

Eixo Temático: Inovação e Sustentabilidade

**UTILIZAÇÃO DE CONTAINER ALIADA A VIRTUALIZAÇÃO PARA MINIMIZAR
O CONSUMO DE ENERGIA EM UM AMBIENTE DE COMPUTAÇÃO**

**USING CONTAINER ALLIANCE VIRTUALIZATION TO MINIMIZE ENERGY
CONSUMPTION IN A COMPUTER ENVIRONMENT**

Velcir Barcaroli, Silviane Lawall Soares, Ricardo Werlang e Jhon Lenon Ozorio Alves

RESUMO

O número de dispositivos conectados à rede tem exigido cada vez mais recursos de computação nos datacenters. Com o advento da internet das coisas essa demanda deve aumentar cada vez mais. Esses recursos computacionais nem sempre são utilizados em sua totalidade, ficando a maioria das vezes ociosos acarretando em desperdício de energia e por consequência trazendo prejuízos ao meio ambiente. A tecnologia de virtualização de servidores tem otimizado muito a utilização dos recursos computacionais, mas a alocação de alguns recursos exclusivos para cada servidor virtual ainda acaba por ter recursos ociosos. Nesse cenário entra a tecnologia de container, que aliada à de virtualização, permite otimizar mais os recursos computacionais. Nesse trabalho é apresentado essa combinação de tecnologias, a metodologia utilizada para a implementação da solução e os resultados obtidos em testes realizados comparando-se dois cenários, onde um cenário é totalmente virtualizado e o outro uma combinação de virtualização e container.

Palavras-chave: Virtualização, Container, Computação, Docker, Xenserver.

ABSTRACT

The number of devices connected to the network has increasingly required computing resources in datacenters. With the advent of the internet of things this demand must increase more and more. These computational resources are not always used in their totality, being most often idle, resulting in waste of energy and consequently bringing damages to the environment. Server virtualization technology has greatly optimized the use of computing resources, but allocating some unique features to each virtual server still has idle resources. In this scenario comes the container technology, which allied with virtualization, allows to optimize more computational resources. This work presents the combination of technologies, the methodology used to implement the solution and the results obtained in tests performed comparing two scenarios, where one scenario is fully virtualized and the other a combination of virtualization and container.

Keywords: Virtualization, Container, Computing, Docker, Xenserver.

1 INTRODUÇÃO

O intenso desenvolvimento tecnológico tem levado cada vez mais um aumento de dispositivos conectados. Esses dispositivos passa a exigir um crescimento exponencial de tráfego, armazenamento, gerenciamento de quantidades cada vez maiores de dados e principalmente muitos recursos de computação. Com o advento da internet das coisas esse crescimento tende a aumentar ainda mais. Esse avanço tecnológico tem provocado uma demanda crescente de recursos computacionais. A virtualização oferece a otimização no uso dos recursos computacionais, trazendo muitos benefícios. O que leva à simplificação da manutenção, redução do espaço físico necessário e, conseqüentemente, à redução dos custos com condicionamento térmico do ambiente. Mesmo tendo otimizado muito os recursos de computação, a virtualização ainda apresenta um problema que é a alocação exclusiva de recursos para cada máquina virtual.

Neste cenário desafiador de otimização dos recursos computacionais visando buscar maneiras de inovar na economia de energia é que a utilização de container vai encontrando seu espaço, mostrando-se capaz de melhorar a otimização dos recursos da virtualização compartilhando os mesmos entre aplicações completamente diferentes.

A gestão eficiente dos recursos computacionais contribui para a redução dos custos das organizações bem como a redução do consumo de energia com refrigeração e processamento melhora as condições do meio ambiente.

Dessa forma, a proposta do presente trabalho tem por objetivo mostrar que é possível melhorar ainda mais a otimização dos recursos computacionais virtualizados adotando a técnica de containers.

A metodologia utilizada para esse trabalho aborda uma revisão bibliográfica, análise documental e a realização de um laboratório experimental para medir os resultados obtidos.

2 REFERENCIAL TEÓRICO

De acordo com um estudo da Consultoria Gartner (2007), os equipamentos de informática são responsáveis por 2% das emissões de CO₂ em todo o mundo, o que corresponde à quantidade emitida por todos os aviões existentes. Os datacenters são responsáveis por 23% da emissão de gases de toda TI, enquanto os PCs e os monitores atingem 40%. (LUNARDI, FRIO e BRU)

Muitos desses recursos computacionais em datacenters que consomem energia fica a maior parte do tempo ociosos. A virtualização tem melhorando consideravelmente a utilização desses recursos. Mesmo assim a ociosidade acontece, já que mesmo com a virtualização alguns recursos tem alocação exclusiva para determinada máquina virtual. A seguir será apresentado os conceitos de virtualização e container e como a combinação dos dois pode otimizar ainda mais a utilização desses recursos.

2.1 VIRTUALIZAÇÃO DE SERVIDORES

A Virtualização usa software para simular as funcionalidades de hardware e assim criar um sistema computacional virtual. Isso permite que vários sistemas operacionais e aplicações distintas sejam executados em um mesmo servidor (VMWARE).

Uma máquina virtual é uma porção de software isolada com um sistema operacional que pode ter uma aplicação sendo executada nele. Desta forma, cada máquina virtual é independente de outras máquinas virtuais. Para prover este desacoplamento, é utilizada uma camada de software chamada hypervisor, que é responsável pela execução das máquinas

virtuais e pelo gerenciamento do uso dos recursos físicos do servidor (CPU, memória, interfaces de rede, dispositivos de armazenamento, gerenciamento de interrupções de entrada e saída, entre outros), assim como sua alocação para cada uma das máquinas virtuais sob seu controle. (ABRANCHES).

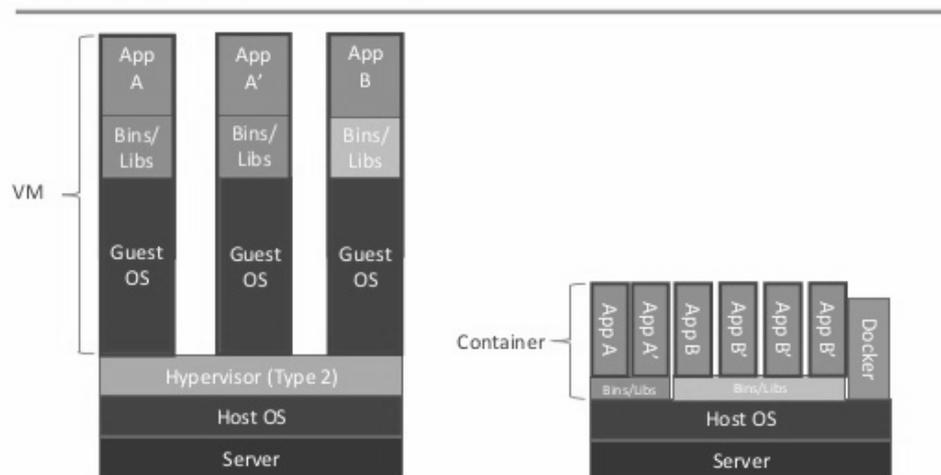
2.1 SERVIÇOS EM CONTAINERS

Containers são ambientes isolados que automatizam a implantação de sistemas. Trata-se de uma solução que permite os administradores de sistemas integrarem e executarem aplicações rapidamente. Seu principal objetivo é proporcionar múltiplos ambientes isolados dentro do mesmo servidor, mas acessíveis externamente via tradução de portas. (TURNBULL).

Como podemos perceber na Figura 1, a solução pode aparentar uma semelhança à estrutura de ambiente virtualizado, porém difere no que tange à necessidade de uma camada intermediária de sistema operacional entre o hospedeiro e as aplicações hospedadas, que é desnecessária, pois ela utiliza o mesmo kernel, criando ambientes isolados a nível de disco, memória e processamento. O recurso de container pode ser implementado pela plataforma Docker. (GOMES e SOUZA)

Figura 1 – Comparação entre container e virtualização

Containers vs. VMs



Fonte: Gomes e Souza.

3 METODOLOGIA

Nesse capítulo é descrito as metodologias utilizadas para se chegar ao resultados e conclusão do presente trabalho.

3.1 TÉCNOLOGIAS UTILIZADAS

Para aplicar os teste de performance dos ambientes foram utilizadas ferramentas que são amplamente utilizadas nos ambientes de produção da maioria dos datacentres. As seguir descreveremos brevemente cada ferramenta utilizada:

- **XenServer** - O Xen foi desenvolvido pelo Systems Research Group da Universidade de Cambridge, e é parte de um projeto maior chamado XenServers, que provê um ambiente de computação global distribuída. O Xen permite compartilhar uma simples

- máquina para vários clientes rodando sistemas operacionais e seus respectivos programas. (ROSSI)
- **Docker** - É uma plataforma que automatiza a implantação de aplicações dentro de ambientes isolados denominados containers (TURNBULL).
 - **Apache Http Server** - Servidor web seguro, eficiente e extensível que forneça serviços HTTP em sincronia com os padrões HTTP atuais. (APACHE).
 - **PHP** - É uma linguagem de script open source de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML. (PHP).
 - **PostgreSQL** - É um poderoso sistema de banco de dados relacional de código fonte aberto muito utilizado para o armazenamento de dados de sistema web. (POSTGRESQL).
 - **AB (Apache HTTP server benchmarking tool)** - Ab é uma ferramenta para testes de servidores web. Ele é projetado para verificar como sua instalação atual do servidor se comporta. Isso mostra especialmente quantos pedidos por segundo sua instalação do é capaz de servir simultaneamente. (AB).
 - **TOP** - Comando do Linux que exibe os processos em execução no sistema. Uma das ferramentas mais importantes para um administrador do sistema. É amplamente utilizado para monitorar a carga do sistema. (FERREIRA).

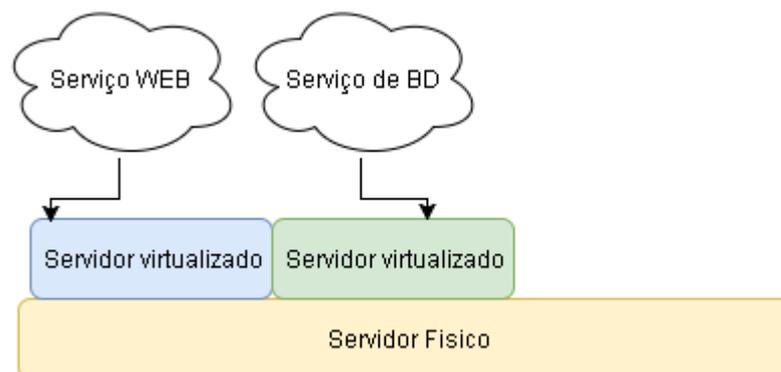
3.2 CENÁRIO DE TESTES

Os serviços oferecido em nosso ambiente de testes foi uma aplicação web desenvolvida em PHP e o servidor de WEB é o Apache. Essa aplicação acessa um banco de dados PostgreSQL. Para a realização dos testes de performance foi montado laboratórios com dois cenários diferentes. Para ambos os cenários utilizou-se um servidor XEON Quad Core com 16GB de memória RAM.

No primeiro cenário foram usadas duas máquinas virtuais criadas no XenServer onde uma das maquinas virtuais tem a função de rodar a aplicação web e na outra roda o banco de dados. Ambas as máquinas virtuais, foi configurada com 1GB de memória RAM e quatro núcleos.

Na figura 2 observa-se o esquemático da configuração e a distribuição dos recursos em cada maquina virtual.

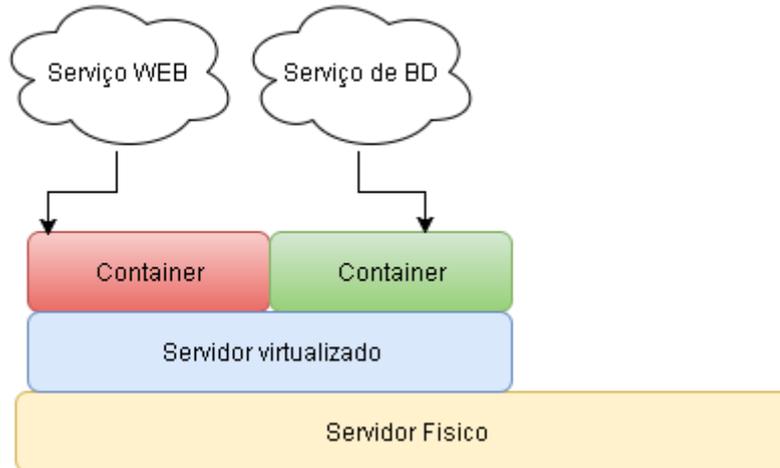
Figura 2 – Esquemático com virtualização



Fonte: autores.

O segundo cenário utilizou-se uma única máquina virtual criada no XenServer com 2GB de memória RAM e quatro núcleos. Dentro dessa maquina virtual foram criados dois containers onde no primeiro container rodou a aplicação web e no segundo o banco de dados. A figura 3 mostra uma única máquina virtual com dois containers onde estão rodando os serviços.

Figura 3 – Esquemático com virtualização e container



Fonte: autores.

O objetivo dos testes foi medir o consumo de recursos de CPU e memória na máquina física enquanto a aplicação era submetida determinadas cargas de requisições. Para fazer as requisições de serviços web foi utilizado o “Apache HTTP server benchmarking tool”, esse aplicativo stressa o servidor simulando o número de usuários e de requisições por usuário. No capítulo a seguir são analisados os resultados obtidos durante a realização dos testes.

4 ANÁLISE DOS RESULTADOS

A ferramenta “ab” tem a seguinte sintaxe “ab -n 100 -c 10 http://host/” onde o parâmetro “-n” refere-se ao número de requisições que serão enviadas, enquanto que o “-c” refere-se ao número de requisições simultâneas. Na tabela a seguir podemos analisar os resultados obtidos quando os cenários citados anteriormente foram submetidos aos testes de performance. Na primeira coluna são mostrados os parâmetros utilizados para os testes. Na segunda coluna pode-se observar a utilização de CPU e memória no primeiro cenário com duas máquina virtualizadas. Na terceira coluna aparecem os dados obtidos com a aplicação dos testes no cenário dois com a utilização de containers em uma única máquina virtualizada.

Tabela 1 – Resultados obtidos

Parâmetros de testes	Virtualização		Virtualização e Container	
	CPU	RAM	CPU	RAM
ab -n 100 -c 10	38,3%	739772	35,5%	643280
ab -n 200 -c 10	42,9%	741761	39,4%	650668
ab -n 300 -c 10	45,2%	773883	41,8%	670028
ab -n 100 -c 20	53,1%	827010	49,2%	731868
ab -n 200 -c 20	54,4%	876308	51,2%	762000
ab -n 300 -c 20	58,3%	902841	56,3%	778312
Média	48,7%	810262	45,6%	706026

Fonte: autores.

Baseados nos testes submetidos observou-se que a utilização de CPU ficou em média 6,8% menor no cenário onde foi utilizado a combinação da virtualização com container. A utilização de memória RAM nos trouxe um resultado melhor ainda ficando em 14,7% menos utilizada.

Nesse cenário de testes a utilização de container juntamente com a virtualização mostrou uma utilização melhor dos recursos físicos de computação que por consequência representam um menor consumo de energia trazendo redução de custos e menos danos ao meio ambiente.

Este trabalho não pode ser conclusivo e sugere-se a coleta de mais dados em períodos maiores e em ambientes mais complexos que estejam em produção já que as tecnologias utilizadas são amplamente utilizadas.

REFERÊNCIAS

AB. **Apache HTTP server benchmarking tool.** <https://httpd.apache.org/docs/2.4/programs/ab.html>. 2017. [Acessado em 30/06/2017]

ABRANCHES, Marcelo Cerqueira de. **Um Mecanismo de Auto Elasticidade com base no Tempo de Resposta para Ambientes de Computação em Nuvem baseados em Containers.** Brasília - DF. 2016.

APACHE. **HTTP Server Project.** <https://httpd.apache.org/>. 2017. [Acessado em 30/06/2017]

BARHAM, P. et al. **Xen and the art of virtualization.** In: SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles. Bolton Landing, NY, USA: ACM Press, 2003. p. 164–177.

FERREIRA, Ricardo. **Comandos Linux: Dominando o comando TOP.** <https://www.linuxdescomplicado.com.br/2013/12/comandos-linux-dominando-o-comando-top.html>. 2013. [Acessado em 30/06/2017]

GOMES, Rafael. SOUZA, Rodrigo. **Docker - Infraestrutura como código, com autonomia e replicabilidade.** Universidade Federal da Bahia (UFBA). Salvador.

LUNARDI, Guilherme Lerch. FRIO, Ricardo Saraiva. BRU, Marília de Marco. **Tecnologia da informação e sustentabilidade: levantamento das principais práticas verdes aplicadas à área de tecnologia.** Universidade Federal do Rio Grande, Rio Grande, Brasil. 2011.

PHP. **O que é o PHP?** https://secure.php.net/manual/pt_BR/intro-what-is.php. 2017. [Acessado em 30/06/2017]

POSTGRESQL. **PostgreSQL About.** <https://www.postgresql.org/about/>. 2017. [Acessado em 30/06/2017]

ROSSI, Fábio Diniz. **Alocação Dinâmica de Recursos no Xen.** Pontifícia Universidade Católica do Rio Grande do Sul Faculdade de Informática. Porto Alegre - RS. 2008.

TURNBULL, James. **The Docker Book: Containerization is the new virtualization.** 2014.

VMWARE. **Virtualization.** <https://www.vmware.com/br/solutions/virtualization.html>. 2017. [Acessado em 30/06/2017]