Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Engenharia de Software

Conjunto de ensaios didáticos para a experimentação de fundamentos de redes de computadores

Autor: Marco Antônio de Lima Costa

Orientador: Prof. Dr. Tiago Alves da Fonseca

Brasília, DF

2022

Marco Antônio de Lima Costa

Conjunto de ensaios didáticos para a experimentação de fundamentos de redes de computadores

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília – UnB

Faculdade UnB Gama – FGA

Orientador: Prof. Dr. Tiago Alves da Fonseca

Brasília, DF

2022

Marco Antônio de Lima Costa
Conjunto de ensaios didáticos para a experimentação de fundamentos de redes de computadores/ Marco Antônio de Lima Costa. – Brasília, DF, 2022-
66 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Tiago Alves da Fonseca

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB
Faculdade UnB Gama – FGA , 2022.

1. FreeBSD. 2. Redes. I. Prof. Dr. Tiago Alves da Fonseca. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Conjunto de ensaios didáticos para a experimentação de fundamentos de redes de computadores

CDU 02:141:005.6

Marco Antônio de Lima Costa

Conjunto de ensaios didáticos para a experimentação de fundamentos de redes de computadores

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Trabalho aprovado. Brasília, DF, :

Prof. Dr. Tiago Alves da Fonseca
Orientador

Prof. Dr. Fernando William Cruz
Convidado 1

Prof. Dr. John Lenon Cardoso Gardenghi
Convidado 2

Brasília, DF
2022

Este trabalho é dedicado ao meu avô Antônio Barbosa de Lima (in memorian), que não pode me ver formado, mas sempre me apoiou e se preocupou com os meus estudos. Ele sempre será um grande exemplo de dignidade e caráter para mim.

Agradecimentos

Agradeço a Universidade de Brasília por ter me recebido de braços abertos e com todas as condições que me proporcionaram dias de aprendizagem muito ricos.

Agradeço aos meus pais e irmãos, que me incentivaram nos momentos difíceis e demonstraram apoio ao longo de todo o período em que me dediquei a este trabalho.

Agradeço a todos os meus professores do curso de Engenharia de Software da Universidade de Brasília pela excelência da qualidade técnica de cada um, em especial ao meu orientador, Prof. Dr. Tiago Alves da Fonseca, por aceitar conduzir o meu trabalho de conclusão de curso.

Agradeço aos amigos, que sempre estiveram ao meu lado, pela amizade incondicional, pelo apoio e compreensão da minha ausência enquanto eu me dedicava à realização deste trabalho.

“Tudo o que temos de decidir é o que fazer com o tempo que nos é dado“
(Gandalf)

Resumo

No mundo atual, o uso das redes de computadores se tornou algo imprescindível no sentido de obter e compartilhar informações. Compreende-se que redes de computadores são um conjunto de computadores autônomos interconectados, de modo que os computadores possam funcionar tanto em conjunto quanto independentes uns dos outros. No contexto de engenharia de software, as redes de computadores passam a ter um papel fundamental no desenvolvimento e provimento de novas aplicações. Portanto, é essencial que alunos que cursam bacharelado em Engenharia de Software obtenham conhecimento para operação básica e resolução de problemas relacionados às camadas de rede. Porém, um dos grandes problemas enfrentados hoje no ensino de rede de computadores é a pouca disponibilidade de equipamentos adequados para a utilização de ferramentas e o conteúdo teórico massivo que é lecionado enquanto o discente participa apenas como ouvinte. Desse modo, o presente trabalho propõe a implementação de um conjunto de experimentos práticos para exercitar a operação básica de uma rede de computadores utilizando os sistemas operacionais Linux e FreeBSD, demonstrando a viabilidade da aprendizagem prática no ensino de redes de computadores.

Palavras-chave: redes de computadores. freebsd. experimentos práticos.

Abstract

In today's world, the use of computer networks has become essential in order to obtain and share information. It is understood that computer networks are a set of autonomous computers interconnected, so that computers can work both together and independently of each other. In the context of software engineering, computer networks play a fundamental role in the development and provision of new applications. Therefore, it is essential that students pursuing a bachelor's degree in software engineering gain knowledge for basic operation and troubleshooting related to the network layers. However, one of the major problems faced today in computer network teaching is the limited availability of adequate equipment for the use of tools and the massive theoretical content that is taught while the student participates only as a listener. Thus, the present work proposes the implementation of a set of practical experiments to exercise the basic operation of a computer network using the Linux and FreeBSD operating systems, demonstrating the feasibility of practical learning in teaching computer networks.

Key-words: computer network. freebsd. practical experiments.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

ARP	Address Resolution Protocol
ARPANET	Advanced Research Projects Agency Network
ASCII	American Standard Code for Information Interchange
BIOS	Basic Input/Output System
BSD	Berkeley Software Distribution
DHCP	Dynamic Host Configuration Protocol
DNAT	Destination Network Address Translation
DNS	Domain Name System
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IID	Identificador de Interface
IoT	Internet of Things
IP	Internet Protocol
ISO	International Organization for Standardization
MAC	Media Access Control
NAT	Network Address Translation
NIC	Network Interface Controller
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
SD	Sistema Distribuído
SMTP	Simple Mail Transfer Protocol
SNAT	Source Network Address Translation
TCC	Trabalho de Conclusão de Curso
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

Sumário

1 Introdução

Atualmente estudos e avanços em Tecnologias de Informação e Comunicação evidenciam um novo planejamento para o desenvolvimento de práticas, contribuindo de maneira significativa para a reestruturação das estratégias de ensinar e aprender; proporcionando diferentes ferramentas e artefatos para apoiar o processo de aprendizagem no âmbito educacional. Por se tratar da área de tecnologia de informação, a disciplina de Redes de Computadores está inserida nessa realidade e, para facilitar a total compreensão deste tema, é necessário o desenvolvimento de atividades práticas com objetivos indispensáveis como: ensinar aos estudantes sobre a pilha de protocolos, visualizar as características de hosts, enlaces e portas, analisar seu comportamento em diferentes topologias e cenários (FERREIRA et al., 2013).

Uma rede de computadores é um sistema composto por computadores e outros dispositivos, em que cada um desses elementos pode se comunicar uns com os outros (KIZZA, 2005). As redes existem de forma que dados possam ser enviados entre lugares distintos e são divididas em diferentes tipos que possuem diferentes características e diferentes funcionalidades (FOROUZAN, 2009).

Para que a comunicação entre diferentes dispositivos se efetive, deve haver um conjunto de regras que cada elemento da rede tem que seguir para realizar a comunicação (KIZZA, 2005). Os protocolos são a implementação dessas regras e um protocolo padrão é aquele que é amplamente adotado por fornecedores e fabricantes de equipamentos (FOROUZAN, 2009).

Para facilitar o entendimento de uma rede de computadores, agrupar protocolos que possuem objetivos semelhantes e separar aqueles que tem objetivos distintos, criaram-se alguns modelos de rede. Os modelos de redes servem para organizar, unificar e controlar os componentes de *hardware* e *software* das comunicações de dados e das redes (FOROUZAN, 2009).

A execução de atividades práticas em disciplinas de Redes de Computadores é essencial para o desenvolvimento de habilidades técnicas fundamentais dos estudantes (FERREIRA et al., 2013). Hassan (2003) disserta, em relação ao ensino na área de redes de computadores, que dentre os problemas e dificuldades encontrados para o desenvolvimento da disciplina estão: material didático e pouca disponibilidade de equipamentos adequados para a utilização de ferramentas. Ele explica que isto ocorre em grande parte devido à velocidade em que os avanços tecnológicos acontecem e ao custo de manutenção de um laboratório experimental. Considerando essas observações, torna-se importante o desenvolvimento de experimentos, em redes de computadores, utilizando um sistema

operacional livre e de código aberto.

1.1 Justificativa

Alunos que cursam bacharelado em Engenharia de Software devem aprender sobre diferentes áreas para que possam planejar o desenvolvimento de um software. Nos dias atuais, raros são os softwares que não estão conectados a uma rede de computadores, seja a internet ou uma rede privada. Portanto, entender o funcionamento de uma rede de computadores se torna essencial para que esse futuro profissional aprenda a configurar uma rede, resolva eventuais problemas e não fique dependente de um profissional da área de redes. Um administrador de sistemas não precisa conhecer a fundo os conceitos de redes, mas saber o suficiente para diagnosticar seus próprios problemas transforma um bom administrador de sistema em um ótimo (LUCAS, 2019).

O FreeBSD é um sistema operacional robusto e de código-fonte aberto que é amplamente utilizado em aplicações práticas, especialmente em redes de computadores. É um sistema operacional completo, o que significa que inclui todas as ferramentas e aplicativos necessários para realizar diversas tarefas, como servidores de rede, firewall, banco de dados, etc. Isso reduz a chance de erro inesperado causado por aplicativos terceiros, pois todas as ferramentas e aplicativos que o sistema precisa já estão incluídos (DELGADO, 2022). O FreeBSD foi um dos primeiros sistemas operacionais a incluir suporte ao protocolo TCP/IP, que é a base da comunicação na internet. Isso demonstra a experiência e o conhecimento da equipe de desenvolvimento do FreeBSD em tecnologias de rede (INC, 2021).

Dentre os sistemas operacionais comumente utilizados hoje em dia (Linux e Windows), o FreeBSD apresenta o menor número de vulnerabilidades reportadas, que leva a ser um fator importante a considerar na escolha de um sistema operacional para experimentos práticos, pois um sistema com menos vulnerabilidades é menos propenso a ser comprometido por ataques externos, ainda mais se tratando de experimentos de redes (CVE, 2022). Outro fator interessante é que o FreeBSD possui compatibilidade binária com o Linux, tornando possível executar aplicativos compilados para o Linux no FreeBSD sem precisar recompilá-los. Isso pode ser útil na utilização de aplicativos já existentes e testados no Linux sem precisar se preocupar com problemas de compatibilidade (DELGADO, 2022).

A Netflix, uma grande empresa de streaming de vídeo, é colaboradora do FreeBSD e usa o sistema operacional em sua infraestrutura de serviços de streaming. Além da Netflix, outras grandes empresas também utilizam o FreeBSD em suas aplicações, como a Apple, Sony, Apache, Cisco, McAfee o que se constitui em indicativo da confiabilidade e da robustez do FreeBSD (DELGADO, 2022).

A monocultura de sistemas operacionais pode ser uma limitação significativa para os alunos que desejam desenvolver habilidades em plataformas diferentes. Por exemplo, se a faculdade optar por adotar apenas o Linux como sistema operacional, os alunos não terão a oportunidade de praticar e aprender a trabalhar com outras plataformas, como Windows, Mac OS ou Unix. Isso pode limitar a capacidade dos alunos de se adaptar a novas plataformas de desenvolvimento de software, o que pode afetar a qualidade do trabalho dos alunos, bem como sua capacidade de se destacar em um mercado competitivo.

Portanto, é extremamente importante que um profissional da área de tecnologia expanda sua gama de conhecimento para lidar com diferentes tecnologias durante sua carreira. O FreeBSD e o Linux são sistemas operacionais que funcionam muito bem para a configuração e administração de uma rede de computadores e, embora ambos sejam baseados no UNIX, cada um apresenta características que permitem clara distinção. Por esse motivo, eles possuem algumas semelhanças que fazem a curva de aprendizado ficar menor quando se trata de migração de um sistema para o outro, mas, ao mesmo tempo, introduz novos conceitos e uma nova bagagem de aprendizado.

O modelo tradicional de ensino propõe que o professor seja o detentor do conhecimento passando o conteúdo enquanto os alunos absorvem tudo de maneira passiva. No ensino por meio da prática, que faz parte da metodologia ativa de aprendizagem, os alunos são estimulados a tomarem a frente, com maior interação e independência, participando ativamente do processo, fazendo com que o conhecimento seja realmente absorvido (PAIVA et al., 2016).

Sendo assim, o aprendizado de conceitos e fundamentos de uma rede de computadores se torna mais efetivo quando o aluno se torna parte desse processo por meio da prática. Portanto, este trabalho tem como proposta apresentar um conjunto de experimentos práticos para exercitar a operação básica de uma rede de computadores.

1.2 Objetivos

1.2.1 Objetivo Geral

O objetivo principal da pesquisa é desenvolver um conjunto de experimentos práticos para operação básica de uma rede de computadores, utilizando o FreeBSD e o Linux, para ser aplicado na disciplina “fundamentos de redes de computadores”, da Faculdade do Gama na Universidade de Brasília.

1.2.2 Objetivos Específicos

1. Desenvolver um conjunto de experimentos práticos para configuração dos principais protocolos de rede;

2. Demonstrar a viabilidade do uso das atividades práticas na disciplina “fundamentos de redes de computadores”;
3. Apresentar um sistema operacional diferente para agregar conhecimento aos alunos;
4. Apresentar, aos discentes de engenharia de software, um conjunto de ferramentas para serem usadas na resolução de problemas referentes a rede de computadores.

1.3 Estrutura do Documento

Este documento está composto pelos seguintes capítulos:

- Capítulo 1 - Introdução: A introdução contém um breve contexto e motivações para a realização deste trabalho;
- Capítulo 2 - Fundamentação Teórica: A fundamentação teórica contém conceitos teóricos considerados necessários para uma melhor compreensão do trabalho que será desenvolvido;
- Capítulo 3 - Proposta de Trabalho: Contém uma explanação da proposta de trabalho à ser desenvolvido;
- Capítulo 4 - Metodologia: Contém descrição da forma que o projeto será desenvolvido, assim como metodologias, políticas e ferramentas que serão utilizadas.
- Capítulo 5 - Resultados: Contém uma explanação dos resultados obtidos durante o desenvolvimento deste trabalho.
- Capítulo 6 - Conclusão: Contém as considerações finais a cerca do trabalho desenvolvido.

2 Fundamentação Teorica

2.1 Sistemas Operacionais

Um sistema operacional pode ser considerado um grande software que faz a interface entre o usuário e os componentes de hardware do computador permitindo, por exemplo, a disponibilização de uma interface gráfica. Seu objetivo é gerenciar o compartilhamento de recursos do sistema (HANSEN, 1973).

Sistemas operacionais atuais permitem o armazenamento prolongado de dados, a existência e a utilização de vários usuários simultaneamente. Dessa forma, eles são responsáveis por gerenciar o uso dos recursos disponíveis no computador, resolver eventuais conflitos de requisições simultâneas, controlar o acesso aos dados armazenados, dentre outras tarefas (HANSEN, 1973).

2.1.1 FreeBSD

O FreeBSD é um sistema operacional de código aberto derivado do BSD (Berkeley Software Distribution), versão do UNIX desenvolvido pela Universidade da Califórnia em Berkeley. Ele é considerado um sistema operacional completo, o que quer dizer que o sistema entrega *kernel*, *drivers*, conjunto de aplicações de espaço de usuário e documentação. É utilizado por grandes empresas ao redor do mundo pois seu código fonte é entregue sob uma licença BSD permissiva, a qual permite que qualquer pessoa possa fazer modificações e não tenha que disponibilizar publicamente essas alterações, o que pode ser um bom atrativo para empresas privadas (DELGADO, 2022).

É válido ressaltar que toda a pilha de protocolos TCP/IP da ARPANET, que foi essencial para a criação e existência da internet, foi inicialmente implementada utilizando o BSD (MANDEL; SIMON; DELYRA, 1997).

2.1.2 Debian

O Debian é uma distribuição do Linux, sistema operacional de código aberto baseado em Unix. Ele é desenvolvido por uma grande comunidade de voluntários e profissionais de todo o mundo e é considerado um sistema operacional bastante estável e confiável. O Debian é uma escolha popular para servidores e desktops devido à sua ampla gama de aplicações disponíveis e sua licença livre que permite ao usuário customizar e distribuir o sistema operacional de acordo com suas necessidades. Porém, diferente do FreeBSD, a licença não é completamente permissiva, as alterações precisam ser disponibilizadas para

que outros usuários também possam usufruir delas (DEBIAN.ORG, 2022).

2.2 Sistemas Distribuídos

Tanenbaum e Steen (2007) define um sistema distribuído como:

> "Um conjunto de computadores independentes que se apresenta a seus usuários como um sistema único e coerente."

Dessa maneira tem-se que um sistema distribuído é composto por diferentes computadores autônomos que se comunicam entre si através de uma rede, dando ao usuário a sensação de que está em um sistema único. Tendo em vista que cada nó desse sistema é um computador independente, esse tipo de sistema permite que ele seja extremamente escalável: basta adicionar ou retirar uma instância para que esse sistema cresça ou diminua sua capacidade. A comunicação entre os nós desse sistema deve ocorrer de forma oculta ao usuário, fazendo com que ele não perceba que está lidando com diferentes computadores. Os sistemas distribuídos também ajudam na disponibilização contínua dos serviços, pois mesmo que algumas partes estejam temporariamente indisponíveis, o sistema deve continuar funcionando sem que o usuário perceba quais são essas partes comprometidas (TANENBAUM; STEEN, 2007).

Essa sensação de estar em um único sistema é devido a uma camada de software intermediária de que os sistemas distribuídos dependem, que é situada logicamente entre uma camada de nível mais alto (usuários e aplicações) e uma camada de nível mais baixo (sistemas operacionais e protocolos de comunicação). Por esse motivo, essa camada é chamada de *middleware* (TANENBAUM; STEEN, 2007).

A Figura 1 representa um sistema distribuído com quatro computadores em rede permitindo que os componentes de uma aplicação se comuniquem com eles mesmos e ao mesmo tempo se comuniquem com diferentes aplicações. Os sistemas operacionais utilizados em cada máquina não necessariamente são os mesmos, porém o middleware faz uma camada de abstração que traz a sensação ao usuário de estar em um sistema único (TANENBAUM; STEEN, 2007).

2.3 Redes de Computadores

Tanenbaum (2003) define rede de computadores como:

> "Conjunto de computadores autônomos interconectados por uma única tecnologia."

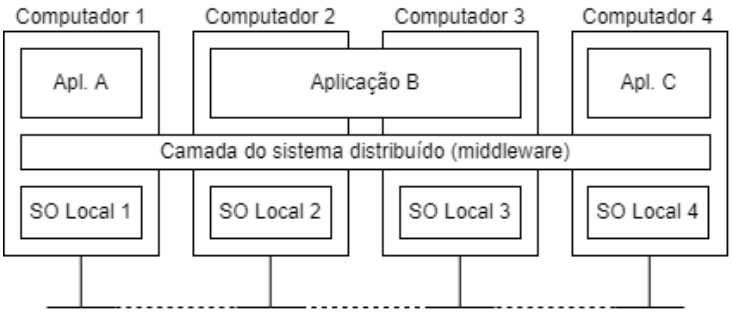

Figura 1 – Sistema distribuído organizado como middleware - Imagem retirada de (TANENBAUM; STEEN, 2007).

Tal definição leva a uma errada compreensão de que uma rede de computadores é um sistema distribuído e vice-versa. De fato, eles são bastantes semelhantes, mas a diferença se dá que, em um sistema distribuído, o *middleware* implementa um único modelo que é apresentado ao usuário, portanto um conjunto de computadores independentes passa a impressão de um único sistema coerente. Já em uma rede de computadores, o *middleware* não está presente fazendo com que esse modelo único também não exista, expondo ao usuário as máquinas reais, deixando claro que ele está lidando com um sistema heterogêneo que possui componentes e sistemas operacionais diferentes (TANENBAUM, 2003).

Quando se tem um *middleware* instalado em uma rede, trata-se de um sistema distribuído. Pode-se dizer então que um sistema distribuído é um sistema de software instalado em uma rede. Portanto, em nível de *hardware*, um sistema distribuído e uma rede são a mesma coisa e fica sob responsabilidade do sistema operacional determinar a diferença entre um sistema distribuído e uma rede (TANENBAUM, 2003).

Para que tudo funcione perfeitamente, uma rede precisa ter conjunto de regras que rejam a comunicação entre dois ou mais hosts. Esse conjunto de regras é chamado de protocolos (ALANI, 2014).

2.3.1 Modelo em camadas

Para padronização e melhor entendimento da operação de uma rede de computadores, foi criado um modelo baseado em camadas. Esse modelo divide em camadas as regras (protocolos) que precisam ser obedecidas para que haja a interconectividade entre dois sistemas, o que facilita a implementação de uma rede pois cada camada tem seus protocolos específicos e bem definidos, fazendo com que esses protocolos consigam executar suas funções de forma mais eficiente. Com essa separação, também fica mais fácil de resolver eventuais problemas, visto que se torna possível isolar cada camada sem que seja necessário mexer na rede como um todo (ALANI, 2014).

2.3.1.1 Open Systems Interconnection (OSI)

Este modelo foi proposto por um grupo da Honeywell Information Systems, liderado por Mike Canepa e publicado na década de setenta pela International Organization for Standardization (ISO). Ele é baseado em sete camadas e cada camada é responsável por fazer uma parte do processo necessário para que duas máquinas se comuniquem. Essas camadas possuem uma hierarquia, fazendo com que cada camada só possa se comunicar com a camada que está imediatamente acima ou abaixo dela. Cada camada trata seus dados de uma maneira específica, de modo que cada uma é capaz de adicionar novos dados aos dados resultantes da camada anterior, processo chamado de encapsulamento, e esses dados tratados recebem o nome de Protocol Data Unit (PDU) (ALANI, 2014).

Este modelo possui uma estratégia de camada em pares, o que significa dizer que cada informação adicionada em uma camada pelo lado do emissor deve chegar à camada de mesmo nível do lado do receptor (ALANI, 2014).

Na Tabela 1, está representada as sete camadas do modelo OSI cujas respectivas responsabilidades estão dispostas da seguinte maneira:

- **Física:** Estabelecer a comunicação real entre os dispositivos;
- **Enlace:** Detectar e corrigir erros que aconteceram na camada anterior, controlar o fluxo da transmissão de dados entre dispositivos;
- **Rede:** Endereçar os dispositivos na rede, determinar o caminho que as informações deverão seguir da origem até o destino (roteamento);
- **Transporte:** Detectar e corrigir erros das camadas inferiores, controlar o fluxo de dados da origem até o destino e ordenar para garantir que os dados cheguem da mesma forma que foram enviados;
- **Sessão:** Viabilizar comunicação entre processos dos diferentes sistemas;
- **Apresentação:** Converter formatos de caracteres para serem usados na transmissão, responsável pela compressão e criptografia;
- **Aplicação:** Oferecer serviços aos usuários finais;

Tabela 1 – Representação das sete camadas do modelo OSI - Tabela retirada de (ALANI, 2014).

Aplicação
Apresentação
Sessão
Transporte
Rede
Enlace
Física

2.3.1.2 Pilha Transmission Control Protocol/Internet Protocol (TCP/IP)

O modelo OSI é um modelo de referência, mas, na prática a internet está modelada de outra forma. A modelagem da internet atualmente segue o modelo TCP/IP, também conhecido como pilha TCP/IP. Este modelo surgiu do projeto Advanced Research Projects Agency NETwork (ARPANET), desenvolvido pelo Departamento de Defesa dos Estados Unidos. Com o advento das tecnologias sem fio, fez-se necessário evoluir o projeto da ARPANET, o que deu origem ao atual modelo TCP/IP (ALANI, 2014).

O modelo TCP/IP funciona da mesma forma que o modelo OSI, por ser baseado em camadas, mas, ao contrário do modelo OSI, utiliza apenas quatro camadas. Como a pilha TCP/IP foi modelada em cima de protocolos existentes, as funções de suas camadas são definidas através dos seus respectivos protocolos. A Tabela 2 mostra as camadas do modelo TCP/IP[1](ALANI, 2014).

Tabela 2 – Representação das quatro camadas do modelo TCP/IP - Tabela retirada de (ALANI, 2014).

Aplicação
Transporte
Internet
Rede

Ao comparar a pilha TCP/IP com o modelo OSI é possivel tirar algumas conclusões:

- A camada *Rede* é a junção das camadas *Física* e *Enlace* do modelo OSI;
- A camada *Internet* tem relação direta com a camada *Rede* do modelo OSI;

[1] Existem divergências em relação à quantidade de camadas da pilha TCP/IP em literatura técnica de fins didáticos. Tanenbaum (2003) e Kurose e Ross (2016) definem a pilha TCP/IP com um modelo de cinco camadas, sendo elas: Físico, Enlace, Rede, Transporte e Aplicação. A diferença na descrição destes autores se dá pelo fato de manterem as camadas de Enlace e Física separadas, enquanto Alani (2014) une essas camadas dando o nome de camada de Rede e a camada de Rede, definida por Tanenbaum e Kurose, recebe o nome de Internet.

- A camada *Transporte* tem relação direta com a camada *Transporte* do modelo OSI;
- A camada *Aplicação* é a junção das camadas *Sessão*, *Apresentação* e *Aplicação* do modelo OSI;

A Figura 2 representa a comparação dos dois modelos.

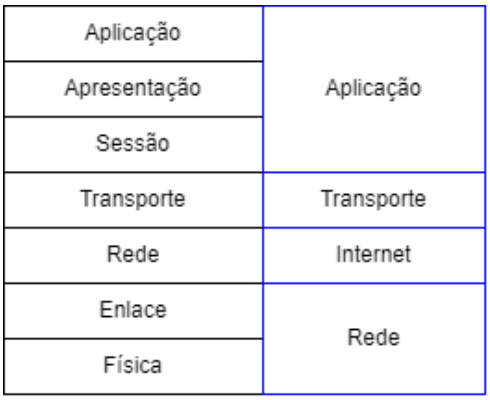

Figura 2 – Modelos de referência OSI e TCP/IP: à esquerda, modelo OSI, à direita, modelo TCP/IP - Imagem retirada de (ALANI, 2014).

2.3.1.3 Camada de Rede

A camada de rede é responsável por garantir que os pacotes de dados sejam transmitidos de forma eficiente entre dispositivos em uma rede (ALANI, 2014).

Os protocolos que operam nesta camada são responsáveis por estabelecer os procedimentos para acessar o meio de transmissão, além de fazer a correspondência entre endereços IP e endereços de hardware (MAC) (ALANI, 2014).

A maioria das configurações nesta camada é simples, como instalar ou ativar a pilha de software TCP/IP. O software necessário geralmente já é pré-instalado em muitos computadores, permitindo que os usuários se conectem diretamente a algumas redes (ALANI, 2014).

- Protocolo ARP (*Address Resolution Protocol*)

O ARP é um protocolo de comunicação utilizado para mapear um endereço de IP para um endereço de hardware, como o endereço MAC de uma placa de rede. Ele é utilizado para descobrir o endereço MAC de um host em uma rede local, permitindo que os pacotes IP sejam encaminhados para o host correto (HIJAZI; OBAIDAT, 2019).

ARP é um protocolo baseado em broadcast, onde uma mensagem é enviada para todos os dispositivos na rede, solicitando que o dispositivo com o endereço IP específico

responda com seu endereço MAC. O dispositivo que possui o endereço IP solicitado responde com seu endereço MAC, permitindo que os pacotes sejam encaminhados para o host correto (ZYDYK, 2021).

2.3.1.4 Camada de Internet

Essa camada tem como principal objetivo selecionar o melhor caminho para os dados viajarem de sua origem ao seu destino, ou seja, roteamento. Em termos de serviços, a camada de internet oferece endereçamento, encaminhamento, roteamento, controle de congestionamento e controle de erros (ALANI, 2014).

O principal protocolo operando nesta camada é o IP. Há, também, um grupo de protocolos de suporte que ajudam o IP a fazer seu trabalho, um deles é o ICMP (ALANI, 2014).

- Protocolo IP (*Internet Protocol*)

O IP é um dos protocolos fundamentais da pilha de protocolos TCP/IP. Ele é responsável por garantir a entrega de pacotes de dados de um host para outro através da rede. Ele faz isso através do uso de endereços IP únicos, que identificam cada dispositivo na rede (KOZIEROK, 2005).

Este protocolo não estabelece uma conexão antes da transferência de informações e não emprega um mecanismo de confirmação para garantir a entrega de seus pacotes. Esta operação fica para os protocolos de camadas mais altas, como o TCP (ALANI, 2014).

Além de garantir a entrega dos pacotes de dados, o IP também é responsável por fragmentar e remontar pacotes de dados, caso sejam muito grandes para serem transmitidos de uma só vez. Esse protocolo possui duas versões: o IPv4 e o IPv6 (KOZIEROK, 2005).

O IPv4 é a versão mais antiga e amplamente utilizada do protocolo IP, possui endereços de 32 bits e é dividido em quatro octetos separados por pontos, cada octeto é representado por um número decimal de 0 a 255. Esses endereços são atribuídos de forma estática ou dinâmica, dependendo das necessidades da rede (TANENBAUM, 2003).

Um endereço IPv4 costuma ser dividido em duas partes: endereço de rede e endereço do host. Endereço de rede é a parte do endereço IPv4 que identifica a rede à qual um dispositivo está conectado, é usado para determinar para qual rede um pacote deve ser enviado e essa parte do endereço é fixa. Endereço de host é a parte do endereço IPv4 que identifica de forma única um dispositivo na rede, é usado para identificar o dispositivo de destino ao qual um pacote deve ser enviado e essa parte do endereço é variável e pode ser configurada livremente (TANENBAUM, 2003).

A parte que é usada para determinar qual parte do endereço IPv4 representa a rede e qual representa o host, é chamada de máscara de sub-rede. Ela é representada por uma notação decimal e indica a quatidade de bits que serão fixos no endereço (TANENBAUM, 2003).

A Figura 3 representa o endereçamento de um IPv4.

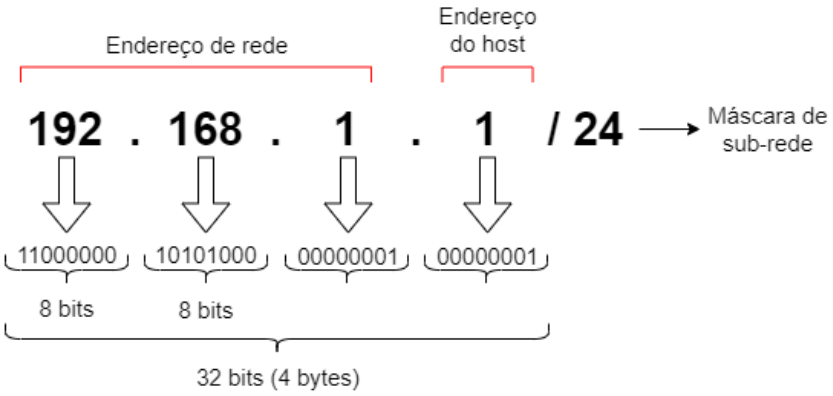

Figura 3 – Endereçamento IPv4.

Um dos problemas do IPv4 é a escassez de endereços disponíveis devido ao seu tamanho de 32 bits, cerca de 4,3 bilhões de endereços únicos, o que levou ao desenvolvimento do protocolo IPv6 com um endereçamento de 128 bits (TANENBAUM, 2003).

O IPv6 é a versão mais recente e possui endereços de 128 bits, o que permite cerca de 3,4 x 10^{38} endereços únicos e é composto por 8 grupos de 16 bits, separados por dois pontos, cada um representando 4 caracteres hexadecimais. Esses endereços também são atribuídos de forma estática ou dinâmica, dependendo das necessidades da rede. (HUITEMA, 1995).

Além disso, o IPv6 inclui uma nova forma de endereçamento, chamada endereçamento de link-local, que permite que os dispositivos em uma rede local sejam endereçados de forma privada e não-roteáveis. (HUITEMA, 1995).

O IPv6 é dividido de forma semelhante ao IPv4, no entanto, o que se chamava de endereço de rede no IPv4, aqui chama-se de prefixo de rede, que é a parte do endereço IPv6 que indica a rede à qual um dispositivo está conectado. A parte que define o prefixo é chamado de tamanho do prefixo. A parte que se chamava endereço de host antes, no IPv6 chama-se identificador de interface (IID) e é usado para identificar de forma única um dispositivo dentro da rede (HUITEMA, 1995).

A Figura 4 representa o endereçamento de um IPv6.

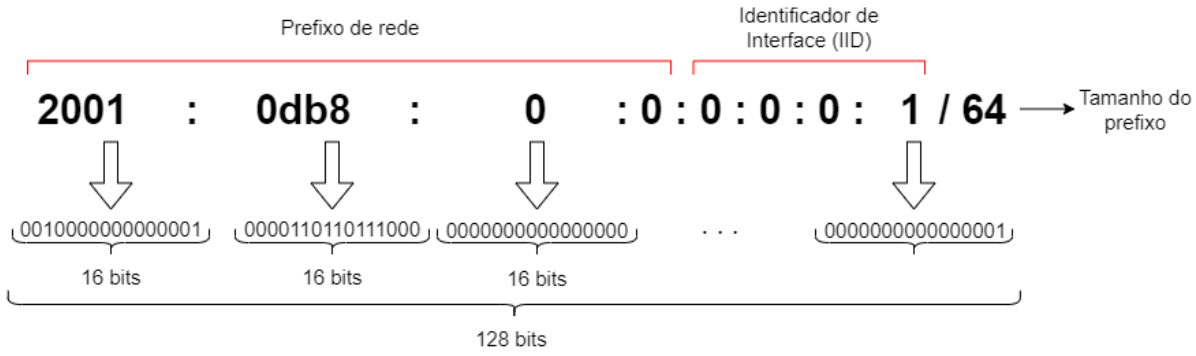

Figura 4 – Endereçamento IPv6.

- Protocolo ICMP (*Internet Control Message Protocol*)

O ICMP é um utilizado para enviar mensagens de erro e de controle entre dispositivos em uma rede, funcionando como um mecanismo de *feedback* entre dispositivos para indicar problemas na comunicação de dados. Ele é utilizado para notificar os dispositivos de erros de comunicação, como pacotes perdidos ou atrasos excessivos, também é utilizado para enviar solicitações de diagnóstico e notificar dispositivos de condições de congestionamento, permitindo que eles ajustem sua taxa de transmissão de dados para evitar sobrecarga na rede. Ele é usado em conjunto com outros protocolos, como o IP, para garantir a confiabilidade e a eficiência da comunicação de dados na rede.

- NAT (*Network Address Translation*)

O NAT é um método utilizado para permitir que dispositivos em uma rede privada acessem uma rede externa usando um único endereço IP público. Ele é utilizado principalmente para conservar endereços IP públicos globais, já que o número de dispositivos conectados à Internet tem aumentado rapidamente, também pode ser utilizado para fornecer segurança adicional à uma rede privada, permitindo que apenas pacotes originados de dentro da rede acessem recursos na Internet (WING, 2010).

NAT funciona mapeando endereços IP privados para endereços IP públicos, ou seja, quando um dispositivo na rede privada inicia uma conexão com uma rede externa, o dispositivo NAT na borda da rede traduz o endereço IP privado para um endereço IP público, permitindo que o pacote seja roteado na Internet. Quando um pacote retorna à rede privada, o dispositivo NAT traduz o endereço IP público de volta para o endereço IP privado, permitindo que o pacote seja entregue ao dispositivo correto (WING, 2010).

A Figura 5 representa o esquema NAT.

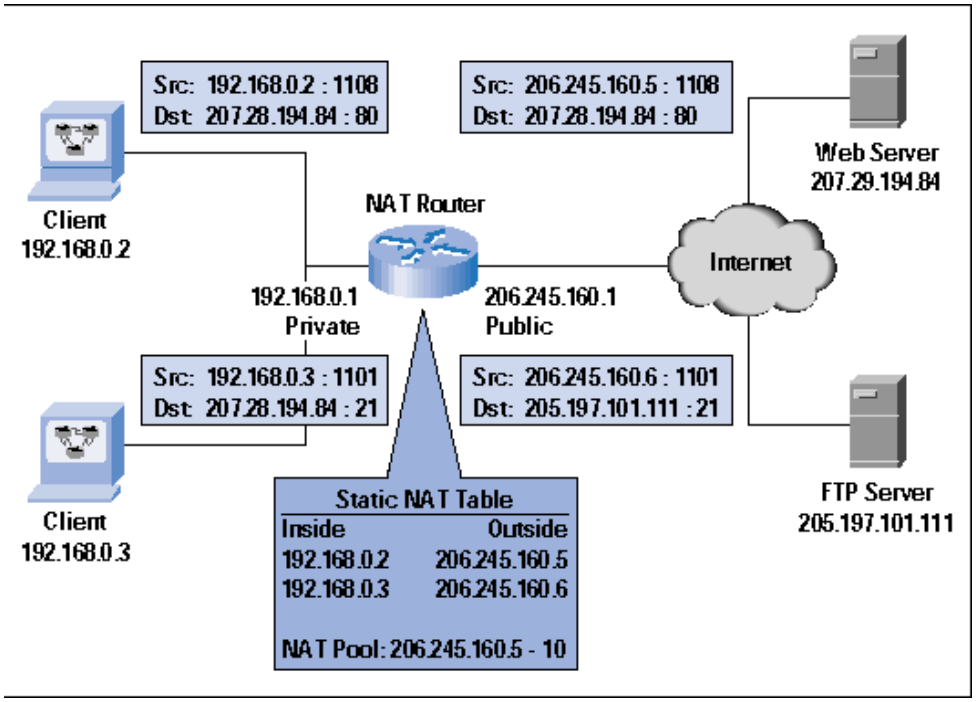

Figura 5 – Representação do esquema NAT - Imagem retirada de (SHARMA, 2014).

O NAT também pode ser combinado com outras técnicas de segurança de rede, como firewall, para fornecer uma camada adicional de proteção (WING, 2010).

- Protocolo DHCP (*Dynamic Host Configuration Protocol*)

O protocolo DHCP é utilizado para atribuir endereços IP dinamicamente a dispositivos em uma rede, permite que um dispositivo, como um computador ou roteador, solicite automaticamente um endereço IP a partir de um servidor DHCP, sem a necessidade de configuração manual (WOUNDY; KINNEAR, 2006).

O DHCP funciona através de uma série de mensagens entre o dispositivo cliente e o servidor DHCP. Quando um dispositivo é ligado à rede, ele envia uma mensagem de solicitação (DHCP Discover) para localizar um servidor DHCP, o servidor DHCP, por sua vez, responde com uma mensagem de oferta (DHCP Offer), que inclui um endereço IP disponível para uso pelo dispositivo, o dispositivo então envia uma mensagem de requisição (DHCP Request) para aceitar o endereço IP oferecido e o servidor DHCP responde com uma mensagem de confirmação (DHCP Ack), confirmando a atribuição do endereço IP (WOUNDY; KINNEAR, 2006).

Além de atribuir endereços IP, o DHCP também pode fornecer outras informações de configuração de rede, como endereços de gateway padrão e endereços de servidores DNS, também, pode ser configurado para fornecer informações de configuração específicas para dispositivos, com base no endereço MAC de um dispositivo (WOUNDY; KINNEAR, 2006).

Uma das principais vantagens do DHCP é a facilidade de gerenciamento de endereços IP em uma rede, como os endereços IP são atribuídos dinamicamente, eles podem ser reutilizados quando um dispositivo é desligado da rede, o que evita o esgotamento de endereços IP. Além disso, como o DHCP automatiza a configuração de endereços IP, ele pode ser particularmente útil em redes de grande escala, onde a configuração manual de endereços IP seria demorada e propensa a erros (WOUNDY; KINNEAR, 2006).

- *Firewall*

Firewall é uma ferramenta de segurança de rede que controla o acesso a uma rede privada e funciona como um filtro, permitindo ou bloqueando o tráfego de rede baseado em regras específicas (NOONAN; DUBRAWSKY, 2006).

Existem dois tipos principais de firewall: firewall baseado em hardware e firewall baseado em software. Firewalls baseados em hardware são dispositivos físicos que são instalados na rede e são geralmente mais rápidos e seguros do que os firewalls baseados em software, são projetados para lidar com grandes volumes de tráfego e podem ser configurados para atuar como gateways ou como roteadores. Já os firewalls baseados em software, são aplicativos que são instalados em computadores individuais e geralmente são menos caros do que os firewalls baseados em hardware e são projetados para lidar com menores volumes de tráfego (NOONAN; DUBRAWSKY, 2006).

Os firewalls também podem ser classificados como firewalls de estado de pacote ou firewalls de estado de conexão. Firewalls de estado de pacote verificam cada pacote individualmente e tomar uma decisão baseada nas regras configuradas. Já os firewalls de estado de conexão, acompanham as conexões ativas e tomar uma decisão com base nas regras configuradas e no estado da conexão (NOONAN; DUBRAWSKY, 2006).

2.3.1.5 Camada de Transporte

A camada de transporte foi projetada para dar à origem e ao destino a capacidade de ter uma conversa de ponta a ponta. No modelo TCP/IP, existem dois protocolos definidos que podem operar nesta camada: TCP e UDP (ALANI, 2014).

- Protocolo TCP (*Transmission Control Protocol*)

O protocolo TCP é utilizado para garantir que as informações transmitidas entre dispositivos em uma rede sejam entregues de forma confiável e precisa, ele estabelece uma conexão antes da transferência de informações e fornece uma forma de sequenciamento, de modo que, mesmo se os segmentos de dados chegarem em uma ordem diferente daquela em que foram enviados, eles possam ser reorganizados. Como o protocolo IP é instável,

o uso do TCP fornece a confiabilidade necessária para garantir que os dados cheguem seguros e inteiros (KUMAR; RAI, 2012).

- Protocolo UDP (*User Datagram Protocol*)

O protocolo UDP é utilizado para aplicações que necessitam de uma entrega rápida e eficiente de dados, sem a necessidade de confirmação de entrega ou garantia de ordem na sequência dos pacotes. Diferente do protocolo TCP, o protocolo UDP não estabelece uma conexão antes da transferência de informações, portanto, o UDP não fornece mecanismos de confiabilidade, porém, em comunicações que são tempo-críticas, como videoconferências, ele pode ser mais interessante (KUMAR; RAI, 2012).

2.3.1.6 Camada de Aplicação

Essa camada tem a função de pegar os dados das aplicações e entregá-los à camada de transporte e coletar dados da camada de transporte e entregá-los às aplicações corretas. Resumindo, ela oferece serviços aos usuários finais (ALANI, 2014).

O modelo TCP/IP possui uma variedade de protocolos de alto nível que atendem a uma ampla gama de aplicações. Alguns desses protocolos são utilizados diretamente pelos usuários, como o FTP e o SSH, enquanto outros são usados diretamente pelas aplicações, como o SMTP e o HTTP. Além disso, existem protocolos que são usados indiretamente ou pelos programas e rotinas do sistema operacional, como o DNS (ALANI, 2014).

A maioria desses protocolos se baseia em comunicação de texto simples, usando o código ASCII. Isso foi escolhido pois é um código geral que pode ser compreendido por quase qualquer tipo de computador. No entanto, essa abordagem tem o risco de que a comunicação possa ser interceptada e compreendida facilmente, o que não é aceitável em muitos casos. Para solucionar essa questão, foram desenvolvidas versões seguras dos protocolos, como o HTTPS e o SSL, que garantem a autenticação dos hosts envolvidos na comunicação e a criptografia dos dados transmitidos (ALANI, 2014).

- Protocolo HTTP (*Hypertext Transfer Protocol*)

O protocolo HTTP é utilizado para transferir informações, é baseado em requisições e respostas, onde um cliente envia uma requisição para um servidor, solicitando informações, e o servidor responde com as informações solicitadas (FIELDING et al., 1999).

Diferente de outros protocolos, como o TCP, que mantém uma conexão aberta entre as partes para garantir a entrega de dados, o HTTP é um protocolo sem estado, o que significa que as requisições são independentes umas das outras e não mantêm nenhum tipo de sessão (FIELDING et al., 1999).

- Protocolo SMTP (*Simple Mail Transfer Protocol*)

O protocolo SMTP é utilizado para enviar mensagens de e-mail entre servidores de correio eletrônico e é baseado em texto simples. Ele usa o protocolo TCP para garantir que as mensagens sejam transmitidas de forma confiável e também suporta a inclusão de anexos de arquivos e recursos de criptografia para garantir a segurança das mensagens transmitidas (POSTEL, 1982).

O SMTP é um protocolo antigo, desenvolvido em 1982, ele é utilizado por todos os sistemas de correio eletrônico, mas tem algumas limitações, ele não suporta recursos avançados como a entrega de mensagens não lidas e a autenticação do usuário. (POSTEL, 1982).

- Protocolo DNS (*Domain Name System*)

O protocolo DNS é um sistema de nomes de domínio que traduz os nomes de domínio em endereços IP e vice-versa, permitindo que os usuários acessem sites e recursos de rede utilizando nomes fáceis de lembrar, em vez de endereços IP numéricos (AWS, 2023).

O DNS funciona como uma hierarquia de servidores, onde cada nível contém informações sobre um conjunto específico de nomes de domínio. Os servidores DNS de nível superior, chamados de raiz, contêm informações sobre os domínios de nível superior, como “com“, “org“ e “edu“. Já os servidores DNS de nível inferior, chamados de autoritativos, contêm informações sobre os nomes de domínio específicos, como "example.com"(AWS, 2023).

- *Proxy*

O proxy é um servidor que atua como um intermediário entre os usuários e outros servidores na internet, é usado para melhorar a segurança, o desempenho e a gerenciabilidade de uma rede e, também, pode ser usados para controlar o acesso à internet, bloqueando sites específicos ou permitindo apenas o acesso a sites aprovados (TANENBAUM, 2003).

Existem vários tipos de proxy, cada um com uma finalidade específica. O proxy de cache, por exemplo, armazena cópias de páginas web frequentemente acessadas para que elas possam ser fornecidas rapidamente aos usuários sem precisar ser baixadas novamente da internet. O proxy de anonimato, por outro lado, oculta a identidade do usuário ao fornecer acesso à internet (TANENBAUM, 2003).

3 Proposta de Trabalho

A proposta do trabalho consiste no desenvolvimento e documentação de um conjunto de experimentos práticos para realizar a operação básica de uma rede de computadores utilizando os sistemas operacionais FreeBSD e Linux. Este conjunto de experimentos servirá para a utilização na disciplina “fundamentos de redes de computadores“, da Faculdade do Gama na Universidade de Brasília.

Cada experimento abordará um tema relacionado a uma determinada camada da pilha TCP/IP, viabilizando, através através da prática, o entendimento dos principais protocolos necessários para que haja uma conexão entre dois computadores.

Como será abordado na Seção 4.1, para cada experimento será gerado um documento que os alunos irão seguir e, no final de todo o conjunto de laboratórios, o aluno terá a base de conhecimento para operação básica de uma rede de computadores. Seguindo a estrutura deste documento, definiu-se o conjunto de experimentos propostos.

3.1 Tema e Introdução

Baseado na ementa da disciplina “fundamentos de redes de computadores“, foram definidos onze temas de experimentos práticos julgados relevantes para o entendimento e operação básica da pilha TCP/IP. Estes experimentos estão divididos de acordo com as camadas do modelo TCP/IP, exercitando a compreensão dos principais protocolos da respectiva camada.

A seguir, estão apresentados os temas que serão abordados, juntamente com uma breve descrição do que será tratado no experimento:

- **Introdução às Redes de Computadores:** Algumas configurações básicas necessárias para o correto funcionamento de equipamentos conectados a redes.
- **Depuração de Problemas na Camada de Aplicação:** Quando um computador está devidamente configurado em uma rede de computadores, é interessante que ele se comunique com outros equipamentos para o provimento de serviços a usuários. Portanto, este experimento apresenta um conjunto mínimo de ferramentas que permitirão a execução de um diagnóstico preciso ao se encarar uma situação de interrupção ou instabilidade de serviço típico de camada de aplicação.
- **Depuração de Problemas na Camada de Transporte:** Para que haja uma conexão por meio de um canal de transmissão confiável, é necessário uma série

de mecanismos que protejam a comunicação quanto a efeitos adversos do meio de transmissão: corrupção de dados; congestionamentos e perdas de pacote. Portanto, este experimento apresenta um conjunto mínimo de ferramentas que permitirão a execução de um diagnóstico preciso ao se encarar uma situação de interrupção ou instabilidade de serviço típico de camada de transporte.

- **Depuração de Problemas na Camada de Internet:** O conjunto de protocolos que cooperam no provimento dos serviços típicos de camada de rede podem ser separados em duas categorias: protocolos que geram tabelas de encaminhamento e os protocolos que utilizam as tabelas de encaminhamento. Portanto, este experimento apresenta um conjunto mínimo de ferramentas que permitirão a execução de um diagnóstico preciso ao se encarar uma situação de interrupção ou instabilidade de serviço típico de camada de internet.
- **Camada de Internet (DHCP):** Alguns serviços de nível de internet complementam as funções básicas de rede, oferecendo funcionalidades que facilitam a administração da rede. Este experimento apresenta a configuração de um dos mais úteis serviços de rede em uso nas redes TCP/IP hoje, é implementado pelo protocolo DHCP (Dynamic Host Configuration Protocol).
- **Camada de Internet (NAT):** O NAT (Network Address Translation) é um esquema que torna viável montar uma rede endereçada por meio de um conjunto de IPs restritos de forma que todos os hosts compartilham um número pequeno de IPs válidos e, ainda sim, são capazes de manter conexões regulares com equipamentos na Internet. Portanto, este experimento apresenta a configuração do serviço que é implementado pelo NAT.
- **Camada de Rede (ARP):** Para o correto funcionamento de redes, alguns serviços de nível de camada de rede são primordiais para a adequada cooperação da pilha de protocolos sobre a qual reside a Internet. Portanto, este experimento apresenta os princípios do protocolo ARP e como acontece suas interações.
- **Operação e Proteção de Redes (Firewall):** Firewall é uma solução de segurança baseada em *hardware* ou *software* (mais comum) que, a partir de um conjunto de regras ou instruções, analisa o tráfego de rede para determinar quais operações de transmissão ou recepção de dados podem ser executadas. Basicamente, o objetivo de um firewall é bloquear tráfego de dados indesejado e liberar acessos bem-vindos. Este experimento tem o intuito de configurar um firewall em uma rede.
- **Camada de Aplicação (Proxy):** Em redes de computadores, um proxy é um servidor (um sistema de computador ou uma aplicação) que age como um intermediário para requisições de clientes solicitando recursos de outros servidores. Este experimento apresenta algumas regras de um proxy.

- **Camada de Aplicação (DNS):** Um servidor DNS (Domain Name System) é comumente usado para traduzir endereços de IP numéricos para nomes de domínio amigáveis para humanos e vice-versa. Portanto, este experimento apresenta a configuração de um servidor DNS.

- **Introdução ao IPv6:** O IPv6 é o sucessor do IPv4 e foi criado para resolver os problemas de escassez de endereços IP que ocorriam no IPv4. Ele fornece uma quantidade praticamente ilimitada de endereços IP, permitindo que as redes cresçam e evoluam de forma mais flexível e escalável. Portanto, este experimento apresenta a configuração de um IPv6.

3.2 Objetivos

Para cada tema, definiram-se, também, os objetivos que descrevem o que se deseja alcançar ao final da execução do experimento. A lista abaixo relaciona os temas e seus respectivos objetivos:

- **Introdução às Redes de Computadores:** Compreender as configurações básicas para navegabilidade em uma rede de computadores, exercitar configurações básicas em diferentes sistemas operacionais e entender como usar ferramentas de diagnóstico para validar configurações.

- **Depuração de Problemas na Camada de Aplicação:** Exercitar uma comunicação típica HTTP e SMTP por meio de ferramenta de diagnóstico (`telnet`) e exercitar as configurações de rede, especialmente no que tange ao serviço de resolução de nomes.

- **Depuração de Problemas na Camada de Transporte:** Exercitar uma comunicação típica TCP por meio de ferramentas de diagnóstico (`telnet`, `netcat`, `netstat`, `sockstat` e `nmap`).

- **Depuração de Problemas na Camada de Internet:** Exercitar os princípios básicos de uma comunicação em redes TCP/IP, com ênfase nos serviços típicos de camada de rede. Conhecer e manipular ferramentas de diagnóstico (`ping`, `traceroute`, `netstat` e `route`) para fixação de conceitos de camada de rede.

- **Camada de Internet (DHCP):** Visualizar a importância dos serviços de atribuição dinâmica de configurações e entender como funciona a implementação do DHCP para configurá-la.

- **Camada de Rede (NAT):** Visualizar a importância dos serviços de compartilhamento de IPs e entender como funciona a implementação do NAT para configurá-la.

- **Camada de Internet (ARP):** Exercitar conceitos referentes à camada de rede e entender o papel do protocolo ARP e como acontecem suas interações.

- **Operação e Proteção de Redes (Firewall):** Entender como funciona a implementação de firewalls.

- **Camada de Aplicação (Proxy):** Demonstrar a utilização de regras de Proxy.

- **Camada de Aplicação (DNS):** Visualizar a importância dos serviços de resolução de nomes e entender como funciona a implementação do DNS para configurá-la.

- **Introdução ao IPv6:** Compreender as configurações básicas para navegabilidade em uma rede de computadores utilizando o IPv6, exercitar configurações básicas em diferentes sistemas operacionais e entender como usar ferramentas de diagnóstico para validar configurações.

3.3 Teoria abordada no experimento

A teoria abordada em cada experimento serve para referir qual conteúdo teórico o experimento prático está apresentando. A Tabela 3 relaciona os temas e seus respectivos conteúdos teóricos.

Tabela 3 – Temas e seus respectivos conteúdos teóricos

Tema	Referências Teóricas
Introdução às Redes de Computadores	Funcionamento básico de uma rede TCP/IP
Depuração de Problemas na Camada de Aplicação	Protocolos de Camada de Aplicação
Depuração de Problemas na Camada de Transporte	Protocolos de Camada de Transporte
Depuração de Problemas na Camada de Internet	Protocolos de Camada de Internet
Camada de Internet (DHCP)	Objetivo e funcionamento do protocolo DHCP
Camada de Internet (NAT)	Objetivo e funcionamento do esquema NAT
Camada de Rede (ARP)	Objetivo e funcionamento do esquema ARP
Firewall	Objetivo e funcionamento de firewalls
Camada de Aplicação (Proxy)	Objetivo e funcionamento de um proxy de aplicação
Camada de Aplicação (DNS)	Objetivo e funcionamento do protocolo DNS
Introdução ao IPv6	Conceitos básico de uma rede TCP/IP utilizando o IPv6.

3.4 Material Necessário

Essa seção apresenta a lista que relaciona os materiais que serão necessários para execução de cada experimento de acordo com o tema definido. Tendo em vista que alguns dos materiais são de uso comum a todos os experimentos, para reduzir o texto, a primeira lista enumera os materiais que serão usados em todos os experimentos, enquanto a segunda lista enumera os materiais específicos de cada experimento.

3.4.1 Materiais de uso comum

Tendo em vista que alguns dos materiais são de uso comum a todos os experimentos, para reduzir o texto, a lista abaixo enumera os materiais que serão usados em todos os experimentos.

1. Interfaces de rede (NIC's)
2. Máquinas com sistema FreeBSD
3. Cabos de rede – par trançado normal
4. Switches ou HUBs
5. Software nas máquinas: ambiente FreeBSD básico

3.4.2 Materiais de uso específico

A lista abaixo tem o intuito de enumerar os materiais específicos de cada experimento.

1. **Depuração de Problemas na Camada de Aplicação:**
   - Ferramentas de diagnóstico: `ifconfig`, `ping`, `traceroute`, `nslookup`, `sockstat`, `host`, `dig`
2. **Depuração de Problemas na Camada de Transporte:**
   - Ferramentas de diagnóstico: `telnet`, `nmap`, `netcat` e `netstat`
3. **Depuração de Problemas na Camada de Internet:**
   - Ferramentas de diagnóstico: `ping`, `traceroute`, `netstat` e `route`

4 Metodologia

Para que a comunicação entre duas máquinas, em uma rede de computadores, se efetive, é essencial a configuração adequada de parâmetros da pilha TCP/IP nos sistemas operacionais que executam em cada um dos ativos conectados em rede. Portanto, este capítulo tem o intuito de apresentar as etapas adotadas no desenvolvimento de cada experimento, que compõem o conjunto de experimentos práticos, para ser feita a operação básica de uma rede de computadores, além de definir um fluxo de atividades desempenhadas neste Trabalho de Conclusão de Curso e trazer a seção de cronograma, que descreve quando as atividades foram ou serão desempenhadas.

4.1 Modelo de Documento do Experimento

No intuito de desenvolver uma certa quantidade de experimentos que, em conjunto, permitem a operação básica de uma rede de computadores, fez-se necessário definir um modelo de documento a ser seguido em cada um desses experimentos. Este modelo apresenta sempre a mesma estrutura:

- **Cabeçalho:** Traz informações a respeito do experimento;
- **Tema:** Define o assunto e serve para nortear o experimento;
- **Introdução:** Breve descrição do conteúdo que será abordado;
- **Objetivos:** Descreve o que se deseja alcançar ao final da execução do experimento;
- **Teoria abordada no experimento:** Serve para referir a qual conteúdo teórico o experimento prático está abordando;
- **Material Necessário:** Descreve quais serão os materiais necessários para executar o experimento;
- **Roteiro:** Detalhamento do experimento. Passo a passo que o aluno terá que seguir para conclusão do ensaio;
- **Questões para Estudo:** Algumas questões extras para incentivar o aluno a buscar mais conhecimento além do que o roteiro proporciona;
- **Referências Teóricas:** Lista fonte de materiais secundários que fornece contexto e explicação teórica para cada parte do experimento;

4.2 Experimento prático

Para o planejamento, configuração e execução dos experimentos práticos, partindo do princípio que os experimentos serão realizados em uma rede isolada com acesso a internet, é necessária a disponibilidade de alguns materiais. A Tabela 4 lista estes recursos.

Tabela 4 – Materiais necessários para execução dos experimentos

Materiais
Duas ou mais interfaces de rede (NICs)
Duas ou mais máquinas com os sistema operacional Linux ou FreeBSD
Cabos de rede – par trançado normal
Switches ou HUBs

Fonte: Autor.

Tendo em vista que o sistema operacional utilizado será o FreeBSD ou o Linux, torna-se necessária a instalação de alguns softwares que não vem instalados por padrão nestes sistemas. A lista abaixo relaciona os softwares que serão necessários para a execução dos experimentos, com uma breve descrição da sua utilidade, alguns já vêm incluídos na distribuição, enquanto outros precisam ser instalados:

- `ifconfig`: Ferramenta de linha de comando usada para configurar e verificar as configurações de rede de um computador. Pode ser usado para atribuir endereços IP, definir gateways, configurar máscaras de sub-rede, definir interfaces de rede e muito mais;
- `route`: Ferramenta de linha de comando usada para gerenciar o roteamento de pacotes em uma rede. Permite que os usuários configurarem, exibam e modifiquem as rotas existentes para conectar um host a outros na rede. Além disso, a ferramenta também pode ser usada para exibir e modificar informações de roteamento, como a tabela de roteamento, as configurações de interface e os endereços IP;
- `ping`: Utilitário de linha de comando usado para testar a conectividade de rede em um determinado destino. Permite ao usuário determinar se um endereço IP ou um nome de host está acessível. O ping também mede a velocidade da conexão ao enviar e receber pacotes de dados para o destino;
- `telnet`: Cliente de terminal remoto que permite que um usuário se conecte a outra máquina remota via TCP/IP e execute comandos. Pode ser usado para testar conectividade de rede e realizar depuração de problemas de rede;
- `netcat`: Ferramenta de rede amplamente utilizada para administração de servidores de rede. Permite a criação de conexões TCP/UDP, envio de pacotes de rede de forma segura, conexões SSH, criação de conexões de túnel, entre outras.

- `netstat`: Ferramenta usada para exibir informações sobre as conexões de rede e protocolos ativos em um sistema FreeBSD. Pode ser usada para verificar se um serviço está escutando em uma porta específica, verificar o estado de uma conexão ou exibir a tabela de roteamento;
- `sockstat`: Ferramenta usada para exibir informações sobre conexões de rede abertas em um sistema. Mostra informações sobre a porta, o processo e o endereço IP associado a cada conexão;
- `traceroute`: Ferramenta usada para descobrir o caminho de rede entre um dispositivo de origem e um destino. Identifica cada dispositivo no caminho com seu endereço IP e fornece informações sobre o tempo de resposta para cada dispositivo;
- `tcpdump`: Ferramenta usada para capturar pacotes de rede e exibir seu conteúdo em formato de texto para fins de análise e diagnóstico;
- `host`: Ferramenta usada para consultar o sistema DNS para obter informações sobre endereços IP, nomes de host e outras informações de domínio. Útil para verificar se um nome de host ou um endereço IP são reconhecidos pelo sistema DNS local ou remoto;
- `squid`: Servidor proxy e cache de internet, projetado para fornecer serviços de acesso ao conteúdo da web, controle de acesso e melhoria da performance.
- `dig`: Utilitário de linha de comando usado para executar consultas DNS. Com ele, é possível realizar testes de DNS e verificar os servidores de nomes, registros DNS e configurações de dominio;
- `nmap`: Ferramenta de segurança de código aberto amplamente utilizada para descobrir hosts e serviços na rede. Permite que você descubra quais portas estão abertas em um servidor, quais serviços (aplicações web, e-mail, FTP, etc.) estão em execução, quais sistemas operacionais estão em execução e muito mais;
- `apache`: Servidor web que fornece aos usuários a capacidade de hospedar seus próprios sites, servindo conteúdo estático e dinâmico aos usuários da rede;
- `nslookup`: Utilitário de linha de comando usado para consultar os servidores DNS e recuperar informações do servidor DNS, como o endereço IP de um hostname, o endereço IP de um servidor de correio ou o endereço IP de um servidor de nomes;
- `bind`: Software que implementa o protocolo DNS em sistemas operacionais como o Linux, Unix e Windows.;
- `isc-dhcp`: Software de servidor DHCP, que é usado para atribuir endereços IP dinâmicos e outros parâmetros de configuração de rede a dispositivos na rede;

Para a instalação de qualquer software que não venha com a distribuição basta executar, no FreeBSD, o comando:

```
# pkg install nome_do_pacote
```

E no Debian, o comando:

```
# apt-get install nome_do_pacote
```

A conexão com a internet é necessária apenas no momento de configuração dos ambientes de experimento, para a instalação de todas as dependências necessárias. Assim que os softwares necessários estiverem devidamente instalados nas máquinas que serão utilizadas, a conexão com a internet se torna desnecessária para a execução dos experimentos.

5 Resultados

Neste capítulo, apresenta-se a análise dos resultados obtidos a partir da implementação de onze experimentos práticos de configurações básicas de rede utilizando os sistemas operacionais FreeBSD e Linux. Aqui serão detalhados os procedimentos realizados, os resultados e as conclusões obtidas com cada experimento, bem como avaliar a eficácia dos experimentos como meio de ensino para alunos da disciplina “fundamentos de redes de computadores“.

5.1 Documentação dos Experimentos

Para documentar os experimentos realizados, foi desenvolvida uma página web utilizando a ferramenta `mkdocs`, essa ferramenta permite a geração de uma página estática e responsiva a partir de documentos escritos em markdown e configurados em um arquivo YAML. A página gerada foi publicada no Github, fazendo uso do Github Pages, para garantir a disponibilidade e acessibilidade dos experimentos.

A página web documenta cada experimento realizado, incluindo uma descrição detalhada do objetivo, das configurações utilizadas e dos passos que devem ser executados. Além disso, os experimentos estão separados em duas seções, FreeBSD e Debian, que define em qual sistema operacional o experimento deve ser executado. Essa página está disponível através do link: <https://markinlimac.github.io/monografia-redes/>.

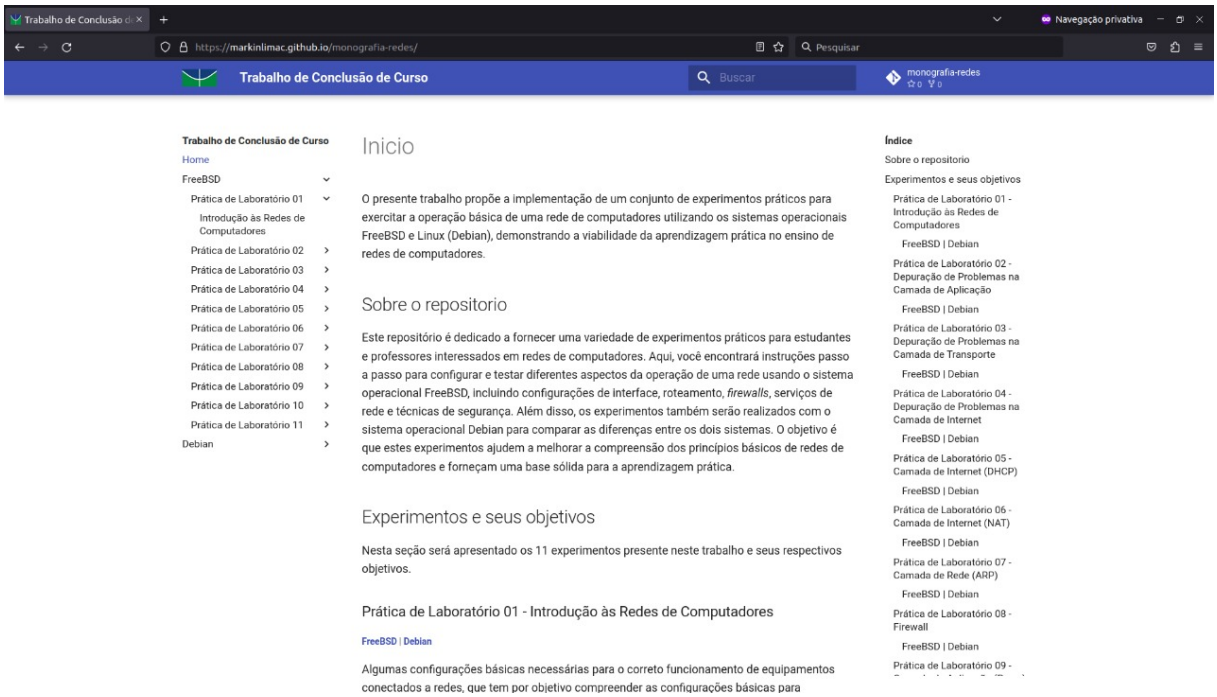

Figura 6 – Página de documentação dos experimentos.

Como pode ser observado na Figura 6, a página possui uma navegação fácil e uma apresentação clara dos experimentos, o que torna a consulta de informações mais simples e eficiente.

5.2 Conjunto de Experimentos Práticos

5.2.1 Introdução às Redes de Computadores

Na prática de laboratório 01, os alunos aprenderão a configurar manualmente o endereço IPv4 de um dispositivo e testar a sua funcionalidade na rede. Primeiramente, os alunos deverão configurar o endereço IPv4, sub-rede e gateway em uma máquina utilizando o sistema operacional FreeBSD ou Linux. Em seguida, deverão testar a comunicação entre as máquinas da rede realizando o `ping` entre elas. Além disso, o experimento introduz de forma básica como pode ser feito a configuração do servidor DNS e incentiva os alunos, de forma indireta, a buscarem informações sobre o servidor DHCP.

Após a realização do experimento, os alunos poderão observar se a configuração do endereço IPv4 foi bem-sucedida ao realizar o `ping` entre as máquinas e verificar se a resposta foi positiva, indicando que as máquinas estão se comunicando corretamente na rede, caso a resposta do `ping` seja negativa, o aluno poderá revisar sua configuração e corrigir os erros encontrados. Ao final deste experimento, o aluno terá compreendido as configurações básicas para navegabilidade em uma rede de computadores e a importância da configuração correta do endereço IP em dispositivos de rede, além de entender como usar ferramentas de diagnóstico para validar as configurações.

5.2.2 Depuração de Problemas na Camada de Aplicação

Na prática de laboratório 02, é necessário que alguns equipamentos estejam previamente configurados na rede, fornecendo serviços que são essenciais: DNS, HTTP e SMTP. Neste experimento, os alunos aprenderão a fazer interações com o DNS através do comando `host`, interações com o HTTP através do comando `telnet` e interações com o SMTP também através do comando `telnet`.

Através deste experimento, os alunos serão capazes de compreender a importância e a utilidade de cada uma dessas ferramentas na identificação de problemas com serviços de camada de aplicação. Eles aprenderão como executar testes básicos de conectividade com o DNS, HTTP e SMTP, terão a capacidade de compreender a saída dos comandos executados e serão capazes de diagnosticar simples problemas de conectividade com os serviços de camada de aplicação. Além disso, o experimento incentiva os alunos, de forma indireta, a buscarem informações sobre o arquivo de mapeamento de nomes de host, cabeçalho de uma requisição HTTP e a segurança padrão do protocolo SMTP.

5.2.3 Depuração de Problemas na Camada de Transporte

Na prática de laboratório 03, é necessário que alguns equipamentos estejam previamente configurados na rede, fornecendo serviços que são essenciais: DNS, HTTP e SMTP. Neste experimento, é ensinado exemplos de como abrir um socket servidor utilizando a ferramenta `netcat`, verificar a abertura desse socket utilizando a ferramenta `sockstat`, interagir com esse socket através da ferramenta `telnet` e também introduz a ferramenta `nmap`, utilizada para realizar varredura de uma rede para descobrimento de hosts e serviços ativos.

Ao finalizar o experimento, os participantes serão capazes de abrir sockets servidores, verificar sua abertura e interagir com eles utilizando os comandos apresentados. Também, aprenderão a realizar varredura de rede utilizando o `nmap`, o que permite que eles realizem diagnósticos precisos em situações de interrupção ou instabilidade de serviços típicos da camada de transporte. Além disso, o experimento incentiva os alunos, de forma indireta, a buscarem informações sobre: estados de portas; o arquivo de configuração usado pelo sistema operacional para conectar aos serviços de rede; diagnóstico no protocolo UDP; e a configuração de um transmissor básico de arquivos usando apenas as ferramentas executadas no experimento.

5.2.4 Depuração de Problemas na Camada de Internet

Na prática de laboratório 04, é necessário que alguns equipamentos estejam previamente configurados na rede, fornecendo serviços que são essenciais: DNS, HTTP, SMTP, DHCP e NAT. Nesse experimento, os alunos aprendem sobre o uso de uma série de ferramentas básicas, como `ifconfig`, `ping`, `netstat` e `traceroute`, para obter informações importantes sobre as interfaces de rede, interagir com outros clientes na mesma rede, verificar a tabela de roteamento, verificar o estado de atividade das interfaces de rede e verificar o caminho percorrido por um pacote, respectivamente.

Esse experimento mostra que o conhecimento destas ferramentas é fundamental para identificar possíveis problemas na camada de internet. Ao utilizar o comando `ifconfig`, por exemplo, é possível verificar informações detalhadas sobre as interfaces de rede, incluindo seus endereços IP, máscara de sub-rede, entre outras informações. Já o comando `ping` permite verificar a conectividade com outros clientes na mesma rede, identificando possíveis problemas de rede. O comando `netstat` permite verificar a tabela de roteamento e o estado de atividade das interfaces de rede. Por fim, a ferramenta `traceroute` permite verificar o caminho percorrido por um pacote na rede, o que é importante para identificar possíveis problemas de roteamento.

O experimento, também, incentiva os alunos, de forma indireta, a buscarem entender a fundo cada parâmetro de saída das ferramentas essenciais para diagnóstico de

problemas na camada de internet.

5.2.5 Camada de Internet (DHCP)

Na prática de laboratório 05, é necessário que a rede possua ponto de saída para a internet, para instalação de ferramentas que serão usadas ao decorrer do experimento. Neste experimento, será ensinado como instalar, configurar e executar um servidor DHCP, destacando os arquivos mais importantes envolvidos na configuração e suas opções. Também, será ensinado como executar o DHCP em apenas uma interface em dispositivos que possuem mais de uma interface, como verificar o arquivo de log do sistema para diagnóstico de eventuais problemas na configuração e será abordada a configuração e validação de clientes na rede para utilização do servidor DHCP configurado.

Após o término do experimento, os participantes deverão ter conhecimento sobre como configurar e executar um servidor DHCP manualmente, conseguindo destacar os arquivos mais importantes envolvidos na configuração e suas variadas opções de configurações, entender como executar o DHCP em apenas uma interface em dispositivos que possuem mais de uma interface, saber verificar o arquivo de log do sistema para diagnóstico de eventuais problemas na configuração e ser capaz de configurar e validar clientes na rede para utilização do servidor DHCP configurado. Além disso, o experimento incentiva os alunos, de forma indireta, a usarem programas que analisam o tráfego de rede para verificar as mensagens que são trocadas entre o cliente e o servidor; configurar de forma manual o DHCP para que dispositivos na rede tenham o endereço de IP previamente definido em função de seu endereço MAC; e buscar explicações mais detalhadas do funcionamento do DHCP.

5.2.6 Camada de Internet (NAT)

Na prática de laboratório 06, os alunos aprenderão a configurar mais de uma interface de rede em um único sistema, de forma que sejam identificadas corretamente pelo sistema operacional, entender o funcionamento do encaminhamento de pacotes, explicando como os pacotes são direcionados de uma interface para outra em uma rede e configurar regras de NAT utilizando ferramentas de filtragem de pacotes, de forma a permitir o compartilhamento de IPs.

Ao final do experimento, os alunos deverão ser capazes de compreender a importância dos serviços de compartilhamento de IPs, ser capazes de configurar uma rede utilizando NAT, explicar o funcionamento do encaminhamento de pacotes, utilizar ferramentas de filtragem de pacotes para configurar regras de NAT e visualizar as traduções NAT ativas, assim como fazer testes usando dispositivos conectados em redes diferentes para ver se pacotes transmitidos de uma conseguem chegar até a outra. Com essas ha-

bilidades, os alunos estarão preparados para solucionar problemas comuns relacionados à configuração de redes com compartilhamento de IPs.

O experimento, também, incentiva os alunos diretamente a pensarem em uma topologia de rede que seja necessário o uso do NAT, fornecendo apenas as informações do segmento de rede.

5.2.7 Camada de Enlace (ARP)

Na prática de laboratório 07, é necessário que alguns equipamentos estejam previamente configurados na rede, fornecendo serviços que são essenciais: HTTP. Nesse experimento, os alunos entendem o papel do protocolo ARP na camada de enlace e aprendem a executar ferramentas de captura de pacotes, verificar as entradas da tabela ARP e interagir com serviços na rede para atualizar a tabela ARP.

Durante o experimento, os participantes poderão verificar a tabela ARP através da execução de comandos específicos e compreender como ela é preenchida ao longo do tempo, conforme novos dispositivos se conectam à rede, também, os alunos aprenderão como capturar pacotes e visualizar informações importantes sobre as interações ARP na rede. Ao entender o papel do ARP e como ele funciona, eles serão capazes de realizar diagnósticos mais precisos em caso de problemas de conectividade na rede. Além disso, o exeperimento, incentiva os alunos, de forma indireta, a buscarem conhecimento em relação a ataques e vulnerabilidades que envolvem o protocolo ARP e sobre desempenho de uma rede que utiliza ARP.

5.2.8 Operação e Proteção de Redes (*Firewall*)

Na prática de laboratório 08, é necessário que alguns equipamentos estejam previamente configurados na rede, fornecendo serviços que são essenciais: SSH. Neste experimento, os alunos aprenderão os conceitos básicos de como funciona a implementação de *firewalls*, incluindo a configuração e teste de regras de filtragem, além de aprender habilitar ferramentas de *firewall* em dispositivos, entender a ordem de regras nas ferramentas de *firewall* e saber verificar o arquivo de log do sistema para diagnóstico de eventuais problemas na configuração.

Ao concluir o experimento, os alunos terão uma compreensão clara da importância do *firewall* na segurança da rede, serão capazes de configurar e testar regras de filtragem, habilitar ferramentas de *firewall* em dispositivos, entender a ordem de regras e saber verificar o arquivo de log do sistema no que diz respeito a *firewall*. Além disso, o experimento incentiva os alunos, de forma indireta, a usarem ferramentas de captura de pacotes para determinar sobre qual protocolo o SSH funciona e como identificar portas possivelmente filtradas em um dispositivo.

5.2.9 Camada de Aplicação (*Proxy*)

Na prática de laboratório 09, é necessário que a rede possua ponto de saída para a internet, para instalação de ferramentas que serão usadas ao decorrer do experimento. Neste experimento, será ensinado como instalar, configurar e executar um servidor *Proxy*, destacando os arquivos mais importantes envolvidos na configuração e suas opções. Também, será ensinado como verificar o arquivo de log do sistema para diagnóstico de eventuais problemas na configuração, a configuração e validação de clientes na rede para utilização do servidor *Proxy* configurado, além de, ensinar montar tanto um SNAT (Source Network Address Translation) quanto um DNAT (Destination Network Address Translation).

Após o término do experimento, os participantes deverão ter conhecimento sobre como configurar e executar um servidor *Proxy* manualmente, conseguindo destacar os arquivos mais importantes envolvidos na configuração e suas variadas opções de configurações (*White List* e *Black List*), saber verificar o arquivo de log do sistema para diagnóstico de eventuais problemas na configuração e ser capaz de configurar e validar clientes na rede para utilização do servidor *Proxy* configurado, seja com SNAT ou DNAT. Além disso, o experimento incentiva os alunos, de forma indireta, a procurarem informações sobre o funcionamento do protocolo HTTP através do servidor *Proxy*, refletirem sobre a diferença de *Firewall* e *Proxy* e se aprofundarem na ferramenta utilizada para configuração do servidor.

5.2.10 Camada de Aplicação (DNS)

Na prática de laboratório 10, é necessário que a rede possua ponto de saída para a internet, para instalação de ferramentas que serão usadas ao decorrer do experimento. Nesse experimento, os alunos irão aprender a instalar, configurar e executar um servidor DNS, compreendendo os arquivos importantes na configuração e suas funcionalidades no serviço que será provido, e configurar e validar manualmente clientes na rede para utilização do servidor DNS configurado.

Ao final do experimento, os alunos deverão ter compreendido como funciona o protocolo DNS e sua importância na Internet, além de ter habilidade para configurar um servidor DNS e validar sua funcionalidade. Eles deverão ser capazes de explicar como os nomes de domínios são resolvidos para endereços IP e como os arquivos importantes na configuração do servidor DNS contribuem para esse processo. Também, os alunos, deverão ser capazes de configurar manualmente clientes na rede para usar o servidor DNS configurado, garantindo a resolução correta de nomes de domínios. Além disso, o experimento incentiva os alunos, de forma indireta, a entender como funciona a configuração do DNS de um servidor de email, buscar explicação de como seria configurado uma zona secundária, entender como funciona o protocolo DNS juntamente com o NAT e como monitorar

e solucionar problemas comuns de desempenho em um servidor DNS.

5.2.11 Introdução ao IPv6

Na prática de laboratório 11, é necessário que a rede possua ponto de saída para a internet, para testes utilizando o IPv6, também é importante ter alguns equipamentos previamente configurados na rede, fornecendo serviços que são essenciais: DHCPv6. Neste experimento, são exploradas as configurações básicas necessárias para o funcionamento correto de equipamentos conectados a uma rede de computadores utilizando o IPv6. Os alunos irão aprender a configurar o IPv6 de escopo global, link local e unique local de forma manual e automática e testar essas configurações para verificar se elas estão funcionando corretamente, fazendo comunicação com outra máquina da rede. Além disso, é apresentado endereços multicast úteis em uma rede, a configuração de um roteador para funcionar com IPv6 e como usar um servidor DHCPv6 previamente configurado para recebimento de endereços de escopo global automaticamente.

No final deste experimento, os alunos terão compreendido as configurações básicas para navegabilidade em uma rede de computadores utilizando o IPv6 e a importância da utilização desse tipo de endereço atualmente, além de entenderem como usar ferramentas de diagnóstico para validar as configurações. Eles serão capazes de configurar um endereço IPv6 manualmente e automaticamente, testar suas configurações e verificar se a comunicação com outra máquina na rede está funcionando corretamente. Também, aprenderão sobre endereços multicast, como configurar um roteador para funcionar com IPv6 e como usar um servidor DHCPv6.

O experimento, também, incentiva os alunos, de forma indireta, a entenderem o funcionamento do SLAAC (*StateLess Address AutoConfiguration*) e do protocolo NDP, a pensarem e relação à configuração de um roteador de forma automática e a existência do NAT em uma rede IPv6.

5.3 Referências Bibliográficas dos Experimentos

As referências bibliográficas são uma parte importante de cada experimento, pois fornecem uma base sólida de informações para a execução dos passos descritos. Estas referências incluem fontes como livros, artigos, documentações técnicas e manuais, e são usadas para garantir a precisão e a validade dos resultados obtidos e, também, permitem que os leitores possam aprofundar seu conhecimento sobre o assunto, ajudando a compreender melhor o experimento e a realizar verificações adicionais se necessário. Portanto, todos os experimentos gerados possuem a seção de rerências bibliográficas para garantir a integridade dos experimentos e a confiabilidade dos resultados obtidos.

5.4 Considerações Acerca dos Experimentos

É importante ressaltar que a validação dos experimentos didáticos é uma parte crucial do processo de ensino-aprendizagem para garantir a qualidade e relevância dos experimentos apresentados.

A validação rigorosa dos experimentos didáticos junto a uma turma de Fundamentos de Redes de Computadores seria interessante, pois esse tipo de validação é mais preciso e confiável, porque permite a aplicação prática dos experimentos em uma situação realista e avalia sua eficiência e relevância. No entanto, devido às limitações de tempo, não foi possível realizar tal validação. Portanto, foi necessário optar por outras formas de verificação e validação dos experimentos.

Com o intuito de validar os experimentos propostos, o autor deste trabalho, usou o Laboratório LDS da Faculdade UnB Gama para executar os ensaios técnicos e avaliar a consistência dos procedimentos de configuração propostos. A utilização destes recursos permitiu a verificação dos resultados obtidos na Seção 5.2 e a confirmação da eficiência dos experimentos propostos.

5.4.1 Tempo Gasto na Execução dos Experimentos

Para avaliar o tempo gasto na execução de cada experimento, foram registrados os tempos de início e término, e posteriormente calculado o tempo total gasto em cada um deles. Verificou-se que o tempo médio para a execução do primeiro experimento foi de aproximadamente 30 minutos, enquanto que para o último experimento, esse tempo aumentou para cerca de 2 horas e 30 minutos. Isso pode ser atribuído ao fato de que, conforme os experimentos avançavam, as tarefas se tornavam mais complexas e exigiam mais tempo para serem concluídas. Além disso, a necessidade de configurar e operar os diversos equipamentos de rede também contribuiu para o aumento do tempo gasto. Portanto, pode-se concluir que a progressão de dificuldade dos experimentos práticos está diretamente relacionada ao tempo necessário para a execução e conclusão das atividades propostas.

5.4.2 Dificuldades Enfrentadas

Durante a implementação deste trabalho, foram enfrentadas algumas dificuldades que impactaram no desenvolvimento e no tempo necessário para conclusão, como: a menor quantidade disponível de referências que abordam os temas no sistema operacional FreeBSD, fazendo com que fosse gasto um esforço considerado na pesquisa e na compreensão e utilização dos recursos disponíveis no sistema operacional; lidar com a grande quantidade de detalhes considerados no desenvolvimento e validação dos experimentos, tornando um processo que requeria muita atenção, esforço e tempo; dificuldade para fixar

o entendimento de alguns protocolos utilizados, o que dificultou a montagem das topologias de redes necessárias para a realização de cada experimento; e lidar com o aprendizado de fundamentos básicos de redes, por falta de conhecimento sólido anterior.

Na parte de execução e validação do presente trabalho, também foram enfrentadas algumas dificuldades que podem ser mitigadas em execuções futuras, por exemplo, a falta de equipamentos necessários ou a presença de equipamentos danificados no laboratório, como, cabos de redes e *switches*. Cabos de redes são fundamentais para a configuração do laboratório, pois são eles que fazem a ligação entre as máquinas. Com a ausência de uma quantidade suficiente de cabos, não é possível configurar a topologia desejada nos experimentos. *Switches* são de extrema importância para a realização dos experimentos, tendo em vista que as máquinas presentes no laboratório possuem no máximo 3 interfaces de rede, o que não permite a configuração de certas topologias indicadas.

5.4.3 Instruções para Futuras Execuções

O laboratório LDS possui um segmento de rede previamente configurado com algumas máquinas que utilizam o sistema operacional Linux (Ubuntu) e com saída para a internet. Para poupar tempo e esforços, é interessante fazer uso desse segmento de rede para *download* de eventuais ferramentas necessárias e para executar validações em experimentos que a topologia exige dois segmentos de redes, onde a rede externa, representada por essa rede previamente configurada, precisa se comunicar com a rede interna que foi configurada no experimento.

No momento da configuração dos equipamentos presentes no laboratório, faz-se necessário a instalação e configuração dos sistemas operacionais que serão utilizados. Este procedimento, junto com a montagem da topologia indicada em cada experimento, pode ser um tanto quanto demorado. Por isso, é indicado que o instrutor do laboratório já tenha em mente a topologia e os materiais que serão utilizados no experimento que será aplicado e tenha o sistema operacional previamente configurado, para *boot*, em alguma unidade de gravação externa, para que não seja desperdiçado tempo tendo que fazer a configuração da mídia com o sistema operacional, só depois, poder configurar a máquina. Além disso, as máquinas podem possuir um sistema de BIOS antigo (*CMOS Setup Utility*), então é importante que o instrutor do laboratório saiba configurar este tipo de BIOS para inicializar o sistema operacional a partir de uma unidade de gravação externa.

5.4.4 Possíveis Erros Esperados

Na execução dos experimentos, alguns erros podem ser comuns e esperados, dentre eles, estão, os erros de incompatibilidade de comandos e ferramentas com a versão do sistema operacional utilizado, a errada configuração da topologia de rede indicada

no experimento, a perda de referência ao servidor DNS ao configurar uma máquina de forma estática e a não habilitação do encaminhamento de pacotes em experimentos que é necessário habilitar o roteamento.

5.4.5 Ambiente Virtualizado

Executar os experimentos em ambiente virtualizado pode ser uma opção viável, pois oferece uma série de vantagens, como a flexibilidade de criação de diferentes topologias de rede, a facilidade de teste e correção de erros e a possibilidade de simular ambientes reais sem afetar a rede física. Além disso, a virtualização também pode ser uma solução mais econômica, pois dispensa o uso de equipamentos físicos e pode ser executada em qualquer ambiente, como em laboratórios ou em computadores pessoais. No entanto, é importante considerar que o uso da virtualização depende da finalidade e objetivos dos experimentos, bem como da capacidade da plataforma virtualizada de simular corretamente as condições reais.

5.5 Acesso a Implementação da Solução

A solução foi construída com o uso de ferramentas gratuitas, e o código fonte utilizado na criação da página web que hospeda os experimentos práticos é regido sob a licença MIT, o que significa que pode ser utilizado gratuitamente.

Os arquivos fonte desta solução podem ser encontrados no repositório: <https://github.com/markinlimac/monografia-redes>.

O repositório contém um arquivo “README.md“ que explica como implementar a solução em outros repositórios. Isso permite que o instrutor faça modificações de acordo com sua preferência e personalize a solução de acordo com suas necessidades.

Para dúvidas a cerca da implementação, entre em contato no email markinlimac@gmail.com ou criando uma Issue no repositório do Github citado anteriormente.

6 Conclusão

Este capítulo tem por finalidade apresentar as considerações finais das atividades que envolvem este Trabalho de Conclusão de Curso. Inicialmente, a Seção 6.1 apresenta o andamento do trabalho, desde a Introdução até o desenvolvimento do conjunto de experimentos práticos e a análise de Resultados, e mostram-se as atividades relacionadas à etapa final do trabalho. Além disso, a Seção 6.2 discute os resultados obtidos para com os objetivos traçados no início deste trabalho. Por fim, a Seção 6.3 apresenta as observações levantadas para trabalhos futuros.

6.1 Andamento do Trabalho

A primeira fase deste trabalho concentrou-se na criação das tarefas que fundamentam e organizam a proposta de projeto em relação ao conjunto de ensaios didáticos para a experimentação de fundamentos de redes de computadores. Tendo em vista os Cronogramas de Atividades estabelecidos para este Trabalho de Conclusão de Curso, pode-se conferir na Tabela 5 o progresso das atividades da etapa inicial deste trabalho.

Tabela 5 – Andamento das Atividades da Fase Inicial

Atividade	**Andamento**
Definição de Tema	Concluída
Levantamento Bibliográfico	Concluída
Formular Introdução da Proposta	Concluída
Definir Referencial Teórico	Concluída
Definir Metodologias	Concluída
Documentar Proposta	Concluída
Formular Resumo do trabalho	Concluída
Apresentar à Banca	Concluída
Aplicar Correções Solicitadas pela Banca	Concluída

Fonte: Autor.

Na segunda fase deste trabalho, o foco foi no desenvolvimento dos experimentos propostos, de modo a colocar em prática as definições alcançadas. A Tabela 6 apresenta o andamento das atividades relacionadas à etapa final deste Trabalho de Conclusão de Curso.

6.2 Resultados Obtidos

Retomando os objetivos específicos, apresentados no Capítulo 1, descritos como:

Tabela 6 – Andamento das Atividades da Fase Final

Atividade	**Andamento**
Desenvolvimento do conjunto de experimentos	Concluída
Análise de Resultados	Concluída
Apresentar à Banca	A Fazer
Aplicar Correções Solicitadas pela Banca	A fazer

Fonte: Autor.

1. Desenvolver um conjunto de experimentos práticos para configuração dos principais protocolos de rede. Resultados: Conjunto de experimentos práticos orientados pela metodologia estabelecida no Capítulo 4, apresentados no Capítulo 5.

2. Demonstrar a viabilidade do uso das atividades práticas na disciplina "fundamentos de redes de computadores". Resultados: Conjunto de ferramentas e experimentos descritos na Seção 5.4 para que os alunos desenvolvam habilidades essenciais para aprofundar seu conhecimento e aplicar as habilidades adquiridas em um ambiente real.

3. Apresentar um sistema operacional diferente para agregar conhecimento aos alunos. Resultado: Uso do sistema operacional FreeBSD, apresentado na Seção 2.1.1, nos experimentos práticos.

4. Apresentar, aos discentes de engenharia de software, um conjunto de ferramentas para serem usadas na resolução de problemas referentes a rede de computadores. Resultados: Conjunto de ferramentas úteis para resolver problemas de rede de computadores, apresentadas no roteiro dos experimentos práticos.

Com base no Capítulo 2, foi possível desenvolver o conjunto de experimentos práticos para operação básica de uma rede de computadores descrito na Seção 1.2.1 e alcançar todos os objetivos descritos na Seção 1.2.2.

6.3 Trabalhos Futuros

Com base no que foi observado no decorrer das experimentações, são apontados os seguintes trabalhos futuros:

- Para uma melhor precisão e confiabilidade dos resultados obtidos sugere-se que a validação dos experimentos seja feita com uma turma, da disciplina Fundamentos de Redes de Computadores, a fim de garantir ainda mais a eficiência e a precisão dos resultados obtidos.

- Utilização de ambientes virtualizados para a execução dos experimentos, a fim de obter maior facilidade na execução dos testes e a criação de ambientes de teste mais complexos.
- Adição de um experimento que diz respeito à crescente área de IoT (Internet of Things), para coleta e análise de dados destes dispositivos.
- Adição de experimentos que abordem segurança da informação e os ataques relacionados aos protocolos comumente usados. Pode-se, inclusive, adicionar experimentos que implementem a segurança de dispositivos IoT na rede.

Referências

ALANI, M. M. Guide to osi and tcp/ip models. Springer, 2014. Citado 9 vezes nas páginas 15, 17, 29, 30, 31, 32, 33, 37 e 38.

AWS, A. *O que é o DNS?* 2023. Disponível em: <https://aws.amazon.com/pt/route53/what-is-dns/>. Citado na página 39.

CVE. *Freebsd : Products and vulnerabilities*. 2022. Disponível em: <https://www.cvedetails.com/vendor/6/Freebsd.html>. Citado na página 24.

DEBIAN.ORG. *Razões para escolher o Debian*. 2022. Disponível em: <https://www.debian.org/intro/why_debian>. Citado na página 28.

DELGADO, S. C. *FreeBSD Documentation*. 2022. Disponível em: <https://docs.freebsd.org/en/books/handbook/introduction/>. Citado 2 vezes nas páginas 24 e 27.

FERREIRA, K. et al. Inserindo um laboratório virtual para o ensino de redes de computadores. In: *ICBL2013–International Conference on Interactive Computer aided Blended Learning, Florianópolis*. [S.l.: s.n.], 2013. Citado na página 23.

FIELDING, R. et al. *Hypertext transfer protocol–HTTP/1.1*. [S.l.], 1999. Citado na página 38.

FOROUZAN, B. A. *Comunicação de dados e redes de computadores*. [S.l.]: AMGH Editora, 2009. Citado na página 23.

HANSEN, P. B. *Operating system principles*. [S.l.]: Prentice-Hall, Inc., 1973. Citado na página 27.

HASSAN, E. B. Laboratório virtual 3d para ensino de redes de computadores. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. [S.l.: s.n.], 2003. v. 1, n. 1, p. 654–663. Citado na página 23.

HIJAZI, S.; OBAIDAT, M. S. Address resolution protocol spoofing attacks and security approaches: A survey. *Security and Privacy*, Wiley Online Library, v. 2, n. 1, p. e49, 2019. Citado na página 32.

HUITEMA, C. *IPv6: the new Internet protocol*. [S.l.]: Prentice-Hall, Inc., 1995. Citado na página 34.

INC, K. *History of FreeBSD – part 4: BSD and TCP/IP*. 2021. Disponível em: <https://klarasystems.com/articles/history-of-freebsd-part-4-bsd-and-tcp-ip/>. Citado na página 24.

KIZZA, J. M. *Computer network security*. [S.l.]: Springer, 2005. Citado na página 23.

KOZIEROK, C. M. *The TCP/IP guide: a comprehensive, illustrated Internet protocols reference*. [S.l.]: No Starch Press, 2005. Citado na página 33.

KUMAR, S.; RAI, S. Survey on transport layer protocols: Tcp & udp. *International Journal of Computer Applications*, Citeseer, v. 46, n. 7, p. 20–25, 2012. Citado na página 38.

KUROSE, J.; ROSS, K. Computer networks and the internet. *Computer networking: A Top-down approach. 7th ed. London: Pearson*, 2016. Citado na página 31.

LUCAS, M. W. *Networking for Systems Administrators*. 5th. ed. USA: Tilted Windmill Press, 2019. Citado na página 24.

MANDEL, A.; SIMON, I.; DELYRA, J. L. Informação: Computação e comunicação. *Revista USP*, n. 35, p. 10–45, nov. 1997. Disponível em: <https://www.revistas.usp.br/revusp/article/view/26865>. Citado na página 27.

NOONAN, W.; DUBRAWSKY, I. *Firewall fundamentals*. [S.l.]: Pearson Education, 2006. Citado na página 37.

PAIVA, M. R. F. et al. Metodologias ativas de ensino-aprendizagem: revisão integrativa. *SANARE-Revista de Políticas Públicas*, v. 15, n. 2, 2016. Citado na página 25.

POSTEL, J. *Simple mail transfer protocol*. [S.l.], 1982. Citado na página 39.

SHARMA, A. *Network Address Translation - Port Address Translation*. 2014. Disponível em: <http://ip-mpls.com/ccna/network-address-translation-port-address-translation/>. Citado 2 vezes nas páginas 15 e 36.

TANENBAUM, A. S. *Redes de computadores*. [S.l.]: Campus, 2003. Citado 6 vezes nas páginas 28, 29, 31, 33, 34 e 39.

TANENBAUM, A. S.; STEEN, M. V. *Sistemas Distribuıdos: princıpios e paradigmas*. [S.l.]: Sao Paulo: Pearson Prentice Hall, 2007. Citado 3 vezes nas páginas 15, 28 e 29.

WING, D. Network address translation: Extending the internet address space. *IEEE internet computing*, IEEE, v. 14, n. 4, p. 66–70, 2010. Citado 2 vezes nas páginas 35 e 36.

WOUNDY, R.; KINNEAR, K. *Dynamic host configuration protocol (DHCP) leasequery*. [S.l.], 2006. Citado 2 vezes nas páginas 36 e 37.

ZYDYK, M. *Address Resolution Protocol (ARP)*. 2021. Disponível em: <https://www.techtarget.com/searchnetworking/definition/Address-Resolution-Protocol-ARP>. Citado na página 33.