J Intell Inf Syst
DOI 10.1007/s10844-014-0322-7

A roadmap for privacy-enhanced secure data provenance

Elisa Bertino · Gabriel Ghinita · Murat Kantarcioglu · Dang Nguyen · Jae Park · Ravi Sandhu · Salmin Sultana · Bhavani Thuraisingham · Shouhuai Xu

Received: 13 September 2012 / Accepted: 2 January 2014
© Springer Science+Business Media New York 2014

Abstract The notion of data provenance was formally introduced a decade ago and has since been investigated, but mainly from a functional perspective, which follows the historical pattern of introducing new technologies with the expectation that security and privacy can be added later. Despite very recent interests from the cyber security community on some specific aspects of data provenance, there is no long-haul, overarching, systematic framework for the security and privacy of provenance. The importance of secure provenance R&D

E. Bertino (✉) · S. Sultana
Purdue University, West Lafayette, IN, USA
e-mail: bertino@cs.purdue.edu

S. Sultana
e-mail: ssultana@purdue.edu

M. Kantarcioglu · B. Thuraisingham
University of Texas at Dallas, Richardson, TX, USA

M. Kantarcioglu
e-mail: muratk@utdallas.edu

B. Thuraisingham
e-mail: thuraisingham@utdallas.edu

G. Ghinita
University of Massachusetts, Boston, USA
e-mail: gabriel.ghinita@umb.edu

R. Sandhu · J. Park · S. Xu · D. Nguyen
University of Texas, San Antonio, USA
e-mail: ravi.sandhu@utsa.edu

J. Park
e-mail: jae.park@utsa.edu

S. Xu
e-mail: shxu@utsa.edu

D. Nguyen
e-mail: dnguyen@cs.utsa.edu

Published online: 31 May 2014

has been emphasized in the recent report on Federal game-changing R&D for cyber security especially with respect to the theme of Tailored Trustworthy Spaces. Secure data provenance can significantly enhance data trustworthiness, which is crucial to various decision-making processes. Moreover, data provenance can facilitate accountability and compliance (including compliance with privacy preferences and policies of relevant users), can be an important factor in access control and usage control decisions, and can be valuable in data forensics. Along with these potential benefits, data provenance also poses a number of security and privacy challenges. For example, sometimes provenance needs to be confidential so it is visible only to properly authorized users, and we also need to protect the identity of entities in the provenance from exposure. We thus need to achieve high assurance of provenance without comprising privacy of those in the chain that produced the data. Moreover, if we expect voluntary large-scale participation in provenance-aware applications, we must assure that the privacy of the individuals or organizations involved will be maintained. It is incumbent on the cyber security community to develop a technical and scientific framework to address the security and privacy challenges so that our society can gain maximum benefit from this technology. In this paper, we discuss a framework of theoretical foundations, models, mechanisms and architectures that allow applications to benefit from privacy-enhanced and secure use of provenance in a modular fashion. After introducing the main components of such a framework and the notion of provenance life cycle, we discuss in details research questions and issues concerning each such component and related approaches.

Keywords Data provenance · Security · Privacy · Trustworthy computing · Cryptography and access control · Risk management · Accountability and compliance

1 Introduction

The notion of data provenance was formally introduced a decade ago (Buneman et al. 2000; 2001) and has since seen increasing interest (Cheney et al. 2009; Moreau 2009; Moreau et al. 2008), especially as our society undergoes unprecedented data explosion, fusion, mining, computation, brokering, sales and theft. While debate continues over nuances and precise meaning (Moreau 2009) there is consensus that at its core data provenance deals with assurances regarding the sources and processes by which data appeared where it now resides or is in use. The value of data provenance has been motivated in multiple domains, including science (Sahoo et al. 2008; Simmhan et al. 2005), business (Curbera et al. 2008), cyber security (Hui et al. 2010; Moitra et al. 2009), and healthcare (Hajnal et al. 2007; Kifor et al. 2006). More recently researchers have noted its benefits in newer domains such as sensor networks (Liu et al. 2010), web mashups (Groth et al. 2009), social networks (Golbeck 2006), data forensics (Lu et al. 2010) and data streams (Vijayakumar and Plale 2006). The importance of secure provenance R&D has been emphasized in the recent report on Federal game-changing R&D for cyber security (Networking et al. 2010), and its predecessor (Networking et al. 2009), especially with respect to the theme of Tailored Trustworthy Spaces.

Secure data provenance offers many benefits. It can significantly enhance data trustworthiness. It can facilitate accountability and compliance, including compliance with privacy preferences and policies of relevant users. Provenance can be an important factor in access control and usage control decisions. It can be valuable in data forensics. It can enhance protection against data leakage by supporting anomaly detection in the flows of information across different systems. Provenance also enhances the meta-data that one can associate with data, thus extending the knowledge that one can extract from data. Along with these potential benefits, data provenance also poses a number of vexing security and privacy challenges. Integrity of provenance is a prerequisite to any kind of assurance that can be placed in it. Sometimes provenance needs to be confidential so it is visible only to properly authorized users. Further, there are privacy challenges in protecting the identity of entities in the provenance from exposure even to authorized users of the provenance information. How can we achieve high assurance of provenance without comprising privacy of those in the chain that produced the data? Moreover, if we expect voluntary large-scale participation in provenance-aware applications how do we assure the individuals or organizations involved that their privacy will be maintained?

As our society becomes increasingly data dependent and data driven, the role and importance of data provenance will only become more prominent. To date, however, there are no comprehensive approaches to provenance management that also assure secuity and privacy of provenance. Devising such approaches entails addressing many challenges. The goal of this paper is to propose a systematic and comprehensive framework for provenance management based on which the various challenges can be identified, discussed, and addressed.

The rest of the paper is organized as follows: Section 2 gives an overview of the foundational framework we envision for privacy-enhanced secure provenance, followed by Section 3 that surveys related work. Section 4, the main part of the paper, highlights the open research challenges with respect to several aspects such as modeling provenance, provenance access control, privacy, integrity, accountability and risk management. Section 5 outlines architectures for privacy-enhanced secure provenance systems at the operating system and data layers. Finally, we conclude in Section 6.

2 A foundational framework

When designing a framework for the secure and privacy-preserving management of provenance it is important to note that security and privacy are not conducive to absolutes. There are complex tradeoffs between the strength of security and privacy guarantees and the assurance with which provenance can be used towards the various ends discussed above. One possible approach is to address such tradeoffs within a risk management perspective. The ultimate goal is to isolate applications and higher-level services from the details of securing provenance, and to provide well-defined platform-independent standardized interfaces to secure provenance services.

The modular structure of the proposed framework is illustrated in Fig. 1. The four pillars of secure provenance, namely Access Control, Privacy, Integrity, and Accountability, are exposed through a uniform and platform-independent interface to the application layer.

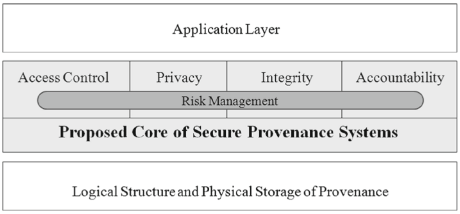

Application Layer
Access Control
Privacy
Integrity
Accountability
Risk Management
Proposed Core of Secure Provenance Systems
Logical Structure and Physical Storage of Provenance

Fig. 1 Framework for privacy-enhanced, secure provenance

A Risk Management infrastructure will span all four aspects. The functionality of secure provenance is encapsulated within a Secure Provenance Core System that consists of software libraries and runtime systems that oversee the generation, storage, dissemination and processing of provenance. The Core layer itself can be built on top of a generic provenance representation in the form of a Directed Acyclic Graph (DAG), as discussed in Section 4.1. From a functional perspective, the proposed framework covers the entire lifecycle of data and their associated provenance. As shown in Fig. 2, we classify data and provenance lifecycle into the following procedures: generation, storage, dissemination, processing, all within the span of accountability and compliance.

3 Related work

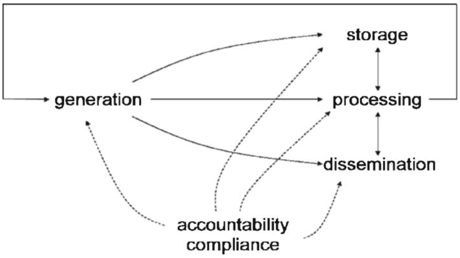
There is a considerable body of research in the general area of data provenance (Moreau 2009). So far, research has mainly focused on the *functional* aspect of provenance management, including the Provenance-aware Storage Systems (PASS) project (Muniswamy-Reddy et al. 2006), the Trio project (Agrawal et al. 2006), the Orchestra project (Green et al. 2007; Ives et al. 2005; Taylor and Ives 2006) and others (Bowers et al. 2006; Cohen et al. 2006; Davidson et al. 2007; Golbeck and Hendler 2008; Groth et al. 2006a; Simmhan et al. 2008). However, the problem of security and privacy encountered in provenance management was not explored until very recently (Braun et al. 2008; Hasan et al. 2007). These discussions have led to a number of studies: for example, ensuring provenance integrity was studied in the setting of file systems in (Hasan et al. 2009), and in the setting of database systems in (Zhang et al. 2009). Furthermore, the on-going End-to-End Provenance System effort (McDaniel et al. 2010) proposes using trusted monitors to enhance the security of provenance at the host level, whereas the relevance of Trusted Computing to provenance management was emphasized in (Lyle and Martin 2010).

To our knowledge, our proposed framework is the first to comprehensively and systematically consider privacy, security, accountability, integrity, sharing and risk issues in managing provenance for multiple purposes, paving the path to game-changing advances in this arena. Specific related research is further identified in the different parts of this paper that focus on particular challenges.

4 Research roadmap

This section discusses research challenges and related approaches concerning the various components of our proposed foundational framework. It begins with research on enhancing

Fig. 2 Lifecycle of data and provenance



the typical DAG model for provenance data, followed by the four pillars of provenance: Access Control, Privacy, Integrity and Accountability, and concludes with the cross-cutting theme of Risk Management.

4.1 Security enhancements of provenance DAG model

The provenance model is the first key element of a provenance framework. Such a model essentially specifies which are the types of entities that compose provenance and it is the basis on which the mechanisms for provenance collection and use are built. Most provenance models are typically based on the notion of directed acyclic graph (DAG), such as the one illustrated in Fig. 3. In the DAG associated with a given data item, each node represents an entity that accessed or modified the data item or its provenance. An entity could be a user, an organization, a process, etc., or some combination thereof. An edge from X to Y signifies that data was sent directly from X to Y. Operations on data are shown as labels associated with each node, e.g., concatenation at node D and set difference at node F so the result DI consists of data items received from node D less those from node E. No_Op denotes that data is simply forwarded without performing any operation. Entities such as C and E are called forwarders. Entities such as D and F which perform data operations are said to be active. Keeping track of forwarders can be useful, for instance, for accountability in the context of data leakage. To find a candidate set of entities that may have caused leakage consideration of forwarders is required. In other contexts forwarders may not matter and the provenance graph can be pruned. Nodes may also include additional information, such as the context (e.g. time, application, operating system and so on) in which an entity has accessed or modified the data item (Sultana and Bertino 2012). A major research challenge consists of identifying and formalizing the security and privacy implications of the DAG model and of the data manipulation, provenance manipulation and provenance querying operations. In other words the challenge is to develop, formalize and analyze a general security-enhanced provenance DAG model. The DAG model has been considered previously as the de-facto standard in representing provenance (Chapman et al. 2008; Heinis and Alonso 2008). However, previous work is concerned with issues such as efficiency of representation with low storage requirements, and does not address provenance security and privacy.

In order to develop suitable privacy and security techniques for provenance models, it is important to take into account three major categories of operations, that is: (i) data manipulation operations and their associated provenance modification operations, (ii) provenance modification operations which modify provenance without changing the underlying data,

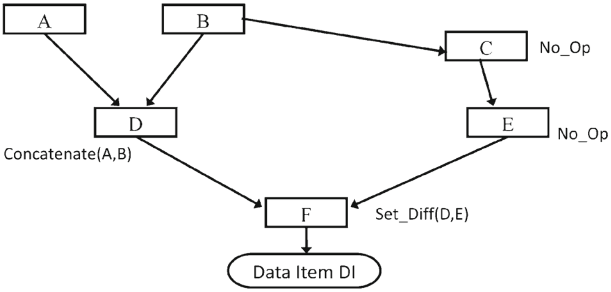


Fig. 3 DAG Representation for Provenance of Data Item DI

and (iii) provenance query operations which look up provenance information for some purpose. Data manipulation operations must be properly captured within the provenance DAG through corresponding provenance modification operations. Data manipulation can range from simple operations such as data concatenation, to complex ones such as statistical tests to determine whether two datasets support each other in a scientific workflow. The DAGs of the operand datasets must be combined, along with information about the operation, to capture the complete provenance. Data operations that remove sensitive keywords may result in the data being more accessible downstream. Conversely, incorporation of sensitive data may increase data sensitivity making it less accessible downstream. Provenance manipulation can occur without data manipulation as in the case of forwarders discussed above. In some cases the forwarder's identity may simply be added to the provenance DAG. In other cases the DAG may be pruned or the identity of entities in some of its nodes modified. For example, while a document is in preparation a detailed provenance DAG may be maintained. When the document is exported the provenance may be abbreviated to attribute the document to an organizational unit rather than to specific individuals. Clearly provenance manipulation operations must be suitably authorized.

For provenance to be useful, querying it securely is essential. Queries may be subject to access control since portions of the provenance may be confidential. It may be necessary to perform queries on encrypted provenance DAGs in a privacy preserving manner. Relevant provenance query operations include the following: Is a specific entity or a specific kind of entity, such as *Professor*, part of a given DAG? Has a certain operation, such as removal of sensitive keywords, been performed on the data? What is the number of nodes or edges in the DAG? The last could be useful in the context of k-anonymity (Samarati and Sweeney 1998), whereby a researcher can view the results of a survey by hospital patients only after a minimum number of patients have completed it.

4.2 Access control

To protect against the disclosure of sensitive provenance information, we need to be able to define fine grained access control policies. In order to define an access control policy for provenance, it is imperative that we identify the parts of the provenance graph that we want to protect. Therefore, we must have a clear definition of the users, their actions and the resources to be protected. As we have discussed, provenance takes the form of a directed acyclic graph (DAG) that establishes causal relationships between data items. Traditional access control models focus on individual data items whereas in provenance we are concerned with protecting both, the data items and their relationships (Braun et al. 2008). In order to protect a resource we need to first identify it in the provenance graph. This identification is one of the major distinguishing factors between a provenance access control model and existing access control models. The challenge here is thus the definition of new access control languages and enforcement mechanisms specific for securing provenance.

Some research has been devoted to the study of access control in provenance. These include the work by Braun et al. (2008), which emphasizes the need for a separate security model for provenance. This work also points out that existing access control models do not support the directed acyclic graph of provenance. In Syalim et al. (2009), the authors present an access control method for provenance over a directed acyclic graph. They build their access control model over a relational database which controls access to nodes and edges. They apply a grouping strategy to the provenance graph to create resources that need to be protected. Instead, we suggest the idea of grouping by defining dynamic paths that are evaluated at query time based on incorporating regular expressions in our policies. In

Corcoran et al. (2007), the authors propose a grouping of provenance into blocks, and then applying a labeling strategy over these blocks. They also provide a language, SELinks, to encode their security policies. However such approach does not support fine-grained access control on provenance.

A possible approach is to extend the access control language by Ni et al. (2009) with support for regular expressions. Such language also incorporates other features of a general access control language such as support for fine-grained access control over the indivisible parts of a provenance graph, and integration of existing access control policies. This provenance language was developed as a generalized model of access control for provenance, but did not address resources with arbitrary path lengths within the provenance graph. Therefore, it suffers from the fact that a resource must be identified before hand, rather than be given as a string which is matched against the graph at execution time.

As an example of use of regular expressions, consider the sample policy given in Fig. 4. In this example, the subject element can be the name of a user or any collection of users, e.g. scientist or senior-scientist, or a special user collection anyuser which represents all users. The record element is the name of a resource. The restriction element is an (optional) element which refines the applicability established by the subject or record. The scope element is an (optional) element which is used to indicate whether the target applies only to the record or its entire ancestry. The condition element is an (optional) element that describes under what conditions access is to be given or denied to a user. The effect element indicates the policy author's intended consequence for a true evaluation of a policy.

The scope element is useful, in particular, when we want to protect the record only if it is along a specified path in the provenance graph. This is achieved by using the predefined value "non-transferable". This element can also be used when we need to protect a path in the provenance graph if a particular record is along that path. This is achieved by the predefined value "transferable". The condition element is necessary when we want to specify system or context parameters for giving access, e.g. permitting access to the provenance when it is being used for research. We achieve fine-grained access control by allowing a record value to be any (indivisible) part of a provenance graph. The regular expressions in the "restriction" element allow us to define policies over paths of arbitrary length in a provenance graph that apply to a subject or record.

A major issue in the design of an access control language based on regular expression is efficiency, as policies need to be efficiently evaluated at run-time when enforcing access control. One approach to address this issue is to adopt a SPARQL (Perez et al. 2009; PrudHommeaux et al. 2006) based framework that can evaluate regular expressions on z RDF graphs efficiently. In Cadenhead et al. (2011b, 2012), it is shown that such approach could scale to large amounts of data.

```
<policy ID="1" >
   <target>
     <subject>anyuser</subject>
     <record>Doc1_2</record>
     <restriction>
        Doc1_2 [WasGeneratedBy] process AND
        process [WasControlledBy] scientist|senior-scientist
     </restriction>
     <scope>non-transferable</scope>
  </target>
  <condition>purpose == research</condition>
  <effect>Permit</effect>
</policy>
```

Fig. 4 Policy language

Another approach for achieving access control protection to provenance is through utilizing special constructs labeled as named abstractions of dependency path patterns (Nguyen et al. 2012a). Dependency path patterns are regular expression-based patterns of edges that represent the causality dependencies between the essential entities (acting users, action processes and data objects) involved in system transactions. Capturing the DAG-like nature of provenance data after such causality dependencies is the essence of the Open Provenance Model (Moreau et al. 2011), which has gained wide acceptance. Abstracting the dependencies in application-specific manners can provide a solid foundational framework upon which simple yet effective access control can be built. Abstraction constructs, labeled dependency names, can be easily utilized in policy specifications for such pursue.

In this approach, Park et al. (2012) propose using these dependency names as a unit for access request as well as a control unit in a multi-layer access control model for provenance data. In essence, there need be protection mechanisms on the dependency names themselves as the first layer. An additional layer can be formed to regulate finer-grained access control to the provenance graph entities that can be covered by the dependency path patterns. More specifically, the set of resulting nodes, reachable by a query that embeds the path patterns, are further processed with vertex-level policies before access decisions are made.

In addition to access control protection of provenance data, Park et al also propose using the dependency constructs to utilize provenance data for access control protection of the underlying data. They term the approach Provenance-based Access Control (Park et al. 2012). In this approach, the dependency names and corresponding dependency path patterns are stored in the system. Dependency names are used in policy specifications while dependency path patterns are embedded in queries to be evaluated against the provenance graphs that are stored in the system. For deciding an access request, the request information and the resulting nodes obtained from executing a path query are evaluated altogether against the policy rule. The PBAC approach is capable of enhanced utilities such as origin-based access control, dynamic separation of duties, etc. Furthermore, PBAC enables simple and effective policy specifications that elevates access control security.

While the framework proposed by Park et al can easily and effectively be deployed in a system with a single source of provenance management, seamless usage of provenance data for access control purposes is not readily achieved in an environment where multiple separate provenance systems are in play. Particularly in a distributed environment, each local system, which captures, stores and maintains its own provenance data, may opt not to share certain sensitive provenance information for various reasons. In such cases, an access request to a local system data, which involve policies that require information from remote systems, will always be falsely disallowed. In order to address this issue, Nguyen et al propose two potential approaches, use of cascading query and sticky provenance data, and demonstrate their usages in a provenance-aware group-centric collaboration scenario (Nguyen et al. 2012b; Park et al. 2011).

A cascading query is essentially a recomputed version of a local PBAC query that can be sent to a remote system to request provenance information. Depend on the use case, such a query comprises a combination of modified components including new target data object, pruned dependency path pattern and revised policy rules. The second approach involves the concept of sticky provenance data, which is the provenance information associated with a local object, being transferred along with the object whenever a cross-system information flow takes place. Sticky provenance data allow appropriate access control decisions to be made without having to inquire provenance information from a remote system. However, complications can arise with sticky provenance data usage. In particular, it may be necessary but potentially infeasible to employ mechanisms to keep a transmitted sticky provenance data up-to-date with the local provenance system status. Furthermore, the complexity of system configuration and management of sticky provenance data can grow unmanageably as the number of involved local and remote systems grow.

4.3 Privacy

Privacy of data and of its associated provenance are closely related. One may be compromised through release of the other. Such cross inferences are among the many challenging aspects of protecting provenance privacy. Figure 5 shows the four possible cases with the following characteristics.

Q1. *Private Data and Private Provenance* This is the most demanding case, e.g., a grade change for a student may need to be privacy protected as data and as private provenance of how the change came about.

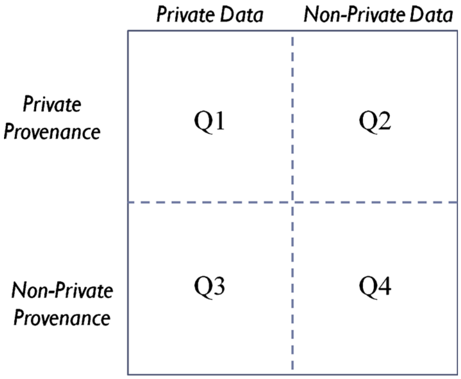
Q2. *Non-Private Data but Private Provenance* Consider a whistleblower in a company that does not properly deploy pollution controls. While the information itself should be made public, the identity of the whistleblower as provenance should be kept protected.

Q3. *Private Data but Non-Private Provenance* In a scientific data repository, some data may originate at a cancer institute, including information that relates to individuals. The cancer institute is known as a regular contributor to the data repository as advertised on its web site. Nevertheless, the fact that specific data originated at the institute is a good indicator of the characteristics of the data (i.e., cells may be harvested from cancer patients), thereby compromising individuals' privacy.

Q4. *Non-Private Data and Non-Private Provenance* This case does not have any privacy requirements, but it can help identify common provenance models and formats. Thereby, if some data or their associated provenance require privacy at some later point in time, the transition can be facilitated without major redesign.

Several data privacy models have been proposed, including k-anonymity (Samarati and Sweeney 1998), ℓ-diversity (Machanavajjhala et al. 2006), t-closeness (Li et al. 2007) and differential privacy (Dwork 2008). These deal with relational data, and it is not clear whether they can be extended to provenance DAGs. Furthermore, none of these takes into account

Fig. 5 Quadrant Representation of Privacy for Data and Provenance



	Private Data	Non-Private Data
Private Provenance	Q1	Q2
Non-Private Provenance	Q3	Q4

the relationship between data and provenance and the potential privacy leaks of one when the other is released.

The main challenge is to develop privacy paradigms, models and metrics suitable for provenance, in context of the above quadrants. In what follows we discuss two approaches: sanitization and cryptography.

Sanitization Approach With the DAG model discussed in Section 4.1, we can sanitize provenance by retaining a subgraph or by generalizing information about entities in some nodes. Sanitization can be done before release or on demand in response to provenance queries. In either case we need sanitization policies that indicate which pieces of the provenance DAG are sensitive and how they can be pruned or generalized. One technique could be to use regular expressions built on the DAG model to represent sensitive paths or nodes. For example, we can specify a sanitization policy where any path that includes a sensitive source should be eliminated. Such regular expressions could also be extended to include properties of the sources, e.g., delete name of the sources if the source is human. The development of a sanitization approach requires addressing several issues. The first issue is the design of policy languages for provenance sanitization, specifying the specific sanitization strategy to be used for the provenance associated with a specific data item. One possible approach is to develop regular-expression based sanitization languages and explore their expressive power and limitations with respect to practical sanitization policies. In Cadenhead et al. (2011b), such approach have been proposed. However other alternatives, such as context free grammar based sanitization languages, need to be investigated. The second issue is related to the sanitization policy composition and policy conflicts. As multiple sanitization policies could apply to the same data, efficient implementation of combined policies and conflict resolution are required. For regular-expression based policies one can adapt common algorithms for finite automaton composition. Efficient enforcement of sanitization policies is also especially important if sanitization happens on demand in real time. To address such real time requirements, different parts of the provenance graph could be tagged with privacy information in advance (e.g., Rachapalli et al. 2012). Finally, limiting provenance sanitization based inference attacks is critical in that sanitized provenance can be combined with background information to infer potentially sensitive information. Since general inference attack is hard to prevent, one needs to explore a risk-based approach to address this problem. By using graph mining techniques, one can analyze whether we can infer some sanitized paths using the existing disclosed provenance information.

Cryptographic Approach Cryptography offers the attractive property of provable security. The DAG-nature of provenance poses new cryptographic challenges for privacy-preserving processing. For instance, how can we determine whether a certain sequence of nodes and operations exist in an encrypted DAG? Or, how can a policy decision point (PDP) determine whether to grant access without authority to access the entire provenance? Intuitively we need encryption schemes that allow appropriate query processing with respect to the plaintext. One research direction to address such issues is to extend existing cryptographic techniques for privacy-preserving queries. In particular, existing encrypted search schemes (e.g., Boneh et al. 2004; Boneh and Waters 2007) can be extended for application to DAG search, and existing attribute-encryption schemes (e.g., Goyal et al. 2006; Lewko et al. 2010; Ostrovsky et al. 2007; Pirretti et al. 2010) can be applied to address the above PDP-decision problem, with acceptable performance. In addition it is critical to devise new cryptographic techniques specialized to support queries on encrypted provenance, including: (i) tailored cryptographic schemes for specific purposes such as encrypting DAG graphs (i.e., their adjacency matrices) while allowing fast processing of DAG nodes identities (e.g., statistical analysis of frequent nodes in a set of encrypted provenance DAG graphs as proposed for other relevant privacy-preserving applications (Kiayias et al. 2008)); (ii) techniques that exploit adjacency-list (rather than adjacency-matrix) representation of DAGs for efficiency; (iii) special key management schemes to achieve better trade-offs between privacy and efficiency.

4.4 Integrity

Perhaps the most fundamental goal of provenance is to guarantee integrity of the data sources and history. Simple application of digital signatures for this purpose can quickly become unwieldy resulting in linear increase in provenance information proportional to size of the provenance DAG (as in Section 4.1). Techniques to avoid this linear increase intuitively require some kind of aggregated signatures, which makes it difficult to simultaneously offer privacy of data source(s) and history and accountability (e.g., techniques such as group signatures (Chaum and van Heyst 1991) are not applicable). The research challenge is to design integrity mechanisms that ensure data source integrity and data history integrity, without these afore-mentioned problems.

Data source and history provenance have been investigated but without considering their security (see, e.g., Buneman et al. 2001; Cheney 2007; Groth 2007).

Hasan et al. (Hasan et al. 2009) investigated the integrity of total-order operation-chain provenance of files in file systems. This was extended to partial-order operation provenance of records and tables in database systems (Zhang et al. 2009). These techniques are neither secure enough nor efficient enough for our purposes (Xu et al. 2010). Also, existing cryptographic techniques (e.g., aggregate signatures (Lysyanskaya et al. 2004)) do not adequately address the problem of provenance integrity because one needs to maintain the DAG-structure (Xu et al. 2010).

At a high level, a suitable approach to integrity requires the novel integration of new cryptographic signing techniques and new cryptographic aggregation techniques. Ideally the resulting solutions meet the following requirements. (i) Security is proven based on standard cryptographic assumptions. (ii) Schemes operate in the weakest Plain Public-Key (PPK) model (Bellare and Neven 2006). (iii) Schemes are non-interactive. (iv) Schemes incur light-weight communication and computational complexities.

Addressing these desiderata requires research along different directions that we discuss in what follows.

1. *Design Families of Cryptographic Methods for Assuring Data Source Provenance Integrity* We motivate and explain this task with two example scenarios. Because a message M endorsed by multiple users might be treated as more trustworthy, we need multisignatures whereby a set of ℓ users/sources can digitally sign M while producing a short signature. Via a new trick to defeat the rogue-key attack inherent to the above-mentioned PPK model and the knowledge of the users identities we (Qian and Xu 2010) very recently constructed a multisignature scheme that is significantly more efficient than the previous best scheme (Boneh et al. 2003). We are investigating whether it is possible to further make the resulting signature verification complexity independent of ℓ via the following idea (inspired by the notion of proxy re-signatures (Ateniese and Hohenberger 2005; Blaze et al. 1998; Libert and Vergnaud 2008)) without incurring two-way interactions between the signers: let each user sign the identities of the ℓ users via an appropriate signature scheme (e.g., a variant of Waters signature (Waters 2005))

so as to allow $user_i$ to translate $user_{i-1}$'s signature into $user_i$'s signature, where $1 < i \leq \ell$. It may be possible then to use $user_1$ and $user_\ell$'s signatures on $M||(user_1, \ldots, user_\ell)$ as a multisignature.

The other scenario is to design editable signature schemes as illustated in the example of Fig. 6. Suppose Alice drafted a message M_1, signed it with her private key, and sent Bob M_1 as well as her signature on M_1. Now, Bob may partially edit M_1 by adding to, and/or *modifying* some portion of, M_1 because he has more accurate information in question. Denote by M_2 the resulting new message, which consists of two parts M_{2A} and M_{2B}, where M_{2A} can be a proper portion of M_1 and M_{2B} is contributed by Bob. How should message M_2 be digitally signed so that the receiver of M_2 can verify its integrity using Alice's and Bob's public keys?

A multi-party sequentially-editable signature scheme has been recently proposed (Qian and Xu 2011), which however requires $O(\ell)$ pairing operations to verify a signature. More efficient schemes are thus needed.

2. *Design Families of Cryptographic Methods for Assuring Data History Provenance Integrity* Consider the example in Fig. 7, where nodes $P_1, \ldots, P_6$ are users.

   Suppose P_1 and P_2 produce and disseminate messages M_1 and M_2, respectively. Suppose P_3 receives M_1 from P_1 and edits M_1 to produce M_3. Suppose P_4 receives M_1 and M_2 from P_1 and P_2, respectively, and edits them to produce a new message M_4. Moreover, P_5 receives M_3 and M_4 from P_3 and P_4, respectively, and edits them to produce M_5. Finally, P_6 receives M_5 and wants to authenticate its history which is the DAG in Fig.7. Ensuring data history provenance will help the users evaluate the trustworthiness of the received messages. At a high level, this requires to ensure the integrity of subgraph that ends at the user in question. This problem was recently formalized and an initial scheme proposed (Xu et al. 2010), where the signature size is proportional to the size of the DAG in question. Alternative more efficient schemes are needed.
3. *Incorporate Privacy and Accountability* Enhancing the above schemes with privacy and accountability is an important research direction. Recent approaches based on group signatures (Ding et al. 2009; Kiayias et al. 2008; Tsudik and Xu 2003; Xu and Yung 2009) help tackle the afore-mentioned linear-increase barrier.

4.5 Accountability

Provenance can be used to enforce accountability and verify compliance. For example, in the case of scientific data, it is important to hold researchers accountable for the data they produce, to avoid the acceptance and propagation of erroneous or fabricated results. At the same time, there may be a need to assure the privacy of the scientists and security of the underlying data. The obvious challenge is how to enforce accountability policies while preserving privacy and security. Using provenance for accountability (e.g., Demsky 2009; Hasan et al. 2009; Weitzner D.J. et al. 2008) has been explored in the past. However, none

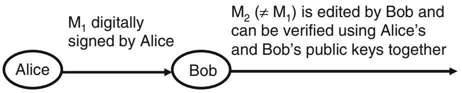
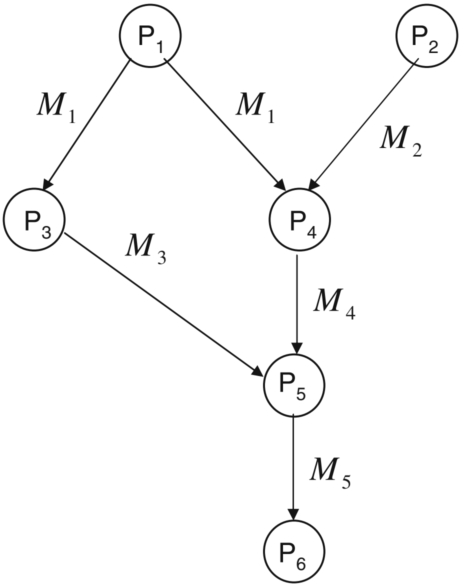

M$_1$ digitally signed by Alice
M$_2$ (≠ M$_1$) is edited by Bob and can be verified using Alice's and Bob's public keys together
Alice
Bob

Fig. 6 An example of two-party sequentially-editable signature: M_2 is edited and "cleaned" by Bob

Fig. 7 Example of data history



of the existing work addresses the privacy issues in enforcing accountability policies using provenance.

For cases where accountability policy requires one to reveal provenance, we may try to limit the privacy disclosure risk. For example, for drug discovery, some of the provenance information needs to be published to convince the public that experiments were conducted under appropriate guidelines. In such cases, we may try to reveal the minimum privacy-sensitive provenance information to satisfy accountability policies. One way to address this problem is to extend the sanitization algorithms discussed in Section 4.3 to search for possible sanitizations that satisfy the accountability policies. In some cases, it is likely that there may be no sanitization approach that can satisfy the accountability policies. In these situations, we may want to check whether the policies are satisfied without revealing the data. To address this problem, a possible approach is to rely on approaches that use secure multi-party computation (SMC) techniques for privacy-preserving data mining (Kantarcioglu and Clifton 2004; Kantarcioglu and Kardes 2009). Finally, there may be situations where we can reveal some sanitized provenance to show that part of the accountability policy is satisfied. For the remaining conditions in the accountability policy, we may use SMC techniques to verify that they are satisfied.

The development of accountability solutions thus requires research along the following directions.

1. *Develop a Language for Accountability Policies* An accountability policy language allows one to specify the provenance information that are required to accountability. A suitable starting point is the accountability language given in Cederquist et al. (2005) which however would need to be extended in order to be compatible with DAG based data provenance model.
2. *Develop Algorithms for Accountability Aware Sanitization* Sanitization search algorithms need to be designed that search the possible sanitization domain (e.g., generalization and deletion of the nodes and edges in the provenance graph) to find

sanitizations that satisfy the accountability policies. As in general the search space may be very large, as part of this search, heuristics must be devised (e.g., generalize sensitive attributes first etc.) to prune or reduce the search space.

3. *Secure Multi-Party Based Accountability Verification* Given an accountability policy, one has to determine whether it is satisfied without revealing sensitive provenance. This in turn requires to determine the basic secure operations that are needed for building such a secure verification component. For example, basic secure protocols such as secure equality, comparison, dot product and set operations most likely suffice to compose complex verification protocols. However research is needed to determine whether other basic secure protocols to build secure verifications are required and to efficiently compose the basic protocols to build a system to efficiently verify large accountability policies.

4.6 Risk management

In order to be compliant with various accountability and other policies, one may need to release provenance information. For those cases where we have to release provenance, it is important to understand the risks involved. Therefore a risk management model is needed based on which one can estimate the likely impact of releasing provenance. To date, various risk management techniques have been applied to access control (e.g., Celikel et al. 2007; Dimmock et al. 2004) and data privacy (e.g., Hong et al. 2004) to mitigate potential risks. However, there is very limited work that considers risk management issues in data provenance (e.g., Cadenhead et al. 2011a)

In order to evaluate the risks of revealing certain provenance information, we need to consider the possible background information (B_i) that the attacker can combine with the provenance to infer sensitive information.

Given this possible background information that can be used by an attacker, an inference function I, and a loss function L, we can estimate the expected loss of revealing certain provenance information. If the expected loss is bigger than the expected utility of releasing the provenance, we may choose not to release it. We can state this approach formally as follows.

1. $\exists B_i \; i = \{1, 2, 3, \ldots, n\}$ a set of most likely backgrounds. Each B_i has a corresponding probability Pr_i that captures the likelihood of an attacker having background information $B_i \in B$, the set of all probable backgrounds.
2. $I : K \times B \rightarrow D$, an inference function that an attacker uses to combine the provenance information released $k \in K$ with background information $B_i \in B$ to infer sensitive information.
3. $L : D \rightarrow \mathbb{R}$, a loss function (where $\mathbb{R}$ is the set of real numbers). $U : K \rightarrow \mathbb{R}$, a utility function that measures the utility of releasing provenance information. The expected loss is then $E[L(d)] = \sum_{i=1}^{n} L(d) \times Pr(B = B_i)$ and determines whether or not provenance will be released.

The development of a suitable risk management approach for provenance requires research along different directions.

1. *Understanding Utility of Provenance* A suitable risk management model for provenance requires understanding of the utility of releasing provenance. Since provenance (e.g., for scientific research data) is used to establish trust to domain information (e.g., experimental results) one can start with a mechanism that assigns trust values to information

based on the provenance released (Dai et al. 2009). Under this scenario, the higher the trust value, the higher the utility of the released provenance. However research is needed that explores the various trust computation metrics based on provenance to see their effect in utility function definitions.

2. *Background Information Modeling* Another important component of a risk management model is the background information that could be used by the attacker. To model this background information, one approach is to create different scenarios where we assume that the adversary knows certain parts of the provenance. Such information could then be used for modeling the adversary's background information. To estimate probabilities needed in a risk management approach, one needs to explore possible probability distribution functions (e.g., power law distributions) where one can assign lower probabilities to larger amounts of background information.
3. *Inference Attack Modeling* Finally, it is important to understand how the attacker could combine all the available data to infer sensitive information. To address this problem, one can explore probabilistic inference models (e.g., Bayesian Belief Networks) and logic base inference models (e.g., first order logic).

5 Architectures for secure provenance management systems

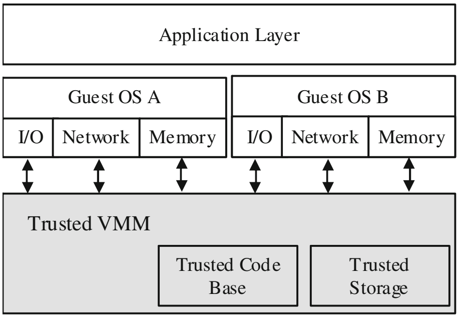
The actual implementation of systems able to collect, store, and make available provenance information while at the same time addressing privacy and security is not a trivial task, also because provenance can be captured at different layers in a computer system. Which layer is relevant may depend on the specific provenance requirements by the application domain. In what follows, in order to discuss architectural issues, we consider two different cases concerning deployment of a provenance system; the first case focuses on provenance at the operating system layer, whereas the second focuses on the data layer in a distributed, loosely-coupled XML document dissemination environment.

5.1 OS layer

One promising approach for mechanisms for privacy-enhanced and secure collection, storage and querying of provenance at the operating system (OS) layer is based on the use of trusted virtual machine monitors (VMMs). Such VMMs run at a higher privilege level than the OS and control the proper generation and manipulation of provenance. The concept of data forensics at the OS level using virtualization has been recently addressed in (Krishnan et al. 2010), which is the first attempt to comprehensively collect information flows from disk, memory and processes using a trusted VMM. The approach monitors accesses to objects on disk and follows the causal chain of these accesses across processes, even after the objects are copied into memory. However, provenance security and privacy are not addressed (e.g., there is no consideration for access control, privacy, risk management, etc). Previously, other approaches such as Antfarm (Jones et al. 2006a) and Geiger (Jones et al. 2006b) took a similar monitoring approach at the VMM level, but considered only partial information collection, either at the OS process level or at the memory level.

An architectural design that addresses the shortcomings of previous approaches is shown in Fig. 8. The fundamental component of the design is the trusted VMM that intercepts all data manipulation operations within the guest operating systems above, including changes to data in memory, data transfers using I/O and transfers over the network. The VMM securely annotates data with provenance information. The VMM trusted base includes the

Fig. 8 Prototype Architecture for Secure Provenance at OS Layer



trusted code base as well as an amount of trusted storage holding cryptographic keys as well as verification objects necessary to check if provenance has been tampered with.

The development of a prototype based on such architectural design requires however to tackle several difficult challenges, such as:

- *Deciding the Data Granularity* Choosing the appropriate granularity at which to capture and record data provenance is crucial for the feasibility of the prototype. A too coarse granularity (e.g., file-level) may restrict functionality, for instance by denying access to entire documents. On the other hand, a too fine granularity (e.g., byte-level) may be difficult to capture efficiently.
- *Performance Overhead* Virtualization-based security architectures do require an additional overhead, and impose certain limitations on the amount of computation and storage that can be performed within the trusted domain. It is thus critical to investigate optimizations that allow collection and processing of provenance without severely affecting system performance. In particular mechanisms need to be developed that allow verification of authenticity with only a small amount of trusted storage required (e.g., hash trees where only the root needs to be signed and stored securely). In addition, methods need to be devised that allow provenance to be stored in a split fashion in both trusted and untrusted storage. Secure reconstruction of split-stored provenance will be done using efficient erasure codes (e.g., digital fountain codes (Luby 2002)).

5.2 Data layer

The design of a provenance system suitable for a distributed system disseminating XML documents requires addressing specific challenges. Due to the loosely coupled nature of the system, solutions that rely on strongly trusted entities, such as Virtual Machine Monitors, are not suitable. Furthermore, it may be difficult to control the process of generation, storage and processing at a remote party. For the above reasons, we will discuss in what follows an approach based on a combination of cryptography and sanitization. Provenance may be encrypted using advanced cryptographic primitives that allow processing directly on top of ciphertexts. In addition, plaintext data may be shared in sanitized form, e.g., using generalization. For instance, within an organization, provenance for a certain document may consist of person (author), supervisor, department, and organization name. If the document leaves the organization, only department and organization information may be disclosed.

Several systems have been developed for provenance-aware dissemination of data in large-scale distributed environments. The PASOA system (Groth et al. 2006b) developed for the UK e-Science program focuses on tracking provenance for over one hundred scientific projects related to distributed data, computation and collaboration. PASOA defines the provenance recording protocol (PReP) that specifies provenance actions on the invocation of services. myGrid (Stevens et al. 2003) also serves the e-Science community, and provides assertion-based provenance tracking mechanisms for a Grid environment specialized in biological sciences. Finally, the Karma system (Simmhan et al. 2006) develops a generic framework for recording provenance in distributed workflows. None of these systems addresses in detail provenance security and privacy requirements.

Figure 9 shows the architecture of a possible system for provenance-centric document dissemination. There are two parties involved: one that queries provenance (in this setting a query can represent updates as well), and a provenance provider party, which is entrusted with secure provenance storage and processing. Note that, the provider need not necessarily be a fully trusted party, since the provenance is stored in encrypted form, and processing is done on ciphertexts. However, the provider is trusted to correctly execute the protocols for processing and enforce appropriate policies, which can be enforced through mechanisms such as attestation or secure co-processors. Both parties interact through a middleware module that encapsulates secure provenance functionality such as encoding of data and queries. Queries received at the provider are dispatched to a processing engine which incorporates privacy-aware features. Before performing provenance operations, the provider will ensure that the client that sent the query has the proper credentials and authorization to perform the operation. This is achieved through the policy enforcement point. Provenance is securely kept in a provenance store, in either sanitized or encrypted form. All accesses to provenance are governed by policies that dictate access control decisions, the sanitization level that must be applied before disclosure, etc.

The design and implementation of such an architectural design requires addressing a number of challenges:

- *Restriction on Supported Operations* Encryption provides strong protection guarantees, but also limits the flexibility of processing. Typically, only simple operations and queries can be performed on encrypted data. Even though recent breakthroughs in homomorphic encryption (Gentry 2009) provide promising results for processing on ciphertexts, the current solutions are still far from practical due to their overhead.
- *Performance Overhead* Advanced cryptographic functionality that allows processing on top of ciphertexts often requires the use of very expensive primitives. Currently, such primitives are orders of magnitude slower than conventional encryption. Techniques need to be investigated that allow parallelization of computation, and approximations devised for the required operations so that they can be implemented more efficiently with the existing cryptographic primitives. Also, the extensive use of encryption would require efficient mechanisms for large-scale key management.

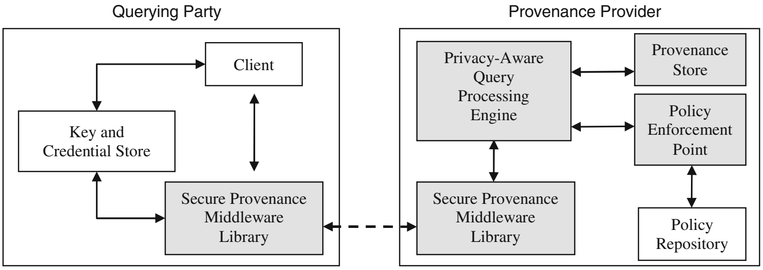


Fig. 9 Prototype Architecture for Secure Provenance at Data Layer

6 Concluding remarks

Any system or application aiming at making an intelligent use of data needs to rely on provenance to enhance the quality of the data, to assess the trustworthiness of data, and to enhance the knowledge extracted from data. This is crucial when data is originated from many different sources and may be transformed in various ways before being accessed by the end-users or applications. In order to rely on provenance one needs to assure that the provenance can be trusted and also however that the privacy and confidentiality of the provenance and the data sources be assured whenever needed. In this paper we have presented a comprehensive research roadmap that identifies research challenges that need to be addressed in order to come up with secure and efficient solutions for provenance management. In the paper we have also outlined preliminary approaches to some of these challenges. An important observation to make is that many of these solutions may be further enhanced by the use of knowledge-based techniques, such as ontologies, and machine learning techniques. For example risk management, which is a crucial component in approaches for secure provenance management, may benefit from ontological descriptions of risks as well as subjects and objects and actions that subjects may execute on the objects. Machine learning techniques may help with the definition and evolution of the different policies involved in secure provenance management, such as access control policies and accountability policies. Finally, intelligent comparison techniques may be used for provenance comparison, to determine for example if two data items have followed closer provenance paths in a distributed system. As future work we plan to investigate many of the research challenges discussed in this paper.

Acknowledgments The work reported in this paper has been partially supported by NSF under grant CNS-1111512.

References

Agrawal, P., Benjelloun, O., Sarma, A., Hayworth, C., Nabar, S., Sugihara, T., Widom, J. (2006). In *VLDB* (pp. 1151–1154).

Ateniese, G., & Hohenberger, S. (2005). In *ACM conference on computer and communications security (CCS'05)* (pp. 310–319).

Bellare, M., & Neven, G. (2006). In *ACM conference on computer and communications security (CCS'06)* (pp. 390–399).

Blaze, M., Bleumer, G., Strauss, M. (1998). In *EUROCRYPT'98date* (pp. 127–144).

Boneh, D., & Waters, B. (2007). Theory of cryptography, (pp. 535–554).

Boneh, D., Gentry, C., Lynn, B., Shacham, H. (2003). In *EUROCRYPT'03* (pp. 416–432).

Boneh, D., Di Crescenzo, R., Ostrovsky, R., Persiano, G. (2004). In *Advances in Cryptology-Eurocrypt 2004* (pp. 506–522). Springer.

Bowers, S., McPhillips, T., Ludäscher, B., Cohen, S., Davidson, S. (2006). In *International provenance and annotation workshop (IPAW)* (pp. 133–147).

Braun, U., Shinnar, A., Seltzer, M. (2008). In *Proceedings of the 3rd conference on hot topics in security USENIX association* (p. 4).

Buneman, P., Khanna, S., Tan, W.C. (2000). FST TCS 2000: Foundations of software technology and theoretical computer science. In S. Kapoor & S. Prasad (Eds.) *Lecture notes in computer science* (vol. 1974, pp. 87–93). Berlin: Springer. doi:10.1007/3-540-44450-5_6.

Buneman, P., Khanna, S., Tan, W.C. (2001). Database Theory-ICDT, (pp. 316–330).

Cadenhead, T., Kantarcioglu, M., Thuraisingham, B. (2011a). In *3th USENIX workshop on the theory and practice of provenance (USENIX Association), TaPP'11.*

Cadenhead, T., Khadilkar, V., Kantarcioglu, M., Thuraisingham, B. (2011b). In *Proceedings of the first ACM conference on data and application security and privacy, CODASPY '11* (pp. 133–144). New York: ACM. doi:10.1145/1943513.1943532.

Cadenhead, T., Khadilkar, T., Kantarcioglu, M., Thuraisingham, B. (2012). In *Proceedings of the 17th ACM symposium on access control models and technologies, SACMAT '12* (pp. 113–116) New York: ACM. doi:10.1145/2295136.2295157.

Cederquist, J., Conn, R., Dekker, M., Etalle, S., den Hartog, J. (2005). In *Sixth IEEE international workshop on policies for distributed systems and networks* (pp. 34–43). doi:10.1109/POLICY.2005.5.

Celikel, E., Kantarcioglu, M., Thuraisingham, B., Bertino, E. (2007). In *Proceedings of the 2007 OTM confederated international conference on the move to meaningful internet systems: CoopIS, DOA, ODBASE, GADA, and IS - Volume Part II, OTM'07* (pp. 1548–1566). Berlin / Heidelberg: Springer-Verlag. http://portal.acm.org/citation.cfm?id=1784707.1784750.

Chapman, A.P., Jagadish, H.V., Ramanan, P. (2008). *Proceedings of the 2008 ACM SIGMOD international conference on management of data* (pp. 993–1006).

Chaum, D., & van Heyst, E. (1991). In *EUROCRYPT* (pp. 257–265).

Cheney, J. (2007). *IEEE Data Engineering Bulletin*, *30*(4), 22.

Cheney, J., Chong, S., Foster, N., Seltzer, M., Vansummeren, S. (2009). In *Proceeding of the 24th ACM SIGPLAN conference companion on object oriented programming systems languages and applications, OOPSLA '09* (pp. 957–964). New York: ACM. doi:10.1145/1639950.1640064.

Cohen, S., Boulakia, S., Davidson, S. (2006). In *Third international workshop on data integration in the life sciences (DILS)* (pp. 264–279).

Corcoran, B., Swamy, N., Hicks, M. (2007). In *On-line proceedings of the workshop on principles of provenance (PrOPr)* (Citeseer).

Curbera, F., Doganata, Y., Martens, A., Mukhi, N., Slominski, A. (2008). On the move to meaningful internet systems: OTM, (pp. 100–119).

Dai, C., Lin, D., Kantarcioglu, M., Bertino, E., Celikel, E., Thuraisingham, B.M. (2009). In *Secure data management* (pp. 49–67).

Davidson, S., Boulakia, S., Eyal, A., Ludascher, B., McPhillips, T., Bowers, S., Anand, M., Freire, J. (2007). *IEEE Data Engineering Bulletin*, *30*(4), 44.

Dimmock, N., Belokosztolszki, A., Eyers, D., Bacon, J., Moody, K. (2004). In *Proceedings of the ninth ACM symposium on access control models and technologies, SACMAT '04* (pp. 156–162). New York: ACM. doi:10.1145/990036.990062.

Demsky, B. (2009). In *Proceedings of the 4rd conference on hot topics in security (USENIX Association).*

Ding, X., Tsudik, G., Xu, S. (2009). *Journal of Computer Security*, *17*(4), 489.

Dwork, C. (2008). In *TAMC* (pp. 1–19).

Gentry, C. (2009). In *Proceedings of the 41st annual ACM symposium on theory of computing, STOC '09* (pp. 169–178).

Golbeck, J. (2006). Provenance and annotation of data. In L. Moreau & I. Foster (Eds.), *Lecture notes in computer Science* (vol. 4145, pp. 101–108). Berlin: Springer. doi:10.1007/11890850_12.

Golbeck, J., & Hendler, J. (2008). *Concurrency and Computation: Practice and Experience*, *20*(5), 431.

Goyal, V., Pandey, O., Sahai, A., Waters, B. (2006). In *ACM Conference on computer and communications security* (pp. 89–98).

Green, T., Karvounarakis, G., Ives, Z., Tannen V. (2007). In *VLDB*.

Groth, P. (2007). The origin of data: Enabling the determination of provenance in multi-institutional scientific systems through the documentation of processes.Ph.D. thesis University of Southampton. http://eprints.ecs.soton.ac.uk/14649/1/ThesisSubmitted.pdf.

Groth, P., Jiang, S., Miles, S., Munroe, S., Tan, V., Tsasakou, S., Moreau, L. (2006a). An architecture for provenance systems. Technical report, University of Southampton. http://eprints.ecs.soton.ac.uk/13216/1/provenanceArchitecture10.pdf.

Groth, P., Jiang, S., Miles, S., Munroe, S., Tan, V., Tsasakou, S., Moreau, L. (2006b). In *Technical report D3.1.1, Ver 0.6*, www.pasoa.org.

Groth, P., Miles, S., Moreau, L. (2009). *ACM Transactions Internet Technology*, *9*(3), 1. doi:10.1145/1462159.1462162.

Hajnal, A., Kifor, T., Pedone, G., Varga, L. (2007). In *Proceedings of HealthGrid 2007* (pp. 330–341).

Hasan, R., Sion, R., Winslett, M. (2007). In *Proceedings of the 2007 ACM workshop on storage security and survivability, StorageSS '07* (pp. 13–18). New York: ACM. doi:10.1145/1314313.1314318.

Hasan, R., Sion, R., Winslett, M. (2009). In *Proceedings of the 7th conference on file and storage technologies (FAST'09)* (pp. 1–14).

Heinis, T., & Alonso, G. (2008). In *Proceedings of the 2008 ACM SIGMOD international conference on management of data* (pp. 1007–1018).

Hong, J.I., Ng, J.D., Lederer, S., Landay, J.A. (2004). In *Proceedings of the 5th conference on designing interactive systems: Processes, practices, methods,and techniques, DIS '04* (pp. 91–100). New York: ACM. doi:10.1145/1013115.1013129.

Hui, P., Bruce, J., Fink, G., Gregory, M., Best, D., McGrath, L., Endert, A. (2010). In *International symposium on collaborative technologies and systems (CTS)* (pp. 489–498). doi:10.1109/CTS.2010.5478473.

Ives, Z., Khandelwal, N., Kapur, A., Cakir, M. (2005). In *CIDR* (pp. 107–118).

Jones, S.T., Arpaci-Dusseau, A.C., Arpaci-Dusseau, R.H. (2006a). In *Proceedings of the annual conference on USENIX '06 annual technical conference* (pp. 1–1).

Jones, S.T., Arpaci-Dusseau, A.C., Arpaci-Dusseau, R.H. (2006b). *SIGOPS Operations Systematics Review*, *40*, 14.

Kantarcioglu, M., & Clifton, C. (2004). *IEEE TKDE*, *16*(9), 1026. http://ieeexplore.ieee.org/iel5/69/29187/01316832.pdf?isnumber=29187&prod=JNL&arnumber=1316832&arnumber=1316832&arSt=+1026&ared=+1037&arAuthor=Kantarcioglu%2C+M.%3B+Clifton%2C+C..

Kantarcioglu, M., & Kardes, O. (2009). *International Journal of Information and Computer Security*, *2*(353). doi:10.1504/IJICS.2008.022488. http://www.ingentaconnect.com/content/ind/ijics/2009/00000002/00000004/art00002.

Kiayias, A., Xu, S., Yung, M. (2008). In *Proceedings of 6th international conference security and cryptography for networks (SCN'08). Lecture notes in computer science* (vol. 5229, pp. 57–76). Springer.

Kifor, T., Varga, L., Vazquez-Salceda, J., Alvarez, S., Willmott, S., Miles, S., Moreau, L. (2006). *IEEE Intelligent Systems*, *21*(6), 38. DOI 9D04F813-E31E-416F-99B7-DBC4D177ACA7.

Krishnan, S., Snow, K.Z., Monrose, F. (2010). In *Proceedings of the 17th ACM conference on computer and communications security* (pp. 50–60).

Lewko, A., Okamoto, T., Sahai, A., Takashima, K., Waters, B. (2010). In *EUROCRYPT* (pp. 62–91).

Li, N., Li, T., Venkatasubramanian, S. (2007). In *ICDE*.

Libert, B., & Vergnaud, D. (2008). In *ACM conference on computer and communications security 2008* (pp. 511–520).

Liu, Y., Futrelle, J., Myers, J., Rodriguez, A., Kooper, R. (2010). In *2010 international symposium on collaborative technologies and systems (CTS)* (pp. 330–339). doi:10.1109/CTS.2010.5478496.

Lu, R., Lin, X., Liang, X., Shen, X.S. (2010). In *Proceedings of the 5th ACM symposium on information, computer and communications security, ASIACCS '10* (pp. 282–292). New York: ACM. doi:10.1145/1755688.1755723.

Luby, M. (2002). In *Annual IEEE symposium on foundations of computer science* (p. 271).

Lyle, J., & Martin, A. (2010). In *2nd USENIX workshop on the theory and practice of provenance (TaPP 10)*.

Lysyanskaya, A., Micali, S., Reyzin, L., Shacham, H. (2004). Advances in cryptology - EUROCRYPT. In C. Cachin & J. Camenisch (Eds.), *Lecture notes in computer science* (vol. 3027, pp. 74–90). Springer.

Machanavajjhala, A., Gehrke, J., Kifer, D., Venkitasubramaniam, M. (2006). In *ICDE*.

McDaniel, P., Butler, K., McLaughlin, S., Sion, R., Zadok, E., Winslett, M. (2010). In *2nd USENIX workshop on the theory and practice of provenance (TaPP 10)*.

Moitra, A., Barnett, B., Crapo, A., Dill, S. (2009). In *Military communications conference, MILCOM 2009. IEEE* (pp. 1–7). doi:10.1109/MILCOM.2009.5379854.

Moreau, L. (2009). Foundations and trends in web science. http://eprints.ecs.soton.ac.uk/18176/1/psurvey.pdf.

Moreau, L., Groth, P., Miles, S., Vazquez-Salceda, J., Ibbotson, J., Jiang, S., Munroe, S., Rana, O., Schreiber, A., Tan, V., Varga, L. (2008). *Communications of the ACM*, *51*, 52. doi:10.1145/1330311.1330323.

Moreau, L., Clifford, B., Freire, J., Futrelle, J., Gil Y., Groth, P., Kwasnikowska, N., Miles, S., Missier, P., Myers, J., Plale, B., Simmhan, Y., Stephan, E., den Bussche, J.V. (2011). *Future Generation Computer Systems*, *27*(6), 743. doi:10.1016/j.future.2010.07.005. http://www.sciencedirect.com/science/article/pii/S0167739X10001275.

Muniswamy-Reddy, K., Holland, D., Braun, U., Seltzer, M. (2006). In *Proceedings of the 2006 USENIX annual technical conference* (pp. 43–56).

Networking, F., Research, I.T., Program, D.N. (2009). (September 2009). http://www.nitrd.gov/pubs/CSIA_IWG_%Cybersecurity_%20Gamechange_RD_%20Recommendations_20100513.pdf.

Networking, F., Research, I.T., Program, D.N. (2010). (May 2010). http://www.nitrd.gov/pubs/CSIA_IWG_%Cybersecurity_%20Gamechange_RD_%20Recommendations_20100513.pdf.

Nguyen, D., Park, J., Sandhu, R. (2012a). In *4th USENIX workshop on the theory and practice of provenance (USENIX Association), TaPP'12*.

Nguyen, D., Park, J., Sandhu, R. (2012b). In *2012 IEEE international Conference on information reuse and integration (IRI)*.

Ni, Q., Xu, S., Bertino, E., Sandhu, R., Han, W. (2009). Secure data management (pp. 68–88).

Ostrovsky, R., Sahai, A., Waters, B. (2007). In *ACM Conference on Computer and Communications Security* (pp. 195–203).

Park, J., Nguyen, D., Sandhu, R. (2011). In *7th international conferenceon collaborative computing: Networking applications and worksharing (CollaborateCom)* (pp. 221–230).

Park, J., Nguyen, D., Sandhu, R. (2012). In *10th annual conference on privacy, security and trust (IEEE), PST 2012*.

Pirretti, M., Traynor, P., McDaniel, P., Waters, B. (2010). *Journal of Computer Security*, *18*(5), 799.

Perez, J., Arenas, M., Gutierrez, C. (2009). *ACM Transactions on Database Systems (TODS)*, *34*(3), 1.

PrudHommeaux, E., Seaborne, A., et al. (2006). *W3C working draft*, *4*.

Qian, H., & Xu, S. (2010). Information Processing Letter (accepted in 2010).

Qian, H., & Xu, S. (2011). In *To appear in the Proceedings of First ACM Conference on Data and Application Security and Privacy (ACM CODASPY'11)*.

Rachapalli, J., Kantarcioglu, M., Thuraisingham, B. (2012). In *4th USENIX workshop on the theory and practice of provenance (USENIX Association), TaPP'12*.

Sahoo, S., Sheth, A., Henson, C. (2008). *IEEE Internet Computing*, *12*(4), 46.

Samarati, P., & Sweeney, L. (1998). In *Proceedings of principles of database systems* (p. 188).

Simmhan, Y.L., Plale, B., Gannon, D. (2005). *SIGMOD Record*, *34*, 31. doi:10.1145/1084805.1084812.

Simmhan, Y.L., Plale, B., Gannon, D. (2006). In *IEEE international conference on web services* (pp. 18–22).

Simmhan, Y., Plale, B., Gannon, D. (2008). *International Journal Web Service Research*, *5*(2), 1.

Stevens, R.D., Robinson, A.J., Goble, C.A. (2003). *Bioinformatics Journal*, *19*(302).

Sultana, S., & Bertino, E. (2012). In *4th international provenance and annotation workshop*.

Syalim, A., Hori, Y., Sakurai, K. (2009). In Advances in information security and assurance, (pp. 51–59).

Taylor, N., & Ives, Z. (2006). In *SIGMOD'06* (pp. 13–24).

Tsudik, G., & Xu, S. (2003). In *ASIACRYPT* (pp. 269–286).

Vijayakumar, N., & Plale, B. (2006). Provenance and annotation of data. In L. Moreau & I. Foster (Eds.), *Lecture notes in computer science* (vol. 4145, pp. 46–54). Berlin: Springer. doi:10.1007/11890850_6.

Waters, B. (2005). In *EUROCRYPT'05* (pp. 114–127).

Weitzner D.J., Abelson, H., Berners-Lee, T., Feigenbaum, J., Hendler, J., Sussman, G.J. (2008). *Communication ACM*, *51*(82). doi:10.1145/1349026.1349043.

Xu, S., & Yung, M. (2009). First international conference on trusted systems (INTRUST'09). In *Lecture notes in computer science* (vol. 6163, pp. 104–128).

Xu, S., Qian, H., Wang, F., Zhan, Z., Bertino, E., Sandhu, R. (2010). In *Proceedings of 11th International Conference Web-Age Information Management (WAIM'10)* (pp. 398–404).

Zhang, J., Chapman, A., Lefevre, K. (2009). In *Proceedings of the 6th VLDB workshop on secure data management (SDM'09)* (pp. 17–32).