



UNIVERSIDADE DA BEIRA INTERIOR  
Engenharia

# Análise de Desempenho da Migração de Instâncias em OpenStack (Versão Final Após a Defesa Pública)

Luzimira de Abreu Ribeiro

Dissertação para obtenção do Grau de Mestre em  
Engenharia Informática  
(2º ciclo de estudos)

Orientador: Prof. Drº. Mário Marques Freire

Covilhã, Março de 2019



Dissertação elaborada no Instituto de Telecomunicações - Delegação da Covilhã e no Departamento de Informática da Universidade da Beira Interior por Luzimira de Abreu Ribeiro, Licenciada em Ciências da Computação pela Ivan Franko National University of Lviv (Ucrânia), sob orientação do Doutor Mário Marques Freire, Investigador Sénior do Instituto de Telecomunicações e Professor Catedrático do Departamento de Informática da Universidade da Beira Interior, e submetida à Universidade da Beira Interior para discussão em provas públicas.

Trabalho realizado no âmbito do Programa Integrado de IC&DT “C4 - Centro de Competências em Cloud Computing” com contrato CENTRO-01-0145-FEDER-000019, cofinanciado pelo Sistema de Apoio à Investigação Científica e Tecnológica - Programas Integrados de IC&DT, pelo Programa Operacional Regional do Centro (Centro 2020) e pela União Europeia através do Fundo Europeu de Desenvolvimento Regional (FEDER).

Cofinanciado por:





# Agradecimentos

A elaboração de uma dissertação de mestrado, faz parte de uma trajetória com muitos desafios. Alcançar este objetivo foi possível graças à bênção e vontade de Deus.

Agradeço a minha família, por confiar e acreditar mim e pelo apoio incondicional que me foi concedido.

Agradeço ao meu mentor e orientador Professor Doutor Mário Marques Freire, pela confiança, apoio e encorajamento que permitiu a conclusão deste trabalho e uma excelente experiência de aprendizagem durante todo o processo de elaboração da dissertação.

Agradeço a Universidade da Beira Interior, ao Departamento de Informática e ao Instituto de Telecomunicações pelo apoio e incentivo.

Agradeço a todos os professores e colegas que tive durante a formação de mestrado, em particular aos meus colegas de estudo que estiveram comigo em todos os projetos. Agradeço a todos que direta e indiretamente ajudaram e contribuíram para o término do meu mestrado e conclusão desta dissertação.



# Dedicatória

À minha Família.



# Resumo

Os sistemas de computação em nuvem, atualmente estão presentes na maioria dos serviços disponibilizados pelas empresas de tecnologias de informação. Isto deve-se ao fácil acesso aos serviços de computação em nuvem bem como a facilidade dos dispositivos em aceder a estes serviços. Outro aspeto muito importante tanto para os utilizadores como para os provedores de serviços em nuvem é a migração de máquinas virtuais. A migração de máquinas virtuais permite equilibrar a carga de trabalho dos servidores e permite que os utilizadores continuem a utilizar os serviços da nuvem sem interrupções por terem os seus servidores virtuais transferidos para outros servidores físicos.

O foco deste trabalho consiste em analisar o desempenho da migração de instâncias em OpenStack, o qual é explorado na perspetiva de uma Infraestrutura como Serviço (Infraestrutura como Serviço (IaaS)), um modelo de serviço de auto atendimento que permite gerir infraestruturas de um centro de processamento de dados. O IaaS representa essencialmente hardware que permite o armazenamento e criação de servidores virtuais, conhecidos como instâncias ou máquinas virtuais.

O sistema foi desenvolvido na plataforma OpenStack. O OpenStack foi instalado manualmente com recursos ao ficheiro de nuvem do OpenStack. A infraestrutura do sistema é composta por dois servidores de computação, com suporte do hypervisor Kernel-Based Virtual Machine (KVM) que permite a criação das máquinas virtuais, um servidor de armazenamento que permite a criação de volumes para a inicialização e armazenamento das máquinas virtuais e do sistema operativo, um servidor *Controller Node* que armazena os serviços adicionais e de rede, que permite a criação de uma topologia de rede para o funcionamento adequado das máquinas virtuais.

Recorrendo ao sistema implementado, o qual é baseado na série Newton do OpenStack foi realizado um estudo de desempenho da migração passiva e da migração ativa de instâncias em OpenStack.

## Palavras Chave

Computação em Nuvem, Hypervisor, OpenStack, IaaS, Back-end, Front-end, Máquina virtual, Migração de Instâncias.



# Abstract

Cloud computing systems are currently present in most of the services provided by information technology companies. This is due to the easy access to cloud computing services as well as the easiness of devices to access these services. Another important aspect for both users and cloud service providers is the migration of virtual machines. Migrating virtual machines allows the balance of server workloads and allows users to continue to use cloud services seamlessly by having their virtual servers transferred to other physical servers.

The focus of this work is to analyze the performance of the migration of instances in OpenStack, which is explored from the perspective of an Infrastructure as a Service (IaaS), a self-service model that allows the management of infrastructures in a data center. A IaaS essentially represents hardware that allows the storage and creation of virtual servers, known as instances or virtual machines.

Cloud computing has gained virtualized computing power from storage provided through an infrastructure as a service, with high performance computing capability, provided by many providers, and used by many users.

The system was developed on the OpenStack platform. OpenStack has been manually installed using the OpenStack cloud file. The system infrastructure consists of two computing servers, with hypervisor KVM support that allows the creation of virtual machines, a storage server that allows the creation of volumes for the initialization and storage of virtual machines and the system operative, a *Controller Node* server that stores the additional and network services, which allows the creation of a network typology for the proper functioning of the virtual machines.

A performance study of passive migration and the active migration of instances in OpenStack was carried out using the implemented system, which is based on the Newton series of OpenStack.

## Keywords

Cloud Computing, Hypervisor, OpenStack, IaaS, Back-end, Front-end, Virtual Machine, Instance Migration.



# Conteúdo

1	Introdução	1
1.1	Foco da Dissertação	1
1.2	Motivação	1
1.3	Definição do Problema	1
1.4	Abordagem Adotada para a Resolução do Problema	2
1.5	Principais Contribuições	2
1.6	Limitações do Trabalho Desenvolvido	2
1.7	Organização da Dissertação	2
2	Background e Estado da Arte	5
2.1	Introdução	5
2.2	História e Trabalhos Relacionados	5
2.3	Conceitos da Computação em Nuvem	6
2.4	Modelos de Serviço e Desenvolvimento	8
2.5	Arquitetura da Computação em Nuvem	12
2.5.1	Características da Computação em Nuvem	13
2.5.2	Vantagens e Desvantagens	16
2.5.3	Sistemas de Computação em Nuvem	17
2.5.4	Tipos de Hypervisor	19
2.6	Conclusão	21
3	Sistema de Computação em Nuvem Implementado e Ferramentas Utilizadas	23
3.1	Introdução	23
3.2	Arquitetura do Sistema de Computação em Nuvem Implementado	23
3.2.1	Requisitos do Sistema	23
3.2.2	Características do Hardware do Sistema de Computação em Nuvem Implementado	24
3.2.3	Plataforma OpenStack	24
3.2.4	Servidor NTP	26
3.3	Arquitetura do Sistema de Computação em Nuvem Baseado no OpenStack Newton	28
3.3.1	Configuração do Controller Node.	28
3.3.2	Network Time Protocol (NTP)	28

3.3.3	Identity Service (Keystone)	29
3.3.4	Image Service (Glance)	29
3.3.5	Compute Service (Nova)	30
3.3.6	Network Service (Neutron)	31
3.3.7	Block Storage (Cinder)	31
3.3.8	Dashboard (Horizon)	31
3.4	Configuração do Compute Node1 e Node2	32
3.4.1	Network Time Protocol (NTP)	32
3.4.2	Instalação e Configuração do Hypervisor KVM	33
3.4.3	Nova-Compute (nova)	33
3.4.4	Network Agent	34
3.5	Configuração do Block-Storage Node	34
3.5.1	Network Time Protocol (NTP)	34
3.5.2	Volume-Service (Cinder)	34
3.6	Conclusão	35
4	Resultados	37
4.1	Introdução	37
4.2	Visão Geral do Sistema	37
4.3	Benchmark nos Servidores Físicos	41
4.3.1	Bechmark de CPU	42
4.3.2	Bechmark de Memória	42
4.4	Migração de Instâncias	43
4.4.1	Migração Passiva e Ativa	44
4.4.2	Comparação entre a Migração Passiva e a Migração Ativa	46
5	Conclusões e Trabalhos Futuros	49
5.1	Principais Conclusões	49
5.2	Trabalhos Futuros	49
	Referências	51

# Lista de Figuras

2.1	Diferença entre Sistema Tradicional e Sistema em Nuvem (Adaptado de [1]). . .	7
2.2	Serviços da computação em nuvem (Adaptado de [2]). . . . .	9
2.3	Modelos de Desenvolvimento (Autor) . . . . .	11
2.4	Arquitetura da Computação em Nuvem (Adaptado de [3]). . . . .	12
2.5	Visão geral da computação em nuvem (Adaptado de [4]). . . . .	14
2.6	Tipos de Hypervisor (Adaptado de [5]). . . . .	19
3.1	Representação esquemática da arquitetura do sistema de computação em nuvem baseado no OpenStack Newton . . . . .	25
3.2	Arquitetura Conceptual do OpenStack (Adaptado de [6]) . . . . .	25
3.3	Sincronização do Servidor NTP com os outros Servidores. . . . .	27
3.4	Verificação da sincronização do Servidor NTP. . . . .	28
3.5	Verificação da sincronização do Controller com o servidor NTP. . . . .	29
3.6	Interação dos serviços em OpenStack. . . . .	29
3.7	Lista de imagens dispiníveis no Sistema. . . . .	30
3.8	Serviços Compute do Controller Node. . . . .	30
3.9	Representação esquemática da interação do compute com outros serviços. . . .	30
3.10	Agentes Neutron no Controller Node. . . . .	31
3.11	Serviço Cinder-Scheduler. . . . .	31
3.12	Verificação do Acesso ao Dashboard. . . . .	32
3.13	Lista de Imagens . . . . .	32
3.14	Verificação da instalação do Servidor KVM nos servidores Compute1 e Compute2. .	33
3.15	Serviços de Computação dos Servidores Compute1 e Compute2. . . . .	34
3.16	Agentes de Rede dos Servidores Controller Node, Compute1 e Compute2. . . . .	34
3.17	Verificação do Serviço NTP do Block-Storage Node. . . . .	34
3.18	Verificação do serviço de armazenamento do servidor Block-Storage. . . . .	35
4.1	Zona de disponibilidade. . . . .	37
4.2	Volumes. . . . .	38
4.3	Tipos de flavor. . . . .	38
4.4	Instâncias criadas no sistema de computação em nuvem. . . . .	39
4.5	Topologia de rede. . . . .	40

4.6	Utilização do Hypervisor. . . . .	41
4.7	Utilização do sistema. . . . .	41
4.8	Benchmark de memória dos servidores físicos . . . . .	43
4.9	Migração . . . . .	44
4.10	Detalhes Instâncias Passivas . . . . .	44
4.11	Resultado da Migração Passiva (part1) . . . . .	45
4.12	Resultado da Migração Passiva (part2) . . . . .	45
4.13	Detalhes da Migração Ativa (part1) . . . . .	45
4.14	Detalhes da Migração Ativa (part2) . . . . .	45
4.15	Instâncias Ativas e Passivas . . . . .	46
4.16	Detalhes da Comparação da Migração Ativa e Passiva (part1) . . . . .	46
4.17	Detalhes da Comparação da Migração Ativa e Passiva (part2) . . . . .	46
4.18	Detalhes da Comparação da Migração Passiva Ativa (part1) . . . . .	47
4.19	Detalhes da Comparação da Migração Passiva Ativa (part2) . . . . .	47
4.20	Tempo de Execução da Tentativa da Migração Ativa e Passiva . . . . .	48

# Lista de Tabelas

2.1	Fornecedores de Serviços (Adaptado de [3], [7], [6]) . . . . .	7
2.2	Compatibilidade do Hypervisor com o Sistema de Computação em Nuvem. . . . .	21
3.1	Características do hardware do sistema de computação em nuvem implementado	24
4.1	Resultados dos testes realizados com o Benchmark de CPU . . . . .	42
4.2	Detalhes de Hardware das Instâncias com mais Recursos . . . . .	44
4.3	Detalhes de Hardware das Instâncias com menos Recursos . . . . .	44



# Lista de Acrónimos

API	Application Programming Interface
CLI	Command-Line Interface
CPU	Unidade Central de Processamento
DHCP	Dynamic Host Configuration Protocol
HaaS	Hardware as a Service
IaaS	Infraestrutura como Serviço
IP	Internet Protocol
KVM	Kernel-Based Virtual Machine
LTS	Long Term Support
NAT	Network Address Translation
NTP	Network Time Protocol
NFS	Network File System
NIST	National Institute of Standards and Technology
PSN	Parceiro do Serviço em Nuvem
PaaS	Plataforma como Serviço
PDA	Personal Digital Assistant
RAM	Random Access Memory
SaaS	Software como Serviço
SQL	Structured Query Language
TI	Tecnologias de Informação
TCP/IP	Transmission Control Protocol/Internet Protocol
USN	Utilizador do Serviço em Nuvem
VMM	Virtual Machine Manager
VM	Virtual Machine
VPN	virtual private network
WEB	World Wide Web
XaaS	Qualquer coisa como Serviço



# Capítulo 1

## Introdução

### 1.1 Foco da Dissertação

A plataforma OpenStack disponibiliza através do serviço dashboard Front-End e da Command-Line Interface (CLI) Back-End um conjunto de recursos de computação, armazenamento e rede em centros de dados tanto para fornecimento público de serviços na nuvem como para sistemas privados de computação em nuvem.

Com o crescimento da utilização dos sistemas de computação em nuvem e o aumento das tecnologias de virtualização nos centros de dados, surgiu a necessidade de serem implementados sistemas com maior escalabilidade e flexibilidade, para suportar carga de trabalho, e possibilitar a migração de servidores virtuais, com recurso à virtualização. A plataforma OpenStack fornece recursos que permitem o desenvolvimento de uma infraestrutura como serviço, para a criação e migração de máquinas virtuais.

A migração de máquinas virtuais tem sido um recurso muito utilizado para o balanceamento da carga de trabalho dos servidores de computação. Neste trabalho foi desenvolvido um sistema de computação em nuvem, com recursos a alguns serviços do OpenStack. O sistema permite a criação de redes virtuais, a criação de máquinas virtuais, e o recurso de migração de máquinas virtuais entre os servidores de computação.

### 1.2 Motivação

Face ao elevado interesse por serviços em nuvem do tipo Infraestrutura como um Serviço (IaaS), têm havido grande interesse em analisar o desempenho da migração de máquina virtuais, nomeadamente ao nível de plataformas de gestão de infraestruturas virtualizadas. Nesta dissertação, pretende-se avaliar o desempenho da migração de máquinas virtuais em sistemas em nuvem, capazes de oferecer não só serviços do tipo Infraestrutura como um Serviço, mas também serviços do tipo Plataforma como um Serviço (Plataforma como Serviço (PaaS)) e software como um Serviço (Software como Serviço (SaaS)). Para tal, foi implementado um sistema baseado na série Newton do sistema OpenStack, tendo sido analisada a migração de máquinas virtuais, que, no jargão do OpenStack, são designadas por instâncias.

### 1.3 Definição do Problema

O problema abordado nesta dissertação consiste em explorar os recursos oferecidos pelo sistema de computação em nuvem OpenStack, com a avaliação de desempenho do tempo da migração de instâncias entre os servidores de computação.

## 1.4 Abordagem Adotada para a Resolução do Problema

A estratégia para a resolução do problema consiste numa abordagem experimental, que envolve desenvolvimento de um serviço do tipo IaaS e análise de desempenho da migração das máquinas virtuais criadas no IaaS, no laboratório do grupo de investigação, com a utilização de duas máquinas físicas para servidor de computação que permite criar as máquinas virtuais, uma máquina física para servidor de armazenamento para as máquinas virtuais, uma máquina física para servidor controlador do sistema, uma máquina física para servidor Network Time Protocol (NTP).

## 1.5 Principais Contribuições

A principal contribuição desta dissertação consiste em avaliar o desempenho do tempo de migração das instâncias, numa infraestrutura como serviço, implementada na versão Newton da plataforma OpenStack.

## 1.6 Limitações do Trabalho Desenvolvido

As limitações do trabalho, estão baseadas na quantidade de máquinas físicas disponibilizadas para o desenvolvimento do sistema de computação em nuvem. A quantidade de máquinas disponíveis limitou o processo de criação e migração das máquinas virtuais. Os requisitos de hardware dos servidores físicos de computação limitou a migração ativa das instâncias com maiores capacidades de hardware. Apesar de terem sido disponibilizadas cinco máquinas físicas, apenas duas máquinas funcionaram como servidor de computação. Foram criadas três máquinas virtuais, com três diferentes tipos de requisitos de hardware, que permitiu a migração passiva e ativa das máquinas virtuais entre os servidores Compute1 e Compute2.

## 1.7 Organização da Dissertação

Para melhor compreensão, a dissertação está estruturada e organizada da seguinte forma:

1. O primeiro capítulo - **Introdução** - aborda assuntos relacionados com o foco da dissertação, descreve a motivação, definição do problema e objetivos, abordagem adotada para a resolução do problema, as principais contribuições, as limitações do trabalho desenvolvido e a organização da dissertação;
2. O segundo capítulo - **Estado da Arte** - descreve os conceitos da computação em nuvem, modelos de serviço, modelos de desenvolvimento, os hypervisors, virtualização e sistemas de computação em nuvem;
3. O terceiro capítulo - **Tecnologias e Ferramentas Utilizadas** - descreve de forma detalhada as ferramentas utilizadas para o desenvolvimento deste projeto, também é feita a instalação do hypervisor nativo KVM. Instalação e configuração do OpenStack sobre o KVM;

4. O quarto capítulo - **Implementação e Testes** - descreve o funcionamento geral do sistema desenvolvido, os recursos disponíveis, e a migração das instâncias;
5. O quinto capítulo - **Conclusões e trabalho Futuro** - conclusões retiradas do trabalho realizado durante o ano letivo, respectivamente a nível técnico, de desenvolvimento, e de competências adquiridas durante o desenvolvimento e implementação desta dissertação.



# Capítulo 2

## Background e Estado da Arte

### 2.1 Introdução

Este capítulo está organizado em 6 Secções. Tem o propósito de apresentar o histórico e uma visão geral que envolve o ecossistema da computação em nuvem. A secção 4.1 é dedicada à introdução e apresenta o capítulo. A Secção 2.2 aborda a história e dos trabalhos relacionados ao tema, aos objetivos e contribuições. A Secção 2.3 descreve os conceitos, objetivos e os respetivos utilizadores da computação em nuvem. A Secção 2.4 apresenta os modelos de serviço, suas camadas de desenvolvimento, e o seu enquadramento e relação existente entre os modelos de serviço, e os modelos de desenvolvimento. A Secção 2.5 descreve a arquitetura e características da computação em nuvem, incluindo as vantagens e desvantagens, sistemas de computação em nuvem e apresenta os tipos hypervisor que podem ser utilizados para o desenvolvimento de um sistema em nuvem. A Secção 3.6 apresenta a conclusão do capítulo bem com algumas considerações relacionadas ao estado da arte da computação em nuvem.

### 2.2 História e Trabalhos Relacionados

Na década de 1960, conforme mencionado em [8] J.C.R. Licklider criou o termo "Intergalactic Computer Network". Este conceito serviu para introduzir o conceito que o mundo veio a conhecer como a Internet. Inicialmente era uma interconexão global entre computador, programas e dados. O termo "cloud" origina das telecomunicações no mundo durante os anos 90, quando os provedores começaram a utilizar o serviço privado virtual de rede (VPN) para comunicação de dados. Segundo Sahu e Pateriya [1] em 1960 John McCarthy introduziu um conceito de desenvolvimento da computação em nuvem. A computação pode ser organizada e utilizada como um pequeno programa utilizado para ampliar o ambiente de trabalho e a eficiência de um sistema. Em 1966 Douglas Parkhill investigou as características da computação em nuvem, publicado no seu livro, "The Challenge of the Computer Utility". Em [9] Tung-Hui Hu descreve os principais momentos da nuvem e de computadores de partilha de tempo, para bunkers da Guerra Fria, que em seguida foram utilizados como centro de processamento de dados.

A Openstack Foundation, [10] define o OpenStack como uma plataforma de código aberto para desenvolvimento IaaS, PaaS e SaaS e em [11] afirma que esta plataforma é composta por três projetos principais, que são: Compute, Block Storage e Image Service. Os recursos da infraestrutura desenvolvida, incluem: Processamento, Armazenamento e Rede que podem ser utilizados para executar plataformas, e serviços de software. Em [12] são apresentados alguns estudos feitos nas estruturas de processamento de dados do projeto OpenStack, e descreve que é possível verificar que a computação em nuvem é uma tecnologia com enorme potencial, uma vez que existe um número indeterminado de utilizadores a transferir informações, que são monitorizadas num centro de processamento de dados de telecomunicações com capacidade para

armazenar um número indeterminado de *Petabytes*.

Nos trabalhos relacionados com o tema desta dissertação, em [13] são apresentadas soluções para migração ativa de máquinas virtuais e conjunto de dados num centro de dados, com 8300 pré-cópia para diferentes tipos de migrações. Em [14] a migração de máquinas virtuais é feita em nuvens heterogêneas da plataforma Openstack da plataforma Openstack para o Vmware. Este tipo de migração é definido como um processo de transferência de processos e cargas de trabalho entre provedores de nuvem que podem diferir em arquiteturas, formatos de imagem e disco e hipervisors. Em [15] é apresentada uma análise da migração ativa na plataforma OpenStack utilizando a rede ótica de alta velocidade.

## 2.3 Conceitos da Computação em Nuvem

Atualmente é comum aceder os conteúdos na Internet de forma independente sem que os utilizadores tenham conhecimento sobre a infraestrutura de hospedagem [16]. Esta infraestrutura é composta por centros de dados que são monitorizados e mantidos em funcionamento 24 horas por dia, pelos provedores de serviços. A computação em nuvem é uma tecnologia que permite o acesso a conteúdos online. A computação em nuvem, conforme mencionado em [2], consiste em recursos de computação configuráveis [17] para descrever um sistema que envolve a entrega de serviços hospedados pela Internet. Disponibiliza software e informações que são fornecidos aos computadores e outros dispositivos sob demanda [18]. A computação em nuvem, [19] permite o acesso conveniente a uma rede compartilhada, com um grupo de recursos computacionais configuráveis como **redes**, **servidores**, **armazenamento**, **aplicações** e **serviços** que podem ser provisionados e lançados para interagir com o provedor de serviços. Conforme ilustra a Figura 2.1, já um sistema normal é composto por hardware, sistema operativo e aplicações, já um sistema em nuvem, é composto por hardware, hypervisor, máquinas virtuais, sistema operativo convidado, aplicações e os serviços disponibilizados em nuvem.

A computação em nuvem é composta por transferência, armazenamento e o processamento de informações sobre a infraestrutura de provedores [20] tem sido atualmente uma tecnologia completa, que, com o desenvolvimento da computação paralela, computação distribuída, computação em grade, é a combinação e evolução da virtualização, computação de utilidade, com os serviços de Software como Serviço (SaaS), Infraestrutura como Serviço (IaaS), Plataforma como Serviço (PaaS) e Qualquer Coisa como Serviço (XaaS). A tabela 2.1 lista alguns provedores de serviços em nuvem, que atualmente são os mais utilizados [21]. É possível gerir e agendar sem [18] alterar recursos de computação, conectados por uma rede e constituindo uma rede de recursos computacionais que fornece ao utilizador, um serviço de acordo com suas necessidades.

As redes que fornecem recursos de computação em nuvem, são chamadas de (nuvem). Os recursos na nuvem podem ser estendidos de forma infindável e podem ser avaliados a qualquer momento, utilizado de acordo com a necessidade do utilizador, e pago de acordo com a utilização. Após a entrega do serviço de computação [22] em nuvem, este é utilizado sob demanda a qualquer momento, através de qualquer rede de acesso, com a utilização de qualquer dispositivo conectado à Internet, que utiliza tecnologias de computação em nuvem. O ecossistema da nuvem é um ecossistema comercial de organizações e pessoas singulares que interagem com algumas funções do ecossistema da nuvem que fornece e consome serviços na nuvem. Os seguintes elementos são identificados em um ecossistema de nuvem:

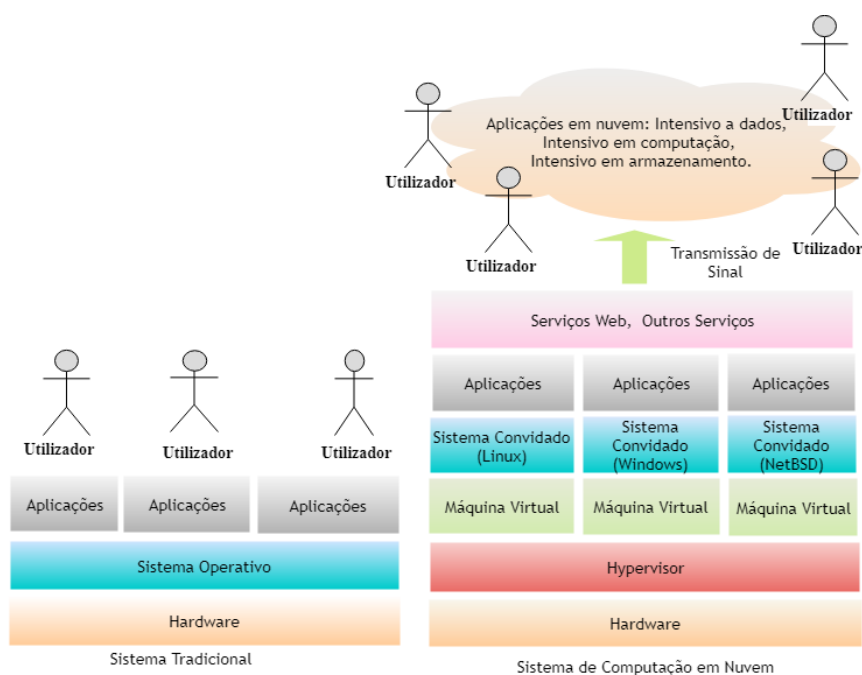


Figura 2.1: Diferença entre Sistema Tradicional e Sistema em Nuvem (Adaptado de [1]).

Tabela 2.1: Fornecedores de Serviços (Adaptado de [3], [7], [6])

N.	Serviços	Provedores
1	Software como Serviço (SaaS)	Google Docs, MobileMe, Zoho, Office 365, QuickBooks online, Salesforce.com, Yahoo Mail, Hotmail, Net Suit, Lotus, WebFiling.
2	Plataforma como Serviço (PaaS)	Microsoft Azure, Google App Engine, Force.com, VMware, Microsoft, Azure, CloudFoundry.com, Google App Engine, NYSE Capital, LongJump, Rollbase.
3	Infraestrutura como Serviço (IaaS)	Amazon EC2, Rakspace, NYSE Capital, Google Cloud Storage, Joyent.
4	Qualquer coisa como Serviço (XaaS)	Networking as a Service, Security as a Service, Desktop as a Service, etc.

- Utilizadores do serviço em nuvem (USN) são pessoas ou organizações que utilizam serviços da nuvem, disponibilizados pelo provedor de serviços (Utilizador, Empresa (incluindo o administrador da empresa, Instituição governamental/pública));
- Provedores de serviços em nuvem (PSN) (uma organização que fornece serviços de SaaS, PaaS, IaaS, XaaS e mantém os serviços de nuvem).
- Colaboradores de serviços na nuvem (CSN), uma pessoa ou organização que presta apoio à construção de serviços de um provedor de serviços na nuvem (Programador de aplicações, Provedor de conteúdo, Provedor de software, Provedor de hardware, Provedor de equipamento, Integrador de sistemas, Auditor).

A gama de serviços apresentada em [23] em nuvem é muito ampla, portanto, são esperados um grande número de requisitos para o desenvolvimento de uma infraestrutura e serviços em nuvem:

1. **Suporte de Sistemas Heterogêneos:** as soluções de gestão de nuvem, devem oferecer suporte à infraestrutura num centro de processamento de dados;
2. **Gestão de Serviços:** é importante para os administradores, possuir ferramentas para a definição e medição para as ofertas de serviços;
3. **Carga de Trabalho Dinâmica e Gestão de Recursos:** a nuvem oferece serviços sob-demanda, com uma capacidade de suportar aos SLAs(Service Level Agreements, contratos de nível de serviço, carga de trabalho e recursos, aumentar o nível de abstração e virtualizar os componentes de um centro de processamento de dados;
4. **Confiabilidade, Disponibilidade e Segurança:** o modelo e a infraestrutura de como os serviços podem ser confiáveis, podem estar disponíveis, e a nuvem continuar as suas operações, enquanto os dados permanecem intactos e protegidos num centro de processamento de dados;
5. **Integração com as Ferramentas de Gestão do Centro de Processamento de Dados:** alguns componentes de gestão, exigem um nível de integração com soluções de gestão de nuvem;
6. **Visibilidade e Relatórios:** a gestão de serviços em nuvem sem visibilidade e mecanismos de relatório, dificulta os níveis de gestão de atendimento aos utilizadores e o desempenho do sistema;
7. **Administrador, Programador e Interfaces do Utilizador Final:** auto-atendimento e modelos de implantação são atributos dos serviços baseados em nuvem que protegem a complexidade do serviço para o utilizador final.

## 2.4 Modelos de Serviço e Desenvolvimento

Os modelos de serviço da computação em nuvem [17] de acordo com o National Institute of Standards and Technology (NIST) permitem o acesso a redes, e servidores de armazenamento, que podem ser provisionados pelo provedor de serviço. Geralmente consistem em quatro diferentes características essenciais que fornecem quatro tipos de serviços genéricos, nomeadamente infraestrutura como serviço, plataforma como serviço [6] software como serviço e qualquer coisa como um serviço. São serviços disponibilizados pela Internet, que podem ser acedidos através de vários dispositivos conforme ilustra a figura 2.2.

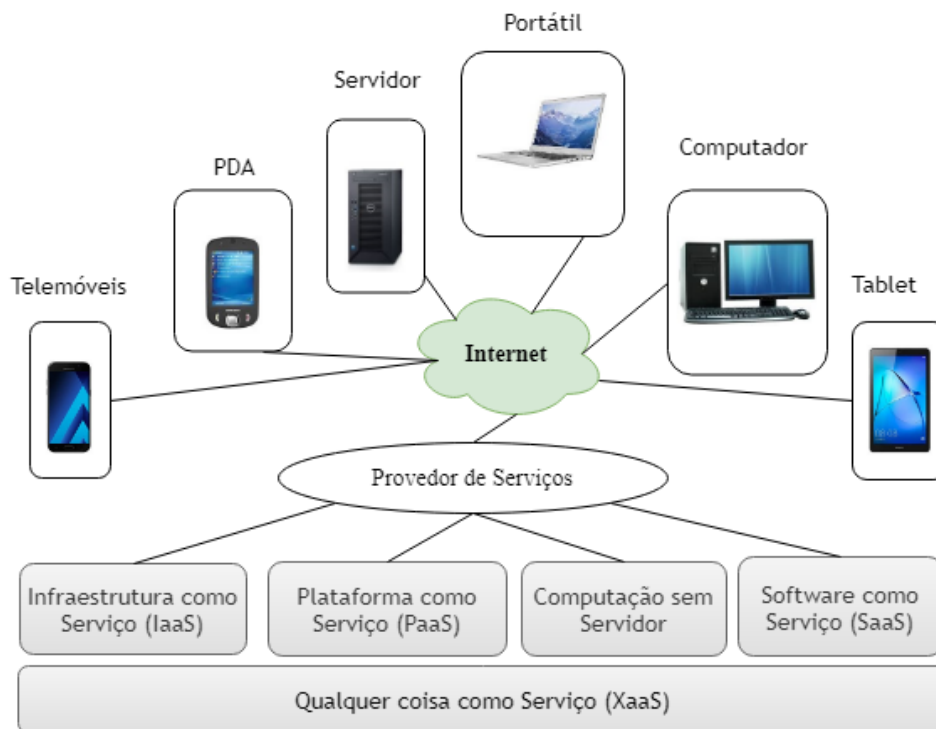


Figura 2.2: Serviços da computação em nuvem (Adaptado de [2]).

- Software como Serviço (SaaS)

O software como serviço (SaaS) é uma aplicação em nuvem [24] que fornece Software como serviço (SaaS) sobre a Internet. Anula a necessidade de instalar e executar uma aplicação no sistema normal de utilizadores. Algumas características importantes de um SaaS são: **acesso baseado em rede** e **gestão de software** comercialmente disponível que são localizações centralizadas que aos utilizadores aceder estas aplicações de forma remota através da Internet [25] que visa substituir as aplicações instaladas e a funcionar no computador dos utilizadores. Não há necessidade de instalar e executar o software no computador se o utilizador utilizar o SaaS. O utilizador ao invés de comprar o software a um preço alto, ele apenas paga por utilização dos recursos, o que pode reduzir os custos significativamente. SaaS aplicações [17] construídas e integradas verticalmente, (como por exemplo um Sistema de email, Gerência de recursos Humanos, Folha de Pagamento, Processamento, Processamento de Base de Dados e outras Aplicações e Processos) são entregues e comprados por utilizadores como Serviços. Existem vários provedores SaaS, conhecidos como [24] provedores de serviços de aplicações. Estes provedores executam uma única aplicação em um centro de processamento de dados e entregam a funcionalidade dos serviços através da Internet para os utilizadores. É importante para a computação [26] em nuvem, garantir que as soluções fornecidas como SaaS, de fato, cumpram as definições geralmente aceites. Sites de portal da Internet, motores [27] de busca na Internet e sites de redes sociais na Internet são essencialmente fornecedores SaaS para utilizadores finais.

- Plataforma como Serviço (PaaS)

A plataforma como serviço [17] é uma camada da computação que disponibiliza ferramentas para desenvolvimento de software e produto como servidores de aplicações, servidores de base

de dados, servidores de portal, middleware. A PaaS oferece os meios [17] para que os Programadores possam desenvolver, executar e administrar aplicações. Os serviços nesta camada, são destinados a suportar a camada superior de serviços de software. Os programadores procuram reduzir tempo e custos na implantação de suas aplicações. Ofertas típicas incluem ambiente de tempo de execução para o código da aplicação, serviços de computação em nuvem, computação de energia, armazenamento e rede. Este nível de serviços pode ser considerado como, uma camada de nível de programador.

A estrutura de preços é frequentemente em torno de: computação de uso por hora; transferência de dados por GB; Input e Output pedidos por utilizadores; armazenamento por GB; pedidos de armazenamento de dados. Todas as cobranças são por cada período de utilização. PaaS fornece uma plataforma [24] de computação utilizando a infraestrutura da nuvem. Tem todas as aplicações normalmente exigidas pelo programador. Portanto o programador não precisa da contrariedade de comprar e instalar software e hardware necessários. A capacidade fornecida ao programador é implantar [19] na nuvem serviços construídos por infraestrutura ou aplicações construídas, com a utilização linguagens de programação e ferramentas suportadas pelo provedor.

O programador não gere ou controla a infraestrutura da nuvem subjacente, incluindo rede, servidores, sistemas operativos ou armazenamento, mas tem controle sobre as aplicações implantadas e, possivelmente, configurações de ambiente de hospedagem para aplicações. A plataforma como serviço (PaaS) oferece [28] serviços de nível superior aos programadores. Portanto, os programadores podem criar aplicações sem instalar ferramentas em seu computador e implantar estas aplicações, sem responsabilidades com as tarefas de administração do sistema. Os programadores de serviços podem [24] desenvolver todos os sistemas e ambientes necessários para o ciclo de vida do software, seja ele em estado de desenvolvimento, teste, implantação e hospedagem de aplicações web. Com a PaaS, o sistema [29] operativo pode ser alterado e atualizado sempre que for necessário, isto permite que equipas geograficamente distribuídas trabalhem juntas em projetos de desenvolvimento de software. O provedor de [30] serviços é responsável por garantir a disponibilidade de servidores, redes, armazenamento e outros serviços de suporte necessários para hospedar a aplicação para o utilizador final.

- Infraestrutura como Serviço (IaaS)

Esta camada da computação em nuvem [17] é essencialmente hardware, pois disponibiliza servidores virtualizados, armazenamento, dispositivos de rede e serviços de hardware, para permitir que as plataformas e aplicações em nuvem funcionem. Estes serviços suportam a camada superior de, plataforma como serviço e software como serviço [19] são fornecidas algumas capacidades ao programador, que são processamento de provisão, armazenamento, redes e outra computação fundamental e recursos onde o programador pode implantar e executar software arbitrário, o que pode incluir sistemas operativos e aplicações. O programador não gere ou controla a infraestrutura da nuvem subjacente, mas tem controle sobre o funcionamento sistemas, armazenamento, aplicações implantadas e, possivelmente, controle limitado de componentes de rede como firewall de hospedagem [25], entrega de grandes recursos computacionais, como a capacidade de processamento, armazenamento e rede. A IaaS também pode ser chamada de Hardware como Serviço (HaaS). A estrutura de preços geralmente é [17] similar à provisão para PaaS. A computação em nuvem [31] foi inspirada pelo símbolo da nuvem que é utilizada frequentemente para representar a Internet em fluxogramas e diagramas.

- Qualquer Coisa como Serviço (XaaS)

Tem referência à entrega de qualquer coisa como [32] serviço, com base nos serviços existentes, relativamente com muitos ou poucos recursos.

- Modelos de Desenvolvimento

Os modelos de desenvolvimento [33] podem ser implementados para corresponder necessidades variadas de utilizadores, programadores ou provedores. Os modelos de desenvolvimento, público, privado híbrido e comunitário, variam da localização do hardware, no caso, qual entidade é responsável pela conservação do sistema e quais entidades podem utilizar os recursos do sistema. Com a evolução do desenvolvimento e a utilização da nuvem, [22] a variedade de tipos de modelos e funcionalidades disponíveis através da nuvem também evoluiu. Atualmente, os modelos existentes são considerados principais de implantação em nuvem como aplicações, plataforma e infraestrutura para organizações que adquirem e utilizam pela Internet conforme ilustra a Figura 2.3.

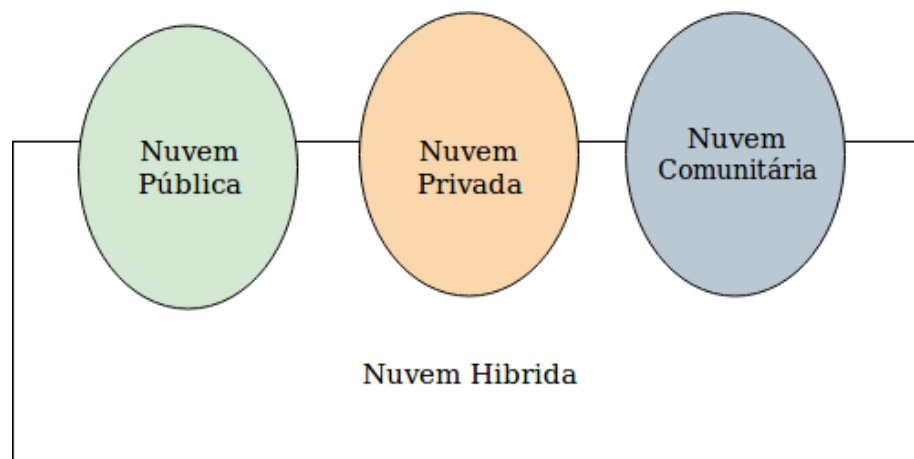


Figura 2.3: Modelos de Desenvolvimento (Autor)

A nuvem pública, é baseada na [34] computação em nuvem no sentido convencional, pelo qual os recursos são provisionados dinamicamente em uma base de auto financiamento, através da Internet, de aplicações Web ou serviços Web de um provedor externo em uma base computacional de utilidade. Esta nuvem pública, [29] utiliza o modelo padrão de computação em nuvem e disponibiliza recursos para os utilizadores através da Internet. Os serviços de nuvem pública podem ser oferecidos em termos de pagamento por utilização ou podem ser grátis. A infraestrutura em nuvem [35] é disponibilizada aos utilizadores em geral ou a uma entidade. A nuvem privada, [34] é um modelo particular de computação em nuvem que envolve um ambiente fundamentado em nuvem diferente e seguro onde apenas um utilizador específico tem total acesso. Tal como acontece com os outros modelos, as nuvens privadas fornecem uma capacidade de computação como serviço dentro de um ambiente virtualizado que utiliza recursos computação física. A infraestrutura de nuvem, [35] é operada exclusivamente para uma empresa.

Uma nuvem de armazenamento híbrida [34] é composta por uma combinação por duas ou mais nuvens de armazenamento. Estas nuvens são muitas vezes necessárias para guardar e fazer cópia de segurança de funções, permitindo que os dados locais sejam reduzido a porções pequenas para uma nuvem pública. A nuvem híbrida [35] tem como função, representar duas ou

mais nuvens separadas, a tornarem-se juntas. Estas nuvens são as **Nuvens Públicas, Privadas, Comunitárias**, ou uma combinação de nuvem virtualizada, instâncias de servidor utilizadas em conjunto com hardware físico real, que permanecem como entidades únicas, mas unidas por uma tecnologia padronizada ou proprietária, que permite a portabilidade de dados e aplicativos como estouro de nuvem para balanceamento de carga entre as nuvens. Na nuvem Híbrida é considerada a utilização de hardware físico e instâncias virtuais do servidor em nuvem em conjunto [34], para fornecer um único serviço comum. A nuvem híbrida também tem [36] um custo e composição semelhante à nuvem pública, segurança e controle semelhante à nuvem privada. Uma nuvem comunitária pode ser estabelecida onde várias empresas têm requisitos semelhantes e procuram partilhar a infraestrutura para realizar alguns dos benefícios de computação em nuvem [35].

## 2.5 Arquitetura da Computação em Nuvem

A plataforma nuvem é composta por servidores físicos ou virtuais, sistemas de armazenamento, centros de processamento de dados, dispositivos de rede, hypervisor e middleware [4]. A arquitetura dos serviços da nuvem consiste numa abordagem escalável em tempo real, com a entrega recursos de computação como um serviço, onde os recursos partilhados, como software e informações são fornecidos a computadores e outros dispositivos como um programa utilizado para substituir falhas do sistema operativo em uma rede. A implementação da computação em nuvem é baseada [3] em tecnologia de virtualização, na qual o sistema hospedado opera sobre o hypervisor. A arquitetura é composta por dois [37] componentes, o primeiro é o **Back-end** parte (CLI) que se refere a própria nuvem, que pode conter várias máquinas de computador, sistemas de armazenamento de dados e servidores. O grupo destas nuvens compõe um sistema inteiro de computação em nuvem. O segundo é o **Front-end** (parte visualizada pelo dashboard) que compreende o dispositivo do utilizador ou uma rede de computadores e algumas aplicações necessárias para aceder ao sistema de computação em nuvem. A arquitetura da computação em nuvem é apresentada Figura 2.4.



Figura 2.4: Arquitetura da Computação em Nuvem (Adaptado de [3]).

### 2.5.1 Características da Computação em Nuvem

Na computação em nuvem, os utilizadores podem aceder os dados, aplicações ou qualquer outro serviço com a ajuda de um navegador de maneira independente do dispositivo utilizado e da localização do utilizador. A infraestrutura que geralmente é fornecida por um provedor, é acedida, através da Internet. O custo é reduzido a um nível significativo. À medida que a infraestrutura é fornecida por um provedor e não precisa ser adquirida para tarefas de computação intensiva ocasionais [24]. Como característica, o acoplamento solto é a principal característica técnica, chave de todos os tipos de sistemas de computação em nuvem [25].

A tolerância a falhas faz da computação em nuvem muito mais adaptada e amplamente adotada a estrutura de rede. Existem também outros tipos de características, tais como Transmission Control Protocol/Internet Protocol (TCP/IP) especialmente baseado na Internet, virtualização e alta segurança. As características da computação em nuvem são muito mais complexas pois existem características que podem ser utilizadas para distinguir cluster, grid e sistemas de computação em nuvem. Os recursos do Cluster estão localizados em um único domínio administrativo com entidade única. Os recursos do sistema de grade são distribuídos e localizados no domínio administrativo, com políticas multi-entidade e de gestão. A plataforma de computação em nuvem possui características dos cima mencionados. A plataforma de computação em nuvem também oferece serviços para os utilizadores sem que os mesmos tenham conhecimento sobre a infraestrutura. O principal objetivo da computação [24] em nuvem é fazer uma melhor utilização de recursos distribuídos, combinando-os para alcançar maiores qualificação de transferência e poder resolver problemas de computação em grande escala. A computação em nuvem cuida da virtualização, escalabilidade, interoperabilidade, qualidade do serviço e integra modelos de desenvolvimento, nomeadamente nuvem pública, nuvem privada, nuvem híbrida e nuvem da comunidade. A computação em nuvem é um conceito de gestão e utilização de recursos distribuídos com base em algumas [38] características essenciais [37].

Um provedor pode fornecer recursos de computação unilateralmente, como tempo de servidor e armazenamento de rede, conforme necessário automaticamente, sem necessidade de interação humana com o provedor de cada serviço. Os serviços estão disponíveis sob demanda, o utilizador pode obter os serviços a qualquer momento. Apenas será necessária uma conexão com a Internet. A nuvem é acedida remotamente pela rede, enquanto o acesso à nuvem é através da Internet; isto significa que é acessível às suas capacidades de computação, software e hardware de qualquer lugar, estas capacidades estão disponíveis através da rede e são acedidas através de mecanismos padrão que promovem o uso por plataformas de *thin ou thick client*, "como por exemplo, *smartphones*, computadores portáteis e Personal Digital Assistant (PDA)s". Os recursos computacionais do provedor são agrupados para atender a múltiplos consumidores utilizando um modelo multi-tenant, com diferentes recursos físicos e virtuais atribuídos e reatribuídos dinamicamente de acordo com as as necessidades do utilizador [19].

O utilizador geralmente não tem controle ou conhecimento sobre a localização exata dos recursos fornecidos, mas pode especificar a localização em um nível de abstração mais alto a localização exata do centro de processamento de dados. Alguns exemplos de recursos incluem armazenamento, processamento, memória, largura de banda de rede e máquinas virtuais o agrupamento de recursos em um local e recursos independentes atende um grande número de utilizadores com todos os seus diferentes dispositivos e os recursos necessários [37]. As capacidades podem ser rapidamente e elasticamente provisionadas, em alguns casos automaticamente, para dimensionar, liberar e escalar rapidamente. Para o utilizador, as capacidades

disponíveis para o provisionamento geralmente parecem ser ilimitadas e podem ser compradas em qualquer quantidade a qualquer momento [19]. Lidar com a nuvem é muito fácil, o utilizador [37] pode simplesmente reduzir ou aumentar a capacidade, e também é mais rápido do que os tipos de computação regulares. Os sistemas de nuvem controlam e otimizam automaticamente a utilização de recursos e realça um recurso de medição em algum nível de abstração apropriado para o tipo de serviço que necessitar. A utilização de recursos pode ser monitorizada, controlada e relatada. Fornece transparência tanto para o provedor como para o utilizador do serviço utilizado [19]. Uma visão geral da computação em nuvem é apresentada na Figura 2.5.

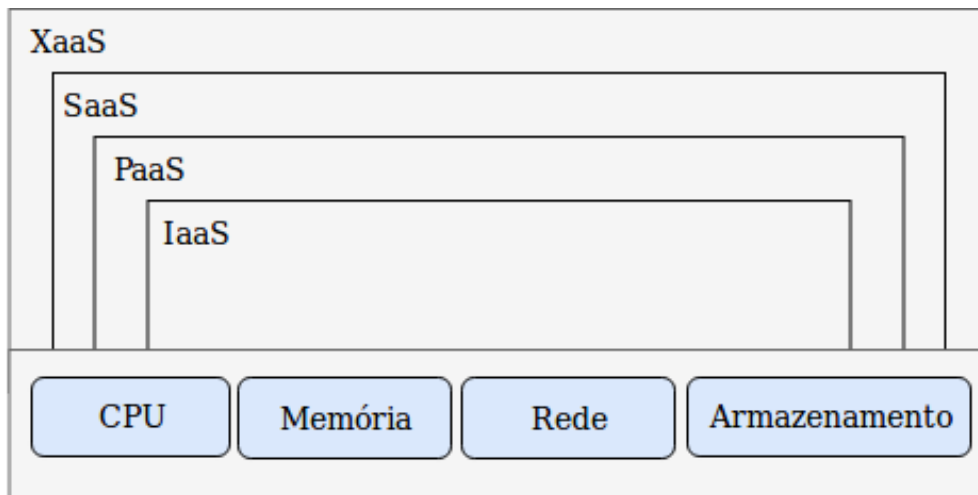


Figura 2.5: Visão geral da computação em nuvem (Adaptado de [4]).

A escalabilidade capacita um sistema para suportar cargas [39] de trabalho crescentes com um desempenho adequado, desde que os recursos de hardware sejam adicionados. No contexto de sistemas distribuídos, a escalabilidade foi definida como a mais recentemente proposta para uma metodologia de medição. Uma maneira escalável e fácil para os programadores aceder a um conjunto elevado de recursos virtualizados que podem ser provisionados de forma dinâmica para se ajustar e suportar a variação de uma carga de trabalho é a utilização da computação. A capacidade de um sistema particular, pode ser considerada para ajustar e suportar um problema. Quando o alcance deste problema aumenta podemos defini-lo como escalabilidade ou uma propriedade desejável de um serviço que oferece uma capacidade de manipulação de quantidades crescentes de cargas de serviço, sem destruir a qualidade mais importante e indispensável de atributos. O cumprimento imprevisto de cargas de trabalho, deriva do desenho do sistema, dos tipos de estruturas de dados, dos algoritmos, da comunicação, e dos mecanismos utilizados para execução dos elementos do sistema. A escalabilidade deve ser transparente para os programadores. Os programadores devem possuir a capacidade de armazenar as suas informações e dados na nuvem sem o conhecimento de onde são mantidos ou como são acedidos [40].

A escalabilidade é considerada uma das principais vantagens do paradigma da nuvem. Especificamente, a vantagem que diferencia nuvens de soluções avançadas de terceirização. Esta escalabilidade pode ser executada na nuvem por meio de diferentes níveis, pode ajudar as empresas que começam com pequenas configurações e crescer para melhores características, e se necessário redução de escala [41]. A flexibilidade da computação em nuvem permite as empresas utilizar recursos além do esperado e corresponder às expectativas dos utilizadores.

A computação em nuvem está preparada para suportar as condições de horário de utilização, configuração servidores de alta capacidade, armazenamento, e hardware. Esta facilidade ajuda as empresas a suportar qualquer tipo de requisito, independente do tamanho do projecto. As Redes *ethernet*, são redes de virtualização, sobreposição e virtualização TCP/IP. A separação dos movimentos do utilizador não são suficientes para uma escalabilidade completa da aplicação. A necessidade de escalar a rede ocorre na segurança do centro de processamento de dados que hospedam várias máquinas virtuais por máquina física [40].

A escalabilidade é muitas vezes alcançada sobre o aprovisionamento de recursos para suportar a demanda aumentada. As nuvens IaaS oferecem aos provedores de aplicações uma maneira eficaz de controlar os recursos utilizados por seus sistemas. Exigem que os programadores de aplicações ou administradores de sistema instalar e configurar todo conjunto de software que os componentes da aplicação necessitam. Em oposição, as nuvens de Plataforma como serviço (PaaS) oferecem ambientes de execução preparados para utilizar e serviços para aplicações. Entretanto, ao utilizar a nuvem PaaS, os programadores podem se concentrar em desenvolver os seus componentes e não configurar os ambientes necessários para os componentes. As nuvens PaaS podem experimentar provedores de alto padrão de utilização, devem ter capacidade para dimensionar os ambientes de execução. O grau em que um sistema tem a capacidade de adaptar-se às mudanças de carga de trabalho e à elasticidade, que permite fornecer e cancelar recursos de forma independente, para que a cada momento, os recursos disponíveis correspondam o máximo possível à procura. Na otimização dinâmica, a quantidade recursos totais adquiridos, a elasticidade é utilizada para objetivos diferentes [42].

Segundo os provedores de serviços, a elasticidade garante melhor utilização dos recursos de computação, economia de energia e permite que diferentes utilizadores, possam ser atendidos ao mesmo tempo. Segundo os provedores, a elasticidade é utilizada para evitar a provisão imprópria de recursos, a destruição do desempenho do sistema e alcançar a redução de custos. A elasticidade pode ser utilizada para outros propósitos, como o aumento da capacidade dos recursos locais. A elasticidade é a base do desempenho da nuvem e pode ser considerada uma grande vantagem e um benefício fundamental da computação em nuvem. Sobre a elasticidade existem requisitos que antecedem sua importância, e também diferenciar sua condição em relação a capacidade de expansão e eficiência. A escalabilidade de um sistema, incluindo todas as camadas de hardware, virtualização e software dentro de seus limites é um pré-requisito para elasticidade. Considerando que a elasticidade está relacionada com a capacidade de um sistema de se adaptar a mudanças nas cargas de trabalho e demanda de recursos, imagina-se a existência de pelo menos um processo de adaptação específico. Este processo é automatizado, um sentido amplo, pode conter etapas manuais. Sem um processo de adaptação definido, um sistema escalável, pode não se comportar de maneira elástica, uma vez que escalabilidade não inclui aspectos temporais. Alguns pontos importantes precisam ser verificados antes da avaliação da elasticidade [39].

Para um bom desempenho, as aplicações em nuvem precisam ter a capacidade para solicitar não apenas servidores virtuais nos vários pontos da rede, mas também canais de rede para fornecer largura de banda e outros recursos de rede para que sejam conectados de forma interna na rede como serviço. Para melhor alcance da qualidade do desempenho da nuvem, as aplicações precisam ser classificadas e projectadas de acordo com o nível de serviço, que permite as empresas de abordar preocupações, migração para uma nova tecnologia. Quando uma empresa requer um serviço, a partir de uma nuvem ou de um centro de processamento de dados, a empresa estabelece um acordo de nível de serviço que identifica métricas importantes, chamadas níveis

de serviço, que a empresa moderadamente pode esperar do serviço. A capacidade de compreender e de confiar na disponibilidade, escalabilidade e desempenho da nuvem é o requisito mais importante [40].

## 2.5.2 Vantagens e Desvantagens

A computação em nuvem oferece vantagens aos utilizadores, programadores e as empresas que adquirem os seus serviços. Disponibiliza, auto serviço padronizado, ofertas, serviços rapidamente provisionados, custos flexíveis etc. Estas vantagens suportam uma variedade de necessidades de negócios para análise e capacidades de auxílio e o desenvolvimento de novos produtos e serviços. Como resultado, mudar para um ambiente de nuvem não é mais uma informação estratégica de Tecnologias de Informação, pode ser também uma decisão de negócios, assim como as funções comerciais e de Tecnologias de Informação de algumas empresas não são mais estritamente autónomas, os componentes de um ecossistema da nuvem são, por definição, interdependentes um no outro. Um ecossistema de nuvem claramente definido e documentado, serve como uma bússola que orienta suas decisões e estratégias de suporte, onde o esforço antecipado necessário é valorizado e reconhecido [43]. A migração para a computação em nuvem pode diminuir o custo de execução e preservar os sistemas de Tecnologias de Informação [44].

Para as empresas em vez de adquirir sistemas e equipamentos de alto custo para os seus negócios, a computação em nuvem pode reduzir os custos com a utilização dos recursos de um provedor de serviços de computação em nuvem. Esta opção, pode ser capaz de diminuir os custos de trabalho pois, o custo das atualizações do sistema, hardware e software originais podem estar incluídos no contrato dos serviços adquiridos. Com a computação em nuvem, estas empresas não precisam pagar remuneração aos técnicos especializados, e também os custos de utilização de energia podem ser reduzidos, e há redução na gestão do tempo. A computação em nuvem oferece diversas vantagens, algumas delas podem variar com base no tamanho e no tipo da empresa que oferece o serviço, e no tipo de empresa que adquire estes serviços. A nuvem permite que pequenas e médias empresas e com recursos limitados, aproveitem os recursos de computação, de armazenamento e de rede líderes de mercado, redimensionados sob demanda para reconduzir o desenvolvimento da empresa [45]. Incorpora serviços de acesso à Internet, virtualização, software de código aberto, e implantação sob demanda. A computação em nuvem utiliza a Internet e servidores centrais para manter dados e aplicações que, por sua vez, permitem uma computação eficiente possui vários benefícios [29].

A otimização ajuda a otimizar uma empresa e investimentos de capitais, comprimir os seus custos de hardware, software, resultar em um custo total menor de propriedade e, em última análise, uma maneira de observar a economia de Tecnologias de Informação operacional e em grande escala; oferece simplicidade e agilidade com redução do tempo e esforço para fornecer recursos adicionais. Os custos permitem a uma empresa trocar as entidades responsáveis pelo serviço, por razões de custos de contratação, ou manter as entidades existente e, assim, habilitar as entidades importantes, para trabalhar produzindo valor e inovação para o negócio da referida empresa. A flexibilidade fornece a pequenas empresas o acesso a serviços de Tecnologias de Informação estariam indisponíveis, e assim permitir que pequenas, médias, e grandes empresas, estejam em categorias iguais para aquisição dos serviços disponibilizados.

A recuperação de dados possui mecanismos para desastres, recuperação e continuidade do negócio através de uma terceirização total do Serviço de Tecnologias de Informação e Comunicação

para uma empresa [17]. Para algumas empresas, é possível escalar ou reduzir o procedimento e a requisição de armazenamento para ajustar-se à situação. Permite flexibilidade conforme as mudanças das necessidades. Em vez de comprar e instalar atualização de alto custo em um computador individual, o provedor de serviços de computação em nuvem pode ser responsável por este trabalho [44]. A computação em nuvem pode ser firmemente escalável e existem benefícios incorporados de eficiência e alta utilização, que resulta em redução de custos e despesas. Em geral, este novo padrão, promete economias de custos, agilidade, inovação, escalabilidade e simplicidade. As ofertas continuam a crescer e as economias de custos tem sido uma boa opção para a atual situação económica [17]. A recuperação de dados, pode melhorar a disponibilidade de [21] recursos de Tecnologias de Informação e possuir vantagens sobre outras técnicas de computação considerando que, grandes empresas estão a ser [17] cativadas para as tecnologias de computação em nuvem e infraestruturas [46].

Um grande problema da nuvem é representado pela segurança. Os utilizadores armazenam os seus dados confidenciais na nuvem. É responsabilidade do provedor de serviços em nuvem estabelecer um mecanismo de segurança para os canais de comunicações, para o armazenamento, envio e receção dos dados. A segurança tem sido uma grande preocupação para provedores de serviços em nuvem [47]. Similar a outros softwares, o hypervisor também pode encontrar [36] falhas de segurança que podem comprometer a confidencialidade dos dados do utilizador. A infra-estrutura de computação em nuvem está sujeita a todas as ameaças que uma infra-estrutura padrão de computação de servidor pode estar submetida.

Os servidores da Web podem ser comprometidos com as vulnerabilidades de script entre sites; base de dados estão sujeitos a ataques de injeção de código; os kernels do sistema operativo podem ser comprometidos pela injeção de código de máquina [48]. A segurança de dados e sistemas é uma parte essencial do desenvolvimento da durabilidade dos negócios enquanto a computação em nuvem reduziu os custos e aumentou a conveniência, a acessibilidade e centralização da computação em nuvem, também cria novas oportunidades para violações de segurança [44]. A falha mais importante na computação em nuvem, é a segurança. A computação em nuvem oferece [49] segurança contra os invasores, enfrenta desafios de segurança de dados semelhante a outros sistemas de comunicação, pois as empresas e os utilizadores armazenam suas informações confidenciais na nuvem, por esta razão, tem surgido muitas preocupações [50] em relação a confidencialidade dos dados, autenticação e controlo no acesso ao sistema, utiliza a criptografia para criar códigos secretos que altera o [51] conteúdo original de uma mensagem, que são utilizados para comunicação secreta de duas entidades numa rede insegura, com a possibilidade de existir invasores entre as comunicações. Atualmente, existem muitas [52] aplicações que utilizam a criptografia para proteger dados e mensagens confidenciais. A criptografia é utilizada para garantir que os dados confidenciais ou mensagens permaneçam inalterados [33].

### 2.5.3 Sistemas de Computação em Nuvem

A nuvem, pode ser considerada como um sistema capaz de abranger um grande número de armazenamento de dados com ligações de rede que podem ser acedidas por qualquer dispositivo que utilize uma conexão padrão à Internet. Apesar de nos últimos anos ter-se confirmado que os sistemas de computação em nuvem têm ganhado mais popularidade, muitos argumentam que as tecnologias subjacentes a a estes sistemas existiram desde o aparecimento da World Wide

Web (WEB). A nuvem é muitas vezes semelhante a web que possui versatilidade que pode ser encontrada em todos os lugares, como as tecnologias de comunicação. A sincronização e armazenamento de dados entre dispositivos num sistema de nuvem, descrito como Armazenamento na nuvem, não devem ser classificados como computação em nuvem, mas o armazenamento distribuído sim, pois a computação em nuvem implica, processamento de dados paralelo ou distribuído em redes de grande escala. O OpenStack é um sistema [53] operativo de nuvem que controla grandes conjuntos de recursos de computação, armazenamento e rede em um centro de processamento de dados, todos geridos através de um painel que permite aos administradores o controle para habilitar seus utilizadores a fornecer recursos através de uma interface web. [54] Software de código aberto que pode ser instalado em mais de um servidor e serve para agregar uma camada de orquestração, (camada de gestão através de interface web, Application Programming Interface (API) ou CLI). Cada servidor será responsável por um ou mais serviços do OpenStack, já que o mesmo é totalmente modular [10].

O OpenStack proporciona uma infraestrutura em nuvem modular e executável em hardware padrão, que nos permite implantar as ferramentas necessárias, tudo a partir de um único ambiente. Eucalyptus utiliza uma infraestrutura existente para desenvolver uma camada de serviços WEB, escalável e segura que que concentre CPU! ( CPU!), rede e armazenamento para fornecer Infraestrutura como Serviço. Os serviços da WEB de Eucalyptus são projetados para nuvens híbridas. O OpenStack composto por vários componentes que interagem uns com os outros, permitindo que os programadores modifiquem módulos existentes. O Eucalyptus é uma implementação de código aberto do *Amazon EC2 (Elastic Compute Cloud)* e é compatível com a maioria das interfaces de negócios [4]. O Nimbus é um conjunto de ferramentas de código aberto estabelecido no fornecimento de capacidades de Infraestrutura como Serviço (IaaS) para a comunidade científica [4] conhecido como uma solução de computação em nuvem que fornece IaaS. O Nimbus é composto por um conjunto de software a ser instalado num nó de serviço e um software separado, instalado num número de nós, no Virtual Machine Manager (VMM). O OpenNebula é um conjunto de ferramentas utilizadas para [55] desenvolver nuvens privadas, públicas e híbridas ou comunitárias .

O OpenNebula foi desenvolvido para ser integrado com infraestruturas heterogêneas de centros de processamento de dados distribuídos, organiza tecnologias de armazenamento, rede e virtualização para implantar diferentes níveis serviços, como grupos de máquinas virtuais conectadas internamente, em infraestruturas distribuídas, combinando recursos do centro de processamento de dados e recursos remotos de computação em nuvem, de acordo com as políticas de alocação [4]. O sistema Aneka funciona como uma plataforma e estrutura para o desenvolvimento de aplicações distribuídas na nuvem, aproveita os ciclos de Unidade Central de Processamento (CPU) de reposição de uma rede heterogênea de computadores e servidores ou ou centro de processamento de dados sob demanda, fornece aos programadores um vasto conjunto de APIs para explorar de forma transparente estes recursos e expressar a lógica comercial de aplicações utilizando as abstrações de programação preferenciais. Os administradores de sistemas podem aproveitar um conjunto de ferramentas para acompanhar as informações fornecidas por instrumentos técnicos e controlar a infraestrutura implantada. Esta, pode ser uma nuvem pública disponível para qualquer utilizador através da Internet, ou uma nuvem privada formada por um conjunto de nós com acesso restrito [56].

## 2.5.4 Tipos de Hypervisor

Os hipervisors são componentes de software ou *firmware* (conjunto de instruções operativas, programadas no hardware de uma máquina) que podem virtualizar recursos do sistema [57]. É considerado uma camada fina de software que geralmente fornece particionamento virtual e capacidades que são executadas diretamente no hardware, mais abaixo dos serviços de virtualização de nível superior [31], que fazem o controle e monitoramento das suas máquinas. O modo de implementação [36] revela a forma de operação como gerentes de virtualização [58]. Existem 2 tipos principais de hypervisor, que são: hypervisor nativos e hypervisor hospedados. Para melhor compreensão, a Figura 2.6 ilustra a diferença entre o hypervisor nativo, e o hypervisor hospedado [5].

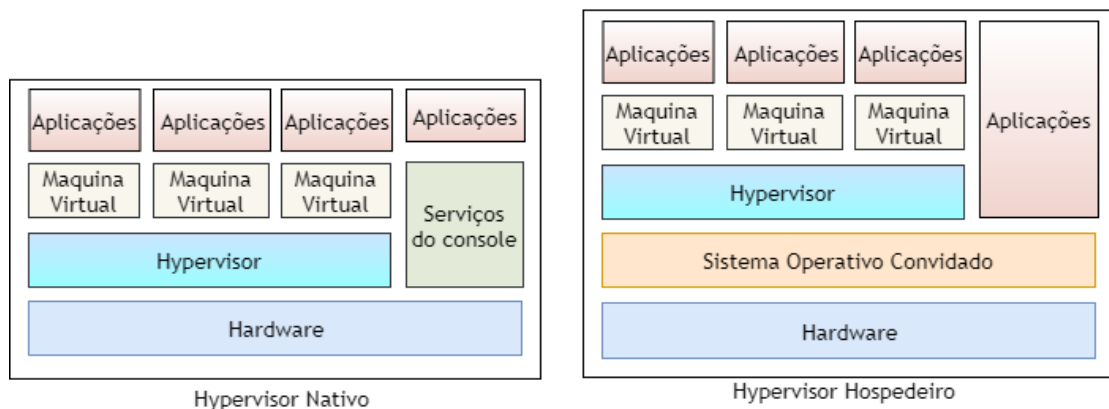


Figura 2.6: Tipos de Hypervisor (Adaptado de [5]).

Os Hypervisor nativos são executados diretamente no hardware da máquina hospedeira e partilham recursos externos como memória e dispositivos entre máquinas convidadas. Fazem o monitoramento do sistema operativo que é executado diretamente acima do hypervisor, e do sistema operativo executado na máquina convidada. Isto acontece porque o sistema operativo da máquina convidada é executado de forma isolada acima do hypervisor. Os hypervisor hospedados são executados como uma aplicação dentro de um sistema e suporte de máquinas virtuais que funcionam como processos individuais. É instalado num sistema operativo existente e hospeda outro sistema operativo. Neste tipo de hypervisor, o problema enfrentado pelo sistema operativo existente, irá afetar também o sistema operativo da máquina hospedada, executada no hypervisor e pode também afetar o bom funcionamento do próprio hypervisor. O hypervisor em execução acima do sistema operativo pode ser protegido, mas o sistema operativo convidado não seria. O sistema operativo hospedado possui uma camada adicional acima da qual o hypervisor reside e uma terceira camada acima do hypervisor.

### 2.5.4.1 Exemplos de Hypervisors

XEN/Citrix e XenServer o hypervisor Xen, cria conjuntos lógicos de recursos do sistema para possibilitar muitas máquinas virtuais partilhar os mesmos recursos físicos. A execução ocorre diretamente no hardware do sistema. Insere uma camada de virtualização entre o hardware do sistema e as máquinas virtuais, e transforma o hardware do sistema num conjunto de recursos informáticos lógicos que pode alocar de forma dinâmica para qualquer sistema operativo convidado. Os sistemas operativos executados nas máquinas virtuais interagem com recursos

virtuais de maneira igual aos recursos físicos [59]. O XenServer é uma plataforma de virtualização de código aberto desenvolvido no hypervisor do Projeto Xen. O Citrix XenServer é a única plataforma de virtualização de classe empresarial comprovada para utilização em nuvem que fornece de forma grátis, funções essenciais de migração em direto e de gestão centralizada para múltiplos nós [60].

O Citrix XenServer é considerado uma solução de virtualização de servidores de grande porte e com redução completa dos custos do centro de processamento dados. O Citrix Essentials para XenServer fornece um conjunto de capacidades de gestão e automatização que aumenta as capacidades do XenServer para ajudar as empresas, os programadores e utilizadores na transformação de centros de processamento de dados para centros de entrega prontos e automatizados. O Citrix XenServer e Citrix Essentials permitem às pequenas, médias e grandes empresas com os seus orçamentos disponíveis beneficiarem da virtualização de servidores.

O VMware ESXi/vSphere é um hypervisor que permite a instalação e utilização de um sistema operativo, e oferece suporte a software de outros sistemas operativos. O VMware é líder global em infraestrutura de nuvem e tecnologia de espaço de trabalho digital, acelera a transformação digital para ambientes de Tecnologias de Informação em evolução. O VMware ESXi/vSphere é destinado para [57] ambientes de virtualização de servidores com capacidades de migração em tempo real, com a utilização do movimento de máquinas virtuais e inicialização de máquinas virtuais da rede. O VMware ESXi suporta uma virtualização completa, o que requer a instalação de todos os drivers de hardware e software relacionados. Implementa versões sombra de estruturas de sistemas como tabelas de páginas e mantém consistência com as tabelas virtuais por prender todas as instruções que tentam atualizar estas estruturas. Neste caso, um nível extra de mapeamento está na página. As páginas virtuais são mapeadas para páginas físicas em toda a tabela de páginas do sistema operativo convidado [60].

O hypervisor faz a tradução física, para a página da máquina, que de modo ocasional é a página correta na memória física. Com a virtualização, um centro de processamento de dados em nuvem, é transformado em infraestruturas informáticas simples, permite que as empresas ofereçam serviços de Tecnologias de Informação flexíveis e confiáveis. VMware vSphere faz a virtualização e associa os recursos de hardware físicos subjacentes em vários sistemas e fornece um conjunto de recursos virtuais para o Centro de processamento de dados. O VMware vSphere gere um elevado conjunto de infraestruturas como CPUs, armazenamento e rede como um ambiente operativo contínuo e dinâmico, gere a complexidade de um Centro de dados [57].

O Microsoft Hyper-V permite aos programadores, executar uma versão de software de um computador, chamado de máquina virtual. Cada máquina virtual, funciona como um computador completo, executa um sistema operativo e programas. Quando o utilizador precisa de recursos de computação, as máquinas virtuais oferecem mais flexibilidade, ajudam a economizar tempo e custos e são uma maneira mais eficiente de utilizar o hardware, e não apenas executar um sistema operativo em hardware físico. O hypervisor executa cada máquina virtual em seu próprio espaço isolado, isto permite ao programador executar mais de uma máquina virtual no mesmo hardware ao mesmo tempo. Se por alguma razão o programador requerer a esta funcionalidade para evitar problemas inesperados que afetam as outras cargas de trabalho, e grupos ou serviços de acesso a diferentes sistemas disponibilizados para os utilizadores [61].

O KVM [37] é um hypervisor de código aberto utilizado para a virtualização completa. Pode ser utilizado como um driver de kernel adicionado ao Linux. O KVM aproveita as vantagens do kernel Linux padrão e virtualização assistida por hardware, e assim descreve o modelo híbrido. O

KVM inclui a capacidade de virtualização, aumenta o kernel tradicional e módulos de utilizador do Linux com um novo processo convidado, que possui modos próprios de kernel e utilização e respostas para a execução de código de sistemas operativos convidados. O KVM fornece um mecanismo para o espaço do utilizador, para injetar interrupções nos sistemas operativos convidados. O espaço do utilizador é um QEMU ligeiramente modificado, que expõe uma plataforma solução de virtualização para um ambiente de computador, onde, Discos, Adaptadores Gráficos e Dispositivos de Rede são incluídos. Os pedidos [60] de sistemas operativos convidados são intercetados e encaminhados no modo de utilizador para ser emulado pelo QEMU.

Alguns tipos de hypervisor têm compatibilidade com alguns tipos de sistemas de computação em nuvem. A diferença de compatibilidade entre hypervisor e sistema de computação em nuvem são apresentados na Tabela 2.2.

Tabela 2.2: Compatibilidade do Hypervisor com o Sistema de Computação em Nuvem.

Hypervisor	OpenStack	Eucalyptus	Nimbus	OpenNebula	Aneka
Xen	Sim	Sim	Sim	Sim	Sim
Citrix XenServer	Não	Não	Não	Não	Não
VMware ESXi	Não	Não	Não	Não	Não
VMware vSphere	Sim	Sim	Não	Não	Não
VMware	Sim	Sim	Não	Sim	Sim
KVM	Sim	Sim	Sim	Sim	Não
MS Hyper-v	Sim	Não	Não	Não	Não

## 2.6 Conclusão

A computação em nuvem tem vindo a representar uma grande evolução no setor das Tecnologias de Informação, pois fornece aos seus utilizadores a possibilidade de aceder aos seus conteúdos e executar tarefas através da Internet. Tudo isto acontece, sem precisar instalar softwares no computador, porque os dados estão guardados em servidores virtuais, não num computador físico e é possível aceder a diferentes serviços oferecidos pela computação em nuvem online. A computação em nuvem oferece numerosas vantagens aos seus utilizadores, e redução de custos para as empresas que aderem a estes serviços. Os utilizadores, podem desfrutar dos serviços oferecidos com confiança de manter as suas informações guardadas pelos provedores de serviço.



# Capítulo 3

## Sistema de Computação em Nuvem Implementado e Ferramentas Utilizadas

### 3.1 Introdução

Neste capítulo são apresentadas as ferramentas necessárias para o desenvolvimento do sistema de computação em nuvem na plataforma OpenStack, com recurso à migração de máquinas virtuais entre os nós de computação e está dividido em 4 secções. A secção 3.1 apresenta o capítulo. A secção 3.2 descreve os requisitos e equipamentos necessários, bem com a infraestrutura de rede para o desenvolvimento da infraestrutura como serviço. A secção 3.3 descreve a instalação de um conjunto de serviços que compõem a configuração do servidor geral *Controller Node*. A secção 3.4 aborda a configuração de serviços e configuração do hypervisor KVM que compõem os servidores, *Compute1* e *Compute2* que fornecem o serviço de computação. A secção 3.5 descreve a configuração do serviço de armazenamento em bloco, que permite a criação de volumes para suporte de armazenamento para as máquinas virtuais. A secção 3.6 apresenta conclusão geral do deste capítulo.

### 3.2 Arquitetura do Sistema de Computação em Nuvem Implementado

#### 3.2.1 Requisitos do Sistema

A plataforma OpenStack permite a configuração de vários servidores. Estes servidores servem para agregar camadas de gestão WEB, API, CLI e hypervisor. Cada servidor é responsável pelo funcionamento de determinados serviços que compõem o sistema de computação em nuvem. Para a implementação desta infraestrutura, serão configurados cinco servidores. O primeiro servidor é o NTP. O servidor NTP é fundamental para estabelecer a hora entre os servidores. O segundo servidor é o *Controller Node* que irá agregar grande parte dos serviços necessários, o terceiro e quarto servidores, são o *Compute1* e *Compute2*, são os servidores onde será instalado o serviço de computação, e o hypervisor KVM que permite para criar as máquinas virtuais do sistema, o quinto servidor, é o servidor *Block-Storage Node* que permite a criação de volumes de armazenamento que serão anexados às instâncias para a instalação do sistema operativo. Cada servidor terá um modelo de implantação com funções distintas. A infraestrutura de rede física é definida de acordo com os requisitos do sistema a ser implementado, pois o servidor *Controller Node* e o *Compute1* e *Compute2* são equipamentos em que a sua configuração requer duas interfaces de rede. uma interface será alocada como a interface de gestão, a segunda interface será alocada como a interface provider para fornecer conectividade à rede virtual que será criada no sistema de computação em nuvem. A utilização destes servidores, é feita de acordo com

os requisitos de implementação e os serviços necessários para a composição sistema de computação em nuvem. Os serviços específicos do OpenStack serão instalados nos seus destinados servidores. Este processo de instalação e configuração dos serviços, é conhecido como: **Openstack Back-End**. O Openstack Back-End está relacionado à fase inicial do desenvolvimento do sistema. No *Back-End* são criadas a base de dados dos serviços, e feita a instalação dos serviços e configuração dos serviços do OpenStack necessários para o desenvolvimento do da infraestruturura. Também são instalados os pacotes do arquivo da nuvem, bem como a configuração do servidor NTP. Para cada versão utilizada do OpenStack é importante certificar o arquivo da nuvem suportado pelo sistema operativo instalado. Também é de extrema importância a utilização do servidor NTP, pois a não sincronização da hora entre os servidores, implica o não funcionamento adequado de alguns serviços.

### 3.2.2 Características do Hardware do Sistema de Computação em Nuvem Implementado

Para o desenvolvimento [10] deste sistema de computação em nuvem, implementado na plataforma OpenStack, inicialmente será necessária uma infraestruturura física, composta por equipamentos que correspondam com os requisitos necessários para melhor desenvolvimento, implementação e funcionamento do sistema. Estes equipamentos são compostos por máquinas físicas com boa capacidade de memória Ram, processador que suporta virtualização, tipo de sistema operativo linux de 64 bits, e um considerável espaço de armazenamento conforme ilustra a Tabela 3.1.

Tabela 3.1: Características do hardware do sistema de computação em nuvem implementado

	NTP-Server	Controller	Compute1	Compute2	Object-Storage
Memória RAM	15,5 GB	15,5 GB	15,5 GB	15,5 GB	15,5 GB
Processador	intel Core i7	intel Core i7	intel Core i7	intel Core i7	intel core i7
Tipo de SO	64-bit	64-bit	64-bit	64-bit	64-bit
Disco	967,6 GB	983,3 GB	983,3 GB	983,3 GB	983,3 GB
Interfaces de Rede	enp1s0	enp1s0, enp2s0	enp1s0,enp2s0	enp1s0,enp2s0	enp1s0

A infraestruturura de rede apresentada na figura 3.1 é necessária para que as máquinas físicas tenham conectividade à Internet, e durante o desenvolvimento do sistema de computação em nuvem, permite que sejam criadas redes virtuais no sistema desenvolvido.

### 3.2.3 Plataforma OpenStack

A plataforma OpenStack é composta por muitos serviços, alguns destes serviços, estão apresentados na Figura 3.2. Estes serviços do OpenStack permitem o desenvolvimento de um sistema de computação em nuvem. Uma implantação do OpenStack engloba tecnologias de informação que [62] abrangem distribuições do sistema operativo Linux, sistemas de base de dados, filas de mensagens, componentes do OpenStack, políticas de controle de acesso, serviços de registo, e ferramentas de monitorização de segurança. O sistema Operativo Linux é muito utilizado pela [63] qualidade e disponibilidade de código, hardware, suporte, implementações de protocolos de comunicação e aplicação, interfaces de programação, ferramentas de desenvolvimento disponíveis, condições de licenciamento, independência do provedor e custo. O kernel do sistema operativo [64] Linux, é muito utilizado em servidores, sistemas incorporados de segurança

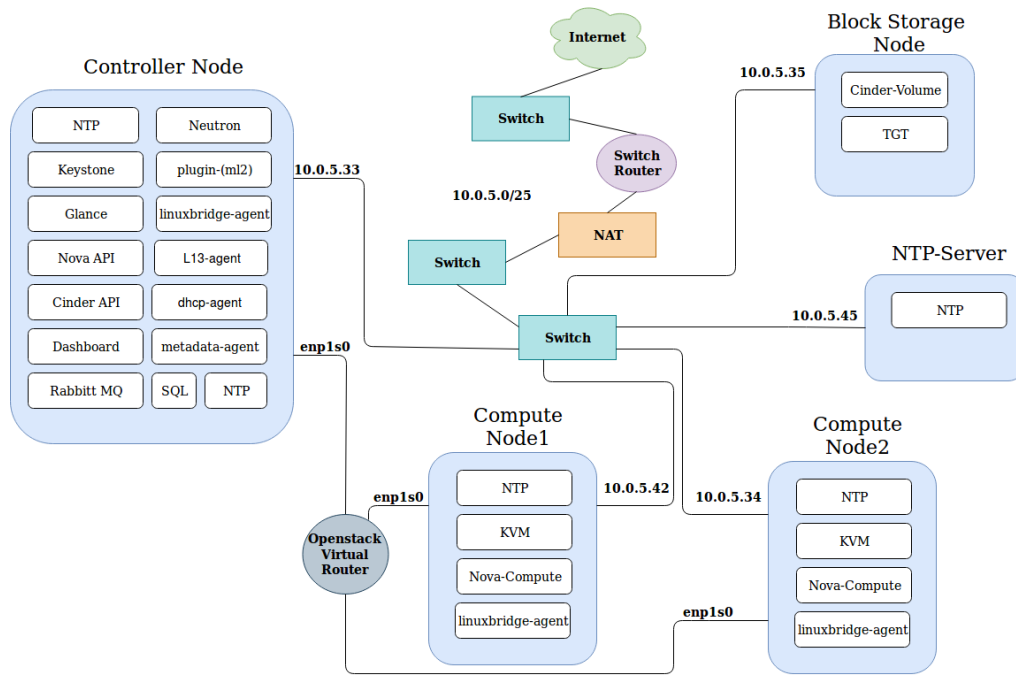


Figura 3.1: Representação esquemática da arquitetura do sistema de computação em nuvem baseado no OpenStack Newton

crítica, electrodomésticos e dispositivos móveis. O kernel do sistema operativo Linux suporta várias funções, como arquitecturas de CPU, sistemas de arquivos e redes. A versão *Ubuntu Desktop 16.04 Long Term Support (LTS)* é um sistema operativo de software de código aberto, [65] destinado para área de trabalho, nuvem, e para os seus componentes conectados à Internet. A versão *Ubuntu Desktop 16.04 LTS* é lançada com todos os recursos necessários, para permitir aos utilizadores de administrar suas áreas de trabalho, que requer a utilização deste sistema.

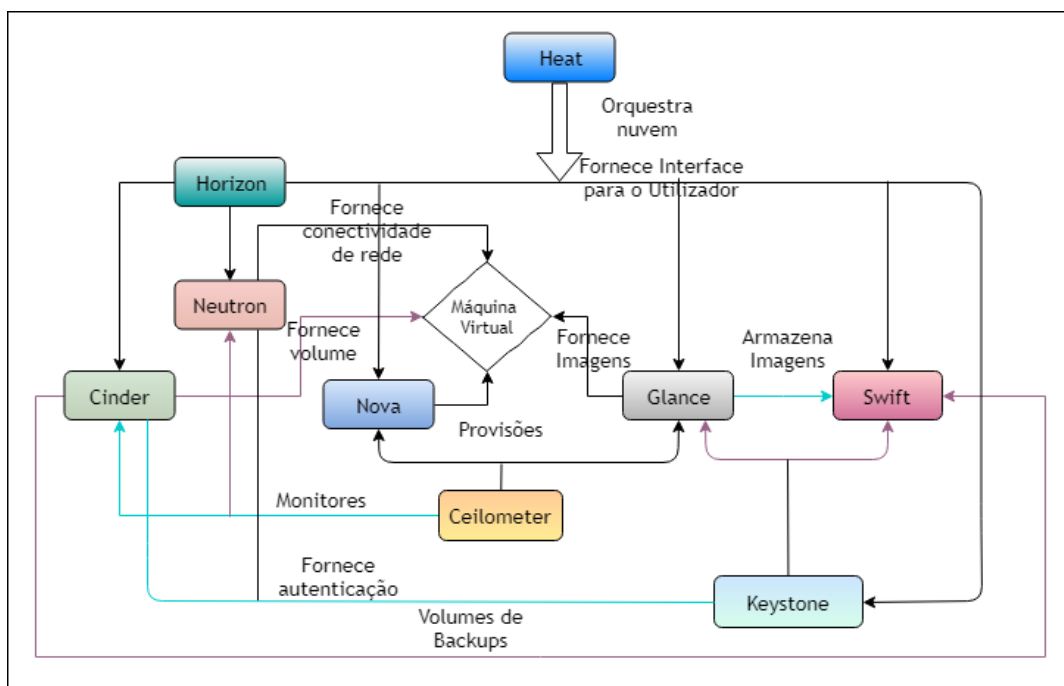


Figura 3.2: Arquitectura Conceptual do OpenStack (Adaptado de [6])

Os serviços apresentados na Figura 3.2, são serviços que fazem parte do conjunto de serviços disponibilizados pela plataforma OpenStack.

- O serviço *Keystone* fornece recursos de autenticação e autorização em toda a infraestrutura da nuvem;
- O serviço *Glance* fornece recursos de gestão de imagem de disco;
- O serviço *Nova* fornece recursos para suportar a criação e gestão de instâncias de máquinas virtuais;
- O serviço *Neutron* fornece recursos de rede para os utilizadores da nuvem;
- O serviço *Cinder* fornece recursos de armazenamento para as instâncias;
- O serviço *Dashboard* fornece interface gráfica para os utilizadores.

Nas secções ??, 3.4 e 3.5 é apresentada informação mais detalhada sobre a instalação e configuração dos serviços acima mencionados.

A cada 6 meses são lançadas novas versões do OpenStack, com novos recursos e novas funcionalidades. Uma particularidade do OpenStack está no ficheiro de nuvem. Um ficheiro de nuvem precisa de ser compatível com a distribuição do sistema operativo Linux. A instalação incorreta do ficheiro de nuvem numa distribuição do Linux não compatível condiciona o desenvolvimento do sistema de computação em nuvem e o funcionamento correto dos serviços do OpenStack. Outra particularidade do OpenStack está nos serviços disponibilizados para cada versão. Um facto é que uma versão do OpenStack pode ter o seu ficheiro de nuvem desenvolvido para um tipo de sistema operativo e ter alguns serviços disponibilizados para outra versão ou outro tipo de sistema operativo. O OpenStack tem diferentes formas de instalação. A instalação pode ser feita através dos Scripts extensíveis **Devstack** e **Packstack** de execução para a instalação rápida do ambiente OpenStack com base nas versões disponíveis.

Os **Scripts** mencionados no parágrafo anterior são utilizados como ambiente de desenvolvimento e como base para implementação dos testes funcionais do projeto OpenStack. Com esta forma de instalação, o programador não tem controlo sobre o sistema, podendo ocorrer erros de difícil perceção e solução, pois estes Scripts de instalação instalam automaticamente grande parte do sistema, sem que o programador tenha o controlo total e conhecimento sobre a configuração do sistema. Uma forma alternativa de instalação do OpenStack consiste na instalação manual, em que o programador cria o sistema de raiz começando na instalação do ficheiro de nuvem, criação da base de dados e instalação e configuração dos serviços necessários para o desenvolvimento do sistema. Com a instalação manual, o programador tem mais facilidade de solucionar possíveis problemas e tem uma perspetiva global e conhecimento mais detalhado de todo ecossistema do sistema implementado. A versão **Newton** do OpenStack é a versão utilizada para o desenvolvimento deste sistema, a qual oferece maior escalabilidade e resiliência de modo a oferecer ao utilizador suporte para uma ampla variedade de cargas de trabalho.

#### 3.2.4 Servidor NTP

O Network Time Protocol (NTP) [66] é um protocolo normalizado da Internet que permite a interação e sincronização do relógio do sistema de todos os computadores na mesma rede. Conforme

apresentado na Figura 3.3 esta sincronização é importante, pois os relógios dos computadores são imprecisos e as horas podem ser alteradas com facilidade, podendo alguns programas de computador ter o seu funcionamento comprometido, caso o computador tiver uma hora errada. O servidor NTP garante a hora exata do servidor e dos computadores com ele sincronizados. Uma característica importante do servidor NTP é que a hora dos computadores está sempre atualizada, isto garante o bom funcionamento dos serviços do OpenStack e melhor desempenho do sistema de computação em nuvem.

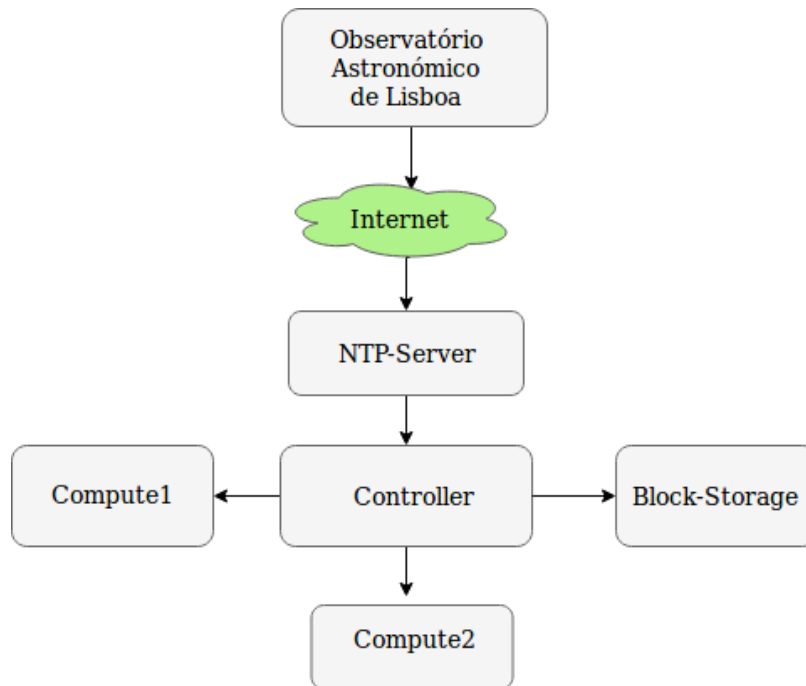


Figura 3.3: Sincronização do Servidor NTP com os outros Servidores.

O servidor NTP está configurado para sincronizar com o servidor do Observatório Astronómico de Lisboa e posterior sincronizar com o Controller Node. O controller Node irá sincronizar o Compute1 e o Compute2 com o Storage Node. A instalação é feita com o comando: `apt-get install ntp`. Após a instalação, o ficheiro `nano /etc/ntp.conf` é editado com algumas configurações. As configurações incluem a remoção da sincronização padrão e substituição pelos servidores do Observatório Astronómico de Lisboa, para que depois seja replicado na rede física interna em utilização (10.0.5.0/25), conforme indicado abaixo.

```
nano /etc/ntp.conf
server ntp02.oal.ul.pt
server ntp04.oal.ul.pt
```

Após a inserção dos servidores do Observatório Astronómico de Lisboa, o servidor NTP é configurado para sincronizar a rede interna 10.0.5.0/25, porque por omissão o servidor NTP é configurado para sincronizar com ele mesmo. Para que possamos indicar o gateway da rede física 10.0.5.1/25 para o servidor NTP, basta acrescentar abaixo do `restrict ::1` a seguinte linha: `restrict 10.0.5.1 mask 255.255.255.128`. A seguir é necessário reiniciar o serviço com os comandos: `systemctl restart ntp` ou `service ntp restart`. Depois é necessário habilitar o start do serviço junto com o boot, com o comando: `systemctl enable ntp`. A verificação do serviço é feita com

o comando: `ntpq -p` e o resultado da sincronização é apresentado na 3.4.

remote	refid	st	t	when	poll	reach	delay	offset	jitter
ntp02.oal.ul.pt	194.117.9.137	2	u	13	64	1	5.256	0.125	1.464
*ntp04.oal.ul.pt	194.117.9.137	2	u	17	64	1	5.500	-0.232	0.634

Figura 3.4: Verificação da sincronização do Servidor NTP.

### 3.3 Arquitetura do Sistema de Computação em Nuvem Baseado no OpenStack Newton

A arquitetura do sistema de computação em nuvem baseado no OpenStack série Newton, implementado no trabalho conducente a esta dissertação, é apresentada na Figura ???. As máquinas físicas encontram-se interligadas em rede de modo a permitir conectividade à Internet e, durante o desenvolvimento do sistema de computação em nuvem, permitir que sejam criadas redes virtuais no sistema desenvolvido. Na figura são também indicados os serviços instalados e configurados em cada nó do sistema.

#### 3.3.1 Configuração do Controller Node.

O *controller node* é a base de todo sistema de computação em nuvem. No controller estão instalados a maioria dos serviços do OpenStack que não são de computação. O *controller Node* é utilizado para executar implementações de serviços do OpenStack para outros nós. O controller node não é utilizado para executar as instâncias de máquinas virtuais. Alguns dos serviços executados pelo *controller node* incluem os serviços de identidade, imagem, rede, parte da gestão do *compute1* e *compute2*, parte da gestão do *block-storage node* e alguns serviços de suporte. Os serviços instalados no *controller node* utilizam uma base de dados para armazenar as informações. Para o desenvolvimento deste sistema o serviço de base de dados instalado é o Structured Query Language (SQL). As operações e informações entre os serviços são coordenadas através de uma fila de mensagens do serviço *rabittmq-server* utilizado pelo OpenStack. O serviço NTP faz a sincronização da hora entre todos os nós (Nodes).

#### 3.3.2 Network Time Protocol (NTP)

No *controller node*, o serviço NTP é configurado para sincronizar com o servidor NTP. Esta sincronização é necessária para atualização da hora do computador e dos serviços do Openstack. Após a sincronização com o servidor e estabelecer a hora no sistema, o *controller Node* será servidor de sincronização para os outros computadores da rede. Para sincronização correta dos serviços entre os nós, será instalado o **Chrony**, no *controller node*, com o comando `apt install chrony`. O **chrony** é uma implementação do NTP. Após a instalação é necessário configurar o ficheiro do serviço **chrony**. O resultado da sincronização do *controller node* com o servidor NTP é apresentado na Figura 3.5.

```
nano /etc/chrony/chrony.conf
```

```
server 10.0.5.45
```

```
allow 10.0.5.0/25
```

```
root@controller:/home/controller# chronyc sources
210 Number of sources = 1
MS Name/IP address          Stratum Poll Reach LastRx Last sample
=====
^* 10.0.5.45                 3      6   77    0  +897ns[ +18us] +/- 56ms
```

Figura 3.5: Verificação da sincronização do Controller com o servidor NTP.

### 3.3.3 Identity Service (Keystone)

O serviço de identidade [10] faz a autenticação e autorização dos utilizadores normais, utilizadores administrativos e utilizadores de serviços do OpenStack. Este serviço é utilizado por todos os componentes do OpenStack. O *Identity Service* é um serviço OpenStack que fornece autenticação de cliente de API, de serviço e autorização de multi-tenant distribuído e implementa a API de Identidade do OpenStack. Também fornece um catálogo [54] central de serviços e nós de extremidade executados numa nuvem particular do OpenStack, que atua como uma diretoria de serviços para outros sistemas OpenStack. Todos os serviços do OpenStack incluem catálogo [11] dos terminais. A autenticação e autorização, são fornecidos pelo *Identity Service* [10] que suporta autenticação baseada em *token* e autorização de serviço para o utilizador. O *Identity Service* interage com os outros serviços como apresentado na Figura 3.6 e fornece um ponto único de integração para gerir autenticação, autorização. O serviço de identidade é o serviço com o qual um utilizador interage. Após a autenticação, o utilizador pode aceder outros serviços do OpenStack. Os outros serviços utilizam o serviço de identidade para garantir que os utilizadores sejam quem dizem ser.

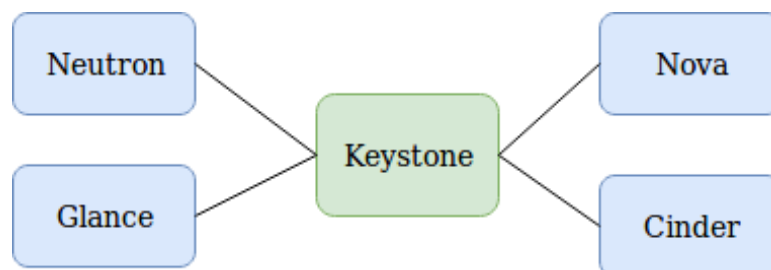


Figura 3.6: Interação dos serviços em OpenStack.

### 3.3.4 Image Service (Glance)

O serviço de imagem permite aos utilizadores, utilizar as imagens disponíveis no sistema conforme ilustra a Figura 3.7 para instalar e inicializar as máquinas virtuais [10]. Os utilizadores podem carregar e pesquisar recursos de dados, utilizados com outros serviços que incluem imagens e definições de metadados, podem registar e recuperar imagens de máquinas virtuais são

serviços de imagem de relance, providenciados pelo *Glance* [11]. Neste serviço, os utilizadores podem fazer um upload e pesquisar recursos de dados que podem ser utilizados com outros serviços.

ID	Name	Status
49a7f026-a4d5-469a-bd42-b69a72e2ef50	ubuntu-18.10 server	active
eb38c378-352f-4f5d-9ae0-9a6ac5802969	ubuntu-18.4 desktop	active
4b82aac-345d-4328-bb78-8f97b258b939	ubuntu-18 live server	active
da3235d8-591b-4d53-984a-813dd0e74e33	ubuntu-16 server	active
d8809aea-60a5-49ae-83bd-1fe8f8956646	ubuntu-16 desktop	active
8de49558-6ebc-4ef8-831b-7b4f838ccc94	ubuntu-14 server	active
83d12df8-0e8f-45b2-bd3b-33d670a2298c	ubuntu-14 desktop	active

Figura 3.7: Lista de imagens dispniveis no Sistema.

### 3.3.5 Compute Service (Nova)

O nova compute [6] é responsável pela gestão das atividades necessárias para desenvolver funções relacionadas com as máquinas virtuais. Foi desenvolvido para provisionar e gerir [11] redes de máquinas virtuais, faz a implantação automática das instâncias de máquina virtual provisionadas. Os serviços de computação configurados no servidor *controller node* são apresentados na Figura 3.8. O *Compute Service* interage com o *Identity Service* para autenticação, com o *Image Service* para imagens de disco e servidor e com o *Horizon* para a interface administrativa conforme ilustra a Figura ?? . O acesso à imagem é limitado por projetos e utilizadores. O OpenStack *Horizon Service* pode ser dimensionado horizontalmente no hardware padrão e baixar imagens para iniciar instâncias.

ID	Binary	Host	Zone	Status	State	Updated At
6	nova-consoleauth	controller	internal	enabled	up	2019-01-10T21:57:07.000000
7	nova-scheduler	controller	internal	enabled	up	2019-01-10T21:57:14.000000
8	nova-conductor	controller	internal	enabled	up	2019-01-10T21:57:07.000000

Figura 3.8: Serviços Compute do Controller Node.

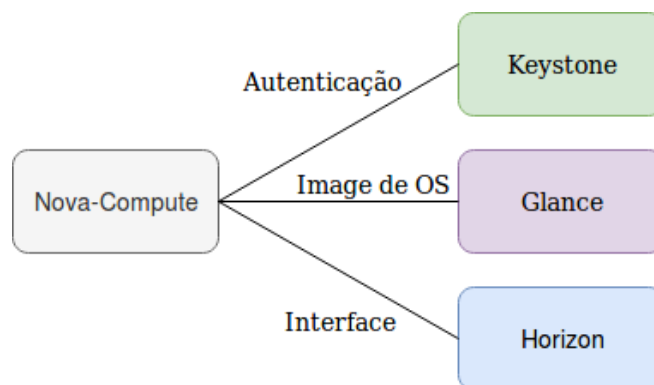


Figura 3.9: Representação esquemática da interação do compute com outros serviços.

### 3.3.6 Network Service (Neutron)

O serviço neutron [10] permite criar e conectar dispositivos de interface geridos por outros serviços OpenStack para redes. Os plug-ins podem ser implementados para acomodar diferentes equipamentos e software de rede. O serviço neutron fornece flexibilidade à arquitetura e à implantação do OpenStack. O serviço neutron é composto por agentes de rede instalados no servidor controller apresentados na Figura 3.10 que funcionam com várias tarefas necessárias e utilizadas para a implementação das redes virtuais pública e privadas do sistema. O *neutron-dhcp-agent*, atribui os endereços IPs automaticamente às máquinas virtuais, o *neutron-l3-agent* fornece suporte para os routers, utiliza o IP do Linux e o iptables para execução e encaminhamento L3 e o Network Address Translation (NAT). O *neutron-metadata-agent* fornece informações de configuração para instâncias e o *neutron-linuxbridge-agent* cria a infraestrutura de rede virtual para instâncias. No *Compute1* e *Compute2* o *Network Service* [54] lida com a criação e gestão de uma infraestrutura de rede virtual na nuvem OpenStack. Os elementos incluem redes, sub-redes e routers.

ID	Agent Type	Host	Availability Zone	Alive	State	Binary
a396b660-332b-451e-956d-6fa31bc374d3	Linux bridge agent	controller	None	True	UP	neutron-linuxbridge-agent
cb8d81a7-7199-4af1-81cf-670de3a7e0c2	Metadata agent	controller	None	True	UP	neutron-metadata-agent
cf28b219-f6c2-4ec0-bf1f-ccc91b5bf0a0	DHCP agent	controller	nova	True	UP	neutron-dhcp-agent
da2c94c4-78da-43ff-a0b1-87e9cc27233f	L3 agent	controller	nova	True	UP	neutron-l3-agent

Figura 3.10: Agentes Neutron no Controller Node.

### 3.3.7 Block Storage (Cinder)

O serviço de armazenamento em bloco [10] fornece recursos de armazenamento de blocos persistentes que as instâncias de computação podem utilizar. Permite aos utilizadores gravar imagens em um dispositivo de armazenamento em bloco para as instâncias. O armazenamento da máquina virtual é feito sem interrupções, adicionado pelo *Cinder*. Uma infraestrutura para gerir volumes e interagir com o OpenStack Compute para fornecer volumes para instâncias são providenciados pelo *Block Storage*. O serviço de armazenamento em bloco instalado no servidor *Controller Node*, apresentado na Figura 3.11, permite gerir volumes e tipos de volume [11] e é responsável por providenciar o armazenamento de bloco, para as instâncias em execução.

Binary	Host	Zone	Status	State	Updated At
cinder-scheduler	controller	nova	enabled	up	2019-01-11T03:50:24.000000

Figura 3.11: Serviço Cinder-Scheduler.

### 3.3.8 Dashboard (Horizon)

O *dashboard* [10] é uma aplicação web, fornece uma interface gráfica para os serviços OpenStack é implementado através do mod wsgi no Apache. Permite ao utilizador modificar o código do *dashboard* para que esteja adequado para diferentes sites . O serviço de *dashboard* conforme ilustra a Figura 3.12 fornece uma interface gráfica de utilizador para os utilizadores finais e administradores. Permite operações como criação de instâncias, gestão de rede e configuração de controlo de acesso.

## Log in

---

**Domain**

**User Name**

**Password**

Figura 3.12: Verificação do Acesso ao Dashboard.

### 3.4 Configuração do Compute Node1 e Node2

Os Compute Node1 e Node2, também designados por Compute1 e Compute2, do OpenStack funcionam como servidores de computação que executam o hypervisor KVM e permitem a criação e implementação de máquinas virtuais. O *Compute1* e *Compute2* fornecem recursos de computação para a execução e migração das máquinas virtuais. Nos servidores de computação nomeadamente *compute1* e *compute2*, é necessário que seja instalado o serviço *chrony*, que permite a sincronização da hora com o servidor *Controller Node* cuja configuração do serviço NTP está descrito na subsecção 3.3.1. A instalação é feita com o comando `apt-get install chrony`. Após a instalação o ficheiro de configuração do serviço *chrony* é editado para sincronizar com o servidor *Controller Node*.

#### 3.4.1 Network Time Protocol (NTP)

Configuração do servidor NTP, nos Compute Node1 e Node2.

```
nano /etc/chrony/chrony.conf
server controller
```

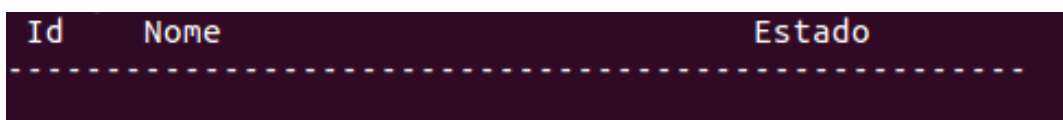
```
210 Number of sources = 1
MS Name/IP address          Stratum Poll Reach LastRx Last sample
=====
^* controller                4      6   377   15   +25us[ +43us] +/-  27ms
```

Figura 3.13: Lista de Imagens

### 3.4.2 Instalação e Configuração do Hypervisor KVM

O hypervisor KVM (Kernel-based Virtual Machine) [67] utiliza a técnica de virtualização completa como um driver de kernel adicionado ao Linux e utiliza a virtualização assistida por hardware. O KVM é um hypervisor de modelo híbrido [35] com o kernel padrão do Linux. O kernel carregado converte o kernel do Linux em hypervisor bare-metal. O KVM emula e fornece a criação de máquinas virtuais no sistema operativo hospedeiro. O KVM é um Kernel Linux e um módulo do FreeBSD e permite que um programa de espaço do utilizador tenha acesso a características de virtualização de hardware dos processadores [68] e que diferentes máquinas convidadas possam ser executadas na parte superior do host kernel através do emulador qemu. Todo o kernel da máquina convidada e o espaço do utilizador convidado permanecem dentro do espaço de endereço qemu isolado do host. O KVM é considerado uma solução de [35] virtualização para o sistema operativo baseado em kernel Linux num hardware x86 que possui extensões para virtualização. A sua composição inclui um módulo do kernel que apresenta um módulo específico do processador e a infraestrutura de virtualização principal. O KVM pode ser instalado nos processadores x86 e o seu desenvolvimento é feito no kernel principal do sistema operativo Linux. Com a sua utilização, o utilizador pode executar várias máquinas virtuais e executar imagens não personalizadas do Windows ou do Linux. O KVM é instalado no *Ubuntu 16.04 LTS* e a Figura 3.14 ilustra a verificação da instalação do hypervisor. O KVM utiliza o kernel do Linux e kernel do sistema [58] hospedeiro, isto ajuda o KVM reutilizar drivers de dispositivos Linux existentes e outras partes do kernel do Linux . O KVM é configurado como o hypervisor padrão e composto por dois componentes principais:

- Dispositivo de caracteres de espaço de kernel driver, que fornece serviços de virtualização de CPU e virtualização de memória através de um arquivo de dispositivo de caracteres `/dev/kvm`;
- Emulador de espaço do utilizador para hardware convidado emulação (QEMU). Requisição de serviço comunicações entre esses dois componentes. A criação de máquinas virtuais é tratada através do sistema IOCTL sobre o ficheiro de dispositivo `/dev/kvm`.



Id	Nome	Estado
-----		

Figura 3.14: Verificação da instalação do Servidor KVM nos servidores Compute1 e Compute2.

### 3.4.3 Nova-Compute (nova)

O serviço *Nova-Compute* [10] suporta a criação de máquinas virtuais, servidores bare metal, e tem suporte limitado para caixas do sistema. O nova-compute é executado como um conjunto de daemons sobre os servidores Linux existentes para fornecer o serviço de computação. O OpenStack *Nova-Compute* pode ser utilizado para hospedar e gerir sistemas de computação em nuvem. Sendo uma parte importante de um sistema Infraestrutura como serviço (IaaS). O serviço *Nova-Compute* funciona em conjunto com o *Compute Serviço* ver Figura 3.15.

ID	Binary	Host	Zone	Status	State	Updated At
9	nova-compute	compute2	nova	enabled	up	2019-01-10T22:02:09.000000
10	nova-compute	compute1	nova	enabled	up	2019-01-10T22:02:13.000000

Figura 3.15: Serviços de Computação dos Servidores Compute1 e Compute2.

### 3.4.4 Network Agent

O agente de rede lida com várias tarefas utilizadas para implementar redes virtuais. O agente de rede instalado no *Compute1* e *Compute2* é o *neutron-linuxbridge-agent* que interage e funciona em conjunto com os agentes de rede instalados no *Controller Node* conforme ilustra a Figura 3.16

ID	Agent Type	Host	Availability Zone	Alive	State	Binary
161b4abb-6300-4633-ba10-c98a5ddbfa49	Linux bridge agent	controller	None	True	UP	neutron-linuxbridge-agent
598cf8f6-d2b3-4ecf-9f5f-8c7b6fd6598e	Linux bridge agent	compute2	None	True	UP	neutron-linuxbridge-agent
8c45fe0e-a288-4259-a174-95f1187be159	L3 agent	controller	nova	True	UP	neutron-l3-agent
a1ee40cc-b2d2-4b1c-9d0e-9f67ba74cd36	Linux bridge agent	compute1	None	False	UP	neutron-linuxbridge-agent
b1284d48-4705-4c43-be19-09a17e171695	Metadata agent	controller	None	True	UP	neutron-metadata-agent
f07812c7-b4b8-4f52-8a2e-0126827dd57b	DHCP agent	controller	nova	True	UP	neutron-dhcp-agent

Figura 3.16: Agentes de Rede dos Servidores Controller Node, Compute1 e Compute2.

## 3.5 Configuração do Block-Storage Node

A API do *Block-Storage* e os serviços do *Block-Storage* geralmente são executados no controller node. Dependendo dos drivers utilizados, o serviço de volume é executado no *Controller Node* descrito na Secção 3.3.7. O serviço *Block-Storage* fornece volumes lógicos para o armazenamento das instâncias.

### 3.5.1 Network Time Protocol (NTP)

Conforme descrito na Secção 3.3.2, o serviço Chrony foi instalado no *Block-Storage Node* para sincronizar com o servidor NTP do *Controller Node* de modo a atualizar a hora do computador e assim garantir o bom funcionamento do serviço NTP *Block-Storage Node* ilustrado na Figura 3.17.

```
nano /etc/chrony/chrony.conf
server controller
```

```
210 Number of sources = 1
MS Name/IP address          Stratum Poll Reach LastRx Last sample
=====
^* controller                4   6   77    8  -2448ns[-8000ns] +/- 205ms
```

Figura 3.17: Verificação do Serviço NTP do Block-Storage Node.

### 3.5.2 Volume-Service (Cinder)

O serviço *Cinder-Volume* ilustrado na Figura 3.18 tem uma interação direta com o serviço *Cinder-Scheduler*, (ver Figura ??) mas também pode interagir com outros provedores de armazenamento,

por meio do serviço rabbitmq, ou uma arquitetura de driver.

```
+-----+-----+-----+-----+-----+-----+
| Binary      | Host      | Zone | Status | State | Updated At |
+-----+-----+-----+-----+-----+-----+
| cinder-volume | block@lvm | nova | enabled | up    | 2019-01-11T03:48:15.000000 |
+-----+-----+-----+-----+-----+-----+
```

Figura 3.18: Verificação do serviço de armazenamento do servidor Block-Storage.

## 3.6 Conclusão

Os sistemas de computação em nuvem proporcionam benefícios tanto para os utilizadores como para os fornecedores de serviço. A versão Newton da plataforma OpenStack é a versão mais estável para criar um serviço do tipo IaaS, pois é uma versão com alta disponibilidade e adaptabilidade e, por norma, disponibiliza um conjunto de serviços que podem ser implementados no sistema desenvolvido. Os serviços disponibilizados pela plataforma OpenStack fornecem os componentes básicos para o desenvolvimento de um sistema em nuvem. Os componentes de um IaaS são hospedados por fornecedores de serviços de nuvem, localizados num centro de dados.



# Capítulo 4

## Resultados

### 4.1 Introdução

Neste capítulo são apresentadas as funcionalidades da infraestrutura em nuvem, resultantes das configurações e implementações dos serviços do OpenStack no *back-end*. O capítulo está dividido em 4 Secções. A Secção 4.1 apresenta o capítulo. A Secção 4.2 aborda de uma forma geral das funcionalidades do sistema, com destaque para os volumes de armazenamento para as máquinas virtuais, dos flavors que permitem definir a arquitetura das máquinas virtuais, dos grupos de segurança e par de chaves, apresenta as instâncias criadas no sistema, apresenta a tipologia de rede utilizada no sistema, fala dos IP flutuantes que permitem aceder as máquinas virtuais através de uma rede pública, fala da utilização do Hypervisor e um resumo da utilização do sistema. A Secção 4.3 descreve o *benchmark* de CPU e o *benchmark* de memória feito nos servidores *Controller Node*, *Compute1* e *Compute2* e *Block-Storage Node*. A Secção 4.4 apresenta a migração passiva e migração ativa, do tempo de execução da migração passiva e migração ativa e da comparação entre os dois tipos de migração.

### 4.2 Visão Geral do Sistema

O Horizon fornece uma interface de utilizador para os serviços do OpenStack. No horizon são apresentados conteúdos que resultam das configurações feitas no *back-end*, para as solicitações dos utilizadores em API no *front-end*. Este sistema é classificado como uma nuvem IaaS privada da plataforma OpenStack. Algumas funcionalidades são descritas nos itens seguintes.

- Availability Zones

Na zona de disponibilidade conforme ilustra a Figura 4.1 são agrupados os nós que executam os agentes de rede. A execução dos agentes de rede permite aos utilizadores associar os seus recursos a uma zona de disponibilidade, para que os mesmos possam obter alta disponibilidade. Para que os recursos de rede tenham alta disponibilidade é necessário utilizar a zona de disponibilidade.

Availability Zone Name	Hosts	Available
internal	• controller (Services Up)	Yes
nova	• compute1 (Services Up) • compute2 (Services Up)	Yes

Figura 4.1: Zona de disponibilidade.

- Volumes de Armazenamento

Os volumes são dispositivos de armazenamento em bloco que são anexados às instâncias, conforme ilustra a Figura 4.2, para permitir o armazenamento de conteúdos. Os volumes podem ser anexados a uma instância em execução ou ser desanexados e, à posterior, ser anexados a outra instância. Os volumes podem ser criados no momento em que a instância está a ser criada ou podem ser criados paralelamente às instâncias. Os utilizadores normais e administradores podem criar volumes, mas apenas os utilizadores administradores podem criar tipos de volumes.

<input type="checkbox"/>	Project	Host	Name	Size	Status	Type	Attached To
<input type="checkbox"/>	admin	block@lvm#LVM	volume5	60GiB	In-use	-	Attached to <a href="#">instance5</a> on /dev/vda
<input type="checkbox"/>	admin	block@lvm#LVM	volume6	60GiB	In-use	-	Attached to <a href="#">instance6</a> on /dev/vda
<input type="checkbox"/>	demo	block@lvm#LVM	demo-volume	80GiB	In-use	-	Attached to <a href="#">demo-instance</a> on /dev/vda
<input type="checkbox"/>	admin	block@lvm#LVM	volume4	80GiB	In-use	-	Attached to <a href="#">instance4</a> on /dev/vda
<input type="checkbox"/>	admin	block@lvm#LVM	volume3	80GiB	In-use	-	Attached to <a href="#">instance3</a> on /dev/vda
<input type="checkbox"/>	admin	block@lvm#LVM	volume2	80GiB	In-use	-	Attached to <a href="#">instance2</a> on /dev/vda
<input type="checkbox"/>	admin	block@lvm#LVM	volume1	80GiB	In-use	-	Attached to <a href="#">instance1</a> on /dev/vda

Figura 4.2: Volumes.

- Flavors

Os Flavors, conforme ilustra a Figura 4.3, definem a capacidade de computação, memória e armazenamento de uma instância. Os flavors podem ser criados, editados ou eliminados pelo utilizador ou pelo administrador. Por serem criados pelo administrador, os flavors são públicos para que outros utilizadores não administradores possam a eles ter acesso. Na versão Newton do Openstack não foram disponibilizado flavors padrão, pelo que os flavors criados neste trabalho são baseados nas versões anteriores à versão Newton.

<input type="checkbox"/>	Flavor Name	VCPUs	RAM	Root Disk	Ephemeral Disk	Swap Disk	RX/TX factor
<input type="checkbox"/>	m1.large	4	8GB	80GB	0GB	0MB	1.0
<input type="checkbox"/>	m1.medium	2	4GB	40GB	0GB	0MB	1.0
<input type="checkbox"/>	m1.small	1	2GB	20GB	0GB	0MB	1.0
<input type="checkbox"/>	m1.tiny	1	2GB	1GB	0GB	0MB	1.0
<input type="checkbox"/>	m1.xlarge	8	16GB	160GB	0GB	0MB	1.0

Figura 4.3: Tipos de flavor.

- Grupos de Segurança e Par de Chaves

Os grupos de segurança funcionam como *firewall* virtual para as instâncias e outros recursos da rede. Os grupos de segurança também disponibilizam regras para especificar as regras de acesso à rede. As regras disponibilizadas pelos grupos de segurança especificam as regras de acesso à rede para as instâncias e outros recursos.

O par de chaves, é na verdade a chave pública de um par de chaves *OpenSSH*. Esta chave *OpenSSH* é utilizada para permitir o acesso às instâncias. O par de chaves pode ser criados no back-end ou no Front-end por utilizadores administrativos, e utilizadores não administrativos. Várias instância podem ser criadas utilizando o mesmo par de chaves.

- Instâncias

As instâncias são as máquinas virtuais que podem ser criadas e executadas nos servidores físicos de computação conforme ilustra a Figura 4.4, dentro do sistema de computação em nuvem. As instâncias são iniciadas e executadas a partir de uma cópia da imagem de um sistema operativo. As alterações feitas na instância não afeta a imagem de sistema operativo. Esta imagem é fornecida pelo serviço glance do OpenStack. A gestão das instâncias é controlada pelo serviço compute.

<input type="checkbox"/>	Project	Host	Name	Image Name	IP Address	Size	Status	Task	Power State
<input type="checkbox"/>	admin	compute2	instance3	ubuntu-18.4 desktop	<ul style="list-style-type: none"> <li>• 192.168.1.13</li> </ul> Floating IPs: <ul style="list-style-type: none"> <li>• 10.0.5.23</li> </ul>	m1.medium	Shutoff	None	Shut Down
<input type="checkbox"/>	admin	compute2	instance2	ubuntu-14 desktop	<ul style="list-style-type: none"> <li>• 192.168.30.3</li> </ul> Floating IPs: <ul style="list-style-type: none"> <li>• 10.0.5.5</li> </ul>	m1.small	Shutoff	None	Shut Down
<input type="checkbox"/>	admin	compute2	instance1	ubuntu-18.4 desktop	<ul style="list-style-type: none"> <li>• 192.168.10.11</li> </ul> Floating IPs: <ul style="list-style-type: none"> <li>• 10.0.5.15</li> </ul>	m1.medium	Shutoff	None	Shut Down
<input type="checkbox"/>	admin	compute1	instance6	ubuntu-16 desktop	<ul style="list-style-type: none"> <li>• 192.168.20.7</li> </ul> Floating IPs: <ul style="list-style-type: none"> <li>• 10.0.5.16</li> </ul>	m1.medium	Active	None	Running
<input type="checkbox"/>	admin	compute1	instance5	ubuntu-16 desktop	<ul style="list-style-type: none"> <li>• 192.168.30.12</li> </ul> Floating IPs: <ul style="list-style-type: none"> <li>• 10.0.5.14</li> </ul>	m1.medium	Active	None	Running
<input type="checkbox"/>	admin	compute1	instance4	ubuntu14-server	<ul style="list-style-type: none"> <li>• 192.168.1.9</li> </ul> Floating IPs: <ul style="list-style-type: none"> <li>• 10.0.5.7</li> </ul>	m1.small	Active	None	Running

Figura 4.4: Instâncias criadas no sistema de computação em nuvem.

- Topologia de Rede do Sistema

A topologia de rede, conforme ilustra a Figura ??, mostra o gráfico topológico das instâncias que estão ligadas á rede. Inicialmente, foi criada a rede pública do tipo provedor. Esta rede pública é a mesma rede dos servidores físicos. Posteriormente, foram criadas as redes privadas. As redes privadas estão ligadas à rede pública através de um router. É através desta ligação à rede pública que as instâncias podem ter acesso à Internet.

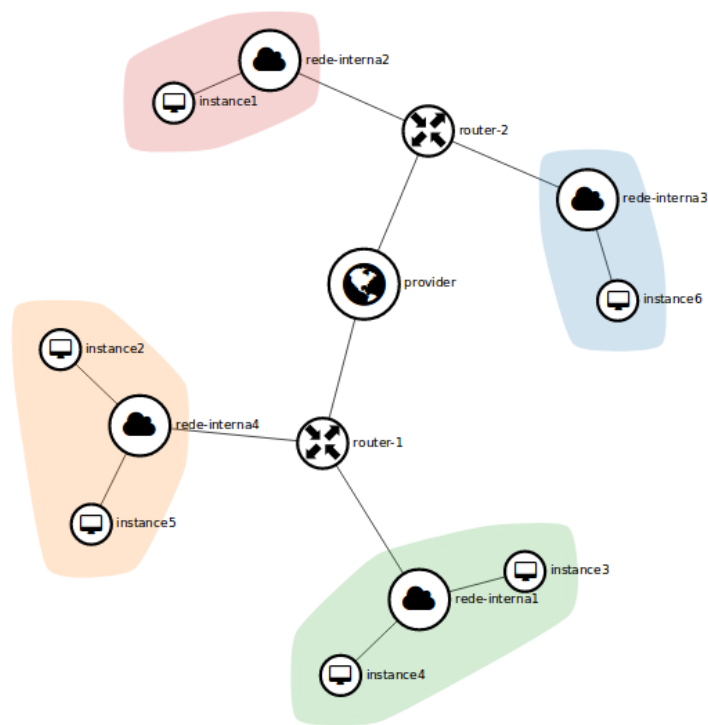


Figura 4.5: Topologia de rede.

- IP flutuante

Os endereços do Internet Protocol (Internet Protocol (IP)) flutuantes são endereços IP adicionados dinamicamente às instâncias em execução, a partir da rede externa. São fornecidos pelo serviço neutron, sem utilizar um servidor Dynamic Host Configuration Protocol (Dynamic Host Configuration Protocol (DHCP)). O sistema operativo das instâncias não detecta a atribuição do endereço IP flutuante, uma vez que é o agente L3 do serviço neutron que é responsável pela interface do IP flutuante. Os endereços IPs flutuantes permitem que as instâncias sejam acedidas a partir de uma rede externa, portanto, exterior à rede do sistema de computação em nuvem. Um endereço IP flutuante é alocado a uma sub-rede associada à rede externa. Ao contrário de um endereço IP flutuante, um endereço IP privado é atribuído à interface de rede de uma instância pelo servidor DHCP da rede privada. As instâncias ligadas à mesma rede conseguem estabelecer comunicações. O endereço IP flutuante e o endereço IP privado podem ser utilizados numa única interface de rede.

- Hypervisor

No resumo do hypervisor são apresentadas, conforme se pode ver na Figura 4.6, as estatísticas dos recursos de utilização em cada servidor de computação ilustrado na Figura 4.6, a utilização total e restante dos recursos disponíveis bem como a quantidade de máquinas virtuais que cada servidor de computação hospeda.

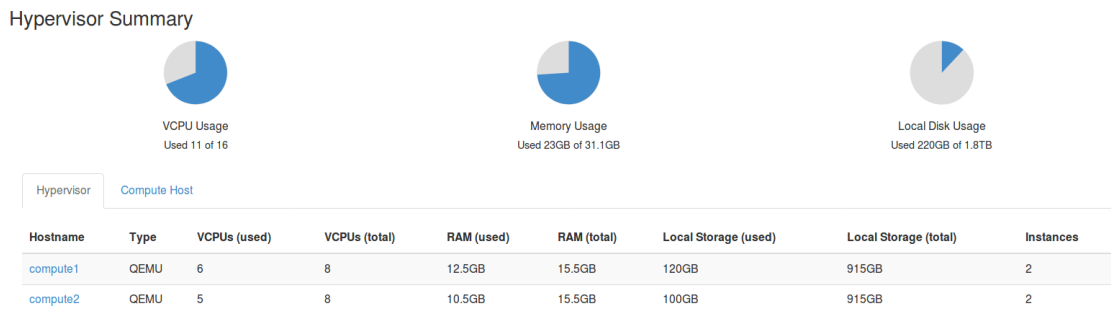


Figura 4.6: Utilização do Hypervisor.

- Resumo Geral do Sistema

No resumo geral do sistema são apresentadas, conforme se pode ver na Figura 4.7, informações gerais sobre a utilização dos recursos disponibilizados para criação e funcionamento das instâncias assim como os limites de utilização dos recursos disponíveis restantes.

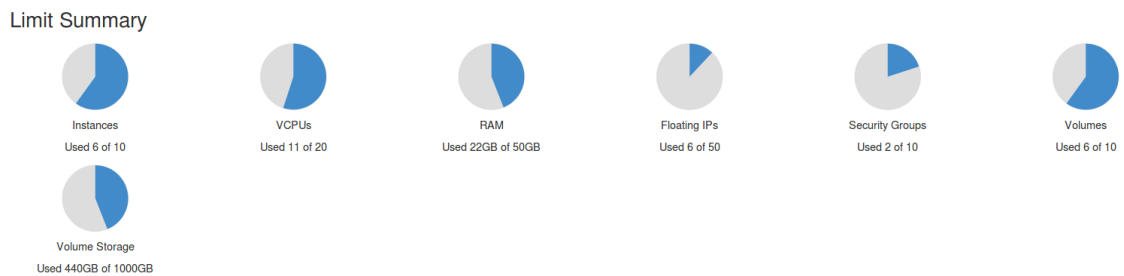


Figura 4.7: Utilização do sistema.

### 4.3 Benchmark nos Servidores Físicos

Após a implementação do sistema de computação em nuvem e a criação das máquinas virtuais, houve a necessidade de avaliar o desempenho de CPU e de Memória dos servidores controller, compute1, compute2 e block-storage, a fim de avaliar as características técnicas e o comportamento real dos servidores com o funcionamento do sistema de computação em nuvem. Na Secção 4.3.1 é descrita a execução do benchmark para avaliar o desempenho de CPU dos servidores, para avaliar a carga de trabalho dos drivers e dos núcleos instalados no sistema operativo. Na secção 4.3.2 é descrita a execução do benchmark de memória dos servidores.

### 4.3.1 Bechmark de CPU

O Geekbench é um *benchmark* de processador *cross-platform*, com um sistema de pontuação que mede a potência do sistema e avalia o desempenho em single e multi-core, com cargas de trabalho através da utilização de *workloads* que simulam cenários reais. Foi realizado um teste a nível de CPU nos servidores controller, compute1, compute2 e block-storage. O teste consiste na execução da ferramenta Geekbench sobre o sistema operativo Ubuntu 16.04, tendo sido obtidos os resultados em single-core que se encontram apresentados na Tabela 4.1.

Tabela 4.1: Resultados dos testes realizados com o Benchmark de CPU

Workload	Controller	Compute1	Compute2	Block-Storage
AES	2.52 (GB/s)	2.45 (GB/s)	5.40 (GB/s)	2.54 (GB/s)
LZMA	6.94 (MB/s)	7.42 (MB/s)	30.8 (MB/s)	7.36 (MB/s)
JPEG	41.1 (Mp/s)	44.1 (Mp/s)	205.6 (Mp/s)	44.3 (Mp/s)
Canny	72.5 (Mp/s)	73.6 (Mp/s)	230.6 (Mp/s)	76.5 (Mp/s)
Lua	5.85 (MB/s)	6.07 (MB/s)	23.0 (MB/s)	6.11 (MB/s)
Dijkstra	3.93 (MTE/sec)	4.41 (MTE/s)	9.72 (MTE/s)	4.42 (MTE/s)
SQLite	151.5 (Krows/s)	158.1 (Krows/s)	650.2 (Krows/s)	159.0 (Krows/s)
HTML5 Parse	26.6 (MB/s)	27.8 (MB/s)	107.2 (MB/s)	27.8 (MB/s)
HTML5 DOM	4.10 (MElements/s)	4.22 (MElements/s)	10.2 (MElements/s)	4.47 (MElements/s)
Hist. Equalization	152.5 (Mp/s)	161.2 (Mp/s)	578.3 (Mp/s)	157.8 (Mp/s)
PDF Rendering	133.4 (Mp/s)	175.3 (Mp/s)	335.4 (Mp/s)	176.5 (Mp/s)
LLVM	634.9 (functions/s)	685.5 (functions/s)	2.51 (Kfunctions/s)	691.8 (functions/s)
Camera	14.6 (images/s)	16.2 (images/s)	72.6 (images/s)	16.2 (images/s)
SGEMM	104.4 (Gflops)	119.0 (Gflops)	212.3 (Gflops)	116.6 (Gflops)
SFFT	14.1 (Gflops)	15.0 (Gflops)	53.6 (Gflops)	14.6 (Gflops)
N-Body Physics	3.76 (Mpairs/s)	3.84 (Mpairs/s)	14.4 (Mpairs/s)	3.95 (Mpairs/s)
Ray Tracing	440.4 (Kpixels/s)	450.3 (Kpixels/s)	2.54 (Mp/s)	465.9 (Kp/s)
Rigid Body Physics	14510.2 (FPS)	15446.4 (FPS)	79622.6 (FPS)	15526.7 (FPS)
HDR	23.3 (Mp/s)	23.8 (Mp/s)	72.7 (Mp/s)	24.3 (Mp/s)
Gaussian Blur	93.6 (Mp/s)	100.8 (Mp/s)	368.2 (Mp/s)	102.4 (Mp/s)
Speech Recognition	46.6 (Words/s)	52.8 (Words/s)	73.6 (Words/s)	53.3 (Words/s)
Memory Copy	9.59 (GB/s)	10.4 (GB/s)	(9.09 GB/s)	14.9 (GB/s)
Memory Latency	67.7 (ns)	65.3 (ns)	71.6 (ns)	(64.9 ns)
Memory Bandwidth	13.0 (GB/s)	12.1 (GB/s)	12.0 (GB/s)	21.5 (GB/s)

### 4.3.2 Bechmark de Memória

Para garantir o bom funcionamento do sistema de computação em nuvem implementado, é necessário verificar o nível de consumo de memória RAM (Random Access Memory) nos servidores *Controller Node*, *Compute1* e *Compute2* e *Block-Storage Node*. Na Figura ?? são apresentados os resultados do teste de benchmarck de memória usand o Geekbench, sendo apresentado para cada servidor o espaço total de armazenamento, o espaço utilizado, o espaço compartilhado, o espaço livre, o espaço de buffer/cache e o espaço disponível no sistema de servidores.

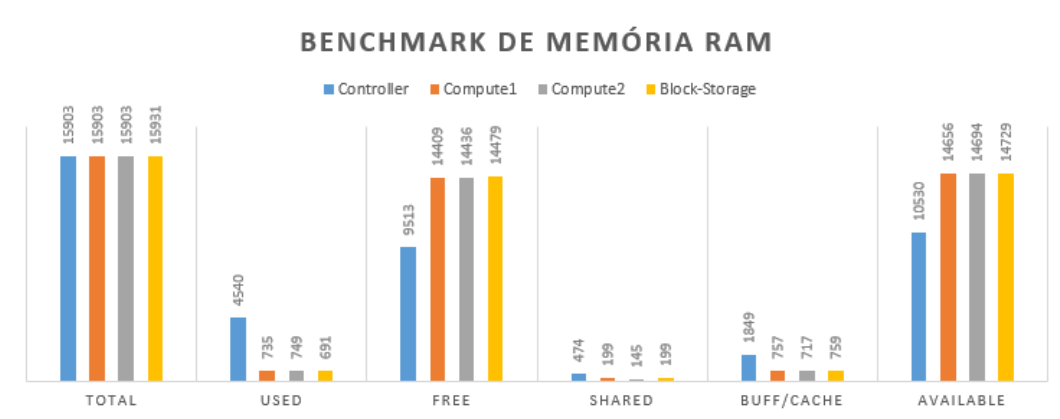


Figura 4.8: Benchmark de memória dos servidores físicos

## 4.4 Migração de Instâncias

A migração de uma instância num sistema desenvolvido na plataforma OpenStack, consiste em mover uma máquina virtual de um servidor para outro. O sistema de computação em nuvem desenvolvido neste trabalho é composto por dois servidores de computação, nomeadamente compute1 e compute2, que permitirão a migração de instâncias pelos nós de computação. O compute service é responsável pela criação e gestão das instâncias e também responsável pela migração das instâncias do compute2 para o compute1 e a seguir, do compute1 para o compute2. No back-end, o compute service é responsável pela transferência dos ficheiros existentes na instância. Existem dois tipos de migração de instâncias.

Na migração passiva, a instância precisa estar encerrada para ser transferida para o outro servidor. Na migração ativa, instância pode ser transferida para outro servidor, com o seu estado em execução, sem que o utilizador perceba, que a sua máquina virtual está a ser transferida para outro servidor físico. A migração passiva ou ativa ajuda a equilibrar a carga de trabalho entre os servidores de computação, ou tornar o servidor de computação livre para a manutenção. Neste trabalho será feita uma migração do tipo ativa e passiva, com volume, pois as instâncias têm como base de armazenamento volumes, e não necessita de armazenamento compartilhado para suportar a migração.

Para que a migração seja feita entre os nós de computação, é necessário que o utilizador **nova** tenha acesso remoto ao compute1 sem password via ssh com a autenticação de chave pública e os UID e GID dos utilizadores **nova** e **libvirt-qemu** sejam exactamente iguais nos dois nós de computação, como por exemplo **uid=128(nova) gid=132(nova) grupos=132(nova) 27(sudo) 130(libvirtd)**.

A migração passiva, será executada do servidor compute2 para o servidor compute1, conforme ilustra a Figura 4.9. As informações de Hardware e armazenamento das instâncias são apresentadas na Tabela 4.2. Neste tipo de Migração, as instâncias estão desligadas. A Figura 4.9 apresenta as instâncias criadas no sistema de computação em nuvem, e o tipo de sistema operativo instalado, os endereços IP normal e o IP flutuante, o par de chaves e o status da instância antes da migração.

Tabela 4.2: Detalhes de Hardware das Instâncias com mais Recursos

Instância	Tipo	VCPUs	RAM	Root Disk	Volume
1	m1.small	1	2 GB	20 GB	95 GB
2	m1.medium	2	4 GB	40 GB	200 GB
3	m1.large	4	8GB	80 GB	160 GB

Tabela 4.3: Detalhes de Hardware das Instâncias com menos Recursos

Instância	Tipo	VCPUs	RAM	Root Disk	Volume
1	m1.tiny	1	2 GB	1 GB	10 GB
2	m1.small	1	2 GB	4 GB	10 GB
3	m1.medium	1	2 GB	5 GB	10 GB

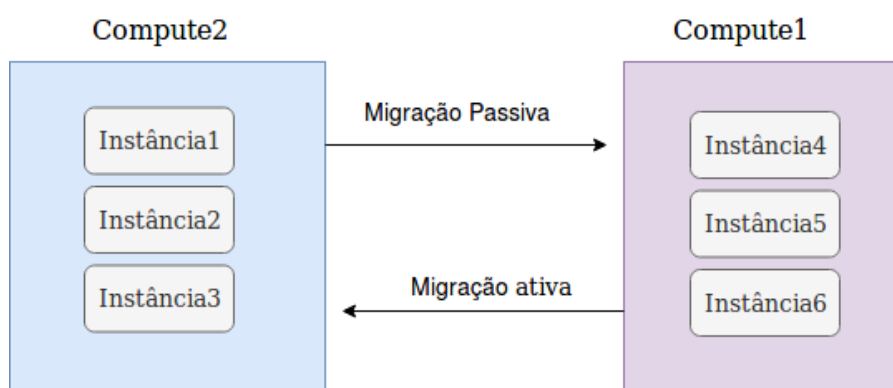


Figura 4.9: Migração

#### 4.4.1 Migração Passiva e Ativa

Na migração passiva, conforme ilustra a Tabela 4.2 as instâncias têm um sistema operativo instalado, têm volumes de armazenamento anexados, e estão desligadas para serem migradas para outro servidor.

Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State
instance3	CentOS	<ul style="list-style-type: none"> <li>192.168.1.9</li> </ul> Floating IPs: <ul style="list-style-type: none"> <li>10.0.5.18</li> </ul>	m1.large	key-pair	Shutoff	nova	None	Shut Down
instance2	Fedora-Workstation	<ul style="list-style-type: none"> <li>192.168.1.7</li> </ul> Floating IPs: <ul style="list-style-type: none"> <li>10.0.5.5</li> </ul>	m1.medium	key-pair	Shutoff	nova	None	Shut Down
instance1	ubuntu-14 desktop	<ul style="list-style-type: none"> <li>192.168.1.4</li> </ul> Floating IPs: <ul style="list-style-type: none"> <li>10.0.5.15</li> </ul>	m1.small	key-pair	Shutoff	nova	None	Shut Down

Figura 4.10: Detalhes Instâncias Passivas

Durante o processo de migração passiva, do servidor compute2 para o compute1. São apresentados nas figuras 4.11 e 4.12, os resultados da migração passiva. Nos resultados apresentados, é possível verificar o servidor de envio e o servidor de destino, o status da migração da instância, tempo inicial e o tempo final e o tipo de migração, que neste resultado é (migration). A Figura 4.11 apresenta o resultado da migração passiva do início ao meio e a Figura 4.12 apresenta o resultado da migração passiva do meio ao fim. O facto da Figura do resultado da migração pas-

siva ser muito longa, esta teve de ser dividida em duas partes nomeadamente (Migração Passiva (part1) e Migração Passiva (part2)).

Id	Source Node	Dest Node	Source Compute	Dest Compute	Dest Host	Status	Instance UUID
15	compute2	compute1	compute2	compute1	10.0.5.42	confirmed	acc07f0f-2e7b-4ff2-a0cb-1542377bba11
16	compute2	compute1	compute2	compute1	10.0.5.42	confirmed	343e341e-e4a0-4024-9e12-142ab62f43f1
17	compute2	compute1	compute2	compute1	10.0.5.42	confirmed	dd0c99ff-917c-4963-af0e-ecaf091100f7

Figura 4.11: Resultado da Migração Passiva (part1)

Old Flavor	New Flavor	Created At	Updated At	Type
2	2	2019-01-14T23:19:07.000000	2019-01-14T23:20:59.000000	migration
3	3	2019-01-14T23:22:38.000000	2019-01-14T23:24:59.000000	migration
4	4	2019-01-14T23:26:31.000000	2019-01-14T23:29:25.000000	migration

Figura 4.12: Resultado da Migração Passiva (part2)

O tempo de execução da migração passiva reflete-se no tempo em que se iniciou a migração apresentado na coluna (Created At) da Figura 4.12, e no tempo em que terminou a migração apresentado na coluna (Updated At) da Figura 4.12.

Na migração ativa, conforme ilustra a Tabela 4.3, primeiro é necessário partilhar o ficheiro de armazenamento das máquinas virtuais entre o servidor controlador e os servidores de computação. Os ficheiros são compartilhados através do servidor Network File System (NFS). A migração será feita do compute2 para o compute1.

Durante o processo de migração ativa, do servidor compute1 para o compute2. São apresentados nas figuras 4.13 e 4.14, os detalhes da migração ativa. Nos detalhes apresentados, é possível verificar o servidor de envio e o servidor de destino, o status da migração da instância, tempo inicial e o tempo final e o tipo de migração, que neste resultado é (live-migration).

A Figura 4.13 apresenta os detalhes da migração ativa do início ao meio e a Figura 4.14 apresenta os detalhes da migração ativa do meio ao fim. O facto da Figura dos detalhes da migração ativa ser muito longa, esta teve de ser dividida em duas partes nomeadamente (Migração Ativa (part1) e (Migração Ativa (part2)).

Id	Source Node	Dest Node	Source Compute	Dest Compute	Dest Host	Status	Instance UUID
15	-	-	compute1	compute2	-	completed	ca4d7e5c-b096-49c9-8cd5-6a787a632bc6
16	-	-	compute1	compute2	-	completed	e57ec5f8-c583-4072-8198-ca5195264766
19	-	-	compute1	compute2	-	completed	ca708961-ff86-4660-a6f9-200d8e55c743

Figura 4.13: Detalhes da Migração Ativa (part1)

Old Flavor	New Flavor	Created At	Updated At	Type
1	1	2019-01-23T00:08:33.000000	2019-01-23T00:08:59.000000	live-migration
6	6	2019-01-23T00:10:08.000000	2019-01-23T00:11:41.000000	live-migration
11	11	2019-01-23T00:50:54.000000	2019-01-23T00:53:52.000000	live-migration

Figura 4.14: Detalhes da Migração Ativa (part2)

O tempo de execução da migração ativa reflete-se no tempo em que se iniciou a migração apresentado na coluna (Created At) da Figura 4.14, e no tempo em que terminou a migração apresentado na coluna (Updated At) da Figura 4.14. A Figura 4.15, apresenta em forma de gráfico, o tempo de execução da migração das instâncias em segundos, com uma diferença de

tempo entre a infraestrutura de cada máquina virtual. Neste tipo de Migração as instâncias estão ligadas e desligadas como ilustrado na Figura 4.15.

A Figura 4.15, apresenta em forma de gráfico, o tempo de execução da migração das instâncias passivas e ativas em segundos, com uma diferença de tempo entre as instâncias com mais recursos (ver tabela 4.2 e as instâncias com menos recursos 4.3).

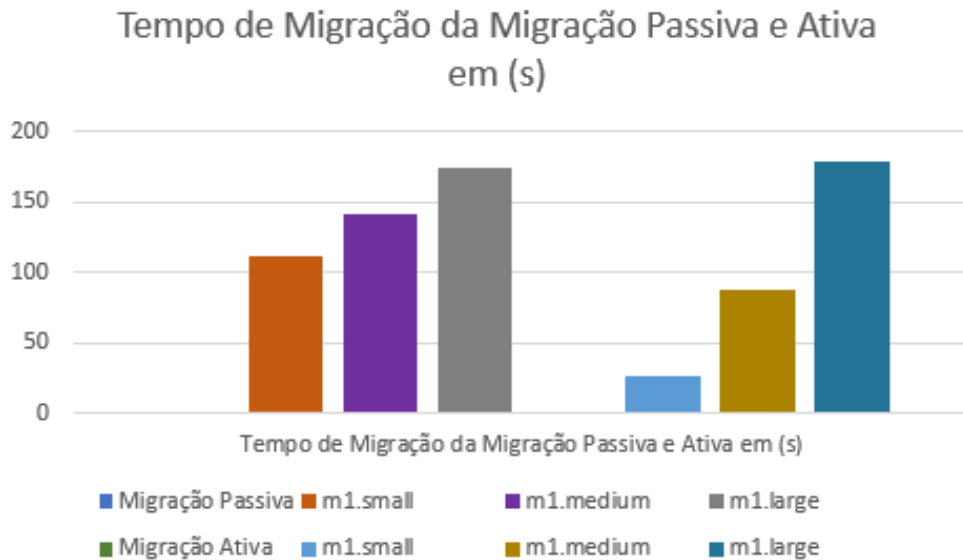


Figura 4.15: Instâncias Ativas e Passivas

#### 4.4.2 Comparação entre a Migração Passiva e a Migração Ativa

A primeira comparação da migração passiva e migração ativa será executada com as instâncias utilizadas na migração passiva, descritas na secção 4.4.1 cujo detalhe de hardware está listado na Tabela 4.2.

Id	Source Node	Dest Node	Source Compute	Dest Compute	Dest Host	Status	Instance UUID
22	-	-	compute1	compute2	-	error	52d75a6c-ec3a-4a6c-8b5c-96085879b6bb
23	-	-	compute1	compute2	-	error	ba4e1022-d83c-49fd-acc0-3ee575ae6723
24	-	-	compute1	compute2	-	error	e348862b-c612-4e38-9109-8457cb1b9437

Figura 4.16: Detalhes da Comparação da Migração Ativa e Passiva (part1)

Old Flavor	New Flavor	Created At	Updated At	Type
12	12	2019-01-24T18:57:15.000000	2019-01-24T19:24:07.000000	live-migration
13	13	2019-01-24T19:31:26.000000	2019-01-24T20:24:56.000000	live-migration
4	4	2019-01-24T22:14:23.000000	2019-01-25T00:01:11.000000	live-migration

Figura 4.17: Detalhes da Comparação da Migração Ativa e Passiva (part2)

Para o processo de comparação da migração passiva e migração ativa, do servidor compute1 para o compute2. São apresentados nas figuras 4.16 e 4.17, os detalhes desta comparação. Nos detalhes apresentados, é possível verificar o servidor de envio e o servidor de destino, o status da migração da instância, tempo inicial e o tempo final e o tipo de migração, que neste resultado é (live-migration), pois anteriormente na Secção 4.4.1, estão apresentados os detalhes

da migração passiva. A Figura 4.16 apresenta os detalhes da comparação da migração ativa e passiva do início ao meio e a Figura 4.17 apresenta os detalhes da comparação da migração ativa e passiva do meio ao fim. O facto da Figura dos detalhes da comparação da migração ativa e passiva ser muito longa, esta teve de ser dividida em duas partes nomeadamente (Detalhes da Comparação da Migração Ativa e Passiva (part1)) e (Detalhes da Comparação da Migração Ativa e Passiva (part2)).

O tempo de execução da tentativa da migração ativa e migração passiva, reflete-se no tempo em que se iniciou a migração apresentado na coluna (Created At) da Figura 4.16, e no tempo em que terminou a migração apresentado na coluna (Updated At) da Figura 4.17. A Figura 4.20, apresenta em forma de gráfico, o tempo de tentativa de execução da migração das instâncias em segundos, com uma diferença de tempo entre os detalhes de hardware de cada máquina virtual. Nos detalhes deste tipo de comparação, não foi possível concluir a a migração das instâncias, por insuficiência das capacidades de hardware dos servidores físicos, descritos na Tabela 3.1, pois as instâncias têm boas capacidades de hardware, conforme lista a Tabela 4.3, por esta razão o status da comparação da migração apresentado na Figura 4.16 é error para a execução que é automaticamente cancelada.

Continuando com o processo de comparação da migração passiva e migração ativa, serão utilizadas os mesmos requisitos de hardware das instâncias utilizadas para a migração ativa, cujo detalhe de hardware está listado na Tabela 4.2.

Id	Source Node	Dest Node	Source Compute	Dest Compute	Dest Host	Status	Instance UUID
26	compute2	compute1	compute2	compute1	10.0.5.42	confirmed	03640cce-1aeb-4319-b9cc-de86543b6a68
27	compute2	compute1	compute2	compute1	10.0.5.42	confirmed	734a1cbd-aa38-444e-9286-9cb0ba18447c
28	compute2	compute1	compute2	compute1	10.0.5.42	confirmed	0fb74e53-a2bf-404d-ae09-7297b21973da

Figura 4.18: Detalhes da Comparação da Migração Passiva Ativa (part1)

Old Flavor	New Flavor	Created At	Updated At	Type
1	1	2019-01-25T22:29:03.000000	2019-01-25T22:31:18.000000	migration
16	16	2019-01-25T22:31:54.000000	2019-01-25T22:35:14.000000	migration
17	17	2019-01-25T22:36:06.000000	2019-01-25T22:40:21.000000	migration

Figura 4.19: Detalhes da Comparação da Migração Passiva Ativa (part2)

O tempo de execução da comparação da migração ativa e migração passiva, reflete-se no tempo em que se iniciou a migração apresentado na coluna (Created At) da Figura 4.18, e no tempo em que terminou a migração apresentado na coluna (Updated At) da Figura 4.18. A Figura 4.20, apresenta em forma de gráfico, o tempo de execução da migração das instâncias em segundos, com uma diferença de tempo entre os detalhes de hardware de cada instância. Os detalhes de hardware das instâncias utilizadas nesta comparação, estão descritos na Tabela 4.3.

## Tempo de Execução da Tentativa da Migração Ativa e Passiva em (s)

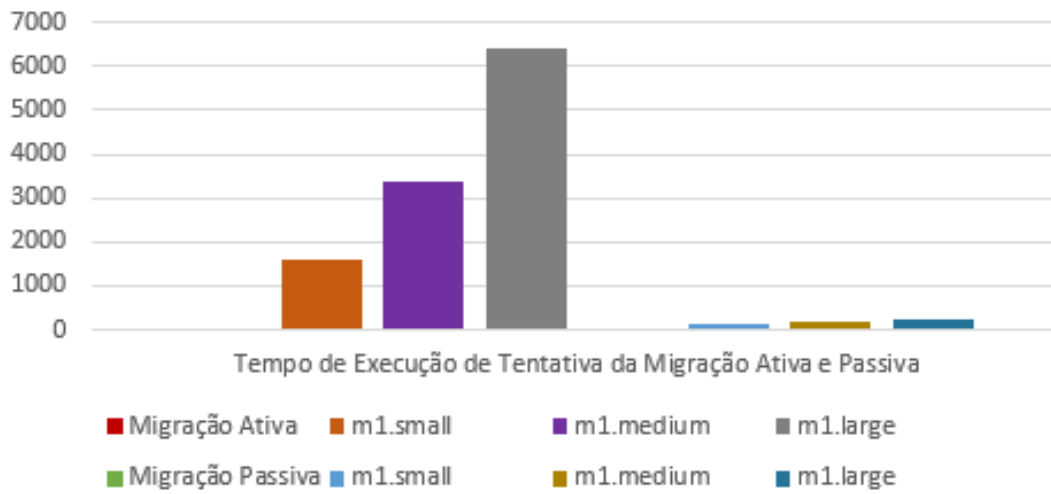


Figura 4.20: Tempo de Execução da Tentativa da Migração Ativa e Passiva

O sistema permite executar a migração passiva e migração ativa. A migração passiva pode ser executada sem restrições nos detalhes de hardware das instâncias, pois as instâncias estão desligadas. Já na migração ativa apenas é possível executar a migração das instâncias com menos recursos de hardware, cuja informação está listada na Tabela 4.3, permite que a migração seja executada e completa conforme ilustra a Figura 4.13.

# Capítulo 5

## Conclusões e Trabalhos Futuros

### 5.1 Principais Conclusões

Este trabalho tem como foco IaaS, que está na camada inferior dos serviços disponibilizados sendo um dos mais utilizados atualmente pelas empresas. Assim sendo, desenvolvemos uma infraestrutura de computação em nuvem, que fornece serviços de computação em nuvem com plataforma OpenStack.

Este é um sistema de computação em nuvem que permite aos utilizadores criar a sua própria rede privada, criar a sua própria máquina virtual com as características de hardware que melhor se adequam às suas necessidades, bem como o sistema operativo a utilizarem nas máquinas virtuais, assim como a possibilidade de aceder às máquinas virtuais fora do sistema com a ajuda do IP flutuante. O sistema é composto por dois servidores de computação, que possibilita a migração de máquinas virtuais, para reduzir a carga de trabalho e para possibilitar a manutenção intercalada dos servidores sem criar restrições aos utilizadores. Com estas funcionalidades garantimos um adequado funcionamento do sistema em tempo integral.

A análise de desempenho da migração passiva das instâncias, realizadas nesta investigação, estão ilustradas na Secção 4.4.1 Outros aspectos relevantes dos resultados obtidos nesta investigação, estão presentes no Capítulo 4

### 5.2 Trabalhos Futuros

O sistema de computação em nuvem desenvolvido neste trabalho, representa em um ambiente laboratorial, uma pequena infraestrutura como serviço de um centro de processamento de dados. O sistema pode ser expandido com a incorporação de mais equipamentos, para aumentar a capacidade do sistema em responder às solicitações relacionadas aos requisitos necessários para o fornecimento de mais serviços. A versão Newton da plataforma OpenStack é composta por um conjunto de serviços que podem ser implementados na IaaS desenvolvida nesta dissertação.

Alguns destes serviços não foram implementados por incompatibilidade do sistema operativo, das máquinas físicas (servidores), tais como: Shared File Systems service (manila) e Container Infrastructure Management service (magnum), pois nós utilizamos o *Ubuntu 16.04 LTS* e esses serviços só são compatíveis com o *Ubuntu 14.04 LTS* e anterior. Pelo que, no futuro trabalharemos com estas versões para implementar esses serviços.



## Bibliografia

- [1] Y. Sahu and R. Pateriya, "Cloud computing overview with load balancing techniques," *International Journal of Computer Applications*, vol. 65, no. 24, 2013. xv, 5, 7
- [2] S. R. and K. N., "A review on data privacy protection and types of attacks in cloud computing," *ARPN Journal of Engineering and Applied Sciences*, vol. 10, pp. 2675-2680, 2015. xv, 6, 9
- [3] M. F. Mushtaq, U. Akram, I. Khan, S. N. Khan, A. Shahzad, and A. Ullah, "Cloud computing environment and security challenges: A review," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, pp. 183-195, 2017. xv, xvii, 7, 12
- [4] C. El Amrani, K. B. Filali, K. B. Ahmed, A. T. Diallo, S. Telolahy, and T. El-Ghazawi, "A comparative study of cloud computing middleware," in *Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on*. IEEE, 2012, pp. 690-693. xv, 12, 14, 18
- [5] G. C. Obasuyi and A. Sari, "Security challenges of virtualization hypervisors in virtualized hardware environment," *International Journal of Communications, Network and System Sciences*, vol. 8, no. 07, p. 260, 2015. xv, 19
- [6] B. K. Adhikari, W. Zuo, and R. Maharjan, "A performance analysis of openstack cloud vs real system on hadoop clusters," in *Edge Computing (EDGE), 2017 IEEE International Conference on*. IEEE, 2017, pp. 194-201. xv, xvii, 7, 8, 25, 30
- [7] A. A. Shaikh, "Attacks on cloud computing and its countermeasures," in *Signal Processing, Communication, Power and Embedded System (SCOPEs), 2016 International Conference on*. IEEE, 2016, pp. 748-752. xvii, 7
- [8] L. M. Kaufman, "Data security in the world of cloud computing," *IEEE Security & Privacy*, vol. 7, no. 4, 2009. 5
- [9] T.-H. Hu, *A Prehistory of the Cloud*. MIT Press, 2015. 5
- [10] O. Foundation. (2010) Openstack overview and architecture, openstack junos, openstack sahara. Acesso em 24-10-2017, 27-11-2017, 19-03-2018, 02-05-2018. [Online]. Available: [www.openstack.org](http://www.openstack.org) 5, 18, 24, 29, 31, 33
- [11] R. Pacevič and A. Kačeniauskas, "The development of vislt visualization service in openstack cloud infrastructure," *Advances in Engineering Software*, vol. 103, pp. 46-56, 2017. 5, 29, 30, 31
- [12] B. Saraladevi, N. Pazhaniraja, P. V. Paul, M. S. Basha, and P. Dhavachelvan, "Big data and hadoop-a study in security perspective," *Procedia computer science*, vol. 50, pp. 596-601, 2015. 5
- [13] A. V. Toutov, A. S. Vorozhtsov, and N. V. Toutova, "Estimation of total migration time of virtual machines in cloud data centers," in *2018 IEEE International Conference "Quality Management, Transport and Information Security, Information Technologies"(IT&QM&IS)*. IEEE, 2018, pp. 389-393. 6

- [14] D. Kargatzis, S. Sotiriadis, and E. G. Petrakis, "Virtual machine migration in heterogeneous clouds: from openstack to vmware," in 2017 IEEE 38th Sarnoff Symposium. IEEE, 2017, pp. 1-6. 6
- [15] M. I. Biswas, G. Parr, S. McClean, P. Morrow, and B. Scotney, "An analysis of live migration in openstack using high speed optical network," in 2016 SAI Computing Conference (SAI). IEEE, 2016, pp. 1267-1272. 6
- [16] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation computer systems*, vol. 25, no. 6, pp. 599-616, 2009. 6
- [17] Z. Mahmood, "Cloud computing: Characteristics and deployment approaches," in *Computer and Information Technology (CIT)*, 2011 IEEE 11th International Conference on. IEEE, 2011, pp. 121-126. 6, 8, 9, 10, 17
- [18] A. Jain and R. Kumar, "A taxonomy of cloud computing," *International journal of scientific and research publications*, vol. 4, no. 7, pp. 1-5, 2014. 6
- [19] P. Mell, T. Grance et al., "The nist definition of cloud computing," 2011. 6, 10, 13, 14
- [20] A. P. Rachna Arora, "Secure user data in cloud computing using encryption algorithms," *International Journal of Engineering Research and Applications (IJERA)*, vol. 3, 2013. 6
- [21] S. Kumar and R. Goudar, "Cloud computing-research issues, challenges, architecture, platforms and applications: a survey," *International Journal of Future Computer and Communication*, vol. 1, no. 4, p. 356, 2012. 6, 17
- [22] O. Corporation). (1995-2018) Cloud computing and cloud services. Acesso em 23-02-2018. [Online]. Available: <http://www.oracle.com> 6, 11
- [23] I. Sheng Liang, IDG Communications. (2018) Requirements for building cloud infrastructure. Acesso em 12-05-2018. [Online]. Available: <https://www.cio.com/article/2412506/cloud-computing/7-requirements-for-building-your-cloud-infrastructure-htmlm> 7
- [24] Y. Jadeja and K. Modi, "Cloud computing-concepts, architecture and challenges," in *Computing, Electronics and Electrical Technologies (ICCEET)*, 2012 International Conference on. IEEE, 2012, pp. 877-880. 9, 10, 13
- [25] C. Gong, J. Liu, Q. Zhang, H. Chen, and Z. Gong, "The characteristics of cloud computing," in *Parallel Processing Workshops (ICPPW)*, 2010 39th International Conference on. IEEE, 2010, pp. 275-279. 9, 10, 13
- [26] D. R. P. S. Mr. Narahari Narasimhaiah, "Theory and features of saas (software as a service) for cloud computing," *International Research Journal of Engineering and Technology (IRJET)*, vol. 2, 2015. 9
- [27] W. Kim, "Cloud computing: Today and tomorrow." *Journal of object technology*, vol. 8, no. 1, pp. 65-72, 2009. 9
- [28] G. Von Laszewski, J. Diaz, F. Wang, and G. C. Fox, "Comparison of multiple cloud frameworks," in *Cloud Computing (CLOUD)*, 2012 IEEE 5th International Conference on. IEEE, 2012, pp. 734-741. 10

- [29] P. Jaiswal and A. Rohankar, "Infrastructure as a service: security issues in cloud computing," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 3, pp. 707-711, 2014. 10, 11, 16
- [30] A. Boukerche and E. Robson, "Vehicular cloud computing: Architectures, applications, and mobility," *Computer Networks*, 2018. 10
- [31] I. Corporation. (2003-2015) Infraestucture as a service overview. Acesso em 08-02-2018. [Online]. Available: [www.ibm.com](http://www.ibm.com) 10, 19
- [32] D. A. Fernandes, L. F. Soares, J. V. Gomes, M. M. Freire, and P. R. Inácio, "Security issues in cloud environments: a survey," *International Journal of Information Security*, vol. 13, no. 2, pp. 113-170, 2014. 11
- [33] M. Srilakshmi, C. Veenadhari, and I. K. Pradeep, "Deployment models of cloud computing: Challenges." *International Journal of Advanced Research in Computer Science*, vol. 4, no. 9, 2013. 11, 17
- [34] C. C. Rao and M. L. Y. R. Kumar, "Cloud: computing services and deployment models," *International Journal of Engineering and computer science*, vol. 2, no. 12, 2013. 11, 12
- [35] L. Savu, "Cloud computing: Deployment models, delivery models, risks and research challenges," in *Computer and Management (CAMAN), 2011 International Conference on*. IEEE, 2011, pp. 1-4. 11, 12, 33
- [36] S. Basu, A. Bardhan, K. Gupta, P. Saha, M. Pal, M. Bose, K. Basu, S. Chaudhury, and P. Sarkar, "Cloud computing security challenges & solutions-a survey," in *Computing and Communication Workshop and Conference (CCWC), 2018 IEEE 8th Annual*. IEEE, 2018, pp. 347-356. 12, 17, 19
- [37] S. M. I. Omer, A. Mustafa, and F. Alghali, "Comparative study between cluster, grid, utility, cloud and autonomic computing," *IOSR Journal of Electrical and Electronics Engineering*, vol. 9, no. 6, pp. 61-67, 2014. 12, 13, 14, 20
- [38] L. Sarga, "Cloud computing: an overview," *Journal of Systems Integration*, vol. 3, no. 4, p. 3, 2012. 13
- [39] C. R. B. Karimella Vikram, Dr. Soundara Rajan, "Cloud computing using elasticity," *International Journal of Computer Science And Technology*, vol. 7, 2016. 14, 15
- [40] M. M. Falatah and O. A. Batarfi, "Cloud scalability considerations," *International Journal of Computer Science and Engineering Survey*, vol. 5, no. 4, p. 37, 2014. 14, 15, 16
- [41] M. R. Nasarul Islam.K.V, "Analysis of various encryption algorithms in cloud computing," *International Journal of Computer Science and Mobile Computing*, vol. 6, pp. 90-97, 2017. 14
- [42] W. Ai, K. Li, S. Lan, F. Zhang, J. Mei, K. Li, and R. Buyya, "On elasticity measurement in cloud computing," *Scientific Programming*, vol. 2016, 2016. 15
- [43] B. Freese, "Defining a cloud ecosystem," *IBM GTS Cloud Advisory Services*, 2014. 16
- [44] D. Subramaniam.T.K, "Secutity attack issues and mitigation techniques in cloud computing environments," *International Journal of UbiComp (IJU)*, vol. 7, no. 1, 2016. 16, 17

- [45] I. RACKSPACE US. (2018) Cloud computing services. Acesso em 11-03-2018. [Online]. Available: <https://www.rackspace.com> 16
- [46] A. Apostu, F. Puican, G. Ularu, G. Suciuc, G. Todoran et al., "New classes of applications in the cloud. evaluating advantages and disadvantages of cloud computing for telemetry applications," Database Systems Journal, vol. 5, no. 1, pp. 3-14, 2014. 17
- [47] K. Chachapara and S. Bhadlawala, "Secure sharing with cryptography in cloud computing," in Engineering (NUiCONE), 2013 Nirma University International Conference on. IEEE, 2013, pp. 1-3. 17
- [48] P. Chouhan and R. Singh, "Security attacks on cloud computing with possible solution," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 6, no. 1, 2016. 17
- [49] A. Singh and D. M. Shrivastava, "Overview of attacks on cloud computing," International Journal of Engineering and Innovative Technology (IJEIT), vol. 1, no. 4, 2012. 17
- [50] G. Booth, A. Soknacki, and A. Somayaji, "Cloud security: Attacks and current defenses," in 8th Annual symposium on information Assurance (ASIA'13), 2013, pp. 4-5. 17
- [51] S. Sanka, C. Hota, and M. Rajarajan, "Secure data access in cloud computing," in Internet Multimedia Services Architecture and Application (IMSAA), 2010 IEEE 4th International Conference on. IEEE, 2010, pp. 1-6. 17
- [52] M. Khari, M. Kumar et al., "Secure data transference architecture for cloud computing using cryptography algorithms," in Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on. IEEE, 2016, pp. 2141-2146. 17
- [53] C. Moss, X. Dai, T.-T. Goh, S. Cheng, I. F. Odianonsen, O. Stephen, A. Anijesu, P. Sodomka, H. Klčov, J. Kříž et al., "Integrating cloud computing and mobile applications: A comparative study based on icloud and sanscode," 2014. 18
- [54] I. Red Hat. (2018) Cloud systems. Acesso em 27-02-2018. [Online]. Available: <https://www.redhat.com> 18, 29, 31
- [55] G. Nimbus. Nimbus overview and architecture. Acesso em 06-02-2018. [Online]. Available: [www.nimbusproject.org](http://www.nimbusproject.org) 18
- [56] M. P. Ltd. (2008-2013) Aneka overview and architecture. Acesso em 12-10-2017. [Online]. Available: <http://www.manjrasoft.com> 18
- [57] I. VMware. (2006-2009) VMware overview and architecture. Acesso em 22-01-2018. [Online]. Available: <https://www.vmware.com> 19, 20
- [58] A. Patel, M. Daftedar, M. Shalan, and M. W. El-Kharashi, "Embedded hypervisor xvvisor: A comparative analysis," in Parallel, Distributed and Network-Based Processing (PDP), 2015 23rd Euromicro International Conference on. IEEE, 2015, pp. 682-691. 19, 33
- [59] I. T. U. (ITU). (2018) Hypervisor overview. Acesso em 19-02-2018. [Online]. Available: <https://www.itu.int> 20
- [60] P. V. V. Reddy and L. Rajamani, "Evaluation of different hypervisors performance in the private cloud with sigar framework," International Journal of Advanced Computer Science and Applications, vol. 5, no. 2, 2014. 20, 21

- [61] M. Corporation). (2018) Cloud computing and cloud services. Acesso em 29-11-2017. [Online]. Available: [www.microsoft.com](http://www.microsoft.com) 20
- [62] H. Leppinen, "Current use of linux in spacecraft flight software," IEEE Aerospace and Electronic Systems Magazine, vol. 32, no. 10, pp. 4-13, 2017. 24
- [63] L. Liang, P. E. McKenney, D. Kroening, and T. Melham, "Verification of the tree-based hierarchical read-copy update in the linux kernel," arXiv preprint arXiv:1610.03052, 2016. 24
- [64] S. Arakawa, H. Miyakawa, T. Takine, and M. Murata, "Evolution of functional core in network-related function calls during linux kernel development," in Consumer Communications & Networking Conference (CCNC), 2018 15th IEEE Annual. IEEE, 2018, pp. 1-5. 24
- [65] C. L. Ubuntu and C. are registered trademarks of Canonical Ltd. (2018) Linux and ubuntu desktop. Acesso em 02-04-2018. [Online]. Available: <https://www.ubuntu.com/desktop> 25
- [66] S. Nalini and J. Andrews, "Mla scheme: Multi-level authentication for data in cloud using ntp-server and biometric," in Computational Intelligence and Computing Research (ICCIC), 2016 IEEE International Conference on. IEEE, 2016, pp. 1-3. 26
- [67] P. V. V. Reddy and L. Rajamani, "Performance comparison of different operating systems in the private cloud with kvm hypervisor using sigar framework," in Communication, Information & Computing Technology (ICCICT), 2015 International Conference on. IEEE, 2015, pp. 1-6. 33
- [68] A. Khandual, "Performance monitoring in linux kvm cloud environment," in Cloud Computing in Emerging Markets (CCEM), 2012 IEEE International Conference on. IEEE, 2012, pp. 1-6. 33