Privacy in a Networked World

Rebecca N. Wright
Computer Science Department and DIMACS Center
Rutgers University
Piscataway, NJ 08854 USA
rebecca.wright@rutgers.edu

August 9, 2007

Abstract

Networked electronic devices have permeated business, government, recreation, and almost all aspects of daily life. Coupled with the decreased cost of data storage and processing, this has led to a proliferation of data about people, organizations, and their activities. This cheap and easy access to information has enabled a wide variety of services, efficient business, and convenience enjoyed by many. However, it has also resulted in privacy concerns. As many recent incidents have shown, people can be fooled into providing sensitive data to identity thieves, stored data can be lost or stolen, and even anonymized data can frequently be reidentified. In this paper, we discuss privacy challenges, existing technological solutions, and promising directions for the future.

1 Introduction

Changes in technology are causing an erosion of privacy. Historically, people lived in smaller communities and there was little movement of people from one community to another. People had very little privacy, but social mechanisms helped prevent abuse of information. As transportation and communications technologies developed, people began to live in larger cities and to have increased movement between communities. Many of the social mechanisms of smaller communities were lost, but privacy was gained through anonymity and scale.

Now, advances in computing and communications technology are reducing privacy by making it possible for people and organizations to store and process personal information, but social

mechanisms to prevent the misuse of such information have not been replaced. While a major issue in computing and communications technology used to be how to make information public, we now have to work hard to keep it private.

The issue of "confidentiality"[1], or protecting information in transit or in storage from an authorized reader, is a well understood problem in computer science. That is, how does a sender Alice send a message M to an intended receiver Bob in such a way that Bob learns M but an eavesdropper Eve does not, perhaps even if Eve is an active attacker who has some control over the communication network? Although some difficulties remain in practical key management and end-host protection, for the most part, this problem is quite well solved by the use of encryption.

In contrast, a modern view of "privacy" is not simply about Eve not learning the message M or anything about its contents, but includes other issues such as Eve learning whether a message was sent between Alice and Bob at all, and questions about what Bob will and will not do with M after he learns it. This view of privacy in the electronic setting was first introduced in the early 1980's by David Chaum [1, 2, 3]. A growing interest in privacy in the computer science research community can be seen by the large number of annual conferences, workshops, and journals now devoted to the topic (e.g., [4, 8, 16, 14]).

Considered in isolation, it is easy to describe how to achieve privacy: one can live in isolation, avoid any use of computers and telephones, pay cash for all purchases, and travel on foot. Obviously, this is not realistic or even desirable for the vast majority of people. The difficulty of privacy arises because of the apparent conflict between utility and privacy—that is, desire of various parties to benefit from the convenience and other advantages provided by use of information, while allowing people to retain some control over "their" information, or at the very least to be aware of what is happening to their information.

The problem is further complicated by the fact that privacy means different things to different people. What some people consider a privacy violation is considered completely innocuous by other people. Or what seems a privacy violation in one context may not seem to be one in another context. For this reason, privacy is not a purely technological issue and cannot have purely technological solutions. Rather, social, philosophical, legal, and public policy issues are also important. However, technology *can* enable new policy decisions to be possible by instantiating solutions with particular properties.

Given the pervasive nature of networked computing, privacy issues arise in a large number

[1]One should note that the terminology distinction of "confidentiality" and "privacy" is not thoroughly standardized, and some people use the term "confidentiality" in the way the term "privacy" is used in this paper.

of settings. These include:

- **Electronic health records.** In the United States, there is a large effort to move towards electronic health records in order to improve medical outcomes as well as reduce the exorbitant cost of healthcare. Unless solutions can be developed that allow medical practitioners access to the right personal health information at the right time and in the right circumstances, while also ensuring that it cannot be accessed otherwise or used inappropriately even by those who have legitimate access, privacy will remain a barrier to adoption.
- **Government surveillance.** In the interest of protecting national security and preventing crime, governments often wish to engage in surveillance activities and link together huge amounts of data in order to identify potential threats before they occur and learn more about incidents and their perpetrators if they occur. Most people want increased security, but many remain concerned about invasion of privacy.
- **Commerce and finance.** Consumers have eagerly adopted on-line commerce and banking because of the great convenience of being able to carry out transactions from home or anywhere else. However, due to the use of personal information such as social security numbers and mother's maiden name being used for authentication coupled with the ease with which this kind of personal can be learned, this has resulted in a huge increase in identity theft. It also allows companies (particularly when data from multiple sources is aggregated by data aggregators) to gain great insight into customers and potential customers, often to the point leaves these customers feeling unsettled.
- **Pervasive wireless sensors.** Sensors such as RFID tags (inexpensive, tiny, chips that broadcast a unique 96-bit serial number when queried and are used in mobile payment applications such as EZPass and Exxon Mobil's Speedpass as well as increasingly embedded in consumer products (or even in consumers themselves in some cases)), mobile telephones and other personal devices that broadcast recognizable identification information, and GPS transmitters such as used in popular car navigation systems. These devices can potentially be used to track individuals' locations and interactions.

There are a large number of different kinds of solutions to various privacy problems, with different kinds of properties. Some aspects in which solutions differ is whether they are transparent (users cannot even tell they are there, but they protect privacy in some way anyway) or visible (users can easily tell they are there and can, or even must, interact with the system while carrying out their tasks); individual-user (in which an individual can unilaterally make a

choice to obtain more privacy, say by using a particular software package or configuring it in a certain way) or infrastucture (in which some shared infrastructure, such as the Internet itself, is modified or redesigned in order to provide more privacy); focused only notification (e.g., allowing or requiring entities to describe their data practices) or also on compliance; and in other ways. However, unless a solution primarily favors utility and functionality over privacy when choices must be made, it will tend not to be widely adopted. I describe one kind of solution, privacy-preserving data mining, in further detail in the following section.

2 Privacy-Preserving Data Mining

Sophisticated use of cryptography can yield solutions with unintuitive properties. The elegant and powerful paradigm of general secure multiparty computation [7, 19] shows how cryptography can be used to allow multiple parties each holding a private input to engage in a computation on their collective inputs in such a way that they all learn the result of the computation but *nothing else* about each other's data; further, this is achievable with computation and communication overhead that is reasonable when described as a function of the size of the private inputs and the complexity of the non-private computation.

Because the inputs to data mining algorithms are typically huge, the overheads of the general secure multiparty computation solutions are intolerable for most applications. Instead, research in this area seeks more efficient solutions for specific computations. Most cryptographic privacy-preserving data mining solutions to date address typical data mining algorithms, such as clustering [15, 9], decision trees [12], or Bayesian networks [13, 17]. Recent work addresses privacy preservation during the preprocessing step [10] and the postprocessing step [18], thereby working towards maintaining privacy throughout the data mining process.

Cryptographic techniques provide the tools to protect data privacy by exactly allowing the desired information to be shared while concealing everything else about the data. To illustrate how to use cryptographic techniques to design privacy-preserving solutions to enable mining across distributed parties, we describe a privacy-preserving solution for a particular data mining task: learning Bayesian networks on a dataset divided among two parties who want to carry out data mining algorithms on their joint data without sharing their data directly.

2.1 Bayesian networks

A Bayesian network (BN) is a graphical model that encodes probabilistic relationships among variables of interest [5]. This model can be used for data analysis and is widely used in data

mining applications.

Formally, a Bayesian network for a set V of m variables is a pair $(B_{\rm s}, B_{\rm p})$. The *network structure* $B_{\rm s} = (V, E)$ is a directed acyclic graph whose nodes are the set of variables. The *parameters* $B_{\rm p}$ describe local probability distributions associated with each variable. There are two important issues in using Bayesian networks: (a) Learning Bayesian networks and (b) Bayesian inferences. Learning Bayesian networks includes learning the structure and the corresponding parameters. Bayesian networks can be constructed by expert knowledge, or from a set of data, or by combining those two methods together. Here, we address the problem of privacy-preserving learning of Bayesian networks from a database vertically partitioned between two parties; in vertically partitioned data, one party holds some of the variables and the other party holds the remaining variables.

2.2 The BN Learning Protocol

A value x is *secret shared* (or simply *shared*) between two parties if the parties have values (*shares)* such that neither party knows (anything about) x, but given both parties' shares of x, it is easy to compute x. Our protocol for BN learning uses composition of privacy-preserving subprotocols in which all intermediate outputs from one subprotocol that are inputs to the next subprotocol are computed as secret shares. In this way, it can be shown that if each subprotocol is privacy-preserving, then the resulting composition is also privacy-preserving.

Our solution is a modified version of the well known *K2* protocol of Cooper and Herskovitz [5]. That protocol uses a score function to determine which edges to add to the network. To modify the protocol to be privacy-preserving, we seek to divide the problem into several smaller subproblems that we know how to solve in a privacy-preserving way. Specifically, noting that only the relative score values are important, we use a new score function g that approximates the relative order of the original score function. This is obtained by taking the logarithm of the original score function and dropping some lower order terms.

As a result, we are able to perform the necessary computations in a privacy-preserving way. We make use of several cryptographic subprotocols, including secure two-party computation (such as the solution of [19], which we apply only on a small number of values, not on something the size of the original database), a privacy-preserving scalar product share protocol (such as the solutions described by [6]), and a privacy-preserving protocol for computing $x \ln x$ (such as [12]). In turn, we show how to use these to compute shares of the parameters α_{ijk} and α_{ij} that are required by the protocol.

Our overall protocol of learning BNs is described as follows. In keeping with cryptographic

tradition, we call the two parties engaged in the protocol Alice and Bob.

Input: An ordered set of m nodes, an upper bound u on the number of parents for a node, both known to Alice and Bob, and a database D containing n records, vertically partitioned between Alice and Bob.

Output: Bayesian network structure B_{s} (whose nodes are the m input nodes, and whose edges are as defined by the values of π_i at the end of the protocol)

As the ordering of variables in V, Alice and Bob execute the following steps at each node v_i. Initially, each node has no parent. After Alice and Bob run the following steps at each node, each node has π_i as its current set of parents.

1. Alice and Bob execute privacy-preserving approximate score protocol to compute the secret shares of $g(i, \pi_i)$ and $g(i, \pi_i \cup \{z\})$ for any possible additional parent z of v_i.
2. Alice and Bob execute privacy-preserving score comparison protocol to compute which of those scores in Step 1 is maximum.
3. If $g(i, \pi_i)$ is maximum, Alice and Bob go to the next node v_{i+1} to run from Step 1 until Step 3. If one z generates the maximum score in Step 2, then z is added as the parent of v_i such that $\pi_i = \pi_i \cup \{z\}$ and Alice and Bob go back to Step 1 on the same node v_i.
4. Alice and Bob run a secure two-party computation to compute the desired parameter α_{ijk}/α_{ij}.

Further details about this protocol can be found in [17], where we also show how a privacy-preserving protocol to compute the parameters B_{p}. Experimental results addressing both the efficiency and the accuracy of the structure-learning protocol can be found in [11].

3 Challenges for the Future

Many challenges remain regarding privacy, both technical and political. These include social and political questions regarding who should have the right and/or responsibility to make various privacy-related decisions about data pertaining to an individual, as well as continued development and deployment of technologies to enable these rights and enforce that such privacy decisions, once made by the appropriate parties, are respected.

References

[1] David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–88, 1981.

[2] David Chaum. Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, 1985.

[3] David Chaum. Achieving electronic privacy. *Scientific American*, pages 96–101, August 1992.

[4] Conference on Computers, Freedom, and Privacy. *Proceedings*. Yearly since 1991.

[5] Greg F. Cooper and Edward Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Mach. Learn.*, 9(4):309–347, 1992.

[6] Bart Goethals, Sven Laur, Helger Lipmaa, and Taneli Mielikäinen. On private scalar product computation for privacy-preserving data mining. In *Proc. of the Seventh Annual International Conference in Information Security and Cryptology*, volume 3506 of *LNCS*. Springer-Verlag, 2004.

[7] O. Goldreich, S. Micali, and A. Wigderson. How to play ANY mental game. In *Proc. of the 19th Annual ACM Conference on Theory of Computing*, pages 218–229, 1987.

[8] International Workshop on Privacy-Enhancing Technologies. *Proceedings*. Yearly since 2002.

[9] G. Jagannathan and R. N. Wright. Privacy-preserving distributed k-means clustering over arbitrarily partitioned data. In *Proc. of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 593–599, 2005.

[10] G. Jagannathan and R. N. Wright. Privacy-preserving data imputation. In *Proc. of the ICDM Int. Workshop on Privacy Aspects of Data Mining*, pages 535–540, 2006.

[11] Onur Kardes, Raphael S. Ryger, Rebecca N. Wright, and Joan Feigenbaum. Implementing privacy-preserving Bayesian-net discovery for vertically partitioned data. In *Proceedings of the ICDM Workshop on Privacy and Security Aspects of Data Mining*, Houston, TX, 2005.

[12] Y. Lindell and B. Pinkas. Privacy preserving data mining. *J. Cryptology*, 15(3):177–206, 2002.

[13] D. Meng, K. Sivakumar, and H. Kargupta. Privacy-sensitive Bayesian network parameter learning. In *Proc. of the Fourth IEEE International Conference on Data Mining*, Brighton, UK, 2004.

[14] Symposium On Usable Privacy and Security. *Proceedings*. Yearly since 2006.

[15] J. Vaidya and C. Clifton. Privacy-preserving k-means clustering over vertically partitioned data. In *Proc. of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 206–215, 2003.

[16] Workshop on Privacy in the Electronic Society. *Proceedings*. ACM, Yearly since 2002.

[17] Z. Yang and R. Wright. Privacy-preserving computation of Bayesian networks on vertically partitioned data. *IEEE Transactions on Data Knowledge Engineering*, 18(9), 2006.

[18] Z. Yang, S. Zhong, and R. N. Wright. Towards privacy-preserving model selection. In *Proc. of the First ACM SIGKDD International Workshop on Privacy, Security, and Trust in KDD*, 2007.

[19] Andrew C.-C. Yao. How to generate and exchange secrets. In *Proc. of the 27th IEEE Symposium on Foundations of Computer Science*, pages 162–167, 1986.