FACULDADE DE TECNOLOGIA ALCIDES MAYA - AMTEC
CURSO TECNOLÓGICO EM REDES DE COMPUTADORES

DANIEL ROBERTO MENDES

INFRAESTRUTURA COMO SERVIÇO: IMPLEMENTAÇÃO DE OPENSTACK COMO SOLUÇÃO PARA NUVENS PRIVADAS

Porto Alegre
2019

DANIEL ROBERTO MENDES

INFRAESTRUTURA COMO SERVIÇO: IMPLEMENTAÇÃO DE OPENSTACK COMO SOLUÇÃO PARA NUVENS PRIVADAS

Projeto de Pesquisa apresentado como requisito parcial para obtenção do título de Tecnólogo em Redes, pelo Curso de Tecnólogo em Redes de Computadores da Faculdade de Tecnologia Alcides Maya - AMTEC

Orientador: Prof. Me. Anderson Santos

Porto Alegre

2019

LISTA DE TABELAS

Tabela 1. Princípio da segurança em computação em nuvem. 32
Tabela 2. Principais serviços do OpenStack. 43
Tabela 3. Componentes de serviço de rede do Neutron. 53

LISTA DE FIGURAS

Figura 1. Diagrama de rede (nuvem). 15
Figura 2. Visão geral de uma nuvem computacional.....16
Figura 3. Arquitetura Tradicional x Virtualização. 20
Figura 4. Quadro de responsabilidades entre cliente e fornecedor. 28
Figura 5. Diagrama de Chao-Kuei. 36
Figura 6. Lançamento das séries do OpenStack. 41
Figura 7. Pilares de serviços do OpenStack. 42
Figura 8. Serviços do OpenStack. 55
Figura 9. Descrição da solução. 57
Figura 10. Simulação de ataque SYN Flood. 58
Figura 11. Desenvolvimento da Nuvem Privada e a simulação de ataque. 60
Figura 12. Tempo de resposta instância cirros_1. 61
Figura 13. Tempo de resposta instância cirros_2. 62
Figura 14. Tempo de resposta Kali Linux para OpenStack. 62
Figura 15. Tempo de resposta ataque de negação de serviço. 63
Figura 16. Tempo de resposta ataque de força bruta. 64
Figura 17. Página inicial de instalação VMware. 79
Figura 18. Contrato de licença VMware. 79
Figura 19. Diretório de instalação VMware. 80
Figura 20. Configuração de experiência de uso VMware. 80
Figura 21. Configuração de atalho VMware. 81
Figura 22. Inicio de instalação VMware. 81
Figura 23. Instalação VMware concluída. 82
Figura 24. Configuração de diretório de instalação VM. 82
Figura 25. Configuração de sistema operacional VM. 83
Figura 26. Configuração de nome VM. 83
Figura 27. Configuração capacidade de disco VM. 84
Figura 28. Página de configuração mínima VM. 84
Figura 29. Página de customização de hardware VM. 85
Figura 30. Página inicial de instalação CentOS. 85

Figura 31. Configuração de idioma CentOS. 86
Figura 32. Página de configuração de instalação CentOS. 86
Figura 33. Seleção de Software CentOS. 87
Figura 34. Configuração disco virtual para instalação do CentOS. 87
Figura 35. Configuração placa de rede/nome de host CentOS. 88
Figura 36. Instalação CentOS. 88
Figura 37. Configuração de senha CentOS. 89
Figura 38. Dashboard OpenStack. 92
Figura 39. Configuração rede_privada_1. 93
Figura 40. Configuração sub-rede rede_privada_1. 94
Figura 41. Configuração detalhes sub-rede rede_privada_1. 94
Figura 42. Configuração rede_privada_2. 95
Figura 43. Configuração sub-rede rede_privada_2. 95
Figura 44. Configuração detalhes sub-rede rede_privada_2. 96
Figura 45. Configuração roteador. 96
Figura 46. Topologia de rede (roteador + rede_privada_1). 97
Figura 47. Adicionar interface ao roteador. 97
Figura 48. Topologia (roteador + rede_privada_1 + rede_privada_2). 98
Figura 49. Criação imagem. 98
Figura 50. Detalhes instância. 99
Figura 51. Origem instância. 99
Figura 52. Flavor instância. 100
Figura 53. Rede instância. 100
Figura 54. Console instância. 101
Figura 55. Teste de comunicação instância cirros_1. 101
Figura 56. Teste de comunicação instância cirros_2. 102
Figura 57. Topologia de rede modo gráfico. 102
Figura 58. Página inicial VMware. 103
Figura 59. Configuração de disco e imagem VMware. 103
Figura 60. Configuração de hardware VMware. 104
Figura 61. Página inicial de instalação Kali Linux. 104
Figura 62. Seleção de idioma Kali Linux. 105
Figura 63. Seleção de localidade Kali Linux. 105
Figura 64. Seleção de hostname Kali Linux. 106

Figura 65. Criação de senha de administrador Kali Linux. 106
Figura 66. Seleção de fuso horário Kali Linux. 107
Figura 67. Seleção de partição de disco Kali Linux. 107
Figura 68. Seleção de disco virtual. 108
Figura 69. Seleção de partição Kali Linux. 108
Figura 70. Demonstração de partições Kali Linux. 109
Figura 71. Formatação de partição Kali Linux. 109
Figura 72. Seleção de instalação Grub. 110
Figura 73. Seleção de partição de instalação Grub. 110
Figura 74. Finalização de instalação Kali Linux. 111
Figura 75. Teste de comunicação Kali Linux. 111
Figura 76. Comando para ataque de negação de serviço. 112
Figura 77. Ataque de negação de serviço (parte 1). 112
Figura 78. Ataque de negação de serviço (parte 2). 113
Figura 79. Comando ataque de força bruta tipo dicionário. 113
Figura 80. Ataque de força bruta (parte 1). 114
Figura 81. Ataque de força bruta (parte 2). 114

LISTA DE SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
NBR	Normas Brasileiras de Regulação
TIC	Tecnologia da Informação e Comunicação
SLA	Service Level Agreement
IBM	Internacional Business Machines
VM	Virtual Machine
TI	Tecnologia da Informação
SOA	Service Oriented Architecture
XML	Extensible Markup Language
NRDC	Natural Resources Defense Council
CPU	Central Processing Unit
API	Application Programming Interface
NBR	Normas Brasileiras de Regulação
GNU	GNU's Not Unix
FSF	Free Software Foundation
OSI	Open Source Initiative
HTTP	Hypertext Transfer Protocol
DoS	Denial of Service
LDAP	Lightweight Directory Access Protocol
AMQP	Advanced Message Queuing Protocol
CLI	Command LIne Interface
RPC	Remote Procedure Call
AWS	Amazon Web Services
TCB	Transmission Control Block
VLAN	Virtual Local Area Network
DHCP	Dynamic Host Configuration Protocol
VPN	Virtual Private Network
SDN	Software Defined Networking
URL	Uniform Resource Locator

ISSO	International Organization for Standardization
QEMU	Quick Emulator
QCOW	QEMU Copy On Write
TTL	Time to Live
FQDN	Fully Qualified Domain Name
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6

SUMÁRIO

1 INTRODUÇÃO **10**
1.1 Definição do Tema ou Problema **11**
1.2 Delimitações do Trabalho **12**
1.3 Objetivos **12**
1.3.1 Objetivo Geral 12
1.3.2 Objetivos Específicos 12
1.4 Justificativa **13**
2 REVISÃO BIBLIOGRÁFICA **14**
2.1 COMPUTAÇÃO EM NUVEM **14**
2.2 CONTEXTO HISTÓRICO **16**
2.3 MODELOS DE COMPUTAÇÃO EM NUVEM **17**
2.4 VIRTUALIZAÇÃO **19**
2.4.1 Tipos de Virtualização 23
2.4.2 Emuladores 23
2.4.3 Virtualização Total 24
2.4.4 Paravirtualização 24
2.4.5 Computação Verde **25**
2.5 SERVIÇOS DE COMPUTAÇÃO EM NUVEM **26**
2.6 SEGURANÇA NA COMPUTAÇÃO EM NUVEM **29**
2.6.1 Ataque de Negação de Serviço **33**
2.6.2 Ataque de Força Bruta **34**
2.7 SOFTWARE LIVRE **35**
2.8 VMWARE WORKSTATION PLAYER **39**
2.9 CENTOS **39**
2.10 OPENSTACK **40**
2.10.1 Serviço de Computação Nova **44**
2.10.2 Serviço de Identidade Keystone **45**
2.10.3 Serviço de Imagem Glance **46**
2.10.4 Serviço de Armazenamento de Objeto Swift **47**

2.10.5 Serviço de Armazenamento em Bloco Cinder 48
2.10.6 Serviço de Controle Horizon 48
2.10.7 Serviço de Telemetria Ceilometer 49
2.10.8 Serviço de Orquestração Heat 50
2.10.9 Serviço de Rede Neutron 52
3 DESCRIÇÃO DA SOLUÇÃO 55
4 METODOLOGIA 59
5 VALIDAÇÃO 61
6 CONCLUSÃO 65
7 CRONOGRAMA 67
REFERÊNCIAS BIBLIOGRÀFICA 68
APÊNDICE A – IMPLEMENTAÇÃO DE NUVEM PRIVADA E SIMULAÇÃO DE ATAQUE 79

1 INTRODUÇÃO

Uma tendência crescente no mundo de TI segundo VELTE (2012) é a virtualização de servidores. Isto é, o software pode ser instalado permitindo que vários servidores virtuais sejam usados. Desta maneira, pode se ter meia dúzia de servidores virtuais funcionando em um único servidor físico. Com isso diminuindo os custos operacionais e implementando uma computação em nuvem.

A computação em nuvem de acordo com BUYYA (2010) consiste em inúmeros fornecedores de TI que fornecem processamento, armazenamento e serviços de hospedagem de aplicativos. Oferecendo promessas de desempenho e disponibilidade garantidas por acordos de nível de serviço (SLA). Os clientes são cobrados com base em suas utilizações de recursos de processamento, armazenamento e transferência de dados. Oferecem acesso baseado via Web que são popularmente chamado de IaaS (Infraestrutura como Serviço), que permite a criação de máquinas virtuais, PaaS (Plataforma como serviço), que fornece ambiente para desenvolvimento e SaaS (Software como serviço), que agrega software na nuvem.

Segundo VELTE (2012) os fornecedores de serviços de computação em nuvem pública têm políticas de privacidade rigorosas e empregam medidas de segurança rígidas, como métodos comprovados de criptografia para autenticar os usuários. Este modelo de serviço traz uma série de desafios de segurança de acordo com GONZALEZ (2013), que precisam ser analisados e endereçados tanto a usuários quanto aos provedores de serviço. GONZALEZ (2013) complementa que esta falta de entendimento ás questões de segurança pode ocasionar em reflexos negativos para as empresas e para os usuários que fazem uso de tais serviços. Um exemplo dessa situação como problema de segurança na computação em nuvem pública, de acordo com CLULEY (2013), aconteceu o vazamento em fevereiro de 2013 das senhas e usuários do aplicativo Evernote, 50 milhões de usuários registrados no serviço foram impactados, tiveram que trocar sua senha.

Segundo BRANCO JR. (2014) para que todo o potencial da computação em nuvem possa ser explorado pelas empresas, são de fundamental importância à segurança e a privacidade dos dados armazenados na nuvem. A CLOUD SECURITY ALLIANCE (2013) produziu um relatório aonde lista as 10 maiores ameaças para a computação em nuvem. O relatório disponibiliza a informação de

que o primeiro lugar no ranking desta pesquisa esta o roubo de dados, seguido do segundo lugar como a perda de dados.

Este projeto tem como objetivo demonstrar uma solução de código aberto, que é a ferramenta OpenStack, para implementação de uma infraestrutura como serviço em nuvem privada, com objetivo de ter um ambiente seguro e controlável. Alguns dos benefícios desta ferramenta são o gerenciamento centralizado, gerenciamento de servidores físicos e virtualizados, implementação em qualquer datacenter com suporte a virtualização, compatibilidade com os principais modelos de máquinas virtuais e integração com ferramentas de gerenciamento de código livre.

1.1 Definição do Tema ou Problema

A Computação em Nuvem segundo VELTE (2012) funciona como uma plataforma que permite a todos utilizar muitas variedades de aplicações via internet, independente da plataforma ou lugar. Esta infraestrutura em nuvem de acordo com VELTE (2012), com o seu acesso via internet, é conhecida como nuvem pública e tem como característica, diminuir a quantidade de equipamentos instalados em uma empresa. De acordo com Conforme CONVERGÊNCIA DIGITAL (2016), a tendência ao uso da nuvem com a virtualização, aplicações e a concentração dos dados em datacenters remotos, transforma a nuvem num território promissor para invasões e vazamentos de dados.

Conforme CABRAL (2019) hackers divulgaram 2,2 bilhões de senhas, com centenas de gigabytes de informação pessoais de usuários de muitos serviços de nuvens públicas conhecidos como Yahoo, LinkedIn e Dropbox. CABRAL (2019) relata que este vazamento de informações privadas é considerado como o maior já verificado por muitos especialistas na área de segurança, onde na maioria das vezes em que isto ocorre, os criminosos cibernéticos vendem os dados, em vez de publicar gratuitamente da internet.

Ataques cibernéticos também ocorrem em provedores de serviços em nuvem pública conforme noticiado pela empresa DYN (2016), que fornece serviços de DNS (Domain Name System), a empresa sofreu uma série de ataques de negação de serviços conhecido como ataque DDoS. DYN (2016) relata que esses ataques provocaram instabilidade nos serviços da empresa, afetando sites como Twitter,

Spotify, SoundCloud e entre outros sites, impactou especialmente o acesso aos servidores localizados nos Estados Unidos.

Tendo em vista o cenário proposto, com os ataques virtuais a nuvens públicas e vazamento de dados pessoais, este projeto visa demonstrar a ferramenta OpenStack de gestão de uma infraestrutura como serviço, como uma solução viável de nuvem privada em relação aos modelos apresentados. A implementação do projeto se dará em ambiente virtualizado, consistindo na orquestração de uma nuvem privada OpenStack.

1.2 Delimitações do Trabalho

Pesquisa e analise das características da implementação de uma infraestrutura como serviço em nuvem privada, pontuando suas funcionalidades, segurança e objetivos.

1.3 Objetivos

O propósito deste projeto será o de estudar a implementação de uma ferramenta de software livre, possibilitará um conhecimento para a orquestração de uma nuvem privada. Os objetivos dividem-se em: geral e específicos.

1.3.1 Objetivo Geral

Implementar a ferramenta de código aberto OpenStack para gerenciamento de infraestrutura como serviço em nuvem privada, visando a segurança e disponibilidade dos dados.

1.3.2 Objetivos Específicos

Os objetivos específicos para este projeto são:

a) Estudar, identificar e analisar as características sobre nuvem privada e infraestrutura como serviço;

b) Implementar, executar e testar a ferramenta para administrar uma infraestrutura como serviço em nuvem privada;

c) Demonstrar os resultados obtidos com a implementação da ferramenta.

1.4 Justificativa

Empresas como o GOOGLE (2018) e a AMAZON (2018) oferecem infraestrutura como serviço em nuvem pública, sua infraestrutura e aplicações são compartilhadas entre seus clientes em todo o mundo. De acordo com o PENSO (2017) com a utilização destes serviços que estão fora das dependências da empresa e muitos clientes interagindo na mesma plataforma, não se tem total confiança no quesito segurança de seus dados e controle de tal.

Conforme a TIINSIDE (2017) que relatou uma pesquisa da Intel Security, 53% das empresas mantêm algum dado sensível na nuvem pública, mas apenas 38% confiam plenamente neste serviço, e apenas 4% escolhem como o único modelo para armazenar seus dados.

A segurança dos dados de uma empresa, armazenados em nuvem pública geram muitas discussões, pois de acordo com a SUPORTI (2017) alguns provedores de nuvem pública declaram em seus contratos de serviços, que a segurança ofertada a empresas é a mesma para usuários domésticos.

Justifica-se esse projeto como uma solução de implementação de uma ferramenta de software livre para orquestração de nuvens privadas em ambientes educacional, pessoal e empresarial, utilizando uma ferramenta para maximizar custos e potencializar o uso da virtualização. A implementação visa à administração e gerenciamento de servidor com máquina virtual em uma infraestrutura como serviço, assim disponibilizando escalabilidade, elasticidade, disponibilidade, processamento em tempo real e segurança de uma nuvem privada.

2 REVISÃO BIBLIOGRÁFICA

O aumento do volume e tráfego de informações juntamente com surgimento de novos tipos de dados de acordo com PEREIRA (2013), provocou a necessidade de desenvolvimento de novas tecnologias com maior capacidade de processamento e armazenamento, menor custo e menores dimensões físicas.

TAURION (2009) descreve que o termo Cloud Computing ou também conhecido como Computação em Nuvem surgiu em 2006 em uma palestra de Erick Schmidt, do Google, sobre como sua empresa gerenciava seus equipamentos de data centers. Hoje, Computação em Nuvem, se apresenta como o cerne de um movimento de profundas transformações do mundo da tecnologia conforme TAURION (2009).

2.1 COMPUTAÇÃO EM NUVEM

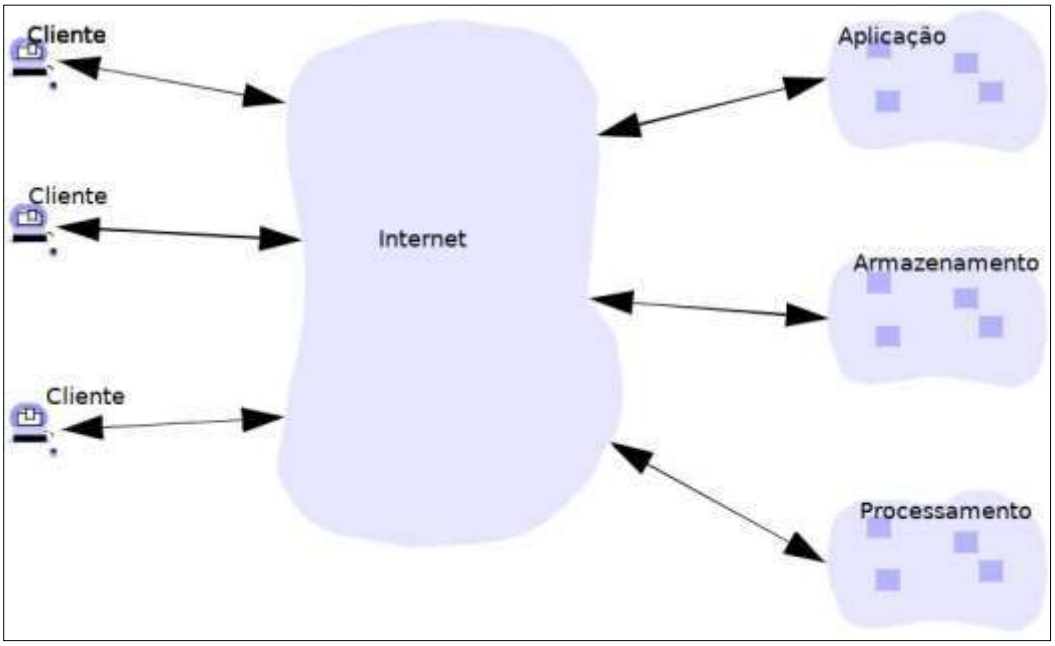
HASHEM (2014) descreve que a computação em nuvem é uma das mudanças mais significativas em TIC (Tecnologia da Informação e Comunicação) e serviços modernos para aplicativos corporativos, se tornou uma arquitetura poderosa para executar computação complexa e em larga escala. As vantagens da computação em nuvem segundo HASHEM (2014) incluem recursos virtualizados, processamento paralelo, segurança e integração de serviços de dados com armazenamento de dados escalável. HASHEM (2014) complementa que a computação em nuvem pode não apenas minimizar o custo e a restrição para automação e informatização por parte de indivíduos e empresas, mas também pode reduzir o custo de manutenção da infraestrutura, o gerenciamento eficiente e o acesso ao usuário.

VERDERAMI (2013) descreve que a computação em nuvem possibilita transformar os sistemas computacionais físicos em uma rede virtual, através de um equipamento computacional com acesso a internet (notebooks, smartphones, computadores de mesa e tablets), possa acessar as aplicações e obter informações e dados de forma muito rápida.

O termo nuvem ou “cloud” de acordo com SOTO (2011) é uma metáfora em relação á forma como a internet é usualmente mostrada nos diagramas de rede, como uma nuvem, como pode ser visto na Figura 1 onde o cliente tem acesso via

internet aos serviços disponibilizados na nuvem como aplicação, armazenamento e processamento.

Figura 1. Diagrama de rede (nuvem).



Fonte: Autor, adaptado COULOURIS (2005).

Uma nuvem é definida conforme COULOURIS (2005) como um conjunto de serviços de aplicativos, armazenamento e computação baseados na internet. O termo nuvem também promove de acordo com COULOURIS (2005) uma visão de tudo como um serviço, a partir de infraestrutura física ou virtual através de software, muitas vezes pago em uma base por uso, em vez de comprado reduzindo os requisitos nos dispositivos dos usuários, permitindo um ambiente de trabalho muito simples ou dispositivos portáteis para acessar uma ampla gama de recursos e serviços.

MELL (2011) define a computação em nuvem como um modelo que possibilita acessar recursos computacionais, conveniente e sob demanda a uma rede compartilhada. Ainda conforme MELL (2011) é um conjunto de recursos de computação configuráveis, conforme esta demonstrada na Figura 2, com servidores, banco de dados, tablets, desktops, notebooks e celulares, estão todos conectados aos serviços disponibilizados pela nuvem computacional, de maneira prática e sob

demanda, com esforço mínimo de gerenciamento ou interação com o provedor de serviços.

Figura 2. Visão geral de uma nuvem computacional.

Fonte: LOPES (2016).

2.2 CONTEXTO HISTÓRICO

Como foi mencionado por MELL (2011) que a computação em nuvem é um modelo de serviços via web, é importante compreender a evolução dos sistemas de computação até o surgimento da Computação em Nuvem.

A primeira evidência de uma arquitetura computacional conforme SOTO (2011) foi apresentada por Charles Babbage em 1856 D.C com o nome de Motor Analítico que representa ainda hoje, a forma mais básica de uma arquitetura de sistemas instalados em computadores modernos.

SOTO (2011) relata que a evolução da computação se divide em quatro gerações:

1. Primeira geração: A IBM financiou a criação do Mark I no ano de 1943, um computador programável eletromecânico. Neste mesmo ano na Inglaterra foi desenvolvido o primeiro dispositivo de computação programável, eletrônico e digital com o nome de Colossos. Com o funcionamento por válvulas e armazenamento de

dados através de cartões perfurados onde os tamanhos destes computadores ocupavam galpões;

2. Segunda geração: A invenção dos transistores marcou o surgimento desta geração. A arquitetura destes computadores continha custo elevado e necessitavam de muito espaço disponível, por isso sua utilização se limitava a universidades e órgãos governamentais;

3. Terceira geração: Esta geração foi marcada pelo surgimento dos circuitos integrados, eram chamadas de minicomputadores que tinha valor e tamanho mais baixos que a geração anterior, assim possibilitando sua utilização por empresas privadas;

4. Quarta geração: Na década de 1970 surgiu o microprocessador e com esta tecnologia foi produzido o popularmente conhecido PC (Personal Computer) que oferecia aos seus usuários maior capacidade de processamento, armazenamento de dados e tamanhos bastante reduzidos.

Fatores como a popularização da internet na década de 1990, o crescimento de produção juntamente com a comercialização de microcomputadores conforme SOTO (2011) foi determinante para que á medida que a conectividade deste equipamento se elevasse á uma plataforma de serviços inteiramente via web surgisse.

Ao final da década de 1990 segundo SOTO (2011) surgiu a Computação em Nuvem e subsequentemente com grande quantidade de serviços web, onde se tinha a necessidade de acessar através de dispositivos com recursos físicos limitados, se teve um avanço na tecnologia de virtualização de hardware, como uma forma de evolução para combater este consumo de serviços web.

2.3 MODELOS DE COMPUTAÇÃO EM NUVEM

Segundo MACHADO (2011) existem dois tipos principais de nuvem, nuvem pública e nuvem privada. A nuvem pública é aquela em que o consumidor dos serviços em nuvem e o provedor desses serviços existem em empresas separadas. A propriedade dos ativos utilizados para prestar serviços em nuvem permanece com o provedor. Por sua vez, nuvem privada é aquela em que o consumidor dos serviços em nuvem e o provedor desses serviços se encontram na mesma empresa. A

propriedade dos ativos em nuvem reside dentro de uma mesma empresa, que presta e consome serviços concomitantemente. Em algumas literaturas, são mencionados outros dois tipos de nuvens: nuvem híbrida e nuvem comunitária.

CSA (2009) e VERAS (2013) demonstram e exemplificam que há quatro modelos de implantação para serviços em nuvem, com variações derivadas e que abordam requisitos específicos:

1. Nuvem Pública: A infraestrutura da nuvem é disponibilizada ao público em geral ou um grande grupo da indústria e é de propriedade de uma organização que vende serviços em nuvem. Disponibilizada publicamente através do modelo pague-por-uso. São oferecidas por organizações públicas ou por grandes grupos industriais que possuem vasta capacidade de processamento e armazenamento.

2. Nuvem Privada: A infraestrutura da nuvem é operada exclusivamente para uma única organização e podem ser administrados pela organização ou por terceiros, e podem existir em instalações ou fora das instalações de uma empresa. Compreende uma infraestrutura de Cloud Computing operada e quase sempre gerenciada pela organização cliente. Os serviços são oferecidos para serem utilizados pela própria organização, não estando publicamente disponíveis para uso geral. Nuvem privada é definida por privacidade, não propriedade, localização ou responsabilidade de gestão.

3. Nuvem Comunitária: A infraestrutura de nuvem é compartilhada por várias organizações e apoia uma comunidade específica que compartilha preocupações (por exemplo, missão, segurança requisitos, políticas ou considerações de conformidade). Pode ser gerenciado por organizações ou terceiros, podem existir em instalações ou fora de instalações das empresas.

4. Nuvem híbrida: A infraestrutura de nuvem é uma composição de duas ou mais nuvens (privada, pública ou comunitária) que permanecem entidades únicas, mas são unidas por padrões ou tecnologia proprietária que permite a portabilidade de dados e aplicações. A nuvem híbrida impõe uma coordenação adicional a ser realizada para uso das nuvens privadas e públicas com impactos de governança.

A computação em nuvem segundo MELL (2011) é composta de cinco características essenciais para entender o funcionamento de uma nuvem:

1. Autoatendimento sobre demanda: O usuário pode provisionar capacidade de computação, tempo de servidor e armazenamento sem exigir interação com o prestador de serviços da nuvem.

2. Amplo acesso á rede: Acesso a diferentes dispositivos (smartphones, tablets, notebooks, etc.) e também diferentes sistemas operacionais. Garantir que todos os seus serviços podem ser acessados de uma maneira padronizada em diferentes dispositivos.

3. Pool (compartilhamento) de recursos: Os recursos devem atender dinamicamente e conforme a demanda dos usuários, mas não disponibilizando o controle e nem a localização exata desses recursos fornecidos. Mas pode ser capaz de especificar a localização em um nível abstração (por exemplo, país, estado ou tipo de datacenter).

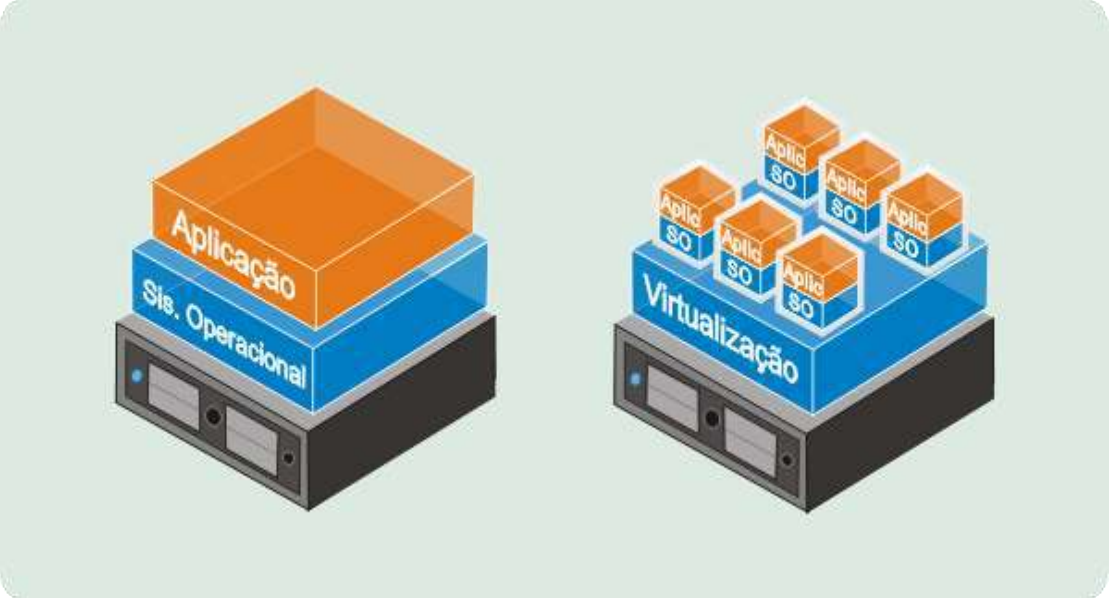
4. Elasticidade rápida: Os recursos devem ser sem limites e podem ser modificados a qualquer hora, promovendo a necessidade de aumentar ou diminuir os recursos de maneira rápida.

5. Medição de serviço: Os sistemas em nuvem controlam e aperfeiçoam automaticamente o uso de recursos, aproveitando uma capacidade de medição em algum nível de abstração apropriado ao tipo de serviço (por exemplo, armazenamento, processamento, largura de banda).

2.4 VIRTUALIZAÇÃO

VELTE (2012) relata que tem uma tendência crescente no mundo de TI que é a virtualização de servidores, onde o software pode ser instalado permitindo que vários servidores virtuais sejam usados, como exemplo, meia dúzia de servidores virtuais funcionando em um único servidor físico. Segue exemplo na Figura 3 conforme Silveira (2014) de uma arquitetura tradicional de servidor, onde se contém o equipamento de hardware (servidor físico) com um sistema operacional e apenas uma aplicação instalada, em comparação com a virtualização de servidores, onde se contém equipamento de hardware (servidor físico) com a ferramenta de virtualização que disponibiliza sistemas operacionais com aplicações instaladas.

Figura 3. Arquitetura Tradicional x Virtualização.



Fonte: SILVEIRA (2014).

MEDEIROS (2015) relata que as máquinas virtuais que conhecemos atualmente tornam possível a utilização de Computação em Nuvem, anteriormente cada servidor era visto como apenas uma única máquina, com o avanço da tecnologia de virtualização é possível armazenar diversos outros servidores virtuais, que por sua vez podem ser usados para diversas finalidades diferentes.

Conforme CHEE (2013) a virtualização se resume á questão dos recursos que estão sendo usada em vários servidores disponíveis e onde a organização precisa maximizar o retorno do investimento e o desempenho. CHEE (2013) relata que muitas empresas de porte médio contêm máquinas físicas no servidor tipicamente dedicado e rodando apenas uma única tarefa, passa a maior parte do tempo ocioso, rodando entre 10% a 20% da utilização da CPU, ficando na ociosidade enquanto espera que ocorra a entrada de dados ou um processamento externo. CHEE (2013) complementa que todos aqueles ciclos de CPU desperdiçados são provavelmente uma extravagância em empresas modernas, e os altos custos associados aos servidores pode ser um desperdício nas economias da empresa, além disso, o custo de energia necessária para alimentar e resfriar muitos servidores separados tem se tornado significativo. CHEE (2013) finaliza que a virtualização pode proporcionar economia de custos em todas as frentes através de upgrades para servidores mais eficientes, (alguns novos servidores têm fontes de alimentação com 95% de eficiência, comparados com unidades de 80% de eficiência

a algumas gerações anteriores), e em muitos casos é possível combinar facilmente os vários tipos de servidores em uma única máquina virtualizada.

A virtualização pode ser conceituada de duas formas conforme VERAS (2010), onde ocorre o particionamento de um servidor físico em vários servidores lógicos, também pode ser definida como uma camada de abstração entre o software e o hardware, com proteção direta do software aos recursos físicos do hardware.

O software que implementa a virtualização de acordo com VERAS (2013) normalmente é do tipo hypervisor, que é conhecido como monitor de máquina virtual (VM), onde representa a camada de abstração, que entrega para o sistema operacional convidado um conjunto de instruções de máquinas equivalente ao processador físico, o servidor físico virtualizado pode então rodar várias instâncias virtuais. CHEE (2013) denomina a instância como uma cópia individual de um determinado sistema operacional rodando em um ambiente virtual.

VERAS (2013) relata que a virtualização simplifica o gerenciamento, permite flexibilizar e ampliar o poder de processamento, com funcionalidades contidas nos softwares de virtualização, também permitem melhorar a disponibilidade e a recuperação de desastres de ambientes de TI de uma maneira mais simples e com menor custo quando comparado a formas tradicionais.

Segundo VERAS (2013) com a virtualização, cada VM utiliza um sistema operacional e suas respectivas aplicações, diversas VM’s podem coexistir no mesmo servidor físico.

VERAS (2010) define a virtualização em três níveis:

1. Nível de hardware: A camada de virtualização é colocada sobre a máquina física e a apresenta ás camadas superiores como um hardware abstrato similar ao original;

2. Nível de Sistema Operacional: A camada de virtualização é um mecanismo que permite a criação de partições lógicas, onde cada partição é vista como uma máquina isolada, mas compartilham o mesmo sistema operacional;

3. Nível de Linguagem de Programação: Nesse nível o objetivo é a definição de uma máquina abstrata sobre a qual executa uma aplicação desenvolvida em uma linguagem de alto nível.

Anteriormente a existência destas técnicas de virtualização conforme LOPES (2012), uma atualização simples de memória em um servidor, acarretaria numa tarefa com nível elevado de trabalho e impossibilitando acesso a diversos serviços,

sem acesso por tempo indeterminado. A escalabilidade vem de encontro à virtualização conforme LOPES (2012) permite a realocação dos recursos de forma dinâmica, permitindo dimensionar uma máquina virtual de acordo com a demanda dos processos existentes, com intuito de aumentar a quantidade de memória, armazenamento e quantidade de núcleos de processamento sem a necessidade de desligamento do equipamento.

LOPES (2012) relata a técnica Live Migration, consiste em migrar máquinas virtuais de um servidor físico para outro, de forma que os processos interligados ao mesmo, não sofram nenhum tipo de impacto durante o procedimento de migração.

De acordo com ALKMIM (2009) as técnicas de migração são constantemente utilizadas para o gerenciamento de balanceamento de carga entre duas ou mais máquinas, além de garantir o funcionamento de servidores de alta disponibilidade, possibilitando transferir uma máquina virtual para outro servidor físico em caso de alguma falha de hardware. Para que o processo de migração seja possível de acordo com ALKMIM (2009), o encapsulamento de todo o estado de software e hardware da máquina virtual seja efetuado, em alguns arquivos cria-se um ambiente propício para a migração, que desta forma, é possível migrar uma máquina virtual para uma máquina física diferente, mesmo que essa possua um diferente hardware.

Conforme LOPES (2012) a flexibilidade é um dos requisitos básicos para um ambiente virtualizado, a utilização de um sistema de armazenamento centralizado se torna indispensável para tornar essa característica possível. LOPES (2009) relata que manter os discos rígidos de uma máquina virtual em uma unidade de armazenamento centralizado, permite facilmente a transportação para outra máquina em caso de falha e ausência de recursos computacionais, que contenham nos hardwares físicos do ambiente.

LOPES (2012) exemplifica três soluções de ferramentas para virtualização:

1. VMware: Ferramenta de virtualização mais popular, definida como infraestrutura de virtualização completa, possui plataforma para versões de desktops a data centers, gestão de automatização, infraestrutura virtual e virtualização de plataformas;

2. XEN: Ferramenta de código aberto que permite a utilização do modelo de Paravirtualização. Seu funcionamento é baseado nos conceitos de domínios e hypervisor. As diversas máquinas virtuais são denominadas de domínios. O

hypervisor é responsável pela comunicação dentre os domínios e o hardware físico, gerenciando as requisições de acordo com o tipo de domínio;

3. VirtualBox: Software de virtualização com parte de seu código fonte aberto. Disponibilizado para uso profissional e doméstico, suporta múltiplas plataformas de sistemas operacionais, possuí interfaces de programação interna bem definida em desenho de cliente/servidor, tornando fácil o controle de várias interfaces em apenas um gerenciamento.

2.4.1 Tipos de Virtualização

Conforme LAUREANO (2006) a virtualização possibilita a utilização de máquinas virtuais e emuladores, que em suas características estão elencadas a execução de um sistema operacional juntamente com suas aplicações sobre outro, utilização de uma aplicação de outra plataforma operacional, execução de múltiplos sistemas operacionais e flexibilização de uma plataforma complexa de trabalho.

LAUREANO (2006) relata a ferramenta que é denominada de emulador, integra uma virtualização, o emulador implementa todas as instruções realizadas pela máquina real em um ambiente abstrato de software, possibilita executar um aplicativo de uma plataforma em outra, é um software criado essencialmente para transcrever instruções de um processador, que seja o alvo para o processador no qual o mesmo esta em funcionamento.

2.4.2 Emuladores

Conforme SANTOS (2013) na emulação de hardware, contém uma arquitetura diferente da utilizada no hardware hospedeiro, onde pode se utilizar uma plataforma de maior desempenho para emular uma arquitetura de hardware antiga que não seja mais comercializada, sem a necessidade de migrar os sistemas legados.

LAUREANO (2006) define que os emuladores podem ser divididos e classificados em quatro tipos:

1. Emulação totalmente baseada em hardware: A solução independe de software para ser utilizada, onde um exemplo de aplicação serie de um processador

emulando uma arquitetura com versão mais antiga para garantir a execução de softwares legados;

2. Emulação parcialmente baseada em hardware: O hardware é projetado para suportar a carga de emulação, mas necessita de uma aplicação em software para que seus recursos sejam utilizados;

3. Emulação baseada em software: A emulação é obtida pelo software que utiliza alguns dos recursos de hardware para disponibilizar a emulação;

4. Emulação totalmente baseada em software: O emulador não necessita de nenhum hardware para executar a emulação, neste caso o software provê todos os recursos necessários.

2.4.3 Virtualização Total

Segundo SILVA (2007) a Virtualização Total é uma técnica que provê uma completa simulação da subcamada de hardware para os sistemas convidados, onde o resultado é um ambiente que todos os sistemas operacionais que são capazes de executar diretamente em um hardware também podem executar em uma máquina virtual.

Na Virtualização Total conforme SANTOS (2013), uma camada de software é utilizada para abstrair os recursos físicos, porém diferentemente desta apenas o hardware do sistema hospedeiro é emulado, de forma que os sistemas operacionais instalados sobre a camada de controle devem ser compatíveis com esta arquitetura.

SILVA (2007) relata que a principal vantagem da Virtualização Total é que não contém a necessidade de modificações nos sistemas operacionais convidados, para que suportem a virtualização, dado que uma completa estrutura de hardware é virtualizada, o que faz que o sistema convidado entenda que esta executando diretamente no hardware.

2.4.4 Paravirtualização

A técnica de Paravirtualização de acordo com SILVA (2007) é a apresentação de uma interface de software para máquinas virtuais que é similar á subcamada de hardware. SILVA (2007) complementa que a técnica permite que o sistema

convidado tenha acesso direto aos recursos do hardware, mas com restrições, que são administradas pelo monitor de máquinas de virtuais.

Na Paravirtualização conforme LAUREANO (2006), o sistema convidado a ser virtualizado, sofre modificações para que a interação com o monitor de máquinas virtuais seja mais eficiente. LAUREANO (2006) relata que embora que a Paravirtualização exija que sistema a ser virtualizado precise ser modificado, permita que o sistema convidado tenha acesso aos recursos do hardware diretamente, onde o acesso é monitorado pelo monitor de máquinas virtuais, fornecendo ao sistema convidado todos os recursos do sistema.

SANTOS (2013) relata que a Paravirtualização também utiliza uma camada de software de controle, como a Virtualização Total, porém neste método, as alterações são inseridas nos sistemas operacionais convidados para que estes cooperem com o processo de virtualização.

2.4.5 Computação Verde

SOTO (2011) relata que algumas tecnologias como as Virtualização e Computação em Nuvem são classificadas como tecnologias Verdes (TI Verde), que contribuem para redução do consumo de energia e também emissões de dióxido de carbono.

A TI Verde segundo LIMA (2013) é considerada uma tendência mundial visando o impacto que os recursos de tecnologia têm no meio ambiente, como um dos princípios de redução no consumo de energia. Outros focos da TI Verde de acordo com LIMA (2013) são a diminuição da utilização de matéria prima, substâncias com níveis tóxicos baixo na fabricação e que isso tenha impacto em seu descarte, com intuito de reciclagem e reutilização.

A Computação em Nuvem conforme LIMA (2013) contém um crescimento constante contendo novas aplicações e serviços, com este aumento de procura á empresas deste ramo, novos datacenters estão sendo construídos e ampliados. LIMA (2013) relata que o consumo de energia com este crescimento tornou-se significativo além do que todo um ambiente de TI é um grande emissor de carbono (CO2), um gás que ocasiona efeito estufa, principal gás relacionado ao aquecimento global.

A Nuvem Verde (do inglês Green Cloud) segundo WERNER (2011) adiciona uma preocupação elevada sobre a infraestrutura de TI, oferecendo o serviço processamento de máquinas virtuais em poucos servidores físicos, assim permitindo que equipamentos ociosos sejam desligados quando estão com período de baixa demanda de trafego de dados. WERNER (2011) relata que consumir energia em baixo uso não interfere no desempenho dos equipamentos, assim alcançando o principio da sustentabilidade, que é o foco da TI Verde.

2.5 SERVIÇOS DE COMPUTAÇÃO EM NUVEM

Segundo WESTPHALL (2013) os serviços que são oferecidos pela computação em nuvem são divididos em três classes ou também conhecido como camadas. WESTPHALL (2013) relata que uma das classes mais simples, como primeiro nível oferecido, é a infraestrutura como serviço, em outra classe de segundo nível, é oferecido à plataforma de desenvolvimento como serviço, em um terceiro nível, os aplicativos são oferecidos como serviço.

VERAS (2013) relata os três principais modelos de serviços de Computação em Nuvem:

Infraestrutura como Serviço (IaaS – Infraestructure as a Service): O provedor tem a capacidade de oferecer uma infraestrutura de processamento, armazenamento de forma transparente e representa uma abstração da infraestrutura propriamente dita. Neste cenário, o usuário não tem o controle da infraestrutura física, mas através de mecanismos de virtualização possui controle sobre as máquinas virtuais. Uma infraestrutura de computação instantânea, provisionada e gerenciada pela internet. Tem como atribuição pagar por somente o que for utilizado. Os recursos são oferecidos separadamente e tem a possibilidade de ser alugado pelo tempo que se necessita. Alguns dos cenários que empresas adotam neste serviço são teste e desenvolvimento, hospedagem de sites, armazenamento com backup e recuperação. O Windows Azure que pertence a Microsoft é um exemplo do serviço citado. MICROSOFT (2018)

Plataforma como Serviço (PaaS – Plataform as a Service): É oferecida pelo provedor para o desenvolvedor de aplicativos. Aplicativos estes que serão executados e disponibilizados na nuvem. A plataforma na nuvem oferece um modelo de computação, armazenamento e comunicação para os aplicativos. Neste nível é

oferecido um ambiente de desenvolvimento aonde o usuário tenha a possibilidade de criar e hospedar suas aplicações e distribuir elas, sem a preocupação se a infraestrutura comporta a aplicação. Este serviço permite que a empresa se foque na concentração na implantação e no gerenciamento das suas aplicações. AMAZON (2018) oferece este serviço.

Software como Serviço (SaaS – Software as a Service): Aplicativos de interesse para uma grande quantidade de usuários, que passam a ser hospedados na nuvem com uma alternativa ao processamento local. Os aplicativos são oferecidos como serviços por provedores e acessados pelos clientes através de aplicações como um navegador de internet (browser). Todo o controle e gerenciamento da rede, sistemas operacionais, servidores e armazenamento é realizado pelo provedor de serviço. Neste modelo não necessitam ter uma própria infraestrutura e nem software instalado, o usuário acessa e-mail, calendário, armazenamento em nuvem, edição de arquivos via texto, planilhas e apresentações, tudo por meio da internet. A empresa Google oferece esse serviço. GOOGLE (2017).

Na Figura 4 é demonstrado conforme MICROSOFT (2018), o quadro de responsabilidades do cliente para os tipos de serviços disponibilizados pela computação em nuvem, onde no primeiro quadro (On-Premises) demonstra que os serviços de aplicações, dados, tempo de execução, middleware, sistema operativo, virtualização, servidores, armazenamento e rede são de responsabilidade do cliente. No segundo quadro nomeado de infraestrutura como serviço (Infrastructure as a servisse), as aplicações, dados, tempo de execução, middleware, sistema operativo são de cargo do cliente, os outros serviços são disponibilizados pela computação em nuvem contratada. No terceiro quadro nomeado de plataforma como serviço (Plataform as a service), as aplicações e dados são a cargo do cliente apenas, os outros serviços tem responsabilidade e disponibilização pela contratante. No quarto quadro nomeado de software como serviço (Software as a service), todos os serviços são a cargo da contratante fornecedora de computação em nuvem.

Figura 4. Quadro de responsabilidades entre cliente e fornecedor.

Fonte: Autor, adaptado MICROSOFT (2018).

SANTOS (2010) exemplifica alguns dos serviços disponibilizados pelo IaaS, Paas e SaaS:

1. Banco de Dados como Serviço: O banco de dados como Serviço tem a capacidade de oferecer serviços de um banco de dados hospedado remotamente. Funcionaria como se fosse um banco de dados local, porém estará hospedado em um datacenter. Sua principal vantagem está na economia com licenças de software e aquisição de hardwares;

2. Governança como Serviço: A governança como Serviço auxilia no gerenciamento de topologias, monitoramento de recursos e virtualização conectada á internet, com base em políticas definidas para dados e serviços;

3. Informação como Serviço: A informação como Serviço tem como conceito e como objetivo de consumir informações hospedadas remotamente, assim como uma integração de softwares, utilizando, por exemplo, APIs (Integração como Serviço) a integração como serviço tende a oferecer as funções e os recursos de um EAI (Enterprise Application Integration), porém, operando em nuvem;

4. Processo como Serviço: Processo como Serviço oferece um recurso remoto que pode reunir muitos outros, criando assim processos de negócio. O aplicativo pode interagir tais como serviços e dados, que combinados, geram uma sequência de processos empresariais;

5. Segurança como Serviço: Segurança como serviço tem a capacidade de oferecer serviços de segurança aplicados a e-mail, navegação entre outros, acoplando uma interface de monitoramento via internet;

6. Armazenamento como Serviço - Armazenamento como Serviço seria o componente mais primitivo da computação em nuvem, explorado pela maioria das outras modalidades. Esta modalidade oferece o armazenamento como serviço dentro de um datacenter, podendo ser acessado por aplicações externas;

7. Testes como Serviço: Testes como Serviço é a capacidade de testar sistemas locais ou mesmo sistemas em nuvem em um ambiente para testes de aplicações nas nuvens também.

TERUEL (2014) relata outros modelos de serviço em Computação em Nuvem:

1. Banco de Dados como Serviço (DAAS – Data as a Service): Esta modalidade é direcionada ao fornecimento de serviços para armazenamento e acesso de volumes de dados. Vantagem de que o detentor da aplicação conta com maior flexibilidade para expandir o banco de dados, compartilhar as informações com outros sistemas, facilitar o acesso remoto por usuários autorizados;

2. Desenvolvimento como Serviço (DEVAAS – Development as a Service): Modelo que se apoia no compartilhamento de ferramentas de desenvolvimento e serviços. Extremamente flexível, permite a mescla de conteúdos de diversas fontes para criar um novo serviço;

3. Comunicação como Serviço (CAAS – Comunication as a Service): Modelo que estabelece uma comunicação unificada por meio de um datacenter, garantindo que a comunicação da empresa fique alocada em um sistema central;

4. Tudo como Serviço (EAAS – Everything as a Service): Unificam a infraestrutura, plataforma, software e suporte em um único serviço;

5. Ensaio como Serviço (TAAS – Testing as a Service): Oferece um ambiente onde os usuários possam simular o comportamento de execução de aplicações e sistemas, sendo comumente usado para testes de segurança anteriormente dos serviços implementados.

2.6 SEGURANÇA NA COMPUTAÇÃO EM NUVEM

A segurança na Computação em Nuvem está diretamente ligada à localização dos dados, conforme COGO (2013) cita que existem três tipos, interna, remota ou distribuída, e que cada um é associado a um modelo de computação em nuvem.

LEIMESTER (2010) relata que a nuvem interna está associada ás nuvens privadas, que se concentra numa rede própria da empresa, garantindo maior segurança para os dados. Conforme LEIMESTER (2010) o usuário tem o conhecimento da localização de seus dados e gestão sobre os mesmos.

A localização remota dos dados conforme SRINIVASAMURTHY (2010) é a forma mais utilizada em aplicações da Computação em Nuvem, onde se tem um debate em crescente escala sobre a segurança dos dados remotos, questões legais de localização dos dados, pois os usuários não contem a domínio perfeito da localização de seus dados.

A localização distribuída dos dados conforme LEIMESTER (2010) é caracterizada pela combinação da localização interna e remota. LEIMESTERT (2010) relata que neste formato, parte dos dados fica localizada remotamente, enquanto outra parte, de maior importância para a empresa, fica localizada internamente.

CASTRO (2010) relata que no cenário corporativo é comum observar que as questões de segurança da informação não são tratadas em um nível de gestão da organização, tendo como consequência a falta de recursos para minimizar os riscos existentes.

Conforme MENEGATT (2012) na computação tradicional os usuários têm total controle sobre seus dados, processos e seu computador, ao migrar para a Computação em Nuvem, todos os serviços e manutenção dos dados são fornecidos por um provedor de nuvem, onde o cliente desconhece quais processos estão em execução ou em qual localização seus dados estão armazenados.

Uma pesquisa realizada pela SYMANTEC (2011) avaliou a situação de Computação em Nuvem na América Latina, esta pesquisa mostrou o quesito segurança como principal preocupação e objetivo das corporações entrevistadas na migração para a nuvem. A pesquisa da SYMANTEC (2011) complementa que 86% dos entrevistados acreditam que a nuvem não causará impacto ou melhorar a postura de segurança, mas de contrapartida eles classificam a segurança como a principal preocupação com o surto de ataques de malware, roubo de dados por ladrão cibernético, compartilhamento inseguro de dados confidenciais via nuvem, uso irregular da nuvem e vazamento de informação.

CSA (2009) divulgou uma lista elaborada juntamente com 29 consultorias e fornecedores de serviço, com sete itens considerados pecados mortais que compõem a segurança na Computação em Nuvem:

1. Perda de dados (vazamento): Com um mau controle de API's, geração de chaves, armazenamento, gestão fraca, pode ocorrer de os aplicativos vazarem dados;
2. Vulnerabilidade de tecnologias compartilhadas: Acordos de nível de serviço (SLA's) garantem o gerenciamento de atualizações, melhores práticas para configuração de servidores e manutenção de rede, evitando configuração duplicada equivocada em múltiplos servidores e máquinas virtuais compartilhem essa informação;
3. Internos maliciosos: Serviços oferecidos pela Computação em Nuvem como disponibilidade e flexibilidade precisam ter a confiança das empresas contratantes. Cada empresa provedora de serviços contém seus próprios níveis de segurança ao acesso aos datacenters;
4. Desvio de tráfego (contas e serviços): Autenticação efetuada de forma insegura pode acarretar em acesso a dados, aplicativos e recursos que se encontram em máquina virtual em nuvem;
5. Interfaces de programação de aplicativos (API's) inseguras: API's com interfaces inseguras permitem que usuários mal intencionados, tenham acesso a seus serviços para invasão de contas;
6. Abuso e uso nefasto da Computação em Nuvem: Acesso de pessoas sem autorização com fins mal intencionados aos serviços hospedados em nuvem;
7. Perfil de risco desconhecido: A transparência vem de encontro aos serviços contratados da nuvem, onde o contratante tem apenas a visão da interface contratada, mas desconhece as plataformas e os níveis de segurança sobre os quais seu serviço contratado está hospedado.

MELO (2018) relata que a segurança da informação possui princípios que devem ser observados, a disponibilidade, integridade, confidencialidade, autenticidade e não-repúdio como é demonstrado na Tabela 1.

Tabela 1. Princípio da segurança em computação em nuvem.

PRINCÍPIOS DE SEGURANÇA	**CENÁRIOS DE RISCO**
Integridade	Manter os dados sem alteração, ou seja, o dado enviado da origem deve ser o mesmo dado recebido pelo destino. Violação das leis de proteção de dados.
Disponibilidade	Manter o caminho entre a origem e o destino sempre ativo.
Confidencialidade	Os dados transmitidos entre as aplicações de vários usuários utilizando o mesmo sistema de armazenamento.
Autenticação	Verificação do sistema das chaves autenticadoras (privadas e públicas) entre entidades que se comunicam.
Não-repúdio	Garantir que a pessoa não negue que tinha assinado a transmissão da mensagem ou arquivo.

Fonte: Autor, adaptado MELO (2018).

Os princípios de confidencialidade e integridade conforme MELO (2018) depende basicamente de medidas de segurança, como exemplo, ter um firewall ativo com rotas e bloqueios definidos contra vírus, worms, spyware e entre outros tipos de dano ao sistema.

A CSA (2017) que é uma organização mundial dedicada a definir e conscientizar sobre as melhores práticas para ajudar a garantir um ambiente seguro de nuvem, publicou um relatório com atualização das doze principais ameaças a segurança em Computação em Nuvem no ano de 2016:

1. Violação de dados;
2. Gerenciamento insuficiente de identidade, credencial e acesso;
3. Interfaces e API’s inseguras;
4. Vulnerabilidades do sistema;
5. Invasão de conta;
6. Invasores maliciosos;

7. Ameaças avançadas persistentes;
8. Perda de dados;
9. Diligência devida insuficiente;
10. Abuso e uso indevido dos serviços de nuvem;
11. Negação de serviço;
12. Compartilhamento de vulnerabilidades de tecnologia.

MENEGATT (2012) relata os princípios de segurança da informação em modelo de Nuvem Pública, envolve a integridade, confidencialidade, autenticidade e não-repúdio. Diferente da Nuvem Pública MENEGATT (2012) afirma que a segurança em Nuvem Privada é muito maior, pois a mesma se encontra dentro de um firewall, que existe um maior controle e estrita aderência ás restrições regulatórias.

CORRÊA (2013) aponta como ponto importante na segurança da informação a disponibilidade dos serviços em uma rede de computadores, onde o responsável pela rede deve se preocupar com as questões de segurança, pra que esteja sempre disponível a rede de computadores para seus usuários finais. A disponibilidade da rede conforme GOMES (2015) pode ser comprometida com um ataque de negação de serviço (DoS - Denial of Service), onde o objetivo é interromper ou prejudicar o fornecimento de um serviço. DEFENSORWEB (2019) relata outro tipo de ataque que prejudica a disponibilidade de serviço, o ataque força bruta tipo dicionário.

2.6.1 Ataque de Negação de Serviço

GOMES (2015) relata que um ataque de negação de serviço (DoS) tem como objetivos de sobrecarregar uma rede de computadores, de modo de tornar o acesso lento ou inoperante, afetar a estabilidade de serviços (e-mail, página web, etc.), explorar vulnerabilidades de sistema operacionais, protocolos de rede, deste modo usuários legítimos ficam impedidos de ter acesso á esses serviços, o servidor fica impossibilitado de processar todas as requisições.

Conforme GOMES (2015) os ataques DoS de negação de serviço contém duas formas de ser execução, o ataque de forma direta (sem disfarce de IP), onde o mesmo tem sua interrupção adicionando um filtro de IP ao firewall, e o ataque de mascaramento (com disfarce de IP) chamado também de IP Spoofing, que aproveita

de brechas de segurança em comutadores de rede que não contém verificação de endereço de remetente.

GOMES (2015) relata o tipo de ataque de negação de serviço Syn Flooding onde máquinas que utilizam o protocolo TCP são os alvos, este ataque explora o processo do handshake (SYN, SYN-ACK e ACK), este processo consiste no envio dos três pacotes utilizado para iniciar uma conexão entre cliente e servidor, o servidor utiliza uma estrutura de dados para armazenar os dados de conexão (handshake) em sua memória chamada de TCB (Transmission Control Block). Conforme GOMES (2015), para estabelecer uma conexão normal, o cliente envia um pacote “SYN” para o servidor solicitando a conexão, o servidor ao receber esse pacote responde ao cliente com “SYN-ACK” (confirmação de recebimento), o servidor reserva um TCB para essa conexão no estado “SYN-RECEIVED”, isso indica que a conexão não teve sua validade confirmada e esta parcialmente aberta. GOMES (2015) explica que após o cliente receber o “SYN-ACK”, envia a resposta “ACK” ao servidor e assim tendo a conexão aberta e o TCB tem seu estado modificado para “ESTABLISHED”. O ataque Syn Flooding segundo GOMES (2015), explora essa funcionalidade de estabelecimento de conexão entre cliente e servidor, o atacante (cliente disfarçado) envia vários requisições de pacotes “SYN” para o servidor que responde com “SYN-ACK”, o atacante fica sem enviar a resposta “ACK”, assim o servidor armazena informação na TCB até que sua capacidade de memória seja esgotada, impossibilitando que o servidor consiga processar mais requisições.

2.6.2 Ataque de Força Bruta

Segundo DEFENSORWEB (2019), o ataque de força bruta (Bruteforce) é realizado com a utilização de ferramentas de software, efetua tentativas de várias combinações de senhas na página (URL) administrativa de um site ou serviço. DEFENSORWEB (2019) relata que um ataque de força bruta ocasiona a indisponibilidade de conexão com site, tornando o servidor sem resposta (offline) com todos os recursos utilizados no ataque.

A CERT (2017) relata que um ataque de força bruta consiste em adivinhar, por tentativa e erro, um nome de usuário e senha, para assim executar processos

que disponibilizam acesso a sites, serviços, servidores com os mesmos privilégios do usuário descoberto.

CERT (2017) explica que se o atacante obtiver o conhecimento de acesso com nome de usuário e senha, o mesmo pode efetuar ações maliciosas:

1. Trocar a senha do administrador ou usuário, impossibilitando o acesso ao site ou servidor invadido;
2. Invadir o serviço de e-mail para ter acesso às mensagens e lista de contatos, além de enviar mensagens se passando por outra pessoa;
3. Acessar rede social privada e enviar mensagens aos seus seguidores contendo códigos maliciosos;
4. Invadir servidor e modificar permissões de usuários, executar ações, apagar arquivos, obter informações confidenciais e instalar códigos maliciosos.

O ataque de força bruta tipo dicionário conforme CERT (2017) ocorre com o uso de ferramentas automatizadas com tentativas de adivinhação, baseada em dicionários de diferentes idiomas, com listas de palavras comumente usadas, substituição de caracteres, informações pessoais de conhecimento prévio do atacante, sequências numéricas e de teclado.

A realização de um ataque de força bruta segundo CERT (2017) pode resultar em um ataque de negação de serviço, devido á sobrecarga produzida pela grande quantidade de tentativas realizadas em um pequeno período de tempo.

2.7 SOFTWARE LIVRE

SABINO (2009) relata a vantagem de se adotar um software livre, onde o mesmo evita a dependência de um fornecedor, assim trazendo uma economia financeira, um dado que normalmente é necessário pagar por atualizações e novas versões do sistema, quando o software tem sua arquitetura fechada.

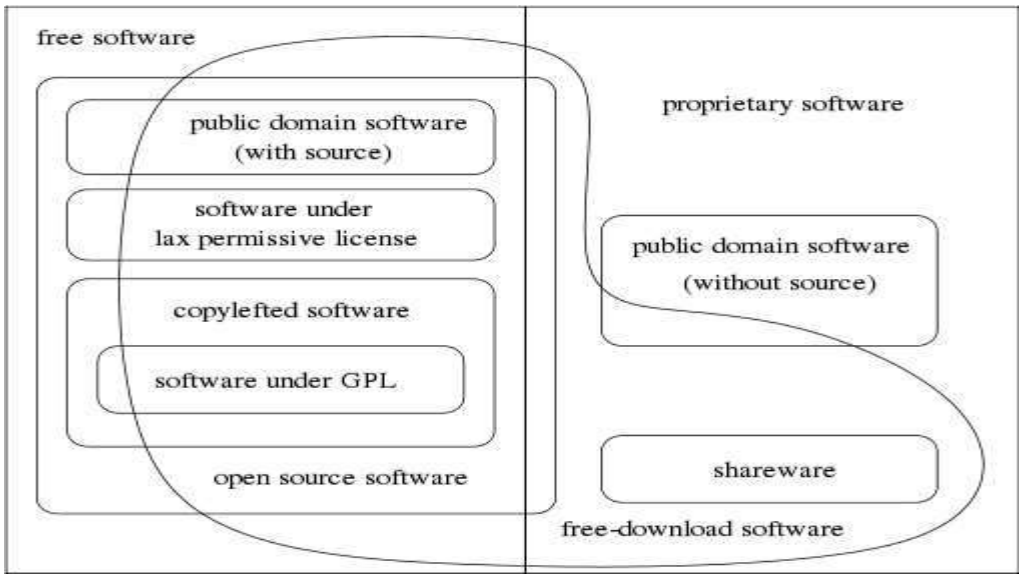
ROSADO (2016) explica que o software livre é um conceito de extrema importância na realidade das tecnologias de informação, e para ser considerado como tal, precisa obedecer a certos parâmetros e características de liberdade. Esta ideia conforme ROSADO (2016) iniciou com Richard Stallman, criador do GNU (sistema operacional composto por software livre) e fundador da Free Software

Foundation ou FSF (entidade sem fins lucrativos criados para servir de base ao movimento do software livre).

STALLMAN (2018) entendeu o software livre como uma referência á liberdade, onde os usuários têm a possibilidade de copiar, executar, distribuir, estudar, modificar e aperfeiçoar o software.

GNU (2019) demonstra por um diagrama na Figura 5 criado por Chao-Kuei as diferentes categorias de software, onde o software livre (free software) compõe todos os domínios de distribuição (public domain, software under, copylefted software, software under GPL, open source software, shareware) e é distribuído de forma livre (free-download software), diferenciando apenas do software proprietário (proprietary software), que distribui para domínios de distribuição, mas sem o código fonte (public domain software without source).

Figura 5. Diagrama de Chao-Kuei.



Fonte: GNU (2019).

O GNU (2019) e a FSF (2019) afirma que para os usuários de um programa de software livre se aplica quatro liberdades essências:

1. Liberdade zero: A liberdade de executar o programa como quiser, para qualquer propósito;
2. Liberdade um: A liberdade de estudar como o programa funciona, e adaptar ás suas necessidades com a precondição que tenha acesso ao código fonte;

3. Liberdade dois: A liberdade de redistribuir cópias e assim você pode ajudar seu vizinho;
4. Liberdade três: A liberdade de melhorar o programa com a precondição que tenha acesso ao código fonte, lançar ao público estas melhorias, beneficiando toda a comunidade.

O software livre conforme STALLMAN (2018) deve apresentar a liberdade dos usuários executarem o programa para qualquer propósito, como copiar e distribuir suas versões para amigos e colegas de trabalho, com intuito de ajudar ambos, estudar como o programa funciona, tendo em mãos acesso a todo o código fonte do programa, para modificar, aperfeiçoar e adaptar ás suas necessidades.

A OSI (2019) explica o rótulo código aberto (open source), criado em uma reunião em três de fevereiro de 1998 na cidade de Palo Alto Califórnia, logo após o anúncio do lançamento do código fonte do Netscape. A OSI (2019) que é uma organização fundada em conjunto por Eric Raymond e Bruce Perens no final de fevereiro de 1998, com a missão de explicar e proteger os softwares de código aberto.

ROSADO (2016) relata que a OSI foi criada com o intuito de educar e defender a utilização de código aberto, é responsável pela revisão e aprovação de licenças open source, em comparação com as liberdades da FSF é mais flexível.

OSI (2019) definiu dez requisitos para que o software possa ser considerado open source:

1. Distribuição livre;
2. Acesso ao código fonte;
3. Permissão para criação de trabalhos derivados;
4. Integridade do código fonte do autor;
5. Não discriminação contra pessoas ou grupos;
6. Nenhuma discriminação contra áreas de atuação;
7. Distribuição de licença;
8. Licença não deve ser específica para um produto;
9. Licença não deve restringir a outros softwares;
10. Licença deve ser neutra em termos de tecnologia.

De acordo com ROSADO (2016) o software livre não se pode caracterizar como software gratuito, este pode ser comercializado desde que não esteja em colisão às regras base da GNU e FSF. ROSADO (2016) explica que existe a

possibilidade de um software proprietário gratuito não tenha a necessidade de comprar a licença de utilização, mas, no entanto, não ser possível a alteração, atualização ou utilização para seu uso total.

A maioria das licenças que acompanham o software livre segundo ROSADO (2016), permitem a modificação e redistribuição, prática estas que são geralmente proibidas por legislações internacionais copyright, propriedade intelectual sobre o que foi criado. Para não infringir a legislação do copyright conforme ROSADO (2016), foi criado o copyleft, que determina que qualquer trabalho derivado possa ser distribuído sob os mesmos termos da licença original, definindo que as cópias, alterações e distribuições podem ser feitas por forma a garantir as liberdades de modificar e distribuir o software assim licenciado.

ROSADO (2016) apresenta algumas das licenças mais utilizadas em software open source:

1. GPL – GNU General Public License é a licença que acompanha os softwares desenvolvidos pelo projeto GNU, esta licença disponibiliza a liberdade de cópia, alteração e distribuição do software por ela protegido bem como a segurança no sentido em que não deve ser adicionadas restrições à mesma descrita na licença, software licenciado com GPL não pode ser utilizado em software proprietário;
2. BSD – Berkeley Software Distribution foi criada originalmente pela Universidade da Califórnia em Berkeley, é a licença que permite quase tudo desde que o criador/autor original seja citado, ao contrário da licença GPL, software com esta licença pode ser incorporado em software proprietário;
3. MIT/X11 – Licença criada pelo Massachusetts Institute of Technology (MIT), uma licença permissiva, semelhante à BSD, acompanha a restrição que o software deverá ser acompanhado da licença;
4. Apache - Licença permissiva e utilizada em um projeto conhecido do software livre como servidor Web Apache. Esta licença apresenta condições para a redistribuição tais como os direitos da licença são perpétuos, são garantidos mundialmente, são gratuitos e sem qualquer tipo de royalties (quantia paga por alguém ao autor/criador pelo direito de explorar, usar e comercializar), não são exclusivos e os direitos não podem ser retirados, sendo irrevogáveis;

5. DFSG – Licença Debian Free Software Guidelines é um contrato social perante a comunidade de software livre, cumpre os critérios da licença GPL. Promete que o sistema Debian e todos os seus componentes serão livres;
6. Freeware – Licença que disponibiliza o software gratuitamente, mas ele não é livre;
7. Shareware – Licença que permite a sua distribuição mediante pagamento da licença, no entanto não permite a distribuição do código.

2.8 VMWARE WORKSTATION PLAYER

O VMware Workstation Player é um aplicativo de virtualização de desktop que se encontra disponível gratuitamente para uso pessoal e educacional conforme relata VMWARE (2019).

VMWARE (2019) destaca que com quase 20 anos de desenvolvimento, o VMware Workstation Player foi criado com base na mesma plataforma que o VMware Workstation Pro e o VSphere, tornando o mesmo uma das soluções mais maduras e estáveis para virtualização local de desktops.

GONZAGA (2018) formulou uma pesquisa onde foi destacado o uso da ferramenta de virtualização VMware como um dos principais softwares utilizados por empresas de TI nos estados da região Sul do Brasil.

FERREIRA (2018) denomina o software VMware com um Hipervisor, uma camada de software situada entre o nível de hardware e o sistema operacional. O Hipervisor de acordo com FERREIRA (2018) tem a responsabilidade de controlar o acesso do sistema operacional visitante (máquina virtual) aos dispositivos de hardware. FERREIRA (2018) complementa que o VMware Player (software gratuito) é denominado de hosted (hospedado), software de virtualização que se executa sobre um sistema operacional pré-existente.

2.9 CENTOS

O sistema operacional CentOS Linux é uma distribuição com uma plataforma gerenciável, previsível, reproduzível e estável derivada das fontes do sistema operacional Red Hat Enterprise Linux (RHEL). CENTOS (2019).

A SOFTWARE LIVRE (2009) relata que a distribuição CentOS proporciona acesso aos inúmeros softwares padrão de indústria derivado do código fonte que é distribuído gratuitamente do sistema operacional RHEL, mantida pelo CentOS Project. De acordo SOFTWARE LIVRE (2009), o CentOS tem total compatibilidade com o RHEL, disponibilizando o mesmo nível de segurança através de updates, diferenciando que a versão enterprise tem suporte pago na aquisição.

Conforme CENTOS (2019), a distribuição não tem custo e é livre para redistribuir, além de que cada versão do CentOS é mantida até 10 anos por meio de atualizações de segurança e suporte. De acordo com CENTOS (2019), uma nova versão é lançada aproximadamente a cada dois anos, cada versão é atualizada periodicamente (aproximadamente a cada seis meses) para suporte a atualização de hardware. CENTOS (2019) complementam que todos estes aspectos resultam em um ambiente Linux seguro, de baixa manutenção, confiável, previsível e reproduzível.

2.10 OPENSTACK

O OpenStack é um sistema operacional em nuvem que controla grandes pools (conjunto fixo pré-definido) de recursos de computação, armazenamento e rede em todo o datacenter, tudo gerenciado por um painel que fornece aos administradores o controle e capacita seus usuários a provisionar recursos por meio de uma interface web. OPENSTACK (2019).

BARBOSA (2018) relata que o OpenStack é uma plataforma de nuvem de código aberto que surgiu em 2010 através de uma iniciativa da Rackspace Hosting e da NASA. Sua estrutura conforme BARBOSA (2018) era baseada na plataforma Nebula, da NASA e no sistema de arquivos em nuvem da Rackspace, fornece suporte a diferentes tipos de hipervisores (Xen, KVM, HyperV, Qemu).

O OpenStack segundo BARBOSA (2018), foi projetado para fornecer serviços em um ambiente de Infraestrutura como Serviço com finalidade de uma implantação de nuvens públicas e privadas.

OPENSTACK (2019) destaca que suas séries são desenvolvidas e lançadas em ciclos de seis meses, após o lançamento inicial de alguma distribuição, lançamentos de pontos estáveis adicionais são lançados posteriormente. Na Figura 6 é demonstrado os lançamentos das séries do OpenStack, com informações de

nome da série (series), seu status atual (status), data de lançamento inicial, a próxima fase da série e a data de sua finalização de desenvolvimento (data fim da vida).

Figura 6. Lançamento das séries do OpenStack.

Series	Status	Data de lançamento inicial	Próxima fase	Data fim da vida
Train	Futuro	2019-10-16 Estimado (cronograma)	Desenvolvimento estimado em *2019-04-11*	
Stein	Desenvolvimento	2019-04-10 Estimado (cronograma)	Mantido estimado em *2019-04-10*	
Rocky	Mantido	2018-08-30	Manutenção prolongada estimada em *2020-02-24*	
Queens	Mantido	2018-02-28	Manutenção prolongada estimada em *2019-08-25*	
Pike	Mantido	2017-08-30	Manutenção estendida estimada em *2019-03-03*	
Ocata	Manutenção Extendida	2017-02-22	Sem estimativa de manutenção mantida.	
Newton	Fim da vida	2016-10-06		2017-10-25
Mitaka	Fim da vida	2016-04-07		2017-04-10
Liberty	Fim da vida	2015-10-15		2016-11-17
Kilo	Fim da vida	2015-04-30		2016-05-02
Juno	Fim da vida	2014-10-16		2015-12-07
Icehouse	Fim da vida	2014-04-17		2015-07-02
Havana	Fim da vida	2013-10-17		2014-09-30
Grizzly	Fim da vida	2013-04-04		2014-03-29
Folsom	Fim da vida	2012-09-27		2013-11-19
Essex	Fim da vida	2012-04-05		2013-05-06
Diablo	Fim da vida	2011-09-22		2013-06-06
Cactus	Fim da vida	2011-04-15		
Bexar	Fim da vida	2011-02-03		
Austin	Fim da vida	2010-10-21		

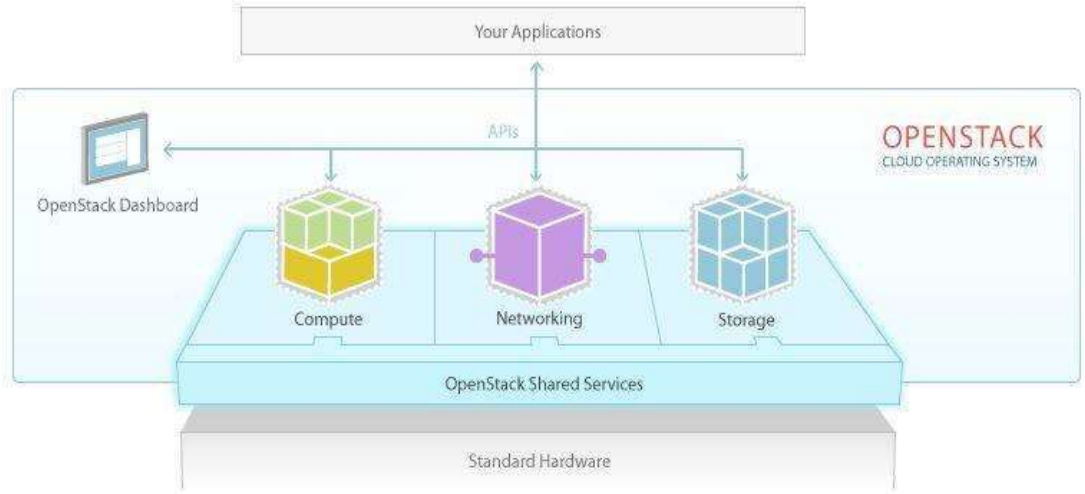
Fonte: Autor, adaptado OPENSTACK (2019).

SOARES (2015) relata que OpenStack é apoiado por mais de 800 organizações, buscando produzir uma plataforma onipresente de computação em nuvem de código aberto para nuvens públicas e privadas, baseando-se nos princípios de entrega de serviços sob demanda de três principais pilares: computação, redes e meios de armazenamento como é demonstrado na Figura 7.

SOARES (2015) complementa sobre os serviços do OpenStack que permitem gestão, orquestração e automação dos recursos de TI que envolvem a construção de uma nuvem privada ou a utilização de nuvens públicas.

A Figura 7 demonstra os pilares principais de serviço do OpenStack, onde a camada relacionada a aplicações (Your applications), se conecta a administração de acesso (OpenStack Dashboard), que posteriormente se conectam com os elementos padrões de recursos para computação (Compute), redes (Networking) e armazenamento (Storage), que são os serviços disponibilizados pelo OpenStack (OpenStack Shared Services).

Figura 7. Pilares de serviços do OpenStack.



Fonte: ICLOUD (2019).

SILVEIRA (2014) relata outros componentes que auxiliam na instalação e na operação da nuvem, como a interface web de administração nomeada de dashboard e a OpenStack Shared Services, que é composta por autenticação, autorização, catálogo de serviços (Identity) e registro de imagens de máquinas virtuais (Image Service).

O OpenStack conforme SOARES (2015) tem gerado uma larga disseminação no âmbito de modelo de nuvem utilizando código aberto, estimulando o baixo custo de sua implementação e manutenção, além de contar com uma grande comunidade para compartilhar conhecimento, técnicas de implantações e correção de erros (bugs).

BARBOSA (2018) define os principais serviços tipo Core que são fornecidos pelo OpenStack conforme a Tabela 2.

Tabela 2. Principais serviços do OpenStack.

Nova (Cloud Management)	Gerencia o ciclo de vidas das instâncias de máquinas virtuais.
Keystone (Identity Service)	Valida as credenciais e informações dos usuários e grupos de usuários. Provê dado sobre projetos e domínios. Valida e administra solicitações de autenticação. Fornece catálogo de todos os serviços e fornece um mecanismo de autorização.
Glance (Image Service)	Armazena e recupera imagens de discos das máquinas virtuais, através de uma API RESTful (interface de programação de aplicação que utiliza requisições HTTP para extrair, inserir, postar e deletar dados).
Swift (Object Storage)	Armazena e recupera objetos de dados não estruturados por meio de uma API RESTful.
Cinder (Block Storage)	Prover um block storage (armazenamento em bloco) persistente para as instâncias em execução, garantindo que continue existindo mesmo após uma máquina virtual seja excluída.
Horizon (Control Panel/Dashboard)	Interface gráfica para todos os módulos, dinamizando o gerenciamento dos recursos da nuvem.
Ceilometer (Accounting)	Fornece o monitoramento e medição dos recursos da nuvem, facilitando tomadas de decisões de prevenção e correção quando ocorrer possíveis falhas.
Heat (Orchestration)	Mecanismo que permite uma automatização da implantação da

	infraestrutura, visto que gerencia o ciclo de vida do ambiente.
Neutron	Prover rede como serviço, criação de infraestruturas de redes dinâmicas. Fornece conectividade entre máquinas virtuais por meio de uma API.

Fonte: BARBOSA (2018).

OPENSTACK (2019) relata serviços opcionais e complementares do OpenStack, o projeto Ironic que é um projeto que fornece máquinas baremetal (em oposição a virtuais) que gerencia o hardware por meio de protocolos de gerenciamento remoto e o projeto Trove, que é o banco de dados como um serviço em nuvem. O projeto Sahara conforme OPENSTACK (2018) é outro serviço complementar e opcional que visa fornecer aos usuários um meio simples de provisionar estruturas de processamento de dados (como Apache Hadoop, Apache Spark e Apache Storm) na nuvem.

2.10.1 Serviço de Computação Nova

O projeto Nova conforme OPENSTACK (2019) tem a função de provisionar instâncias de computação, também conhecidas como servidores virtuais. OPENSTACK (2019) explica que o projeto Nova suporta a criação de máquinas virtuais e servidores baremetal (servidor físico de inquilino único).

REDHAT (2019) relata que o projeto Nova é uma ferramenta de acesso e gerenciamento total para os recursos computacionais do OpenStack, incluindo programações, criações e exclusões.

O projeto Nova conforme SILVEIRA (2014) se comunica com grande parte dos serviços do OpenStack, é considerada a principal parte da infraestrutura como serviço, requisitando serviços de rede, armazenamento, identidade e imagem. SILVEIRA (2014) relata os componentes de serviços que compõem o projeto Nova:

1. Nova-api: Suporta as API’s do tipo OpenStack Compute, Amazon EC2;
2. Nova-api-metadata: Recebe requisições das máquinas virtuais do tipo metadado;

3. Nova-compute: Responsável por receber ações de criar, pausar ou terminar uma VM (virtual machine), através da fila de mensagens;
4. Nova-scheduler: Identifica requisições de VM através da fila de mensagens e determina em qual nó de computação a VM será alocada;
5. Nova-conductor: Intermedia a comunicação entre nova-compute e o banco de dados, o que elimina o acesso direto;
6. Nova-network: Através da fila de mensagens recebe requisições de rede para execução de manipulação de rede;
7. Nova-dhcpbridge: Identifica endereços de IP alocados por VM e registra no banco de dados;
8. Nova-consoleauth: Autoriza tokens (gerador eletrônico de senhas) para usuários que utilizam serviço proxy de console;
9. Nova-novncproxy: Intermedia conexões VNC (Virtua Network Computing) para acessar VM em execução. Suporta clientes baseados em navegadores web;
10. Nova-xvpnvncproxy: Intermedia conexões VNC para acessar VM em execução. Suporta cliente Java desenvolvido pelo OpenStack;
11. Nova-cert: Gerencia certificados x509 (certificado digital que utiliza a infraestrutura de uma chave pública x509 padrão, para associar uma chave pública a uma identidade contida em um certificado);
12. Nova-client: Usuário utiliza para interagir com Nova;
13. Nova-manage: Utilitário para administradores da nuvem.

2.10.2 Serviço de Identidade Keystone

SILVEIRA (2014) relata a importância do Keystone no OpenStack, onde os recursos necessitam de restrição de acesso, o Keystone auxilia no gerenciamento da identificação, autenticação e nos privilégios de usuários que auxilia na segurança e confiabilidade do ambiente de nuvem.

O Keystone é um serviço do OPENSTACK (2019) que fornece autenticação de cliente via API REST, onde os clientes obtêm esse token de acesso para outras API’s de serviço, fornecendo suas credenciais válidas ao serviço de autenticação.

PYTHON (2019) relata que o Keystone fornece mecanismos de autenticação, autorização e descoberta de serviço via protocolo HTTP para os serviços do projeto OpenStack.

O serviço de identidade Keystone conforme OPENSTACK (2019) tem integração com serviços de diretório como o LDAP, permite múltiplas formas de autenticação, como token ou usuário e senha.

OPENSTACK (2019) relata que o Keystone gerencia os catálogos de serviços pertencentes ao OpenStack, provendo aos usuários uma lista dos serviços instalados na nuvem, e para os administradores, provendo um ambiente centralizado para criar usuários, projetos, permissões e políticas.

2.10.3 Serviço de Imagem Glance

OPENSTACK (2018) relata que o serviço de imagem Glance fornece aos usuários um modo de fazer upload de imagens e definições de metadados, os serviços de imagem incluem a descoberta, registro e recuperação de imagens de máquinas virtuais (VM). O Glance conforme OPENSTACK (2018) tem uma API RESTful que permite a consulta dos metadados da imagem da VM, como a recuperação da imagem real. Segundo SILVEIRA (2014) o Glance apenas armazena metadados e informações da imagem, provê um repositório central de imagens para os usuários, mas não é responsável por armazenar a imagem.

ROSADO (2016) descreve os serviços internos do serviço Glance:

1. Glance-api: Principal serviço do projeto que recebe os pedidos referente ás imagens com as quais serão criadas as máquinas virtuais;
2. Glance-registry: Serviço responsável por armazenas e tratar metadados referentes ás imagens, como o seu tamanho e tipo;
3. Repositório para ficheiros das imagens: O Glance suporta vários tipos de repositórios, como o próprio sistema de ficheiros do servidor onde está instalado o serviço, o próprio serviço Swift do OpenStack, entre outros.

2.10.4 Serviço de Armazenamento de Objeto Swift

O projeto Object Store (Armazenamento de Objeto), conhecido como Swift conforme OPENSTACK (2019) oferece um software de armazenamento em nuvem, para que o usuário possa armazenar e recuperar muitos dados com uma API simples. OPENSTACK (2019) relata que o Swift é construído para dimensionamento e otimizado para durabilidade, disponibilidade e simultaneidade em todo o conjunto de dados, ideal para armazenar dados não estruturados que podem crescer sem limite.

O Swift é usado para armazenamento de dados redundante e escalável segundo OPENSTACK (2019), usando clusters de servidores padronizados para armazenar petabytes de dados. OPENSTACK (2019) relata que o Swift é um sistema de armazenamento de longo prazo para grandes quantidades de dados estáticos que podem ser recuperados e atualizados, utiliza uma arquitetura distribuída sem nenhum ponto central de controle, proporcionando maior escalabilidade, redundância e permanência. Os objetos são gravados conforme OPENSTACK (2019) em vários dispositivos de hardware, com o software OpenStack como responsável por garantir a replicação e a integridade dos dados no cluster, onde os clusters de armazenamento são dimensionados horizontalmente adicionando novos nós. Se um nó falhar, OPENSTACK (2019) explica que o software OpenStack replica seu conteúdo de outros nós ativos, utilizando lógica de software para garantir replicação e distribuição de dados em diferentes dispositivos.

MIRANDA (2017) relata como a arquitetura do Swift é composta:

1. Servidor Proxy: Responsável por conectar as demais arquiteturas ao Swift;
2. Servidor Ring: Mapeamento entre o nome das entidades que estão armazenadas em disco a sua disposição física. Contém a responsabilidade de estabelecer quais dispositivos serão ativados ou replicados em casos que ocorra algum tipo de falha;
3. Servidores de Objeto: Serve como um dispositivo de armazenamento de objetos, armazenados como arquivos binários;
4. Servidores Contêiner: Cataloga os objetos que estão presentes dentro dos contêineres, não contém a localização física de cada objeto, mas tem o conhecimento de onde ele se aloja em cada um dos contêineres;

5. Servidores de Conta: Cataloga os contêineres que pertencem a uma determinada conta;
6. Replicador: Conserva o sistema em um estado constante em caso de falha ou erro.

2.10.5 Serviço de Armazenamento em Bloco Cinder

O serviço Cinder conforme OPENSTACK (2019) disponibiliza um armazenamento em bloco para a nuvem OpenStack, projetado de forma a apresentar recursos de armazenamento para usuários finais. OPENSTACK (2019) complementa que o serviço Cinder virtualiza o gerenciamento de dispositivos de armazenamento em bloco e fornece aos usuários finais uma API de autoatendimento, para solicitar e consumir esses recursos, sem exigir conhecimento de onde seu armazenamento está realmente implantado ou em que tipo de dispositivo.

Segundo ROUSE (2017), em um cenário típico, os volumes Cinder fornecem armazenamento persistente para máquinas virtuais de tipo guest, conhecidas como instâncias, gerenciadas pelo OpenStack. Conforme ROUSE (2017) o serviço Cinder permite disponibilizar um catálogo de dispositivos de armazenamento baseados em blocos com características diferentes. ROUSE (2017) exemplifica este catálogo um tipo de volume de armazenamento potencial, pode ser uma opção de alto desempenho para armazenamento de banco de dados, enquanto outro volume de armazenamento pode ser dedicado ao armazenamento de backup com desempenho inferior. ROUSE (2017) complementa que o serviço Cinder possui também recursos básicos de armazenamento, como replicação, gerenciamento de snapshot (registro do estado de um sistema em determinado ponto no tempo) e clones de volume.

2.10.6 Serviço de Controle Horizon

De acordo com OPENSTACK (2017) o serviço Horizon é a implementação canônica do dashboard do OpenStack, fornece uma interface de usuário baseada na web, demonstrando muitos serviços do OpenStack em andamento.

ROUSE (2017) relata que o Horizon é uma interface gráfica baseada na web, onde administradores e usuários pode acessar a nuvem para gerenciar serviços de computação, armazenamento e rede do OpenStack.

Conforme ROUSE (2017) o Horizon tem funcionalidades de lançar instâncias de máquinas virtuais, visualizar o tamanho e o estado atual da implantação de nuvem do OpenStack, gerenciar redes e definir limites nos recursos de nuvem disponíveis para os usuários. Estas e outras funcionalidades de serviço do OpenStack conforme ROUSE (2017), são acessadas por meio de interfaces de programação de aplicativos (API's). ROUSE (2017) complementa que para os usuários finais, o Horizon atua como um portal de autoatendimento para provisionar recursos da nuvem.

2.10.7 Serviço de Telemetria Ceilometer

O projeto Ceilometer conforme OPENSTACK (2017) é um serviço de coleta de dados que fornece a capacidade de normalizar e transformar dados em todos os componentes principais do OpenStack atuais, com o trabalho em andamento para suportar futuros componentes do OpenStack.

OPENSTACK (2017) relata que o Ceilometer é um componente do projeto de telemetria (tecnologia que permite medição e comunicação de informações), onde seus dados podem ser usados para fornecer faturamento ao cliente, rastreamento de recursos e alarme em todos os componentes principais do OpenStack.

Segundo OPENSTACK (2019) os requisitos de telemetria de um ambiente OpenStack são vastos e variados, incluem casos de uso, como medição, monitoramento, alarme e entre outros.

O Ceilometer é um serviço que fornece medições de recursos em nuvem segundo REDHAT (2019), que relata os componentes de seu serviço:

1. Ceilometer-agent-compute: Executado em cada nó de compute para pesquisar estatísticas de utilização de recursos;
2. Ceilometer-agent-central: Um agente que é executado em um servidor de gerenciamento central para pesquisar estatísticas de utilização sobre recursos não vinculados a instâncias ou a nós do compute:

3. Ceilometer-coletor: Agente que é executado em um ou mais servidores de gerenciamento central para monitorar as filas de mensagens. As mensagens de notificação são processadas e transformadas em mensagens de telemetria e enviadas de volta para o barramento de mensagens usando o tópico apropriado. As mensagens de telemetria são gravadas no armazenamento de dados sem modificação;
4. Ceilometer-alarm-evaluator: Serviço de alarme que aciona transições de estado em alarmes;
5. Ceilometer-alarm-notifier: Serviço de alarme que executa ações necessárias quando os alarmes são acionados;
6. Ceilometer-notification: Um agente que envia as métricas para o serviço de coletor de vários serviços do OpenStack;
7. MongoDB database: Para dados de uso coletados de agentes coletores. Apenas os agentes coletores e o servidor de API têm acesso ao banco de dados;
8. Ceilometer-api: É executado em um ou mais servidores de gerenciamento central para fornecer acesso aos dados no banco de dados;
9. RabbitMQ server: Fornece a fila de mensagens do AMQP. O RabbitMQ (também usado por outros serviços) lida com o gerenciamento de transações do OpenStack, incluindo enfileiramento, distribuição, segurança, gerenciamento, armazenamento em cluster e federação. O envio de mensagens torna-se especialmente importante quando uma implantação do OpenStack é dimensionada e seus serviços estão sendo executados em várias máquinas.

2.10.8 Serviço de Orquestração Heat

Segundo OPENSTACK (2019) o projeto Heat implementa um mecanismo de orquestração, para iniciar vários aplicativos compostos em nuvem com base em modelos na forma de arquivos de texto que podem ser tratados como código. OPENSTACK (2019) relata que a missão do programa OpenStack Orchestration é criar um serviço acessível por humanos e máquinas para gerenciar todo o ciclo de vida da infraestrutura e dos aplicativos nas nuvens do OpenStack.

O Heat conforme OPENSTACK (2018) é um serviço para orquestrar aplicativos em nuvem compostos usando um formato de modelo declarativo por meio de uma API REST nativa do OpenStack.

OPENSTACK (2018) relata as características do serviço Heat:

1. O Heat fornece uma orquestração baseada em modelos para descrever um aplicativo em nuvem, executando chamadas de API do OpenStack apropriadas para gerar aplicativos em nuvem e em execução;
2. Template Heat descreve a infraestrutura de um aplicativo em nuvem em arquivos de texto que podem ser lidos e gravados por humanos e pode ser gerenciado por ferramentas de controle de versão;
3. Templates especificam as relações entre recursos (por exemplo, este volume esta conectado a este servidor). Isso permite que o Heat chame as API's do OpenStack para criar toda a sua infraestrutura na ordem correta para iniciar completamente o aplicativo;
4. O software integra outros componentes do OpenStack. Os templates permitem a criação da maioria dos tipos de recursos do OpenStack (como instâncias, ip's flutuantes, volumes, grupos de seguranças, usuários etc.), além de algumas funcionalidades mais avançadas, como alta disponibilidade de instâncias, escalonamento automático de instâncias e pilhas aninhadas;
5. O Heat gerencia principalmente a infraestrutura, mas os templates integram-se bem ás ferramentas de gerenciamento de configuração de software;
6. Operadores podem personalizar os recursos do Heat instalando plug-ins.

REDHAT (2019) relata os componentes do serviço de orquestração Heat:

1. Openstack-heat: Uma ferramenta CLI que se comunica com o heat-api para executar API's do AWS CloudFormation;
2. Openstack-heat-api: Uma API REST nativa do OpenStack que processa solicitações de API enviando-as ao mecanismo do Heat por meio do RPC;
3. Openstack-heat-api-cfn: Fornece uma API de consulta da AWS compatível com o AWS CloudFormation e processa as solicitações da API enviando-as aos mecanismos do Heat por meio do RPC;
4. Openstack-heat-engine: Orquestra o lançamento de template e fornece eventos de volta ao consumidor da API;

5. Openstack-heat-api-cloudwatch: Fornece monitoramento (coleta de métricas) para o serviço de orquestração;
6. Openstack-heat-cfntools: Pacote de scripts auxiliares (por exemplo, cfn-hup, que manipula atualizações para metadados e executa ganchos personalizados).

2.10.9 Serviço de Rede Neutron

OPENSTACK (2019) destaca que a missão do serviço de rede Neutron é implementar serviços e bibliotecas associadas, para fornecer abstração de rede sob demanda, escalonável e independente de tecnologia. O Neutron conforme OPENSTACK (2019) é um projeto do OpenStack para fornecer “rede como um serviço” entre dispositivos de interface (VM’s), com uma API flexível permitindo que redes complexas (VLAN’s, DHCP, IPv6 e etc.) sejam construídas.

Segundo REDHAT (2019) o serviço Neutron lida com a criação e gerenciamento de uma infraestrutura de rede virtual na nuvem OpenStack. REDHAT (2019) relata que os elementos desse gerenciamento incluem redes, sub-redes, roteadores, serviços avançados com firewall e redes privadas virtuais (VPN).

OPENSTACK (2019) relata que o Neutron oferece aos locatários da nuvem uma API para criar topologias de rede avançadas, configurar políticas de rede avançada à nuvem, criar topologia de aplicativo da web de multicamadas, ativar plug-ins de inovação (arquitetura aberta e fechada de software) que introduzem recursos avançados de rede.

Conforme REDHAT (2019) o serviço Neutron fornece flexibilidade aos administradores de nuvem para decidir quais serviços individuais devem ser executados em quais sistemas físicos, é definido por software e pode reagir de maneira fácil e rápida ás mudanças de necessidade de uma rede, como criar e atribuir novos endereços IP. REDHAT (2019) exemplifica as vantagens de utilizar o Neutron:

1. Usuários podem criar redes, controlar o tráfego, conectar servidores e dispositivos a uma ou mais redes;
2. Neutron oferece modelos de rede flexíveis, para que os administradores possam alterar o modelo de rede para se adaptar ao volume e á locação;

3. IP’s podem ser dedicados ou flutuantes (IP’s flutuantes permitem o reenvio de tráfego dinâmico);
4. Neutron oferece um limite de 4094 VLAN’s (4094 redes), isso se traduz em um limite de 16 milhões de túneis na nuvem e um limite de 4094 túneis por nó de computação.

REDHAT (2019) descreve os componentes do serviço de rede Neutron conforme a Tabela 3:

Tabela 3. Componentes de serviço de rede do Neutron.

Openstack-neutron-server	Um processo (daemon) do Python que gerencia solicitações de usuários. Configurado com um plug-in que implementa as operações da API do OpenStack Neutron, usando um conjunto específico de mecanismos de rede. Plug-ins utilizados são o OpenvSwitch e LinuxBridge, que usam mecanismos de rede Linux nativos, outros plug-ins interagem com dispositivos externos ou controladores SDN.
Openstack-neutron-ml2	Plug-in que gerencia os drivers de rede, fornecendo serviços de roteamento e comutação.
Network Agents (Agentes de rede)	Serviço executado em cada nó para executar a configuração de rede local para as máquinas virtuais e os serviços de rede do nó.
Openstack-neutron-dhcp-agent	Agente que fornece serviços DHCP para redes de locatários.
Servidor RabbitMQ (rabbitmq-server)	Fornece a fila de mensagens AMQP. O RabbitMQ (também usado por outros serviços) lida com o gerenciamento de

	transações do OpenStack, incluindo enfileiramento, distribuição, segurança, gerenciamento, armazenamento em cluster e federação. O envio de mensagens torna-se especialmente importante quando uma implantação do OpenStack é dimensionada e seus serviços estão sendo executados em várias máquinas.
Base de dados (Database)	Fornece armazenamento persistente.

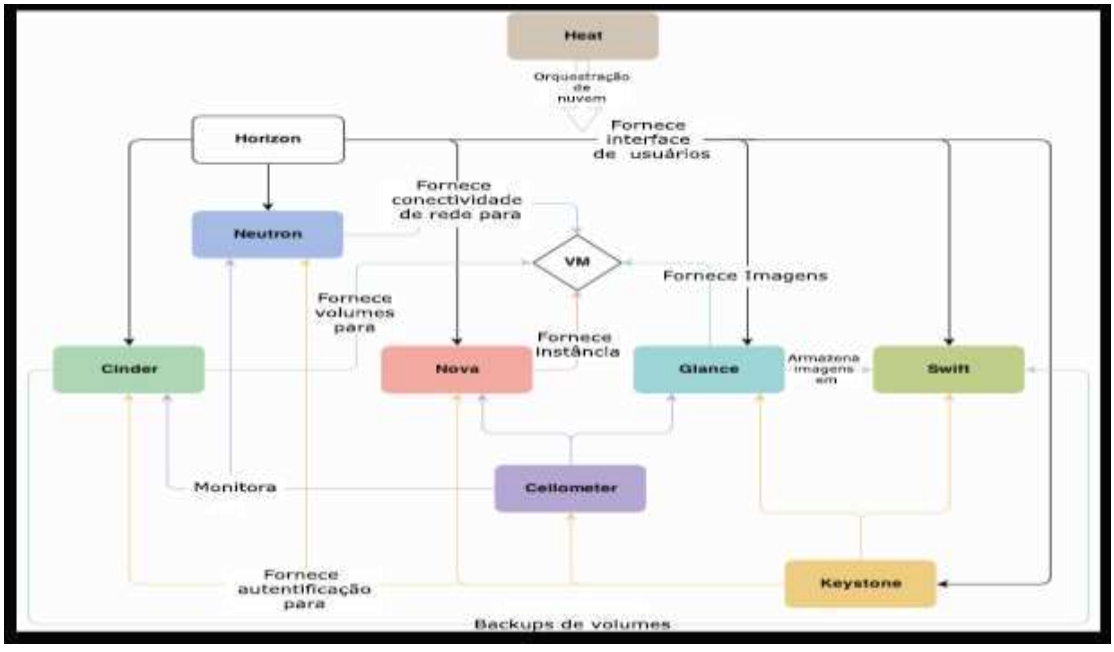
Fonte: REDHAT (2019).

3 DESCRIÇÃO DA SOLUÇÃO

O projeto propõe demonstrar softwares de âmbito livre e gratuito em específico o OpenStack, como alternativa de baixo custo e segurança. A escolha pela utilização do software OpenStack se caracteriza principalmente por ser uma ferramenta que contém uma comunidade de desenvolvedores ativos, onde se contém atualizações constantes, assim disponibilizando uma confiança de que a mesma irá conter uma base sólida de implementação.

O cenário proposto para a implementação do projeto é composto por máquina virtual. O ambiente virtual visa à orquestração de uma infraestrutura como serviço em nuvem privada. Na Figura 8 é demonstrado como os serviços do OpenStack se relacionam. O serviço Heat implementa a orquestração da nuvem, onde todos os serviços de comunicam, o serviço Horizon disponibiliza a interface gráfica, o serviço Neutron implementa a ferramenta de rede, o serviço Cinder disponibiliza armazenamento para á máquina virtual (VM), o serviço Nova disponibiliza a instância para a VM, o serviço Glance fornece a imagem para VM, o serviço Swift armazena a imagem para a VM, o serviço Ceilometer fornece telemetria e gráficos, o serviço Keystone disponibiliza a autenticação aos serviços do OpenStack.

Figura 8. Serviços do OpenStack.



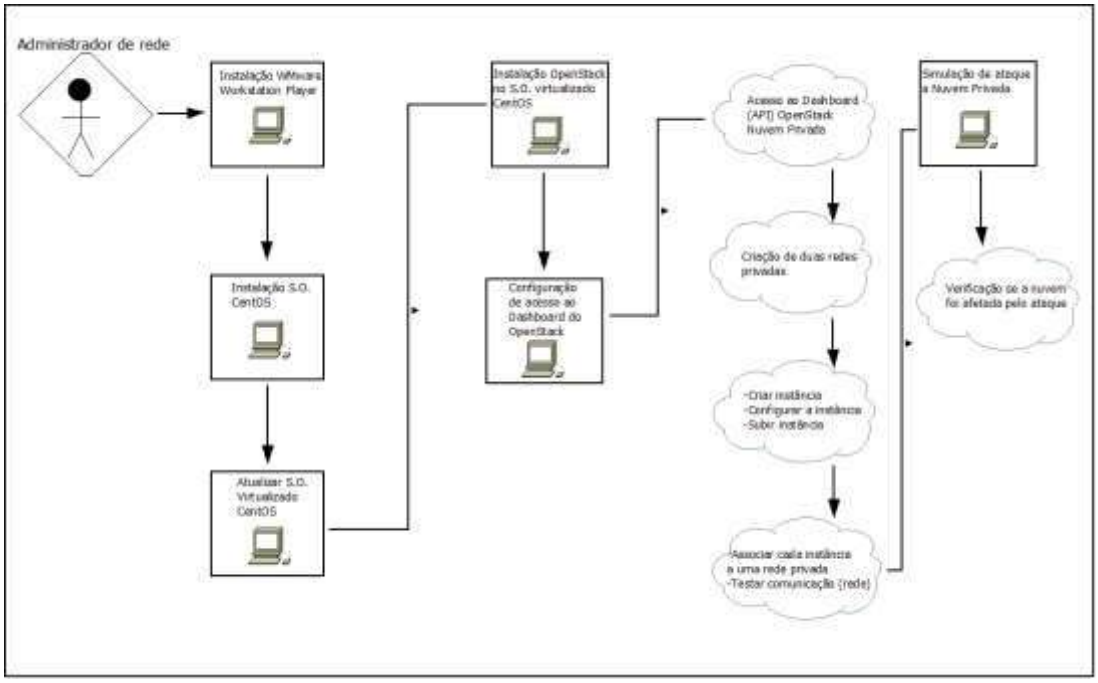
Fonte: Autor, adaptado VMWARE (2014).

O hardware utilizado no projeto é composto por um roteador DHCP Multilaser Modelo RE033 e dois microcomputadores Itautec Modelo ST4272, processador i3 2100 2ª geração 3.1ghz 3mb Lga1155, 12 Gigabytes de memória e 500 Gigabytes de armazenamento, a base de sistema operacional dos dois microcomputadores é o CentOS na versão X64-1810. O ambiente virtual será feito pelo software VMware Workstation Player na versão 15, que é disponibilizada sua utilização gratuitamente no âmbito educacional, doméstico, pessoal e não comercial. VMWARE (2019). A máquina virtual criada no VMWARE Workstation Player irá conter de configuração de memória 4 Gigabytes, capacidade de armazenamento de 40 Gigabytes, placa de rede 10/100, processador de dois núcleos 2vCPU. O sistema operacional virtualizado será o CentOS na versão X64-1810 disponibilizada no site do CENTOS (2019). Após instalação do sistema operacional virtualizado, o mesmo terá sua base de dados atualizada pelo repositório da Red Hat Enterprise. O software de implementação de infraestrutura como serviço em nuvem será o OpenStack na versão (release) Rocky. Após instalação do OpenStack será feito o acesso ao Dashboard (API), por browser de internet com endereço IP aplicado na configuração de instalação. O acesso ao Dashboard é efetuado por usuário e senha de administrador da plataforma. Na tela principal do OpenStack será efetuado a criação de duas redes distintas (rede_privada_1 e rede_privada_2) com um range de IP's diferentes, criar um roteador virtual (roteador) para a comunicação das duas redes, assim simulando uma infraestrutura como serviço. Criar duas instâncias (cirros_1 e cirros_2) com a distribuição Linux CirrOS, uma distribuição mínima do Linux segundo OPENSTACK (2019) com o tamanho de espaço de 12,1 Megabytes, projetada para uso como imagem de teste em nuvens. Cada instância criada terá como configuração 512 Megabytes de memória, quantidade de armazenamento de 1 Gigabytes, placa de rede 10/100 e processador de dois núcleos 2vCPU. A instância cirros_1 será associada a rede_privada_1, a instância cirros_2 será associada a rede_privada_2, subir as duas instâncias e testar comunicação de rede via comando "Ping", seguido do endereço IP do roteador virtual e físico.

Com a criação da infraestrutura como serviço em nuvem privada, o projeto visa efetuar um ambiente de simulação de ataque à nuvem privada, verificar como a mesma se comporta em tal situação. Na Figura 9 é descrito a elaboração de um fluxograma, onde o administrador de rede tem a detenção da orquestração da instalação do software de virtualização e suas configurações, suas respectivas

atualizações, acessos, criação das redes privadas e instâncias, associar as instâncias as respectivas redes privadas, testes de comunicação, a simulação de ataque e a verificação de disponibilidade da nuvem privada.

Figura 9. Descrição da solução.

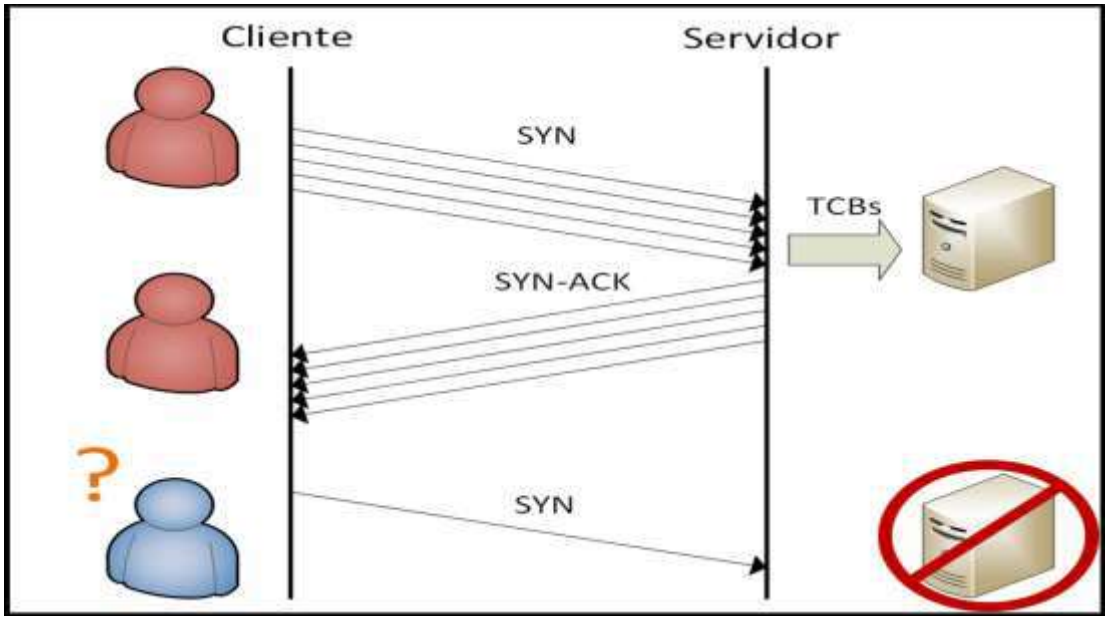


Fonte: Autor (2019).

A simulação de ataque será efetuada em ambiente virtualizado com a distribuição Kali Linux baseada no Debian na versão Kali Linux 64 Bit, esta ferramenta disponibiliza inúmeras tarefas de segurança da informação conforme KALI (2019). A máquina virtual com a distribuição Kali Linux terá configuração de 4 Gigabytes de memória, capacidade de armazenamento de 40 Gigabytes, placa de rede 10/100, processador de dois núcleos 2vCPU. Uma das tarefas a ser testada é o teste de penetração via ataque de negação de serviço (DoS), tipo SYN Flood (inundação de SYN), com o intuito na qual a simulação de ataque envia uma sequencia de requisições SYN para a nuvem privada, visando uma sobrecarga direta na camada de transporte e indiretamente na camada de aplicação do modelo OSI. A Figura 10 demonstra um ambiente de simulação de ataque à nuvem privada, onde o atacante (Kali Linux) disfarçado pela figura de cliente (figura vermelha), aplica uma sequencia de requisições SYN ao servidor (nuvem privada OpenStack), o protocolo TCB gerencia as requisições de recebimento (SYN) e envio da resposta

(SYN-ACK) ao atacante. A simulação de ataque visa que com a inundação de requisições SYN o servidor irá ficar inoperante, sem a possibilidade de resposta e acesso de um suposto cliente verdadeiro (figura azul) ao servidor.

Figura 10. Simulação de ataque SYN Flood.



Fonte: GOMES (2015).

O software da distribuição Kali Linux que será utilizado é o HPING, o ataque do HPING consiste no envio de uma grande quantidade de pacotes com flags (mecanismos lógicos) setadas SYN para a nuvem privada, o intuito deste ataque é que após o mesmo verificar a disponibilidade e resposta do serviço (comando “Ping” seguido do endereço IP da nuvem privada) conforme as requisições do ataque em andamento. Outra tarefa a ser testada é o ataque de força bruta do tipo dicionário, onde o ataque visa descobrir a senha para acesso à nuvem privada, a ferramenta do Kali Linux a ser utilizada será o THC Hydra, esta ferramenta irá atacar a página inicial de acesso á nuvem privada com tentativa de várias combinações de senha e login para acesso a mesma. Com o ataque de força bruta em andamento á nuvem privada, será verificada a resposta do serviço (comando “Ping” seguido do endereço IP da nuvem privada) e disponibilidade de acesso com login e senha de administrador credenciado.

4 METODOLOGIA

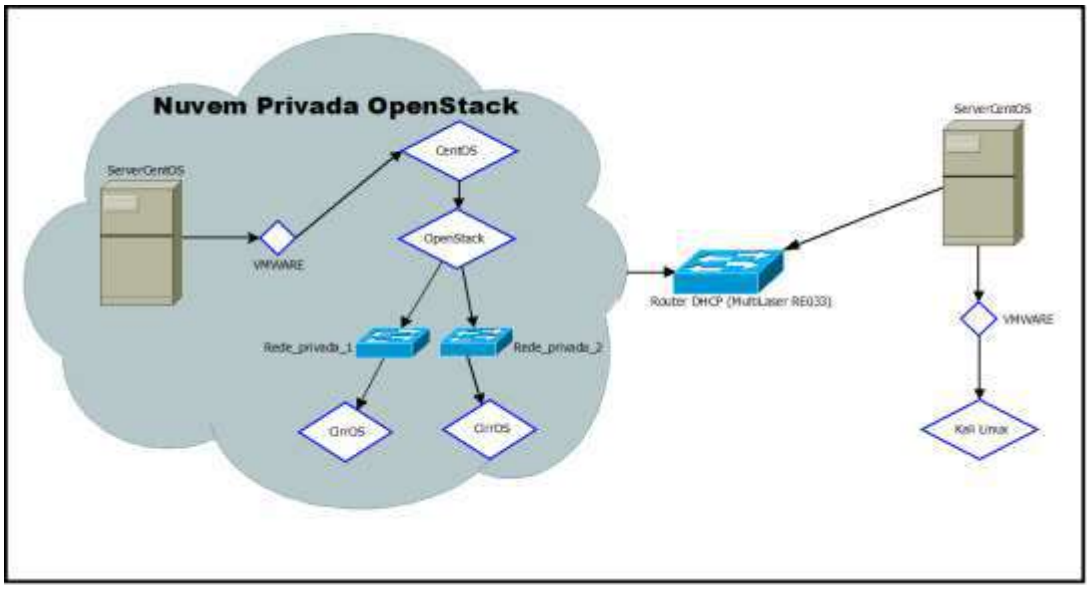
A metodologia utilizada neste projeto se caracteriza por pesquisa de método indutivo de tecnologias que englobam a Computação em Nuvem. A pesquisa foi efetuada com o método de revisão de leitura de livros, sites web, artigos e projetos. A revisão da leitura consistiu na realização de método de pesquisa do conteúdo dos conceitos sobre a Computação em Nuvem, modelos da Computação em Nuvem, arquitetura da Computação em Nuvem, a importância da adoção da Computação em Nuvem Verde para a sustentabilidade, serviços da Computação em Nuvem, segurança da Computação em Nuvem, definição e importância da utilização de software livre, ferramenta de software livre de virtualização VMware, sistema operacional de software livre CentOS, software livre de implementação de IaaS de Nuvem Privada OpenStack.

Para execução de validação de funcionamento da Nuvem Privada, foi efetuado a implementação da Nuvem Privada, com a experimentação do trabalho juntamente com a configuração necessária para o funcionamento do ambiente computacional. O método de testes implementado a Nuvem Privada se foca na disponibilidade dos serviços, processos em tempo real e segurança. A segurança terá um modelo de teste simulado de ataque à nuvem.

Na fase de desenvolvimento da Nuvem Privada e a simulação de ataque demonstrado na Figura 11, foi implementado um servidor (ServerCentOS), instalado o software de virtualização (VMWARE), virtualizado o sistema operacional CentOS, implementado a ferramenta OpenStack no sistema operacional virtualizado CentOS, efetuado a criação de duas redes privadas (rede_privada_1 e rede_privada_2) distintas conectadas um roteador virtual criado no OpenStack, respectivamente duas máquinas virtuais com o sistema operacional CirrOS. Estas ferramentas foram nomeadas de Nuvem Privada OpenStack. A nuvem privada criada é conectada a um roteador físico (Router DHCP). O primeiro teste de processamento em tempo real e disponibilidade é efetuado entre a máquina virtual CirrOS e o roteador físico, o segundo teste é entre a máquina virtual e o roteador virtual criado no OpenStack. Efetuado verificação de média de tempo de resposta com a utilização dos testes. A simulação de ataque implementada é composta por um servidor (ServerCentOS), instalado o software de virtualização (VMWARE), virtualizado o sistema operacional

Kali Linux, efetuado a simulação de ataque com o Kali Linux do tipo negação de serviço, com intuito de gerar de indisponibilidade e ataque de força bruta tipo dicionário, com intuito para gerar indisponibilidade e quebrar a senha de administrador á Nuvem Privada OpenStack, Efetuado calculo de média de resposta enquanto ocorre e não ocorre as simulações de ataque.

Figura 11. Desenvolvimento da Nuvem Privada e a simulação de ataque.



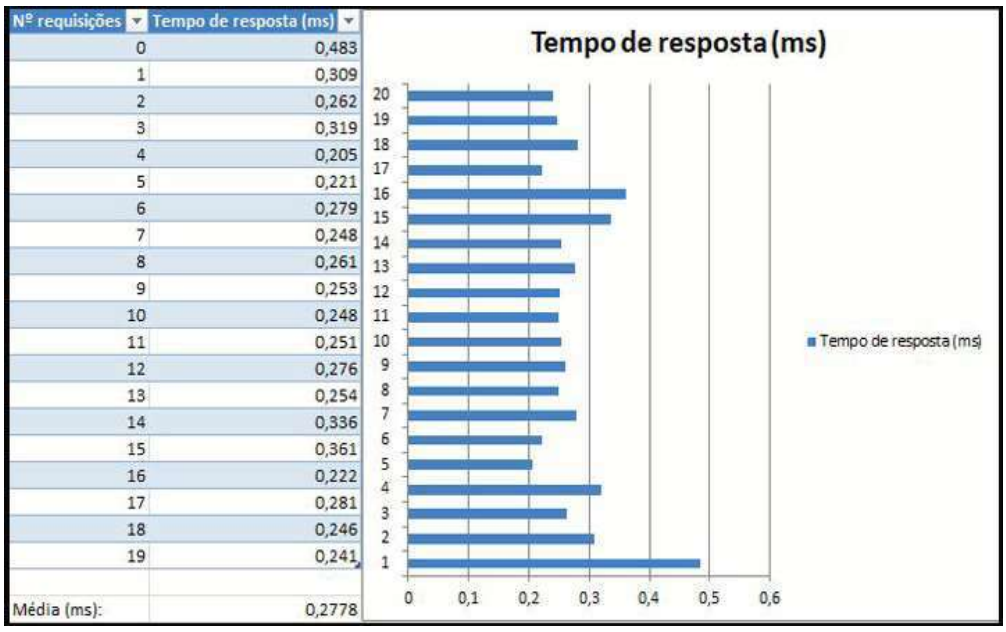
Fonte: Autor (2019).

5 VALIDAÇÃO

O método efetuado para validação da implementação de infraestrutura como serviço em nuvem privada se concretizou em testes de comunicação de rede, verificando sua disponibilidade de serviço, processamento em tempo real, simulação de ataque de negação de serviço e força bruta tipo dicionário.

O teste da instância cirros_1 (CirrOS) foi realizado entre o mesmo citado e o Roteador DHCP (Multilaser RE033), acionado o comando “ping” com 20 pacotes de transmissão, com TTL de 64 segundos no terminal de comando da instancia cirros_1, foi efetuado visando o tempo de resposta entre o sistema operacional e o Roteador DHCP. Os dados coletados chegaram ao resultado de que o maior tempo de resposta encontrado foi de 0,483ms (milissegundos), efetuado o calculo da média de resposta entre os 20 pacotes de transmissão, encontrado o resultado de 0,2778ms como é demonstrado na Figura 12.

Figura 12. Tempo de resposta instância cirros_1.



Nº requisições	Tempo de resposta (ms)
0	0,483
1	0,309
2	0,262
3	0,319
4	0,205
5	0,221
6	0,279
7	0,248
8	0,261
9	0,253
10	0,248
11	0,251
12	0,276
13	0,254
14	0,336
15	0,361
16	0,222
17	0,281
18	0,246
19	0,241
Média (ms):	0,2778

Fonte: Autor (2019).

O teste de comunicação da instância cirros_2 foi efetuado por terminal de comando do próprio sistema operacional para o roteador virtual criado no OpenStack, acionado o comando “ping” com quantidade de 20 pacotes de

transmissão e TTL de 64 segundos, o maior tempo de resposta encontrado foi de 0,573ms, o cálculo da média de resposta encontrado foi de 0,3015ms como é demonstrado na Figura 13.

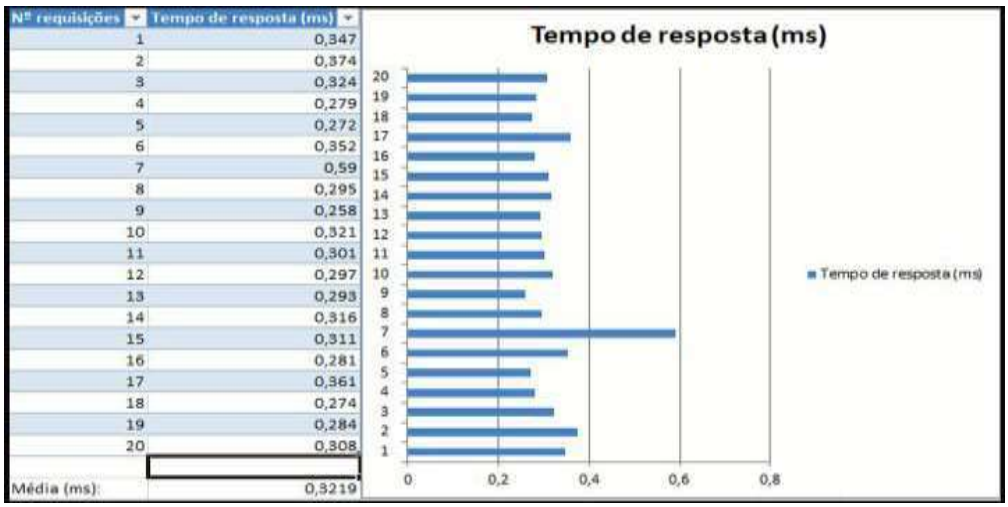
Figura 13. Tempo de resposta instância cirros_2.

Nº requisições	Tempo de resposta (ms)
0	0,573
1	0,22
2	0,276
3	0,276
4	0,299
5	0,223
6	0,273
7	0,271
8	0,329
9	0,244
10	0,222
11	0,224
12	0,361
13	0,271
14	0,303
15	0,359
16	0,343
17	0,299
18	0,321
19	0,343
Média (ms):	0,3015

Fonte: Autor (2019).

Após a instalação e configuração do ambiente de simulação de ataque com o sistema operacional Kali Linux, foi efetuado um teste de comunicação de rede entre o Kali e a nuvem privada OpenStack. O teste foi efetuado no terminal de comando do Kali com o comando “ping” num total de 20 pacotes de transmissão e TTL de 128 segundos, o maior tempo encontrado foi de 0,59ms, o cálculo da média de tempo resposta encontrado foi de 0,3219ms como é demonstrado na Figura 14.

Figura 14. Tempo de resposta Kali Linux para OpenStack.

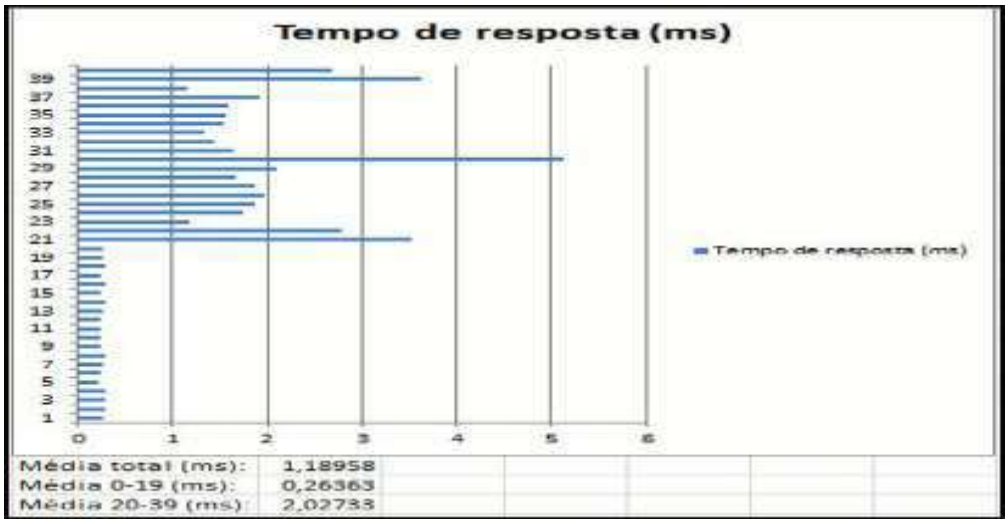


Nº requisições	Tempo de resposta (ms)
1	0,347
2	0,374
3	0,324
4	0,279
5	0,272
6	0,352
7	0,59
8	0,295
9	0,258
10	0,321
11	0,301
12	0,297
13	0,293
14	0,316
15	0,311
16	0,281
17	0,361
18	0,274
19	0,284
20	0,308
Média (ms):	0,3219

Fonte: Autor (2019).

O ataque de negação de serviço foi efetuado no terminal de comando do Kali Linux assim como o teste de comunicação de rede, o teste de comunicação foi efetuado com o comando “ping” do pacote de transmissão 0 ao 39, totalizando 40 amostras, da amostra 0 á 19 onde não é efetuado a simulação de ataque, efetuado o cálculo de média do tempo de resposta e encontrado o resultado de 0,26363ms, da amostra 20 á 39 é efetuado a simulação de ataque de negação de serviço, sua média de tempo foi calculada e foi encontrado o tempo de 2,02733ms, o tempo médio calculado entre as 40 amostras foi de 1,18958ms como é demonstrado na Figura 15.

Figura 15. Tempo de resposta ataque de negação de serviço.



Fonte: Autor (2019).

A simulação de ataque de força bruta tipo dicionário foi efetuado no terminal de comando do Kali Linux, como também o teste de comunicação, o teste de comunicação foi efetuado com o comando “ping” do pacote de transmissão 0 ao 39, totalizando 40 amostras realizadas, da amostra 0 á 19 onde não é efetuado a simulação de ataque, foi efetuado o cálculo de média do tempo de resposta, encontrado o resultado de 0,35311ms, da amostra 20 á 39 é efetuado a simulação de ataque de força bruta, sua média de tempo foi calculada e foi encontrado o tempo de 0,33481ms, o tempo médio calculado entre as 40 amostras foi de 0,3435ms como é demonstrado na Figura 16.

Figura 16. Tempo de resposta ataque de força bruta.

Fonte: Autor (2019).

6 CONCLUSÃO

A implementação da infraestrutura como serviço em nuvem privada foi concluída com êxito, a escolha por softwares livres e gratuitos é a indicada, acarreta em investimento apenas no hardware utilizado, sem dificuldades encontradas na utilização do software de virtualização VMware, sistema operacional CentOS e Kali Linux, a maior dificuldade encontrada foi na documentação do OpenStack disponibilizada no site oficial e seus desenvolvedores, para instalação e configuração do software, a documentação é incompleta ou desatualizada, não é de bom entendimento para aprendizado ou pessoas que tenham pouco conhecimento com sistemas operacionais de arquitetura Linux.

A ferramenta OpenStack com seus serviços demonstra uma possível solução para os paradigmas de Computação em Nuvem, a orquestração de redes virtuais com sistemas operacionais virtuais por software livre são de grande valia, se tem possibilidades para uso doméstico, acadêmico e empresarial de inúmeros portes, estes serviços virtualizados tem uma camada a mais de segurança para um possível ataque. A implementação de infraestrutura como serviço em nuvem privada teve pontos positivos, as instâncias tiveram bons resultados de disponibilidade encontrada no teste de comunicação de rede. Os resultados encontrados em testes de comunicação entre as instâncias e o roteador físico e virtual foram positivos, tempo de resposta dentro do esperado, que num ambiente de utilização, teria acesso normal. A simulação de ataque de negação de serviço teve êxito, pois no momento em que estava em andamento, os serviços tiveram demora nas respostas de teste de comunicação de rede, o que num ambiente usual, acarretaria falta de acesso ao serviço ou demora de resposta. A simulação de ataque de força bruta tipo dicionário não teve êxito, além de o nome de usuário e senha para acesso a nuvem privada OpenStack, não foram roubados, o tempo de resposta não conteve oscilação e os serviços não ficaram indisponíveis. A nuvem privada OpenStack para ter uma segurança ao ataque de negação de serviço, deve se implementar outras camadas de segurança ao redor da nuvem privada, com utilização de softwares e hardwares, assim tendo total confiança e funcionalidade plena dos serviços. Conclui-se que a ferramenta OpenStack apresentou com seus serviços que é possível aderir á nuvem privada, com uma camada de segurança mínima, dependendo do

administrador de rede ter os corretos conhecimentos da sua funcionalidade, o trato dos dados, instâncias e redes.

Para um projeto futuro considerasse o desenvolvimento de uma implementação de balanceamento de carga, sendo possível a disponibilidade ininterrupta do serviço de uma infraestrutura como serviço em nuvem privada.

7 CRONOGRAMA

Atividades	**Meses**					
	mar.	abr.	mai.	jun.	jul.	
Revisão bibliográfica	x	x	x			
Estudo das tecnologias do projeto	x	x	x			
Descrição da solução			x			
Metodologia			x	x		
Resultados obtidos			x	x		
Elaboração dos apêndices			x	x		
Conclusão				x		
Revisão final do texto				x	X	
Data limite de entrega do Projeto						04/07/19

REFERÊNCIAS BIBLIOGRÀFICA

CABRAL, Isabela. Hackers divulgam 2,2 bilhões de senhas de graça na Internet; proteja-se. 2019. Disponível em: https://www.techtudo.com.br/noticias/2019/02/hackers-divulgam-22-bilhoes-de-senhas-de-graca-na-internet-proteja-se.ghtml Acesso em: 15 jun. 2019.

ALKMIM, Gustavo P.; UCHÔA, Joaquim Quinteiro. Uma Solução de Baixo Custo para a Migração de Máquinas Virtuais. 2009. Disponível em: https://www.researchgate.net/publication/228501218_Uma_Solucao_de_Baixo_Custo_para_a_Migracao_de_Maquinas_Virtuais Acesso em: 02 julho 2018.

AMAZON, Amazon Web Services. Disponível em: https://aws.amazon.com/pt/ Acesso em: 15 jun. 2019.

AMAZON. Elastic Compute Cloud (EC2) – Servidor e hospedagem na nuvem – AWS. Disponível em: https//aws.amazon.com/pt/ec2/. Acesso em: 30 maio 2018.

BRANCO JR., Eliseu C.; MACHADO, Javam C.; Monteiro, Jose Maria. Estratégias para Proteção da Privacidade de Dados Armazenados na Nuvem. 2014. Disponível em: http://www.inf.ufpr.br/sbbd-sbsc2014/sbbd/proceedings/artigos/pdfs/14.pdf Acesso em: 15 jun. 2019.

BUYYA, Rajkumar.; BROBERG, James.; GOSCINSKI, Andrzej. Cloud Computing: Principles and Paradigms. 1º Ed. John Wiley and Sons, 2010, p. 17.

COGO, Gabriel Silva. Análise da Intenção de Adoção da Computação em Nuvem por Profissionais da área de TI. Disponível em: https://lume.ufrgs.br/handle/10183/78039 Acesso em: 15 jun. 2019.

LEIMESTER, Stefanie; RIEDL, Christoph; BOHM, Markus; KRCAMR, Helmut. The Business Perpective of Cloud Computing: Actors, Roles, and Valeu Networks. 2010. Disponível em: https://pdfs.semanticscholar.org/e6c5/a574bcb34e247371e47338f5e16286ddb6d9.pdf?_ga=2.254030157.479994144.1554147488-372792476.1550494656 Acesso em: 15 jun. 2019.

SRINIVASAMURTHY, Shilpashree; LIU, David Q. Survey on Cloud Security. 2010. Disponível em: http://salsahpc.org/CloudCom2010/Poster/cloudcom2010_submission_67.pdf Acesso em: 15 jun. 2019.

CASTRO, Rita de C. C.; SOUSA, Verônica L. Pimentel de. Segurança em Cloud Computing: Governança e Gerenciamento de Riscos de Segurança. 2010. Disponível em: https://s3.amazonaws.com/academia.edu.documents/34088078/26-05-S5-1-68740-Seguranca_em_Cloud.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1530559704&Signature=tklB1i%2BwCZF9zD2WuLSC6qnZlJY%3D&response-content-disposition=inline%3B%20filename%3DSeguranca_em_Cloud_Computing_Governanca.pdf Acesso em: 15 jun. 2019.

CHEE, Brian J. S.; FRANKLIN JR., Curtis. Computação em Nuvem. Cloud Computing. Tecnologias e Estratégias. M.Books. São Paulo. 2013. p. 58-62-210.

CLOUD SECURITY ALLIANCE –CSA. The Notorius Nine: Cloud Computing Top Threats in 2013. Disponível em: https://cloudsecurityalliance.org/download/the-notorious-nine-cloud-computing-top-threats-in-2013/ Acesso em: 15 jun. 2019.

CLULEY, Graham. Evernote Hacked – “Almost 50 million Passwords Reset After Security Breach”. 2013. Disponível em: https://nakedsecurity.sophos.com/2013/03/02/evernote-hacked-almost-50-million-passwords-reset-after-security-breach. Acesso em: 15 jun. 2019.

CONVERGÊNCIA DIGITAL, Nuvem Cibernética Vira Território Promissor Para Hackers. Disponível em: http://convergenciadigital.uol.com.br/cgi/cgilua.exe/sys/start.htm?UserActiveTemplate=site&infoid=44205&sid=97 Acesso em: 15 jun. 2019.

COULOURIS, George; DOLLIMORE, Jean; KINDBERG, Tim; BLAIR, Gordon. Distributed Systems: Concepts and Design. Fifth Edition. 2005. Disponível em: http://www.gecg.in/papers/ds5thedn.pdf Acesso em: 15 jun. 2019.

CSA. Cloud Security Alliance. Cloud Security Alliance Releases Updates to ‘The Treacherous 12: Cloud Computing Top Threats in 2016’. 2017. Disponível em: https://downloads.cloudsecurityalliance.org/assets/research/top-threats/treacherous-12-top-threats.pdf/ Acesso em: 15 jun. 2019.

CSA. Cloud Security Alliance. Sedurity Guidance for Critical Areas of Focus un Cloud Computing V2,1. 2009. Disponível em: https://cloudsecurityalliance.org/csaguide.pdf Acesso em: 15 jun. 2019.

DYN, DDoS Attack Against Dyn Managed DNS. 2016. Disponível em: https://www.dynstatus.com/incidents/nlr4yrr162t8 Acesso em: 15 jun. 2019.

GONZALEZ, Nelson Mimura; MIERS, Charles Christian; REDÌGOLO, Fernando Frota; ROJAS, Marco Antônio Torrez; CARVALHO, Tereza Cristina Melo de Brito. Segurança nas nuvens computacionais: Uma Visão dos principais problemas e soluções. 2013. Disponível em: https://www.revistas.usp.br/revusp/article/viewFile/61683/64572 Acesso em: 15 jun. 2019.

GOOGLE, Google Cloud. Disponível em: https://cloud.google.com/?hl=pt-br Acesso em: 15 jun. 2019.

GOOGLE. G Suite - Gmail, Drive, Documentos e muito mais. Disponível em: https://gsuite.google.com.br/intl/pt-BR/. Acesso em: 15 jun. 2019.

HASHEM, Ibrahim Abaker Targio; YAQOOB, Ibrar; ANUAR, Nor Badrul; MOKHTAR, Salimah; GANI, Abdullah; KHAN, Samee Ullah. (2014). The rise of “Big Data” on cloud computing: Review and open research issues. Information Systems. 47. p. 98-115.

LAUREANO, Marcos. Máquinas Virtuais e Emuladores – Conceitos, Técnicas e Aplicações. 2006. Disponível em: http://www.mlaureano.org/aulas_material/so/livro_vm_laureano.pdf Acesso em: 15 jun. 2019.

LIMA, Fagner Silva de. Adoção da TI Verde na Administração das Redes de Computadores. 2013. Disponível em: https://pt.slideshare.net/fagnerlima91/fagner-lima-adoo-da-ti-verde-na-administrao-das-redes-de-computadores Acesso em: 15 jun. 2019.

LOPES, Giuseppe Alves; JUNIOR, Jair de Mello. EDUCLOUD 2: Implementando mecanismos de elasticidade em uma nuvem privada para ambientes acadêmicos. 2012. Disponível em: http://revistaseletronicas.pucrs.br/ojs/index.php/graduacao/article/view/12420/8338 Acesso em: 15 jun. 2019.

LOPES, Ney da Silva; CARRERO, Marcos Aurélio. BOAS PRÁTICAS DA TI VERDES ADOTADAS PELAS EMPRESAS COMO FORMA DE USO EFICIENTE DOS RECURSOS ENERGÉTICOS. 2016. Disponível em: https://cadernopaic.fae.edu/cadernopaic/article/viewFile/200/160 Acesso em: 15 jun. 2019.

MACHADO, Claiton Prado; LOUREIRO,César A. H.. Comparação de ferramentas de software livre para administração de nuvem privada. 2011. Disponível em: http://www.ulbra.inf.br/joomla/images/documentos/TCCs/2011_02/PROJETO_RC_CLAITON_PRADO_MACHADO.pdf Acesso em: 15 jun. 2019.

MEDEIROS, Monnalisa Christina Pereira de. Análise da Implantação de Computação em Nuvem: Estudo de Caso na Alfa Informática.Net – Currais Novos. 2015. Disponível em: https://monografias.ufrn.br/jspui/bitstream/123456789/2110/3/An%C3%A1lise%20da%20implanta%C3%A7%C3%A3o_Monografia_Medeiros.pdf Acesso em: 15 jun. 2019.

MELL, Peter; Grance, Timothy. The NIST Definition of Cloud Computing. Gaithersburg: National Institute of Standards and Technology, 2011,p.7. Disponível em: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-145.pdf Acesso em: 15 jun. 2019.

MELL, Peter; GRANCE, Timothy. The NIST Definition of Cloud Computing. 2011. Disponível em: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-145.pdf Acesso em: 15 jun. 2019.

MENEGATT, Josimar. Segurança e Privacidade na Cloud Computing. 2012. Disponível em: https://www.ppgia.pucpr.br/~jamhour/RSS/TCCRSS11/Josimar%20Menegatt%20_%20Artgo%20_%20Josimar%20Menegatt%20_%20Seguran%FEa%20e%20Privacidade%20na%20computa%FE%D2o%20em%20nuvem.pdf Acesso em: 15 jun. 2019.

CORRÊA, André Luiz Riccó; MARTINS, Henrique Pachioni. Monitoramento de Ataques de Negação de Serviço: Um caso Prático Utilizando Slowloris. 2013. Disponível em: http://fatecbauru.edu.br/mtg/source/Monitoramento%20de%20ataques%20de%20nega%C3%A7%C3%A3o%20em%20servi%C3%A7o.pdf Acesso em: 15 jun. 2019.

MELO, Ricardo Rebelo Silva. Segurança em Nuvem em uma Rede Corporativa. 2018. Disponível em: http://ebrevistas.eb.mil.br/index.php/OC/article/view/1686/1395 Acesso em: 15 jun. 2019.

MICROSOFT. Plataforma e serviços de computação em nuvem do Microsoft Azure. 2018. Disponível em: https://azure.microsoft.com/pt-br/. Acesso em: 15 jun. 2019.

PENSO. Diferenças Entre Os Tipos De Nuvem: Nuvem Pública. Disponível em: https://www.penso.com.br/diferencas-entre-os-tipos-de-nuvem-nuvem-publica/ Acesso em: 15 jun. 2019.

PEREIRA, A.L.; Appel, A.P.: Modeling and storing complex network with graph-tree. In: New Trends in Databases and Information Systems, Workshop Proceedings of the 16th East European Conference, ADBIS 2012, Pozna, Poland, September 17-21, pp.305–315, (2013).

SANTOS, Augusto Carvalho dos; PEDROSA, BrunoGil; SANTOS, Henrique; GODINHO, Higor Gonçalves; SILVA, Leandro; SCHETTINO, Rafael do Carmo; SANTOS, Rodrigo Carvalho dos; MENDES, Sérgio Henrique Amaral. Computação em Nuvem: Conceitos e Perspectivas. 2010. Disponível em: http://virtual.ietec.com.br/file.php/1/Biblioteca/Modelos_de_Trabalhos_T_cnicos_j_realizados/Modelos_de_Trabalhos_IFTI/Computacao_em_nuvem.pdf Acesso em: 15 jun. 2019.

SANTOS, Igor Lucas Coelho; LIMA, Iremar Nunes de. Virtualização em Servidores de Bando de Dados. 2013. Disponível em: http://blog.newtonpaiva.br/pos/wp-content/uploads/2013/02/E2-SI-30.pdf Acesso em: 15 jun. 2019.

SILVA, Rodrigo Ferreira da. Virtualização de Sistemas Operacionais. 2007. Disponível em: http://lncc.br/~borges/doc/Virtualizacao%20de%20Sistemas%20Operacionais.TCC.pdf Acesso em: 15 jun. 2019.

SILVEIRA, Rafael Turnes. OpenStack com Open vSwitch. 2014. Disponível em: https://wiki.sj.ifsc.edu.br/wiki/images/9/93/TCC_RafaelTurnesSilveira.pdf Acesso em: 15 jun. 2019.

SOTO, Julio Alba. OpenNebula: implantação de uma nuvem privada e orquestração das máquinas virtuais no paradigma da Computação em Nuvem. 2011. Disponível em: http://www.cgeti.ufc.br/monografias/JULIO_ALBA_SOTO.pdf Acesso em: 15 jun. 2019.

SUPORTI, Nuvem Pública: Você sabe Do Que Estamos Falando? Disponível em: http://suporti.com/nuvem-publica/ Acesso em: 15 jun. 2019.

SYMANTEC. Pesquisa sobre a Situação de Cloud Computing. Resultados América Latina. 2011. Disponível em: http://www.symantec.com/content/pt/br/enterprise/images/theme/state-of-cloud/State-of-Cloud-Report-LAM-PORT-FN.pdf Acesso em: 15 jun. 2019.

CSA. Cloud Security Alliance. Security Guidance for Criticals Areas of Focus in Cloud Computing V2.1. 2009. Disponível em: https://cloudsecurityalliance.org/wp-uploads/2011/07/csaguide.v2.1.pdf Acesso em: 15 jun. 2019.

TAURION, Cezar. Cloud computing: computação em nuvem: transformando o mundo da tecnologia da informação. Rio de Janeiro: Brasport, 2009. p.205.

TERUEL, Domingos. Segurança em Cloud Computing. Desafios e Gerenciamento de Riscos. 2014. Disponível em: https://culturacolaborativa.socialbase.com.br/wp-content/uploads/2015/01/ebook_seguranca_cloud_computing.pdf Acesso em: 15 jun. 2019.

TIINSIDE, Aumento Da Confiança Na Nuvem Traz Mais Preocupações Com A Segurança. Disponível em: http://tiinside.com.br/tiinside/seguranca/artigos-seguranca/19/03/2017/aumento-da-confianca-na-nuvem-traz-mais-preocupacoes-com-seguranca/ Acesso em: 15 jun. 2019.

VELTE, Antonhy T. ; VELTE, Toby J. ; ELSENPETER, Robert. Cloud Computing: Computação em Nuvem - Uma Abordagem Prática. Elsevier/alta Books, 2012.

VERAS, Manoel. Arquitetura de Nuvem – Amazon Web Services (AWS). Rio de Janeiro: Brasport 2013. p. 9 - 91.

VERAS, Manoel. Datacenter: Componente Central da Infraestrutura de TI. Rio de Janeiro: Brasport 2010. p. 8 – 100.

VERDERAMI, Beatriz Monteiro; ROSA, Rodrigo. AVALIANDO O USO DA COMPUTAÇÃO EM NUVEM NA TI PARA PEQUENAS E MÉDIAS EMPRESAS BRASILEIRAS. v. 2, n. 1, 2013 Disponível em:http://revistas.ung.br/index.php/computacaoaplicada/article/view/1404/1190 Acesso em: 15 jun. 2019.

WESTPHALL, Carlos; VILLARREAL, Sergio. PRINCIPÍOS E TENDÊNCIAS EM GREEN CLOUD COMPUTING. 2013. Disponível em: http://www.periodicosibepes.org.br/index.php/reinfo/article/view/1050/pdf Acesso em: 15 jun. 2019.

FSF. Free Software Foundation. 2019. Disponível em: https://www.fsf.org/ Acesso em: 15 jun. 2019.

GNU. O Sistema Operacional GNU. 2019. Disponível em: https://www.gnu.org/philosophy/philosophy.html Acesso em: 15 jun. 2019.

GNU. Categorias de softwares livres e não livres. 2019. Disponível em: https://www.gnu.org/philosophy/categories.pt-br.html Acesso em: 15 jun. 2019.

SABINO, Vanessa; KON, Fabio. Licenças de Software Livre História e Características. 2009. Disponível em: http://www.ccsl.org.br/files/relatorio-licencas.pdf Acesso em: 15 jun. 2019.

STALLMAN, Richard. Por que o Software Deveria Ser Livre. 2018. Disponível em: http://www.gnu.org/philosophy/shouldbefree.html Acesso em: 15 jun. 2019.

ROSADO, Tiago André Pais. Implementação de uma infraestrutura de Cloud Privada Baseada em Openstack. 2016. Disponível em: https://comum.rcaap.pt/bitstream/10400.26/18871/1/Tiago-Andre-Pais-Rosado.pdf Acesso em: 15 jun. 2019.

OSI. Open Source Initiative. 2019. Disponível em: https://opensource.org/history Acesso em: 15 jun. 2019.

OSI. The Open Source Definition (Annotated). 2019. Disponível em: https://opensource.org/osd-annotated Acesso em: 15 jun. 2019.

GONZAGA, Emerson Corbellini. A Melhor Solução de Virtualização no Mercado Segundo os Profissionais de TI da Região Sul do Brasil. 2018. Disponível em:https://riuni.unisul.br/bitstream/handle/12345/5722/EMERSON_CORBELLINI_GONZAGA-%5b51114-11301-3-750846%5dFINAL_AD4_artigo_sem_logo_apos_defesa_corrigido.pdf?sequence=1&isAllowed=y Acesso em: 15 jun. 2019.

FERREIRA, Leonardo da Silva. Eficiência Energética na Consolidação de Servidores: Uma Comparação Entre a Virtualização e a Conteinerização. 2018. Disponível em: https://monografias.ufrn.br/jspui/bitstream/123456789/8336/1/TCC%20-%20Leonardo%20da%20Silva%20Ferreira.pdf Acesso em: 15 jun. 2019.

VMWARE. Workstation Player. 2019. Disponível em: https://www.vmware.com/br/products/workstation-player.html Acesso em: 15 jun. 2019.

WMWARE. Avaliar o VMware Workstation Player. 2019. Disponível em: https://www.vmware.com/br/products/workstation-player/workstation-player-evaluation.html Acesso em: 15 jun. 2019.

CENTOS. CentOS Linux. 2019. Disponível em: https://www.centos.org/about/ Acesso em: 15 jun. 2019.

CENTOS. O que é o CentOS Linux?. 2019. Disponível em: https://wiki.centos.org/ Acesso em: 15 jun. 2019.

CENTOS. Download CentOS. 2019. Disponível em: https://www.centos.org/download/ Acesso em: 15 jun. 2019.

SOFTWARE LIVRE. Software Livre Brasil. 2009. Disponível em: http://softwarelivre.org/centos Acesso em: 15 jun. 2019.

BARBOSA, Roseli da R; REGO, Paulo A. L.; BONFIM, Michel S.; CALLADO, Arthur de C.. Análise de Desempenho das Tecnologias de Virtualização de Rede da Plataforma OpenStack. 2018. Disponível em: http://portaldeconteudo.sbc.org.br/index.php/wperformance/article/view/3334 Acesso em: 15 jun. 2019.

OPENSTACK. Welcome to OpenStack Documentation. 2019. Disponível em: https://docs.openstack.org/rocky/?_ga=2.182280096.609572395.1552998822-377616459.1552998822 Acesso em: 15 jun. 2019.

OPENSTACK. OpenStack Releases. 2019. Disponível em: https://releases.openstack.org/ Acesso em: 15 jun. 2019.

OPENSTACK. OpenStack Compute (Nova). 2019. Disponível em: https://docs.openstack.org/nova/latest/ Acesso em: 15 jun. 2019.

OPENSTACK. OpenStack identity (”Keystone”). 2019. Disponível em: https://wiki.openstack.org/wiki/Keystone Acesso em: 15 jun. 2019.

OPENSTACK. OpenStack API Documentation. 2019. Disponível em: https://developer.openstack.org/api-ref/identity/v3/index.html#what-s-new-in-version-3-11-rocky Acesso em: 15 jun. 2019.

OPENSTACK. Keystone, the OpenStack Identity Service. 2019. Disponível em: https://docs.openstack.org/keystone/latest/ Acesso em: 15 jun. 2019.

OPENSTACK. Welcome to Glance’s documentation. 2018. Disponível em: https://docs.openstack.org/glance/latest/ Acesso em: 15 jun. 2019.

OPENSTACK. Swift. 2019. Disponível em: https://wiki.openstack.org/wiki/Swift Acesso em: 15 jun. 2019.

OPENSTACK. Introduction to Object Storage. 2019. Disponível em: https://docs.openstack.org/swift/pike/admin/objectstorage-intro.html Acesso em: 15 jun. 2019.

OPENSTACK. Cinder. 2019. Disponível em: https://wiki.openstack.org/wiki/Cinder Acesso em: 15 jun. 2019.

OPENSTACK. Horizon: The OpenStack Dashboard Project. 2017. Disponível em: https://docs.openstack.org/horizon/latest/ Acesso em: 15 jun. 2019.

OPENSTACK. Welcome to Ceilometer's documentation. 2017. Disponível em: https://docs.openstack.org/ceilometer/latest/ Acesso em: 15 jun. 2019.

OPENSTACK. Telemetry. 2019. Disponível em: https://wiki.openstack.org/wiki/Telemetry Acesso em: 15 jun. 2019.

OPENSTACK. Welcome to the Heat Documentation. 2018. Disponível em: https://docs.openstack.org/heat/latest/ Acesso em: 15 jun. 2019.

OPENSTACK. Heat. 2019. Disponível em: https://wiki.openstack.org/wiki/Heat Acesso em: 15 jun. 2019.

OPENSTACK. Get Images. Disponível em: https://docs.openstack.org/image-guide/obtain-images.html Acesso em: 15 jun. 2019.

OPENSTACK. Neutron. 2019. Disponível em: https://wiki.openstack.org/wiki/Neutron Acesso em: 15 jun. 2019.

OPENSTACK. Welcome to Neutron's documentation. 2019. Disponível em: https://docs.openstack.org/neutron/pike/ Acesso em: 15 jun. 2019.

OPENSTACK. Welcome to Ironic's documentation! 2019. Disponível em: https://docs.openstack.org/ironic/pike/ Acesso em: 15 jun. 2019.

OPENSTACK. Welcome to Trove's documentation! Disponível em: https://docs.openstack.org/trove/latest/ Acesso em: 15 jun. 2019.

OPENSTACK. Welcome to Sahara! 2018. Disponível em: https://docs.openstack.org/sahara/latest/ Acesso em: 15 jun. 2019.

ROUSE, Margaret. OpenStack Block Storage (cinder). 2017. Disponível em: https://searchstorage.techtarget.com/definition/Cinder-OpenStack-Block-Storage Acesso em: 15 jun. 2019.

ROUSE, Margaret. OpenStack Horizon. 2017. Disponível em: https://searchcloudcomputing.techtarget.com/definition/OpenStack-Horizon Acesso em: 15 jun. 2019.

REDHAT. Introdução ao OpenStack. 2019. Disponível em: https://www.redhat.com/pt-br/topics/openstack Acesso em: 15 jun. 2019.

REDHAT. OPENSTACK TELEMETRY (CEILOMETER). 2019. Disponível em: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux_OpenStack_Platform/6/html/Component_Overview/section-telemetry.html Acesso em: 15 jun. 2019.

REDHAT. OPENSTACK ORCHESTRATION (HEAT). 2019. Disponível em: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux_OpenStack_Platform/6/html/Component_Overview/section-orchestration.html Acesso em: 15 jun. 2019.

REDHAT. OPENSTACK NETWORKING (NEUTRON). 2019. Disponível em: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux_OpenStack_Platform/6/html/Component_Overview/section-networking.html Acesso em: 15 jun. 2019.

ICLOUD. Visão Geral Sobre OpenStack Private Cloud. 2019. Disponível em: https://www.icloud.com.br/945/visao-geral-sobre-openstack-private-cloud Acesso em: 15 jun. 2019.

MIRANDA, Lucas Carvalho. Utilização do OpenStack em Conjunto com as Tecnologias SDN – Software-Defined Networking e NFV – Network Functions Virtualization. 2017. Disponível em: https://repositorio.ufu.br/handle/123456789/21753 Acesso em: 15 jun. 2019.

SOARES, Jackson. Implementação de um Serviço de Telemetria do OpenStack em um Ambiente Cloud Computing. 2015. Disponível em: https://monografias.ufrn.br/jspui/bitstream/123456789/1408/1/Implementa%C3%A7%C3%A3o-de-um-servi%C3%A7o-de-Telemetria-do-OpenStack-em-um-ambiente-Cloud-Computing-Jackson-Soares.pdf Acesso em: 15 jun. 2019.

SILVEIRA, Rafael Turnes. OpenStack com Open vSwitch. 2014. Disponível em: https://wiki.sj.ifsc.edu.br/wiki/images/9/93/TCC_RafaelTurnesSilveira.pdf Acesso em: 15 jun. 2019.

PYTHON. Keystone 15.0.0. 2019. Disponível em: https://pypi.org/project/keystone/ Acesso em: 15 jun. 2019.

KALI. What is Kali Linux? 2019. Disponível em: https://docs.kali.org/introduction/what-is-kali-linux Acesso em: 15 jun. 2019.

GOMES, Lucas de Carvalho; ARAUJO, Marcos Seefelder de Assis; CAMPOS, Vinicius Silva. Negação de Serviço e Botnets. 2015. Disponível em: https://www.gta.ufrj.br/grad/15_1/dos/index.html Acesso em: 15 jun. 2019.

DEFENSORWEB. Ataques de força bruta (bruteforce), como se proteger. 2019. Disponível em: https://www.defensorweb.com/2019/05/04/ataques-de-forca-bruta-bruteforce-como-se-proteger/ Acesso em: 15 jun. 2019.

CERT. Cartilha de Segurança para Internet. 2017. Disponível em: https://cartilha.cert.br/ataques/ Acesso em: 15 jun. 2019.

VMWARE. VMware + OpenStack. 2014. Disponível em: http://vmwarebrasil.blogspot.com/2014/11/vmware-openstack.html Acesso em: 15 jun. 2019.

APÊNDICE A – IMPLEMENTAÇÃO DE NUVEM PRIVADA E SIMULAÇÃO DE ATAQUE

O apêndice tem por finalidade demonstrar as instalações e configurações necessárias para a implementação de nuvem privada OpenStack e a simulação de ataque com o Kali Linux.

Na Figura 17 é demonstrado a primeira página de configuração de instalação da ferramenta de virtualização VMware Workstation Player 15.

Figura 17. Página inicial de instalação VMware.

Fonte: Autor (2019).

A Figura 18 demonstra o contrato de licença para a utilização do VMware.

Figura 18. Contrato de licença VMware.

Fonte: Autor (2019).

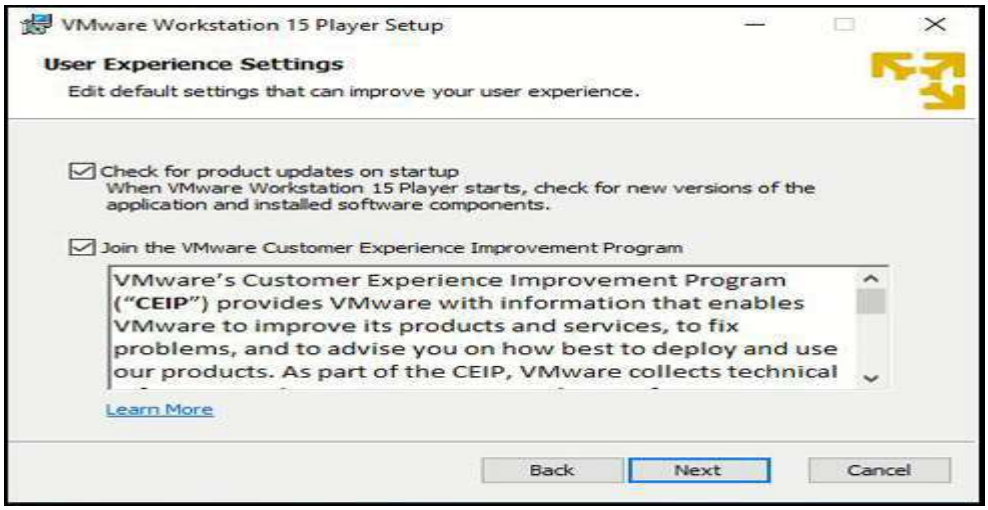
Na figura 19 é demonstrado no botão “change” a escolha do diretório para a instalação do VMware. A opção “Enhanced Keyboard Driver” é uma opção para driver de modelos de teclados que não são considerados padrão.

Figura 19. Diretório de instalação VMware.

Fonte: Autor (2019).

Na Figura 20 é demonstrado duas opções de experiência ao usuário, a opção “Check for product updates on startup” irá verificar na inicialização do VMware se contém novas versões da ferramenta para instalar. Na opção “Join the VMware Customer Improvement Program” possibilita que seja enviado informações de erros na ferramenta VMware.

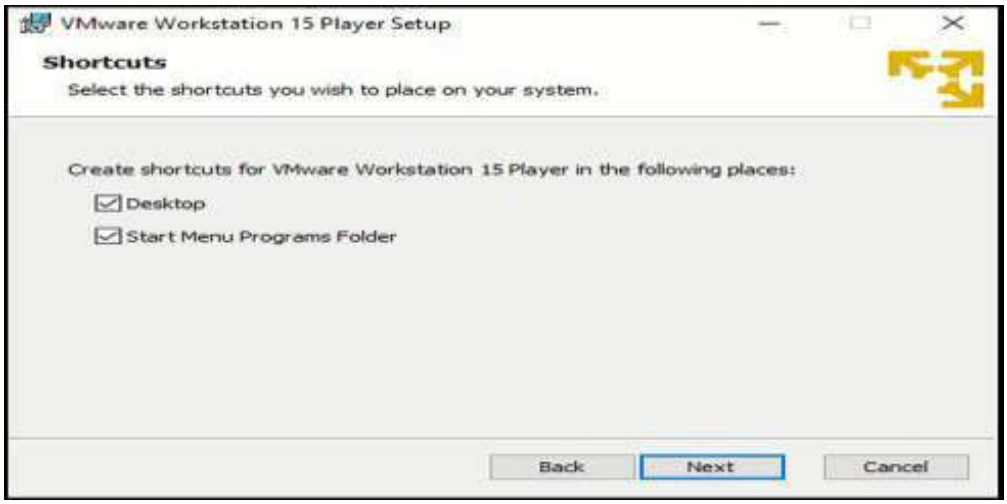
Figura 20. Configuração de experiência de uso VMware.



Fonte: Autor (2019).

Na Figura 21 demonstra as opções de criar ícone de atalho da ferramenta VMware na Área de trabalho e no menu principal do sistema operacional.

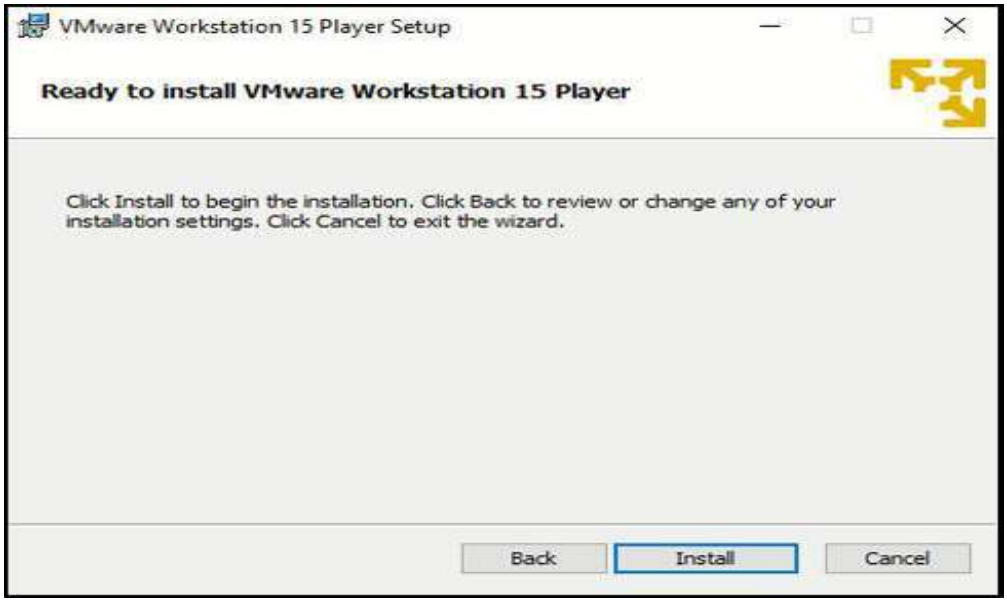
Figura 21. Configuração de atalho VMware.



Fonte: Autor (2019).

Na Figura 22 é demonstrada a possibilidade de retornar “Back” caso o administrador queira modificar alguma configuração de instalação, o botão “Install” inicia a instalação da ferramenta e o botão “Cancel” cancela a instalação e descartando todas as configurações feitas anteriormente.

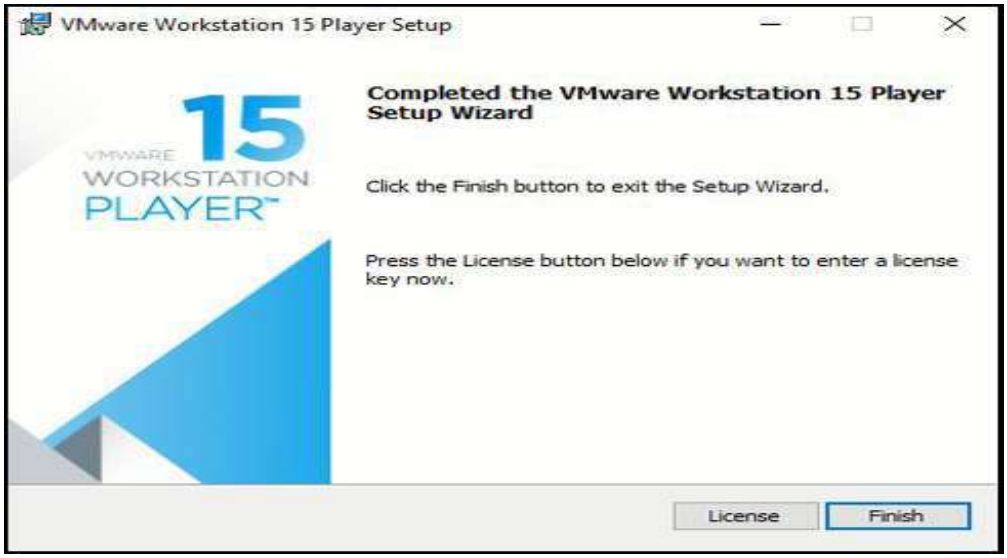
Figura 22. Inicio de instalação VMware.



Fonte: Autor (2019).

Na Figura 23 demonstra que a instalação foi efetuada com sucesso.

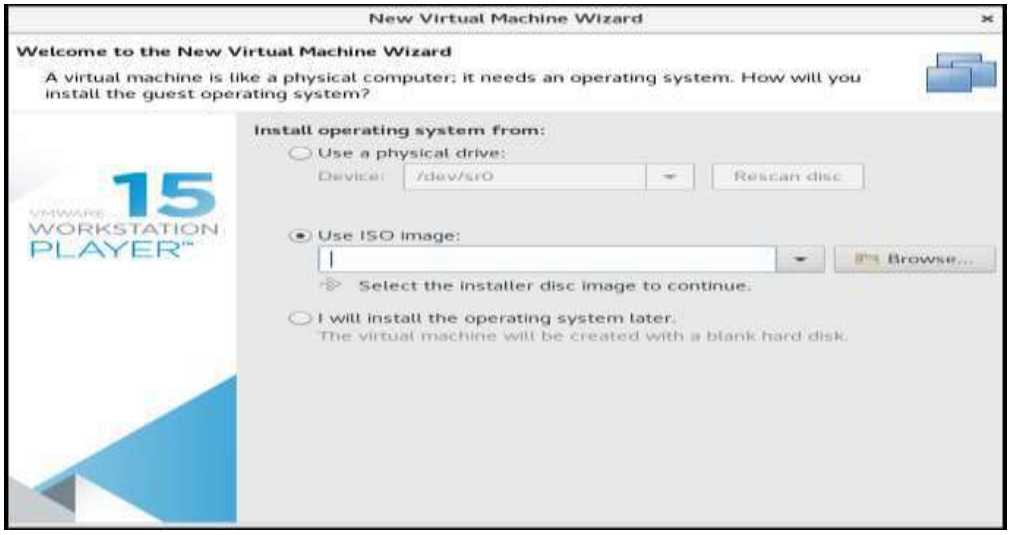
Figura 23. Instalação VMware concluída.



Fonte: Autor (2019).

A página principal da ferramenta VMmare demonstra a opção de criar uma nova máquina virtual (Nem Virtual Machine), a página a seguir na Figura 24 demonstra que a utilização de instalação do sistema operacional virtualizado será de uma imagem no formato ISO, o botão “Browse” disponibiliza a escolha no diretório a qual a imagem ISO se encontra armazenada.

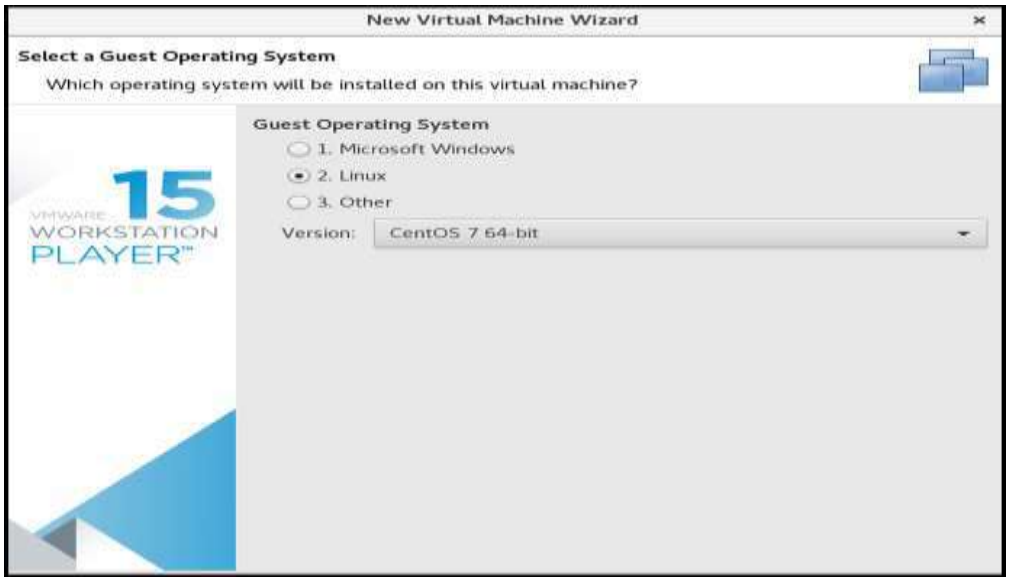
Figura 24. Configuração de diretório de instalação VM.



Fonte: Autor (2019).

Na Figura 25 foi escolhido a opção “Linux” de sistema operacional e sua versão “CentOS 7 64-bit”.

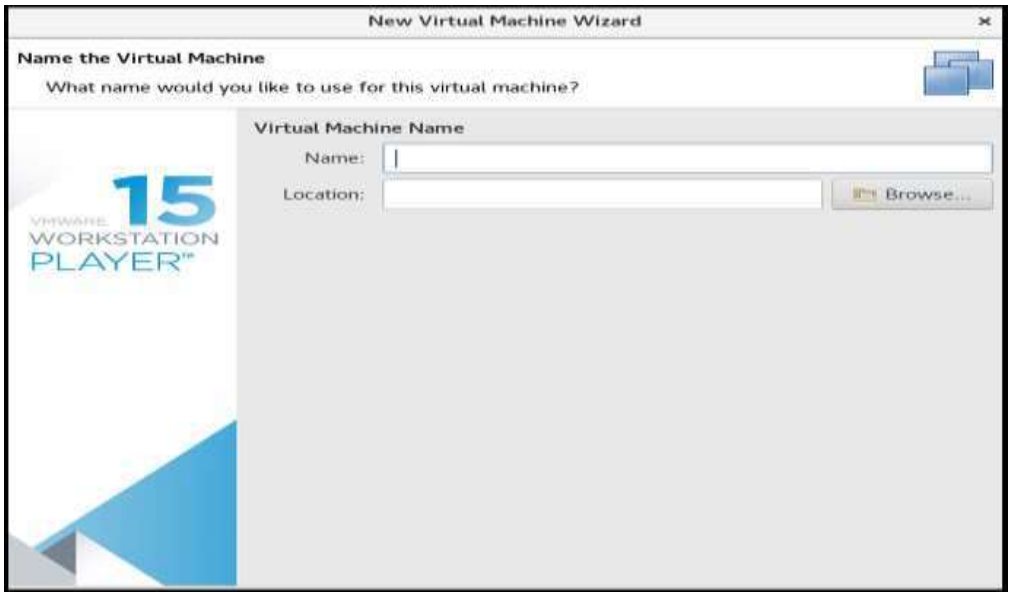
Figura 25. Configuração de sistema operacional VM.



Fonte: Autor (2019).

Na Figura 26 tem a opção de criar um nome (Name) para a máquina virtual e a localização (Browse) no diretório em que a mesma será armazenada.

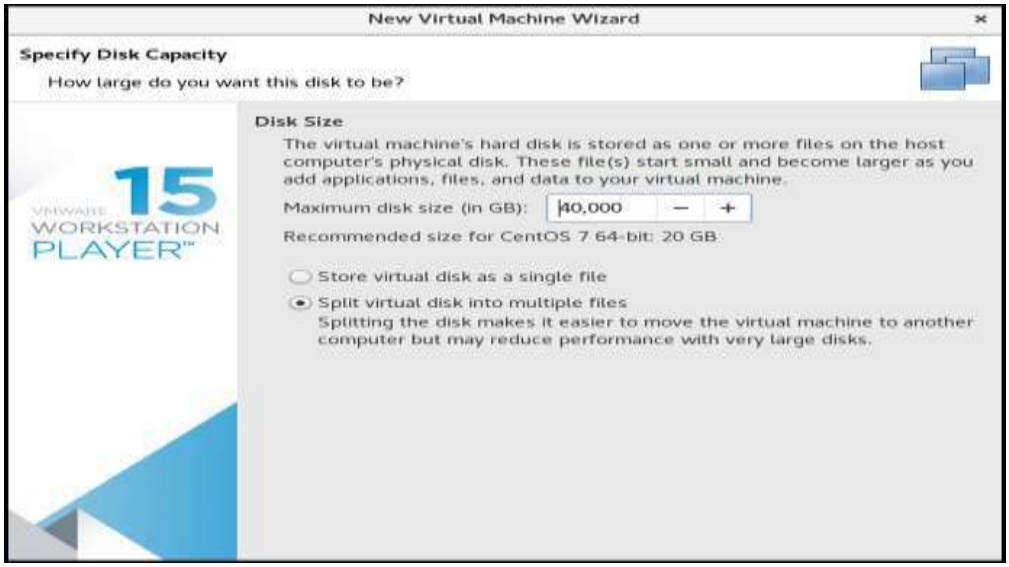
Figura 26. Configuração de nome VM.



Fonte: Autor (2019).

Na Figura 27 é configura a quantidade de armazenamento da máquina virtual para 40 Gibabits, efetuado também a configuração de dividir o disco virtual em múltiplos arquivos (Split virtual disk into multiple files).

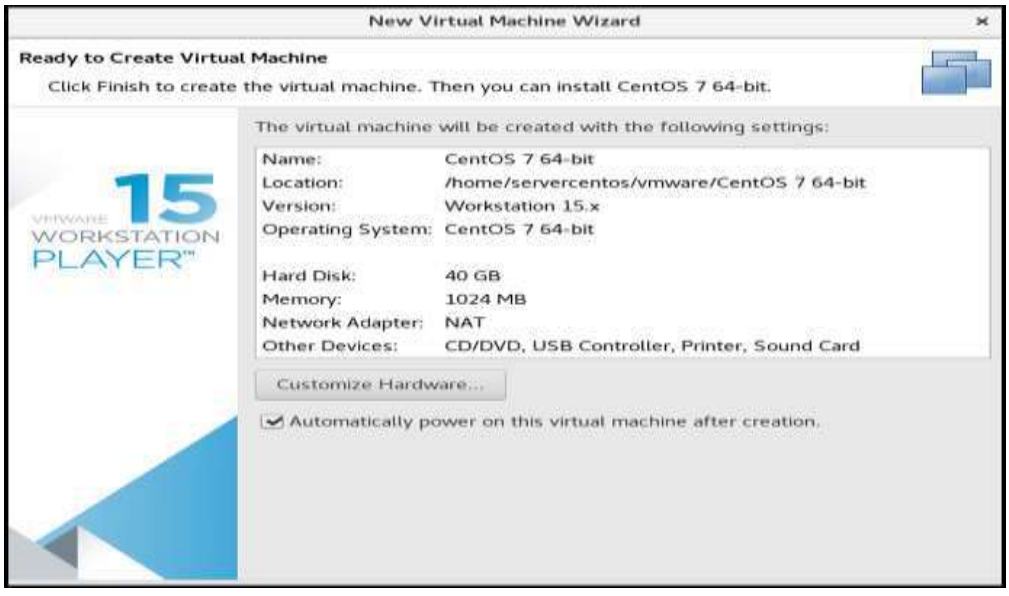
Figura 27. Configuração capacidade de disco VM.



Fonte: Autor (2019).

Na Figura 28 se encontra uma página de configurações mínimas de criação da máquina virtual, no botão “Customize Hardware” será adicionado mais um núcleo de processador e mais 3072MB de memória, num total de 4096MB.

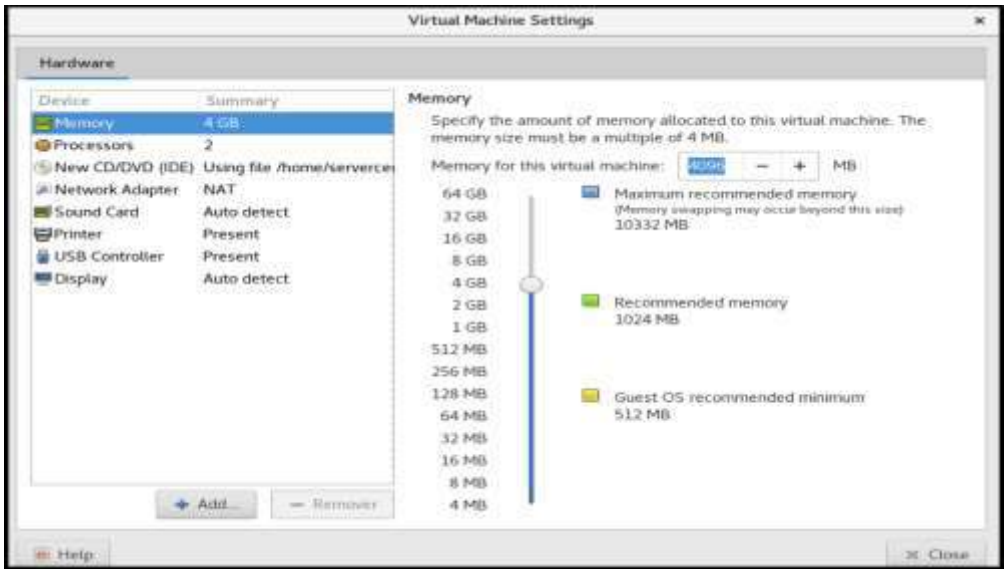
Figura 28. Página de configuração mínima VM.



Fonte: Autor (2019).

Na Figura 29 é demonstrado a adição de memória RAM e processamento.

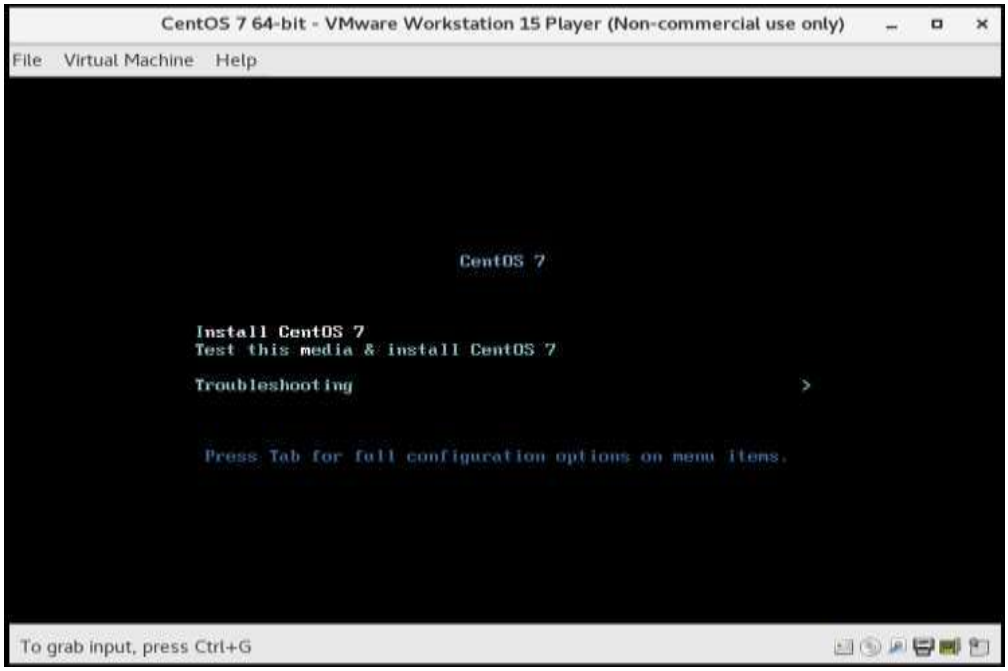
Figura 29. Página de customização de hardware VM.



Fonte: Autor (2019).

Após a criação da máquina virtual e sua inicialização é demonstrado na Figura 30 a página inicial de instalação do sistema operacional CentOS.

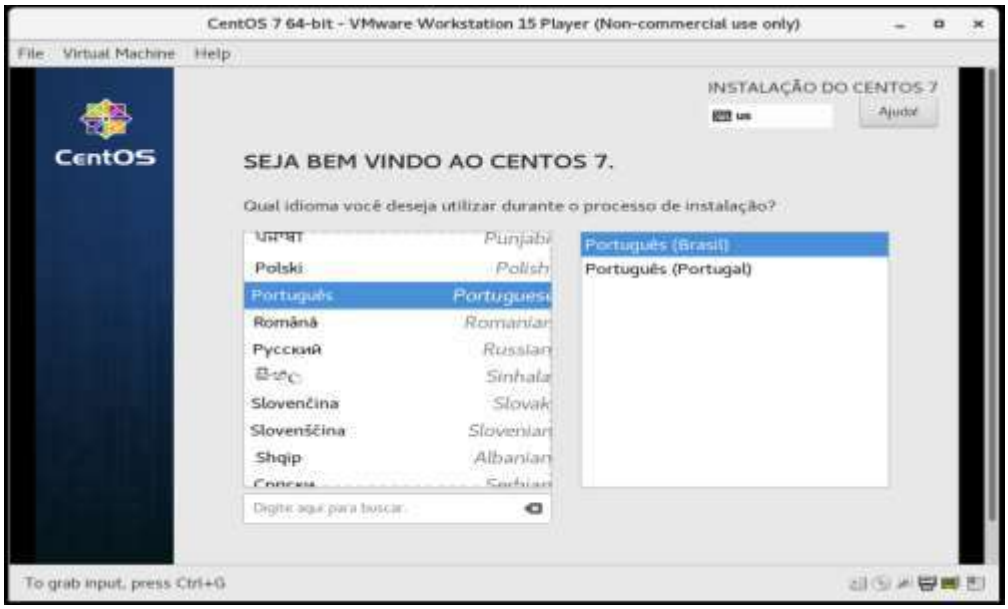
Figura 30. Página inicial de instalação CentOS.



Fonte: Autor (2019).

Na Figura 31 é configurado o idioma do sistema operacional para língua Português do Brasil.

Figura 31. Configuração de idioma CentOS.



Fonte: Autor (2019).

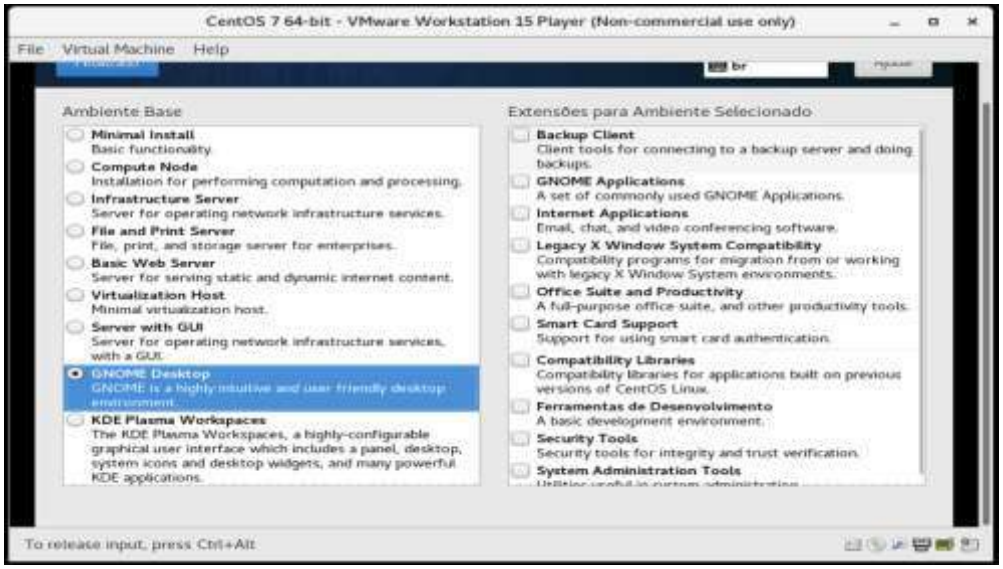
Na Figura 32 é demonstrado a página de configuração de instalação do CentOS.

Figura 32. Página de configuração de instalação CentOS.

Fonte: Autor (2019).

Selecionado opção de “Seleção de Software” da Figura 32 anterior, foi configurado na Figura 33 a visualização “Gnome DeskTop” interface gráfica no sistema operacional.

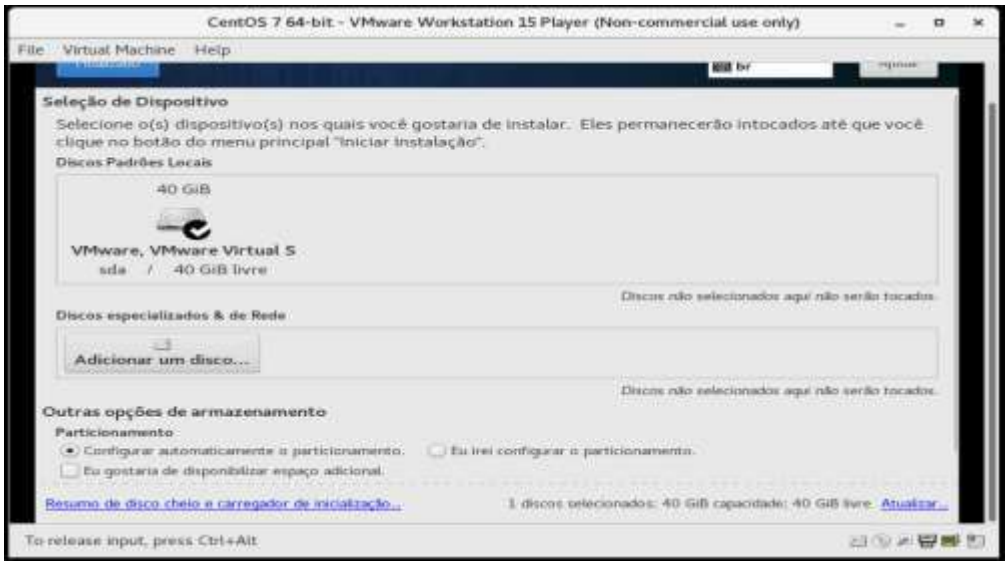
Figura 33. Seleção de Software CentOS.



Fonte: Autor (2019).

Acionado opção “Fonte de instalação” da Figura 33 anterior, e posteriormente na Figura 34 foi escolhido o disco virtual criado no VMware para a instalação do CentOS.

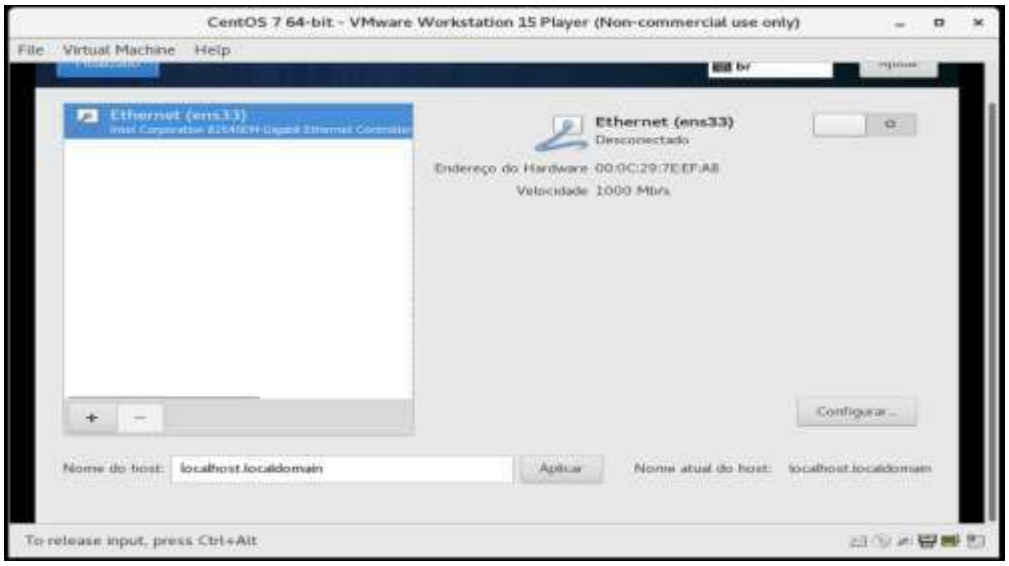
Figura 34. Configuração disco virtual para instalação do CentOS.



Fonte: Autor (2019).

Acionado opção “Rede & Nome do Host” da Figura 34, posteriormente na Figura 35 foi acionada a placa de rede e nome do host.

Figura 35. Configuração placa de rede/nome de host CentOS.



Fonte: Autor (2019).

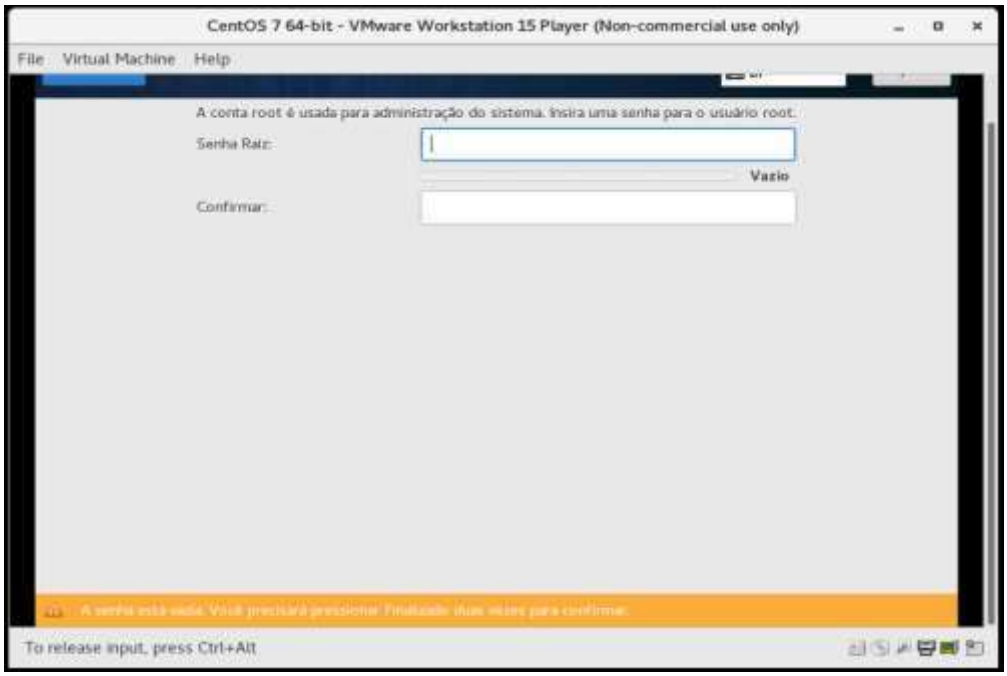
Na Figura 36 é demonstrado a inicialização da instalação do CentOS. Enquanto a instalação ocorreu foi configurado na opção “Senha Raiz” uma senha para acesso como administrador “root”.

Figura 36. Instalação CentOS.

Fonte: Autor (2019).

Na Figura 37 é configurada uma senha para o acesso como administrador ao CentOS.

Figura 37. Configuração de senha CentOS.



Fonte: Autor (2019).

Após a instalação do sistema operacional CentOS e seu acesso, é feito as configurações necessárias para a instalação do OpenStack, com terminal de comando do CentOS, é necessário atualizar o sistema operacional com o comando:

yum update

O próximo passo é instalar algumas ferramentas complementares com o comando:

yum install vim wget git ntp firewalld

Habilitado o serviço de firewall e gerência de rede:

systemctl enable firewalld NetworkManager

Habilitado serviço de sincronização de relógios:

systemctl enable ntpd

Habilitado inicialização dos serviços de firewall, gerência de rede e sincronização de relógios:

systemctl start firewalld ntpd NetworkManager

Verificado o status inicialização dos serviços de firewall, gerência de rede e sincronização de relógios:

systemctl status firewalld ntpd NetworkManager

Efetuado paralisação dos serviços de firewall e gerência de rede:

systemctl stop firewalld NetworkManager

Desabilitado os serviços de firewall e gerência de rede:

systemctl disable firewalld NetworkManager

Editado e desabilitado a estrutura SeLinux (Security Enhanced Linux) configuração da camada de segurança extra do CentOS:

vim /etc/selinux/config

Desabilitar o SeLinux:

SELINUX=DISABLED

Editado a configuração de transposição do nome da máquina para um endereço IP:

vim /etc/hosts

Adicionado a configuração abaixo com o IP do micro, nome do micro e o seu nome de domínio qualificado (FQDN):

192.168.1.102 dhcppc2 dhcppc2.localdomain

Atualizado o servidor de horas:

ntpdate br.pool.ntp.org br.pool.ntp.org br.pool.ntp.org

Reiniciado a máquina virtual e após seu inicio é verificado se os serviços de firewall, gerência de rede e sincronização de relógios estão desativados:

systemctl firewalld NetworkManager ntpd status

Iniciar apenas serviço ntpd:

systemctl start ntpd

Configurado a placa de rede para melhor visualização e organização:

cd /etc/sysconfig/network-scripts

Editado a configuração da placa de rede física:

vim enp0s25

Efetuado nova configuração:

DEVICE=enp0s25

DEVICETYPE=enp0s25

TYPE=Ethernet

BOOTPROTO=static

IPADDR=192.168.1.102

NETMASK=255.255.255.0

GATEWAY=192.168.1.1

DNS=8.8.8.8

ONBOOT=yes

Reiniciado a placa de rede física:

systemctl restart network

Iniciado a instalação do OpenStack:

yum install centos-release-openstack-rocky

Atualizado sistema operacional novamente:

yum update

Instalado utilitário de instalação do OpenStack:

yum install openstack-packstack

Utilizado o utilitário Packstack para instalar OpenStack:

packstack --allinone --provision-demo=n

O comando “allinone” direciona que todos os serviços do OpenStack vão ser instalados em um único sistema operacional, o comando “provision-demo=n” delimita que o projeto demo não será instalado.

Após a finalização da instalação é instalado ferramentas de gerenciamento de serviço OpenStack:

yum install openstack-utils

Verificado se todos os serviços do OpenStack estão acionados:

openstack-service status

Localizado a pasta com as configurações da placa de rede física com intuito de criação de uma interface de rede virtual para o OpenStack:

cd /etc/sysconfig/network-scripts/

Copiado a configuração da placa de rede física para uma nova placa de rede virtual:

cp ifcfg-enp0s25 ifcfg-br-ex

Editado a placa de rede virtual:

vim ifcfg-br-ex

Configuração aplicada à placa de rede virtual:

DEVICE=enp0s25

DEVICETYPE=enp0s25

TYPE=Ethernet

BOOTPROTO=static

IPADDR=192.168.1.102

NETMASK=255.255.255.0

GATEWAY=192.168.1.1

DNS1=8.8.8.8

ONBOOT=yes

Editado a placa de rede física:

vim enp0s25

Configuração aplicada à placa de rede física:

DEVICE=enp0s25

HWADDR= "Aqui vai o número Mac Address da placa de rede física"

TYPE=OVSPort

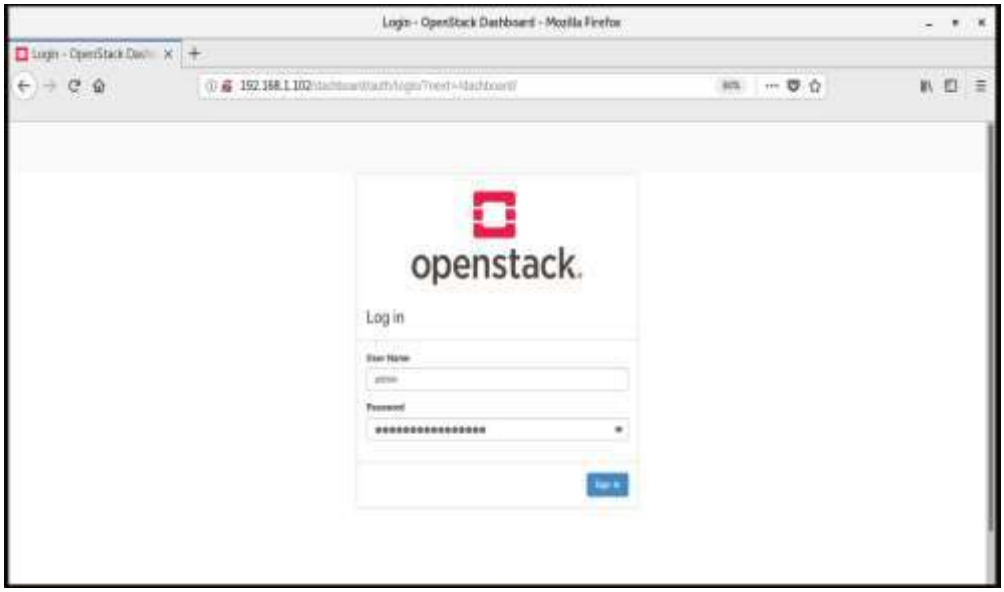
DEVICETYPE=ovs

OVS_BRIDGE=br-ex

ONBOOT=yes

Na Figura 38 é efetuado acesso ao Dashboard do OpenStack via navegador browser de internet, o Mozilla Firefox, o endereço utilizado para acesso é o 192.168.1.102. Usuário e senha utilizados se utilizados para acesso como administrador se encontra no arquivo keystonerc_admin.

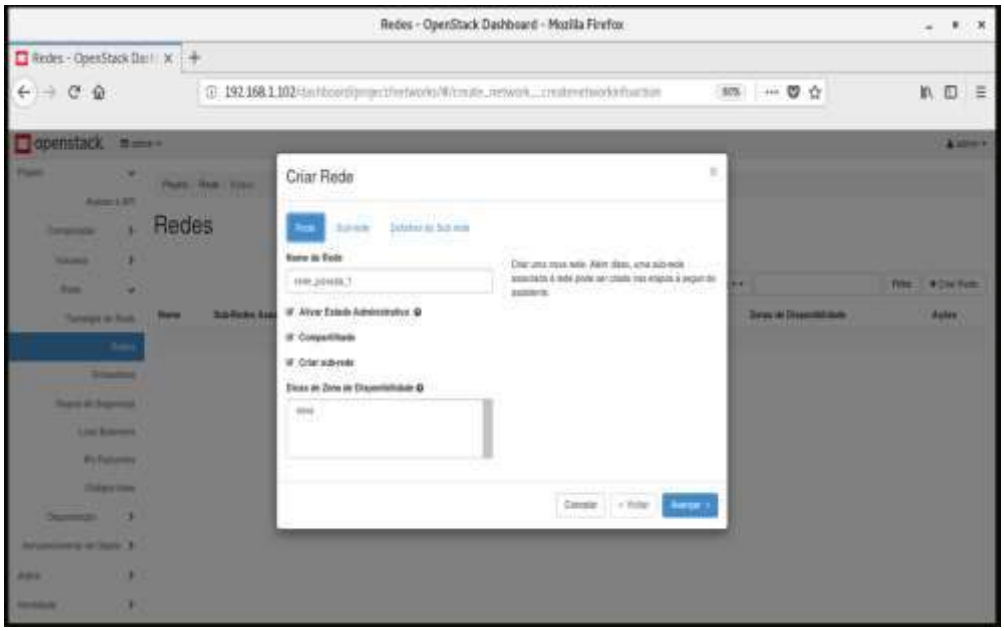
Figura 38. Dashboard OpenStack.



Fonte: Autor (2019).

Na Figura 39 é demonstrado o acesso ao painel de gerenciamento do OpenStack, acessado o menu “Redes” e após clicado no botão “+ Criar Rede”, a página “Criar Rede” se apresenta, a rede foi nomeada como “rede_privada_1”, ativado as opções de “Ativar Estado Administrativo” para o administrador ter a possibilidade de configurar novamente a rede quando já criada, “Compartilhado” para a rede ter a possibilidade de ser compartilhada com outras redes e “Criar sub-rede” que irá habilitar a opção de sub-rede.

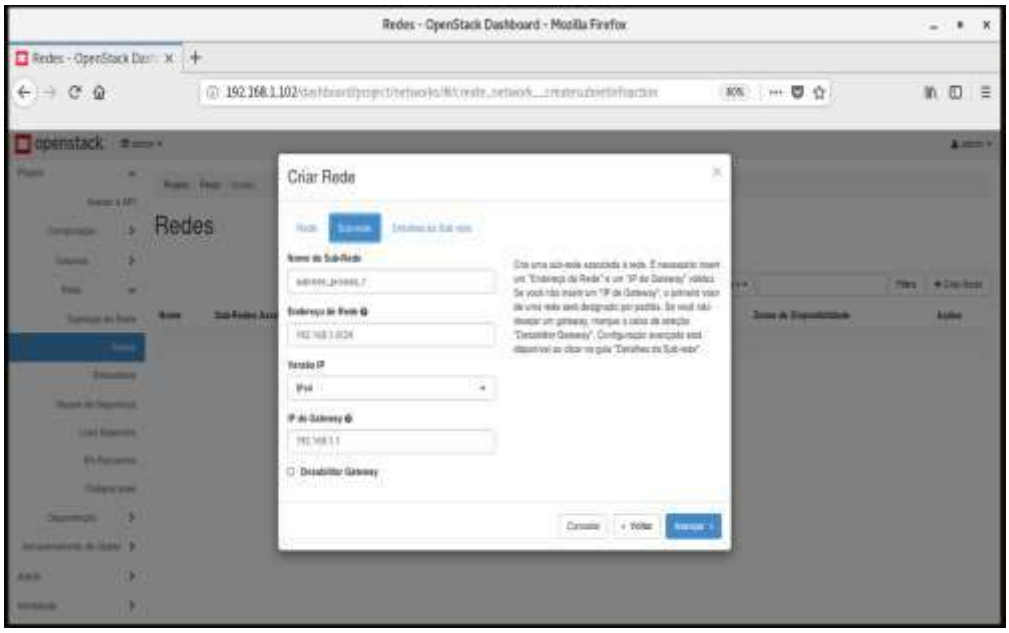
Figura 39. Configuração rede_privada_1.



Fonte: Autor (2019).

Na Figura 40 é demonstrado o menu “Sub-rede”, nomeado a Sub-rede como “subrede_privada_1”, adicionado o endereço de IP 192.168.1.0/24, que será o endereço inicial para a rede, configurado como versão do IP para IPV4,o IP gateway setado é o 192.168.1.1 que é o roteador (Multilaser RE033) conectado á nuvem privada OpenStack.

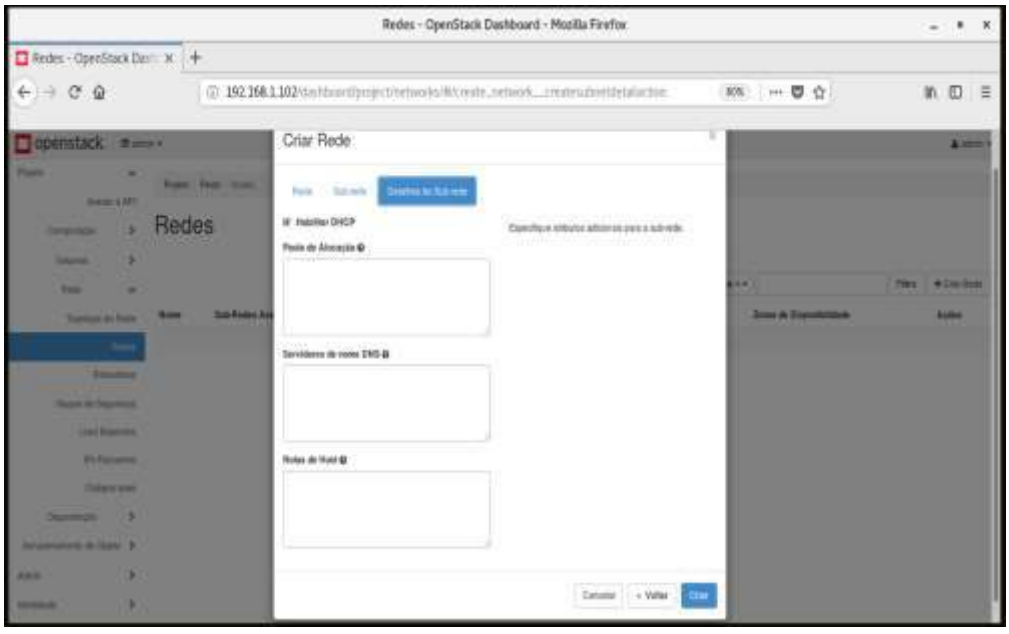
Figura 40. Configuração sub-rede rede_privada_1.



Fonte: Autor (2019).

Na Figura 41 é habilitada a opção de DHCP e após o botão “Criar”.

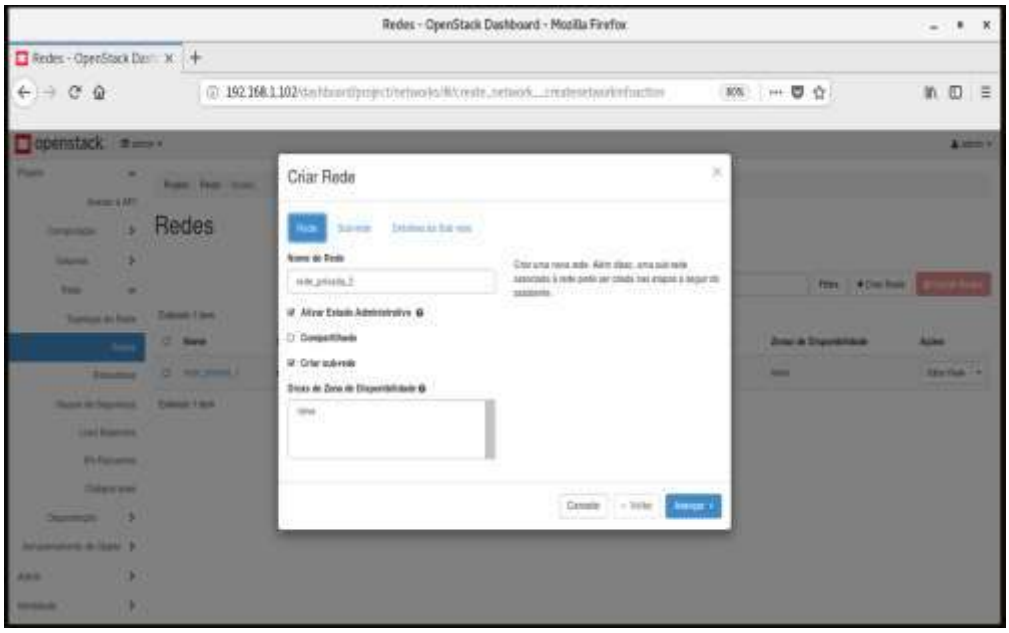
Figura 41. Configuração detalhes sub-rede rede_privada_1.



Fonte: Autor (2019).

Na Figura 42 é criada outra rede nomeada de “rede_privada_2”. Habilitada para ativar estado administrativo, não compartilhado e criação de sub-rede.

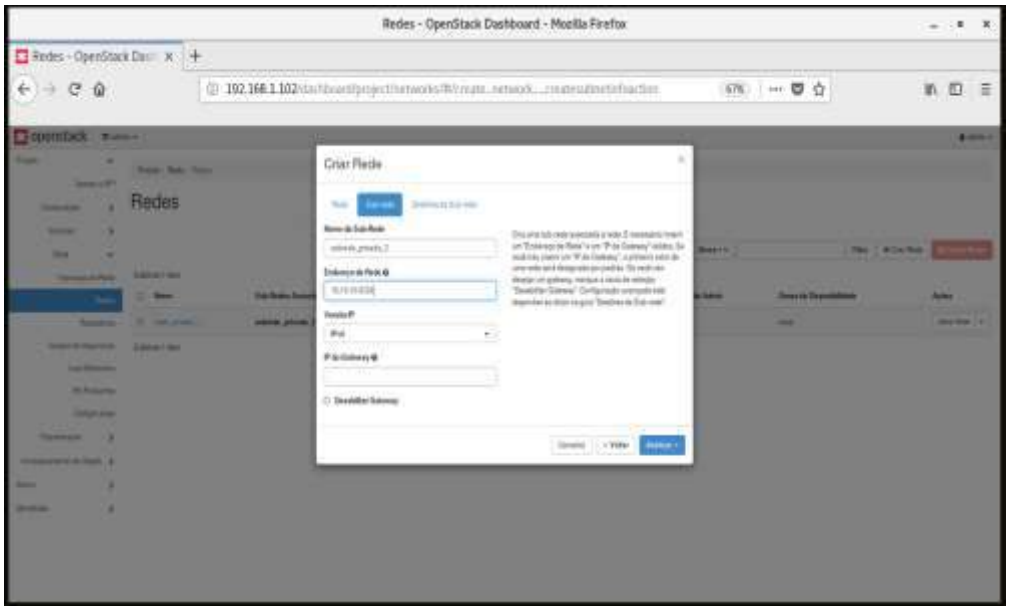
Figura 42. Configuração rede_privada_2.



Fonte: Autor (2019).

Na Figura 43 é configurado o nome da sub-rede para “rede_privada_2”, endereço de rede para 10.10.10.0/24, versão do IP IPv4 e sem endereço de gateway,o OpenStack oferece o serviço de gateway virtual se o administrador de rede necessita da criação de uma nova rede.

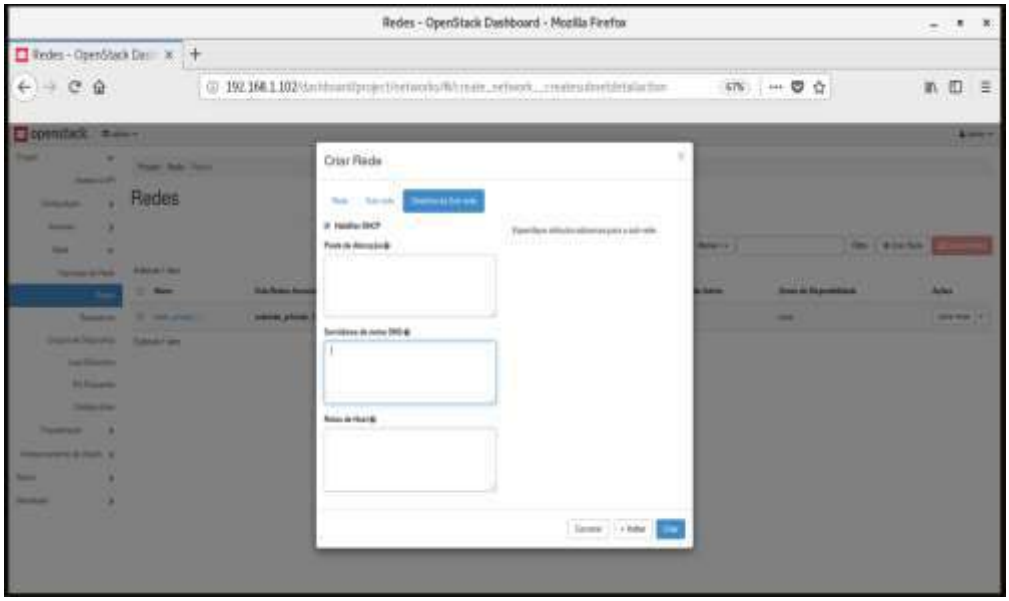
Figura 43. Configuração sub-rede rede_privada_2.



Fonte: Autor (2019).

Na figura 44 é habilitado o DHCP e acionado o botão “Criar”.

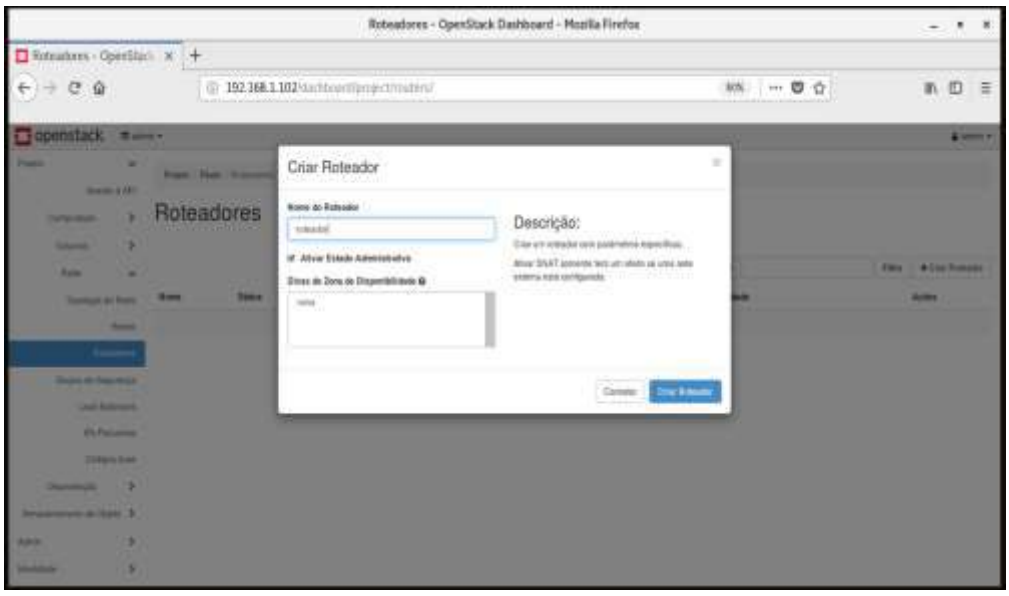
Figura 44. Configuração detalhes sub-rede rede_privada_2.



Fonte: Autor (2019).

Na Figura 45 é demonstrado o acesso ao menu “Roteadores”, acionado o botão “+ Criar Roteador”, na página “Criar Roteador” foi configurado o nome do roteador como “roteador”, ativado estado administrativo e acionado o botão “Criar Roteador”.

Figura 45. Configuração roteador.



Fonte: Autor (2019).

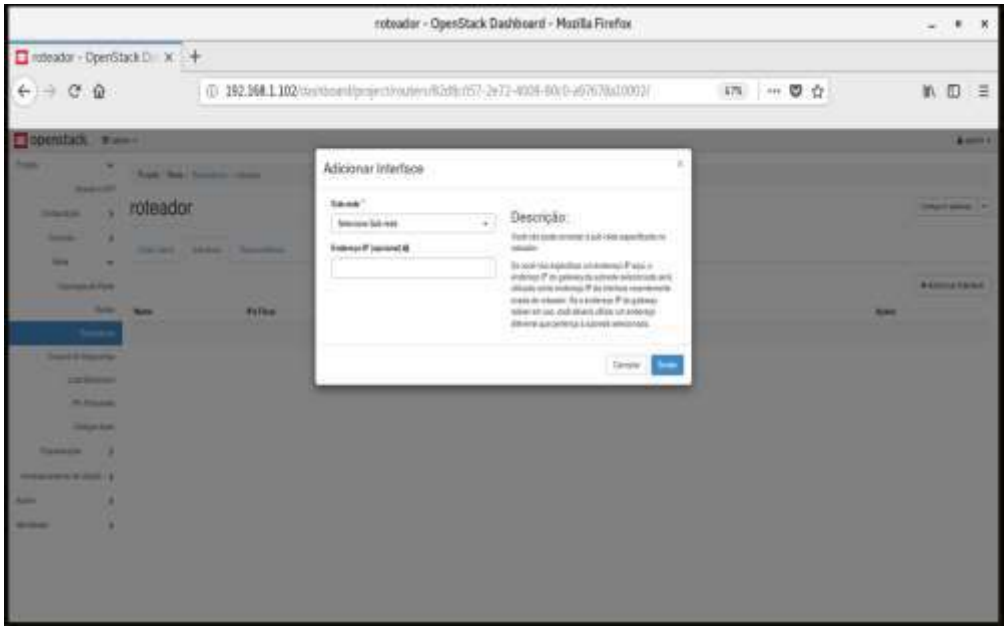
Na Figura 46 é demonstrado o menu “Topologia de Rede”, a página apresenta as duas redes privadas criadas e o roteador conectado a “rede_privada_1.

Figura 46. Topologia de rede (roteador + rede_privada_1).

Fonte: Autor (2019).

Na Figura 47 foi acionada a opção de “Adicionar Interface” clicando no roteador conforme a Figura 46 acima. Na página “Adicionar Interface” é adicionado a sub-rede “rede_privada_2”.

Figura 47. Adicionar interface ao roteador.



Fonte: Autor (2019).

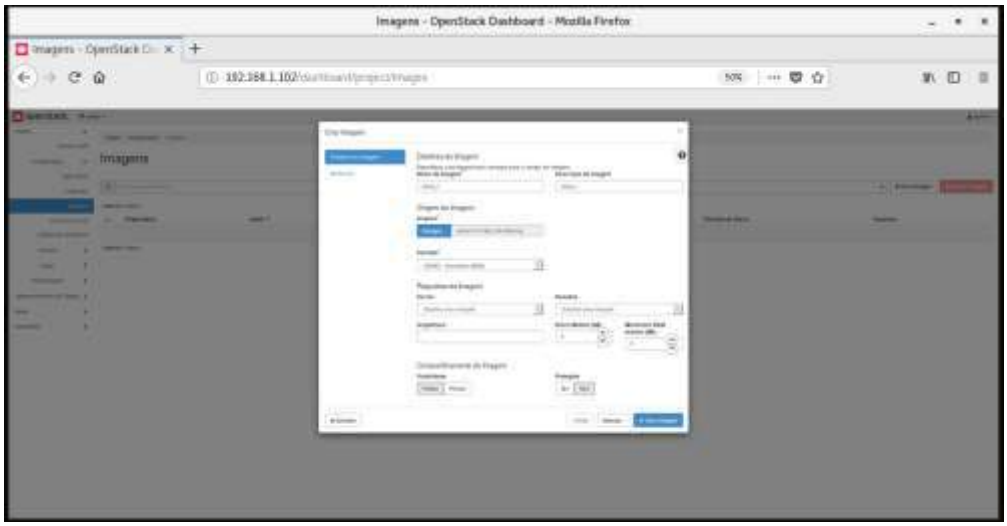
Na Figura 48 são demonstradas as duas redes privadas conectadas ao roteador virtual.

Figura 48. Topologia (roteador + rede_privada_1 + rede_privada_2).

Fonte: Autor (2019).

Na Figura 49 é demonstrado o acesso ao menu “Imagens” e acionado o botão “+ Criar Imagem”, na página “Criar Imagem” é nomeado e a descrição da imagem como cirros_1, a origem da imagem foi escolhida uma imagem da distribuição Linux CirrOS. O formato de leitura escolhido é o QCOW2 (formato de arquivos que otimiza armazenamento em disco) emulador de QEMU (virtualização completa de um sistema dentro de outro). Acionado o botão “Criar Imagem”.

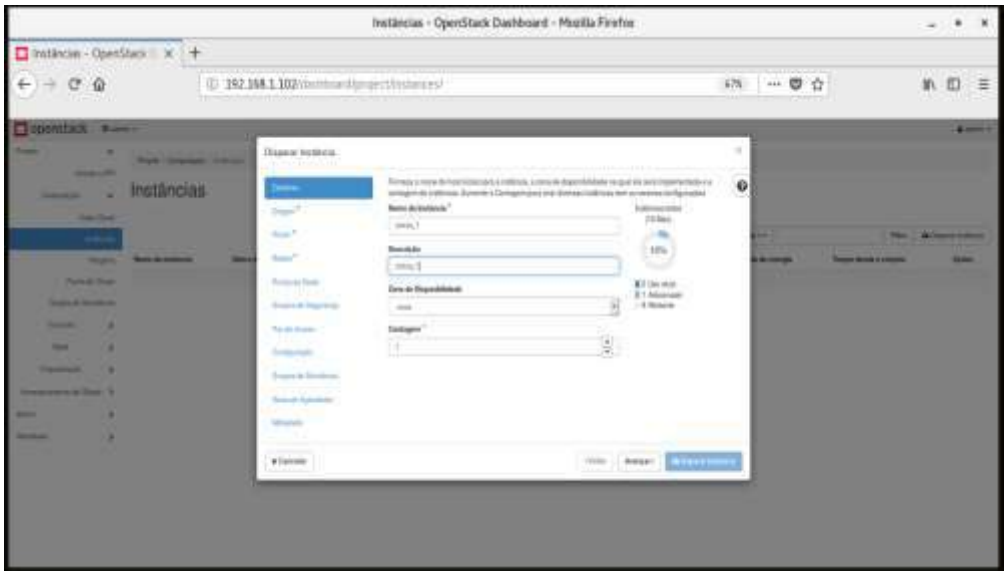
Figura 49. Criação imagem.



Fonte: Autor (2019).

Na Figura 50 é demonstrado o menu “Instâncias” e acionado o botão “Disparar Instância. Na página “ Disparar Instância” e na aba “Detalhes” é nomeado a instância como nome e descrição “cirros_1”.

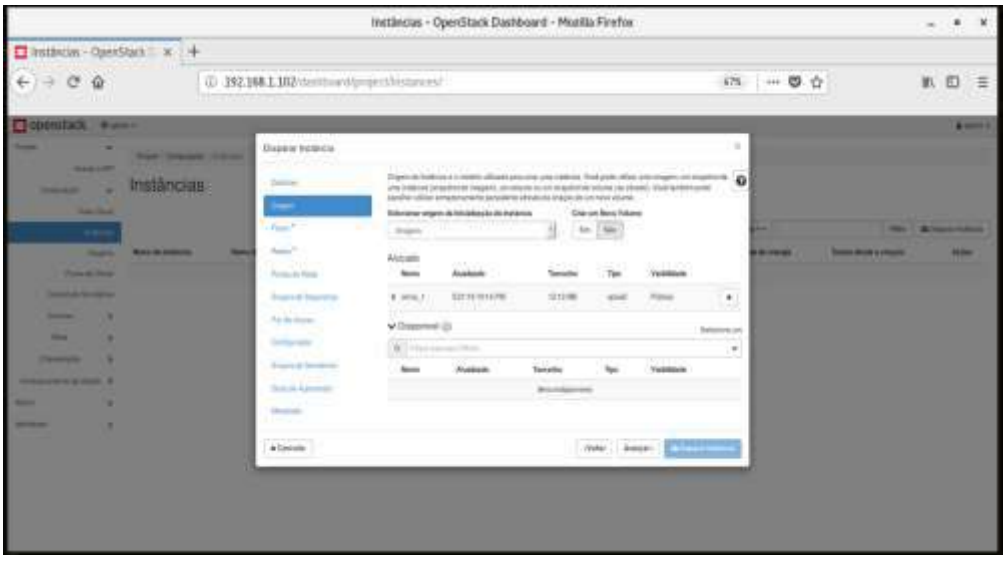
Figura 50. Detalhes instância.



Fonte: Autor (2019).

Na Figura 51 é demonstrada a aba “Origem” onde é acionado o botão “Não” para criar um novo volume e configurado para a imagem “cirros_1”.

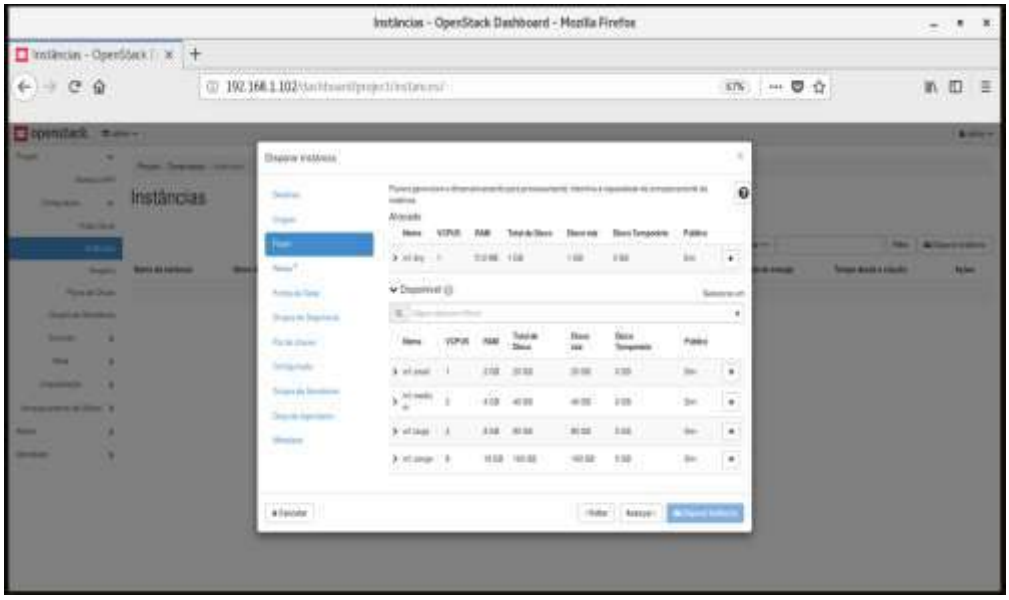
Figura 51. Origem instância.



Fonte: Autor (2019).

Na Figura 52 é demonstrado a aba “Flavor” que é alocado a configuração “m1.tiny” para a instância, as configurações são de um processador virtual (1 Vpcu), 512MB (memória RAM) e com um total e disco raiz de 1GB.

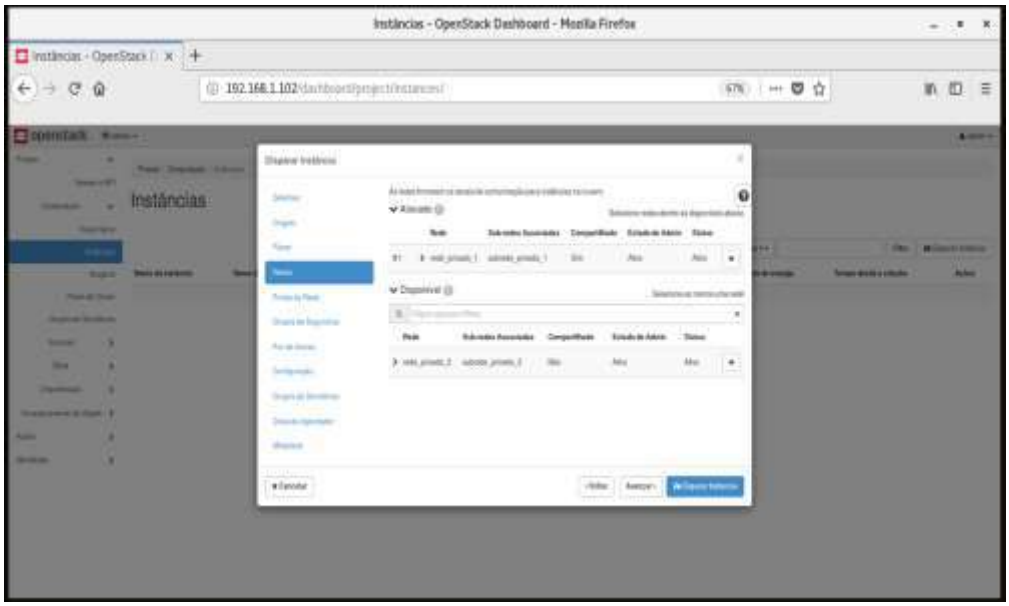
Figura 52. Flavor instância.



Fonte: Autor (2019).

Na Figura 53 é alocado na aba “Redes” a configuração de “rede_privada_1” para a instância. Acionado o botão “Disparar Instância”.

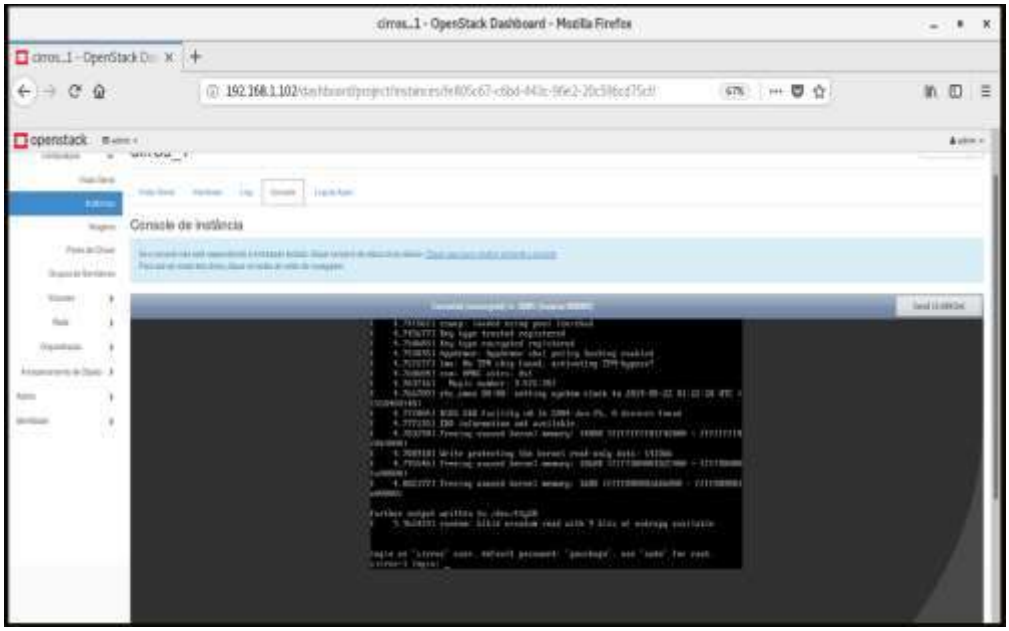
Figura 53. Rede instância.



Fonte: Autor (2019).

Na Figura 54 é demonstrado que ao acionar a instância “cirros_1” e a aba “Console”, se tem a visualização da máquina virtual e acesso a instância “cirros_1”.

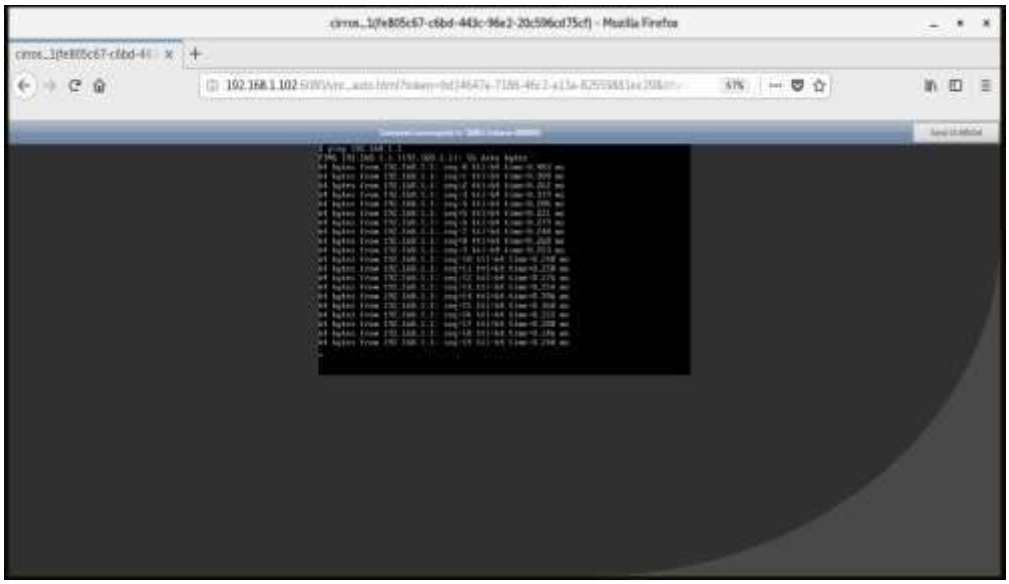
Figura 54. Console instância.



Fonte: Autor (2019).

Na Figura 55 é demonstrado o teste de conexão fora da nuvem privada até o roteador gateway de endereço 192.168.1.1 (Multilaser RE033).

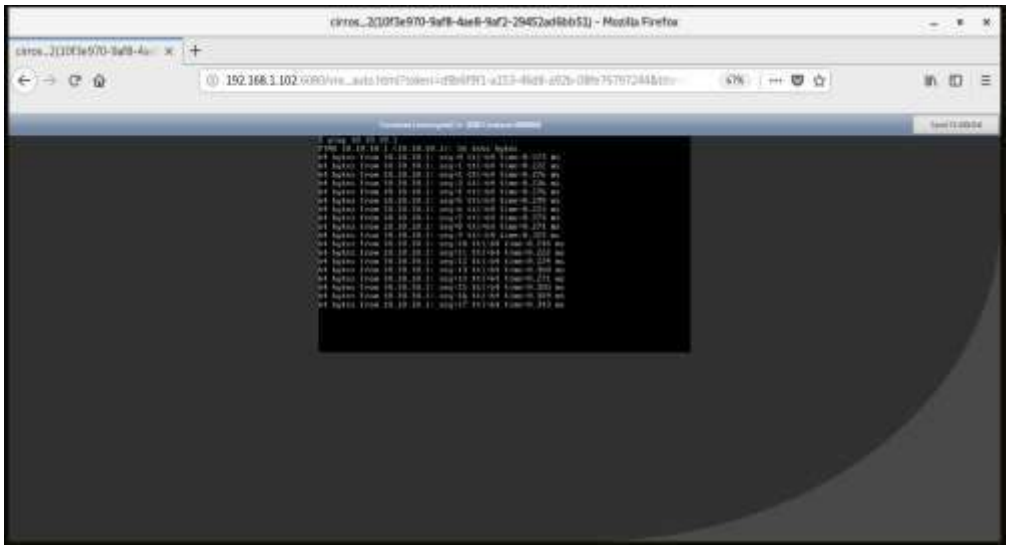
Figura 55. Teste de comunicação instância cirros_1.



Fonte: Autor (2019).

A configuração da instância “cirros_2” segue os passos semelhantes ao do “cirros_1”, a diferença de configuração se encontra aba “Redes”, onde a configuração acionada é a “rede_privada_2” para a instância. Na Figura 56 é demonstrado o teste de comunicação com a rede virtual criada (Gateway 10.10.10.1).

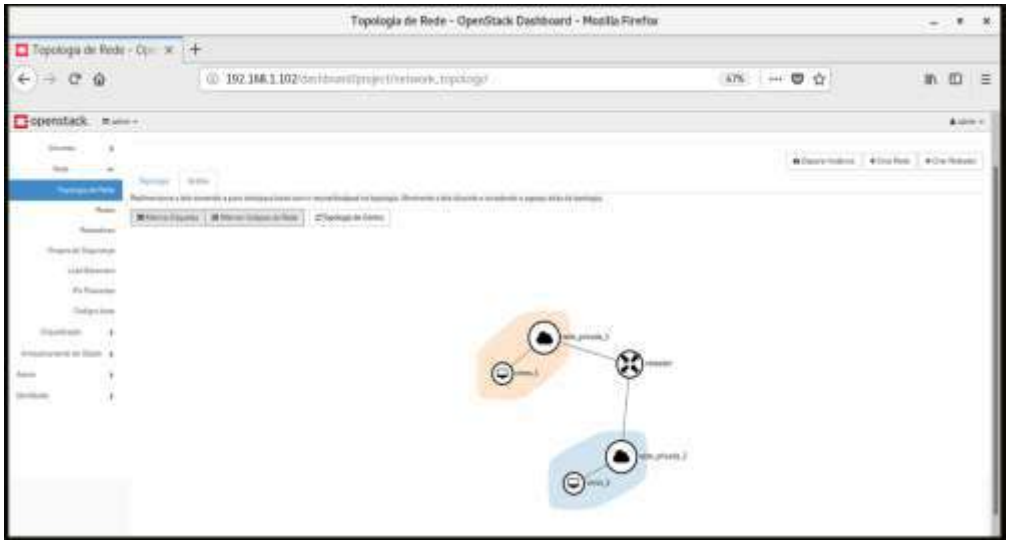
Figura 56. Teste de comunicação instância cirros_2.



Fonte: Autor (2019).

Na Figura 57 é demonstrado no menu “Topologia de Rede” na aba “Gráficos” toda a infraestrutura virtual criada, demonstrando a implementação de uma infraestrutura como serviço em nuvem privada.

Figura 57. Topologia de rede modo gráfico.



Fonte: Autor (2019).

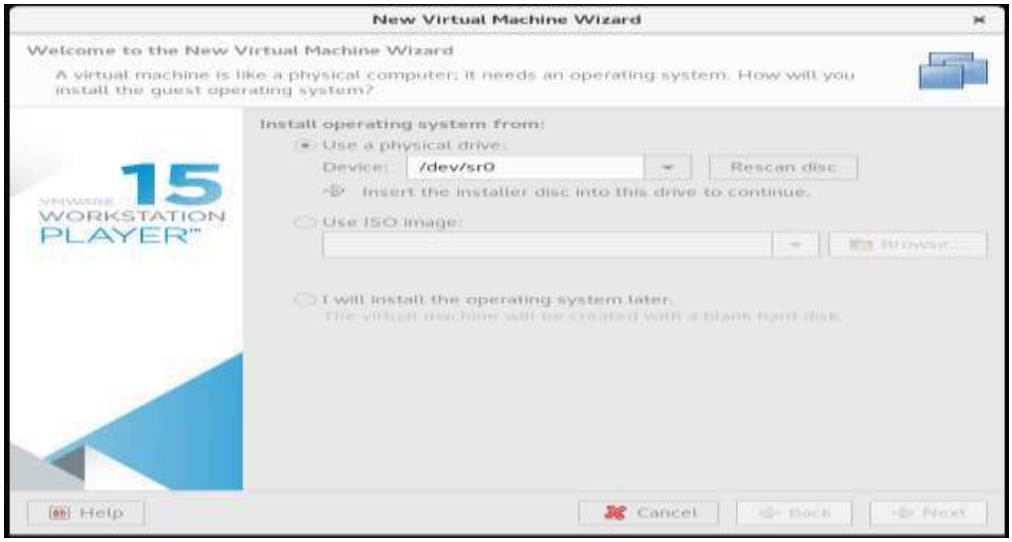
Após a implementação da infraestrutura como serviço foi elaborado a ferramenta de simulação de ataque, na Figura 58 é demonstrado a página inicial do VMware para criação de uma máquina virtual (Create a New Virtual Machine) com o sistema operacional Kali Linux.

Figura 58. Página inicial VMware.

Fonte: Autor (2019).

Na Figura 59 é representada a escolha do disco físico a ser utilizado e a utilização de uma imagem ISO para instalação do Kali Linux.

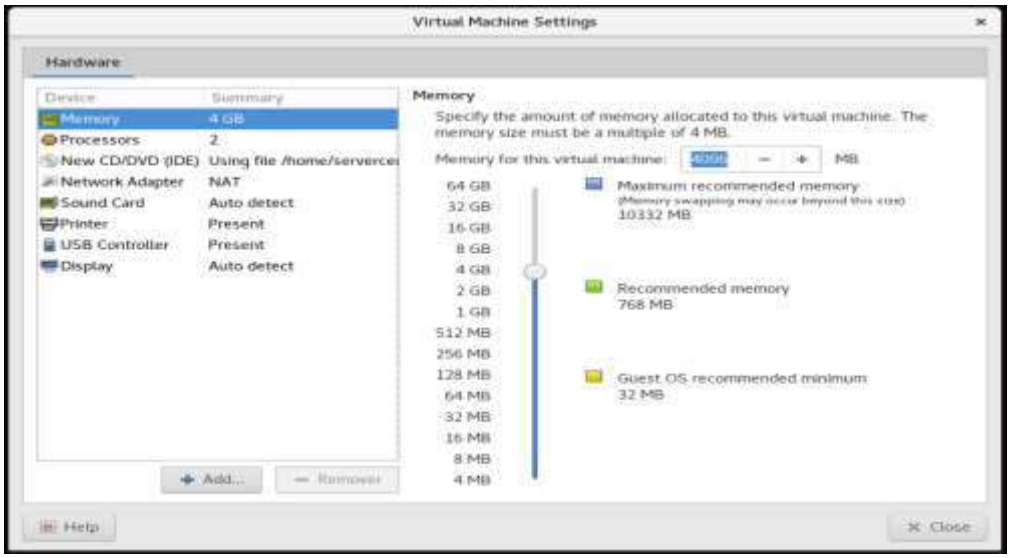
Figura 59. Configuração de disco e imagem VMware.



Fonte: Autor (2019).

Na Figura 60 é demonstrado às configurações implementadas para a criação da máquina virtual.

Figura 60. Configuração de hardware VMware.



Fonte: Autor (2019).

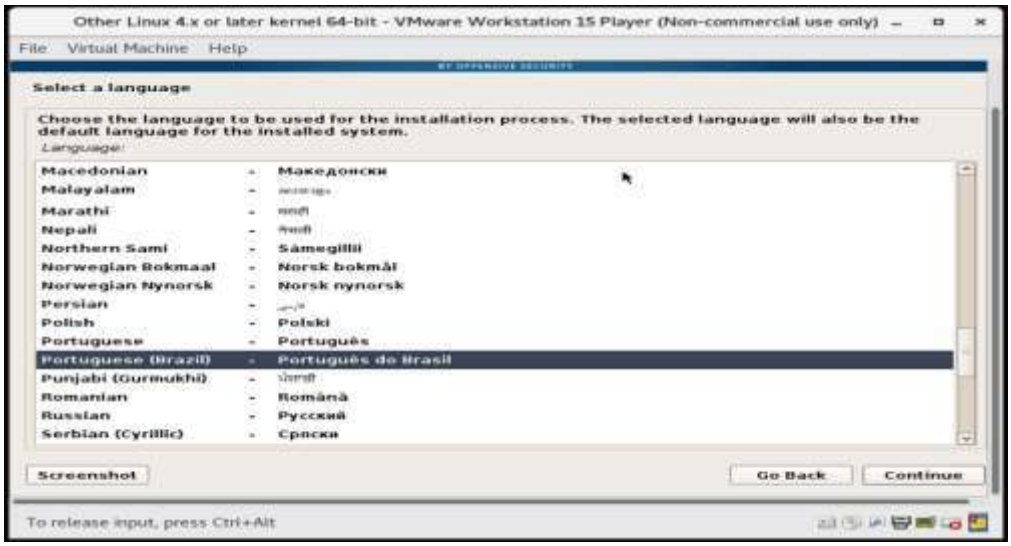
Na Figura 61 é demonstrada a página inicial de instalação do Kali Linux, selecionado a opção “Graphical install”.

Figura 61. Página inicial de instalação Kali Linux.

Fonte: Autor (2019).

Na Figura 62 é demonstrada a escolha de linguagem (Portuguese Brazil) para instalação.

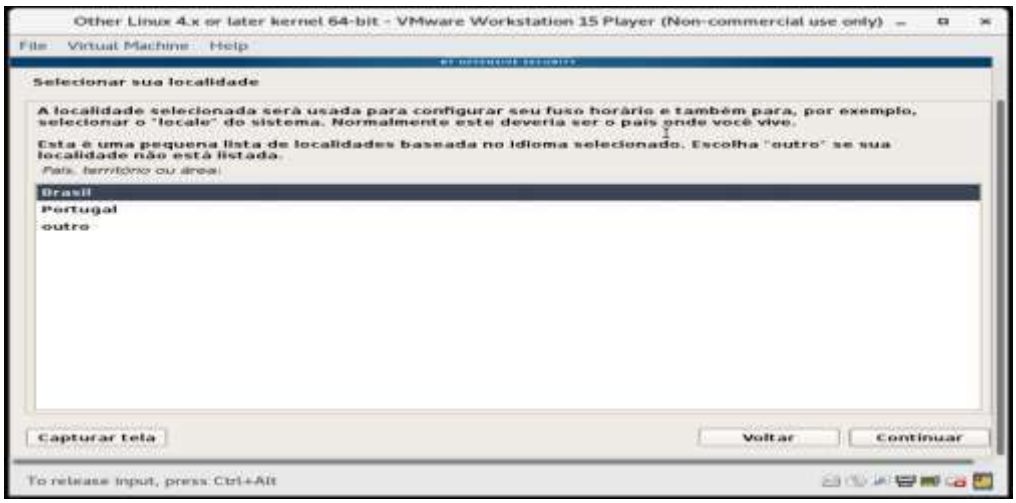
Figura 62. Seleção de idioma Kali Linux.



Fonte: Autor (2019).

Na Figura 63 é demonstrada a seleção de localidade (Brasil) para a instalação.

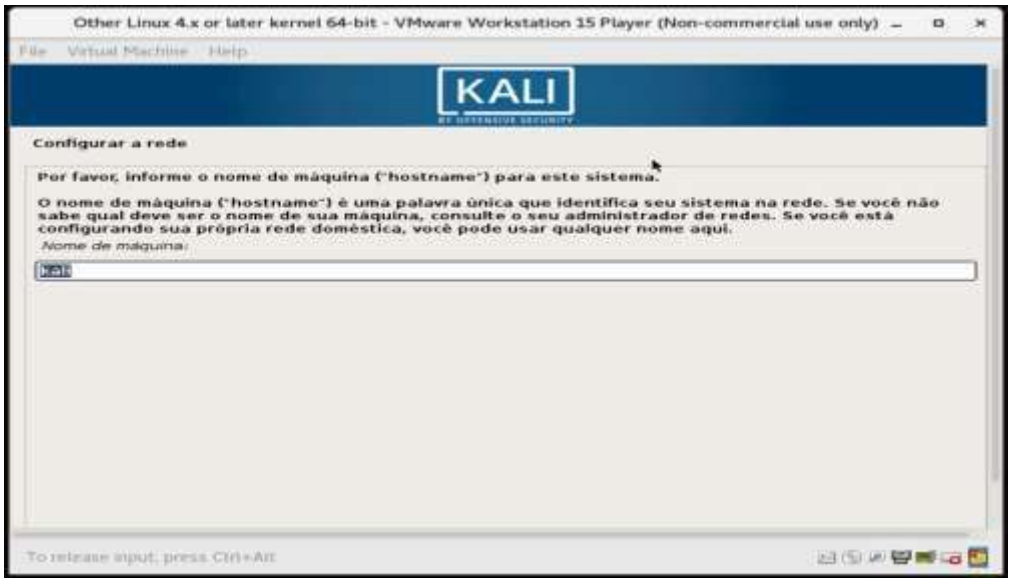
Figura 63. Seleção de localidade Kali Linux.



Fonte: Autor (2019).

Na Figura 64 é demonstrada a escolha do nome de máquina (Kali) para o sistema operacional.

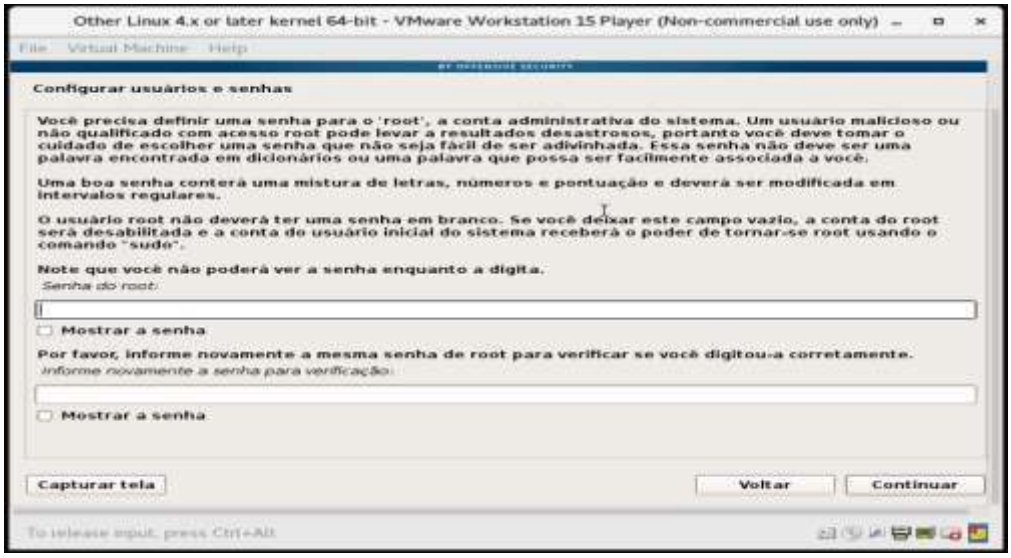
Figura 64. Seleção de hostname Kali Linux.



Fonte: Autor (2019).

Na Figura 65 é demonstrada a criação uma senha para acesso (root) administrador ao sistema operacional Kali Linux.

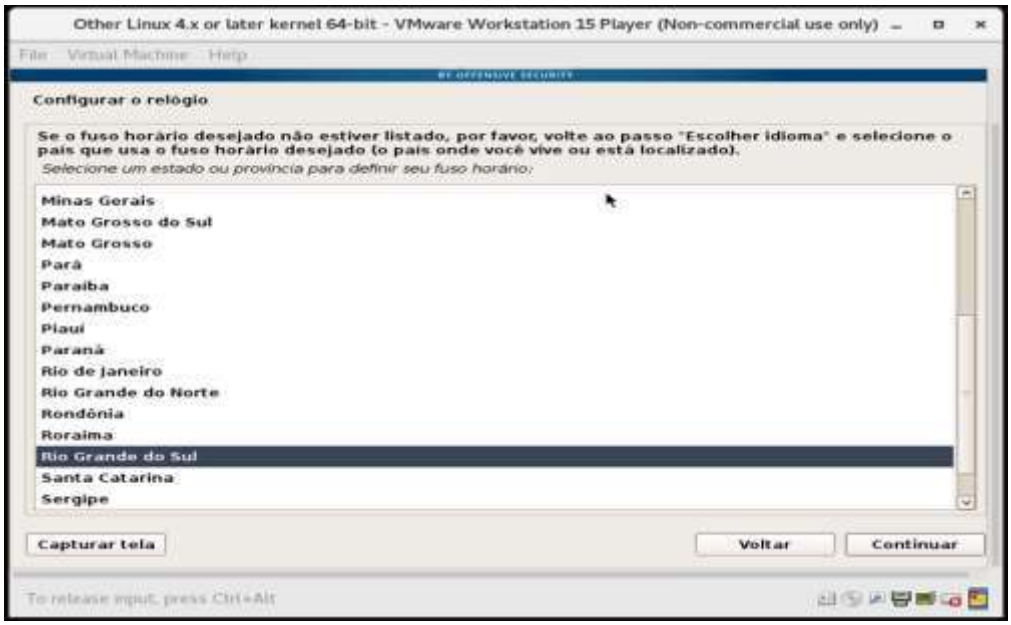
Figura 65. Criação de senha de administrador Kali Linux.



Fonte: Autor (2019).

Na Figura 66 é demonstrada a seleção de fuso horário (Rio Grande do Sul) para o Kali Linux.

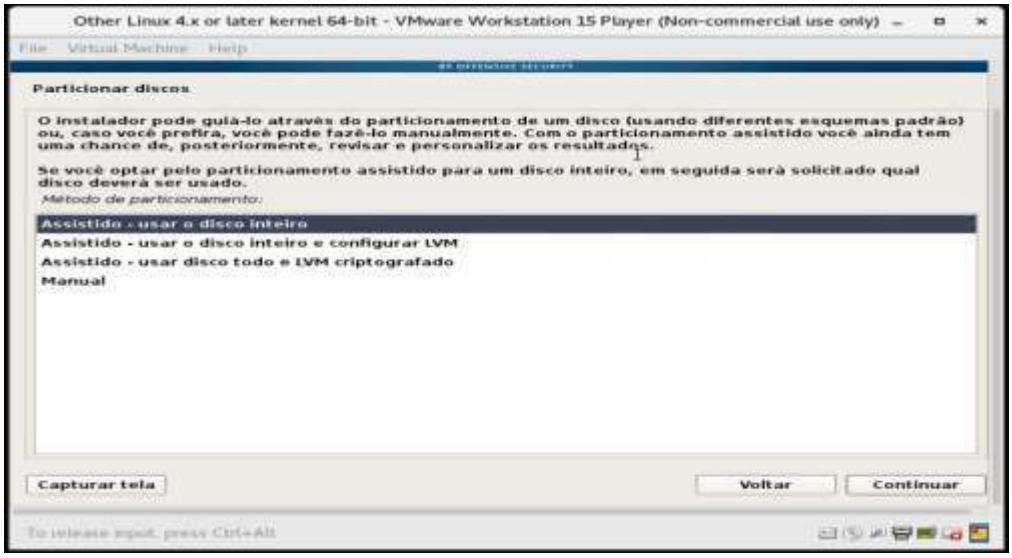
Figura 66. Seleção de fuso horário Kali Linux.



Fonte: Autor (2019).

Na Figura 67 é demonstrada a utilização da configuração de instalação para usar o disco inteiro.

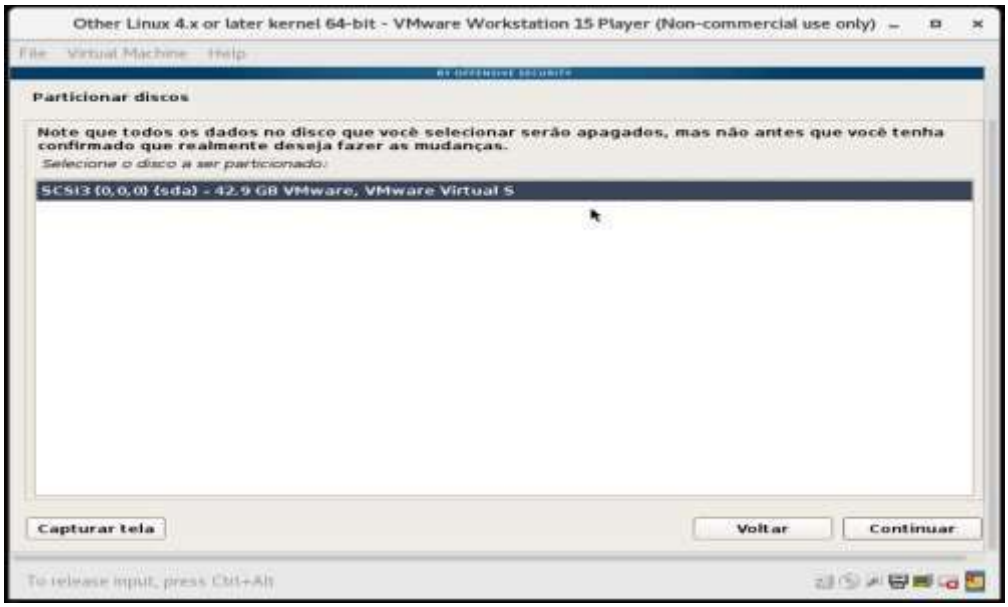
Figura 67. Seleção de partição de disco Kali Linux.



Fonte: Autor (2019).

Na Figura 68 é demonstrada o disco virtual criado pelo VMware a ser instalado o Kali Linux.

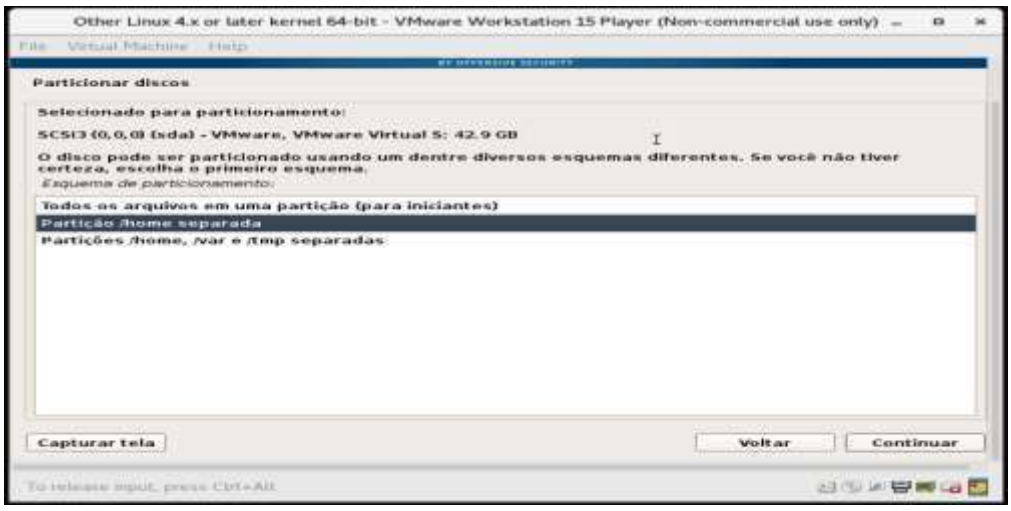
Figura 68. Seleção de disco virtual.



Fonte: Autor (2019).

Na Figura 69 é demonstrada a seleção onde a partição (/home) será separada da principal, esta partição separa os ficheiros pessoais de qualquer utilizador do Kali Linux.

Figura 69. Seleção de partição Kali Linux.



Fonte: Autor (2019).

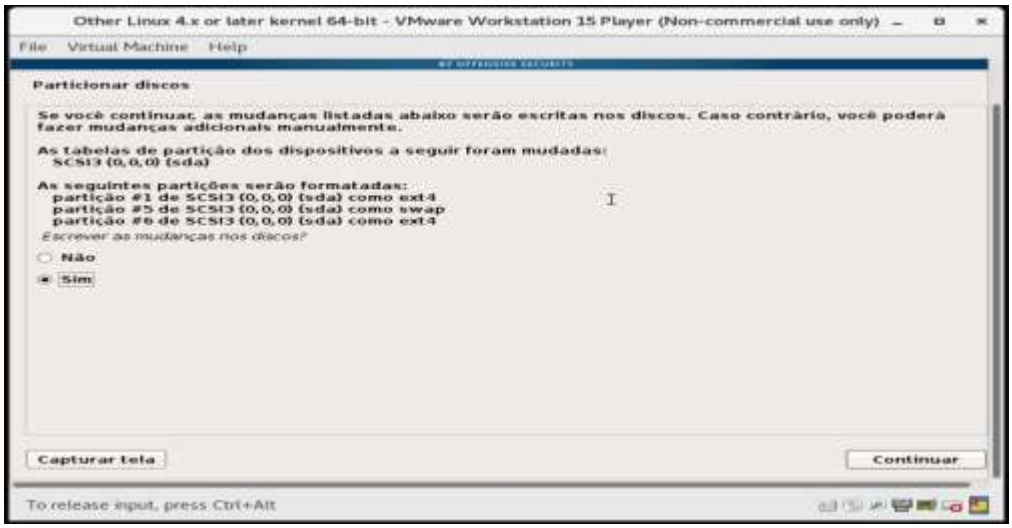
Na Figura 70 são demonstradas as mudanças de particionamento no disco virtual.

Figura 70. Demonstração de partições Kali Linux.

Fonte: Autor (2019).

Na Figura 71 é demonstrada a confirmação de formatação das partições para instalação do Kali Linux.

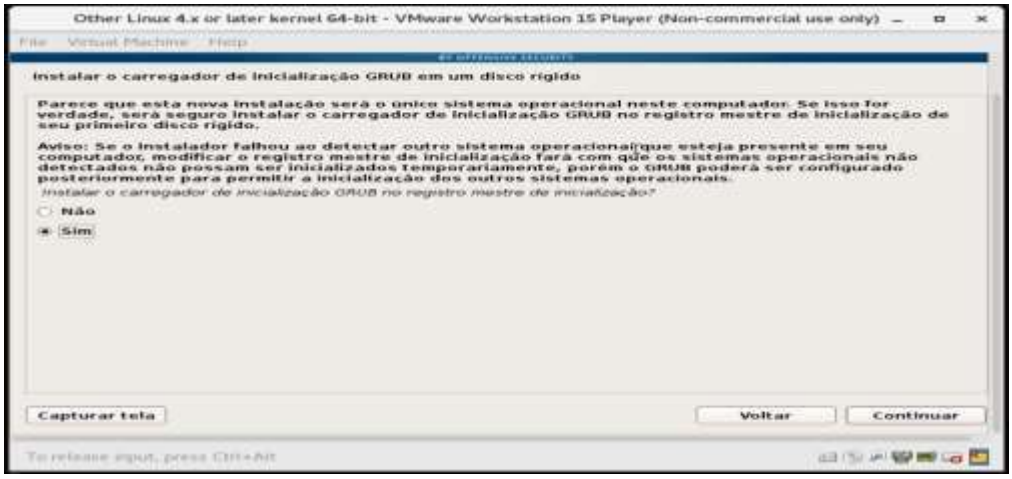
Figura 71. Formatação de partição Kali Linux.



Fonte: Autor (2019).

Na Figura 72 é demonstrada a seleção para instalação de inicialização “Grub”, esta ferramenta impede a instalação de outros sistemas operacionais neste disco virtual como método de segurança.

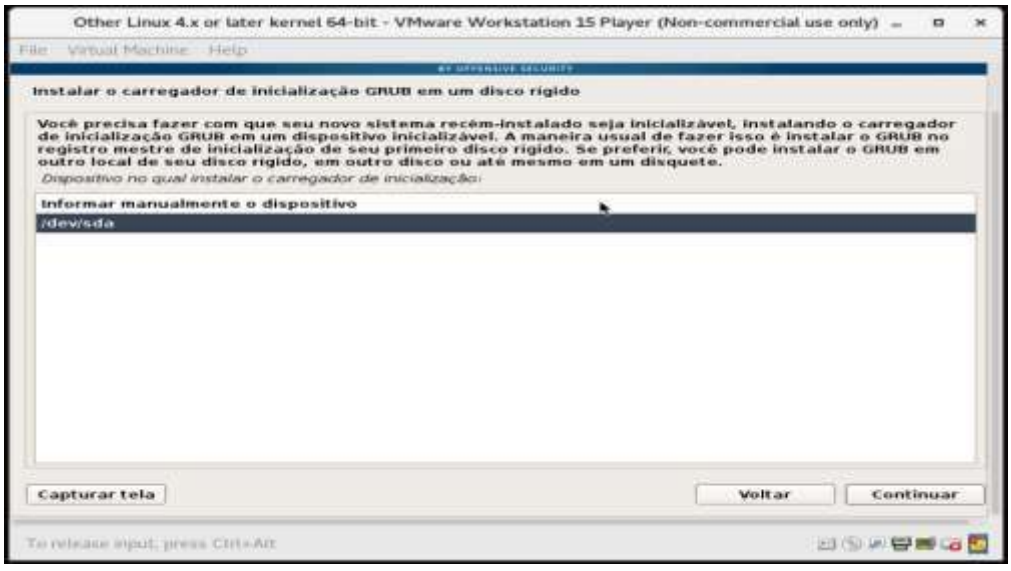
Figura 72. Seleção de instalação Grub.



Fonte: Autor (2019).

Na Figura 73 é demonstrada a escolha do local na partição para a instalação do “Grub”.

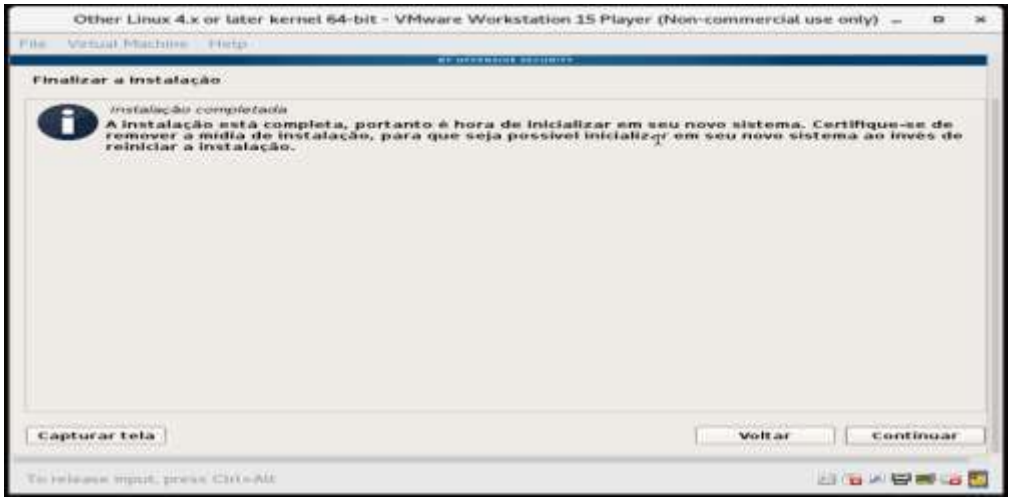
Figura 73. Seleção de partição de instalação Grub.



Fonte: Autor (2019).

Na Figura 74 é demonstrada a finalização da instalação do Kali Linux.

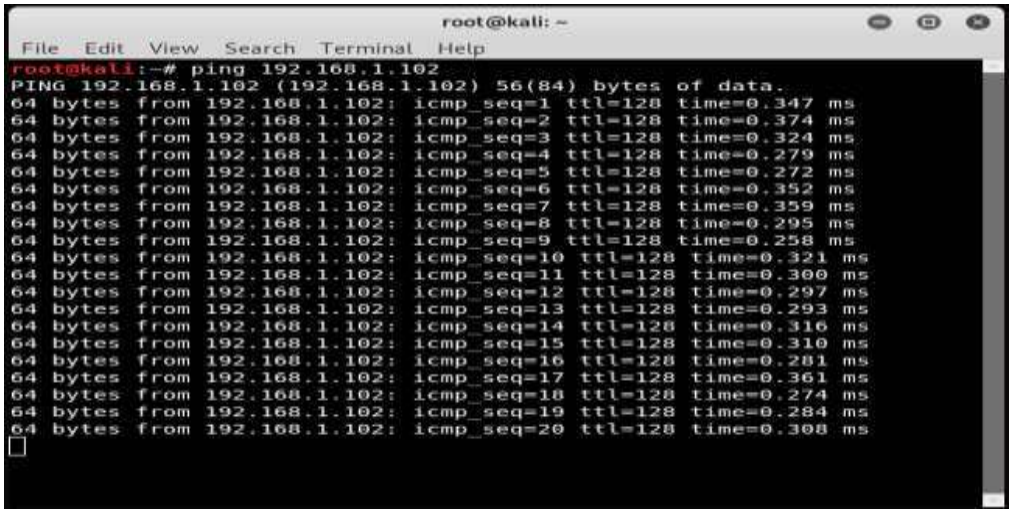
Figura 74. Finalização de instalação Kali Linux.



Fonte: Autor (2019).

Na Figura 75 é demonstrada a tela do terminal de comando do Kali Linux rodando o teste de comunicação (ping 192.168.1.102) por uma sequencia de vinte vezes a nuvem privada OpenStack.

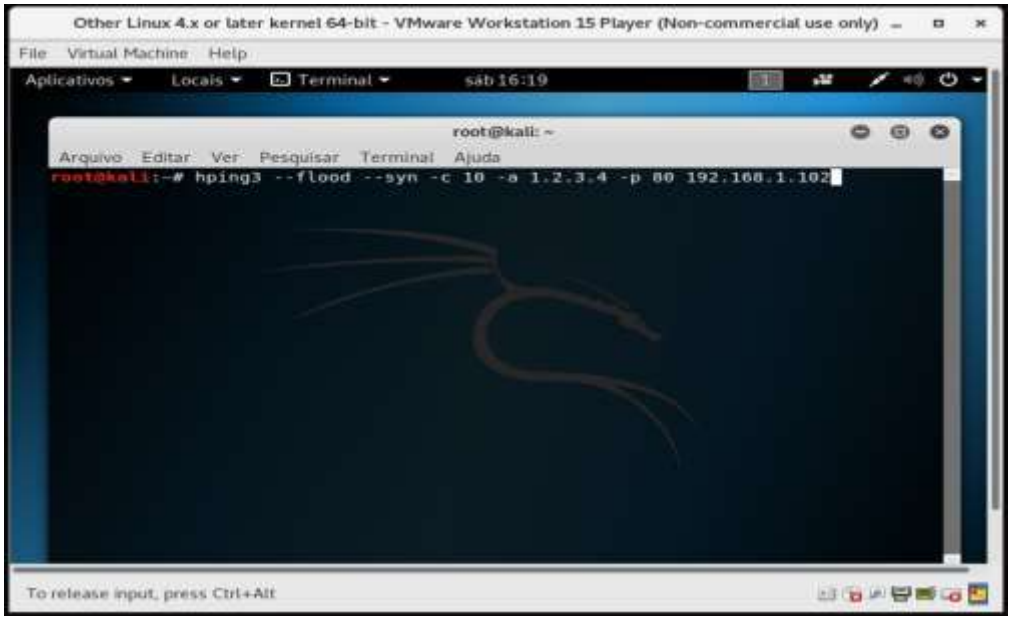
Figura 75. Teste de comunicação Kali Linux.



Fonte: Autor (2019).

Na Figura 76 é demonstrada o terminal de comando do Kali Linux com o comando configurado para simulação de ataque de negação de serviço a nuvem privada OpenStack.

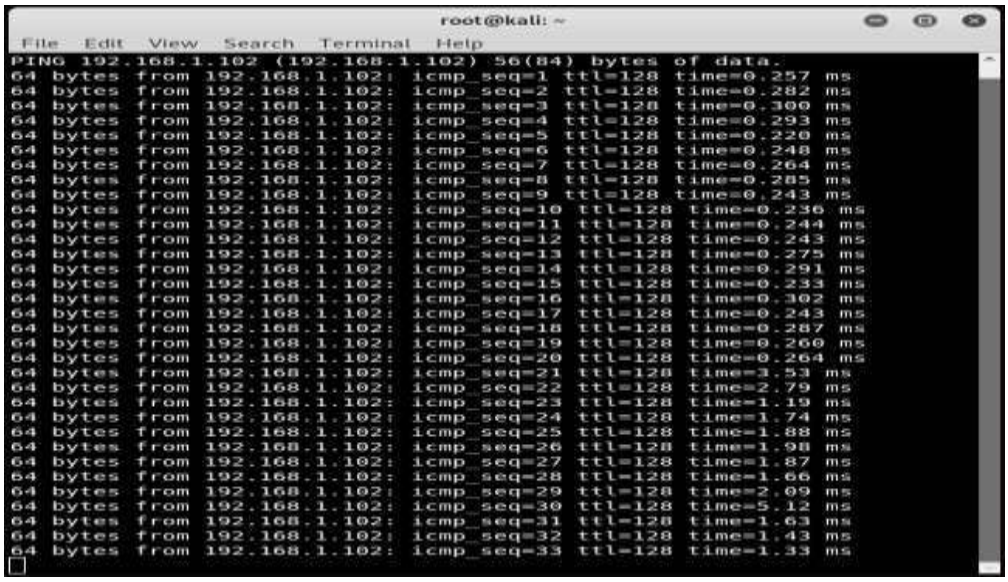
Figura 76. Comando para ataque de negação de serviço.



Fonte: Autor (2019).

Na Figura 77 é demonstrada teste de comunicação com a nuvem privada OpenStack, onde o ataque de negação de serviço se iniciou no pacote de transmissão número 21 e termina no pacote 40 que é demonstrado na Figura 78.

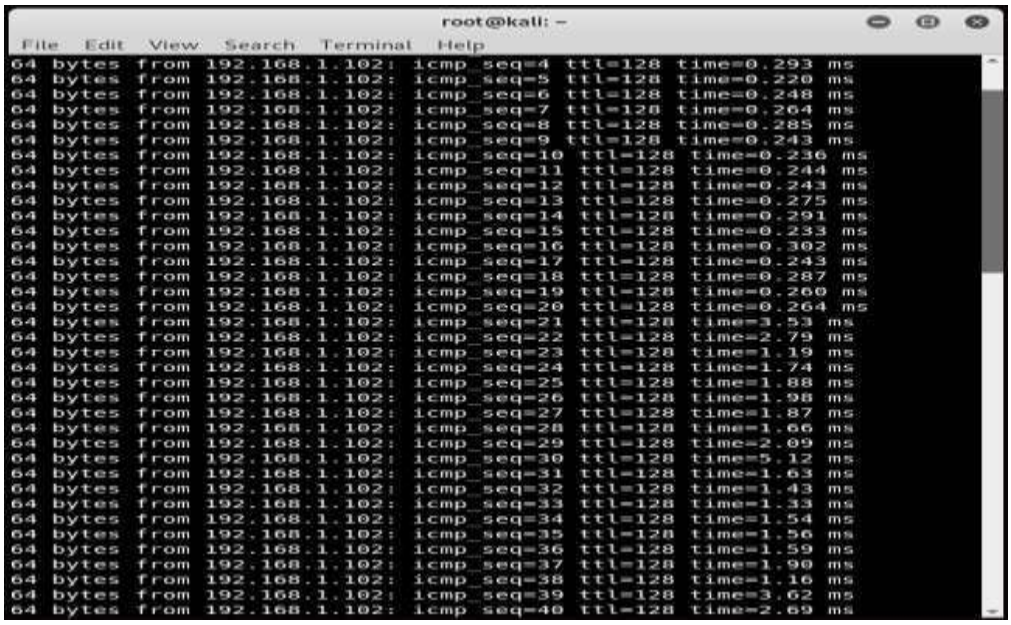
Figura 77. Ataque de negação de serviço (parte 1).



Fonte: Autor (2019).

Na Figura 78 é demonstrada ataque de negação de serviço até o pacote de transmissão número 40, ao qual após isso o ataque foi cancelado.

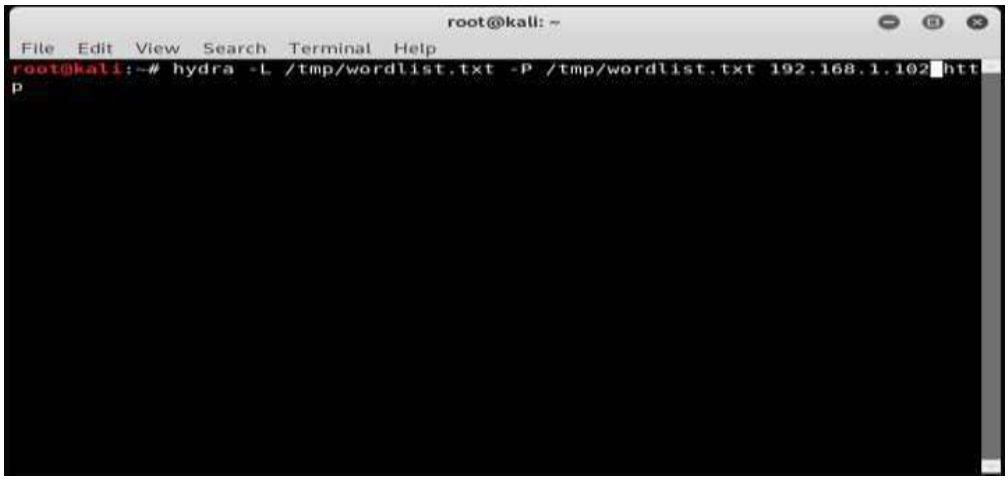
Figura 78. Ataque de negação de serviço (parte 2).



Fonte: Autor (2019).

Na Figura 79 é demonstrada no terminal de comando do Kali Linux a sintaxe para a simulação de ataque de força bruta tipo dicionário á nuvem privada OpenStack.

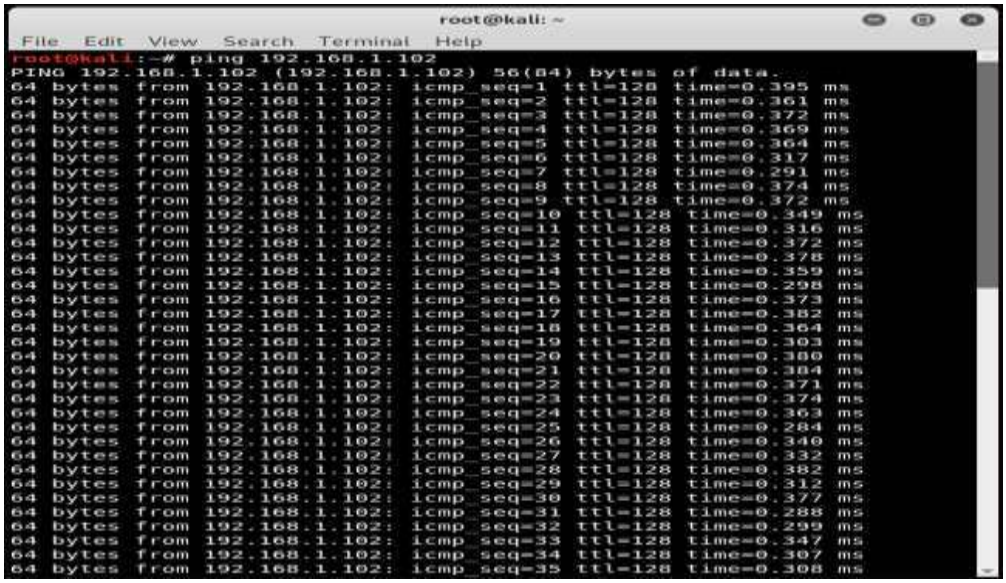
Figura 79. Comando ataque de força bruta tipo dicionário.



Fonte: Autor (2019).

Na Figura 80 é demonstrada teste de comunicação com a nuvem privada OpenStack, onde a simulação de ataque de força bruta se iniciou no pacote de transmissão número 21 e finaliza no pacote 40 que é demonstrado na Figura 81.

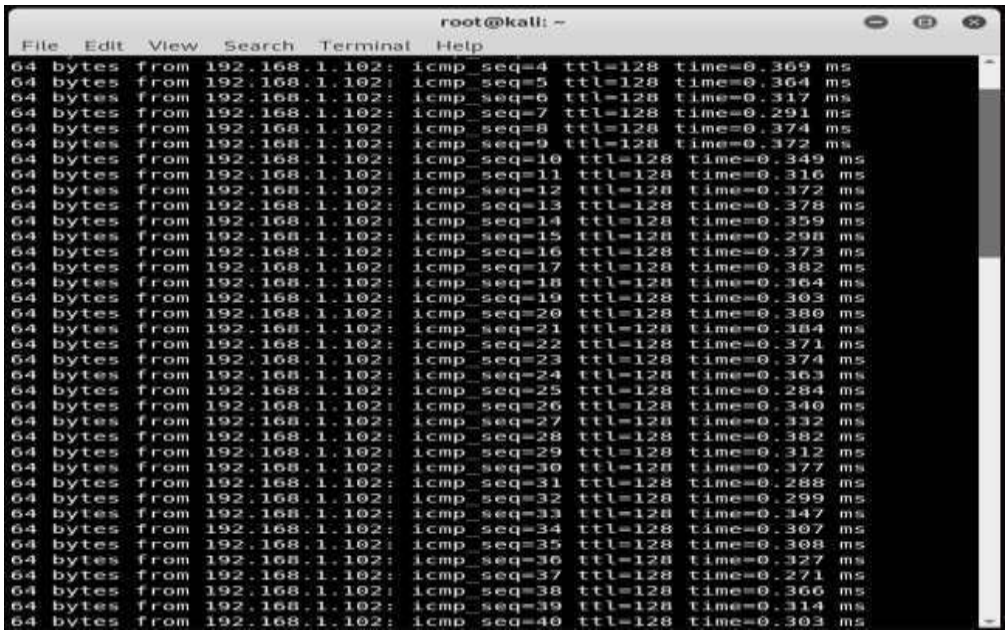
Figura 80. Ataque de força bruta (parte 1).



Fonte: Autor (2019).

Na figura 81 são demonstrados os pacotes de transmissão até o número 40.

Figura 81. Ataque de força bruta (parte 2).



Fonte: Autor (2019).