A Trusted Naming Service for Things of Internet

IMRAN ALI JOKHIO*, SANA HOOR JOKHIO**, AND FAISAL KARIMSHAIKH***

RECEIVED ON 12.12.2011 ACPTED ON 15.03.2012

ABSTRACT

The growing popularity of RFID (Radio Frequency Identification) systems cannot be overlooked due to their wide range of application areas. However, some RFID applications face security and privacy threats due to the exposure of tags over wider distances. There are security methods but these can lack security and scalability when authenticating RF tags in MORIS (Multi Organizational RFID System). RFID systems operating within a single domain have only security and privacy issues due to the security method on tags, but in a multi-organizational system there are even more security and privacy issues in the system architecture rather than just in the security method on tag side. In this paper, we have designed a broker service to EPC (Electronic Product Code) system architecture for secure and confidential data management of the RFID data events. Secure and confidential authentication broker service is an extension to the EPC network architecture for a MORIS. The central authentication broker service ensures to avoid illicit access to the confidential RFID tag's data such as EPC code, whilst managing a large-scale authentication data to ensure secure and deterministic response to the authentic entities of the system.

Key Words: **Things of Internet, RFID Security, ToI Management, EPC System Architecture.**

1. INTRODUCTION

ONS (Object Naming Service) [1], a component of EPC system architecture, is built on the EPC code to direct a tag to its EPCIS (EPC Information System) [2-3]. In RFID system where the EPC code of tags is confidential, a tag does not send it. In this case in order to retrieve the EPC code from a tag, a reader and the tag need mutual authentication. Prior to authentication a reader device does not know EPC code of a tag, so as where this authentication information lies in the back-end system called EPCIS. A security method full filling all the requirements of security and privacy has a 'chalk and cheese' response in each authentication session. Therefore discovering an EPCIS service in the EPC system is a challenge. The research community is well aware of the security and privacy issues [4-7] in an RFID system. Consequently a number of security methods for authentication are proposed. For an extensive review of the proposed security methods, [8-11] can be referred. These are mainly based on HB (Hopper and Blum) [12], one-way keyed hash function. But, to the best of our knowledge the issue of authentication service discovery in the back system architecture whilst an EPC is not known, is not well addressed.

Rest of the paper is organized as: Section 2 explains EPC system architecture and its major components i.e. tag data

* Assistant Professor, Department of Software Engineering, Mehran University of Engineering & Technology, Jamshoro.
** Assistant Professor, Department of Computer Systems Engineering, Mehran University of Engineering & Technology, Jamshoro.
*** Assistant Professor, Department of Telecommunicatin Engineering, Mehran University of Engineering & Technology, Jamshoro.

structure; ONS service structure and the EPC information system. Authentication problem in a MORIS is explained and related proofs are also elaborated in Section 3. Section 4 discusses the need of a central trusted naming service and the proposed authentication broker service architecture, its integration with existing EPC system is explained. Various processes of the trusted naming service are also explained Section 4. In order to validate and verify the proposed system architecture various experiments are conducted and reported in Section 5. Finally this research work concludes with possible future work in Section 6.

2. SYSTEM ARCHITECTURE FRAMEWORK

In order to envisage a compatibility mismatch of RFID architecture caused by the role of the ONS service [1] and the EPC code's transmission over an unsecured channel, the architecture services and data flow across the system architecture is discussed in this section. An overview of the EPC network architecture framework and its major components are shown in Fig. 1.

2.1 EPC Tags

To integrate and use RFID technology in a SCM (Supply Chain Management) system or any other potential application, Auto ID Labs and EPCglobal came up with a naming scheme [1] to uniquely To integrate and use RFID technology in a SCM or any other potential application, Auto ID Labs and EPCglobalcame up with a naming scheme [1] to uniquely identify physical objects called the EPC [13]. When the idea of EPC was applied to a number of other application areas such as manufacturing, electronics, healthcare and transportation, more was added than the item level identification of the physical object, such as the unique identification of batch products, component types, and physical configurations. For the sake of greater applicability of the EPC system architecture a number of standards were specified for the EPC code as shown in Table 1.

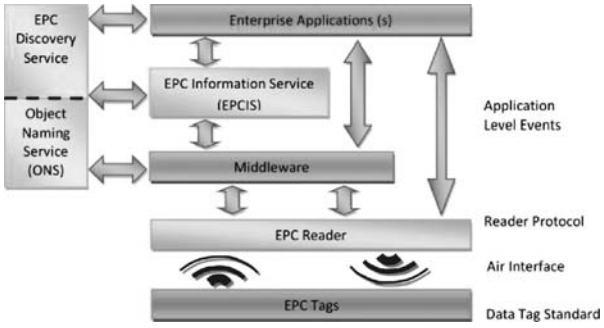

FIG. 1. RFID SYSTEM OVERVIEW

TABLE 1. EPC TAG DATA STANDARD

EPC Scheme	Tag Encoding	Corresponding GSI Key	Typical Use
SGTIN (Serialized Global Trade Identification)	SGTIN-196, SGTIN-198	GTIN (with added serial number)	Trade item
SSCC (Serial Shipping Container Code)	SSCC-96	SSCC	Pallet load or other logistics unit load
SGLN (Serialized Global Location Number)	SGLN-96, SGLN-195	GLN (with or without additional extension)	Location
GRAI (Global Returnable Asset Identifier)	GRAI-96, GRAI-170	GRAI (serial number mandatory)	Returnable/reusable asset
GIAI (Global Individual Asset Identifier)	GIAI-96, GIAI-202	GIAI	Fixed asset
GDTI (Global Document Type Identifier)	GDTI-96, GDTI-113	GDTI (serial number mandatory)	Document
GSRN (Global Service Relation Number)	GSRN-96	GSRN	Service relation (e.g., loyalty card)
GID (General Identifier)	GID-96	[none]	Unspecified

2.2 Object Naming Service

The EPC system architecture has an ONS. which functions as a look up service, converting an EPC to its EPCIS service Uniform Resource Locator, in the same way that a domain name is converted to an IP address using DNS. The ONS structure and architecture is very similar to DNS, it is actually just an application of DNS [1].

Conceptually, an ONS service is a single look up query service. When an application such as a reader, middleware etc. wants to locate an EPCIS service of an EPC code it sends a query to its local DNS resolver service, which is typically a part of a computer operating system. The DNS resolver service is responsible for carrying out the query process and returning a result. From an application's point of view this appears to be a single operation, but the ONS performs a multi-step look up service as shown in Fig. 2. When an ONS service built on top of a DNS service receives a query, it consults a Root ONS service that is controlled by EPCglobal. The Root ONS service identifies a local ONS service of an EPC code's EPC Manager organization. The application or reader middleware completes the lookup to locate an EPC code's EPCIS service by sending a request to the identified local ONS service [14].

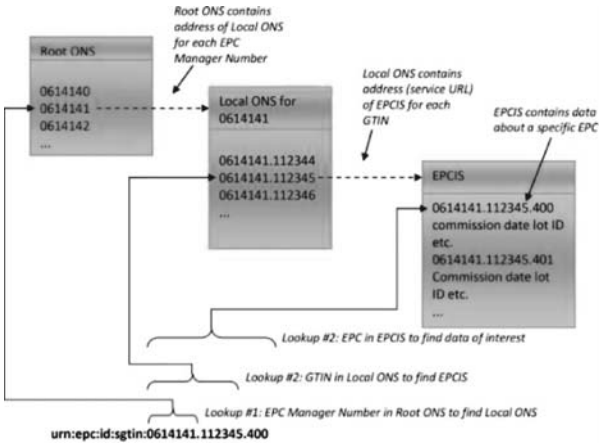

FIG. 2. ONS LOOK UP A MULTI STEP LOOKUP QUERY SERVICE

2.3 Electronic Product Code Information System

The EPCIS is a repository, which stores historical RFID data events. The information contained in EPCIS is shared within and across organizations making it more useful for supply chain or other business contexts. EPCIS has two interfaces: EPCIS Capturing Interface, and the EPC Accessing Interface. The former interface transforms the received RFID data events into EPC-related business events for a business process. It may also transform multiple RFID events into a single event if required for the filtration of data. The later interface is useful for an application, which is implementing a business process related to EPC events data. While sharing RFID events, data and information, appropriate access controls may also be put into the practice at the business process level. This is beyond the scope and it is not investigated in this paper. Beside these business process level-accessing interfaces, there is a tag's authentication service in an EPCIS service to validate and verify a claimed ID of tags with a tag security method. The EPC system architecture recommends an authentication scheme or security method between a reader and tag. An authorized reader will have access to EPCIS that contains information for the authentication of tag. This is to ensure that a tag may not send its data to any reader that is not a trusted reader, and a fake tag may also try to be a true source of information, which needs to be verified.

3. AUTHENTICATION PROCESS IN A MORIS

In order to explain the authentication problem and respective authentication delay overhead in EPC system architecture for MORIS when a security method of a tag does not send an EPC code or any other fixed identifier, a system model of MORIS is defined and described in this section.

3.1 MORIS System Model

Let *S* be a MORIS with r_v organizations or EPC managers, and m reader devices $R_j \forall_j=1,2,3,.....m$. Each organization or EPC manager has an EPCIS service denoted as $E_i \forall_i=1,2,3,.....n$, and each E_i have K_i tags $T_{Eir} \forall_r=1,2,3,.....K_i$. A security model employed in tags is *Sec(a,x,b)* and it maintains the confidentiality of a tag ID *x*, where *a* and *b* are l-bit strings.

3.2 Tag Security Model

A generic light-weight cryptographic security method for tags $T_{Eir} \forall_r=1,2,3,.....K_i$ in a MORIS S is described along with its properties, definitions and assumptions.

Definition 1: A function *y=f(x)* is said to be a One Way Function, for a given *y*, computationally it is very difficult or impossible to find *x'* or x such that $y \Rightarrow f(x)=f|(x')$ in time *t*.

Using the Definition 1 of a one-way hash function following definitions can be made for generic security method *Sec(a,x,b)*.

Definition 2: Computationally its very difficult or impossible to drive an x from *Sec(a,x,b)* for given *a* and *b*.

Definition 3: In an authentication session *S*, $R_s=Sec(a,x,b)$ is unique for *a* is generated by a tag, *b* is generated by a reader, and both *a* and *b* are l-bit PRNG random numbers i.e. $s:R \Leftarrow Sec(a,x,b) \forall a,b \in N$. *N* is a set of PRNG random numbers.

Therefore the following lemmata can be made whilst considering the above definitions.

Lemma-1: As *Sec* is a cryptographic function and it can be deduced from Definition 2 that *R* cannot be predicted by an attacker for given *a* and *b* when *x* is unknown.

Lemma 2: To find *x* during an authentication process *s* for given a_s and b_s, a blind search has to be carried out on each tag access as:

$Sec(a_s,x,b_s) \Rightarrow s \Leftarrow Sec(a_s,x_i,b_s) \forall x_i=x_1,x_2,x_3,.....x_n \in X$. *X* is a set of secret tag ID for tags in the system *S*.

3.3 Authentication

When a security method *Sec(a,x,b)* is implemented on a tag T_{Eir}, above definitions and lemmata suggest that a central authentication with a trusted third party is needed. There is no other way of getting authentication of a tag in a MORIS without knowing an EPC code, and a tag to protect the security and privacy requirements may not transmit the EPC. However a simple way of getting authentication of a tag can be based on a flooding routing idea i.e. a flooding authentication request to all EPCIS services E_i of the system *S*. In this case let us consider that an authentication service at an EPCIS service *Ei* takes $P=p(T_{Eir})$ time units to process a request.

3.4 Proof of Correctness

Lemmata 1 and 2 suggest that while searching or processing an authentication reques to fatag in the EPCIS so for the system *S*, an average authentication process time across all EPCIS services for a tag T_{Eir} isa *S*:

$$P = \frac{\sum_{i=1}^{n} \sum_{r=1}^{k_i} p\left(T_{E_{ir}}\right)}{\sum_{i=1}^{n} k_i} \quad (1)$$

But a tag T_{Eir} is held and owned by only one of the EPCIS E_i, the processing across theEPCIS services is redundant as given Equation (2):

$$\sum_{i=1}^{n} \sum_{r=1}^{k_i=1} p\left(T_{E_{ir}}\right) - \sum_{r=1}^{k_i=1} p\left(T_{E_{ir}}\right) \quad (2)$$

For each tag access or authentication request, each E_i will have a redundant processing overhead as given in the Equation (3).

$$\sum_{r=1}^{k_i=1} p\left(T_{E_{ir}}\right) \quad (3)$$

Besides this redundancy across all the E_i, the system S will send the tag's authentication requesting information to *n-1* EPCISs that may result in many security vulnerabilities. Sending a tag T_{Eir} for an authentication request to each Ei have a possibility of leakage of useful information *n-1* times that may be treated as business important data and may also be considered as personal privacy invasion.

4. TRUSTED NAMING SERVICE

A federated authentication system such as Liberty Alliance [15], Shibboleth [16] and WS-Federation [17] etc. do not provide a potential solution to address the security and privacy issues in the RFID systems. These federated authentication systems need an IdP information. This IdP information is confidential information, leakage of which may cause high security risks. Hence, the existing federated authentication systems cannot be extended. Similarly the ONS service cannot be used as a naming service in RFID system where EPC code of the tags is confidential and an authentication process is required before revealing the EPC it. Therefore a trusted naming service is needed.

4.1 Authentication Brokering

The trusted naming service may serve as an authentication broker service for RFID tags. Every organization in a MORIS may have an authentication service, and the reader devices cannot access/locate the right authentication service for the tags authentication. The authentication and discovery service information such as EPC code and security method parameters are confidential, and the organizations may not share this information without a trust relationship with an entity. Therefore there is a need of central authentication brokering in the EPC system architecture with an appropriate trust relationship. A central ABS (Authentication Brokering Service) may service as an authentication and discovery service in the EPC system. All the organizations of a MORIS may send the authentication information such as tag security method parameters, keys, policy etc. The authentication brokering service may authenticate the tags and send an authentication session token to the EPCIS service. This paper proposes a central TNS (Trusted Naming Service) as an authentication broker for the EPC system. The proposed TNS service may have the role of the ONS service as defined and described in the EPC network. In addition to it, there must be an authentication and authorization process for the tags and reader before revealing the EPC code of the tags.

4.2 Proposed TNS Architecture

The proposed design of an authentication broker service is an enhancement of the EPC system architecture as shown in Fig. 3. It can be deployed as a web service among the other components of the EPC network architecture to provide enhanced security, privacy and confidentiality.

In a central or federated authentication system there is always an initial registration process so that a business partner may register with a trusted central third party. After registration, it may define and describe its users or tags and trusted readers. The definition and description of a tag may contain the EPC code, the security method

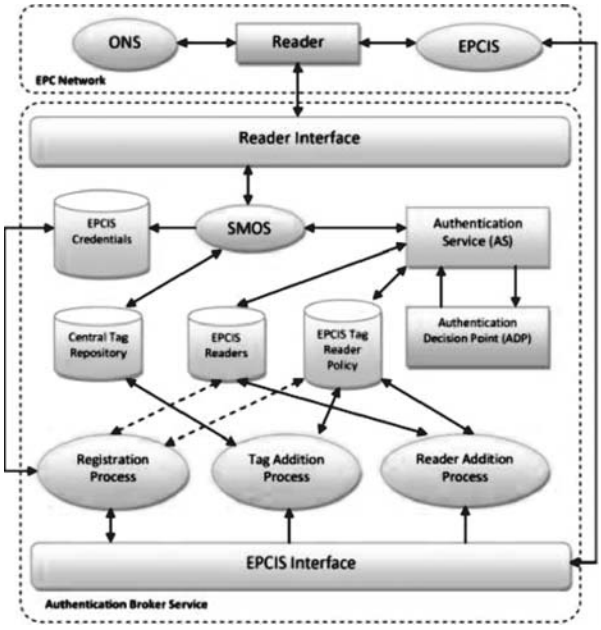

FIG. 3. ABS (TNS) BASED EPC SYSTEM ARCHITECTURE

of a tag, a policy profile and if necessary the trusted readers may also contain a policy profile etc. Once an EPC manager or EPCIS service is registered and has defined and described its tag's security methods, policy and trusted readers, it may add the tags to ABS before deploying them in an application. In order to register, add, define and describe the tags and readers, an EPC manager needs a number of interfaces to an authentication broker service and it has to be integrated with the existing EPC system services. The integration and enhancement of the EPC network with the authentication broker service does not affect the tag-reader communication; it is as per a security method employed in between these two. Once the registration and initialization is done and tags are deployed for some application, a reader query a tag with a hello message and tag replies with an encrypted message as defined in the tag's security method. The tag's message comprises of an encrypted EPC code with some other parameters used in a security method. The reader device invokes the RI (Reader Interface) of the authentication broker service by sending the tag's message and Reader ID rid. The authentication broker service holds all the registered EPC codes of EPCIS services, so it finds the EPC using the encrypted data it received from a reader.

When the EPC code of a tag is found for a tag's message, the authentication decision is taken as per policy definition and description of the tag and reader. If the tag's message cannot be translated to an EPC code, it means that it is not a valid tag so authentication fails. On successful authentication, ABS checks whether the reader with *rid* is authorized to query about the tag.Upon successful authentication, ABS sends a positive response and the EPC to the reader device. The reader sends it to an ONS service to obtain an EPCIS URI. The ONS will send the URI of an EPCIS to the reader. Finally, the required information/data about the tag can be successfully logged in or pulled out from the EPCIS. For an unsuccessful authentication or authorization, appropriate messages are sent to reader device.

It should be noted that, it is assumed that communication channel between the reader device and ABS is secure. This can easily be done using SSL/TLS. The components and repositories of the ABS are kept secure and communication between them is also secure. This can be done by keeping all the components and repositories in a domain and putting appropriate access control restrictions on them. Figs. 4-5 show the sequence diagram of the TNS based EPC Network. ABS has four processes, i.e. (i) Registration (ii) AddTag (iii) AddReader and (iv) Authentication and Authorization.

4.2.1 Registration Process

An EPCIS has to register with ABS in order to authenticate its tags. During the registration request, an EPCIS sends its URI, username password pair or any other login credentials and an interface, *(EName, c1, intf1)* as shown in Fig. 4. While processing registration request, ABS creates a separate database of readers for each EPCIS and stores the interface along with its username and passwords in a central database. This interface will be used later by ABS while authenticating the tags as depicted in the authentication process in Fig. 5. Upon successful registration, ABS sends back two interfaces with login credentials *(intf2, c2)*, these two interfaces serve the AddTag and AddReader processes. The username, password pair or any other credentials serve the login of an EPCIS to the ABS. These need not necessarily be username and password, it can be a digital certificate assigned to an EPCIS service; digital certificate with username and password; or a MAC/IP based digital certificate etc.

4.2.2 AddTag and AddReader Process

After the registration process, an EPCIS may add its tags and trusted readers. The metadata of the tags is stored in a CTR (Central Tag Repository) but readers are kept in the reader's database of each EPCIS as shown in Fig. 3. While

adding a tag, an EPCIS sends the tag data along with its policy profile if there is any and the login credentials as *(EName, c2, EPC, policy1, Sm)*, where Sm is the description of the security model employed on a tag. Upon this request, the respective ABS database repository is updated.

Similarly, for a reader, an EPCIS sends *rid* along with the policy profile of the reader. The policy profile for every reader need not be distinct. A group of reader devices may have a single policy profile. While describing these tags and readers, ABS also checks the authentication of an EPCIS by their credentials, which were issued during the registration process. Once the tags and trusted readers of an EPCIS are defined and described to ABS, authentication of the tags may take place.

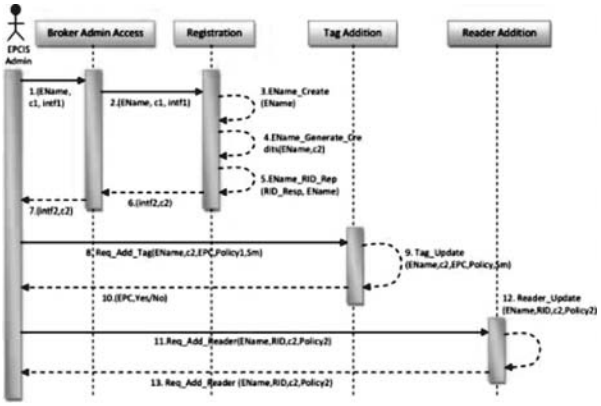

FIG. 4. ABS REGISTRATION AND ADD PROCESS

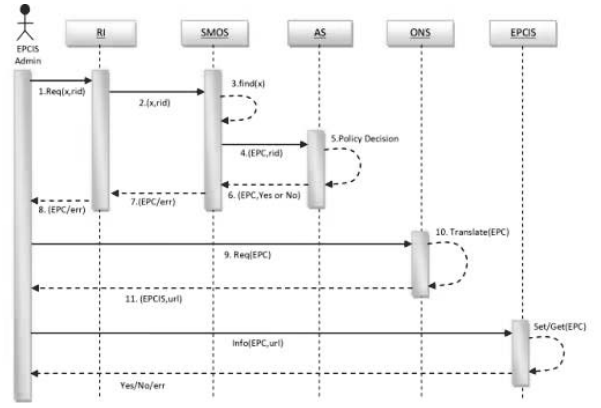

FIG. 5. AUTHENTICATION AND AUTHORIZATION PROCESS IN ABS

4.2.3 Authentication and Authorization Process

The authentication process begins when a reader queries the broker about a tag by accessing the RI interface as shown in Fig. 5. The SMOS (Security Model Operation Store) component of the ABS searches a tag by encrypting and matching the tags. The encryption and matching of a tag is done by accessing the CTR, which holds all the tags of the system. This searching of a tag is done against the received encrypted tag data *x* from the reader. If a tag is found in the CTR by SMOS, it forwards the EPC code and the reader ID to AS (Authentication Service). The AS takes a decision at ADP (Authentication Decision Point) as per the policy definition of the tag and reader. The AS retrieves the policy definition of the tag's x and the reader's *rid* from the respective repositories as shown in Fig. 3 and takes a decision at ADP. If a tag is successfully authenticated and the reader ridis verified to be authorized to query the tag x, the EPC code of the tag is sent to the reader, otherwise appropriate error message is sent to the reader. When the tag and reader are mutually authenticated, the reader device gets the EPC code of the tag. The rest of the process is same as described in the EPC network architecture.

The authentication broker service based EPC network architecture does not disturb the existing architecture of the EPC network. The reader device is still querying the ONS for the EPC to *URI* translation. Similarly, the reader interface is logging or pulling out the EPC events data in the EPCIS service. The new architecture solves the confidentiality breach problem in the EPC network for a MORIS. In the ABS, a separate repository for the policy definitions of the tags and reader of every EPCIS is maintained. This will provide a fine grained and scalable authentication decision service. Furthermore, the SMOS repository is a separate repository to the policy definitions repository. This is also helpful in separating the searching overhead of the encrypted EPC data caused by any security method deployed between a tag and a reader. The searching overhead of a tag may be maintained for a

deterministic time response by using the load balancing techniques in the SMOS. Beside confidentiality and security of the EPC code during an authentication process, the deterministic behavior of authentication is another challenge so the distributed design approach is adapted for tag searching in CTR to maintain a certain level of QoS.

However, a real-time system may have certain requirements, as there can be defined a certain authentication delay, and beyond that any delay might not be acceptable. To assure a certain delay while authenticating tags, the distributed design of the ABS can be deployed. In distributed design the ABS may have a common interface for readers but CTR is distributed. When a reader requests a tag authentication the common interface to the reader sends a request to each distributed broker authentication service. Ultimately, central broker will get a successful authentication from one of the distributed broker services thus providing the required QoS of the authentication.

5. EXPERIMENTAL EVALUATION

For security and feasibility of the proposed authentication system of a practical real- world application, such as SCM, simulations are done for the experimental evaluation. Consequently, a series of experimental simulations has been implemented to assess the processing overhead and scalability of the proposed system architecture. In these experiments tags and reader devices are simulated with java classes. A tag class contains an EPC identifier, a hash function and a PRNG function which generates 96-bit random number a. A reader device class also has a PRNG function and methods to access TNS, ONS and EPCIS interfaces. The EPCglobal specifications of the EPC network architecture and their futuristic vision of the RFID technology recommends the use of web services for the development of a RFID supply chain management system. Therefore, in the experiments, the components of EPC network and TNS are implemented as WS (Web Services). It may also be noted that calculated authentication delays do not include any SOAP or XML processing delays, because the experiments are aimed at calculating authentication process delay rather than end to end delay of the authentication service. Interfaces and operations of these web services are same as described previously. The objectives of the experiments are as follows:

- To verify ***definition-1*** by experimental evaluation.
- To prove and show how the redundant process overhead will affect each EPCIS of a MORIS in the presence ofa security model on tag which may not compromise onthe confidentiality of EPC code.
- To analyze the average authentication delays of ABS architecture by experimental evaluation.

5.1 Brute Force Search Experiment

In the very first experiment, a simulation of an authentication service having a tag's repository of 1000 tags is designed. 100 authentication request queries for different tags are sent to it. Authentication time response for 100 tags is shown Fig. 6. It can be clearly observed that delay is not constant for all tags; some tags get authenticated quicker than the others. It is so because of the unpredictable property (definition-1) of security model in a tag i.e. a tag's reply is different on each access. During the authentication process every tag data is processed until a match is found so authentication delay depends on how many tags are in the repository and how far the tag in question is placed in the repository's starting entry as it is a blind search. In the next experiments, two main scenarios are designed; one is EPC network architecture without ABS and other one is with ABS. In these two experiments a security model is implemented on tag as described in Section 3.1. Therefore during authentication, a reader accesses each and every EPCIS of the system in the first scenario, while in the second scenario authentication is done centrally at ABS. The redundant authentication delay of the two scenario is shown in Figs. 7-8, respectively.

5.2 Authentication Delay Experiments

Another experiment using ABS based EPC network architecture is simulated. In the very first experiment, it is demonstrated that authentication delay for different tags is different, so in this experiment we calculated the average authentication delay during an authentication process. In this experiment we increase the number of tags in the system to evaluate the average authentication delay behavior ABS. Fig. 9 shows the average authentication delay while increasing the total.

6. CONCLUSION

There have been many efforts to design a secure and scalable security model for tag-reader. But the research community is lacking to consider a very important issue, that even if there is a security model on tag and EPC code is transmitted over insecure wireless channel, it can cause security and privacy issues. If the EPC code is not transmitted then current EPC system architecture has no way of finding an authentication service or EPCIS. Therefore, the design of ABS is proposed and its integration with the existing EPC network standards is also presented. With the SABS architecture a tag does not expose an EPC code or any other identifier. The proposed system architecture for authentication data management is not based on a specific security model, so any security model can be used with it. The advantage of having a central authentication is, less network traffic generated by the reader devices,and RFID authentication

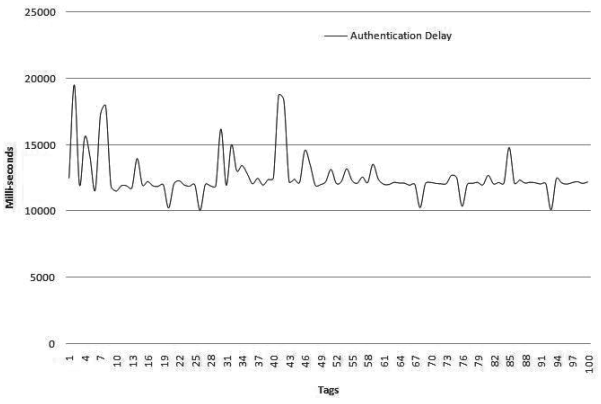

FIG. 6. AUTHENTICATION DELAY OF TAGS.

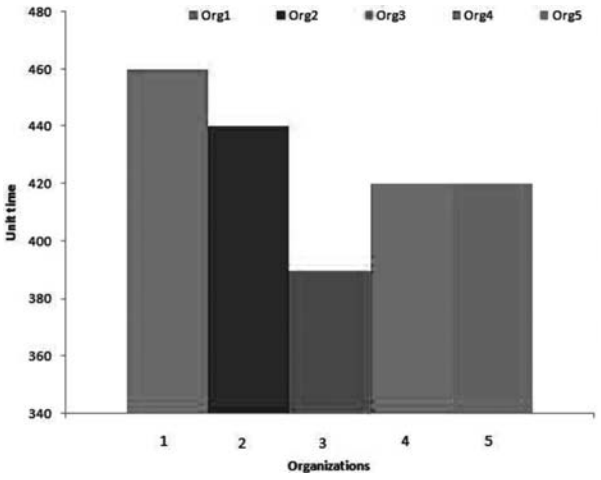

FIG. 7. REDUNDANT AUTHENTICATION DELAY

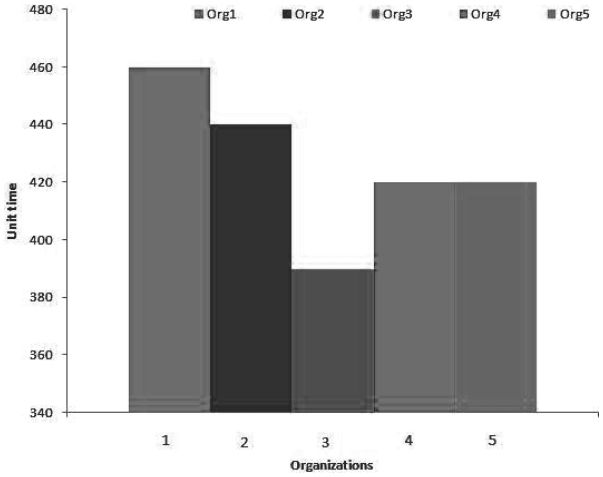

FIG. 8. REDUNDANT AUTHENTICATION DELAY

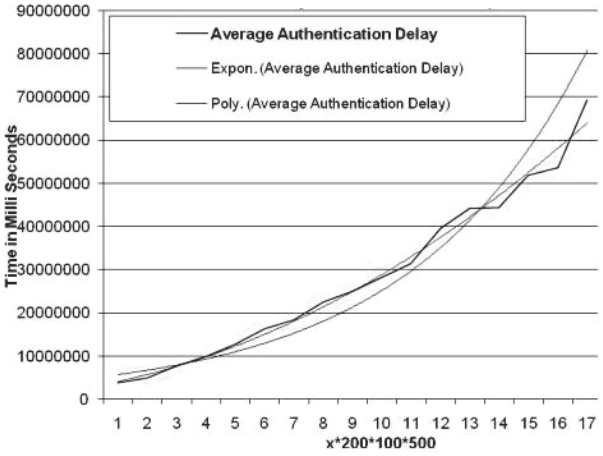

FIG. 9. AVERAGE AUTHENTICATION DELAY IN ABS

data is not compromised at all. It is easier and vital to have a central trusted authentication service rather than trusting all the entities of the system. To address the scalability, ABS architecture is designed in a layered approach. This leads ABS to a loosely coupled system in single domain and each layer can be managed separately to ensure response time of a large deterministic authentication service. Our ongoing research work involves design of a self-adaptive data management layer for deterministic response time of ABS, to ensure a certain level of authentication QoS.

ACKNOWLEDGEMENTS

We are thankful to MUET (Mehran University of Engineering & Technology), Jamshoro, Pakistan, for providing the required facilities and funding to conduct this research. Special thanks to Prof. Jie Xu, who was the Ph.D. Thesis Supervisor of the first author. We also appreciate the dynamic leadership of Prof. Dr. Abdul Qadeer Khan Rajput, Vice-Chancellor, to support research activities in Mehran UET.

REFERENCES

[1] EPCglobal, "Epcglobal Object Name Service (ONS) 1.0.1", Ratified Standard Specification, May, 2008.

[2] Kin, D.W.E., Leong, S., and MunLeng, N.G., "EPC Network Architecture", Auto-ID Center MIT, White Paper, 2005.

[3] Traub, G.A.K., "EPC Network Architecture Framework", Auto-ID Center/EPCglobal, Technology Reprint, July, 2005.

[4] Tsougas, N., "RFID Security and Privacy Concerns", Proceedings of the 5th ACM Workshop on Wireless Security, pp. 31-31, New York, USA, 2006.

[5] Henrici, D., "RFID Security and Privacy: Concepts, Protocols, and Architectures", Lecture Notes in Electrical Engineering, Springer Publishing Company, Incorporated, 2008.

[6] Kerschbaum, F., and Sorniotti, A., "RFID-Based Supply Chain Partner Authentication and Key Agreement", Proceedings of the Second ACM Conference on Wireless Network Security, pp. 41-50, New York, USA, 2009.

[7] Neumann, P.G., "Risks to the Public", SIGSOFT Software Engineering Notes, Volume 32, No. 5, pp. 17-25, 2007.

[8] Koralalage, K.S., and Cheng, J., "A Comparative Study of RFID Solutions for Security and Privacy: Pop vs. Previous Solutions",Proceedings of the International Conference on Information Security and Assurance, IEEE Computer Society, pp. 342-349Washington, DC, USA, 2008.

[9] Lim, T.L., Li, T., and Yeo, S.L., "A Cross-Layer Framework for Privacy Enhancement in RIFD Systems", Pervasive Mobile Computer, Volume 4, No. 6, pp. 889-905, 2008.

[10] Seo, Y., "A Study on Scalable and Untraceable Authentication Protocol of RFID Tags", M.E. Thesis, School of Engineering Information and Communications University, Daejeon, Korea, December, 2007.

[11] Langheinrich, M., "A Survey of RIFD Privacy Approaches", Personal Ubiquitous Computer, Volume 13, No. 6, pp. 413-421, 2009.

[12] Hopper, N., and Blum, M., "A Secure Human-Computer Authentication Scheme", Carnegie Mellon University, CMU-CS-139, Technical Report, 2000.

[13] EPCglobal, "EPC Generation 1 Tag Data Standards", Standard Specifications 1.27, EPCglobal, May, 2005.

[14] EPCglobal, "EPCglobal Tag Data Translation (TDT)", Technical Report 1.4, EPC-Global, June, 2009.

[15] Liberty for Users or Vendors? "Computer Fraud and Security", No. 8, March, 2002.

[16] Needleman, M., "The Shibboleth Authentication/ Authorization System", Serials Review, Volume 30, No. 3, pp. 252-253, 2004.

[17] WS Federation Standard Version 1.1 http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-fed/WS- Federation-V1-1B.pdf, Last accessing date, December, 2011.