

Uma avaliação de infraestruturas de sistemas de informações corporativos de alta disponibilidade e redundância

David Coelho dos Santos, Jacyana Suassuna Nunes, Brunno Santiago, André Campos Bezerra, Itamir de Moraes Barroca Filho

Instituto Metr pole Digital – Universidade Federal do Rio Grande do Norte (UFRN)

Caixa Postal 1524 – 59078-970 – Natal – RN – Brasil

{david.coelho,jacyana,brunno.santiago,andre.campos,itamir}@imd.ufrn.br

***Abstract.** Enterprise information systems are increasingly present in public and private institutions around the world. Thus, with the increase, importance and frequent use of these systems in these environments, grows the concern with the infrastructure of servers, since it is one of the main responsible for their availability to their users. In this context, this paper aims to describe an evaluation based on the execution of stress tests in two scenarios of infrastructures of information systems of high availability and redundancy.*

***Resumo.** Os sistemas de informa es corporativos est o cada vez mais presentes nas institui es p blicas e privadas em todo o mundo. Assim, com o aumento, import ncia e uso frequente desses sistemas nesses ambientes, cresce a preocupa o com a infraestrutura de servidores, j  que ela   uma das principais respons veis pela disponibilidade deles para seus usu rios. Nesse contexto, este trabalho tem o objetivo de descrever uma avalia o baseada na execu o de testes de estresse em dois cen rios de infraestruturas de sistemas de informa o de alta disponibilidade e redund ncia.*

1. Introdu o

Atualmente os sistemas de informa o est o cada vez mais presentes em organiza es p blicas e privadas em todo mundo. Al m disso, de acordo com o Gartner [CIO 2016], as plataformas de tecnologia digital, em especial os sistemas de informa o, s o umas das principais tend ncias de tecnologias estrat gicas para 2017. Isso deve-se principalmente a import ncia desses sistemas de informa es no controle dos processos de neg cio das organiza es e produ o de informa es geradas pelos dados capturados por eles. Segundo Laudon [Laudon, 2010], al m dos sistemas de informa o ajudarem nos processos, eles produzem as informa es que as organiza es precisam ter para tomar decis es, controlar suas opera es, analisar problemas e criar novos produtos e servi os. Dessa forma, com o aumento, import ncia e uso frequente de sistemas de informa es para esses fins, cresce a preocupa o com a infraestrutura de servidores, j  que ela   uma das principais respons veis pela disponibilidade dos sistemas para utiliza o dos seus usu rios.

Considerando o uso de sistemas de informa es em organiza es p blicas, no Instituto Metr pole Digital (IMD) da Universidade Federal do Rio Grande do Norte, existem cinco sistemas em produ o - Inova, respons vel pelo controle dos processos de incubaq o na Incubadora Inova Metr pole; o Keys, respons vel pelo controle das reservas de salas e entregas de chaves no IMD, o Probat rio, respons vel pelo

acompanhamento dos servidores técnicos e docentes em estágio probatório; o Patrimônio, responsável pelo controle patrimonial dos bens do Instituto; e o Labs, responsável pelo acompanhamento das produções e planejamento dos projetos de pesquisa realizados pelos núcleos integradores do IMD.

Assim, com o aumento da demanda por serviços providos por esses sistemas houve uma elevação nos acessos a eles, o que aumentou a necessidade por disponibilidade e segurança. Esses sistemas de informação são responsáveis pela implementação de mapeamentos específicos de processos de cada área e atualmente são acessados por uma faixa de 500 usuários. Como eles também controlam determinadas ações, como por exemplo, a entrega de chaves e reserva de sala, é primordial que eles sempre estejam disponíveis quando os usuários necessitarem dos serviços providos por eles. Dessa forma, com esse aumento de demanda, surgiram problemas como: sobrecarga no servidor; diminuição de desempenho dos serviços, aumentando a lentidão durante os picos de acessos; e indisponibilidade dos sistemas. Portanto, no intuito de melhorar a disponibilidade dos sistemas de informação, a Diretoria de Tecnologia da Informação implementou para avaliação dois cenários com duas soluções de infraestrutura robustas com alta disponibilidade e redundância, por meio de uso de tecnologias atuais tais como, por exemplo, containers.

Dessa forma, o objetivo deste trabalho é reportar os resultados da avaliação realizada por meio de testes de estresse executados nas soluções implementadas, e a experiência de criação dessas infraestruturas de sistemas de informação de alta disponibilidade e redundância. Com essa avaliação foi possível indicar qual o melhor cenário para resolver os problemas gerados pelos picos de acessos dos usuários nos sistemas do IMD. Por fim, esse trabalho é estruturado da seguinte forma: na Seção 2 apresenta-se o método para criação da infraestrutura proposta; na Seção 3 apresenta-se os resultados dos testes de estresse para avaliação dessa infraestrutura; e por fim, na Seção 4 são apresentadas as conclusões deste trabalho.

2. Método

Nesta seção são apresentados os cenários implementados para a infraestrutura de sistemas de informação do IMD, detalhando as tecnologias utilizadas e como os servidores foram organizados. Para propor a avaliação das infraestruturas de sistemas de alta disponibilidade e redundância, a metodologia realizada adotou os seguintes passos: estudo das tecnologias atuais de infraestrutura de sistemas; estudo sobre Apache Mod_jk, Tomcat e Docker para sistemas de informação; prova de conceito com Apache Mod_jk e Tomcat e os sistemas do IMD; prova de conceito com Docker com os sistemas do IMD; realização de testes de estresse com o JMeter para avaliação da disponibilidade dos sistemas nas infraestruturas propostas; e avaliação dos resultados obtidos.

2.1. Cenário com Apache e Mod_jk

No que diz respeito ao cenário com Apache e Mod_jk, ele é composto por três máquinas virtuais, denominadas dubai.imd.ufrn.br com 4 vCPUs, 4GB da RAM, SO CentOS 7 e função de balanceador; pequim.imd.ufrn.br e xangai.imd.ufrn.br ambos com 4 vCPUs,

4GB da RAM, CentOS 7 e função de servidor de aplicação. A máquina virtual `dubai.imd.ufrn.br` atua como o balanceador de requisições e é responsável por direcioná-las para os serviços disponíveis nas máquinas `xangai.imd.ufrn.br` e `pequim.imd.ufrn.br`. Para esse balanceamento são utilizados o Servidor Web Apache e o conector `Mod_jk`. Em poucas palavras, o Apache aguarda solicitações HTTP do cliente e quando essas solicitações chegam ele faz o que é necessário para atender às solicitações, fornecendo o conteúdo desejado [Apache 2016].

2.2. Cenário com Docker

Atualmente o uso de containers vem se tornando frequente em novos projetos, e com uso dessa tecnologia, tudo o que é necessário para fazer um software funcionar é embalado em recipientes isolados [Docker 2017]. Dessa forma, pensando em containers, foi criado um cenário com Docker que é composto por três máquinas virtuais, denominadas `madrid.imd.ufrn.br`, com 4 vCPUs, 4GB da RAM, CentOS 7 e função de balanceador; `roma.imd.ufrn.br` e `vенеza.imd.ufrn.br` cada uma com 4 vCPUs, 4GB da RAM, CentOS 7 e função de docker manager/worker/leader. Ainda sobre esse cenário, ele utiliza um recurso denominado *swarm*, presente a partir do Docker 1.12, para efetuar a paralelização e orquestração de containers. As máquinas `roma.imd.ufrn.br` e `vенеza.imd.ufrn.br` consistem em dois nós com containers réplicas e distribuídos a fim de manter a alta disponibilidade e redundância dos sistemas do IMD. Esta abordagem permite que cada sistema seja executado e contido em seu pool de recursos e evita que caso um sistema pare seja necessário parar todos os outros que rodam no mesmo servidor de aplicação Tomcat.

3. Resultados

Para a realização da avaliação dos cenários propostos utilizou-se o software JMeter versão 3.1 r1770033. Com esse software foram realizados testes de estresse nas aplicações, simulando-se a quantidade de usuários simultâneos acessando ao mesmo tempo cada um dos cinco sistemas disponíveis no ambiente. Foram adicionados os seguintes componentes para cada plano de teste dos sistemas: Grupo de usuários virtuais - com 1500 usuários por sistemas; Componente de Requisição HTTP - Para configuração da url destino em cada sistema; Gerenciador de cookie HTTP - Para apagar cookies em cada uma das interações.

A fim de testar a escalabilidade do ambiente e das aplicações, foram executados testes com aumento de carga gradativa, considerando 500, 1000 e 1500 usuários virtuais simultâneos por aplicação. O objetivo disso foi determinar o comportamento do balanceador e das aplicações em condições normais e em altos picos de requisições. Por fim, para a avaliação comparativa entre os cenários foram considerados os seguintes critérios: latência e tempo de resposta.

3.1. Resultados para o cenário com Apache e Mod_jk para 1500 usuários simultâneos por aplicação

Durante o período de testes, foram observados aumento do atraso de respostas às requisições (latência) com pico máximo de 3200 ms e um pico máximo de tempo de

resposta de 36400 ms, conforme gráficos da Figura 01. O servidor balanceador teve carga máxima em torno de 950 acessos por segundo. Mesmo durante os intervalos de picos, não foi observada em momento algum a falta de respostas às requisições. Também não houveram períodos de indisponibilidade e todos os sistemas estavam totalmente acessíveis, mesmo quando observado um relativo atraso no tempo de resposta durante alguns períodos do teste. De modo geral, a infraestrutura correspondeu de forma satisfatória com médias de tempo para todos os indicadores analisados, conforme os seguintes tempos: tempo de Hit no balanceador – 448 ms, 54 ms; latência – 1.528,32 ms; tempo de resposta – 8.835,15 ms.

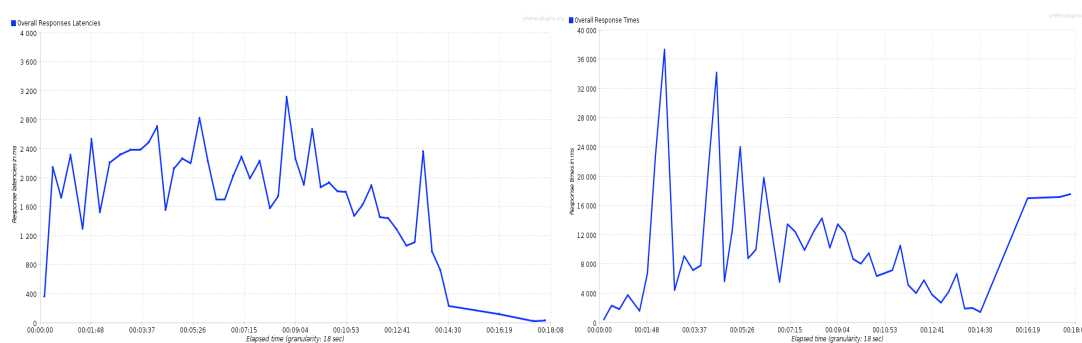


Figura 01 - Gráficos de latência (direito) e tempo de resposta (esquerdo) referente aos testes no ambiente Apache-Mod_jk.

3.2. Resultados para o cenário com Docker e Containers para 1500 usuários simultâneos por aplicação

Durante todo o período de testes, foram observados aumento do atraso de respostas às requisições (latência) com pico máximo de 6000 ms e um pico máximo de tempo de resposta de 51000 ms, conforme apresentado na Figura 02. O servidor balanceador teve carga máxima em torno de 710 acessos por segundo. Em relação ao comportamento dos sistemas, a abordagem Docker – containers obteve os seguintes tempos médios para os indicadores: tempo de Hit no proxy - 343,18 ms; latência: 36.838,97 ms; tempo de resposta: 179.623,19 ms.



Figura 02 - Gráficos de latência (direito) e tempo de resposta (esquerdo) referente aos testes no ambiente Docker.

4. Conclusões

Esse trabalho apresentou uma avaliação de duas infraestruturas de sistemas corporativos de alta disponibilidade e redundância, considerando como variáveis de avaliação a média do tempo de latência e resposta das aplicações. Nessa avaliação foi utilizado o JMeter para execução dos testes de estresse e um plano de testes considerando um total de 1500 usuários simultâneos, conforme apresentação na Seção 3. Além disso, para cada cenário proposto, as máquinas possuíam recursos de memória e processamento semelhantes. Com os resultados obtidos foi possível perceber que o cenário proposto com Docker se apresentou 22x mais lento com relação ao tempo médio de resposta e com uma latência 36x maior.

Por fim, como trabalhos futuros, pretende-se continuar os estudos das tecnologias de Docker e a realização de novas avaliações considerando números diferentes de usuários. Além disso, também pretende-se utilizar o Docker no desenvolvimento de sistemas e na integração contínua.

Referencias

- CIO (2016), <http://cio.com.br/tecnologia/2016/10/26/gartner-aponta-tendencias-estrategicas-para-2017-na-area-de-ti/>. Acesso em 29 de março de 2017.
- LAUDON KENNETH C.; LAUDON, J. P. (2010) Management Information Systems: managing the digital system. 12th. ed. [S.l.]: Pearson.
- APACHE (2016), http://tomcat.apache.org/connectors-doc/webserver_howto/apache.html. Acesso em 29 de março de 2017.
- DOCKER (2017), <http://https://www.docker.com/what-docker>. Acesso em 30 de março de 2017.