ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

IJIRSET

International Journal of Innovative Research in Science, Engineering and Technology

Volume 3, Special Issue 3, March 2014

2014 International Conference on Innovations in Engineering and Technology (ICIET'14)
On 21st&22ndMarch Organized by

K.L.N. College of Engineering, Madurai, Tamil Nadu, India

Distributed Secure and Privacy- Preserving Information Using Brokering System

Catherin Jenifer.S[1], Anandha Kumar.M[2]

Department of Computer & Communication Engineering, M.A.M College of Engineering, Tamilnadu, India.[1]

Department of Information & Technology Engineering, Assistant Professor, M.A.M College of Engineering, Tamilnadu, India.[2]

Abstract-Interaction between entities that may not trust each other is now commonplace on the Internet. It focuses on the specific problem of sharing information between distrusting parties. Previous work in this area shows that privacy and utility can co-exist, but often do not provide strong assurances of one or the other. To sketch a research agenda with several directions for attacking these problems, considering several alternative systems that examine the privacy vs. utility problem from different angles. Therefore to propose a novel approach to preserve privacy of multiple stakeholders involved in the information brokering process. First of all to define two privacy attacks, namely attribute-correlation attack and inference attack, and propose two countermeasure schemes such as automaton segmentation and query segment encryption to securely share the routing decision-making responsibility among a selected set of brokering Servers. With comprehensive security analysis and experimental results, shows that our approach seamlessly integrates security enforcement with query routing to provide system-wide security with insignificant overhead.

Keywords – Access control, information sharing, privacy.

I.INTRODUCTION

Along with the explosion of information collected by organizations in many realms ranging from business to government agencies, there is an increasing need for interorganizational information sharing to facilitate extensive collaboration. While many efforts have been devoted to reconcile data heterogeneity and provide interoperability, the problem of balancing peer autonomy and system coalition is still challenging. Most of the existing systems work on two extremes of the spectrum, adopting either the query-answering model to establish pair wise client-server connections for on-demand information access, where peers are fully autonomous but there lacks system wide coordination, or the distributed database model, where all peers with little autonomy are managed by a unified DBMS.

Unfortunately, neither model is suitable for many newly emerged applications, such as healthcare or law enforcement information sharing, in which organizations share information in a conservative and controlled manner due to business considerations or legal reasons. Take healthcare information systems as example. Regional Health Information Organization (RHIO) [1] aims to facilitate access to and retrieval of clinical data across collaborative healthcare providers that include a number of regional hospitals, outpatient clinics, payers, etc. As a data provider, a participating organization would not assume free or complete sharing with others, since its data is legally private or commercially proprietary, or both. Instead, it requires to retain full control over the data and the access to the data. Meanwhile, as a consumer, a healthcare provider requesting data from other providers expects to preserve her privacy in the querying process.In the context of sensitive data and autonomous data providers, a more practical and adaptable solution is to construct a data-centric overlay (e.g., [2], [3]) consisting of data sources and a set of brokers that make routing decisions based on the content of the queries [4]-[5]. Such infrastructure builds up semantic-aware index mechanisms to route the queries based on their content, which allows users to submit queries without knowing data or server location.

In previous study [5], [6], such a distributed system providing data access through a set of brokers is referred to as *Information Brokering System* (IBS) always involve some sort of consortium (e.g., RHIO) among a set of organizations. Databases of different organizations are connected through a set of brokers, and metadata (e.g., data summary, server locations) are pushed to the *local brokers*, which further some of the metadata to other brokers. Queries are sent to the local broker and routed according to the metadata until reaching the right data server(s). In this way, a large number of information sources in different organizations are loosely federated to provide a unified, transparent, and on-demand data access.While the IBS approach provides scalability and server autonomy, privacy concerns arise, as brokers are no longer assumed fully trustable—the broker functionality may be outsourced to third-party providers and thus vulnerable to be abused by insiders or compromised by outsiders.

It presents a general solution to the privacy-preserving information sharing problem. First, to address the need for privacy protection. Here, propose a novel IBS, namely Privacy Preserving Information Brokering(PPIB). It is an overlay infrastructure consisting of two types of brokering components, brokersand coordinators. The brokers, acting as mix anonymizer [7] are mainlyresponsible for user authentication and query forwarding. The coordinators, concatenated in a tree structure, enforce access control and query routing based on the embedded nondeterministic finite automata—the query brokering automata. To prevent curious or corrupted coordinators from inferring private information, we design two novel schemes to segment the query brokering automata and encrypt corresponding query segments so that routing decision making is decoupled into multiple correlated tasks for a set of collaborative coordinators. While providing integrated in-network access control and content-based query routing, the proposed IBS also ensures that a curious or corrupted coordinator is not capable to collect enough information to infer privacy, such as “which data is being queried”, “where certain data is located”, or “what are the access control policies”, etc. Experimental results show that PPIB provides comprehensive privacy protection for on-demand information brokering, with insignificant overhead and very good scalability.

In this paper is organized as follows introduce the related work in Section II, and discuss the privacy requirements and threats in the information brokering scenario in Section III, and Section IV, its present two core brokering schemes and the types as follows. Thendiscuss the construct the maintenance in Section V, evaluate the performance in Section VI, and conclude future work in Section VII.

II. RELATED WORKS

Research areas such as information integration, peer-to-peer file sharing systems and publish-subscribe systems provide partial solutions to the problem of large-scale data sharing. In this section, the discussed about the Information integration system, Automation segmentation and XML query routing.

A. INFORMATION BROKERING SYSTEM

Information integration approaches focus on providing an integrated view over a large number of heterogeneous data sources by exploiting the semantic relationship between schemas of different sources [8]-[9]. The PPIB study assumes that a global schema exists within the consortium, therefore, information integration is out of our scope.

While PPIB aims to locate relevant data sources for a given query and route the query to these data sources.PPIB addresses more privacy concerns other than anonymity, and thus faces more challenges.

B. NON-DETERMINISTIC FINITE AUTOMATON

It adopts an NFA-based query rewriting access control scheme proposed recently in [15], [5], which has a better performance than previous view-based approaches [12].

It adopt the *Nondeterministic Finite Automaton* (NFA) based approach as presented in [15], which allows access control to be enforced outside data servers, and independent from the data. The NFA-based approach constructs NFA elements for four building blocks of common XPath axes. So that, XPath expressions, as combinations of these building blocks, can be converted to an NFA, which is used to match and rewrite incoming XPath queries. Please refer to [15] for more details on the QFilter approach.This allows access control to be enforced outside data servers, and independent from the data.Each packet would be sent segmental and time delay occurs.

C.XML QUERY ROUTING

Research on distributed access control is also related to work gives a good overview on access control in collaborative systems [10]. In this part, earlier approaches implement access control mechanisms at the nodes of XML trees and filter out data nodes that users do not have authorization to access [11], [12]. These approaches rely much on the XML engines. View-based access control approaches create and maintain a separate view for each user [13], [14], which causes high maintenance and storage costs.

The eXtensible Markup Language (XML) has emerged as the *de facto* standard for information sharing due to its rich semantics and extensive expressiveness. We assume that all the data sources in PPIB exchange information in XML format, i.e., taking XPath[16] queries and returning XML data. Note that the more powerful XML query language, XQuery, still uses XPath to access XML nodes. In XPath, predicates are used to eliminate unwanted nodes, where test conditions are contained within square brackets. To specify the authorization at the node level, fine-grained access control models are desired. [13].

In particular, specialized data structures are maintained on overlay nodes to route XML queries. In [3], a robust mesh has been built to effectively route XML packets by making use of self-describing XML tags and the overlay networks. Kouds*et al.* also proposed a decentralized architecture for ad hoc XPath query routing across a collection of XML databases [4]. To share data among a large number of autonomous nodes, [18] studied content-based routing for path queries in peer-to-peer systems.

III. OVERVIEW OF PPIB SYSTEM

In this part, first discuss the framework of PPIB and then focus the privacy problem with the revelation of brokers.Eventually, to define some basic notions.

A. THE ARCHITECTURE OF PPIB SYSTEM FOR INFORMATION SHARING

It proposes a new model, namely *Privacy Preserving Information Brokering* (PPIB). PPIB has threetypes of brokering components: *brokers*, *coordinators*, and a *central authority*.

*1) Brokers:*Fig.1 shows the architecture of PPIB. Data servers and requestors from different organizations connect to the system through local brokers. Brokers are interconnected through coordinators. A local broker functions as the "entrance" to the system. It authenticates the requestor and hides his identity from other PPIB components.

*2)Coordinators:*Coordinators are responsible for content-based query routing and access control enforcement. With privacy-preserving considerations, we cannot let a coordinator hold any rule in the complete form. Instead, we propose a novel *automaton segmentationscheme* to divide (metadata) rules into segments and assign each segment to a coordinator. Coordinators operate collaboratively to enforce secure query routing. A*query segmentencryption scheme* is further proposed to prevent coordinators from seeing sensitive predicates. The scheme divides a query into segments, and encrypts each segment in a way that to each coordinator enroute only the segments that are needed for secure routing is revealed.

*3) Central authority:*The CA is assumed for offline initiation and maintenance. It handles key management and metadata maintenance.

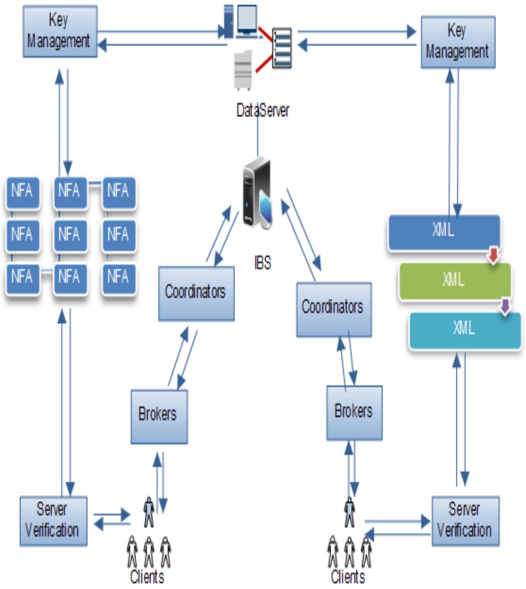

Fig.1 Architecture of PPIB

The architecture of PPIB is shown in Fig.2 where users and data servers of multiple organizations are connected via a broker-coordinator overlay. In particular, the brokering process consists of four phases:

- **Phase 1:** To join the system, a user needs to authenticate himself to the local broker. After that, the user submits an XML query with each segment encrypted by the corresponding public level keys, and a unique session key K_Q.K_Q is encrypted with the public key of the data servers to encrypt the reply data.
- **Phase 2:** Besides authentication, the major task of the broker is metadata preparation: (1) it retrieves the role of the authenticated user to attach to the encrypted query; (2) it creates a unique Q_{ID} for each query, and attaches Q_{ID}, $\langle K_Q \rangle$ pk_{DS} and its own address to the query for data servers to return data.
- **Phase 3:** Upon receiving the encrypted query, the coordinators follow automata segmentation scheme and query segment encryption scheme to perform access control and query routing along the coordinator tree as described. At the leaf coordinator, all query segments should be processed and encrypted by the public key of the data server. If a query is denied access, a failure message with Q_{ID} will be returned to the broker.
- **Phase 4:** In the final phase, the data server receives a safe query in an encrypted form. After decryption, the data server evaluates the query and returns the data, encrypted by K_Q, to the broker that originates the query.

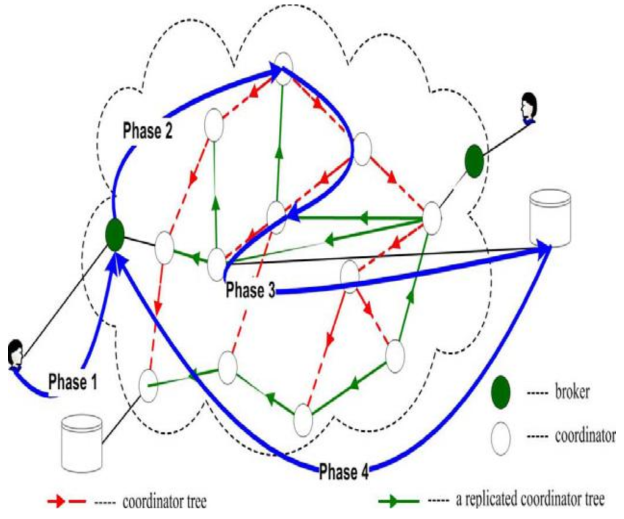

Fig.2we explain the query brokering process in four phases.

B. PROBLEM DEFINITION

In adopt the semi-honest[17] assumption for the brokers, and assume two types of adversaries, *(a) External attackers:*passively eavesdrop communication channels; and *(b) Curious or corrupted brokering components:*while following the protocols properly to fulfill brokering functions, try their best to infer sensitive or private information from the querying process. The attacker could further infer the privacy of different stakeholders through *attribute-correlation attacks* and *inference attacks*.

Attribute-correlation attack:Predicates of an XML querydescribe conditions that often carry sensitive and private data (e.g., name, SSN, credit card number, etc.) If an attacker in percepts a query with multiple predicates or composite predicate expressions, the attacker can "correlate" the attributes in the predicates to infer sensitive information about data owner. This is known as the attribute correlation attack.

For example: A tourist Anne is sent to ER at CaliforniaHospital. Doctor Bob queries for her medical records through a Medicare IBS. Since Anne has the symptom of leukemia, the query contains two predicates: [pName="Anne"], and [symptom="leukemia"]. Any malicious broker that has helped routing the query could guess "Anne has a blood cancer" by correlating the two predicates in the query.

Inference attack:More severe privacy leak occurs when anattacker obtains more than one type of sensitive information and learns explicit or implicit knowledge about the stakeholders through association. By "implicit", we mean the attacker infers the fact by "guessing".

For example: an attacker can guess the identity of a request or from her query location (e.g., IP address). Meanwhile, the identity of the data owner could be explicitly learned from query content (e.g., name or SSN). Attackers can also obtain publicly-available information to help his inference. If an attacker identifies that a data server is located at a cancer research center, he can tag the queries as "cancer-related".

IV. PRIVACY-PRESERVING QUERY BROKERING SCHEME

The Broker [5] approach has severe privacy vulnerability is discussed problem statement.If the Broker is compromised or cannot be fully trusted. To tackle the problem, presents the PPIB infrastructure with two core schemes. In this part, first explain the solution details of *automata segmentation* and *query segmentencryption* schemes.

A.AUTOMATON SEGMENTATION (NFA)

It analyzes the attack models with distinct backdrop knowledge. In the context of distributed information brokering, multiple organizations join a consortium and agree to share the data within the consortium. While different organizations may have different schemas, they assume a global schema exists by aligning and merging the local schemas. Thus, the access control rules and index rules for all the organizations can be crafted following the same shared schema and captured by a global automaton. The key idea of automaton segmentation scheme is to logically divide the global automaton into multiple independent yet connected segments, and physically distribute the segments onto different brokering components, known as coordinators.

The atomic unit in the segmentation is an NFA state of the original automaton. Each segment is allowed to hold one or several NFA states. They further define the granularity levelto denote the greatest distance between any two NFA states contained in one segment. Given a granularity level, for each segmentation, the next states will be divided into one segment with a probability. Obviously, with a larger granularity level, each segment will contain more NFA states, resulting in less segments and smaller end-to-end overhead in distributed query processing. However, a coarse partition is more likely to increase the privacy risk. The tradeoff between the processing complexity and the degree of privacy should be considered in deciding the granularity level.As privacy protection is of the primary concern of this work.To ensure the segments are logically connected, they also make the last states of each segment as "dummy" accept states, with links pointing to the segments holding the child states of the original global automaton.To suggest a structure of PPIB is distributed information for privacy protection.

To reserve the logical connection between the segments after segmentation, here define the following heuristic segmentation rules:

- NFA states in the same segment should be connected via parent-child links.
- Sibling NFA states should not be put in the same segment without their parent state.
- The "accept state" of the original global automaton should be put in separate segments.

B.QUERY SEGMENT ENCRYPTION(XML)

It computes a final score for each possible segmentation by adding the MWE scores of individual segments. Then we pick the segmentation that yields the highest segmentation score. Here we use a dynamic programming approach to search over all possible segmentations. Uses query logs as the only resource and can effectively capture the structural units in queries. That uses only query logs. The query segment encryption scheme is proposed following three types.

1) Level-Based Preencryption: According to the automaton segmentation scheme, query segments are processed by a set of coordinators along a path in the coordinator tree. A straightforward way is to encrypt each query segment with the public key of the coordinator specified by the scheme. Hence, each coordinator only sees a small portion of the query that is not enough for inference, but collaborating together, they can still fulfill the designed function. The key challenges in this approach is that the segment-coordinator association is unknown beforehand in the distributed setting, since no party other than the CA knows how the global automaton is segmented and distributed among the coordinators.

2) Post encryption: The processed query segments should also be protected from the remaining coordinators in later processing, so post encryption is necessary. In a simple scheme, It assume all the data servers share a pair of public and private keys, {$pk_{DS,}$ sk_{DS}}, where $pk_{DS}$is known to all the coordinators. Each coordinator first decrypts the query segment(s) with its private level key, performs authorization and indexing, and then encrypts the processed segment(s) with pk_{DS} so that only the data servers can view it.

3) *Commutative Encryption*: Commutative encryption algorithms [19], [20] have the property of being commutative, where an encryption algorithm is commutative if for any two commutative keys e_1 and e_2 and a message m, ‹‹m ›e_1›e_2=‹‹m› e_2, e_1. Therefore, we assign a new commutative level key$e_i$to nodes at level i, and further assume nodes at level share e_i with nodes at level i+2. It adopts Pohlig - Hellman exponentiation cipher with modulus as our commutative encryption algorithm to generate the commutative keys.

V. MAINTENANCE

KEY MANAGEMENT

The CA is assumed for offline initiation and maintenance. With the highest level of trust, the CA holds a global view about all the rules and plays a critical role in automaton segmentation and key management. There are four types of keys used in the brokering process: query session key K_Q, public/private level keys {pk, sk}, commutative level keys {e, d}, and public/private data server keys {$pk_{DS,}$ sk_{DS}}. Except the query session keys created by the user, the other three types of keys are generated and maintained by the CA. The data servers are treated as a unique party and share a pair of public and private keys, while each of the coordinators has its own pairs of level key and commutative level key. Along with the automaton segmentation and deployment process, the CA creates key pairs for coordinator at each level and assigns the private keys with the segments. The level keys need to be revoked in a batch once a certificate expires or when a coordinator at the same level quits the system.

VI. PERFORMANCE ANALYSIS

In this part, analyze the performance of proposed
PPIBsystem using system scalability and end- to-end query processing system. In system results, coordinators are coded in Java(JDK 5.0) and results are collected from coordinators runningon a Windows desktop (3.4 G CPU). This is wildly used in the research community,as good imitation of real world applications.

In the results show that PPIB provides comprehensive privacy protection for on-demand information brokering, with insignificant overhead and very good scalability.

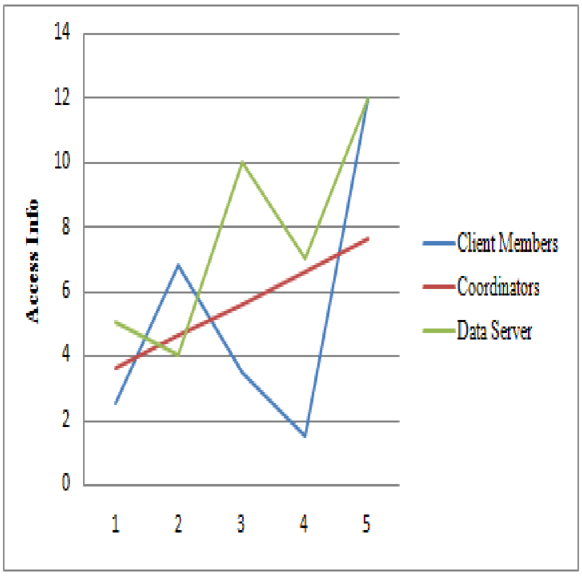

Fig.3Distributed Shared Data

A. SYSTEM SCALABILITY

In this task, it evaluates the scalability of the PPIB system againstcomplicity of ACR and data size (number of data objects and data servers) and then following aspects.

1) Complicity of XML Schema and ACR: When the segmentation scheme is determined, the demand of coordinators is determined by the number of ACR segments, which is linear withthe number of access control rules. Assume finest granularity automaton segmentation is adopted, we can see that the increase of demanded number of coordinators is linear or even better. This is because similar access control rules with same prefix may share XPath steps, and save the number of coordinators.Moreover, different ACR segments or, logical coordinators may reside at the same physical site, thus reduce the actual demand of physical sites. In this framework, the numbers of coordinators m, and the height of the coordinator tree h, are highly dependent on how access control policies are segmented. In this part, the segments are received fully.

*2) Data Size:*Fig.3when data volume increases (e.g., addingmore data items into the online auction database), the number of indexing rules also increases. This results in increasing of the number of leaf-coordinators. However, in PPIB, query indexing is implemented through hash tables, which is scalable. Thus, the system is scalable when data size increases. Also shared secure and privacy- preserving information using brokering system

B. END-TO-END QUERY PROCESSING TIME

In the results, the total forward query processing time is calculated as,

$$T_{\text{forward}} \simeq 1.9 \times 5.7 + 100\ (5.7 + 1) \simeq 681(\text{ms}).$$

It is obvious that network latency T_N*(N_{HOP}+1) dominates total forward end-to-end query processing time, because the value of T_C is negligible compared with T_N. since T_N remains the same (as an estimation from Internet traffic), N_{HOP} becomes the deterministic factor that affects end-to-end query processing time. Note that for other information brokering systems, although they use different query routing scheme, network latency is not avoidable. As a conclusion, the proposed PPIB approachachieves privacy-preserving query brokering and accesscontrol with limited computation.

VII. CONCLUSION AND FUTURE WORK

Little attention provided to privacy of user, data, and metadata during the design stage, existing information brokering systems suffer from a spectrum of vulnerabilities associated with user privacy, data privacy, and metadata privacy. Here, propose PPIB, a new approach to preserve privacy in XML information brokering. Through an innovative automaton segmentation scheme, in network access control, and query segment encryption, PPIB integrates security enforcement and query forwarding while providing comprehensive privacy protection. The analysis shows that it is very resistant to privacy attacks. End-to-end query processing performance and system scalability are also evaluated and the results show that PPIB is efficient and scalable.

As future work, many directions are ahead for future research. First, at present, site distribution and load balancing in PPIB are conducted in an ad-hoc manner. The next step of research is to design an automatic scheme that does dynamic site distribution. Several factors can be considered in the scheme such as the workload at each peer, trust level of each peer, and privacy conflicts between automaton segments. Designing a scheme that can strike a balance among these factors is a challenge. Second, to quantify the level of privacy protection achieved by PPIB. Finally, Plan to minimize or even eliminate the participation of the administrator node, who decides such issues as automaton segmentation granularity. A main goal is to make PPIB self-reconfigurable.

ACKNOWLEDGMENT

I would like to express my special thanks of gratitude to Mr.A.SelvaReegan as well as our guide Mr.M.AnandhaKumar who helped me in completing my work. Secondly I would like to thank my parents, my friends and well wishers who were a constant source of Inspiration.

REFERENCES

[1] W. Bartschat, J. Burrington-Brown, S. Carey, J. Chen, S.Deming, and S. Durkin, *“Surveying the RHIO landscape: A description of current {RHIO} models, with a focus on patient identification,”*J. AHIMA, vol. 77, pp. 64A–64D, Jan. 2006.

[2] X. Zhang, J. Liu, B. Li, and T.-S. P. Yum, *“CoolStreaming/DONet: A data-driven overlay network for efficient live media streaming,”* in Proc. IEEE INFOCOM,Miami, FL, USA, 2005, vol. 3, pp. 2102–2111.

[3] A.C. Snoeren, K. Conley, and D. K. Gifford, *“Mesh-based content routing using XML,”* in Proc. SOSP, 2001, pp. 160-173.

[4] N. Koudas, M. Rabinovich, D. Srivastava, and T. Yu, *“Routing XML queries,”* in Proc. ICDE’04, 2004, p. 844.

[5] F. Li, B. Luo, P. Liu, D. Lee, P. Mitra,W. Lee, and C. Chu, *“In-brokeraccess control: Towards efficient end-to-end performance information brokerage systems,”* in Proc. IEEE SUTC, Taichung, Taiwan, 2006, pp. 252–259.

[6] F. Li, B. Luo, P. Liu, D. Lee, and C.-H. Chu, *“Automaton segmentation: A new approach to preserve privacy in XML information brokering,”* in Proc. ACM CCS’07, 2007, pp. 508–518.

[7] D. L. Chaum, *“Untraceable electronic mail, return addresses, and digital pseudonyms,”*Commun. ACM, vol. 24, no. 2, pp. 84–90, 1981.

[8] M. Genesereth, A. Keller, and O.Duschka, *“Informaster: An information integration system,”* in Proc. SIGMOD, Tucson, AZ, USA, 1997.

[9] J. Kang and J. F. Naughton, “*On schemamatching with opaque column names and data values,”* in Proc. SIGMOD, 2003, pp. 205–216.

[10] W. Tolone, G.-J. Ahn, T. Pai, and S.-P. Hong, “*Access control in collaborative systems,”*ACM Comput. *Surv.*, vol. 37, no. 1, pp. 29–41,2005.

[11] S. Cho, S. Amer-Yahia, L. V. S. Lakshmanan, and D. Srivastava, *“Optimizing the secure evaluation of twig queries,”* in Proc. VLDB, 2002, pp. 490–501.

[12] M.Murata, A. Tozawa, andM. Kudo, *“XML access control using static analysis,”* in Proc. ACM CCS, 2003, pp. 73–84.

[13] S. Rizvi, A. Mendelzon, S. Sudarshan, and P. Roy, *“Extending query rewriting techniques for fine-grained access control,”* in Proc. SIGMOD’04, Paris, France, 2004, pp. 551–562.

[14] T. Yu, D. Srivastava, L. V. S. Lakshmanan, and H. V. Jagadish, *“Compressed accessibility map: Efficient access control for XML,”* in Proc.VLDB, China, 2002, pp. 478–489.

[15] B. Luo, D. Lee, W. C. Lee, and P. Liu, *“Qfilter: Fine-grained runtime XML access control via NFA-based query rewriting enforcement mechanisms,”* in Proc. *CIKM*, 2004, pp. 543–552.

[16] A.Berglund, S. Boag, D. Chamberlin, M. F. Fernndez, M. Kay, J. Robie, and J. Simon, *“XML Path Language (XPath)”*. ver. 2.0, 2003 [Online].Available:http://www.w3.org/TR/xpath20/

[17] R. Agrawal, A. Evfimivski, and R. Srikant, *“Information sharing across private databases,”* in Proc. 2003 ACM SIGMOD, San Diego, CA, USA, 2003, pp. 86–97.

[18] G. Koloniari and E. Pitoura, *“Content-based routing of path queries in peer-to-peer systems,”* in Proc. EDBT, 2004, pp. 29–47.

[19] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Zhu, *“Tools for privacy preserving distributed data mining,”*ACM SIGKDD ExplorationsNewsletter, vol. 4, no. 2, pp. 28–34, 2003.

[20] H. Y. S. Lu, *“Commutative cipher based en-route filtering in wireless sensor networks,”* in Proc. VTC, Sep. 2004, vol. 2, pp. 1223–1227.