

ARQUITETURA PARA CONSUMO E ANÁLISE DE DADOS EM TEMPO REAL APLICADA A REDES DE SENSORES IOT

Alisson Solitto da Silva¹, Fábio Dacêncio Pereira¹

¹Fundação de Ensino Eurípides Soares da Rocha (UNIVEM)
CEP – 17.525-901 – Marília – SP – Brasil

alisson.solitto@gmail.com, prof.fabiopereira@gmail.com

Abstract. *With the steady advancement of IoT (Internet of Things) sensor networks, many surveys face problems finding a dataset that exposes updated real-time data from an IoT environment. The analysis of data produced by IoT sensor networks are relevant research topics, but many still use a lagged dataset provided by Berkeley University. This work proposes the creation of a modular and updated architecture for the collection, enrichment, storage and availability of real-time data generated by the IoT sensor networks, in order to be used by algorithms for analysis and vulnerability detection in IoT networks.*

Resumo. *Com o avanço constante das redes de sensores IoT (Internet of Things) muitas pesquisas enfrentam problemas ao encontrar um dataset que exponha dados atualizados e em tempo real de um ambiente IoT. A análise de dados produzidos por redes de sensores IoT são temas de pesquisas relevantes, mas muitas ainda utilizam um dataset defasado disponibilizado pela Universidade Berkeley. Esse trabalho propõe a criação de uma arquitetura modular e atualizada para a coleta, enriquecimento, armazenamento e disponibilidade dos dados em tempo real gerados pelas redes de sensores IoT, com o objetivo de ser utilizado por algoritmos de análise e detecção de vulnerabilidades em redes IoT.*

1. Introdução

Com o constante avanço da tecnologia a quantidade de objetos conectados à Internet está em uma constante crescente. Um relatório da Gartner Inc. “prevê que 6,4 milhões de objetos estarão conectados e em uso em todo o mundo no ano de 2016, um aumento de trinta por cento em relação ao ano de 2015 e chegará a 20,8 bilhões até 2020” [Van Der Meulen 2015]. Esse grande número de objetos conectados à Internet geram e disseminam uma miríade de informações que podem interagir com outros sistemas e proporcionar o poder de tomada de decisão.

A busca por melhorias em arquiteturas de redes de sensores IoT, assim como, projetos de ferramentas de análise de dados produzidos por estas redes são temas de pesquisas relevantes atualmente. Em uma revisão sistemática do tema, observou-se que o *dataset* disponibilizado pela Universidade Berkeley [Bodik et al. 2004], é referenciado por inúmeros trabalhos, tais como: [Rassam et al. 2015], [Waskita et al. 2016] e [Junior et al. 2017], com foco na análise e detecção de vulnerabilidades e anomalias em redes de sensores IoT.

Esse *dataset* consiste em uma base dados contendo informações de uma rede de 54 sensores e registra dados de quatro tipos de grandezas: iluminação, temperatura, umidade e corrente. Contudo esses dados foram coletados no laboratório Intel Berkeley Research entre 28 de fevereiro e 5 de abril de 2004.

Neste contexto, surge a problemática que envolve esta pesquisa. Muitos trabalhos ainda utilizam o *dataset* mencionado, consideramos que para o cenário atual de redes de sensores IoT esse *dataset* já se tornou obsoleto. Não há uma diversidade de tipos de sensores para a coleta de dados no ambiente no *dataset* mencionado, as tecnologias que constituíam a arquitetura do experimento estão defasadas há anos e principalmente não permitem a homologação de algoritmos de detecção de vulnerabilidade e anomalias, pois não oferece mecanismos de análise em tempo real da rede de sensores sem fio.

O objetivo do presente trabalho é criar um ambiente que disponibilize dados públicos e atualizados em tempo real do InnovaSpace Coworking[Univem 2017] localizado no campus do UNIVEM. Nesse ambiente foi feita a instalação inicialmente de dez sensores para a coleta de dados em tempo real seguindo a arquitetura proposta nesse projeto.

O objetivo final é criar um sistema independente, onde outros pesquisadores possam usar além dos dados coletados a arquitetura e as soluções de análise para projetar suas próprias redes de sensores IoT. Desta forma possibilitando novas pesquisas para identificação de comportamentos anômalos na rede de sensores e disponibilizando os dados do ambiente em tempo real de forma pública.

2. Arquitetura Proposta

Para a criação de uma arquitetura que disponibilize dados em tempo real de uma rede de sensores IoT de forma pública e contextualizada é necessário a análise de vários fatores. As principais preocupações são a segurança e a capacidade computacional dos sensores, os dispositivos responsáveis por observar e coletar os dados do ambiente não possuem muitos recursos para disponibilizar esse dado em alto nível para a tomada de decisão de maneira segura.

A arquitetura visa modelar uma rede de sensores IoT para que seja possível empregar as principais funcionalidades que circunda a Internet das Coisas, tais como: interoperabilidade, comunicação M2M (*Machine-to-Machine*), análise de grande volume de dados, análise de anomalias na rede e predição de dados futuros.

Desta forma os pesquisadores podem criar sua própria rede de sensores IoT utilizando a estrutura central da arquitetura para o envio dos dados, sendo possível o armazenamento e a disponibilização dos dados de forma pública e em tempo real em prol de, diversificar os dados e ambientes gerando novas pesquisas com esse novo *dataset*.

3. Modelo da Arquitetura

A arquitetura (Figura 1) para o desenvolvimento do projeto é dividida em seis camadas que estão descritas ao longo do artigo, as camadas das extremidades (INTERNET OF THINGS (A), HIGH LEVEL APPLICATION (E) e USER INTERFACE) são as camadas desacopladas, tornando possível o desenvolvimento de ambientes e aplicações por outros pesquisadores sem a preocupação com a infraestrutura e o armazenamento dos dados.

As camadas responsáveis por estas funcionalidades estão acessíveis para a utilização de forma pública.

As camadas centrais (BROKER MQTT (B), MIDDLEWARE (C) e PERSISTENCE (D)) podem ser escaláveis, ou seja, expandidas de forma horizontal com a adição de mais nós aos sistema garantindo o processamento de acordo com a demanda.

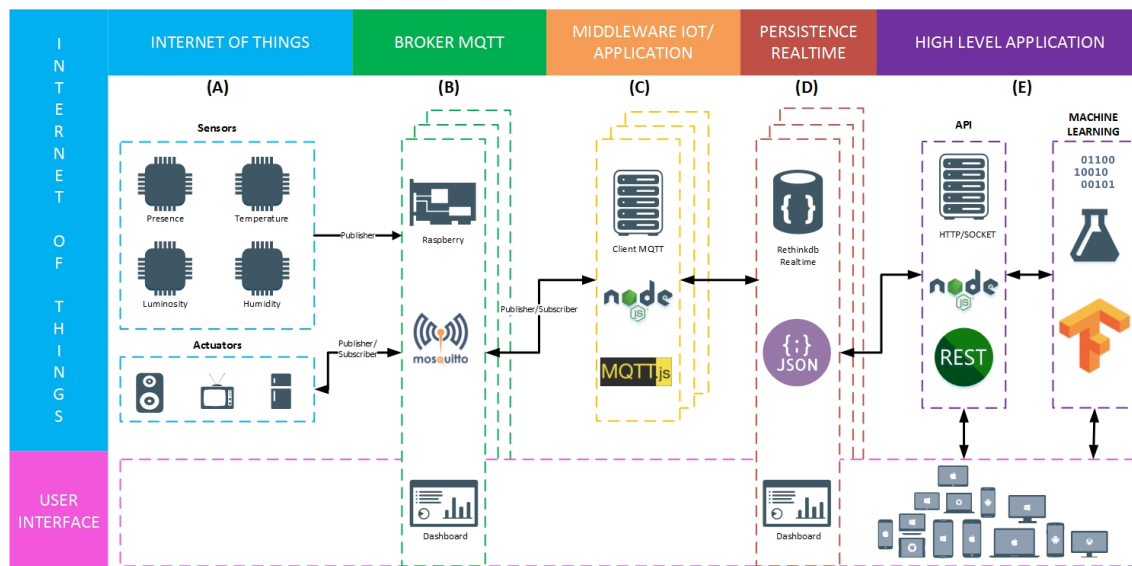


Figura 1. Modelo de arquitetura proposta para a criação da rede de sensores IoT

3.1. Internet of Things

Nesta camada (Item A, Figura 1) encontra-se os dispositivos IoT instalados no ambiente, esses dispositivos podem ser divididos em sensores e atuadores. Os sensores são um dos pilares da IoT, estes são responsáveis por coletar os dados do ambiente onde estão inseridos, já os dispositivos atuadores recebem informações ou comandos para atuar no ambiente.

Um dos grandes objetivos da IoT é que os dispositivos se comuniquem sem a intervenção humana e instruções explícitas. Partindo deste princípio a camada de Internet of Things presente na arquitetura pretende utilizar o protocolo de comunicação MQTT (*Message Queue Telemetry Transport*), tornando possível a comunicação M2M entre sensores e atuadores no ambiente.

3.2. Broker MQTT

O broker (Item B, Figura 1) faz parte do uso do protocolo MQTT, este é responsável por gerir a troca de mensagens entres os dispositivos da rede (*publish/subscriber*). Quando um dispositivo da rede de sensores IoT deseja publicar uma informação, este envia uma mensagem (*publish*) para o broker, os atuadores que desejam receber (*subscriber*) as mensagens que são enviadas pelos sensores “assinam” as suas publicações de interesse.

Um broker pode ser um dispositivo embarcado (Raspeberry Pi) ou até mesmo um computador ou servidor. A troca de mensagens feita pelos dispositivos operando o protocolo MQTT utilizam o conceito de tópicos, semelhante a uma URI (*Uniform Resource Identifier*).

Para o desenvolvimento dessa camada é utilizado o Mosquitto [Light 2013], um broker MQTT de código aberto desenvolvido em C e mantido pela fundação Eclipse.

3.3. Middleware IoT/Application

O *middleware* (Item C, Figura 1) é um cliente MQTT, responsável por receber todas as mensagens que são publicadas pelos sensores no broker MQTT, essa camada é responsável pelo mapeamento e persistência dos dados das mensagens publicadas na rede de sensores IoT. Além de fazer a persistência, a camada também trabalha em mais alto nível se comparado com os sensores, publicando dados enriquecidos para os atuadores na rede.

O desenvolvimento do cliente broker em mais alto nível é feito utilizando a biblioteca MQTT.js [Rudd 2012]. Essa biblioteca utiliza a linguagem JavaScript e possui o código fonte aberto.

3.4. Persistence Realtime

Para posteriores análises e consultas todos os dados publicados na rede de sensores IoT são persistidos em um *dataset* realtime (Item D, Figura 1). Com a persistência dos dados produzidos no ambiente é possível efetuar análises utilizando diferentes métodos.

O banco de dados RethinkDB [Walsh et al. 2009] é um banco de dados NoSQL de código aberto. A escolha da tecnologia RethinkDB para o armazenamento está principalmente relacionada por possibilitar a consulta de resultados atualizados em tempo real.

Os dados persistidos nessa camada utilizam o formato JSON, esses dados já estão enriquecidos com outros metadados, sendo possível a geração da informação no contexto.

3.5. High Level Application

Na camada de mais alto nível da arquitetura (Item E, Figura 1) estão disponíveis os serviços para o consumo do *dataset*.

A API (*Application Programming Interface*) segue o estilo arquitetural REST (*Representational State Transfer*) [Fielding and Taylor 2000] que define um conjunto de princípios a serem aplicados na construção de sistemas com uma arquitetura orientada a recursos (URI), esse recurso pode ser tudo aquilo que deve ser acessado pelo cliente e transferido entre o mesmo e um servidor.

Além da API na camada *High Level Application* são disponibilizados algoritmos para a detecção de anomalias e análise preditiva com base no *dataset* dos dados coletados na rede de sensores IoT, utilizando técnicas de *machine learning* com o auxílio da ferramenta TensorFlow [Abadi et al. 2016].

3.6. User Interface

As aplicações para o usuário estão definidas nessa camada, na camada BROKER MQTT e PERSISTENCE REALTIME são disponibilizadas duas aplicações para a análise e monitoramento dos dados, isoladamente em cada camada.

Na camada USER INTERFACE é possível a criação de diferentes aplicações, nessa arquitetura os sistemas são desenvolvidos utilizando JavaScript, uma linguagem amplamente aceita sendo possível o desenvolvimento de software para vários sistemas operacionais.

4. Considerações Finais

Atualmente a dificuldade de encontrar um *dataset* público e atualizado de uma rede de sensores IoT tem sido um grande problema para a iniciação de novas pesquisas, evolução e melhoria desses ambientes.

Com a criação do projeto seguindo a arquitetura proposta torna-se possível a criação de um *dataset* atualizado e disponível publicamente em tempo real para o uso de algoritmos de detecção de anomalias. Além de uma diversidade de sensores, informações e ambientes, auxiliando e incentivando as pesquisas na área de Internet das Coisas. Atualmente o projeto está limitado a uma rede de dez sensores IoT, análises com relação ao desempenho, volume dos dados e segurança da arquitetura ainda estão em fase de estudos.

Referências

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283.
- Bodik, P., Hong, W., Guestrin, C., Madden, S., Paskin, M., and Thibaux, R. (2004). Intel lab data. *Online dataset*. Disponível em: <http://db.csail.mit.edu/labdata/labdata.html> (Acessado em 15.03.2018).
- Fielding, R. T. and Taylor, R. N. (2000). *Architectural styles and the design of network-based software architectures*. University of California, Irvine Doctoral dissertation.
- Junior, N. F., Silva, A., Guelfi, A., and Kofuji, S. T. (2017). Iot6sec: reliability model for internet of things security focused on anomalous measurements identification with energy analysis. *Wireless Networks*, pages 1–24.
- Light, R. (2013). Mosquitto-an open source mqtt v3. 1 broker. Disponível em: <http://mosquitto.org> (Acessado em 15.03.2018).
- Rassam, M. A., Zainal, A., and Maarof, M. A. (2015). Principal component analysis-based data reduction model for wireless sensor networks. *International Journal of Ad Hoc and Ubiquitous Computing*, 18(1-2):85–101.
- Rudd, A. (2012). Mqtt.js is a client library for the mqtt protocol. Disponível em: <https://github.com/mqttjs/MQTT.js> (Acessado em 16.03.2018).
- Univem, Citec, C. (2017). Innovaspace coworking. Disponível em: <http://www.innovaspace.com.br> (Acessado em 18.03.2018).
- Van Der Meulen, R. (2015). Gartner says 6.4 billion connected ‘things’ will be in use in 2016, up 30 percent from 2015. *Comunicado de Imprensa*. Disponível em: <http://www.gartner.com/newsroom/id/3165317> (Acessado em 17.03.2018).
- Walsh, L., Akhmechet, V., and Glukhovsky, M. (2009). Rethinkdb-rethinking database storage.
- Waskita, A., Suhartanto, H., and Handoko, L. T. (2016). A performance study of anomaly detection using entropy method. In *Computer, Control, Informatics and its Applications (IC3INA), 2016 International Conference on*, pages 137–140. IEEE.