

Vítor Alano de Ataides

Um Estudo Sobre Computação em Nuvem e Plataformas de Código Aberto da Nuvem

Trabalho Individual apresentado ao Programa de Pós-Graduação em Computação da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Mestre em Ciência da Computação

Orientador: Prof. Dr. Maurício Lima Pilla
Coorientador: Prof. Dr. Laércio Lima Pilla

Pelotas, 2015

**Insira AQUI a ficha catalográfica
(solicitada na página da biblioteca)**

Dedico a todas pessoas que não me deram spoiler de Star Wars.

**Sua dissertação está muito fraca.
Você não escreveu nem 20 linhas.**

— PROFESSORA DE PORTUGUÊS DO ENSINO MÉDIO

RESUMO

ATAIDES, Vítor Alano de. **Um Estudo Sobre Computação em Nuvem e Plataformas de Código Aberto da Nuvem**. 2015. 31 f. Trabalho Individual (Mestrado em Ciência da Computação) – Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2015.

Com o rápido desenvolvimento das tecnologias de processamento e de armazenamento, recursos computacionais ficaram mais baratos, mais poderosos e mais ubiquoamente disponíveis. Esta tendência tecnológica permitiu a realização de um novo modelo computacional chamado Computação em Nuvem, no qual recursos são oferecidos como serviços de utilidade geral que podem ser alocados e desalocados por usuários pela internet sob demanda. A emergência da Computação em Nuvem causou um grande impacto nas indústrias de tecnologia da informação nos últimos anos.

Neste trabalho foi feito um estudo aprofundado sobre a Computação em Nuvem, onde foi estudado suas definições, principais características, os modelos de negócio, a arquitetura, os tipos de Nuvem e os principais desafios da área. Além desse estudo este trabalho também apresentou uma revisão sobre as Plataformas da Nuvem de Código Aberto, onde foi feito um estudo mais aprofundado sobre o software OpenStack. Sobre ele é apresentado seu histórico, sua arquitetura base e suas propriedades.

Palavras-chave: Palavrachave-um, palavrachave-dois, palavrachave-tres, palavrachave-quatro.

ABSTRACT

ATAIDES, Vítor Alano de. **Titulo do Trabalho em Ingles**. 2015. 31 f. Trabalho Individual (Mestrado em Ciência da Computação) – Programa de Pós-Graduação em Computação, Centro de Desenvolvimento Tecnológico, Universidade Federal de Pelotas, Pelotas, 2015.

With the rapid development of processing and storage technologies and the success of the Internet, computing resources have become cheaper, more powerful and more ubiquitously available than ever before. This technological trend has enabled the realization of a new computing model called cloud computing, in which resources are provided as general utilities that can be leased and released by users through the Internet in an on-demand fashion.

This paper presents a further study of Cloud Computing, its settings, main characteristics, business models, architecture, types of Cloud and the main challenges of the area. In addition to this study this work also presented a review of the Open Source Cloud Plataforms, where he made a further study on the OpenStack software. About OpenStack is presented its history, its base architecture and its properties.

Keywords: keyword-one, keyword-two, keyword-three, keyword-four.

LISTA DE FIGURAS

Figura 1	Modelo de negócio da computação em nuvem.	15
Figura 2	Arquitetura da Computação em Nuvem segundo (ZHANG; CHENG; BOUTABA, 2010)	15
Figura 3	Arquitetura da Computação em Nuvem segundo (HASSAN, 2011) .	17

LISTA DE TABELAS

Tabela 1	Comparação dos estudos sobre as plataformas da Nuvem	22
----------	--	----

LISTA DE ABREVIATURAS E SIGLAS

IaaS	Infrastructure-as-a-Service
PaaS	Plataform-as-a-Service
SaaS	Software-as-a-Service
DaaS	Data-as-a-Service
NIST	National Institute of Standards and Technology
APIs	Application Programming Interfaces
LBaaS	Load Balancing as a Service
VM	Virtual Machine
SLOs	Service Level Objectives

SUMÁRIO

1	INTRODUÇÃO	10
2	COMPUTAÇÃO EM NUVEM	12
2.1	Definição	12
2.2	Características	13
2.3	Modelo de Negócio	14
2.4	Arquitetura	15
2.5	Tipos de Nuvem	16
2.6	Desafios	18
2.6.1	Provisionamento Automático de Serviço	18
2.6.2	Migração de Máquinas Virtuais	18
2.6.3	Consolidação de Servidores	19
2.6.4	Gerenciamento de Energia	19
2.6.5	Segurança dos Dados	19
3	PLATAFORMAS DA NUVEM DE CÓDIGO ABERTO	21
3.1	OpenStack	21
3.1.1	Arquitetura Base	23
3.1.2	Propriedades	27
4	CONCLUSÃO	28
	REFERÊNCIAS	29

1 INTRODUÇÃO

A Computação em Nuvem, o sonho antigo da computação como commodity, tem o potencial de transformar grande parte de indústria da tecnologia da informação. Fazendo com que os softwares sejam ainda mais atrativos como serviço e transformando a maneira como os hardwares são projetados e comprados (ARMBRUST et al., 2009). Com a Computação em Nuvem desenvolvedores com idéias inovadoras não precisam mais de grande capital inicial para lançar suas idéias. Além disso, empresas com tarefas extremamente grandes podem coseguir seus resultados tão rápido quanto seus programas conseguem escalar, afinal usar 1000 servidores por uma custa o mesmo que 1 servidor por 1000 horas. Essa elasticidade de recursos, sem custo adicional, é revolucionária na história da tecnologia da informação (ARMBRUST et al., 2009).

A Computação em Nuvem refere-se tanto a aplicações entregues como serviços pela internet quanto a hardware e sistemas que fornecem esses serviços. Os serviços veem sendo chamados de *Software-as-a-Service* (SaaS). O hardware e softare do datacenter é o que é chamado de Nuvem. Quando uma Nuvem está disponível para o público geral, chamamos de Nuvem Pública e o serviço vendido de Utility Computing. Se uso o termo Nuvem Privada para referenciar a datacenters internos de empresas ou outras organizações não disponíveis para o público geral.

Além do modelo SaaS existe os modelos *Plataform-as-a-service* (PaaS) que visa oferecer plataformas de desenvolvimento como serviço e *Infraestructure-as-a-Service* (IaaS) que visa oferecer infraestrutura como serviço. Diversas soluções IaaS veem sendo desenvolvidas, muitas delas são plataformas de código aberto como: OpenStack, Nimbus, Eucalyptus e OpenNebula.

O objetivo deste trabalho é fazer um estudo sobre Computação em Nuvem e um estudo sobre Plataformas da Nuvem de Código Aberto.

Este trabalho esta dividido em 4 capítulos. O primeiro capítulo é a Introdução. No segundo capítulo é apresentado um estudo sobre Computação em Nuvem, seu histórico, definições, características, modelos de negócio, arquitetura, tipos de nuvem e desafios. No terceiro capítulo é apresentado um estudo sobre as plataformas da Nuvem de Código Aberto, onde é feito um estudo mais aprofundado sobre o software

OpenStack.

2 COMPUTAÇÃO EM NUVEM

Com o rápido desenvolvimento das tecnologias de processamento e de armazenamento, recursos computacionais ficaram mais baratos, mais poderosos e mais ubi-quoamente disponíveis. Esta tendência tecnológica permitiu a realização de um novo modelo computacional chamado Computação em Nuvem, no qual recursos são oferecidos como serviços de utilidade geral que podem ser alocados e desalocados por usuários pela internet sob demanda. A emergência da Computação em Nuvem causou um grande impacto nas indústrias de tecnologia da informação nos últimos anos, onde grandes empresas como Google, Amazon e Microsoft rivalizam para prover plataformas de nuvem mais potentes, confiáveis e economicamente eficientes, enquanto que as empresas de negócios procuram adequar seus modelos de negócio para ganhar benefícios desse novo paradigma.

Apesar do atual destaque à Computação em Nuvem, seu conceito não é novo e tão pouco suas tecnologias (HAMDAQA; TAHVILDARI, 2012). Em 1960 John McCarthy acreditava que um dia corporações conseguiriam vender recursos computacionais como “commodity” (HASSAN, 2011).

O objetivo deste capítulo é fazer um estudo sobre a Computação em Nuvem, suas características, definições, desafios e arquitetura. Ele é dividido em 6 seções. Na primeira 2.1 seção são apresentados as principais definições de computação em nuvem. Na segunda 2.2 seção são apresentadas suas principais características. Na terceira 2.3 seção, os modelos de negócio possibilitados pela nuvem são explicados. Na quarta 2.4 seção é feita uma análise da arquitetura da nuvem. Na quinta 2.5 seção é feita uma revisão dos os tipos de nuvem. Por fim, na sexta 2.6 seção os principais desafios da área são apresentados.

2.1 Definição

A Nuvem já foi definida de várias formas diferentes. Em (VAQUERO et al., 2008), 22 definições diferentes de Nuvem são estudadas e, a partir dessas definições, é proposta uma nova definição que contempla as demais:

“Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized SLAs (Service-level agreements).”

A NIST (National Institute of Standards and Technology) em “The NIST Definition of Cloud Computing” (2011) define a Computação em Nuvem abaixo:

“Cloud Computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e. g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models and four deployment models.”

As cinco características essenciais, os três modelos de serviço e os quatro tipos de nuvem citados acima são descritas nas seções seguintes.

2.2 Características

Em “The NIST Definition of Cloud Computing” (2011) é definido que a Computação em Nuvem é composta de cinco características essenciais. Essas características são descritas abaixo:

Autoatendimento sob demanda: Um cliente da nuvem pode alocar recursos como processamento e armazenamento, sem que para isso seja necessário qualquer interação humana.

Ampla acesso a rede: Os recursos da nuvem devem estar disponíveis através de uma rede (Internet, por exemplo) e podem ser acessados por mecanismos que permitam o acesso por uma gama heterogênea de plataformas como celulares, tablets e notebooks.

Pooling de recursos: Os recursos computacionais do provedor estão agrupados para servir a múltiplos consumidores, utilizando um modelo *multi-tenancy*, onde diferentes recursos físicos e virtuais são dinamicamente alocados e desalocados de acordo com a demanda do consumidor. Existe um senso de independência de localização que significa que o consumidor não tem controle nem conhecimento da localização exata dos recursos alocados.

Elasticidade rápida: Os recursos podem ser elasticamente provisionados, ou seja, quando é necessário mais recursos eles são alocados de forma automática e quando desnecessários esses recursos são desalocados, dando a ideia de que os recursos são ilimitados.

Serviços mensurados: A nuvem deve ter a capacidade de controlar e otimizar a utilização dos recursos. Dessa forma a utilização dos recursos pode ser monitorada, controlada e reportada, tornando seu uso transparente tanto para o provedor quanto para o consumidor.

2.3 Modelo de Negócio

A computação em nuvem implementa um modelo de negócio orientado a serviços. Em outras palavras, os recursos de hardware e recursos a nível de plataforma são oferecidos como serviços sob demanda. De acordo com Zhang et al. (2010) os serviços oferecidos pelas nuvens podem ser agrupados em três categorias: *Infrastructure as a Service* (IaaS), *Platform as a Service* (PaaS) e *Software as a Service* (SaaS).

Infrastructure as a Service: IaaS refere-se ao provisionamento sob demanda dos recursos de infraestrutura, normalmente máquinas virtuais (*virtual machines*, VMs). O dono de uma nuvem que oferece IaaS é chamado de provedor IaaS. Amazon EC2 (AMAZON, 2015), GoGrid (GOGRID, 2015) e FlexiScale (AGUIAR, 2012) e (SMARTERTOOLS, 2015) são exemplos de provedores IaaS.

Platform as a Service: PaaS refere-se ao provisionamento de recursos de plataforma, como sistemas operacionais e frameworks de desenvolvimento de software. Exemplos de provedores PaaS são: Google App Engine (GOOGLE, 2015), Microsoft Windows Azure (MICROSOFT, 2015) e Salesforce (SALESFORCE, 2015).

Software as a Service: SaaS refere-se ao provisionamento de aplicações sob demanda. Exemplos de provedores SaaS incluem: Rackspace (RACKSPACE, 2015) e SAP Business By Design (SAP, 2015).

O modelo de negócio da computação em nuvem é apresentado na Figura 1. De acordo com a arquitetura em camadas da nuvem é possível que um provedor PaaS funcione sobre um provedor IaaS. Porém o que é muito comum atualmente é que a mesma organização seja o provedor PaaS e IaaS, como a Google e a SalesForce.com. Por esse motivo, provedores PaaS e IaaS são frequentemente chamados de provedores de infraestrutura ou provedores de nuvem.

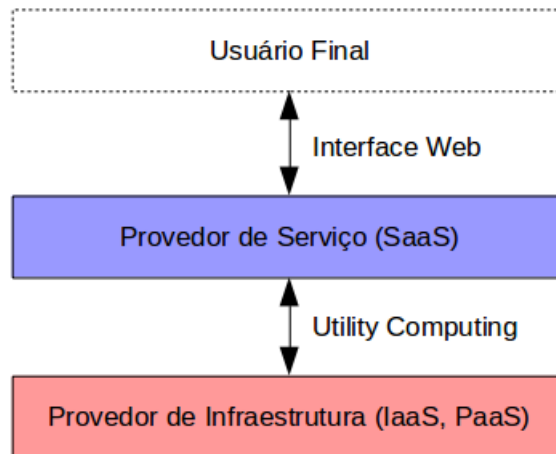


Figura 1: Modelo de negócio da computação em nuvem.

2.4 Arquitetura

Para Zhang et al. (2010) a arquitetura da computação em Nuvem é dividida em quatro camadas: a camada de hardware, a camada de infraestrutura, a camada de plataforma e a camada de aplicação, como apresentado na Figura 2. As camadas são detalhadas abaixo:

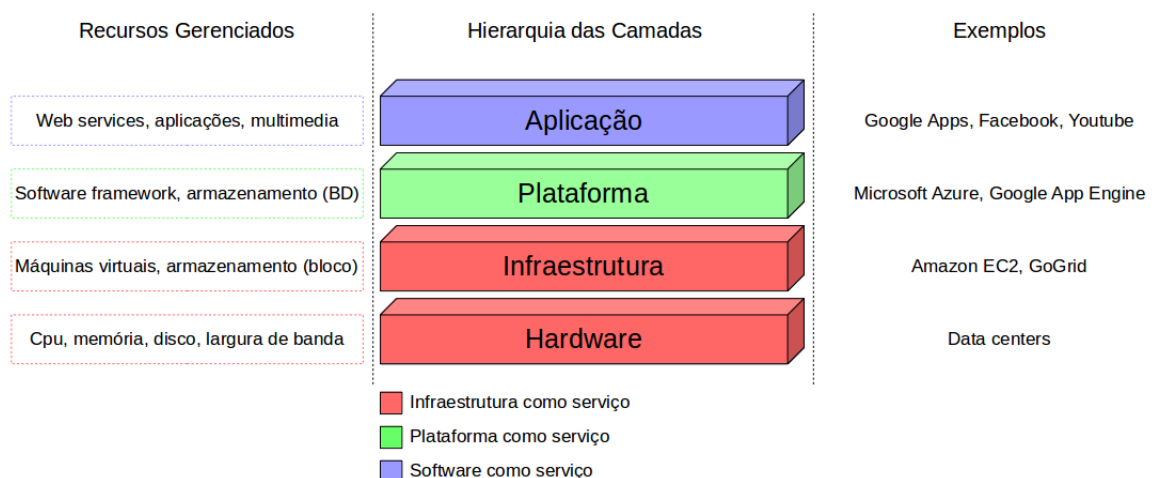


Figura 2: Arquitetura da Computação em Nuvem segundo (ZHANG; CHENG; BOU-TABA, 2010)

Camada de Hardware: essa camada é responsável pela gestão dos recursos físicos, incluindo servidores, roteadores, switches, sistemas de resfriamento e energia. Na prática, a camada de hardware é tipicamente implementada nos data centers. Um data center normalmente contém milhares de servidores organizados em racks e interconectados através de switches ou roteadores. Problemas típicos na camada de hardware envolvem configuração de hardware, tolerância a fa-

lhas, gerenciamento do tráfego, gerenciamento de energia e gerenciamento dos recursos de resfriamento.

Camada de Infraestrutura: também conhecida como camada de virtualização, a camada de infraestrutura cria uma pool de armazenamento e recursos computacionais através do particionamento dos recursos físicos utilizando tecnologias de virtualização como Xen (CITRIX, 2015), KVM (LINUX-KVM, 2015) e VMware (VMWARE, 2015). Como as principais características da computação em nuvem, como alocação dinâmica de recursos, só são possíveis por causa das tecnologias de virtualização, a camada de infraestrutura é considerada um componente essencial na arquitetura.

Camada de Plataforma: construída sobre a camada de infraestrutura, a camada de plataforma consiste em sistemas operacionais e frameworks. O propósito da camada de plataforma é facilitar o lançamento de aplicações em máquinas virtuais. Por exemplo, o Google App Engine opera na camada de plataforma provendo uma API com suporte para armazenamento e lógica de negócio para aplicações web típicas.

Camada de Aplicação: no nível mais alto da hierarquia, a camada de aplicação consiste nas aplicações da nuvem. Diferentes das aplicações tradicionais, as aplicações da nuvem fazem uso do escalonamento automático disponível na Nuvem para alcançar um melhor desempenho, disponibilidade e baixo custo de operação.

Comparado aos ambientes de hospedagem de serviços tradicionais como servidores dedicados, a arquitetura da nuvem é mais modular. Cada camada é fracamente acoplada, com camadas acima e abaixo, permitindo que cada camada evolua separadamente. Essa modularidade permite que a computação em nuvem dê suporte a uma gama de requerimentos das aplicações enquanto reduz o sobrecusto de manutenção e de administração.

Para Hassan et al. (2011) a arquitetura também é dividida em quatro camadas com uma hierarquia baseada na abstração, porém a arquitetura é definida diretamente ligada aos modelos de negócio, como pode ser observado na Figura 3. Nessa definição é adicionando um novo modelo de negócio: dados como serviço, que é descrito como o serviço que oferece base de dados para armazenamento das informações do cliente.

2.5 Tipos de Nuvem

Zhang et al. (2010) e Mell et al. (2011) descrevem 4 tipos de nuvem: Nuvem Pública, Nuvem Privada, Nuvem Híbrida e Nuvem Comunitária.

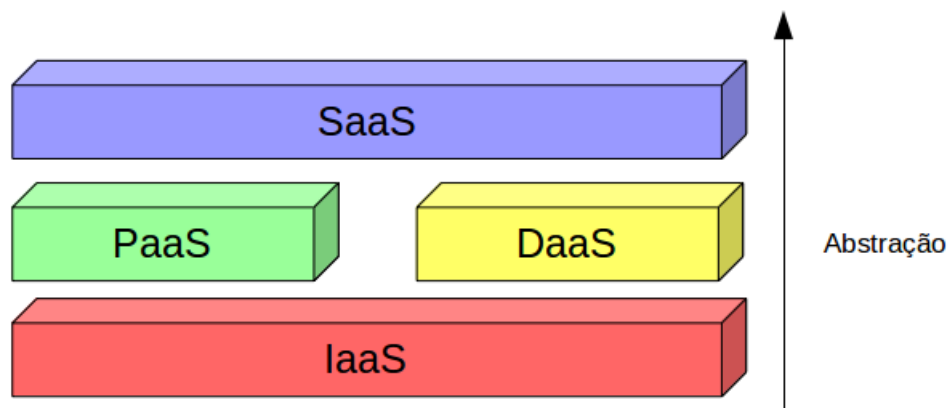


Figura 3: Arquitetura da Computação em Nuvem segundo (HASSAN, 2011)

Nuvem Pública: nuvem onde os provedores de serviços oferecem seus recursos ao público geral. Nuvens públicas oferecem vários benefícios aos provedores de serviço: por exemplo, não é necessário para os provedores investimentos iniciais em infraestrutura. No entanto, as nuvens públicas oferecem um baixo grau de controle sobre os dados, a rede e as configurações de segurança.

Nuvem Privada: também conhecida como nuvem interna, as nuvens privadas são para uso exclusivo de uma organização. A nuvem privada pode ser construída e gerenciada pela própria empresa ou por provedores externos. Esse tipo de nuvem oferece um maior controle de desempenho, segurança e confiabilidade. No entanto, as nuvens privadas são criticadas por serem muito similares aos servidores tradicionais.

Nuvem Híbrida: uma nuvem híbrida é a combinação da nuvem privada e da nuvem pública, na tentativa de minimizar as limitações das duas abordagens. Nesse tipo de nuvem, parte dos serviços de infraestrutura estão na nuvem privada enquanto que as outras partes são executadas em uma nuvem pública. Nuvens híbridas oferecem mais flexibilidade que as nuvens públicas e do que as nuvens privadas. Elas oferecem mais controle sobre dados, rede e segurança do que as nuvens públicas mantendo as facilidades de expansão.

Nuvem Comunitária: a nuvem comunitária funciona praticamente como uma nuvem privada que pode ser utilizada por duas ou mais organizações. Ela pode ser gerenciada tanto por uma organização, pelas duas organizações, ou por um provedor.

2.6 Desafios

Apesar da Computação em Nuvem estar amplamente presente na indústria, as pesquisas nessa área ainda estão em fase inicial (ZHANG; CHENG; BOUTABA, 2010). Muitos problemas existentes ainda não foram resolvidos, enquanto novos desafios continuam surgindo das aplicações nas indústrias. Esta seção apresenta um resumo dos principais desafios da Computação em Nuvem.

2.6.1 Provisionamento Automático de Serviço

Uma das principais características da Computação em Nuvem é a capacidade de adquirir e liberar recursos sob demanda. O objetivo de um provedor de serviço neste caso é de alocar e desalocar recursos da nuvem a fim de satisfazer os Service Level Objectives (SLOs), enquanto minimiza o custo operacional. No entanto, não é óbvio como o provedor de serviço alcançará esse objetivo. Em particular, não é fácil de determinar como mapear SLOs tais como requisitos de QoS para requisitos dos recursos de baixo nível como requisitos de CPU e de memória. Além disso, para alcançar alta agilidade para responder as flutuações de demanda, as decisões de provisionamento devem ser feitas em tempo real.

2.6.2 Migração de Máquinas Virtuais

A virtualização é uma das principais tecnologias utilizadas em infraestruturas em nuvem. Através do uso da virtualização, é possível compartilhar a mesma máquina física com múltiplos sistemas operacionais e/ou aplicações de usuários finais, promovendo o isolamento entre os mesmos. Da mesma forma, o seu uso torna possível realizar balanceamento de carga entre estruturas físicas, através de ações como a migração de máquinas virtuais entre servidores.

A migração de VMs evoluiu das técnicas de migração de processos (OSMAN et al., 2002). Algumas ferramentas já implementam o conceito de live-migration, onde as VMs são movidas em um processo que envolve paradas extremamente curtas que variam de dezenas de milissegundos a um segundo. Clark et al. (2005) mostra que a migração de um SO inteiro e todas suas aplicações como uma unidade permite que muitas das dificuldades encontradas nas abordagens de migração de processos sejam evitadas.

O maior benefício da migração de VMs é a capacidade de evitar a sobrecarga de pontos específicos da infraestrutura de servidores. No entanto, essa não é uma tarefa simples. Atualmente, a detecção de pontos de sobrecarga e o disparo de uma migração não tem agilidade suficiente para responder às mudanças de carga de trabalho repentinas (ZHANG; CHENG; BOUTABA, 2010). Além disso, é necessário transferir o estado dos dados em memória de forma consistente e eficiente, considerando

os recursos para as aplicações e para os servidores físicos.

2.6.3 Consolidação de Servidores

A consolidação de servidores é uma abordagem que tem como objetivo maximizar a utilização de recursos enquanto minimiza o consumo de energia. A técnica de migração de VMs é frequentemente utilizada na consolidação de servidores, onde VMs são movidas de múltiplos servidores subutilizados para um servidor. Dessa forma os servidores restantes são colocados no estado de economia de energia. O problema de consolidar servidores de forma ótima pode ser visto como uma variação do problema bin-packing (CHEKURI; KHANNA, 1999), que é um problema de otimização NP-difícil.

2.6.4 Gerenciamento de Energia

Atingir uma maior eficiência no consumo de energia também é um dos principais desafios da computação em nuvem. Hamilton et al. (2006) estima que o custo de alimentação energética e refrigeração somam 53 por cento dos gastos operacionais dos data centers. Em 2006, os data centers nos Estados Unidos consumiram mais de 1.5 por cento de toda energia gerada naquele ano, e o crescimento dessa porcentagem tem projeção de crescimento de 18 por cento ao ano. Isso leva a uma enorme pressão sobre os provedores de infraestrutura para a redução do consumo energético. O objetivo não é apenas reduzir o gasto energético, mas também se adequar aos regulamentos do governo e as normas ambientais.

2.6.5 Segurança dos Dados

A segurança dos dados é outro tópico de pesquisa muito importante na computação em nuvem. Visto que os provedores dos serviços tipicamente não têm acesso aos sistemas de segurança dos data centers, eles dependem do provedor de infraestrutura para garantir a segurança de seus dados. Os provedores de infraestrutura devem garantir:

Confidencialidade para garantir a segurança de acesso e transferência de dados.

Auditabilidade para atestar se as configurações de segurança de uma aplicação foram modificadas ou não.

Confidencialidade é normalmente alcançada utilizando protocolos de criptografia, enquanto que a auditabilidade é alcançada utilizando técnicas de atestado remoto. Atestados remotos tipicamente necessitam de um trusted platform module (TPM) para gerar um sumário do sistema infalsificável como prova da segurança do sistema. Porém num ambiente como a nuvem onde as VMs podem migrar dinamicamente de um servidor para outro, utilizar atestado remoto não é o suficiente. É necessário que

seja criado um mecanismo confiável em cada camada da arquitetura. Inicialmente, para ser confiável, a camada de hardware deve usar um TPM. A camada de infraestrutura, responsável pela migração das VMs, deve utilizar monitores de VMs confiáveis. A migração de VMs só deve ser permitida se tanto o servidor de destino quanto fonte são confiáveis.

3 PLATAFORMAS DA NUVEM DE CÓDIGO ABERTO

As Plataformas da Nuvem de Código Aberto foram criadas com a intenção de suprir a necessidade de que as soluções IaaS forneçam privacidade e controle sobre os ambientes virtualizados (A.; DINIZ; S., 2015). Portanto essas plataformas eram usadas principalmente para contruir Nuvens privadas. Atualmente elas são utilizadas para Nuvens públicas, privadas e híbridas. Com o aparecimento de diferentes Plataformas da Nuvem de Código Aberto, mesmo tendo as características específicas de cada plataforma, escolher a mais adequada é uma tarefa complicada (A.; DINIZ; S., 2015).

Muitos trabalhos foram desenvolvidos com objetivos de comparar as diferentes soluções para IaaS. A tabela 1 apresenta os trabalhos que comparam as soluções para nuvem, destacando as soluções analisadas, suas limitações e foco.

Alguns trabalhos analisaram algumas soluções mais profundamente considerando um ponto de vista específico. Amrani et. al. (2012) foca na aparência e nas features, enquanto que Laszewski et. al. (2012) foca em comparar a escalabilidade das plataformas e Cordeiro et. al. (2010) nas placement policies, arquitetura das plataformas e na estrutura de rede das plataformas analisadas. Baser (2012) detalha sobre o OpenStack e o CloudStack, porém apenas o OpenStack é analisado profundamente.

Esse capítulo tem como objetivo fazer um estudo sobre as soluções de código aberto para IaaS, onde será feito um estudo mais aprofundado sobre o software OpenStack.

3.1 OpenStack

OpenStack é um software pra Nuvem que oferece a capacidade de controlar uma enorme quantidade de recursos de computação, de rede e de armazenamento. Ele ainda prove ao usuário recursos sob demanda (SOFTWARE, 2015). O desenvolvimento do OpenStack começou em 2010, inicialmente sendo desenvolvido pela Rackspace Hosting e pela NASA (OPENSTACK, 2015) com o objetivo de prover uma solução de código aberto para construção de Nuvens privadas. A missão do OpenStack é per-

Tabela 1: Comparação dos estudos sobre as plataformas da Nuvem

Estudo	Plataformas Analisadas	Foco/Limitação
Voras et. al. (2011)	Open Nebula, Eucalyptus, Ubuntu Enterprise Cloud, openQRM, Abiquo, Red Hat Cloud Foundation, OpenStack, Nimbus e mOSAIC	Breve introdução das plataformas, nenhuma comparação fornecida
Zeng et. al. (2012)	Amazon EC2, IBM smart cloud, Google App Engine, Windows Azure, Hadoop, Eucalyptus, OpenNebula e Nimbus	Breve introdução das plataformas, nenhuma comparação fornecida
Endo et. al. (2010)	XCP, Nimbus, OpenNebula, Eucalyptus, TPlataform, ECP (Enomals Elastic Computing) e Apaches VCL	Breve introdução das plataformas e breve comparação fornecida
Amrani et. al. (2012)	Eucalyptus, OpenNebula e Nimbus	Limitado às novas características e à aparência
Von Laszewski et. al. (2012)	Eucalyptus, OpenNebula, Nimbus e OpenStack	Foco em escalabilidade
Cordeiro et. al. (2010)	XCP, Eucalyptus e OpenNebula	Foco na arquitetura das plataformas e na rede
Baset (2012)	OpenStack e CloudStack	Detalhes aprofundados apenas para o OpenStack

mitir que qualquer organização crie e ofereça serviços de Computação em Nuvem. Disponível como uma solução de código aberto, o OpenStack foi construído com 4 princípios em mente:

Open Source: todo código é publicado sob a licença Apache 2.0, permitindo a comunidade a utilizá-lo livremente.

Open Design: a cada 6 meses a comunidade de desenvolvimento realiza um *design summit* para recolher requerimentos e escrever especificações para as próximas versões.

Open Development: durante toda fase de desenvolvimento de uma nova versão é mantido publicamente em um repositório o código fonte.

Open Community: manter as comunidades de desenvolvimento e de usuários engajadas através de um processo aberto e transparente.

A primeira versão do OpenStack se chamava Austin e foi lançada em outubro de 2010. Desde então o OpenStack adotou uma política de realizar 2 lançamentos grandes por ano, totalizando 12 lançamentos até hoje (o próximo será o Mikata, esperado para fevereiro de 2016). O versão Austin continha apenas o componente de armazenamento de objetos (Swift) e o componente de computação (Nova) e apresentava restrições importantes como suporte limitado a objetos de 5GB. O suporte a objetos maiores foi introduzido na segunda versão, chamado Bexar, junto com o serviço de registro de imagem e o serviço de entrega (Glance). Após isso, novos módulos foram introduzidos apenas na quinta versão, chamada de Essex, que introduziu uma interface gráfica para o usuário (Horizon) e um componente de segurança (Keystone). A sexta versão, Folsom, trouxe a rede como um dos projetos principais do OpenStack (antigamente chamado de Quantum, atualmente chamado de Neutron) e um componente de armazenamento de bloco (Cinder). Folsom foi a primeira versão a incorporar os três mais importantes módulos do OpenStack: Swift, Nova e Neutron. A partir dessa versão os módulos existentes foram fixados e melhorados e alguns serviços compartilhados novos foram adicionados como: serviço de telemetria (Ceilometer) e serviço de orquestração (Heat), foram lançados na oitava versão, Havana. Na nona versão, Icehouse, foi lançado um serviço de banco de dados (Trove). O serviço de processamento de dados (Sahara) foi lançado na décima versão, Juno. Na décima primeira versão, Kilo, foi introduzido o serviço de *bare metal* (Ironic). Na mais nova versão, Liberty, foi lançado um serviço de busca (Searchlight).

3.1.1 Arquitetura Base

Como em qualquer outra plataforma para Nuvem, a infraestrutura por baixo do OpenStack é formada por hardware padrão, que pode conter qualquer peça de dis-

positivos físicos como servidores, discos ou dispositivos de rede. No intuito de prover serviços na nuvem, o OpenStack desenvolveu camadas de virtualização, promovendo uma abstração da infraestrutura física para o usuário final. Essas camadas de virtualização são desenvolvidas numa arquitetura multicomponente.

A arquitetura do OpenStack consiste em 3 componentes principais: computação (Nova), rede (Neutron) e armazenamento (Swift). Além desses 3 componentes, o OpenStack vem desenvolvendo muitos outros serviços, projetados para trabalharem juntos para prover uma solução IaaS completa. A integração desses serviços é facilitada através de APIs (*application programming interfaces*) que são fornecidas por cada serviço (OPENSTACK, 2015).

3.1.1.1 Computação (Nova)

O componente de computação, de codinome Nova e escrito em Python, é responsável por gerenciar grandes redes de VMs e eventualmente por escalonar as VMs entre as máquinas físicas disponíveis (OPENSTACK, 2015). O Nova é uma aplicação distribuída que consiste em 6 componentes: Nova-api, Message Queue, Nova-Compute, Nova-Network, Nova-Volume e Nova-Scheduler. O Nova suporta o ciclo-de-vida completo de uma instância na Nuvem, desde a requisição para inicializar uma VM até seu término. Os seis componentes são detalhados abaixo:

Nova-api: aceita e responde as chamadas do usuário de computação do usuário final pela API. Além de prover sua própria API, o Nova-api é compatível com a API da Amazon EC2, oferecendo o potencial de integração com os serviços de Nuvem da Amazon. Esse componente lida com a organização das atividades como: executar uma instância e aplicação de políticas de execução.

Nova-compute: é responsável por criar e terminar as instâncias de VM através das APIs do hypervisor. OpenStack suporta vários hypervisors capacidade de aceitar outros hypervisor através de sua biblioteca padrão.

Nova-volume: gerencia a criação, a ligação e a desligação de volumes persistentes para instâncias de computação. Existem dois tipos de armazenamentos suportados por uma VM: (1) Armazenamento Efêmero, que é associada com uma única instância. Um bloco de armazenamento efêmero é conectado ao ciclo-de-vida de uma instância, quando essa instância terminar os dados são deletados; (2) Armazenamento de Volume: é persistente e independente de qualquer instância particular.

Nova-network: responsável por todas tarefas relacionadas a rede. Tarefas como trocar as regras da tabela de ips ou setar as interlaces de *bridge*.

Nova-schedule: responsável pelo escalonamento de VMs entre as máquinas físicas. Enquanto os algoritmos de escalonamento podem ser definidos pelo usuário, o Nova-schedule suporta por padrão 3 algoritmos: (1) Simple: tenta encontrar o *host* mais livre, (2) Chance: escolhe um *host* aleatório disponível da tabela de serviço, (3) Zone: escolhe um *host* aleatório em uma zona disponível. Ao permitir que os usuários definam seus próprios algoritmos de balanceamento faz desse componente muito importante para a construção de um sistema tolerante a falhas e com balanceamento de carga.

Queue: é uma central para a passagem de mensagens entre os componentes. É normalmente implementado com RabbitMQ, mas suporta outros tipos de fila de mensagens.

Database: armazena a maior parte do estado de execução e do estado de construção da infraestrutura da Nuvem. Por exemplo, o Database fornece informação das instancias que estão disponíveis para uso e em uso, redes disponíveis e informação de armazenamento. O Nova, teoricamente, suporta qualquer banco de dados baseado em SQL, mas os bancos de dados mais usados são sqlite3, MySQL e PostgreSQL.

Todos os componentes da arquitetura do OpenStack seguem uma política *shared-nothing* baseada em mensagens. *Shared-nothing* significa que cada componente de cada grupo de componentes pode ser instalado em qualquer servidor, de maneira distribuida, enquanto que ser baseado em mensagens garante que a comunicação entre todos os componentes seja realizada através de uma fila de mensagens.

3.1.1.2 Rede (Neutron)

O componente de rede de qualquer plataforma na Nuvem tem funções importantes: (1) oferecer acessibilidade aos recursos e serviços, (2) prover a ligação dos endereços entre diferentes serviços e (3) configurar automaticamente a rede.

A arquitetura do Neutron consiste em quatro redes físicas distintas:

Rede de Gerenciamento: usada para comunicação interna entre os componentes do OpenStack.

Rede de Dados: usada para comunicação com VMS relacionada a dados na Nuvem.

Rede Externa: usada para prover acesso a Internet à VMs.

Rede da API: expõe todas APIs do OpenStack.

3.1.1.3 Armazenamento

O componente de armazenamento tem como finalidade gerenciar os recursos armazenados. OpenStack tem suporte tanto para Armazenamento de Objetos quanto para Armazenamento de Bloco.

O Swift, componente responsável pelo armazenamento de objetos, provê uma API de acesso a plataforma que pode ser integrada diretamente com aplicações ou utilizada para *backup*, arquivamento e retenção de dados (OPENSTACK, 2015). No armazenamento de objetos os dados são salvos em múltiplos dispositivos de hardware, o OpenStack fica responsável por garantir a replicação dos dados e a integridade entre os *clusters*.

O armazenamento de bloco é responsabilidade do componente chamado de Cinder. Por gerenciar os recursos de armazenamento em blocos, o armazenamento em bloco é apropriado para cenários que precisam de desempenho como armazenamento de banco de dados.

3.1.1.4 Interface Gráfica (Horizon)

O OpenStack fornece uma interface gráfica tanto para usuários quanto para administradores, capaz de controlar os recursos de computação, armazenamento e rede. Através do Horizon os administradores podem gerenciar os usuários e os limites de acesso para cada usuário.

3.1.1.5 Serviços Compartilhados

Os Serviços Compartilhados do OpenStack é um conjunto de serviços presentes nos 3 principais módulos do OpenStack com o objetivo de facilitar as operações de gerenciamento da Nuvem. Serviços como de identidade, de imagem, de telemetria, de orquestração e de banco de dados (OPENSTACK, 2015):

Serviço de identidade (Keystone): é o serviço de segurança, usado para proteger o acesso e uso dos recursos.

Serviço de Imagem (Glance): é o repositório para discos virtuais e das imagens usadas pelas VMs.

Serviço de Telemetria (Ceilometer): capaz de analisar a utilização dos recursos e a performance dos serviços do OpenStack.

Serviço de Orquestração (Heat): permite que os desenvolvedores de aplicações descrevam e automatizem o lançamento de infraestruturas na Nuvem.

Serviço de Banco de Dados (Trove): permite que usuários utilizem as características dos banco de dados relacionais. Usuários e administradores de banco

de dados podem criar e gerenciar múltiplas instâncias de banco de dados.

3.1.2 Propriedades

O OpenStack é construído seguindo uma filosofia livre: no sentido de evitar estar diretamente ligado a tecnologias específicas e fornecer ao usuário a liberdade de escolher as tecnologias que melhor se encaixam as suas necessidades (OPENSTACK, 2015). Abaixo é feita uma análise sobre as principais propriedades do OpenStack.

Live migration: O OpenStack suporta dois tipos de *live migration*: baseada em armazenamento compartilhado e *live migration* por bloco.

Balanceamento de carga: O OpenStack suporta balanceamento de carga em diferentes níveis. A propriedade de *live migration* permitiu que os administradores do sistema distribuam as cargas de trabalho entre os servidores físicos através do ajuste da localização da VM. Além disso, é possível controlar as cargas de trabalho a nível de VM. O OpenStack tem um projeto em desenvolvimento chamado Load Balancing as a Service (LBaaS) que tem objetivo fornecer um serviço de balanceamento de carga para usuário final.

Tolerância a falhas: a tolerância falhas também pode ser feita em diferentes níveis, dependendo da maneira que o IaaS foi configurado. No nível de VM, com objetivo de prever falhas, o usuário pode desenvolver algoritmos de escalonamento (além dos três já existentes). A nível de armazenamento e de banco de dados, a tolerância a falhas é alcançada através da replicação e sincronização garantindo que a falha em um dispositivo não pare todo sistema.

Segurança: OpenStack tem um serviço separado (Keystone) que fornece um gerenciamento central de autenticação para os usuários e o sistema operacional da nuvem.

Compatibilidade: o OpenStack é compatível com o Amazon EC2 e Amazon S3. As aplicações de cliente feitas para a Amazon Web Service podem ser usadas com OpenStack a partir de um esforço mínimo (OPENSTACK, 2015). O OpenStack tem suporte para vários *hypervisors*: Xen, KVM, HyperV, VMWare, etc.

4 CONCLUSÃO

Neste trabalho foi feito um estudo aprofundado sobre a Computação em Nuvem, onde foi estudado suas definições, principais características, os modelos de negócio, a arquitetura, os tipos de Nuvem e os principais desafios da área. Além desse estudo este trabalho também apresentou uma revisão sobre as Plataformas da Nuvem de Código Aberto, onde foi feito um estudo mais aprofundado sobre o software OpenStack. Sobre ele é apresentado seu histórico, sua arquitetura base e suas propriedades.

Como trabalho futuro será feito uma revisão mais aprofundada sobre as outras Plataformas da Nuvem de Código Aberto como CloudStack, OpenNebula, Nimbus e Eucalyptus.

REFERÊNCIAS

A., B.; DINIZ, d. S.; S., I. Open Source Solutions for Building IaaS Clouds. In: SCALABLE COMPUTING: PRACTICE AND EXPERIENCE, 2015. **Anais...** SCPE, 2015. p.187–204. (Volume 16, Number 2).

AGUIAR, M. **Título da Monografia**. 2012. Dissertação de Mestrado — PPGC/UFPEL, Pelotas/RS.

AMAZON. **Amazon EC2**. Acessado em: 4/10/2015, aws.amazon.com/pt/ec2/.

AMRANI, C. E.; FILALI, K. B.; AHMED, K. B.; DIALLO, A. T.; TELOLAHY, S.; EL-GHAZAWI, T. A Comparative Study of Cloud Computing Middleware. In: IEEE/ACM INTERNATIONAL SYMPOSIUM ON CLUSTER, CLOUD AND GRID COMPUTING (CCGRID 2012), 2012., 2012, Washington, DC, USA. **Proceedings...** IEEE Computer Society, 2012. p.690–693. (CCGRID '12).

ARMBRUST, M.; FOX, A.; GRIFFITH, R.; JOSEPH, A. D.; KATZ, R. H.; KONWINSKI, A.; LEE, G.; PATTERSON, D. A.; RABKIN A. STOICA, I.; ZAHARIA, M. Above the Clouds, A Berkeley View of Cloud Computing. In: TECHNICAL REPORT NO. UCB;EECS-2009-28, 2009. **Anais...** UCB, 2009.

BASET, S. A. Open Source Cloud Technologies. In: THIRD ACM SYMPOSIUM ON CLOUD COMPUTING, 2012, New York, NY, USA. **Proceedings...** ACM, 2012. p.28:1–28:2. (SoCC '12).

CHEKURI, C.; KHANNA, S. On Multi-dimensional Packing Problems. In: TENTH ANNUAL ACM-SIAM SYMPOSIUM ON DISCRETE ALGORITHMS, 1999, Philadelphia, PA, USA. **Proceedings...** Society for Industrial and Applied Mathematics, 1999. p.185–194. (SODA '99).

CITRIX. **XenServer**. Acessado em: 4/10/2015, www.citrix.com/products/xenserver/overview.html.

CLARK, C.; FRASER, K.; HAND, S.; HANSEN, J. G.; JUL, E.; LIMPACH, C.; PRATT, I.; WARFIELD, A. Live Migration of Virtual Machines. In: ND CONFERENCE ON SYMPOSIUM ON NETWORKED SYSTEMS DESIGN & IMPLEMENTATION - VOLUME 2, 2., 2005, Berkeley, CA, USA. **Proceedings...** USENIX Association, 2005. p.273–286. (NSDI'05).

CORDEIRO, T.; DAMALIO, D.; PEREIRA, N.; ENDO, P.; PALHARES, A.; GONÇALVES, G.; SADOK, D.; KELNER, J.; MELANDER, B.; SOUZA, V.; MANGS, J.-E. Open Source Cloud Computing Platforms. In: GRID AND COOPERATIVE COMPUTING (GCC), 2010 9TH INTERNATIONAL CONFERENCE ON, 2010. **Anais...** [S.l.: s.n.], 2010. p.366–371.

ENDO, P. T.; GONÇALVES, G. E.; KELNER, J.; SADOK, D. A survey on open-source cloud computing solutions. In: SYMPOSIUM ON COMPUTER NETWORKS AND DISTRIBUTED SYSTEMS, 2010. **Anais...** [S.l.: s.n.], 2010.

GOGRID. **GoGrid - A Datapipe Company**. Acessado em: 4/10/2015, www.datapipe.com/gogrid/.

GOOGLE. **Google App Engine**. Acessado em: 4/10/2015, appengine.google.com/.

HAMDAQA, M.; TAHVILDARI, L. Cloud Computing Uncovered: A Research Landscape. In: HURSON, A.; MEMON, A. (Ed.). . [S.l.]: Elsevier, 2012. p.41 – 85. (Advances in Computers, v.86).

HAMILTON, J. **Cooperative Expendable Micro-Slice Servers (CEMS)**: Low Cost, Low Power Servers for Internet-Scale Services.

HASSAN, Q. F. Demystifying Cloud Computing. **CrossTalk**, [S.l.], p.16–21, 2011.

LASZEWSKI, G. v.; DIAZ, J.; WANG, F.; FOX, G. C. Qualitative comparison of multiple cloud frameworks. In: 2012. **Anais...** [S.l.: s.n.], 2012.

LINUX-KVM. **Kernel Virtual Machine**. Acessado em: 4/10/2015, www.linux-kvm.org/page/Main_Page.

MELL, P. M.; GRANCE, T. **SP 800-145. The NIST Definition of Cloud Computing**. Gaithersburg, MD, United States: [s.n.], 2011.

MICROSOFT. **Microsoft Azure**. Acessado em: 4/10/2015, azure.microsoft.com/.

OPENSTACK. **OpenStack**. Acessado em: 15/10/2015, <http://openstack.org>.

OSMAN, S.; SUBHRAVETI, D.; SU, G.; NIEH, J. The Design and Implementation of Zap: A System for Migrating Computing Environments. **SIGOPS Oper. Syst. Rev.**, New York, NY, USA, v.36, n.SI, p.361–376, Dec. 2002.

RACKSPACE. **Rackspace**. Acessado em: 4/10/2015, www.rackspace.com.

SALESFORCE. **Salesforce**. Acessado em: 4/10/2015, www.salesforce.com.

SAP. **SAP**. Acessado em: 4/10/2015, www.sap.com.

SMARTERTOOLS. **Portal - FlexiScale Technologies Limited**. Acessado em: 4/10/2015, support.flexiscale.com/Main/.

SOFTWARE, O. C. **Chapter 1 – Introduction to OpenStack**. Acessado em: 4/10/2015, <http://docs.openstack.org/training-guides/content/module001-ch001-intro-text.html>.

VAQUERO, L. M.; RODERO-MERINO, L.; CACERES, J.; LINDNER, M. A Break in the Clouds: Towards a Cloud Definition. **SIGCOMM Comput. Commun. Rev.**, New York, NY, USA, v.39, n.1, p.50–55, Dec. 2008.

VMWARE. **VMware**. Acessado em: 4/10/2015, www.vmware.com/products/esxi-and-esx/overview.html.

VORAS, I.; MIHALJEVIC, B.; ORLIC, M.; PLETIKOSA, M.; ZAGAR, M.; PAVIC, T.; ZIMMER, K.; CAVRAK, I.; PAUNOVIC, V.; BOSNIC, I.; TOMIC, S. Evaluating open-source cloud computing solutions. In: MIPRO, 2011 PROCEEDINGS OF THE 34TH INTERNATIONAL CONVENTION, 2011. **Anais...** [S.l.: s.n.], 2011. p.209–214.

ZENG, W.; ZHAO, J.; LIU, M. Several public commercial clouds and open source cloud computing software. In: COMPUTER SCIENCE EDUCATION (ICCSE), 2012 7TH INTERNATIONAL CONFERENCE ON, 2012. **Anais...** [S.l.: s.n.], 2012. p.1130–1133.

ZHANG, Q.; CHENG, L.; BOUTABA, R. Cloud computing: state-of-the-art and research challenges. **Journal of Internet Services and Applications**, [S.l.], v.1, n.1, p.7–18, 2010.