UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
CURSO DE ENGENHARIA DE REDES DE COMUNICAÇÃO

Gabriel Inácio Machado Trindade

ESTUDO DE IMPLEMENTAÇÃO DE UMA INFRAESTRUTURA SEGURA DE NUVEM PRIVADA

Brasília
2023

Gabriel Inácio Machado Trindade

ESTUDO DE IMPLEMENTAÇÃO DE UMA INFRAESTRUTURA SEGURA DE NUVEM PRIVADA

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de Bacharel em Engenharia de Redes de Comunicação e aprovado em sua forma final pelo Departamento de Engenharia Elétrica pela Universidade de Brasília.
Orientador: Prof. Dr. Georges Daniel Amvame Nze

Brasília
2023

Gabriel Inácio Machado Trindade

ESTUDO DE IMPLEMENTAÇÃO DE UMA INFRAESTRUTURA SEGURA DE NUVEM PRIVADA

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de Bacharel em Engenharia de Rede de Comunicação e aprovado em sua forma final pelo Departamento de Engenharia Elétrica pela Universidade de Brasília.

Brasília, 10 de Fevereiro de 2023.

Banca Examinadora:

Prof. Georges Daniel Amvame Nze, Dr.
EnE/UnB - Orientador

Prof. Fábio Lúcio Lopes de Mendonça, Dr.
EnE/UnB - Avaliador Interno

Prof. Welber Santos de Oliveira, Esp.
Estácio/Brasília - Avaliador Externo

AGRADECIMENTOS

A minha família, meus pais, Adailton e Monica, e aos meus irmãos, Juliana e João Victor, por todo os conselhos, incentivos, apoio e acreditarem em meu potencial.

A minha namorada, Dáfynne Mello, que me acompanha desde o ensino médio e que é a minha maior parceira. Agradeço por todo o carinho durante todos esses anos, estar sempre presente em minhas conquistas e me aconselhar nas minhas frustrações.

Aos meus familiares e amigos que torcem por mim, agradeço por todo apoio durante esse período de graduação.

Ao meu orientador, Prof. Dr. Georges Daniel, por toda a orientação e disposição de como se decorreu a jornada singular durante o Projeto Final de Graduação.

Por fim, ao corpo docente da UnB, especialmente para ao departamento de Engenharia Elétrica, por todo o ensinamento durante esses anos de graduação.

"Privacidade não é sobre ter algo a esconder. É sobre ter algo para proteger. E esse algo é quem você é. É algo em que você acredita. É quem você quer se tornar. Privacidade é o direito de si mesmo. É o que lhe permite compartilhar com o mundo quem você é nos seus próprios termos."

- Edward Snowden

RESUMO

Este trabalho tem como objetivo implementar uma infraestrutura de nuvem privada segura utilizando isolamento de rede por meio de um firewall, inspeção profunda de pacotes e configuração de uma conexão segura. Baseando-se em trabalhos que adotaram a implementação de uma plataforma de nuvem, foi obtido o sucesso em identificar vulnerabilidades existentes em um ambiente de nuvem e adotar estratégias para mitigar a exploração de vulnerabilidades por parte de um usuário malicioso. Aspira-se que esse trabalho possa servir de base para estudos futuros que envolvam a implementação de um ambiente de rede computacional seguro, pois a infraestrutura aqui montada possui potencial para ser escalada para um nível de ambiente real.

Palavras-chave: Nuvem privada; Segurança em nuvem; Firewall; IDS/IPS; VPN.

ABSTRACT

This paper aims to implement a secure private cloud infrastructure using network isolation through a firewall, deep packet inspection and configuration of a secure connection. Based on papers that adopted the implementation of a cloud platform, success was achieved in identifying existing vulnerabilities in a cloud environment and adopting strategies to mitigate the exploitation of vulnerabilities by a malicious user. It is hoped that this work can serve as a basis for future studies involving the implementation of a secure computing network environment, as the infrastructure assembled here has the potential to be scaled to a real environment level.

Keywords: Private cloud; Cloud security; Firewall; IDS/IPS; VPN.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

API	*Application Programming Interface*
ARM	*Advanced RISC Machine*
AWS	*Amazon Web Services*
CIA	*Confidentiality, Integrity and Availability*
CID	Confidencialidade, Integridade e Disponibilidade
CPU	*Central Processing Unit*
DHCP	*Dynamic Host Configuration Protocol*
DMZ	*Demilitarized Zone*
DNS	*Domain Name System*
DoS	*Denial of Service*
FTP	*File Transfer Protocol*
GNS3	*Graphical Network Simulator-3*
HTTP	*Hypertext Transfer Protocol*
IaaS	*Infrastrucuture as a Service*
ICMP	*Internet Control Message Protocol*
IDS	*Intrusion Detection System*
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
IP	*Internet Protocol*
IPS	*Intrusion Prevention System*
ISO	*Internacional Organization for Standards*
LAN	*Local Area Network*
LTS	*Long-term support*
MAC	*Media Access Control*
NAT	*Network Address Translation*
NIC	*Network Interface Controller*
NIST	*National Institute of Standards and Technology*
OSI	*Open Systems Interconnection*
OSPF	*Open Shortest Path First*
PaaS	*Platform as a Service*
PC	*Personal Computer*
RAM	*Random Access Memory*
SaaS	*Software as a Service*
SLA	*Service Level Agreement*
SO	Sistema Operacional
SPI	*Software-Platform-Infrastructure*
SSH	*Secure Socket Shell*
TCP	*Transmission Control Protocol*

TI	Tecnologia da Informação
TLS	*Transport Layer Security*
UDP	*User Datagram Protocol*
UTM	*Unified Threat Management*
VLAN	*Virtual Local Area Network*
VPN	*Virtual Private Network*
WAN	*Wide Area Network*

SUMÁRIO

1 INTRODUÇÃO

Este projeto final de graduação tem como finalidade implementar um sistema capaz de prover um serviço de nuvem privada segura para um usuário que se encontra fora do perímetro da rede de uma empresa. Com esse propósito, será utilizado ferramentas e estratégias para mitigar a superfície de ataque de um possível invasor.

Para dar isolamento a rede onde ficará hospedado o serviço de nuvem, será implementada uma rede DMZ (*Demilitarized Zone*), onde todo tráfego com destino a essa rede passará pelo firewall e um sistema de detecção e prevenção de intrusão que realizarão uma inspeção profunda desse pacotes recebidos. Após essa inspeção o firewall e IDS/IPS irão agir de acordo com sua configuração.

1.1 PROBLEMA

Com o estabelecimento da Internet e o avanços das tecnologias relacionadas a computação em nuvem, cada vez mais empresas e usuários finais encontram-se consumindo serviços fornecidos por nuvens computacionais. Porém, como ocorreu um uso crescente por parte dos usuário e empresas, aumentou o número de pessoas mal-intencionadas que tentam obter acesso, informações e dados dos consumidores desses serviços. Por um lado a modernização das tecnologias facilitou o trabalho na rede, mas por outro lado também modernizou os ataques que uma rede pode sofrer. A ferramentas computacionais de ataque tiveram uma evolução, tornando-as mais acessíveis, automatizando descobertas de vulnerabilidades e sofisticou as suas sintaxes.

Com a finalidade de combater essas crescentes ameaças virtuais, as ferramentas utilizadas para a segurança cibernética tiveram que acompanhar os avanços dos cibercriminosos para melhorar a segurança dos sistemas computacionais e redes. Ferramentas precisaram ser adotadas, como o uso de criptografia para velar o conteúdo das mensagens trocadas entre entidades, Firewalls que são utilizados para gerenciar o tráfego de um rede e isolá-la, VPNs que garantem a comunicação segura de informações sensíveis por um meio inseguro e sistemas de detecção e prevenção de intrusão que são utilizados para reconhecer pacotes/conexões legítimos de usuário e redes por meio de uma inspeção profunda.

Deste modo, devido a sofisticação gradativa e a frequência crescente dos ataques cibernéticos, decidiu-se criar uma arquitetura de infraestrutura que seja capaz de analisar o tráfego em rede em tempo real, sendo realizada uma filtragem dos pacotes com destino a rede DMZ e configurar os dispositivos responsáveis pela segurança a tomarem a melhor decisão com relação a pacotes mal-intencionados.

1.2 OBJETIVOS

Nas seções abaixo estão descritos o objetivo geral e os objetivos específicos deste trabalho de conclusão de curso.

1.2.1 Objetivo Geral

Este projeto final de graduação tem como objetivo final a implementação de uma infraestrutura de rede que seja capaz de diminuir as possibilidades de exploração de vulnerabilidades em um ambiente de nuvem privada, por meio da filtragem e inspeção profunda de pacotes. Tornando um sistema eficaz, escalável e robusto com a capacidade gerar informações em tempo real para um administrador de rede e de fornecer segurança para os usuário e para a empresa na qual está fornecendo o serviço de computação em nuvem.

1.2.2 Objetivos específicos

Para que o trabalho obtivesse sucesso, foi-se necessário cumprir alguns objetivos específicos, como criar uma topologia que fosse capaz de suportar e gerar informações reais sobre os ataques cibernéticos, implementar um serviço de nuvem privada que fosse capaz de ser acessível por um usuário externo a rede, realizar configurações com base nas análises dos pacotes, realizar simulação de ataque para testar o sistema e implantar uma estratégia capaz de mitigar a superfície de ataque do sistema implantado.

1.3 ORGANIZAÇÃO DO TRABALHO

A partir desse capítulo o trabalho está dividido em 5 partes. O capítulo 2 tratará do referencial teórico, o capítulo 3 apresentará a fundamentação teórica, o capítulo 4 abordará a metodologia, o capítulo 5 mostrará os resultados e as análises e o capítulo 6 abordará as conclusões do trabalho.

2 REVISÃO BIBLIOGRÁFICA

Este capítulo será apresentado uma revisão sistemática da literatura realizada sobre o tema Implementação de Infraestrutura Segura de Nuvem Privada, com o foco em identificar respostas para as questões propostas além de apresentar trabalhos relevantes já publicados com aplicações em ambientes de experimentação, tanto em cenários reais como em ambientes simulados.

Este capítulo está organizado da seguinte forma: a Seção 2.1 apresenta qual foi o método da revisão aplicada no trabalho e sua etapas, sendo divido na Seção 2.1.1 no define as questões da pesquisa e Seção 2.1.2 que define a estratégia de busca e seleção dos trabalhos relacionados. Na Seção 2.2, são apresentados os trabalhos relacionados escolhidos que contribuíram para a tomada de decisão durante o trabalho.

2.1 REVISÃO SISTEMÁTICA DOS TRABALHOS RELACIONADOS

A revisão sistemática é um tipo de revisão que se propõe a responder um pergunta específica de forma objetiva e imparcial. Para isso utiliza métodos sistemáticos e definidos a priori na identificação e seleção dos estudos, extração dos dados e análise dos resultados para mapear o estado da arte em relação a formas da implementação da infraestrutura de nuvem e suas possível complicações. Esse mapeamento foi composto por duas etapas, que serão detalhados nas subseções seguintes

1. Definição de questões da pesquisa;
2. Busca e seleção dos estudos relevantes;

2.1.1 Questões da pesquisa

Neste mapeamento se definiu as questões de pesquisa que guiou a condução do estudo:

- **1^o questão:** Como ocorre a implementação de um serviço de nuvem de forma segura?
- **2^o questão:** Quais são os tipos de ataques mais comuns em um serviço nuvem?
- **3^o questão:** Quais são as implicações em uma implementação de um serviço de baseado nuvem?

2.1.2 Estratégia de busca e seleção

As buscas ocorreram de forma manual e tomou como referência três métodos: a fonte de da busca, os idiomas desejados e palavras-chave na busca.

- **Fontes de busca:** bases de dados na área da engenharia e computação: IEEE Xplore - IEEE (Instituto de Engenheiros Eletricistas e Eletrônicos), Google

Scholar, Science Direct, busca por trabalhos de conclusão de curso, mestrados e doutorados em acervos onlines de universidades federais.

- **Idiomas dos trabalhos:** português, para acessar publicações nacionais, além de ser o idioma nativo e inglês, por ser o idioma mundialmente aceito para trabalhos científicos
- **Palavras-chave:** em português "*segurança em nuvem*", "*infraestrutura de nuvem*", *"computação em nuvem"*, *"plataformas de infraestrutura como serviço"* e "*implementação*". Em inglês "*cloud security*", "*cloud security issues*", *"infrastruture network"* e *"cloud"*.

2.2 TRABALHOS RELACIONADOS

Primeiramente, foi buscado artigos e livros que definissem os conceitos que serão abordados. Depois, foi buscado exemplos de trabalhos relacionados com a proposta de implementação de um serviço em nuvem. Por fim, foi encontrado artigos que levantam questões pertinentes sobre a implementação de um serviço em nuvem e a migração de um serviço para a nuvem, seus riscos e soluções.

Com o objetivo de buscar a definição sobre o que é computação de nuvem, seus modelos de serviço, características essenciais e modelos de implantação. O Instituto Nacional de Padrões e Tecnologia, conhecido como NIST (*National Institute of Standards and Technology*), em 2011 publicou sua definição sobre computação em nuvem (NIST, 2011). Krutz e Vines (KRUTZ; VINES, 2010) também contribuíram com um livro, que além de definir computação em nuvem, define a arquitetura, risco e desafios de uma segurança em nuvem computacional.

Posteriormente, foi buscado trabalhos que implementaram serviços em nuvem. Em 2016, Rafael César Merlo dos Santos (SANTOS, R. C. M. dos, 2016) realizou a implantação da plataforma de código aberto *CloudStack* no Laboratório de Bioinformática e Dados (LaBiD) de maneira a disponibilizar uma nuvem privada de infraestrutura para suportar os projetos. Como, segundo o autor, na época a Universidade de Brasília possuía nenhuma infraestrutura de nuvem privada, foi necessário montar sua própria infraestrutura para realização de seu trabalho, como pode ser visto na Figura 1. Foi necessário realizar o levantamento das máquinas disponíveis para instalação da plataforma *CloudStack*, como capacidade de processamento, capacidade de memória de disco e RAM. O autor apresentou detalhes da escolha da plataforma de infraestrutura como serviço, da implantação da nuvem privada, em passo a passo, mostrou os testes realizados e seus resultados obtidos.

Lúcio da Silva Gama Júnior (SILVA GAMA JÚNIOR, 2017) discorreu em seu mestrado sobre Virtualização de Redes, Computação em Nuvem e Virtualização de Funções de Rede, sendo um mestrado focado em virtualização. Neste, o autor utilizou a plataforma de infraestrutura como serviço chamada OpenStack, realizou o levantamento das máquinas

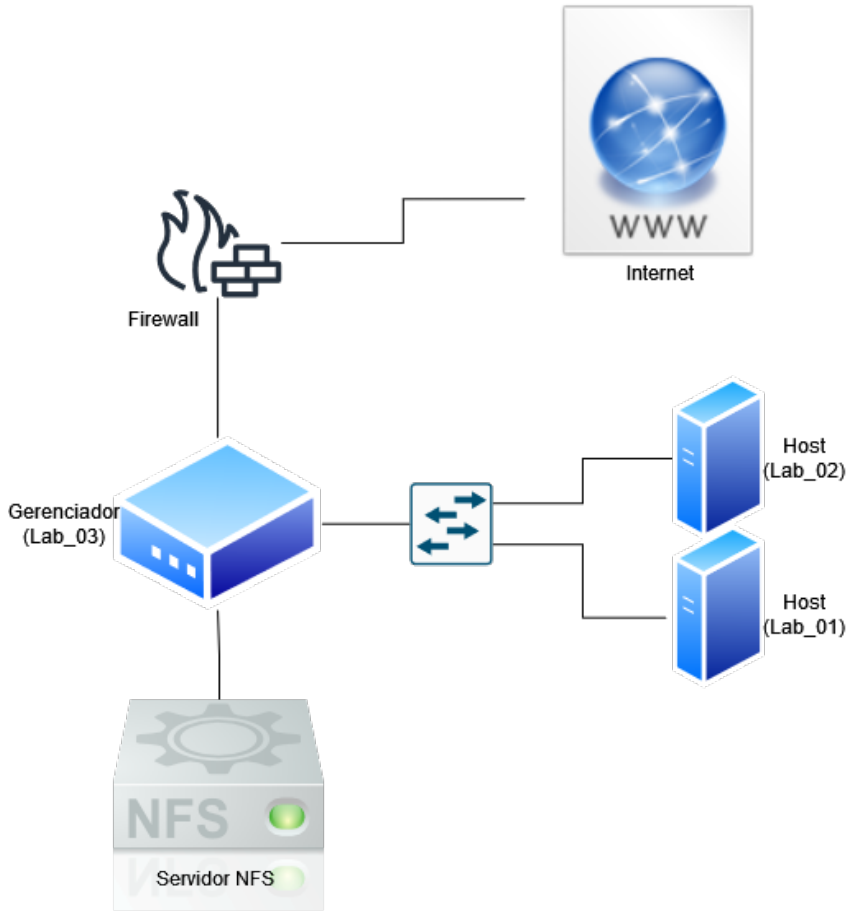


Figura 1 – Arquitetura utilizada para implementação da nuvem privada, adaptado de (SANTOS, R. C. M. dos, 2016, p. 37)

(processador, memória, capacidade e interfaces de rede disponíveis) e SO e definiu a maquina controladora, maquinas de computação e máquina de armazenamento, seguindo a documentação do OpenStack.

Kennedy Bezerra da Silva Linz (SILVA LINZ, 2020), em 2020, teve como objetivo realizar um estudo sobre a implementação de uma nuvem híbrida na Agência Estadual de Meio Ambiente de Pernambuco. O autor escolheu por usar o provedor AWS (*Amazon Web Services*) para realizar a implementação, detalhando todo planejamento da implementação.

Molina, Silveira e Santos em seu artigo (MOLINA; SILVEIRA; SANTOS, F. V. dos, 2015) realizaram um estudo de caso envolvendo a implantação de um ambiente seguro na rede de computadores da Prefeitura Municipal de Palmeira das Missões. Definiram uma infraestrutura física e lógica que por meio da criação de VLANs e definição da DMZ teve como objetivo atingir um nível de segurança e gerência de redes proporcionando uma maior confiabilidade e integridade das informações que trafegam na rede.

Por fim, com a finalidade de ter conhecimento sobre algumas questões relacionado a segurança na nuvem, possíveis riscos e suas soluções, foi buscado artigos tratam sobre esse assunto. Em artigo foi mostrado o acordo entre as partes, SLA (*Service Level Agreement*),

que definiu como:

> "Um acordo de nível de serviço é um documento que define o relacionamento entre duas partes: o provedor e o cliente. Este é claramente um item extremamente importante da documentação para ambas as partes. Se usado corretamente deve: identificar e definir as necessidades do cliente; fornecer em estrutura para compreensão; simplificar questões complexas, reduzir áreas de conflito; incentivar o diálogo em caso de disputas; e eliminar expectativas irrealistas"(KANDUKURI; PATURI; RAKSHIT, 2009, p. 1)

No qual é mostrado qual é o conteúdo típico do SLA, como sendo: definição do(s) serviço(s); gestão de desempenho; gerenciamento de problemas; deveres e responsabilidades do cliente; garantias e recursos; segurança; recuperação de desastres e continuidade de negócios; e terminação.

Dependo da escolha do modo de operação, diferentes vantagens e desvantagens irão aparecer no cenário, na qual Sarmah "discute as várias soluções propostas, suas vantagens e desvantagens que ajudaria a optar pela solução de de nuvem mais adequada"(SARMAH, 2019, p. 1). Sarmah lista, também, os desafios e riscos de segurança da nuvem sendo eles a responsabilidade compartilhada sendo difícil de gerenciar, a incerteza dos dados e os riscos relacionados ao uso de recursos compartilhados. Ele finaliza listando algumas soluções de mitigação de riscos e soluções oferecidos para proteger a nuvem:

- Mecanismos de segurança para provedores:
  - Padrões de segurança
  - Auditorias e Certificações
  - Padrão ISO2700x
- Soluções de segurança para clientes:
  - Localização de dados
  - Proteção de dados
  - Autenticação de usuário
  - Autorização e Controle de Acesso
  - Rastreabilidade

No fim, Sarmah, afirma que deve-se

> "...considerar uma transição para a nuvem de forma gradual, começando com tratamentos de baixo risco e realizando análises de risco que abrangem o ambiente técnico, jurídico e de negócios para realizar essa transição com toda a segurança para ambas as organizações clientes. e pessoas cujos dados estão sendo migrados para a nuvem."(SARMAH, 2019, p. 5)

É isso que Rosado *et al* (ROSADO *et al.*, 2012) faz uma análise da segurança na migração para ambientes em nuvem. Neste artigo, começa pontuando os benefícios e os desafios da segurança na computação em nuvem e os problemas de segurança em nuvem dependendo do modelo de implantação (nuvem pública, privada e híbrida). Por

fim, faz uma análise das abordagens de migração para a nuvem, como migração baseada em modelo de sistemas de software legado para a nuvem; migração de aplicativos legados para o serviço de nuvem; reutilização e migração de aplicativos legados para serviços de nuvem interoperáveis; benchmark de criptografia transparente de dados para migração de aplicativos da Web na nuvem; realizar um estudo de caso de migração de um sistema corporativo de TI para IaaS; utilizar ferramentas de suporte à decisão para migração para a nuvem em uma empresa; migrar um serviço para uma arquitetura de nuvem; e utilizar um serviço dinâmico e migração de dados nas nuvens.

3 FUNDAMENTOS TEÓRICOS

O propósito deste capítulo é conceituar assuntos ligados ao paradigma da computação distribuída, sendo seu maior foco em computação em nuvem, já que o objetivo central deste trabalho é a implementação de uma infraestrutura segura de nuvem privada. Além de dar um panorama geral sobre as camadas da rede, seu funcionamento e seus protocolos e, por fim, mostrar alguns dos possíveis ataques maliciosos que ocorrem em redes de computadores.

A Seção 3.1 abordará os conceitos relacionados aos sistemas distribuídos. A Seção 3.2 definirá as características e os modelos de serviço e implantação em uma nuvem computacional. A Seção 3.3 mostrará os riscos e responsabilidades que consumidores e provedores compartilham. A Seção 3.4 apresentará o modelo em camadas da arquitetura de redes de computadores e seus protocolos. A Seção 3.5 abordará mecanismos e estratégias de como deixar uma rede de computadores mais segura. E, por fim, a Seção 3.6 mostrará alguns possíveis ataques que ocorrem em um ambiente de nuvem computacional.

3.1 SISTEMA DISTRIBUÍDO

Existem várias definições de sistemas distribuídos na literatura, porém, mesmo havendo diferenças entre suas definições, é possível encontrar semelhanças em alguns pontos sobre sua definição. Coulouris define um sistemas distribuído como sendo "aquele no qual os componentes localizados em computadores interligados em rede se comunicam e coordenam suas ações apenas passando mensagem"(COULOURIS, 2007, p. 15). Já Tanenbaum define como "um conjunto de computadores independentes que se apresenta a seus usuários como um sistema único e coerente"(TANENBAUM, 2007, p. 1). Isso nos leva ao conceito chave sobre sistemas distribuído: a transparência ao usuário. O uso de múltiplos computadores e processadores deve ser invisível ao usuário. Sendo assim, uma definição que abrange os conceitos é que um sistema distribuído é uma coleção de computadores independente ligados por uma rede, que do ponto de vista do usuário, o sistema se comporta como um único computador.

3.1.1 Características de um sistema distribuído

O compartilhamento de recursos é o principal fator de motivação para a construção de sistemas distribuídos, porém existem outras características inerentes ao sistema distribuído que são: heterogeneidade, abertura, concorrência, escalabilidade, tolerância a falhas, segurança e transparência.

- Heterogeneidade: os sistemas distribuídos "devem ser construídos a partir de uma variedade de redes, sistemas operacionais, hardware e linguagem de programação diferentes"(COULOURIS, 2007, p. 36). Para resolver esse problema,

é criado uma camada *middleware* para funcionar como um software intermediador, no qual diminui a complexidade e a heterogeneidade dos diversos sistemas existentes e fornece a comunicação entre as aplicações de forma transparente às mesmas.

- Abertura: define que os sistemas distribuídos devem ser extensíveis e, esse grau de abertura, pode ser determinado medindo-se a possibilidade de acrescentar novos serviços ou recursos compartilhados sem causar problemas para os já existentes.
- Concorrência: em um sistema distribuídos com múltiplos acessos simultâneos gera uma concorrência para seus recursos e "em um ambiente concorrente, cada recurso deve ser projetado para manter a consistência nos estados e seus dados"(COULOURIS, 2007, p. 36).
- Escalabilidade: capacidade que o sistema deve possuir de aumentar significativamente o número de recursos e/ou o número de usuários sem perder desempenho.
- Tolerância a falhas: é ter a consciência que nenhum sistema é imune a falhas, porém é necessário ter uma tolerância e conter os efeitos que uma falha pode gerar de uma maneira que o sistema não pare de funcionar e mascare essas falhas de seus usuários.
- Segurança: criptografias podem ser usadas de maneira a proporcionar uma proteção adequada para os recursos compartilhados e manter informações sigilosas em segredo quando são transmitidas pela rede. Isso não resolve ataques de negação de serviços.
- Transparência: como dito anteriormente, é a característica mais importante dos sistemas distribuídos. Seu "objetivo é tornar certos aspectos da distribuição invisíveis para o programador de aplicativos, para que este se preocupe apenas com o projeto de seu aplicativo em particular"(COULOURIS, 2007, p. 36). Ou seja, um sistema, apesar de distribuído, deve parecer ao usuário como uma única máquina.

Com isso, essas características definidas sobre o modelo de um sistema distribuído, é possível definir o que é computação em nuvem.

3.2 COMPUTAÇÃO EM NUVEM

Assim como a definição sobre o que um sistema distribuído, não ocorreu uma concordância entre pesquisadores e engenheiros sobre o que é computação em nuvem. Contudo em 2011, o NIST (*National Institute of Standards and Technology*), entidade de padronização ligada ao governo americano, publicou sua definição de computação em nuvem e como melhor usá-la. Segundo NIST,

> "a computação em nuvem é um modelo para permitir acesso de rede onipresente, conveniente e sob demanda a um conjunto compartilhado de recursos de computação (rede, servidores, armazenamento, aplicativos e serviços) configuráveis que podem ser rapidamente provisionados e liberados com o mínimo esforço de gerenciamento ou interação com o provedor de serviços"(NIST, 2011, p. 2).

Com esta definição, o modelo de nuvem proposto pelo NIST é composto por cinco características essenciais, três modelos de serviço e quatro modelos de implantação.

3.2.1 Características essenciais

A definição do NIST de computação em nuvem afirma que o modelo de nuvem compreende cinco características essenciais. Essas características são: autoatendimento sob demanda, amplo acesso a rede, agrupamento de recursos, elasticidade rápida e medição de serviço.

1. **Autoatendimento sob demanda (*On-demand self-service*)**:
   Essa característica é sobre o

   > "consumidor poder provisionar unilateralmente os recursos de computação, como tempo de servidor e armazenamento de rede, conforme necessário, automaticamente, sem exigir interação humana com cada provedor de serviço"(NIST, 2011, p. 2).

   Com o autoatendimento sob demanda, o consumidor pode agendar o uso de serviços em nuvem, como bem assim entender, além de gerenciar e implantar esses serviços.

   > "Para ocorrer uma maior eficácia para o consumidor, a interface de autoatendimento deve ser amigável e fornecer meios eficazes para gerenciar as ofertas de serviço. Essa facilidade de uso e eliminação da interação humana proporciona eficiência e economia de custos tanto para o usuário quanto para o provedor de serviços em nuvem"(KRUTZ; VINES, 2010, p. 9 e 10).

2. **Amplo acesso a rede (*Broad network access*):**
   O NIST define o amplo acesso a rede no qual os recursos

   > "estão disponíveis na rede e são acessados por meio de mecanismos padrão que promovem o uso de plataformas heterogêneas de clientes, como por exemplo, telefones celulares, tablets, labtops e estações de trabalho"(NIST, 2011, p. 2).

   Ou seja, para que a computação em nuvem seja uma alternativa eficaz, os *links* de comunicação devem estar disponíveis, devem fornecer acesso para diferentes dispositivos e diferentes sistemas operacionais e devem garantir que seus serviços sejam acessados de maneira padronizada pelos diferentes dispositivos, assim tendo uma interação heterogênea para seus usuários.
3. **Agrupamento de recursos (*Resource pooling*):**
   Um provedor de nuvem deve ter um agrupamento de recurso grande e flexível

para atender às necessidades do consumidor, fornecer economias de escala, atender a vários consumidores usando um modelo multilocatário e atender aos requisitos de nível de serviço, com diferentes recursos físicos e virtuais atribuídos e reatribuídos dinamicamente de acordo com a demanda do consumidor de forma que não ocorra diminuição no desempenho (NIST, 2011, p. 2) (KRUTZ; VINES, 2010, p. 10).
Para o NIST

> "Há um senso de independência de localização, pois o cliente geralmente não tem controle ou conhecimento sobre a localização exata dos recursos fornecidos, mas pode especificar a localização em um nível mais alto de abstração (por exemplo, país, estado ou data center). Exemplos de recursos incluem armazenamento, processamento, memória e largura de banda de rede."(NIST, 2011, p. 2)

4. **Elasticidade rápida (*Rapid Elasticity*):**

> "A elasticidade rápida refere-se à capacidade da nuvem de expandir ou reduzir os recursos alocados de forma rápida e eficiente para atender aos requisitos da característica de autoatendimento. Essa alocação pode ser feita de maneira automática e aparecer par o usuário como um grande conjunto de recursos dinâmicos que podem ser pagos conforme o necessário e quando necessário."(KRUTZ; VINES, 2010, p. 11 e 12).

"Para o consumidor, os recursos disponíveis para provisionamento muitas vezes podem parecer ser ilimitados e podem ser apropriados em qualquer quantidade a qualquer momento"(NIST, 2011, p. 2), diretamente ligado a característica de agrupamento de recursos (Item 3).

5. **Medição de serviço (*Measured service*):**
Devido às características já apresentadas com relação aos serviços da computação em nuvem, a quantidade de recursos de nuvem usados por um consumidor pode ser alocada e monitorada de forma dinâmica e automática. O cliente pode então ser cobrado com base no uso medido apenas dos recursos de nuvem que foram alocados para a sessão específica.
A definição do NIST é que os

> "sistemas em nuvem controlam e otimizam automaticamente o uso de recursos, aproveitando uma capacidade de medição[1] em algum nível de abstração apropriado para o tipo de serviço (por exemplo, armazenamento, processamento, largura de banda e contas de usuário ativas). O uso de recursos pode ser monitorado, controlado e relatado, proporcionando transparência tanto para o provedor quanto para o consumidor do serviço utilizado"(NIST, 2011, p. 2).

3.2.2 Modelos de Serviço

Já há algum tempo, o esquema de classificação geralmente aceito para computação em nuvem foi cunhado como modelo SPI (*Software-Platform-Infrastructure*). Essa sigla

[1] Normalmente, isso é feito com base em pagamento por (*pay-per-use*) ou cobrança por uso(*charge-per-use*).

representa os três principais modelos serviços que são prestados por meio da nuvem: SaaS (*Software as a Service*), ou software como serviço; PaaS (*Platform as a Service*), ou plataforma como serviço; e IaaS (*Infrastrucuture as a Service*), ou infraestrutura como serviço.

Na Figura 2, mostra a relação entre os modelos de serviço e a arquitetura hierárquica.

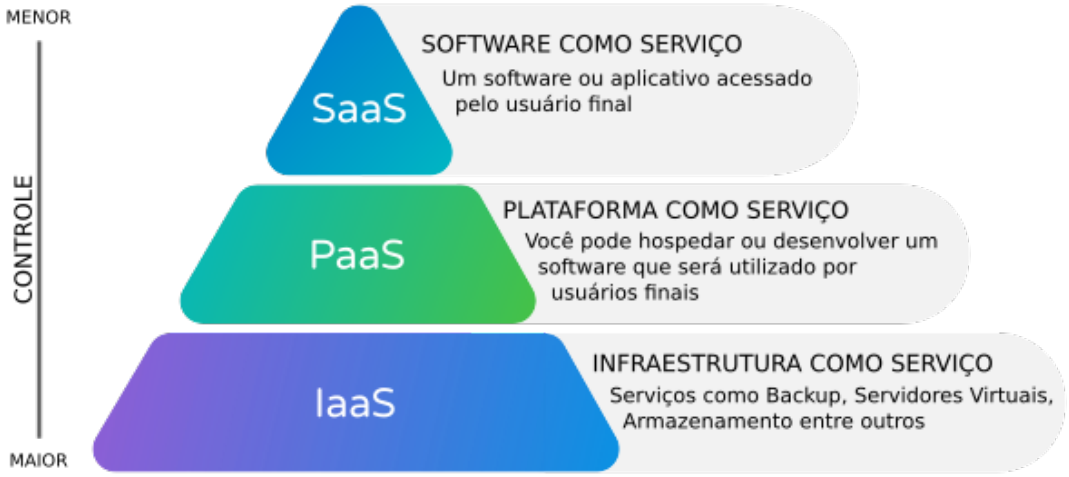


Figura 2 – Modelos de serviços e sua hierarquia.

Fonte: https://tecnomega.com.br/blog/conheca-os-modelos-iaas-paas-saas/

3.2.2.1 Software como Serviço (Software as a Service) - SaaS

No topo da pilha de modelos temos o software como serviço, que é um aplicativo entregue como um serviço ao consumidor. O NIST define que SaaS é

> "A capacidade de fornecer ao consumidor o uso dos aplicativos do provedor em execução em uma infraestrutura de nuvem[2]. Os aplicativos são acessíveis a partir de vários dispositivos clientes por meio de uma interface *thin client*, como um navegador da Web (por exemplo, e-mail baseado na Web) ou uma interface de programa. O consumidor não gerencia ou controla a infraestrutura de nuvem subjacente, incluindo rede, servidores, sistemas operacionais, armazenamento ou mesmo recursos de aplicativos individuais, com a possível exceção de configurações limitadas de aplicativos específicos do usuário."(NIST, 2011, p. 2)

Assim, o software é entregue como um serviço através da Internet, eliminando a necessidade de instalar e executar o aplicativo no computador do cliente, simplificando a

[2] Uma infraestrutura de nuvem é a coleção de hardware e software que permite as cinco características essenciais da computação em nuvem. A infraestrutura de nuvem pode ser vista como contendo uma camada física e uma camada de abstração. A camada física consiste nos recursos de hardware necessários para dar suporte aos serviços em nuvem fornecidos e normalmente inclui componentes de servidor, armazenamento e rede. A camada de abstração consiste no software implantado na camada física, que manifesta as características essenciais da nuvem. Conceitualmente, a camada de abstração fica acima da camada física.

manutenção e o suporte. Normalmente, as aplicações SaaS são independentes de plataforma e podem ser acessadas a partir de vários dispositivos clientes, tais como *workstations, laptops, tablets* e *smarthphones* que executam diferentes SOs. Com isso, os usuários são capazes de acessar a aplicação a partir de qualquer lugar via *browser*. Alguns exemplos de SaaS: Microsoft365[3], Salesforce[4], Facebook[5] e Slideshare[6].

3.2.2.2 Plataforma como Serviço (Platform as a Service) - PaaS

Abaixo do SaaS, na pilha de modelos, temos o PaaS, na qual abstrai muitas funções de nível de pilha de aplicativos padrão e fornece essas funções como um serviço. O NIST define PaaS como sendo

> "A capacidade de fornecer ao consumidor implantar na infraestrutura de nuvem aplicativos criados ou adquiridos pelo consumidor, criados usando linguagens de programação, bibliotecas, serviços e ferramentas suportadas pelo provedor[7]. O consumidor não gerencia ou controla a infraestrutura de nuvem subjacente, incluindo rede, servidores, sistemas operacionais ou armazenamento, mas tem controle sobre os aplicativos implantados e possivelmente definições de configuração para o ambiente de hospedagem de aplicativos."(NIST, 2011, p. 2 e 3)

Isto significa que é fornecido ao usuário a capacidade de desenvolver e implementar aplicativos na nuvem usando as ferramentas de desenvolvimento, as APIs e as bibliotecas. A responsabilidade do usuário é desenvolver, configurar e gerenciar aplicações na infraestrutura da nuvem. O serviço não é a aplicação pronta, mas as ferramentas para desenvolver as aplicações. Exemplos de PaaS são: Microsoft Azure Platform[8], SalesForce Platform[9], Heroku[10], Google App Engine[11] e Digital Ocean App Platform[12].

3.2.2.3 Infraestrutura como Serviço (Infrastrucuture as a Service) - IaaS

Na base da pilha de modelos de serviços temos IaaS, no qual abstrai muitas das tarefas que estão relacionadas ao gerenciamento e a manutenção de um *data center* físico e a sua infraestrutura. O NIST definiu IaaS como sendo

> "A capacidade de fornecer ao consumidor processamento, armazenamento, redes e outros recursos de computação fundamentais onde o consumidor é capaz de implantar e executar software arbitrário, que pode incluir sistemas operacionais

[3] https://www.microsoft.com/pt-br/microsoft-365
[4] https://www.salesforce.com/br/
[5] https://pt-br.facebook.com/
[6] https://pt.slideshare.net/
[7] Esse recurso não exclui necessariamente o uso de linguagens de programação, bibliotecas, serviços e ferramentas compatíveis de outras fontes
[8] https://azure.microsoft.com/pt-br/
[9] https://www.salesforce.com/products/platform/overview/
[10] https://www.heroku.com/
[11] https://cloud.google.com/appengine
[12] https://www.digitalocean.com/products/app-platform

> e aplicativos. O consumidor não gerencia ou controla a infraestrutura de nuvem subjacente, mas tem controle sobre sistemas operacionais, armazenamento e aplicativos implantados; e possivelmente controle limitado de componentes de rede selecionados (por exemplo, firewalls de host)."(NIST, 2011, p. 3)

Isto significa, em outros termos, que o provedor de nuvem aloca aos usuários instâncias de máquinas virtuais, isto é, infraestrutura de computador, usando a tecnologia de virtualização possibilitando o usuário acessar um ambiente padrão de funcionamento do sistema e poder instalar e configurar todas as camadas acima dela. A tecnologia de virtualização é a principal técnica para habilitar a IaaS. Alguns exemplos de IaaS: Amazon Elastic Compute Cloud (EC2)[13] e Google Compute Engine (GCE)[14].

3.2.3 Modelos de Implantação

Dentro de cada um dos três modelos de serviços descritos, há modelos de implantação. Os modelos de implantação não têm relação técnica e funcional com cada um dos modelos de entrega, ou seja, qualquer um dos modelos de entrega pode existir em qualquer um dos cenários de implantação, embora um emparelhamento específico de modelo de serviço e implantação possa ser mais comum do que outros, como por exemplo, SaaS em nuvem pública.

Além disso, com base no uso da nuvem por uma organização e seu relacionamento com a empresa como um todo, esses modelos de implantação de nuvem são geralmente chamados de nuvens externas ou internas. Cada um desses modelos, no entanto, deve compartilhar os princípios fundamentais da computação em nuvem:

- Cada modelo de implantação emprega dispositivos conectados à Internet.
- Cada modelo fornece escalabilidade dinâmica de recursos virtuais.
- Os usuários de cada modelo geralmente não têm controle sobre a tecnologia que está sendo usada.

Os modelos de implantação são: nuvem privada, nuvem comunitária, nuvem pública e nuvem híbrida.

1. **Nuvem privada:**
   Basicamente, é uma nuvem de servidores e software para utilização sem um ponto de acesso público, operadas exclusivamente para uma única organização e acessíveis apenas para membros dessa organização. Assim, o gerenciamento da rede é feito pela própria organização ou por terceiros. A infraestrutura utilizada pertence à organização, por isso possui um alto custo, tendo que comprar, construir e operar toda a infraestrutura.
   O NIST definiu a nuvem privada como sendo uma

   > "infraestrutura de nuvem provisionada para uso exclusivo por uma única organização composta por vários consumidores (por exemplo,

[13] https://aws.amazon.com/pt/ec2/
[14] https://cloud.google.com/compute

> unidades de negócios). Pode ser de propriedade, gerenciado e operado pela organização, um terceiro ou alguma combinação deles, e pode existir dentro ou fora das instalações."(NIST, 2011, p. 3)

É adequada para organização nas quais a segurança é muito importante e/ou para organizações com grande demanda por TI (Tecnologia da Informação). É amplamente defendido que as nuvens privadas possuam mecanismos de segurança e confiabilidade mais severos que nuvens públicas, pois são implantadas dentro do firewall das organizações

2. **Nuvem comunitária:**
   Neste modelo, empresas ou organizações reúnem em *pool* seus recursos na nuvem para resolver um problema comum. A infraestrutura de nuvem é compartilhada por várias organizações e suporta uma comunidade específica que tenha as mesmas preocupações, requisitos de segurança e política de gerência. O gerenciamento é feito pelas organizações ou por terceiros, localmente ou remotamente. O NIST definiu como uma

   > "infraestrutura de nuvem fornecida para uso exclusivo por uma comunidade específica de consumidores de organizações que compartilham preocupações (por exemplo, missão, requisitos de segurança, política e considerações de conformidade). Pode ser de propriedade, gerenciado e operado por uma ou mais organizações da comunidade, um terceiro ou alguma combinação deles, e pode existir dentro ou fora das instalações."(NIST, 2011, p. 3)

   Adequado para organizações que querem acessar as mesmas aplicações e dados, e querem compartilhar também o custo da nuvem com o grupo.

3. **Nuvem pública:**
   É um modelo na qual a nuvem é disponibilizada ao público ou para grandes grupos industriais e aberta para qualquer usuário que possa pagar pela alocação dos recursos. É operada pelo provedor da nuvem, que é responsável pela manutenção e segurança da mesma. A definição do NIST:

   > "A infraestrutura de nuvem é provisionada para uso aberto pelo público em geral. Pode ser de propriedade, gerenciado e operado por uma organização empresarial, acadêmica ou governamental, ou alguma combinação deles. Ele existe nas instalações do provedor de nuvem."(NIST, 2011, p. 3)

4. **Nuvem híbrida:**
   Neste modelo, a infraestrutura é composta por dois ou mais modelos de implementação. Cada nuvem permanece como uma entidade única, porém estão juntas pelo uso de tecnologia proprietária ou padronizada que garante a portabilidade de dados e aplicações. Para o NIST a nuvem híbrida é uma

   > "infraestrutura de nuvem que tem em sua composição duas ou mais infraestruturas de nuvem distintas (privadas, comunitárias ou públicas) que permanecem entidades únicas, mas são unidas por tecnologia padronizada ou proprietária que permite a portabilidade de dados e aplicativos (por exemplo, estouro de nuvem para balanceamento de carga entre nuvens)."(NIST, 2011, p. 3)

A nuvem privada pode ter seus recursos ampliados pela reserva de recursos em uma nuvem pública. Com a nuvem híbrida é possível manter os níveis de serviço mesmo no caso de flutuações rápidas na necessidade de recursos.

3.2.4 Considerações Finais

O uso de computação na nuvem vem crescendo com o passar dos anos e esse aumento se deve por alguns fatores que a nuvem proporciona tanto ao prestador de serviço na nuvem tanto ao consumidor desse serviço. Em um serviço sem nuvem a empresa que fornece o seu próprio software precisa pensar nos servidores, na fonte de energia dedicada a esses servidores e possuir peças em estoque para, caso aconteça algum problema, ocorra a substituição da peça defeituosa. Além disso, é necessário manter uma equipe para configuração e monitoramento para que possa resolver problemas de emergência. Porém, quando o software é baseado em nuvem, essas preocupações e os custos com a infraestrutura desaparecem, pois esses valores são bastantes previsíveis, pois o provedor na nuvem é o responsável por essas preocupações e ele deve assegurar que seja um processo tranquilo para o cliente e que a aplicação não sofra com interrupções no seu serviço em troca de um custo fixo e razoável.

Outro fator para o uso da computação na nuvem é sua segurança. É permitido que seja implementado estratégias, práticas e tecnologias de ponta, com uma visão mais ampla dos padrões globais de ameaças. Normalmente os provedores de serviços na nuvem trabalham com um orçamento voltado para a segurança maior, pois precisam garantir a segurança de todos os clientes, mantendo os benefícios de cada empresa. Com isso, os provedores podem fornecer a mesma uma infraestrutura fortalecida, um cuidadoso monitoramento e a aplicação de protocolos de segurança tanto para uma empresa de pequeno ou médio porte tanto para uma empresa que possui altos padrões e requisitos exigentes.

Um último fator importante para o crescimento do uso da tecnologia na nuvem é a capacidade de colocar todos os clientes em um mesmo nível. A computação na nuvem democratiza o aplicativo de software corporativo, pois o fator de uma plataforma ter dezenas ou milhares de acessos não é mais relevante. Os provedores e empresas possuem uma flexibilidade alta de aumentar ou diminuir a capacidade de sua plataforma, o que não afeta o usuário final e muitas vezes nem fica ciente do que está acontecendo na nuvem.

Por fim, é notório que o uso de computação na nuvem possui grandes vantagens para os provedores e consumidores deste serviço. Utilizando esta tecnologia é possível reduzir a pegada de carbono da empresa, possibilita uma alta colaboração entre os funcionário por meio do compartilhamento de aplicativos e documentos, os provedores se tornam responsáveis pela atualização e manutenção do serviço, uma maior facilidade e agilidade na recuperação de possíveis desastres e há uma maior segurança nos dados armazenados em nuvem devido a possibilidade deles serem acessados de qualquer lugar independentemente

da perda ou dados aos aparelhos. Com isso, chegamos ao ponto central da discussão, a segurança na nuvem.

3.3 SEGURANÇA EM NUVEM

A segurança na nuvem, também conhecida como segurança computacional na nuvem, inclui conjuntos de políticas, controles, procedimentos e tecnologias que funcionam em conjunto para proteger sistemas, dados e infraestrutura baseados na nuvem. Como mencionado anteriormente, a computação na nuvem dá às empresas acesso a um novo patamar, fornecendo um melhor atendimento ao cliente, garantindo coleta e armazenamento de dados aprimorados, maior flexibilidade por meio de trabalho remoto e escalabilidade rápida e uma maior conveniência por meio de sistemas interconectados com compartilhamento de dados e arquivos.

Existem certos riscos de configuração e o perigo sempre presente de criminosos cibernéticos para que o ambiente de nuvem de qualquer empresa tenha uma segurança eficaz. Com a segurança na nuvem é possível melhorar o nível de proteção dos ativos digitais e mitigar os riscos associados a possíveis erros humanos, assim, reduzindo a probabilidade de que uma organização sofra uma perda prejudicial devido a uma violação evitável. Por isso, a responsabilidade pela segurança na nuvem se torna uma responsabilidade compartilhada entre o cliente e o provedor. A maioria das violações de segurança acontecem devido a falta de confiança na segurança do fornecedor, ou seja, falta de diligência prévia.

Na Tabela 1 é possível ver como funciona a responsabilidade compartilhada para cada modelo de serviço em nuvem. Em um ambiente local, também conhecido como *on-premise*, tudo se torna responsabilidade do usuário/empresa. Em um ambiente em nuvem o usuário pode ter menos responsabilidade, porém, em todos os tipos de modelo, o usuário será sempre responsável pela governança, risco, conformidade e segurança dos dados, contudo, nunca será responsável pela segurança física do local.

No modelo IaaS o usuário tem mais controle sobre a nuvem, ou seja, possuindo maior responsabilidade, sendo elas a segurança da aplicação, gerenciamento de identidade e acesso, segurança dos dados e governança, risco e conformidade. A responsabilidade compartilhada entre provedor e cliente é sobre a segurança da rede e segurança da aplicação e o provedor tendo somente a responsabilidade sobre a segurança física.

Para o modelo PaaS o cliente possui menos responsabilidade do que com relação ao IaaS. Nesse modelo, o usuário tem como responsabilidade a segurança dos dados e governança, risco e conformidade. O provedor tem como responsabilidade a segurança física, segurança da infraestrutura do host e a segurança da rede. E, por fim, a responsabilidade compartilhada é sobre a segurança da aplicação e o gerenciamento de identidade e acesso.

No modelo SaaS o provedor tem mais responsabilidade sobre a nuvem, tendo como sua responsabilidade a segurança física, segurança da infraestrutura do host, segurança da rede e segurança da aplicação. A responsabilidade do usuário/empresa é a mesma do

Tabela 1 – Modelo de responsabilidade compartilhada em nuvem.

Responsabilidade para cada modelo de nuvem	**Local**	**IaaS**	**PaaS**	**SaaS**
Governança, risco e conformidade				
Segurança dos Dados				
Gerenciamento de identidade e acesso				
Segurança da aplicação				
Segurança da rede				
Segurança da infraestrutura do host				
Segurança física				

Provedor	
Compartilhada	
Usuário/Empresa	

Fonte: Autor.

que no modelo PaaS, governança, risco, conformidade e segurança dos dados, restando o gerenciamento de identidade e acesso como responsabilidade compartilhada entre as entidades.

O modelo de responsabilidade compartilhada tem como objetivo cumprir o que muitos autores denominam como sendo a tríade CIA (*Confidentiality, Integrity and Availability*), ou, em português, CID (Confidencialidade, Integridade e Disponibilidade), da segurança da informação e são pilares importantes da garantia de software em nuvem (KRUTZ; VINES, 2010, p. 63).

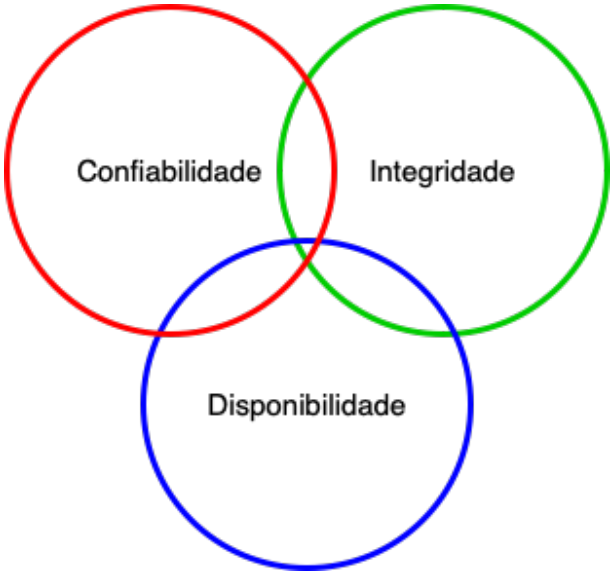


Figura 3 – Pilares da segurança da informação.

Fonte: Autor

A confiabilidade é sobre prever a divulgação não autorizada, intencional ou não intencional, de informações. Ela está relacionada com as área de direitos de propriedade

intelectual, criptografia, canais secretos, análise de tráfego e inferência. Um sistema que promete garantir a confidencialidade, no caso de um terceiro que capture informações trocadas entre duas partes, esse não será, nem poderá ser, capaz de extrair qualquer informação do conteúdo capturado.

A integridade das informações em nuvem exigem que três princípios sejam atendidos (KRUTZ; VINES, 2010, p. 64). O primeiro é que não são feitas modificações nos dados por pessoal ou processos não autorizados. O segundo é que modificações não autorizadas, não são feitas nos dados por pessoal ou processo autorizado. Por fim, o terceiro é que os dados são internamente e externamente consistentes. É a integridade que garante que os dados sejam corretos, autênticos e confiáveis, garantindo que não foram alterados ou adulterados.

A disponibilidade é o que garante o acesso confiável e que o sistemas aplicativos e dados estejam disponíveis paras os usuários autorizados de forma oportuna para eles, ou sejam, quando precisarem. A rede, os sistemas e os aplicativos devem estar constantemente ativos e funcionando de forma a garantir que não ocorra interrupção. Um ataque de negação de serviço é um exemplo de ameaça contra a disponibilidade.

Além disso, é preciso que toda a infraestrutura da nuvem seja projetada de uma forma que ela possa sustentar os requisitos de segurança, indo desde implementação da rede aos componentes e suas configurações. Para isso, a rede possui uma arquitetura em camadas, cada uma com a finalidade e responsabilidade de realizar uma tarefa específica que lhe foi atribuída.

3.4 MODELO EM CAMADAS

Em uma arquitetura de camadas, cada camada possui a responsabilidade de prestar seu serviço para a camada acima e consumir os serviços que vem da camada abaixo dela. Com essa arquitetura se torna mais fácil analisar de uma parcela e bem definida de uma sistema grande e complexo. "Essa simplificação tem considerável valor intrínseco, pois provê modularidade, tornando muito mais fácil modificar a execução do serviço prestado pela camada"(KUROSE; ROSS, 2013, p. 36). Essa modularidade vem da falta de necessidade de montar uma rede utilizando somente equipamentos de um mesmo fabricante. A padronização dos protocolos permite que equipamentos que realizam uma função dentro do modelo de camada possam ser substituídos por outros equipamentos de outras marcas realizando a mesma função, sem afetar o funcionamento das redes.

Analisando a pilha de protocolos da Internet, também conhecido como modelo TCP/IP, temos cinco camadas, que de cima para baixo são: aplicação, transporte, rede, enlace e física. O modelo OSI (*Open Systems Interconnection*), desenvolvido pela Organização Internacional de Normalização, ISO (*Internacional Organization for Standards*), divide em sete camadas a arquitetura da Internet, sendo eles de cima para baixo: aplicação, apresentação, sessão, transporte, rede, enlace e física. A Figura 4 exemplifica de forma

simples a conexão de dois sistemas diferentes. No modelo TCP/IP a camada de aplicação engloba a camada de apresentação e sessão referente ao modelo OSI

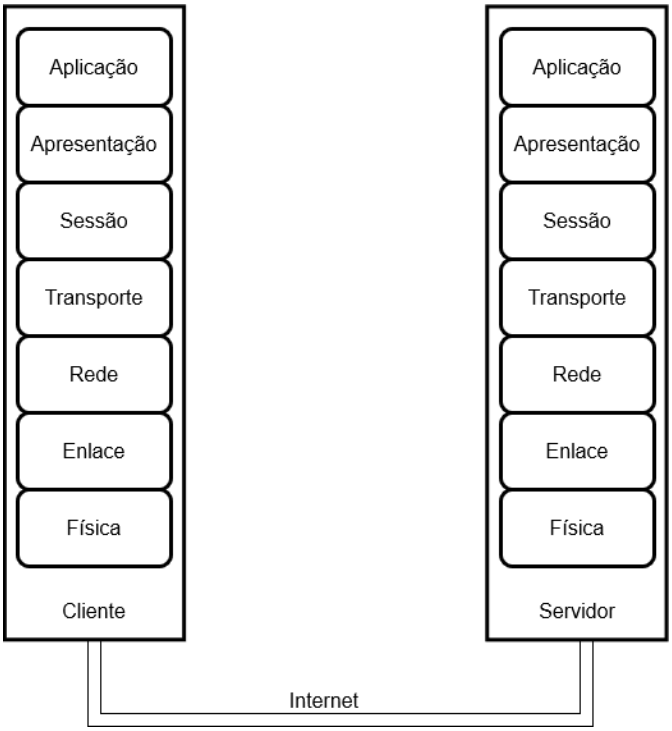


Figura 4 – Simplificação do modelo OSI, adaptado de (KUROSE; ROSS, 2013).

Será abordada nesta sessão somente as camadas de aplicação, transporte, rede e enlace, pois são as camadas relacionadas com a proposta deste trabalho.

- **Camada de aplicação:** No topo do modelo OSI é onde estão as aplicações e seus protocolos, tais como HTTP, FTP e DNS. "Um protocolo de camada de aplicação é distribuído por diversos sistemas finais, e a aplicação em um sistema final utiliza o protocolo para trocar pacotes de informação com a aplicação em outro sistema final"(KUROSE; ROSS, 2013, p. 38).
- **Camada de transporte:** "A camada de transporte da Internet carrega mensagens da camada de aplicação entre os lados do cliente e servidor de um aplicação"(KUROSE; ROSS, 2013, p. 38). A função principal da camada de transporte é realizar a comunicação de maneira transparente sendo capaz de prover uma comunicação fim-a-fim. Sendo seus protocolos o TCP, para serviços orientados a conexão, e o UDP, para serviços não orientados a conexão.
- **Camada de rede:** Responsável por mover as informação pela rede, de um hospedeiro para outro. Ela "provê o serviço de entrega do segmento à camada de transporte no hospedeiro de destino"(KUROSE; ROSS, 2013, p. 38). É nessa camada que encontramos o protocolo IP, o qual define os campos do

datagrama[15] e como os sistemas finais e roteadores agem nesse campo.

- **Camada de enlace:** O serviço básico da camada de enlace é mover um datagrama de um nó[16] até outro nó adjacente por um único meio de comunicação (KUROSE; ROSS, 2013, p. 323). O serviço que a camada de enlace podem variar dependendo do protocolo empregado no enlace.

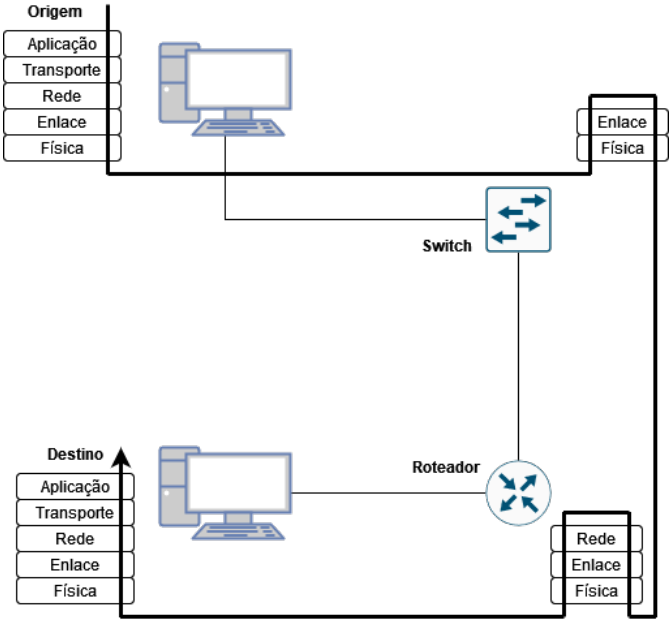


Figura 5 – Arquitetura de rede simples, adaptado de (KUROSE; ROSS, 2013, p. 40).

A Figura 5 mostra um arquitetura de rede simples que contêm hospedeiros, roteador e *switch*, cada um contendo um conjunto diferente de camadas, o que mostra suas diferentes funcionalidade. O roteador é um equipamento de camada de rede tendo sua principal serviço a "função de repasse - a transferência, propriamente dita, de pacote dos enlaces de entrada de saída"(KUROSE; ROSS, 2013, p. 235). São responsáveis por rodar os algoritmos de roteamento, ou protocolos de roteamento, que trocam e calculam as informações que são utilizadas para configurar as tabela de repasse.

Já um *switch*, ou também um comutador, é responsável por "receber quadros[17] da camada de enlace e repassá-los para enlace de saída". "O comutador em si é transparente aos hospedeiros e roteadores na sub-rede"(KUROSE; ROSS, 2013, p. 352). Por fim, é muito comum confundirem a função de ambos que, de maneira resumida, os roteadores são comutadores que transmitem pacotes usando endereços da camada de rede, endereçamento IP, e os *switchs* transmitem pacotes usando endereços MAC.

[15] Como são conhecidos os pacotes na camada de rede.
[16] Qualquer dispositivo que rode um protocolo da camada de enlace.
[17] Como são conhecidos os pacotes na camada de enlace.

3.4.1 Protocolos utilizados

Os protocolos mais utilizados foram de camada de rede, ou camada 3, e isso se deve principalmente pelo fato de como a estratégia do estudo do referente trabalho ocorreu. Os protocolos utilizados foram: IP, DHCP, NAT, ICMP e OSPF.

O protocolo IP (*Internet Protocol*) é utilizado como identificação de um dispositivo conectador à Internet, no jargão, é utilizado como "endereço IP". Em suma, o endereço IP é o que permite que informações possam ser enviadas entre dispositivos em um rede, contendo nele as informações sobre a localização do dispositivo. Esses endereços são escritos em notação decimal separada por pontos com um comprimento de 32 bits (equivalente a 4 bytes), tendo um total de 2^{32} endereços IP possíveis, ou seja, por aproximação, há 4 bilhões de endereços IP possíveis. (KUROSE; ROSS, 2013, p. 250)

O DHCP (*Dynamic Host Configuration Protocol*) - Protocolo de Configuração Dinâmica de Hospedeiros - "permite que um hospedeiro obtenha um endereço IP de maneira automática"(KUROSE; ROSS, 2013, p. 255). É conhecido como um protocolo *plug-and-play* tornando bastante atraente a um administrador de rede utilizá-lo do que configurar manualmente.

O NAT (*Network Address Translation*) - tradução de endereço de rede - é responsável por traduzir um endereço IP privado para endereço IP público, ou vice-versa. "Um domínio com endereços privados refere-se a uma rede cujos endereços somente têm significado para equipamentos pertencentes àquela rede". "Na essência, o roteador que usa NAT está ocultando do mundo exterior os detalhes da rede residencial"(KUROSE; ROSS, 2013, p. 258).

O protocolo de mensagens de controle da internet, ICMP (*Internet Control Message Protocol*), é utilizado por hospedeiros e roteadores para realizar a comunicação de informações de camada de rede entre si, sendo mais comum seu uso para comunicação de erros. As mensagens ICMP são carregadas dentro de datagramas IP, como carga útil (KUROSE; ROSS, 2013, p. 260).

O OSPF (*Open Shortest Path First*) é utilizado para roteamento dentro de sistemas autônomos, intra-AS.

> "É um protocolo de estado de enlace que usa inundação de informação de estado de enlace e um algoritmo de caminho de menos custo de Dijkstra. Com o OSPF, um roteador constrói um mapa topológico completo de todo o sistema autônomo. O protocolo OSPF tem como verificar se os enlaces se encontram operacionais via mensagem HELLO enviada a um vizinho ligado adjacente."(KUROSE; ROSS, 2013, p. 286)

3.5 SEGURANÇA EM REDE

Em uma rede segura, o remetente deseja que somente o destinatário entenda o conteúdo da mensagem que ele enviou, mesmo que eles estejam se comunicando por um meio inseguro, em que um intruso possa interceptar qualquer dado que seja transmitido. No

lado do destinatário, ele quer ter certeza de que está se comunicando com o remetente certo (KUROSE; ROSS, 2013, p. 496). Isso, como já foi mostrado anteriormente na Figura 3, satisfaz os pilares da segurança da informação.

3.5.1 Redes virtuais privadas - VPNs

Com a finalidade de resolver o problema de enviar informações sensíveis por um meio inseguro, como a Internet, muitas empresas adotam a solução VPN (*Virtual Private Network*) sob a Internet pública. A VPN fornece a capacidade de estabelecer uma conexão de rede que é protegida, pois elas criptografam todo o tráfego e, também, mascaram a identidade do usuário para a rede pública.

Como a VPN oculta seu endereço IP, ela permite que a rede redirecione o tráfego para um servidor remoto que foi pré configurado como um servidor VPN, ou seja, um host que está conectado a uma VPN, o servidor VPN se torna a fonte de seus dados.

3.5.2 Firewall

A adoção de um firewall em uma rede se torna viável quando é preciso isolar uma rede interna de uma empresa ou organização da Internet em geral. Um firewall permite um controle de acesso entre o mundo externo e os recursos da rede, permitindo que um administrador gerencie o fluxo de tráfego de e para seus recursos (KUROSE; ROSS, 2013, p. 538). Os firewalls podem ser classificados em três categorias: filtros de pacotes, filtros de estado e gateways de aplicação.

Um firewall filtro de pacotes é responsável por examinar cada datagrama, determinando se deve passar ou ser descartado/bloqueado baseados nas regras específicas que foram definidas por um administrador. Essas regras podem ser (KUROSE; ROSS, 2013, p. 539):

- Endereço IP de origem e/ou de destino.
- Tipo de protocolo: TCP, UDP, OSPF, ICMP, etc.
- Porta TCP ou UDP de origem e/ou de destino.
- Bit de *flag* do TCP: SYN, ACK, etc.
- Tipo de mensagem ICMP.
- Regras diferentes para datagramas que entram e saem da rede.
- Regras diferentes para diferentes interfaces do firewall.

Um firewall filtro de estado rastreia as conexões TCP e utilizam esse conhecimento para tomar a decisão sobre a filtragem, que é realizada em cada pacote isolado. Isso ocorre pois o firewall nota o inicio de uma conexão observando a apresentação de três vias (SYN, SYNACK e ACK) e observa o fim de uma conexão ao ver um pacote FIN.

Alem de examinar os cabeçalhos IP, TCP e UDP e tomar as decisões com base em dados da aplicação, os firewall gateway de aplicação é um servidor específico de aplicação, no qual todos os dados de uma determinada aplicação devem passar. É possível que sejam executados vário gateways de aplicação em um mesmo host, porém cada gateway é considerado um servidor separado tendo seus próprios processos. Em algumas redes internas é possível encontrar vários gateways de aplicação, como gateways para FTP, Telnet, HTTP e e-mail.

3.5.3 Sistemas de detecção e prevenção de invasão - IDS/IPS

Com a finalidade de reconhecer quais são os pacotes/conexões legítimas de usuários confiáveis e reais, é preciso ir além de somente analisar um datagrama ou pacote e realizar uma inspeção mais profunda, ou seja, analisar além dos campos de cabeçalhos e ir dentro dos dados da aplicação que o pacote carrega, sendo assim possível detectar alguns tipos de ataques.

Esse trabalho de inspecionar profundamente um pacote é realizado por um sistema de detecção de invasão, IDS (*Intrusion Detection System*), e por um sistema de prevenção de invasão, IPS (*Intrusion Prevention System*). Um IDS analisa o tráfego de rede e gera alertas quando observa algum tráfego potencialmente malicioso. Um IPS também analisa o tráfego de rede, porém realiza a filtragem de pacote, ou seja, caso detecte algum tráfego malicioso, ele impede que esses pacotes cheguem em seu destino. Com isso, a principal diferença entre os dois é que o IDS é um sistema de monitoramento e o IPS é um sistema de controle, porém ambos, se usados corretamente, se complementam em suas funcionalidades.

Utilizando um IDS/IPS é possível detectar alguns ataques, como o mapeamento de rede, varreduras de portas, varreduras de pilha TCP, ataques DoS, ataques de inundação de largura de banda, worms e vírus, ataques de vulnerabilidade de SO e ataques de vulnerabilidade de aplicações (KUROSE; ROSS, 2013, p. 544).

3.5.4 Zona desmilitarizada - DMZ

Uma zona desmilitarizada, DMZ (*Demilitarized Zone*), é uma rede que tem como função principal a separação dos serviços que têm acesso externo da rede local, aumentando a segurança e, por consequência, diminui possíveis riscos de um dano que possa ser causado por um usuário malicioso, isso tanto interno ou externo a DMZ.

Com a finalidade de permitir essa confidencialidade aos dados que estão dentro da DMZ, é usado um firewall para realizar o controle de tráfego de rede entre a rede e a Internet, e possivelmente uma rede interna.

3.6 ATAQUES CIBERNÉTICOS COMUNS EM AMBIENTES DE NUVEM

Em um ambiente de nuvem computacional pode ocorrer vários tipos de ataques cibernéticos, como o roubo de dados, injeção de malwares, ataques de negação de serviço, ataques que envolvem engenharia social, explorações de vulnerabilidades e apropriação de contas.

O ataque que envolve o roubo de dados tem como objetivo acessar e extrair informações confidenciais que estão armazenadas na nuvem. A injeção de malware pode ocorrer com a implantação nas máquinas virtuais da nuvem para coletar informações sensíveis e monitorar atividades. O ataque de negação de serviços, também conhecido como DoS (*Denial of Service*), é um ataque que tem como objetivo de sobrecarregar os recursos de uma nuvem, tornando-a inacessível para usuários legítimos, aqueles que realmente querem se conectar. O ataque de engenharia social visa enganar os usuários com o objetivo de que revelem informações sensíveis ou baixem vírus, malwares ou ransonwares. A exploração de vulnerabilidades acontece quando invasores buscam brechas na configurações de segurança da nuvem para obter acesso não autorizado. Por fim, a apropriação de contas acontece quando os invasores tentam adquirir credenciais de usuários para acessar informações na nuvem ou realizar ações maliciosas.

Como a nuvem é uma tecnologia que está em constante evolução, novas maneiras de ataques surgem a qualquer momento, por isso é importante sempre atualizar e configurar de maneira metódica as regras de firewall e fazer o uso de ferramentas que auxiliam o trabalho de defesa, para que a superfície de ataque seja a menor possível.

4 METODOLOGIA

Este capítulo apresenta como ocorreu a implantação da nuvem privada de IaaS no GNS3, que se deu por meio da plataforma Openstack. Também explicará como acontece o acesso do usuário cliente a plataforma de infraestrutura como serviço.

A Seção 4.1 apresentará a criação da topologia da rede dentro do GNS3. A Seção 4.2 abordará a instalação da plataforma Openstack. A Seção 4.3 trará a interação entre o usuário cliente e o usuário malicioso no sistema. E, enfim, a Seção 4.4 mostrará ferramentas e um ataque que exploram a superfície de ataque do sistema.

4.1 CRIAÇÃO DA TOPOLOGIA DE AMBIENTE EM NUVEM

Para se trabalhar com um ambiente nuvem, foi escolhido construir uma topologia em um ambiente que pudéssemos emular um ambiente real de rede, com dispositivos que são realmente usados, por isso, optou-se por utilizar o software de emulação GNS3. O GNS3 é um simulador de redes que emula os mais diversos equipamentos ativos de uma rede, como roteadores, switchs, servidores, computadores pessoais (PCs), telefones, firewall e etc. Pegando como exemplo a categoria des roteadores, o GNS3 consegue emular o sistema operacional de um roteadore real e operar as respectivas configurações. O software integra-se coma as ferramentas de virtualização virtual e QEMU, sendo que há um conjunto de imagens, com sistemas operacionais, prontas para serem baixadas e utilizadas, chamados de *appliances*.

4.1.1 Configuração da topologia

A topologia é composta por dois roteadores, um firewall, um switch, uma máquina Ubuntu Desktop 20.04 LTS, uma máquina Ubuntu Server 20.04 LTS, um máquina Windows 10 e uma máquina Kali Linux, como mostra a Figura 6. Os roteadores formam a rede que liga a rede privada a rede mundial de computadores. O firewall, o switch e as duas máquinas Ubuntu compõem a DMZ.

Para a rede principal foi escolhido a faixa de IP 172.21.2.0/24, que foi dividida em quarto sub-redes, ou seja, com máscara /26, sendo a rede entre roteador R1 e roteador R2 a sub-rede 172.21.2.0/26, a rede entre R2 e firewall a sub-rede 172.21.2.64/26, a rede entre R1 e cliente a sub-rede 172.21.2.128/26 e a rede entre R1 e usuário malicioso a sub-rede 172.21.2.192/26.

Para a rede DMZ foi escolhido a faixa de IP 192.168.1.0/24. Não foi necessário dividir essa rede entre os dispositivos, pois só iria gerar um aumento da complexidade da rede DMZ. A atribuição de endereçamento IP para essa rede se deu por meio da implementação de um servidor DHCP configurado no firewall.

Os rotadores rodam o sistema operacional de rede de código aberto chamado VyOS

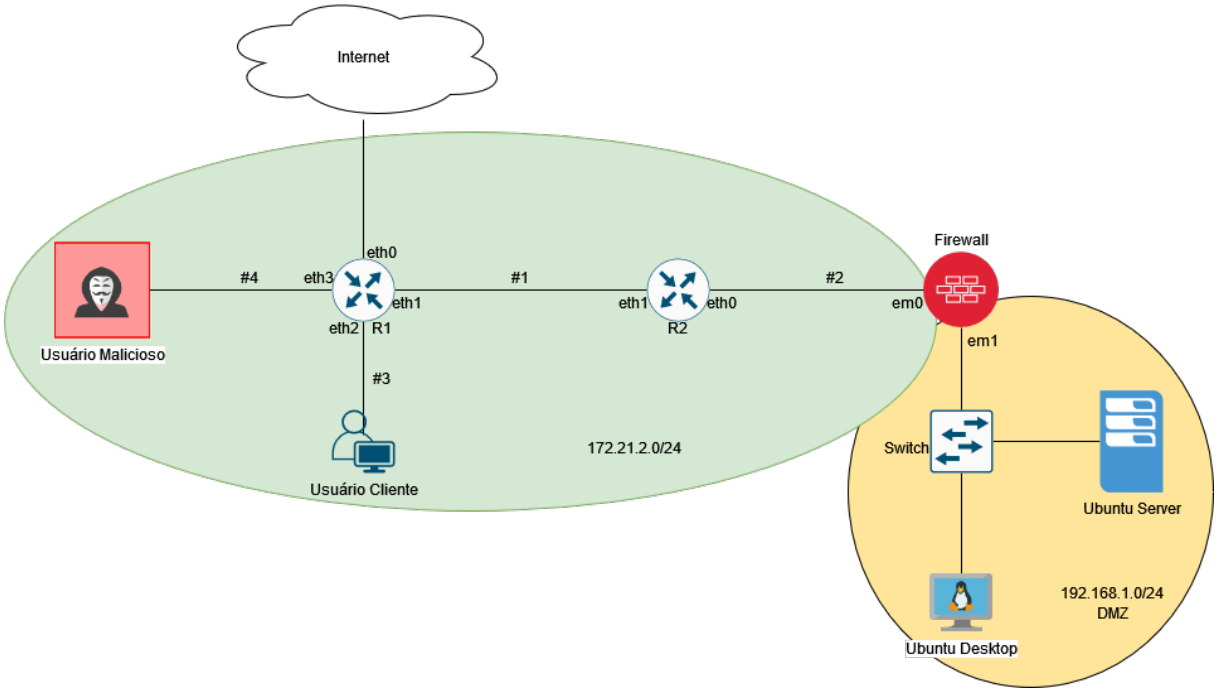


Figura 6 – Topologia implementada

Fonte: Autor.

Tabela 2 – Parâmetros de configuração das sub-redes

Sub rede	**IP de sub rede**
#1	172.21.2.0/26
#2	172.21.2.64/26
#3	172.21.2.128/26
#4	172.21.2.192/26

Fonte: Autor.

que é um SO baseado no Debian GNU/Linux e fornece uma plataforma de roteamento gratuita. Ele pode ser utilizado como um dispositivo de roteador e firewall para implantações em nuvem, pois é executado nos sistemas padrão amd64, i586 e ARM. Ambos roteadores foram configurados para utilizar o OSPF como protocolo de roteamento, porém, somente o roteador R1 foi configurado com o protocolo NAT, pois é o roteador que faz contato com outra rede, pela interface eth0.

O roteador R1 possui quarto conexões, com a Internet, com o roteador R2, com o usuário cliente e com o usuário malicioso. O roteador R2 possui duas conexões, com o roteador R1 e com o firewall pfSense. Os endereços IPs de cada interface para cada roteador foi definido seguindo a Tabela 2 como referência e esse endereços podem ser visualizados na Tabela 3. É notório que somente a interface do roteador R1 não foi definida com um endereço de IP estática, mas sim recebendo um enderenço IP via DHCP.

Para o firewall, foi utilizado a *appliance* do pfSense. O pfSense é um software de

Tabela 3 – Endereços IPs das interfaces dos roteadores R1 e R2

Roteador	**Interface**	**Endereço IP**
R1	eth0	DHCP
	eth1	172.21.1.1
	eth2	172.21.2.129
	eth3	172.21.2.192
R2	eth0	172.21.2.2
	eth1	172.21.2.65

Fonte: Autor.

código livre e gratuito que tem como sistema operacional como base o FreeBSD, que é um SO que pode ser usado como um Linux firewall e até mesmo como um roteador de redes. O pfSense tem como principais características o seu fácil gerenciamento, possuindo uma interface web fácil de ser gerenciada, permitindo o administrador da rede visualizar o ambiente em tempo real. É considerado um firewall UTM (*Unified Threat Management* ou Gerenciamento Unificado de Ameaças), antispam, antispyware, antivírus e detector de intrusos, podendo, também, realizar filtragem de conteúdos. Além disso, possui licença aberta permitindo a customização do pfSense de acordo com as necessidades.

Em um primeiro cenário, a configuração do firewall foi feita para proteger e controlar o acesso a rede de nuvem. Inicialmente, foi feita uma regra que só permitiria o acesso à DMZ a partir de conexões TCP, para que usuários externos a rede pudessem se conectar (realizando um login em uma plataforma de nuvem), porém, após algumas análise de segurança, que serão abordadas nas próximas seções, foi criado um segundo senário no qual a regra inicial foi substituída por uma regra em que o acesso só era permitido a partir de uma conexão VPN. A VPN foi configurada pelo próprio pfSense, utilizando o OpenVPN.

O OpenVPN utiliza o protocolo UDP (*User Datagram Protocol*) ou TCP (*Transmission Control Protocol*) para realizar o transporte de dados. Optou-se pelo uso do OpenVPN ao invés do IPsec, pois o OpenVPN é uma solução de VPN de software livre, possui uma configuração e gerenciamento mais simples. Neste trabalho, a conexão VPN foi configurada para ser através de um túnel UDP utilizando a porta padrão do OpenVPN, a 1194. Portanto, a regra final no firewall somente permitirá conexões feitas por meio desse túnel UDP na porta 1194. A Figura 7 exemplifica o funcionamento da conexão VPN.

Também foi instalado a ferramenta de código aberto chamada Suricata que fornece o serviço IDS/IPS. Ela é projetada para monitorar o tráfego de rede e detectar atividades maliciosas, como explorações de vulnerabilidades, invasões, ataques de negação de serviço, etc. Ela utiliza uma arquitetura de assinaturas para detectar ameaças, ou seja, ele precisa ser atualizado continuamente com as últimas assinaturas de ameaças conhecidas. Optou-se pelo Suritcata ao invés do Snort, pois o Suricata oferece suporte a protocolos adicionais, incluindo IPv6, HTTP e TLS, além de possuir recursos avançados de detecção de ameaças,

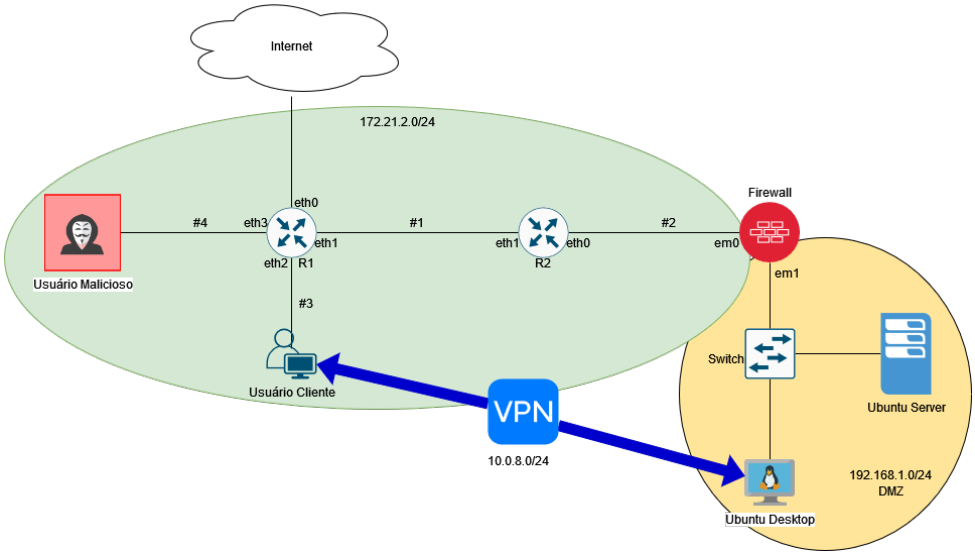


Figura 7 – Topologia com a VPN.

Fonte: Autor.

como o comportamento baseado em inteligência artificial. Ambos, são ferramentas de IDS eficazes, porém o Suricata é considerado mais avançado e escalável, com recursos adicionais que o colocam na frente do Snort.

O Suricata foi configurado para monitorar a interface WAN do firewall e para criar alertas de tráfego suspeito, assim, o bloqueio de IP só aconteceria no caso de uma intervenção manual do administrador da rede após verificar o logs de alerta.

4.2 PLATAFORMA DE IAAS - OPENSTACK

Na atualidade, existem diversas plataformas para a implementação de nuvem IaaS, de forma a filtrá-las, foi necessário realizar uma busca de uma plataforma que atendesse os requisitos para o trabalho em questão, que são: ser uma plataforma de código aberto, que fosse instalável e suportável em um ambiente de testes controlado (GNS3) e que suportasse componentes de computação, armazenamento e rede. Em (LINUXLINKS, 2020), (YOUTECHDIET, 2023), (LINUX.COM, 2014) e (CLOUD COMPUTING PROJECTS, 2023) é possível encontrar uma consistência quando citam a plataforma OpenStack, na qual essa plataforma é umas das melhores plataformas de nuvem IaaS atualmente no mercado, como mostra a figura Figura 8.

O OpenStack é um sistema operacional em nuvem que pode controlar grupos de recursos de computação, armazenamento e rede em uma datacenter, sendo possível gerenciá-los e provisioná-los por meio de APIs que fornecem para os administradores o controle enquanto capacita seus usuários a provisionar recursos por meio de uma interface web (OPENSTACK, 2023c). Além de possuir a funcionalidade de infraestrutura como serviço, ele possui componentes adicionais que permitem orquestrar, gerenciar falhas e

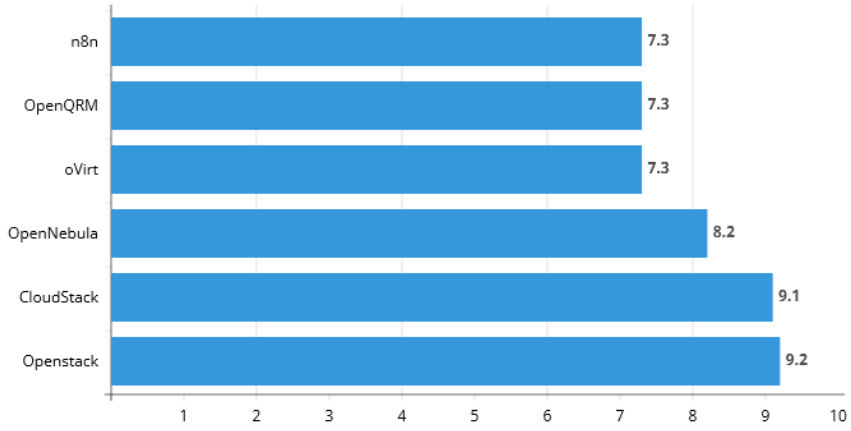


Figura 8 – Melhores softwares IaaS de código aberto, adaptado de (LINUXLINKS, 2020)

serviços, para garantir a alta disponibilidade de aplicativos do usuário.

Para isso, o OpenStack é dividido em serviços que permitem componentes sejam conectados dependendo da necessidade do usuário. Essa lista, que contêm mais de 50 serviços, pode ser visualizada na Figura 9, que contêm todos os serviços nativos do OpenStack, separados por funcionalidade. Para que o trabalho não fique muito extenso, optou-se por explicar somente os componentes que foram utilizados.

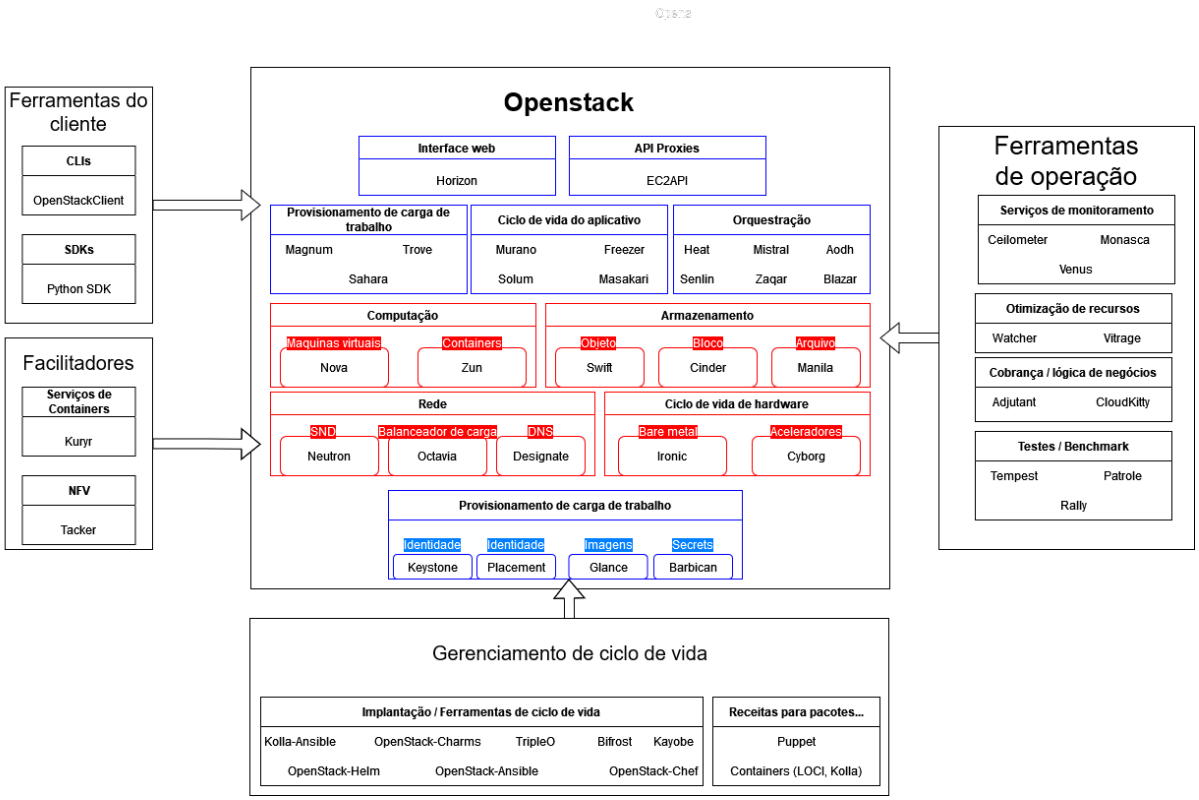


Figura 9 – Serviços OpenStack, adaptado de (OPENSTACK, 2023c)

4.2.1 Instalação

A instalação da plataforma OpenStack pode ser realizada de diversas maneiras, mas por conta da documentação oficial ser de fácil entendimento, optou-se pela instalação seguindo-a. A plataforma foi instalada em duas máquinas Ubuntu 20.04 LTS, um nó de controle, Desktop, e outro nó de computação e armazenamento compartilhado, Server. Os requisitos para os nós de controle, armazenamento e rede estão listados na Tabela 4.

Tabela 4 – Requisitos de hardware recomendados

Nó	**CPU**	**RAM**	**Armazenamento**	**NIC**
Controlador	1-2	8 GB	100 GB	2
Computação	2-4+	8+ GB	100+ GB	2
Armazenamento	1-2	4 GB	100+ GB	1

Fonte: (OPENSTACK, 2023b)

Porém para um ambiente de prova de conceito, utilizando instâncias CirrOS, os requisitos passam a ser os descritos na Tabela 5, o que torna a instalação da plataforma possível dentro do GNS3.

Tabela 5 – Requisitos para uma prova de conceito

Nó	**CPU**	**RAM**	**Armazenamento**
Controlador	1	4 GB	5 GB
Computação	1	2 GB	10 GB

Fonte: (OPENSTACK, 2023a)

4.2.2 Componentes instalados

Para a instalação dos componentes do OpenStack é preciso atentar-se que alguns componentes possuem pré-requisitos outros componentes para serem instalados, como por exemplo, o componente de computação Nova depende dos componentes Keystone, Neutron, Glance e Placement. Dito isso, a ordem de instalação dos componentes, para que não ocorresse problemas de dependências, foi:

1. **Keystone:** é o serviço OpenStack que fornece autenticação de API cliente, descoberta de serviço e autorização multilocatário distribuída implementando a API de identidade do OpenStack.
2. **Horizon:** é o serviço que fornece, via web, uma interface gráfica (dashboard) e um painel de serviço de forma a facilitar a interação com os demais serviços OpenStack.
3. **Neutron:** é um projeto OpenStack responsável por gerenciar a rede e fornecer *conectividade de rede como serviço* entre os dispositivos de interface que são gerenciados por outros serviços.

4. **Manila:** é o serviço que fornece aos sistemas o serviço de arquivos compartilhados. Tem como objetivos ter uma arquitetura baseada em componentes, ser altamente disponível, ser tolerante a falhas, ser recuperável e possuir padrões abertos.
5. **Glance:** é o projeto que fornece o serviço de imagem para que os usuários possam realizar o upload dados que podem ser usados com outros serviços, isso inclui imagens e definições de metadados.
6. **Placement:** é o serviço que fornece uma API HTTP para rastrear inventários e usos de recursos de nuvem para ajudar outros serviços a gerenciar e alocar seus recursos.
7. **Nova:** responsável pelo serviço computacional responsável por fornecer uma maneira de provisionar instâncias de computação, oferecendo o suporte para a criação de máquinas virtuais

Na Figura 10 é possível ver o resultado da consulta ao servidor no qual está instalado os serviços fornecidos.

```
osboxes@osboxes:~$ openstack service list
+----------------------------------+-----------+-----------+
| ID                               | Name      | Type      |
+----------------------------------+-----------+-----------+
| 09a0ceb7c4ac463a8e87fbc88096e7bb | manila    | share     |
| 246152e7833a46e59d7abea752b5afd1 | manilav2  | sharev2   |
| 300181c3031d41c59ad7ded04aa06d98 | placement | placement |
| 477bf5a35e514656bc2a1617c7f6f8a2 | neutron   | network   |
| d35bac7c451b41aca449f8780dc6e6c6 | glance    | image     |
| dcdea64f4fd84ac39bbbff48809a3073 | keystone  | identity  |
| e77d8b33a68e413bae75a96c673d5e2e | nova      | compute   |
+----------------------------------+-----------+-----------+
```

Figura 10 – Consulta para saber os componentes instalados

Fonte: Autor.

Por meio do dashboard, fornecido pelo componente Horizon, através da conta de administrador, é possível ver os serviços e suas respectivas configurações, como urls e portas, como mostra a Figura 11.

4.2.3 Configuração dos nós controle e computação

No nó de controle foi configurado todos os serviços listados anteriormente e possui endereço de IP fixo 192.168.1.105, como mostra a Figura 12, no qual o cliente cadastrado pode acessar uma tela de login para poder provisionar um serviço.

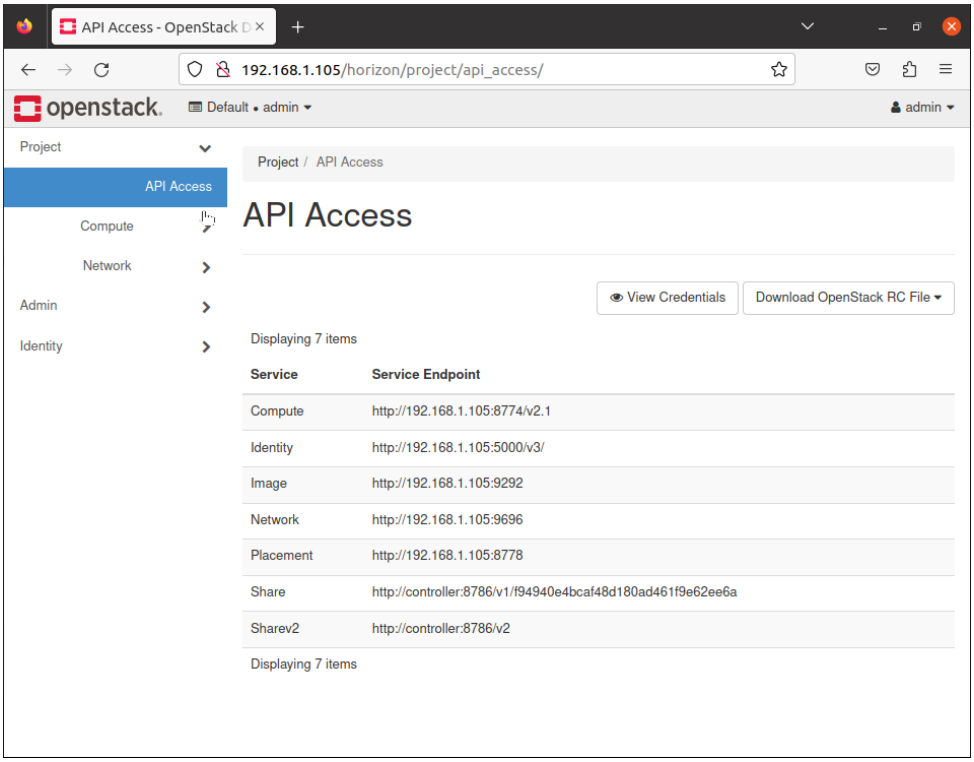


Figura 11 – Consulta através do dashboard para saber os componentes instalados

Fonte: Autor.

```
osboxes@osboxes:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 0c:99:f4:18:00:00 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 192.168.1.105/24 brd 192.168.1.255 scope global dynamic noprefixroute ens3
       valid_lft 4873sec preferred_lft 4873sec
    inet6 fe80::d9d4:48c:a95f:218c/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Figura 12 – Endereço IP do nó de controle

Fonte: Autor.

O nó de computação foi configurado para ter um endereço de IP fixo de 192.168.1.106, como mostra a Figura 13. Essa máquina é responsável de provisionar os serviços dos usuários.

4.3 FUNCIONAMENTO

Aqui será explicado como ocorreu o acesso do usuário cliente e do usuário malicioso.

Figura 13 – Endereço IP do nó de computação

Fonte: Autor.

4.3.1 Acesso do cliente

O cliente, que se encontra na sub rede 3, 172.21.2.128/26, como mostra a Tabela 2, possui uma máquina Windows 10 e está configurada com endereço IP fixo 172.21.2.130, como mostra a Figura 14. Esse usuário é esperado que só consiga o acesso ao sistema de gerenciamento de nuvem, caso possua uma conta, usuário e senha, já configurada por um administrador na plataforma, caso contrário, seu acesso fica restrito somente a página de *login* da página de dashboard fornecida pelo componente do OpenStack Horizon.

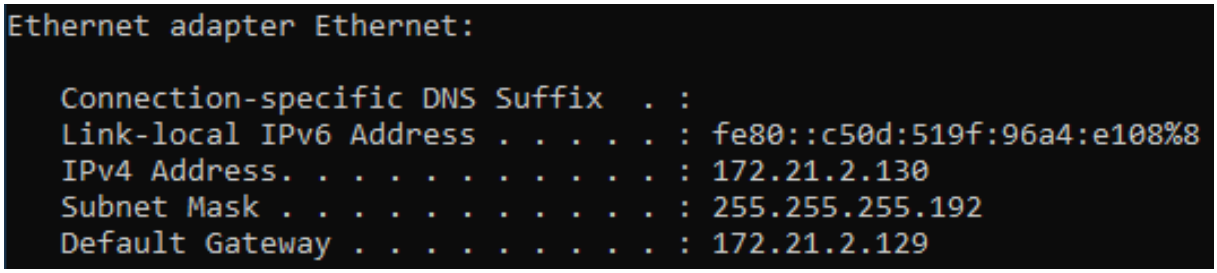


Figura 14 – Endereço IP do usuário cliente

Fonte: Autor.

Após as análises de segurança seu acesso passou a ser somente permitido através de uma conexão via VPN, como mostra a Figura 15.

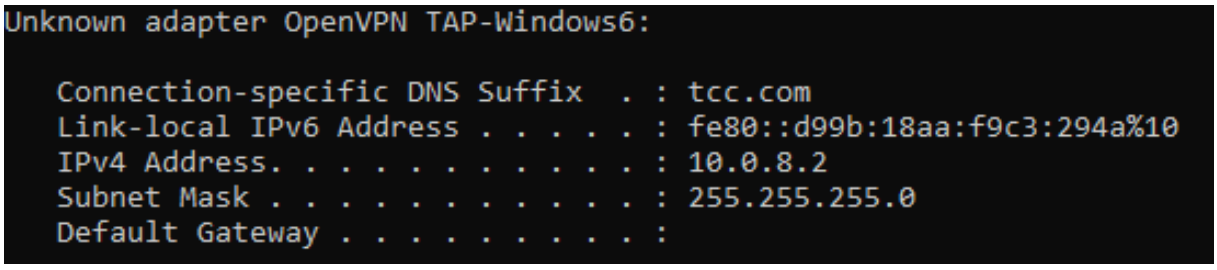


Figura 15 – Endereço IP do túnel UDP no usuário cliente

Fonte: Autor.

4.3.2 Acesso do usuário malicioso

O usuário malicioso, que se encontra na sub rede 4, 172.21.2.192/26, como mostra a Tabela 2, possui uma máquina Kali Linux e está configura com o endereço IP fixo 172.21.2.194, como mostra a Figura 16. Inicialmente, tirando o fato de o usuário malicioso não possuir uma conta cadastrada, era possível que ele acessasse a página de login, porém, com a configuração do acesso restrito à VPN, ele perdeu o acesso a essa página.

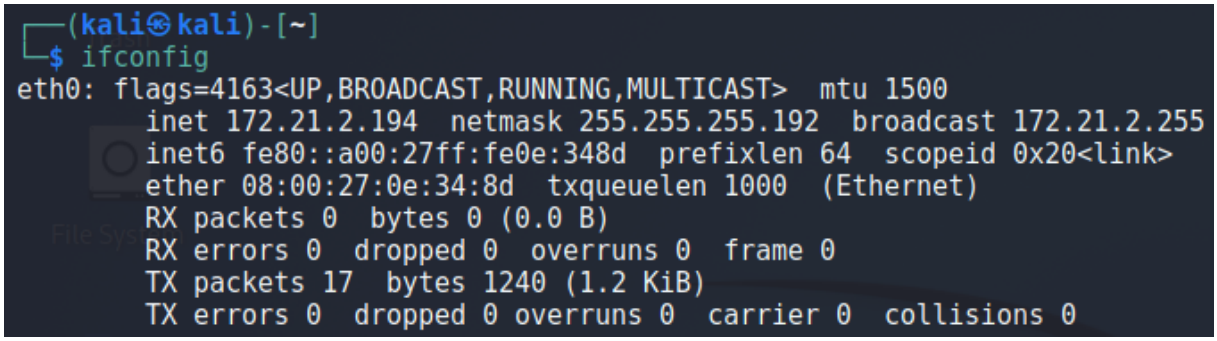


Figura 16 – Endereço IP do usuário malicioso

Fonte: Autor.

4.4 SUPERFÍCIE DE ATAQUE

Com a finalidade de demonstrar a superfície de ataque do sistema, foi utilizado o framework Metasploit em sua versão 6.3.0-dev. O Metasploit é utilizado por profissionais de segurança e pesquisadores de cibersegurança para fazer testes de penetração e avaliar a segurança de sistemas e redes. Fornece diversos exploits prontos para usar e uma plataforma para desenvolver e executar exploits personalizados. Ademais, o Metasploit pode ser integrado com ferramentas de segurança, como scanners de vulnerabilidades e sistemas de gerenciamento de vulnerabilidades. A documentação da ferramenta se encotram e seu site[1].

Para demonstrar um ataque, devido a popularidade deste tipo de ameaça atualmente, foi escolhido o ataque de negação de serviço. Para executar esse ataque, foi utilizado a ferramenta SlowHTTPTest que, por meio do prolongamento de diversas conexões HTTP que utilizam diferentes técnicas, permite a simulação de um ataque de negação de serviço na camada de aplicação. A documentação da ferramenta é possível encontrar em seu site[2].

[1] https://www.kali.org/tools/metasploit-framework/
[2] https://www.kali.org/tools/slowhttptest/

5 RESULTADOS E ANÁLISES

Neste capítulo será apresentado os resultados das análises das configurações já apresentadas no Capítulo 4, tanto para o cenário que possui VPN quanto para o cenário no qual a VPN foi configurada.

A organização desse capítulo está da seguinte maneira: a Seção 5.1 apresentará os resultados das análises feitas no cenário sem VPN configurada e a Seção 5.2 abordará os resultados das análises realizadas para o cenário com a VPN configurada.

5.1 CENÁRIO 1: COM VPN

Como mostra a Figura 17 e mencionado anteriormente, o firewall permitia conexões TCP de qualquer lugar para os endereços na LAN, a rede DMZ. Por causa dessa regra, tanto o usuário cliente quanto o usuário malicioso só conseguiam acessar o serviço por meio de uma uma requisição web, como mostra a Figura 18 e Figura 19.

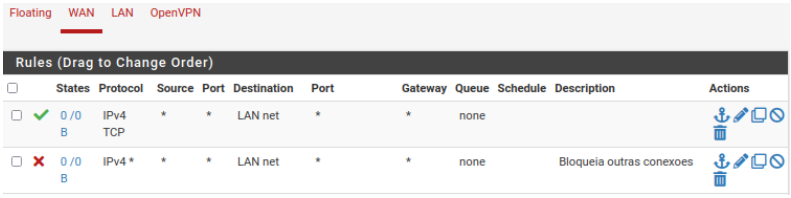


Figura 17 – Regra firewall para o cenário sem VPN

Fonte: Autor.

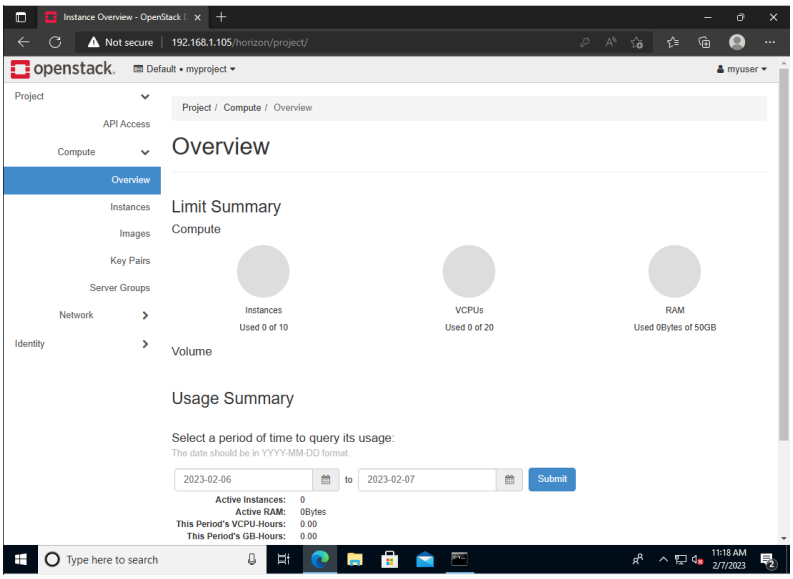


Figura 18 – Usuário cliente acessando a plataforma via web.

Fonte: Autor.

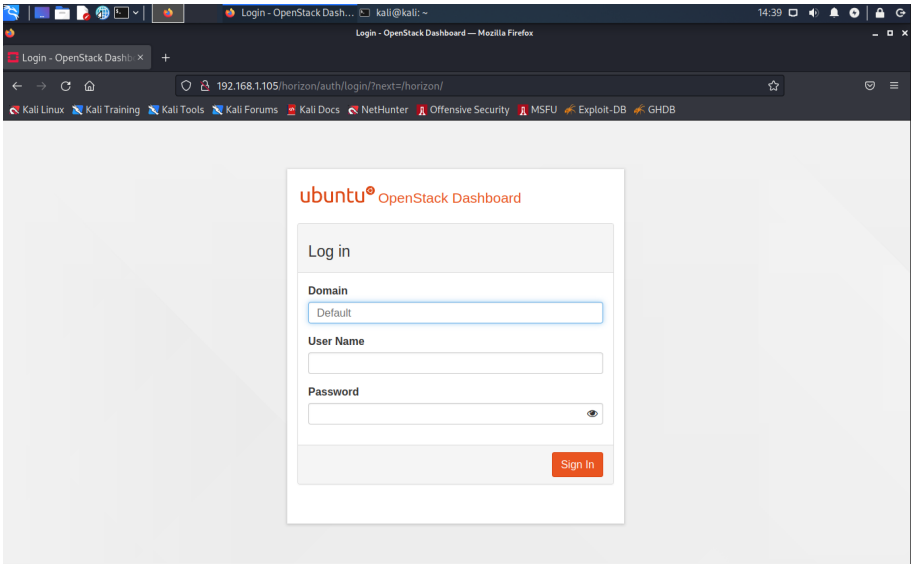


Figura 19 – Usuário malicioso acessando a plataforma via web.

Fonte: Autor.

A diferença entre as duas figuras é que na Figura 18 o usuário está em sua conta, pois é um usuário que possui uma conta cadastrada na plataforma, e na Figura 19 o usuário malicioso está na tela de login, pois não é um usuário autorizado, portanto, não possui conta cadastrada na plataforma.

Como mencionando anteriormente, inicialmente seu acesso só era permitido através do protocolo TCP, ou seja, tanto o cliente quanto o usuário malicioso não possuem permissão de enviar protocolos ICMP através do comando *ping* ao servidor, como mostra a Figura 20 e a Figura 21.

```
C:\Users\IEUser>ping 192.168.1.105

Pinging 192.168.1.105 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.105:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Figura 20 – Ping entre usuário cliente e nó de controle.

Fonte: Autor.

5.1.1 Mapeamento da rede e serviços

Primeiramente em um cenário de ataque, o invasor realiza uma varredura da superfície de ataque do sistema alvo. Pelo framework Metasploit foi possível realizar a varredura da rede pelo comando *db_nmap*, no qual realiza o escaneamento da rede através

```
┌──(kali㉿kali)-[~]
└─$ ping -c 4 192.168.1.105
PING 192.168.1.105 (192.168.1.105) 56(84) bytes of data.

--- 192.168.1.105 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3053ms
```

Figura 21 – Ping entre usuário malicioso e nó de controle.

Fonte: Autor.

das portas abertas do alvo. Após realizar o comando com o nó de controle como alvo, foi obtido o resultado mostrado na Figura 22. É possível notar serviços como SSH na porta 22, HTTP na porta 80 e MySQL na porta 3306. Em cada serviço é possível ver a ferramenta e sua versão adotada. Também é possível notar portar dos serviços que os componentes do Openstack utilizam como a porta 9696, utilizado pelo componente Neutron, e a porta 8774, utilizado pelo componente Nova.

```
Services
========

host            port   proto  name            state  info
----            ----   -----  ----            -----  ----
192.168.1.105   22     tcp    ssh             open   SSH-2.0-OpenSSH_8.2p1 Ubuntu-4ubuntu0.5
192.168.1.105   80     tcp    http            open   Apache/2.4.41 (Ubuntu)
192.168.1.105   3306   tcp    mysql           open   5.5.5-10.3.37-MariaDB-0ubuntu0.20.04.1
192.168.1.105   4369   tcp    epmd            open
192.168.1.105   5000   tcp    upnp            open
192.168.1.105   5672   tcp    amqp            open
192.168.1.105   6080   tcp    gue             open
192.168.1.105   8774   tcp                    open
192.168.1.105   8775   tcp                    open
192.168.1.105   8778   tcp    uec             open
192.168.1.105   9292   tcp    armtechdaemon   open
192.168.1.105   9696   tcp                    open
192.168.1.105   25672  tcp                    open
```

Figura 22 – Escaneamento da rede.

Fonte: Autor.

Um semelhança sobre os serviços listados é que todos rodam sobre o protocolo TCP, devido a regra no firewall, no qual só permitiu que encontrasse serviços utilizando esse protocolo. Essa análise sobre os serviços fornece ao invasor variedades de explorar venerabilidades que possam estar presente no sistema.

5.1.2 Ataque de negação de serviço

Com a finalidade de demonstrar uma das possíveis variedades de vulnerabilidade, foi realizado um ataque de negação de serviço sobre o serviço web na porta 80 utilizando a ferramenta SlowHTTPTest.

Utilizando essa ferramenta deixar o serviço indisponível por alguns segundos, como mostra a Figura 23 e Figura 24. No comando foi configurado para que o ocorresse 1000 conexões, no modo *SLOW HEADERS*, com intervalo entre os dados de 10 segundos, com 200 conexões por segundo, com o alvo "http://192.168.1.105/horizon/auth/login/?next=/horizon/", com o tamanho máximo de 24 bytes e com um timeout de 3 segundos.

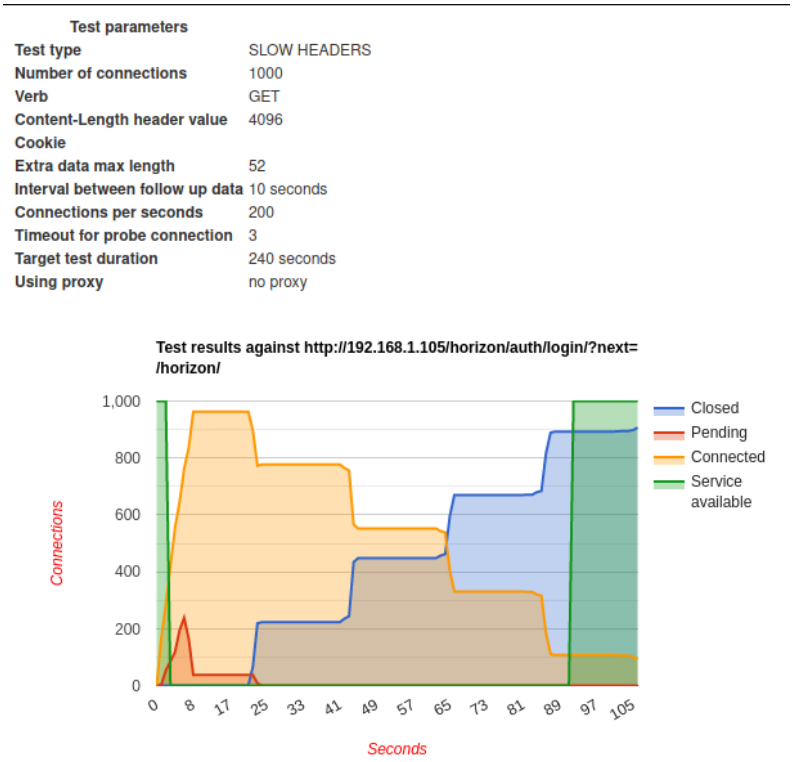


Figura 23 – Resultado do teste de negação de serviço

Fonte: Autor.

Pela Figura 23 é possível ver de forma cronológica como aconteceu o ataque. Nos primeiros oito segundos ocorreu um grande número de requisição ao alvo, que não suportou o número de acessos e ficou indisponível. Algumas conexões ficaram como pendentes, mas que logo foram aceitas, tornando o sistema ainda mais indisponível. Com o passar o tempo, essas conexões que foram estabelecidas começaram a ser fechadas, invertendo a quantidade de conexões estabelecidas com a quantidade de conexões fechadas. Porém, mesmo com a inversão das duas curvas, demorou cerca de 30 segundos para que o sistema voltasse a ficar disponível.

A Figura 24 mostra a saída do comando *slowhttptest* para aos 20 segundos do teste. É possível notar que nesse momento o ataque conseguiu realizar 963 conexões bem sucedidas, 37 conexões pendentes e o sistema já se encontrava interrompido.

Realizando esse simples teste, o sistema ficou fora do ar por volta de 1 minuto e 20 segundos, porém, em um cenário real, onde dezenas de invasores possuem centenas de máquinas que realizam milhares de ataques como esse, um sistema pode ficar horas ou até dias indisponível.

5.1.3 Visão do sistema

Como foi configurado um sistema de detecção e prevenção de intrusão, foi possível identificar o escaneamento da rede e as conexões provenientes do ataque de negação de

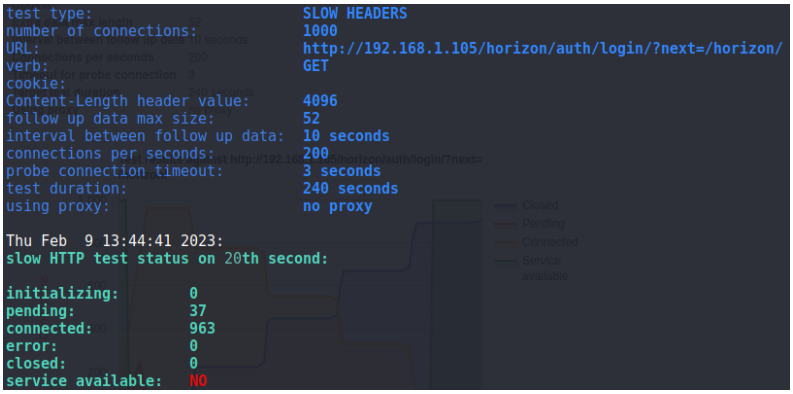


Figura 24 – Resultado do comando *slowhttptest*.

Fonte: Autor.

serviço, como mostra a Figura 25 e a Figura 26. A Figura 25 mostra alguns logs de alerta gerados pela varredura de rede do Metasploit, já a Figura 26 mostra em específico o escaneamento do serviço MySQL na porta 3306.

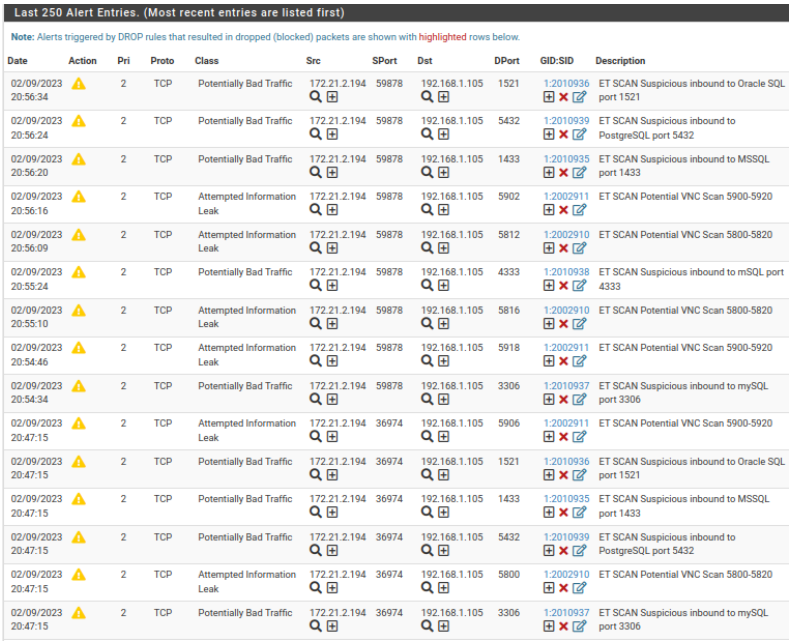


Figura 25 – Log de detecção de varredura de rede do Suricata.

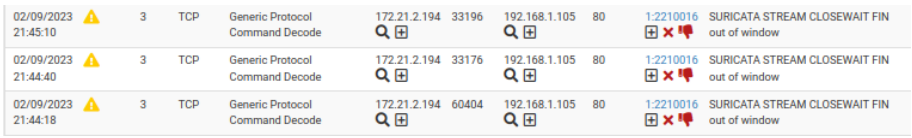
Fonte: Autor.

Figura 26 – Log de detecção de varredura de rede do serviço MySQL do Suricata.

Fonte: Autor.

Também foi possível identificar tráfegos suspeitos provenientes do ataque de negação de serviço, como mostra a Figura 27. Nessa imagem é possível ver que o Suricata alertou

sobre um tráfego suspeito vindo do usuário malicioso pelas portas 33196, 33176 e 60404 para o nó de controle na porta 80.



02/09/2023 21:45:10	3	TCP	Generic Protocol Command Decode	172.21.2.194	33196	192.168.1.105	80	1:2210016	SURICATA STREAM CLOSEWAIT FIN out of window
02/09/2023 21:44:40	3	TCP	Generic Protocol Command Decode	172.21.2.194	33176	192.168.1.105	80	1:2210016	SURICATA STREAM CLOSEWAIT FIN out of window
02/09/2023 21:44:18	3	TCP	Generic Protocol Command Decode	172.21.2.194	60404	192.168.1.105	80	1:2210016	SURICATA STREAM CLOSEWAIT FIN out of window

Figura 27 – Log de detecção do ataque DoS.

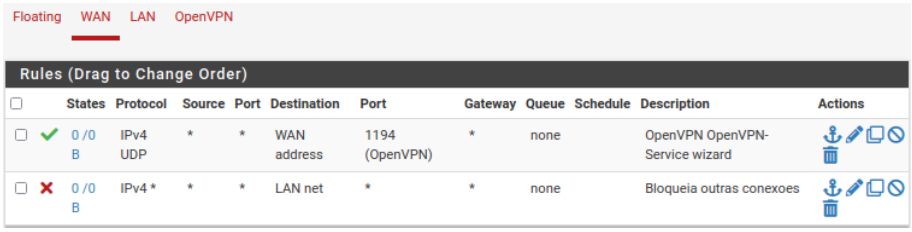
Fonte: Autor.

Com esses alertas o administrador de rede pode avaliar e tomar a melhor decisão sobre o que fazer com o tráfego, podendo bloquear a origem, o destino, ambos e/ou próximos tráfegos de redes semelhantes.

5.2 CENÁRIO 2: COM VPN

Neste cenário foi configurado uma VPN, na qual somente os tráfegos dos hosts que estiverem conectados a ela poderiam acessar o serviço da plataforma Openstack. Resumidamente, o usuário cliente se conectava diretamente a rede DMZ e consumia o serviço de nuvem.

Como mostra a Figura 28, só será permitido conexões na interface WAN por meio do protocolo UDP na porta 1194 e qualquer outro tipo de conexão será bloqueada.



Floating WAN LAN OpenVPN

Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
☐ ✓	0 /0 B	IPv4 UDP	*	*	WAN address	1194 (OpenVPN)	*	none		OpenVPN OpenVPN-Service wizard	
☐ ✗	0 /0 B	IPv4 *	*	*	LAN net	*	*	none		Bloqueia outras conexoes	

Figura 28 – Regra da interface WAN para conexão VPN.

Fonte: Autor.

Na Figura 29 é possível ver a regra feita para a interface na qual é responsável pela conexão VPN. A regra configurada permite que a conexão por meio da VPN tenha total acesso a rede DMZ, como mostra Figura 30, na qual o cliente conectado a VPN consegue enviar mensagens ICMP, pelo comando *ping*, para o nó de controle.

Para o usuário malicioso o cenário mudou de forma significativa. Ainda não é possível enviar mensagens ping para o nó de controle, porém seu acesso a plataforma de login foi bloqueado, como mostra a Figura 31, pois ele não está conectado a VPN.

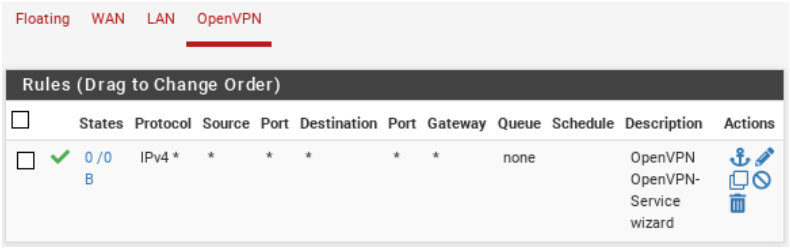


Figura 29 – Regra da interface OpenVPN.

Fonte: Autor.

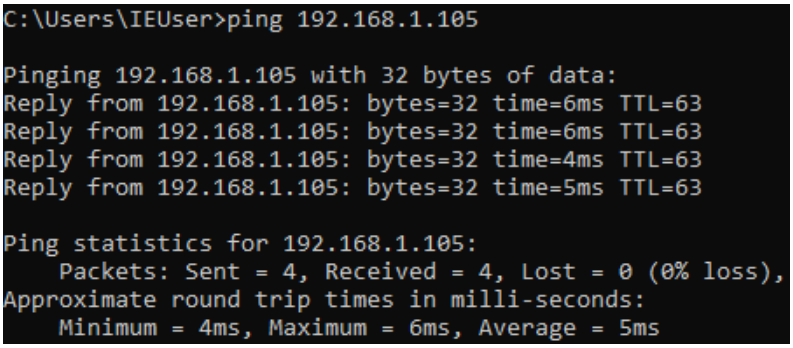


Figura 30 – Ping do usuário cliente conectado a VPN para o nó de controle.

Fonte: Autor.

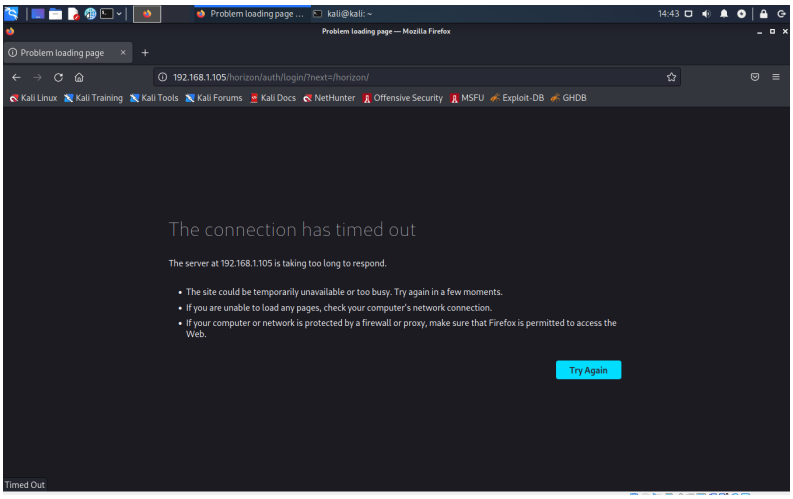


Figura 31 – Usuário malicioso acessando a plataforma via web sem conexão VPN

Fonte: Autor.

5.2.1 Mapeamento da rede e serviços

Assim como no cenário sem a conexão VPN, foi realizado uma varredura na rede pelo serviços disponíveis na rede do alvo. Utilizadno o mesmo framework Metasploit, foi realizado um escaneamento utilizando o comando *db_namp*. Após o comando com o nó de controle como alvo, foi obtido o resultado mostrado na Figura 32. Como esperado, é

possível notar que somente o serviço VPN aparece no mapeamento da rede pela porta configurada, a 1194. Isso deve pela regra do firewall, Figura 28, que bloqueia qualquer tráfego de rede que chega pela interface WAN que não seja pelo protocolo UDP na porta 1194.

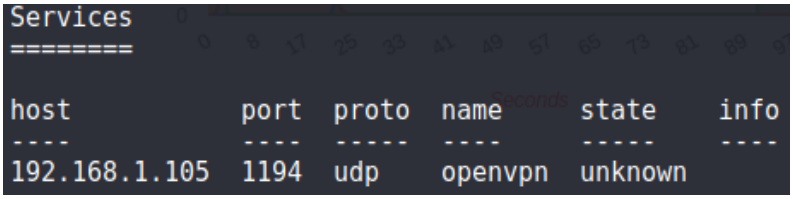


Figura 32 – Escaneamento da rede com VPN.

Fonte: Autor.

Com isso, é possível notar que houve um redução da superfície de ataque disponível para um usuário malicioso explorar vulnerabilidades que possam comprometer o sistema. Tornando, assim, um cenário mais de mais difícil acesso e menos propenso para um atacante realizar ações maliciosas.

5.2.2 Ataque de negação de serviço

Para demonstrar um avanço na segurança do sistema com a redução da superfície de ataque, foi realizado o mesmo ataque de negação de serviço feito no cenário sem VPN configurada, com os mesmos parâmetros.

A Figura 33 mostra que o comando *slowhttptest* teve um resultado no qual não foi possível estabelecer conexão. Isso ocorreu pelo fato de a porta 80, estar sendo filtrada pelo firewall, não permitindo que qualquer tráfego que ocorra por essa porta chege a rede DMZ.

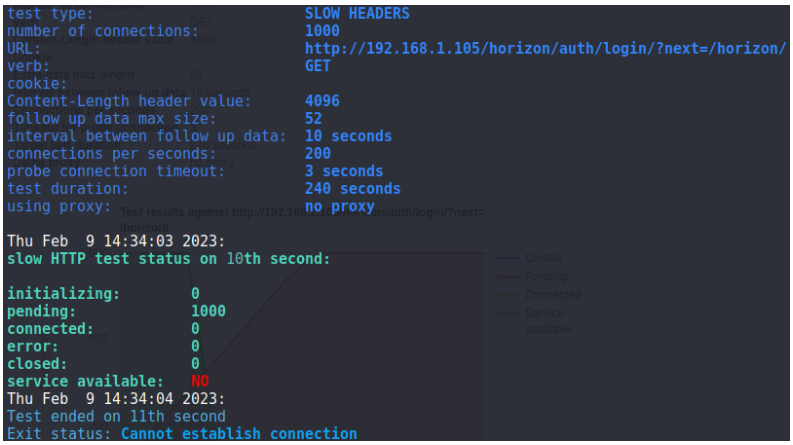


Figura 33 – Resultado do comando *slowhttptest* no cenário com VPN.

Fonte: Autor.

A Figura 34 mostra a cronologia do ataque, que só durou 11 segundos. Nos primeiros 6 ocorreu um grande fluxo de dados em direção ao alvo, que não obtiveram sucesso em sua conexão, permanecendo com o status de pendente até o final do teste. Como o acesso ao serviço foi negado, o comando assumiu que o serviço já se encontrava fora do ar, por isso entre o segundo 2 e 3 mudou o status do sistema para indisponível.

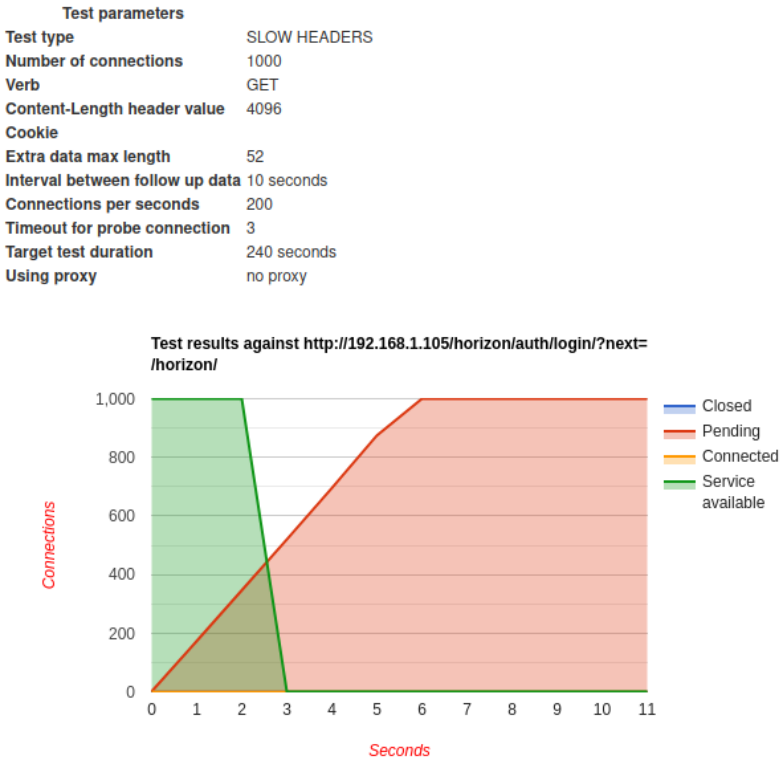


Figura 34 – Resultado do teste de negação de serviço no cenário com VPN.

Fonte: Autor.

É notório que a configuração da VPN mudou completamente o cenário para um atacante e por isso que vem sendo adotada em larga escala para serviços sensíveis e para trabalhos remotos, que estão se tornando cada vez mais comuns em um mundo pós-pandemia.

5.2.3 Visão do sistema

Para o sistema de detecção e prevenção de intrusão, Suricata, o mapeamento de rede no qual foi descoberto a porta UDP 1194 não ocorreu, ou seja, ele foi incapaz de identificar um tráfego suspeito na rede em busca do serviço. Também, como esperado, não apresentou nenhum alerta sobre o ataque de negação de serviço de serviço, porque, de fato, ele não aconteceu, devido as requisições serem bloqueadas pelo firewall.

5.3 COMPARAÇÕES

É notório que no primeiro cenário, no qual não há uma conexão VPN, a superfície de ataque é maior, tendo 13 portas identificadas, o que fornece a um atacante um grande número de possibilidades para explorar vulnerabilidades ligadas aos serviços ligados a essas portas. Porém, no cenário no qual há um conexão VPN a superfície de ataque é bem menor, tendo somente uma porta identificada, fornecendo uma opção limitada de exploração de vulnerabilidade ligada a porta para um possível atacante.

Ademais, o estabelecimento da regra de conexão somente por VPN tirou o trabalho de identificar, bloquear e alertar do IDS/IPS, se tornando mais um mecanismo de contensão contra falhas mais localizadas no sistema.

Por fim, a conexão VPN fornece segurança tanto para o sistema saber que está sendo acessado por um usuário confiável, quanto para o usuário que acessa um sistema protegido.

6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Esse trabalho se propôs em estudar uma maneira de implementar uma infraestrutura segura de nuvem privada. Tendo em consideração o sucesso da topologia apresentada no Capítulo 4 e os resultados obtidos nas análises efetuadas durante o Capítulo 5, concluise que o objetivo geral e os objetivos específicos desse trabalho de conclusão de curso foram devidamente satisfeitos.

É notório que o uso do firewall permitiu um isolamento à rede DMZ, permitindo um maior controle e gerenciamento do tráfego com destino a rede isolada. As regras configuradas nas interfaces e a solução adotada como firewall, o pfSense, se mostraram bastante robustas, escaláveis e eficazes com os testes realizados.

A plataforma Openstack se mostrou um plataforma de simples configuração, documentação fácil e componentes que atendem seu propósito proposto, o que justifica sua ampla adoção em ambientes reais.

O uso de um sistema de detecção e prevenção de intrusão se provou eficaz, visto que, no cenário sem a conexão VPN, conseguiu identificar os tráfegos de escaneamento de rede e conexões provenientes que um ataque de negação de serviço. O que demonstrou sua grande força em ambientes reais de redes de computadores.

A conexão VPN demonstrou que o seu uso é de bastante importância quando o assunto é comunicação segura. Demonstrou-se uma excelente alternativa para ambientes no qual é preciso realizar um controle de acesso, pois tira a sobrecarga dos IDSs e IPSs de analisar todo o tráfego que entra em uma rede, poupando muito trabalho computacional do firewall. Ela unifica o meio de acesso a uma rede em cima de um protocolo e um meio criptografado, tornando um meio que antes era inseguro, agora, seguro.

Por fim, independentemente dos resultados positivos obtidos, ainda é possível realizar a mesma infraestrutura de arquitetura de serviços para estudar outros meios de proteção em um ambiente seguro. Aumentar a complexidade da rede, criando sub redes e gerenciar o acesso das outras sub rede a rede DMZ. Utilizar outras soluções e estratégias para se obter um meio seguro de comunicação entre as partes e prevenir ataques cibernéticos. Além disso, é possível aplicar esse estudo em um ambiente real, com serviços e servidores reais, para atestar e comprovar a eficácia da proposta de infraestrutura aqui apresentada.

REFERÊNCIAS

CLOUD COMPUTING PROJECTS. **Cloud Computing Open Source Projects**. [*S.l.*]. Disponível em: https://cloudcomputingprojects.net/cloud-computing-open-source-projects/. Acesso em: 29 jan. 2023.

COULOURIS, George; DOLLIMORE, Jean; KINDBERG, Tim. **Sistemas Distribuídos**: Conceitos e Projeto. 4. ed. [*S.l.*]: Bookman, 2007.

EMMS, Steve. **Best Free and Open Source IaaS Software**. [*S.l.*], 2020. Disponível em: https://www.linuxlinks.com/iaas/. Acesso em: 29 jan. 2023.

KANDUKURI, Balachandra Reddy; PATURI, Ramakrishna; RAKSHIT, Dr. Atanu. **Cloud Security Issues**. Pune, India: IEEE International Conference on Services Computing, 2009. P. 4.

KRUTZ, Ronald L.; VINES, Russell Dean. **Cloud Security**: A Comprehensive Guide to Secure Cloud Computing. 2. ed. Indianaplois, Indiana: Wiley Publishing, Inc., 2010.

KUROSE, James F.; ROSS, Keith W. **Redes de computadores e a Internet**: uma abordagem top-down. 6. ed. São Paulo: Pearson Education do Brasil, 2013.

MELL, Peter; GRANCE, Timothy. **The NIST Definition of Cloud Computing**: Recommendations of the National Institure of Standards and Technology. [*S.l.*], 2011. Disponível em: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf. Acesso em: 22 dez. 2022.

MOLINA, Denison; SILVEIRA, Sidnei Renato; SANTOS, Fernando Veux dos. **Implantação de Um Ambiente de Segurança de Redes de Computadores: Um Estudo de Caso na Prefeitura Municipal de Palmeira das Missões - RS**. Rio Grande do Sul: [*s.n.*], 2015. P. 41.

OPENSTACK. **Environment**. [*S.l.*]. Disponível em: https://docs.openstack.org/install-guide/environment.html. Acesso em: 29 jan. 2023.

OPENSTACK. **Overview**. [*S.l.*]. Disponível em: https://docs.openstack.org/install-guide/overview.html. Acesso em: 29 jan. 2023.

OPENSTACK. **What is openstack?** [*S.l.*]. Disponível em: https://www.openstack.org/software/. Acesso em: 29 jan. 2023.

ROSADO, David G.; GÓMEZ, Rafael; MELLADO, Daniel; FERNÁNDEZ-MEDINA, Eduardo. **Security Analysis in the Migration to Cloud Environments**. Business Intelligence Architect, Alpha Clinical Systems Inc, USA: Future Internet, 2012. P. 19.

SANTOS, Rafael César Merlo dos. **Implantação de Infraestrutura como Serviço em uma Nuvem Computacional Privada**. Brasília: Universidade de Brasília, 2016. P. 76.

SARMAH, Simanta Shekhar. **Cloud Migration - Risks and Solutions**. Business Intelligence Architect, Alpha Clinical Systems Inc, USA: Science e Technology, 2019. P. 5.

SILVA GAMA JÚNIOR, Lúcio da. **Virtualização de Funções de Rede em Nuvem para Instituições Públicas**. São Cristóvão, Sergipe: UNIVERSIDADE FEDERAL DE SERGIPE, 2017. P. 87.

SILVA LINZ, Kennedy Bezerra da. **Estudo para implementação de uma nuvem híbrida na Agência Estadual de Meio Ambiente de Pernambuco -CPRH/PE**. Recife, Pernambuco: Universidade Federal do Sergipe, 2020. P. 55.

TANENBAUM, Andrew S.; STENN, Maarten Van. **Sistemas Distribuídos**: Princípios e paradigmas. 2. ed. [*S.l.*]: Pearson, 2007.

WILLIAMS, Alexander. **The Top Open Source Cloud Projects of 2014**. [*S.l.*], 2014. Disponível em: https://www.linux.com/news/top-open-source-cloud-projects-2014/. Acesso em: 29 jan. 2023.

YOUTECHDIET. **Top 7 Open Source IaaS Platforms**. [*S.l.*]. Disponível em: https://yourtechdiet.com/blogs/open-source-iaas-platforms/. Acesso em: 29 jan. 2023.