Trading Privacy for Trust in Online Interactions

Leszek Lilien[1,3] and Bharat Bhargava[2,3]

[1] WiSe (Wireless Sensornets) Lab, Department of Computer Science
Western Michigan University, Kalamazoo, MI 49008, U.S.A.

[2] RAID Lab, Department of Computer Science
Purdue University, West Lafayette, IN 47907, U.S.A.

[3] Center for Education and Research in Information Assurance and Security (CERIAS)
Purdue University, West Lafayette, IN 47907, U.S.A.

1. INTRODUCTION

Any interaction—from a simple transaction to a complex collaboration—can start only after an adequate level of trust exists between interacting entities. One of the more important components of trust of an entity E in its interaction partner is its reliance that the partner is both willing and able to protect E's privacy. This is true both in social systems and in the cyberspace.

The need for privacy is broadly recognized by individuals, businesses, the government, the computer industry, and academic researchers. Examples are shown in Figure 1. The growing recognition of the importance of privacy is motivated not only by users' sensitivity about their personal data. Other factors include business losses due to privacy violations, and enactments of federal and state privacy laws.

The role of trust and privacy is fundamental in social systems as well as in computing environments. The objective of this chapter is presenting this role in online interactions, emphasizing the close relationship between trust and privacy. In particular, we show how one's degree of privacy can be traded for a gain in the level of trust perceived by one's interaction partner.

We begin with a brief overview of these two basic notions in Section 2, presenting the background for research on trust, privacy, and related issues. First, we define trust and privacy, and then discuss their fundamental characteristics. Selecting the most relevant aspects of trust and privacy for a given computing environment and application is in and by itself a significant challenge (since both trust and privacy are very complex, multi-faceted concepts).

Privacy and trust in computing environments are as closely related and as interesting in various aspects of their interplay as they are in social systems (Bhargava ***et al***., 2004) --[6]. On the one hand, a high level of trust can be very advantageous. For example, an online seller might reward a highly trusted customer with special benefits, such as discounted prices and better quality of services. To gain trust, she can reveal private digital credentials—certificates, recommendations, or past interaction histories. On the other hand, a mere perception of a threat to users' privacy from a collaborator may result in substantial lowering of trust, again in both computing and social settings. In particular, any sharing of an entity's private information depends on satisfactory limits on its further dissemination, such as a partner's solid privacy policies. Just a potential for a privacy violation by an interaction partner impedes sharing of sensitive data among the interacting entities, which results in reduced effectiveness of the interaction and, in the extreme cases, even in the termination of the interaction. For instance, a user who learns that an ISP has carelessly revealed any customer's email will look for another ISP.

The possibility of trading privacy for trust, the main topic of this chapter, is explored in some depth in Section 3. It categorizes types of privacy-for-trust tradeoff, and shows how entities can trade their privacy for trust in an optimal way.

The remaining sections conclude this chapter, look into the future, and provide references and additional reading suggestions. Section 4 presents our view of future trends in research on privacy and trust. Section 5 includes conclusions, and Section 6 presents future research directions for privacy and trust in computing. Section 7 includes references, and Section 8 suggests additional reading material.

Recognition of the need for privacy by *individuals* (Cranor et al., 1999)

- 99% unwilling to reveal their SSN
- 18% unwilling to reveal their favorite TV show

Recognition of the need for privacy by *businesses*

- Online consumers worrying about revealing personal data held back $15 billion in online revenue in 2001 (Kelley 2001)

Recognition of the need for privacy by the *Federal Government*

- Privacy Act of 1974 for federal agencies (Privacy Act, 2004)=[44]
- Health Insurance Portability and Accountability Act of 1996 (HIPAA) (HIPAA Summary, 2003), (Mercuri, 2004)

Recognition of the need for privacy by *computer industry research* (examples)

- IBM—incl. Privacy Research Institute (IBM Privacy, 2007)
  - § Topics include: pseudonymity for e-commerce, EPA and EPAL—enterprise privacy architecture and language, RFID privacy, privacy-preserving video surveillance, federated identity management (for enterprise federations), privacy-preserving data mining and privacy-preserving mining of association rules, Hippocratic (privacy-preserving) databases, online privacy monitoring
- Microsoft Research—incl. Trustworthy Computing Initiative (Trustworthy Computing, 2003)
  - § The biggest research challenges: Reliability / Security / Privacy / Business Integrity
  - § Topics include: DRM—digital rights management (incl. watermarking surviving photo editing attacks), software rights protection, intellectual property and content protection, database privacy and privacy-preserving data mining, anonymous e-cash, anti-spyware

Recognition of the need for privacy by *academic researchers* (examples)

- Trust negotiation with controlled release of private credentials, privacy-trust tradeoff
- Trust negotiation languages
- Privacy metrics
- Anonymity and k-anonymity
- Privacy-preserving data mining and privacy-preserving database testing
- Privacy-preserving data dissemination
- Preserving location privacy in pervasive computing, and privacy-preserving location-based routing and services in networks,
- Trust negotiation with controlled release of private credentials
- Genomic privacy

Figure 1. Recognition of the need for privacy by different entities.

2. BACKGROUND: TRUST, PRIVACY, AND RELATED WORK

The notions of trust and privacy require an in-depth discussion of their background. It is provided in this section.

2.1. Trust and Its Characteristics

2.1.1. Definition of Trust

We define *trust* as as "reliance on the integrity, ability, or character of a person or thing" (American Heritage, 2000)--[1]. Use of trust is often implicit. Quite frequently it is gained offline (Bhargava *et al.*, 2004) --[6]. A user, who downloads a file from an unfamiliar Web site, trusts it implicitly by not even considering trust in a conscious way.

A user who decides to buy an Internet service from an Internet service provider may build her trust offline by asking her friends for recommendations.

An entity E expects that its interaction partner is both *willing* and *able* to protect E's privacy. This indicates that dimensions of trust include: *integrity* and *competence* of a trustee. That is, the integrity dimension of trust is a belief that a trustee is honest and acts in favor of the truster, and the competence dimension of trust is a belief in a trustee's ability or expertise to perform certain tasks in a specific situation. Predictability can be attached as a secondary measure to both an integrity belief and a competence belief (Zhong *et al*., 2006).

2.1.2. Implicit and Explicit Trust

Trust is truly ubiquitous and beneficial in social systems. The need for trust in one's interaction partner exists in all social interactions, irrespective of the fact whether the partner is an individual, an institution (e.g., a bank, a hospital, a used car dealer), or an artifact (e.g., a car, an Internet browser, a software house).

Trust is a powerful paradigm that enables smooth operation of social systems, also under conditions of uncertainty or incomplete information. It has been comprehensively used and well tested in social interactions and systems. For example, trust is constantly—if often unconsciously—applied in interactions between: people, businesses, institutions, animals (e.g., a guide dog) or artifacts (e.g., "Can I rely on my car for this long trip?").

Trust has to be approached differently in closed and open systems. In the former, trustworthiness of interaction partners is known to an initiator of interaction before the interaction starts, and in the latter it is not known. An example of a *closed social system* is a small village where people know each other (or at least know each other's reputations). Trust is used *implicitly* since each villager knows what to expect of everybody else. In short, "Mr. X 'feels' how much to trust Ms. Y." An example of an *open social system* is a large city where trust must be used *explicitly* to avoid unpleasant surprises (such as being harmed by a dishonest or incompetent car mechanic or dentist). A city dweller needs to ask around to find a trusted entity she needs (such as a trustworthy car mechanic or dentist), inquiring friends, office mates, etc. She can inquire among friends, office mates, etc., or check professional „reputation databases," such as AAA's Approved Auto Repair Network, or the Better Business Bureau (BBB).

Trust has proven its usefulness in social systems. We need similarly ubiquitous, efficient and effective trust mechanisms in the cyberspace. We have both closed systems—such as a LAN serving a research lab—and opened environments—such as the World Wide Web or WiFi networks. Only the latter include users who are not known in advance to their interaction partners. An access control system for a WiFi hot spot is an example of such a partner in an open system that must determine the permitted actions of each unknown user before an interaction can start.

We believe that many users or computer systems err by not considering trust issue at all. They do not assume trust implicitly. They simply ignore the issue of trust. Without even knowing it, they trust *blindly* (i.e., trust without evidence or verification). For example, this error is made by any operating system that trusts all application programs, allowing any program to run. As another example, too many users do not even know that they show a naïve trust by accessing unknown web site, which can harm them or their computers.

Still, closed computing environments systems (analogous to a small village) have been working well without applying the notion of trust, at least explicitly. However, it becomes more and more difficult to handle open computing systems (analogous to a big city) without the assistance from the powerful trust paradigm. In the security area, for example, the confidentiality-integrity-availability (CIA) paradigm has served sufficiently well in closed systems but it has to be replaced or augmented with trust-based solutions in open environments (such as the Web). Using the trust paradigm simplifies security problems by reducing complexity of interactions among system components, both human and artificial ones.

Summarizing, an adequate degree of trust is required to enable interaction, from a simple transaction to a complex collaboration, in social or computer systems. Parties to an interaction must build up trust in each other, irrespective of the fact whether these are human or artificial partners, and whether the interaction is offline or online.

2.1.3. Selected Trust Characteristics

Trust is a very complex and multi-faceted notion. A researcher wishing to use trust in computing systems must cope with the challenging choice of the optimal subset of trust characteristics. A vast variety of different trust-based systems can result from selecting different subsets. Some of the choices will make systems based on them ineffective or inefficient.

Some of the choices for trust characteristics include the following:

1. Symmetric and asymmetric trust.
   The former assumes that "A trusts B" implies "B trusts A," which in general is not true. Therefore, asymmetric trust is more general. Symmetric trust can be viewed as its special case, which can be assumed only in very special circumstances or applications.

2. Degrees of trust vs. binary trust
   The former is more precise, allowing for *degrees* of trust (from multi-level to continuous trust), while the latter, is *all-or-nothing* trust, which forces to specify a single trust threshold above which full trust can be assumed. Binary trust is insufficient in general, and can be assumed only for very special and limited settings.

3. Explicit or implicit trust
   Implicit trust is used by either ignorant or naïve interaction parties. For instance, a user, who downloads a file from an unfamiliar Web site, trusts it implicitly by not even considering trust in a conscious way. The consequences might include penetration by malware.

   Explicit trust allows for its clear specification, assuring that trust considerations are not ignored. Given A's need for determining trustworthiness of B, only explicit trust allows for determination of the party that vouches for trustworthiness of B, and assumes risks when this trust is breached. It may, but does not have to be the case, that B vouches for its own trustworthiness (e.g., via its behavior in earlier interactions with A).

   Explicit trust might be gained offline. For instance, a person who decides to buy an Internet service from an Internet service provider (ISP) may build her trust offline by asking her friends for trustworthy ISPs.

4. Direct or indirect trust.
   Direct trust between A and B (as in: "A trusts B") is limited to cases when A has gained a degree of trust in B from previous interactions. (This may, but does not have to, mean that B gained any degree of trust in A.)

   It is obvious that the domain of trust can be significantly extended by relying not only on direct trust but also on *indirect trust*. For indirect trust, A does not need to trust B to be willing to interact with it. It is sufficient that A finds an intermediary C such that A has a sufficient degree of trust in C and C trusts B. (To be more precise, in this case A needs to trust to a sufficient degree in C's recommendations about trustworthiness of B).

   C becomes a *trusted third party* (*TTP*). A TTP can be any entity accepted by Entity A, in particular, it can be an institution set up to provide indirect trust, also on a commercial basis.

5. Type of trusted entities.
   Should trust be lavished only on humans? The answer is clearly "no." We trust our refrigerators, cars, cellphones, PDAs, or RIF tags in stores. As is the case with humans, this trust can be breached if the devices are loyal to other parties than their owners or primary users (such as a leaseholders of a sensor-rich apartment). Loyalty decides who the entrusted party works for. For example, sensors and recorders in a car can work not for the driver but for an insurer, a browser can work for a commercial advertiser, and a sensor network in one's home can be hijacked by a nosy neighbor or—in the worst case—by the Big Brother.

6. Number of trusted entities.
   The most critical distinction is between trusting somebody or trusting nobody. The latter leads to paranoid

behavior, with extremely negative consequences on system performance (inclusing costs). We believe that "You can't trust everybody but you have to trust somebody."

Trusting more partners improve performance as long as trust is not abused. Any breach of trust causes performance penalties. An optimal number of trusted entities should be determined.

7. Responsibility for breaches of trust.
   If no TTP is involved, is the trustor or the trustee responsible for deciding on the degree of trust required to offer or accept a service? As a consequence, is the trustor or the trustee ultimately for possible breaches of trust?

   In commercial relationships, most often a buyer determines whether the seller is trustworthy enough and then—at least once the warranty period is over—bears the costs of broken trust. There are, however, cases when it is the seller pays for abuses by the buyer (as in the case when terrorists are not prevented from boarding a plane).

   If a TTP is involved in a trust relationship, it may be held responsible for to the extent allowed by its legal obligations.

2.1.4. Caveats

A few words of caution are in order (Bhargava *et al*., 2004). First, using a trust model too complex for an application domain (i.e., including superfluous trust aspects) hurt flexibility or performance. Second, excessive demands for evidence or credentials result in laborious and uncomfortable trust-based interactions, while insufficient requirements make them too lax. (In the latter case, who wants to be friends with someone who befriends crooks and thieves?) Third, exaggerating the need for explicit trust relationships hurts performance. For example, modules in a well-integrated (hence, closed) system should rely on implicit trust, just as villagers do. Also, in a crowd of entities, only some communicate directly, so only they need to use trust. But even not all of them need to use trust explicitly.

2.2. Privacy and Its Characteristics

2.2.1. Definition of Privacy

We define *privacy* as "the right of an entity (normally a person), acting in its own behalf, to determine the degree to which it will interact with its environment, including the degree to which the entity is willing to share information about itself with others" (Internet Society, 2007)--[13]. We fully embrace the possibility—indicated by the words "an entity (normally a person)"—to extend the scope of the notion of privacy from a *person* to an *entity*. The latter may be an organization, an artifact (software in particular), etc. The extension is consistent with the use of the notion of trust" also in relationship to artifacts (American Heritage, 2000)--[1], and with the common practice of antropomorphization of intelligent system components (such as objects and agents) in computer science. The extension is useful for discussion of privacy not only for humans but also for artificial entities (acting, more or less directly, on behalf of humans).

2.2.2. Selected Privacy Characteristics

Privacy has 3 dimensions: (a) *personal privacy* of an entity—demanding protecting an entity against undue interference (such as physical searches) and information that violates moral sense of the entity; (b) *territorial privacy*—calling for protection of the area surrounding the entity (such as laws on trespassing); and (c) *informational privacy*—requiring protection of gathering, compilation and dissemination of information (Fischer-Hübner, 2001).

Any interaction involves exchange of data. It is hard to find any data that (at least in conjunction with other data, also offline data) does not carry any private information on its sender. Hence, *informational privacy* is endangered

since each interaction involves release or dissemination of such private information.

The release of private data can be controlled in various degrees: from none to full control. It can also be categorized as voluntary, "pseudo-voluntary," or mandatory (incl. the case of information release as required by law). The pseudo-voluntary data dissemination is particularly deceitful since it appears to give a user a freedom to decline sharing his private information but only at the cost of denying the user an access to a desirable service. As a simple example, a person who refuses for privacy reasons (including fears of more spam) to enter the email address on a web site can be denied the site's services. Quite often, in the name of a real need or just a convenience the user is forced or pressured to provide private data. (This tradeoff between privacy and convenience should be studied.)

The amount or degree of privacy lost by disclosing a piece of information is affected by the identity of the recipients of this information, possible uses of this information, and related private information disclosed in the past. First, the recipients of private information include not only direct but also all indirect recipients, who receive some of this private information from entities other than the user. For example, a doctor, the direct recipient of private patient's information, passes some of this information to the insurer, an indirect recipient. Any indirect recipient can disseminate information further. In our example, the insurer can pass some information to user's employer. Second, possible uses of information vary from completely benevolent to the most malicious ones, with the latter including identity theft. Third, related private information disclosed in the past has a life of its own, like a genie out of the bottle. At best it is limited only by the controls that its owner was able to impose on its dissemination (e.g., asking a company not to sell to or share it with other businesses). At worst, it can be retrieved and combined with all pieces of information about the owner, destroying much of owner's privacy.

2.2.3. Threats to Privacy

Threats to privacy can be classified into four categories (Fischer-Hübner, 2003):

1. Threats to privacy at application level
2. Threats to privacy at communication level
3. Threats to privacy at system level
4. Threats to privacy in audit trails

In the first category, threats to privacy at the application level are due to collection and transmission of large quantities of personal data. Prominent examples of these types of threats are projects for new applications on the information highway, e.g.: public administration networks, health networks, research networks, electronic commerce, teleworking, distance learning, and private use.

In the second category, threats to privacy at the communication level include risks to anonymity of communication, such as: (a) threats to anonymity of sender, forwarder, or receiver; (b) threats to anonymity of service provider; and (c) threats to privacy of communication (e.g., via monitoring, logging, and storage of transactional data).

In the third category, threats to privacy at the system level are due to attacks on the system I order to gain access to its data. For example, attacks on system access level can allow the attacker access to confidential databases.

In the fourth category, threats to privacy in audit trails are due to wealth of information included in system logs or audit trails. A special attention should be directed to consider logs and trails that gained an independent life, away from the system from which they were derived.

Another view of threats to privacy (Fischer-Hübner, 2003) categorizes the threats as:

1. Threats to aggregation and data mining.
2. Threats due to poor system security.
3. Government-related threats due, for example, to the facts that: (a) the government has a lot of people's most

private data (incl. data on taxes / homeland security / etc.; (b) it is difficult to find the right balance between people's privacy on the one hand and homeland security concerns on the other hand.

4. Threats due to use of Internet, e.g., intercepting of unencrypted e-mail, recording of visited web site, and attacks via Internet.
5. Threats due to corporate rights and business practices since, for instance, companies may collect data that even the U.S. government is *not* allowed to gather.
6. Threats due to many traps of "privacy for sale," that is, temptations to sell out one's privacy. Too often online offers that seem to be "free" are not really free since they require providing the "benefactor" with one's private data (e.g., providing one's data for a "free" frequent-buyer card)

2.2.4. Escalation of Threats to Privacy in Pervasive Computing

Pervasive computing will exacerbate the privacy problem (cf. Bhargava *et al*., 2004). Unless privacy is adequately protected, the progress of pervasive computing will be slowed down or derailed altogether. People will be surrounded by zillions of computing devices of all kinds, sizes, and aptitudes (Sensor Nation, 2004). Most of them will have limited or even rudimentary capabilities and will be quite small, such as radio frequency identification tags and smart dust. Most will be embedded in artifacts for everyday use, or even human bodies (with possibilities for both beneficial and apocalyptic consequences).

Pervasive devices with inherent communication capabilities might even self-organize into huge, *opportunistic* sensor networks (Lilien *et al*., 2006), (Lilien *et al*., 2007) able to spy anywhere, anytime, on everybody and everything within their midst. Without proper means of detection and neutralization, no one will be able to tell which and how many snoops are active, what data they collect, and who they work for. Questions such as "Can I trust my refrigerator?" will not be jokes—the refrigerator will be able to snitch on its owner's dietary misbehavior to the owner's doctor.

Will pervasive computing force us to abandon all hope for privacy? Will a cyberfly, with high-resolution camera eyes and supersensitive microphone ears, end privacy as we know it? Should a cyberfly[1] be too clever to end up in the soup, the only hope might be to develop cyberspiders. But cyberbirds might eat those up. So, we'll build a cybercat. And so on and so forth …

Radically changed reality demands new approaches to computer security and privacy. Will a new privacy category appear—namely, protecting artificial entities' privacy? We believe that socially based paradigms, such as trust-based approaches, will play a big role in pervasive computing. As in social settings, solutions will vary from heavyweight ones for entities of high intelligence and capabilities (such as humans and intelligent systems) interacting in complex and important matters, to lightweight ones for less intelligent and capable entities interacting in simpler matters of lesser consequence.

2.3. Interplay of Privacy and Trust

Privacy and trust can be in a symbiotic or in an adversarial relationship. We concentrate here on the latter, when users in interactions with businesses and institutions face tradeoffs between a loss of their privacy and the corresponding gain of trust by their partners. (An example of the former is the situation when a better privacy provided by a commercial web site results in its customers' higher degree of trust.)

Users entering an online interaction want to gain a certain level of trust with the least loss of their privacy. This is the level of trust that is required by her interaction partner, e.g. a Web site, to provide a needed service, e.g., an online purchase of a gift. The interaction partner will asks for certain private information, such as certain credentials, e.g., her card and cardholder information. These credentials, when provided online, are indeed digital credentials—despite the

[1] A successful construction of a cyberfly or "the first robot to achieve liftoff that's modeled on a fly and built on such a small scale" was just reported ((Ross, 2007).

fact that non-digital credentials, such as a credit card, are their basis.

This simple scenario shows how privacy and trust are intertwined. The digital credentials are used to build trust, while providing the credentials reduces user's degree of privacy. It should be noted that in a closed environment, a user could receive certain service with revealing much less private information. For example, a student can order free educational software just by logging into a password-protected account, without any need for providing his credit card information. Obviously, entering only one's login and password is less revealing than providing one's credit card information.

All elements system "elements" that affect trust, affect also the interplay of privacy and trust. Trust is affected by a large number of system elements, including: (a) quality and integrity data; (b) trustworthiness of end-to-end communication, including sender authentication, message integrity, etc.; and (c) security of network routing algorithms, including dealing with malicious peers, intruders, security attacks, etc.

Privacy and trust can not be provided for free or traded for free (under any cost measures). Only in an ideal world we would never loose our privacy in any interaction, would be fully trusted at the same time, and would be provided these benefits at no cost. In reality, we can only approach this optimum by providing *minimal* privacy disclosures—ones that are absolutely necessary to gain a level of trust required by the interaction partners. The mechanisms providing *minimal* privacy disclosures and trust carry costs (incl. costs of computation, communication, storage, extra traffic, additional delays, etc).

It is obvious that gaining a higher level of trust may require a larger loss of privacy. It should also be obvious that revealing more private information beyond certain point will produce no more trust gains, or at least, no more useful trust gains. For example, a student wishing to enter a tavern must show a proof of his age (a loss of privacy for trust gain). Showing his driver's license is entirely sufficient, and showing his passport, his tax statements, etc. would produce no more trust gains.

It should also be obvious that for each required level of trust we can determine (at least in theory) the minimal loss of privacy required to produce this level of trust. This means that users can (and usually want) to build a certain level of trust with this minimal loss of privacy. We want to automate the process of finding this optimal privacy-for-trust tradeoff, including automatic evaluation of a privacy loss and a trust gain. To this end, we must first provide appropriate measures of privacy and trust, and then quantify the tradeoff between privacy and trust. This quantification will assist a user in deciding whether or not to trade her privacy for the potential benefits gained from trust establishment. A number of questions, including the following, must be answered. How much privacy is lost by disclosing a specific piece of information? How much trust is gained by disclosing given data? How much does a user benefit by having a given trust gain? How much privacy a user is willing to sacrifice for a certain amount of trust gain? Only after answering these questions, we can design algorithms and mechanisms that will assist users in making rational privacy-for-trust decisions. Developed mechanisms can empower a user's decision making process, or even automate it based on policies or preferences predefined by the user. In the latter case, user provides only her policies or preferences to the system and then accepts system's decisions.

2.4. Related Work

2.4.1. Related Work on Privacy

Many conferences and journals, not only in the area of computer science or other technical disciplines, focus on privacy. We can mention only a few publications that affected our search for a privacy-for-trade solution presented in this chapter.

Reiter and Rubin (Reiter and Rubin, 1999)=[48] use the size of the anonymity set to measure the degree of anonymity. The anonymity set contains all the potential subjects that might have sent/received data. The size of the anonymity set does not capture the fact that not all senders in the set have an equal probability of sending a message. This may help the attacker in reducing the size of the set of potential senders. Therefore, the size of the anonymity set may be a misleading measure, showing a higher degree of privacy than it really is.

Another approach (Diaz et al., 2002)-=[23], (Serjantove and G. Danezis, 2002)=[56] uses entropy to measure the level of privacy that a system achieves. Differential entropy is used in (Agrawal and C. Aggarwal, 2001)=[5] to quantify the closeness of an attribute value estimated by an attacker to its original value. These papers assume a static model of the attacker, in the sense that the attacker does not accumulate information by watching the system over the time.

The Scrub system (Sweeney, 1996)=[59] can be used to de-identify personal patient's information. Privacy is ensured by filtering identifying information out of data exchanged between applications. The system searches through prescriptions, physician letters, and notes written by clinicians to replace with generic data information identifying patients, such as their names, phone numbers, and addresses. A database of personally-identifying information is used to detect the occurrences of such information. The database contains data such as first and last names, addresses, phones, social security numbers, employers, and birth dates. In addition, the system constructs templates for different information formats (e.g., different formats for writing phone numbers and dates). These templates are used to detect variants of personal information.

Collecting pieces of information from different sources and putting them together to reveal personal information is termed *data fusion* (Sweeney, 2001a)=[61]. Data fusion is more and more invasive due to the tremendous growth of information being electronically gathered on individuals (Sweeney, 2001b)=[62]. The Scrub system does not provide a sufficient protection against data fusion, that is, it does not assure complete anonymity. The Datafly system (Sweeney, 1998)=[60], (Sweeney, 2002b=[64] maintains anonymity, even if data are linked with other sources. While maintaining a practical use of data, Datafly automatically aggregates, substitutes, and removes information to maintain data privacy. Datafly achieves data privacy by employing the k- anonymity algorithm (Sweeney, 2002b)=[64], which provides a formal guarantee that an individual can not be distinguished from at least k - 1 other individuals.

Platform for Privacy Preferences (P3P) is the best-known protocol and a suite of tools for specifying privacy policies of a Web site, and preferences of Web users (Cranor , 2003)=[Cran03]. P3P is not intended to be a comprehensive privacy "solution" that would address all principles of Fair Information Practices (Trade Commission, 1998)=[UFTC98]. AT&T Privacy Bird is a prominent implementation of P3P (Privacy Bird, 2004)=[APBT04]. It is a tool that can be added to a web browser to keep its user aware of web site privacy policies. It can be used as a part of the proposed metadata-based privacy scheme.

We do not discuss here general security solutions which contribute to privacy protection. Examples include protecting software, mobile objects or agents from many types of attacks by either: (i) running them only on dedicated and tamper-resistant platforms—e.g., on secure coprocessors (Tygar & Yee, 1994)=[TyYe94]=[22]=68]; or (ii) by providing security on commodity hardware—e.g., a single partitioning a hardware platform into many isolated virtual machines or "closed boxes" (Garfinkel, 2003)=[GPCR03]=[12]; hardware and the closed box mechanism together form a trusted party. Examples include also protection of a software client (code) from a malicious host by *obfuscating*, *tamper-proofing*, or *watermarking* the code (Collberg & Thomborson, 2000)=[CoTh00]=19].

2.4.2. Related Work on Trust

The problem of establishing and maintaining trust in dynamic settings has attracted many researchers. One of the first formalized models of trust in computer science (Marsh, 1994)=[11] introduced the concepts widely used by other researchers, such as context and situational trust.

A comprehensive social trust model, based on surveying more than 60 papers across a wide range of disciplines, has been proposed by McKnight & Chervany (McKnight & Chervany, 2001) =[13c]. It has been validated via empirical experimental study (McKnight *et al.*, 2002) =[12c]. The model defines five conceptual trust elements (cf. Cofta, 2006): trusting behavior, trusting intention, trusting belief, institution-based trust, and disposition to trust. First, *trusting behavior* is an action that increases a truster's risk or makes the truster vulnerable to the trustee.

Second, *trusting intention* indicates that a truster is willing to engage in trusting behaviors with the trustee. A trusting intention implies a trust decision and leads to a trusting behaviors. Two subtypes of trusting intention are: (i) willingness to depend: the volitional preparedness to make oneself vulnerable to the trustee; and (ii) subjective probability of depending: the likelihood that a truster will depend on a trustee.

Third, *trusting belief* is a truster's subjective belief in the fact that a trustee has attributes beneficial to the truster. The followings are the four attributes used most often: (i) *competence*: a trustee has the ability or expertness to perform certain tasks; (ii) *benevolence*: a trustee cares about a truster's interests; (iii) *integrity*: a trustee is honest and keeps commitments; and (iv) *predictability*: a trustee's actions are sufficiently consistent (so future action can be predicated based on the knowledge of previous behavior).

Fourth, *institution-based trust* is the belief that proper structural conditions are in place to enhance the probability of achieving a successful outcome. Two subtypes of institution-based trust are: (i) *structural assurance*: the belief that structures deployed promote positive outcomes, where structures include guarantees, regulations, promises etc.; and (ii) situational normality: the belief that the properly ordered environments facilitate successful outcomes.

Finally, *disposition to trust* characterizes a truster's general propensity to depend on others across a broad spectrum of situations. Two subtypes of disposition to trust are: (i) faith in humanity: the general assumptions about trustees' integrity, competence, and benevolence (i.e.. a priori trusting beliefs); and (ii) *trusting stance*: a preference for the default trust-based strategy in relationships.

Zacharia and Maes proposed two reputation systems, SPORAS and HISTOS (Zacharia & Maes, 2000)=[24]. Reputations in SPORAS are global, i.e., a principal's reputation is the same from the perspective of any querier. HISTOS has the notion of personalized reputation, i.e. different queriers may get different reputation values about the same principal. In addition to the reputation value, a reputation deviation is provided to measure the reliability of the value. Discarding notorious identity is unprofitable in SPORAS and HISTOS, because a newcomer starts with the lowest reputation value. Carbo et al. propose a trust management approach using fuzzy reputation (Carbo *et al*., 2003)=[5]. The basic idea is similar to that of SPORAS.

A distributed personalized reputation management approach for e-commerce is proposed by Yu et al. (Yu & Singh, 2002a)=[22], (Yu & Singh, 2002b)=[23]. The authors adopt the ideas from Dampster-Shafer theory of evidence to represent and evaluate reputation. If two principals *a* and *b* have direct interactions, *b* evaluates *a*'s reputation based on the ratings of these interactions. This reputation is called local belief. Otherwise, *b* queries on TrustNet for other principals' local beliefs about *a*. The reputation of *a* is computed based on the gathered local beliefs using Dampster-Shafer theory. How to build and maintain TrustNet is not mentioned in the paper. Aberer and Despotovic simplify this model and apply it to manage trust in a P2P system (Aberer & Despotovic, 2001)=[1].

Sabater and Sierra propose a reputation model for gregarious societies called Regret system (Sabater & Sierra, 2002)=[20]. The authors assume that a principal owns a set of sociograms describing the social relations in the environment. The Regret system structure has three dimensions. The individual dimension models the direct experience between two principals. The social dimension models the information coming from other principals. The ontology dimension models how to combine reputations on different aspects. Witness reputation, neighborhood reputation, and system reputation are defined. The performance of this approach highly depends on the underlying sociograms. The paper does not discuss how to build sociograms.

Lik Mui *et al*. uses Bayesian analysis approach to model reputation and trust (Mui, 2002)=[16], (Mui *et al*., 2002)=[17]. Many reputation models and security mechanisms assume the existence of a social network (Barnes & Cerrito, 1998)=[2]. The idea of evaluating reputation based on implicit feedbacks has been investigated. Pujol *et al*. propose an approach to extract reputation from the social network topology that encodes reputation information (Pujol *et al*., 2002)=[18]. Morinaga *et al*. propose an approach to mining product reputations on the web (Morinaga *et al*., 2002)=[15]].

2.4.3. Related Work on Privacy-trust Optimization

The research on automated trust negotiation (ATN) investigates the issues of iteratively exchanging credentials between two entities to incrementally establish trust (Yu, Winslett, and Seamons, 2003)=[73]. This approach considers the tradeoff between the length of the negotiation, the amount of information disclosed, and the computation effort. The major difference between ATN and the proposed research is that we focus on the tradeoff between privacy and trust. Our research leads to an elaborate method for estimating the privacy loss, due to disclosing a piece of information, and for making rational decisions.

Wegella *et al.* present a formal model for trust-based decision making (Wegella *et al.*, 2003)=[71]. An approach is provided to manage trust lifecycle with considerations of both trust and risk assessments. This approach and our research on trust and evidence formalization (Bhargava and Y. Zhong, 2002)=[13] can be extended to use the trustworthiness of an information receiver to decide whether or not to disclose private information to him.

Seigneur and Jensen propose an approach to trade minimal privacy for the required trust (Seigneur & Jensen, 2004)=[55]. Privacy is based on a multiple-to-one linkability of pieces of evidence to a pseudonym. It is measured by *nymity* (Goldberg, 2000)=[31]. The authors assume the presence of a partial order of *nymity* levels for the measurement of privacy. Our proposed research considers multiple-to-multiple relationships between pieces of evidence and private attributes.

3. TRADING PRIVACY FOR TRUST

3.1. Problems in Trading Privacy for Trust

To gain trust, a customer must reveal private digital credentials—certificates, recommendations, or past interaction histories. He is faced with a number of tough questions:

- How much privacy is lost by disclosing a specific credential? (To make the answer even more difficult, the amount of privacy loss is affected by credentials and information disclosed in the past.)
- How many credentials should a user reveal? If alternative credentials are available (e.g., either a driver license or a passport indicates birth data), which ones should be revealed?
- How much trust is gained by disclosing a given credential? This is referred to as the *trust gain*. Also, what is the minimal degree of privacy that must be sacrificed to obtain a required amount of trust gain? Which credentials should be presented to satisfy this minimum requirement?
- How much does a user benefit by having a given trust gain?
- How much privacy a user is willing to sacrifice for a certain amount of trust gain?

These questions alone show how complex and difficult is optimization of the privacy-for-trust exchange. Obtaining an optimal solution without a technical support is practically impossible. There is only a small chance that intuitive approaches to this process will result in outcomes close to the optimal ones.

3.2. A Solution for Trading Privacy for Trust

This section presents our proposed solution facilitating privacy-for-trust trading and enabling optimal outcomes of this process. It discusses in turn proposed approaches for: building and verifying trust, protecting privacy, and trading privacy for trust.

3.2.1. Building and Verifying Trust

We focus on methods of building trust in opened and dynamic computing environments, which are more

challenging than the closed ands static settings.

Digital credentials are common means of building trust in open environments. Credentials include certificates (Farrell & Housley, 2002)=[26], recommendations, or past transaction histories (Fujimura & Nishihara, 2003)=[28]. Since credentials contain private information (e.g., user's identity and shopping preferences), their use involves on "trading" privacy for trust. We need to consider problems with credentials, including their imperfect and non-uniform trustworthiness. Since no credentials are perfect, means to verify trust are necessary. We present basic ways of verifying trust.

A. Trust Metrics

Trust cannot be built or verified without having measures of trust, which can be determined in many ways. We propose a three-step method for defining a trust *gain* metric.

In the first step, we need to determine multilevel trust metrics with n trust levels, measured on a numeric scale from 1- n, where n could be an arbitrarily large number. Such metric is generic, applicable to a broad range of applications, with the value of n determined for a particular application or a set of applications. The case of $n = 2$ reduces multilevel trust to the simplistic case of binary trust (it might still be useful in simple trust-based applications), with trust levels named, perhaps, *full_service* and *no_service*. Selecting $n = 5$ results in having 5 trust levels that could be named: *no_service*, *minimal_service*, *limited_service*, *full_service*, and *privileged_service* going from the lowest to the highest level.

Trust levels could be defined by a service provider, the owner of a Web site on which it resides (which might be different from the service provider), or any other entity that is an intermediary between the service provider and the customer or end user. The number of levels n could be increased when the site outgrows its old trust metric, or when the user becomes more sophisticated and needs or wants to use more trust levels.

In the second step, a *trust benefit function* $B(t_i)$, associated with each trust level t_i, needs be defined. The default trust benefit function for a service can be defined by the same party that defined test levels in the preceding step (i.e., by a service provider, by the owner of a Web site on which it, or intermediary). An optional trust benefit function, overriding the default one, can also be defined by an individual customer, allowing for more user-specific benefit metric.

In order to compute the trust gain, a trust benefit function $B(t_i)$ is associated with each trust level t_i,. The value of the benefit function can either be provided by the service provider (the owner of a Web site), or be evaluated by an individual user using her own utility function.

In the third step, *trust gain*, denoted by $G(t_2, t_1)$, can be calculated based on the benefit function. $G(t_2, t_1)$, indicates how much a user gains if the user's trust level, as seen by the user's interaction partner, increases from t_1 to t_2. The following simple formula is used to compute the trust gain:

$$\text{trust gain} = G(\textit{new\_trust\_level}, \textit{old\_trust\_level}) = B(\textit{new\_trust\_level}) - B(\textit{old\_trust\_level})$$

B. Methods for Building Trust

Some of the many generic means of building trust are listed in Figure 2. They include familiarity with the entity to be trusted or its affiliation with a familiar entity, as well as building trust by first-hand *experience* or second-hand *reputation*.

Building trust by *familiarity* with X

- Person: face, voice, handwriting, *etc*.
- Institution: company name, image, good will, *etc*.
- Artifact: manufacturer name, perceived quality, *etc*.

Building trust by *affiliation* of X with person/institution/artifact Y

- Trust or distrust towards Y rubs off on X

Building trust by first-hand *experience* with X's activities/performance

- Good or bad experience (trust or distrust grows)

Building trust by second-hand *reputation* of X determined by evidence or credentials

- Reputation databases (e.g., BBB, industry organizations, etc.) with „good" evidence or lack of „bad" evidence)
- Credentials: X's driver license, library card, credit card

Figure 2. Basic means of building trust among partners.

Rather than looking at ways of building trust in general, we differentiate them depending on the relative strengths of the interacting parties. The *strength* of a party *P1* participating in an interaction with another party, *P2*, is defined by *P1*'s capability to demand private information from *P2*, and *P1*'s means available in case when *P2* refuses to comply. As a simple example, a bank is stronger than a customer requesting a mortgage loan. As another example, two small businesses negotiating a contract are, in most cases, equally strong.

We concentrate on asymmetric trust relationships, in which one party is stronger and another weaker, for example, trust relationships between individuals and institutions, or between small business and large businesses. We ignore trust relationships with "same-strength" partners, such as individual-to-individual interactions and most B2B interactions. We will interchangeably use the terms: "a weaker partner" and "a customer," as well as "a stronger partner" and "a company."

Example means of building trust by a company in a customer include receiving a cash payment for a service provided, or checking partner's records in the e-Bay reputation databases. Example means of building trust by a customer in a company include asking friends about company's reputation, or checking its reputation in Better Business Bureau databases.

Multiple means of building trust by a stronger partner in the weaker partner are shown in Figure 3. They can assist a company in a fight against fraud attempts by a customer. All these means can be divided into privacy-preserving means of the weaker partner, and the means not preserving privacy. Only the first item listed in Figure 3 ("Ask partner for an *anonymous payment* for goods or services"), belongs to the privacy-preserving means (by the virtue of preserving customer's anonymity). All others compromise customer's privacy and result in disclosing private information. This indicates that much more often than not successful interactions with the stronger party require that a weaker party trade its privacy loss for a trust gain required by this stronger party.

There are also multiple means of building trust by a weaker partner in the stronger partner, with some of them shown in Figure 4. All these means can assist a customer in a fight against fraud attempts by a company. It is clear that customer's major weapon is information on the company and its reputation.

C. Methods for Verifying Trust

Since no credentials are perfect, means to verify trust are necessary. This as true in computing as in social life.[2] The basic ways of verifying trust are shown in Figure 5.

Verification must be careful, not based on mere appearances of trustworthiness (which could be exploited by

[2] This includes politics. A Russian proverb „Trust but verify" was made famous in the mid 1980's by President Reagan, at the start of the historic negotiations with the General Secretary Gorbachev.

fraudsters). The cyberspace can facilitate more careful verification than is the case in the offline world, in which a careful verification might be too costly or too inconvenient.

Quite often a business order sent from Company A to Company B is processed without a careful verification. The reasons include the following factors: (a) verification is expensive, (b) *implicit* trust prevails in business, (c) risk of fraud or swindle is low among reputable businesses, and (d) Company B might be „insured" against being cheated by its business partners (that is, a trusted third-party intermediary assumes transaction risk; for example a buyer's bank could guarantee a transaction).

Ask partner for an *anonymous payment* for goods or services

- Cash / Digital cash / Other

---------- *above this line – privacy-preserving, below – privacy-revealing* ----------

Ask partner for a non-anonymous payment for goods or services

- Credit card / Traveler's Checks / Other

Ask partner for specific private information

Checks partner's credit history

Computer authorization subsystem observes partner's behavior

- Trustworthy or not, stable or not, …
- Problem: Needs time for a fair judgment

Computerized trading system checks partner's records in reputation databases

- e-Bay, PayPal, …

Computer system verifies partner's digital credentials

- Passwords, magnetic and chip cards, biometrics, …

Business protects itself against partner's misbehavior

- Trusted third-party, security deposit, prepayment, buying insurance, …

Figure 3. Means of building trust by a stronger partner in her weaker partner.

3.2.2. Protecting Privacy

Protecting privacy requires defining privacy metrics as a prerequisite. Privacy measures are discussed first. Methods for protecting privacy, relying on metrics, are presented next.

A. Privacy Metrics

We cannot protect privacy if we do not know how to measure it. This indicates the importance of privacy metrics. More specifically, we need privacy metric to determine what degree of data and communication privacy is provided by privacy protection methods. The metric has to work in any existing or future combination of users, techniques, and systems. It has to support or deny claims made by any such combination that a certain level for privacy will be maintained by it.

Ask around

- Family, friends, co-workers, …

Check partner's history and stated philosophy

- Accomplishments, failures and associated recoveries, …
- Mission, goals, policies (incl. privacy policies), …

Observe partner's behavior

- Trustworthy or not, stable or not, …
- Problem: Needs time for a fair judgment

Check reputation databases

- Better Business Bureau, consumer advocacy groups, …

Verify partner's credentials

- Certificates and awards, memberships in trust-building organizations (e.g., BBB), …

Protect yourself against partner's misbehavior

- Trusted third-party, security deposit, prepayment,, buying insurance, …

Figure 4. Means of building trust by a weaker partner in his stronger partner.

Verify one's experience

- Check own notes about X's activities/performance

Verify reputation evidence / credentials

- Call back to verify phone number
- Check user feedback about quality of artifact (online)
- Check reputation DB (e.g., consumer reports, BBB) for data

Verify affiliation

- Check with employer if X still employed
- Check reputation of Y with which X is affiliated

Figure 5. Basic ways of verifying trust toward Entity X.

This gives rise to at least two heterogeneity-related challenges. First, different privacy-preserving techniques or systems claim different degrees of data privacy. These claims are usually verified using ad hoc methods customized for each technique and system. While this approach can indicate the privacy level for each technique or system, it does not allow comparisons of different techniques or systems using various user models.

Second, privacy metrics themselves are usually ad hoc and customized for a user model and for a specific technique or system.

Requirements for good privacy metrics call for unified and comprehensive privacy measures to provide quantitative assessments of privacy levels achieved by diverse privacy-preserving techniques. A good privacy metric has to compare different techniques/systems confidently. It also has to account for: (a) operation of a broad range of privacy-preserving techniques; (b) dynamics of legitimate users—such as how users interact with the system, and awareness that repeated patterns of data access can leak information to a violator; (c) dynamics of violators—such as how much information a violator may gain by watching the system for some time; and (d) costs associated with metric implementation—such as storage, injected traffic, CPU cycles, and delay.

We propose to design metrics for assessing the privacy level of a given system. The metrics must be general enough to be used for comparing different privacy-preserving techniques. The objective of the privacy metrics is to answer questions such as: Given a specific user model, how much data privacy is maintained, and at what cost?

Two metrics are proposed to assess the privacy achieved by a given system: an anonymity set size metric and an information-theoretic (entropy-based) metric. The first metric can provide a quick assessment of privacy, while the second gives a more detailed insight into the privacy aspects of the system.

A.1. Effective Anonymity Set Size Metric Since anonymity is defined as the state of being indistinguishable within a set of subjects (Pfitzmann & Kohntopp, 2000)=[42], we can use the *size* of the anonymity set as privacy metrics. The basic idea is that of "hiding in a crowd" illustrated in Figure 6. As shown, hiding among n entities provides more privacy than hiding among 4 entities (for $n >> 4$). Clearly, the larger the set of indistinguishable entities, the lower probability of identifying any one of them. This approach can be generalized to "anonymize" not only identities of entities but also the values of their attributes: a selected attribute value is hidden within the domain of its all possible values.

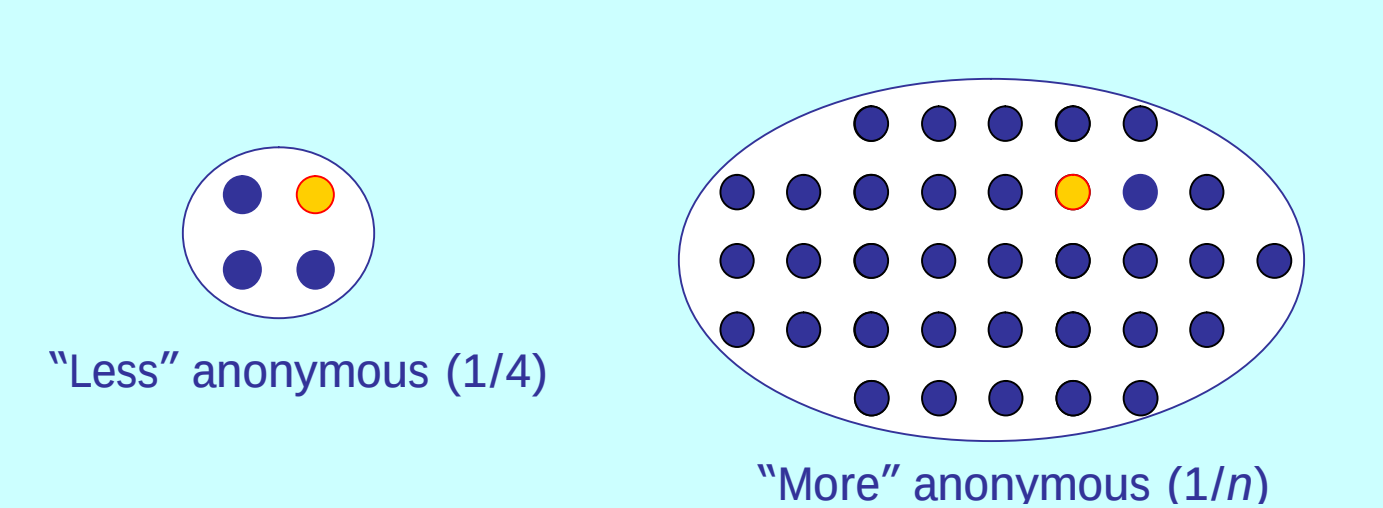

Figure 6. "Hiding in a crowd" underlying anonymity set size metrics.

We need to present this metric more precisely. The set of subjects, or values, is known as the *anonymity set*, denoted by A. Using the size of the anonymity set directly may indicate a stronger privacy than what it really is. The probability distribution that the violator can assign to individual subjects of the set should be considered. To illustrate this problem, consider a system that claims that a subject receiving data cannot be distinguished from $|A|$ other subjects of the anonymity set A. Suppose that a violator has noticed that a half of the nodes in A rarely receive messages. Then, he assigns to these nodes a very low probability of receiving a data item. The violator has effectively reduced the anonymity set size to $|A|/2$. To counter this problem, we define the anonymity set as: $A = \{(s_1, p_1), (s_2, p_2), \ldots, (s_n, p_n)\}$; where p_i represents the probability assigned to the subject s_i. Thus, we can determine the *effective* size of the anonymity set as:

$$L = |A| \sum_{i}^{|A|} \min(p_i, 1/|A|) \qquad (1)$$

Note that the maximum value for L is $|A|$, which happens when all entities in A are equally likely to access data, i.e., $p_i = 1/|A|$, $1 \le i \le n$. Equation (1) captures the fact that the anonymity set size is effectively reduced when the probability distribution is skewed, that is, when some entities have a higher probability of accessing data than the others.

A.2. Information-theoretic (Entropy-based) Metric Entropy measures the randomness in the system, and therefore, it measures the uncertainty that one has about that system (Cover & Thomas, 1991)=[21]. Building on this notion, we propose to use entropy to measure the level of privacy that a system achieves at a given moment. The idea is

that when an attacker gains more information about the system, the uncertainty about subjects that send or receive data, and thus their entropy, is decreased. By comparing a current entropy value with the maximum possible entropy value for the system, we can learn how much information the attacker has gained about the system. Therefore, the privacy of the system can be measure based on how much private information was lost.

a) Entropy Calculation Example Privacy loss $D(A,t)$ at time t, when a subset of attribute values A might have been disclosed, is given by:

$$D(A,t) = H^*(A) - H(A,t)$$

where: $H^*(A)$ is the maximum entropy (computed when probability distribution of p_i's is uniform), and $H(A, t)$ is entropy at time t given by:

$$H(A,t) = \sum_{j=1}^{|A|} w_j \left(\sum_{\forall i} (-p_i \ \log_2(p_i)) \right)$$

with w_j denoting weights capturing relative privacy "value" of the attributes.

Consider a private phone number: $(a_1\ a_2\ a_3)\ a_4\ a_5\ a_6 - a_7\ a_8\ a_9\ a_{10}$, where the first three digits constitute the area code. Assume that each digit is stored as a value of a separate attribute. Assume also that the range of values for each attribute is [0—9], and that all attributes are equally important, *i.e.*, for each $j \in$ [1-10], $w_j = 1$.

The maximum entropy exists when an attacker has no information about the probability distribution of the values of the attributes. As a consequence, the attacker assigns a ***uniform*** probability distribution to the attribute values. In this case, for each j, $a_j = i$ with $p_i = 0.1$ for each i, and we get:

$$H^*(A) = \sum_{j=0}^{9} \left(w_j \sum_{i=1}^{10} (-0.1\ \log_2(0.1)) \right) = 33.3$$

Suppose that after time t, the attacker can figure the state in which the phone number is registered, which may allow him to learn the three leftmost digits (at least for states with a single area code). Entropy at time t is given by:

$$H(A,t) = 0 + \sum_{j=4}^{10} w_j \left(\sum_{i=0}^{9} (-0.1\ \log_2(0.1)) \right) = 23.3$$

Note that attributes a_1, a_2, and a_3 contribute 0 to the entropy value because the attacker knows their correct values. Information loss at time t is:

$$D(A,t) = H^*(A) - H(A,t) = 10.0$$

b) Decrease of system entropy with attribute disclosures Decrease of system entropy with attribute disclosures is illustrated in Figure 7. The white circle indicates the size of the attribute set for private data under consideration, the darker circles within them indicate the sizes of subsets of disclosed attributes, the vertical lines to the left of the white circles indicate the maximum entropy H^*, and vertical bars to the left of the white circles (superimposed on the "H^* lines") indicate the current entropy level. Let us first consider Cases (a) – (c) in which we assume a fixed size of private data (this explains why the white circles in these three cases have the same size). When entropy reaches a certain higher threshold value H^2, as shown in Case (b), a controlled data distortions method (increasing entropy of these data) must be invoked to protect privacy of data. When entropy drops below a certain lower threshold level H^1, as shown in Case (c), data destruction must be triggered to destroy all private data (as the ultimate way of preventing data disclosure).

Let us add a bit more detail to this example. Consider private data that consists of a set of attributes, e.g., a name, a social security number, and a zip code. Each attribute has a domain of values. The owner of private data first computes the maximum entropy H^* for all attributes. She also determines two values for entropy mentioned above: the higher value H^2 (the threshold for triggering controlled data distortions) and the lower value H^1 (the threshold for triggering data destruction). Every time private data is being shared or transferred from one entity to another, entropy is recomputed using Equation (2). The new value of entropy, H^{new}, is calculated based on how much additional information is being disclosed at this time If H^{new} stays above H^2, no actions to prevent privacy disclosure are needed. If H^{new} drops below H^2 but stays above H^1 ($H^2 > H^{new} > H^1$), a controlled data distortions method is invoked on the

private data to increases its entropy. (Examples of distortion mechanisms include generalization and suppression (Sweeney, 2002a)=[63] or data evaporation (Lilien & Bhargava, 2006)=[p2d2].) Finally, if H^{new} drops below H^1 ($H^1 > H^{new}$), data destruction is invoked to destroy private data.

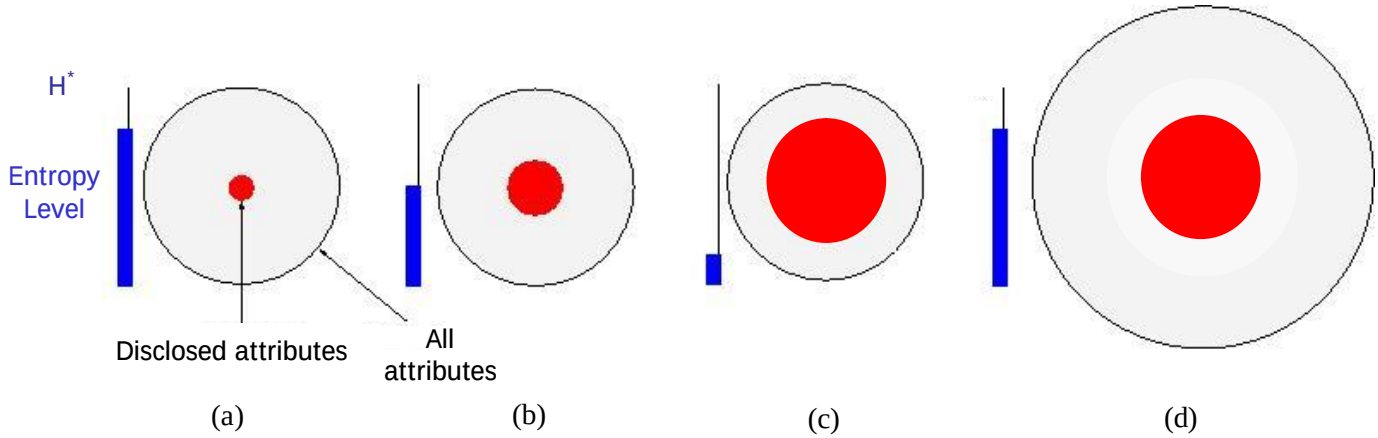

Figure 7. Dynamics of entropy.

The example above assumed that the size of the private data attribute set is fixed. The entropy metric can also be used in cases when the private data attribute set is allowed to grow or shrink. Case (d) in Figure 7, as compared with Case (c), illustrates the situation in which private data attribute set grew. This growth is indicated by the larger diameter of the white circle, indicating a larger attribute set for private data. The sizes of subsets of disclosed attributes, indicated by the red circles, are identical in Cases (d) and (c)—do not be fooled by the optical illusion that the red circle in (d) is smaller than in (c). As a result, entropy for Case (d) is higher than for Case (c), as indicated by a higher vertical bar for Case (d). This uses the principle of "hiding in the crowd" again.

Entropy can be increased not only by increasing the size of the private data attribute set, as shown above, but also by making its subset of disclosed attributes less valuable. For example, suppose that a bank releases the current account balance of a customer to the insurer. This balance is valid for a specific period of time. After this period, the value of knowing this private piece of information decreases, because the customer could have changed her balance. In computing the new value of entropy, the balance is assumed to be a private data again. This leads to a *gradual* increase in entropy. In another example, a bank can increase entropy *rapidly*: to make the stolen credit card numbers useless, it quickly changes the credit card numbers for the compromised accounts.

B. Methods for Protecting Privacy

Privacy controls for sensitive data are necessary. Without them, many interaction opportunities are lost. Examples are patients' symptoms hidden from doctors, given up business transactions, lost research collaborations, and rejected social contacts

Privacy can be supported by technical or legal controls. Examples of legal controls are the EPA privacy act (Privacy Act, 2004)=[44], and HIPAA Privacy Rule (HIPAA Summary, 2003, and Mercuri, 2004) intended to protect privacy of individuals. Yet, there are many examples of privacy violations even by federal agencies. The sharing of travellers' data by JetBlue Airways with the federal government was one such incident (Privacy Act, 2004)=[44].

Technical controls for facilitating or enabling privacy controls must complement legal controls. Such privacy protection mechanisms should empower users (peers, nodes, etc.) to protect various aspects of privacy, including user identity, privacy of user location and movement, as well as privacy in collaborations, data warehousing, and data dissemination. Each party that obtained through an interaction other party's sensitive data must protect privacy of these data. Forwarding them without proper privacy guarantees to other entities could endanger partner's privacy.

Both partners need appropriate protection mechanisms and should be assisted with technical solutions. On the one

hand, the responsibility of the stronger partner for protecting privacy is larger. The reason is that the stronger partner obtains more sensitive data of her counterpart then a weaker partner. In many cases, the stronger partner might be the only party that obtains private data.

On the other hand, the weaker partner should not rely entirely on the integrity of the stronger counterpart. He needs mechanisms to protect sensitive data released by him even—or *especially*—when they are out of his hands. This means that at least the following two kinds of mechanisms are needed. The first one must assist in minimizing the amount of private information that is disclosed. A system for privacy-for-trust exchange, presented in this chapter, is an example of a mechanism of this kind.

Mechanisms of the second kind provide protection for further dissemination of sensitive data that are already disclosed, setting clear and well-defined limits for such dissemination. They assure that data disclosed to a stronger party are not freely disseminated by her to other entities. For example, a mechanism of this kind assures that only some of data revealed by a patient to his doctor are forwarded by the doctor to an insurer or a nurse, and most sensitive data are never forwarded to anybody.[3] An example of this kind is the solution named *P2D2* (Privacy-Preserving Data Dissemination) (Lilien & Bhargava, 2006), which enables control of further dissemination of sensitive data by integrating privacy protection mechanisms with the data they guard. P2D2 relies on the ideas of *bundling* sensitive data with metadata, an *apoptosis*—that is, a clean self-destruction (Tschudin, 1999)—of endangered bundles, and an adaptive *evaporation* of bundles in suspect environments.

B.1. Technical Privacy Controls Technical privacy controls, also known as *Privacy-Enhancing Technologies* (*PETs*), can be categorized into the following categories (Fischer-Hübner, 2001):

a) Protecting user identities

b) Protecting usee identities

c) Protecting confidentiality and integrity of personal data

We take in turn a look at these categories of technologies.

a) Protecting User Identities (Fischer-Hübner, 2001) User identities can be protected via *anonymity, unobservability, unlinkability,* and *pseudonymity* of users. First, *anonymity* ensures that a user may use a resource or service without disclosing the user´s identity (Common Criteria 1999). The special cases of anonymity are: *sender anonymity*, ensuring that a user is anonymous in the role of a sender of a message, and *receiver anonymity*, ensuring that a user is anonymous in the role of a receiver of a message.

We can define the following six *degrees of sender anonymity*—from the one fully protecting the sender, to the one exposing the sender (Fischer-Hübner, 2001): (i) *absolute privacy*, when the sender enjoys full privacy w.r.t. to being considered as the sender of the message; (ii) *beyond suspicion*, when the sender appears to be no more likely to be the originator of a sent message than any other potential sender in the system; (iii) *probable innocence*, when the sender appears to be no more likely to be the originator of a sent message than not to be the originator; (iv) *possible innocence*, when there is a nontrivial probability that the real sender is someone else; (v) *exposed*, when the sender is highly likely to be the originator of a sent message; and (vi) *provably exposed*, when the sender is identified beyond any doubt as the originator of a sent message.

Second, *unobservability* ensures that a user may use a resource or service without others being able to observe that the resource or service is being used (Common Criteria 1999). Third, *unlinkability* ensures that a user may make use of resources and services without others being able to link these uses together (Common Criteria 1999). Its special case is *unlinkability of a sender and a recipient*, when a sender and a recipient cannot be identified as communicating with each other.

Fourth, *pseudonymity*: "ensures that a user acting under a pseudonym may use a resource or service without disclosing his identity (Common Criteria 1999).

[3] A special case of protection of this kind is preventing *any* forwarding of disseminated data by any party that received it directly from their owner.

b) Protecting Usee Identities (Fischer-Hübner, 2001) In this case, the protected entity is the subject described by data, that is, a usee—not the user of data. Usee identities can be protected, e.g., via *depersonalisation*, providing *anonymity* and *pseudonymity* of data subjects.

Depersonalisation (*anonymization*) of data subjects can be classified as a *perfect depersonalization*, when data are rendered anonymous in such a way that the usee (the data subject) is no longer identifiable, and a *practical depersonalization*, when a modification of personal data is such that information concerning personal or material circumstances can either no longer be attributed to an identified or identifiable individual, or can be so attributed only with a disproportionate amount of time, expense and labor. Attackers attempt to circumvent depersonalization by *reidentification* attacks.

c) Protecting confidentiality and integrity of personal data (Fischer-Hübner, 2001) Protecting confidentiality and integrity of personal data can be protected by a number of methods and technologies, including privacy-enhanced identity management, limiting access control (incl. formal privacy models for access control, also security models enforcing legal privacy requirements), using enterprise privacy policies, making data inaccessible with cryptography or steganography (e.g., hiding a message in an image), and utilizing specific tools (such as Platform for Privacy Preferences or P3P (Marchiori *et al* , 2002)=[40]).

B.2. Legal Privacy Controls For completeness of our presentation, we will take a look at legal privacy controls. Since the focus of this chapter is on computing technology solutions, this overview is concise.

Despite the fact that definitions of privacy vary according to context and environment, the belief that privacy is a fundamental human right, even one of the most important rights of the modern age, is internationally recognized At a minimum, individual countries protect inviolability of the home and secrecy of communications [Green, 2004].

There are two types of privacy laws in various countries [Green, 2004]: comprehensive laws, and sectoral laws. *Comprehensive laws* are general laws that govern the collection, use and dissemination of personal information by public and private sectors. They are enforced by commissioners or an independent enforcement body. The disadvantages of this approach include lack of resources for oversight and enforcement agencies, as well as the governmental control over the agencies. Comprehensive privacy laws are used in Australia, Canada, the European Union, and the United Kingdom.

Sectoral laws focus on specific sectors and avoid general laws. They benefit from being able to use a variety of specialized enforcement mechanisms, selecting the ones best suited for the sector they apply to. Their disadvantage is the need for a new legislation for each new sectoral technology. Sectoral privacy laws are used in the United States.

Disparate national privacy laws require international (or regional) agreements to bridge different privacy approaches. An example is the Safe Harbor Agreement, reached in July 2000, between the United States and the European Union (Safe Harbor, 2007). It has been criticized by privacy advocates and consumer groups in both the United States and European Union for inadequate enforcement, and relying too much on mere promises of participating companies.

3.2.3. Trading Privacy for Trust

An interacting entity can choose to trade its privacy for a corresponding gain in its partner's trust in it (Zhong & Bhargava, 2004). We believe that the scope of a privacy disclosure should be proportional to the benefit expected by the entity that discloses its private information. For example, a customer applying for a mortgage must (and is willing to) reveal much more personal data than a customer buying a book.

Having measures of trust and privacy defined above will allow precise observation of these two quantities, and precise answers to questions such as: (a) What degree of privacy is lost by disclosing given data? (b) How much trust is gained by disclosing given data? (c) What degree of privacy must be sacrificed to obtain a certain amount of trust gain?

A. Symmetric and Asymmetric Trust

Trust relationships can be symmetric—which occurs between partners of similar stature, or asymmetric—which occurs when one partner's position is stronger vis-à-vis the other's. The ***strength*** of a party participating in the relationship is defined by its capability to demand private information from the other party, and her means available in case when the other party refuses to comply. As a simple example, a bank is stronger than a customer requesting a mortgage loan.

B. Symmetric and Asymmetric Privacy-for-trust Negotiations

To realize a privacy-for-trust tradeoff, two interacting parties, P1 and P2, must negotiate how much privacy needs be reveled for trust. We categorize such negotiations as either: (1) symmetric—when both parties are of similar strength; or (2) asymmetric—when one party's position is stronger vis-à-vis the other's. In turn, symmetric privacy-for-trust negotiations can be either: (1a) privacy-revealing negotiations, in which parties disclose their certificates ***or*** policies to the party; or (1b) privacy-preserving negotiations, in which parties preserve privacy of their certificates ***and*** policies.

We compare all three kinds of privacy-for-trust negotiations—that is, (1a), (1b), and (2)—in terms of their behavior during the negotiations. This behavior includes defining trust level necessary to enter negotiations, growth of trust level during negotiation, and the final trust level sufficient for getting a service.

Symmetric negotiations are very popular in the research literature. Asymmetric negotiations, to the best of our knowledge have been defined by us.

B.1. Trust Growth in Symmetric Trust Negotiations Symmetric trust negotiations involve partners of similar strength.

a) Trust growth in privacy-revealing symmetric trust negotiations Considering trust growth in symmetric trust negotiations, let us focus first on privacy-revealing symmetric trust negotiations. They allow to reveal private certificates ***or*** policies to the negotiation partner.

Negotiations of this type can start only if an initial degree of trust exists between the parties. They must trust each other enough to reveal to each other some certificates and policies right away. From this point on, mutual trust grows in a stepwise manner as more private information is revealed by each party. Negotiation succeeds when a "sufficient" mutual trust is established by the time when the negotiation ends. It is "sufficient" for the requestor to obtain the desired service. This procedure is summarized in Figure 8.

An initial degree of trust necessary

- Must trust enough to reveal (some) certificates / policies right away

Stepwise trust growth in each other as more (possibly private) info about each other revealed

- Proportional to the number of ceritficates revealed to each other

Succeed if sufficient mutual trust established when negotiation completed

- „Sufficient" for the task at hand

Figure 8. Trust growth in privacy-revealing symmetric trust negotiations.

b) Trust growth in privacy-preserving symmetric trust negotiations Let us turn now to privacy-preserving

symmetric trust negotiations. They ensure that no private certificates *or* policies are revealed to the negotiation partner.

In contrast to privacy-revealing symmetric trust negotiations, negotiations of this type can without any initial trust. There are no intermediate degrees of trust established during the negotiations. They continue without mutual trust up to the moment when they succeed or fails. They succeed when a sufficient mutual trust is established by the time when the negotiation ends. This procedure is summarized in Figure 9.

Initial distrust

- No one wants to reveal any info to the partner

No intermediate degrees of trust established

- „From distrust to trust"

Succeed if sufficient mutual trust established when negotiation completed

- „ Sufficient" for the task at hand

Figure 9. Trust growth in privacy-preserving symmetric trust negotiations.

B.2. Trust Growth in Asymmetric Trust Negotiations Asymmetric trust negotiations involve partners of dissimilar strength, with one party's position visibly stronger vis-à-vis the other's.

Negotiations of this type can start only if at their start the weaker partner has a sufficient trust into the stronger partner. This trust is "sufficient" when the weaker party is ready for revealing *all* private information required to gain stronger party's trust necessary for obtainig a service from the stronger party. As negotiations continue, the weaker partner trades a (degree of) privacy loss for (a degree of) a trust gain as perceived by the stronger partner. It should be clear that the former looses a next degree of privacy when she reveals the next private certificate to the latter. (The only exception to privacy loss is the "no privacy loss" case in the *anonymity-preserving* example in „Stronger Building Trust in Weaker" shown in Figure 3).

Negotiations succeed when, by the time when the asymmetric trust negotiations end, the stronger party gains a sufficient trust into the weaker party to provide it the requested service. This procedure is summarized in Figure 10.

B.3. Summary of Privacy-for-trust Trading in Symmetric and Asymmetric Trust Negotiations Figure 11 summarizes trust growth in symmetric and asymmetric trust negotiations.

Initially, Weaker has a „sufficient" trust into Stronger

- Weaker must trust Stronger sufficiently to be ready for revealing *all* private information required to gain Stronger's sufficient trust

Weaker trades a (degree of) privacy loss for (a degree of) a trust gain as perceived by Stronger

- A next degree of privacy „lost" when a next certificate revealed to Stronger

„Sufficient" trust of Stronger into Weaker established when negotiation completed

- „ Sufficient" for the task at hand

Figure 10. Trust growth in asymmetric trust negotiations.

Trust growth in symmetric „disclosing" trust negotiations

- Initial degree of trust / stepwise trust growth / establishing mutual „full" trust
- Trades info for trust (information is *private* or *not*)

Trust growth in symmetric „preserving" trust negotiations (from distrust to trust)

- Initial distrust / no stepwise trust growth / establishes mutual „full" trust
- No trading of info for trust (info is private or not)

Trust growth in asymmetric trust negotiations

- Initial „full" trust of Weaker into Stronger and *no* trust of Stronger into Weaker / stepwise trust growth / establishes „full" trust of Stronger into Weaker
- Trades *private* info for trust

Figure 11. Summary of trust growth in symmetric and asymmetric trust negotiations.

C. Privacy-for-trust Optimization in Asymmetric Trust Negotiations The optimization procedure for trading privacy for trust in asymmetric trust negotiations presented below follows our approach (Zhong & Bhargava, 2004). It includes four steps:

1. Formalize the privacy-trust tradeoff problem.
2. Measure *privacy loss* due to disclosing a private credential set.
3. Measure *trust gain* due to disclosing a private credential set.
4. Develop algorithms and build a system that *minimize privacy loss* for required trust gain.

We distinguish two forms of privacy-for-trust optimization. The first one minimizes the loss of privacy by the weaker partner necessary for obtaining, in the eyes of the stronger partner, a certain trust level required to get a service. This is the form discuss in more detail below.

The second form of optimization finds the degree of privacy disclosure (loss) by the weaker partner necessary for maximizing the trust level obtained from the stronger partner. We do not discuss this form, just noticing that it is needed in situations when the weaker partner's benefits obtained form the stronger partner are proportional to the trust level attained in the eyes of the stronger partner.

We assume that a user has multiple choices on what private information to disclose (e.g, in response to an age query, a user can show a driver license, a passport, a birth certificate, etc.). Each user can make this selection decision independently.

C.1. Formulating the Tradeoff Problem Suppose that the private attributes we want to conceal are $a_1, a_2, \ldots, a_m$. A user has a set of credentials $\{c_1, c_2, \ldots, c_n\}$. A credential is classified as a direct or a linking one. A *direct credential* contains information that reveals private attribute values. A *linking credential* is associated with a set of credentials. An example of a linking credential is the ownership of a pseudonym. A service provider obtaining this credential can access a new set of credentials of the user under this pseudonym. A credential set can be partitioned by a service provider into *revealed* and *unrevealed* credential subsets. The partitions are denoted as ***R(s)*** and ***U(s)***, respectively, where s is the identity of a service provider.

The tradeoff problem can now be formulated as follows: choose from ***U(s)*** the next credential ***nc*** to be revealed in a way that minimizes privacy loss while satisfying trust requirements. In a more formal notation it can be represented as:

$$\min\{\Pr\textit{ivacyLoss}(nc \cup R(s)) - \Pr\textit{ivacyLoss}(R(s)) \mid \textit{nc satisfies trust requirements}\}$$

This problem can be investigated in two scenarios:

1. Service providers never collude to discover customer's private information. An individual version ***R(s)*** is maintained for each service provider and privacy loss is computed based on it.

2. Some service providers collude to discover customer's private information. A global version R_g that consists of all credentials disclosed to any service provider is maintained. Since the difference between $R(s)$ and R_g is transparent to the evaluation of privacy loss and trust gain, they both are denoted as R in further considerations**.**

The tradeoff problem changes to a multivariate problem if multiple attributes are taken into consideration. It is possible that selecting nc_1 is better than nc_2 for a_1 but worse for a_2. We assume the existence of an m-dimensional weight vector [w_1, w_2, …, w_m] associated with these private attributes. The vector determines the protection priority for the private attributes a_1, a_2, …, a_m, respectively. We can minimize either: (a) the weighted sum of privacy losses for all attributes, or (b) the privacy loss of the attribute with the highest protection weight.

Another factor affecting the tradeoff decision would be the purpose of data collection. It can be specified in the service provider's privacy policy statement, for instance, by using P3P (Marchiori *et al* , 2002)=[40]. *Pseudonymous analysis* and *individual decision* are two data collection purposes defined in P3P. The former states that the collected information will not be used to attempt to identify specific individuals. The latter tells that information may be used to determine the habits, interests, or other characteristics of individuals. A user could make different decisions based on the stated purpose. Furthermore, the service provider's trustworthiness to fulfill the declared privacy commitment can be taken into consideration.

C.2. Estimating Privacy Loss In order to make the tradeoff decision, we use metrics and inference procedures for representing and estimating privacy loss.

We distinguish the query-dependent and query-independent privacy losses. *Query-dependent privacy loss* for a credential *nc* is defined as the amount of information that *nc* provides in answering a specific query. The following example illustrates the query-dependent privacy loss for a credential. Suppose that the user's age is a private attribute. The first query asks: "Are you older than 15?" The second query tests the condition for joining a silver insurance plan, and asks: "Are you older than 50?". If a user has already presented a valid driver license, we are 100% sure that the answer to the first query is "yes" but the probability of answering "yes" to the second query by a person with a driver license is, say, 40%. Privacy loss for a revealed credential (a driver license) is here query-dependent since it varies for different queries: it is a full privacy loss (100%) for the first query, and only a partial ("probabilistic") privacy loss (40%) for the second query. This example also makes it clear that the value of revealed information (such as a driver license) can vary for different queries.

It is time for an example illustrating the query-independent privacy loss for a credential. Suppose that a user has already presented her driver license. It implies that she is older than 16. If she uses her Purdue undergraduate student ID as the next piece of evidence, a high query-independent privacy loss ensues—since this credential greatly reduces the probable range of her age. Let us consider the third query asking: "Are you an elementary school student?" The student ID is redundant as a credential for this because her previous revealed credential (the driver license) has already excluded this possibility. This shows that a credential having a high query-independent privacy loss may not necessarily be useful to answer a specific query.

Two types of methods can be used to measure a privacy loss: *probabilistic* methods and the predefined lattice method.

a) Probabilistic Methods for Estimating Privacy Loss The first type are *probabilistic methods*, one for evaluating query-dependent privacy losses and another for evaluating query-independent privacy losses. More specifically, the first probabilistic method evaluates the query-independent privacy loss for disclosing a credential c_i with respect to one attribute a_j that has a finite domain { v_1, v_2, …, v_k }. The probability of $a_j = v_i$ before disclosing the credential is $Prob(a_j = v_i \mid R)$. The probability of $a_j = v_i$ with a given credential c_i disclosed is $Prob(a_j = v_i \mid R \cup c_i)$. The privacy loss is measured as the difference between entropy values (Young, 1971)=[72]:

$$\mathrm{Pr}\,ivacyLoss_{a_j}(c_i \mid R) = \sum_{i=1}^{k} -P_i \log_2(P_i) - \sum_{i=1}^{k} -P_i^* \log_2(P_i^*)$$

where $P_i = Prob(a_j = v_i \mid R)$ and $P_i^* = Prob(a_j = v_i \mid R \cup c_i)$.

The second probabilistic method evaluates the query-dependent privacy loss based on the knowledge of the complete set of potential queries. Let $q_1, q_2, \ldots, q_n$ denote the n queries. Let pr_i be the probability that q_i is asked, and w_i be the corresponding weight indicating the protection priority of this query. We can now evaluate the privacy loss of disclosing a credential c_i in response to a query q_k. Suppose that there are r possible answers to the query. The domain of an attribute a_j is divided into r subsets $\{ qv_1^j, qv_2^j, ..., qv_r^j \}$ based on the query answer set. The privacy loss with respect to attribute a_j and query q_k is computed as:

$$\Pr ivacyLoss_{a_j,q_k}(c_i \mid R) = \sum_{i=1}^{r} -P_i \log_2(P_i) - \sum_{i=1}^{r} -P_i^* \log_2(P_i^*)$$

where $P_i = Prob(a_j \in qv_i^k \mid R)$ and $P_i^* = Prob(a_j \in qv_i^k \mid R \cup c_i)$.

The query-dependent privacy loss with respect to attribute a_j is evaluated by the following formula:

$$\Pr ivacyLoss_{a_j}(c_i \mid R) = \sum_{k=1}^{n} (\Pr ivacyLoss_{a_j,q_k} * \mathrm{pr}_k * w_k)$$

Bayes networks (Jensen, 1996)=[36] and kernel density estimation can be used for conditional probability estimation.

b) The Predefined Lattice Method for Estimating Privacy Loss The second type of methods that can be used for measuring a privacy loss is represented by the *predefined lattice method*. This method assumes that each credential is associated with a *tag* indicating its privacy level with respect to attribute a_j. The *tag set* is organized as a lattice (Donnellan, 1968)=[25a] in advance. Tags are assigned to each subset of credentials as follows. Suppose that T_B and T_A are two tags and $T_B \leq T_A$. T_A and T_B are assigned to credentials c_A and c_B, respectively, if the information inferred from c_A is more precise than what can be inferred from c_B. c_A determines a possible value set V_A for a_j, and c_B determines another set V_B. The formula to compute the privacy loss is:

$$\Pr ivacyLoss_{a_j}(c_i \mid \varnothing) = Tag(c_i)$$

$$\Pr ivacyLoss_{a_j}(c_i \mid R) = \mathrm{lub}(\Pr ivacyLoss_{a_j}(c_i \mid \varnothing), \Pr ivacyLoss_{a_j}(R \mid \varnothing)$$

where *lub* is the least upper bound operator (Donnellan, 1968)=[25a].

C.3. Estimating Trust Gain The way to compute the trust gain has already been shown in Section 3.2.1-A on Trust Metrics. It requires defining a trust benefit function $B(t_i)$ associated with each trust level t_i. Then the *trust gain* G can be calculated as follows:

trust gain = G(*new_trust_level*, *old_trust_level*) = B(*new_trust_level*) - B(*old_trust_level*)

C.4. PRETTY—a System Minimizing Privacy Loss for a Required Trust Gain A prototype software PRETTY (PRivatE and TrusTed sYstems) was developed at in the Raid Laboratory at Purdue University (Zhong & Bhargava, 2004)=[25b]. PRETTY utilizes the server/client architecture as shown in Figure 12. It uses as its component the existing TERA (Trust-Enhanced Role Assignment) prototype also developed in the Raid Lab at Purdue (Zhong *et al.*, 2004)=[75]. TERA evaluates the trust level of a user based on her behavior. It decides whether a user is authorized for an operation based on the policies, the credentials, and the level of trust. A user's trust value is dynamically updated when more data on her behavior becomes available.

The client component of PRETTY consists of the user application, the credential manager, the evaluator of trust gain and privacy loss, the privacy negotiator, and a set of privacy policies. The server component consists of the server application, the TERA server, the privacy negotiator, the set of privacy policies, the database, and the data disseminator.

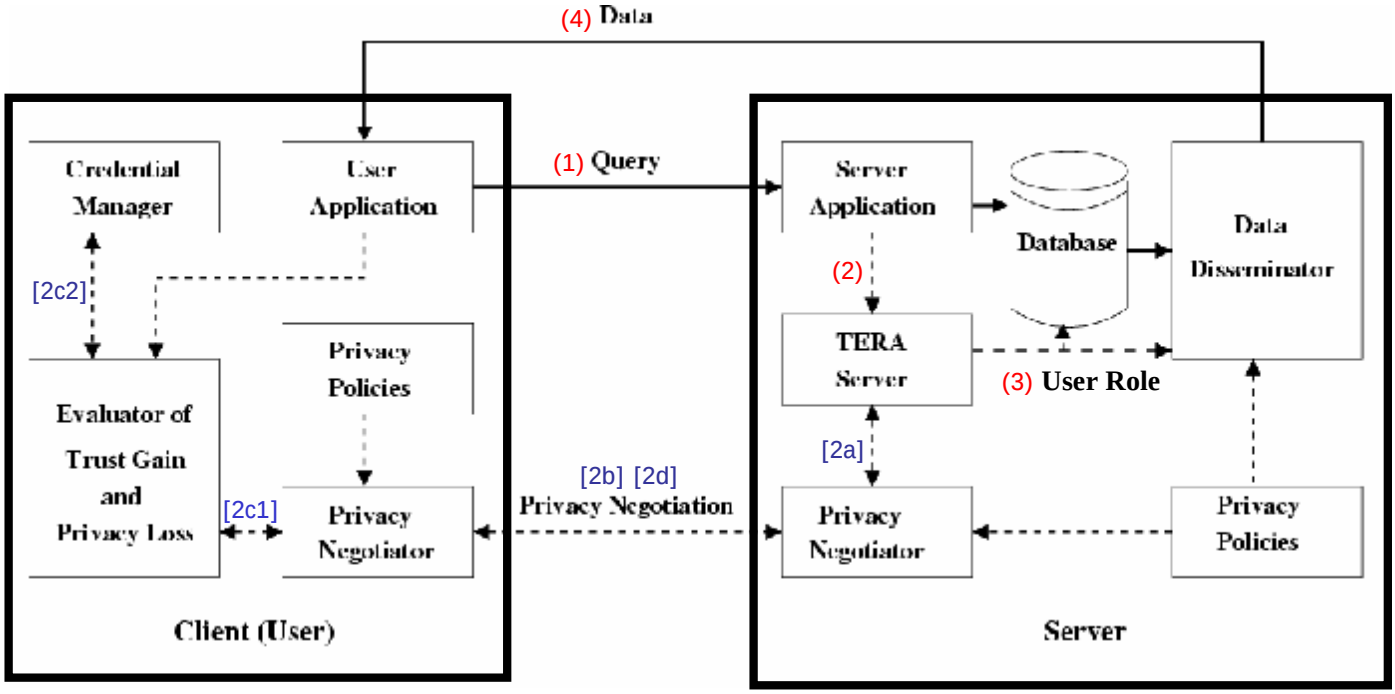

Figure 12. Architecture of PRETTY.

A typical interaction is illustrated in Figure 12. An arrow number in the figure (1 – 4, some with letter suffixes) corresponds to an item number in the interaction description below. A number in parentheses in the figure denotes an unconditional path (e.g., "(1)")—a path followed by all interactions, and a number in brackets denotes a conditional path (e.g., "[2a]")—a path followed only by some interactions. An interaction takes place as follows:

1) User application sends query to server application.
2) Server application sends user information to TERA server for trust evaluation and role assignment.
    2a) If a higher trust level is required for query, TERA server sends the request for more user's credentials to privacy negotiator.
    2b) Based on server's privacy policies and the credential requirements, privacy negotiator interacts with user's privacy negotiator to build a higher level of trust.
    2c) Trust gain and privacy loss evaluator selects credentials that will increase trust to the required level with the least privacy loss. Calculation considers credential requirements and credentials disclosed in previous interactions. (This item includes two actions: [2c1] and [2c2].)
    2d) According to privacy policies and calculated privacy loss, user's privacy negotiator decides whether or not to supply credentials to the server.
3) Once trust level meets the minimum requirements, appropriate roles are assigned to user for execution of his query.
4) Based on query results, user's trust level and privacy polices, data disseminator determines: (i) whether to distort data and if so to what degree, and (ii) what privacy enforcement metadata should be associated with it.

The *evaluator of trust gain and privacy loss* implements the ideas present in this paper. It allows users to specify the representation of a privacy loss and the strategies and prompts a user to enter the utility function for the ongoing interaction. The evaluator either automatically makes the tradeoff decision, or provides a user with the evaluation results for privacy loss and trust gain. The following methods are used for automatic decision making: (a) if the lattice-based representation of privacy loss is used, the decision procedure is to choose the least upper bound of privacy loss

of the candidate credentials; (b) if the numeric representation of privacy loss is used, search algorithms using the privacy loss as heuristic functions can be used. We are still developing mechanisms to precisely evaluate conditional probabilities for privacy loss. We plan to study the effectiveness and efficiency of the entropy-based and lattice-based privacy loss evaluation methods using PRETTY.

4. FUTURE TRENDS FOR PRIVACY AND TRUST RESEARCH

Technical privacy- and trust-related solutions will continue their strong impact on online consumer protection. The future trends related to privacy will be determined, among others, by the following challenges (Bhargava *et al*., 2003):

1. Defining and measuring privacy and its multifaceted aspects.
   How to define and assess quality, safety, and privacy of personal data? How to define metrics for this assessment?
2. Defining, analyzing, and managing privacy policies.
   How to define privacy policies? How to best perform privacy requirement analysis and stakeholder analysis? How to address privacy of primary and secondary uses of information? How to optimize digital rights management (DRM)?
3. Determining technologies endangering privacy in computing environments.
   What technologies (or system components) endanger privacy in computing environments, and how to prevent this? As an example, how to prevent pervasive computing from illegitimate monitoring and controlling people? How to assure anonymity in more and more pervasive computing environment? How to balance anonymity with accountability under these circumstances?
4. Finding new privacy-enhancing technologies.
   What technologies can be utilized or exploited to provide privacy, and how to use them to this end? For example, is there a way to insert "privacy monitors" or tools that provide alerts when privacy is endangered due to inference or careless transactions? What are the best ways of privacy-preserving data mining and querying? How to monitor Web privacy and prevent privacy invasions by undesirable inferences? How to address the issue of "monitoring the monitor," including identification and prevention of situations when incorrect monitor data result in a personal harm?

The future trends related to trust will be determined among others, by the following challenges (Bhargava *et al*., 2003):

1. Improving initiation and building of trust.
   How to create formal models of trust, addressing the issues of different types of trust (e.g., trust towards data, or users, or system components)? How to define trust metrics able to compare different trust models? How should trust models select and accommodate trust characteristics? How should the models of trust handle both direct evidence and second-hand recommendations related to the trusted subjects or objects? How trusted parties can be used to initiate and build trust? How timeliness, precision, and accuracy affect the process of trust building?
2. Maintaining and evaluating trust.
   How to collect and maintain trust data (e.g., credentials, evidence on the behavior of the trusted objects, recommendations)? How and when to evaluate trust data? How to discover betrayal of trust, and how to enforce accountability for damaging trust? How to prevent trust abuse, for example, by means of revocation of access rights? How to motivate users to be good citizens and to contribute to trust maintenance?
3. Constructing practical trust solutions.
   How to scale up trust models and solutions? What is the impact of trust solutions on system performance and economics? How to guarantee performance and economy of trust solutions? How and what economic incentives and penalties can be used for trust solutions?

4. Engineering trust-based applications and systems.
   How to experiment with and implement trust-based applications and systems for e-government, e-commerce, and other applications? How to enhance system performance, security, economics, etc., with trust-based ideas (such as enhancing role-based access control with trust-based mappings)?

5. CONCLUSION

Providing tools for privacy-for-trust exchange is critical to the further development of online interactions. Without privacy guarantees, there can be no trust, and without at least some trust no interactions can even commence—unless a party is totally oblivious to the dangers of privacy loss, up to the point of identity theft. Normally, people will avoid any *negotiations* if their privacy is threatened by a prospective negotiation partner. Without trust-building negotiations, no *trust* can be established.

The stakes are becoming higher since privacy guarantees are becoming absolutely essential as we progress towards pervasive computing. More pervasive devices have the higher potential for violating privacy. Unless adequate technical privacy controls and privacy-for-trust support is provided, possibilities of huge privacy losses will scare people off, crippling the promise of pervasive computing.

The objective of this chapter was presentation of an approach and a tool for protecting privacy in privacy-for-trust exchanges. We began with summarizing the role of trust and privacy in online interactions, emphasizing the tradeoff between these two phenomena. After an overview of trust with its characteristics and privacy with its characteristics, we discussed the issue of interplay of privacy and trust, emphasizing privacy-for-trust tradeoff. Next, an overview of problems facing a person wishing to trade privacy for trust was followed by a description of our proposed solution. It started with a look at trust metrics and means for building and verifying trust. We then discussed technical means for protecting privacy, preceded by a presentation of privacy metrics: an effective anonymity set size and an entropy-based metric.

We categorized the processes of trading privacy for trust into symmetric privacy-for-trust negotiations and asymmetric privacy-for-trust negotiations, dividing the former into privacy-revealing and privacy-preserving subcategories. The presented privacy-for-trust solution is intended for optimization in asymmetric trust negotiations. It involves four steps: formulating the tradeoff problem, estimating privacy loss, estimating trust gain, and minimizing privacy loss for a required trust gain. We provided a brief description of PRETTY, a system minimizing privacy loss for a required trust gain.

5. FUTURE RESEARCH DIRECTIONS IN TRADING PRIVACY FOR TRUST IN ONLINE INTERACTIONS

We have shown that privacy and trust enable and facilitate collaboration and communication. We indicated their growing role in open environments. To increase the benefits of privacy- and trust-related solutions, a number of research directions should be pursued (cf. (Bhargava *et al.*, 2003)). For privacy-related solutions, the following research problems should be addressed (*ibid*):

1. Privacy metrics.
   Issues of privacy of users or applications, on the one hand, and privacy (secrecy, confidentiality) of data, on the other hand, intertwine. Metrics for personal and confidential data usage should be developed. They should include measures of who and how accesses data, what data are accessed, and for how long. Metrics and methods for measurements of privacy-related aspects of data quality should be provided. Researchers should also propose measures of accuracy in information extraction with respect to privacy, since inaccurate information can obstruct accountability or harm privacy (like in a case of a wrongly identified individual).

2. Privacy policy monitoring and validation.
   We need to better understand how to monitor and validate privacy policies. We need to develop technologies that ensure the correct enforcement of privacy policies. This research should include addressing the issues of monitoring and validating privacy aspects of data integration, separation, warehousing, and aggregation. An interesting issue, related to validation, is licensing of personal data by their owners for specific uses (an example is Ms. Smith agreeing to receive house-for-sale advertising by licensing her e-mail rights to a real estate advertiser).

3. Information hiding, obfuscation, anonymity, and accountability.
   Research should address different ways of assuring anonymity via information hiding and obfuscation, ranging from steganography through location security and hiding message source and destination from intermediate nodes to approaches used for digital elections. At the same time, for accountability, we need to investigate how to prevent illegitimate or improper information hiding. We need models supporting accountable anonymity that do not depend on a trusted third party. As an example, accountability suffers when data provenance obfuscation or user anonymity hinder intruder identification. Other interesting issues are information hiding and anonymity preservation in negotiations among parties with variable degrees of mutual trust.

4. New privacy-enabling and privacy-disabling technologies.
   We need more research on the impact of new technologies on preserving privacy. In particular research on privacy for pervasive computing is needed, since pervasive computing results in an easy information flow. Unless proper access control is provided, this flow threatens to ruin anonymity with perfect accountability (e.g., not only GPS-enabled devices but even cell phones and RFID tags on purchased products introduce the risk of monitoring of location of individuals). Similarly, permanent availability (or "always-on" connectivity) complicates protection against denial-of-service attacks. Interesting aspects of trust-related privacy are raised by data-sharing peers, including limiting data disclosures on the as-needed basis, and avoiding sharing irrelevant or highly sensitive data (such as trade secrets). Another important issue is privacy-preserving data mining on massive datasets.

5. Interdisciplinary privacy research.
   One important direction of interdisciplinary work is proposing comprehensive and rich privacy models based on social and ethical privacy paradigms. Another direction is considering public acceptance of privacy requirements and rules, and their enforcement.

In turn, for trust-related solutions, the following research problems should be addressed (*ibid*):

1. A better utilization of the social paradigm of trust.
   Utilization of the powerful social paradigm of trust, based on the analogies to uses of the notion of trust in social systems, should be explored in many ways. Finding out what makes trust work in existing social systems, and transferring this to a computing world is a big challenge. This work calls for a strong cooperation with social scientists.

2. Liability of trust.
   We need to provide methods, algorithms, and tools to identify which components and processes of the system depend on trust. We also need to find out to which extent and how security of a system may be compromised if any of these trust-dependent components fails. As an example, the role of data provenance explanations in trust-based systems needs be investigated.

3. Scalable and adaptable trust infrastructure.
   A high priority should be given to building scalable and adaptable trust infrastructures, including support for trust management and trust-based negotiations. In particular, support should be made available for gaining insight from different applications, for exploring the issue of dynamic trust, for building interoperable tools for the trust infrastructure, for developing flexible and extensible standards, and for investigating trust-based negotiations.

4. Benchmarks, testbeds, and development of trust-based applications.

We need benchmarks and testbeds for experimenting with diverse roles of trust in computing systems. The experiments should form a strong basis for the development of prototype trust-based applications, such as ones for crisis and emergency management for homeland security, broad collaborations among researchers or government agencies, or medical information sharing between healthcare providers. Trust-based solutions for new and emerging technologies should be studied. An example is using trust for ensuring data integrity and privacy in sensor networks deployed in trustless environments.

5. Interdisciplinary trust research.
   There is a strong need for trust-related interdisciplinary research outside of the realm of computer science and engineering. In addition to already-mentioned interdisciplinary work on the social paradigm of trust, it should include research on ethical, social, and legal issues, both human-centered and system-centered. Another important interdisciplinary work should focus on economic incentives for building trust, and disincentives and penalties for committing fraud.

Trust and privacy are strongly related to security. Therefore, in addition to the separate research directions for privacy and trust specified above, we can also indicate threads of research common not only to them, but also to security. This means research on intersecting aspects of trust, privacy *and* security (TPS) (Bhargava *et al*., 2003). The first common thread includes the tradeoffs, including not only the tradeoff between privacy and trust, but also performance vs. TPS, cost and functionality vs. TPS, and data monitoring and mining vs. TPS. The second common thread contains policies, regulations, and technologies for TPS. This includes creation of flexible TPS policies, appropriate TPS data management (including collection, usage, dissemination, and sharing of TPS data), and development of domain- and application-specific TPS approaches (such as TPS solutions for commercial, government, medical, and e-commerce fields). The third and the fourth threads are development of economic models for TPS, and investigation of legal and social TPS aspects.

ACKNOWLEDGMENTS

This research was supported in part by the NSF Grant IIS-0242840 and IIS-0209059. The authors thank Dr. Yuhui Zhong and Dr. Yi Lu, currently at Microsoft, for their contributions discussing related research on trust and privacy-for-trade solutions; and Dr. Mohamed Hefeeda, currently at the Simon Fraser University, for contributing Figure 7. Any opinions, finding, conclusions or recommendation expressed in the paper are those of the authors and do not necessarily reflect the views of the funding agencies or institutions with which the author is affiliated.

7. REFERENCES

(Aberer & Despotovic, 2001)=[1] Aberer, K. & Despotovic, Z. (2001). Managing Trust in a Peer-2-Peer Information System. In *Proc. of the 2001 ACM CIKM International Conference on Information and Knowledge Management, Atlanta, Georgia* (pp. 310–317). New York, NY: ACM.

(Agrawal and CAggarwal, 2001)=(Agrawal & Aggarwal, 2001)=[5] Agrawal, D., & Aggarwal, C. (2001). On the design and quantification of privacy preserving data mining algorithms. In *Proc. of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS'01* (pp. 247–255). Santa Barbara, CA: ACM.

(Barnes & Cerrito, 1998)=(Barnes *et al*., 1998)=[2] Barnes, G.R., Cerrito, P.B., & Levi I (1998). A mathematical model for interpersonal relationships in social networks. *Social Networks, 20*(2), 179-196.

(Bhargava *et al*., 2003) Bhargava, B., Farkas, C., Lilien, L., & Makedon, F. (2003). *Trust, Privacy, and Security: Summary of a Workshop Breakout Session at the National Science Foundation Information and Data Management (IDM) Workshop held in Seattle,Washington, Sep. 14–16, 2003* (Tech. Rep. No. 2003-34). West Lafayette, Indiana: Purdue University, Center for Education and Research in Information Assurance and Security (CERIAS).

(Bhargava *et al*., 2004)=[6] Bhargava, B., Lilien, L., Rosenthal, A., & Winslett, M. (2004). Pervasive Trust. *IEEE Intelligent Systems, 19*(5), 74-77.

(Bhargava and Y. Zhong, 2002)=[13] Bhargava, B., & Zhong, Y. (2002). Authorization Based on Evidence and Trust. In Kambayashi, Y., Winiwarter, W., & Arikawa, M. (Eds.) *Proc. of 4th Intl. Conf. on Data Warehousing and Knowledge Discovery, DaWaK 2002. Lecture Notes in Computer Science, Vol. 2454* (pp. 94-103). Heidelberg, Germany: Springer.

(Rezgui *et* al., 2003)=[50]=(Bouguettaya & Eltoweissy, 2003)=[50]=[ReBE03] Rezgui, A., Bouguettaya, A. R.A., & Eltoweissy, M.Y. (2003). Privacy on the Web: facts, challenges, and solutions. *IEEE Security and Privacy, 1*(6): 40–49.

(Carbo *et al*., 2003)=[5] Carbo, J., Molina, J., & Davila, J. (2003). Trust management through fuzzy reputation. *International Journal of Cooperative Information Systems, 12*(1), 135–155.

(Common Criteria 1999) =(Class FPR, 1999) Class FPR: Privacy (1999). In *Common Criteria for Information Technology Security Evaluation. Part 2: Security Functional Requirements. Version 2.1. Report CCIMB-99-032* (pp.109-118). Ft. Meade, Maryland: National Information Assurance Partnership (NIAP). Retrieved June 5, 2007, from http://www.niap-ccevs.org/cc-scheme/cc_docs/cc_v21_part2.pdf

(Collberg & Thomborson, 2000)=[19] =-(Collberg & Thomborson, 2002)=[19] Collberg, C., & Thomborson, C. (2002). Watermarking, Tamper-Proofing, and Obfuscation-Tools for Software Protection. *IEEE Transactions on Software Engineering, 28*(8), 735-746.

(Cofta, 2006) Cofta, P. (2006). Impact of convergence on trust in ecommerce. *BT Technology Journal, 24*(2), 214-218.

(Cover & Thomas, 1991)=[21] Cover, T., & Thomas, J. (1991). *Elements of Information Theory*. Hoboken, NJ: John Wiley & Sons.

(Cranor et al., 1999)=[9] Cranor, L.F., Reagle, J., & Ackerman, M. S. (1999). *Beyond Concern: Understanding Net Users' Attitudes about Online Privacy* (Tech. Rep. No. TR 99.4.3). Middletown, NJ : AT&T Labs-Research. Retrieved June 5, 2007, from citeseer.ist.psu.edu/cranor99beyond.html.

(Cranor , 2003)=[Cran03] Cranor, L.F. (2003). P3P: making privacy policies more useful. *IEEE Security and Privacy, 1*(6), 50–55.

(Diaz et al., 2002)-=[23] = (Diaz *et al*., 2003)-=[23] Diaz, C., Seys, S., Claessens, J., & Preneel, B. (2003). Towards Measuring Anonymity. In Dingledine, R., & Syverson, P.F. (Eds.), *Proc. of the 2nd International Workshop on Privacy Enhancing Technologies, PET 2002, San Francisco, California, April 2002. Lecture Notes in Computer Science, Vol. 2482* (pp. 184-188). Heidelberg, Germany: Springer.

(Donnellan, 1968)=[25a] Donnellan, T. (1968). *Lattice Theory*. Oxford, NY: Pergamon Press.

(Farrell & Housley, 2002)=[26] Farrell, S., & Housley, R. (2002). RFC3281: An internet attribute certificate profile for authorization. The Internet Society. Network Working Group. Retrieved June 5, 2007, from http://www.ietf.org/rfc/rfc3281.txt

(Fischer-Hübner, 2001) Fischer-Hübner, S. (2001). *IT-Security and Privacy-Design and Use of Privacy-Enhancing Security Mechanisms*. *Lecture Notes on Computer Science, Vol. 1958*. Heidelberg, Germany: Springer.

(Fujimura & Nishihara , 2003)=[28] Fujimura, K., & Nishihara, T. (2003). Reputation rating system based on past behavior of evaluators. In *Proc. of the 4th ACM Conf. on Electronic Commerce, San Diego, CA, USA* (pp. 246–247). New York, NY: ACM Press.

(Garfinkel, 2003)=[12] Garfinkel, T., Pfaff, B., Chow, J., Rosenblum, M., & Boneh, D. (2003). Terra: A Virtual Machine-Based Platform for Trusted Computing. In *Proc. of 19th ACM Symposium on Operating Systems Principles, SOSP 2003, Bolton Landing, New York* (pp. 193-206). New York, NY: ACM Press. Retrieved June 5, 2007, from http://citeseer.ist.psu.edu/667929.html

(Goldberg, 2000)=[31] Goldberg, I. (2000). *A Pseudonymous Communications Infrastructure for the Internet*. Ph.D. thesis, University of California at Berkeley. Retrieved June 5, 2007, from http://www.isaac.cs.berkeley.edu/iang/thesis-final.pdf

(Green, 2003) Green, A.M. (2003). *International Privacy Laws. Sensitive Information in a Wired World* (Tech. Rep. No. CS 457). New Haven, Connecticut: Yale University.

(IBM Privacy, 2007) IBM Privacy Research Institute (2007). Armonk, New York: IBM. Retrieved June 5, 2007, from http://www.research.ibm.com/privacy/

(Internet Society, 2007)--[13] = (Internet Security, 2007)--[13] Internet Security Glossary (2007). The Internet Society. Retrieved June 5, 2007, from www.faqs.org/rfcs/rfc2828.html

(Jensen, 1996)=[36] Jensen, F.V. (1996). *An introduction to Bayesian networks*. London, United Kingdom: UCL Press.

(Kelley 2001) Kelley, C.M., with Denton, A., & Broadbent, R. (2001). *Privacy Concerns Cost eCommerce $15 Billion*. Cambridge, MA: Forrester Research.

(Lilien & Bhargava, 2006)=[p2d2] Lilien, L., & Bhargava, B. (2006). A Scheme for Privacy-preserving Data Dissemination. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, *36*(3), 503-506.

(Lilien *et al.*, 2006) Lilien, L., Kamal, Z.H., Bhuse, V., & Gupta, A. (2006). Opportunistic Networks: The Concept and Research Challenges in Privacy and Security. In Reiher, P., Makki, K., & Makki, S. (Eds.), *Proc. of the International Workshop on Research Challenges in Security and Privacy for Mobile and Wireless Networks, WSPWN06, Miami, Florida* (pp. 134-147).

(Lilien *et al.*, 2007) Lilien, L., Gupta, A., & Yang, Z. (2007). Opportunistic Networks for Emergency Applications and Their Standard Implementation Framework. In *Proc. of the First International Workshop on Next Generation Networks for First Responders and Critical Infrastructure, NetCri07, New Orleans, LA* (pp. 588-593).

(Marchiori *et al.*, 2002)=[40] =(Marchiori, 2002)=[40] Marchiori, M. . (2002). *The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. W3C Recommendation*. W3C. Retrieved June 5, 2007, from http://www.w3.org/TR/P3P/

(Marsh, 1994)=[11] Marsh, S. (1994). *Formalizing Trust As a Computational Concept*. PhD thesis, University of Stirling, U.K.

(McKnight *et al.*, 2002) =[12c] McKnight, D., Choudhury, V., & Kacmar, C. (2002). Developing and validating trust measures for e-commerce: An integrative topology. *Information Systems Research, 13*(3), 334–359.

(McKnight & Chervany, 2001) =[13c] McKnight, D.H., & Chervany, N.L. (2001). Conceptualizing Trust: A Typology and E-Commerce Customer Relationships Model. In *Proceeding of the 34th Annual Hawaii International Conference on System Sciences, HICSS-34, Vol. 7, Island of Maui, Hawaii* (10 pages). Washington, D.C.: IEEE Computer Society. Retrieved June 5, 2007, from http://csdl2.computer.org/comp/proceedings/hicss/2001/0981/07/09817022.pdf

(Mercuri, 2004) Mercuri, R.T. (2004). The HIPAA-potamus in Health Care Data Security. *Communications of the ACM, 47*(7): 25-28.

(Morinaga *et al.*, 2002)=[15] Morinaga, S., Yamanishi, K., Tateishi, K., & T. Fukushima, T (2002). Mining Product Reputations on the Web. In *Proceeding of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 341–349). New York, NY: ACM Press. Retrieved June 5, 2007, from citeseer.ist.psu.edu/morinaga02mining.html

(Mui, 2002)=[16] Mui, L. (2002) . *Computational Models of Trust and Reputation: Agents, Evolutionary Games, and Social Networks*. PhD thesis, Massachusetts Institute of Technology.

(Mui *et al*., 2002)=[17] Mui, L., Mohtashemi, M., & Halberstadt, A. (2002). A Computational Model of Trust and Reputation for E-businesses. In *Proceeding of the 35th Annual Hawaii International Conference on System Sciences, HICSS'02, Track 7, Island of Hawaii, Hawaii* (9 pages). Washington, D.C.: IEEE Computer Society. Retrieved June 5, 2007, from http://csdl2.computer.org/comp/proceedings/hicss/2002/1435/07/14350188.pdf

(Pfitzmann & Kohntopp, 2000)=[42] Pfitzmann, A., & Köhntopp, M. (2000). Anonymity, Unobservability, and Pseudonymity—A Proposal for Terminology. In Federrath, H. (Ed.), *Designing Privacy Enhancing Technologies, Proc. of the Workshop on Design Issues in Anonymity and Observability, Berkeley, California. Lecture Notes in Computer Science, Vol. 2009* (pp. 1-9). Heidelberg, Germany: Springer.

(Pujol *et al*., 2002)=[18] Pujol, J.M., Sangesa, R., & Delgado, J. (2002). Extracting reputation in multi agent systems by means of social network topology. In *Proceeding of the First International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS '02, Bologna, Italy* (pp. 467–474). New York, NY: ACM Press. Retrieved June 5, 2007, from citeseer.ist.psu.edu/pujol02extracting.html

(Privacy Act, 2004)=[44] Privacy Act (2004). Washington, D.C.: U.S. Environmental Protection Agency. Retrieved June 5, 2007, from http://www.epa.gov/privacy/

(Privacy Bird, 2004)=[APBT04] =(Privacy Bird, 2007)=[APBT04] Privacy Bird Tour (2007). Retrieved June 5, 2007, from http://www.privacybird.org/tour/1_3_beta/tour.html

(Trade Commission, 1998)=[UFTC98] =(Privacy Online, 1998)=[UFTC98] Privacy Online: A Report to Congress (1998). Washington, D.C.: U.S. Federal Trade Commission.

(Reiter and Rubin, 1999)=[48] Reiter, M., & Rubin, A. (1999). Crowds: Anonymity for Web transactions. *Communications of the ACM*, *42*(2), 32–48.

(Ross, 2007) Ross, R. (2007). Robotic Insect Takes Off. Researchers have created a robotic fly for covert surveillance. *Technology Review*, July 19, 2007. Retrieved July 20, 2007, from http://www.technologyreview.com/Infotech/19068/

(Sabater & Sierra, 2002)=[20] Sabater, J., & Sierra, C. (2002). Social ReGreT, a reputation model based on social relations. *ACM SIGecom Exchanges*, *3*(1), 44–56.

(Seigneur & Jensen, 2004)=[55] Seigneur, J.-M., & Jensen, C.D. (2004). Trading privacy for trust. In Dimitrakos, T. (Ed.), *Proceedings of the Second International Conference on Trust Management, iTrust 2004, Oxford, United Kingdom. Lecture Notes in Computer Science, Vol. 2995* (pp. 93-107). Heidelberg, Germany: Springer.

(Sensor Nation, 2004) Sensor Nation. Special Report (2004). *IEEE Spectrum*, *41*(7).

(Serjantove and G. Danezis, 2002)=[56] =(Serjantov & Danezis, 2003)=[56] Serjantov, A., & Danezis, G. (2003). Towards an Information Theoretic Metric for Anonymity. In Dingledine, R., & Syverson, P.F. (Eds.), *Proc. of the 2nd International Workshop on Privacy Enhancing Technologies, PET 2002, San Francisco, California, April 2002. Lecture Notes in Computer Science, Vol. 2482* (pp. 259-263). Heidelberg, Germany: Springer.

(HIPAA Summary, 2003) =(Summary HIPAA, 2003) Summary of the HIPAA Privacy Rule (2003). Washington, D.C.: The U.S. Department of Health and Human Services.

(Sweeney, 1996)=[59] Sweeney, L. (1996). Replacing Personally-Identifying Information in Medical Records, the Scrub System. In Cimino, J.J. (Ed.), *Proceedings, American Medical Informatics Association* (pp. 333-337). Washington, D.C.: Hanley & Belfus. Retrieved June 5, 2007, from http://privacy.cs.cmu.edu/people/sweeney/scrubAMIA1.pdf

(Sweeney, 1998)=[60] Sweeney, L. (1998). Datafly: A System for Providing Anonymity in Medical Data. In Lin, T.Y., & Qian, S. (Eds.), *Database Security XI: Status and Prospects. IFIP TC11 WG11.3 Eleventh International Conference on Database Security, Lake Tahoe, California, August 1997* (pp. 356-381). Amsterdam, The Netherlands: Elsevier Science.

(Sweeney, 2001a)=[61] Sweeney, L. (2001a). *Computational Disclosure Control: A Primer on Data Privacy Protection*. PhD thesis, Massachusetts Institute of Technology.

(Sweeney, 2001b)=[62] Sweeney, L. (2001b). Information explosion. In Zayatz, L., Doyle, P., Theeuwes, J., & Lane, J. (Eds.), *Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies* (26 pages). Washington, D.C.: Urban Institute. Retrieved June 5, 2007, from http://privacy.cs.cmu.edu/people/sweeney/explosion2.pdf

(Sweeney, 2002a)=[63] Sweeney, L. (2002a). Achieving k-anonymity privacy protection using generalization and suppression. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 10*(5), 571–588.

(Sweeney, 2002b)=[64] Sweeney, L. (2002b). k-anonymity: a model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 10*(5), 557–570.

(American Heritage, 2000)--[1] =(The American, 2000)--[1] *The American Heritage Dictionary of the English Language. 4th ed.* (2000). Boston, MA: Houghton Mifflin.

(Trustworthy Computing, 2003) Trustworthy Computing White Paper (2003). Redmond, Washington: Microsoft. Retrieved June 5, 2007, from http://www.microsoft.com/mscorp/twc/twc_whitepaper.mspx

(Tschudin, 1999)=[Tsch99]=[21]=[67] Tschudin, C. (1999). Apoptosis — the Programmed Death of Distributed Services. In Vitek, J., & Jensen, C.D. (Eds.), *Secure Internet Programming. Security Issues for Mobile and Distributed Objects. Lecture Notes in Computer Science, Vol 1603* (pp. 253-260). Heidelberg, Germany: Springer.

(Tygar & Yee, 1994)=[TyYe94]=[22]=68] Tygar, J.D., & Yee, B. (1994). Dyad: A System for Using Physically Secure Coprocessors. In *Proc. Joint Harvard-MIT Workshop Technological Strategies for Protecting Intellectual Property in the Networked Multimedia Environment*. Annapolis, Maryland: Interactive Multimedia Association. Retrieved July 20, 2007, from http://www.cni.org/docs/ima.ip-workshop/Tygar.Yee.html

(Wagealla *et al*., 2003)=[71] Wagealla, W., Carbone, M., English, C., Terzis, S., Lowe, H. & Nixon, P. (2003). A Formal Model for Trust Lifecycle Management. In *Proc. of the 1st International Workshop on Formal Aspects of Security and Trust, FAST 2003, Pisa, Italy, September 2003* (pp. 181-192). Retrieved July 20, 2007, from http://www.iit.cnr.it/FAST2003/fast-proc-final.pdf

(Safe Harbor, 2007) =(Welcome Safe, 2007**)** Welcome to the Safe Harbor (2007). Trade Information Center. Retrieved June 5, 2007, from http://www.export.gov/safeharbor/

(Young, 1971)=[72] Young, J.F. (1971). *Information theory*. New York, NY: Wiley Interscience.

(Yu & Singh, 2002a)=[22] Yu, B., & Singh, M.P. (2002a). Distributed reputation management for electronic commerce. *Computational Intelligence, 18*(4), 535–549.

(Yu & Singh, 2002b)=[23] Yu, B., & Singh, M.P. (2002b). An evidential model of distributed reputation management. In *Proceeding of the First International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS '02, Bologna, Italy, July 2002* (pp. 294–301). New York, NY: ACM Press.

(Yu *et al*., 2003)=[73] Yu, T., Winslett, M., & Seamons, K.E. (2003). Supporting structured credentials and sensitive policies through interoperable strategies for automated trust negotiation. *ACM Transactions on Information and System Security, 6*(1), 1–42.

(Zacharia & Maes, 2000)=[24] Zacharia, G., & Maes, P. (2000). Trust management through reputation mechanisms. *Applied Artificial Intelligence, 14*, 881–907.

(Zhong & Bhargava, 2004)=[25b] Zhong, Y., & Bhargava, B. (2004). Using Entropy to Trade Privacy for Trust. In *Proceedings of the NSF/NSA/AFRL Conference on Secure Knowledge Management, SKM 2004, Amherst, NY, September 2004* (pp. **XXX**). **CITY, ST: PUBL.**

(Zhong *et al*., 2004)=[75] Zhong, Y., Lu, Y., and Bhargava, B. (2004). *TERA: An authorization framework based on uncertain evidence and dynamic trust* (Tech. Rep. No. CSD-TR 04-009). West Lafayette, Indiana: Purdue University.

(Zhong *et al*., 2006) Zhong, Y., Lu, Y., Bhargava, B., & Lilien, L. (2006). *A Computational Dynamic Trust Model for User Authorization* (Working Paper). West Lafayette, Indiana: Purdue University.

8. ADDITIONAL READING IN TRADING PRIVACY FOR TRUST IN ONLINE INTERACTIONS

(8) Additional Reading
A list of 25-50 additional readings (e.g. journal articles, book chapters, case studies, etc.) should be offered by the author(s) of each chapter. As the experts, we feel as though the contributing authors are the best source for suggestions on additional readings in their respective fields!

********** NOT FINISHED - NEED MORE (10-30) REFERENCES – ESP. NEW ONES, >= 2005 **********

Cahill, V., Gray, E., Seigneur, J.-M., Jensen, C.D., Chen, Y., English, C., Wagealla, W., Terzis, S., Nixon, P., di Marzo Serugendo, G., Bryce, C., Carbone, M., Krukow, K., & Nielsen, M. (2003). Using trust for secure collaboration in uncertain environments. *IEEE Pervasive Computing*, *2*(3), 52–61.

Frosch-Wilke, D. (2001). Are E-Privacy and E-Commerce a Contradiction in Terms? - An Economic Examination. In *Proceedings of Informing Science Challenges to Informing Clients: A Transdisciplinary Approach, Krakow, Poland, June 2001* (pp. 191-197). Retrieved June 5, 2007, from http://www.informingscience.org/proceedings/IS2001Proceedings/pdf/FroschWilkeEBKAreEP.pdf

Grandison, T., and M. Sloman, M. (2000). A survey of trust in internet applications. *IEEE Communications Surveys*, *3*(4), 2–16.

Langheinrich, M. (2001). Privacy by Design - Principles for Privacy-Aware Ubiquitous Systems. In Abowd, G.D., Brumitt, B., & Shafer, S. (Eds.) *Proceedings of the 3rd International Conference on Ubiquitous Computing, UbiComp 2001, Atlanta, Georgia, September-October 2001. Lecture Notes in Computer Science, Vol. 2201* (pp. 273-291). Heidelberg, Germany: Springer.

Martin, D. M., Jr., Smith, R. M., Brittain, M., Fetch, I., & Wu, H. (2001). The privacy practices of Web browser extensions. *Communications of the ACM*, *44*(2), 45–50.

Resnick, P., Kuwabara, K., Zeckhauser, R. & Friedman, E. (2000). Reputation systems. *Communications of the ACM*, *43*(12), 45–48.

Richardson, M., Agrawal, R., and Domingos, P. (2003). Trust Management for the Semantic Web. In *Proceedings of the 2nd International Semantic Web Conference. Lecture Notes in Computer Science, Vol.* 2870 (pp. 351–368). Heidelberg, Germany: Springer.

Rousseau, D.M., Sitkin, S.B., Burt, R.S., & Camerer C. (1998). Not so different after all: A cross-discipline view of trust. *Academic Management Review*, *23*(3), 393–404.

Sandberg, C.K. (2002). Privacy and customer relationship management: can they peacefully co-exist. *William Mitchell Law Review*, *28*(3), 1147–1162.

Westin, A. (1967). *Privacy and Freedom*. New York, NY: Atheneum, 487 pages.

Westin, A. (2003). Social and Political Dimensions of Privacy. *Journal of Social Issues*, *59*(2), 431–453.