Trust and Punishment*

Sandro Etalle
Distributed and Embedded Systems
University of Twente
sandro.etalle@utwente.nl

Jerry den Hartog
Distributed and Embedded Systems
University of Twente
jerry.denhartog@utwente.nl

Stephen Marsh
National Research Council Canada
Institute for Information Technology
steve.marsh@nrc-cnrc.gc.ca

ABSTRACT

In recent years we have witnessed a great increase in the interest in Trust Management (TM) techniques both from the industrial and the academic sectors. The booming research has also determined a duality in the very definition of TM system which can lead to confusion. In one of the two categories of TM systems a great deal of work has yet to be done in advancing the inherently adaptive nature of trust. This position paper examines reasons for the success of TM, the two broad TM categories, and, for reputation-based TM, issues of 'Regret Management' and accountability that are necessary enhancements on the road leading to much more sophisticated TM architectures.

1. INTRODUCTION

Trust Management is an approach to making decisions about interacting with something or someone we do not completely know, establishing whether we should proceed with the interaction or not. Consider for example the decision whether to grant access to a resource. Traditional access control schemes make authorization decisions based on the identity, or the role of the requester. However, when the resource owner and the requester are unknown to one another, making access control based on identity is ineffective.

In trust management systems, see e.g. [5, 4, 22, 16, 14, 2] decisions are made based on statements made by multiple principals. The decision who can be trusted, e.g. to access a resource, is taken not just by a single principal but by taking into account information from other principals. In this way the decision is, at least in part, *delegated* to the other principals. This is particularly important in presence of autonomous systems which typically operate in a dynamic and decentralized environments.

Trust management has become something of a hot topic, largely because it works, at least to some extent, and is based on a concept (that of trust) that makes inherent sense to us as humans in society. However, extant systems are not the final word, and each has its own issues. Most importantly, in many systems, the issue of accountability is poorly dealt with, and so punishing transgressions is harder than it should be or not as effective as it could be. In this paper we explore trust management in its current state, discuss issues associated in particular with reputation-based trust management, and arrive at a need to more fully address accountability, which we propose can be started with a discussion of another human-social concept, that of regret, leading to a notion of 'Regret Management'.

This paper is organized as follows. In the next section, trust management is introduced and two broad trust management concepts explored. Section 3 brings the concept of punishment and accountability into the trust management arena, moving through the concept of regret to a position on a more formal notion of regret management in section 4. We conclude with thoughts for the future in section 5.

2. TRUST MANAGEMENT

Trust management systems can be divided into two large categories: *rule based* and *reputation based* trust management (the latter are often referred to as *reputation systems*). Because of this duality, confusion often arises when talking about TM systems and the TM area in general. To put things in the right perspective, we now describe and compare these two categories.

2.1 Rule based TM

Let us first illustrate rule based trust management systems by means of a rather standard example, taken from [16]. Consider the situation in which a bookstore adopts the policy of giving a 10% discount to students of an accredited university. In our scenario, Alice comes to the book store and claims she is entitled to the discount. The bookstore will not have a list of all students of all accredited universities. Instead it will have a delegation rule stating that accredited universities are trusted to decide who are students. Furthermore, the bookstore will also not have a list of all accredited universities. Instead it will delegate this decision to an official accreditation body. In this way the bookstore can capture its policy in a very simple rule:

```
IF   The accreditation body states that
```

*This research has been supported by the following projects: BSIK/Poseidon, STW/I-Share.

AUTONOMICS 2007, 28-30 October 2007, Rome, Italy

DOI 10.4108/ICST.AUTONOMICS2007.2107

```
     X is a ‘‘university’’
AND  X states that Y is a ‘‘student’’
THEN Y is entitled to a discount
```

This is a canonical, simple trust management rule. It may look like things here are too simple to be true, and indeed the pragmatics behind checking a simple rule such as this one can become integrated. Let us, for a bit, explore why.

Returning to our scenario, we need to check that Alice is a student. Alice is a student at the University of Twente (UT) which is an accredited university so she is entitled to the discount. Remains the issue of how the bookstore checks this fact. For this it will need to obtain two extra statements: the first one from the accreditation body (AB) that UT is an accredited University and the second one from the UT that Alice is a student at the UT. In TM systems, such statements are often digitally signed to ensure their authenticity and integrity, and are called *credentials* or *certificates*. In this case the certificates from the UT and the AB need to be found and combined. Their combination forms a so-called *credential chain*.

Credential Chain Discovery. Finding such chains of credentials is a central topic in rule based TM and is strongly related to the decisions on the storage location of credentials. We now illustrate why referring to our example: one could store both credentials on the student ID of Alice; this makes it easy to find the chain as all credentials are available at the moment that Alice makes her request. However, it is unhandy for Alice to store not only all the credentials about herself, but also the credentials about the issuers of the credentials she owns (like the accreditation credential issued by AB). Moreover, storing such credentials by Alice would create problems in case of policy change. Another way is to store the credential 'Alice is a student' on her card but store the 'UT is an accredited University' credential at the UT or the accreditation body. Looking at the credential on Alices' ID card the bookstore will know that it needs to check the UT (mentioned in Alices' credential) or the accreditation body (mentioned) in the bookstore trust policy to find the missing credential(s). Finally one could consider the case that all credentials are store in a database at the UT. This case illustrates a potential problem for credential chain discovery: Alice does not have a credential and the bookstore only has a link to the accreditation body which does not have any credentials either; thus the available credentials cannot be used to find that the missing onces stored at the UT.

Trust Negotiation. A second important research topic in rule based TM systems is *trust negotiation* which relates to the issue of protecting credentials themselves. Credentials are often confidential (e.g., someone's medical record, or a credential proving that someone is a policeman), and should not be automatically disclosed to anyone requesting them. The mechanism to prevent unwanted disclosure of credentials is called *trust negotiation*; in our example the student card is used for all campus facilities so the student may want to be careful and to disclose it only to an *legitimate* bookstore. This requires that the bookstore first shows his credentials to the student; the shop, in turn may have additional requirements that should be satisfied before he shows his own credential, leading to trust negotiation [25, 24]: an iterative process of revealing credentials to build the required trust on both sides.

2.2 Reputation based TM

Reputation based trust management is now a well researched area [27]. The interest in reputation based systems comes from e.g. expert and auction systems [21], like*AllExperts* (http://www.allexperts.com), where everyone can ask an expert volunteer a question from the selected area. The user can then rate the expert so that other users be informed on the quality of advice given by different experts. An example of an auction system is *eBay* (http://www.ebay.com). In eBay, every user is welcome to leave a positive, negative or neutral feedback after each transaction. Sellers and buyers in eBay can rate each others and by this they can discourage (or encourage) prospective users to enter into business with another eBay user.

As an example consider the situation of a student (Anton) who wants to organize a joint book ordering to get a bulk discount and save on shipping costs. The more participants, the higher the savings; however, Anton does need to be sure that the people joining will actually show up and buy the book once it arrives, and letting them pay in advance may not be a viable option. In this case Anton can use a reputation based system: he asks people he trusts to join and to recommend others that they trust. Anyone with sufficiently strong recommendations may join the action. A scenario for this example: Anton's friend Alice joins the action, Anton's friend Bob does not join but recommends Charlie as being a very reliable person. As a result Charlie gets a sufficiently high trust value to also join in the book order.

Reputation-based TM systems support the decision making for questions such as (*"should Anton trust Charlie for the purpose of this transaction?"*), in a highly automated fashion: after deciding the required level of trust for a transaction one can even let the ultimate decision be made in a completely automatic way. Crucial practical questions in realizing this are: (1) how to express trust values also in relation with the kind of transaction they are used for – Anton may decide to trust Charlie if the books cost only 10 Euro, but would require additional guarantees in case the book cost was 1000 Eur; (2) to which extent should trust be *transitive*: if it is not transitive then the system becomes trivial; (3) how to adjust trust values in response to events (more on this later). Addressing these questions in the engineering of a reputation system leads to research issues such as trust metric definition and calculation based on local history and information provided by peers as well as recommendation exchange protocols for efficient and secure collection and exchange of such information. See e.g., the EigenTrust algorithm [15] and TrustRank [11].

It has been observed that reputation is an important factor which naturally supports the process of building trust among people [12, 21]. The role of a reputation system is then to collect, distribute, and aggregate feedbacks (reputations) concerning participants' past behavior [21]. The past behavior is usually expressed using a so called *trust metric*, which describes the agent's trust in another agent - most

often within some well defined context [1]. In defining trust and reputation, authors often refer to social sciences [1, 20] or economy and politics [21, 9]. In most of the formal approaches to reputation based trust management there is a clear distinction between a so called *direct* and *recommendation* trust [27, 13, 1, 26].

2.3 Rule Based vs. Reputation based systems

We now further elaborate on the two categories of TM systems by pointing out their main differences and similarities.

Analogies. The main analogy lies in the fact that, to reach a decision, in both kinds of systems need to combine information coming from different sources. In particular, the entity making the decision has to *trust* (at least to some extent) these sources and the information they provide. For instance, in our bookstore example (rule-based), the bookstore needed to trust the credentials issued by the UT and by the accreditation authority (note that if the UT started giving out student credentials to anyone asking for it, regardless whether he is a student or not, the bookstore would suffer monetary damage, as it would have to start giving discount to too many people; in this respect we can really say that bookstore must *entrust* the UT). Similarly, in our reputation-based example, Anton has to *trust* Bob's recommendation.

A second important analogy lies in the intrinsic *openness* of both categories of system: in both cases anyone can join or leave the system and any participating entity is also an *authority* that can issue her own credentials and her own recommendations. It will be up to the other participants to decide whether her credentials (or recommendations) will be trusted or not.

Differences. The main difference between the two systems lies in the role that *risk* plays in the system: while risk plays a central role in reputation based systems, it is at best a secondary factor in rule-based systems. In our example, once the bookstore has collected the credential showing that Alice is a student, it can proceed with the transaction and forget about it afterwards. The case of the recommendations that Anton collects for his bulk order is different: the fact that he trusts Alice and Bob (and - by transitivity - Charlie) does not mean that they will eventually show up with the money. Every time a reputation system is put in place, it is there to enable one to manage the risks that are embedded in a certain action. Think for instance of the reputation systems of eBay.

A consequence of this difference is that while rule based systems are basically boolean yes/no systems (Alice is either a student or not, there is no way in-between), reputation systems must employ multivalued domains to express *to which extent* someone is trusted. In rule based TM systems principals are either completely trusted for a purpose or not authorized for it. E.g. universities decide who are students; if a university says someone is a student, he is a student by definition, or put differently, the university states a fact rather than an opinion. The behaviour of the principals typically has no (directly) influence on the trust in the principal; only changes in the policies effect this trust. On the other hand, reputation based systems are numeric [6] and highly dynamic with each action influencing the reputations.

This brings us to what in our opinion is the most salient difference between rule and reputation systems: the fact that in reputation based systems, trust *varies in time*. Consider, for instance, our bulk-order example: assume that when the book finally arrives Alice promptly comes to collect her copy while Charlie does no show up. Because of this experience, the Alice's reputation (in the eyes of Anton) will increase, while Charlie's reputation will decrease, and – since Charlie was recommended by Bob – also Bob's reputation will suffer some damage. We subscribe to the view that a reputation based system has to reflect the "subjective probability by which an individual, A, expects that another individual, B, performs a given action on which A's welfare depends" (This is actually the disputed definition of trust originally given by Gambetta [10]. Here we do not want to get into the providing or subscribing to a definition of trust, rather, we look at the pragmatics of what trust management systems are and how they work or should work.) This probability has to reflect the actual behaviour of the various agents and therefore must be updated each time new information becomes available – that is, each time a new transaction takes place; in particular, misbehaviour should always yield a reduction of trust (cf. the reputation system in eBay).

On the other hand, in rule-based system there is usually no reason to update the rules and the credentials unless really exceptional situation arise. In any case, it should be clear that there is no need to update the system after each transaction.

3. PUNISHMENT, REGRET AND ACCOUNTABILITY

If we look at the pragmatics of *why* present reputation system manage to do reasonably well the job they are supposed to do we see that we should extend our horizon and take other factors into consideration: *regret* and ultimately *accountability*. For instance, when we engage in an eBay transaction with someone who has a very high reputation (say "98% positive feedback"), there are two reasons why we might decide to trust the seller.

- First, the fact that the previous transaction the seller has been involved in were successful ones as testified by the high percentage of positive feedback leads us to trust that also our transaction will be satisfactory for everyone; this factor coincides with the old definition of trust as *subjective probability*.
- Second, the fact that the seller often engages in eBay transactions and has a high reputation may lead us to put some confidence in the idea that the seller *wants* to maintain a high reputation to carry on with his business. In particular he will not want to defraud us because in that case we would give him a bad feedback and this will ruin his reputation.

The second point is important in that it shows that reputation systems work not only because they can help determining the subjective probability a transaction will be successful, but also because in their pragmatics they give

participants a way to damage each other in case of misbehaviour, which acts as deterrence. Below we discuss this in more detail, consider the strength of a deterrence, the role such deterrence could play in trust management systems and how this could be formalized into a concrete tool for making decision rather than the current informal and ad-hoc use.

If we accept that reputation based trust management systems allow the adaptation of trust over time, this leads to questions both about how such values can be adapted and, importantly, about how such adaptation can be used in the decision process. It is commonly accepted that a negative behaviour will reduce the corresponding level of trust, and this works quite well on a one to one basis. If I trust you less (if at all) there is less likelihood I will be burned another time. For instance, should Charlie not turn up to buy the book from Anton, the next time Anton wants to order books, he's unlikely to trust Charlie enough to allow him to join. That being said, Anton is still left holding the book, and the bill. In this system, Anton has not been able to hold Charlie to account for his (lack of) actions. What is needed, for Anton, is a way to ensure that either Charlie does turn up, or that if he doesn't, he is 'punished' and made to regret his actions.

We conjecture that there exists a potential to use trust management systems as a means of enforcing the *accountability* of others in a transaction or relationship. In this sense, both parties in a transaction need to know that there is a recourse to a system that will lead to the punishment of a transgressor.

Such a solution tends to alleviate the need for a trust in the other and incorporates it into a trust for the enforcing system – if a truster can have confidence (or 'trust') that a system does hold transgressors to account (and alleviates any hardship they themselves might suffer) *and knows the trustee has similar confidence*, then there is less need to consider trust in trustee. This is a topic in need of further exploration, and we will leave this aside for the moment and focus on a phenomenon that can help in this enforcement: that of regret.

3.1 Regret

'Trust is only required if a bad outcome would make you regret your decision.' [19, page 98].

Regret is not a new concept for study. In 1982, both Bell [3] and Loomes and Sugden [17] independently introduced the concept of regret theory in economics, itself based on the social psychological theory of counterfactual thinking [23], amongst other things. In game theory, the Savage/Regret Minimax theory has been extant for many years [18], itself based again on interpretation of more psychological theories. Much of what is presented here will reflect this previous work in some way.

Regret allows an action or happening to be looked upon as negative, and further allows the actors, or observers, to reinforce behaviours or associated feelings or emotions (such as trust) to ensure that the likelihood of such a thing happening again is reduced. It is, therefore, a powerful motivational force in interactions with others. Further, because it can have an effect in trust management systems, it is necessary to study, formalise, and concretise regret and its link with trust management to the extent that it becomes a computational tool similar to the current status of trust. Consider once more Anton and Charlie – should Charlie fail to turn up for his book, Anton would find himself regretting entering into the transaction with Charlie in the first place. In a formal model, the amount of that regret would be associated with the value of the book (and the bill) Anton is left with. Regret is in this sense a post-situational analysis tool that can help in the adjustment of trust in a trust management system.

Regret can also be a pre-situational decision making tool. Consider Charlie's point of view – what is needed is a way to make Charlie regret not turning up for the book. A formal notion of regret would put a value on the notion (again, perhaps proportional to the value of Anton's loss) that was large enough to ensure that Charlie had an actual loss of some kind. Charlie should know that if he fails to turn up, he will regret it. Knowing this before making the decision to *become* trusted by Anton would help Charlie make the decision about whether or not to enter into the transaction. The consideration is then – 'if I say I'll buy the book, I really have to because otherwise I lose money (or reputation). Can I afford the book in the first place?'

Further, consider if Charlie does enter into the transaction in good faith and then falls on hard times and is unable to collect and pay for the book. He may in an informal sense be able to express regret, just as Anton may be able to express his regret at trusting Charlie in the first place. Expressions and feelings of regret are both a means to being able to hold others accountable for their actions, and a means of ameliorating punishment. Anton can perceive Charlie's regret (perhaps in a formal setting, Charlie can try to make reparations) and this may make Anton less inclined to 'punish' Charlie. However, this aspect of regret is more difficult to automate because of its susceptibility to deceit, and we don't explore it further here.

In a automated formal model that uses regret as a tool, there are two questions a potential truster can ask that take regret into account:

- How much regret the truster can make the trustee feel if the trustee did something bad.
- How much this is going to cost (for instance, the cost of a legal approach or, perhaps, the cost to one's own reputation for being suckered and admitting it).

Problems with many present systems are that agents could vanish, use cheap pseudonyms and re-appear as someone else someplace else, or be unreachable because of barriers, and the available sanctions are not that onerous. These are less of an issue the more, for instance, your reputation means to you; e.g. an eBay vendor will not want to lose a carefully built up reputation by vanishing or by getting bad feedback.

We conjecture that in fact part of the trust we are developing and using now in reputation management systems is in fact a

surrogate for accountability using regret mechanisms. Trust is an *a priori* consideration tool, regret is both *a priori* and *a posteriori*, but both serve similar roles in decision making. More importantly, both can happily be implemented in a model that ever more closely resembles the human traits that give each its name.

4. REGRET MANAGEMENT – ACCOUNTABILITY

While regret is at turns an informal notion and (part of) a potential formal model, it serves as a tool for bringing the concretized notion of accountability to the fore, and for implementing automated systems that are able to hold transgressors accountable for their actions.

A 'regret management' system has the following properties:

- It is capable of assessing to some value the amount of regret a truster has after a trustee transgresses.
- It is capable of ensuring that the transgressor is 'assigned' that regret – that is, punished in some material (meaningful to the transgressor) way in a form proportional to the truster's regret.
- It is open and clear enough, and 'trusted' by both parties to be able to make these things happen. An 'untrusted' regret management system is as good as no system at all.

Ultimately, such a regret management system would ensure that transgressors are held accountable for their actions by ensuring that they regret these actions. As noted above, if this system were to exist, it would serve both parties as a pointer to helping them in making trusting decisions in the first place.

A posteriori access control [7] by using logging and auditing is a step towards such an open system. The logging and auditing mechanism makes it possible to find transgressions and also provides confidence to users that they will be able to hold others accountable. It also makes clear whom they are relying on when they trust the system (e.g. the auditing authority). A main additions that would be needed is a way of formalizing the amount of regret that can be assigned.

5. CONCLUSIONS

Trust management systems are a means of helping systems and people make decisions about actions and interactions under situations of risk and uncertainty. Regret, as a means of judging the results of an action, the cost of that action, and what might have been otherwise, is a powerful tool for trust-reasoning systems. Considering regret leads to a consideration of accountability, something we believe has been lacking in extant systems.

Existing reputation based systems often offer a form recourse in damaging reputation through negative feedback. However, the value of such regret is not formalized in any way and thus also not taking into account in the a priori trust evaluation (at least not formally). Thus these existing mechanisms may be considered to be insufficient to really achieve accountability.

To achieve accountability in a trust management system it is necessary to more fully describe regret as it relates to trust management, creating a formal model that can be implemented. Further, solid reputation management systems that truly take accountability into consideration and can enforce it are needed. Finally, once regret and accountability are considered, it is necessary to examine how a damaged trust following transgressions can be rebuilt taking regret into account. The lack of this vital last step will result in, to paraphrase Gandhi, a blind and toothless world.

Acknowledgments

We would like to acknowledge the contribution that W. H. Winsborough and D. Li gave to this paper by means of uncountable valuable discussion and a joint paper [8], which has been a source of inspiration for this work.

6. REFERENCES

[1] A. Abdul-Rahman and S. Hailes. Supporting Trust in Virtual Communities. In *Proc. 33rd Hawaii International Conference on System Sciences*, volume 6, page 6007. IEEE Computer Society Press, 2000.

[2] D. Artz and Y. Gil. A survey of trust in computer science and the semantic web. *Journal of Web Semantics*, 2007. to appear.

[3] D. E. Bell. Regret in decision making under uncertainty. *Operations Research*, 30:961–981, 1982.

[4] M. Blaze, J. Feigenbaum, J. Ioannidis, and A. Keromytis. The KeyNote Trust-Management System, Version 2. IETF RFC 2704, 1999.

[5] M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized Trust Management. In *Proc. 17th IEEE Symposium on Security and Privacy*, pages 164–173. IEEE Computer Society Press, 1996.

[6] P. Bonatti, C. Duma, D. Olemdilla, and N. Shahmehri. An Integration of Reputation-based and Policy-based Trust Management. In *Proc. Semantic Web and Policy Workshop*, 2005.

[7] J. G. Cederquist, R. J. Corin, M. A. C. Dekker, S. Etalle, J. I. den Hartog, and G. Lenzini. Audit-based compliance control. *International Journal of Information Security*, 6(2-3):133–151, 2007.

[8] M. R. Czenko, S. Etalle, D. Li, and W. H. Winsborough. An introduction to the role based trust management framework rt. Technical Report TR-CTIT-07-34, June 2007.

[9] C. Dellarocas. Analyzing the Economic Efficiency of eBay-like Online Reputation Reporting Mechanisms. In *Proc. 3rd ACM conference on Electronic Commerce*, pages 171–179. ACM Press, 2001.

[10] D. Gambetta. *Can We Trust Trust?* Basil Blackwell, 1988. Reprinted in electronic edition from Department of Sociology, University of Oxford, chapter 13, pp. 213-237.

[11] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with trustrank. In *VLDB*, pages 576–587, 2004.

[12] S. L. Jarvenpaa, N. Tractinsky, and M. Vitale. Consumer Trust in an Internet Store. *Inf. Tech. and Management*, 1(1-2):45–71, 2000.

[13] A. Jøsang. The Right Type of Trust for Distributed Systems. In *NSPW '96: Proc. Workshop on New Security Paradigms*, pages 119–131. ACM Press, 1996.

[14] A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 43(2):618–644, 2007.

[15] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The Eigentrust Algorithm for Reputation Management in P2P Networks. In *Proc. 12th International Conference on World Wide Web*, pages 640–651. ACM Press, 2003.

[16] N. Li, J. Mitchell, and W. Winsborough. Design of a Role-based Trust-management Framework. In *Proc. IEEE Symposium on Security and Privacy*, pages 114–130. IEEE Computer Society Press, 2002.

[17] L. Loomes and R. Sugden. Regret theory: An alternative theory of rational choice under uncertainty. *Economic Journal*, 92:805–824, 1982.

[18] R. D. Luce and H. Raiffa. *Games and Decisions*. Dover Publications, 1957.

[19] N. Luhmann. Familiarity, confidence, trust: Problems and alternatives. In D. Gambetta, editor, *Trust*, chapter 6, pages 94–107. Blackwell, 1990.

[20] L. Mui, M. Mohtashemi, and A. Halberstadt. A Computational Model of Trust and Reputation for E-businesses. *Hicss*, 07:188, 2002.

[21] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman. Reputation systems. *Commun. ACM*, 43(12):45–48, 2000.

[22] R. Rivest and B. Lampson. SDSI – A Simple Distributed Security Infrastructure, October 1996. Available at http://theory.lcs.mit.edu/∼rivest/sdsi11.html.

[23] N. J. Roese and J. M. Olson, editors. *What Might Have Been: The Social Psychology of Counterfactual Thinking*. Lawrence Erlbaum Associates, Mahwah, NJ, 1995.

[24] K. E. Seamons, M. Winslett, T. Yu, B. Smith, E. Child, J. Jacobson, H. Mills, and L. Yu. Requirements for Policy Languages for Trust Negotiation. In *Proc. 3rd International Workshop on Policies for Distributed Systems and Networks (POLICY 2002)*, pages 68–80. IEEE Computer Society Press, 2002.

[25] W. H. Winsborough and N. Li. Towards practical automated trust negotiation. In *POLICY*, pages 92–103. IEEE Computer Society Press, 2002.

[26] L. Xiong and L. Liu. PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities. *IEEE Trans. Knowl. Data Eng.*, 16(7):843–857, 2004.

[27] R. Yahalom, B. Klein, and T. Beth. Trust Relationships in Secure Systems – A Distributed Authentication Perspective. In *RSP: IEEE Computer Society Symposium on Research in Security and Privacy*. IEEE Computer Society, 1993.