

Análise comparativa de desempenho entre as tecnologias de virtualização e containerização para hospedagem de serviços web

Diego William Fernandes de Carvalho e Tiago Alexandre Dócusse. Instituto Federal de Educação, Ciência e Tecnologia de São Paulo. diego.william@aluno.ifsp.edu.br

Contêineres, Máquinas Virtuais e Serviços Web

Introdução

Com o aumento na capacidade de processamento dos computadores e na quantidade de memória principal disponível, a virtualização se tornou uma tecnologia interessante na hospedagem de serviços. A virtualização compartilha recursos para simplificar a gestão de atividades, atendendo várias demandas da área tecnológica (GOLDEN e SCHEFFY, 2008). A containerização e as máquinas virtuais são dois exemplos dessa tecnologia. A máquina virtual é ideal para aplicações que precisam de diferentes sistemas operacionais ou diferentes versões do mesmo para ser executada, enquanto no contêiner, vários sistemas isolados compartilham o mesmo sistema operacional do hospedeiro (CHAE, LEE e LEE, 2017).

Objetivos

Este estudo busca analisar o desempenho e o consumo de recursos de serviços *web* hospedados em contêineres e máquinas virtuais. Foi feita uma comparação sobre qual ambiente de virtualização obteve o melhor resultado ao utilizar recursos como processador e memória.

Materiais e Métodos

Foi criado um *site* utilizando o PHP (com o framework Laravel), pois além de ser possível utilizá-lo em sistemas operacionais Linux e contêineres baseados em Linux, ele também oferece uma boa arquitetura para se trabalhar com banco de dados (SHARMA, 2020).

Utilizando o Microsoft SQL Server 2019, foi criado e preenchido (com vários registros), um banco de dados teste, com 8 tabelas e relacionamentos N para N entre elas. Após isso, foi configurado a máquina virtual, com sistema operacional Linux, e também o contêiner, utilizando o *Docker*. Estes ambientes foram escolhidos pois as comparações entre máquina virtual e contêiner fornece aos profissionais informações sobre a qualidade dos

serviços disponibilizados para virtualização (YADAV, SOUSA e CALLOU, 2018).

Visando obter uma análise de processamento e memória, o *site* realizou diferentes consultas SQL (*join*, *exists*, *union*, *not exists* e *aninhada*) no banco teste, que foi hospedado na máquina virtual e no contêiner.

Resultados e Discussão

Devido à diferença de tempo retornado a cada teste, as consultas foram processadas cinco vezes, obtendo uma média da velocidade de processamento e também do valor, em *megabytes*, que cada ambiente de virtualização consumiu do servidor.

Para simular cenários com diferentes quantidades de recursos, foram obtidos resultados de diferentes configurações de núcleo e memória na máquina virtual e no contêiner.

Na Figura 1, em (a), tem-se os resultados para o processamento das consultas ao utilizar o *Docker*; em (b), os resultados para as máquinas virtuais; em (c), o resultado para o consumo de memória do hospedeiro ao processar as consultas utilizando o *Docker*; em (d), o resultado para as máquinas virtuais.

O servidor possuía 8000 megabytes de memória. Além disso, o sistema operacional era o Windows 10.

Figura 1 - Gráficos com comparativos dos resultados

Ambiente	Teste 1	Teste 2	Teste 3	Teste 4	Teste 5	Teste 6
Docker com WSL	52,6 ms	33 ms	38,6 ms	2 ms	9,4 ms	23,2 ms
Docker sem WSL e 4 núcleos	44,70 ms	30,57 ms	44,68 ms	2 ms	8,33 ms	28,72 ms
Docker sem WSL e 2 núcleos	39,72 ms	26,33 ms	39,03 ms	2 ms	7,19 ms	25,02 ms
Docker sem WSL e 1 núcleo	48,6 ms	37,43 ms	51,32 ms	2 ms	7,38 ms	23,16 ms

(a)

Ambiente	Teste 1	Teste 2	Teste 3	Teste 4	Teste 5	Teste 6
Máquina Virtual - 4 núcleos e 4096 mb	81,2 ms	51,2 ms	85,3 ms	3,06 ms	11,92 ms	24,78 ms
Máquina Virtual - 2 núcleos e 5280 mb	77,1 ms	54,7 ms	76,6 ms	2,5 ms	12,8 ms	26,3 ms
Máquina Virtual - 1 núcleo e 4096 mb	68,2 ms	47,8 ms	71,4 ms	2,2 ms	11,4 ms	26,1 ms

(b)

Ambiente	Teste 1	Teste 2	Teste 3	Teste 4	Teste 5	Teste 6
Docker com WSL	4801,8 mb	4448 mb	4728,7 mb	1815 mb	3505 mb	4853,3 mb
Docker sem WSL e 4096 mb	1713,6 mb	1474,5 mb	1649,3 mb	16,3 mb	783,1 mb	2004,5 mb
Docker sem WSL e 2048 mb	2284,1 mb	1957,7 mb	2831,3 mb	20,7 mb	786,3 mb	3716,3 mb

(c)

Ambiente	Teste 1	Teste 2	Teste 3	Teste 4	Teste 5	Teste 6
Máquina Virtual - 4 núcleos e 4096 mb	2565,7 mb	1968,6 mb	2646,2 mb	15 mb	807 mb	2643,6 mb
Máquina Virtual - 2 núcleos e 5280 mb	2216,2 mb	1995,6 mb	2336,2 mb	15 mb	755,8 mb	2759,9 mb
Máquina Virtual - 1 núcleo e 4096 mb	2543,1 mb	1976,9 mb	2536,2 mb	15 mb	797,2 mb	2968,7 mb

(d)

Observando as imagens, nota-se o diferencial de consumo do *Docker* com o WSL (Windows Subsystem for Linux) ativado e desativado. O WSL executa um processo que virtualiza um contêiner linux nativamente, sem emulação, o que consome mais memória e processamento. Ao desativá-lo, o consumo de recurso é menor, pois o Docker passa a utilizar um hospedeiro Windows. Observa-se também que ao disponibilizar menos memória ao contêiner, o consumo do *Docker* aumenta, pois ele passa a funcionar no seu limite de capacidade.

Comparando as diferentes configurações das máquinas virtuais, o sistema com 4 núcleos obteve a melhor velocidade de processamento, devido a sua capacidade de dividir mais tarefas do que as outras máquinas que possuíam menos núcleos disponibilizados. No entanto, para os testes realizados neste trabalho, não se obteve muita diferença no consumo de memória do hospedeiro.

Comparando os resultados da máquina virtual com o *Docker*, ela foi mais lenta em todas as consultas. Isso acontece porque a máquina virtual emula um *hardware*, enquanto o contêiner é executado direto na máquina hospedeira.

Conclusões

A hospedagem do *site* no *Docker* (com o WSL desativado) consumiu menos memória e foi mais rápida em relação à execução do mesmo em uma máquina virtual. A quantidade de núcleo disponibilizada também se mostrou um diferencial no desempenho.

Agradecimentos

Os autores gostariam de agradecer ao Instituto Federal de Educação, Ciência e Tecnologia de

São Paulo (IFSP), Câmpus Barretos, pela bolsa recebida para o desenvolvimento do projeto.

Bibliografia

GILLANI, K e LEE, J. **Comparison of Linux virtual machines and containers for a service migration in 5G multi-access edge computing.** *Korean Institute of communications and Information Sciences*, Coreia do Sul, 17 de dez. de 2018.

GOLDEN, B. e SCHEFFY, C. **Virtualization for dummies.** *Sun AMD – Special Edition.* Indianapolis: Wiley Publishing Inc., 2008.

SHARMA, R. **Why Laravel is the best PHP framework to use in 2020.** FreeCodeCamp, 2020. Disponível em: <<https://www.freecodecamp.org/news/why-laravel-is-the-best-php-framework-to-use-in-2020/>>. Acesso em: 1 de junho de 2021.

YADAV, R., SOUSA, E. e CALLOU, G. **Performance comparison between Virtual Machines and Docker Containers.** *IEE Latin America Transactions*, New York, 8 de agosto de 2018, vol. 16, no. 8, pp. 2282-2288.