

Planejamento de capacidade para sistemas Linux, usando VPS para o fornecimento de SaaS.

Saulo Gomes

Curso de Bacharelado em Ciência da Computação – Universidade Federal Rural de Pernambuco (UFRPE) – Campus de Recife
Rua Dom Manoel de Medeiros, s/n, Dois Irmãos - CEP: 52171-900 – Recife/PE - Brasil
saulobr88@gmail.com

Resumo. *Com a crescente dependência tecnológica que a sociedade atualmente possui, planejamento e gerência de capacidade são assuntos de considerável importância para a continuidade do bom fornecimento de serviços na internet. Utilizando conceitos de computação em nuvem e análise de desempenho de sistemas o seguinte trabalho tem como objetivo apresentar um conjunto de softwares e técnicas para medição para ambientes computacionais que utilizam serviços sobre o protocolo de rede HTTP utilizando Web Service. Como resultado é apresentado a relação entre aplicação e desempenho de infraestrutura, informação importante para a tomada de decisão referente a investimento em mais recursos computacionais.*

1. Introdução e Motivação

Com o aumento dos dispositivos que se conectam a internet e o grande volume de dados gerados pelos usuários de serviços online [EFE. 2015], se faz necessário mensurar quais são os requisitos para fornecer software como serviço [Melo et al. 2006]. Poder avaliar o desempenho de um sistema que está alocado em um provedor de infraestrutura com a finalidade de saber quando é a hora mais adequada para um upgrade ou para distribuição de componentes (Banco de Dados, APIs, Repositórios etc) é um recurso importante nos dias atuais. Assim, conhecer os limites dos recursos atuais e planejar a evolução dos componentes é de grande importância para a sustentabilidade de negócios, garantindo a elasticidade da Nuvem e qualidade para os usuários.

Ambientes virtualizados possuem consideráveis vantagens, como por exemplo a possibilidade de aumento da capacidade computacional de forma rápida com impacto reduzido no fornecimento de algum serviço. Fornecedores de infraestrutura como a **Amazon** e **Digital Ocean** fornecem planos baseados em tempo de uso, dessa forma saber o quanto é necessário dos recursos computacionais é um requisito para a aquisição de ambientes na nuvem.

O trabalho a seguir apresenta um método para simular e medir o quanto uma aplicação web, usando o protocolo de rede HTTP, precisará para fornecer uma quantidade específica de usuários com requisições concorrentes, levando em consideração que não deve haver atrasos no tempo de serviço, já que isso implica queda na qualidade de serviço. Com um conjunto de softwares para monitoramento e fornecimento de serviços em sistemas que usam o Kernel Linux, sistema adotado em

grande escala para servir páginas na internet e outros recursos acessados via rede de computadores.

Este trabalho está organizado como segue: A seção 2 apresenta a fundamentação teórica com definições que envolvem nuvem computacional, serviços na internet e planejamento de capacidade. A seção 3 apresenta trabalhos relacionados. A seção 4 apresenta a metodologia utilizada para a medição. A seção 5 apresenta o ambiente juntamente com os softwares envolvidos, por fim na Seção 6 são mostradas as considerações finais e indicações de trabalhos futuros.

2. Fundamentação Teórica

Para uma melhor compreensão do trabalho de medição realizado se faz necessário alguns conhecimentos, tais como:

VPS é a sigla para *Virtual Private Server* (Servidor Virtual Privado), um termo utilizado para informar que o computador, máquina, é de uso exclusivo de quem contratou, uso privado, e que é uma máquina virtual. Segundo [Araujo & Cavalcante, 2012], a virtualização de servidores se refere à captação de recursos computacionais, tais como o processo de funcionamento de dois ou mais computadores em um conjunto de sistemas de hardware. A virtualização de servidores é uma das bases que sustentam o modelo da computação em nuvens que proporciona a economia de recursos e práticas de TI verde.

Elasticidade e escalonamento na nuvem, Segundo [Buyya et al, 2011] a computação na nuvem propicia a ilusão de recursos computacionais infinitos disponíveis para o uso. Portanto, os usuários têm a expectativa de que a nuvem seja capaz de fornecer rapidamente recursos em qualquer quantidade e a qualquer momento. É esperado que os recursos adicionais possam ser providos, possivelmente de forma automática, quando ocorre o aumento da demanda e retidos, no caso da diminuição desta demanda.

Linux, é um termo utilizado para se referir a sistemas operacionais que utilizem o núcleo Linux, **Kernel Linux**. O Linux foi inicialmente desenvolvido pelo cientista da computação finlandês Linus Torvalds, inspirado no sistema Minix. É um sistema operacional muito utilizado para computadores que fornecem recursos pela rede de computadores.

SaaS é a sigla para *Software as a Service* (Software como serviço). Segundo [Souza et al. 2009] o modelo de SaaS proporciona softwares com propósitos específicos que são disponíveis para os usuários através da Internet. Os softwares são acessíveis a partir de vários dispositivos do usuário por meio de uma interface *thin client* como um navegador Web.

O SaaS foi desenvolvido para o uso de ferramentas da web. No SaaS, o usuário não administra ou controla a infraestrutura subjacente, incluindo rede, servidores, sistemas operacionais, armazenamento, ou mesmo as características individuais da aplicação, exceto configurações específicas [Velte, 2011].

Planejamento de Capacidade é um termo pertencente a avaliação de desempenho, segundo [Jain, 1991] planejamento de capacidade é assegurar que os recursos computacionais adequados estarão disponíveis para futuras cargas de trabalho de forma eficaz em termos de custos, enquanto sejam cumpridos os objetivos de desempenho. De forma geral envolve prever o quanto será necessário para fornecer

algum serviço ou recursos para uso futuro, envolvendo o crescimento de uso usando métodos de monitoramento dos recursos computacionais e padrão de uso.

3. Trabalhos Relacionados

Em [Mohamed et Al, 2014] é apresentado uma abordagem que usa linguagem de modelagem para domínio específico, onde é aplicado em um exemplo de servidor HTTP virtualizado. Esse trabalho possui técnicas que apresentam dados simples de interpretação e por meio de regressão linear é apresentado o comportamento para previsão de uso futuro baseado na quantidade de usuários concorrentes.

Em [Menescé et al, 2004] são apresentados vários conceitos sobre engenharia de planejamento de capacidade, além de modelos matemáticos para o planejamento.

4. Metodologia Proposta

Utilizando o software para coleta de dados Collectl, presente nos repositórios de distribuições Linux, serão coletados dados referentes ao uso do Processador, memória, entrada e saída de disco e rede. Será utilizado um Bash Script que será executado 5 segundos antes do gerador de carga iniciar as requisições.

A aplicação fornecida pelo servidor web será uma API simples, para simular um ambiente de informações distribuídas, ambiente comumente utilizado por aplicações de dispositivos móveis e sistemas distribuídos que usam o protocolo HTTP para troca de informações. A base de dados utilizada é a **Sakila Database**, base fornecida pela equipe do MySQL para utilização em livros, tutoriais, artigos e experimentos de forma geral. A consulta utilizada nos testes retorna as 200 linhas da tabela 'actor' da base de dados.

Os softwares Jmeter e Apache Bench, fornecidos pela Fundação Apache, serão utilizados para a simulação de carga no ambiente.

5. Medição do ambiente virtual

A máquina virtual será executada pelo programa VirtualBox, a máquina terá apenas um núcleo de processamento e 1GB de memória principal, com essa configuração estará próxima da configuração fornecida pela Amazon para o serviço gratuito de instância EC2, que é uma máquina virtual disponível pelo prazo de um ano.

O perfil dos softwares do servidor são: Distribuição Linux Ubuntu Server, Servidor Web Apache, Interpretador PHP e Servidor MySQL. Foram utilizadas as versões mais recentes disponíveis nos repositórios da distribuição.

Por padrão o Servidor Web Apache possui uma configuração para responder até 150 requisições concorrentes.

Para a geração da carga de trabalho será utilizado o Apache Jmeter, versão 2.13 r1665067. A carga gerada terá um comportamento crescente de requisições concorrentes, de acordo com a seguinte progressão: 1, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150. Com um tempo de inicialização de 5 segundos para cada grupo de usuários do Jmeter. O Apache Bench usará a mesma progressão.

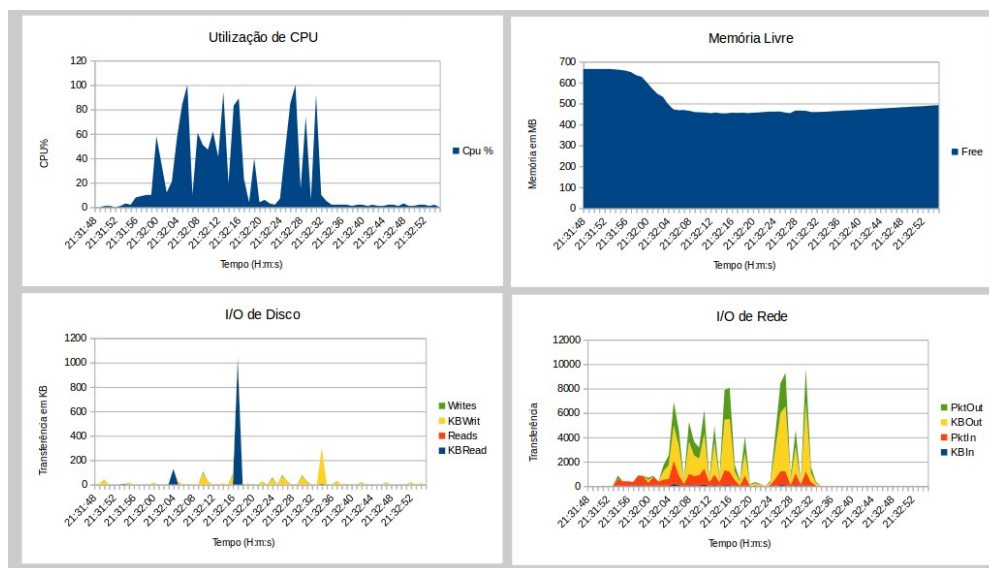


Figura 1. Carga de trabalho com o Jmeter.

6. Considerações Finais

Foram realizados 3 testes iguais utilizando o Jmeter, os dados coletados são apresentados na **Figura 1** na forma de gráfico. Nesses testes a máquina virtual foi submetida a carga poucos segundos após ser iniciada, garantindo assim que não haviam outras tarefas ocupando recursos da mesma, apenas os processos necessários para o bom funcionamento do sistema como um todo. A **Figura 2** apresenta o comportamento dos recursos submetidos ao teste do Apache Bench, onde não havia tempo de pausa entre a progressão das requisições. É possível observar que há uma relação entre os I/O da rede e a Utilização de CPU.

Em momentos que a utilização de CPU chega a 100% há um aumento no tempo de resposta. Nos momentos em que o uso da CPU está acima de 90%, esse tempo já se apresenta mais longo, normalmente o tempo é entre 1,2 a 0,1 segundos, nos momentos de pico de utilização o tempo chega à 4 segundos, essa informação é apresentada como saída do comando Apache Bench, um valor não aceitável em alguns tipos de serviços online.

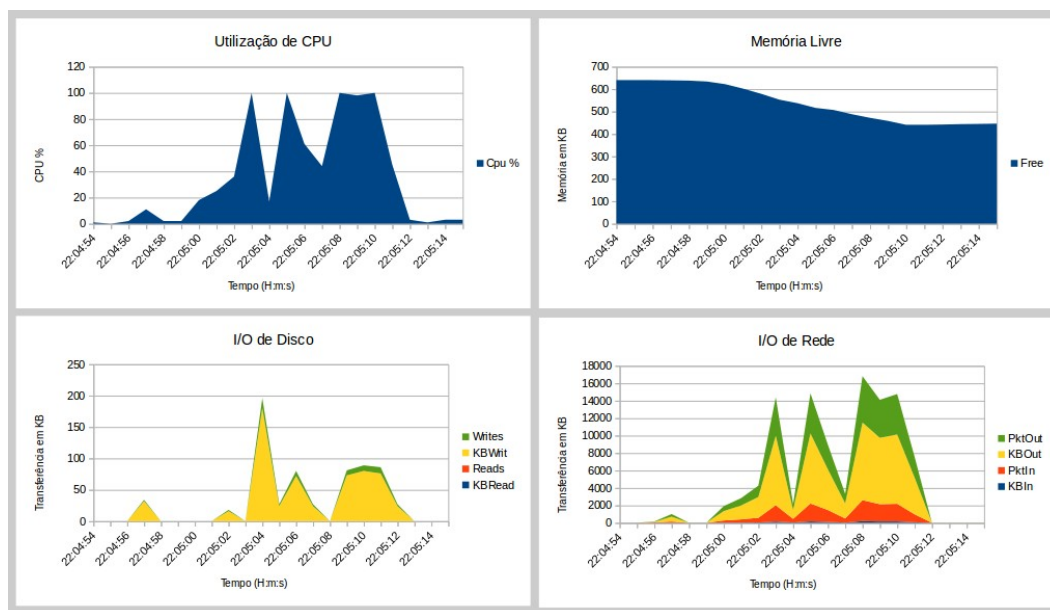


Figura 2. Carga de Trabalho com o Apache Bench

O uso de memória não fica comprometido com esse tipo de aplicação, a entrada e saída de dados no disco é mais utilizada em altos valores de requisições, acima de 90 requisições, devido ao fato do Servidor Web escrever mais informações nos arquivos de *logs*.

Os scripts, logs e planilha com os dados analisados estão disponíveis em [Gomes, 2015], para futura reprodução de experimento. Existem alterações que podem ser realizadas para melhorar o desempenho de alguns recursos, tais como alteração nas configurações do SGDB, do Servidor Web e de opções do Kernel Linux.

Entre os trabalhos futuros destacam-se:

1. Uso mais amplo da API (*Web Service*), simulando mais de uma requisição para cada usuário único;
2. Utilizar um processador Multicore, alterando o módulo **mpm_prefork** pelo **mpm_work** do servidor Web Apache;
3. Realizar *tuning* de Kernel, SGDB e Servidor Web. Um *trade-off* entre o uso de memória e carga de CPU.

Com a crescente dependência tecnológica que a sociedade atualmente possui, planejamento e gerência de capacidade são assuntos que fazem parte do cotidiano de profissionais que trabalham com infraestrutura computacional, para garantir a qualidade de serviço e competitividade entre fornecedores de soluções na nuvem se faz necessário a difusão de técnicas e estudos sobre ambientes dessa natureza. A realização desse tipo de estudo comprova a necessidade de investimento em infraestrutura para o bom fornecimento de serviço.

Referências

- [Araujo & Cavalcante. 2012] Araujo, B. D. L; Cavalcante, V. M. R. M. (2012) **Computação em Nuvens: Contribuição para a Sustentabilidade Econômica e Ecológica**. I ERIPE (Revista da Escola Regional de Informática de PE).
- [Buyya et al, 2011] Buyya, Rajkumar; Broberg, James; Goscinski, Andrzej. **Cloud Computing – Principles and Paradigms**; 1. ed New Jersey, U.S. : John Wiley & Sons, Inc. 2011. 664p. ISBN: 978-0-470-88799-8
- [EFE. 2015] Agencia EFE. **Internet terá 100 bilhões de dispositivos conectados em 2020, projeta consultoria**. Disponível em: < <http://exame.abril.com.br/tecnologia/noticias/internet-lidara-com-100-bilhoes-de-dispositivos-em-2020> >. Acessado em: 24 out. 2015.
- [Gomes, 2015] Gomes, S. **Artigo_Capacity_Planning_2015**. Disponível em: < https://github.com/saulobr88/Artigo_CapacityPlanning_2015 >. Acessado em 22 nov 2015.
- [Jain. 1991] Jain, R. **The Art of Computer Systems Performance Analysis: techniques for experimental design, measurement, simulation and modeling**, John Wiley, 1991, ISBN: 0-471-50336-3
- [Melo et al. 2006] Melo, C. A., Arcoverde, D. F., Moraes, E. R. A., Pimentel, J. H. C., Freitas, R. Q. (2006) **Software como Serviço: Um Modelo de Negócio Emergente**. UFPE.
- [Menescé et al, 2004] Menescé, D. A. Almeida, V. A. F. Dowdy L. W. **Performance by Design - Capacity Planning by Example**. Prentice Hall PTR, 2004. ISBN: 0-13-090673-5. 552p.
- [Mohamed et Al, 2014] Mohamed, R. P. Kolovos, D. S. Paige, R. F. **Resource Requirement Analysis for Web Applications Running in a Virtualised Environment**, 2014 IEEE 6th International Conference on Cloud Computing Technology and Science.
- [Souza et al. 2009] Souza, F. R. C., Moreira, L. O., Machado, J. C. (2009). **Computação em Nuvem - Conceitos, Tecnologias, Aplicações e Desafios**. III Escola Regional de Computação Ceará – Maranhão – Piauí (ERCEMAPI).
- [Velte, 2011] Velte, A; Velte, T. J.; Elsenpeter, R. **Computação em Nuvem: Uma abordagem prática**. 1 ed. Rio de Janeiro: Alta Books, 2011.