KSII The 13th Asia Pacific International Conference on Information Science and Technology (APIC-IST) 2018.

An Open Source Conformance Testing Tool for Standardized IoT Platforms

JaeYoung Hwang*†, NakMyoung Sung†, SeungMyeong Jeong†, Lewis Nkenyereye*, JaeSeung Song*
Department of Computer and Information Security, Sejong University, Seoul, South Korea*,
Korea Electronics Technology Institute (KETI), Sungnam, Gyeonggi-do, South Korea†
[e-mail: forest62590@sju.ac.kr, {nmsung, sm.jeong}@keti.re.kr, {nkenyele, jssong}@sejong.ac.kr]
Corresponding author: JaeSeung Song

Abstract

Many vendors are releasing their products such as IoT platforms or devices to market with the growth of the IoT. However, if these IoT platforms or devices do not ensure the interoperability and conformance with the IoT standards specified by many IoT platforms or devices, it could lead to the fragmentation of the IoT ecosystem. Thus, many IoT vendors would face interoperability challenges. In order to overcome this issue, conformance and interoperability testing are considered as a key footstep towards the success of IoT ecosystem. In this paper, we focus on the conformance testing as a prerequisite of the interoperability testing and introduce an open source conformance testing tool to conduct the IoT standard conformance testing

Keywords*: *Conformance Testing, Eclipse Titan, Internet of Things, oneM2M Standard, TTCN-3

1. Introduction

In recent years, Internet of Things (IoT) is playing a role as a key to improve the citizens' daily life and mitigate many social challenges such as traffic jam, emission from motor vehicles and so on. In addition, it is expected that connected devices will reach 30 billion by 2020 [1].

With the growth of the connected devices, interconnection issue caused by different devices using a variety of IoT standards will lead to the fragmentation of the IoT environment [2]. In addition, International Telecommunication Union (ITU) insisted that interoperability issue is the main obstacle which hinders IoT development and adoption by the market [3]. Therefore, interoperability issue has to be overcome and furthermore, conformance testing also has to be considered as a prerequisite of interoperability testing. In this regards, Interoperability testing is to test that end-to-end functionality required by that standard among devices or platforms; whereas Conformance testing is to check whether the implementation using a particular standard is well developed based on requirements of that standard [4].

In this paper, we focus on the conformance testing as a first step to overcome the interoperability issues. More specifically, we use the oneM2M IoT standard as one of the IoT standards and introduce an open source oneM2M conformance testing framework which is called *oneM2MTester* to conduct the conformance testing. The *oneM2MTester* development phase is part of oneM2M conformance testing project and it was initialized in January 2016 with the collaboration of 10 international members. Its ultimate purpose is to provide a conformance testing framework to support the testers to check whether their platforms or devices satisfy the oneM2M standard.

The structure of this paper is as follows. Section 2 introduces the basic concepts on the conformance testing tool. Section 3 proposes the

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2017-0-00501, Development of Self-learnable common IoT SW Engine)

open source-based conformance testing architecture and a practical running example is introduced in Section 4. Finally, in Section 5, we conclude this paper and mention our future work.

2. Basic Concepts to Understand the Conformance Testing Tool

This section introduces the basic concepts to understand conformance testing tool based on oneM2M standard.

2.1 oneM2M standard

oneM2M is IoT standard to mitigate the fragmentation problems regarding the IoT and it was initialized by 7 Standard Development Organizations (SDO) with around 200 partners in 2012. In addition, it has been developing various Technical Specifications (TS) for conformance testing and interoperability testing as follows[1].

- TS-0013: Interoperability Testing
- TS-0015: Testing Framework
- TS-0017: Implementation Conformance Statements
- TS-0018: Test Suite Structure and Test Purposes
- TS-0019: Abstract Test Suite Inplementation eXtra Information for test

2.2 TTCN-3 and Eclipse Titan

Testing and Test Control Notation version 3 (TTCN-3) was developed for the purpose of testing and is being maintained by European Telecommunications Standards Institute (ETSI). Because TTCN-3 is an intermediate language which means that it is a little bit different from other languages such as JAVA, C++, C and so on. Therefore, a new system is required to compile and understand the TTCN-3 [5].

Eclipse Titan[2], which is an open source conformance testing framework is an alternative to provide compilation and execution features to TTCN-3 environment.

2.3 Abstract Test Suite (ATS)

ATS is the test case based on TTCN-3 and it is being developed and maintained by ETSI. ATS confirms whether System Under Test (SUT) is well developed to comply with the targeted IoT standards. Currently, ETSI has validated the oneM2M Release 1 ATS, and oneM2MTester is supporting it.

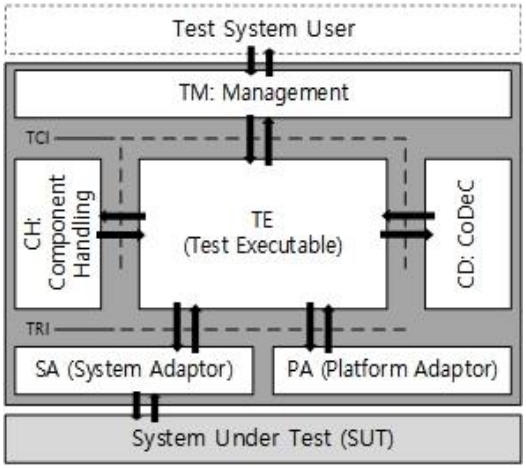

Fig. 1. Basic concept of the TTCN-3

3. Overview of Conformance Testing Tool

Figure 1. describes the basic concept of TTCN-3. Each component interacts with each other to execute test cases and for the sake of simplicity, we describe the three components including System Adapter (SA), CoDec (CD) and Platform Adapter (PA). Those componets are being modified to support the oneM2M standard.

- SA: By using the SA, testing system can communicate with SUT.
- CoDec: It is in charge of encoding and decoding the message.
- PA: It supports the timer functionality and external function concept to use the external library such as JavaScript Object Notation (JSON) or Extensible Markup Language (XML) parser.

As we mentioned earlier, TTCN-3 is a language that describes only the testing behaviors of what users want to perform. In order to perform actual communication with the SUT for the testing, conversion to a native language including UNIX-socket API, thread handling or other communication functionalities is essential. Therefore, the Eclipse Titan converts TTCN-3 as a C++ language and provide the C++ plugins called Test Ports (TPs). TPs is in charge of sending or receiving the actual data and also maintating the transport connections[2]. By using TPs, Eclipse Titan can support the Internet

[1]oneM2M: http://www.onem2m.org/technical/published-drafts
[2]Eclipse Titan: https://projects.eclipse.org/projects/tools.titan

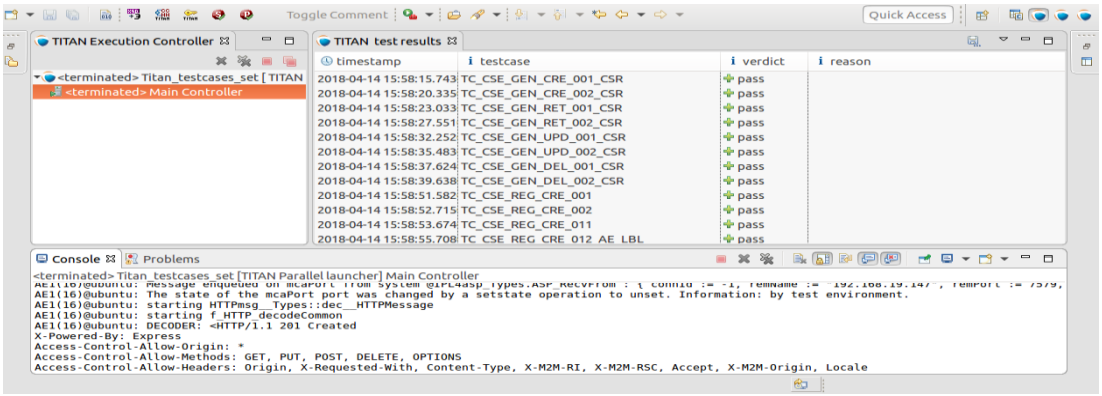

Fig. 2. Conformance testing with oneM2MTester

Protocol (IP) stack including IoT essential protocols such as HyperText Transfer Protocol (HTTP), Message Queueing Telemetry Transport (MQTT), and Constrained Application Protocol (CoAP). Currently, oneM2M has released oneM2M binding specifications[3] to bind the oneM2M primitives into application protocols and since Eclipse Titan only supports the basic IP stack, additional mechanism is required to understand the oneM2M specification. This mechanism can be realized by modifying the exsiting SA and CoDeC. Basically, TTCN-3 does not support the JSON and XML is used as oneM2M serialization type. Therefore, by using external function concept of PA, CoDec can understand the XML, JSON or the other serizliation types.

4. Running the Conformance Testing Tool

We conducted a conformance testing by using oneM2MTester within the open source oneM2M server called Mobius to show the feasibility of oneM2MTester. For example, when testers execute the "TC_CSE_GEN_CRE_001_CSR" test case which is developed for the purpose of "Check whether the SUT accepts the creation of a <container> resource". When Mobius works well according to the test behavior described in TC_CSE_GEN_CRE_001_CSR, it means that <container> resource of oneM2M specification is created successfully. Consequently, testing tool will provide a pass verdict. However, it will show fail or inconclusive verdict if the Mobius does not create the <container> resource successfully because of following the oneM2M standard incorrectly. As shown in Fig 2., oneM2MTester shows a list of verdicts as a testing result.

5. Future Direction and Conclusion

In today's world, with the growth of the necessity of IoT to improve the social challenges or improve the quality of citizens' life, connected devices are also increasing rapidly which could lead to the fragmentation of the IoT ecosystem if vendors do not ensure the interoperability and conformance of their products. Accordingly, interoperability testing and conformance testing can be the solution to overcome afore-mentioned limitations. In this paper, as one of the activities of the conformance testing, we introduce the opensource-based conformance testing framework to test the IoT platforms or devices by using Eclipse Titan. In addition, we conducted and showed conformance testing example for the practical example.

For the future work, we are planning to support the remaining tasks of the oneM2MTester such as MQTT and CoAP protocols.

References

[1] B. Ahlgren, M. Hidell, and E. C.-H. Ngai, "Internet of things for smart cities: Interoperability and open data," IEEE

[3] oneM2M TS-0008, CoAP- Protocol Binding, v3.0.0
oneM2M TS-0009, HTTP-Protocol Binding, v3.0.0
oneM2M TS-0010, MQTT-Protocol Binding,v2.7.1
http://www.onem2m.org/technical/published-drafts

Internet Computing, vol. 20, no. 6, pp. 52–56, 2016.

[2] H. Park *et al.*, "Recent advancements in the internet-of-things related standards: a onem2m perspective," ICT Express, vol. 2, no. 3, pp. 126–129, 2016

[3] E. E. Kim and S. Ziegler, "Towards an open framework of online interoperability and performance tests for the internet of things," in Global Internet of Things Summit (GIoTS), 2017. IEEE, 2017, pp. 1–6.

[4] S. Moseley, S. Randall, and A. Wiles, "Experience within etsi of the combined roles of conformance testing and interoperability testing," in Standardization and Innovation in Information Technology, 2003. The 3rd Conference on. IEEE, 2003, pp. 177–189.

[5] J. Z. Szabo and T. Csondes, "Titan, ttcn-3 test execution environment," *Infocommunications Journal*, vol. 62, no. 1, pp. 27–31, 2007.