

1 INTRODUÇÃO

Computação em nuvem vem sendo utilizada devido ao seu modelo permitir um melhor aproveitamento dos recursos de hardware e software acarretando em redução de custos, melhor eficiência energética, elasticidade, flexibilidade e disponibilidade de recursos por demanda (ARMBRUST et al., 2010).

Com o aumento da demanda por de serviços prestados pela nuvem os ambientes crescem constantemente para conseguir atender a demanda (SINGH; SOOD; SHARMA, 2011; WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015). O crescimento constante dos ambientes da nuvem acaba aumentando muito a complexidade de sua gerência (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015; HALL, 2012; URGONKAR; SHENOY; ROSCOE, 2002) tornando esta inviável de ser realizada por administradores. Em (KEPHART; CHESS, 2003) é apresentado um modelo de computação autônoma como uma solução para lidar com grandes ambientes computacionais heterogêneos e complexos como ambientes de computação em nuvem.

É necessário realizar o monitoramento constante da nuvem para ter ciência de quanto recurso foi usado por seus usuários e então poder estimar um valor a ser cobrado. Além disso o monitoramento também pode ser usado para viabilizar técnicas de otimização e auxiliar numa melhor gerência destes ambientes. Para viabilizar tais otimizações é importante que o monitoramento forneça dados precisos e em tempo hábil sem afetar o funcionamento da nuvem (VIRATANAPANU et al., 2010)

O monitoramento em tempo real da nuvem é muito desafiador. Primeiro deve-se levar em conta o grande número de serviços e usuários, que torna a nuvem um ambiente com uma gama muito grande de informação para ser processada (SHAO et al., 2010; CLAYMAN; GALIS; MAMATAS, 2010). Além disso tem-se a heterogeneidade dos componentes da nuvem, logo deve-se ter em mente que será necessário implementar meios para monitorar cada um desses componentes, o que é algo muito custoso (SHAO et al., 2010).

Monitorar uma nuvem requer mais cuidado que monitorar outros tipos de software, logo, sobre cargas criadas pelo processo de monitoramento na nuvem é um dos principais obstáculos do monitoramento na nuvem (SHAO et al., 2010; CLAYMAN; GALIS; MAMATAS, 2010). Além disso, após implantado o sistema de monitoramento, qualquer modificação necessária deve ser feita sem a necessidade de desligar ou reiniciar o

ambiente (CLAYMAN; GALIS; MAMATAS, 2010).

Problemas aparentes de um monitoramento mau sucedido são a queda de desempenho da nuvem devido a sobre cargas de processamento com monitoramento. Possíveis tomadas de decisões incorretas da administração da nuvem devido a dados inconsistentes.

Perda de dados causadas pelo fato de ter que desligar ou reinicializar a nuvem para poder implantar modificações feitas na ferramenta de monitoramento.

Atualmente as ferramentas de orquestração de computação em nuvem (levantadas durante o desenvolvimento deste trabalho) em sua maioria não realizam ou realizam mas não disponibilizam os dados resultantes do monitoramento dos recursos utilizados por usuários ou pela nuvem. O que estas ferramentas costumam adotar é o monitoramento momentâneo do ambiente, sem a criação de um histórico de uso de recursos, para indicar aos administradores o estado atual do ambiente.

O que existe atualmente para monitoramento são extensões que são adicionados a essas ferramentas de orquestração, porém estas extensões são de difícil instalação e não são completos pois não permitem o monitoramento de uma série de recursos físicos e algumas costumam apresentar dados inconsistentes com o ambiente, além de ser mais uma camada de software para gerenciar.

O Zenoss é um exemplo dessas extensões que apresentam alguns problemas. No caso do Zenoss, na versão do CloudStack, ele monitora os recursos relativos ao dom0 e não da nuvem como um todo, não dando uma noção do estado real do ambiente.

Ao atuar nessa área, deve-se usar boas técnicas de manipulação de dados devido a grande gama de dados que podem ser monitorados na nuvem. Ser o menos invasivo possível, a nuvem deve manter seu amplo funcionamento como se nada tivesse acontecendo, ou seja, ser invisível para os usuários da nuvem. Além disso, deve-se tratar a precisão das informações capturadas. Quanto mais rápida as informações sobre o estado atual do ambiente forem disponibilizadas, mais eficientemente se conseguirá administrar a nuvem.

2 FUNDAMENTACAO

2.1 COMPUTAÇÃO EM NUVEM

A computação em nuvem é um modelo ubíquo, conveniente e de acesso compartilhado a recursos computacionais como servidores, armazenamento, aplicações e redes. Estes recursos compartilhados podem ser rapidamente provisionados e liberados com um esforço mínimo de gestão ou interação com o provedor de serviços (MELL; GRANCE, 2011).

2.1.1 Características

Acesso aos recursos sob demanda - O usuário pode escolher quais serviços/componentes ele irá usar dentro do que é disponibilizado pela nuvem sem a necessidade de interação com administradores(MELL; GRANCE, 2011).

Disponibilidade - A nuvem deve estar sempre disponível para o usuário pela internet independente da plataforma de acesso (tablet, smartphone, desktop etc).

Compartilhamento de recursos - A nuvem deve ser capaz de disponibilizar os seus recursos para diferentes usuários independente se for o mesmo recursos físico para usuários distintos (arquivos de diferentes usuários no mesmo HD).

Elasticidade - Conforme um usuário vai demandando mais recursos, a nuvem deve alocar mais recursos para este usuário evitando a degradação dos serviços prestados, assim como se um usuário parar de demandar tantos recursos, a nuvem deve desalocar recursos excedentes.

Monitoramento de serviços - A nuvem deve ser capaz de monitorar o uso de recursos por cada serviço, proporcionando transparência aos fornecedores e usuários do serviço.

Modelos de serviço (MELL; GRANCE, 2011):

Software como um serviço (SaaS) : A nuvem fornece uma aplicação final via rede para usuários usarem através de um cliente que pode ou não ser disponibilizado pela nuvem. O usuário não tem controle algum sobre a infra estrutura que hospeda a aplicação ou da plataforma que a aplicação usa.

Plataforma como um serviço (PaaS) : A nuvem disponibiliza diferentes plataformas para o usuário escolher qual mais o agrada e usar

esta plataforma para hospedar suas próprias aplicações. Neste nível o usuário tem controle total das aplicações, e pode configurar por completo a plataforma.

Infra estrutura como um serviço (IaaS) : A nuvem fornece toda a infra estrutura para o usuário, como processador, disco, memória, rede. Neste nível o usuário tem uma máquina virtual pronta para uso, ele define quais plataformas ele irá usar e quais aplicações ele irá rodar em cada plataforma, porém a parte de hardware ele não tem total acesso, endereços físicos de memória e algumas instruções do processador não serão acessíveis ao usuário.

2.1.2 Tipos de nuvens (MELL; GRANCE, 2011)

Nuvem privada : A infra estrutura da nuvem é disponibilizada apenas para uma única organização usar. A nuvem pode ser gerenciada pela própria organização, por alguma outra organização ou por ambos.

Nuvem comunitárias : A infra estrutura da nuvem é disponibilizada apenas para uma comunidade de organizações que tem interesses em comum. A nuvem pode ser gerenciada por alguma(s) dessas organizações(s), terceiros ou por ambos.

Nuvem pública : A nuvem é aberta para todo o público. A é gerenciada por alguma empresa, organização do governo/acadêmica ou alguma combinação destas.

Nuvem híbrida : A infra estrutura da nuvem é uma composição de duas ou mais infra estruturas (privada,comunitária,pública).

2.2 VIRTUALIZAÇÃO

A virtualização consiste em abstrair o hardware do sistema operacional que será hospedado nesse hardware. Esta abstração é chamada de monitor de máquina virtual (VMM) ou hypervisor e, basicamente, oculta os recursos físicos do sistema operacional (OS). Uma vez que os recursos de hardware são diretamente controlados pelo VMM e não pelo sistema operacional (SAHOO; MOHAPATRA; LATH, 2010) . Com esse tipo de abstração em cima do hardware, é possível que os mesmos componentes de hardware suportem diferentes sistemas operacionais como diversas unidades lógicas. Estas unidades lógicas em que o hardware é dividido são nomeadas de Virtual Machine (Máquinas virtuais, VM). Embora essas VMs sejam abstrações do hardware supridas pelo

hypervisor, os sistemas operacionais e aplicações enchem esses ambientes virtuais como reais. Para o usuário final, ter uma máquina virtual ou uma máquina física não irá mudar em nada a sua experiência de uso do computador (SAHOO; MOHAPATRA; LATH, 2010).

2.2.1 Tipos de virtualização

1. Virtualização completa: O VMM atua como o gerente das máquinas virtuais e roda encima de um outro Sistema Operacional que irá fazer a comunicação com o hardware. Toda a comunicação que as máquinas virtuais irão fazer com o hardware será intermediada pelo VMM. Todos os endereços de I/O serão mapeados para endereços lógicos disponibilizados pelo hypervisor (SAHOO; MOHAPATRA; LATH, 2010). Este tipo de virtualização é muito complexa e custosa de se fazer devido as camadas de software extras, porém é a mais versátil.
2. Virtualização da camada do Sistema Operacional: Este tipo de virtualização permite a criação de várias instâncias do mesmo sistema operacional totalmente independentes rodando em paralelo. Quando comparado com outras técnicas de virtualização, esta tende a ser mais eficiente que o anterior e peca apenas na garantia de total isolamento entre VMs (SOLTESZ et al., 2007). Neste tipo de virtualização, os clientes estão presos a criarem VMs exclusivamente do mesmo tipo de sistema operacional do hypervisor.
3. Virtualização da camada de Hardware: Esta abordagem é comumente usado no mercado de servidores devido ao seu elevado isolamento de máquinas virtuais e seu grande desempenho. Neste cenário o hypervisor roda diretamente encima do hardware e controla diretamente o acesso dos sistemas operacionais das VMs aos recursos de hardware (SAHOO; MOHAPATRA; LATH, 2010).

2.2.2 Vantagens e Desvantagens da virtualização

A virtualização permite um melhor aproveitamento de todos os recursos de hardware disponíveis. Permite a alteração de peças de hardware sem a necessidade de parar o funcionamento das VMs por longos períodos, basta migrar a VM para outro dispositivo de hardware. A

criação de máquinas prontas para o uso é rápido como o apertar de um botão. O custo de ter um máquina virtual é muito menor do que o custo para ter uma máquina física com a mesma configuração (SAHOO; MOHAPATRA; LATH, 2010). Como problemas temos uma grande sobrecarga de processamento para gerenciar a inúmera quantidade de máquinas virtuais. O desenvolvimento de virtualizadores não é algo simples e é de inúmera importância ter redundância de hardwares pois se um componente de hardware der defeito, todas as VMs vinculadas a tal componente serão penalizadas (POPEK; GOLDBERG, 1974) .

2.2.3 Virtualizadores

Xen : É um software de virtualização livre para arquiteturas x86, x86-64, IA-32, IA-64 e PowerPC. O Xen suporta o tipo de virtualização 3. O código do Xen foi comprado pela empresa Citrix que abriu o código e disponibilizou para comunidade além de uma boa documentação.

KVM : É um virtualizador do tipo 1 que roda numa plataforma linux. O KVM suporta as mesmas arquiteturas do Xen, porém deixa a desejar em desempenho e confiabilidade, além da documentação não ser muito bem formalizada e a comunidade não chega ser tão ativa.

VMWare (Free): Este hypervisor suporta apenas arquiteturas x86. Assim como o KVM, o VMware é um virtualizador do tipo 1. Um problema do VMware é a falta de suporte para placas com barramento PCI, a usabilidade parece ser difícil se comparada com os outros hypervisors.

2.3 MONITORAMENTO NA NUVEM

Monitoramento na nuvem: Consiste em obter métricas relativas a um ambiente específico num intervalo de tempo específico. As métricas podem ou não serem filtradas em uma busca (CLAYMAN; GALIS; MAMATAS, 2010).

As métricas monitoradas devem ser específicas para cada ambiente, não se deve monitorar métricas desnecessárias (que não serão aproveitadas) pois serão apenas ruídos (CLAYMAN; GALIS; MAMATAS, 2010). O monitoramento de qualquer ambiente da nuvem deve ser realizado usando ferramentas que possam lidar com um grande número de parâmetros e objetos a serem monitorados.

Poder monitorar qualquer objeto adicionado recentemente e dei-

xar de monitorar qualquer objeto removido automaticamente (CLAYMAN; GALIS; MAMATAS, 2010). Estar disponível quando solicitada. Suportar diferentes tamanhos de cargas. Ser eficiente quanto a disponibilidade das métricas monitoradas, sempre disponibilizar a métrica assim que obtida (MIAN; MARTIN; VAZQUEZ-POLETTI, 2013). Ser precisa, fornecer dados condizentes com a situação em que o ambiente se encontra.

A ferramenta que for utilizada para o monitoramento também deverá ser capaz de se gerenciar sem a necessidade de um ser humano para sua manutenção (MIAN; MARTIN; VAZQUEZ-POLETTI, 2013). Suportar estensões de funcionalidade sem precisar de modificações bruscas (MIAN; MARTIN; VAZQUEZ-POLETTI, 2013).

2.4 ORQUESTRAÇÃO DA NUVEM

Computação em nuvem é algo complexo devido aos seus inúmeros componentes heterogêneos. Estes componentes devem ser gerenciados de forma a conseguir trabalhar em conjunto mesmo com as suas diferentes arquiteturas. Realizar isso num ambiente vasto como a nuvem com uma grande quantidade destes componentes é algo impraticável para um administrador humano (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015).

As ferramentas de orquestração na nuvem servem para abstrair do administrador toda a parte de interação com os hypervisors e serviços de armazenamento. Possibilitando que o administrador se preocupe apenas com o ambiente em seu estado superficial, sem ter que saber o que se passa em níveis mais baixos.

Essas abstrações proporcionadas pelas ferramentas de orquestração tornam possível que o administrador da nuvem possa delimitar recursos físicos para cada usuário sem a necessidade de um conhecimento específico do hypervisor que irá realizar tal função. Assim como também possibilita que usuários sem conhecimento nenhum de computação possam criar suas próprias máquinas virtuais sem nenhuma dificuldade.

2.5 FERRAMENTAS DE ORQUESTRAÇÃO

Dentre as ferramentas pesquisadas (CloudStack, OpenStack, Eucalyptus), o CloudStack foi a escolhida. O CloudStack possui su-

porte para os hypervisors vSphere (VMWare), KVM, Xen e Hyper-V, que são suportados também pelo OpenStack e, com execução do Hyper-V, pelo Eucalyptus também.

Então por que foi escolhido o CloudStack? Devido a usar sistemas de arquivos conhecidos, como iSCSI e NFS. Isso economizaria tempo de pesquisa, pois as outras ferramentas, implementam sistemas de arquivos próprios, o que requer tempo para uma total compreensão do funcionamento dos mesmos. Além disso o CloudStack é gerido pela fundação Apache, que zela pela boa documentação de suas ferramentas e possui uma comunidade em ampla atividade.

REFERÊNCIAS

- ARMBRUST, M. et al. A view of cloud computing. **Communications of the ACM**, ACM, v. 53, n. 4, p. 50–58, 2010.
- CLAYMAN, S.; GALIS, A.; MAMATAS, L. Monitoring virtual networks with lattice. In: IEEE. **Network Operations and Management Symposium Workshops (NOMS Wksp)**, 2010 **IEEE/IFIP**. [S.l.], 2010. p. 239–246.
- HALL, P. Opportunities for csps in enterprise-grade public cloud computing. **OVUM**, May, 2012.
- KEPHART, J. O.; CHESS, D. M. The vision of autonomic computing. **Computer**, IEEE, v. 36, n. 1, p. 41–50, 2003.
- MELL, P.; GRANCE, T. The nist definition of cloud computing. Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology Gaithersburg, 2011.
- MIAN, R.; MARTIN, P.; VAZQUEZ-POLETTI, J. L. Provisioning data analytic workloads in a cloud. **Future Generation Computer Systems**, Elsevier, v. 29, n. 6, p. 1452–1458, 2013.
- POPEK, G. J.; GOLDBERG, R. P. Formal requirements for virtualizable third generation architectures. **Communications of the ACM**, ACM, v. 17, n. 7, p. 412–421, 1974.
- SAHOO, J.; MOHAPATRA, S.; LATH, R. Virtualization: A survey on concepts, taxonomy and associated security issues. In: IEEE. **Computer and Network Technology (ICCNT)**, 2010 **Second International Conference on**. [S.l.], 2010. p. 222–226.
- SHAO, J. et al. A runtime model based monitoring approach for cloud. In: IEEE. **Cloud Computing (CLOUD)**, 2010 **IEEE 3rd International Conference on**. [S.l.], 2010. p. 313–320.
- SINGH, G.; SOOD, S.; SHARMA, A. Cm-measurement facets for cloud performance. **International Journal of Computer Applications**, International Journal of Computer Applications, 244 5th Avenue, # 1526, New York, NY 10001, USA India, v. 23, n. 3, p. 37–42, 2011.

SOLTESZ, S. et al. Container-based operating system virtualization: a scalable, high-performance alternative to hypervisors. In: ACM. **ACM SIGOPS Operating Systems Review**. [S.l.], 2007. v. 41, n. 3, p. 275–287.

URGAONKAR, B.; SHENOY, P.; ROSCOE, T. Resource overbooking and application profiling in shared hosting platforms. **ACM SIGOPS Operating Systems Review**, ACM, v. 36, n. SI, p. 239–254, 2002.

VIRATANAPANU, A. et al. On demand fine grain resource monitoring system for server consolidation. In: IEEE. **Kaleidoscope: Beyond the Internet?-Innovations for Future Networks and Services, 2010 ITU-T**. [S.l.], 2010. p. 1–8.

WEINGÄRTNER, R.; BRÄSCHER, G. B.; WESTPHALL, C. B. Cloud resource management: A survey on forecasting and profiling models. **Journal of Network and Computer Applications**, Elsevier, v. 47, p. 99–106, 2015.