The Challenge of Assuring Data Trustworthiness

Elisa Bertino[1], Chenyun Dai[1], and Murat Kantarcioglu[2]

[1] Computer Science Department and CERIAS, Purdue University
West Lafayette, Indiana, USA
{bertino,daic}@cs.purdue.edu
[2] Computer Science Department, University of Texas at Dallas
Richardson, Texas, USA
muratk@utdallas.edu

Abstract. With the increased need of data sharing among multiple organizations, such as government organizations, financial corporations, medical hospitals and academic institutions, it is critical to ensure that data is trustworthy so that effective decisions can be made based on these data. In this paper, we first discuss motivations and requirement for data trustworthiness. We then present an architectural framework for a comprehensive system for trustworthiness assurance. We then discuss an important issue in our framework, that is, the evaluation of data provenance and survey a trust model for estimating the confidence level of the data and the trust level of data providers. By taking into account confidence about data provenance, we introduce an approach for policy observing query evaluation. We highlight open research issues and research directions throughout the paper.

Keywords: Data Integrity and Quality; Security; Policies.

1 Introduction

Today, more than ever, there is a critical need for organizations to share data within and across the organizations so that analysts and decision makers can analyze and mine the data, and make effective decisions. However, in order for analysts and decision makers to produce accurate analysis and make effective decisions and take actions, data must be trustworthy. Therefore, it is critical that data trustworthiness issues, which also include data quality and provenance, be investigated for organizational data sharing, situation assessment, multi-sensor data integration and numerous other functions to support decision makers and analysts. Indeed, today's demand for data trustworthiness is stronger than ever. As many organizations are increasing their reliance on data for daily operations and critical decision making, data trustworthiness, and integrity in particular, is arguably one of the most critical issues. Without integrity, the usefulness of data becomes diminished as any information extracted from them cannot be trusted with sufficient confidence.

The problem of providing "good data" to users is an inherently difficult problem which often depends on the semantics of the application domain. Also solutions for improving the data, like those found in data quality, may be very expensive and may require access to data sources which may have access restrictions, because of data

X. Zhou et al. (Eds.): DASFAA 2009, LNCS 5463, pp. 22–33, 2009.

sensitivity. Also even when one adopts methodologies to assure that the data is of good quality, errors may still be introduced and low quality data be used; therefore, it is important to assess the damage resulting from the use of such data, to track and contain the spread of errors, and to recover.

The many challenges of assuring data trustworthiness require articulated solutions combining different approaches and techniques. In this paper we discuss some components of such a solution and highlight relevant research challenges.

The rest of this paper is organized as follows. We start by a quick survey of areas that are relevant to the problem of data trustworthiness. We then present a comprehensive framework for policy-driven data trustworthiness, and discuss relevant components of such framework. Finally, we outline a few conclusions.

2 State of the Art

Currently there is no comprehensive approach to the problem of high assurance data trustworthiness. The approach we envision, however, is related to several areas that we discuss in what follows.

Integrity Models. Biba [3] was the first to address the issue of integrity in information systems. His approach is based on a hierarchical lattice of integrity levels, and integrity is defined as a relative measure that is evaluated at the subsystem level. A subsystem is some sets of subjects and objects. An information system is defined to be composed of any number of subsystems. Biba regards integrity threat as that a subsystem attempts to improperly change the behavior of another by supplying false data. A drawback of the Biba approach is that it is not clear how to assign appropriate integrity levels and that there are no criteria for determining them. Clark and Wilson [4] make a clear distinction between military security and commercial security. They then argue that security policies related to integrity, rather than disclosure, are of the highest priority in commercial information systems and that separated mechanisms are required for the enforcement of these policies. The model by Clark and Wilson has two key notions: well-formed transactions and separation of duty. A well-formed transaction is structured such that a subject cannot manipulate data arbitrarily, but only in constrained ways that ensure internal consistency of data. Separation of duty attempts to ensure the external consistency of data objects: the correspondence among data objects of different subparts of a task. This correspondence is ensured by separating all operations into several subparts and requiring that each subpart be executed by a different subject.

Semantic Integrity. Many commercial DBMS enable users to express a variety of conditions, often referred to as *semantic integrity constraints*, that data must satisfy [18]. Such constraints are used mainly for *data consistency and correctness*. As such semantic integrity techniques are unable to deal with the more complex problem of data trustworthiness in that they are not able to determine whether some data correctly reflect the real world and are provided by some reliable and accurate data source.

Data Quality. Data quality is a serious concern for professionals involved with a wide range of information systems, ranging from data warehousing and business intelligence to customer relationship management and supply chain management. One industry study estimated the total cost to the US economy of data quality problems at

over US$600 billion per annum [5]. Data quality has been investigated from different perspectives, depending also on the precise meaning assigned to the notion of data quality. Data are of high quality "if they are fit for their intended uses in operations, decision making and planning" [6]. Alternatively, the data are deemed of high quality if they correctly represent the real-world construct to which they refer. There are a number of theoretical frameworks for understanding data quality. One framework seeks to integrate the product perspective (conformance to specifications) and the service perspective (meeting consumers' expectations) [7]. Another framework is based in semiotics to evaluate the quality of the form, meaning and use of the data [8]. One highly theoretical approach analyzes the ontological nature of information systems to define data quality rigorously [9]. In addition to these more theoretical investigation, a considerable amount of research on the data quality has been devoted to investigating and describing various categories of desirable attributes (or dimensions) of data. These lists commonly include accuracy, correctness, currency, completeness and relevance. Nearly 200 such terms have been identified and there is little agreement on their nature (are these concepts, goals or criteria?), their definitions or measures. Tools have also been developed for analyzing and repairing poor quality data, through the use for example of *record linkage techniques* [22].

Reputation Techniques. Reputation systems represent a key technology for securing collaborative applications from misuse by dishonest entities. A reputation system computes reputation scores about the entities in a system, which helps single out those entities that are exhibiting less than desirable behavior. Examples of reputation systems may be found in several application domains; E-commerce websites such as eBay (ebay.com) and Amazon (amazon.com) use their reputation systems to discourage fraudulent activities. The EigenTrust [10] reputation system enables peer-to-peer file sharing systems to filter out peers who provide inauthentic content. The web-based community of Advogato.org uses a reputation system [19] for spam filtering. Reputation techniques can be useful in assessing data sources and data manipulation intermediaries; however their use for such purpose has not been yet investigated.

3 A Comprehensive Approach

Our envisioned approach [2] to data trustworthiness is based on a comprehensive framework composed of four key elements (see Figure 1). The first element is a mechanism for associating *confidence values* with data in the database. A confidence value is a numeric value ranging from 0 to 1 and indicates the trustworthiness of the data. Confidence values can be generated based on various factors, such as the trustworthiness of data providers and the way in which the data has been collected. The second element is the notion of *confidence policy*, indicating which confidence level is required for certain data when used in certain tasks. The third element is the computation of the confidence values of a query results based on the confidence values of each data item and lineage propagation techniques [11]. The fourth element is a set of strategies for incrementing the confidence of query results at query processing time. Such element is a crucial component in that it makes possible to return query results meeting the confidence levels stated by the confidence policies.

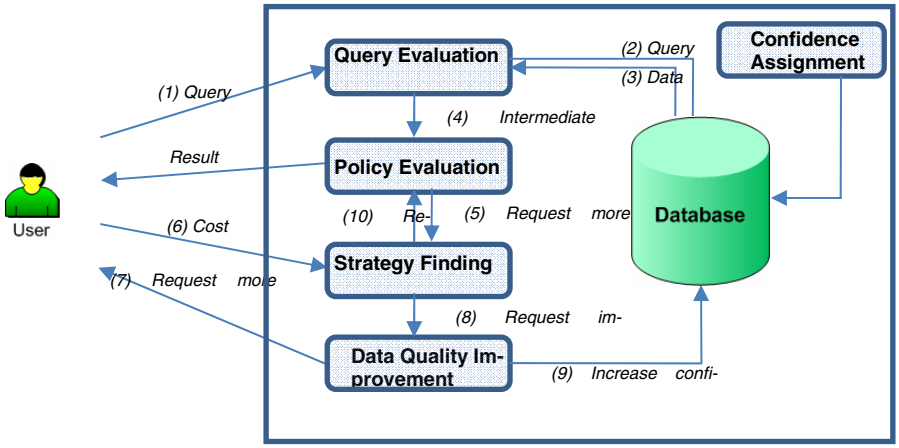

Fig. 1. A framework for assuring data trustworthiness

The notion of confidence policy is a key novel notion of our approach. Such a policy specifies the minimum confidence level that is required for use of a given data item in a certain task by a certain subject. As a complement to the traditional access control mechanism that applies to base tuples in the database before any operation, the confidence policy restricts access to the query results based on the confidence level of the query results. Such an access control mechanism can be viewed as a natural extension to the Role-based Access Control (RBAC) [12] which has been widely adopted in commercial database systems. Therefore, our approach can be easily integrated into existing database systems.

Because some query results will be filtered out by the confidence policy, a user may not receive enough data to make a decision and he/she may want to improve the data quality. To meet the user's need, an approach for dynamically incrementing the data confidence level (the fourth element of our solution) is required. Our envisioned approach will select an optimal strategy which determines which data should be selected and how much the confidence should be increased to satisfy the confidence level stated by the confidence policies.

4 Assigning Confidence Levels to Data

A possible approach to assign confidence levels, also referred to as *confidence scores*, to data is based on the trustworthiness of data provenance [1]. By data provenance we refer to information about the source of the data and the entities, such as agents, applications, users, who have accessed and/or manipulated the data before the data has been entered in the destination database. Though several research efforts have been devoted to data provenance [13, 14, 15, 16], the focus has mainly been on the collection and semantic analysis of provenance information. Little work has been done with respect to the trustworthiness of data provenance. Evaluating the trustworthiness of data provenance requires answering questions like "From which sources did the data originate from? How trustworthy are such data sources? Which entities (applications

or users or systems) handled the data? Are these entities trustworthy?" To address these challenges, Dai et al. [1] have proposed a *data provenance trust model* (trust model, for short) which estimates the confidence level of data and the trustworthiness of data sources. In what follows, we first survey such a trust model and then we discuss open research issues.

A Data Provenance Trust Model. The model developed by Dai et al. [1] takes into account three different aspects that may affect data trustworthiness: data similarity, data conflict, and path similarity. Similar data items are considered as support to one another, while conflicting data items compromise the confidence level of one another. Besides data similarity and data conflict, the source of the data is also an important factor for determining the trustworthiness of the data. For example, if several independent sources provide the same data, such data is most likely to be true. We also observe that a data is likely to be true if it is provided by trustworthy data sources, and a data source is trustworthy if most data it provides are true. Due to such inter-dependency between data and data sources, the model is based on an iterative procedure to compute the scores to be assigned to data and data sources. To start the computation, each data source is first assigned an initial trust score which can be obtained by querying available information about data sources. At each iteration, the confidence level of the data is computed based on the combined effects of the aforementioned three aspects, and the trustworthiness of the data source is recomputed by using the confidence levels of the data it provides. When a stable stage is reached, that is, when changes to the scores (of both data and data sources) are negligible, the trust computation process stops.

Table 1. An example data set

RID	SSN	Name	Gender	Age	Location	Date
1	479065188	Tom	Male	38	Los Angeles	3pm 08/18/2007
2	47906518	Tom	Male	38	Los Angeles	3pm 08/18/2007
3	479065188	Bob	Male	38	New York	7pm 08/18/2007
4	4790651887	Tom	Male	38	Pasadena	3pm 08/18/2007

Data similarity in this model refers to the likeness of different items. Similar items are considered as supportive to each other. The challenge here is how to determine whether two items are similar. Consider the example in Table 1. We can observe that the first two items are very similar since they both report the same locations of Tom at the same date. The only difference between these two items is a possible typo error in the person's SSN. In contrast, the third item is different from the first two because it reports a totally different location. In order to determine sets of data items that are very similar likely describe the same real world item, the model employs a clustering algorithm. The clustering process results in a set of item; each such set represents a single real-world data item. For each item the effect of data similarity on its confidence score is determined in terms of the number of items in the same cluster and the size of the cluster. The use of clustering techniques requires developing distance functions to measure the similarities among items and the cost function which the clustering algorithm tries to minimize. The distance functions are usually determined by the

type of data being clustered, while the cost function is defined by the specific objective of the clustering problem. Well known approaches can be used for such functions, such as the edit distance for string data types, or hierarchy-based distances for categorical data.

Data conflict refers to inconsistent descriptions or information about the same entity or event. A simple example of a data conflict is that the same person appears at different locations during the same time period. It is obvious that data conflict has a negative impact on the confidence level of items. There are various reasons for data conflicts, such as typos, false data items generated by malicious sources, or misleading knowledge items generated by intermediate parties. Data conflict largely depends on the knowledge domain of the specific application. Therefore, the trust model by Dai et al. [1] allows users to define their own data conflict functions according to their application-dependent requirements. To determine if two items conflict with each other, data users first need to define the exact meaning of conflict, which we call *data consistency rules*. Consider the example in Table 1 again. The attribute value of "SSN" in the first item is the same as that in the third item, but the attribute value of "Name" in the first item is different from that in the third one. This implies a data conflict, since we know that each single SSN should correspond to only one individual. We can further infer that there should be something wrong with either source providers (airports) or intermediate agents (police stations) whichever handled these two items. The data consistency rule we would use in this example is that *if* $r1$("*SSN*") = $r2$("*SSN*"), *then* $r1$("*Name*") = $r2$("*Name*") (such rule can be simply modeled as functional dependency). If two items cannot satisfy the condition stated by such data consistency rule, these two items are considered conflicting with each other; if two items conflicts, their confidence level will in general be lower. To facilitate automatic conflict detection, a simple language needs to be provided allowing data users and/or domain experts to define data consistency rules; such language can then be implemented by using the trigger and assertion mechanisms provided by DBMS.

Path similarity models how similar are the paths followed by two data items from the sources to the destination. Path similarity is important in that it is used to evaluate the provenance independence of two or more data items. A path in the model by Dai et al. [1] is represented by a list of identifiers; the first element of such list is the data source, whereas the subsequent elements are the identifiers of all the intermediate parties that processed and/or simply retransmitted the data. The similarity of two paths is computed by comparing the lists corresponding to these paths.

Open Research Issues. The above model is still preliminary and requires addressing several research issues which we discuss in what follows.

- *Similarity/dissimilarity of data.* A key component of the model is represented by the factors concerning the similarity/dissimilarity of data. In addition to the techniques mentioned in the above paragraph, like the edit distance, one needs to include modeling techniques able to take into account data semantics. For example, consider the fourth item in Table 1; such item can be considered very similar (or supportive of the first two items) by observing that Pasadena is part of the Los Angeles area. Such an inference requires taking into account knowledge about spatial relationships in the

domain of interest. Possible approaches that can be used include semantic web techniques, like ontologies and description logics.

- *Secure data provenance*. An important requirement for a data provenance trust model is that the provenance information be protected from tampering when flowing across the various parties. In particular, we should be able to determine the specific contribution of each party to the provenance information and the type of modification made (insert/delete/update). We may also have constraints on what the intermediate parties processing the data and providing provenance information can see about provenance information from previous parties along the data provisioning chain. An approach to address such problem is based on approaches for controlled and cooperative updates of XML documents in Byzantine and failure-prone distributed systems [20]. One could develop an XML language for encoding provenance information and use such techniques to secure provenance documents.
- *Data validation through privacy-preserving record linkage*. In developing solutions for data quality, the use of record linkage techniques is important. Such techniques allow a party to match, based on similarity functions, its own records with records by another party in order to validate the data. In our context such techniques could be used not only to match the resulting data but also to match the provenance information, which is often a graph structure. Also in our case, we need not only to determine the similarity for the data, but also the dissimilarity for the provenance information. In other words, if two data items are very much similar and their provenance information is very dissimilar, the data item will be assigned a high confidence level. In addition, confidentiality of provenance information is an important requirement because a party may have relevant data but have concerns or restrictions for the data use by another party. Thus application of record linkage technique to our context thus requires addressing the problem of privacy, the extension to graph-structured information, and the development of similarity/dissimilarity functions. Approaches have been proposed for privacy-preserving record linkage [22, 23]. However those approaches have still many limitations, such as the lack of support for graph-structured information.
- *Correlation among data sources*. The relationships among the various data sources could be used to create more detailed models for assigning trust to each data source. For example, if we do not have good prior information about the trustworthiness of a particular data source, we may try to use distributed trust computation approaches such as EigenTrust [10] to compute a trust value for the data source based on the trust relationships among data sources. In addition, even if we observe that the same data is provided by two different sources, if these two sources have a very strong relationship, then it may not be realistic to assume that the data is provided by two independent sources. An approach to address such issue is to develop "source correlation" metrics based on the strength of the relationship among possible data sources. Finally, in some cases, we may need to know "how important is a data sources within our information propagation network?" to reason about possible data conflicts. To address such issue one can apply various social network centrality measures such as degree, betweenness, closeness, and information centralities [21] to assign importance values to the various data sources.

5 Policies

Policies provide a high level mechanism for expressing organizational requirements and policies and simplify administration and management. In our context, a key type of policy is represented by the *confidence policy*, regulating the use of the data according to requirements concerning data confidence levels. Such a policy is motivated by the observation that the required level of data trustworthiness depends on the purpose for which the data have to be used. For example, for tasks which are not critical to an organization, like computing a statistical summary, data with a medium confidence level may be sufficient, whereas when an individual in an organization has to make a critical decision, data with high confidence are required. An interesting example is given by Malin et al. [17] in the context of healthcare applications: for the purpose of generating hypothesis and identifying areas for further research, data about cancer patients' diseases and primary treatment need not be highly accurate, as treatment decisions are not likely to be made on the basis of these results data alone; however, for evaluating the effectiveness of a treatment outside of the controlled environment of a research study, accurate data is desired. In what follows, we first survey a policy model [2] addressing such requirement and then discuss open research issues.

A Confidence Policy Model. A policy in the confidence policy model by Dai et al. [2] specifies the minimum confidence that has to be assured for certain data, depending on the user accessing the data and the purpose the data access. In its essence, a confidence policy contains three components: a *subject specification*, denoting a subject or set of subjects to whom the policy applies; a *purpose specification*, denoting why certain data are accessed; a *confidence level*, denoting the minimum level of confidence that has to be assured by the data covered by the policy when the subject to whom the policy applies requires access to the data for the purpose specified in the policy. In this policy model, subjects to which policies apply are assumed to be roles of a RBAC model, because this access control model is widely used and well understood. However such policy model can be easily extended to the case of attribute-based access control models, as we discuss in the research issues paragraph.

The confidence policies are thus based on the following three sets: R, Pu and $R+$. R is a set of roles used for subject specification; a user is human being and a role represents a job function or job title within the organization that the user belongs to. Pu is a set of data usage purposes identified in the system. $R+$ denotes non-negative real numbers. The definition of a confidence policy is thus as follows.

[*Confidence Policy*]. Let $r \in R$, $pu \in Pu$, and $\beta \in R+$. A confidence policy is a tuple $< r, pu, \beta >$, specifying that when a user under a role r issues a database query q for purpose pu, the user is allowed to access the results of q only if these results have confidence value higher than β.

The confidence policies <Secretary, summary, 0.1>, and <Manager, investment, 0.8> specify, respectively, that a secretary can use data with low confidence value for the purpose a summary reports, whereas a manager must use data with high confidence value when making investment decisions.

Open Research Issues. The development of policies for integrity management requires however addressing several research issues which we discuss in what follows.

- *Expressive power.* A first issue is related to improve the expressivity of the confidence policy model. The simple model outlined in the previous paragraph needs some major extensions. It should be possible to support a more fine-grained specification of confidence requirements concerning data use whereby for a given task and role, one can specify different confidence levels for different categories of data. The model should support the specification of subjects, in terms of subject attributes and profiles other than the subject role. If needed, exceptions to the policies should be supported; a possible approach is to support *strong policies*, admitting no exceptions, and *weak policies*, admitting exceptions. If exceptions are allowed for a policy (set of policies), the policy enforcement mechanism should support the gathering of evidence about the need for exceptions. Such evidence is crucial in order to refine confidence policies.
- *Policy provisioning*. An important issue is related to the confidence policy provisioning. Provisioning refers to assigning confidence policies to specific tasks, users, and data and, because it very much depends from the applications and data semantics, it may be quite difficult. To address such issue, one approach is the use of machine learning techniques.
- *Data validation policies*. Even though confidence policies have a key role in our framework, policies are also required to manage data validation activities, periodically or whenever certain events arise. To address such requirement, one possibility is to design a *data validation policy* (DVP) language that includes the following components: (i) A set of events that trigger the execution of some validation actions. An event can be a data action (read, insert, delete, update) or a user-defined event such as a specific time or a particular situation; for example a source of certain data has been found to be invalid and thus the validation process needs to determine which data, users and application programs may have been affected by the invalid data. Notice that it should also be possible to specify that a validation must be executed before any access is made by a given subject or set of subjects, or even when the data is being accessed (see also next section). (ii) A validation procedure which performs the actual integrity analysis of the data. Such procedure may be complex in that it can involve human users and may result in a number of actions, possibly organized according to a workflow. (iii) A set of actions to be undertaken as consequence of the validation. A large variety of actions are possible, such as blocking all accesses to data, blocking the execution of an application program, invoking some data repair procedures, making changes to the metadata associated with the data. It is important to notice that even though it would be desirable to perform data validation very frequently, the impact on performance may be significant; therefore the availability of a language, like the DVP language, will make easier for the data administrators to fine tune the system according to trade-offs among performance, cost and integrity.

6 Policy Complying Query Evaluation

A critical issue in enforcing confidence policies in the context of query processing is that some of the query results may be filtered out due to confidence policy violation. Therefore a user may not receive enough data to make a decision and he may want to improve the data quality. A possible approach [2] is based on dynamically incrementing the data confidence level. Such approach selects an optimal strategy which determines which data should be selected and how much the confidence should be increased to comply with the confidence level stated by the policies. Such approach assumes that each data item in the database is associated with a cost function that indicates the cost for improving the confidence value of this data item. Such a cost function may be a function on various factors, like time and money. As part of the approach several algorithms have been investigated to determine the increment that has the lowest cost. In what follows we first briefly discuss components of the system proposed by Dai et al. [2] and then we discuss open research issues.

A Policy Complying Query Evaluation System. The query evaluation system (see Figure 1) consists of four main components: *query evaluation*, *policy evaluation*, *strategy finding*, and *data quality improvement*. We elaborate on the data flow within the system. Initially, each base tuple is assigned a confidence value by the *confidence assignment* component (based on the model described in Section 4). A user inputs query information in the form <*Q*, *purpose*, *perc*>, where *Q* is a normal SQL query, *purpose* indicates the purpose for which the data returned by the query will be used, and *perc* indicates percentage of results that the user expects to receive after the confidence policy enforcement. The *query evaluation* component then computes the results of *Q* and the confidence level of each tuple in the result based on the confidence values of base tuples. The intermediate results are sent to the *policy evaluation* component. The *policy evaluation* component selects the confidence policy associated with the role of the user who issued *Q* and checks each tuple in the query result according to the selected confidence policy. Only the results with confidence value higher than the threshold specified in the confidence policy are immediately returned to the user. If less than *perc* of the results satisfy the confidence policy, the *strategy finding* component is invoked to devise an optimal strategy for increasing the confidence values of the base tuples and report the cost of such strategy to the user. If the user agrees about the cost, the *strategy finding* component will inform the *data quality improvement* component to take actions to improve the data quality and then update the database. Finally, new results will be returned to the user.

Open Research Issues. The development of policy complying query evaluation framework requires addressing several research issues which we discuss in what follows.

- *Cost models*. In the above discussion, we assumed that cost models for the quality improvement are available for each tuple. However suitable cost models need to be developed, also depending on the optimization criteria adopted, like time and financial cost.

- *Heuristics for confidence increases*. Algorithms need to be devised for determining suitable base tuples for which the increase in the confidence values can lead to the minimum cost. Because it is likely that finding the optimal solution may be computationally very expensive, heuristics need to be devised.
- *Query based data validation strategies*. Data validation and correction is often an expensive activity and thus needs to be minimized and executed when high-confidence data are actually required. Also because data validation and correction may take some time, approaches must be in place to make sure that the data are corrected by the time they are needed. To address such issue, approaches must be devised that, based on knowing in advance the queries issued by the users and the timeline of these queries, are able to determine which data must be validated and corrected while at the same time minimizing the costs.

7 Conclusions

In this paper we have discussed research directions concerning the problem of providing data that can be trusted to end-users and applications. This is an important problem for which multiple techniques need to be combined in order to achieve good solutions. In addition to approaches and ideas discussed in the paper, many other issues needed to be addressed to achieve high-assurance data trustworthiness. In particular, data need to be protected from attacks carried through unsecure platforms, like the operating system, and unsecure applications, and from insider threats. Initial solutions to some of those data security threats are starting to emerge.

Acknowledgments. The authors have been partially supported by the AFOSR grant FA9550-07-0041 "Systematic Control and Management of Data Integrity, Quality and Provenance for Command and Control Applications".

References

1. Dai, C., Lin, D., Bertino, E., Kantarcioglu, M.: An Approach to Evaluate Data Trustworthiness based on Data Provenance. In: Jonker, W., Petković, M. (eds.) SDM 2008. LNCS, vol. 5159, pp. 82–98. Springer, Heidelberg (2008)
2. Dai, C., Lin, D., Kantarcioglu, M., Bertino, E., Celikel, E., Thuraisingham, B.: Query Processing Techniques for Compliance with Data Confidence Policies. Technical Report, CERIAS (submitted for publication) (2009)
3. Biba, K.J.: Integrity Considerations for Secure Computer Systems. Technical Report TR-3153, Mitre (1977)
4. Clark, D.D., Wilson, D.R.: A Comparison of Commercial and Military Computer Security Policies. In: IEEE Symposium on Security and Privacy, Oakland, CA (1987)
5. Eckerson, W.: Data Warehousing Special Report: Data quality and the bottom line (2002), `http://www.adtmag.com/article.aspx?id=6321`
6. Juran, J.M.: Juran on Leadership for Quality – an Executive Handbook. Free Press, New York (1989)
7. Kahn, B., Strong, D., Wang, R.: Information Quality Benchmarks: Product and Service Performance. In: Communications of the ACM, vol. 45, pp. 184–192. ACM, New York (2002)

8. Price, R., Shanks, G.: A Semiotic Information Quality Framework. In: IFIPInternational Conference on Decision Support Systems: Decision Support in anUncertain and Complex World, Prato, Italy (2004)
9. Wand, Y., Wang, R.Y.: Anchoring Data Quality Dimensions in Ontological Foundations. In: Communications of the ACM, vol. 39, pp. 86–95. ACM, New York (1996)
10. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The EigenTrust Algorithm for Reputation Management in P2P Networks. In: Twelfth International World Wide Web Conference, pp. 640–651. ACM, New York (2003)
11. Dalvi, N.N., Suciu, D.: Efficient Query Evaluation on Probabilistic Databases. In: Thirtieth International Conference on Very Large Data Bases, pp. 864–875. Morgan Kaufmann, San Fransisco (2004)
12. Ferraiolo, D.F., Sandhu, R.S., Gavrila, S.I., Kuhn, D.R., Chandramouli, R.: Proposed NIST Standard for Role-Based Access Control. In: ACM Transactions on Information and System Security, vol. 4, pp. 224–274. ACM, New York (2001)
13. Simmhan, Y.L., Plale, B., Gannon, D.: A Survey of Data Provenance Techniques. Technical Report, Indiana University, Bloomington (2007)
14. Buneman, P., Khanna, S., Tan, W.C.: Why and where: A characterization of data provenance. In: Van den Bussche, J., Vianu, V. (eds.) ICDT 2001. LNCS, vol. 1973, pp. 316–330. Springer, Heidelberg (2000)
15. Lanter, D.P.: Design of a Lineage-Based Meta-Data Base for GIS. Cartography and Geographic Information Systems 18, 255–261 (1991)
16. Greenwood, M., Goble, C., Stevens, R., Zhao, J., Addis, M., Marvin, D., Moreau, L., Oinn, T.: Provenance of e-Science Experiments – Experiences from Bioinformatics. In: UK OST e-Science All Hands Meeting (2003)
17. Malin, J.L., Keating, N.L.: The Cost-Quality Trade-off: Need for Data Quality Standards for Studies that Impact Clinical Practice and Health Policy. Journal of Clinical Oncology 23, 4581–4584 (2005)
18. Ramakrishnan, R., Gehrke, J.: Database Management Systems. McGraw-Hill, New York (2000)
19. Levien, R.: Attack Resistant Trust Metrics. PhD thesis, University of California, Berkeley, CA, USA (2002)
20. Mella, G., Ferrari, E., Bertino, E., Koglin, Y.: Controlled and cooperative updates of XML documents in byzantine and failure-prone distributed systems. In: ACM Transactions on Information and System Security, vol. 9, pp. 421–460. ACM, New York (2006)
21. Jackson, M.O.: Social and Economics Networks. Princeton University Press, Princeton (2008)
22. Batini, C., Scannapieco, M.: Data Quality: Concepts, Methodologies and Techniques. Springer, Heidelberg (2006)
23. Scannapieco, M., Figotin, I., Bertino, E., Elmagarmid, A.: Privacy Preserving Schema and Data Matching. In: ACM SIGMOD International Conference on Management of Data, pp. 653–664 (2007)
24. Inan, A., Kantarcioglu, M., Bertino, E., Scannapieco, M.: A Hybrid Approach to Private Record Linkage. In: 24th IEEE International Conference on Data Engineering, pp. 496–505 (2008)