Efficient Fine Grained Access Control for RFID Inter-Enterprise System

Bayu Anggorojati, Neeli Rashmi Prasad, and Ramjee Prasad

Center for TeleInFrastruktur (CTIF) Aalborg University, Denmark
E-mail: {ba,np,prasad}@es.aau.dk

Received 24 August 2013; Accepted 12 November 2013;
Publication 23 January 2014

Abstract

Access control management is a very challenging task in an inter-enterprise RFID system due to huge amounts of information about things or objects that can be collected and accessed to and from the system. Furthermore, the information stored in the inter-enterprise RFID system contains sensitive and confidential data related to the activities of the organization involved around the RFID system. Hence, the efficiency and high-granularity are critical in the design of access control for such system. This paper presents a novel access control model which is efficient and fine grained for such a system. A detail definition and mechanism of the access control model are described in the paper. A system implementation is developed for the evaluation purpose. An important performance measure in big data processing is delay in processing time, thus the evaluation aims at measuring the access control processing time. The evaluation results show that the model is consistent, and is able to achieve less delay than the inter-enterprise RFID system without access control at a certain point.

Keywords: access control, policy, security, RFID, IoT.

1 Introduction

RFID technology allows the everyday things to be interconnected to the internet world, thus the key component towards the full deployment of the IoT vision [3]. From the system components and architecture perspectives, RFID

Journal of Cyber Security, Vol. 2 NO. 3 & 4, 221–242.
doi: 10.13052/jcsm2245-1439.232

system employed by any organization in its activity may consist of three sub-systems, namely RF, enterprise, and inter-enterprise sub-system [11]. Inter-enterprise sub-system in particular, is the most important component that enables the objects or things visibility and tracking throughout their life cycle, i.e. in supply chain industry, etc. When the information about thousands of things or objects is able to be gathered and accessed, consequently access control management of such information is a great challenge.

The technical specification, including the standard interfaces and data format, that enables the inter-enterprise information sharing of RFID events data is specified in the EPCIS specification [7] issued by EPC global. The EPCIS repository, i.e. software implementation of this specification, in particular aims at receiving application-agnostic RFID data, translate it into a corresponding business events (e.g. business process, business location, event time, etc), and then make the events available and accessible by upstream applications. Since the EPCIS repository potentially contains sensitive and confidential data of any individual or organization, the access to such information through its interfaces needs to be managed properly. In this regard, it is important to mention that the access control mechanism in the EPCIS specification is left open to each specific implementation. Additionally, efficiency, fine level of granularity, and trust are important keys to the access control design since highly dynamic and huge amount of events data is expected to be generated by potentially thousands of tagged objects which are of any interest for individuals or organizations that have even had any relationship before.

There are two main contributions of this paper. First, a dynamic and efficient access policy mechanism of an object or group of objects, based on the attributes and vocabularies of EPCIS is introduced. This access policy takes the profile of the accessing entity (i.e. individual or organization that requests to access RFID events information – this term will shortly be referred as *user* throughout the rest of this paper), and results in a suitable set of access rule of the corresponding entity to the object(s). This way, the access policy can be dynamically reuse for any user that even had no relationship before. Second, fine grained policy access enforcement method to handle large amounts of RFID events information, using the created rules and contextual information, is presented. For evaluation purpose, a system implementation of the proposed access control model is developed and tested.

The remainder of this paper is organized as follows: An overview of related works in access control is given in Section [2]. The problems, requirements, and realistic assumptions along with a real life example for designing an

access control framework in RFID is described in Section [3]. The proposed access control framework, along with the definition of access policy of the object(s), mechanism to generate access rules for the accessing entity, and the access policy enforcement mechanism, is explained in Section [4]. The system implementation of the proposed access control model is presented in section [5]. The evaluation results and findings are discussed in Section [6]. Some qualitative discussion regarding important features of secure system and access control constituted in the proposed model as well as comparison with existing access control model are presented in Section [7]. Finally, the conclusion and future directions of this work are given in Section 8.

2 Related works

Study in various types of access control models have been quite well established within the computing and information technology field. Our particular interests are in incorporating the contextual information and dynamically create access rules based on a pre-defined set of policies. XACML [13] is an XML framework to describe access control policies for web based resources. The XACML specification incorporates some contextual information into access decisions, but it has no formal context-aware access control model. In addition, the access decision from the evaluated policies in XACML is only limited to four pre-defined categories, i.e. Permit, Deny, NotApplicable, or Indeterminate, which greatly reduces the granularity of access decision results.

Role Based Access Control RBAC [12] is an access control model that is widely used and further derived into different models, due to its suitability in almost any organization which consists of different roles with some levels of hierarchy. Temporal aspects of RBAC were addressed in TRBAC [14], which focuses on temporal availability and dependency of roles. GTRBAC [10] is an extension of TRBAC model that is capable in expressing a wide range of temporal constraints – in particular time periodicity as well as duration, and de/activating as well as enabling constraints – on roles. An XML specification of GTRBAC has been introduced in [6] and the extension of X-GTRBAC which incorporates trust in assigning roles to users has been presented in [5]. Although these models support context-awareness but the role based model, i.e. with user-to-role and role-to-permission mapping, does not fit with the requirement of RFID inter-enterprise system.

CCAAC [1], another type of access control model that supports contextual information and is based on capability. In addition, CCAAC provides a

framework where a valid capability as a mean for an access request to be granted, is created for any user based set of access policies attached to an object or group of objects. Here, *object* refers to resource to be accessed by any user. The CCAAC offers efficient and dynamic way of managing access control through the evaluation of user's profile and contextual information via the corresponding access policies upon the capability request to certain object(s), which is important when dealing with huge numbers of objects and users simultaneously, e.g. in IoT or RFID system. Moreover, it also supports access delegation and revocation. However, the type of action and access decision result limits the level granularity, and the context-awareness is not formally modelled.

A fine grained access enforcement specially designed for the EPCIS events data through a rule-based policy language for Auto-ID events, called AAL, has been introduced in [9]. In addition, an efficient policy enforcement mechanism and implementation based on SQL query rewriting was presented. The main drawback of AAL as presented in [9] is that the access policy is manually assigned to users which is impractical in the real life situation. The dynamic generation, assignment, and revocation of access policies were not considered as well.

3 System requirements

The RFID events data in the EPCIS repository, which is known as EPCIS events, is categorized into four types of events namely *Object Event*, *Aggregation Event*, *Quantity Event*, and *Transaction Event* [7]. Each of them describes different type of event taking place in relation to the RFID tag, which is represented by EPC ID, in the business process within the company. In addition, two types of data, i.e. RFID application-agnostic and master/company data, are comprised in the EPCIS repository. Here, the master data, e.g. event Time, action, bizStep, etc, provides some necessary business context to interpret the EPCIS events [7].

A simplified example of an inter-enterprise RFID system deployment involving an EPCIS repository is depicted in Fig. 1. In this example, the EPCIS repository is owned by a company c_1, and is accessed by companies c_2 and c_3. The RFID events consisting of RFID and master data generated by a BEG module owned by c_1 is captured via capture interface and stored in the EPCIS repository as EPCIS events. The EPCIS events stored in EPCIS repository can then be accessed by other companies through the query interface. According to a comprehensive definition and description of the EPCIS specification [7], the

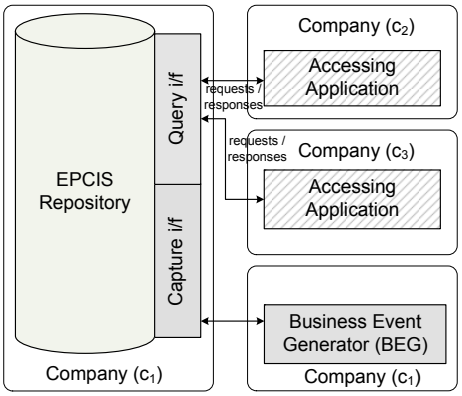

Figure 1 The interactions of EPCIS repository interfaces in an inter-enterprise RFID system

EPCIS events can be interpreted and give a valuable information regarding the business activities of company c_1 by business context information and RFID data. For example, the interpreted EPCIS event can give a figure about production volume, sales activity of certain products, inventory status, etc. Obviously, these are very sensitive information that a company want to reveal as minimum as possible to other parties by managing access to EPCIS event data in high level of granularity. In addition, the the RFID tags' owner may want to restrict the access to particular information related to the activity of the tags which might reveal another type of sensitive information apart from the business related information.

Based on the identified problems, the following requirements for access control in an inter-enterprise RFID system should be foreseen:

- **Context-awareness:** The access control system should be design to support rich business context information that are contained in the EPCIS events.
- **Dynamic rule:** Providing a fine grained access control in a highly dynamic EPCIS events data that is generated continuously, it is almost impossible to assign a static access rights for particular users to certain part of data or attributes. Therefore, dynamic access rule should be generated based on the specified rules in the access policy and the requested set of information. In addition, the policy should support flexible inclusion of new events.
- **Dynamic access assignment:** There are certainly various types of users that are trying to gain access to the EPCIS events data which are probably

not known before by the EPCIS repository's system administrator, i.e. the responsible person to create the access policy. Hence, a dynamic mechanism to assign access rights to users is also required.
- **Object based policy:** The access policy based on particular RFID tag IDs, e.g. on the Object Class or serial numbers level, is also required to restrict the access to information related to tag's activities by the tag's owner.

4 Proposed Access Control Model

4.1 Assumptions

Based on the identified problems and how the inter-enterprise RFID system operates, the following assumptions are made as a baseline to design an efficient, dynamic, fine grained access control for inter-enterprise RFID system:

- The authentication phase has been carried out before the access control process takes place.
- The EPCIS events stored in the EPCIS repository are only events generated by the EPCIS repository's owner, e.g. a company.
- The set of contextual information, i.e. attributes of EPCIS events, such as bizStep, action, disposition, readPoint, epcList, etc, and their values are known to the EPCIS repository's owner. The values of some attributes are always fixed, e.g. $action = \{'ADD', 'OBSERVE', 'DELETE'\}$, while the values of attributes like *epcList* are dynamics but the company has a full knowledge of all EPC IDs involved in their business transactions.
- The EPCIS owner may or may not know the users that are accessing its EPCIS events through the query interface. But the user's profile, such as company name, location, business area, etc, can be obtained through a trusted means, e.g. via trusted third party organization.
- For each query of EPCIS events, the user may optionally specify particular Event Type(s) and some contexts or event attributes, e.g. event Time, action, bizStep, etc, as stated in the EPCIS specification [7]. It is important to note that the default query operation according to the EPCIS specification is that all the available information will be returned unless specified otherwise in the query.
- The RFID tag's owners have the knowledge of their tags' IDs in order to create access policies for an individual tag or a set of tags.

These assumptions lead us to propose an access control model that will be explained in the following subsections.

4.2 Definitions

4.2.1 Elements, Attributes, and Values

First of all, the data structure in our proposed access control framework is based upon $Element \rightarrow Attribute \rightarrow Value$ ternary relationship which maps each element to its attributes and their values. *Element* is a set of elements which could be user profile, P, or EPCIS event, E. Each of the *element* consists of several attributes and each of the attribute can have a set of possible values.

4.2.2 EPCIS Event

Let E_i be the i^{th} EPCIS event within the set of EPCIS events element. As mentioned previously, according to [**?**] E_i could be an *ObjectEvent*, *AggregationEvent*, *QuantityEvent*, or *TransactionEvent*. Each of E_i consists of a set of attributes as defined in details in [**?**]. Let us call the j^{th} attribute of an E_i as AE_j. Here, the relationship between E_i and AE_j can be expressed in the following notation: $E_i = \{AE_1, \cdots, AE_n\}$.

The value of each AE_j, let us call it as VE_{jk}, can either be a single value or a list of values, e.g. in the case of the *epcList*. Moreover, the value of some attributes may be among some pre-defined values. In any case, the relationship between AE_j and VE_k can be generally expressed as $AE_j = \{VE_{j1}, \cdots, VE_{jn}\}$.

4.2.3 User profile

In addition to our previous assumption, the administrator can define a set of user profile based on several attributes and values. Now, let P_i be the i^{th} user profile among a set of user profiles defined by the administrator. Similar to E_i, each P_i consists of several user profile attributes, e.g. AP_j, and each AP_j may consists of a set of values, i.e. VAP_{jk}, defined by the administrator.

$$AP_j = \{VAP_{j1}, \cdots, VAP_{jn}\} \tag{1}$$

4.2.4 Objects

The *Object* field can be expressed as a single tag ID or as a pattern which can apply to a set of tags, according to the TDS specification [8]. In mathematical form, the *Object* O can simply be expressed as a set of object values: $O = \{VO_i, \cdots, VO_n\}$

4.2.5 Condition

Condition is the key component to provide dynamic rule and fine-grained access control. It is important to mention that the *condition* in this proposed model is not a requirement for granting an access or not. Rather, it is defined as a set of constraint applied to the contextual information. In our case, the contextual information is specific to the business contexts or the EPCIS event attributes (AE_j).

The way the condition is expressed in the proposed access control policy follows the white-listing principle, meaning that the access to information that is not explicitly mentioned in the condition is not allowed, which is the opposite of the nature of the EPCIS query specification as explained earlier. This principle is valid in general, and especially in our case where the EPCIS event attributes are defined in great details.

With this principle in mind, a condition can be expressed in general as subset of AE_j, i.e. $AE_j^s \subseteq AE_j$, with respect to a set of all possible values of a particular AE_j. This implies that a condition can set whether to allow all, partly, or none of the values to be accessible. For example, let say an EPCIS event attribute consists some pre-defined values, e.g. $AE_j = \{a, b, c\}$. Some possible condition for such an attribute AE_j are $AE_j^s = AE_j = \{a, b, c\}$, $AE_j^s = \{b, c\}$, or $AE_j^s = \varnothing$, which means that the values of AE_j is shown fully, partially (e.g. only $\{b, c\}$), and none. This way of describing the condition is also to fulfil the purpose of maintaining the consistency upon the existence of multiple rules.

On a practical stand point in writing a condition, there are two possible ways to specify it: first, by explicitly stating what set of values, i.e. VE_{jk} are accessible; second, by describing values in certain ranges using a comparison operator. The first way is more static and suitable for attributes that have some pre-defined values, e.g. action, business location, disposition, etc, whereas the later one is more dynamic and capable of specifying a condition for event that has not happened yet, such as the *event Time*. In such a case, general statement of the condition of the rule i with respect to AE_j^s is as follows:

$$C_i = AE_{i1}^s \wedge \cdots \wedge AE_{ij}^s \wedge \cdots \wedge AE_{in}^s \tag{2}$$

4.2.6 Rule

A *rule* consists of a set of *conditions*, C and an EPCIS event type, ET. General representation of a rule using a **if**-**then** relationship is: $ET \Rightarrow C$. A set of rules together with a user profile P or a set of objects O forms a policy that will be explained in the next Subsection (4.2.7).

4.2.7 Policy

In the proposed model, there are two types of *policies*, namely user based and object based policies. A *user based policy* consists of a set of *profiles* P and a set of rules, while an *subject based policy* consists of a set of *objects* O and a set of rules. A set of different rules that are applicable for a *request*, i.e. matched to the *request*'s P or O, and having the same ET value are combined into a new applicable rule using a *disjunction* operation after resolving all the conflicting policies. In addition, multiple policies may also be applicable to an access *request*, thus a policy combining mechanism should take place in order to create a final applicable rule for an access *request*. Finally, the final applicable rule would then be evaluated with the *conditions* presented within the access *request* in order to get a final access decision. Please note that the access decision is not in a form of static *permit* or *deny*, rather it is in a set of access constraints or conditions that explicitly authorize which data can be accessed by the user. The whole mechanisms of the rule and policy combining, and then evaluate them with the access *request* are explained in great details in the Subsection 4.3.

4.2.8 Request

An access *request* must consists of a *profile* P, and optionally a set of EPCIS event types ET and a set of *conditions* C. Both ET and C are optional fields in a *request* due to the nature of the EPCIS query specification. For a *request* to be evaluated with a certain policy, the *request's* profile P_{REQ} should match with the *policy's* profile P_{pol}. If a single or set of ET(s) are specified in the *request*, then only rules that matched with the specified ETs are applicable, otherwise all rules in the policy are applicable for the *request*.

4.3 Access Rules Evaluation

Access rules evaluation is at the heart of the whole model in order to provide a fine-grained access control. The results of this evaluation is a final access decision in a form of a set of explicit access constraints.

There are two important steps in the access rules evaluation. First, all applicable rules are combined by using union operation. Second, the combined rules' condition is then evaluated against the condition presented by access request using an intersection operation. The general expression of this evaluation is depicted as follows:

$$C^{REQ} \cap (C_1 \cup \cdots \cup C_p) \tag{3}$$

Algorithm 1 access rules evaluation procedures

```
Procedure RulesEvaluation(C^REQ, SC)
    for all AE^s_ij in C_i in SC do
        for all C_i in SC do
            AE^s'_j = UnionAll(AE^s_xj)
        end for
        Combine Rules = Conjunction(AE^s'_j)
    end for
    Condition = ""
    for all AE^s'_j in Combine Rules do
        if Condition ≠ "" then
            Condition += ∧
        end if
        for all AE^r_k in C^REQ do
            if j == k
                Condition += AE^s'_j ∩ AE^r_k
            end if
        end for
        if not Found(k == j) then
            Condition += AE^s_j
        end if
    end for
end procedure
```

where:

- p: number of matched access rules.
- $C_i = (AE^s_{ij} \wedge \cdots \wedge AE^s_{in})$: a set of conditions specified in the i^{th} matched access rule.
- $C^{REQ} = \{AE^r_k, \cdots, AE^r_m\}$: a set of EPCIS attributes conditions specified in user request.

The access rules evaluation as expressed in (3) is obtained through some steps illustrated in the pseudo-code 1.

The procedure starts by combining all conflict-free conditions of the applicable rules using union set operation. Once a combined rule is obtained, the access rule evaluation is started by grouping the similar event type attribute, i.e. AE pair, from between the C^{REQ} and the $CombineRules$, and then a set intersection operation is applied, i.e. $AE^{s'}_j \wedge AE^r_k$, where $j = k$. In case of the pair of $AE^{s'}_j$ is not specified in the C^{REQ}, then only the condition as specified in AE^s_j will be applied. At last, the final expression of access rule evaluation as depicted in (3) is obtained.

5 System Implementation

Two important components in this system implementation are the Policy repository as well as evaluation, and Access Policy enforcement components. In addition, the overall implementation is designed to be modular and complies with the EPCIS query interface specification. For this purpose, the proposed access control is implemented through web service interface and be part of the Aspire RFID middleware [2] module. The Aspire RFID middleware itself is one of the open source implementation of RFID middleware based on EPCglobal specifications.

5.1 System Architecture

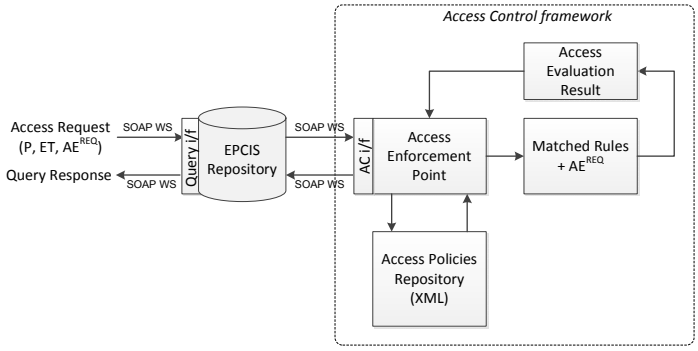

Figure 2 System architecture of the implemented access control model

The system architecture of the implemented access control model is depicted in Fig. 2. The Access Control Framework is communicating with the EPCIS repository through web service interface. Once the incoming query or access request is received, the access policy enforcement point will query the Access Policies Repository to find the matching policies with the request according to the procedure explained in Section 4 earlier. Once a set of matching policies or rules is found, it will be evaluated against the access request. Finally, the result of the access control evaluation will be sent back to EPCIS repository through the Access Enforcement Point via the web interface.

The access policy enforcement is an important component in a policy based access control implementation after an access decision has been taken. In our implementation, the access control policy enforcement is implemented in a form of a modified *Query Params*, i.e. the query parameters defined in the EPCIS specification. The advantage of this method is that the implementation

complies fully with the EPCIS query interface specification while stay modular without a necessity to greatly modify the EPCIS server implementation. Moreover, it does not depend of the actual implementation of the database technology for the EPCIS repository. However, the main drawback is that this method does not fully leverage the proposed access control model due to the nature of *Query Params* description in the EPCIS specification where the parameter that is not explicitly mentioned in the *Query Params* will be included in the query results. Nevertheless, it still allows us to perform an evaluation over most of the important functionalities of the proposed access control model. Additionally, if the access decision return an empty result, i.e. no matching policy is found in the repository, the query results of the EPCIS query interface implementation will return a *Security Execption*.

The policy repository and evaluation component implementation uses XML and JAXB technologies. The proposed access control policy specification as explained in Section 4 is translated into a set of XML specification and is practically stored as an XML file. Thanks to the JAXB technology, the access control policies can be created as an XML file and the stored access control policies can be translated into Java Objects to be further evaluated.

The usage of XML for describing policies in the proposed access control model is motivated due to the fact that the XML has been well accepted and widely used in the heterogeneous IT enterprise environment across different platforms. Among others, XACML [13] is the most famous XML based access control system. Particularly in our system implementation, the XML is used it model the data structure of the proposed access policy elements as described in Section 4.

6 Evaluation Results and Discussion

A series of experiments or evaluation have been carried out based on our implementation in order to measure the performance of the proposed access control model. The evaluation encompasses two main purposes. First, it is meant to validate the functionality of the proposed access control model. Second, it aims at measuring the performance in terms of delay time. Delay is an important parameter to be evaluated due to the fact that time is a critical performance metric in dealing with big data processing.

6.1 Evaluation Procedures

For the testing purpose, one policy for a particular user profile is prepared. The policy consists of different rules with several conditions for different EPCIS

event types. It is important to note that since generating EPCIS events already requires some complex procedures, we focuses on generating only one type of business event in the test scenario. Hence, the generated EPCIS events are always having fixed event attributes values (VE_ij), except for the *event Time*, *record Time* and EPC ID. As a result, the final access control enforcement mainly depends on conditions related to *event Time* and EPC ID parameters or attributes in the practical experiments.

Regarding the first objective of this evaluation, it is shown that the access control works as it should be. The query results only returns the EPCIS event data that fulfil the conditions set in the policy and the final access decision. On the other hand, a *Security Exception* will be returned if there is no matching policy found in the access policy repository.

Concerning the second objective of the evaluation, the delay performance of the proposed access control model is compared against the EPCIS system without any access control applied. For this purpose, we have tested our proposed access control model on a system with Intel Core i7 2.80 GHz CPU, 8 GB RAM, running Windows 7, and Java 6.31. In addition, an Apache Tomcat 6.35 server is used to deploy all the Aspire RFID Middleware modules as well as our access control module, and a MySQL 5.2 Server is used as the EPCIS events data repository. Although the measurement results strongly dependant on the implementation, nevertheless we can expect to draw some qualitative conclusion out of the results and further improve the system implementation if necessary.

Two measurement scenarios are carried out to observe the delay performance and behaviour of the implemented proposed access control model. The delay is defined as the time consumed between the query request being sent and received by the EPCIS query client, i.e. $T_{received} - T_{sent}$. For each scenario, the same measurement point is done repeatedly for 700 times. For the purpose of explanation in the rest of this paper, we will refer the case when the access control is used as the *first case* and the *second case* refers to the case when access control is not used.

6.2 The Impact of Varying the Number of Read Tags

The first scenario aims to observe the delay time behaviour when the number of read tags in each event stored in the EPCIS repository is varied. In this scenario, the number of EPCIS events are fixed to three events at the EPCIS repository. It should be noted that the query results of the *first case* always returns one EPCIS events, while the *second case* always returns all three events since no access control is applied.

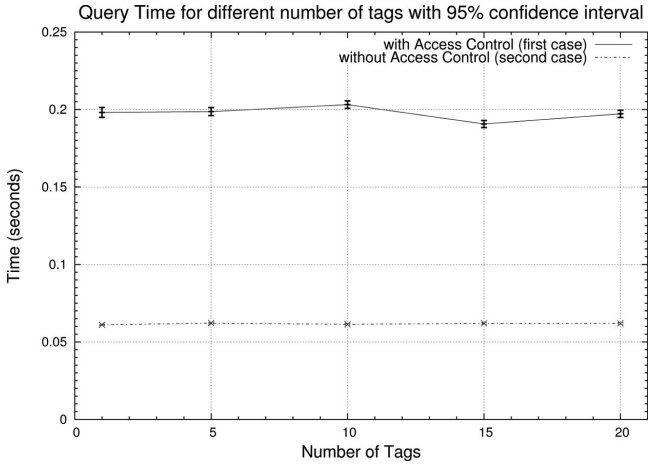

Figure 3 Query delay time for different number of tags with 95% confidence interval

There are two important findings that can be derived based on the results of this scenario which is shown in Fig. 3. First, the average delay of the *first case* is three times higher than that of the *second case*. This finding is quite expected because it involves extensive XML processing, e.g. creating and parsing of SOAP message and parsing the access policy, which is time consuming. The delay performance could be improved in general through a more tightly coupled implementation with the expense of sacrificing the flexibility and modularity of such an open system. Second, the EPCIS query delay time for both cases is relatively constant regardless of the number of tags read in each EPCIS event. Although some small variation is shown in the *first case*, the confidence interval margin is rather high and it could be due to the inconsistency of web service invocation of access control service. Thus it is quite safe to neglect the small variation. Nevertheless, the second finding is quite interesting because the amount of data queried from the database would contribute to the query time delay.

6.3 The Impact of the Number of the EPCIS Events Data in the Repository

Based on the second finding in the first scenario, we would like to check the impact of varying the number of EPCIS events in the repository to the query time delay. Knowing that the number of tags does not change the delay behaviour, the number of tag included in the EPCIS events stored within the

repository is fixed to only one tag in this scenario. Similar to the first scenario, the *first case* always returns one EPCIS events while the *second case* always returns all the EPCIS events available in the repository.

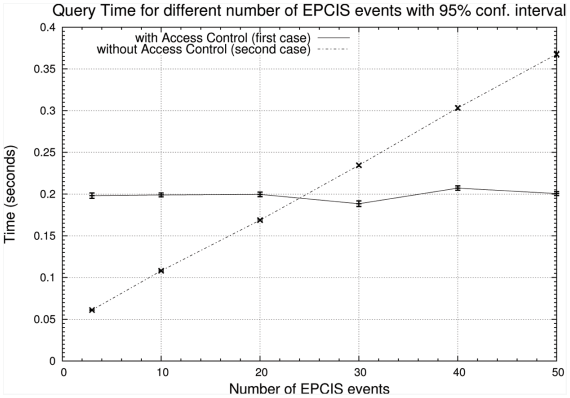

Figure 4 Query delay time for different number EPCIS events in the repository with 95% confidence interval

The results of the second scenario depicted in Fig. 4 shows that the number of EPCIS events data queried from the repository does impact the query time delay linearly. Obviously, the query delay time of the *first case* is relatively constant since it always return one EPCIS event as a result of the access control mechanism. Based on this finding, it can be concluded that both cases would achieve the same query delay time when the ratio of the EPCIS events data returned between the *first* and *second* case is around $1 : 25$. Consequently, the delay gap between both cases as shown in Fig. 3 would be bigger if the result of access control evaluation would return more than one EPCIS events data. This ratio could be improved through a more efficient implementation.

7 Discussions

Earlier in this paper, we have mentioned several requirements of access control model in an inter-enterprise RFID system, and how the proposed model fulfilled those requirements. In this section, we will present several important features of a secure system – in particular an access control for big data system – that are constituted in the proposed access control model. Furthermore, some qualitative comparisons with other access control models will also be given.

- **High-granularity:** The proposed access control model provides high-granularity since it does not allows the access only based on *permit* or *deny* action, but based on some specific types of data or context information. The level of granularity can be defined in the access policy by the system or security administrator of the system.
- **High privacy:** The high-granularity of access control policy provides high level of privacy to the business activities related information stored in the EPCIS repository.
- **Trust:** The access policy is applied to each user automatically based on some pre-defined user profiles, and a user is assigned to a particular user-profile based on trust relationship and contextual information.
- **Flexibility:** The proposed access control policy model offers high flexibility which is favourable in a big data system. It allows relative time definition instead of fixed time, EPC ID definition based on the EPC pattern, and automatic user assignment to user-profile through trust information.
- **Inter-operability:** The proposed access control model complies with the EPCIS Specification which is an open standard. The proposed system is also implemented as a web service which is highly inter-operable, i.e. independent of any specific programming language or server implementation.

In comparison with other existing access control, the proposed access control model is better as compared to them in the following aspects:

- *Trust-based X-GTRBAC[5]*: Our proposed access control model allows a way to incorporate trust in assigning user-profile to users which is quite similar to the approach presented in [15]. However, [15] does not fit the requirement of providing high-granularity access to data, particularly in an inter-enterprise RFID system.
- *AAL [9]*: A quite similar approach of a fine grained access enforcement specially designed for the EPCIS events data that is proposed in our access control model, was also introduced in [9]. However, automatic way of assigning some access policy to a user was not considered in [9], which gives a lot of burden to the system administrator, thus unrealistic for the real system. Moreover, its SQL query rewriting method for the access enforcement, does not inter-operable with the EPCIS implementation that is not based on the RDBMS.

8 Conclusion

Access control management in an inter-enterprise RFID system is a great challenge, since such system allows the tracking and monitoring of large numbers of things, i.e. a path towards realizing the IoT vision. The most challenging access control problems in such a system are in providing high-granularity access of RFID events data, known as EPCIS events, with flexibility and efficient access policy management over a very dynamic and huge amounts of EPCIS events data. A novel access control model to address these problems has been proposed in this paper along with complete definition of access control model and evaluation through the system implementation of the model. The findings in the evaluation show that the proposed access control model is consistent and could achieve less query time delay than the inter-enterprise RFID system without access control if the ratio of the returned EPCIS events data is less than $1 : 25$.

The proposed access control model relies on the trusted third parties entity to obtain the user profile, but the mechanism on dealing with such trust management has not been addressed yet. Incorporating trust into the access control model, i.e. through PKI CA, is one possibility of the future work. Another direction targeted in the near future is to improve the current system level implementation, especially in addressing the limitation described in Section 5.

Reference

[1] Anggorojati, P. N. Mahalle, N. R. Prasad, and R. Prasad. Secure access control and authority delegation based on capability and context awareness for federated iot. In Fabrice Theoleyre and Ai-Chun Pang, editors, *Internet of Things and M2M Communications*. River Publisher, 2013.

[2] ASPIRE. http://wiki.aspire.ow2.org.

[3] L. Atzori, A. Iera, and G. Morabito. The internet of things: A survey. *Computer Networks*, 54(15):2787–2805, 2010.

[4] E. Bertino, P. A. Bonatti, and E. Ferrari. Trbac: A temporal role-based access control model. *ACM Trans. Inf. Syst. Secur.,* 4(3):191–233, August 2001.

[5] R. Bhatti, E. Bertino, and A. Ghafoor. A trust-based context-aware access control model for web-services. *In Web Services, 2004. Proceedings. IEEE International Conference on*, pages 184–191, july 2004.

[6] R. Bhatti, A. Ghafoor, E. Bertino, and J. B. D. Joshi. X-gtrbac: an xml-based policy specification framework and architecture for enterprise-wide access control. *ACM Trans. Inf. Syst. Secur.,* 8(2): 187–227, May 2005.
[7] EPCglobal. Epc information services (epcis) version 1.0.1 specification. September 2007.
[8] EPCglobal. Gs1 epc tag data standard 1.6 - ratified standard. September 2011.
[9] E. Grummt and M. Muller. Fine-grained access control for epc information services. In *Proceedings of the 1st International Conference on The Internet of Things,* IOT'08, pages 35–49, Berlin, Heidelberg, 2008. Springer-Verlag.
[10] J. B. D. Joshi, E. Bertino, U. Latif, and A. Ghafoor. Ageneralized temporal role-based access control model. Knowledge and Data Engineering, *IEEE Transactions on*, 17(1):4–23, jan. 2005.
[11] T. Karygiannis, B. Eydt, G. Barber, Lynn Bunn, and T. Phillips. Guidelines for securing radio frequency identification (rfid) systems - recommendations of the national institute of standards and technology. *NIST Special Publication,* April 2007.
[12] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-based access control models. *Computer*, 29(2):38–47, feb 1996.
[13] XACML. https://www.oasis-open.org/standards#xacmlv2.0

Acronyms

6LoWPAN IPv6 over Low-power Wireless Personal Area Network
AAL Auto-ID Authorization Language
ACL Access Control List
ACS Access Control Servers
AVISPA Automated Validation of Internet Security Protocols and Applications
BEG Business Event Generator
CASM Context Aware Security Manager
CA Certificate Authority
CA Certification Authority
CCAAC Capability-based Context Aware Access Control
CWAC Context aWare Access Control
DoS Denial of Service
EPCIS Electronic Product Code Information Services
EPC Electronic Product Code
FM Federation Manager
GTRBAC General Temporal RBAC
GW Gateway
ICAP Identity based Capability
IdP Identity Provider
IoT-DS IoT Directory Service
IoT-FM IoT Federation Manager
IoT Internet of Things
ITU International Telecommunication Union
JAXB Java Architecture for XML Binding
MAC Message Authentication Code
MAGNET My Adaptive Global NETwork
NFC Near Field Communication
OS Object Service
PE Policy Engine
PKI Public Key Infrastructure
PN-F Personal Network Federation
PNDS Personal Network Directory Service
PNDS PN Directory Service
PN Personal Network
PTD Personal Trusted Device
RBAC Role Based Access Control
RDBMS Relational Data Base Management System
RFID Radio Frequency Identification

RF Radio Frequency
SDP Security Decision Point
SecaaS Security as a Service
SOAP Simple Object Access Protocol
SQL Structured Query Language
TDS Tag Data Standard
TRBAC Temporal RBAC
VID Virtual Identity
WPAN Wireless Personal Area Network
WSN Wireless Sensor Network
XACML Extensible Access Control Markup Language
XML eXtensible Markup Language

Biographies

Bayu Anggorojati received his B.E. degree in Electrical Engineering in 2005 from Institut Teknologi Bandung, Bandung, Indonesia. He received his MSc in Mobile Communication in 2007 from Aalborg University, Aalborg, Denmark. He joined Wireless Security and Sensor Network group within Network and Security Section (now CTIF Section) in the Electronic System of Aalborg University as a research assistant from 2007 till now. He has been involved in a number of EU-funded R&D projects, including FP7 CP Betaas for M2M & Cloud, FP7 CIP-PSP LIFE 2.0, FP7 IP ISISEMDICt for Demetia, and FP7 IP ASPIRE RFID and Middleware. He is currently pursuing his PhD degree at CTIF Section in Electronic System Department of Aalborg University, Denmark. His research interests include Radio Resource Management in OFDMA based system; Access Control, Authentication, and Key Management in the IoT/M2M and Cloud system.

Neeli Prasad is leading a global team of 20+ researchers across multiple technical areas and projects in Japan, India, throughout Europe and USA. She has a Master of Science degree from Delft University, Netherlands and a PhD degree in electrical and electronic engineering from University of Rome Tor Vergata, Italy. She has been involved in projects totaling more than $120 million – many of which she has been the principal investigator. Her notable accomplishments include enhancing the technology of multinational players including Cisco, HUAWEI, NIKSUN, Nokia-Siemens and NICT as well as defining the reference framework for Future Internet Assembly and being one of the early key contributors to Internet of Things. She is also an advisor to the European Commission and expert member of governmental working groups and cross-continental forums. Previously, she has served as chief architect on large-scale projects from both the network operator and vendor side looking across the entire product and solution portfolio covering wireless, mobility, security, Internet of Things, Machine-to-Machine, eHealth, smart cities and cloud technologies. She has more than 250 publications and published two of the first books on WLAN. She is an IEEE senior member and an IEEE Communications Society Distinguished Lecturer.

Ramjee Prasad is currently the Director of the Center for TeleInfrastruktur (CTIF) at Aalborg University, Denmark and Professor, Wireless Information Multimedia Communication Chair. Ramjee Prasad is the Founding Chairman

of the Global ICT Standardisation Forum for India (GISFI: www.gisfi.org) established in 2009. GISFI has the purpose of increasing of the collaboration between European, Indian, Japanese, North-American and other worldwide standardization activities in the area of Information and Communication Technology (ICT) and related application areas. He was the Founding Chairman of the HERMES Partnership - a network of leading independent European research centres established in 1997, of which he is now the Honorary Chair. He is the founding editor-in-chief of the Springer International Journal on Wireless Personal Communications. He is a member of the editorial board of other renowned international journals including those of River Publishers. Ramjee Prasad is a member of the Steering, Advisory, and Technical Program committees of many renowned annual international conferences including Wireless Personal Multimedia Communications Symposium (WPMC) and Wireless VITAE. He is a Fellow of the Institute of Electrical and Electronic Engineers (IEEE), USA, the Institution of Electronics and Telecommunications Engineers (IETE), India, the Institution of Engineering and Technology (IET), UK, and a member of the Netherlands Electronics and Radio Society (NERG), and the Danish Engineering Society (IDA). He is also a Knight ("Ridder") of the Order of Dannebrog (2010), a distinguishment awarded by the Queen of Denmark.