Peer-to-Peer Privacy Preserving Reputation Inquiry: An Agent Assistant Approach

Bon K. Sy[1, 2]

[1]Queens College/CUNY
Computer Science Department
65-30 Kissena Blvd.
Flushing NY 11367
U.S.A.

[2]Graduate Center/CUNY
Computer Science Department
365 Fifth Ave
New York NY 10016
U.S.A.

Abstract
This research presents a privacy preserving peer-to-peer communication mechanism that allows peers using their personal agents to obtain reputation information of each other through a pair of trustworthy mediator proxies. A mediator proxy is considered trustworthy, if even when it is compromised, it can guarantee three conditions: (1) the anonymity of the identity of the responders and the target being inquired, (2) the privacy of the content in an inquiry and a response, and (3) the boundary limit of the reputation summary with no possibility of combining the response of multiple inquiries to reverse engineer the reputation rating of an individual responder.

Key words:
Privacy preserving, trustworthy mediator proxy, double-blind communication, homomorphic encryption.

1. Introduction

Let's consider the following reputation inquiry situation:

> *Given n participants in a reputation environment, each party needs to know the reputation of each other. When a party, let's say P1, is interested in the reputation of party P2, party P1 will select m "trusted colleagues" (from the group of n) serving as referees. These "trusted colleagues" are individualized referees for providing feedbacks about party P2. The reputation of party P2, based on the m (< n) referees selected by party P1, is derived based on the boundary limit of a linear sum of the weighted numerical scores of m referees.*

In the scenario just described, let's assume there are four parties *P1, P2, P3*, and *P4*; i.e., *n=4*. Party *P1* wants to know about (the reputation of) party *P2*. Party *P2* does not know party *P1* is asking about him/her, and party *P2* certainly does not know from whom *P1* will solicit the opinion. From party *P1* perspective, he has the choice of asking only party *P3 (m=1)*, or *P4 (m=1)*, or both *(m=2)*. In each of the three cases, parties *P3* and *P4*, are not supposed to know the solicitation is from *P1*, and certainly are not supposed to know *P1* is asking about *P2* ─ even parties *P3* and *P4* each will have an opinion about *P2*.

There are many applications to the reputation inquiry just described; e.g., reputation-based network security and reputation-based medical referral. Let's suppose a patient wants to know the reputation of a physician from, for example, other participants in an online blog. The patient could post the reputation inquiry to the online blog, and hope that those who have been treated by the physician could offer useful feedbacks on the physician. In a typical online blog, all participants, including the physicians, can see all the postings of each other. Consequently, the (alias) identity of the patient, the feedback providers, and the reputation feedback, will now all be exposed to the public, and subject to manipulation; e.g., the physician himself could create an alias identity and enter biased feedback to himself. Preferably, the identity of the patient, the inquiry content, and the response, can be concealed for privacy and security reason.

The goal of this paper is to show a privacy preserving peer-to-peer communication protocol for reputation inquiry; whereas the goal of a reputation inquiry is to obtain a boundary limit on a summary response composed of a linear combination of the weighted ratings from a group of selected "trusted peers".

Our proposed communication protocol is based on (1) a homomorphic encryption to guarantee the privacy of the content in an inquiry and a response, (2) a pair of trustworthy mediator proxies to guarantee the anonymity of the identity of the inquirer and the responders, as well as the privacy of the content of an inquiry and a response, (3) an algebraic transformation for masquerading the

Manuscript received July 5, 2007
Manuscript revised July 25, 2007

inquiry to prevent privacy leak in a covert channel, (4) an injection of a random noise to produce a useful boundary limit of a summary response, and yet to leave no possibility of combining the response of multiple inquiries to reverse engineer the reputation rating of an individual responder, and (5) an agent entity to extract need-to-know summary response.

In section 2 previous work as related to this research is presented. In section 3 the challenge and the requirement for privacy preserving reputation inquiry in a peer-to-peer communication environment is discussed. A naïve peer-to-peer communication based on homomorphic encryption is illustrated in section 4, followed by a discussion on its inadequacy due to the presence of a covert channel in section 5. We then present an improved version of the privacy preserving inquiry protocol in section 6. In section 7 an example illustration is shown, followed by a review discussion in section 8 on the strength and weakness of the proposed peer-to-peer privacy preserving communication. In section 9 the privacy preserving inquiry protocol is extended to incorporate agent entities of the peers. An application example and evaluation are shown in section 10, followed by the conclusion and future research in section 11.

2. Related Work

There are several basic research questions related to the reputation inquiry in a privacy preserving peer-to-peer communication:

Q1. How is the notion of reputation defined?
Q2. How is the identity of an inquirer protected?
Q3. How is the identity of a responder protected?
Q4. How is the content of an inquiry protected?
Q5. How is the response to an inquiry protected?

The notion of reputation has been studied across a number of diverse disciplines [1-7]. The user feedbacks posted after the completion of a transaction in eBay is perhaps one of the most well known reputation system. Reputation in eBay is simply a function of the cumulative positive and non-positive ratings for a seller or a buyer over the history of being an eBay member. As pointed out elsewhere [8], one of the most noticeable effects, so called Pollyanna, is the disproportional large number of positive feedbacks and rare negative feedbacks. Public disclosure of the rating and the rater information is one of the many factors attributes to the Pollyanna effect. There are also studies on the vulnerability of a reputation system [9] and the risk of misbehave because of the lack of reputation consequence [10]. For example, Sybil attack [11] is not uncommon in an environment when a participant in the reputation system could easily create multiple identities.

In this research we do not attempt to (re)define the notion of reputation. We simply adopted one of the simplest notions of reputation as suggested elsewhere [12], which is some contextualized ratings that a peer receives from others. In particular, we conceive reputation as a boundary limit of the sum of the weighted contextualized ratings of some selected "trusted peers." The specific focus of this research is on developing a communication protocol that could enforce a mechanism borrowed from economics called *Strategy proof* [13]. *Strategy proof* basically says that no participant could artificially alter the structure of the strategy to optimize the result. That is, artificially inflating/deflating ratings will not help to improve one's reputation. Consequently, the Pollyanna effect and the risk of Sybil attack should at least be partially alleviated.

Two main approaches are typically encountered in regard to protecting the identity of the participating parties as raised in questions (Q2) and (Q3); namely, store-and-forward proxy, and broadcasting [14]. An example store-and-forward proxy approach is Publius system [15] that relies on encryption and threshold key distributed to a static, system-wide list of servers to protect the identity of a publisher. Broadcasting as discussed elsewhere [16], on the other hand, protects the identity of a responder. Personal privacy protocol (P^5) as discussed elsewhere [14] provides mutual (inquirer and responder) anonymity through transmitting an inquiry and a response to a broadcast group as opposed to an individual party. While our proposed approach relies on the general concept behind both approaches to achieve inquirer-responder anonymity, it also distinguishes itself in two unique and novel aspects.

First, a data structure hashing the identity of peers is applied to encode a “position reference” for each trusted peer referee into a unique index of the inquiry vector. We then apply a homomorphic encryption to the inquiry that allows a referee to respond directly. In other words, the referee could respond to the encrypted inquiry without the inquirer sharing the decryption key, which is typically required to reveal the inquiry content. Second, a mediator proxy is involved in broadcasting an inquiry to all participating parties so that each of the m responders (referred to as "trusted peers" earlier) becomes indistinguishable from the remaining *n-m* participants in a peer-to-peer environment.

As different from the P^5 [14] or APFS [16], the identity of the inquirer is concealed through the use of a pair of mediator proxies. But our proxy approach is different from that of the Publius system because our mediator proxies do not get involved in an encryption process. Therefore, our mediator proxies are completely ignorant and have no access to the information about the identity of the responder(s) that is protected in the encrypted inquiry. In regard to the two aspects just mentioned, one distinct

advantage is its ability to provide an additional layer of privacy protection due to the encryption of the responder identity into the inquiry vector.

For protecting the privacy of the inquiry content and response, k-anonymity and cryptographic application are two general concepts commonly encountered. The basic idea behind k-anonymity is to introduce poly-instantiation so that an entity value is indistinguishable from *(k-1)* other objects assuming the same value. Exemplary privacy preserving techniques based on k-anonymity could be found elsewhere [17] [18]. Cryptographic application to privacy preserving has ranged from applying standard encryption techniques and PKI for protecting the "secrets" of the inquiry content and response [19], to creating a dining cryptographer network such as Herbivore [20]. Certain trust assumption is made in the cryptographic application to privacy preserving communication; particularly, the trustworthiness of the parties involved in the communication process. Our proposed approach could be considered as one kind of cryptographic application to privacy preserving communication, but with a provable privacy assurance similar to that of Herbivore [20].

3. Privacy Requirement & Challenge

The challenge of a reputation system is its trustworthiness and the risk of undesirable bias injection. Consider party *P1* solicits the opinion about *P2* from parties *P3* and *P4*, and party *P4* solicits the opinion about party *P3* from party *P2*. If these solicitations are held in public, parties *P2* and *P3* will each know that the other party is being solicited for feedbacks. Consequently, both parties may artificially inflate/deflate their opinion about each other in exchange for a favor/revenge, thus introducing undesirable bias. When this happens, the integrity of the reputation inquiry is compromised and its trustworthiness becomes questionable.

The success of a reputation inquiry depends on its ability to guarantee the privacy of each party on expressing its opinion about each other. A commonly encountered strategy is to introduce a mediator proxy to achieve a double-blind process. In doing so, administrative policy is required to make sure the mediator proxy to maintain the confidentiality of the information flowing through it. However, the compliance of the policy may not be enforceable and its success relies on the voluntarily participation. For example, eBay feedback system is one such case that relies on voluntarily participation of the buyers and sellers.

Even if voluntarily participation exists, we must also require no privacy leak from the mediator proxy. Note that in a traditional double-blind process, the mediator proxy has the information about the identity of the inquirer (party *P1* in the example shown in section 1), the identity of the target (party *P2*), and the identities of the referees (parties *P3* and *P4*). If there is a security breach on the mediator proxy, then the privacy of all parties in the above example is compromised.

To guarantee privacy, the mediator proxy must be completely trustworthy. A mediator proxy is completely trustworthy if it leaks no information even when it is compromised. The mediator proxy could be compromised due to, for example, passive sniffing by the peers on the communication channel between the mediator proxy and the inquirer/referee(s), or an (il)legal interception of the communication channel by an intruder/authority. Three conditions are required for a mediator proxy to be completely trustworthy.

First, there is no information leak on the identity of the target and the referees by the mediator proxy. In other words, the identity of the target (party *P2*) and the identity of the referees (parties *P3* and *P4*) remain anonymous to the mediator proxy. Second, there is no information leak on the response of the referees. Specifically, the response from each individual referee and the summary response are not intelligible to the mediator proxy. Only the summary (not individual) response is intelligible to the inquirer (party P1 in the example). Therefore, leaking the information about the specific response of each individual referee by the mediator proxy is not possible. Third, the mediator proxy is capable of introducing random noise to the summary (i.e., weighted sum of the scores) of the referees ─ even the summary of the referees is not known by the mediator proxy. The random noise is introduced in such a way that a useful boundary on the summary of the referees is derivable and guaranteed, while the precise score of each individual referee remains private. Our proposed research in privacy preserving reputation inquiry guarantees the privacy of each party just mentioned, thus providing provable trustworthiness.

4. Naïve Peer-to-Peer Communication

The basic idea behind our proposed research can be summarized in the following two key steps:

Step 1:

In a reputation inquiry environment of n parties, each party maintains an n^2x1 vector with $n(n-1) + 1$ zero entries, while the remaining $n-1$ entries contain the rating score of the peers; e.g., $V^{P3} = (0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 2.5\ 1.5\ 0\ 4.5\ 0\ 0\ 0\ 0)^T$ is the rating vector maintained by party *P3* in the example (shown in section 1) with the rating scores 2.5, 1.5, and 4.5 for parties *P1, P2*, and *P4* respectively. Similarly, $V^{P4} = (0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 3.5\ 1.75\ 0.45\ 0)^T$ is an example rating vector maintained by party P4 with the rating scores 3.5, 1.75, and 0.45 for parties *P1, P2*, and *P3* respectively. Specifically, the non-zero elements

in the rating vector of party *i (=1 .. 4)* is at the entries *n(i-1)+k* except the *n·(i-1)+i* entry; where *k = 1 ... 4, n = 4*.

Step 2:

A novel application of a simple exponential encryption $E(k, m) = k^m$ is employed to facilitate the communication between the inquirer and the mediator proxy, and between the mediator proxy with all parties in a *broadcasting mode*; where *k* is a dynamically changing encryption key and *m* is the message to be encrypted.

a. Phase 1 communication between inquirer and the mediator proxy:

Referring to the example where *P1* solicits inputs from parties *P3* and *P4* about *P2*. Let's assume the reputation of *P2* will be based on the weighted sum *2·SC+3·SD* as defined by *P1*; where *SC* and *SD* are the scores about *P2* by *P3* and *P4* respectively. The objective is for *P1* to inquire from the mediator proxy the summary response based on the sum of the weighted *SC* (i.e., 2·1.5) and *SD* (i.e., 3·1.75).

Party *P1* will first define a "secret key", let's say, $k = 2.92$. Then party *P1* will construct a *16x1* inquiry vector $(k^{m1}\ k^{m2}\ ...\ k^{m16})$; where $mi = 0$, or $k^{mi} = 1$ for $i = 1 ... 16$ except $i = 10$ or 14. In other words, the inquiry vector = $(1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2.92^2\ 1\ 1\ 1\ 2.92^3\ 1\ 1)^T$.

b. Communication between the mediator proxy and all parties:

Upon receiving an inquiry vector, the mediator proxy will broadcast the inquiry vector to all parties in the reputation inquiry environment. Each party will perform a simple "product of exponent" operation $\Pi_{i=1}^{16} (k^{mi})^{vi}$; where *vi* is the i^{th} element in the rating vector of the party. For example, the product of exponent over the inquiry vector and the rating vector of *P3* in the above example will result in a response (by party *P3*) $R^{P3} = \Pi_{i=1}^{16} (k^{mi})^{vi} = 2.92^{2\cdot1.5} = 2.92^3$ because the rating vector of party *P3* is $V^{P3} = (0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 2.5\ 1.5\ 0\ 4.5\ 0\ 0\ 0\ 0)^T$ and the inquiry vector is $(1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2.92^2\ 1\ 1\ 1\ 2.92^3\ 1\ 1)^T$.

c. Phase 2 communication between the mediator proxy and the inquirer:

Upon receiving the response from all parties, the mediator proxy will "calculate" the responses from all parties by performing a multiplication operation $S = \Pi_{i=2}^{n} R^{Pi}$. In the above example, $S = 2.92^{2\cdot1.5} \cdot 2.92^{3\cdot1.75} = 2.92^{2\cdot1.5+3\cdot1.75}$. Afterwards, the mediator proxy will compose the final response by adding a random noise S^{rnd} to the response so that the final response becomes $S^{(1+rnd)}$; where *rnd* is a value from some random number generator *RND(seed, param-val)* with a *seed* value, and a distribution characterized by the set of statistical parameters defined in *param-val* over the range of random values $[rnd_{min} ... rnd_{max}]$. An example distribution is a one-sided Gaussian distribution with unit value as its mean and variance defined in *param-val*.

The final response $S^{(1+rnd)}$, together with the details of the random number generator, will be transmitted by the mediator proxy to the inquirer (party *P1*). Upon receiving the final response, the inquirer decrypts the final response using the $Log_k(\bullet)$ operator. In the example above, it is $Log_{2.92}(2.92^{(2\cdot1.5+3\cdot1.75)(1+rnd)}) = (8.25)(1+rnd)$. Based on the given random number generator, the inquirer will then be able to derive a (lower/upper) boundary on the reputation of the target (i.e., party *P2* in the example).

5. Covert Channel

The peer-to-peer communication just described is considered naïve because (1) the mediator proxy is still a risk for privacy leak, and (2) the inquiry vector in step 2a presents a covert channel for re-identifying the referees and even inference on the precise inquiry.

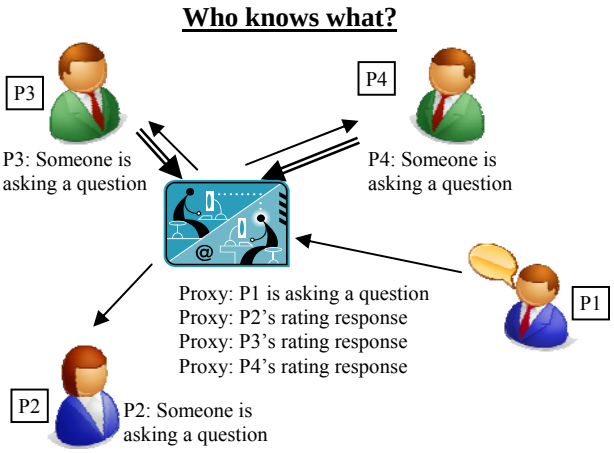

Figure 1: Delineation on "who knows what"

Referring to figure 1, when P1 sends the inquiry to the mediator proxy in step 2a, and when all parties responds to the mediator proxy in step 2b, the mediator proxy will know the encrypted inquiry and the response to the encrypted inquiry. In other words, the mediator proxy could decode the response by simply taking the log of the response with the encrypted inquiry value as the base of the log; i.e., for an encrypted inquiry item *em* $(=k^{mi})$ that the mediator proxy receives, the mediator proxy could decode the i^{th} element in the rating vector of a responder *vi* by taking $Log_{em}(k^{mi})^{vi} = vi$, thus entailing the risk of a privacy leak.

In addition, referring to the example inquiry vector $(1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 2.92^2\ 1\ 1\ 1\ 2.92^3\ 1\ 1)^T$ in the previous section, any entry with a "1" is from an encryption result $E(k, mi = 0)$ irrespective to the choice of the encryption key. When $mi = 0$, this is equivalent to skipping the opinion of a referee with an index at the corresponding *"1"* entry of the inquiry vector because any opinion weighted by *0* is nullified. Therefore, one could deduce

from the inquiry vector that the reputation about *P2* is being inquired from parties *P3* and *P4*.

Furthermore, since exponential encryption is a common knowledge to all parties, the ratio between the encrypted values (other than *"1"*) of two integer coefficient weighting factors may reveal further information about the encryption key. For example, $2.92^3/2.92^2$ will return a value that is a geometric multiple (and in this case the exact value) of the encryption key! Fortunately there is a solution for the challenge due to the covert channel.

6. Improved Peer-to-Peer Communication

There are three key concepts for preventing the covert channel; namely, employing two (instead of one) mediator proxies to achieve separation of duty, algebraic transformation of an intended inquiry, and an enhanced homomorphic encryption satisfying $E(k, m1+m2) = E(k,m1)E(k,m2)$, *and* $[E(k,m)]^c = E(k,mc)$; where k is an encryption key, and the addition, product, and exponent are standard arithmetic operators.

A pair of mediator proxy to achieve separation of duty

Referring to figure 1, the risk of privacy leak by the mediator proxy is due to the reliance on one single entity in the communication protocol to handle the inquiry and the response. A remedy to mitigate the risk of privacy leak is to introduce an additional mediator proxy, and to separate the duty of handling inquiry and response by two independent mediator proxies as shown below:

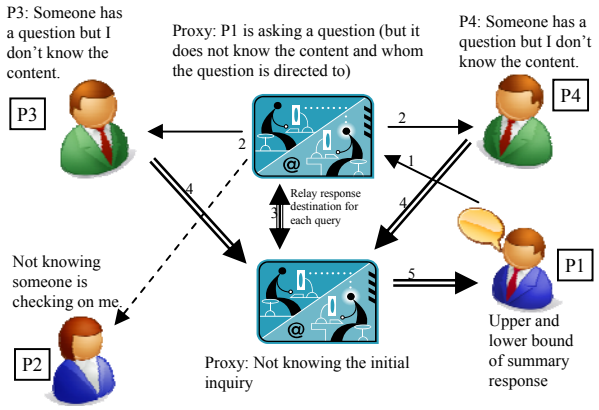

Figure 2: Improved Privacy Preserving Protocol

Referring to figure 2, P1 sends the encrypted inquiry to one mediator proxy ─ referred to as inquiry handler. The inquiry handler notifies the other mediator proxy ─ referred to as response handler. In addition, the inquiry handler also informs the peer responders about the identity of the response handler, as well as the location to which the response handler should send the summary response.

Algebraic transformation and homomorphic encryption

Let's refer to the example inquiry $2 \cdot SC+3 \cdot SD$ as defined by *P1*; where *SC* and *SD* are the rating about *P2* by *P3* and *P4* respectively. Instead of encrypting and posting directly the inquiry, two arbitrary queries would be composed in such a way that some linear combination of the two queries will result in the cancellation of all terms except the weighting factors for *SC* and *SD*. For example, consider the following two inquiry vectors that relate to the original inquiry in form of $IV_2 - 2 \cdot IV_1$:

$$IV_1 = (2.3\ 4.2\ 1.3\ 2.4\ 9.6\ 2.7\ 8.3\ 7.6\ 2.9\ \mathbf{6.6}\ 3.2\ 4.3\ 5.1\ \mathbf{9.9}\ 1.7\ 2.4)^T$$

$$IV_2 = (4.6\ 8.4\ 2.6\ 4.8\ 19.2\ 5.4\ 16.8\ 15.2\ 5.8\ \mathbf{15.2}\ 6.4\ 8.6\ 10.2\ \mathbf{22.8}\ 3.4\ 4.8)^T$$

Instead of using the exponential encryption k^m, an enhanced version of the homomorphic encryption will take the form of $E(k,a,m)=k^{am}$; where *(k, a)* are encryption secrets and *m* is the message to be encrypted. Suppose two different encryption keys *(k1=2.92, a1=3.1)* and *(k2=3.29, a2=2.9)* are used for IV_1 and IV_2 respectively, the encrypted inquiry vectors would appear to have no regular patterns among the numbers in the vectors.

Let $E(2.92,3.1,R^{IV1})$ and $E(3.29,2.9,R^{IV2})$ be the encrypted response vectors for IV1 and IV2 respectively, and *rnd1* and *rnd2* are the corresponding random noise. The following linear combination allows the reconstruction of the boundary limit for the summary response to the original inquiry $(2 \cdot SC + 3 \cdot SD)$:

$$\sum_{i=10,14} Log_{3.29}(E(3.29,2.9,R^{IV2})_i)/2.9 - 2 \cdot [Log_{2.92}(E(2.92,3.1,R^{IV1})_i)/3.1]$$
$$= (2 \cdot SC + 3 \cdot SD) + (2 \cdot SC \cdot rnd1 + 3 \cdot SC \cdot rnd2)$$

In general, an inquiry on the reputation of a party from a group of referees can be decomposed into queries of which some linear combination will result in the original inquiry. Below we summarize the algorithmic steps for the improved privacy preserving peer-to-peer communication in an environment of *n* parties:

Step 1: (initialization)

Each party *Pi* maintains an *nx1* (n ≥ i) vector R^{Pi} storing the rating score of each peer with the i^{th} entry being *0* (because an individual entity does not rate itself). Each party also has access to the following functions:

$make\_RV(R^{Pi},i) = T_i \times R^{Pi}$

where T_i is a transformation matrix of size n^2xn for mapping R^{Pi} of size *nx1* to a rating vector V^{Pi} of size n^2x1 with zero padding, and $T_i^T = [Z_{nxn(i-1)}\ \ I_{nxn}\ \ Z_{nxn(n-i)}]$ is composed of a zero matrix of size $nxn(i-1)$, an identity matrix of size *nxn*, and another zero matrix of size $nxn(n-i)$

$make\_Response(V^{Pi}, IV^{Pk}) = \Pi_{j=1}^{nxn} (k^{mj})^{vj}$

vj is j^{th} entry in the V^{Pi}, and k^{mj} is the j^{th} entry in IV^{Pk}

Step 2:

a. Phase 1 communication between inquirer and the mediator proxy:

2-a.1: Inquirer *Pi* defines the target party *Pj*, the set of "trusted peer referees" *Ri_j = {Pk|Pk* is a referee selected by *Pi* to offer an opinion on *Pj}*, and a weighting factor *wj_k* associated with each *Pk* in *Ri_j*. In other words, the reputation inquiry (on *Pj*), denoted by RI_{Pj}, can be mathematically represented by a n^2x1 vector with the *n(k-1)+j* entry being *wj_k*, and all other entries being zero; where $Pk \in Ri\_j$, and *n* is the number of peers in the reputation system environment.

2-a.2: Denote the number of referees in *Ri_j* by $|Ri\_j|$, the inquirer composes $|Q|$ numbers of arbitrary query vectors $Q_1 \ldots Q_{|Q|}$ satisfying one condition: there exists a linear combination $\sum_{i=1}^{|Q|} a_i Q_i = RI_{Pj}$ for some a_is. The coefficients a_is can be determined algebraically that will retro-fit the condition $\sum_{i=1}^{|Q|} a_i Q_i = RI_{Pj}$.

2-a.3: For each Q_v where $v = 1 \ldots |Q|$, the inquirer defines an encryption key (k_v,a_v). Each element in the query vector Q_v is then encrypted using (k_v,a_v) to compose the inquiry vector IV_v; i.e., $E(k_v,a_v,Q_v) \rightarrow IV_v$ for $v = 1 \ldots |Q|$.

b. Communication between the mediator proxy and all parties:

Upon receiving an inquiry vector, the inquiry handler broadcasts the inquiry vector to all parties, and notifies the response handler the destination of summary response. Each party *Pi* ($i = 1 \ldots n$ except the inquirer) will call the function $make\_RV(R^{Pi},i)$ (defined in step 1) to compose its rating vector V^{Pi} and subsequently $make\_Response(V^{Pi}, IV^{Pk})$ to compose a response vector *RV_Pi*.

c. Phase 2 communication between the mediator proxy and the inquirer:

Upon receiving the response from all parties, the response handler will combine the responses from all parties to generate a summary response vector $S\_IV_v = \Pi_{Pi} RV\_Pi$. Afterwards, the response handler will compose the final response by adding a random noise $S\_IV_v^{rnd}$ to the response so that the final response becomes $S\_IV_v^{(1+rnd)}$; where *rnd* is an "unpredictable" value for each query from a random number generator *RND(seed, param-val)* with a *seed* value, and a distribution characterized by the set of statistical parameters defined in *param-val* over the range of random values $[rnd_{min} \ldots rnd_{max}]$.

The final response $S\_IV_v^{(1+rnd)}$, together with the information about the random number generator, is transmitted by the response handler to the inquirer *Pi*.

Step 3:

Upon receiving the response for each IV_v, the inquirer *Pi* decrypts the final response using the $(1/a_v)Log_{kv}(\bullet)$ operator to obtain the unencrypted response R_Q_v for the query Q_v; where $v = 1 \ldots |Q|$.

Upon deriving the responses $R\_Q_1 \ldots R\_Q_{|Q|}$, construct the linear combination $\sum_{i=1}^{|Q|} a_i (R\_Q_i)_j$ as described in step 2-a.2 using the relevant j^{th} entries in R_Q_i to obtain a boundary limit for the reputation inquiry RI_{Pj} about *Pj*.

7. Illustration

In this section we show an example illustration for the improved peer-to-peer communication presented in section 6. Consider a five-party (*P1 ... P5*) environment in which each party maintains the following information:

@ Step1:

R^{P1}=(0 2.6 3.7 1.7 6.9)^T R^{P2}=(1.3 0 2.6 5. 1 4.7)^T
R^{P3}=(3.2 2. 4 0 4.3 5.2)^T R^{P4}=(2.7 3.2 4.5 0 2.3)^T
R^{P5}=(4.5 4.2 3.1 2.3 0)^T

Furthermore, the matrix representation of T_2^T in $make\_RV(R^{P2},2)$ is shown below for an illustration purpose:

Column	**1 … 5**	**6**	**7**	**8**	**9**	**10**	**11 … 25**
Row 1	0 … 0	1	0	0	0	0	0 … 0
2	0 … 0	0	1	0	0	0	0 … 0
3	0 … 0	0	0	1	0	0	0 … 0
4	0 … 0	0	0	0	1	0	0 … 0
5	0 … 0	0	0	0	0	1	0 … 0

$make\_RV(R^{P2},2) = T_2 \times R^{P2}$
= [0 0 0 0 0 **1.3 0 2.6 5.1 4.7** 0 … 0]^T

@ Step 2-a.1

Let's assume *P3* solicit the reputation of *P1* from its trusted peer group defined by *P2* and *P5*, then *Ri_j = R3_1 = {P2 P5}* for $i = 3$ and $j = 1$. Let's further assume *P3* assigns a weighting factor of *0.8* for the response of *P2* and *0.2* for the response of *P5*; i.e., $w1\_2 = 0.8$ and w1_5 = 0.2. The reputation inquiry RI_{P1} can then be expressed mathematically as a 5^2x1 vector shown below:

RI_{P1}=[0 0 0 0 0 0.8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0.2 0 0 0 0]^T

Note that the only non-zero values are *wj_k* at the *n(k-1)+j* entry with $n = 5, j=1,$ and $k=2, 5$.

@Step 2-a.2:

In this example, $|R3\_1| = 2$. RI_{P1} could be expressed in terms of the linear combination of two queries defined by *P3* as shown below:

Q_1=[0.997 0.611 0.266 0.84 0.376
0.28 0.009 0.276 0.588 0.838
0.485 0.744 0.458 0.744 0.599
0.735 0.572 0.152 0.425 0.517
0.31 0.169 0.492 0.7 0.148]^T

Q_2=[0.204 0.125 0.055 0.172 0.077
-1.092 0.002 0.057 0.12 0.172
0.099 0.152 0.094 0.153 0.123
0.151 0.117 0.031 0.087 0.106
-0.225 0.035 0.101 0.143 0.03]^T

Note that $RI_{P1} = 0.142 \cdot Q_1 - 0.693 \cdot Q_2$

@Step 2-a.3:

The followings are the encrypted *Q1* and *Q2* using the encryption secrets defined earlier (i.e., *k1=2.92, a1=3.1* for Q_1, and *k2=3.29, a2=2.9* for Q_2):

IV_1=[27.42 7.624 2.421 16.294 3.485
2.535 1.03 2.5 7.05 16.159
5.007 11.829 4.578 11.857 7.315
11.492 6.696 1.654 4.106 5.572
2.8 1.753 5.124 10.222 1.632]^T
IV_2=[2.025 1.541 1.207 1.812 1.305
0.023 1.006 1.216 1.516 1.809
1.409 1.693 1.383 1.693 1.528
1.682 1.499 1.113 1.351 1.442
0.46 1.127 1.416 1.641 1.11]^T

Note that the k^{th} entry in IV_1 is $2.92^{3.1 \cdot Q1k}$; where $Q1_k$ is the k^{th} entry in $Q1$. Similarly, the k^{th} entry in IV_2 is $3.29^{2.9 \cdot Q2k}$; where $Q2_k$ is the k^{th} entry in $Q2$.

@Step 2-b:

After the inquiry handler receives IV_1 and IV_2 from $P3$, the inquiry handler notifies the response handler that the summary response is to be sent to $P3$. In addition, the inquiry handler also notifies *P1, P2, P4 and P5* to send the response to the response handler. IV_1 and IV_2 are then broadcasted to *P1, P2, P4*, and *P5* by the inquiry handler. The functions $make\_RV(R^{Pi},i)$ and $make\_Response(V^{Pi}, IV^{Pk})$ are then called by each party to compose a response vector RV_Pi for each of the IV_j, i =1,2,4,5, and j=1,2. To maintain the focus on only the most relevant information, we only show one such response from *P2* for IV_1:

1 1 1 1 1
3.351 1 10.836 2.12E+04 4.78E+05
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1

@Step 2-c:

After the response handler receives the responses from parties *P1, P2, P4, P5*, a summary response vector $S\_IV_v = \Pi_{Pi} RV\_Pi$ is generated for each query (i.e., v=1,2). Different levels of the random noise are then added to the response for each query. Below shows an example final response vectors for the queries:

Response for $IV_1 = S\_IV_1^{(1+rnd)}$ (where *rnd = 0.15)*
= [1 434.266 43.071 234.162 20076.443
4.017 1 15.49 94314.61 3.399E+06
1 1 1 1 1
1961 1093.74 13.537 1 94.033
206.266 15.048 338.654 467.975 1]^T

Response for $IV_2 = S\_IV_2^{(1+rnd)}$ (where *rnd = 0.13)*
= [1.000 3.566 2.198 3.133 7.956
0.004 1.000 1.774 10.997 23.290
1.000 1.000 1.000 1.000 1.000
4.889 4.326 1.725 1.000 2.588
0.019 1.764 3.382 3.623 1.000]^T

@Step 3:

Upon receiving the response for IV_v *(v=1,2),* the inquirer *P3* decrypts the response using the $(1/a_v)Log_{kv}(\bullet)$ operator on each element in the summary response vector $S\_IV_v^{(1+rnd)}$ *(v=1,2)* and the following results are obtained:

R_Q_1 = [0 1.828 1.133 1.642 2.982
0.419 0 0.825 3.448 4.527
0 0 0 0 0
2.282 2.106 0.784 0.000 1.368
1.604 0.816 1.754 1.851 0]^T
R_Q_2 = [0 0.368 0.228 0.331 0.601
-1.612 0 0.166 0.694 0.912
0 0 0 0 0
0.460 0.424 0.158 0 0.275
-1.147 0.164 0.353 0.373 0]^T

Let r_q1_i and r_q2_i be the ith entry in R_Q_1 and R_Q_2 respectively; where i = *1…25*. Note that only two terms of r_q1_i and r_q2_i are relevant to the original inquiry; which are i=*6* and i=*21*. The sum of these terms, $\sum_{i=6,21}(0.142 \cdot r\_q1_i - 0.693 \cdot r\_q2_i)=2.199$, produces an upper boundary limit for the query RI_{P1} that inquires into the reputation of *P1* based on the linear combination of the rating by *P2* (i.e., 1.3) with a weighting factor 0.8, and the rating by *P5* (i.e., 4.5) with a weighting factor 0.2. It could be noted that $\sum_{i=6,21}[(0.142 \cdot r\_q1_i/1.15) - (0.693 \cdot r\_q2_i/1.13)]$ is the exact linear combination of the rating information being sought (i.e., 0.8·1.3+0.2·4.5=1.94); whereas the scaling 1.15 and 1.13 are from *(1+rnd)* introduced by the response handler to query 1 and 2 respectively.

8. Discussion

Referring to step 2 in section 6, the secret key is never shared by the inquirer. Without the secret key, one could not tell from the inquiry vector the specific inquiry who is the target and who is/are the referee(s). Furthering in step 2b, the inquiry handler broadcasts the inquiry to all parties, thereby eliminating the possibility of a security compromise due to a covert channel revealing the identity of the intended receiver(s). Consequently, the privacy of the responders and the target being inquired is protected ─ the first condition of a trustworthy mediator proxy described earlier. In addition, the responses from the referees are unintelligible to other peers and the response handler (because the inquiry handler does not share the original encrypted inquiry vector(s) IV_i). Only the inquirer can decrypt the response as the secret key holder ─ the second condition of a trustworthy mediator proxy. Finally, the injection of the random noise by the response handler eliminates the possibility of the inquirer to make multiple identical inquiries to algebraically re-derive the specific response of each individual referee, thereby the privacy of the response of a referee is also protected ─ the third condition of a trustworthy mediator proxy.

Although our peer-to-peer communication protocol delivers the response to a reputation inquiry with an assurance on the stipulated privacy preserving conditions, there is a vulnerability that has not yet been addressed in our approach. Referring to the example in the previous section, there are additional information in the response vectors R_Q_1 and R_Q_2; i.e., the boundary limit of the ratings of all individuals by all parties. For example, the inquirer *P3* knows the second entry in Q_1 is 0.611, or $(2.92^{3.1 \cdot 0.611})^{v12(1+rnd)} = 434.266$ (the second entry observed in the response for IV_1; i.e., $S\_IV_1^{(1+rnd)}$) because $k1=2.92$, $a1=3.1$; where *v12* is the rating of *P2* by *P1*. In other words, $v12(1+rnd) = (\log_{2.92}434.266)/(3.1 \cdot 0.611)$, or $v12 \leq \log_{2.92}434.266/(3.1 \cdot 0.611) = 2.9924$ which is 15% more than the actual rating of *P2* by *P1* as injected by the mediator proxy.

The risk of the vulnerability just mentioned is a potential exploit if the statistical behavior of the random number generator allows reverse identification, thus the possible value(s) of *rnd*. This may allow the inquirer to infer the ratings of all individuals by all parties with a certain level of confidence. Yet the quality of an inquiry response suffers if the information about the statistical behavior of the random number generator is concealed excessively. An alternative is an independent entity that will act as an agent on behalf the inquirer to receive the encrypted summary response from the mediator proxy, and to strip the additional information in the response vectors prior to passing to the inquirer only the necessary information related to a reputation inquiry.

9. Agent Assistant for Privacy Preserving

In the agent assistant setup for the privacy preserving reputation inquiry, figure 3 shows the steps of interaction among the inquirer, the pair of mediator proxies, and the responders using the previous example; i.e., P1 solicits the reputation of P2 from P3 and P4.

In summary, the protocol for agent assistant privacy preserving reputation inquiry shown in figure 3 is comprised of seven steps:

1. An inquirer specifies an inquiry for its agent.
2. The agent derives new inquiry vector(s) algebraically that can reconstruct the original inquiry vector through some linear combination. In addition, the agent chooses an encryption key and applies homomorphic encryption to encrypt each elements in each of the new inquiry vector(s). Each element is then sent one-by-one to the mediator proxy handling the inquiry.
3. Upon receiving an inquiry, the inquiry handler notifies the response handler to anticipate incoming responses and the destination of the inquirer for the summary response to be sent to.
4. The inquiry handler broadcasts the encrypted inquiry vector(s) to all peers.
5. Each peer responder generates the response by raising each element of each encrypted inquiry vector to the power of the rating score using the *make_Response* function as described in step 1 in section 6, and sends the response to the response handler (mediator proxy).
6. Upon receiving the replies, the response handler combines the responses to generate a summary response, and then a random noise is introduced to the replies as described in step 2c in section 6. The summary response is then sent to the agent of the inquirer using the destination information provided by the inquiry handler in step 3.
7. Upon receiving the summary response to the encrypted inquiry vector(s), the agent decrypts the response and composes the linear combination of the response vectors to reconstruct the response vector to the original inquiry, and extracts only the need-to-know boundary limit information to send to the inquirer.

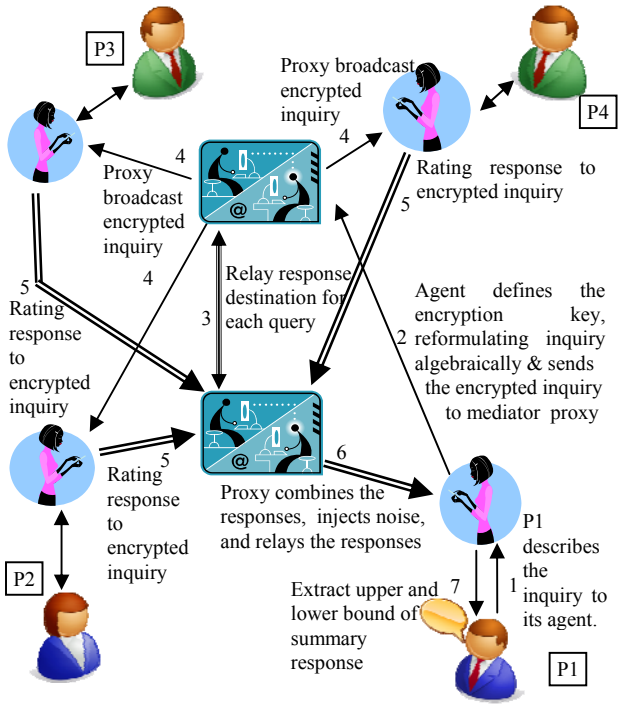

Figure 3: Agent assistant Reputation inquiry

10. Security Application and Evaluation

A preliminary experimental study was conducted to investigate its potential for a real world application, and particularly for reputation-based network security.

In this preliminary experimental study, signature based intrusion detection alerts over a period of three days from sensors of six subnets are used. We model each intrusion detection sensor as an agent for cross sharing information to develop a holistic view of the security status. Due to security implication, the stake holders of the intrusion detection logs prefer to conceal their identity even they are all interested in cross sharing information in their security logs. As such, the proposed privacy preserving reputation system approach is applied to achieve intrusion detection information sharing.

Over the period of the three-day preliminary study, 4618 alerts were generated by the six intrusion detection sensors. 3.68% of the alerts have a source IP indicating an origination from internal (i.e., one of the six subnets) while the rest 96.32% shows an origination from the external. For the purpose of this preliminary study, only the partial record of an alert is used; i.e., the source and destination IP addresses and ports. Based on the source IP shown in an alert, we located the Internet Service Provider (ISP) who handles the routing of the traffic originated from the source IP. We then performed a reverse look up on the country origin of the ISP.

A total of 16 countries were found in the alerts. The "creditability score" (reputation) of each country is derived based on the potential threat of the traffic activities, as grouped by the country origin, logged in the alerts. In this experiment, two-point penalty is assigned for each alert with a destination port less than 1024, which indicates a potential anomaly of server-side contact as detected by the intrusion detection sensor. One-point penalty is assigned if the destination port in an alert is above 1024. For each IDS sensor, a *threat* rating score for each one of the 16 countries is derived based on the normalized sum of the penalty points from alerts with a source IP handled by an ISP of the corresponding country; whereas the normalization factor is twice the total number of alerts as generated by the IDS sensor. The creditability score of a country is the negative of the threat score. As such, the creditability score of a country as assigned by an IDS sensor is always between 0 and -1. As seen by an IDS sensor, creditability score zero means that the corresponding country has not generated network traffic activities that have alarmed the sensor. On the other hand, a creditability score (close to) -1 means that the network traffic activities (originated from a country) are considered to have posed the most serious threat. A 16x1 vector is created by each IDS sensor to maintain the reputation score of the 16 countries; whereas the vector indices are the result of a no-collision hashing that takes a country name as an input.

In this experiment, the participants of the peer-to-peer communication are the six IDS sensors, and each participant posts two queries to the mediator proxies. The objective of each IDS sensor is to determine whether the cumulative threat as measured by the sum of the threat score of the worst three "offenders" (i.e., countries with the three lowest creditability scores), is typical to other IDS sensors. Although one query is sufficient, two queries are posted so that an upper and a lower boundary can be obtained. In this experiment, a random number generator with a uniform distribution is used by the response handler. The random number generator is used to generate a random value between zero and one in dealing with the query, thus resulting in a lower bound for the response to the query. Likewise, a random value (>1) is generated for the query. This results in an upper bound for the response to the query.

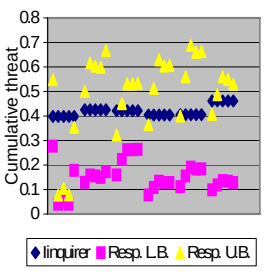

Table 1: Boundary interval of response

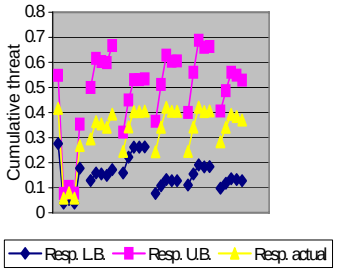

Table 2: Cumulative threat of responders

Table 1 shows the boundary interval of the query response in relation to the threat score of the inquirer. Table 2 shows the upper (U.B.) and lower (L.B.) boundary of the threat score of each country derived by each IDS sensor, as well as the actual value. Table 1 shows the relevancy and accuracy of the response, while table 2 shows the degree of privacy preserving of the creditability score information of each IDS sensor.

11. Conclusion

This paper presents a privacy preserving peer-to-peer communication protocol for reputation inquiry. Our proposed approach involves a pair of trustworthy mediator proxies and guarantees three conditions: (1) the anonymity of the identity of the responders and the target being inquired, (2) the privacy of the content in an inquiry and a response, and (3) the boundary limit of the reputation summary with no possibility of combining the response of multiple inquiries to reverse engineer the reputation rating of an individual responder.

We show an algebraic transformation and the use of an enhanced homomorphic encryption to protect inquiry privacy from the threat of a covert channel existed in any communication process permitting multiple queries. An example illustration is presented to show the mechanism of each step and is used to discuss its strength and its limitation. Finally, we describe the extension of the improved privacy preserving communication protocol to incorporate agent assistants to ascertain the delivery of only need-to-know summary response. Our future study will focus on analyzing the potential privacy leak arises from the necessity of sharing the statistical information for interpreting the boundary limit of an inquiry response.

Acknowledgements

This paper is an extended version of the paper nominated for the best paper award in the 2007 IEEE Workshop on Information Assurance. This work was supported in part by the Research Award Program of PSC-CUNY Research Foundation.

References

[1] J. Andreoni, J.H. Miller "Rational Cooperation in the Finitely Repeated Prisoner's Dilemma: Experimental Evidence," The Economic Journal, 103(418), 1992.
[2] R. Selten, "The Chain Store Paradox," Theory and Decision, 9, 1978, pp. 127-159.
[3] J. Sabater, C. Sierra, "REGRET: A Reputation Model for Gregarious Societies," 4th Workshop on Deception, Fraud, and Trust in Agent Societies, 2001.
[4] L. Mui, M. Mohtashemi, C. Ang, P. Szolovits, A. Halberstadt, "Ratings in Distributed Systems: A Bayesian Approach," 11th Workshop on Information Technologies and Systems (WITS), New Orlean, 2001.
[5] B. Yu, M.P. Singh, "Towards a Probabilistic Model of Distributed Reputation Management," 4th Workshop on Deception, Fraud, and Trust in Agent Societies, Montreal, Canada, 2001.
[6] G.B. Pollock, L.A. Dugatkin, "Reciprocity and the Evolution of Reputation," Journal of Theoretical Biology, 159, pp. 25-37, 1992.
[7] S. Wasserman, K. Faust, Social Network Analysis: Methods and Applications, Cambridge U. Press, 1994.
[8] P. Resnick, R. Zeckhauser, "Trust Among Strangers in Internet Transactions: Empirical Analysis of eBay's Reputation System," NBER Workshop on Empirical Studies of Electronic Commerce Paper, 2000.
[9] C. Dellarocas, "Immunizing Online Reputation Reporting Systems Against Unfair Ratings and Discriminatory Behavior," Proc. 2nd ACM Conference on Electronic Commerce, 2000.
[10] L.C. Freeman, "Centrality in Social Networks: I. Conceptual Clarification," Social Networks, 1, 1979.
[11] http://en.wikipedia.org/wiki/Sybil_attack
[12] G. Zacharia, P. Maes, "Collaborative Reputation Mechanisms in Electronic Marketplace," Proc. 32nd Hawaii International Conference on System Sciences.
[13] http://en.wikipedia.org/wiki/Strategyproof
[14] R. Sherwood, B. Bhattacharjee, A. Srinivasan, "P^5: A protocol for Scalable Anonymous Communication," in Proc. IEEE Symp. Security and Privacy, Oakland, CA, May 2002, pp. 58-70.
[15] M. Waldman, A.D. Rubin, L.F. Cranor, "Publius: A robust, tamper-evident, censorship-resistant, web publishing system," in Proc. USENIX Security Symp., 2000, pp. 59-72.
[16] V. Scarlata, B. Levine, C. Shields, "Responder anonymity and anonymous peer-to-peer file sharing," in Proc. IEEE ICNP, Riverside, CA, 2001, pp. 272-280.
[17] L. Sweeney, "Achieving k-anonymity privacy protection using generalization and suppression," International Journal on Uncertain, Fuzziness, Knowledge-Based System, vol. 10, no. 5, Oct. 2002.
[18] L. Sweeney, "K anonymity: A model for protecting privacy," International Journal on Uncertain, Fuzziness, Knowledge-Based System, vol. 10, no. 5, Oct. 2002.
[19] A. Heinemann, J. Kangasharju, F. Lyardet, M. Muhlhauser, "Ad Hoc Collaboration and Information Services Using Information Clouds," Proc. of the 3rd Workshop on Applications and Services in Wireless Networks (ASWN 2003), T. Braun, N. Golmie, J. Schiller, Eds., Bern, Switzerland: Institute of Computer Science and Applied Mathematics, University of Bern, 2003.
[20] S. Goel, M. Robson, M. Polte, E. Sirrer, "Herbivore: A scalable and efficient protocol for anonymous communications," Dept. Computer Information Science, Cornell University, Ithaca, NY, CIS Tech. Rep. TR2003-1890, Feb 2003.

Bon Sy received his Ph.D. (1988) in Electrical and Computer Engineering from Northeastern University in Boston, Massachusetts. He is currently a full Professor of the University Graduate Center and Queens College of the City University of New York (CUNY). Professor Sy has over 60 prestigious journal and conference articles to document his funded research projects. He is the holder of the patent on his data mining technique for model discovery, and the lead author of the book entitled "Information-Statistical Data Mining" published by Kluwer Publishing (now Springer) at 2003 in its International Series in Engineering and Computer Science. He is a CISSP as certified by the International Information Systems Security Certification Consortium.