Internet of Things: Hotspot-based Discovery Service Architecture with Security Mechanism

Degang Xu[1,2], Zhao Wu[1], Zhongbo Wu[1], Qilin Zhang[1], Leihua Qin[2], Jingli Zhou[2]
(Corresponding author:Degang Xu)

School of Mathematics and Computer Science, Hubei University of Arts and Science[1]
No. 296, Longzhong Road, Xiangyang, 441053, China
School of Computer, Huazhong University of Science and Technology[2]
(Email: dgx.hust@gmail.com)
(Received Mar. 25, 2013; revised and accepted Nov. 6, 2014)

Abstract

In the emerging Internet of Things (IoT), as a means to fulfill item-level lookup, apart from the functional requirements with high performance and robustness, lookup service or discovery service playing a critical role should meet security and privacy requirements. However, existing lookup service and discovery service of IoT mainly rely on a centralized or a chain-style framework, have some drawbacks or bottlenecks to prevent them from being widely adopted, while the issue of locating hotspot resource has received much less attention, as well as the item-level lookup service is still missing. Therefore, we first present a distributed hotspot-based discovery service architecture based on double-Chord-ring for IoT, and then give its framework and some relevant mechanisms. Here we primarily focus on the goals of meeting security and privacy requirements. Additionally, we further discuss and analyze our solution.

Keywords: Internet of Things, Object Name Service, Discovery Service, Object Discovery Service, Security Mechanism

1 Introduction

The Internet of Things (IoT), an emerging global Internet-information architecture, has the purpose of providing an IT-infrastructure facilitating the exchanges of goods and services in global supply chain networks in a secure and reliable manner [3, 31], where lookup service or discovery service plays a critical role. Therefore, as an essential and critical component for a variety of application scenarios of the IoT (specifically, the EPCglobal Network, an industry proposal to build a global information architecture for objects carrying RFID tags with Electronic Product Codes (EPC)), lookup service should take some measures to enhance the security and privacy of the architecture.

Two of the key components of IoT lookup service architecture required to implement track and trace capabilities are the Object Name Service (ONS) and the Discovery Service (DS) envisaged to provide pointers to multiple providers of information across a supply chain not only the manufacturer. In the EPCglobal architecture [16], the most influential architecture and potential future nucleus of IoT, DS is still in development, ONS only provides a pointer to the information service provided by the manufacturer of the object. Moreover, ONS [15] is based on the well-known Domain Name System (DNS), each query must start from Root ONS. Thus, the ONS will inherit all of the well-documented DNS weaknesses, such as the limited redundancy in practical implementations and the creation of single points of failure [29]. For ONS, this architecture will have a deep impact on the reliability, security and privacy of the involved stake holders and their business processes, especially for information clients.

In IoT, a lookup service to locate item-level information stored at potentially unknown supply chain partners is still missing, and current lookup service and discovery service mainly rely on a centralized or a chain-style framework, e.g. EPCglobal Architecture, Affilias DS [1, 25], BRIDGE Project [24] and the Distributed ePedigree Architecture [17]. Apart from the issue of security and privacy, these systems have some drawbacks, such as poor scalability, load imbalance, poor reliability owing to the presence of single points of failure, or bottlenecks, which prevent them from being widely adopted. Moreover, to the best of our knowledge, the issue of locating hotspot resource in IoT has received much less attention.

In the last years, Peer-to-Peer (P2P) network has become one of the most popular applications in the Internet, and the P2P paradigm has emerged as an alternative to centralized and hierarchical architectures. The approaches to enhance the performance and robustness of lookup service by using structured P2P systems (e.g. Chord [28]) based on Distributed Hash Tables (DHT) that have a high potential as a replacement for ONS as

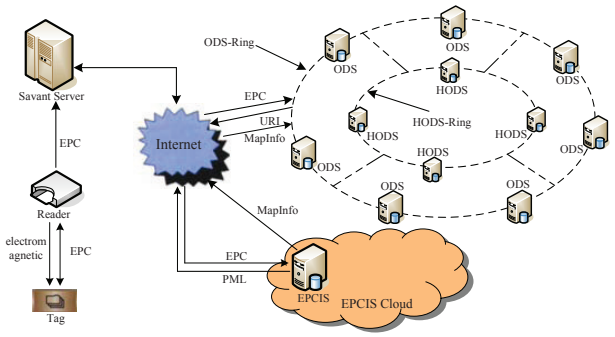

Figure 1: The framework of HDSA

well. Therefore, we present HDSA, a distributed hotspot-based IoT discovery service architecture adopting double-Chord-ring, give its framework and some relevant mechanisms. Here we mainly focus on the goals of meeting security and privacy requirements.

The rest of this paper is structured as follows. In Section 2, we give the framework of HDSA and some relevant mechanisms for security consideration. Then, we further discuss and analysis our solution in Section 3. Section 4 describes some work in related disciplines. Finally, we draw conclusions and outline future work in Section 5.

2 Hotspot-based Discovery Service Architecture

In this section, we present a distributed hotspot-based IoT discovery service architecture HDSA. The basic framework of HDSA is shown in Figure 1.

2.1 The Base Composition of HDSA

HDSA consists of three major parts Savant Server, ODS (Object Discovery Service) System and EPCIS Cloud.

2.1.1 Savant Server

Savant [6], designed to process the streams of tag or sensor data (event data) coming from of one or more reader devices, is a middleware software system that sits between tag readers and enterprise applications. Its intent is to address the unique computational requirements presented by EPC applications. Savant performs filtering, aggregation, and counting of tag data, reducing the volume of data prior to sending to Enterprise Applications. The Savant itself is a container for processing modules defined by Auto-ID standards or users and other third parties. More details about Savant can be found in [6]. The Savant server is a server installed with savant software. Each company deploys only one Savant Server logically, although in practice, every one may deploy more than one Savant Server.

2.1.2 ODS System

In HDSA, from functionality, we combine ONS and DS into ODS, and introduce Chord into ODS, the structured Chord overlay networks are the network substrate of the applications about ODS and information interaction. Every participant, such as manufacture, distributor, or retailer, deploys a dedicated ODS node, and all the ODS nodes are organized in a Chord ring and ordered following the hash values of their IPs.

Within a continuous period T, when the accessed times of an EPC exceeds the accessed times threshold of EPC (AT), the EPC is called hotspot resource (HR). Only depends on ODS-Ring, as the HR objects are frequently accessed by the client application of some organizations, it generates vast network traffic flows that may lead to the network congestion of ODS-Ring. To relieve this problem and balance the flows of query, we add a HR ODS-Chord-Ring (HODS-Ring) into HDSA. On the other hand, HODS brings appropriate hotspot data redundancy and backup, helps to enhance the reliability of the system. ODS-Ring and HODS-Ring together form an ODS System. The ODS-Ring is responsible for the queries of all objects (include HR), and the HODS-Ring is only responsible for the queries of HR objects. In real world, HODS nodes, may be derived from the participants'ODS servers which have better hardware configuration, also may are responsible by the third party or be constructed and maintained by the government, according to network region. Whichever method to be selected, it is depended on the concrete situation or the relevant regulation of real world.

In one company, the ODS/HODS node may each be implemented by multiple physically separate servers that act as backups for each other to increase the scalability and reliability of the entire system. It is important to note that the number of HODS nodes must be far smaller than the one of ODS nodes, which is analyzed in detail in our another paper.

2.1.3 EPCIS Cloud

The EPCIS is a role defined in EPCglobal Network Architecture Framework [16], which provide for storage and retrieval of filtered and processed information related to EPC-tagged objects about different events within the supply-chain. In HDSA, each participant (company or organization) maintains at least one EPCIS server. All the EPCIS servers constitute an EPCIS Cloud.

Normally, a company needs only one EPCIS Server in theory. As a kind of service, each company may deploy more than one EPCIS Server (multiple redundant servers, one for backup of another one) as needed, but logically, the external feature of multiple EPCIS Servers within a company is still one server through the relevant mechanism of main-backup. For the sake of cost saving, for every stakeholder, its ODS may be combined together with its EPCIS server.

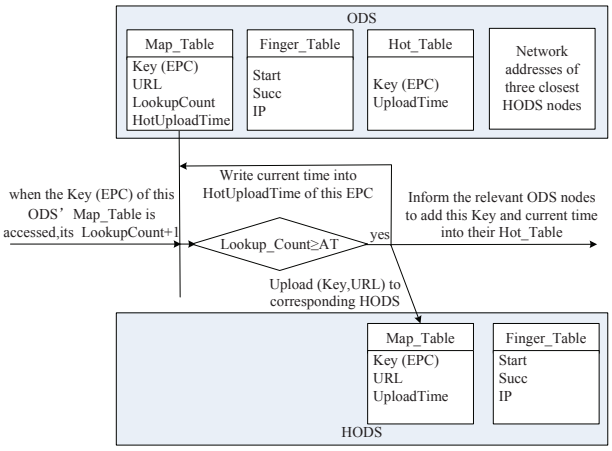

Figure 2: ODS storage & data flow chart

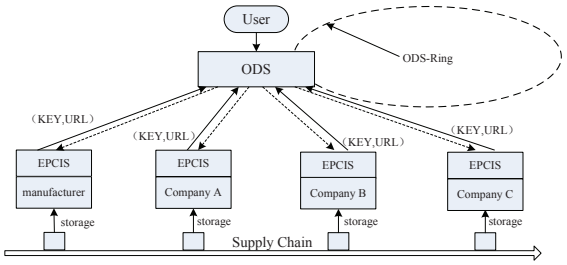

Figure 3: Information publishing principle chart

2.2 Storage Mechanism

In our HDSA framework, the DS is designed not to duplicate or aggregate information stored within each individual EPCIS repository, but to store only sufficient relevant information to be able to create the list of links.

Each of the ODS node has and maintains a finger table, a mapping table (map-table), a hotspot table (hot-table), and holds the network addresses of three closest HODS nodes to facilitate the information interaction between it and HODS-Ring. Here, finger table is similar to the one of Chord [28]. Hot-table includes two fields: key represents the hash value of hotspot EPC, and upload time represents the time when the EPC become as HR. Map-table mainly stores the list of mappings between EPC and the network addresses (IP address or URL) of its corresponding EPCIS servers, the lookup count (i.e. the accessed times of the EPC). The information storage in ODS System are shown in Figure 2, where Key (EPC) represents the hash value of EPC. Only one finger table and one map-table are in HODS node, where the roles and content of these tables are similar to the ones in ODS node.

2.3 Information Interaction Mechanism

In our HDSA, the information interactions principally include three aspects: publishing of resource information, interaction of hotspot information and switch of lookup between ODS-Ring and HODS-Ring.

2.3.1 Information Publishing

The current EPCIS standard [9, 21] does not involve a specific communication mechanism between an EPCIS and a lookup service. Thus, here, we give an information publishing mechanism to send the association between an EPC and the URL of the relevant EPCISs to corresponding ODS node.

When an EPC-tagged object flows along the supply chain, the relevant EPCIS servers must publish in time the relevant event information related to the EPC to the corresponding ODS node that determined by hashing the object EPC. The event information registered or published to ODS by EPCIS (see Figure 3) is mainly the mapping information associated to the current EPC that is passing the company that possesses this EPCIS. The mapping information is a 2-tuple (KEY, URL), includes two fields: the SHA1 (or MD5) hash value of the EPC, the URL where the information related to the EPC is available.

2.3.2 Interaction of Hotspot Information

As shown in Figure 2, for one ODS node, once someone EPC of its map-table becomes as HR, it simultaneously do three things. They are: (1) uploading the relevant information of this EPC to corresponding HODS nodes map-table, (2) writing current time into the field of Hot Upload Time of its map-table, (3) and informing the relevant ODS nodes which are consulting the EPC to add this Key and current time into their hot-table. On the other hand, with the moving through the supply chain, for a HR, once new mapping information about it is published to the corresponding ODS node by relevant EPCIS Server, this ODS node upload the newest information to relevant HODS node to assure the performance of real-time of mapping information.

2.3.3 Switch of Lookup

When an ODS node receives an EPC query of subscriber or end user, if it finds that the EPC is in its hot-table, it will forward the query to its available closest HODS node (one of the three closest HODS nodes). Then, the HODS node performs the query in inside ring, and returns query result to this ODS node.

2.4 Lookup Mechanism

As the complete information about an individual object may be fragmented across multiple organizations, the role of lookup service is to locate all the providers of the fragments of information that constitute the complete supply-chain or lifecycle history for an object. Here, we give relevant lookup mechanism: to obtain the address list of EPCIS server related to EPC-tagged object being queried

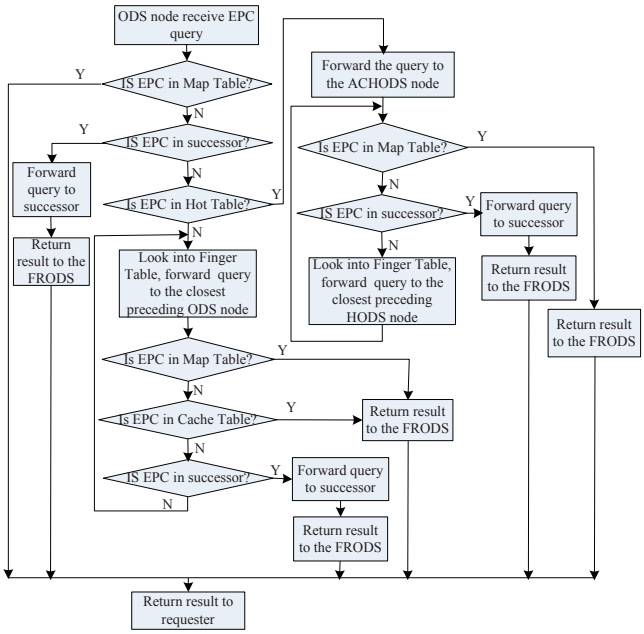

Figure 4: Flow chart of routing in chord-based ODS system

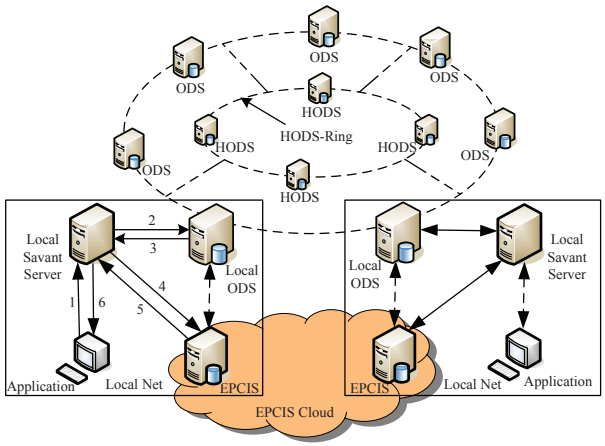

Figure 5: Base composition of company deployment & lookup data flow of application level

through the routing in ODS system, and then to get the complete information about this object through the lookup of application level. In our architecture framework, the hash value of EPC of object is used as an index of query.

2.4.1 Routing in Chord-based ODS System

As shown in Figure 4, routing in Chord-based ODS System is based on the following EPC lookup algorithm.

Firstly, the ODS node which is the first receiver of a query from a requester (Savant server) on a given EPC, looks into its map-table (we call this ODS node FRODS). If this EPC is found, it returns directly relevant address information to requester. Otherwise, it judges whether the key (the hash value of the EPC) is located between itself and its immediate successor. If is, it forwards the query to the immediate successor that is responsible for the requested EPC-tagged object, and then, the immediate successor returns directly desired result to FRODS. If not, it looks into hot-table, if this EPC is found, it forwards the query to ACHODS (one of the three closest HODS node and is available) which is responsible for this query within HODS-Ring and return net result (relevant address information) to it. If FRODS does not find EPC in hot-table, it looks into its finger table for the closest ODS node to the key (the hash value of the EPC) that has a lower or equal identifier, and forwards the query message to this ODS node which is called the closest predecessor refer to this key.

The closest predecessor does the work similar to what done by the FRODS. The main difference between them is that the closest predecessor does not look into its hot-table. In this way, the query message is routed through the ODS-Ring until it reaches a node containing mapping information related to the requested EPC-tagged object, this node then returns directly desired result to FRODS. Finally, FRODS returns the result to the requester.

Likewise, when HODS node receives the query forwarded by other node (FRODS or other HODS node), it takes the handling similar to what done by the closest predecessor in ODS-Ring. The main difference is that HODS lookups EPC only in its map-table.

2.4.2 Lookup of Application Level

In Figure 5, the parts within rectangle line represent the base composition of a company deployment, and the first to the sixth step of the query procedure of application level are described as follows:

1) The application issues EPC query to local Savant server through whose application software interface.

2) The local Savant server calls the query module to forward the query to the local ODS server.

3) The local ODS server calls lookup module to consult ODS system (as depicted in Section 2.4 1). If it finds successfully the desired result a list of network addresses of relevant EPCIS servers related to the EPC, it returns the result to the local Savant server. Otherwise, it analyzes the causes of failure and sends this analysis result to the local Savant server.

4) The local Savant server analyzes the result from the local ODS Server. If the result is failure causes, it switches to exception handling, and informs application. If is the list of EPCIS addresses, it accesses in parallel the corresponding EPCISs of EPCIS Cloud according to the list of addresses, to collect the detail information related to the EPC.

5) Every corresponding EPCIS sends its lookup result using PML (Physical Markup Language) file format to the local Savant server.

6) The local Savant server resolves these lookup results from previous step and sends them to the application through the application software interface.

Finally, the application integrates the collated results and displays them through the user-friendly interface.

2.5 Security Mechanism

In IoT, lookup services or Discovery Services are not only critically dependent upon the high efficiency of lookup and the integrity, but also the confidentiality they offer their customers. Therefore, in HDSA, apart from all information interaction use secure channels, the following security measures are given to meet the corresponding security and privacy requirement.

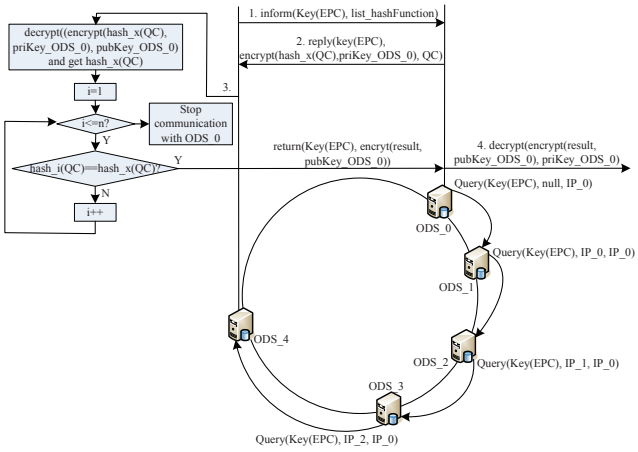

Figure 6: Base composition of company deployment & lookup data flow of application level

2.5.1 Mutual Authentication

To enable the retrieved address and object information could be authenticated, the responder and requester of object information must be authenticated mutually via certificate services. In HDSA, all participants (organization/company) need to obtain two certificates from certificate authority (CA), one certificate used for query is called query certificate (QC) and installed on the Savant server and the ODS server, another certificate used for sending lookup result to requester is called response certificate (RC) and installed on EPCIS server.

When a participant approaches CA to obtain certificates, its identities (for responder or requester) are verified and after complete verification, the corresponding certificate is issued. Communicating participants use them for mutual authentication. Before Savant server communicates with the corresponding EPCIS server, a mutual verification of them is performed. The certificates (QC of Savant server and RC of EPCIS server) which they obtained from Certificate authority are verified.

2.5.2 User Account Management

Some information about some item is confidential to some user and should only be accessed by the user has the corresponding right. Therefore, we provide corresponding access right for user to prevent unauthorized read of information.

Before beginning to perform the operation of lookup, every user must have an account including user name, password, and access right. The account information of user is stored in the user-account-table in the relevant Savant server.

In our solution, the Savant servers of all participants are assigned a corresponding level right (is called lookup-right) to lookup object information. The levels of their lookup-rights do not have to be same or different. Whatever level to be assigned, it is depended on the concrete situation or the relevant regulation of real world. When registering to the Savant server of someone participant, user must provide some necessary identity information and credentials to be verified whether the user has the qualification to obtain the level of lookup-right. If has, the Savant server creates corresponding user account for the user in its user-account-table, give user name and password, and assign its lookup-right to the access right of the user, and add the base identity information of the user into this table. Otherwise, the Savant server refuses to create account for the user, so the user could not to issue lookup via the Savant server.

2.5.3 Security Measure in Communication between Savant and EPCIS

After mutual authentication between Savant server and the corresponding EPCIS server is completed, a session key is created, Savant sever begin to issue query encrypted by the session key to EPCIS server. EPCIS server receives and decrypts the query information from the Savant server by the session key to get the EPC and the access right of the query user. Then, according to the access right, the corresponding information of item related to the EPC is extracted from the EPCIS servers lookup result, and sent to Savant server after being encrypted by the session key. Finally, the Savant server receives and decrypts the information from EPCIS server by the session key to obtain the desired lookup result.

2.5.4 Security Measure in ODS System

In ODS system, for the sake of backtracking processing when the lookup fails, the query message mainly include three parts: the hash value of EPC, the IP address of FRODS and the IP address of the closest source ODS node (e.g. ODS_1 is the closest source ODS node of ODS_2 in Figure 6) of current ODS node. When the destination ODS node (e.g. ODS_4 in Figure 6) is found, it

immediately initiates communication to FRODS (ODS_0 is the FRODS in Figure 6). As shown in Figure 6, the detailed procedures are described as follow:

1) ODS_4 informs ODS_0 that it has the desired information about Key(EPC) and the hash algorithms supported by it include: hash_0, hash_1, $\cdots$, hash_n.

2) ODS_4 decrypts the encrypted data from ODS_0 by the public key of ODS_0 and get hash_x(QC). Then, it uses hash_i ($i = 1, 2, \cdots, n$) one by one to get hash_i(QC), and compares hash_i(QC) and hash_x(QC). If they are equal, ODS_4 returns the lookup result (list of addresses) encrypted by the public key of ODS_0 on the Key(EPC) to ODS_0; Otherwise, when i is equal to n, hash_i(QC) is still not equal to hash_x(QC), ODS_4 stops communication with ODS_0 and disconnects with it.

3) ODS_0 decrypts the encrypted lookup result from ODS_4 by its private key and gets the desired lookup result about Key(EPC).

3 Discussion and Analysis

In the following, according to privacy enhancing technologies and Security and privacy needs of IoT, we give corresponding discussion and analysis for our solution and security mechanism.

3.1 Privacy Enhancing Technologies

It is quite difficult to enhance the privacy of user of network and service infrastructures. In order to achieve information privacy goals, a number of Privacy Enhancing Technologies (PET) have been developed. They can be described in brief as follows [12, 30]:

- Virtual Private Network (VPN) is extranet established by close groups of business partners. This solution may reduce the confidentiality and integrity risks, but it does not allow for a dynamic global information exchange and is impractical with regard to third parties beyond the borders of the extranet.

- Transport Layer Security (TLS) could improve confidentiality and integrity of the IoT on the base of an appropriate global trust structure, but it would be negatively affected the performance of lookup.

- DNS Security Extensions (DNSSEC) use public-key cryptography to guarantee origin authenticity and integrity of delivered information. However, DNSSEC could only assure global ONS information authenticity if the Internet community as a whole adopts it.

- Onion Routing cryptographically transform and mix Inter-net traffic from many different sources to impede matching a particular IP packet to a particular source. However, onion routing would affect the usability of ONS and Discovery Services because of the latency and performance issues.

- Private Information Retrieval (PIR) systems conceal which client is interested in what information, once the EPCIS have been located. However, in the global lookup system such as the ONS, problems of scalability and key management, as well as performance issues, make this method impractical.

Some of the above- mentioned methods could be combined to create alternatives for enhancing security and private of IoT.

To further increase security and privacy, DHT-based (Distributed Hash Tables) Peer-to-Peer (P2P) system that generally shows good scalability and performance in real-world applications, is a good method. In recent years, as one of the most popular applications in the Internet, the P2P paradigm has emerged as an alternative to centralized and hierarchical architectures. The advantages of DHTs include, among other aspects, self-organizing, load-balance, less traffic that data placement and search procedures generate. Chord, one of the most popular DHT-based systems, has not only simple design idea and good features of distribution, scalability, stability, and load balancing, and it is the focus of the present research, such as [7, 8, 14, 18, 19, 26, 27]. Though both existing EPCIS and DHT have not offered any encryption measure or access control [11], the encryption of EPCIS connection and the authentication of customer could be implemented without major difficulties, using common Internet and web service security frameworks [12], and the authentication of customer can be done by issuing shared secrets or using public-key cryptography [11]. Therefore, combining some of the above- mentioned methods and implementing access control at the actual EPCIS itself, a distributed hotspot-based discovery service architecture based on Chord for IoT, is a good solution to meet the functional requirements (with high performance and robustness) and security & privacy requirements.

3.2 Security and Privacy Requirements

From the technologies point of view, the lookup service (or discovery service) architecture of the IoT has an impact on the security and privacy of the involved stake holders. As described in [11] and [30], a high degree of reliability is needed in business processes and private enterprises using IoT technology will have to include the following security and privacy requirements into their risk management concept governing the business activities in general:

- Resilience to attacks: The system can avoid single points of failure and adjust itself to node failures.

- Data authentication: Retrieved address and object information should be authenticated.

- Access control: Information providers must be able to implement access control on the data provided [3].

- Client privacy: Measures need to be taken that only the information provider is able to infer from observing the use of the lookup system related to a specific customer; at least, inference should be very hard to conduct.

As HDSA is constructed on the base of Chord protocol, all storage-load and lookup-load are distributed evenly on the entire system, which make it can avoid single of failure and enhance its adaptivity by performing the stabilization operation inherited from Chord. The mutual authentication between the Savant server performing query operation and the corresponding EPCIS server enables the retrieved address and object information could be authenticated in HDSA. Though the user account management, HDSA provides corresponding access right for user to prevent unauthorized read of information, thus obtains some access control capabilities.

In our solution, information publishing, query, and storage are all done using the hash value of EPC, which make the information is not understandable or meaningful to non-requester (non-subscriber) of query or eavesdroppers or attackers, and thus enhance the protection on privacy of client and security of information to certain extent. On the other hand, due to the adoption of P2P, the number of ODS (and HODS) nodes the adversary or attackers have to monitor is very large and is increasing, which make the inference be very hard to conduct for the attackers. Additionally, the security measures in ODS System ensure the correctness and completeness of the returned discovery service information, in particular the addresses of the relevant EPCIS, and the security measure in communication between Savant and EPCIS ensures the correctness and completeness of the object information itself, which be able to reduce the whole systems confidentiality and integrity risks.

In ODS system, the hash value of EPC is used as the index to locate the corresponding EPCIS servers address and the number of ODS (or HODS) nodes is very large, which make the eavesdroppers cannot know the actual value of EPC and eavesdropping more difficult. Therefore, the mutual authentication between ODS (or HODS) nodes is not very necessary. So, our solution does not require the mutual authentication, which can help the improvement of lookup performance.

Technically, measures ensuring the IoT architectures resilience to attacks, data authentication, access control and client privacy need to be established. On the other hand, international legislator [30] would best establish an adequate legal framework considering the underlying technology, so the requirements of security and privacy can better meet.

4 Related Works

In the last years, the achievements of EPCglobal standardization efforts are substantial and the diffusion of the EPCglobal network continues. However, DS is not yet specified, the granularity of ONS lookup is still limited to product type, rather than serial-level lookup in the up-to-date version of ONS 2.0.1 [15], and ONS is based on DNS suffering from well-studied weaknesses in robustness, configuration complexity, and security. Relying heavily on Root ONS to implement traceability applications, makes the centralized architecture of hierarchical ONS is vulnerable to single point failure and workload-imbalanced due to excessive lookup-load of Root ONS. Alias DS is compliant to the architecture framework of EPCglobal, its basic characteristics are hierarchical lookups and DNS-based naming and translation. The BRIDGE project, supported by the EU and coordinated by GS1, addressed a wide spectrum of problems related to the implementation of RFID in Europe, whose prototype is very similar to the EPCglobal approach. The two kinds of approach and the EPCglobal approach share the same advantages and disadvantages [10].

Barchetti et al. [2] focused on the implementation of DS developed as an extension of FossTrak open framework [13] based on centralized framework. MUTLER et al. [22] presented a centralized aggregating DS for the EPCglobal Network and showed how to overcome scalability challenges through notification in a real-world environment. MANZANARES-LOPEZ et al. [21] proposed an distributed discovery service for EPCglobal network in nested package scenarios, which is based on the implementation of a DHT-based (Pastry) ONS and a totally distributed DS. Although they solve some problems, they all ignore the enhancement of security and privacy.

HUANG et al. [17] propose a distributed ONS architecture by combining ONS and DHT to find out drug counterfeit points in the pharmacy supply chains, and the distributed ONS be constructed involving the EPCIS servers of all the participants. This approach of locating drug counterfeit points through forward tracing or reverse tracing is called Daisy Chain approach. Although the approach is implemented in a distributed and relatively secure manner, all the processes rely on rewritable tags and must be supervised by an entity to be able to identify the offender [4, 21]. In addition, to collect information about an item, traversing all relevant EPCISs along the supply chain for a given EPC cannot be parallelized and therefore raises high lookup latency because each EPCIS must be queried sequentially [4, 22].

In preceding studies, researchers pay less or no attention on the issue of locating hotspot resource object in IoT. In our solution, ODS and DS are combined into ODS from functionality and Chord lookup algorithm is introduced into ODS, which not only make the entire system is easier and more efficient to implement item-level tracking & tracing, but also remove the faults DNS-based ONS. The hotspot-based HODS-Ring further improves the efficiency of lookup, balances the flows of query and reduces the probability of congestion of ODS-Ring, which make the whole system is more efficient and robust. In contrast to the Distributed ePedigree Architecture proposed by Huang et al. in HDSA, the information collec-

tion from all relevant EPCISs along the supply chain for a given EPC can be parallelized and therefore achieves low lookup latency because each EPCIS can be queried concurrently, which further improves the lookup efficiency in practice. Additionally, the introduction of security mechanism ensures our architectures resilience to attacks, data authentication, access control and client privacy.

5 Conclusions

For the current development status and existing problems of lookup service in IOT, especially for the issue of security and privacy, this paper proposes HDSA with security mechanism, which points out one of viable directions to foster the development of IOT. So far, many research problems and implementation issues are still unsolved, and require more efforts from both academia researchers and industrial practitioners, though lookup service is a vital research direction in IOT. Future work should focus on the latest security techniques and protocols (such as [5, 20, 23]) and further enhance the security of the Internet of Things. Additionally, to introduce Cloud Storage and Cloud Computation into HDSA will also be our next work.

Acknowledgments

This study was supported by the National Natural Science Funds Fund of China [61172084, 61202046], Natural Science Foundation of Hubei Province of China [2013CFC026], and the project of Study on Information Security of Chord-based Object Discovery for the Internet of Things. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Afilias, "Afilias discovery services: Enabling secure, selective visibility in global supply chains," Tech. Rep. White Paper, 2008.

[2] U. Barchetti, A. Bucciero, M. De Blasi, L. Mainetti, and L. Patrono, "Implementation and testing of an epcglobal-aware discovery service for item-level traceability," in *Proceedings of International Conference on Ultra Modern Telecommunications and Workshops (ICUMT'09)*, pp. 1–8, St. Petersburg, Russia, Oct. 2009.

[3] C. Floerkemeier, M. Langheinrich, E. Fleisch, F. Mattern, and S. E. Sarma, *The Internet of Things*, Berlin/Heidelberg: Springer, 2008.

[4] J. J. Cantero, M. A. Guijarro, G. Arrebola, E. Garcia, J. Baos, M. Harrison, and T. Kelepouris, "Traceability applications based on discovery services," in *Proceedings of IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008)*, pp. 1332–1337, Hamburg, Sept. 2008.

[5] C. H. Wei, M. S. Hwang, and A. Y.H. Chin, "An authentication protocol for low-cost rfid tags," *International Journal of Mobile Communications*, vol. 9, no. 2, pp. 208–223, 2011.

[6] S. Clark, K. Traub, D. Anarkat, and T. Osinski, "Auto-id savant specification 1.0,", Tech. Rep. Auto-ID Center, 2003.

[7] R. Cuevas, M. Uruena, and A. Banchs, "Routing fairness in chord: Analysis and enhancement," in *Proceedings of IEEE INFOCOM'09*, pp. 1449–1457, April 2009.

[8] Y. P. Deng and H. Du, "Improved chord algorithm based on physical topology," *Computer Engineering and Design*, vol. 33, no. 10, pp. 3734–3738, 2012.

[9] EPCglobal, "EPC information service (EPCIS) ver. 1.0.1," tech. rep., Sept. 2007.

[10] S. Evdokimov, B. Fabian, S. Kunz, and N. Schoenemann, "Comparison of discovery service architectures for the internet of things," in *Proceedings of 2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'10)*, pp. 237–244, California, USA, June 2010.

[11] B. Fabian and O. Gunther, "Distributed ons and its impact on privacy," in *Proceedings of IEEE International Conference on Communications (ICC'07)*, pp. 1223–1228, Glasgow, Scotland, June 2007.

[12] B. Fabian and O. Gunther, "Security challenges of the epcglobal network," *Communications of the ACM*, vol. 52, no. 7, pp. 121–125, 2009.

[13] C. Floerkemeier, C. Roduner, and M. Lampe, "Rfid application development with the accada middleware platform," *Systems Journal*, vol. 1, no. 2, pp. 82–94, 2007.

[14] A. Forestiero, E. Leonardi, C. Mastroianni, and M. Meo, "Self-chord: A bio-inspired p2p framework for self-organizing distributed systems," *IEEE/ACM Transactions on Networking*, vol. 18, no. 5, pp. 1651–1664, 2010.

[15] GS1, "GS1 object name service (ons) 2.0.1," tech. rep., Jan. 2013.

[16] GS1, "The GS1 epcglobal architecture framework-ver. 1.5," tech. rep., Mar. 2013.

[17] D. J. Huang, M. Verma, A. Ramachandran, and Z. B. Zhou, "A distributed epedigree architecture," in *Proceedings of the 11th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS'07)*, pp. 220–230, Mar. 2007.

[18] W. A. Imtiaz, S. Shil, and A. M. Rahman, "Three layer hierarchical model for chord," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 11, pp. 96–101, 2012.

[19] C. Jennings, B. Lowekamp, E. Rescorla, S. A. Baset, and H. Schulzrinne, "Resource location and discovery (reload)," Tech. Rep. IETF Internet-Draft, 2008.

[20] W. Khedr, "On the security of moessners and khans authentication scheme for passive epcglobal c1g2 rfid tags," *International Journal of Network Security*, vol. 16, no. 5, pp. 369–375, 2014.

[21] P. Manzanares-Lopez, J. P. Munoz-Gea, J. Malgosa-Sanahuja, and J. C. Sanchez-Aarnoutse, "An effcient distributed discovery service for epcglobal network in nested package scenarios," *Journal of Network and Computer Applications*, vol. 34, no. 3, pp. 925–937, 2011.

[22] J. Muller, J. Oberst, S. Wehrmeyer, J. Witt, A. Zeier, and H. Plattner, "An aggregating discovery service for the epcglobal network," in *Proceedings the 43rd Hawaii International Conference on System Sciences*, pp. 1–9, Honolulu, HI, Jan. 2010.

[23] A. Nitaj, "Cryptanalysis of ntru with two public keys," *International Journal of Network Security*, vol. 16, no. 2, pp. 112–117, 2014.

[24] University of Cambridge, AT4 wireless, BT Research, and SAP Research, "Bridge wp02-working prototype of serial-level lookup service," Tech. Rep. BRIDGE project, 2008.

[25] A. Rezafard, "Extensible supply-chain discovery service problem statement," Tech. Rep. IETF Internet-Draft, 2008.

[26] L. Schmidt, R. Dagher, R. Quilez, N. Mitton, and D. Simplot-Ryl, "Dht-based distributed ale engine in rfid middleware," Tech. Rep. Research Report 7316, INRIA, 2010.

[27] Y. Shimano and F. Sato, "Dynamic reconfiguration of chord ring based on physical network and finger table information," in *Proceedings of 2012 15th International Conference on Network-Based Information Systems (NBiS'12)*, pp. 66–73, Melbourne, Australia, Sept. 2012.

[28] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 4, pp. 149–160, 2001.

[29] R. H. Weber, "Internet of things - need for a new legal environment," *Computer Law and Security Review*, vol. 25, no. 6, pp. 522–527, 2009.

[30] R. H. Weber, "Internet of things new security and privacy challenges," *Computer Law and Security Review*, vol. 26, no. 1, pp. 23–30, 2010.

[31] Lu Yan, Y. Zhang, L. T. Yang, and H. Ning, *The Internet of things: from RFID to the next-generation pervasive networked systems*, New York/London: Auerbach, 2008.

Degang Xu received the BE degree in computer science and technology, M.S. degree in computer application technology and the Ph.D. degree in computer system architecture from Huazhong University of Science and Technology in 2000, 2007 and 2013, respectively. He is currently a lecturer in the School of Mathematics and Computer Science, Hubei University of Arts and Science, CHINA. His research interests include computer network and security, cloud computing, DHT-based P2P network, and Internet of Things.

Zhao Wu received the BE degree in computer application from China University of Geoscience, M.S, degree in computer application from Wuhan University of Technology and the Ph.D. degree in Computer software and theory from Wuhan University, in 1999, 2003 and 2007, respectively. He is currently a associate professor in the School of Mathematics and Computer Science, Hubei University of Arts and Science, CHINA. His research interests include cloud computing, service computing, and internet of things. He is a member of the Association for Computing Machinery.

Zhongbo Wu received the BE degree in information technology from Central China Normal University, MS degree in computer application from Huazhong University of Science and Technology and the PhD degree in computer application from Renmin University of China, in 2001, 2004 and 2010, respectively. He is currently an associate professor in the School of Mathematics and Computer Science, Hubei University of Arts and Science, CHINA. His research interests include cloud computing, database, and internet of things. He is a member of the IEEE.

Qilin Zhang received the BE degree in computer science and technology from Huazhong University of Science and Technology in 2003, MS degree in computer application technology from Wuhan university of technology in 2006 and the PhD degree in communication and information system from Wuhan University 2013, respectively. He is currently a lecturer in the School of Mathematics and Computer Science, Hubei University of Arts and Science, CHINA. His research interests include computer network and security, smart grid and Internet of Things.

Leihua Qin received the Ph.D. degree in computer system architecture in 2007 from Huazhong University of Science and Technology. He is now an associate professor at School of Computer Science and Technology of HUST. His main research interests include computer network and cloud computing, network storage system and Internet of Things.

Jingli Zhou received the B.E. degree in 1969. She is a Professor and doctor advisor at Huazhong University of Science and Technology. She had been a visiting scholar in USA from 1995 to 1996 and has been honor of the State Department Special Allowance since 1999. Her main field of research: computer network, network storage system and multimedia signal processing.