

INF01015 - Gerencia de Aplicações em Rede

Desenvolvimento de ferramenta de gerência distribuída baseada no Iperf3

Fabício S. Almeida¹, Eduardo Bassani¹, Oberdan Santos¹

¹Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil

`fsalmeida@inf.ufrgs.br, ebchandelier@inf.ufrgs.br, ocsantos@inf.ufrgs.br`

1. Introdução

Este trabalho consiste no desenvolvimento e documentação de uma ferramenta distribuída para gerenciamento de dispositivos de rede utilizando microprocessos e como base de recursos o Iperf3, que consiste em um aplicativo/biblioteca de testes de velocidade e qualidade de conexões de rede.

A ferramenta de gerencia desenvolvida visará a atuação em redes onde a conexão entre dois pontos deve ser monitorada em relação a sua banda de transferência, sua qualidade de transmissão (porcentagem de retransmissões observada) e uso de CPU no dispositivo monitorado provendo as seguintes funcionalidades.

- **Monitoramento de banda disponível:** A ferramenta fica monitorando a banda disponível ponto a ponto analisado e ecoando na interface, ela também sinaliza de forma visual por meio de coloração da caixa de texto onde o valor é exibido se o mesmo está abaixo do valor mínimo definido nas configurações de monitoramento.
- **Monitoramento de índice de retransmissões:** A ferramenta fica monitorando o índice de retransmissões (em porcentagem) ponto a ponto analisado e ecoando na interface, ela também sinaliza de forma visual por meio de coloração da caixa de texto onde o valor é exibido se o mesmo está acima do valor máximo definido nas configurações de monitoramento.
- **Monitoramento de CPU remota:** A ferramenta fica monitorando o consumo de CPU do dispositivo monitorado e ecoando na interface, ela também sinaliza de forma visual por meio de coloração da caixa de texto onde o valor é exibido se o mesmo está acima do valor máximo definido nas configurações de monitoramento.

Todos os parâmetros medidos são dispostos também em gráficos (Figure 1) e (Figure 2), afim de fornecer uma melhor experiencia de uso ao gerenciador da rede analisada.

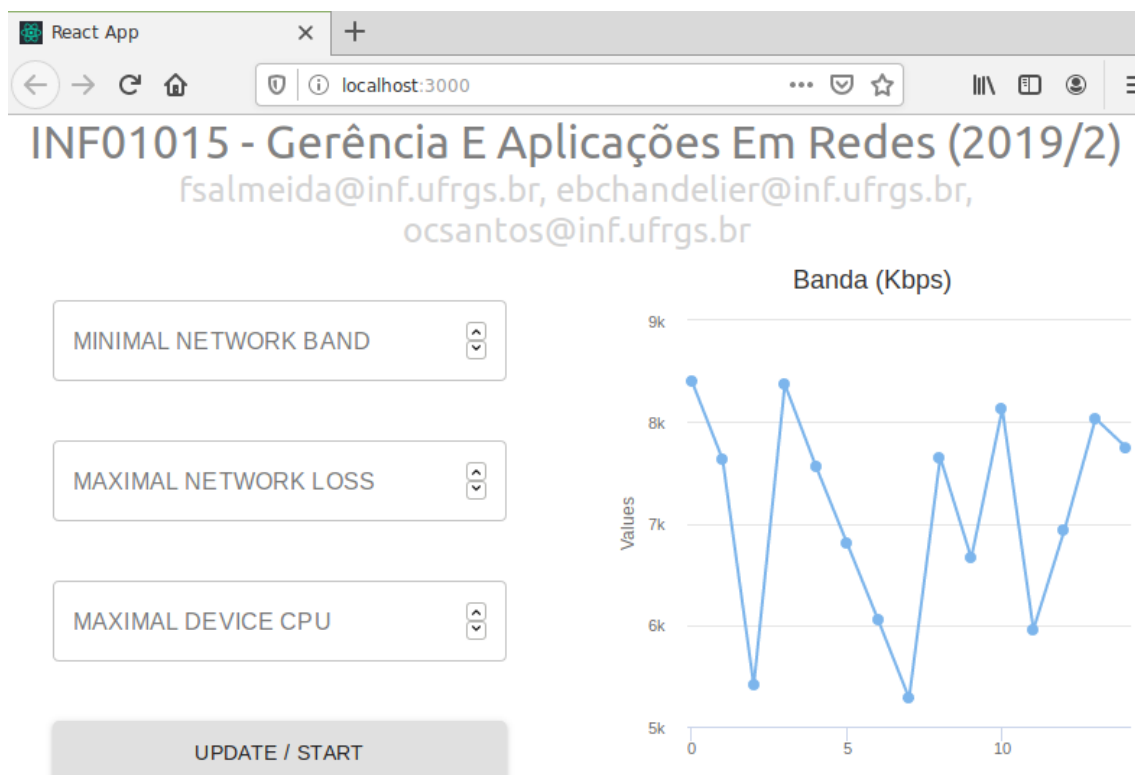


Figure 1. Interface parte 1

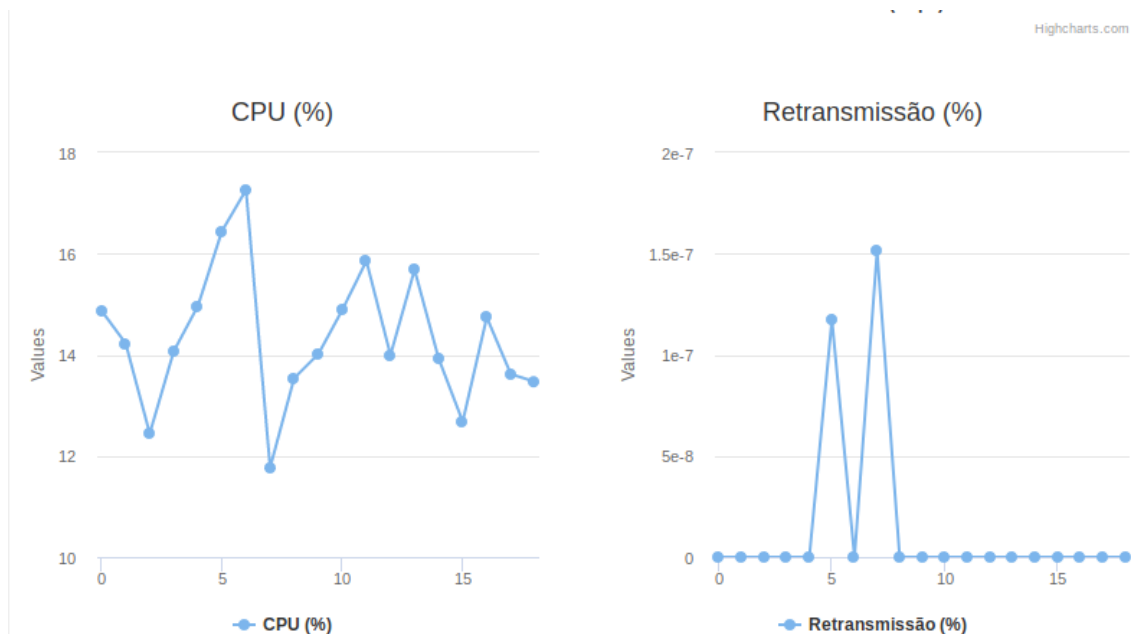


Figure 2. Interface parte 2

2. Forma de implementação utilizada

A ferramenta de gerência produzida teve como princípio de desenvolvimento o uso de micro serviços, onde utilizamos a ferramenta Docker como ambiente de execução. A

ferramenta consiste em dois micro serviços detalhados a seguir (Figure 3):

- **Micro serviço 1 - Backend e frontend:** Micro serviço responsável pela execução do backend desenvolvido em Python com auxílio da biblioteca Iperf3, onde executamos uma aplicação que realiza conexões a dispositivos monitorados conforme a demanda e realiza a obtenção de dados para mostrar na interface, além disto nela é executada uma aplicação frontend desenvolvida em Javascript com o auxílio do framework React que é responsável pela entrada de configurações e emissão de valores medidos e alertas.
- **Micro serviço 2 - Ponto de conexão remoto:** Micro serviço responsável pela execução de uma aplicação desenvolvida em Python com o auxílio da biblioteca Iperf3 que serve como ponto de conexão e coleta de dados para monitoramento. Não realiza nenhuma função de alerta emissão de resultados.

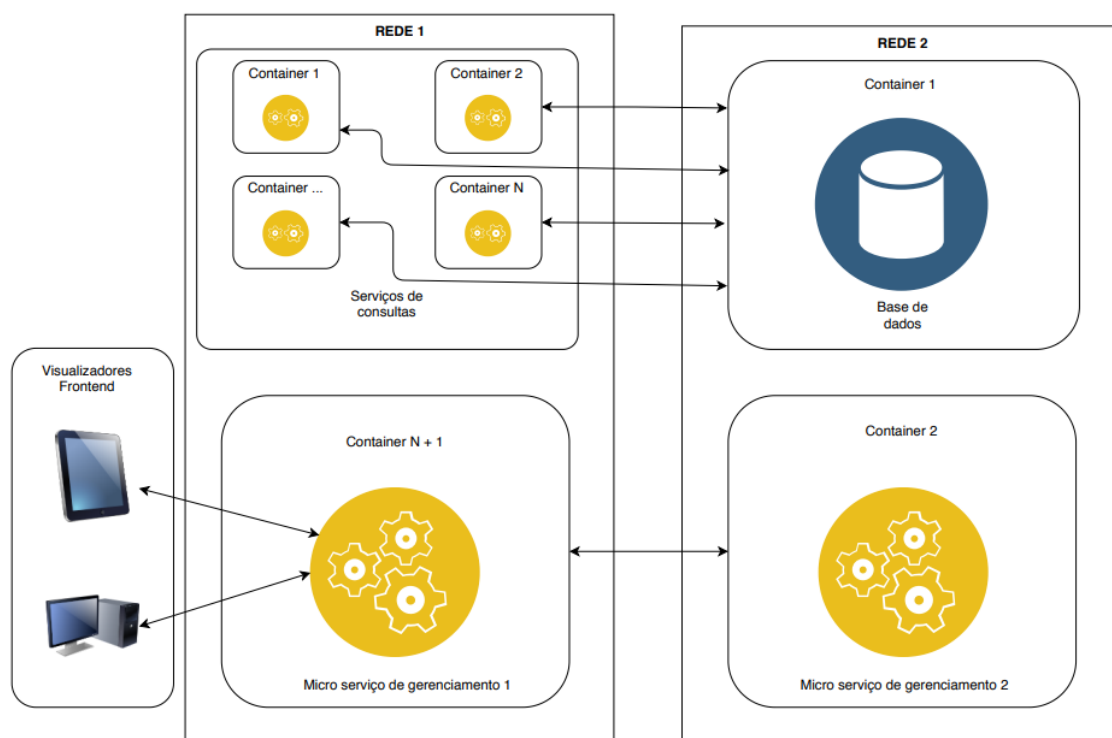


Figure 3. Arquitetura de gerenciamento

3. Guia de implantação

Para implantação do sistema de gerencia desenvolvido precisamos de pelo menos um computador com sistema operacional de distribuição Linux 19.04 ou superior, onde pode ser executado tanto o ambiente modelado como Rede 1, como o ambiente modelado como Rede2 (Figure 3). A seguir apresentaremos os passos a serem executados para execução da aplicação:

- **1. Instalar o Git:** O sistema de gerencia encontra-se no gitHub, onde para obtê-lo você necessitará de cliente git. Caso você não o tenha instalado pode fazer por meio da linha de comando, ou se já tiver instalado pular para o próximo passo:

- *sudo apt install git*
- **2. Obter ferramenta:** Para realizar o clone da ferramenta você deve executar a seguinte linha de comando:
 - *git clone https://github.com/fszczesny/gerenciaT2.git*
- **3. Preparação do ambiente:** O sistema de gerencia foi projetado utilizando micro serviços, desta forma necessitamos a instalação de algumas ferramentas de execução de dependências externas. Para simplificar o processo foi criado o script de execução chamado "config.sh", onde para executa-lo você de estar por meio do terminal de comandos dentro do repositório clonado no passo anterior e executar os seguintes comandos:
 - *chmod 777 config.sh*
 - *./config.sh*
- **4. Implantação ambiente Rede 2:** Para implantação do conjunto de serviços modelados como Rede 2 (Figure 3) foi criado o script de execução chamado "runServer.sh", onde para executa-lo você de estar por meio do terminal de comandos dentro do repositório clonado no passo anterior e executar os seguintes comandos:
 - *cd src/*
 - *chmod 777 runServer.sh*
 - *./runServer.sh*
- **5. Implantação ambiente Rede 1:** Para implantação do conjunto de serviços modelados como Rede 1 (Figure 3) primeiramente você deve editar o arquivo "client.docker-compose.yml" nas linhas "SERVER-IP" e colocar o IP da maquina que está executando o ambiente Rede 2. Logo após para implantação do serviço foi criado o script de execução chamado "runClient.sh", onde para executa-lo você de estar por meio do terminal de comandos dentro do repositório clonado no passo anterior e executar os seguintes comandos:
 - *cd src/*
 - *chmod 777 runClient.sh*
 - *./runClient.sh*
 - *cd front-charts*
 - *npm install*
 - *npm start*

Para escalar clientes de consulta no banco de dados da aplicação monitorada foi desenvolvido um script de replicação dos containers de consulta chamado "runScaleClient.sh", onde para executa-lo você de estar por meio do terminal de comandos dentro do repositório clonado no passo anterior e executar os seguintes comandos:

- *cd src/*
- *chmod 777 runScaleClient.sh*
- *./runScaleClient.sh X*

Onde "X" é o numero de clientes de consulta que serão iniciados.

- **6. Acesso a interface:** Para acesso a interface de visualização dos resultados obtidos necessitamos que em qualquer dispositivo conectado a mesma rede do ambiente Rede 1 ocorra a conexão por meio de um browser no endereço:
 - *IP-FERRAMENTA:3000*

Onde o IP-FERRAMENTA se refere ao IP do computador que executa a aplicação de gerencia localizada no ambiente Rede 1.

4. Conclusão

Podemos concluir que a gerencia de rede com a utilização do Iperf como base é um tanto quanto limitada quando se utiliza apenas ele como ferramenta, pois o mesmo possui um conjunto de funcionalidades pequeno quando se leva em conta o numero de informações retornadas devido seu único foco em medições de banda e estado da rede.

5. Referências

Ao longo da realização deste trabalho de pesquisa foram consultados alguns materiais disponíveis em sites. Inicialmente foram consultados materiais sobre Docker em fontes como:

- Docker website. Docker, 2019: www.docker.com - acesso em novembro de 2019.
- Docker Hub. Iperf3 Docker, 2019: hub.docker.com/r/networkstatic/iperf3 - acesso em novembro de 2019.

Após foram realizados consultas em outros materiais com o foco no desenvolvimento. São eles:

- Iperf website. Iperf, 2019: iperf.fr - acesso em novembro de 2019.
- Pypi. Biblioteca Iperf3, 2019: pypi.org/project/iperf3 - acesso em novembro de 2019.
- Flask. Introdução ao desenvolvimento web com Python, 2019: pythonclub.com.br/what-the-flask-pt-1-introducao-ao-desenvolvimento-web-com-python.html - acesso em novembro de 2019.
- NodeJs. Documentação NodeJs, 2019: nodejs.org/en - acesso em novembro de 2019.