I.J. Computer Network and Information Security, 2011, 1, 47-53
Published Online February 2011 in MECS (http://www.mecs-press.org/)

A Model of Workflow-oriented Attributed Based Access Control

Guoping Zhang
School of Computer & Communication Engineering
China University of Petroleum, Dong Ying, China
Email: zhanggp@upc.edu.cn

Jing Liu
School of Computer & Communication Engineering
China University of Petroleum, Dong Ying, China
Email: liujing415jsj@163.com

***Abstract*—the emergence of "Internet of Things" breaks previous traditional thinking, which integrates physical infrastructure and network infrastructure into unified infrastructure. There will be a lot of resources or information in IoT, so computing and processing of information is the core supporting of IoT. In this paper, we introduce "Service-Oriented Computing" to solve the problem where each device can offer its functionality as standard services. Here we mainly discuss the access control issue of service-oriented computing in Internet of Things. This paper puts forward a model of Workflow-oriented Attributed Based Access Control (WABAC), and design an access control framework based on WABAC model. The model grants permissions to subjects according to subject atttribute, resource attribute, environment attribute and current task, meeting access control request of SOC. Using the approach presented can effectively enhance the access control security for SOC applications, and prevent the abuse of subject permissions.**

***Index Terms*—Internet of Things, Service-Oriented, Access Control, Task, Attribute, SAML, XACML**

I. Introduction

With the development of networks, Internet of Things attracts the attention of people recently. The emergence of "Internet of Things" [1] breaks previous traditional thinking, which integrates physical infrastructure and network infrastructure into unified infrastructure. While there is no global consensus on the meaning of IoT, it is clear that the main idea behind the IoT concept is the ability to connect loosely defined smart objects and enable them to interact with other objects, the environment, or more complex and legacy computing devices [2]. As we are moving towards the "Internet of Things", millions of devices will be interconnected, providing and consuming information available on the network and cooperate. It is also clear that computing and processing of information is the core supporting of IoT. A typical structure of IoT is shown in Fig. 1. Each product is embedded an electronic tag which includes a unique code about the product. The particular product information corresponding to the code is stored in local

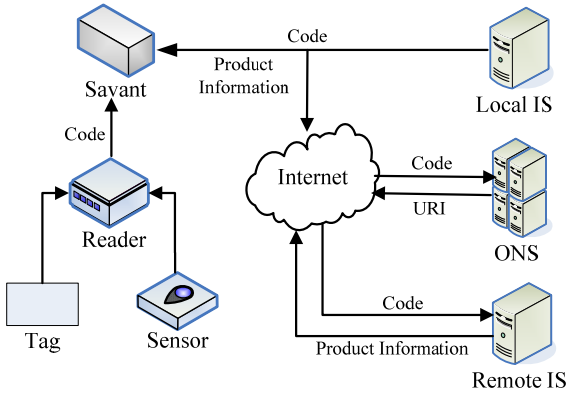

Figure 1. The Simple Structure of IoT.

information server or remote information server. The product code in the tag is read by the reader and that code will be forwarded to the Savant. If the product information is stored in local information server, the Savant will directly communicate the information to the the enterprise application. If not, the code will consult the ONS over the Internet to obtain the location of the information (URI) corresponding to that particular code. And then get the product information from remote information server via the URI.

In this paper, we introduce "Service-Oriented Computing" (SOC) to solve the problem. SOC is the computing paradigm that utilizes services as fundamental elements for developing applications/solutions. To build service-oriented applications relies on the service-oriented architecture (SOA), which is a logical way of designing a software system to provide services to either end-user applications or other services distributed in a network through published and discoverable interfaces [3,4,5]. In IoT, each device can offer its functionality as standard services, and mobile clients offer increasingly sophisticated methods to capture information, to make use of context information and to interact directly with objects in the real world [6,7]. Global Sensor Web is a typical sensor network system using service-oriented ideology, its goal is to make people access or control all

sensors, instruments and imaging equipments through network.

However, a key challenge in a service-oriented environment is the design of effective access control schemas. In SOC, services will be invoked by a large number of temporary subjects, and at the same time authentication and authorization need to cross several security domains frequently, raising new demands for access control of SOC.

In this paper, we mainly discuss the access control issue of service-oriented computing in Internet of Things, and present a Workflow-oriented Attributed Based Access Control model (WABAC). This paper is organized as follows: Section 2 describes the basic concept and definition of WABAC model. In Section 3, we discuss the implementation of WABAC model, offering an access control framework. Next Section gives an example about service-oriented computing application. Section 5 introduces related work. Finally, the conclusion is given and the future work is pointed out.

II. WABAC Model

A The basic concept

1) Subject Attribute(SA): A subject is an entity that takes action on a resource, such as a user, application, or mobile devices, and its set is recorded as S. Each subject has associated attributes which define the identity and characteristics of the subject, such as subject's identifier, IP address, Email address and so on [8].

The set of subject attributes is recorded as SA, if there are K subject attributes, denote them by SA_k, k=1,2,…,K.

2) Resource Attribute(RA): A resource is an entity that is acted upon by a subject, such as service, data or smart device, and its set is recorded as R. As with subjects, resources have attributes (e.g., resource's identifier, geographical position or creation date) that can be leveraged to make access control decisions.

The set of resource attributes is recorded as RA, if there are M resource attributes, denote them by RA_m, m=1,2,…,M.

3) Environment Attribute(EA): Environment Attribute describes situational environment or context in whicjh the information access occurs. Environment attributes such as current date, or the network's security level, are different from subject attributes or resource attributes, but may be used in applying an access control policy.

The symbol of E expresses environment, and the set of environment attributes is recorded as EA, if there are N environment attributes, denote them by EA_n, n=1,2,…,N.

4) Attribute Distribution(ATTR): Attribute Distributi -on means attribute assignment for subject, resource and environment. ATTR(s), ATTR(r), and ATTR(e) are attribute assignment relations for subject s, resource r, and environment e, respectively:

$$ATTR(s) \subseteq SA_1 \times SA_2 \times ... \times SA_K$$
$$ATTR(r) \subseteq RA_1 \times RA_2 \times ... \times RA_M$$
$$ATTR(e) \subseteq EA_1 \times EA_2 \times ... \times EA_N$$

We also use the function notation for the value assignment of individual attributes. For example:

IP(s) = "101.126.1.15"
ServiceName(r) = "TVOnline"
CurrentTime(e) = "09:30"

5) Task(T): Task is a fundamental unit of business work or business activity. It is distinguishable action which may be relevant to multiple users or include several subtasks [9]. For example, Electronic Toll Collection flow includes three tasks: information collection, vehicle billing and bank charges.

6) Task State(TS): No less than any course, task has also its own lifecycle in which including five states, i.e. reday, active, hold, end and invaild that showen in Fig. 2.

7) Authorization Unit(AU): Authorization Unit is an abstraction of task, being made up of access subject, O and access permissions, P. If AU need external user to participate in, see the external user as access subject, O. If not, set O as null. P is the least set of permissions that complete a task. According to the executing process of workflow, AU can be divided into four categories: order unit, select unit, loop unit and concurrent unit.

8) Authorization Dependency (AD): Authorization Dependency is the relations between AU in workflow. For any AU_1 and AU_2, the main dependency includes the following:

a) Order Dependency: AU_2 can be activated only after AU_1 has been finished, being written as $\mathbf{AU_1} \longrightarrow \mathbf{AU_2}$.

b) Defeat Dependency: AU_2 can be activated only after AU_1 has defeated, being written as $\mathbf{AU_1} \overset{\times}{\longrightarrow} \mathbf{AU_2}$.

c) Divided Permission Dependency: AU_1 and AU_2 must be executed by different user, being written as $\mathbf{AU_1} \longleftrightarrow \mathbf{AU_2}$.

d) Agent Dependency: AU_1's permissions can be surrogated to AU_2 when AU_1 is aborted, being written as $\mathbf{AU_1} \overset{d}{\longrightarrow} \mathbf{AU_2}$.

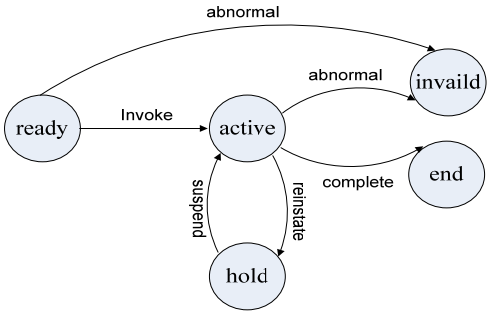

Figure 2. The State Transition of Task.

B. WABAC Model

WABAC = { S, R, E, P, T, AU } where: S is subject; R is resource; E is current environment; P is the set of permissions; T is the set of tasks; AU is the set of authorization unit. This model includes the following relations:

1) Permission Assignment: a many-to-many permissi -on to subject assignment relation. Permission assignment depends on subject attribute, resource attribute, environment attribute, and current task.

2) Authorization Dependency: Authorization Depend -ency decides the execution flow of workflow. The

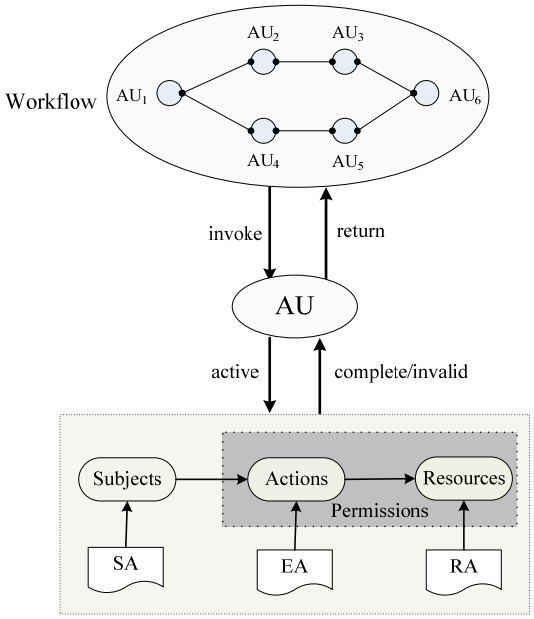

Figure 3. The Access Control View of WABAC.

relationship between AU is $AU \times AU \subseteq 2^D$, D={Order Dependency, Defeat Dependency, Divided Permission Dependency, Agent Dependency}.

3) Permission Transformation: In the process of access, subject's permissions change with task. Here we use authorization unit sequence to show permission transformation, such as $\{AU_1, AU_2, \ldots, AU_n\}$, AU_1, AU_2, …, AU_n are specific authorization unit.

WABAC model supports two famed security control principles in authorization:

(1) **principle of least privilege:** Grant least permissions for subject in the executing processes of task. If a task is aborted abnormally, then freeze subject's permissions. When a task is completed successfully, retrieve subject's permissions at once, in order to prevent the abuse of permissions.

(2) **separation of duties principle:** Different tasks in a system may need different subjects to implement, e.g., in Electronic Toll Collection flow, vehicle billing and bank charges need different subjects to complete, dong this by divided permission dependency of authorization units.

The model of WABAC can realize fine-grained access control of cross-domain system, also manage subject's permissions dynamically. This model is suit for access control of SOA, especially workflow based distributed computing system. Fig. 3 depicts the access control view of WABAC. The following will discuss the implemention of WABAC model, and give an access control framework.

Ⅲ. IMPLEMENTATION OF WABAC

A.. Presentation of Attribute

This model makes access control decision on the basis of certain attributes, so needs an effective carrier to declare these attributes, especially subject attribute. Here select SAML to declare subject attribute. SAML (Security Assertion Markup Language) which is published by OASIS is an XML-based standard for web service security, and provides a standards-based approach to the exchange of information, including attributes, that are not easily conveyed using other WS-Security token formats [10]. SAML defines three different assertions: authentication assertion, attribute assertion and authorization decision assertion which carries authentication information, attribute information and authorization decision information respectively. In SAML standard, we can also transport SAML protocol messages between participants by lower-level communication or messaging protocols, e.g., the SOAP-over-HTTP binding can be used to exchange SAML request/response protocol messages. A SOAP message sender obtains a SAML assertion by means of the SAML Request/Response protocol. A simple example about SAML attribute query message being transported within a SOAP envelope can be seen in Fig. 4. After SOAP message sender obtains SAML assertion about subject attribute, SAML assertion is carried within the header of SOAP envelope, and is sent to system access port.

```
<?xml version="1.0" encoding="UTF-8"?>
 <env:Envelope
   xmlns:env=”http://www.w3.org/2003/05/soap/envelope/”>

  <env:Body>
   <samlp:AttributeQuery
     xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
     xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
     ID="aaf23196-1773-2113-474a-fe114412ab72"
     Version="2.0"
     IssueInstant="2006-07-17T20:31:40Z">
    <saml:Issuer>http://example.sp.com</saml:Issuer>

    <saml:Subject>
     <saml:NameID
       Format="urn:oasis:names:tc:SAML:1.1:nameidformat:
       X509SubjectName">
       C=US, O=NCSA-TEST,
       OU=User, CN=trscavo@uiuc.edu
     </saml:NameID>
    </saml:Subject>

    <saml:Attribute
      NameFormat="urn:oasis:names:tc:SAML:2.0:attrnamef
      ormat:uri" Name="urn:oid:2.5.4.42"
      FriendlyName="givenName">
    </saml:Attribute>

   </samlp:AttributeQuery>
  </env:Body>
 </env:Envelope>
```

Figure 4. A Simple Example about Attribute Query.

B. Access Policy Language

In 2003, OASIS put forward XML Access Control Markup Language named XACML which provides support for attributed based access control. XACML is an XML-based language, which defines the syntax and


```
<?xml version=1.0" encoding="UTF-8"?>
<Policy xmlns="urn:oasis:names:tc:xacml:1.0:policy"
   PolicyId="SimplePolicy" RuleCombiningAlgId="rule-combining-algorithm:deny-overrides">
<Description> Video Store access control policy </Description>

<Target>
  <Subjects>
    <AnySubject/>
  </Subjects>
  <Resources>
    <AnyResource/>
  </Resources>
  <Actions>
    <AnyAction/>
  </Actions>
</Target>

<Rule RuleId= "urn:oasis:names:tc:xacml:1.0:example:SimpleRule" Effect="Permit">
    <Description> The manager of video store can read any resource </Description>
    <Target>
     <Subjects>
       <Subject>
         <SubjectMatch MatchId=" urn:oasis:names:tc:xacml:1.0:function:string-equal">
           <SubjectAttributeDesignator
              AttributeId= "urn:example:role" DataType="http://www.w3.org/2001/XMLSchema#string"/>
           <AttributeValue DataType= "http://www.w3.org/2001/XMLSchema#string">manager</AttributeValue>
         </SubjectMatch>
       </Subject>
     </Subjects>
     <Resources>
       <AnyResource/>
     </Resources>
     <Actions>
       <Action>
          <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
            <ActionAttributeDesignator
               AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
               DataType="http://www.w3.org/2001/XMLSchema#string"/>
            <AttributeValue  DataType="http://www.w3.org/2001/XMLSchema#string">read</AttributeValue>
          </ActionMatch>
       </Action>
     </Actions>
  </Rule>
    …
</xacml:Policy>
```

Figure 5. A Simple Policy in XACML.

semantics of a language for expressing and evaluating access control policies, provides an access control framework to control all of authorization process. XACML framework is constituted of multiple components, including Policy Administration Point(PAP), Policy Decision Point(PDP), Policy Enforcemtne Point(PEP), Policy Information Point(PIP) and Context Handler. XACML is intended to be suitable for a variety of application environments by the XACML context [11]. Compared with other policy language, XACML access policy is based on subject attribute, resource attribute and environment attribute, and can achieve fine-grained access control. In addition, it can define new functions, data structure and combinational logic algorithm as needed. A simple access control policy in XACML is seen in Fig. 5.

Both SAML and XACML are XML-based security standard, and can be identificated by people and computer at the same time. SAML assertions provide a method to distribute security-related information, one of the most important purposes is as input to Access Control decisions. XACML provides an access control framework to make decision who can access which resource. As a result, SAML and XACML can be used alone, and also can be used together [12,13,14].

Here we use SAML as carrier of subject attribute, XACML as access policy language, make the suitable

extension of XACML framework, and put forward WABAC access control framework that showen in Fig. 6.

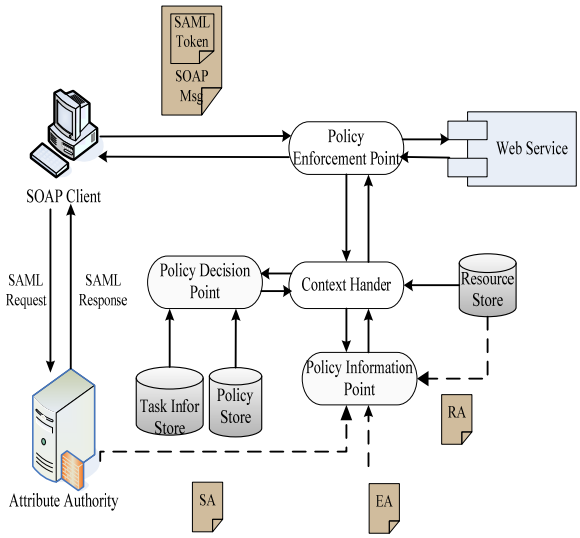

Figure 6. WABAC Access Control Framework.

- Before subject accesses system, subject must have SAML attribute declaration which is published by Attribute Authority. Then insert SAML declaration into SOAP Head, and send SOAP request to system access port via SOAP client.
- When system gets subject's access request, generate corresponding task instances, allowing each task to be in a ready state. As for these tasks, they will be carried out automatically in definite logical order.
- Once certain task is activated, Policy Enforcement Point (PEP) shoule pick up subject attributes and current task information from SOAP request message. At the same time, write current task state into Task Information Store, and create a standard XACML authorization request, send it to Policy Decision Point (PDP).
- After receiving authorization request, PDP will make authorization decision in the light of corresponding policies from Policy Store and current task state from Task Information Store. If need other attributes in authorization decision, get them through Policy Information Point (PIP).
- Context Handler gets the authorization decision, and transforms it into a format which can be accepted by PEP. Then PEP enforces the authorization decision. When certain task is completed successfully, start next task at once.

Ⅳ. AN EXAMPLE

Take Electronic Toll Collection system as an example of WABAC application. Electronic Toll Collection system includes information collection service, vehicle billing service and bank charges service, mainly solving auto fare collection in toll station. The system workflow is following:

- When vehicle is close to toll station, send access request to system. The system creates information collection task, vehicle billing task and bank charges task that will Be carried out automatically according to information collection->vehicle billing->bank charges.
- Information collection equipments at side of road take pictures of vehicle, read electronic tag which is installed in vehicle, and get the only identification code of vehicle and other information that will be sent to control centre.
- Control centre looks for vehicle attributes (e.g., vehicle type, the owner name) from vehicle information store depending on the only identification code of vehicle, which is used to judge whether the vehicle is charged on. If need to charge, send request to vehicle billing service, and the service returns billing result to control centre.
- After sending the owner account information and billing result to bank charges service, this service will deduct certain cost from the owner account, add to vehicle toll account, and return charge result to control centre.
- Finally, send charge result to vehicle terminal, if bank charges task is completed successfully, show "Success Charge". The system will send "permission" command to the exit intercept equipment, allowing the vehicle to pass through the exit.

Ⅴ. RELATED WORK

Since the early 1960s and 1970s, it appeared several of access comtrol models, i.e. Discretionary Access Control (DAC), Mandatory Access Control (MAC), Role Based Access Control (RBAC) [15]. The study of access control in distributed, heterogeneous and service-oriented environemnt is becoming the hot topic in the field of network security.

SHEN Haibo proposes a CGRBAC model for access control in Web services by introducing global roles and local roles [16]. The global role activation depends on the security-relevant environmental context, while local roles activation depends on whether its corresponding local services have been invoked by global services. The model can simplify the role assignment and management in heterogeneous system, and provide more fine-grained access control and improved security for Web services applications.

Xu Feng et al. present a service-orient role-based access control model and security architecture model for Web Services [17]. In this model, a new technology is presented to implement the RBAC on the Web Services by designing the secure cookies and secure SOAP messages. Compared with other RBAC, this model is

easier to express dynamic characters of role and realize single sign-on (SSO) in distributed environment by introducing the notion of Service and Actor.

Xiangning Zhou et al. put forward an access control model of workflow system integrating RBAC and TBAC [18]. In the workflow system, TBAC model can directly achieve the access control, but sometimes need to describe the dynamic characteristics in the system. As result, it is necessary that integrating RBAC and TBAC. This model combines the needed relationships in the workflow, and can express the workflow's control mechanism clearly.

More papers [19], [20], [21] improve RBAC model and applied it to the access control of different applications. However, RBAC model will be faced with trivial role management work, and also can not solve large number of temporary users in SOC environment. Therefore, RBAC model is not suitable for access control of Service-Oriented Computing in Internet of Things.

In orded to realize resource sharing between coalitions, SPARTA ISSO first proposed Attributed Based Access Control (ABAC) in 2001 [22]. ABAC model defines permissions based on just about any security-relevant characteristics, known as attributes, mostly including subject attribute, resource attribute and environment attribute. It is more flexible and powerful to describe complex, fine-grained access control semantics [23]. In addition, ABAC model can be easily transformed and integrated into primal systems, and its access control framework is very suitable for distributed system. But it can not manage permissions dynamically, such as revocation of permissions, and not satisfy “principle of least privilege”.

Our paper puts forward Workflow-oriented Attributed Based Access Control model(WABAC) which has the following features:

- Introducing the concept of “Attribute”. Using attribute to describe detailedly entity’s properties (e.g., role may be an attribute of subject) .We can flexibly set access control policy, meeting fine-grained access control requirement.
- Using “Workflow control method”. Workflow is a business process which is composed of multiple relevant tasks. In WABAC, when subject sends an access request, system creates the corresponding task instances automatically. Grant least permissions for subject according to attributes and current task, meeting “principle of least privilege”.
- Managing permissions dynamically. When a task is actived, subject has its own permissions. If a task is aborted abnormally, freeze subject’s permissions. Once a task is completed successful -ly, retrieve subject’s permissions.

Ⅵ. CONCLUSION

In this paper, we focus on the access control of Service -Oriented Computing in Internet of Things. We present a workflow-oriented attributed based access control model and access control framework for SOC application. Compared with other models, this model can achieve fine-grained access control, and manage permissions dynamically by introducing the notion of Attribute and Task, supportting principle of least privilege and separation of duties principle. It is very suit for access control of the service-oriented architectures especially workflow based distributed computing system. But a key issue existing in attribute-based access control is how to protect senstive attribute (e.g., age, location, or bank account) when need to obtain attribute credentials, especially in IoT, there will be a lot of private message needed to protect. In the future, we will use Apache Axis2 platform to deploy this model, and design an effective method to protect senstive attribute.

REFERENCES

[1] International Telecommunication Union UIT, “ITU Internet Reports 2005:The Internet of Things,” 2005.
[2] Carlo Maria Medaglia and Alexandru Serbanati, “An Overview of Privacy and Security Issues in the Internet of Things,” The Internet of Things:20th Tyrrhenian Workshop on Digital Communications, DOI 10.1007/978-1-4419-1674-7_38, pp. 389–395, 2010.
[3] Papazoglou M.P, “Service-oriented computing: Concepts, Characteristics and directions,” In: Proceedings of the 4th International Conference on Web Information Systems Engineering, 2003.
[4] Michael N.Huhns and Munindar P.Singh, “Service-Oriented Computing: Key Concepts and Principles,” IEEE Internet Computing, February 2005, pp. 75–81.
[5] W3C Working Group Note, “Web Services Architecture,” l1 February 2004.
[6] Patrik Spiess and Stamatis Karnouskos, “SOA-based Integration of the Internet of Things in Enterprise Services,” IEEE International Conference on Web Services, 2009.
[7] Sven Siorpaes et al., “Mobile Interaction with the Internet of Things,” In Adjunct Proceedings of the 4th International Conference on Pervasive Computing (Pervasive 2006), ISBN 3-85403-207-2, May 2006.
[8] Eric Yuan and Jin Tong, “Attributed Based Access Control (ABAC) for Web Services,” Proceedings of the IEEE International Conferenceon Web Services (ICWS 05), 2005, pp.560–569.
[9] R.K.Thomas and R.S.Sandhu, “Task-based Authorization Controls (TBAC): A Family of Models for Active and Enterprise-oriented Authorization Management,” Proceedings of the IFIP WG11.3 Workshop on Database Security, Auguest 1997.
[10] http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security
[11] http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml
[12] Han Tao, “XACML-based Access Contorl Model for Web Service,”Wireless Communications Networking and Mobile Computing 2005 Proceedings, 2005.9, pp. 1140–1144.
[13] Markus Lorch, Dennis Kafura, and Sumit Shah, “An XACML-based policy management and authorization service for globus resources,” Grid Computing 2003 Proceedings, 2003.11, pp. 208–210.
[14] Torsten Priebe, Wolfgang Dobmeier, Christian Schläger, and Nora Kamprath, “Supporting Attribute-based Access Control in Authorization and Authentication

Infrastructures with Ontologies," Proceedings of the First International Conference on Availability,Reliability and Security (ARES 06), April 2006.
[15] Ravi S. Sandhu et al., "Role-Based Access Control Models," IEEE Computer, February 1996, pp. 38–47.
[16] SHEN Haibo and HONG Fan, "A Context-Aware Role-Based Access Control Model for Web Services," Proceedings of the 2005 IEEE International Conference on e-Business Engineering (ICEBE 05).
[17] Xu Feng, Lin Guoyuan, Huang Hao, and Xie Li, "Role-based Access Control System for Web Services," Proceedings of the Fourth International Conference on Computer and Information Technology (CIT 04).
[18] Xiangning Zhou and Zhaolong Wan, "An Access Control Model of Workflow System Integrating RBAC and TBAC," In IFIP International Federation for Information Processing, vol. 252, 2007, pp.246–251.
[19] R. Bhatti, E. Bertino, and A. Ghafoor, "A Trust-based Context-Aware Access Control Model for Web Services," IEEE International Conference on Web Services (ICWS'04) Proceedings, March 2004.
[20] Min Wu, Jiaxun Chen and Yongsheng Ding, "Study on Role-Based Access Control Model for Web Services and its Application," Proceedings of the 5th WSEAS International Conference on Telecommunications and Informatics, May 27-29, 2006, pp. 41–45.
[21] Manachai Toahchoodee et al., "A Trust-Based Access Control Model for Pervasive Computing Applications," Data and Applications Security 2009, LNCS 5645, pp. 307–314, 2009.
[22] http://www.isso.sparta.com/documents/
[23] Lingyu Wang, Duminda Wijesekera and Sushil Jajodia, "A logic-based framework for attribute based access control," Proceedings of the 2004 ACM workshop on Formal methods in security engineering, ISBN: 1-58113-971-3, 2004.

Guoping Zhang holds a bachelor degree in the application of computer from China University of Petroleum, Dong Ying, China in 1992, and a master's degree in the application of computer from China University of Petroleum, Dong Ying, China in 2001.

He is working for School of Computer and Communication Engineering in China University of Petroleum since 1992. He was promoted to the lecturer in 1998, and was promoted to the associate professor in 2003. During his working time, he was engaged in computer specialized teaching, involved in many research projects, and gained school excellent technology achievement award. His main research interests are in the areas of information system and information integration, data base, and data grid. He has published many papers in international conference proceedings and journals in these areas.

Dr. Zhang is a member of IBM, and has been serving on international program commitees, e.g., of the International Conference on Computer Science and Information Technology.

Jing Liu holds a bachelor degree in computer science and technology from Inner Mongolia Finace and Economics College in 2009.

She is getting a master degree of computer science and technology in China University of Petroleum now. Her research direction is the access control of service-oriented computing in Internet of Things.

