



UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA

Autorizada pelo Decreto Federal nº 77.496 de 27/04/76
Recredenciamento pelo Decreto nº 17.228 de 25/11/2016



PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
COORDENAÇÃO DE INICIAÇÃO CIENTÍFICA

XXV SEMINÁRIO DE INICIAÇÃO CIENTÍFICA DA UEFS SEMANA NACIONAL DE CIÊNCIA E TECNOLOGIA - 2021

Implementação da Arquitetura SOFT-IoT em Ambiente de Emulação Fogbed

**Uellington da Conceição Damasceno¹; Antonio Augusto Teixeira Ribeiro
Coutinho**²;

1. Bolsista PROBIC, Graduando em Engenharia de Computação, Universidade Estadual de Feira de Santana, e-mail: udamasceno@ecomp.uefs.br
2. Orientador, Departamento de Tecnologia (DTEC), Universidade Estadual de Feira de Santana, e-mail: acoutinho@uefs.br

PALAVRAS-CHAVE: Internet das Coisas; Computação em Névoa; Emulação.

INTRODUÇÃO

Os desafios relacionados à Internet das Coisas (*Internet of Things*, IoT) (GUBBI et al., 2013) vêm motivando propostas de arquiteturas amplamente distribuídas tanto à nível de infraestrutura quanto de serviço. Diferentes iniciativas apresentam formas de coletar, processar, distribuir e armazenar dados fora dos centros de nuvem para reduzir a latência e diminuir a quantidade de dados transferidos através do núcleo da rede. Atualmente, a convergência das tecnologias de nuvem com a computação em névoa (*fog computing*) (BONOMI et al., 2014) estão no centro dos estudos de soluções IoT inovadoras.

Neste contexto, o conceito de névoa das coisas (*Fog of Things*, FoT) apresentado por Prazeres e Serrano (2016) propõe o emprego dos dispositivos presentes na borda de rede como servidores locais, além do uso de *gateways* para controle de atividades e entrega de serviços aos clientes do sistema IoT. Como implementação do conceito de FoT, a plataforma SOFT-IoT (*Self-Organizing FoT for the IoT*) foi desenvolvida por Prazeres e Serrano (2016) através de uma infraestrutura *Enterprise Service Bus* (ESB) baseada na especificação modular OSGi¹, compatível com o paradigma orientado à serviços.

Entretanto, os custos envolvidos com infraestruturas de testes (*testbeds*) complexas para prototipagem de componentes da IoT (ABREU et al., 2020) tem fomentado alternativas acessíveis para avaliar soluções em estágio inicial de desenvolvimento. O ambiente Fogbed (COUTINHO et al., 2018) é um emulador de névoa implementado para possibilitar testes de componentes e aplicações de forma escalável.

Ambientes de prototipagem são amplamente usados na modelagem de sistemas de nuvem, porém ainda estão em um estágio inicial em sistemas de névoa (ABREU et al., 2020). Nosso objetivo neste trabalho é validar a prototipagem rápida e o teste escalável de componentes e serviços IoT reais com suporte à névoa, descrevendo as tecnologias e implementando suas funcionalidades em ambiente virtualizado. Para isso, foi desenvolvida uma solução de balanceamento de carga entre *gateways* completamente distribuída e segura usando a plataforma SOFT-IoT, com uso do ambiente Fogbed para emulação da infraestrutura de rede e dispositivos necessários ao estudo de caso.

¹ <https://www.osgi.org/>

MATERIAL E MÉTODOS OU METODOLOGIA

A primeira etapa do trabalho foi um estudo sobre a arquitetura SOFT-IoT e a implementação de seus componentes principais: *FoT-Devices* (sensores e atuadores) e os *FoT-Gateways*. Os *FoT-Devices* usam *Message Queuing Telemetry Transport* (MQTT)² para implementar *The Accessible Thing Universe Protocol* (TATU) como padronização na comunicação entre os *FoT-Devices* e os *FoT-Gateways*. Os *FoT-Gateways* podem ser instalados em dispositivos de baixo custo, com processamento e memória limitados. O *middleware* implementado é dividido em duas camadas (PRAZERES et al., 2017): uma camada orientada a mensagens e outra camada orientada a serviços.

A seguir foi iniciada a implementação do estudo de caso, um algoritmo para balanceamento de *gateways* usando tecnologia de livro razão distribuída (*Distributed Ledger Technology*, DLT) (GREVE et al., 2018). Na solução, foi usada uma DLT baseada na IOTA³, conhecida como Tangle. A comunicação no plano de controle entre *FoT-Gateways* e *FoT-Devices* foi realizada através da rede Tangle usando um agrupamento lógico (*overlay*) hierárquico em diferentes camadas. Para melhorar o desempenho da comunicação, foi modificada a implementação padrão do nó IOTA (*IOTA Reference Implementation*, IRI) incorporado um filtro de mensagens na rede Tangle.

A próxima etapa foi o estudo da *Application Programming Interface* (API) do ambiente Fogbed. O ambiente de emulação permite instanciar e testar componentes de névoa distribuídos em um cluster de máquinas *host* em uma rede local. Porém, para possibilitar a execução virtual e escalável da arquitetura SOFT-IoT em diferentes máquinas físicas é necessário realizar a containerização de *FoT-Gateways* e *FoT-Devices*. Durante o processo de containerização foi necessária a configuração dos módulos do SOFT-IoT para permitir suporte a seus protocolos e serviços. Além disso, foi realizada a configuração das imagens para definir os recursos de memória e processamento disponíveis através da interface de containers Docker⁴.

Por fim, através da API de topologia do Fogbed, foi definida uma topologia para avaliar o desempenho dos nós virtuais de névoa propostos no estudo de caso. Para isso, foram implementados *scripts* de configuração dinâmicos para rede emulada entre dispositivos virtuais simulados e instâncias de névoa virtuais. Na execução do ambiente, foram usados computadores do Laboratório de Redes e Sistemas Distribuídos (LARSID) da Universidade Estadual de Feira de Santana (UEFS). Devido ao isolamento do laboratório pela pandemia de COVID-19 no Estado da Bahia, também foram usados recursos do bolsista e do orientador na metade final do trabalho.

RESULTADOS E/OU DISCUSSÃO

Os principais resultados desse trabalho foram: (i) a implementação da arquitetura SOFT-IoT-DLT e dispositivos virtuais capazes de processar as primitivas necessárias para participar do algoritmo de balanceamento; (ii) a virtualização dos componentes do

² <https://mqtt.org/>

³ <https://iota.org/>

⁴ <https://www.docker.com/>

SOFT-IoT através de containers Docker para emular o funcionamento dos *FoT-Gateways* e dos *FoT-Devices* no ambiente de emulação Fogbed.

A arquitetura SOFT-IoT-DLT estende a arquitetura SOFT-IoT incorporando serviços voltados ao balanceamento de carga de *gateways*. A Figura 1 mostra a solução composta pelos módulos *soft-iot-dlt-auth*, *soft-iot-dlt-load-monitor*, *soft-iot-dlt-client-tangle*, *soft-iot-dlt-load-balancer* e *soft-iot-dlt-id-manager*, específicos do estudo de caso, mas que possuem uma implementação baseada no módulo *soft-iot-mapping-devices* encontrado na arquitetura SOFT-IoT.

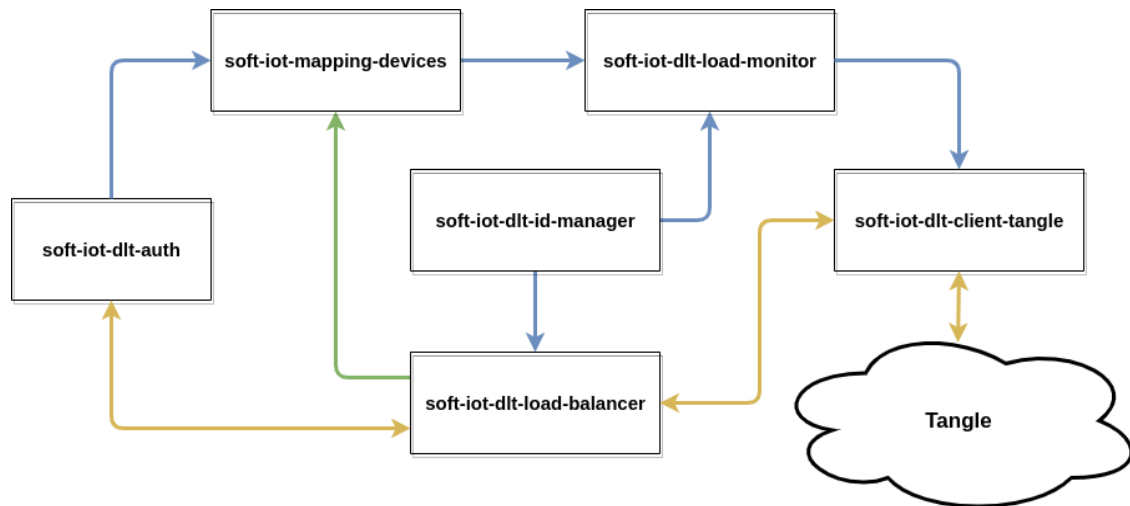


Figura 1. Arquitetura SOFT-IoT-DLT implementada no estudo de caso.

A direção das setas da Figura 1 tem como objetivo demonstrar o fluxo de interação entre os serviços (módulos) da arquitetura. As setas azuis indicam que o serviço efetua uma operação de leitura, as setas verdes representam uma operação de escrita, e as amarelas representam as duas operações. Os nomes dos módulos foram escolhidos com o intuito de facilitar o entendimento de suas funcionalidades.

O módulo *soft-iot-dlt-load-monitor* foi projetado para executar o monitoramento da carga do *gateway*. Entretanto, atualmente esse módulo verifica apenas o monitoramento da quantidade de dispositivos conectados ao *FoT-Gateway* para decidir sobre o balanceamento. Isso ocorre pela dificuldade de se obter os dados sobre o desempenho de um processo executado dentro de um container Docker.

O *soft-iot-dlt-client-tangle* é responsável pela comunicação entre os módulos da arquitetura SOFT-IoT-DLT, realizada através da rede Tangle. É através deste módulo que os pedidos de leitura e escrita na rede Tangle são solicitados pelos outros módulos.

É no módulo *soft-iot-dlt-load-balancer* que o algoritmo de balanceamento de carga acontece. Ao ser inicializado, o componente *load-balancer* acessa o serviço *soft-iot-dlt-id-manager* para saber o grupo hierárquico do *FoT-Gateway* no qual está sendo executado. Logo após, o *load-balancer* usa o serviço de leitura da rede Tangle para se inscrever no tópico de transações do grupo do *FoT-Gateway* no qual está sendo executado. A partir desse momento, o *load-balancer* inicia o algoritmo de balanceamento.

O *soft-iot-dlt-id-manager* é o módulo responsável por gerenciar as credenciais de outro *FoT-Gateway* que está inserido em seu grupo. Por meio do *soft-iot-dlt-id-manager* outros módulos de monitoramento e balanceamento conseguem direcionar as mensagens

para o grupo correto. Ele possui duas interfaces de serviços, uma para gerenciar o grupo do *FoT-Gateway* e outra para as credenciais de identificação do *FoT-Gateway*.

O módulo *soft-iot-mapping-devices* da arquitetura SOFT-IoT-DLT é uma bifurcação do módulo original da arquitetura SOFT-IoT. Essa bifurcação foi necessária pois à versão original somente é permitido operações de leitura da lista de dispositivos conectados. Entretanto, a versão utilizada no SOFT-IoT-DLT possui dois novos serviços que permite à leitura, escrita, remoção e atualização dos dispositivos conectados.

CONSIDERAÇÕES FINAIS

O trabalho conduzido faz parte de um projeto maior, que visa avaliar a escalabilidade na emulação de ambientes IoT. Durante a implementação do estudo de caso apresentado, foi necessário configurar em ambiente Docker imagens de nós de névoa compatíveis com a arquitetura SOFT-IoT. Esse procedimento permitiu executar os elementos da arquitetura SOFT-IoT através do emulador Fogbed. Devido às restrições impostas pela pandemia de COVID-19, ainda não foram realizados experimentos para avaliar a escalabilidade do protótipo usando um ambiente virtual em diferentes máquinas *host*. Entretanto, os passos para a configuração dos experimentos foram realizados de forma remota, faltando apenas sua execução para validar a prototipagem da arquitetura SOFT-IoT e SOFT-IoT-DLT em ambiente de emulação Fogbed.

REFERÊNCIAS

- GUBBI, J. et al. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions, *Future generation computer systems* 29.7: 1645-1660.
- BONOMI, F. et al. (2014). Fog computing: A platform for internet of things and analytics. In: *Big data and internet of things: A roadmap for smart environments*. Springer, Cham, p. 169-186.
- PRAZERES, C., SERRANO, M. (2016). SOFT-IoT: Self-organizing fog of things. In: *2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*. IEEE, p. 803-808.
- PRAZERES, C. et al. (2017). Design and implementation of a message-service oriented middleware for fog of things platforms. In *Proceedings of the Symposium on Applied Computing, SAC '17*, pp. 1814–1819.
- COUTINHO, A. et al. (2018). Fogbed: A rapid-prototyping emulation environment for fog computing, in *Communications Workshops (ICC Workshops), 2018 IEEE International Conference*, pp. 1-7.
- GREVE, F. G. et al. (2018). Blockchain e a Revolução do Consenso sob Demanda. *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)-Minicursos*.
- ABREU, D. P. et al. (2020). A comparative analysis of simulators for the Cloud to Fog continuum. *Simulation Modelling Practice and Theory*, vol.101, pp.102029.