com base em informações presentes nos cabeçalhos dos pacotes de dados, como endereços IP, números de porta e protocolos de transporte.

De acordo com Kurose e Ross (2013), os filtros de pacotes são bastante eficazes em prevenir ataques simples e diretos, mas apresentam limitações quando se trata de lidar com tráfego malicioso disfarçado ou sofisticado. Por essa razão, outros tipos de *firewall*s foram desenvolvidos para aumentar a segurança da rede.

Segundo Mendes (2015), o sistema de regras é constituído por um conjunto de instruções que definem como os pacotes serão tratados pelo *firewall*. Cada regra é composta por uma série de critérios, que podem incluir informações como o endereço de origem, o endereço de destino, o número da porta e o protocolo utilizado.

Os *firewalls* de aplicação, também conhecidos como "application *firewall*" ou "*gateway firewall*", operam na camada de aplicação do modelo OSI e têm a capacidade de analisar todo o conteúdo das requisições e respostas de aplicação, incluindo cabeçalhos e corpos de mensagens. Segundo os autores Kurose e Ross (2013), eles são capazes de bloquear o acesso a determinados serviços e aplicativos, filtrar o tráfego com base em conteúdo específico e até mesmo alterar o conteúdo das mensagens. Portanto, os proxies são uma opção mais avançada para o controle de segurança na rede.

De acordo com Mendes (2015), os *firewall*s de aplicação são capazes de proteger contra-ataques ainda mais avançados, como os que exploram vulnerabilidades específicas em protocolos de aplicação. Eles são considerados mais sofisticados e abrangentes do que os *firewall*s de filtros de pacotes, uma vez que operam na camada de aplicação do modelo OSI, permitindo uma análise mais detalhada do tráfego de rede. Portanto, é recomendável a utilização de *firewall*s de aplicação para garantir uma segurança mais efetiva na rede.

Os *firewall*s de estado, também conhecido como "*stateful firewall*", representam uma técnica de *firewall* que monitora o tráfego de rede e mantém registros do estado de conexões estabelecidas. De acordo com Tanenbaum e Wetherall (2011), o *firewall* de estado é capaz de verificar se o tráfego recebido corresponde a uma conexão já estabelecida ou se é uma nova conexão. Isso é feito verificando as informações de cabeçalho das mensagens, como os endereços IP de origem e destino e as portas de origem e destino.

Conforme Stallings (2005), quando uma nova conexão for estabelecida, o *firewall* de estado cria uma entrada em sua tabela de estado, contendo informações como os endereços IP de origem e destino, portas de origem e destino, número de sequência e número de confirmação. Durante o fluxo da comunicação, o *firewall*

verifica as mensagens recebidas em relação ao estado das conexões estabelecidas, permitindo o tráfego que corresponde a conexões válidas e bloqueando o tráfego que não corresponde.

Segundo Parker (2023), o *firewall* de estado é uma técnica mais avançada do que os *firewall*s de filtros de pacotes, uma vez que é capaz de analisar o estado das conexões e permitir ou bloquear o tráfego com base nessa análise. Isso permite uma maior proteção contra ataques maliciosos, como os de negação de serviço (DoS), que tentam sobrecarregar um sistema com tráfego falso.

O *firewall* de estado é uma técnica de *firewall* que monitora o tráfego de rede e mantém registros do estado das conexões estabelecidas, permitindo apenas o tráfego que corresponde a conexões válidas. Essa técnica é mais avançada do que os *firewall*s de filtros de pacotes e oferece uma maior proteção contra-ataques maliciosos.

É importante considerar o tipo adequado para a rede em questão, levando em conta suas necessidades de segurança e orçamento disponível. Além disso, é necessário compreender a arquitetura de um *firewall*, incluindo suas camadas e componentes, para garantir a efetividade do sistema de segurança.

2.16 Arquiteturas de *firewall*

De acordo com Kurose e Ross (2013), a arquitetura de *firewall* é um conjunto de princípios e técnicas usados para implementar a segurança de rede por meio da distribuição de *firewall*s. Essa distribuição é feita com o objetivo de proteger a rede de possíveis ameaças externas ou internas, limitando o acesso a recursos e serviços disponíveis na rede. A arquitetura de *firewall* é fundamental para garantir a segurança da rede, pois define como os *firewall*s serão posicionados e configurados para controlar o tráfego de rede e impedir que ameaças maliciosas possam comprometer a segurança do sistema. É importante lembrar que a escolha da arquitetura de *firewall* deve ser feita com base nas necessidades de segurança da rede e no grau de proteção desejado.

A arquitetura de *firewall*s é uma parte crucial da segurança de rede, e pode ser implementada de várias maneiras. Segundo Kurose e Ross (2013), a arquitetura de

*firewall*s pode ser feita de três formas: *Dual-Homed Host*, *Screened Host* e *Screened Subnet.*

A arquitetura *Dual-Homed Host* é uma estrutura de *firewall* em que um *firewall* é instalado em uma máquina que possui duas interfaces de rede, uma conectada à rede interna e outra à rede externa. Conforme Kurose e Ross (2013), essa arquitetura foi criada na década de 1990, período em que os *firewall*s eram comumente implementados em hardware dedicado. Ela é útil para controlar o acesso à rede interna e limitar o tráfego de rede entre redes distintas.

Nessa arquitetura, todo o tráfego da rede interna passa pelo *firewall*, que é responsável por filtrar o tráfego indesejado e permitir o tráfego legítimo. A interface da rede interna é protegida por um conjunto de regras de filtragem, que limita o acesso à rede externa. A interface da rede externa é protegida por um conjunto de regras que limita o acesso à rede interna. Assim, um atacante que tenha acesso à interface da rede externa não pode se conectar diretamente à rede interna.

Como afirmado por Mendes (2015), o *Dual-Homed Host* é uma arquitetura simples, que pode ser implementada em um servidor comum com duas placas de rede, e que permite proteger a rede interna de uma organização contra acessos não autorizados. No entanto, é importante destacar que essa arquitetura tem limitações em relação à escalabilidade e ao desempenho, sendo mais indicada para redes menores.

Screened Host é uma arquitetura de *firewall* que consiste em um *firewall* instalado em uma máquina que possui três interfaces de rede: uma conectada à rede interna, uma conectada à rede externa e uma conectada a uma rede desmilitarizada (DMZ). Segundo Parker (2023), a DMZ é uma rede semitransparente que fornece um ambiente seguro para hospedar serviços que precisam ser acessados tanto pela rede interna quanto pela rede externa.

Nessa arquitetura, todo o tráfego da rede interna passa pelo *firewall*, que é responsável por filtrar o tráfego indesejado e permitir o tráfego legítimo. A interface da rede interna é protegida por um conjunto de regras de filtragem, que limita o acesso à rede externa. A interface da rede externa é protegida por um conjunto de regras que limita o acesso à rede interna e à DMZ. Já a interface da DMZ é protegida por um conjunto de regras que limita o acesso à rede interna e permite o acesso à rede externa apenas para os serviços que estão hospedados na DMZ.

De acordo com Mendes (2015), a arquitetura *Screened Host* é recomendada para empresas que possuem serviços que precisam ser acessados tanto pela rede interna quanto pela rede externa, como servidores de e-mail e servidores Web. Essa arquitetura é mais complexa que a *Dual-Homed Host*, mas oferece maior segurança e flexibilidade.

Screened Subnet é uma arquitetura de *firewall* que usa duas camadas de proteção para aumentar a segurança da rede. Essa arquitetura foi descrita por Kurose e Ross (2013) como uma evolução da arquitetura *Dual-Homed Host*, sendo mais escalável e mais adequada para redes maiores.

Na arquitetura de *Screened Subnet*, o *firewall* é instalado em uma máquina que tem três interfaces de rede: uma conectada à rede interna, outra conectada à rede externa e uma terceira conectada a uma sub-rede protegida, também chamada de DMZ (zona desmilitarizada). A sub-rede protegida é criada para hospedar serviços que precisam ser acessíveis a partir da Internet, como servidores de e-mail e de Web. O tráfego da rede externa é filtrado pelo *firewall* da mesma forma que na arquitetura de *Dual-Homed Host*. No entanto, na arquitetura de *Screened Subnet*, a sub-rede protegida adiciona uma camada extra de segurança. Os servidores na sub-rede protegida são protegidos por um conjunto adicional de regras de filtragem, que limita o acesso à rede interna.

Dessa forma, a arquitetura de *Screened Subnet* permite proteger a rede interna de uma organização de acessos não autorizados, enquanto permite que os serviços na sub-rede protegida sejam acessíveis a partir da Internet. Essa arquitetura é mais complexa do que a arquitetura de *Dual-Homed Host*, mas oferece um nível superior de segurança e flexibilidade.

No entanto, é importante destacar que as arquiteturas de *firewall* não garantem a segurança absoluta da rede. Como mencionado por Parker (2023), os *firewall*s são apenas uma das camadas de segurança que devem ser implementadas em uma rede, e é importante considerar outras medidas de segurança, como a criptografia, a autenticação de usuários e a detecção de intrusões.

De qualquer forma, é inegável a importância das arquiteturas de *firewall* na segurança da rede. Conforme descrito por Mendes (2015), as arquiteturas de *firewall* são uma das principais medidas de segurança em redes de computadores, e sua implementação é fundamental para evitar o acesso não autorizado à rede e garantir a integridade dos dados.

2.17 *Firewall* de Hardware e Software.

*Firewall*s são ferramentas fundamentais para garantir a segurança de redes, servidores e computadores contra ameaças externas e internas. Existem duas principais soluções de *firewall* disponíveis no mercado: o *firewall* de *hardware* e o *software firewall* de *software*.

O termo "*Firewall* de *hardware*" ou “*Firewall* Físico" refere-se a uma solução de segurança baseada em dispositivos físicos que operam como *firewall*s. Essa solução é executada em um dispositivo de hardware dedicado, projetado especificamente para proteger uma rede contra ameaças externas e internas. Como descreve Parker (2023), os *firewalls* de hardware são uma das soluções mais comuns e eficazes para proteger redes, servidores e computadores de ataques maliciosos, pois são capazes de filtrar o tráfego de rede com base em regras de segurança pré-definidas.

Segundo Stallings (2005), os *firewalls* de hardware dedicados têm a vantagem de possuírem desempenho superior aos *firewall*s baseados em *software*, uma vez que são projetados para realizar exclusivamente a função de filtragem de pacotes. Além disso, os *firewalls* de *hardware* geralmente possuem recursos adicionais de segurança, como a detecção de intrusões e a prevenção de ataques de negação de serviço.

Já o *Firewall* de *software* é uma solução de segurança baseada em *software* que é instalada em um computador para proteger uma rede contra acessos não autorizados. Essa solução utiliza uma combinação de regras de filtragem de pacotes, inspeção de estado e outras técnicas de segurança para monitorar e controlar o tráfego de rede. Conforme Kurose e Ross (2013), o *firewall* de *software* é uma solução flexível que pode ser implementada em uma ampla variedade de sistemas operacionais e *hardware* de computador. No entanto, a eficácia desse tipo de *firewall* depende muito da configuração correta das regras de segurança.

Já Parker (2023) afirma que o *firewall* de *software* é uma solução mais acessível do que o *firewall* de hardware, pois não requer um dispositivo dedicado. No entanto, a eficácia do *firewall* de *software* pode ser prejudicada por outros *software*s que estejam sendo executadas no mesmo computador.

Uma alternativa aos *firewall*s de hardware dedicados é a implementação de *firewall* baseado em servidor. Nesse caso, um servidor comum é utilizado para executar o *software* de *firewall*, que é responsável por filtrar o tráfego de rede. Mendes

(2015) destaca que essa solução é mais flexível do que os *firewall*s de hardware dedicados, uma vez que é possível utilizar hardware mais comum e configurar o *software* de acordo com as necessidades específicas da organização.

Outra vantagem dos *firewall*s baseados em servidor é a capacidade de personalização do *software* de *firewall*. Como afirma Mendes (2015), é possível configurar o *firewall* de *software* de acordo com as necessidades específicas da organização, permitindo que os administradores de rede criem regras de segurança personalizadas e adaptadas às suas necessidades.

Além disso, os *firewall*s baseados em servidor também podem ser implementados em sistemas operacionais e plataformas de hardware diferentes, aumentando a flexibilidade dessa solução de segurança. Segundo Kurose e Ross (2013), isso permite que as organizações escolham a plataforma que melhor se adapte às suas necessidades e orçamento.

A implementação de *firewall*s baseados em servidor não é complexa e não requer conhecimentos especializados em hardware. De acordo com Mendes (2015), o *software* de *firewall* pode ser facilmente instalado em um servidor existente e configurado pelos administradores de rede da organização. Isso torna a solução mais acessível e econômica do que os *firewall*s de hardware dedicados.

2.18 Linux

O sistema operacional Linux é amplamente utilizado em servidores, principalmente em *firewall* baseado em servidor, pois permite o controle de acesso à rede e a detecção de possíveis ataques, conforme explicado por Rash (2007). No entanto, o Linux não se limita a essa função, ele é um sistema operacional completo e robusto, utilizado em diversos dispositivos e áreas, desde servidores Web até sistemas embarcados em dispositivos móveis. De acordo com Hertzog et al. (2017), o Linux se destaca pela sua flexibilidade e personalização, permitindo que seus usuários adaptem o sistema às suas necessidades.

A história do Linux começou em 1991, quando Linus Torvalds, um estudante finlandês, criou o sistema operacional como um *hobby*. O objetivo de Torvalds era criar um sistema operacional similar ao Unix, porém com código aberto e disponível para todos. Com o tempo, o Linux foi ganhando adeptos e se popularizou, tornando-

se uma das opções mais utilizadas em servidores e em dispositivos de Internet das Coisas (IoT), como mencionado por Kanagachidambaresan (2021).

O Linux é um sistema operacional de código aberto, o que significa que seu código fonte é disponibilizado para todos e pode ser modificado e adaptado livremente. Essa característica torna o Linux uma ferramenta poderosa para desenvolvedores e programadores, permitindo que eles criem soluções personalizadas e específicas para suas necessidades.

Além da detecção e controle de acesso à rede, o uso do Linux como *firewall* em um servidor apresenta diversas outras vantagens. Rash (2007) destaca que o Linux possui uma arquitetura modular e escalável, permitindo que seja configurado de acordo com as necessidades específicas de cada ambiente. Além disso, o sistema conta com diversas ferramentas de monitoramento de rede, que auxiliam na identificação de possíveis ameaças.

Outra vantagem do uso do Linux em *firewall* é a sua segurança. O sistema operacional é conhecido por ser mais seguro em relação a outros sistemas operacionais, como o Windows. Segundo Newcomb (2017), a segurança do Linux é atribuída ao seu modelo de código aberto, que permite que especialistas em segurança possam auditar o código fonte e identificar possíveis vulnerabilidades.

A facilidade de configuração e administração é outra vantagem do Linux em *firewall*. Molloy (2016) destaca que o Linux oferece diversas ferramentas de configuração e gerenciamento, como o *Iptables*, que permite a configuração de regras de *firewall* de forma simples e eficiente.

Por fim, o custo é uma vantagem importante do Linux em *firewall*. Devido à sua licença de código aberto, o sistema operacional é gratuito, o que reduz significativamente o custo de implementação e manutenção em comparação com sistemas operacionais proprietários, como destaca Hertzog et al. (2017).

O uso do Linux como *firewall* em um servidor apresenta diversas vantagens, como modularidade, segurança, facilidade de configuração e administração e custo reduzido. Essas vantagens tornam o Linux uma opção atraente para empresas e organizações que buscam uma solução eficiente e econômica para proteger suas redes.

2.19 Single Board Computers

Com a flexibilidade dada pelo sistema operacional Linux, o uso do hardware normalmente utilizado em servidores pode ser substituído por *Single Board Computers* (SBCs), como destaca Kanagachidambaresan (2021). Estes dispositivos contêm todos os componentes necessários para um computador em uma única placa, como processador, memória, armazenamento, interfaces de rede e USB.

De acordo com Monk (2021), os SBCs são adequados para aplicações de IoT devido à sua capacidade de lidar com fluxos de dados em tempo real, baixo consumo de energia e facilidade de implantação em ambientes distribuídos. Além disso, esses dispositivos são mais acessíveis em termos de custo e possuem uma ampla variedade de opções de conectividade, como Wi-Fi e Bluetooth.

De acordo com Rash (2007), a utilização de *Single Board Computers* (SBCs) como *firewall*s pode ser uma solução mais econômica do que a aquisição de equipamentos especializados de *firewall*. Os SBCs são mais acessíveis em termos de custo e sua baixa exigência de recursos de hardware permite que eles sejam implantados em dispositivos de baixo custo, como o *Raspberry Pi*. Com isso, é possível obter uma solução de segurança para redes corporativas e domésticas de forma mais acessível e personalizada.

Segundo Molloy (2016), os SBCs com sistemas operacionais Linux são ideais para a implantação de *firewall*s devido à sua flexibilidade e capacidade de personalização. Esses dispositivos podem ser configurados para bloquear ou permitir o tráfego de rede com base em regras de *firewall* personalizadas, e o sistema operacional Linux oferece uma ampla variedade de ferramentas de segurança e monitoramento de rede.

Além disso, a utilização de SBCs como *firewall*s pode ser uma solução mais econômica do que a aquisição de equipamentos especializados de *firewall*, como destaca Monk (2021). Os SBCs são mais acessíveis em termos de custo, e sua baixa exigência de recursos de hardware permite que eles sejam implantados em dispositivos de baixo custo, como o *Raspberry Pi*.

Os Single Board Computers são dispositivos cada vez mais utilizados em soluções de IoT devido à sua versatilidade e facilidade de uso. Além disso, esses dispositivos também podem ser utilizados como *firewall*s baseados em *software*,

fornecendo segurança e proteção para redes corporativas e domésticas de forma econômica e personalizada.

2.20 *Raspberry Pi*

O *Raspberry Pi* é um computador de placa única (SBC) desenvolvido pela *Raspberry Pi* Foundation. Segundo Kanagachidambaresan (2021), o objetivo inicial do *Raspberry Pi* era oferecer uma plataforma de baixo custo para a educação em programação e computação, mas rapidamente se tornou popular em diversas áreas, desde projetos de automação residencial até soluções corporativas de IoT.

Ele é caracterizado por sua flexibilidade, já que pode ser utilizado para diferentes finalidades e programado em diversas linguagens. Newcomb (2017) destaca que o *Raspberry Pi* é uma plataforma ideal para criadores que desejam experimentar e desenvolver projetos com tecnologia.

Além disso, o *Raspberry Pi* possui um hardware completo em uma única placa, incluindo processador, memória RAM, interfaces de rede e USB, além de suporte a vídeo e áudio. Segundo Molloy (2016), o *Raspberry Pi* é capaz de rodar sistemas operacionais completos, como o Linux, e oferece uma grande variedade de periféricos e acessórios, permitindo a expansão de suas funcionalidades.

O *Raspberry Pi* também é conhecido por sua baixa exigência de recursos de hardware, o que o torna uma opção econômica e acessível para diferentes projetos. Hertzog et al. (2017) destacam que o *Raspberry Pi* é um dispositivo poderoso o suficiente para executar uma distribuição de teste de penetração como o Kali Linux, permitindo a realização de testes de segurança em redes e sistemas.

O *Raspberry Pi* 4, lançado em 2019, é a quarta geração do *Raspberry PI*. Ele apresenta um processador ARM Cortex-A72 de 64 bits, que é significativamente mais rápido do que os processadores utilizados em versões anteriores. Além disso, possui até 8 GB de memória RAM, opções de armazenamento em cartão SD ou em discos externos, conectividade Wi-Fi e Bluetooth, além de interfaces Ethernet, USB e HDMI (*Raspberry Pi*).

2.21 OpenWrt

O OpenWrt (*open wireless router*) é uma distribuição Linux de código aberto projetada especificamente para roteadores e dispositivos embarcados, oferecendo um conjunto poderoso de recursos para construção de *firewall*s e aprimoramento da segurança de redes (OpenWrt). o OpenWrt destaca-se por sua altíssima flexibilidade e personalização, permitindo uma configuração adaptada às necessidades específicas do ambiente de rede.

Uma das principais vantagens do OpenWrt segundo Granderath e Schonwalder é a capacidade de transformar um roteador comum em um dispositivo de segurança avançado. Com ele, é possível implementar *firewall*s de camada 3 e camada 7, filtrar o tráfego de entrada e saída, criar túneis VPN para proteger a comunicação de rede e muito mais. O sistema operacional é altamente modular, permitindo a instalação e configuração apenas dos serviços e aplicativos de segurança necessários, resultando em um consumo eficiente de recursos.

O OpenWrt inclui o *firewall* Netfilter/Iptables, semelhante ao mencionado para o *Raspberry Pi*, oferecendo configurações avançadas para controlar estritamente o tráfego de rede. Além disso, suporta o uso de pacotes de software adicionais, como Snort e Suricata, que são sistemas de detecção de intrusões (IDS) capazes de monitorar o tráfego de rede em tempo real, identificando comportamentos suspeitos e ameaças para uma resposta rápida e eficaz.

Outro ponto forte do OpenWrt é a capacidade de monitorar o tráfego de rede de forma detalhada, permitindo a identificação de comportamentos suspeitos e o rastreamento de eventos de segurança. Essa capacidade é essencial para a proteção proativa da rede, especialmente em ambientes corporativos.

3 Implantação do *firewall*

Este capítulo tem como objetivo mostrar o processo de implantação, expondo os procedimentos utilizador para instalação e configuração do *firewall*. O método de implantação é mostrando suas respectivas funcionalidades.

3.1 Componentes de hardware

Esta seção descreverá os componentes de hardware para a implementação do *firewall*.

3.1.1 *Raspberry Pi* 4

Raspberry Pi 4, desenvolvida pela *Raspberry Pi* Foundation, destaca-se como uma opção robusta para implementação em sistemas de *firewall*, oferecendo recursos que atendem às demandas específicas dessa aplicação.

De acordo com as especificações fornecidas pelo site oficial da *Raspberry Pi* Foundation (www.raspberrypi.com), a *Raspberry Pi* 4 é alimentada pelo chipset Broadcom BCM2711 Quad-core Cortex-A72, proporcionando desempenho excepcional para tarefas intensivas em processamento. A flexibilidade na capacidade de memória, variando de 1GB a 8GB de LPDDR4-3200 SDRAM, permite adaptação eficiente às necessidades de um *firewall*.

Figura 3- *Raspberry Pi*

Fonte: Elaborado pelo autor

A conectividade avançada, incluindo redes sem fio IEEE 802.11ac, Bluetooth 5.0 e Ethernet Gigabit, é fundamental para a comunicação eficaz em ambientes de segurança. As portas USB 3.0 e 2.0 possibilitam a conexão fácil de dispositivos externos, enquanto a compatibilidade com o padrão GPIO de 40 pinos facilita a integração em sistemas existentes de *firewall*.

A *Raspberry Pi* 4, com base nas especificações do site oficial, oferece uma combinação de potência de processamento, conectividade avançada e flexibilidade que a torna uma escolha vantajosa para implementações em sistemas de *firewall*.

3.1.2 Adaptador USB para *Ethernet*

O adaptador USB para Ethernet conta com o chipset Realtek RTL8153, conforme apresentado na figura 5. De acordo com a documentação da Realtek, esse chipset suporta velocidades de até 1 Gigabit por segundo (Gbps), utilizando a interface USB 3.0. Além disso, é compatível com os protocolos de rede do IEEE (*Institute of Electrical and Electronics Engineers*). Este dispositivo será empregado para expandir as entradas Ethernet das Raspberry, utilizando a porta USB disponível.

Figura 4 - Adaptador USB

Fonte: Elaborado pelo autor

3.2 Instalação do OpenWrt

Esta seção descreverá o processo de instalação do sistema operacional OpenWrt no Raspberry PI para a implementação do *firewall*.

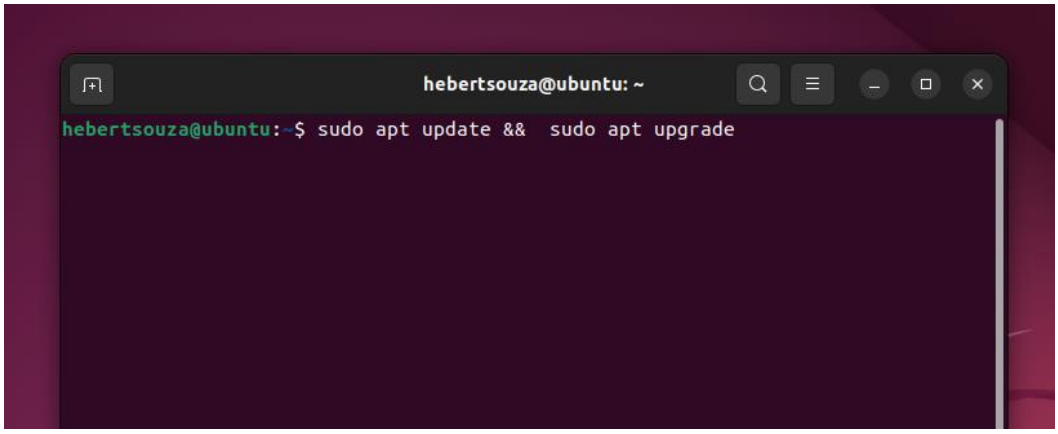
3.2.1 Configuração inicial

Para iniciar a configuração é necessário baixar o código-fonte do OpenWrt. Isso permitirá a criação da imagem de instalação adaptada ao hardware do *Raspberry Pi*. Esse procedimento é realizado utilizando um computador de mesa com o sistema operacional Linux Ubuntu Desktop 22.04.3 LTS. A preparação do ambiente de construção da imagem é fundamental para garantir um processo adequado de compilação e adaptação do sistema operacional ao hardware específico do *Raspberry Pi*.

Para prosseguir com a preparação do ambiente para a construção da imagem do OpenWrt adaptada ao hardware do *Raspberry Pi*, é fundamental realizar a atualização do sistema Linux e seus respectivos pacotes. Esse procedimento,

conforme indicado na figura 6, inicia com um comando que busca a listagem dos pacotes disponíveis para atualização. Em seguida, o sistema realiza o download desses pacotes, visando garantir que o sistema operacional e os pacotes associados estejam atualizados, fator crucial para o posterior processo de compilação do OpenWrt para o *Raspberry Pi*.

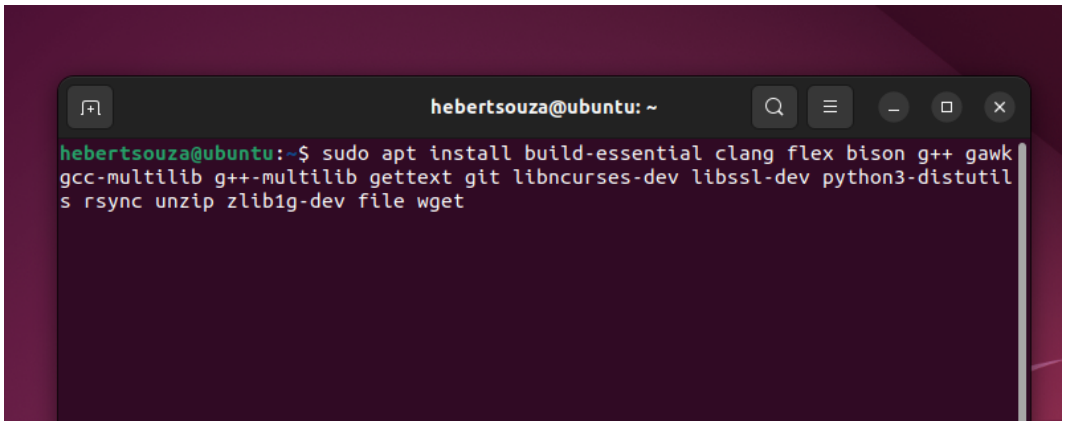
Figura 5- Atualizar Linux



Fonte: Elaborado pelo autor

Para a construção da imagem do OpenWrt, é necessário instalar os pacotes fundamentais, conforme ilustrado na figura 7. Esses pacotes são essenciais para a compilação e configuração do OpenWrt e na criação da imagem personalizada para o hardware específico do *Raspberry Pi*.

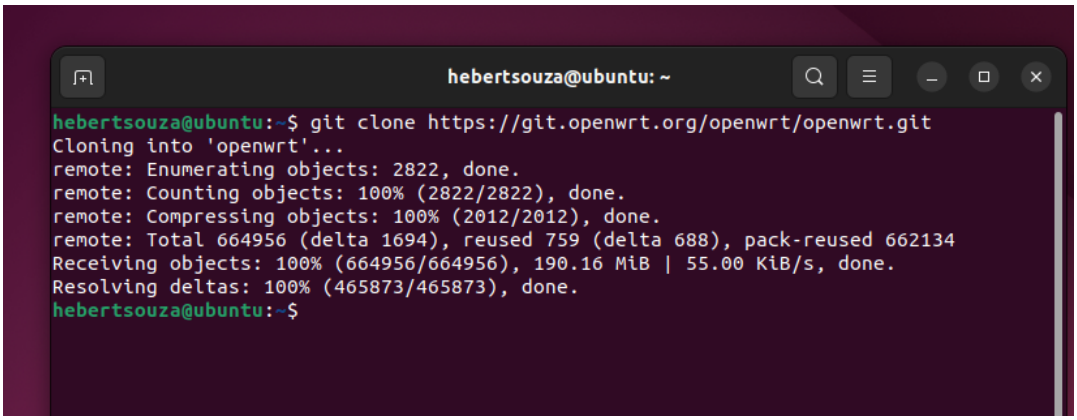
Figura 6- Instalando pacotes



Fonte: Elaborado pelo autor

Após a atualização e instalação dos programas, é necessário criar um diretório de trabalho e clonar o repositório do OpenWrt, como indicado na forma e figura 8.

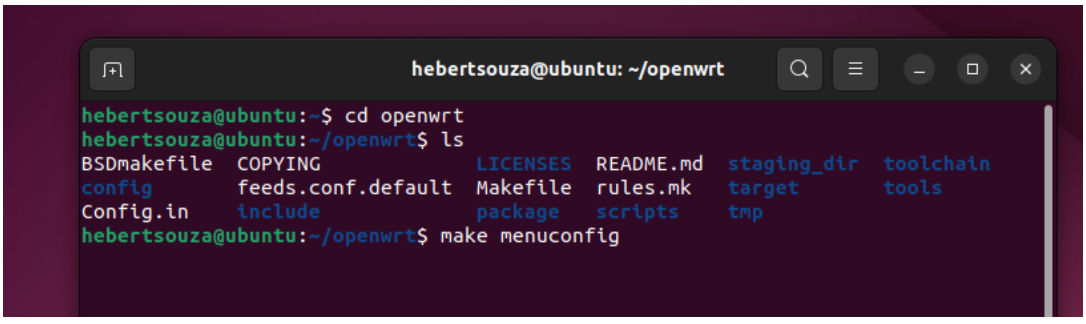
Figura 7 - Copiando do repositório



Fonte: Elaborado pelo autor

Com a cópia do projeto do OpenWrt, é o momento de iniciar a configuração do ambiente de construção da imagem. Para isso, é necessário inserir os comandos conforme demonstrado na figura 9. Esse comando tem a função de acessar a pasta do projeto e iniciar o menu de configuração do construtor.

Figura 8 - Pasta do OpenWrt

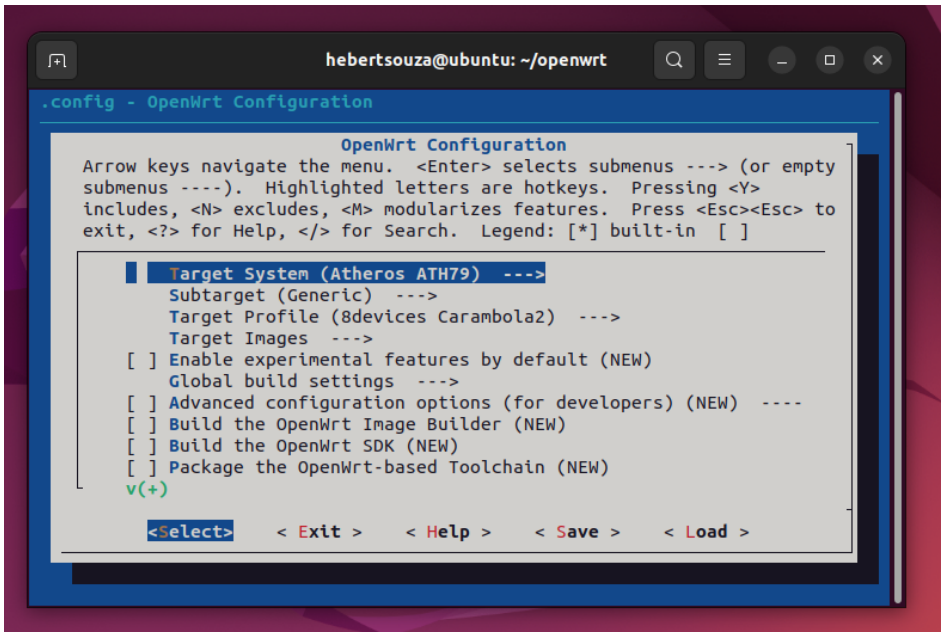


Fonte: Elaborado pelo autor

Através do menu de configuração do construtor, é possível configurar e montar a imagem com as características desejadas para o *firewall*, conforme ilustrado na

figura 10. Este processo permite definir e ajustar as especificidades necessárias para a construção da imagem do OpenWrt destinada ao *firewall*.

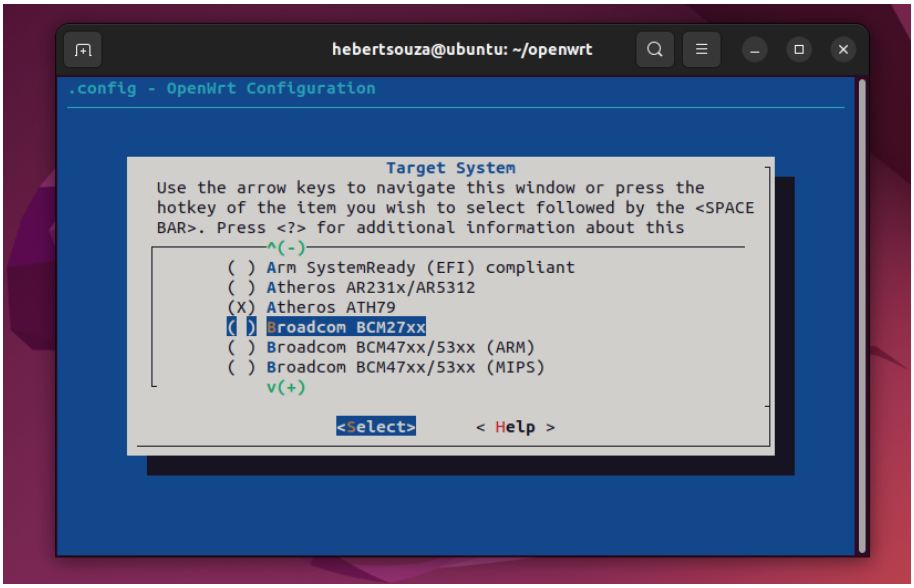
Figura 9 - Configurador do OpenWrt



Fonte: Elaborado pelo autor

Selecionando a primeira opção, pode-se escolher o "target" para a arquitetura do processador. Dado que o Raspberry é equipado com um processador Broadcom 2711, foi selecionada a arquitetura correspondente à linha de processadores Broadcom 27xx, conforme demonstrado na figura 11. Esta escolha é fundamental para garantir a compatibilidade e adaptação da imagem do OpenWrt ao processador específico do Raspberry.

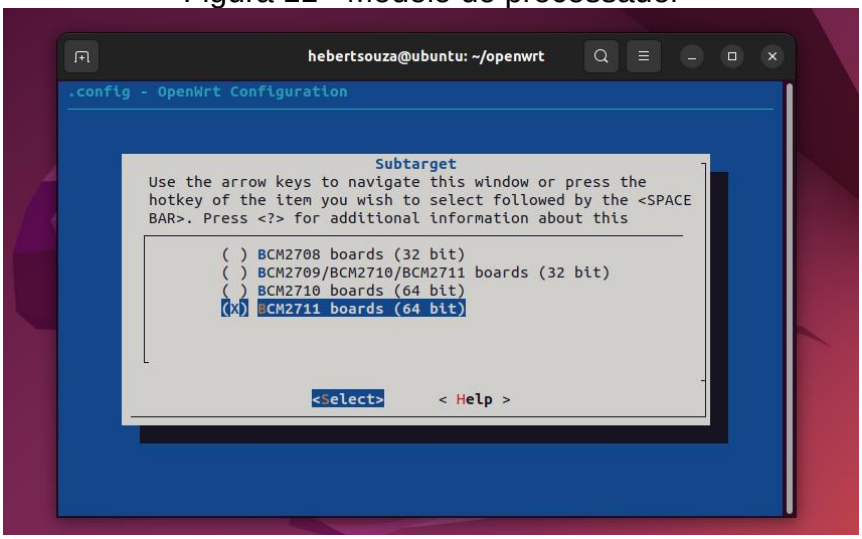
Figura 10 - Arquitetura da *Raspberry Pi*



Fonte: Elaborado pelo autor

Após selecionar o sistema de destino, o próximo passo é especificar o processador na opção “subtarget”. Optou-se por selecionar a opção correspondente ao modelo do processador Broadcom 2711 do *Raspberry Pi* 4, como ilustrado na figura 12. Esta etapa é essencial para direcionar e ajustar as configurações da imagem do OpenWrt para o processador específico do *Raspberry Pi*.

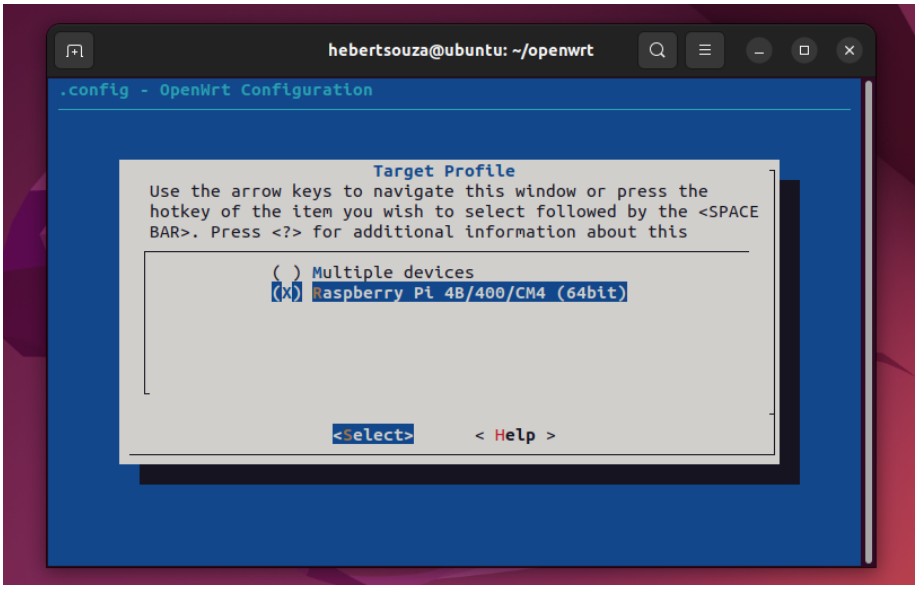
Figura 11 - Modelo do processador



Fonte: Elaborado pelo autor

Na etapa "Target Profile", opta-se pela seleção da opção "*Raspberry Pi* 4/n/400/cm4", conforme demonstrado na figura 13. Essa escolha representa um conjunto reconfigurado do kernel para o *Raspberry Pi* 4, fornecendo configurações específicas e pacotes essenciais para o funcionamento do sistema operacional. Este perfil é crucial para garantir que o sistema esteja devidamente equipado com os pacotes necessários e configurado adequadamente para operar no hardware do *Raspberry Pi* 4.

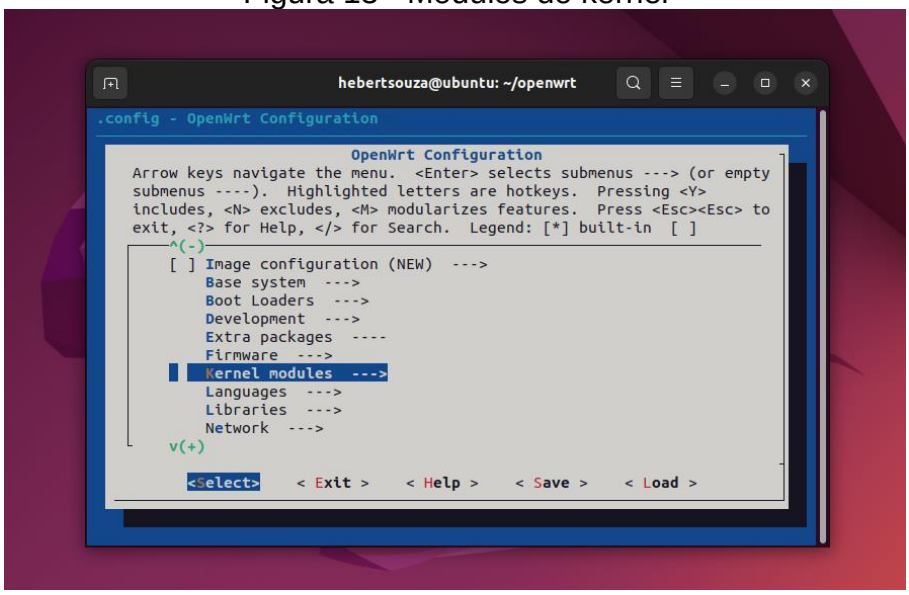
Figura 12 - Target profile



Fonte: Elaborado pelo autor

Nesta fase, a configuração dos módulos do kernel, ilustrada na figura 14, é fundamental para a personalização da imagem antes de ser gravada no dispositivo de armazenamento. Isso permite que a imagem seja replicada em outros dispositivos sem a necessidade de instalar os módulos do kernel Linux posteriormente à construção da imagem do sistema. Essa personalização é valiosa para garantir uma imagem pronta e completa para uso, facilitando sua implantação em múltiplos dispositivos sem a necessidade de instalações adicionais.

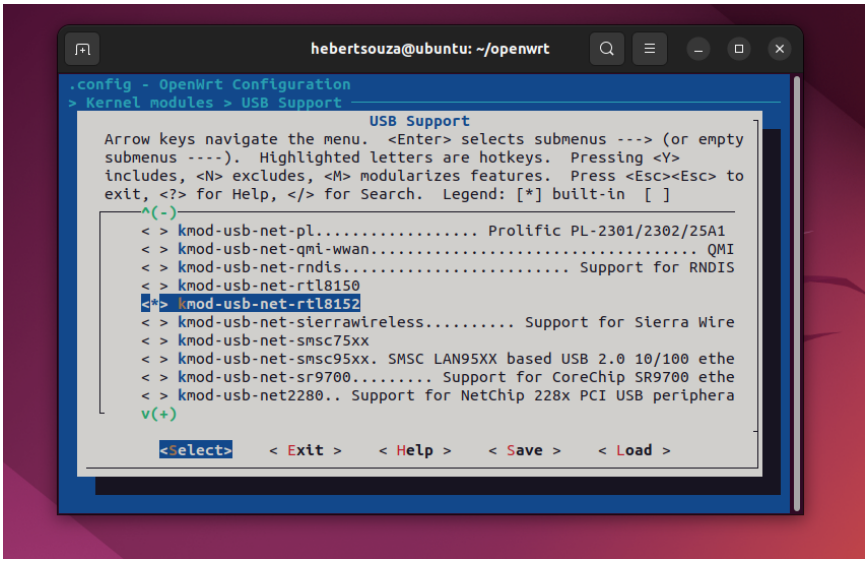
Figura 13 - Módulos do kernel



Fonte: Elaborado pelo autor

No caso da utilização de um adaptador USB para *ethernet* com o *chip* Realtek 8153, é necessário instalar o *driver* correspondente à sua versão dentro dos módulos, conforme demonstrado na figura 15. Este procedimento garante a integração do driver específico necessário para o funcionamento correto e compatibilidade com o adaptador USB Realtek 8153 dentro dos módulos do sistema.

Figura 14 - Modulo do adaptador Ethernet

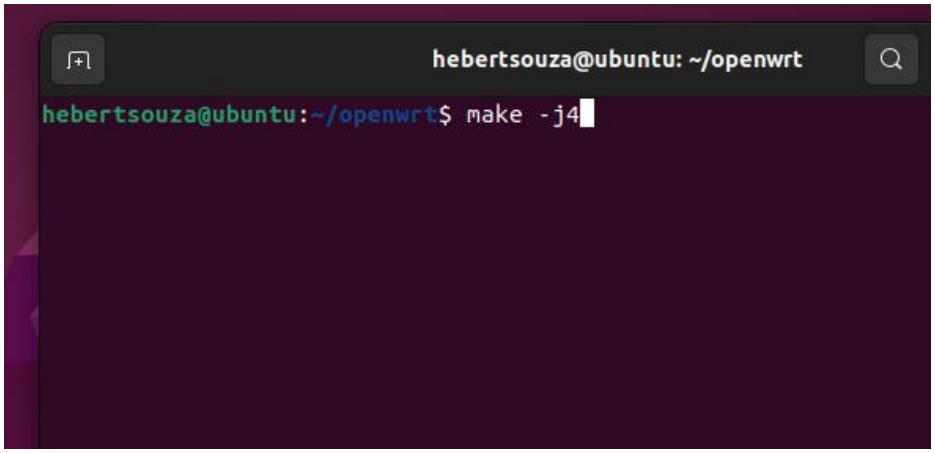


Fonte: Elaborado pelo autor

Após realizar a configuração, basta clicar em "Salvar" e, em seguida, digitar o comando conforme indicado na figura 16 para dar início ao processo de construção

da imagem. Esse comando é responsável por iniciar a compilação do sistema conforme configurado, resultando na geração da imagem customizada do OpenWrt com as especificações definidas.

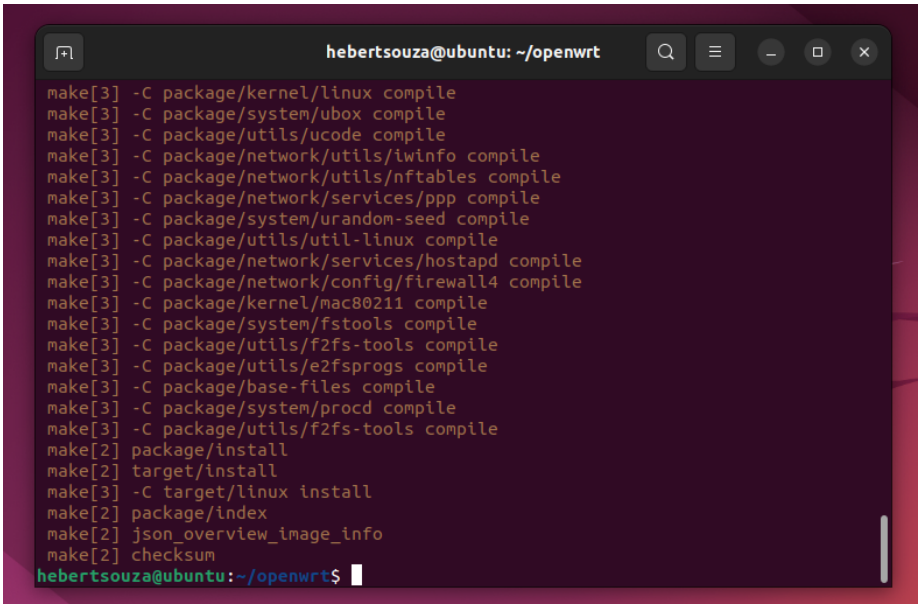
Figura 15 - Construindo a imagem



Fonte: Elaborado pelo autor

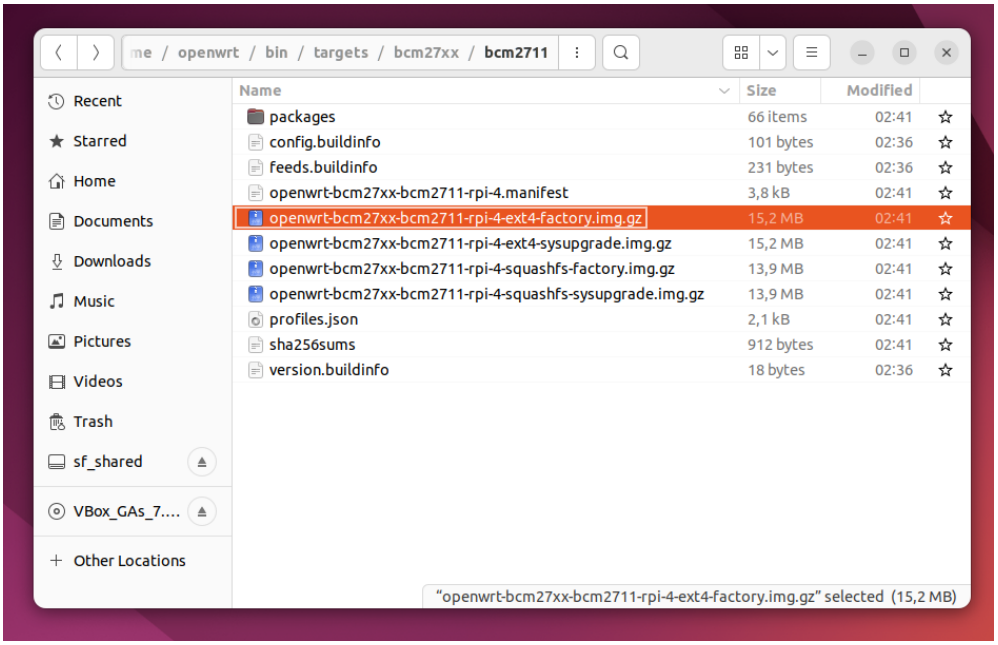
Após fazer a construção dos, como pode ser visto na figura 17, é produzido os arquivos de imagem do sistema operacional do OpenWrt, coma extensão do armazenamento selecionado, como mostrado na figura 18

Figura 16 - Construindo a imagem



Fonte: Elaborado pelo autor

Figura 17 - Imagem final do OpenWrt



Fonte: Elaborado pelo autor

3.2.2 Gravando a imagem

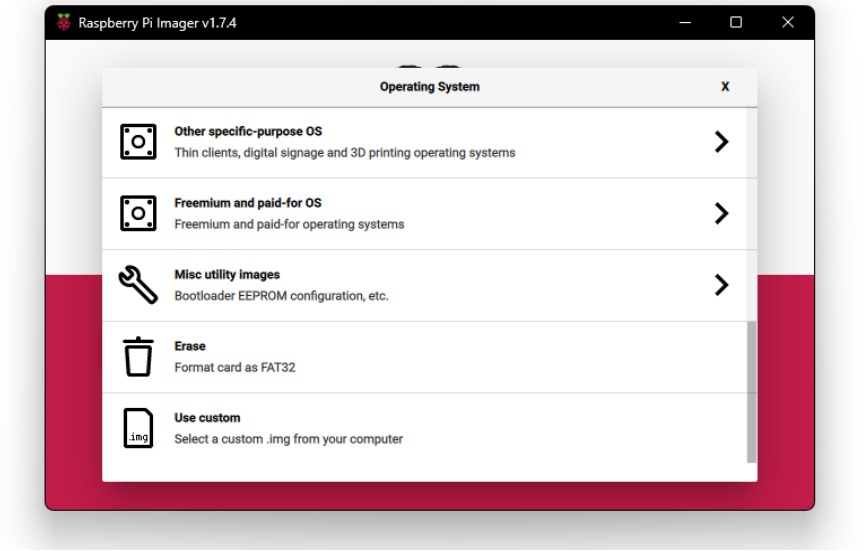
Nesta seção, descreve-se o processo de gravação da imagem do OpenWrt, desenvolvida para o *Raspberry Pi* 4. Para esta tarefa, foi selecionado o *Raspberry Pi Imager,* um software fornecido pela *Raspberry Pi*, projetado para a rápida e simples gravação de imagens em unidade de armazenamento. A interface deste software está ilustrada na figura 19.

Figura 18 - Interface do *Raspberry Pi* Imager

Fonte: Elaborado pelo autor

Para gravar a imagem do OpenWrt desenvolvida, basta clicar no botão "CHOOSE OS" para selecionar o sistema. Na sequência, é necessário navegar até a última opção e escolher o sistema operacional "Custom", conforme ilustrado na figura 20. Este procedimento permitirá a seleção da imagem do OpenWrt personalizada, preparando-se para a gravação no dispositivo de armazenamento desejado.

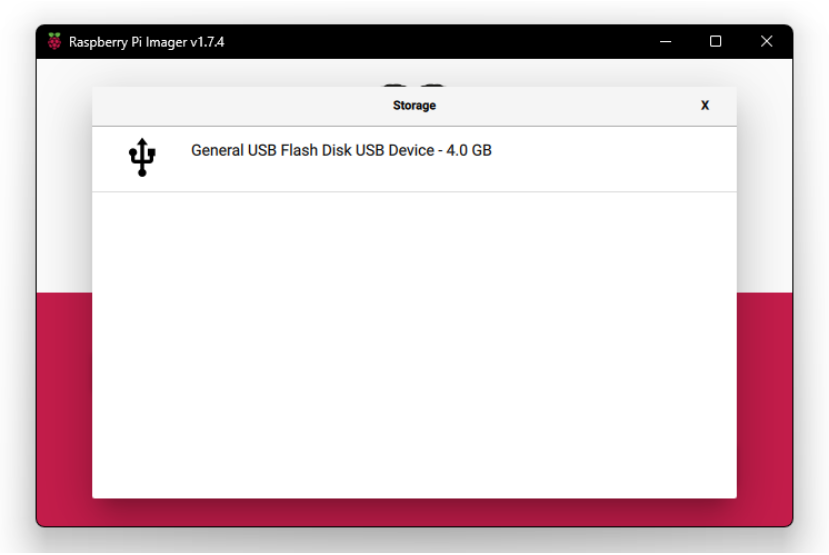
Figura 19 - Selecionar imagem do OpenWrt



Fonte: Elaborado pelo autor

Após a seleção do sistema operacional "Custom", é necessário, no segundo botão, escolher o dispositivo de armazenamento correspondente ao que será utilizado no *Raspberry Pi*, como indicado na figura 21. Este passo é crucial para garantir que a imagem personalizada do OpenWrt seja gravada no dispositivo de armazenamento específico que será utilizado no *Raspberry Pi*.

Figura 20 - Escolhendo a unidade de armazenamento



Fonte: Elaborado pelo autor

Completada essa etapa, basta clicar em "Write" e confirmar a formatação da unidade de armazenamento, seguindo as orientações fornecidas, para que o software possa realizar a gravação da imagem do sistema operacional OpenWrt no dispositivo de armazenamento selecionado, conforme demonstrado na figura 23. Este processo permite a correta gravação da imagem e a preparação do dispositivo para execução do OpenWrt no *Raspberry Pi*.

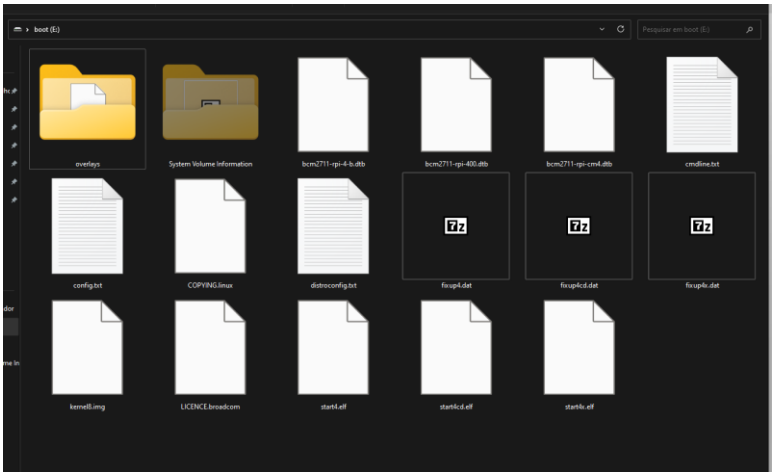
Figura 21 - Gravando a imagem

Fonte: Elaborado pelo autor

3.2.3 Configurações antes da inicialização

Nesta seção, é abordado o processo de configuração do OpenWrt após a gravação da imagem, que contém o conteúdo do sistema operacional, conforme ilustrado na figura 24. Essa imagem representa praticamente todo o sistema operacional que será executado pela *Raspberry Pi*.

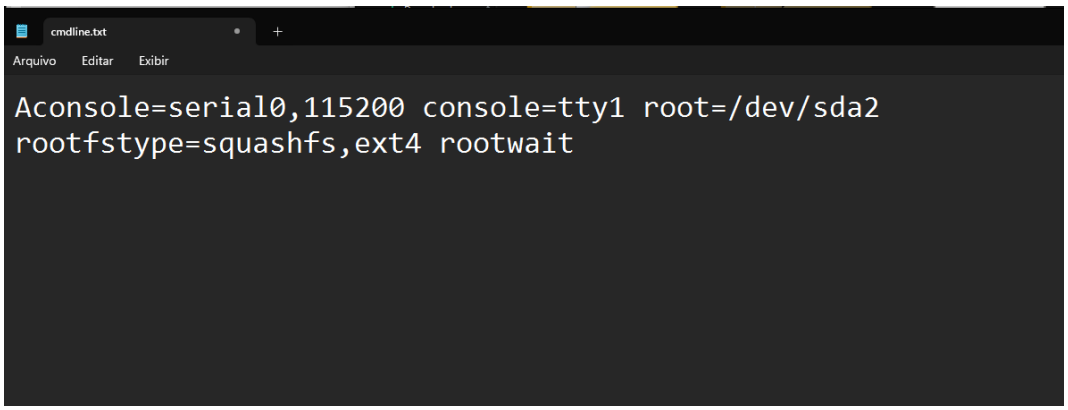
Figura 22 - Arquivos gravados na unidade de armazenamento



Fonte: Elaborado pelo autor

Existem arquivos específicos para as configurações, como o cmdline.txt, que configura os parâmetros de inicialização do sistema. Nesse caso, é importante fazer o mapeamento correto da unidade de armazenamento, conforme ilustrado na figura 25, onde está sendo mapeada a unidade sda2. Além disso, há o arquivo de configuração, onde são definidos os parâmetros necessários para o funcionamento da imagem do sistema operacional. Esses arquivos são essenciais para ajustar as configurações iniciais e garantir o funcionamento adequado do OpenWrt na *Raspberry Pi*.

Figura 23 - Arquivo de configuração

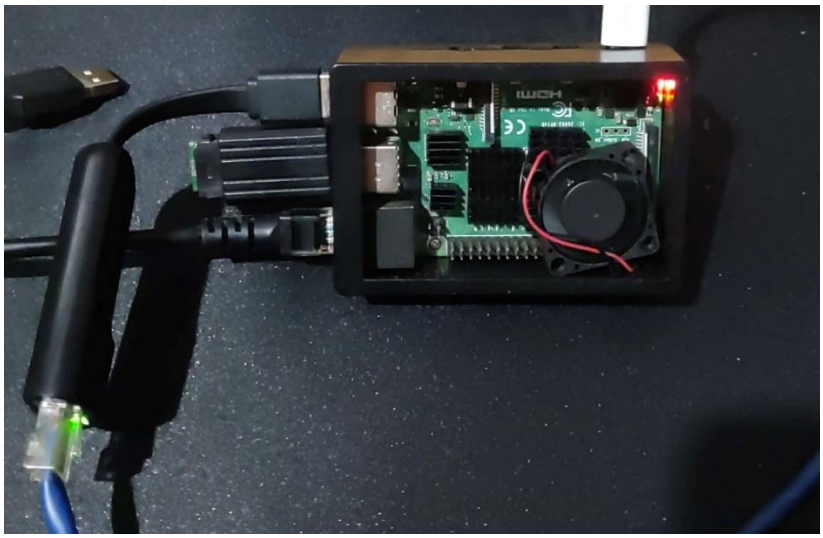


Fonte: Elaborado pelo autor

3.2.4 Configurações do OpenWrt

Nesta seção, é abordado o processo de configuração do OpenWrt após a pré-configuração e montagem da Raspberry, conforme ilustrado na figura 26. Nesse estágio, já é possível conectar a Raspberry e começar a realizar a configuração do *firewall*, iniciando a customização e ajustes de acordo com as necessidades e objetivos específicos.

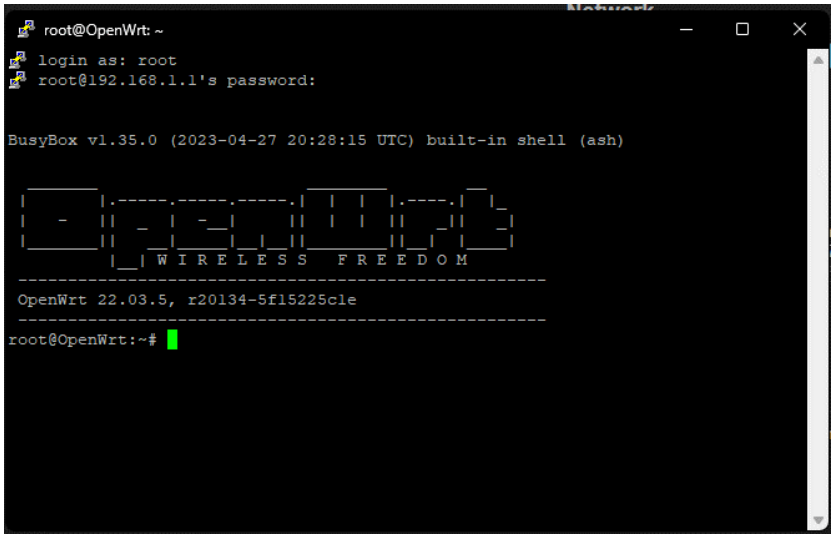
Figura 24 - Raspberry montada



Fonte: Elaborado pelo autor

Ao iniciar o sistema e estabelecer a conexão via SSH, é necessário configurar a senha padrão do sistema. Após esse procedimento, é apresentada a tela inicial, conforme mostrado na figura 27. Esta tela oferece as opções iniciais e os recursos básicos para a configuração e gerenciamento do OpenWrt, permitindo ajustar as configurações do *firewall* e personalizar a segurança de acordo com as necessidades específicas.

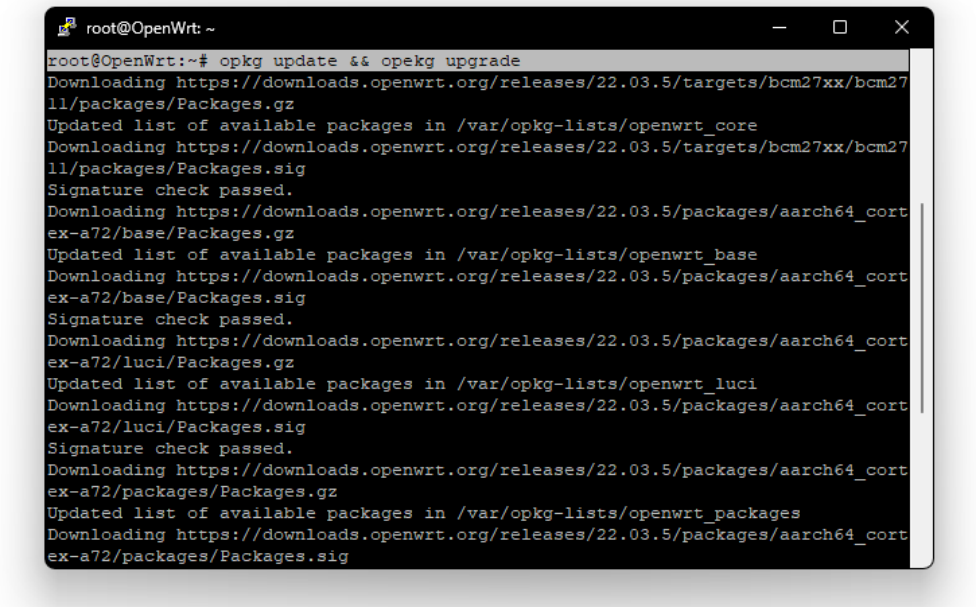
Figura 25 - Tela de abertura do OpenWrt



Fonte: Elaborado pelo autor

Após essa etapa inicial, é importante atualizar o sistema para garantir acesso a todos os pacotes mais recentes. O comando utilizado para essa atualização pode ser visualizado na figura 28. Isso assegura que o sistema esteja atualizado e apto a acessar as últimas versões e atualizações dos pacotes disponíveis.

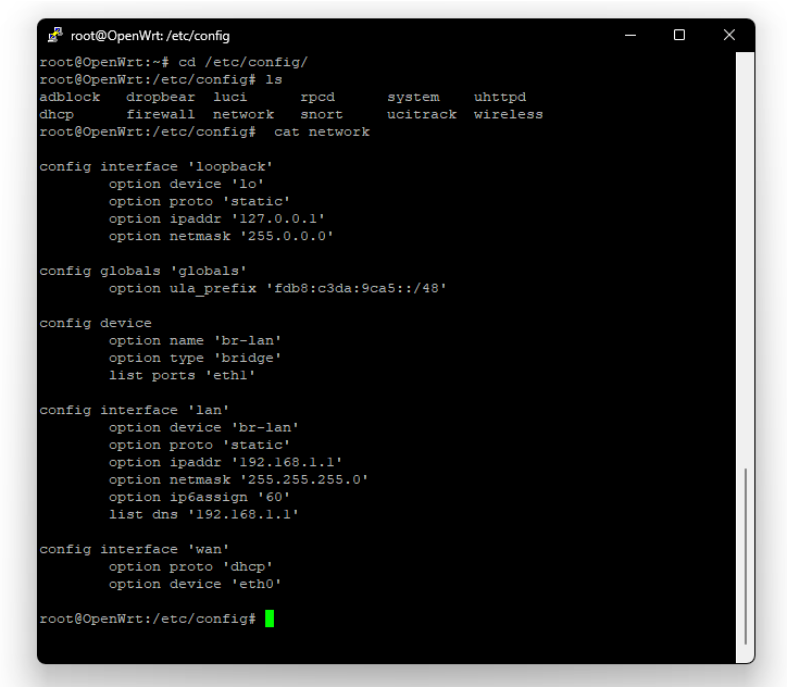
Figura 26 - Atualizando o OpenWrt



Fonte: Elaborado pelo autor

Após completar essa etapa, é viável ajustar as configurações do Raspberry. Os arquivos para tal ajuste estão localizados na pasta “/etc/config”. O primeiro arquivo para modificar é o arquivo "network", responsável pela configuração da conexão de Internet e seu funcionamento. Após realizar as configurações, é possível visualizar o resultado dessas alterações nas informações exibidas, conforme ilustrado na figura 29.

Figura 27 - Arquivo de configura



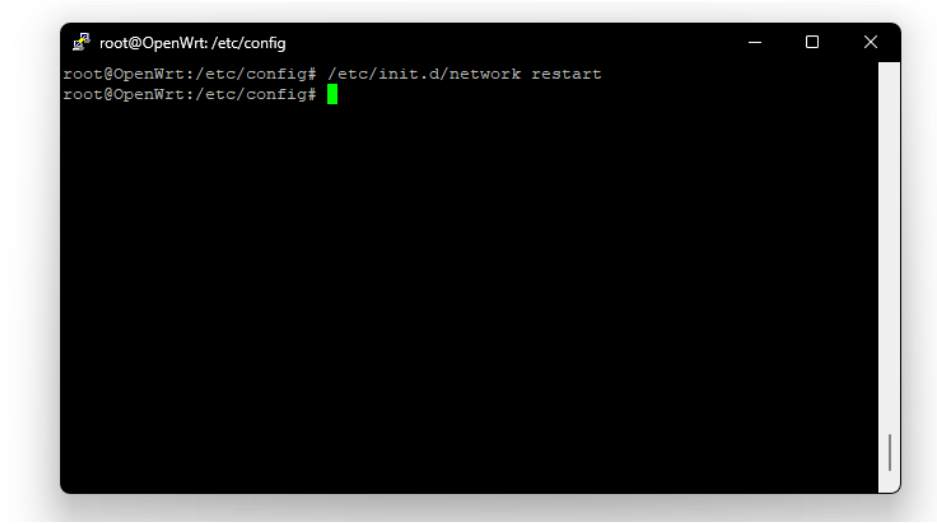
Fonte: Elaborado pelo autor

As configurações de “loopback” geralmente se referem à parte local da conexão do sistema. Por outro lado, as configurações “globais” são informações que se aplicam a todas as interfaces de rede. Na seção “devices” são ajustadas as configurações para as interfaces virtuais. No cenário mencionado, está sendo configurada uma rede local com uma ponte sobre a entrada “eth1”, que é o adaptador USB Ethernet utilizado.

Posteriormente, na seção “lan”, é realizada a configuração da rede local. É definido que “br-lan” será a interface de rede com um IP estático, como 192.168.1.1. O parâmetro “ip6assign” ativa o protocolo IPv6, enquanto o comando list dns configura o Raspberry como o servidor DNS da rede LAN. Na figura 29, são apresentadas as configurações para a rede “wan”, a qual se refere à conexão com a Internet. No cenário mencionado, está sendo utilizado o modo DHCP e a conexão é feita através da entrada Ethernet do próprio *Raspberry Pi*.

Após realizar essas configurações, é necessário reiniciar o serviço de rede do OpenWrt, conforme demonstrado na figura 30. Isso assegura que as alterações feitas nas configurações de rede sejam aplicadas e que o sistema passe a funcionar conforme as novas definições.

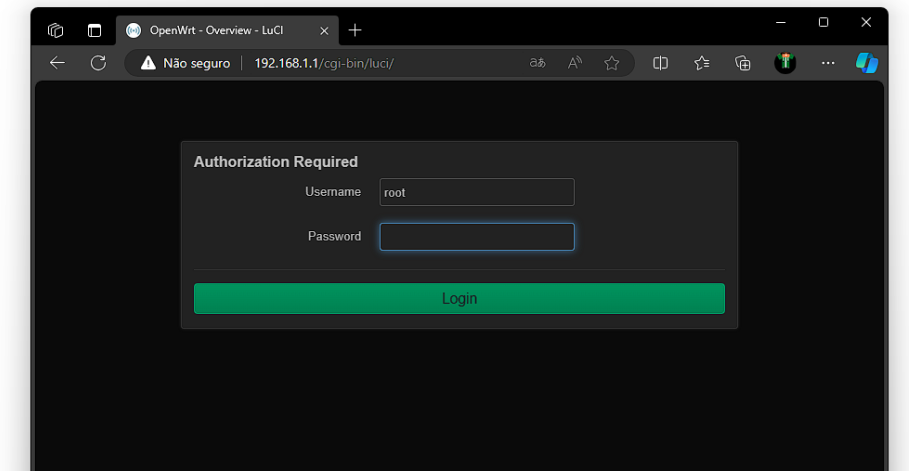
Figura 28 - Reiniciando a interface de Internet



Fonte: Elaborado pelo autor

Realizando esses procedimentos, a Raspberry consegue estabelecer a conexão com a Internet e rotear os dados entre as interfaces, funcionando efetivamente como um roteador de Internet. Agora, é possível configurar o sistema através da interface Web fornecida pelo OpenWrt. Considerando que o endereço IP padrão para o Raspberry é 192.168.1.1, basta inseri-lo no navegador para acessar a interface. Conforme demonstrado na figura 31, será exibida a tela de login, na qual é necessário inserir o usuário "root" e a senha criada anteriormente.

Figura 29 - Interface Web do OpenWrt

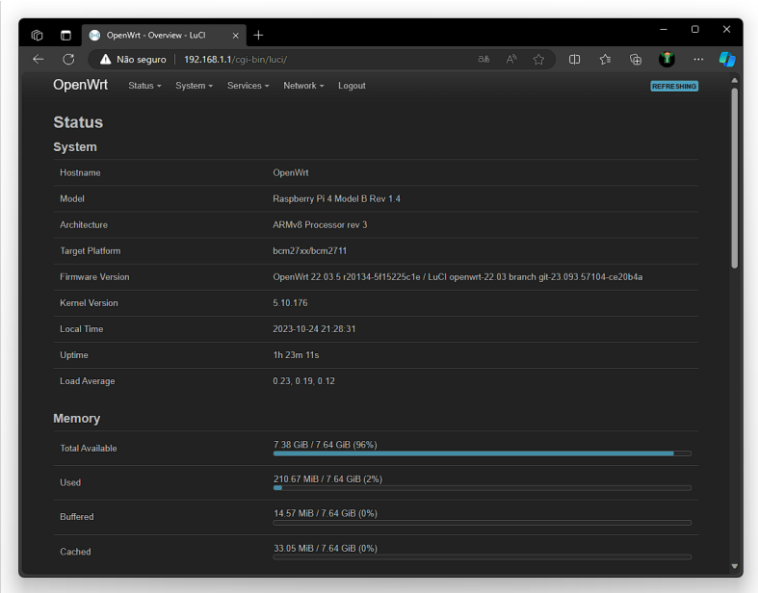


Fonte: Elaborado pelo autor

Após o login, é exibida a tela ilustrada na figura 32, onde são apresentadas informações sobre o hardware que está executando o OpenWrt, como o uso de

memória, além de detalhes adicionais. Também é exibida a barra de menu, caracterizando essa interface como uma interface de administrador, permitindo acesso a configurações e gerenciamento detalhados do sistema.

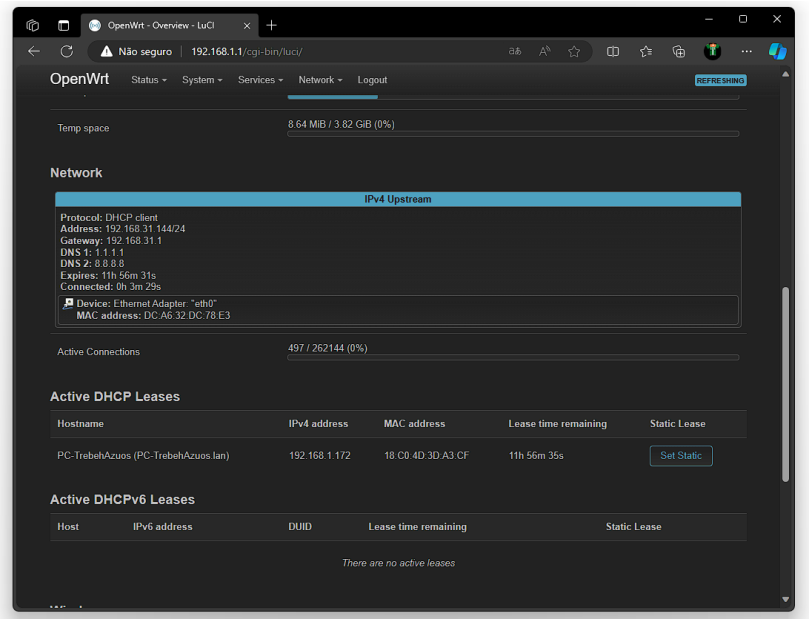
Figura 30 - Tela de início do OpenWrt



Fonte: Elaborado pelo autor

Essa interface exibe informações detalhadas, principalmente sobre os dispositivos conectados e ativos, além do número de conexões que o sistema atualmente mantém, como evidenciado na figura 32. Essa visualização oferece uma perspectiva abrangente do status e da atividade da rede, permitindo um controle mais refinado e específico.

Figura 31 - Mostrando as redes do OpenWrt

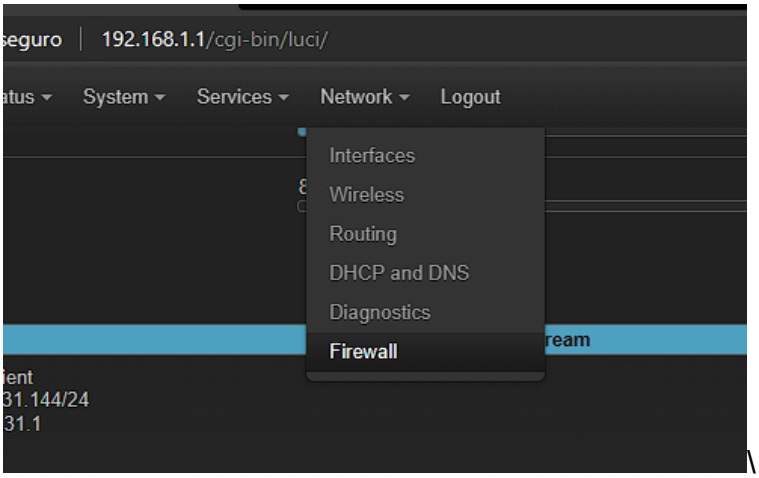


Fonte: Elaborado pelo autor

3.2.5 Configurações do *firewall*

Usando a interface Web, é possível configurar o *firewall* acessando a seção "network" e, em seguida, "*firewall*", conforme indicado na figura 33. Isso concede acesso à tela de configuração do *firewall*, onde é possível ajustar as configurações de segurança e definir as políticas de filtragem de tráfego.

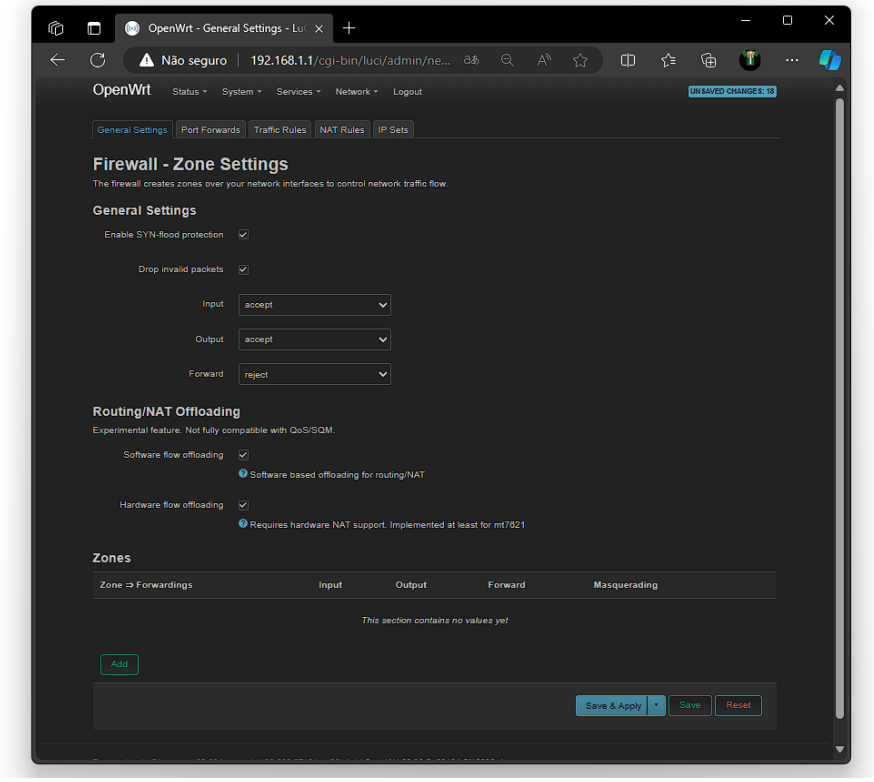
Figura 32 - Navegando para o *Firewall*



Fonte: Elaborado pelo autor

Ao acessar a tela de *firewall*, é exibida a interface ilustrada na figura 34. Nessa área, encontram-se as configurações gerais, permitindo a configuração das zonas do *firewall*. Isso possibilita a definição e personalização das zonas de segurança para o controle do tráfego de rede.

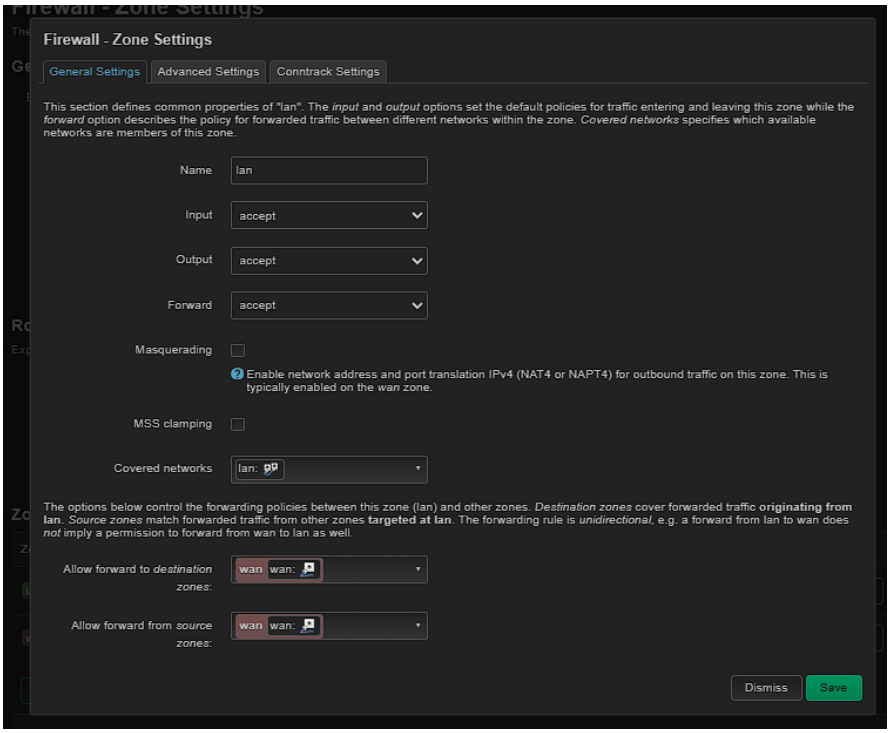
Figura 33 - Tela de Configuração do *Firewall*



Fonte: Elaborado pelo autor

Ao clicar no botão abaixo de "add", é possível criar uma zona de *firewall*, conforme ilustrado na figura 35. Nesse caso, está sendo configurada a zona "lan", determinando suas funcionalidades para a rede "wan". Este processo permite estabelecer parâmetros específicos de segurança e comportamento para diferentes áreas da rede.

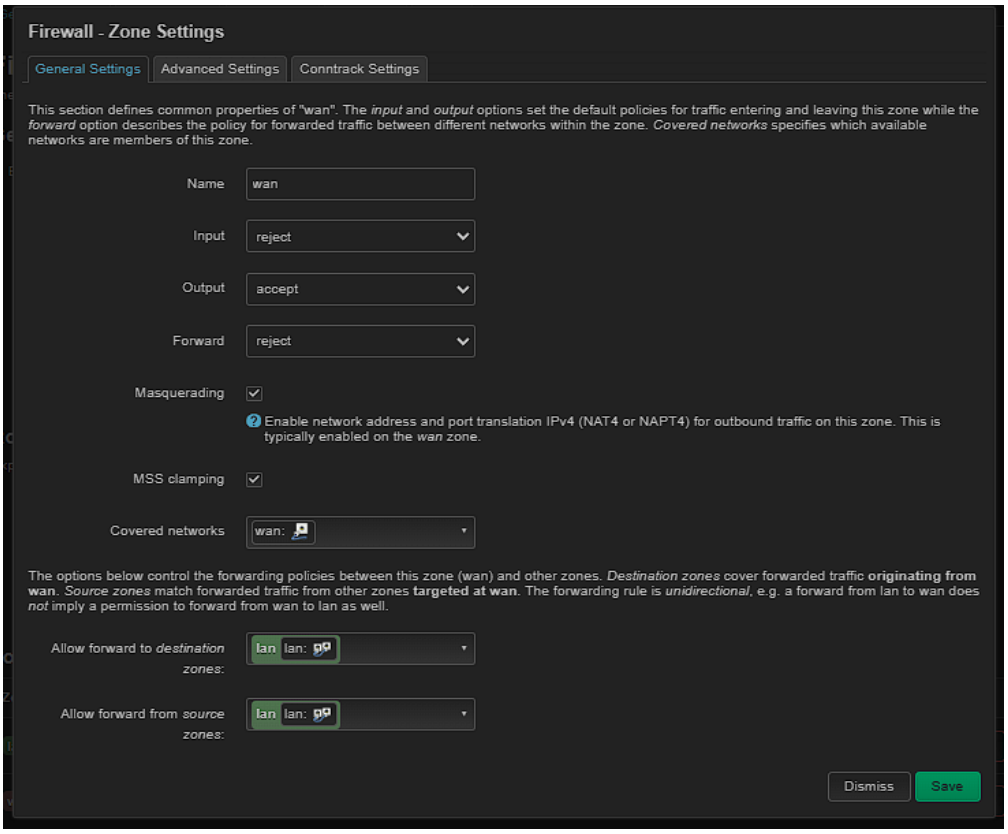
Figura 34 - Configurando o *firewall* para rede wan



Fonte: Elaborado pelo autor

Após configurar a rede "lan", é necessário configurar a rede "wan", como ilustrado na figura 36. Nessa etapa, o mascaramento de IP será habilitado para aumentar a segurança da rede "lan". Isso evita a exposição direta dos IPs da "lan" na rede "wan", contribuindo para uma camada adicional de proteção.

Figura 35 - Configurando o *firewall* para rede lan



Fonte: Elaborado pelo autor

4 Teste Com o *Firewall*

Este capítulo tem como objetivo mostrar o processo de teste do firewall, expondo os procedimentos utilizados para realizar os testes do firewall, de forma que seja possível extrair informações para que possam ser analisadas posteriormente.

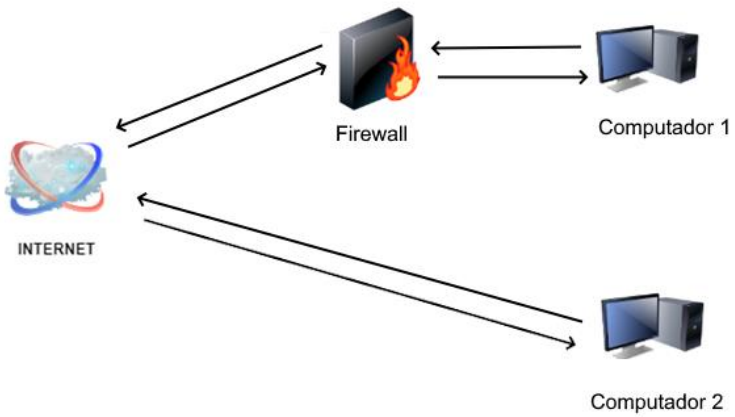
4.1 Ambiente de teste

Esta seção descreve o ambiente de teste do *firewall* implementado.

4.1.1 Estrutura da rede de teste

O teste é conduzido em um ambiente de rede local composto por dois computadores, um *firewall* e uma conexão com um roteador para acesso à Internet. A configuração da rede de teste está ilustrada na Figura 37. O propósito deste ambiente é avaliar o funcionamento e o desempenho do *firewall*. O objetivo do teste é determinar se o *firewall* afeta o desempenho da rede e, caso isso ocorra, quantificar o grau desse impacto.

Figura 36 - Cenário de teste



Fonte: Elaborado pelo autor

4.1.2 Qualidade de serviço

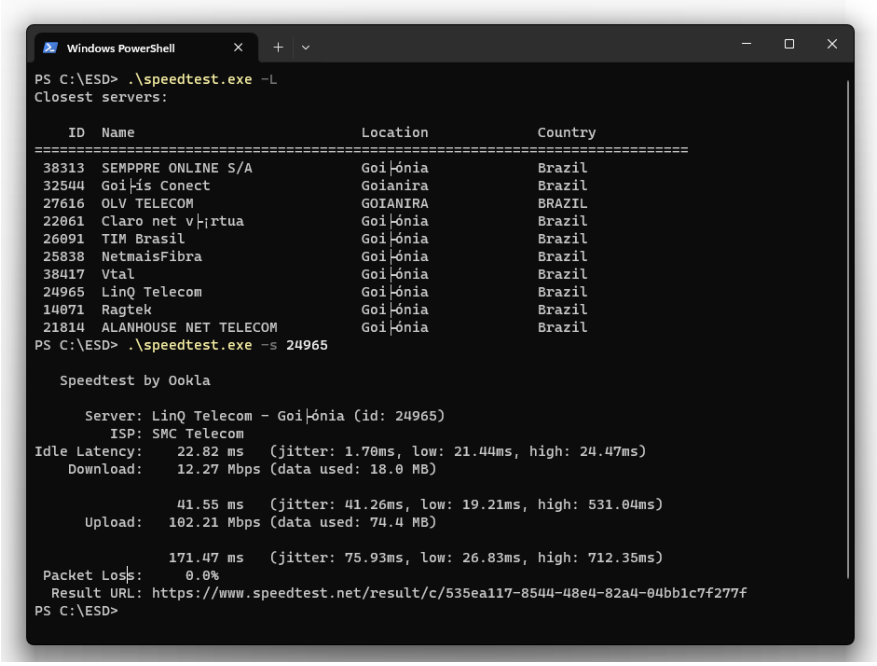
O teste de Qualidade de serviço (Quality of Service, QoS) segundo o Chadda (1999) refere-se à capacidade de uma conexão de Internet fornecer serviços de forma consistente e confiável, garantindo uma experiência satisfatória para os usuários. Isso envolve vários parâmetros, incluindo:

- **Velocidade de Conexão:** A largura de banda disponível para a transferência de dados, medida em megabits por segundo (Mbps) ou gigabits por segundo (Gbps).
- **Latência:** O tempo que leva para um pacote de dados percorrer a rede de origem para destino, influenciando a responsividade da conexão
- **Perda de Pacotes:** A porcentagem de pacotes de dados que não são entregues com sucesso, afetando a integridade das comunicações.
- **Jitter:** A variação no atraso de entrega de pacotes, o que pode resultar em uma experiência de uso inconsistente, especialmente
- **Disponibilidade:** A confiabilidade da conexão, indicando o tempo em que a Internet está disponível e funcional.

Para esse teste de QoS foi escolhido a ferramenta speedtest-cli, uma ferramenta multiplataforma, pois segundo Brito (2023), essa ferramenta verifica a velocidade da Internet utilizando apenas o terminal, além de que ela tem todas as informações citadas por Chadda para poder fazer uma métrica ideal de uma qualidade de serviço.

Com a ferramenta instalada na máquina, deve-se escolher um servidor e utilizar ele para os testes, a realização do teste é feita com o *firewall* e sem o *firewall* na rede, como descrito na figura 38, mostrando como é a informações mostrada na ferramenta speedtest-cli, pode-se ver uma lista de servidores e que todas as informações que buscando para fazer metodologia de testa está disponível.

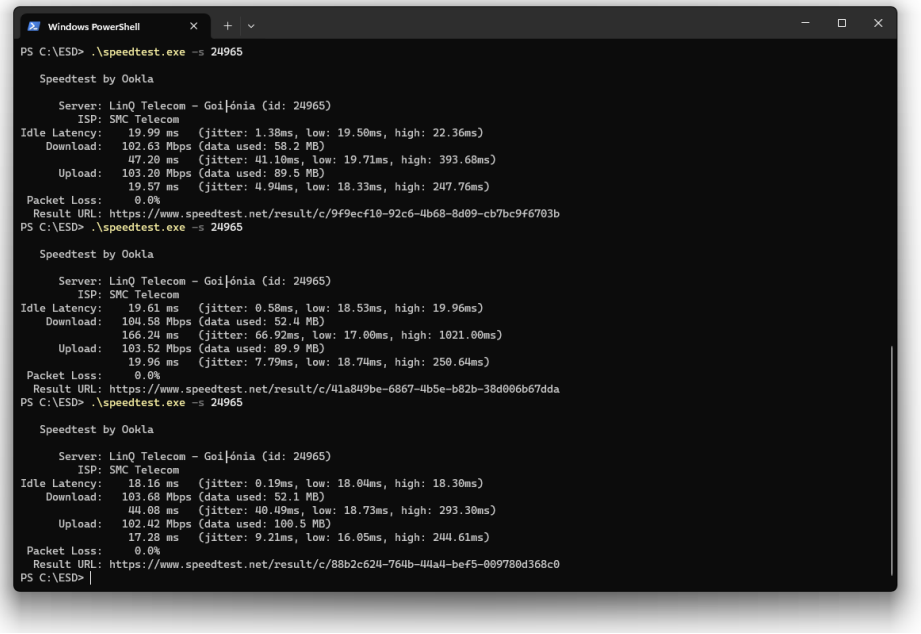
Figura 37 - Speedtest em execução



Fonte: Elaborado pelo autor

Para buscar uma maior homogeneidade nos testes, foi realizada uma série de três testes em cada situação, utilizando sempre o mesmo servidor de rede. Esse procedimento visa padronizar as condições dos testes, assegurando uma avaliação mais consistente. Esses detalhes podem ser observados na figura 39.

Figura 38 - Teste do SpeedTest sem *Firewall*



Fonte: Elaborado pelo autor

Com base nos dados do teste realizado na rede sem a utilização do *firewall*, conforme apresentado na Tabela 1, a latência média em ociosidade e de upload situa-se em torno de 19ms, enquanto a de download é de aproximadamente 50ms.

Tabela 1 - Resultados sem firewall

Latência em ocioso	Latência de download	Latência de upload	Taxa de download	Taxa de upload
19.99ms	47.29ms	19,57ms	102.63 mbps	103.20 mbps
19.61ms	166.24ms	19.96ms	104.58 mbps	103.52 mbps
18.16ms	44.08ms	17.28ms	103.68 mbps	102.42 mbps

Fonte: Elaborado pelo autor

Outro dado importante é o *jitter,* segundo Chadda o *jitter* é a variação no tempo de chegada de pacotes em uma rede, representando a diferença na latência entre pacotes, esse dado é importante pois representa o tempo entre o envio e a resposta do envio do pacote, na tabela 3 pode ser visto com os dados extraídos dos testes de QoS sem o *firewall*.

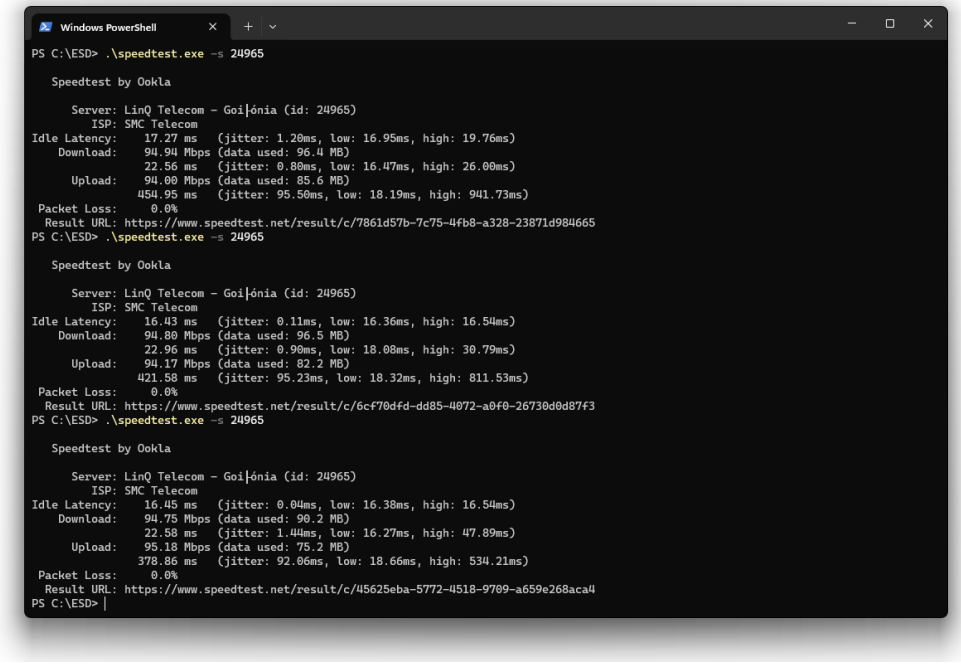
Tabela 2 - Resultados de jitter sem firewall

Jitter em ocioso	Jitter em download	jitter em upload
1,38ms	41,10ms	4.94ms
0,58ms	66.69ms	7.79ms
18,16ms	40.49ms	9.21ms

Fonte: Elaborado pelo autor

Ao realizar os testes com a inclusão do *firewall* na rede, os resultados obtidos são detalhados e ilustrados na figura 40.

Figura 39 - Teste do SpeedTest com *Firewall*



Fonte: Elaborado pelo autor

Com base nos dados da Tabela 3, percebe se mudanças significativas nos parâmetros de latência e taxas de transferência. Houve uma notável diminuição na latência ociosa e uma redução expressiva na latência de download, que caiu de 50ms para 22ms. No entanto, em contrapartida, houve um aumento na latência de upload. Além disso, as taxas de download e upload também demonstraram uma redução. Esses resultados indicam um impacto direto da implementação do *firewall*, afetando de forma distinta as operações de upload e download, assim como as velocidades de transferência de dados.

Tabela 3 - Resultados com firewall

Latência em ocioso	Latência de Download	Latência de upload	Taxa de download	Taxa de upload
17.27ms	22.56ms	454.95ns	94.94 mbps	94.94 mbps
16.43ms	22.96ms	421.58ms	94.90mbps	94.17 mbps
16.45ms	22.58ms	378.86ms	94.75mbps	95.18 mbps

Fonte: Elaborado pelo autor

Com o firewall ativo, observam-se os dados de jitter. Neste teste, é possível avaliar o nível de impacto do firewall na rede, visto que o jitter representa o tempo de resposta entre o envio e a resposta do servidor ao qual o pacote chegou. Os valores correspondentes estão apresentados na Tabela 4

Tabela 4 - Resultados de jitter sem firewall

Jitter em ocioso	Jitter em download	jitter em upload
1,28 ms	0,80ms	95.50ms
0,11 ms	0.90ms	95.23ms
0,04 ms	1.44ms	92.06ms

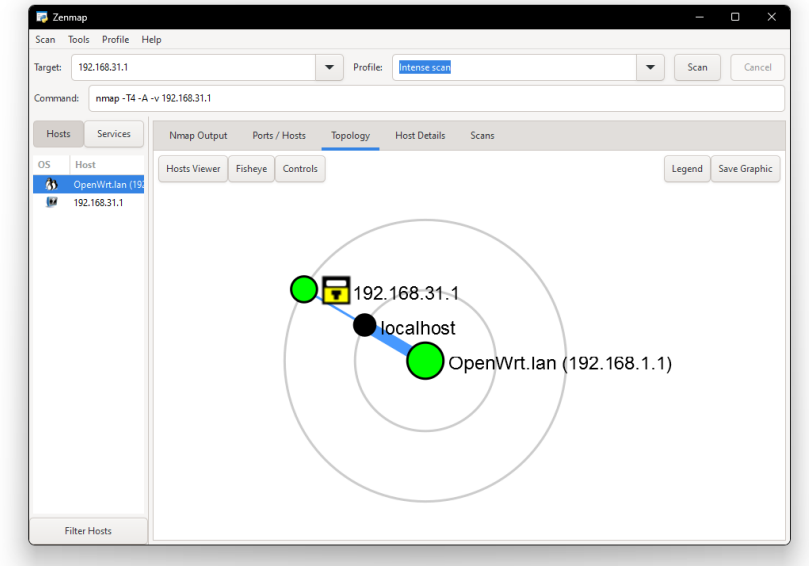
Fonte: Elaborado pelo autor

O comparativo de desempenho da rede, com e sem a presença de um firewall, revela resultados significativos. Partindo dessas premissas e analisando os dados das tabelas 2 e 4, observa-se que a implementação do firewall resulta em uma redução notável na latência de download, embora ocorra um aumento correspondente na latência de upload. A latência ociosa permanece estável, indicando consistência em períodos de baixa atividade. Apesar das alterações nas taxas de transferência, os valores continuam sendo gerenciáveis.

4.1.3 Teste de regras de *firewall*

Este teste visa avaliar a eficácia das regras de *firewall*, analisando sua aplicação e capacidade de manter um ambiente seguro contra ameaças externas. Para conduzir essa avaliação, utiliza se a ferramenta de análise de rede NMap, evidenciada na figura 41. Conforme Daibert (2020), o NMap, abreviação de *Network Mapper* (ou Mapeador de Redes), é uma aplicação especializada na transmissão de pacotes direcionados a um alvo específico, buscando informações essenciais para orientar a subsequente coleta de dados. Esta ferramenta opera na criação e manipulação de pacotes em estado bruto, permitindo a obtenção de variadas informações ao serem transmitidos, tais como a disponibilidade do alvo, o estado das portas, a identificação de serviços ativos, a determinação de versões de sistemas e a identificação de possíveis vulnerabilidades.

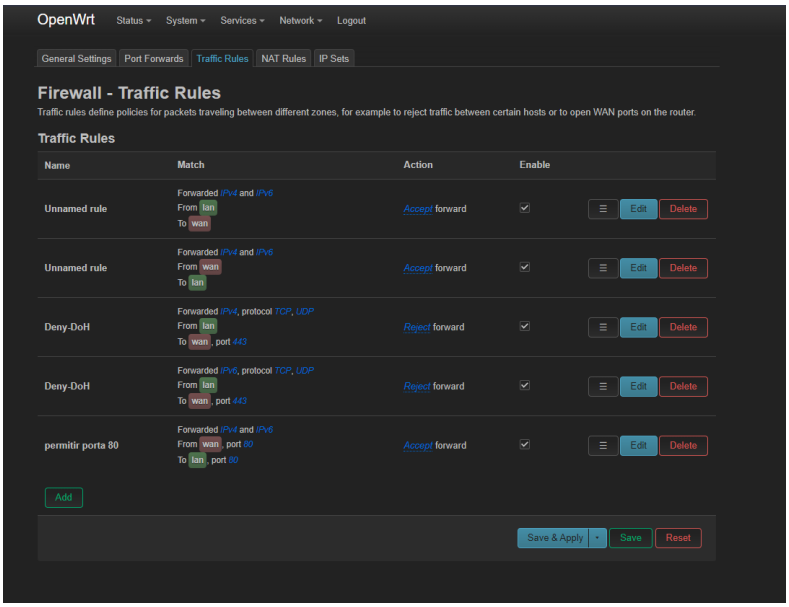
Figura 40 - Interface do NMap



Fonte: Elaborado pelo autor

Para verificar se o *firewall* está ativo, utiliza se a ferramenta NMap para escanear as portas. Para configurar as regras no *firewall*, utiliza se a interface Web do OpenWrt. Na tela do *firewall*, basta clicar em "*Traffic Rules*", como demonstrado na figura 42. Isso permite acessar e definir as regras que regulam o tráfego de dados, controlando o acesso e a segurança da rede.

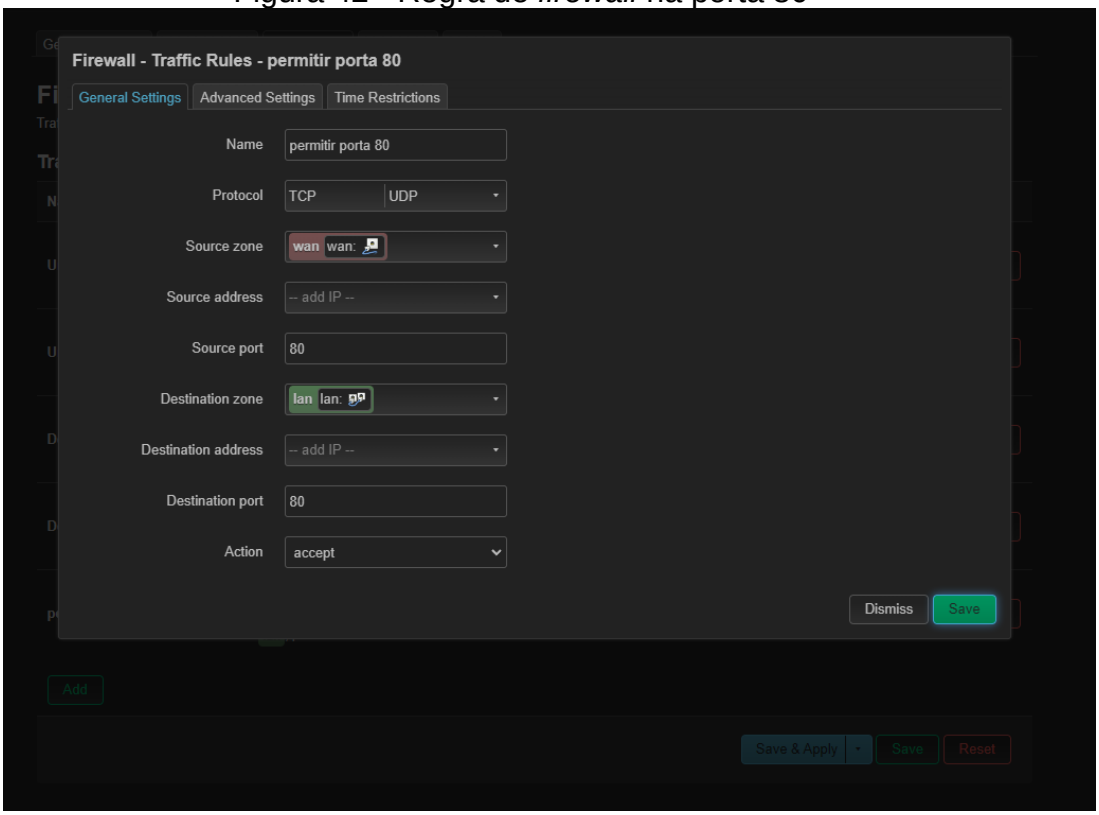
Figura 41 - Interface de regras de *firewall*



Fonte: Elaborado pelo autor.

Para criar uma regra, deve-se clicar em “Add” , os protocolos de rede, a interface que será afetada pela regra e até mesmo especificar o IP para o qual a regra será aplicada. No caso presente, conforme ilustrado na figura 43, o *firewall* está configurado no modo de bloqueio de todo acesso externo. Foi estabelecida uma regra para permitir o tráfego na porta 80, a qual corresponde ao serviço de Web HTTP. Esta ação permite o tráfego específico por meio dessa porta, ao mesmo tempo em que mantém o bloqueio para outros acessos externos.

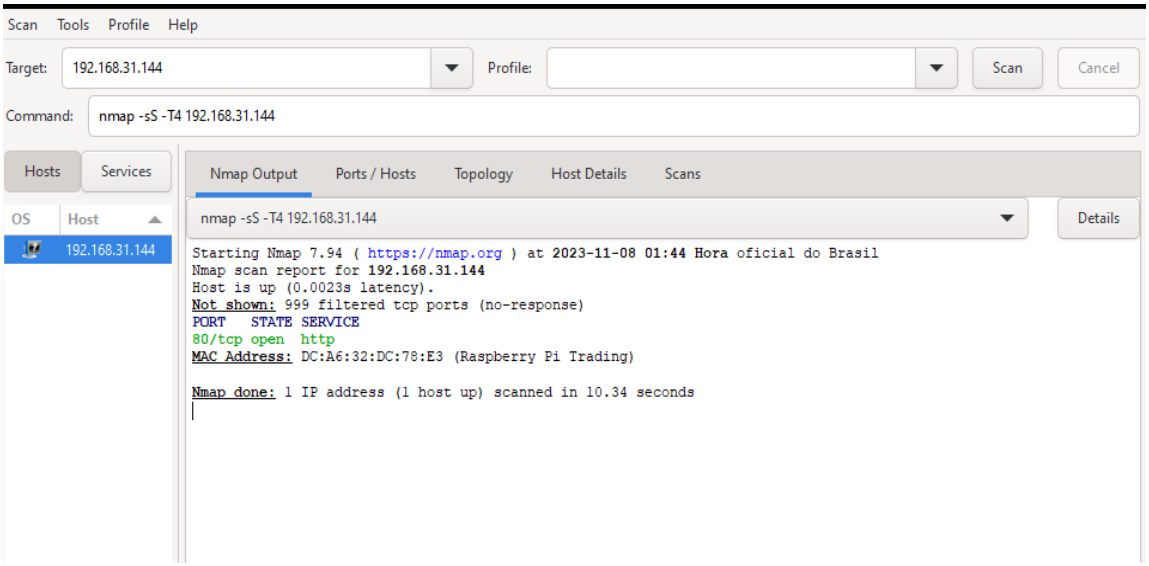
Figura 42 - Regra de *firewall* na porta 80



Fonte: Elaborado pelo autor.

Utilizando o NMap a partir de um computador externo ao *firewall*, basta escanear o IP da *Raspberry Pi*, conforme ilustrado na Figura 44, mostra que existe permissão para o tráfego de dados na porta 80, enquanto os demais serviços encontram-se inacessíveis devido ao bloqueio padrão implementado pelo *firewall*. Isso abre a oportunidade de testar outras portas e serviços disponíveis.

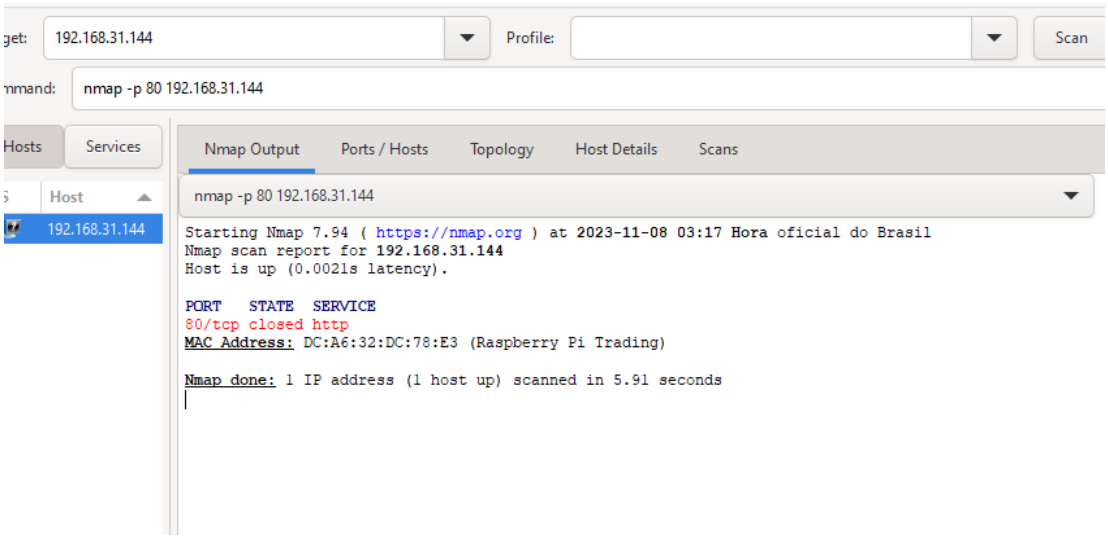
Figura 43 - Porta 80 aberta



Fonte: Elaborado pelo autor.

Ao modificar a regra para bloquear a porta 80, o NMap mostra a porta se encontra bloqueada, como indicado na figura 45. Esse teste específico valida a eficácia da regra implementada, demonstrando que a alteração surtiu o efeito desejado, restringindo o acesso à porta 80. Esse resultado comprova a funcionalidade do *firewall*, evidenciando sua capacidade de controlar e direcionar o tráfego, fornecendo uma camada adicional de segurança ao ambiente de rede.

Figura 44 - Porta 80 fechada



Fonte: Elaborado pelo autor.

4.1.4 Teste de Impacto do *Firewall* na Rede Utilizando o iPerf3

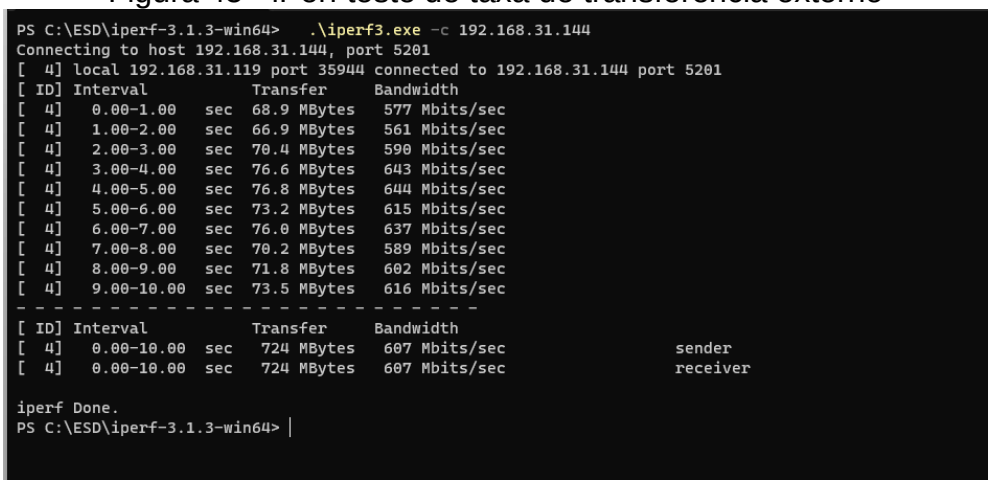
Com o intuito de criar um ambiente de simulação representativo e avaliar de forma abrangente o impacto de *firewall*s em redes, optou-se pela utilização da ferramenta iPerf3. Segundo a documentação oficial, o iPerf3 é reconhecido como uma "ferramenta para medições ativas da largura de banda máxima alcançável em redes IP" (ESnet / Laboratório Nacional Lawrence Berkeley, 2022). Essa aplicação destaca-se pela sua versatilidade, permitindo a customização de diversos parâmetros relacionados ao tempo e à taxa de transferência de pacotes.

Durante a execução de cada teste, o iPerf3 fornece relatórios detalhados que englobam aspectos cruciais, tais como largura de banda, perda de pacotes e outros parâmetros relevantes. Essa abordagem meticulosa se mostra instrumental para a análise do desempenho da rede, oferecendo uma visão holística no contexto da presença de *firewall*s.

O escopo deste experimento visa simular a dinâmica da entrada e saída de dados, representando fielmente cenários de redes reais. O iPerf3, devido à sua capacidade de realizar testes com cargas de pacotes variadas e suportar múltiplas conexões simultâneas, proporciona um ambiente de simulação robusto, assemelhando-se a uma pequena rede.

O primeiro teste tem como objetivo mensurar a taxa de transferência da rede ao acessar o *firewall* externamente. Este procedimento permite a avaliação da taxa máxima alcançável dentro da rede. A execução do teste é viabilizada pelo comando exemplificado na Figura 46, onde uma análise de taxa de transferência é conduzida ao longo de um período de 10 segundos.

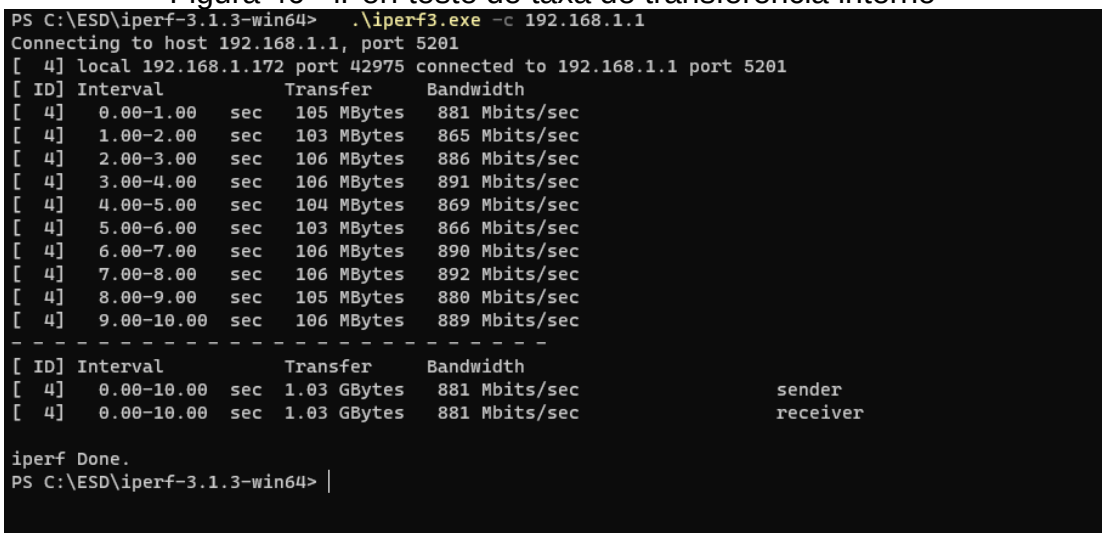
Figura 45 - iPerf teste de taxa de transferência externo



Fonte: Elaborado pelo autor.

Esse conjunto de testes meticulosamente configurados visa não apenas verificar a capacidade do *firewall* em lidar com diferentes cargas de tráfego, mas também compreender seu impacto no desempenho geral da rede. O iPerf3, ao ser empregado nesse contexto, proporciona uma metodologia robusta para investigação e análise, contribuindo para a geração de dados confiáveis e significativos neste estudo. Ao estender a avaliação para a infraestrutura interna do *firewall*, foram realizados testes adicionais utilizando o iPerf3, cujos resultados estão ilustrados na Figura 47. Destaca-se que, de maneira notável, o acesso externo demonstrou uma redução em relação ao acesso interno.

Figura 46 - iPerf teste de taxa de transferência interno



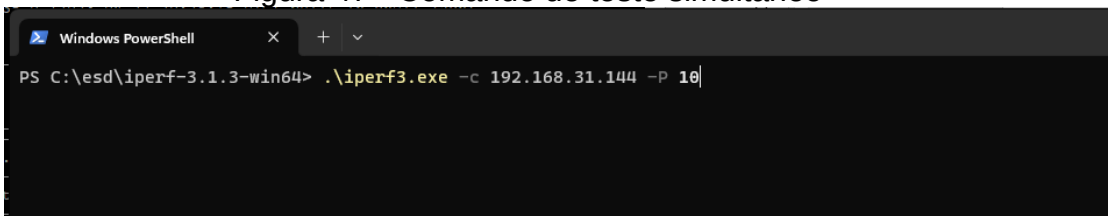
Fonte: Elaborado pelo autor.

Os dados obtidos indicam que a taxa máxima de transferência no acesso externo foi limitada a 607 megabits por segundo, enquanto, internamente, o acesso atingiu uma taxa de 881 megabits por segundo. Esta discrepância sugere que a presença do *firewall* tem um impacto significativo na eficiência do tráfego de dados, resultando em uma diferença apreciável na largura de banda alcançável entre acessos externos e internos.

Esses resultados preliminares corroboram a hipótese de que o *firewall* desempenha um papel crucial na modelagem do fluxo de dados na rede. A limitação observada no acesso externo pode ser atribuída às camadas de segurança adicionadas pelo *firewall*, introduzindo um elemento de controle e filtragem que afeta diretamente o desempenho da comunicação externa.

Prosseguindo com a avaliação do desempenho da rede, o próximo teste foi conduzido para simular um cenário realista de múltiplos acessos, refletindo o ambiente onde há vários fluxos simultâneos de dados. A execução deste teste é realizada mediante o comando apresentado na Figura 48, no qual uma carga de rede simulando 10 conexões simultâneas é aplicada.

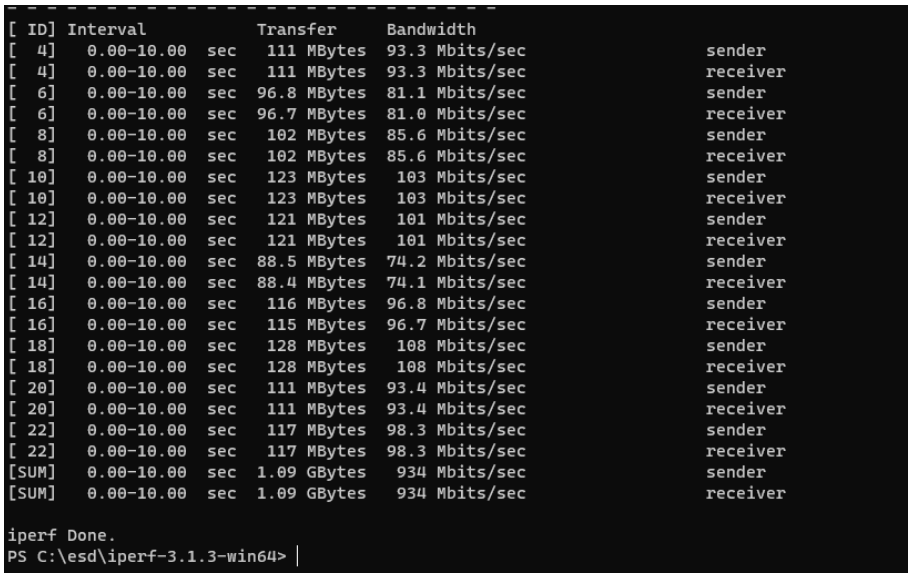
Figura 47 - Comando de teste simultâneo



Fonte: Elaborado pelo autor.

Os resultados obtidos, representados na Figura 49, revelam que o limite de 934 megabits por segundo foi atingido. Esta constatação indica que a taxa máxima da interface de rede de 1 gigabit por segundo pode ser alcançada em um ambiente com múltiplas conexões. Destaca-se que o OpenWrt, operando no *Raspberry Pi*, demonstrou eficiência ao balancear a carga da rede, distribuindo aproximadamente a largura de banda disponível para cada uma das conexões. Este comportamento sugere que o *Raspberry Pi*, mesmo em um ambiente com várias conexões simultâneas, consegue gerenciar o tráfego sem impactar adversamente o desempenho geral da rede.

Figura 48 - Teste simultâneo interno

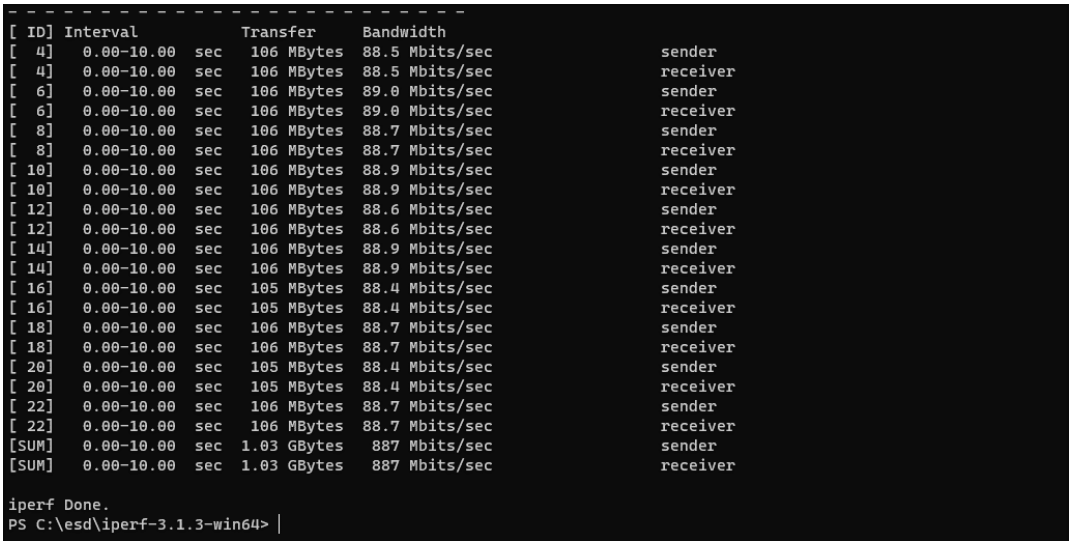


Fonte: Elaborado pelo autor.

Ao replicar o mesmo teste, desta vez internamente na rede protegida pelo *firewall*, os resultados apresentados na Figura 50, revelam uma divisão mais homogênea da banda para as 10 conexões simultâneas, mantendo-se em torno de 89 megabits por segundo. A taxa máxima foi limitada a 887 megabits por

segundo. Esses dados confirmam que a taxa de transferência da interface de rede está sendo efetivamente alcançada mesmo em um cenário com múltiplas conexões simultâneas.

Figura 49 - Teste simultâneo externo



Fonte: Elaborado pelo autor.

A conclusão derivada destes testes de desempenho é que o *firewall*, implementado no *Raspberry Pi* e operando com o OpenWrt, demonstra uma capacidade robusta, para lidar com a carga de trabalho gerada por 10 computadores na rede, no mínimo testado. Este tipo de teste é especialmente relevante, pois evidencia a eficácia do *firewall* em ambientes de pequeno porte, para médio porte, destacando seu desempenho diante de cenários realistas de múltiplos acessos simultâneos. A análise minuciosa desses resultados contribui para uma compreensão aprofundada do papel e da efetividade do *firewall* no contexto específico deste ambiente de teste.

5 CONSIDERAÇÕES FINAIS

Este estudo detalhado abordou a implementação de um *firewall* em uma *Raspberry Pi* 4, empregando o sistema operacional OpenWrt. Em meio à crescente importância da segurança de rede para organizações que dependem da tecnologia da informação, a estratégia essencial de utilizar um *firewall* ganha destaque. O enfoque foi explorar a implementação de um *firewall* em um Single Board Computer (SBC), utilizando o *Raspberry Pi* como base.

Ao analisar as características técnicas e funcionais do *Raspberry Pi*, destaca-se sua viabilidade como plataforma para atuar como *firewall* em ambientes de pequeno porte. O estudo não apenas identificou essas qualidades, mas também avaliou os principais desafios e riscos de segurança enfrentados por redes de menor escala. Além disso, examina-se como um *firewall* baseado em *Raspberry Pi* pode mitigar eficazmente essas preocupações.

Os resultados obtidos oferecem *insights* valiosos, proporcionando uma visão aprofundada sobre o desempenho e o impacto desse *firewall* em ambientes de rede locais. Ao equilibrar os aspectos cruciais de segurança e desempenho, este estudo não só ressalta a eficácia do *firewall* implementado, mas também fornece uma compreensão aprimorada sobre como o *Raspberry Pi* pode desempenhar um papel significativo na segurança de redes de pequeno porte, destacando aspectos cruciais explorados nos testes.

Na avaliação da Qualidade de Serviço (QoS), observam-se as nuances interessantes no modo como o *firewall* influencia diferentes métricas de desempenho. A redução notável na latência de download indica uma contribuição positiva, mas esse ganho é acompanhado por um aumento correspondente na latência de upload. Esse aumento na latência de upload pode ser atribuído ao processamento do filtro de *firewall*, que introduz uma sobrecarga adicional durante a transferência de dados, impactando o tempo de resposta na direção de envio. As taxas de download e upload também apresentaram reduções significativas, enfatizando a necessidade de um ajuste preciso para equilibrar efetivamente segurança e desempenho.

Esses resultados destacam a complexidade inerente à implementação de *firewall*s e ressaltam que, embora haja melhorias perceptíveis em certos aspectos, como a latência de download, é fundamental considerar o impacto total nas diversas métricas de desempenho. A compreensão de que o aumento da latência de upload está

associado ao processamento do filtro de *firewall* adiciona uma camada de informação crucial para a interpretação desses resultados, proporcionando insights mais profundos sobre as compensações entre segurança e eficiência na rede.

Ao investigar as regras de *firewall*, torna-se evidente que a *Raspberry Pi* 4, em conjunto com o OpenWrt, proporciona flexibilidade e eficácia na capacidade de controlar e direcionar o tráfego de acordo com políticas específicas. A implementação de regras específicas, como permitir ou bloquear o acesso a determinadas portas, revelou-se altamente bem-sucedida, destacando a importância crítica de uma configuração precisa para atender às necessidades específicas de segurança da rede.

É essencial ressaltar que o bloqueio ou liberação estratégica de portas desempenha um papel significativo na mitigação de ameaças e na proteção dos recursos da rede. Ao bloquear portas não essenciais ou vulneráveis, o *firewall* cria uma barreira adicional contra possíveis ataques cibernéticos, dificultando a exploração de vulnerabilidades específicas. Essa abordagem proativa ajuda a fortalecer a segurança da rede, limitando os potenciais vias de entrada para atividades maliciosas.

Além disso, a capacidade de definir regras personalizadas no *firewall* proporciona um controle preciso sobre o tráfego, permitindo que os administradores de rede adaptem as políticas de segurança conforme necessário. Isso não apenas aumenta a adaptabilidade do sistema de segurança, mas também garante que as medidas de proteção estejam alinhadas com os requisitos específicos da rede, proporcionando uma defesa mais eficiente contra ameaças em constante evolução.

Os testes de impacto do *firewall*, conduzidos por meio do iPerf3, revelaram aspectos cruciais relacionados à taxa de transferência e à capacidade de gerenciamento de múltiplos acessos simultâneos. Ao acessar a rede externamente, foi observada uma redução significativa na taxa de transferência, evidenciando a influência direta do *firewall* na eficiência do tráfego externo. Essa diminuição na taxa de transferência externa pode ser atribuída às camadas adicionais de segurança implementadas pelo *firewall*, introduzindo uma filtragem mais rigorosa que impacta diretamente o desempenho.

Entretanto, em cenários de múltiplos acessos, o *Raspberry Pi* demonstrou uma notável capacidade de gerenciar a carga da rede de forma eficiente, distribuindo de maneira equitativa a largura de banda disponível entre as conexões simultâneas. Esse equilíbrio eficiente destaca o potencial da *Raspberry Pi* como uma solução robusta

para *firewall*s em ambientes de pequeno porte, onde a capacidade de lidar com múltiplos acessos simultâneos é crucial para manter o desempenho e a estabilidade da rede. Essa capacidade de gerenciamento eficaz contribui para a eficiência geral do sistema de segurança, tornando a *Raspberry Pi* uma escolha viável para ambientes que demandam equilíbrio entre segurança e desempenho.

Comparando esses resultados com soluções de mercado, percebe se que, embora a *Raspberry Pi* 4 apresente uma redução na taxa de transferência em acessos externos, sua capacidade de gerenciamento de múltiplos acessos simultâneos destaca-se positivamente. Soluções comerciais podem oferecer desempenho superior, mas muitas vezes a um custo significativamente maior. Portanto, a escolha entre uma *Raspberry Pi* e uma solução de mercado dependerá das necessidades específicas da rede, do orçamento disponível e da importância atribuída ao equilíbrio entre segurança e desempenho.

Em síntese, este estudo representa uma contribuição valiosa para a compreensão prática da implementação de *firewall*s em dispositivos de baixo custo, destacando a eficácia da *Raspberry Pi* 4 nesse contexto. À medida que a segurança da rede se torna cada vez mais crucial, a *Raspberry Pi* 4 se posiciona como uma escolha viável para ambientes específicos que não demandam recursos massivos. A capacidade do dispositivo em desempenhar efetivamente as funções de *firewall*, aliada à flexibilidade proporcionada pelo OpenWrt, ressalta sua posição como uma alternativa acessível e adaptável para ambientes de pequeno porte.

No entanto, é imperativo que os administradores de rede estejam cientes das compensações inerentes entre segurança e desempenho ao optarem por essa solução. A configuração precisa das regras de *firewall* emerge como uma prática essencial, assim como a consideração minuciosa das necessidades específicas da rede. Esses elementos são fundamentais para otimizar o equilíbrio entre segurança robusta e desempenho eficiente. Portanto, a implementação bem-sucedida da *Raspberry Pi* 4 como *firewall* depende da habilidade de gerenciar estrategicamente essa interação delicada, garantindo que as medidas de segurança adotadas não comprometam excessivamente o desempenho da rede. Em última análise, ao adotar essa abordagem consciente e estratégica, a *Raspberry Pi* 4 se posiciona como uma solução sólida para requisitos de segurança em ambientes de pequeno porte, oferecendo uma resposta acessível e eficaz às demandas contemporâneas de proteção cibernética.

No anexo 1 apresenta-se o termo de autorização para publicação.

5.1 Trabalhos Futuros

Propõe-se em trabalhos futuros:

- aplicação do *firewall* em um ambiente mais realístico
- Implementação de uns múltiplos dispositivos *firewall*
- Implementação de Políticas de Segurança Específicas no *firewall*
- Estudo de Vulnerabilidades e Ataques no *firewall*

REFÊRENCIAS

ABNT. Associação Brasileira de Normas Técnicas. **ABNT NBR ISO/IEC 27001:2013 - Tecnologia da informação - Técnicas de segurança - Sistemas de gestão de segurança da informação** - Requisitos. Rio de Janeiro, 2013.

ABNT. Associação Brasileira de Normas Técnicas. **ABNT NBR ISO/IEC 27032:2012 - Tecnologia da informação - Técnicas de segurança - Diretrizes para cibersegurança**. Rio de Janeiro, 2012.

BERNERS-LEE, T.; Fischetti, M. **Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web**. HarperBusiness, 2000.

BRASIL. Lei nº 12.965, de 23 de abril de 2014. **Estabelece princípios, garantias, direitos e deveres para o uso da Internet no Bras**il. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2014/lei/l12965.htm>. Acesso em 07 de mar. de 2023.

BRASIL. Lei nº 13.709, de 14 de agosto de 2018. **Dispõe sobre a proteção de dados pessoais e altera a Lei nº 12.965, de 23 de abril de 2014 (Marco Civil da Internet).** Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/L13709.html>. Acesso em 07 de março de 2023.

BROWN, R. **About the OpenWrt/LEDE project**. Disponível em: <https://openwrt.org/about>. Acesso em: 10 ago. 2023.

CHADDA, Ankur. **Metodologia de Teste de Qualidade de Serviço. Dissertação** (Mestrado em Ciência da Computação) - Universidade de New Hampshire, 2004. Disponível em: <https://www.iol.unh.edu/sites/default/files/knowledgebase/routing/QoS_Testing_Methodology.pdf>. Acesso em: 08 de set. de 2023.

DAIBERT, Marcelo. **NMAP: um estudo sobre a ferramenta de busca e análise de vulnerabilidades em redes**. 2020. Disponível em: <https://revista.unifagoc.edu.br/index.php/caderno/article/view/890>. Acesso em: 08 de set de 2023.

ESnet / Laboratório Nacional Lawrence Berkeley. **iPerf3 - Ferramenta para medições ativas da largura de banda máxima em redes IP**. 2022. Disponível em: <https://iperf.fr/>. Acesso em: 08 de out de 2023.

EU. (2016). **Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)**. Disponível em: <https://eur-lex.europa.eu/eli/reg/2016/679/oj>. Acesso em 07 mar 2023.

FOROUZAN, Behrouz A. **Comunicação de dados e redes de computadores**. 3. ed, Bookman, 2006.

GHEDIN, Alessandro, e Rustam Lalkaka. **HTTP/3: the past, the present, and the future**. Disponível em: <http://blog.cloudflare.com/http3-the-past-present-and-future>. Acesso em: 24 de abr. de 2023.

GOUVEIA, R. A.; SILVA, J. S. DA. LGPD - Lei Geral de Proteção de Dados. Biblioteca Digital de Monografias da Famig, 26 de jun. de 2020.

GRANDERATH, M., & SCHONWALDER, J. **A Resource Efficient Implementation of the RESTCONF Protocol for OpenWrt Systems**. In: NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium. IEEE Xplore. Disponível em: https://ieeexplore.ieee.org/abstract/document/9110458. Acesso em: 1 de ago. de 2023.

HERTZOG, Raphaël, et al. **Kali Linux Revelado: Dominando a Distribuição de Teste de Penetração**. Offsec Press, 2017.

HINTZBERGEN, Jule, et al. **Fundamentos de Segurança da Informação: Com base na ISO 27001 e na ISO 27002**. Brasport, 2018.

KANAGACHIDAMBARESAN, G. R. Papel dos Computadores de Placa Única (SBCs) na Prototipagem Rápida de IoT. Springer, 2021.

KUROSE, James F., e KEITH W. Ross. **Redes de computadores e a Internet: uma abordagem top-down**. 6. ed, Pearson, 2013.

Ltd, Raspberry Pi. **Raspberry Pi 4 Model B Specifications**. Raspberry Pi, Disponível em: <https://www.raspberrypi.com/products/raspberry-pi-4-modelAcessado 7 de mai de 2023.

MENDES, Douglas Rocha. **Redes de Computadores: Teoria e Prática**. 2. ed, Novatec, 2015.

MOLLOY, Derek. **Exploring Raspberry Pi: interfacing to the real world with embedded Linux**. Wiley, 2016.

MONK, Simon. **Programming the Raspberry Pi: getting started with Python.** Third edition, McGraw Hill Education, 2021.

NEWCOMB, Aaron. **Linux for Makers: Understanding the Operating System That Runs Raspberry Pi and Other Maker SBCs**. Maker Media, Inc, 2017.

RASH, Michael. **Linux firewalls: attack detection and response with iptables, psad, and fwsnort**. No Starch Press, 2007.

REALTEK. **RTL8153B-VB-CG.** Disponível em: <https://www.realtek.com/en/products/connected-media-ics/item/rtl8153b-vb-cg>. Acessado 10 de ago. de 2023.

STALLINGS, William. **Redes e sistemas de comunicação de dados**. Elsevier, 2005.

TANENBAUM, Andrew S., e David J. Wetherall. **Redes de computadores**. 5. ed, Pearson Prentice Hall, 2011.

TORRES, G. **Redes de computadores**. 2. ed. Novaterra, 2016.

Web's Advantages - World Wide Web Foundation. Disponível em: <https://Webfoundation.org/about/vision/why-the-Web>. Acesso em: 24 nov. 2023.Acessado 9 de abr. de 2023.

ANEXO 1 - Termo de autorização de produção acadêmica

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
GABINETE DO REITOR
Av. Universitária, 1069 • Setor Universitário
Caixa Postal 86 • CEP 74605-010
Goiânia • Goiás • Brasil
Fone: (62) 3946.1000
www.pucgoias.edu.br • reitoria@pucgoias.edu.br

RESOLUÇÃO nº 038/2020 – CEPE

ANEXO I

APÊNDICE ao TCC

Termo de autorização de publicação de produção acadêmica

O(A) estudante Hebert Cardoso de Souza
do Curso de Engenharia de Computação, matrícula 2017200330019 2,
telefone: 62936939838 e-mail Hebertcardoso@gmail.com
na qualidade de titular dos direitos autorais, em consonância com a Lei nº 9.610/98 (Lei dos Direitos do Autor), autoriza a Pontifícia Universidade Católica de Goiás (PUC Goiás) a disponibilizar o Trabalho de Conclusão de Curso intitulado Implementação de Firewall em Raspberry Pi, gratuitamente, sem ressarcimento dos direitos autorais, por 5 (cinco) anos, conforme permissões do documento, em meio eletrônico, na rede mundial de computadores, no formato especificado (Texto(PDF); Imagem (GIF ou JPEG); Som (WAVE, MPEG, AIFF, SND); Vídeo (MPEG, MWV, AVI, QT); outros, específicos da área; para fins de leitura e/ou impressão pela internet, a título de divulgação da produção científica gerada nos cursos de graduação da PUC Goiás.

Goiânia, 30 de AGOSTO de 2023.

Assinatura do autor: Hebert Cardoso de Souza
Nome completo do autor: Hebert Cardoso de Souza

Assinatura do professor–orientador:
Nome completo do professor-orientador: Marcio Antonio Cabral de Araújo