

An MQTT-Based Context-Aware Autonomous System in oneM2M Architecture

Geonwoo Kim, Seongju Kang, Jiwoo Park, and Kwangsue Chung

Abstract. *Having a seamless connection between independently developed platforms has recently become an important issue. In addition, research is underway on autoconfiguration to minimize human interference while configuring devices for users to utilize Internet of Things (IoT) applications. OneM2M is a standardization group established to develop a standard Internet service platform. In the oneM2M environment, data exchange between each component is performed through HTTP, constrained application protocol (CoAP), and message queuing telemetry transport (MQTT). MQTT enables efficient data transmission in low-power and unreliable networks. However, MQTT is not suitable for service discovery for autoconfiguration given that it does not support multicasting and resource directories for managing IoT resources. In this paper, we present an architecture to provide an autoconfiguration mechanism by using the MQTT publish/subscribe messaging pattern. To realize fully autonomous systems, the proposed system autonomously interacts with devices by recognizing the context using semantic Web technology. We verify the feasibility of the proposed system through its implementation in the real environment. It is confirmed that the proposed approach is able to automatically configure MQTT-based devices and provide context-aware services using a standard interface.*

1. Objetivo

O objetivo dos autores deste artigo é apresentar um sistema autônomo, ciente de contexto, que permite a descoberta de serviço e operação dos dispositivos de IoT (Internet das Coisas) de forma automática, usando o protocolo de mensagens de publicação/assinatura MQTT.

2. Justificativa

A IoT é uma coleção enorme e heterogênea de protocolos, serviços e soluções e um dos pontos-chave da sua implantação é a interoperabilidade entre dispositivos e aplicativos em várias arquiteturas, plataformas e tecnologias de rede [1]. No entanto, devido ao rápido crescimento do mercado de IoT, o número de dispositivos aumentou dramaticamente, dificultando assim a conexão de plataformas desenvolvidas de forma independente e a configuração de dispositivos e serviços [2].

Para mitigar o ecossistema fragmentado de IoT, diversos setores e organizações internacionais de padrões corroboraram para publicar especificações padrão para o sistema IoT. Dessa forma, um projeto de parceria internacional **oneM2M** foi criado para produzir uma especificação de camada de serviço M2M mundial, para soluções máquina-a-máquina [3].

A troca de dados entre cada componente no ambiente oneM2M é realizada usando o protocolo de mensagens MQTT. No entanto, como o MQTT opera com base no TCP, ele não suporta *multicast* e diretórios de recursos para gerenciar os recursos de IoT. Portanto, MQTT, de forma isolada, não é adequado para descobrir serviços para configurar automaticamente os dispositivos IoT.

3. Proposta

Para resolver esse problema, os autores deste artigo apresentaram uma arquitetura para fornecer um mecanismo de autoconfiguração de dispositivos IoT, usando o sistema de mensagens de publicação/assinatura MQTT.

Para criar sistemas totalmente autônomos, a proposta interage autonomamente com dispositivos, reconhecendo o contexto usando tecnologia da Web semântica.

4. Trabalhos relacionados

Sem padrões, os sistemas e serviços de IoT seriam desenvolvidos de forma independente para diferentes domínios verticais, causando problemas de alta fragmentação e aumentando custos de desenvolvimento e manutenção. Dessa forma, diversos setores e organizações trabalham em conjunto para publicar especificações para o sistema de IoT, incluindo protocolos de comunicação leves, mecanismos de descoberta de serviço e tecnologias da Web semântica.

Os autores apresentam várias pesquisas relevantes para este foco, conforme:

A. Padrão oneM2M

A iniciativa global oneM2M tem o compromisso de padronizar uma plataforma de camada de serviço comum para M2M/IoT. O oneM2M pode ser usado para conectar dispositivos no campo com servidores de aplicativos M2M em todo o mundo. Além disso, facilita a troca de dados entre diferentes dispositivos e servidores de aplicativos, e se torna ideal para casos de uso que exigem muita interconexão [3].

B. Protocolo MQTT

MQTT é um protocolo de comunicação projetado para permitir a interação de vários recursos, dentro de uma arquitetura publicar/assinar. No conceito básico de entrega de mensagens por meio do protocolo MQTT, um cliente representa um programa ou dispositivo que troca mensagens com outro cliente por meio de um servidor, chamado de intermediário de mensagem. As interações são sempre moderadas pelo servidor, e assim os clientes não precisam se conhecer. Os clientes se comunicam solicitando o servidor para assinar ou publicar mensagens sobre um determinado tópico [4].

C. Protocolo de descoberta de serviço

Acesso e descoberta de serviço são duas funcionalidades básicas essenciais para aplicações IoT. A função do protocolo de descoberta de serviço é fornecer um método de acesso a um dispositivo descobrindo um serviço na rede local.

O protocolo de descoberta de serviço é dividido na abordagem centralizada e a abordagem distribuída. A abordagem centralizada gerencia as informações de serviço na rede por meio de um ou mais diretórios de recursos. A abordagem distribuída descobre os serviços por meio da interação entre dispositivos sem um diretório de recurso.

As principais funções e procedimentos do protocolo de descoberta de serviço são: publicação, registro, descoberta e resolução. A publicação informa outro dispositivo com informações como o tipo de serviço, método de acesso, e suas características. O registro armazena as informações de serviço de um dispositivo em um elemento específico na rede, como um diretório de recursos para que a descoberta de serviço possa ser executada [5].

D. Web semântica

Web semântica é um movimento colaborativo para organizar a informação de maneira legível para computadores e máquinas através de padrões de formatação de

dados. No ambiente da Web semântica, cada recurso é identificado por meio de um International Resource Identifier (IRI) e é definido por meio de um modelo de dados chamado Resource Description Framework (RDF) [6].

As tecnologias semânticas são capazes de desempenhar funções críticas na gestão de dados e conhecimento em plataformas IoT. Para criar sistemas totalmente autônomos para fornecer serviços sensíveis ao contexto no ambiente IoT, é necessário suporte à interoperabilidade no nível semântico dos recursos para além da interoperabilidade a nível de comunicação [7].

Segundo os autores, até onde eles sabem, nenhum trabalho ainda foi feito no sentido de possibilitar a configuração automática, ciente de contexto, que permita a descoberta de serviço e operação de dispositivos de IoT, usando MQTT.

5. Materiais e métodos usados pelos autores para elaborar a proposta

No sistema de autoconfiguração proposto pelos autores, todos os procedimentos devem ser processados dinamicamente sem intervenção humana quando um dispositivo é conectado à rede. Nesse papel, o sistema de autoconfiguração emprega o protocolo de descoberta de serviço DNS-SD/mDNS para conexão do cliente com o *broker* MQTT de forma autônoma.

A. Visão geral do sistema

A estrutura geral do sistema proposto é mostrada na Fig. 1. O *gateway* e o dispositivo existem na mesma rede e operam como um nó mDNS, implementando a função DNS-SD para descoberta de serviço. O *gateway* e os dispositivos têm um módulo de comunicação MQTT para registro de serviço, descoberta e comunicação de dados.

O *broker* e o diretório de recursos existem no *gateway*. O diretório de recursos gerencia o dispositivo e as informações de serviço na rede local e retorna o resultado da operação de pesquisa no formato JSON de acordo com a solicitação de descoberta. O *gateway* atua como um MN-CSE na arquitetura oneM2M, que gerencia o servidor Web e o recurso e também fornece CSFs.

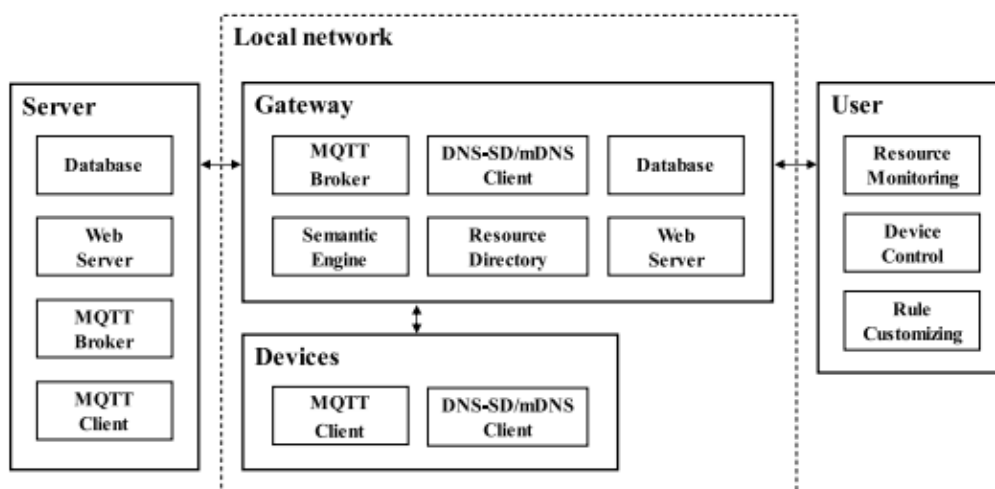


Fig. 1. Visão geral do sistema autônomo ciente de contexto baseado em MQTT

O Motor Semântico recebe as informações do dispositivo através do diretório de recursos e cria gráficos RDF. A plataforma de servidor se comunica com cada *gateway*

local e gerencia todo o *gateway*. Os usuários podem monitorar e operar dispositivos na rede local usando um sistema baseado na interface Web e receber serviços automatizados de acordo com as regras definidas pelo usuário.

B. Ambiente de Implementação

Para verificar a proposta os autores construíram um ambiente experimental. Os *gateways* e dispositivos foram implementados usando uma placa *Tinker e Raspberry Pi 3*, que são placas embutidas prontas para uso, e OCEAN, que é uma plataforma de código aberto que está em conformidade com o padrão oneM2M.

O *broker* MQTT do *gateway* usa *Mosquitto*, e o protocolo DNS-SD/mDNS para descoberta de serviço e publicação foi implementado usando Node-mDNS. O *Semantic Engine* foi implementado usando o framework *Jena*, da biblioteca de código aberto Java, que suporta o desenvolvimento de funções semânticas, como a RDF, semântica de banco de dados e SPARQL.

6. Resultados apresentados

Para analisar o desempenho foi medido o tempo do serviço de descoberta, dependendo do número de serviços na rede local e o tempo de correspondência de regra, dependendo do número de regras definidas pelo usuário. O atraso médio foi calculado com base em dez medições em cada caso.

O tempo de descoberta do serviço mede o atraso de quando o usuário e o dispositivo publicam a solicitação de descoberta de serviço usando MQTT, para quando o resultado for recebido. Resultados experimentais mostraram que uma média de 74 ms foram consumidos em um ambiente onde existem 1000 serviços na rede local, e que o atraso tem algumas diferenças dependendo do tempo de comunicação MQTT.

Os resultados da medição do tempo médio de correspondência de regra, dependem do número de regras definidas pelo usuário. Uma vez que o procedimento de correspondência de regras é realizado por meio de consulta no banco de dados semântico, levou mais tempo em comparação com o tempo de descoberta do serviço. Além disso, conforme o número de regras definidas pelo usuário aumenta, o atraso aumenta porque o número de gráficos RDF definidos para cada uma das condições IF e THEN é maior.

7. Conclusão

Neste artigo, um sistema para fornecer serviços autônomos, ciente de contexto, em um ambiente configurado com dispositivos baseados em MQTT foi proposto. O sistema proposto realiza a descoberta do serviço implementando um esquema de solicitação/resposta assíncrona usando o padrão de mensagens MQTT publicar/assinar. Além disso, para que o sistema seja totalmente autônomo, o *gateway* interage autonomamente com dispositivos, reconhecendo o contexto usando a tecnologia Web semântica.

Considerando as características do MQTT, o processo de descobrir um *gateway* para registrar serviços usa DNS-SD/mDNS. Depois que os serviços são registrados no diretório de recursos, os dispositivos se tornam recursos oneM2M e são capazes de acessar dispositivos e descobrir serviços usando MQTT.

Foi realizada uma simulação em ambiente real que analisou a viabilidade e o desempenho do sistema proposto. Pelos resultados, confirmou-se que os dispositivos e serviços são capazes de configurar automaticamente em um ambiente composto de

dispositivos baseados em MQTT e que o sistema proposto é capaz de fornecer serviços sensíveis ao contexto.

8. Visão crítica sobre o artigo

O artigo foi publicado em 30 de maio de 2019 e foi escrito por um professor e três alunos do curso de Doutorado em Eletrônica e Telecomunicações da Universidade Kwangwoon, de Seul na Coreia do Sul. Desde a sua publicação, o artigo teve 15 citações e mais de 600 visualizações.

Considero o artigo bem completo e bem escrito, com uma boa fundamentação e um rico detalhamento na simulação do experimento e análise de resultados.

Uma questão bastante enfatizada no artigo é o fato de que as tecnologias da Internet das Coisas variam conforme a aplicação, e, portanto, padronizações são necessárias. Na verdade, sem plataformas e padrões para orientar o desenvolvimento de produtos e serviços, a IoT pode se tornar uma bagunça caótica.

Expandindo a leitura para outras publicações, foi possível verificar que diversas iniciativas vem sendo criadas para estabelecer padrões tecnológicos que trazem viabilidade, interoperabilidade, segurança, integridade, disponibilidade, escalabilidade e desempenho para aplicações de IoT, porém, ainda há muito que se avançar nesse sentido.

9. Referências

- [1] J. A. Stankovic, “Research direction for the Internet of Things,” IEEE Internet Things J., vol. 1, no. 1, pp. 3–9, Feb. 2014.
- [2] S. Kim, K. Lee, and J. Jeong, “DNS naming services for service discovery and remote control for Internet of Things devices,” in Proc. Int. Conf. ICT Converg., Oct. 2017, pp. 1157–1162.
- Holton, M. and Alexander, S. (1995) “Soft Cellular Modeling: A Technique for the Simulation of Non-rigid Materials”, Computer Graphics: Developments in Virtual Environments, R. A. Earnshaw and J. A. Vince, England, Academic Press Ltd., p. 449-460.
- [3] oneM2M—Standards for M2M the Internet of Things. Accessed: Sept. 19, 2021. [Online]. Available: <http://www.onem2m.org>
- [4] A. Banks and R. Gupta, MQTT Version 3.1.1, OASIS Stand., Burlington, MA, USA, Apr. 2014.
- [5] B. C. Villaverde, R. De Paz Alberola, A. J. Jara, S. Fedor, S. K. Das, and D. Pesch, “Service discovery protocols for constrained machine-to-machine communication,” IEEE Commun. Surveys Tuts., vol. 16, no. 1, pp. 41–60, 1st Quart., 2014.
- [6] A. I. Maarala, X. Su, and J. Riekk, “Semantic reasoning for context-aware Internet of Things applications,” IEEE Internet Things J., vol. 4, no. 2, pp. 461–473, Apr. 2017.
- [7] C. E. Kaed, I. Khan, A. V. D. Berg, H. Hossayni, and C. Saint-Marcel, “SRE: Semantic rules engine for the industrial Internet of Things gateway,” IEEE Trans. Ind. Informat., vol. 14, no. 2, pp. 715–724, Feb. 2018.