

ANÁLISE DOS MECANISMOS DE MONITORAMENTO PARA AVALIAÇÃO DE DESEMPENHO EM UM CLUSTER VIRTUAL DE BAIXO CUSTO

Lucas G. O. Silva¹, Débora C. M. Queiroz¹, Aldo C. L. Figueiredo¹, Ananias P. Neto²

1. Estudante de Telecomunicações do Instituto Federal de Educação Ciência e Tecnologia do Pará (IFPA)
2. Professor do IFPA - Orientador

Resumo

Este artigo apresenta um estudo comparativo dos mecanismos de monitoramento, tais como, GET HTTP, PING e TCP aplicados em um *cluster* virtual de baixo custo, para garantir alta disponibilidade do serviço HTTP, utilizados em Servidores Web. A proposta principal deste estudo é investigar a viabilidade dos mecanismos de monitoramento de um *cluster* virtual de baixo custo para alta disponibilidade de dados, aliada a avaliação de desempenho da plataforma proposta, como forma de mensurar quantitativamente e qualitativamente a infraestrutura desenvolvida. O estudo é baseado nas métricas de vazão, memória e disponibilidade do serviço HTTP. Resultados obtidos mostram que a técnica com uso do PING é uma boa alternativa para o monitoramento do *cluster* virtual de alta disponibilidade, comparadas às demais técnicas.

Palavras-chave: Sistemas Distribuídos; Alta Disponibilidade; Armazenamento de Dados.

Introdução

A popularização de diversos aplicativos móveis usados em áreas como educação, saúde, economia, entretenimento, entre outros, têm obrigado muitos profissionais na área da tecnologia da informação a utilização de sistemas computacionais com maior capacidade de armazenamento e processamento, para poder processar e disponibilizar grandes quantidades de dados [1]. No entanto, o custo dessa infraestrutura não é viável por questão econômica e, desde modo, em muitos casos os serviços são providos pelas empresas de grande porte. Atualmente, existem inúmeros serviços de comunicações e armazenamento que transmitem voz, vídeo e dados através de complexas redes de transmissão e armazenamento computacional, impactando na qualidade do serviço e na alta disponibilidade dos recursos. O uso e disseminação de conteúdo pela rede mundial de computadores é uma realidade que vem crescendo no cotidiano tecnológico e, com essa expansão, a tendência é o desenvolvimento de novas plataformas e/ou infraestruturas que atendam a necessidade do usuário final [2].

Neste contexto, para garantir plataformas com maior capacidade de armazenamento, garantido alta disponibilidade dos serviços e recursos, se faz necessário o uso de soluções de sistemas distribuídos, tais como, os *clusters* de alta disponibilidade necessários para manter os serviços sempre ativos e tolerantes a falhas. Segundo [3], os sistemas de computação de alto desempenho estão prontos para garantir recursos computacionais superiores aos recursos normalmente disponíveis como, por exemplo, servidores de médio e pequeno porte. Um dos desafios na concepção dos sistemas distribuídos consiste no desenvolvimento de plataformas que viabilizem tanto, a alta disponibilidade de dados, quanto a visão centrada no usuário na utilização dos recursos e serviços disponíveis [4].

Este trabalho propõe desenvolver um *cluster* de baixo custo utilizando conceitos e técnicas de virtualização de servidores para armazenamento de dados, além de proporcionar alta disponibilidade de serviços e recursos em *cluster* de servidores web, baseado na composição de *softwares* livres. O objetivo é investigar a viabilidade dos mecanismos de monitoramento, GET HTTP, PING e TCP do *cluster* virtual de baixo custo para alta disponibilidade de dados.

Metodologia

A metodologia inicial deste trabalho consiste no desenvolvimento do *cluster* de baixo custo escolhida, por meio dos recursos como, Sistema Operacional Linux, distribuição *Ubuntu Server*, plataformas de virtualização, *Oracle VM hypervisorVirtual Box* [5] e *software* de monitoramento *Zabbix* [6], todos *softwares* livres. Um *cluster* é definido como um sistema distribuído de computadores independentes e interligados, cujo objetivo é suprir a necessidade de um grande poder computacional com um conjunto de computadores de forma transparente ao usuário e principalmente com baixo custo [7]. A plataforma do *cluster web* foi desenvolvida utilizando a proposta de um nó do *cluster* configurado como máquina mestre, também denominado de *Front-End*, conforme ilustra a Figura 1, sendo que o *Front-End* distribui as requisições aos demais servidores do *cluster*, também denominados de nós escravos. A máquina mestre, estabelece uma conexão TCP com o servidor (camada de transporte), sempre que um cliente fizer uma requisição HTTP e oferece distribuição de requisição em função de conteúdo, em outras palavras, ela decide qual servidor deve repassar a requisição. Isso garante o balanceamento de carga. Na aplicação foi desenvolvido também um *Web Site*, com a finalidade de disponibilizar os recursos armazenados no *cluster web*, garantido alta disponibilidade dos recursos armazenado.

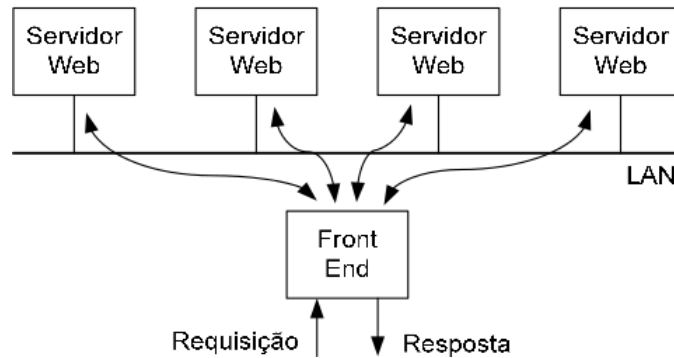


Figura 1. Cenário da plataforma de baixo custo, *cluster web* proposto.

O cenário da Figura 1 foi desenvolvido na infraestrutura dos laboratórios de informática do IFPA, *Campus* Belém, compostos de infraestrutura de redes de computadores e Internet apropriados a proposta da pesquisa. A infraestrutura pertence ao Curso Superior de Tecnologia em Sistemas de Telecomunicações, envolvendo alunos do curso técnico e superior. Após o desenvolvimento da plataforma realizou-se a avaliação do desempenho da plataforma proposta, através de experimentos, utilizando métricas de avaliação de desempenho, tais como vazão, memória e disponibilidade do serviço HTTP.

O cenário final do *cluster* virtual utilizado neste estudo está ilustrado na Figura 2, no qual foi implementado na sua plataforma um *cluster*, composto de 9 (nove) computadores, sendo 1 (um) *Servidor Front-End* (servidor *master*) e 8 (oito) *Servidores Web* Virtuais, com sistema operacional *Ubuntu Server*, 1 (um) *Servidor de Monitoramento*, com *software Zabbix* e 1 (um) *switch*. Os *hardwares* utilizados para a realização dos experimentos possuem as especificações descritas a seguir: Computadores HP Compaq 8200 Elite Small, com processador Intel Core I5, memória de 4 GB, DDR 3, Sistema Operacional Windows 7 Professional.

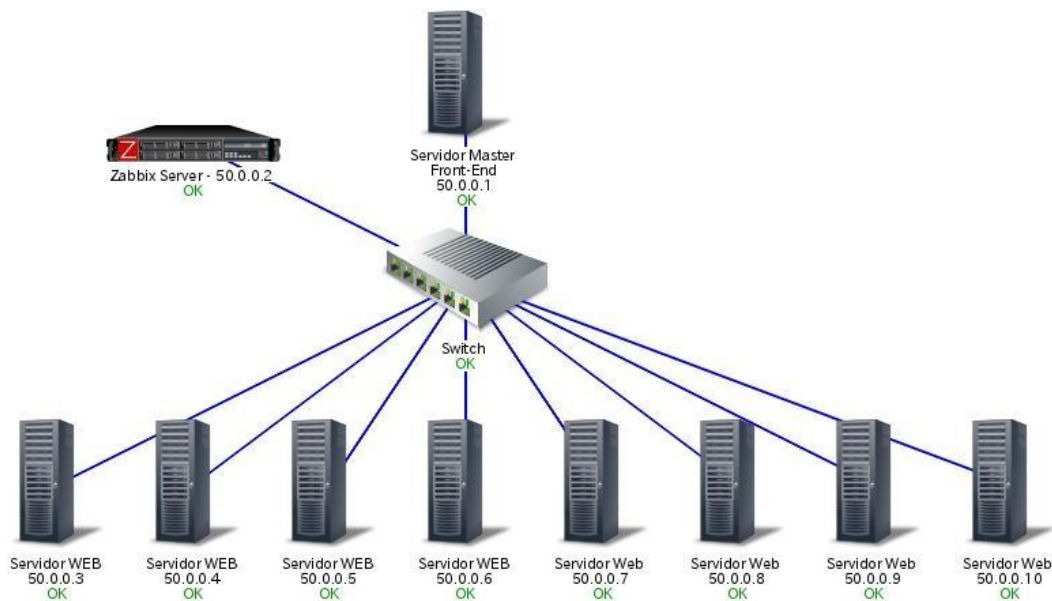


Figura 2. Mapa lógico da plataforma do *cluster* virtual.

Resultados e Discussão

Neste estudo foram avaliados 3 (três) técnicas de requisições e monitoramento dos servidores virtuais via *Front-End*, tais técnicas são: PING, GET HTTP e TCP. A finalidade é verificar qual delas apresenta melhor desempenho para a plataforma virtual desenvolvida, no que se refere ao atendimento das requisições HTTP que acessam os servidores. Para a análise e coleta dos dados foi utilizado, neste experimento, o *software Zabbix*. O servidor *Zabbix* é um *software open source* que monitora diversos parâmetros de uma rede de computadores, utilizando inúmeras métricas, para avaliar o desempenho.

Na avaliação da plataforma desenvolvida, especificamente no aglomerado computacional, utilizou as seguintes métricas: vazão da rede, disponibilidade do serviço HTTP e memória do *cluster* virtual. Desta forma, a primeira análise realizada foi quanto ao uso de memória do *cluster* virtual relacionado ao atendimento das requisições HTTP, considerando o atendimento utilizando o GET HTTP, Figura 3 (a), utilizando o PING, Figura 3 (b) e utilizando o TCP, Figura 3 (c), monitorados durante 1h.

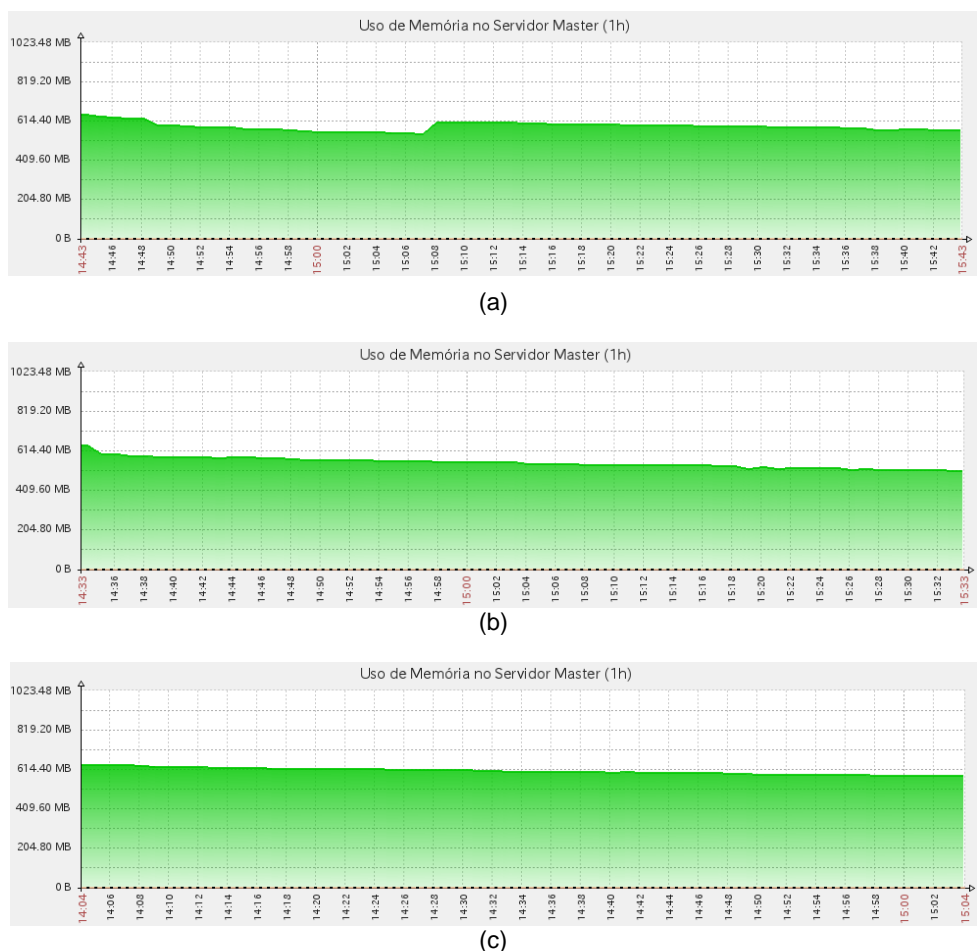


Figura 3. Uso da memória do *cluster* virtual: (a) GET HTTP, (b) PING e (c) TCP.

Os resultados para o uso da memória do *cluster* demonstram que o atendimento das requisições HTTP mais efetivo, para processamento das atividades do *cluster* virtual, foi do PING. Os resultados apresentam média de uso de memória para o atendimento com o PING de 546.51 MB, para o atendimento com o GET HTTP de 580.06 MB e para o atendimento com o TCP de 598.66 MB. O fato do TCP apresentar pior resultado esta relacionado ao grande controle de fluxo do protocolo, proporcionando um maior consumo de memória.

Na garantia de alta disponibilidade do serviço, do *cluster* virtual, os resultados individuais de cada atendimento de requisições HTTP apresentaram resultados satisfatórios para ambos os métodos utilizados, garantido que o serviço WEB esteve ativo durante todo período da avaliação, pelo período de 1h, garantindo alta disponibilidade do serviço, conforme ilustrado Figura 4.

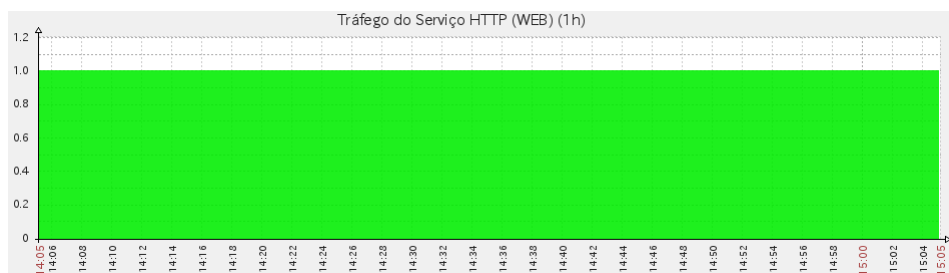


Figura 4. Disponibilidade do Serviço HTTP (Web) para GET HTTP, PING e TCP.

Para a vazão na rede, a Figura 5 (a) apresenta o desempenho para o atendimento com o GET HTTP, a Figura 5 (b) mostra o atendimento com o PING e a Figura 5 (c) ilustra o atendimento com o TCP. Nota-se que a vazão na rede, referente ao tráfego das requisições de entrada no *cluster* é maior para o atendimento de requisições HTTP com o uso do PING, apresentando uma vazão média de 194.03 Kbps. Esse crescimento da vazão média do atendimento com o PING em comparação ao atendimento com o GET HTTP, vazão média de 179.01 Kbps e com o TCP, vazão média de 177.04 Kbps, confirma o melhor perfil de desempenho, devido a abordagem de requisições reduzidas utilizadas pelo PING, no qual o seu controle de entrada é otimizado pelas mensagens curtas aumentando portanto a vazão da rede.

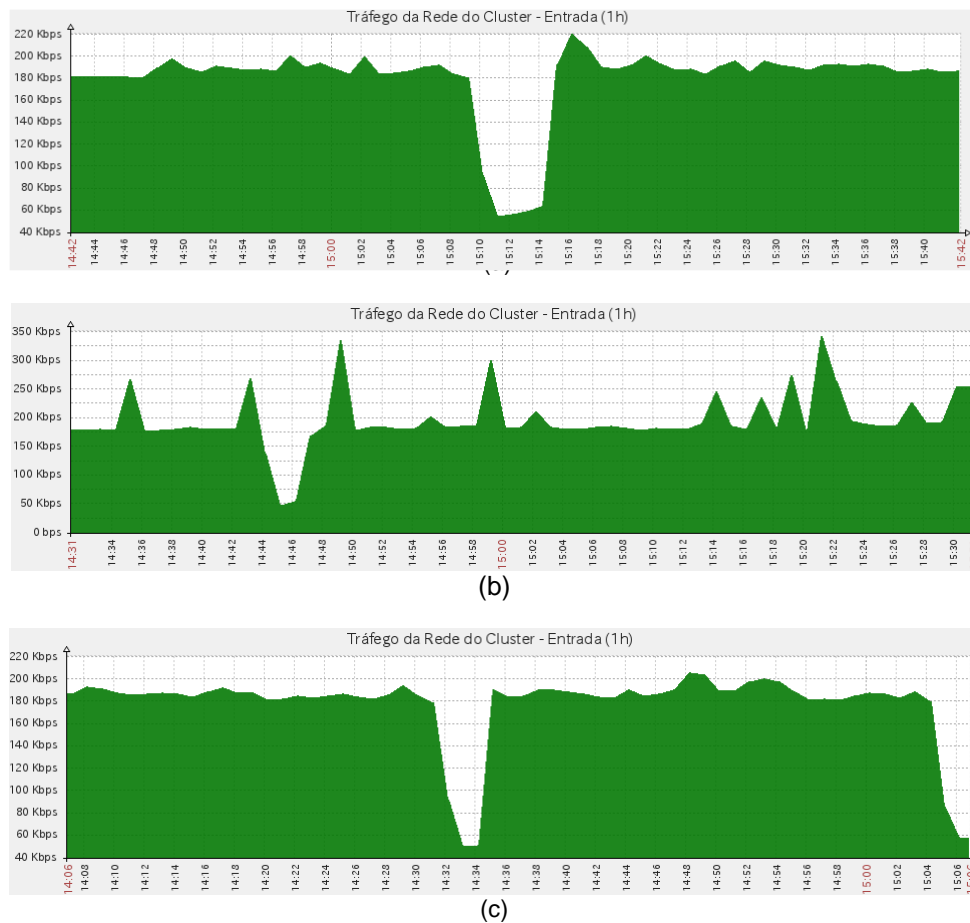


Figura 5. Vazão da Rede do Cluster Virtual: (a) GET HTTP, (b) PING e (c) TCP.

Conclusões

Conclui-se que para os resultados do uso de memória, disponibilidade do serviço HTTP e vazão na rede todas as formas de atendimento tiveram resultados próximos, porém para o atendimento com o PING obteve-se o melhor desempenho comparado com o GET HTTP e TCP que são mais onerosos. Para os resultados da vazão, o atendimento das requisições com o TCP obteve o pior desempenho, visto que o monitoramento do tráfego de pacotes na rede é mais rigoroso para o protocolo TCP. O melhor caso obtido foi utilizando o atendimento das requisições com o uso do PING, devido a utilização de mensagens curtas para as requisições e monitoramento do recursos no *cluster* virtual proposto.

Referências bibliográficas

- [1] E. Britannica, "Cloud Computing", 2020. Acesso em 13 fevereiro de 2020. Disponível em: <<https://www.britannica.com/technology/cloud-computing>>.
- [2] J. R. Luz, M. Martini e E. R. Kaufmann, "Cluster de Alta Disponibilidade da Informação com Software Livre", *Unoesc & Ciência*, vol. 7, n. 1, p. 23-34, Janeiro, 2016.
- [3] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the Suitability of Fog Computing in the Context of Internet of Things", *IEEE Transactions on Cloud Computing*, pp. 99:1–1, 2015.
- [4] F. Zafar, A. Khan, S. Malik, M. Ahmed, A. Anjum, M. Khan, N.Javed, M.Alam and F. Jamil "A Survey of Cloud Computing Data Integrity Schemes: Design Challenges, Taxonomy and Future Trends", *Elsevier Computer & Security*, vol. 67, p. 29-49, March, 2017.
- [5] VirtualBox. Disponível em: <https://www.virtualbox.org>. Acessado em 20/01/2020.
- [6] Zabbix. Disponível em: <http://www.zabbix.org>. Acessado em 22/01/2020.
- [7] D. Beserra, A. Borba, S. Solto, M. Andrade, A. Araújo, "Desempenho de Ferramentas de Virtualização na Implementação de Clusters Beowulf Virtualizados em Hospedeiros Windows", *X Workshop em Clouds e Aplicações. XXX Simpósio Brasileiro de Redes de Computadores*, 2012.