ISSN 1598-0170 (Print)
ISSN 2287-1136 (Online)
http://www.jksii.or.kr

Access Control as a Service for Information Protection in Semantic Web based Smart Environment☆

Isma Farah Siddiqui[1] Scott Uk-Jin Lee[2*]

ABSTRACT

Pervasive computing and Internet of Things (IoT) have recently received considerable interest to deploy solutions for the future Internet. Smart environments are integrated with Semantic Web to provide context-awareness to the processed information. Self-learning techniques have been adopted within smart solutions for efficient retrieval of data but do not process data with privacy parameters for in-place authorization. To overcome this issue, we present a novel approach of deploying access control as a service mechanism within Semantic Web based smart environment by using eXtensible Access Control Markup Language (XACML). The proposed XACML as a Service (XACMLaaS) approach offers fine-grained access control for protecting information within smart environment. In this paper, we have defined mathematical rules for each components of proposed access control service layer. These rules are for implementation of access control using XACML. The proposed approach allows the adaptation of authorization of information at component level and provides scalable solution for authorization policies and rule enforcement within smart environment.

☞ keyword : Access Control, Smart Environment, Semantic Web, XACML.

1. Introduction

Smart information has become a vital resource with the deployment of heterogeneous smart sensors and devices over Internet. The emergence of wireless sensor networks within smart environment aims to provide “anytime-anywhere” computing [1]. This hype brings together pervasive computing paradigm and Ambient Intelligence (AmI) to the IoT and, support users of smart environment with seamless accessibility and provision with personalized adaptiveness to personal information. Traditionally, smart environment utilizes back-end network infrastructure and state-of-art interoperable data processing techniques to incorporate AmI [2]. However, the heterogeneity in infrastructure demands for consistent ways to uniformly process information. The semantically correct and sound information is essential for intelligent decision-making.

Semantic Web technologies contextually process information with consistency [3]. The raw data acquired from a smart device is transformed into Resource Description Framework (RDF), the standard data model for Semantic Web. The interoperability issues are managed with use of REST-full APIs for HTTP [4]. The contextual meaning comes by adding ontology to RDF data using Web Ontology Languages (OWL). The use of defined ontologies over information gives ease in data accessibility which further inevitably needs privacy protection and security. For candid accessibility, information needs to be protected by access control. XACML is an XML based access control language which is standardized by OASIS [5]. Decisions to access information like permit or deny are made on bases of set of XACML policies [6].

In order to provide an effective protection to information, we present a novel approach to incorporate access control policies with embedded context-awareness. In this research, we present a prototype layered architecture for Semantic Web based Smart Environment (SWSE). We focus on providing authorized access by in-place component service layer. The main contributions of this paper are highlighted as follows:

[1,2] Department of computer science and engineering, Hanyang University ERICA Campus, Ansan-si, Gyeonggi-Do, 426-791, Republic of Korea
* Corresponding author (scottlee@hanyang.ac.kr)
[Received 13 May 2016, Reviewed 26 May 2016, Accepted 11 July 2016]
☆ A preliminary version of this paper was presented at ICONI 2015 and was selected as an outstanding paper.
☆ This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korean government (MSIP) (No. NRF-2016R1C1B2008624)

- A novel approach to incorporate access control authorization as a service layer within a Semantic Web based smart environment for authorized information handling.
- A novel approach of XACMLaaS by utilizing XACML components as a service mechanisms.
- We provide an inter-operative information filter procedure with an independent policy enforcement mechanism through XACML components.
- Our proposed service layer offers policy enforcement mechanism between layers of user application and information storage.
- The paper also showcase a dedicated usecase environment as Semantic Web based Smart Education Environment (SWSEE) with proposed access control authorization as a service.

2. RELATED WORK

Recently, a number of researchers worked on authorized access control over cloud platforms. “Authorization as a Service” was first used by Lang [7] and was proposed for providing model driven security. Another study on XACML standard and its use as service over cloud was made by Laborde et al [8]. However, up to our knowledge no works has been proposed for ensuring access control authorization as a service for Semantic Web based smart environment.

S-CRETA [9], provides use of a real-time AmI system for a smart classroom. The assistance system exploits the use of ontologies to model the context and extract information to use it for machine learning, further targets for high-level activity recognition. However, this work is not highlighting any aspect of privacy in smart learning. Cadenhead et al. [10] gave approach of access control security over RDF graphs. Their work is targeted towards assured information sharing operated over cloud infrastructure. Alfredo et al. [11] gave idea of access control at triple level. The work focused on access control to synchronize simultaneous accessibility to shared RDF resources in a multi-agent system. However, authors did not consider its application within smart environment. Khadilkar et al. [12] studied about information sharing and privacy in private clouds. The approach incorporated XACML with Semantic Web techniques, however, not well applicable over smart environment architecture because of additional runtime processing of ontology based information over device data.

Unlike the above mentioned approaches, in this research we are giving an innovative approach to exploit the semantic information obtained from physical IoT devices and protecting it by using access control authorization as a service in a layer framework.

3. SEMANTIC WEB BASED SMART ENVIRONMENT

3.1 CONTEXT-AWARENESS USING RDF/OWL

In Semantic Web, RDF provides an open solution for describing data as information and their possible aggregation for future knowledge inference. The heterogeneous data is processed and changed into information by added context using OWL/RDF. Context ontology of this information makes it able to be utilized for querying, inferencing and reasoning. Raw data acquired from a smart device goes towards RDF transformation and ontology annotations. The transformation of data into RDF is out of scope of this paper, hence not discussed in detail.

The Semantic Sensor Network (SSN) ontology provided by W3C is best candidate for modeling sensory information in domain of smart environment [13]. This ontology serves as a prototype ontology for Semantic Web based smart environment.

3.2 MOTIVATION

Information sharing in smart environment is considered unsafe as being sensitive for malicious activities. The information over Semantic Web is available with no restricted access and authorization, which leads security and privacy issues. In a typical data-centric smart environment, some basic principles for legitimacy are confidentiality and integrity of information. Confidentiality means privacy of valuable information and, integrity ensures protection of information from tempering.

Implementing an access control authorization over smart environment, offers privacy and ensures integrity through authorized access to information. An environment with efficient

smart devices must be protected by access control as a service against malicious activities. In real world scenario, attacks can be generated from any open point within environment, devices, communication channels, ports, interface, web service etc. An overview of a few possible security threats is given here:

3.2.1 External Threats

Smart environment manages information mainly about users and devices. The smart environment needs to maintain an authorization layer to prevent external attacks. Some threat scenarios which needs to be highly considered are discussed here as follows:

Threat 1: Unauthorized Environment Management

Smart devices which operate within smart environment are manually installed and managed by actors involved in smart environment. Device management is made via associated user interface or by automated system modules. The accessibility to device management system needs to be authorized to prevent external attacks and undesired tempering of system resources and their behavior.

Threat 2: Unauthorized External Data Access

Smart environment holds confidential data of its users and smart devices. External attackers might try to penetrate within the system and avail unauthorized access to stored data, leads towards data tempering for introducing malicious behavior and possible illegal redistribution. Smart environment needs to provide authorized access mechanism as service at every level of information accessibility to prevent illegitimate usage.

Threat 3: Tempering and devastation

Smart environment can experience devastation on application level, device management level, as well on data storage level. Remote users, if gets unauthorized access can involve easily in cyber-criminal activities such as DDOS and remote monitoring. There is a definite need of providing service based privacy mechanism for intrusion prevention.

3.2.2 **Internal Threats**

Several internal threats within smart environment may initiate with unrestricted open access to system resources. In a pervasive environment, the restricted components cannot be protected physically and are seen as a big loophole. Once, a system component is compromised with illegal accessibility, it can produce malicious data for undesirable activities. Providing access control as a service module to device system level is essential to protect such internal invasions.

Threat 4: Information integrity

Malicious data sent from a single device can result in unexpected malfunctioning of overall smart environment. The integrity of information coming from smart devices needs to be well monitored. Access control service may possibly be utilized for ensuring authorized device utilization on component level.

Threat 5: Unauthorized internal information leakage

Internal users may act illegitimately once they by-pass their assigned roles. The authorized access acts as an initial barrier to overcome various internal threats.

4. PROPOSED METHODOLOGY

4.1 XACMLaaS: ACCESS CONTROL AUTHORIZATION AS A SERVICE

For the authorization dissemination throughout smart environment architecture, we have used XACML as basic policy defining language. We have proposed approach of policy specification and execution which includes description of resource and its attributes, and retrieval of description using SPARQL querying approach with enforced policy decisions. The architecture of XACML mechanism is flexible to incorporate access control solutions from Attribute Based Access Control (ABAC), Role Based Access Control (RBAC) or Policy Based Access Control (PBAC). To best of our knowledge, the mentioned policies related to XACML are yet not integrated to smart environment with Semantic Web technologies. The architecture of access control service layer using XACML components is illustrated in Figure 1.

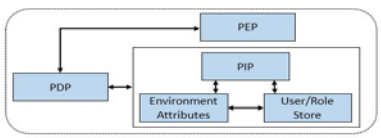

Figure 1. Access Control Service Layer

Each component of this access control service layer offers different set of rules to grant access to information which is termed as resource from this point. The access control rules enforced by each component are defined using mathematical definitions as follows.

4.1.1 Policy Information Point (PIP):

This component serves as a source for identifying values of XACML access request attributes. If there are missing attributes in the request sent from PEP, then PIP identifies them before the PDP evaluates the policy.

In a smart environment given with *n* resources, *m* users, and set of four actions (read, write, edit and delete), we have a maximum of $(4 \times n)$ possible permissions, while associated with user we have maximum of $[m \times (4 \times n)]$. In order to show the operations performed by PIP, we define possible set of actions as below:

- Let 'T' be a set of RDF resources.
- Then, set of 'T' are closed under operations of Select$\odot$, Create$\uplus$, Append$\oplus$, and Delete$\otimes$.
- Let 'Q' be a SPARQL query made over 'T'.
- Let '$T1$' is queried RDF resource and '$Q1$' is query made over 'T_1', then we have four (04) possible actions as:

1. Select: $(m \times (Q1 \odot T1))$
2. Create: $(m \times (Q1 \uplus T1))$
3. Append: $(m \times (Q1 \oplus T1))$
4. Delete: $(m \times (Q1 \otimes T1))$

4.1.2 Policy Decision Point (PDP):

This component evaluates applicability of an access control policy over a resource and returns an authorization decision to the PEP. In order to process PDP, set of rules are to be defined and their variables are used as follows:

- Let '$\mathcal{P}$' be a permit, '$\mathcal{D}$' be a partial Deny and, '$\mathbb{D}$' be a full Deny, for request over resource '$\mathcal{R}$'.
- Let '$\mathcal{F}$' be False, '$\mathcal{T}$' be True, to show availability of resource.
- Let '$\mathcal{I}$' be Immediate and '$\mathcal{N}$' be Not-applicable, over action '$\mathcal{A}$' and, giving multiple decisions as 'n'.

Definition 1. *Deny Override*

For resource authorization, if there is any denial of operation over resource then all decision over that operation will be denied.

If any resource $\mathcal{R}$ is partially denied to be accessed that means ($\mathcal{R}$=$\mathcal{D}$) then, single decision is made as:

$$[(1 \times \mathcal{D}) = \mathbb{D}]$$

Similarly, multiple decisions are made as:

$$[(n \times \mathcal{D}) = \mathbb{D}]$$

Definition 2. *Immediate Applicable*

If resource is available for access and unambiguously gives permit/deny for all operations than action taken is immediate for permit/deny. Similarly, if resource is not available then action is immediate given for not-applicable request. If any resource R is available i.e. $(\mathcal{R} = \mathcal{T})$ and multiple decision are permitted, i.e. $(n = \mathcal{P})$ then action is immediate i.e.

$$[\mathcal{A} = \mathcal{I} \times (\mathcal{P} \vee \mathcal{D})]$$

Similarly, if any resource $\mathcal{R}$ is not available i.e. $(\mathcal{R} = \mathcal{F})$ then action is immediate and not-applicable. i.e.

$$[\mathcal{A} = \mathcal{I}(\mathcal{N})]$$

Definition 3. *Only One Applicable*

This rule evaluates all policies to conclude final evaluation.

- All policies are evaluated and none is applicable on a resource than there is not-applicable decision for all operation request on that resource.
- All policies are evaluated and more than one is applicable on resource then decision is immediate.
- All policies are evaluated and only one is applicable on resource then action is evaluated on only that single policy.

These are given here as:

- If for all decision, there need to be an action. $(\forall n \exists \mathcal{A};\ n \rightarrow \mathcal{A})$, and there are some decisions which are Not-applicable$(\exists n \rightarrow \mathcal{N})$, then all actions are Not-applicable over that resource. i.e.

$$[(\mathcal{A} = \mathcal{N}) \rightarrow \mathcal{R})].$$

- If for all decision, there need to be an action i.e. $(\forall n \exists \mathcal{A};\ n \rightarrow \mathcal{A})$, and there are some decisions which

are applicable ($\exists n \rightarrow !\mathcal{N}$), then all actions are immediate over resource. i.e.

$$[(\mathcal{A} = \mathcal{I}) \rightarrow \mathcal{R})].$$

- If for all decision, there need to be an action i.e. $(\forall n \exists \mathcal{A};\ n \rightarrow \mathcal{A})$, and all decisions are applicable $(\forall n \rightarrow !\mathcal{N})$, then all actions are immediate over resource. i.e.

$$[(\mathcal{A} = \mathcal{I}) \rightarrow \mathcal{R})].$$

Definition 4. *Permits Override*

This rule works as if set of policy evaluates on permit, then action is immediately set to permit, while if all policy sets are not applicable then action is immediately not-applicable.

For resource '$\mathcal{R}$', if there is some decision, for which there are permit action, i.e.$(\exists n \rightarrow \mathcal{A} \times \mathcal{P})$ then, for all decisions there is an immediate permission, i.e.

$$(\forall n\ = \mathcal{I} \times \mathcal{P}).$$

Similarly, for all decisions, for which there are Not-applicable actions, i.e. $(\forall n \rightarrow \mathcal{N})$ then, for all decisions there is an immediate Not-applicable decision, i.e.

$$(\forall n = \mathcal{N}).$$

4.1.3 Policy Enforcement Point (PEP):

This entity performs access control authorization by working over decision requests and enforce policy decisions for authorized access to a resource. The SPARQL query request from query endpoint will be serviced after authorization by PEP. PEP sends XACML request to the PDP and receives decision for authorization i.e. permit or deny access to a resource, and acts on the query request.

All rulings from PDP are enforced by PEP based on filters for decision of pass or fail the request further to SPARQL query engine for further query processing, based on following definitions.

1) Let '*F*' be the Filter having value '*True*' or '*False*'.
2) Let there be '*Permit*' or '*Deny*' from PEP to query engine for entertaining query request further along.

Definition 5. If ruling from PDP is found then filter '*F*' passes request as *True*, and forwards query permit to query engine, i.e.

$$(F = True \rightarrow Permit).$$

Definition 6. If ruling from PDP is not found then filter '*F*' passes request as *False*, and forwards query deny to query engine, i.e.

$$(F = False \rightarrow Deny).$$

4.1.4 Environment Attributes and User/Role Store:

These two components acts as storage for information about attributes, users, roles and groups.

4.2 LAYERED ARCHITECTURE OF SEMANTIC WEB BASED SMART ENVIRONMENT

The overall architecture for Semantic Web based smart environment, with multiple layers for information processing and management is illustrated in Figure 2. The architecture is partially implemented using Java based tools for Semantic Web toolkit – Apache Jena, ARQ, TDB and Fuseki [14]. The details of each layer is given as follows:

4.2.1 Semantics Layer:

Semantics Wrapper

Semantics wrapper is a Java component created using RDF APIs for mapping raw data from smart devices and formatting it into meaningful. The RDF data is forwarded to semantics aggregator.

Semantics Aggregator

This component gathers RDF triples and has the capability to aggregate different triples and store as new triples in the triple database. The aggregator is also responsible to generate rule based triples with re-gathering and merging values from existing triples.

4.2.2 Database Layer:

Triple Database(TDB) Stores and Mediator

The RDF triples are stored in a Jena TDB store and Mediator is used to implement higher level services for synchronization of RDF information.

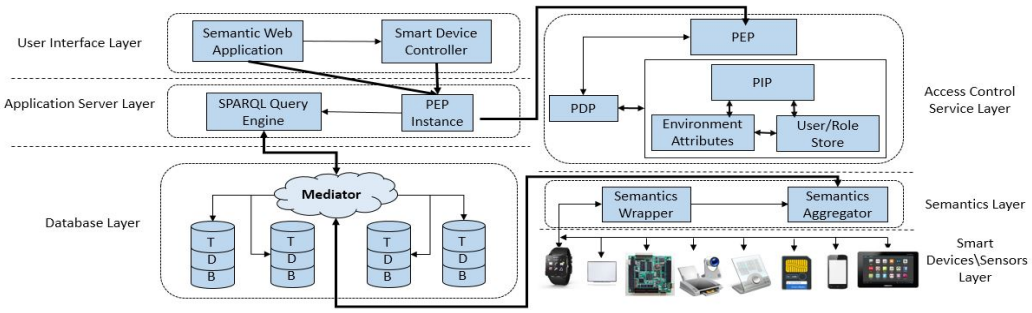

Figure 2. Access Control Authorization as a Service in Semantic Web based Smart Environment

4.2.3 Access Control Service Layer:

This layer utilizes XACML in Java-based tools to enforce authorization as a service for ensuring privacy of information. The PEP module receives SPARQL query request from end user application and authenticates the role of user after it receives authorization decisions from PDP. The PDP evaluates the authorization decision for resource attribute and return it to PEP.

4.2.4 Application Server Layer:

SPARQL Query Engine

The triples from TDB are queried using SPARQL queries. The query engine acts as a HTTP local host server and supports to give access to query endpoints using RESTful services. The server layer hosts multiple query engines to support distributed smart environment.

PEP (Policy Enforcement Point) Instance

The instance of PEP is hosted over server and query request made from user interface layer is directed towards this module for user authentication and authorization. In distributed smart environment, there exist multiple instances of PEP for concurrent access control.

4.2.5 User Interface Layer:

Semantic Web Application

This application is created using Fuseki, which provides SPARQL server to be accessed by end users. It accepts queries and forwards to query engine. Permission to process query is provided after permit decision is evaluated at PEP at access control service layer.

Smart Device Controller Application

This is Java application to enforce event-based actions or time-based queries using query engine. Using the obtained RDF results, it generates rule-based decisive commands for smart devices, and manage behavior of connected smart sensors and devices.

5. SEMANTIC WEB BASED SMART EDUCATION ENVIRONMENT (SWSEE): A USECASE SCENARIO

In order to encapsulate the feasibility of access control as a service in smart environment, we present transition of “smart education environment” to “Semantic Web based Smart Education Environment (SWSEE)”. Figure 3 highlights the key domains and scope of SWSEE. The features of smart environment are inhibited for procurement of smart education through utilization of heterogeneous smart devices.

Traditionally smart education environment enables data sharing among different connected devices through web services via Internet and mobile communication. However, proposed SWSEE aims to share device data in RDF format by using Semantic Web’s REST-full services. The data representation as RDF enables its utilization for reasoning and inferencing based on associated ontology [15, 16]. The inferred decisions can be used in isolation by desired smart environment audience or as event-based decision rules to exhibit controlled device output.

The proposed layered architecture of XACMLaaS when applied as SWSEE would be practiced by a Semantic Web

application whose targeted role audience broadly includes: teacher, student and administrative staff. Kim et al. [17] in their research have explored smart literacy standards and discussed about smart learning environment. Yet, there is a huge lapse of access control at this level of information sharing. The combination of attribute and role based access control is required for secure information sharing within SWSEE, which is effectively possible with XACMLaaS.

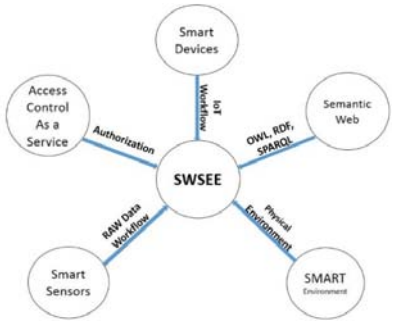

Figure 3. Semantic Web based Smart Education Environment – Domain and Scope

6. CONCLUSION

Integration of smart environment with Semantic Web provides context-awareness to the smart information. For candid utilization, this information needs to be protected from unauthorized access. In this paper, we present a novel approach of deploying access control as a service within Semantic Web based smart environment architecture by using XACML. The proposed approach of XACMLaaS targets fine-grained access control for smart device data using defined set of layer components. The proposed service layout shows efficient adaptation of XACML components within layered architecture of smart environment. We discussed various rules in detail for individual component of proposed access control service layer for an effective access control enforcement. This novel research provides authorized information accessibility, rule-based decision making and reasoning capabilities to intelligently manage data of smart devices.

In future, we aim to fully integrate proposed service layered architecture with smart devices and perform real-time simulations for smart environment.

Reference

[1] M. Weiser, “The computer for the 21st century,” *Scientific American*, Vol. 265, no. 3, pp. 94 – 104, 1991. http://dx.doi.org/10.1038/scientificamerican0991-94

[2] C. Ramos, J. C. Augusto, D. Shapiro, “Ambient intelligence-the next step for artificial intelligence,” *Intelligent Systems*, IEEE, vol. 23, no. 2, pp. 15 – 18, 2008. DOI: http://dx.doi.org/10.1109/mis.2008.19

[3] T. Berners-Lee, J. Hendler, O. Lassila, “The Semantic Web,” *Scientific American*, vol. 284, no. 5, pp. 28 – 37, 2001. dx.doi.org/10.1038/scientificamerican0501-34

[4] E. Viljamaa, J. Kiljander, J.-P. Soininen, A. Ylisaukko-oja, “A smart control system solution based on Semantic Web and uID,” in *Int’l. Conf. on Mobile Ubiquitous Computing, Systems, Services and Technologies(UBICOMM2011)*, Nov. 2011, pp. 105 – 110.

[5] B. Parducci, H. Lockhart, R. Levinson, M. McRae, “Extensible access control markup language – version2.0,” *OASIS Standard*, 2005. https://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf

[6] H. Shen, “A semantic-aware attribute-based access control model for web services,” in *Algorithms and Architectures for Parallel Processing*. Springer, pp. 693 – 703, 2009. http://dx.doi.org/10.1007/978-3-642-03095-6_65

[7] U. Lang, “OpenPMF SCaaS: authorization as a service for cloud & SOA applications,” *in Int’l Conf. on Cloud Computing Technology and Science*, IEEE, Nov. 2010, pp. 634-643. http://dx.doi.org/10.1109/cloudcom.2010.13

[8] R. Laborde, F. Barrère, and A. Benzekri, “Toward authorization as a service: a study of the XACML standard,” *In Proc. of the 16th Communications & Networking Symposium*, Apr. 2013, p. 9. http://dl.acm.org/citation.cfm?id=2499995

[9] K. Maria, E. Vasilis, and A. Grigoris, “S-CRETA: Smart classroom real-time assistance,” *Ambient Intelligence-Software and Applications*. Springer, pp. 67 – 74, 2012. http://dx.doi.org/10.1007/978-3-642-28783-1_9

[10] T. Cadenhead, M. Kantarcioglu, V. Khadilkar, and B. Thuraisingham, "Design and implementation of a cloud-based assured information sharing system," *Computer Network Security*. Springer, pp. 36 – 50, 2012. http://dx.doi.org/10.1007/978-3-642-33704-8_4

[11] D'Elia, J. Honkola, D. Manzaroli, and T. S. Cinotti, "Access control at triple level: Specification and enforcement of a simple RDF model to support concurrent applications in smart environments," *Smart Spaces and Next Generation Wired/Wireless Networking*, Springer, pp. 63 – 74, 2011. http://dx.doi.org/10.1007/978-3-642-22875-9_6

[12] V. Khadilkar, T. Cadenhead, M. Kantarcioglu, and B. Thuraisingham, "Assured information sharing (AIS) using private clouds," *High Performance Cloud Auditing and Applications*, Springer, pp. 215 – 255, 2014. http://dx.doi.org/10.1007/978-1-4614-3296-8_9

[13] M. Compton, P. Barnaghi, L. Bermudez, R. GarcıA-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, "The SSN ontology of the W3C semantic sensor network incubator group," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 17, 2012, pp. 25 – 32. http://dx.doi.org/10.1016/j.websem.2012.05.003

[14] Apache Jena https://jena.apache.org/ (last accessed on July 06, 2016)

[15] W. Jun, S.K. Hong, "A Study on Development of Smart Literacy Standards of Teachers and Students in Smart Learning Environments", *Journal of Internet Computing and Services (JICS)*, Vol. 14, no. 6, pp.59-70, 2013. http://dx.doi.org/10.7472/jksii.2013.14.6.59

[16] S. Kamal, R. Ibrahim, I. Ghani, "An Improved Combined Content-similarity Approach for Optimizing Web Query Disambiguation", *Journal of Internet Computing and Services (JICS)*, Vol. 16, no. 6, pp.79-88, 2015. http://dx.doi.org/10.7472/jksii.2015.16.6.79

[17] T. Kim, J. Y. Cho, B. G. Lee, "A Study on the Development Strategy of Smart Learning for Public Education", *Journal of Internet Computing and Services (JICS)*, Vol. 16, no. 6, pp.123-131, 2015. http://dx.doi.org/10.7472/jksii.2015.16.6.123

◐ Authors ◑

Isma Farah Siddiqui

2006, Bachelors of Software Engineering, Mehran UET, Pakistan.
2008, Masters of Engineering in Information Technology, Mehran UET, Pakistan.
2012~Present, Ph.D in Computer Science at Hanyang University ERICA , S. Korea.
2006~Present: Faculty Member, Dept. of Software Engg, Mehran UET, Pakistan.
Research Interests: Information Management, Semantic Web, Smart Environment, IoT.
E-mail: isma2012@hanyang.ac.kr

Scott Uk-Jin Lee

2004 University of Auckland, Bachelor of Engineering in Software Engineering (학사)
2009 University of Auckland, Doctor of Philosophy in Computer Science (박사)
2011～현재 한양대학교 컴퓨터공학과 교수
관심분야 : 소프트웨어공학, 웹
E-mail : scottlee@hanyang.ac.kr