Implementation of An Access Control Technology for Internet of Things Environments

Daewon Kim and Jeongnyeo Kim
Information Security Research Division
Electronics and Telecommunications Research Institute
Daejeon, Korea
emails: {dwkim77, jnkim}@etri.re.kr

***Abstract*—In this paper, we introduce an implementation for the access control on Internet of Things (IoT) environments. For the implementation, we designed the components and processes required for the access control technology. To show the feasibility, we implemented the technology based on an oneM2M architecture and experimented the processing steps.**

Keywords-Internet of Things; security; access control; role based access control; oneM2M.

I. INTRODUCTION

For security, Internet of Things (IoT) platforms need some access control technologies to control the information access authorities of the joined devices. Normally, under the activation of an access control technology, each device can access only its own resources and the owner or manager device can access the resources of other devices.

Among various IoT platforms, oneM2M-based platforms have been actively researched and implemented. OneM2M [1] is the global standard for machine-to-machine (M2M) communications and the IoT. Eight regional standards associations and over 200 companies are participating in the oneM2M standards. The important point is that oneM2M provides many useful specifications for constructing the oneM2M-based IoT platforms to researchers and developers.

For controlling the resource access, oneM2M defines and describes various components such as resources, attributes, and parameters [2]. However, the information is not sufficient to be practically implemented because it provides the general descriptions and procedures for the access control mechanism. In this paper, we present the implementation for an access control on an oneM2M platform. Among various access control researches, our implementation is based on Role Based Access Control (RBAC) [3][6].

For the access control, some studies have been processed. Representatively, there are Capability-based Access Control (CapBAC) [4][5] and RBAC [3][6]. They basically use token structures and certificates. An important problem of such researches is, even for sending small main contents to other IoT devices, that they may include the security information of bigger size than main contents.

The contribution of our paper is to introduce an implementation technique for the access control in IoT environments and to show the feasibility controlling the accesses only with a lightweight permission information which is known as a token identifier.

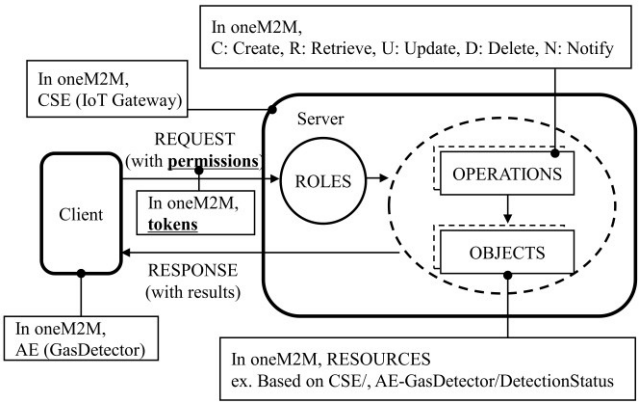

Figure 1. The simplied relationship of RBAC and oneM2M.

The rest of the paper is organized as follows. In Section 2, we introduce the background information related to the RBAC and oneM2M. In Section 3, we present the operation processes for our implementation. In Section 4, we show the experiment results for the feasibility and finally we conclude the paper in Section 5.

II. BACKGROUNDS

Figure 1 shows the simplified relationship of RBAC [3] and oneM2M. The core of RBAC is that permissions are assigned to roles rather than to devices. The role is a job function for some associated semantics regarding the authorities of devices. The semantics mean the operations for objects.

In the oneM2M architecture, common service entity (CSE) and application entity (AE) have the relationship of a server and a client. Mainly, AE sends service requests to CSE and CSE responds the processed results to AE. The resources, which are the objects in RBAC, can be addressed as the paths of the data storage including some values. For controlling the resources, the oneM2M entities can use 5 operations, such as create, retrieve, update, delete, and notify. The permissions of RBAC can be represented as the tokens in the oneM2M architecture.

 ISBN: 978-1-61208-596-8

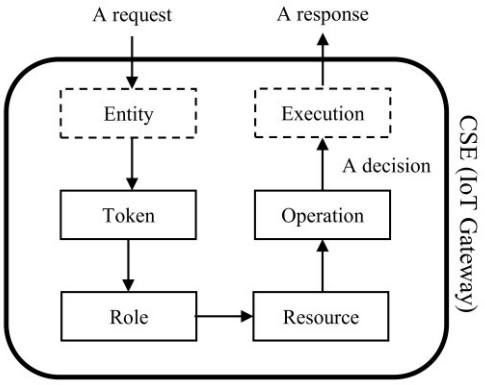

Figure 2. The processes for our access control technology.

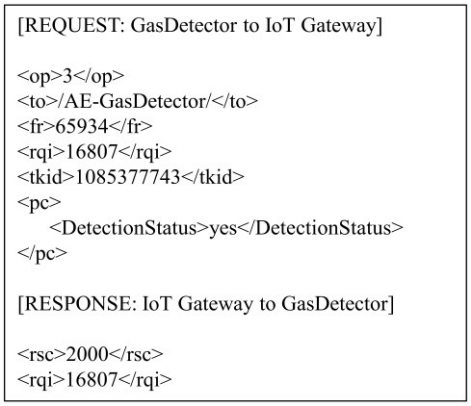

Figure 3. The request and response messages for the scenario.

III. THE OPERATION PROCESSES FOR IMPLEMENTING OUR ACCESS CONTROL TECHNOLOGY

Figure 2 shows the processes of our implementation to control the resource accesses. If one of the processes fails, the IoT Gateway sends an error response to the request originator. The solid line rectangles are the core processes for the access control. Each core process is composed of an identifier (ID) table and the job function.

A request is received to the IoT Gateway. The request normally includes a source ID, a destination ID, a target resource, an operation for the resource, and a token. First, in the Entity processing, the source ID is verified whether it is already registered into the Entity ID table. In the Token processing, the request token is verified whether it is already registered into the Token ID table. From the Token ID table, a role ID related to the token is extracted. From the Role ID table by the Role processing, a resource ID and the permitted operations are extracted. In the Resource processing, a resource related to the resource ID is extracted. Finally, if the target resource and operation in the request message are same with the resource and operation extracted from the tables in the IoT Gateway, the resource access of request is executed successfully.

IV. EXPERIMENTS

In this section, for verifying the feasibility, we show the detailed operations of our implementation about a scenario.

A. A Simple Operation Scenario based on An oneM2M Architecture

For an experiment, as an oneM2M-based practical scenario, in Figure 1, we assume that the IoT Gateway has the value 'no' in the resource CSE/AE-GasDetector/DetectionStatus. When gas leak is detected by the GasDetector, it sends an update request to the IoT Gateway for changing the resource value from 'no' to 'yes'.

B. Request and Response Messages for The Scenario

Figure 3 shows the request and response messages for the above scenario. The pseudo XML type messages are based on the oneM2M primitive parameters. Table I shows the brief descriptions for the XML tags.

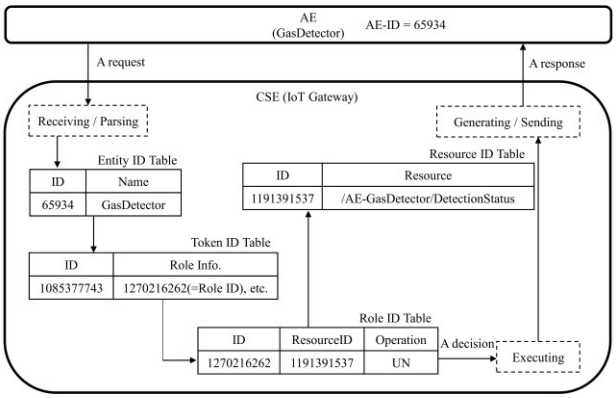

Figure 4. The detailed operations for the scenario.

TABLE I. THE BRIEF DESCRIPTIONS OF FIGURE 3

Short Name	Full Name	Description
op	OPeration	UPDATE(3)
to	TO	destination or target resource
fr	FRom	source
rqi	ReQuest Id	request identifier
tkid	ToKen ID	token identifier
pc	Primitive Content	serialized representation for accessing the target resource
rsc	Response Status Code	OK(2000)

Through the request message of Figure 3, the GasDetector requests to update the value 'no' of target resource '/AE-GasDetector/DetectionStatus' to 'yes'.

C. The Detailed Operations for The Scenario

Figure 4 shows the detailed operations about the request and response messages of Figure 3. We assume that the GasDetector ID and a token ID have been already allocated. The IoT Gateway receives a request message from the

GasDetector the source ID 65934 in the tag <fr> is verified whether it is already registered into the Entity ID table. The request token 1085377743 in the tag <tkid> is verified whether it is already registered into the Token ID table. A role ID 1270216262 related to the token 1085377743 is extracted from the Token ID table. A resource ID 1191391537 and the permitted UN operations, which are update and notify, are extracted from the Role ID table. The IoT Gateway executes the request and sends an OK response to the GasDetector because the update and notify operations are permitted about the target resource '/AE-GasDetection/DetectionStatus'.

V. CONCLUSIONS

In this paper, we introduced an RBAC-based implementation for controlling the IoT resource accesses. Through the experiments for an operation scenario, we showed the implementation feasibility and the operation clarity. As the future works, our implementation will be extended for some management issues, such as role creation, role assigning, and the device to device direct communication.

ACKNOWLEDGMENT

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2015-0-00508, Development of Operating System Security Core Technology for the Smart Lightweight IoT Devices).

REFERENCES

[1] "oneM2M Functional Architecture," oneM2M-TS-0001, vol. 2.10.0, Aug. 2016.

[2] "oneM2M Security," oneM2M-TR-0008, vol. 2.0.0, Aug. 2016.

[3] "American National Standard for Information Technology-Role Based Access Control", American National Standard Institute, Inc., ANSI INCITS 359-2004, 2004.

[4] R. Hummen, H. Shafagh, S. Raza, T. Voig, and K. Wehrle, "Delegationbased authentication and authorization for the IP-based Internet of Things," in SECON. IEEE, 2014.

[5] S. Cirani, M. Picone, P. Gonizzi, L. Veltri, and G. Ferrari, "IoT-OAS: An OAuth-based authorization service architecture for secure services in IoT scenarios," Journal of Sensors, 2015.

[6] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli, "Proposed nist standard for role-based access control," TISSEC, 2001.

 ISBN: 978-1-61208-596-8