Trust Management Survey

Sini Ruohomaa and Lea Kutvonen

University of Helsinki, Finland
{Sini.Ruohomaa, Lea.Kutvonen}@cs.helsinki.fi
http://www.cs.helsinki.fi/group/tube/

Abstract. Trust is an important tool in human life, as it enables people to cope with the uncertainty caused by the free will of others. Uncertainty and uncontrollability are also issues in computer-assisted collaboration and electronic commerce in particular. A computational model of trust and its implementation can alleviate this problem.

This survey is directed to an audience wishing to familiarize themselves with the field, for example to locate a research target or implement a trust management system. It concentrates on providing a general overview of the state of the art, combined with examples of things to take into consideration both when modelling trust in general and building a solution for a certain phase in trust management, be it trust relationship initialization, updating trust based on experience or determining what trust should have an effect on.

1 Introduction

Trust plays an important role in virtual organisations, countering uncertainty caused by the business requirement for openness. The requirement seeks to make marketable services openly available to all potential, highly autonomous clients, which increases a service provider's vulnerability to an attack. As there is no central authority to provide support for traditional authentication for a rapidly changing actor base, making sensible authorisation decisions concerning new, previously unknown partners is difficult. Manual updates to policy or access control settings quickly become laborious, which drives organisations into making only very broad decisions concerning large parts of the user base to avoid the overly heavy process of personalizing the security settings. Trust management can provide a basis for more detailed and better-informed authorisation decisions, while allowing for a high level of automation.

This paper aims to provide an overview of trust management research in the field of computer science, without going too deeply into any implementation specifics. It was written as a part of a state-of-the-art analysis for the TuBE (Trust Based on Evidence) project [1]. The paper is organized in two parts. The first part discusses trust as a concept, how it has been modelled and how the concept could be introduced to computer security. Noteworthy ideas are drawn from all sources alike.

P. Herrmann et al. (Eds.): iTrust 2005, LNCS 3477, pp. 77–92, 2005.

The second part describes the different tasks of the trust management system: determining initial trust, observing the trustee's actual behaviour and updating trust accordingly. Technologies and ideas to support a particular task are brought up together with the general discussion of the challenges related to that task. This forms a loose application taxonomy through which trust management research is presented. The task set, or trust life cycle, is taken to begin from choosing a partner and determining a suitable initial trust in them. Reputation systems are discussed as an aid in this task, although their usability does not end there. After the partner is allowed to use the provided services, the system moves to observation and gathering evidence of the partners' behaviour. Intrusion detection and prevention systems (IDS/IPS) play a central role here. From the data gathered from various pieces of evidence and possible updates from a reputation system, different actions can be taken, the most central being the adjustment of the trust estimate.

The rest of this paper is organized as follows: Chapter 2 discusses concepts for trust management and how to model it, covering the aforementioned first part of the paper. Chapter 3 covers the second part, discussing the initialization of trust relationships, identifying methods for observing the trustee and considering the various decisions to base on trust and the observations. Finally, chapter 4 offers some conclusions.

2 On the Nature of Trust

Trust and reputation, a closely related term, are firmly rooted in sociology, and those roots should not be forgotten [2]. However, trust is quite a complicated phenomenon, the concept itself carrying many meanings. As our interest in purely sociological or psychological studies on trust has been limited, this section will only give a brief overview of the trust phenomenon before moving on to how systems should trust. It is not certain that we want to even try imitating human trust fully in computer systems, as humans do not seem to always make fully rational trust decisions [3, 4].

2.1 Concepts for Trust Management

In the following chapters of this work, we consider trust as it is directed at independent actors. The **trustor** is a service provider practicing electronic commerce on the Internet, and the **trustee** is either a business partner or an individual requiring access to the trustor's services, as represented by an identifiable agent in the network. The trustees are independent in the sense that their actions cannot directly be controlled by outsiders such as the service provider, i.e. the trustor. The business partners and individuals behind the trustee agents can have control over several different identities through agents which cannot be reliably traced back to them or to each other.

Trust seems to essentially be a means for people to deal with uncertainty about the future and their interaction partners. Stephen Marsh considers the protection of law, a lack of options for possible outcomes and other kinds of

limitations, reducing the aforementioned independence of actors, as examples of factors reducing the need to trust [5]. In a more technical environment, "trusted" hardware for monitoring [6] or cryptographically secure communications [7] also work towards reducing uncertainty.

Trust is defined as *the extent to which one party is willing to participate in a given action with a given partner, considering the risks and incentives involved* (adjusted from [8, 9]). **A trust decision** is binary and based on the balance between trust and risk, and it has some sort of effect on the trustee. Usually it is made with a class of applicable situations in mind, such as concerning a particular trustee in performing a certain action only. **Actions** involve using services provided by the trustor. The effect of the trust decision on the trustee varies: depending on the situation, trust may directly affect whether the action is *authorised* at all. Even if access is granted, there may be a need to *limit resources* available during the action or tighten the *observation* of the trustee.

The effect of trust comes with a **risk**: an authorisation decision means that we expose something in our control to attack or abuse, while reduced observation means that misbehaviour may proceed undetected and more resources allocated means that more of them may go to waste or be misused. The connection of risk and trust is emphasised by many researchers, such as [5, 8, 10]. Risks are tied to assets [11]. If money or system security are assets, the related risks involve losing money or experiencing a breach in security. The risks considered here are limited to those known to the trustor; balancing against something unknown may prove difficult. The **action importance**, or its business value, affects trust similarly to good reputation, increasing the willingness to make a positive trust decision.

Reputation is defined as a *perception a party creates through past actions about its intentions and norms* [2]. Reputation exists only in a community which is observing its members in one way or another, and is as such meaningless outside its native environment. It can be transmitted from one community context to another, however, by means of recommendations. As the definition implies, it is affected by experience; directly if the experience is shared by the entire community, or through negotiations if only a sub-community has borne witness.

A **recommendation** is simply *an attempt at communicating a party's reputation from one community context to another*. A poor recommendation may be detrimental to one's reputation, and there is no separate term for "negative recommendation". The word *attempt* should remind us that the source and target communities are seldom compatible enough to be able to use a recommendation directly. Instead, the recommendation may be tuned down in various means, including tagging it as "uncertain information" or giving it a lowered weight in appropriate calculations. In order for reputation to exist in larger than the trivial one-member communities, the members must come to an agreement about their shared perception for each given party in that community. Various reputation systems suggest different ways of coming to this agreement. There is no objective truth to be found—or lost—in reputation itself, but some perceptions come closer to the target party's real intentions and norms than others.

Despite the earlier dismissal of human trust as somewhat irrational and overly complex, it is an important research topic for computer scientists as well. After all, there is a human actor somewhere behind the user agents and client programs, making trust-involving decisions e.g. on whether to use our trust management system or not. If the human users disagree with their trust management system more often than not, they will probably change the system accordingly. Research on how to appear trustworthy to users shows that trustworthiness estimates are determined by much more than just past success or failure to comply to expectations [11, 12, 13, 14]. It has been suggested that contracts, as they make implicit expectations explicit, can encourage more trust [13, 14]. An agreement on social norms does help in building human trust, but binding contracts can also be used to reduce uncertainty in electronic commerce. In case of a breach of contract, a means for dealing with the violation may be given by the contract itself, with the backing of law, if necessary and available. Contracts are in this sense another measure of control, like insurances, and they may well reduce risk even more than encourage more trust to balance with the remaining risk. The effect of contracts in trust management has been discussed in relation to transaction modelling as well [15, 16].

2.2 The Trust Management Model

Trust management research has its roots in authentication and authorisation. In the context of authentication, trust is established by means such as digital certificates. These certificates are proof of either identity directly or membership in a group of good reputation. Authentication-related trust is discussed for example in [17, 7]. Policy languages, such as [18, 19, 20], then make it possible to automatically determine whether certain credentials are sufficient for performing a certain action, to authorise the trustee. The Sultan trust management framework [21] includes a language for describing trust and recommendation relationships in the system. Constraints can be attached to these relationships, and through them the relationships can be connected to the Ponder policy language [22].

Credentials are sufficient when the system is either convinced of the trustee's identity or knows her to be a member of some sufficiently trusted group—or one of the credentials may be an authorisation certificate, signed by someone with the right to delegate such authority. At the authentication level, trust is monotonic across time and attached to a certain identity or membership. Updating the level of trust based on evidence of actual behaviour is not yet considered; the focus is on credentials matching policy.

Sufficiently flexible policy systems provide the backbone for a trust management system. Tonti et al. compare three languages for policy representation and reasoning [23]. KAoS [24, 25], Rei [26] and the aforementioned Ponder are used as a basis for sketching some general properties desirable in future work on policy semantics. Delegent has strong roots in authorisation administration research [27], and it has also been developed into software [28].

To make trust more dynamic, the behaviour of the trustee should be considered as well. In 2000, monitoring users could be achieved by intrusion

detection systems, but the information gleaned was not being used to evolve trust or reputation. None of the existing systems then yet covered monitoring and re-evaluation of trust [29]. Since then, behaviour history collection has been included in one form or another in numerous trust models. Behaviour information can be gathered locally [30, 9], or it can be received as third-party observations through a reputation system [31]. Involving third parties, however, requires some sort of trust in their statements, as well as comparability between the trustor's and the third party's views on reputation. Reputation systems are discussed in more detail in chapter 3.1.

Much work has also gone to identifying factors which either are considered to affect trust directly or which are used together with trust decisions. It was mentioned earlier that uncertainty is involved in increasing the need for trust. Uncertainty is not always problematic to the trustor, however, but mostly when it is related to risk. While the exact relationship between risk and trust is not entirely clear [8, 10], it is agreed that increased risk and increased need to trust go hand in hand [32, 33]. Risk is relative to the trustor taking the risk; for example, the risk of losing a specific monetary sum is less important if the sum is only a small fraction of the usable funds. It has also been noted that increased potential profits in making a decision to trust encourages coping with relatively higher risk [9]. Potential profits can be considered a part of the action importance mentioned in chapter 2.1. The protection of law and other factors limiting the need to trust according to Marsh may also be considered means to reduce risk [5].

Applications where a more dynamic trust management is beneficial may have a rapidly-changing user base. Newcomers create a problem for a trust management system based on behaviour history alone. The system must determine how much these unknown individuals should be trusted, sometimes without knowing anything about them. While certification may provide a means to introduce an initial trust out of band, it may not be plausible for some applications. Similarly, reputation systems are only helpful if the user has interacted with other systems gathering reputation before. For fully unknown users, a default level of trust must be determined. If it is set too low, the user may not be allowed access to the system at all, which makes proving trustworthiness through one's actions rather difficult [34]. If it is set very high, there may be a need to limit the possibility for users to "start over" by re-registration after misbehaving. Otherwise the punishment from having behaved badly becomes void, as a user with a trustworthiness estimate below that given to a newcomer will re-register herself to the system to become one herself.

2.3 The Trust Information Model

The problem of somehow representing human thought and feeling in a computer system is quite evident in trust management, albeit still in a somewhat limited sense compared to some other fields. Sociologists and psychologists, as well as economists in the field of game theory, have attempted to model trust and concepts closely related to it, such as reputation and reciprocity. Reciprocity is the *mutual exchange of deeds (such as favor or revenge)* [2]. That is, if one

participant in a highly reciprocal society tends to be very cooperative, others should be cooperative towards him as well. The term has not been included in many trust models this far.

Current trust models have been criticised for not making the relationship between trust and reputation clear and for treating them as independent of context or time [2]. Grandison and Sloman [21] find that while the current (in 2002) logic-based frameworks suffer from problems related to applicability and limit themselves to a subsection of the trust management problem, the existing solutions such as PolicyMaker [19], KeyNote [20], REFEREE [18] and TrustBuilder, a negotiation architecture for sensitive credential exchange [17], merely concentrate on certificates and access control, with no trust re-evaluation based on available information.

Early forms of trust management, as represented by the aforementioned four systems, began by automating authentication and authorisation decisions with the help of varying sets of credentials. In this kind of setting, a trust level is fixed in relationship to passed credentials, and trust is not re-evaluated based on experience information. It is outsourced, in a sense, to certificate authorities and the like, and the system using this kind of trust management is merely deciding how much it "trusts" a given credential or its issuer in the context of determining a reputation of sorts.

Research on trust can be divided into three groups based on its context. On the lowest, most fundamental level, trust is a part of the *infrastructure*. Early trust research has been concentrating on this level. As electronic commerce has gained a foothold and open systems become more common, trust forms an important part on the *service* level as well, where much of this paper is positioned. There are still many problems to be solved on this level before research on the highest level, the *community*, can proceed freely.

Marsh was one of the first to introduce a computational model for trust in his doctorate thesis [5]. His model is relatively simple, based on a scalar value of trust, and does not discuss reputation. Abdul-Rahman and Hailes criticize the model for overvariabilisation [35]. Mayer looked for a differentiation between factors contributing to trust, trust itself and its outcomes [8]. Two years later Essin wrote a socio-technologically focused model for trust and policy, with a goal to make them work better in computer systems [36].

Various different aspects of trust are highlighted in the different ways it is defined. Gambetta sees trust as a subjective *probability* in the trustee performing a particular action ([37], used by e.g. Abdul-Rahman and Hailes [35]). Not far apart, Mui et al. consider it as a subjective *expectation* about the trustee's future behaviour [2]. On the other hand, Mayer, Jøsang and Lo Presti define trust as an extent of *willingness* to depend on somebody [8, 9]. Demolombe places his definitions of trust in a framework of modal logic, and considers it to be a strong *belief about a given property* of the trustee, for example sincerity, cooperativity or credibility [38].

Jøsang draws our attention to being clear about the target of trust [39]. He points out that a machine or a program (a rational entity) does not trust; it

only implements the trust policy given by a human (a passionate entity). On the other hand, while trusting another passionate entity concerns speculation on things such as their motives and intentions, trusting a rational entity, who only acts according to lists of commands and rules, is based on trusting its ability to withstand manipulation by a passionate entity [39]. This also implies that when placing trust in an agent, which is a rational representation of a passionate entity, i.e. a human user, we are not only placing trust in the user behind the agent, but also indirectly in the person who coded the agent and anyone who would be capable of assuming control of the agent. While all this can be summarised as trusting the user to only use a "secure" agent, being explicit about the implications of that trust will aid not only fellow scientists but, possibly even more importantly, the users of the trust models, frameworks and implemented solutions produced from trust management research.

Egger [40, 12] has developed a model for trust-relevant factors from a customer's perspective. Some factors are relevant for the perspective of a service provider as well, such as reputation, propensity to trust and transference. As mentioned before, risk is an important factor in trust management. Mayer points out, in light of his definition of trust as a willingness to depend on someone, that risk is not directly tied to the willingness itself, but on the behavioural manifestation of a willingness to be vulnerable [8].

While Marsh's trust model represented trust in the form of a scalar [5], SECURE represents it as a range to include a measure of uncertainty in the value [41]. Jøsang and Lo Presti include a three-dimensional decision surface for balancing trust and risk in their model [9]. Trust could also be represented as an n-dimensional vector, with parameters such as the trustee's reputation, the action to perform and the risk and business importance related to it. As described before, a trust decision related to a particular action in a given situation remains binary, with the system possibly providing also a third option for "yes, if the following constraints are met".

3 The Tasks of a Trust Management System

In the previous section, we built some theoretical basis for trust management, categorized the effects of trust to the trustee into those related to authorisation, observation and resource allocation, and identified different factors weighed in a trust decision. In this section, we take a look at different challenges set for a trust management system. The section begins with the initialization of trust relationships, and goes on to identify different means to observe the trustee's behaviour during the actions. Finally, actions to take based on the new experience are discussed.

3.1 Initializing a Trust Relationship

Sometimes partners can be found with traditional out-of-band means like word of mouth, but in a highly dynamic and possibly automated environment a discovery service of some sort is necessary [42]. The lack of background information

constitutes a problem both for determining an initial trust in a partner as well as choosing the suitable partner for one's current needs. A search using a discovery service may result in a plethora of potential partners, some of which may be incompetent or even malicious. Once a number of potential partners has been found, a reputation system may aid in locating the most trustworthy one, based on their past behaviour with other principals in the network.

A reputation system aggregates information about the past behaviour of a group of entities in the form of the community's shared perception of them. This information may include information from book reviewers' perceived fairness to on-line companies' perceived competence and reliability. Reputation systems have been found to benefit of computer-aided human-to-human interaction, by reducing the level of uncertainty about new acquaintances to an endurable level [31, 33].

Experience or reputation information gathering and storage can be organized centrally or be distributed across peers. EBay (`http://www.ebay.com`), an on-line marketplace, gathers performance ratings from its users distributedly, but the results are kept on a central server [33]. Organizations like The International Chamber of Commerce, who act as advisors and certifiers in first-trade situations and provide information about other organizations' reputation, represent a fully centralized approach [43]. Everyone wishing to use a reputation system must be able to trust the information provider to not insert false ratings or omit information at will. While this may be a small challenge in a centralized approach, it is considerably more difficult to achieve with a fully distributed approach. The problem escalates when reputation is more valuable; competitors may be given bad ratings to disrupt their business, or good ratings may be sold for money, unrelated to actual performance. Gathering negative feedback may also be a problem: human reviewers in eBay tend to avoid giving negative feedback to others and prefer to resolve conflicts out-of-band *before* making a rating [31].

Obreiter suggests the use of evidence in the form of trade receipts, which can be used as a sort of certificate of having behaved well at some point [44]. Pinocchio rewards honest participation in a recommendation system, which might rate users as well as e.g. books, with a sort of virtual currency which is then needed to use the system. It approximates an honest user (measured as a scalar) as one who does not disagree with other users more than other users on the average disagree with each other. On the other hand, it also punishes poor "reputation" as a recommender by stopping the rewards for a probationary period [45].

Dishonesty in the expression of perceptions should be somewhat difficult to detect, let alone prove, but in the context of reputation systems we can understand dishonesty as relatively similar to Pinocchio's view—either too agreeing or too disagreeing to be likely to be useful for others. Kalcklösch and Herrmann apply statistical methods in ad hoc networks where trust information communication is automatic [46]. While these methods may not be the approach of choice for an established web service provider looking for partners, they serve well in their context. One must note that solutions on different levels, from infrastructure to service to community level, have very different needs.

Even if the recommending party is known to be honest and knowledgeable, their statements may be useless if the principals are not known by the same name by the recommender and the receiver, or if the principles behind the recommendations are not comparable to those of the receiver. A good reputation as a trader in an on-line auctioning system does not necessarily mean that the user should be given e.g. wider access privileges in a distributed software development project. This makes representing trust or reputation as a single numeric value somewhat problematic: If a reputation statement says that a user is trustworthy by "3 on a scale from 1 to 5", what does it mean in the receiver's context? Has the default been 1 or 3 and how easily is it increased or decreased? If this is a trader's reputation, should I trust them to sell a car to me if they got their reputation by selling pocket lighters [31]? This causes difficulties for porting ratings from one system to another as well. Recommendations remain an *attempt* at communicating reputation information between communities.

Resnick et al. describe three requirements for a successful reputation system: first, the entities must be long-lived and have use for reputation; second, feedback must be captured, distributed and made available in the future; third, the feedback must be used to guide trust decisions [31]. The first property implies some problems that newcomers have with reputation systems. Besides having the problem of finding a trustworthy information provider, they must gain a reputation themselves [34].

The usability of reputation information from outside sources is not limited to choosing a partner. It can also be included as a factor in the trust estimate of a partner, along with their locally gathered reputation based on first-hand experience. Initially, as there is no local information about the partner's behaviour, external reputation information may hold considerable weight in a trust decision.

Besides reputation systems, various kinds of authentication and credential systems may help determine an initial level of trust through e.g. membership in a group with a good reputation. The Web Services standard WS-Trust approaches authorisation and authentication via security tokens requested from on-line servers [47]. Karabulut proposes a hybrid public key infrastructure model to ease the delegation of trust, in the sense of allowing third parties to produce credentials usable for authorisation trust management in the target system [48].

A trust management system also tends to have some sort of a default value to assign to complete strangers. This value represents the system's general tendency to trust, or its *trust propensity*. This default may be raised or lowered based on a general view of the world the system operates in. If the average partner seems to be a somewhat unreliable opportunist, the trust propensity may be reduced. On the other hand, if the system operates in an environment of honest cooperation, the trust propensity may be increased.

As the initial trust value is even at best based on the experiences of others with the partner, it may prove to be a poor estimate. Observing the partner's actions and updating their local reputation based on the observation strengthens the system against misplaced expectations. Evolving reputation and trust is discussed in more detail in chapter 3.3.

3.2 Observation

Observation can be done in two different roles: either as an active participant of a collaboration, or as an outsider, a silent third party. In the first case, the actions of the observed are seen through a personal context, which gives more depth to the analysis. Intrusion detection software can benefit greatly from "insider" information from the observed application, if it is available. As an example, Zamboni [49] suggests using internal sensors inside the observed applications themselves, but such modifications are not always possible.

The principles and research in the field of intrusion detection can be put to use in observing users or partners in a trust management system. The traditional approach to intrusion detection looks at system calls or network traffic, while application-level intrusion detection adds "insider" understanding to the analysis by being aware of the particular applications observed instead of trying to understand network traffic or system calls for the entire system.

We can divide intrusion detection into two main approaches. Anomaly detection attempts to model normal behaviour, often by learning from experience gained during an observation period [50, 51], and considers abnormalities potential signs of an attack. The second approach, misuse detection, constructs models to match the attacks instead [52]. While such specifications are less likely to yield false positives than detecting previously unseen behaviour in general, keeping them up to date is problematic—only known attacks can be detected. The approaches are not mutually exclusive, as is shown by IDES [53].

Specification-based anomaly detection attempts to combine the best of both worlds [54]. A specification for normal behaviour can be built with the help of e.g. source code. In the context of Web Services, contracts of acceptable behaviour may have already been made, possibly with the help of e.g. the Web Service Description Language (WSDL) [55]. If a suitable set of interface specifications for a particular Web Service can be found, it could be used as a basis for the specification of acceptable behaviour as well.

Thorough observation ties up resources, which may make it simply impossible to keep close track of what every user is doing at all times. Herrmann and Krumm, who study monitoring and trust or lack thereof directed towards a set of system components, suggest adjusting the intensity of monitoring and behaviour checks according to the level of trust in the observed component, its hosting environment and its vendor [56].

Suspicious activity can in the most straightforward case be actual misbehaviour in the form of breaking system policy or not following other forms of orchestration. It can, however, also be an action which either should only be taken by actors in a different role or is simply highly unusual behaviour for the observed. A change in an actor's behaviour may give reason to suspect that communications with the actor have been compromised, either on the way or in the source by subverting the actor or its representee somehow. The exact reason behind the unusual behaviour is not necessarily of consequence; the actor is not behaving as it should, and the observing system wants to protect itself against these possibly malicious influences.

When an observation system has detected suspicious activity, a decision must be made on what to do with the information. In the literature, the most visibly noted actions are updating the trust value or reputation (see the following chapter) and, if the analysis is done in real time, stopping the suspicious activity altogether. An Intrusion Prevention System (IPS) extends the concept of intrusion detection by also considering preemptive measures.

Automated reaction to detected attacks requires very accurate, real-time intrusion detection. The anomaly intrusion detection approach, with is tendency towards false positives, would therefore be sub-optimal if used alone in intrusion prevention. On the other hand, misuse intrusion detection would also miss some attacks due to not knowing them beforehand. Specification-based anomaly detection shows some promise, but building specifications of normal behaviour may not be feasible for all applications. It has been applied to network protocols [54], and could maybe find a place in the field of Web Services. Specifications of acceptable behaviour could potentially be composed based on the architecture's various specification languages, mainly the Web Service Description Language (WSDL) combined with probably necessary additional semantic information. Taking the step from merely being a language validator to observing trust-relevant activity may be challenging, however.

The idea of preventing policy-breaking or otherwise suspicious activity is not new. Access control lists have for long prevented users without specific identity-tied privileges from accessing certain files or services, and policy languages can be used to further limit access according to other constraints. They can also be used to lower the resources allocated for a slightly risky task which is not considered to be in direct conflict with policy, and as mentioned earlier, the task can be allowed to proceed normally, but under tighter observation as with trust decisions discussed earlier. Similar adjustments could be based on trust instead of more static, pre-set constraints.

Besides detecting suspicious activity, an observation system could be used as a witness of "normal" behaviour. Good experiences lead to better or at least more "certain" reputation in many reputation systems where the users themselves act as witnesses. On the other hand, if a reputation estimate includes a measure of confidence, i.e. how certain the estimate is, a lengthy period of observation showing behaviour in agreement with the current reputation may be taken as increased confidence in the reputation estimate.

3.3 Evolving Reputation and Trust

The evolution of reputation stands at the heart of a trust management system. It also seems to be a subject which is seldom discussed in detail in a practical context. One reason for this may be the need for configurability; research should not impose any particular policy on trust updates upon its applications. Some detailed examples in the right context can prove invaluable, however.

Mathematical models give tools and formulae for dealing with experience as it is represented as a binary for "cooperated vs. defected" [2] or by scalars [57]. The SECURE project provides a formal model of incorporating new evidence

to trust information [30, 41]. The Sultan project has also included an experience collection module in its architecture description [21, 58]. Translating experience into updates in reputation seems to largely be work in progress.

As the user's reputation is updated based on their actions, information about the changes can be sent as recommendations to reputation systems spanning larger communities, such as those used by the local reputation system to estimate the initial reputation of newcomers. The information can then be used to adjust the user's reputation in the target community as well. This requires that the recommendation includes a representation of the user's identity that is recognized in both communities. It is noteworthy that the reputation changes communicated across systems are not an objective truth by our definition, and the updates involve agreements on how the information is dealt with. This topic is central in the development of reputation systems.

4 Conclusions

Trust management is a young area of research. Trust as a concept has many very different applications, which causes divergence in trust management terminology. Also, conceptually separating trust from reputation is not always done, or nothing is said about how one affects the other [2]. Yet if either is forgotten, the remaining term's definition is left to bear both the aspect of perceptions and predictions as well as the willingness to depend and the related analysis of risks and benefits. Similarly, associating trust specifically to known actions instead of principals in general can make trust models more adaptable and understandable.

There has been some progress in the field of updating trust and reputation based on evidence of the actors' behaviour in the system. Yet while some projects include experience-collection modules in their systems [58, 10], practical studies on how to translate various suspicious or encouraging events into updates of reputation or trust are scarce. Theoretical models considering the topic assume that experiences have already been coded into either binary or scalar [2, 57]. Observation alone is a difficult task to automatize well; intrusion detection systems seek an automated way to answer to "is this an attack or just something resembling one?", and face similar problems. High configurability is a requirement for the observation system, or at least for its interpretation engine. As collaborative systems allow autonomic and dynamic policy changes at individual enterprises, conflicts in policy or expectations need to be detected run-time; static verification is no longer sufficient.

As a phenomenon, trust is such a multi-faceted research target that finding a satisfactory representation of it for computer systems must either be done based on a relatively limited context or not at all. The three-level view of trust research, from infrastructure to communities, was presented to keep these limitations of context in mind when evaluating earlier work. Still, there is work to do on all levels. An increased automation in trust management is needed for collaborative systems, especially for routine tasks. There should be room for human interven-

tion, however, for exceptions of the rule or new kinds of situations where the routine rules may not be applicable.

On one hand, it is reassuring to remember that trust is only a tool and as such can be simplified and toned down to suit our purposes. On the other hand, a tool which gives poor counsel due to not considering factors the user would want to give weight to is a tool easily abandoned. A tool might also be considered faulty to the degree of being unusable even if it knows better, according to a suitable definition of better, but constantly disagrees with its users in ways they do not comprehend.

References

1. Kutvonen, L., Viljanen, L., Ruohomaa, S.: The TuBE approach to trust management in collaborative enterprise systems. (2005) Manuscript.
2. Mui, L., Mohtashemi, M., Halberstadt, A.: A computational model of trust and reputation. In: 35th Annual Hawaii International Conference on System Sciences (HICSS'02). Volume 7., IEEE Computer Society (2002)
3. Fogg, B., Soohoo, C., Danielson, D., Marable, L., Stanford, J., Tauber, E.R.: How do people evaluate a web site's credibility? Technical report, Stanford Persuasive Technology Lab (2002)
4. Jonker, C.M., Schalken, J.J.P., Theeuwes, J., Treur, J.: Human experiments in trust dynamics. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004., Springer-Verlag (2004) 206–220
5. Marsh, S.: Formalising Trust as a Computational Concept. PhD thesis, University of Stirling, Department of Computer Science and Mathematics (1994)
6. Baldwin, A., Shiu, S.: Hardware security appliances for trust. In: Trust Management: First International Conference, iTrust 2003, Heraklion, Crete, Greece, May 28–30, 2003. Proceedings. Volume LNCS 2692/2003. (2003) 46–58
7. Djordjevic, I., Dimitrakos, T.: Towards dynamic security perimeters for virtual collaborative networks. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 191–205
8. Mayer, R.C., Davis, J.H.: An integrative model of organizational trust. The Academy of Management Review **20** (1995) 709–734
9. Jøsang, A., Presti, S.L.: Analysing the relationship between risk and trust. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 135–145
10. English, C., Terzis, S., Wagealla, W.: Engineering trust based collaborations in a global computing environment. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 120–134
11. Brændeland, G., Stølen, K.: Using risk analysis to assess user trust - a net-bank scenario -. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LCNS 2995/2004. (2004) 146–160
12. Egger, F.N.: From Interactions to Transactions: Designing the Trust Experience for Business-to-Consumer Electronic Commerce. PhD thesis, Eindhoven University of Technology (2003)

13. Grimsley, M., Meehan, A., Tan, A.: Managing Internet-mediated community trust relations. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 277–290
14. Ishaya, T., Mundy, D.P.: Trust development and management in virtual communities. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 266–276
15. Gordijn, J., Akkermans, H.: Designing and evaluating e-Business models. IEEE Intelligent Systems **16** (2001) 11–17
16. Tan, Y.H., Thoen, W., Gordijn, J.: Modeling controls for dynamic value exchanges in virtual organizations. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 236–250
17. Winsborough, W.H., Seamons, K.E., Jones, V.E.: Automated trust negotiation. In: DARPA Information Survivability Conference and Exposition, 2000. DISCEX '00. Proceedings. Volume 1., IEEE (2000) 88–102
18. Chu, Y.H., Feigenbaum, J., LaMacchia, B., Resnick, P., Strauss, M.: REFEREE: Trust management for Web applications. Computer Networks and ISDN Systems **29** (1997) 953–964
19. Blaze, M., Feigenbaum, J., Lacy, J.: Decentralized trust management. In: Proceedings of the IEEE Symposium on Security and Privacy, IEEE (1996)
20. Blaze, M., Feigenbaum, J., Keromytis, A.D.: KeyNote: Trust management for public-key infrastructures (position paper). In: Security Protocols: 6th International Workshop, Cambridge, UK, April 1998. Proceedings. Volume LNCS 1550/1998., Springer-Verlag (1998) 59–63
21. Grandison, T., Sloman, M.: Specifying and analysing trust for Internet applications. In: Proceedings of 2nd IFIP Conference on e-Commerce, e-Business, e-Government I3e2002, Lisbon, Portugal. (2002)
22. Damianou, N., Dulay, N., Lupu, E., Sloman, M.: The Ponder policy specification language. In: Workshop on Policies for Distributed Systems and Networks (Policy2001), HP Labs Bristol, 29-31 Jan 2001. Volume 1995. (2001) 18–
23. Tonti, G., Bradshaw, J.M., Jeffers, R., Montanari, R., Suri, N., Uszok, A.: Semantic Web languages for policy representation and reasoning: A comparison of KAoS, Rei, and Ponder. In: The SemanticWeb - ISWC 2003. Volume LCNS 2870/2003. (2003) 419–437
24. Uszok, A., Bradshaw, J.M., Jeffers, R.: KAoS: A policy and domain services framework for grid computing and Semantic Web services. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 16–26
25. Bradshaw, J.M.: KAoS: An open agent architecture supporting reuse, interoperability, and extensibility. In: Proceedings of Tenth Knowledge Acquisition for Knowledge-Based Systems Workshop. (1995)
26. Kagal, L., Finin, T., Joshi, A.: A policy language for a pervasive computing environment. In: Proceedings of IEEE 4th International Workshop on Policies for Distributed Systems and Networks (POLICY 2003), IEEE (2003) 63–74
27. Firozabadi, B.S., Sergot, M.: Revocation in the privilege calculus. In: Workshop on Formal Aspects of Security and Trust (FAST2003) at FM2003. Volume IIT TR-10/2003., IIT-CNR, Italy (2003) 39–51 URL `http://www.iit.cnr.it/FAST2003/fast-proc-final.pdf` (TR-10/2003).

28. Rissanen, E.: Server based application level authorisation for Rotor. IEE Proceedings Software **150** (2003) 291–295
29. Grandison, T., Sloman, M.: A survey of trust in Internet applications. IEEE Communications Surveys and Tutorials **3** (2000) 2–16
30. Wagealla, W., Carbone, M., English, C., Terzis, S., Nixon, P.: A formal model on trust lifecycle management. In: Workshop on Formal Aspects of Security and Trust (FAST2003) at FM2003. Volume IIT TR-10/2003. IIT-CNR, Italy (2003) 184–195 URL `http://www.iit.cnr.it/FAST2003/fast-proc-final.pdf` (TR-10/2003).
31. Resnick, P., Zeckhauser, R., Friedman, E., Kuwabara, K.: Reputation systems. Communications of the ACM **43** (2000) 45–48
32. Gray, E., Seigneur, J.M., Chen, Y., Jensen, C.: Trust propagation in small worlds. In: Trust Management: First International Conference, iTrust 2003, Heraklion, Crete, Greece, May 28–30, 2003. Proceedings. Volume LNCS 2692/2003. (2003) 239–254
33. Jøsang, A., Hird, S., Faccer, E.: Simulating the effect of reputation systems on e-markets. In: Trust Management: First International Conference, iTrust 2003, Heraklion, Crete, Greece, May 28–30, 2003. Proceedings. Volume LNCS 2692/2003. (2003) 179–194
34. Barber, K.S., Fullam, K., Kim, J. In: Challenges for Trust, Fraud and Deception Research in Multi-agent Systems. Volume 2631/2003 of Lecture Notes in Artificial Intelligence. Springer-Verlag (2003) 8–14
35. Abdul-Rahman, A., Hailes, S.: Supporting trust in virtual communities. In: Hawaii International Conference on System Sciences 33,HICSS. (2000)
36. Essin, D.J.: Patterns of trust and policy. In: Proceedings of 1997 New Security Paradigms Workshop, ACM Press (1997)
37. Gambetta, D.: Can we trust trust? Trust: Making and Breaking Cooperative Relations (2000) 213–237 Electronic edition.
38. Demolombe, R.: Reasoning about trust: A formal logical framework. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 291–303
39. Jøsang, A.: The right type of trust for computer networks. In: Proceedings of the ACM New Security Paradigms Workshop, ACM (1996)
40. Egger, F.N.: "Trust me, I'm an online vendor": Towards a model of trust for e-Commerce system design. In: Conference on Human Factors in Computing Systems, CHI'00 extended abstracts on Human factors in computing systems, ACM Press (2000)
41. Cahill, V., et al.: Using trust for secure collaboration in uncertain environments. Pervasive Computing **2** (2003) 52–61
42. Kutvonen, L.: Automated management of inter-organisational applications. In: Proceedings of the Sixth International Enterprise Distributed Object Computing Conference (EDOC '02). (2002) 27–38
43. Tan, Y.H.: A trust matrix model for electronic commerce. In: Trust Management: First International Conference, iTrust 2003, Heraklion, Crete, Greece, May 28–30, 2003. Proceedings. Volume LNCS 2692/2003. (2003) 33–45
44. Obreiter, P.: A case for evidence-aware distributed reputation systems overcoming the limitations of plausibility considerations. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 33–47

45. Fernandes, A., Kotsovinos, E., string, S., Dragovic, B.: Pinocchio: Incentives for honest participation in distributed trust management. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 64–77
46. Kalcklösch, R., Herrmann, K.: Statistical trustability (conceptual work). In: Trust Management: First International Conference, iTrust 2003, Heraklion, Crete, Greece, May 28–30, 2003. Proceedings. Volume LNCS 2692/2003. (2003) 271–274
47. Kaler, C., Nadalin, A., et al.: Web Services Trust Language (WS-Trust). (2004) Version 1.1.
48. Karabulut, Y.: Implementation of an agent-oriented trust management infrastructure based on a hybrid PKI model. In: Trust Management: First International Conference, iTrust 2003, Heraklion, Crete, Greece, May 28–30, 2003. Proceedings. Volume LNCS 2692/2003. (2003) 318–331
49. Zamboni, D.: Using Internal Sensors for Computer Intrusion Detection. PhD thesis, Purdue University (2001)
50. Teng, H.S., Chen, K., Lu, S.C.Y.: Adaptive real-time anomaly detection using inductively generated sequential patterns. In: 1990 IEEE Symposium on Research in Security and Privacy, May 7–9, 1990, IEEE Computer Society (1990) 278–284
51. Forrest, S., Hofmeyr, S., Somayaji, A., Longstaff, T.: A sense of self for Unix processes. In: 1996 IEEE Symposium on Security and Privacy, May 6–8, 1996, Oakland, California. (1996)
52. Kumar, S., Spafford, E.H.: A Pattern Matching Model for Misuse Intrusion Detection. In: Proceedings of the 17th National Computer Security Conference, Baltimore, Maryland, October 1994. (1994) 11–21
53. Denning, D.: An intrusion-detection model. IEEE Transactions on Software Engineering **13** (1987) 222–232
54. Sekar, R., Gupta, A., Frullo, J., Shanbhag, T., Tiwari, A., Yang, H., Zhou, S.: Specification-based anomaly detection: a new approach for detecting network intrusions. In: Proceedings of the 9th ACM Conference on Computer and Communications Security, Washington, DC, USA. (2002) 265–274
55. Chinnici, R., Gudgin, M., Moreau, J.J., Schlimmer, J., Weerawarana, S.: Web Services Description Language (WSDL) version 2.0 part 1: Core language, W3C working draft 10 November 2003. Technical report, World Wide Web Consortium (2003)
56. Herrmann, P., Krumm, H.: Trust-adapted enforcement of security policies in distributed component-structured applications. In: Proceedings of the 6th IEEE Symposium on Computers and Communications. Hammamet, Tunisia, IEEE Computer Society Press (2001) 2–8
57. Liu, J., Issarny, V.: Enhanced reputation mechanism for mobile ad hoc networks. In: Trust Management: Second International Conference, iTrust 2004, Oxford, UK, March 29–April 1, 2004. Proceedings. Volume LNCS 2995/2004. (2004) 48–62
58. Grandison, T.W.A., Sloman, M.: Sultan - a language for trust specification and analysis. In: Eighth Workshop of the HP OpenView University Association, Berlin, June 24-27, 2001, HP OpenView University Association (2001) URL `http://www.hpovua.org/PUBLICATIONS/PROCEEDINGS/8_HPOVUAWS/Papers/Paper01.2-Grandison-Sultan.pdf`.