Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 2, February 2014, pg.533 – 538

REVIEW ARTICLE

A Review on Privacy Problems in Distributed Information Brokering System and Solutions

Mr. Ashutosh Kamble[1], Prof. Deepak Kapgate[2], Prof. Prakash Prasad[3]

[1]Department of CSE, GHRAET, Nagpur (M.S), India
[2]Department of CSE, GHRAET, Nagpur (M.S.), India
[3]Department of Information Technology, Nagpur (M.S.), India
[1]kontact.ashu@gmail.com
[2]deepakkapgate32@gmail.com
[3]prakashprasad@yahoo.com

Abstract*— There is an increasing need for information sharing via on-demand access in different organization. Information Brokering Systems (IBSs) have been introduced to connect large-scale lightly-associated data sources with the help of a brokering overlay. In this system, the brokers make routing decisions to direct client queries to the requested data servers. Some present IBSs consider that brokers are trusted and thus only adopt server-side access control for data confidentiality. But, the privacy of location of data and information about consumer can still be concluded from metadata (such as query and access control rules) exchanged in the IBS, but little attention has been paid for its protection. This paper presents an overview on information sharing in distributed environment through information brokering system and problems associated with it. It also defines two attacks- attribute correlation attack and inference attack.

***Keywords*— Information Brokering, Access control, information sharing, data confidentiality**

I. INTRODUCTION

Nowadays enterprise information systems are designed as distributed network systems, where existing information systems and new components are connected together via a middleware. A globalization in a sphere of business creates need for decentralized systems with a correct data distribution, distributed processing, reservation of resources and a reliable communication infrastructure. In recent years, we have observed an explosion of information shared among organizations in many realms ranging from business to government agencies. To facilitate efficient large scale information sharing, many efforts have been devoted to reconcile data heterogeneity and provide interoperability across geographically distributed data sources. The distributed information systems are designed as a network of communicating and partially independent components where each component performs its specific task, by itself or with help of other components.

A Distributed Information Brokering System (DIBS) is a peer-to-peer overlay network that comprises diverse data servers and brokering components helping client queries locate the data server(s). All existing information brokerage systems view or handle access control and query brokering as two orthogonal issues. Access control is a security issue that concerns information confidentiality, while query brokering is a system issue that concerns costs and performance.

Most of the existing systems work on two extremes of the spectrum: (1) peers are fully autonomous in the query-answering model for on-demand information access but there is no system-wide coordination; so that participants create pairwise client-server connections for information sharing; (2) in the traditional distributed database systems, all the

participates lost autonomy and are managed by a unified DBMS. Unfortunately, neither of them is suitable for many newly emerged applications in which organizations share information in a conservative and controlled manner.

The rest of the paper discusses the following: Various approaches for information sharing in distributed environment are discussed in Section II, the privacy requirements and threats in the information brokering scenario is stated in Section III, and related work in Section IV. Finally, the conclusion is presented in Section V.

II. APPROACHES FOR DISTRIBUTED INFORMATION SHARING

Distributed Information System (DIS) is seen as a collection of autonomous information systems which can collaborate with each other. Consider the example of healthcare information systems, such as Regional Health Information Organization (RHIO) [1]. It aims to facilitate access to and retrieval of clinical data across collaborative health providers. An RHIO is formed with multiple stakeholders. As a data provider, a consumer would not accept free or complete sharing with others, since its data is legally private or commercially registered, or both. In fact, there should be full control over the data and access to the data. Meanwhile, as a consumer, a health provider requesting data from other providers expects to protect private information (e.g. requestor's identity, interests) in the querying process.

In such scenarios, sharing a complete copy of the data with others or "pouring" data into a centralized repository becomes impractical. To address the need for autonomy, federated database technology has been proposed [2], [3], to manage locally stored data with a federated DBMS and provide unified data access. However, the centralized DBMS still introduces data heterogeneity, privacy, and trust issues. Meanwhile, the peer-to-peer information sharing framework is often considered a solution between "sharing nothing" and "sharing everything". In its basic form, every pair of peers establishes two symmetric client-server relationships, and requestors send queries to multiple databases. This approach assumes 2n relationships for n peers, which is not scalable.

In the context of sensitive data and autonomous data owners, a more practical and adaptable solution is to construct a data centric overlay (e.g. [4], [5]), including the data sources and a set of brokers helping to locate data sources for queries [6], [7], [8], [9]. 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 our previous study [9], [10], such a distributed system providing data access through a set of brokers is referred to as Information Brokering System (IBS).

As shown in Figure 1, applications atop IBS always involve some sort of consortium (e.g. RHIO) among a set of organizations. Databases of various organizations are connected through a set of brokers, and metadata (e.g. data summary, server locations) are "pushed" to the local brokers, which further "advertise" (some of) the metadata to other brokers. Each query is sent to the local broker. The brokers, route this query according to the metadata until reaching the right database(s). In this way, a large number of information sources in different organizations are loosely federated to provide fused, clear, and on-demand data access.

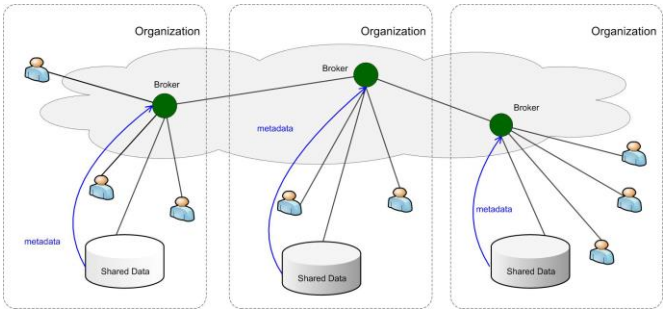

Fig. 1. An overview of the IBS infrastructure.

The IBS approach provides server autonomy and scalability. With this the privacy concerns arise, as brokers are no longer assumed fully trustable – they may be abused by insiders or compromised by outsiders. For overcoming this, a novel IBS, named Privacy Preserving Information Brokering (PPIB) (Fig.2) has been proposed. It is an overlay infrastructure consisting of two types of brokering components: brokers and coordinators. The brokers, acting as mix anonymizers [11], are mainly responsible for user authentication and query forwarding. The coordinators are concatenated in a tree structure. They enforce access control and query routing based on the embedded nondeterministic finite automata. To prevent corrupted coordinators from inferring private information, two novel schemes were proposed: (a) to segment the query brokering automata, and (b) to encrypt corresponding query segments. These two schemes provide full capability to enforce in-network access control and to route queries to the right data sources. With this they ensure that a corrupted coordinator is not capable to collect enough information to infer privacy. PPIB provides

comprehensive privacy protection for on-demand information brokering, with insignificant overhead and very good scalability.

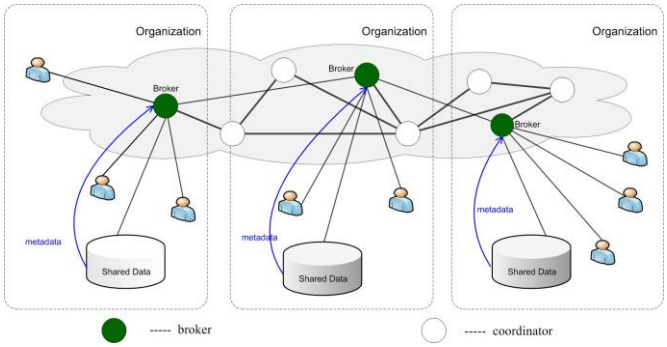

Fig. 2. The architecture of PPIB.

III. VULNERABILITIES AND ATTACKS

Information Brokering System (IBS) over a peer-to-peer connection has been proposed to support information sharing among loosely associated data sources. It consists of various data servers and brokering components. These components help client queries to locate the data servers. The information brokers deal with the IBS systems and are responsible for providing and processing the data between heterogeneous entities, hence they are also called as the Data Brokers. In real world, directly or indirectly everybody gets affected with such Information Brokering systems as the Information Brokers collect information about consumers from a variety of public and non-public sources such as website cookies etc. and sell them to businesses who want to target their advertisements and special offers.

A. *Threats*:

The privacy threats that arise in the DIBS systems are as follows:

1) *User Privacy*:

Generally speaking, we can summarize the user privacy as who, where, and what". "Who" refers to the identity of a user, where" denotes his/her location at the moment of sending a query, and "what" represents the interest and purpose that can be inferred from his/her query. User location could be easily retrieved by analyzing the IP packet of the query. User identity is a key concern of user privacy, which can be obtained either from authentication process or by associating user location information with other public data. The “what" privacy may not be known directly, but some reasonable inference from the content of the query can be made. Although user identity, user location, and query content are privacy-sensitive matters, one cannot apply popular privacy preserving techniques directly in the DIBS. This is because a broker needs to learn this privacy-sensitive information to full query brokering. For example, since data in DIBS is only accessible by legitimate users, user identity cannot be represented by anonymity as other privacy-preserving applications do. In other words, the broker is responsible for authenticating user identity. As a result, to what extent user's privacy is preserved highly depends on how we minimize the disclosure of this privacy-sensitive information. This requires a new mechanism where the broker cannot infer the privacy of individuals while still fulfilling its designated functions

2) *Data Privacy*:

In DIBS, data owners collect data independently and manage it with autonomous data servers. While providing data access to legitimate users, data servers have to release certain privacy-sensitive information that needs to be protected. In general, two questions, “where is the data stored?" and “who stores what data?", can express privacy concerns of data. The first question concerns data location privacy, and the second question, denoted data object distribution privacy, inquires which type of data is contained in a particular data server. Unlike other large public databases or data warehouse, data owners in the proposed DIBS are highly conservative about their data privacy. They only share data and data distribution within the consortium.

3) *Metadata Privacy*:

Two types of metadata are involved in the information brokering process in a DIBS, query indexing guidelines and access control rules. The former describes where the data objects are distributed among all the data servers, and the latter assigns accessibility to legitimate users according to access control policy provided by data owners. It is obvious that the metadata is highly relevant to both the privacy of data location and the privacy of data object distribution. However, to facilitate information brokering, these metadata have to be stored at the intermediate brokering components, which may be abused by the insider or compromised by the outsider according to our assumptions. As a result, the metadata becomes an obvious and easier target of attacks. Risk rises when unsecured or dishonest brokering components try to abuse or leak this privacy-sensitive information. In existing DIBS approaches, a compromised broker can obtain data location information from indexing guidelines or access control policy since these information are stored in brokers to facilitate routing and access control. Even if we can adopt some encryption schemes to hide these sensitive information

from brokers, a compromised broker can probe the whole system by sending snooping queries. In this way, a compromised broker is more dangerous to the system than ordinary malicious users.

B. *Attacks*:

1) *Attribute-correlation attack*: In this attack, an attacker intercepts a query (in plaintext), which typically contains a number of predicates. Every predicate defines a condition, which sometimes involves reserved and sensitive data (e.g. name, SSN or credit card number, etc.). If a query has multiple predicates or composite predicate expressions, the attacker can "correlate" the corresponding attributes to infer sensitive information about the data owner. This attack is known as the attribute correlation attack:

   Example 1: A tourist Anne is sent to the emergency room at California Hospital. Doctor Bob queries for her medical records through a Medicare IBS. As Anne has the symptom of leukemia, the query has two predicates: [name="Anne"], and [symptom="leukemia"]. Any malicious broker that has helped routing the query could guess "Anne has a blood cancer" by correlating two predicates in the query. 2 Unfortunately, sensitive query content, including the predicates, cannot be simply encrypted since such information is necessary for content-based query routing. Therefore, we are facing a paradox of the requirement for content-based brokering and the risk of attribute-correlation attacks.

2) *Inference attack*: More severe privacy leak occurs when an attacker obtains more than one type of sensitive information, and further associates them to study explicit and implicit knowledge about the stakeholders. By "implicit", we mean the attacker infers the fact by "guessing". For example, an attacker can guess the identity of a requestor 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 in the predicate). Attackers can also obtain publicly available information to help his inference. For example, if an attacker identifies a data server located at a cancer research center, he can tag the related queries as "cancer-related". There are three distinct combinations of private information, and the reasonable inferences: (1) From query location & data location, the attacker concludes information about who (i.e. a specific requestor) is interested in what (i.e. a specific type of data). (2) From query location & query content, the attacker infers knowledge about who is interested in what, or where who is, if the query has predicates describing one's interests (e.g. symptom or medicine) or identifying a personnel (e.g. name or address). (3) From query content & data location, the attacker deduces which data server has which data. Hence, the attacker could continuously monitor user queries or generate artificial queries to learn the data distribution of the system, which could be used to conduct further attacks.

IV. RELATED WORK

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. Information integration approaches focus on providing an integrated view over large numbers of heterogeneous data sources by exploiting the semantic relationship between schemas of different sources [13], [14], [15]. The PPIB study considers that a universal schema exists within the association, therefore, information integration is out of our scope.

Peer-to-peer systems are designed to share files and data sets (e.g. in collaborative science applications). Distributed hash table technology [16], [17] is adopted to locate replicas based on keyword queries. However, although such technology has recently been extended to support range queries [18], the coarse granularity (e.g. files and documents) still makes them short of our expressiveness needs. Further, P2P systems may not provide complete set of answers to a request while we need to locate all relevant data.

Addressing a conceptually dual problem, XML publish subscribe systems (e.g. [19], [20]) are probably the closely related technology to the proposed research: while PPIB locates relevant data sources for a given query and route the query to these data sources, the pub/sub systems locate relevant consumers for a given document and route the document to these consumers. However, due to this duality, there are different concerns: they focus on efficiently delivering the same piece of information to a large number of consumers, while we are trying to route large volume but small-size queries to much fewer sites. Accordingly, the multicast solution in pub/sub systems does not scale in this situation and there is a need to develop new mechanisms.

One idea is to build an XML overlay architecture that supports expressive query processing and security checking atop normal IP network. In particular, specialized data structures are maintained on overlay nodes to route XML queries. In [5], 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 suggest a distributed architecture for ad hoc XPath query routing across a collection of XML databases [6]. To share data among a large number of autonomous nodes, [21] studies content-based routing for path queries in peer-to-peer systems. Different from these approaches, PPIB seamlessly integrates query routing with security and privacy protection.

Research on anonymous communication provides a way to protect information from unauthorized partners. Different protocols have been proposed to allow a message sender dynamically selecting a set of other users and relaying its request [22], [23]. Such approaches can be incorporated into PPIB to protect locations of data requestors and data servers from being known by irrelevant parties in communication. PPIB addresses more privacy concerns other than anonymity, aiming at enforcing access control during query routing, and thus faces more challenges.

Finally, many researches have been proposed on distributed access control. [24] present an overview on access control in collaborative systems. In summary, earlier approaches implement access control mechanisms at the nodes of XML trees and filter out data nodes that users do not have authorizations to access [25], [26]. These approaches rely much on the XML engines. View-based access control approaches create and maintain a separate view (e.g. a specific portion of XML documents) for each user [27], [28], which causes high maintenance and storage cost. Recently, a NFA-based query re-writing access control scheme has been proposed [29], [9], which has a better performance than view-based approaches [26].

V. CONCLUSION

For meeting the increasing need for information sharing IBS was introduced. With little attention drawn on 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. To overcome this, PPIB, a new approach to preserve privacy in XML information brokering, was proposed. PPIB integrates security enforcement and query forwarding while providing comprehensive privacy protection.

Many directions are ahead for future research. First, at present, site distribution and load balancing in PPIB are conducted in an ad-hoc manner. Also, there is research scope to design an automatic scheme that does dynamic site distribution. A number of 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, there is space to quantify the level of privacy protection achieved by PPIB. A main goal is to make PPIB self-reconfigurable.

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," Journal of AHIMA 77, pp. 64A–D, January 2006.

[2] A. P. Sheth and J. A. Larson, "Federated database systems for managing distributed, heterogeneous, and autonomous databases," ACM Computing Surveys (CSUR), vol. 22, no. 3, pp. 183–236, 1990.

[3] L. M. Haas, E. T. Lin, and M. A. Roth, "Data integration through database federation," IBM Syst. J., vol. 41, no. 4, pp. 578–596, 2002.

[4] X. Zhang, J. Liu, B. Li, and T.-S. P. Yum, "CoolStreaming/DONet. A data-driven overlay network for efficient live media streaming," in Proceedings of IEEE INFOCOM, 2005.

[5] A. C. Snoeren, K. Conley, and D. K. Gifford, "Mesh-based content routing using XML," in SOSP, pp. 160–173, 2001.

[6] N. Koudas, M. Rabinovich, D. Srivastava, and T. Yu, "Routing XML queries," in ICDE '04, p. 844, 2004.

[7] G. Koloniari and E. Pitoura, "Peer-to-peer management of XML data: issues and research challenges," SIGMOD Rec., vol. 34, no. 2, 2005.

[8] M. Franklin, A. Halevy, and D. Maier, "From databases to dataspaces: a new abstraction for information management," SIGMOD Rec., vol. 34, no. 4, pp. 27–33, 2005.

[9] F. Li, B. Luo, P. Liu, D. Lee, P. Mitra, W. Lee, and C. Chu, "In-broker access control: Towards efficient end-to-end performance of information brokerage systems," in Proc. IEEE SUTC, 2006.

[10] 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 ACM CCS '07, pp. 508–518, 2007.

[11] D. L. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," Communications of the ACM, vol. 24, no. 2, 1981.

[12] R. Agrawal, A. Evfimivski, and R. Srikant, "Information sharing across private databases," in Proceedings of the 2003 ACM SIGMOD, 2003.

[13] M. Genesereth, A. Keller, and O. Duschka, "Informaster: An information integration system," in SIGMOD, (Tucson), 1997.

[14] I. Manolescu, D. Florescu, and D. Kossmann, "Answering XML queries on heterogeneous data sources," in VLDB, pp. 241–250, 2001.

[15] J. Kang and J. F. Naughton, "On schema matching with opaque column names and data values," in SIGMOD, pp. 205–216, 2003.

[16] I. Stoica, R. Morris, D. Liben-Nowell, D. Karger, M. Kaashoek, F. Dabek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup protocol for Internet applications," in IEEE/ACM Transactions on Networking, vol. 11 of 1, 2003.

[17] R. Huebsch, B. Chun, J. Hellerstein, B. Loo, P. Maniatis, T. Roscoe, S. Shenker, I. Stoica, and A. Yumerefendi, "The architecture of PIER: an Internet-scale query processor," in CIDR, pp. 28–43, 2005.
[18] O. Sahin, A. Gupta, D. Agrawal, and A. E. Abbadi, "A peer-to-peer framework for caching range queries," in ICDE, 2004.
[19] A. Carzaniga, M. J. Rutherford, and A. L. Wolf, "A routing scheme for content-based networking," in Proc. Of INFOCOM, 2004.
[20] Y. Diao, S. Rizvi, and M. J. Franklin, "Towards an Internet-scale XML dissemination service," in VLDB Conference, (Toronto), August 2004.
[21] G. Koloniari and E. Pitoura, "Content-based routing of path queries in peer-to-peer systems.," in EDBT, pp. 29–47, 2004.
[22] M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions," ACM TISS, vol. 1, no. 1, pp. 66–92, 1998.
[23] P. F. Syverson, D. M. Goldschlag, and M. G. Reed, "Anonymous connections and onion routing," in IEEE Symposium on Security and Privacy, (Oakland, California), pp. 44–54, 4–7 1997.
[24] W. Tolone, G.-J. Ahn, T. Pai, and S.-P. Hong, "Access control in collaborative systems," ACM Comput. Surv., vol. 37, no. 1, 2005.
[25] S. Cho, S. Amer-Yahia, L. V. S. Lakshmanan, and D. Srivastava, "Optimizing the secure evaluation of twig queries.," in VLDB, 2002.
[26] M. Murata, A. Tozawa, and M. Kudo, "XML access control using static analysis," in ACM CCS, 2003.

