A Trust Evaluation Method Based on Logic and Probability Theory

Reto Kohlas, Jacek Jonczy, and Rolf Haenni

Abstract We introduce a trust evaluation method applicable in a decentralized setting, in which no universally trusted authority exists. The method makes simultaneous use of logic and probability theory. The result of the qualitative part of the method are logical arguments for and against the reliability of an entity. The quantitative part returns the probability that the reliability of an entity can be deduced under the given assumptions and pieces of evidence, as well a corresponding probability for the counter-hypothesis. Our method is a true generalization of existing methods, in particular the Credential Networks. It relies on digital signatures for authenticating messages and accounts for many-to-many relationships between entities and public keys. Moreover, it includes eight different types of trust relations, namely the assumption or the statement that an entity is honest, competent, reliable, or malicious, and their corresponding negations.

1 Introduction

Members of global social networks and e-commerce systems regularly face the question whether they can trust other, a priori unknown entities. A rating system or a *trust evaluation* method can provide decision support in such a situation. It indicates arguments for the reliability of an entity (or a numerical value representing an entity's reliability, respectively) by taking available trust assumptions, recommendations and discredits into account.

The credibility of a statement, for example a recommendation, generally depends on the reliability of its author (i.e., the honesty and the competence of its author). In

---

Reto Kohlas and Jacek Jonczy
Institute of Computer Science and Applied Mathematics, University of Berne, 3012 Berne, Switzerland, e-mail: kohlas,jonczy@iam.unibe.ch

Rolf Haenni
Bern University of Applied Sciences, 2501 Biel/Bienne, Switzerland, e-mail: rolf.haenni@bfh.ch

a digital setting, messages should be authenticated, since the identity of the alleged sender of a message can typically be forged without effort. The method presented in this paper makes use of *public-key cryptography* and digital signature schemes for message authentication.

The use of public-key cryptography requires the *authentication of public keys*, i.e., the establishment to which physical entity a public key belongs. *Public-key certificates* are digitally signed statements which approve the authenticity of a public-key entity for a physical entity.[1] They contribute thus to public-key authentication and are useful for those physical entities who cannot exchange their public keys personally.

The main purpose of this paper is to introduce a novel trust evaluation method that relies on logic and probability theory. It uses digital signatures for message authentication and extends previously proposed approaches.

1.1 Existing Trust Evaluation Methods

Some authors have noted that early methods for evaluating trust or for authenticating public keys tend to return counter-intuitive results. Deficiencies in PGP's Web of Trust for instance have been identified in [21, 18, 13], principles that such methods should ideally fulfill have been stated in [23, 18]. In search of improved techniques, a vast number of methods has been proposed in the last decade.

Some methods combine the confidence values *specifically*, in the sense that their way of combining the confidence values has been exclusively conceived for trust evaluation. Examples of such specific methods are [2, 22, 23, 26, 1, 19, 20]. Other methods treat trust evaluation as a special case of accepting or rejecting a *hypothesis* (that a public key is authentic or that an entity is reliable) under *uncertain assumptions* and *pieces of evidence* (public-key certificates, recommendations, discredits). Such methods use *formal techniques for reasoning under uncertainty*, and are often based on a *probabilistic* interpretation of the confidence values. Examples are Maurer's Probabilistic Model [21] (based on Probabilistic Logic), Jøsang's Certification Algebra [14] (based on Subjective Logic), Haenni's Key Validation Method [6] and the Credential Networks [12] (both based on the Theory of Probabilistic Argumentation).

We here briefly describe Maurer's probabilistic method (MPM), since it allows us to exemplify in Subsection 1.2 in which sense we intend to extend existing probabilistic methods. The basic idea behind MPM is the combination of *logic* and *probability theory*. MPM's *deterministic* model consists of two so-called *inference rules*. The first inference rule asserts that if a reasoner A knows the authentic public key of X ($\mathrm{Aut}_{A,X}$), if X is trusted by A for issuing public-key certificates ($\mathrm{Trust}_{A,X,1}$)[2], and

[1] For reasons explained in Subsection 2.1, these entities are called *public-key* and *physical entities*.

[2] The third index is an integer and corresponds to the *trust level* (its exact meaning is irrelevant for the discussion in this paper).

if X issues a public-key certificate for Y ($\text{Cert}_{X,Y}$)[3], then A can conclude to possess the authentic public key of Y ($\text{Aut}_{A,Y}$). Formally, this rule translates into

$$\forall X\ \forall Y:\ \text{Aut}_{A,X} \wedge \text{Trust}_{A,X,1} \wedge \text{Cert}_{X,Y} \vdash \text{Aut}_{A,Y}.$$

The second inference rule (which we do not print here) describes the role of recommendations for evaluating the reliability of a physical entity. Note that MPM considers positive recommendations only (i.e., there are no statements asserting that some entity is unreliable).

The *probabilistic* model of MPM lets A assign a probability to every assumption. Each probability, also called confidence value, is intended to stand for A's degree of belief with respect to the truth of the judged assumption. MPM then defines confidence value for the hypothesis $\text{Aut}_{A,B}$ as function of the initially attributed probabilities. This confidence value corresponds to the probability that $\text{Aut}_{A,B}$ can be deduced from A's initial view by applying consecutively the two inference rules of the deterministic model.

1.2 Motivation

In accordance with other contributions, we propose to use a probabilistic framework as the basis of our method. However, we suggest to revisit existing probabilistic methods with respect to the type of assumptions and certificates (credentials) they take into account. The following list discusses some important and often neglected modeling aspects:

- *Physical entities may use multiple public-key entities.* Most methods assume that each physical entity uses *at most* one public-key entity. In MPM, for example, the supposed public key of X is precisely for this reason not included in the statement $\text{Aut}_{A,X}$. As a consequence, statements signed by different public-key entities are usually considered independent. However, it is often impossible in a decentralized system to limit the number of keys used, since each entity can generate as many public-key entities and distribute as many public keys as desired. If some physical entity controls two public-key entities, then statements signed by these two public-key entities are by no means independent.
- *Two physical entities can share the use of a public-key entity.* It is usually impossible to assure that one public-key entity is controlled by only *one* physical entity. A key holder can for instance disclose the passphrase for accessing the private key to another physical entity, and thereby share control of the public-key entity. Such sharing of public-key entities can be problematic. If both physical entities control the public-key entity, it is not possible to uniquely assign state-

[3] Note that X uses her public-key entity to issue a certificate for Y's public key. But neither X nor Y's public key are parameters in the statement $\text{Cert}_{X,Y}$, because all physical entities are assumed to control *exactly one* public-key entity.

ments signed by the public-key entity to either of the physical entities. As a consequence, if the two physical entities are not equally trusted, it is impossible to determine the credibility of the signed statement in a unique way.

- *The opposite of trust is twofold.* Trust is often modeled as positive assumption only allowing to conclude what trusted introducers say. If an introducer is *not* trusted (e.g., in MPM the statement $\text{Trust}_{A,X,1}$ would not be valid), no conclusions are drawn within these methods. But it is possible that malicious entities lie. In the context of public-key authentication, persuading someone to use the "wrong" public key allows to decrypt messages and make statements in someone else's name; providing "false" statements about reliability could convince somebody to enter a deal to the cheating entity's advantage. In Subsection 2.3 we shall therefore differentiate between two opposites of trust: first, as an entity's belief that a given introducer is *incompetent*, and second as the stronger assumption that the introducer is *malicious*, in which case the contrary of what the introducer says can be deduced.
- *Negative statements.* Many existing methods are monotonic. In MPM for instance, if A adds a public certificate to her view, the confidence value $conf(\text{Aut}_{A,B})$ remains the same or increases (but it does not decrease). There is never evidence for the hypothesis that a public key is *not* authentic. The reason for the monotonicity of the methods lies in the fact that only positive statements are taken into account. However, negative and positive statements are equally important. If a honest introducer observes that someone else is spreading false information, or that a public key is not authentic, this honest introducer should have a means at hand to warn other participants. We intend therefore to include different types of negative statements in our model.

1.3 Goal and Outline

The goal of this paper is to propose a trust evaluation method that considers the modeling aspects mentioned in the previous subsection. We base our method on the Theory of Probabilistic Argumentation [16, 9, 7] (TPA), which allows us to cope with conflicting assumptions and evidence. Moreover, hypotheses can be evaluated qualitatively and quantitatively. The qualitative part of the method provides logical arguments for and against a hypothesis. The results of the quantitative evaluation are two corresponding probabilities of derivability.

The emphasis of this paper lies primarily in the preciseness and not in the practicability of the proposed method. By suggesting a more accurate model we hope to understand the mechanisms behind trust evaluation better. Aspects of efficiency and usability will be part of future work.[4]

This paper is organized as follows. In Section 2 we introduce our model. Section 3 describes the logical and probabilistic evaluation of hypotheses concerning

[4] We are confident that practicable implementations are possible, as recent experiences in the context of the Credential Networks have shown [11].

reliability and public-key authenticity. We conclude with Section 4 by discussing the contributions of our paper and directions for future research.

2 A Model for Reliability and Public-Key Authenticity

We start by recapitulating an existing entity-relationship model [17], which has been conceived for the public-key authentication problem (Subsection 2.1). The relationships defined within this model are used as predicates in the evidential language $\mathscr{L}$ (described in Subsection 2.2), which allows to formalize the terms of public-key authenticity and trust in Subsection 2.3. Finally, we introduce the concept of a trust and authenticity network in Subsection 2.4.

2.1 Entities and Relationships

The model we consider consists of two types of *entities*. A *physical world entity* (*physical entity* for short) is someone that exists in the reality of the "physical" world. Examples are natural persons (human beings) or legal persons (companies, governmental agencies, sports clubs, etc.). A *public-key entity* consists of a public key and a private key, as well as a signature generation and a signature verification algorithm. Access to the private key is needed for generating signatures in the public-key entity's name.

Finding unique and adequate names for physical entities can be difficult, especially for natural persons. Here we put the naming problem aside and assume that each entity is known under exactly one, unique identifier. We use $\mathtt{p_1}, \ldots \mathtt{p_m}$ to denote tht physical entities of our model, and $\mathtt{k_1}, \ldots, \mathtt{k_n}$ the $\mathtt{n}$ public-key entities. The symbol $\mathtt{b}$ represents the entity whose reliability is evaluated. Corresponding capital letters refer to variables, i.e., the indexed variable $\mathtt{P_i}$ to a physical entity and $\mathtt{K_i}$ to a public-key entity. We use $\mathscr{P}$ to denote the set of physical entities, and $\mathscr{K}$ stands for the set of public-key entities. Entities of our model can stand in the following *relationships*:

- A physical entity `controls` a public-key entity whenever she has access to its private key. Access occurs through knowledge of a password or passphrase, through possession of a physical device such as a smartcard, or through a biometric attribute. The same public-key entity can be controlled by more than one physical entity. A physical entity can control more than one public-key entity.
- The relationship `signs` involves a public-key entity and a statement. It holds if there exists a digital signature under the statement, which has been generated by using the private key of the public-key entity. Note that `signs` does not indicate which physical entity is using or controlling a public-key entity.
- The relationship `authors` stands for the fact that it was in a physical entity's *intention* to be the author of a statement. Authoring a statement can mean to say

it, to write it on a piece of paper, to type it into a computer, or to create any other representation of the statement. The digital signature provides evidence that a physical entity has authored the signed statement.

2.2 An Evidential Language

We use a formal language $\mathscr{L}$ to model assumptions, pieces of evidence, and their logical relationship in the context of trust evaluation. Due to limited space, we cannot provide the exact formal definitions of $\mathscr{L}$ here, but we give at least the basic idea behind $\mathscr{L}$.

The relationships introduced in the previous subsection are used as predicates in $\mathscr{L}$, the elements of $\mathscr{L}$ are called $\mathscr{L}$-formulas. $\mathscr{L}$ is a *many-sorted logic* without function symbols.[5] "Many-sorted" means that all variables and the arguments of the predicates are of a specified sort. We consider three sorts, namely the physical entities, the public-key entities, and the statements. The atoms of $\mathscr{L}$ are the relationships introduced in the previous subsection; a distinguished predicate symbol is the equality sign. An atomic $\mathscr{L}$-formula is a predicate symbol together with arguments of appropriate sort. An argument of predicate is either a constant symbol or a variable. Examples of atomic $\mathscr{L}$-formulas are $\mathtt{controls}(\mathtt{p_1},\mathtt{k_3})$, $\mathtt{controls}(\mathtt{P_1},\mathtt{k_3})$, or $\mathtt{P_1} = \mathtt{p_2}$ (note that in the two latter formulas $\mathtt{P_1}$ stands for a *variable* of sort physical entity, and not for a constant). $\mathscr{L}$ contains the usual logical connectives: $\wedge$ (logical and), $\vee$ (logical or), $\neg$ (not), $\rightarrow$ (material implication), $\leftrightarrow$ (bidirectional material implication), and $\forall$ (universal quantifier). In the sequel, let $\mathtt{L}$ and the indexed variable $\mathtt{L_i}$ stand each for an $\mathscr{L}$-formula.

2.3 Formalizing Reliability and Public-Key Authenticity

2.3.1 Reliability and Maliciousness

We differentiate among three types of introducers in our model. A physical entity is *reliable* if she is competent and honest; statements authored by a reliable principal are believable. The second type of introducers are those who are *incompetent*. If a statement $\mathtt{L}$ is authored by an incompetent introducer, it is impossible to decide whether $\mathtt{L}$ is true or false; $\mathtt{L}$ can be true by chance, independently of the introducer's honesty. Therefore statements made by incompetent entities should be simply ignored. The third type of physical entities are the *malicious* introducers. A malicious entity is competent but dishonest, and tries to deceive other physical entities by spreading credentials that contain a false statement. Under the assumption that someone is malicious one can conclude the contrary of what the suspected in-

[5] Except for constant symbols such as $\mathtt{p_1}$, which can be seen as 0-ary function symbols.

troducer says. We therefore define the reliability ($\texttt{rel}$) and maliciousness ($\texttt{mal}$) of a physical entity depending on her competence ($\texttt{comp}$) and honesty ($\texttt{hon}$) as follows:[6]

Rule 1: *Reliable physical entity*.

$$\forall.\mathtt{P}\ :\quad \big(\mathtt{rel(P)}\ \leftrightarrow\ (\mathtt{comp(P)} \wedge \mathtt{hon(P)})\big) \tag{1}$$

Rule 2: *Malicious physical entity*.

$$\forall.\mathtt{P}\ :\quad \big(\mathtt{mal(P)} \leftrightarrow\ (\mathtt{comp(P)} \wedge \neg\mathtt{hon(P)})\big) \tag{2}$$

The logical relationship between a physical entity $\mathtt{P}$'s honesty and her competence, as well as the truth of a statement $\mathtt{L}$ authored by $\mathtt{P}$ are captured by the following two rules. On the one hand, if $\mathtt{P}$ is believed to be reliable, the statement $\mathtt{L}$ authored by $\mathtt{P}$ can be believed. On the other hand, if $\mathtt{P}$ is assumed to be malicious, we invert the truth of the uttered statement $\mathtt{L}$:

Rule 3: *Statement authored by a reliable physical entity*.

$$\forall.\mathtt{P}\ \forall.\mathtt{L}:\quad \big((\mathtt{authors(P,L)}\ \wedge\ \mathtt{rel(P)}) \rightarrow\ \mathtt{L}\big) \tag{3}$$

Rule 4: *Statement authored by a malicious physical entity*.

$$\forall.\mathtt{P}\ \forall.\mathtt{L}:\quad \big((\mathtt{authors(P,L)}\ \wedge\ \mathtt{mal(P)}) \rightarrow \neg\mathtt{L}\big) \tag{4}$$

2.3.2 Public-Key Authenticity

Public-key authenticity of $\mathtt{K}_1$ for $\mathtt{P}_1$ means that $\mathtt{P}_1$, but no other entity $\mathtt{P}_2$, controls $\mathtt{K}$. This formally translates into Rule (5).

Rule 5: *Definition of public-key authenticity*.

$$\begin{aligned}&\forall.\mathtt{P}_1\ \forall.\mathtt{P}_2\ \forall.\mathtt{K}\ :\\ &\mathtt{aut}(\mathtt{P}_1,\mathtt{K})\ \leftrightarrow\ \big(\mathtt{controls}(\mathtt{P}_1,\mathtt{K}) \wedge ((\mathtt{P}_1 \neq \mathtt{P}_2) \rightarrow \neg\mathtt{controls}(\mathtt{P}_2,\mathtt{K}))\big)\end{aligned} \tag{5}$$

Because the variables $\mathtt{P}_1$, $\mathtt{P}_2$, and $\mathtt{K}$ are universally quantified, Rule (5) is valid for all physical entities $\mathtt{P}_1$ and $\mathtt{P}_2$, as well as all public-key entities $\mathtt{K}$.

Rule (6) formalizes a simplified view of the security of a digital signature scheme: If only $\mathtt{P}$ has access to $\mathtt{K}$ (i.e., $\mathtt{aut(P,K)}$ holds), and if there is a digital signature under the statement $\mathtt{L}$ by $\mathtt{K}$, then $\mathtt{P}$ authored the statement $\mathtt{L}$:

Rule 6: *Ascribing digital signatures to physical entities*.

$$\forall.\mathtt{P}\ \forall.\mathtt{K}\ \forall.\mathtt{L}\ :\quad \big((\mathtt{aut(P,K)}\ \wedge\ \mathtt{signs(K,L)})\ \rightarrow\ \mathtt{authors(P,L)}\big) \tag{6}$$

[6] We do not have to define incompetent introducers at this point, since the assumptions that someone is incompetent allows no conclusion.

2.4 Trust and Authenticity Networks

For evaluating a hypothesis concerning the reliability of a physical entity or the authenticity of a public key, a reasoner A takes certain *assumptions* and collects a set of *credentials*. Assumptions and credentials are either with respect to the authenticity of public keys or the reliability of entities. Assumptions are *subjective*; A decides which assumptions are acceptable for her. A credential is a statement which is either digitally signed by a public-key entity or authored by a physical entity.

A's assumptions and credentials form what we call her *Trust and Authenticity Network* (*TAN*). A TAN can be depicted by a *multigraph*. We use drawn-through arrows for authenticity assumptions and credentials, similarly to [21, 6, 12]. The graph in Fig. 1 (a) shows A's assumption that k_1 is authentic for p_1, the graph in Fig. 1 (b) represents the statement that k_2 is authentic for p_2, and is digitally signed by k_1. An example of a *negative* authenticity credential (i.e., a statement that a public key is not authentic) is depicted in Figure 1 (c); negative statements are indicated by the negation sign $\neg$. For the moment, we consider only assumptions and credentials about the `aut` predicate, but it is conceivable to incorporate `controls` statements in a future method. Dashed arrows represent trust assumptions and credentials. Whereas

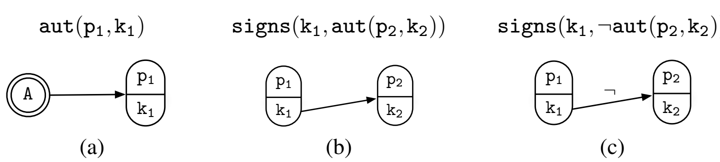


Fig. 1 An authenticity assumption and two authenticity credentials.

A's assumption that p_1 is reliable constitutes a positive trust assumption, her belief that p_1 is incompetent or cheating is negative. We use the following abbreviations: R for a `rel` assumption and credential, I for `incomp`, and M for `mal`. Figure 2 shows

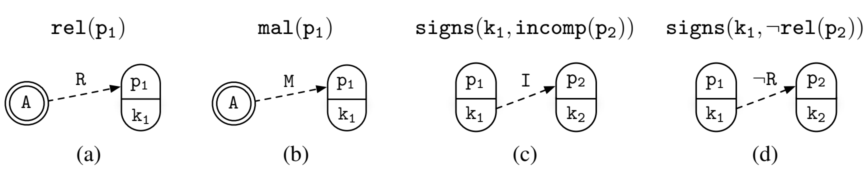


Fig. 2 Two trust assumptions (of entity A) and two trust credentials (digitally signed by the public-key entity k_1).

examples of trust assumptions and credentials. In the graph of Fig. 2 (a), A believes that $\mathtt{p}_1$ is reliable, in Fig. 2 (b) A assumes that $\mathtt{p}_1$ is malicious. The graph in Fig. 2 (c) shows a statement, digitally signed by $\mathtt{k}_1$, which asserts that $\mathtt{p}_2$ is incompetent. Finally, the graph in Figure 2 (d) provides an example of a negated trust statement: the key owner of $\mathtt{k}_1$ claims that $\mathtt{p}_2$ is not reliable (which is not equal to the statement that $\mathtt{p}_2$ is malicious). TAN-assumptions and credentials can be connected, which results in a multigraph as depicted in Fig. 3.

3 Reasoning about Reliability and Public-Key Authenticity

Our reasoner A is possibly *uncertain* about several of her assumptions; A might doubt the reliability of an introducer; authenticity assumptions can be uncertain if the identification process of an alleged public-key entity owner is error-prone. In analogous manner, credentials can also be uncertain; an introducer can express her uncertainty about an assertion contained within a credential by assigning a *weight* to it. The logical and probabilistic reasoning allows A to evaluate her hypotheses under uncertain assumptions and credentials.

3.1 Logical Reasoning

In this subsection we explain the basic ideas behind *scenarios*, *assumptions* and *arguments*, which are TPA's building blocks for reasoning logically about hypotheses. The definitions coincide to some extent with those provided in [7].

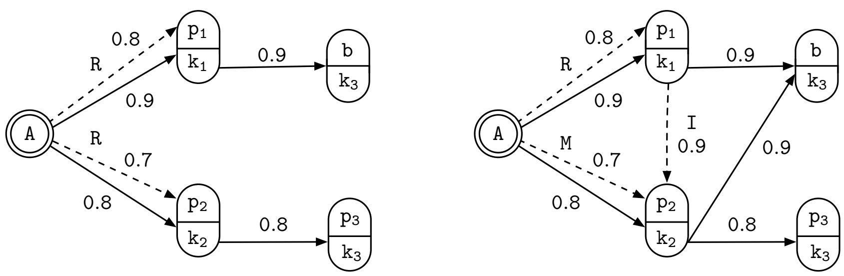
3.1.1 Assumptions

An *assumption* (in the sense of TPA) is a basic unit of concern which is uncertain from the point of view of an entity A. With respect to a TAN, all edges in the multigraph are assumptions, as discussed in the introduction of this section. From a syntactic point of view, an assumption is an $\mathscr{L}$-formula which consists of a predicate symbol and constant symbols of appropriate sort. In the example of Fig. 3 (a), $\mathtt{aut}(\mathtt{p}_1,\mathtt{k}_1)$, $\mathtt{aut}(\mathtt{p}_2,\mathtt{k}_2)$, $\mathtt{rel}(\mathtt{p}_1)$ $\mathtt{rel}(\mathtt{p}_2)$, $\mathtt{signs}(\mathtt{k}_1,\mathtt{aut}(\mathtt{b},\mathtt{k}_3))$, and $\mathtt{signs}(\mathtt{k}_2,\mathtt{aut}(\mathtt{p}_3,\mathtt{k}_3))$ are assumptions.

3.1.2 Scenarios

A *scenario* is specified by a truth value assigned to each of the assumptions. Given n assumptions, 2^n scenarios exist. A scenario is denoted by the symbol S. If an assumption A is true (false) in S, we write $S(A) = 1$ ($S(A) = 0$). It is assumed that there is exactly one scenario which represents the real state of the world. Unfortunately, $\texttt{A}$ does not know which scenario meets this condition.

With respect to a knowledge base (and hence with respect to a given TAN), the set of scenarios can be divided into *conflicting* and *consistent* scenarios. A conflicting scenario stands in contradiction with the knowledge base, a consistent scenario on the other hand is non-conflicting. In Fig. 3 (a), the scenario in which all assumptions hold is conflicting. An informal explanation is the following: from the two $\texttt{signs}$ and $\texttt{aut}$ assumptions we can conclude - by applying Rule (6) - that $\texttt{p}_1$ authored $\texttt{aut}(\texttt{b},\texttt{k}_3)$ and $\texttt{p}_2$ authored $\texttt{aut}(\texttt{p}_3,\texttt{k}_3)$. Since $\texttt{p}_1$ and $\texttt{p}_2$ are trusted, by using Rule (2) $\texttt{aut}(\texttt{b},\texttt{k}_3)$ and $\texttt{aut}(\texttt{p}_3,\texttt{k}_3)$ can be derived. But Rule (5) asserts that $\texttt{aut}(\texttt{b},\texttt{k}_3)$ and $\texttt{aut}(\texttt{p}_3,\texttt{k}_3)$ cannot hold both at same time. Hence the scenario is conflicting. All other scenarios are consistent with respect to the TAN of Fig. 3.



(a) (b)

Fig. 3 Two simple TANs.

With respect to a hypothesis h, the set of consistent scenarios can be divided into *supporting, refuting*, and *neutral* scenarios [7]. A supporting scenario is a consistent scenario that allows the deduction of h. A refuting scenario is a scenario supporting the counter-hypothesis $\neg h$. A neutral scenario with respect to h is a consistent scenario which is neither supporting nor refuting h. An example of a supporting scenario for $\texttt{aut}(\texttt{b},\texttt{k}_3)$ is

$$\begin{aligned}
&S(\texttt{aut}(\texttt{p}_1,\texttt{k}_1)) = 1, \quad && S(\texttt{rel}(\texttt{p}_1)) = 1, \quad && S(\texttt{signs}(\texttt{k}_1,\texttt{aut}(\texttt{b},\texttt{k}_3))) = 1,\\
&S(\texttt{aut}(\texttt{p}_2,\texttt{k}_2)) = 1, \quad && S(\texttt{rel}(\texttt{p}_2)) = 0, \quad && S(\texttt{signs}(\texttt{k}_2,\texttt{aut}(\texttt{p}_3,\texttt{k}_3))) = 1.
\end{aligned}$$

The assumptions $\texttt{aut}(\texttt{p}_1,\texttt{k}_1)$, $\texttt{signs}(\texttt{k}_1,\texttt{aut}(\texttt{b},\texttt{k}_3))$, and $\texttt{rel}(\texttt{p}_1)$ allow to conclude $\texttt{aut}(\texttt{b},\texttt{k}_3)$ (by Rule (5) and Rule (2)). The scenario is not conflicting, since by the assumed falsity of $\texttt{rel}(\texttt{p}_2)$ the lower "certification path" in Fig. 3 (a) is broken; hence $\texttt{A}$ cannot conclude $\texttt{aut}(\texttt{p}_3,\texttt{k}_3)$ (otherwise this would lead to a contradiction). An example of a refuting scenario for $\texttt{aut}(\texttt{b},\texttt{k}_3)$ is

$$\begin{aligned}
&S(\texttt{aut}(\texttt{p}_1,\texttt{k}_1)) = 1, \quad && S(\texttt{rel}(\texttt{p}_1)) = 0, \quad && S(\texttt{signs}(\texttt{k}_1,\texttt{aut}(\texttt{b},\texttt{k}_3))) = 1,\\
&S(\texttt{aut}(\texttt{p}_2,\texttt{k}_2)) = 1, \quad && S(\texttt{rel}(\texttt{p}_2)) = 1, \quad && S(\texttt{signs}(\texttt{k}_2,\texttt{aut}(\texttt{p}_3,\texttt{k}_3))) = 1.
\end{aligned}$$

The scenario is supporting $\texttt{aut}(\texttt{p}_3,\texttt{k}_3)$, and since we do not accept a public key as authentic for two physical entities, $\neg\texttt{aut}(\texttt{b},\texttt{k}_3)$ follows.

	$qs(\bot)$
(1)	$\texttt{aut}(\texttt{p}_1,\texttt{k}_1)\wedge\texttt{aut}(\texttt{p}_2,\texttt{k}_2)\wedge\texttt{rel}(\texttt{p}_1)\wedge\texttt{rel}(\texttt{p}_2)\wedge$ $\texttt{signs}(\texttt{k}_1,\texttt{aut}(\texttt{b},\texttt{k}_3))\wedge\texttt{signs}(\texttt{k}_2,\texttt{aut}(\texttt{p}_3,\texttt{k}_3))$
	$sp(\texttt{aut}(\texttt{b},\texttt{k}_3))$
(1)	$\texttt{aut}(\texttt{p}_1,\texttt{k}_1)\wedge\texttt{rel}(\texttt{p}_1)\wedge\texttt{signs}(\texttt{k}_1,\texttt{aut}(\texttt{b},\texttt{k}_3))\wedge\neg\texttt{aut}(\texttt{p}_2,\texttt{k}_2)$
(2)	$\texttt{aut}(\texttt{p}_1,\texttt{k}_1)\wedge\texttt{rel}(\texttt{p}_1)\wedge\texttt{signs}(\texttt{k}_1,\texttt{aut}(\texttt{b},\texttt{k}_3))\wedge\neg\texttt{rel}(\texttt{p}_2)$
(3)	$\texttt{aut}(\texttt{p}_1,\texttt{k}_1)\wedge\texttt{rel}(\texttt{p}_1)\wedge\texttt{signs}(\texttt{k}_1,\texttt{aut}(\texttt{b},\texttt{k}_3))\wedge\neg\texttt{signs}(\texttt{k}_2,\texttt{aut}(\texttt{p}_3,\texttt{k}_3))$
	$sp(\neg\texttt{aut}(\texttt{b},\texttt{k}_3))$
(1)	$\texttt{aut}(\texttt{p}_2,\texttt{k}_2)\wedge\texttt{rel}(\texttt{p}_2)\wedge\texttt{signs}(\texttt{k}_2,\texttt{aut}(\texttt{p}_3,\texttt{k}_3))\wedge\neg\texttt{aut}(\texttt{p}_1,\texttt{k}_1)$
(2)	$\texttt{aut}(\texttt{p}_2,\texttt{k}_2)\wedge\texttt{rel}(\texttt{p}_2)\wedge\texttt{signs}(\texttt{k}_2,\texttt{aut}(\texttt{p}_3,\texttt{k}_3))\wedge\neg\texttt{rel}(\texttt{p}_1)$
(3)	$\texttt{aut}(\texttt{p}_2,\texttt{k}_2)\wedge\texttt{rel}(\texttt{p}_2)\wedge\texttt{signs}(\texttt{k}_2,\texttt{aut}(\texttt{p}_3,\texttt{k}_3))\wedge\neg\texttt{signs}(\texttt{k}_1,\texttt{aut}(\texttt{b},\texttt{k}_3))$

Table 1 $qs(\bot)$, $sp(\texttt{aut}(\texttt{b},\texttt{k}_3))$, and $sp(\neg\texttt{aut}(\texttt{b},\texttt{k}_3))$ for the TAN of Fig. 3 (a).

3.1.3 Arguments

A compact logical representation of scenarios is achieved by means of *arguments*. Technically, an *argument* is a conjunction of assumption literals. There are conflicting, supporting anf refuting arguments,, analogously to the different types of scenarios. The expression $qs(\bot)$ represents the set of minimal conflicting assumptions; $sp(h)$ and $sp(\neg h)$ stand for the sets of minimal arguments supporting and refuting h, respectively.

The arguments of the TANs discussed in this paper have been determined by translating first the TAN into a *Propositional Argumentation System* (i.e., a knowledge base in which all variables have been instantiated and the universal quantifiers have been removed). The so-obtained propositional knowledge base was implemented in ABEL [10], a framework for evaluating propositional knowledge bases qualitatively and quantitatively.

3.1.4 Examples

Table 1 shows the minimal argument sets for the example depicted Fig. 3 (a). As mentioned, there is only one conflicting scenario. Hence we have only one conflicting argument containing all assumptions. The common part of the supporting arguments for $\texttt{aut}(\texttt{b},\texttt{k}_3)$ are the three assumptions of the upper certification path of our example. The assumptions $\neg\texttt{aut}(\texttt{p}_2,\texttt{k}_2)$, $\neg\texttt{rel}(\texttt{p}_2)$, and $\neg\texttt{signs}(\texttt{k}_2,\texttt{aut}(\texttt{p}_3,\texttt{k}_3))$ all guarantee that the argument is not conflicting. Each argument stands for four scenarios (because there are two missing assumptions in each argument supporting $\mathtt{aut}(\mathtt{b},\mathtt{k}_3)$). The supporting arguments for $\neg\mathtt{aut}(\mathtt{b},\mathtt{k}_3)$ are in a certain sense symmetric to the arguments for $\mathtt{aut}(\mathtt{b},\mathtt{k}_3)$. They actually correspond to the supporting arguments for $\mathtt{aut}(\mathtt{p},\mathtt{k}_3)$. Note that Table 1 lists only the *minimal* arguments. For example, the argument

$$\mathtt{aut}(\mathtt{p}_1,\mathtt{k}_1)\wedge\mathtt{rel}(\mathtt{p}_1)\wedge\mathtt{signs}(\mathtt{k}_1,\mathtt{aut}(\mathtt{b},\mathtt{k}_3))\wedge\neg\mathtt{aut}(\mathtt{p}_2,\mathtt{k}_2)\wedge\mathtt{rel}(\mathtt{p}_2)$$

supports also $\mathtt{aut}(\mathtt{b},\mathtt{k}_3)$, but is contained in argument (1) of $sp(\mathtt{aut}(\mathtt{b},\mathtt{k}_3))$ of Table 1.

	$qs(\bot)$
(1)	$\mathtt{aut}(\mathtt{p}_1,\mathtt{k}_1)\wedge\mathtt{rel}(\mathtt{p}_1)\wedge\mathtt{signs}(\mathtt{k}_1,\mathtt{incomp}(\mathtt{p}_2))\wedge\mathtt{mal}(\mathtt{p}_2)$
(2)	$\mathtt{aut}(\mathtt{p}_1,\mathtt{k}_1)\wedge\mathtt{aut}(\mathtt{p}_2,\mathtt{k}_2)\wedge\mathtt{rel}(\mathtt{p}_1)\wedge\mathtt{signs}(\mathtt{k}_1,\mathtt{aut}(\mathtt{b},\mathtt{k}_3))\wedge$ $\mathtt{mal}(\mathtt{p}_2)\wedge\mathtt{signs}(\mathtt{k}_2,\mathtt{aut}(\mathtt{b},\mathtt{k}_3))$
	$sp(\mathtt{aut}(\mathtt{b},\mathtt{k}_3))$
(1)	$\mathtt{aut}(\mathtt{p}_1,\mathtt{k}_1)\wedge\mathtt{rel}(\mathtt{p}_1)\wedge\mathtt{signs}(\mathtt{k}_1,\mathtt{aut}(\mathtt{b},\mathtt{k}_3))\wedge\neg\mathtt{mal}(\mathtt{p}_2)$
(2)	$\mathtt{aut}(\mathtt{p}_1,\mathtt{k}_1)\wedge\mathtt{rel}(\mathtt{p}_1)\wedge\mathtt{signs}(\mathtt{k}_1,\mathtt{aut}(\mathtt{b},\mathtt{k}_3))\wedge$ $\neg\mathtt{signs}(\mathtt{k}_1,\mathtt{incomp}(\mathtt{p}_2))\wedge\neg\mathtt{signs}(\mathtt{k}_2,\mathtt{aut}(\mathtt{b},\mathtt{k}_3))$
(3)	$\mathtt{aut}(\mathtt{p}_1,\mathtt{k}_1)\wedge\mathtt{rel}(\mathtt{p}_1)\wedge\mathtt{signs}(\mathtt{k}_1,\mathtt{aut}(\mathtt{b},\mathtt{k}_3))\wedge$ $\neg\mathtt{signs}(\mathtt{k}_1,\mathtt{incomp}(\mathtt{p}_2))\wedge\neg\mathtt{aut}(\mathtt{p}_2,\mathtt{k}_2)$
	$sp(\neg\mathtt{aut}(\mathtt{b},\mathtt{k}_3))$
(1)	$\mathtt{aut}(\mathtt{p}_2,\mathtt{k}_2)\wedge\mathtt{mal}(\mathtt{p}_2)\wedge\mathtt{signs}(\mathtt{k}_2,\mathtt{aut}(\mathtt{b},\mathtt{k}_2))\wedge\neg\mathtt{aut}(\mathtt{p}_1,\mathtt{k}_1)$
(2)	$\mathtt{aut}(\mathtt{p}_2,\mathtt{k}_2)\wedge\mathtt{mal}(\mathtt{p}_2)\wedge\mathtt{signs}(\mathtt{k}_2,\mathtt{aut}(\mathtt{b},\mathtt{k}_2))\wedge\neg\mathtt{rel}(\mathtt{p}_1)$
(3)	$\mathtt{aut}(\mathtt{p}_2,\mathtt{k}_2)\wedge\mathtt{mal}(\mathtt{p}_2)\wedge\mathtt{signs}(\mathtt{k}_2,\mathtt{aut}(\mathtt{b},\mathtt{k}_2))\wedge$ $\neg\mathtt{signs}(\mathtt{k}_1,\mathtt{aut}(\mathtt{b},\mathtt{k}_3))\wedge\neg\mathtt{signs}(\mathtt{k}_1,\mathtt{incomp}(\mathtt{p}_2))$

Table 2 $qs(\bot)$, $sp(\mathtt{aut}(\mathtt{b},\mathtt{k}_3))$, and $sp(\neg\mathtt{aut}(\mathtt{b},\mathtt{k}_3))$ for the TAN of Fig. 3 (b).

Figure 3 (b) shows an example which is more complicated. In contrast to the previous example, A believes that $\mathtt{p}_2$ is malicious. There is an additional digitally signed trust credential which claims incompetence for $\mathtt{p}_2$. The owner of $\mathtt{k}_2$ provides conflicting information, as she claims simultaneously public-key authenticity of $\mathtt{k}_3$ for b and $\mathtt{p}_3$.

The qualitative evaluation provides some interesting insights: The second conflicting argument in Table 2 is equal to the only conflicting argument of our first example. Argument (1) of $qs(\bot)$ conflicts with the given TAN because $\mathtt{aut}(\mathtt{p}_1,\mathtt{k}_1)$, $\mathtt{rel}(\mathtt{p}_1)$, and $\mathtt{signs}(\mathtt{k}_1,\mathtt{incomp}(\mathtt{p}_2))$ allow to conclude that $\mathtt{p}_2$ is incompetent. This, however, conflicts with the assumption $\mathtt{mal}(\mathtt{p}_2)$, which stands for the assumption that $\mathtt{p}_2$ is competent (and dishonest).

All supporting arguments for $\mathtt{aut}(\mathtt{b},\mathtt{k}_3)$ in the second TAN contain the three assumptions of the upper certification path. Again, some negated assumptions have to be added to guarantee the consistency of the supporting arguments. For example, the first supporting argument contains the literal $\neg\mathtt{mal}(\mathtt{p}_2)$. By adding this assumption, a contradiction can be prevented. Note that - in contrast to the previous example - there are no supporting arguments for $\mathtt{aut}(\mathtt{p}_3, \mathtt{k}_3)$. If $\mathtt{p}_2$ is indeed *not* malicious, she is `incompetent` *or* `honest`. From this clause $\mathtt{aut}(\mathtt{p}_3, \mathtt{k}_3)$ can *not* be deduced.

3.2 Probabilistic Reasoning

The idea of the probabilistic part of TPA (and hence of our method) is that from `A`'s point of view each scenario corresponds with a certain probability to the real state of the world. `A` has to choose the probabilities such that the sum of the probabilities assigned to the scenarios equals one. Given the exponential number of scenarios, it is infeasible for `A` to estimate the probability of each single scenario. It is often justifiable to consider the statements of a TAN as being stochastically independent. In this case, `A` assigns a probability to all of her authenticity and trust assumptions. The weights assigned to each credential are represented by a probability, too. Under

	$dqs(\bot)$	$dsp(h_1)$	$dsp(\neg h_1)$	$dsp(h_2)$	$dsp(\neg h_2)$
TAN of Fig. 3(a)	0.290	0.504	0.222	0.222	0.504
TAN of Fig. 3(b)	0.486	0.403	0.282	0.000	0.654

Table 3 Qualitative evaluation of TANs: $h_1 = \mathtt{aut}(\mathtt{b}, \mathtt{k}_3)$, $h_2 = \mathtt{aut}(\mathtt{p}_3, \mathtt{k}_3)$.

the independence assumption, the probability of a scenario can be computed as the product of the marginal probabilities of the assumptions.

Formally, let A_i stand for the ith assumption, and let p_i be the probability attached to A_i. Given a scenario S, let S^+ denote the assumptions which are positive in S, and S^- the assumptions that occur negatively:

$$S^+ = \{A \in S \mid S(A) = 1\}, \quad S^- = \{A \in S \mid S(A) = 0\}.$$

The probability $P(S)$ of scenario S is then defined as

$$P(S) = \prod_{A_i \in S^+} p_i \cdot \prod_{A_i \in S^-} (1 - p_i).$$

The *degree of conflict*, denoted by $dqs(\bot)$, is obtained by summing up the *probabilities of the conflicting scenarios*. It is a measure of how conflicting the assumptions are with respect to the knowledge base (i.e., the TAN). Let $dqs(h)$ stand for the sum of $dqs(\bot)$ and the probabilities of all the scenarios supporting h (i.e., the sum of the probabilities of all scenarios allowing the deduction of h, including the conflicting ones). The *degree of support* for the hypothesis h, denoted by $dsp(h)$, is the probability that h can be derived, provided that the real scenario is not conflicting.

Formally, the degree of support corresponds to

$$dsp(h) = \frac{dqs(h)}{1 - dqs(\perp)}.$$

Table 3 shows the degrees of support for our two examples of Fig. 3. In example of Fig. 3 (a), $\texttt{aut}(\texttt{b},\texttt{k}_3)$ and $\neg\texttt{aut}(\texttt{p}_3,\texttt{k}_3)$ are quite probable. In the TAN of Fig. 3 (b), $\texttt{aut}(\texttt{b},\texttt{k}_3)$ is less probable, but we have no evidence for $\texttt{aut}(\texttt{p}_3,\texttt{k}_3)$. In both cases, A either accepts $\texttt{aut}(\texttt{b},\texttt{k}_3)$ or collects additional evidence to gain more certainty for or against the validity of $\texttt{aut}(\texttt{b},\texttt{k}_3)$. A discussion of how to validate a hypothesis based on $dsp(h)$ and $dsp(\neg h)$ can be found in [8].

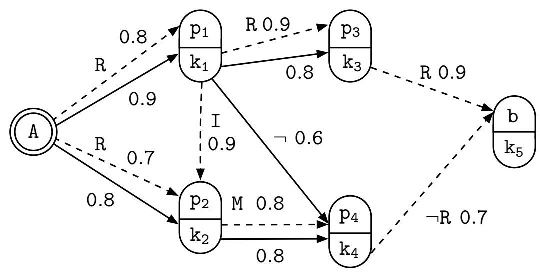


Fig. 4 A more complex TAN.

We end the presentation of our probabilistic method by analyzing the more complicated TAN depicted in Figure 4. The TAN involves most trust and authenticity statements introduced in this paper. It contains a negative authenticity statement ($\texttt{signs}(\texttt{k}_1,\neg\texttt{aut}(\texttt{p}_4,\texttt{k}_4))$), and a negative trust statement ($\texttt{signs}(\texttt{k}_3,\neg\texttt{rel}(\texttt{b}))$). We are interested in the evaluation of the reliability of b, i.e., the hypothesis of interest is $h = \texttt{rel}(\texttt{b})$. Although the TAN is not that large, it has already a considerable complexity: there are twelve edges in the TAN, and hence $2^{12} = 4096$ possible scenarios. The qualitative evaluation yields two conflicting and seven supporting arguments (which we do not write down here). One of the two conflicting arguments corresponds to

$$\begin{aligned}&\texttt{aut}(\texttt{p}_1,\texttt{k}_1) \wedge \texttt{aut}(\texttt{p}_2,\texttt{k}_2) \wedge \texttt{rel}(\texttt{p}_1) \wedge \texttt{rel}(\texttt{p}_2) \wedge \\ &\texttt{signs}(\texttt{k}_1,\neg\texttt{aut}(\texttt{p}_4,\texttt{k}_4)) \wedge \texttt{signs}(\texttt{k}_2,\texttt{aut}(\texttt{p}_4,\texttt{k}_4)).\end{aligned}$$

The above argument is conflicting, since it allows the deduction of $\texttt{aut}(\texttt{p}_4,\texttt{k}_4)$ *and* $\neg\texttt{aut}(\texttt{p}_4,\texttt{k}_4)$. The degree of conflict $dqs(\perp)$ is quite high and is approximately 0.473; the degree of support $dsp(\texttt{rel}(\texttt{b}))$ is roughly 0.442. Interestingly, there is no argument for the hypothesis that b is not reliable ($dsp(\texttt{rel}(\texttt{b})) = 0$).

4 Conclusion

We have introduced a trust evaluation method, which can also be used for authenticating public keys. The used, extended model considers the possibility that a public-key entity is shared by different physical entities, and that a physical entity controls several public-key entities at the same time. Negative statements are an integral part of the method. Reliability is decomposed into honesty and competence, which allows to differentiate between incompetent and malicious physical entities. The assumptions and the available evidence from the perspective of a physical entity A can be represented by an evidential language and by a multigraph. We make use of the Theory of Probabilistic Argumentation, which allows to cope with conflicting assumptions. TPA provides logical arguments *and* probabilities of derivability for and against the hypotheses in question.

Future work consists in investigating the applicability of our method in concrete systems, and in devising specific algorithms for the evaluation of trust and authenticity networks. Possible extensions of the model are the inclusion of trust scopes and time aspects, as well as modeling the revocation of statements and public keys.

Acknowledgements This research was supported by the Swiss National Science Foundation, Project No. PP002-102652/1.

References

1. The official ebay website. *http://www.ebay.com*, April 2006.
2. T. Beth, M. Borcherding, and B. Klein. Valuation of trust in open networks. In *ESORICS'94, 3rd European Symposium on Research in Computer Security*, LNCS 875, pages 3–18. Springer, 1994.
3. M. Burrows, M. Abadi, and R. Needham. A logic of authentication. *ACM Transactions on Computer Systems*, 8(1):18–36, February 1990.
4. W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, 1976.
5. L. Gong, R. Needham, and R. Yahalom. Reasoning About Belief in Cryptographic Protocols. In Deborah Cooper and Teresa Lunt, editors, *Proceedings 1990 IEEE Symposium on Research in Security and Privacy*, pages 234–248. IEEE Computer Society, 1990.
6. R. Haenni. Using probabilistic argumentation for key validation in public-key cryptography. *International Journal of Approximate Reasoning*, 38(3):355–376, 2005.
7. R. Haenni. Probabilistic argumentation (submitted). *Elsevier*, 2007.
8. R. Haenni, J. Jonczy, and R. Kohlas. Two-layer models for managing authenticity and trust. In R. Song, L. Korba, and G. Yee, editors, *Trust in E-Services: Technologies, Practices and Challenges*. 2006.
9. R. Haenni, J. Kohlas, and N. Lehmann. Probabilistic argumentation systems. In D. M. Gabbay and P. Smets, editors, *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, volume 5: Algorithms for Uncertainty and Defeasible Reasoning, pages 221–288. Kluwer Academic Publishers, Dordrecht, Netherlands, 2000.
10. R. Haenni and N. Lehmann. ABEL: an interactive tool for probabilistic argumentative reasoning. In *ECSQARU'03, 7th European Conference on Symbolic and Quantitative Approaches to Reasoning under Uncertainty*, pages 588–593, Aalborg, Denmark, 2003.

11. J. Jonczy. Evaluating trust and authenticity with CAUTION. In *iTrust'06, 4rd International Conference on Trust Management*, pages 449–453, Pisa, Italy, 2006.
12. J. Jonczy and R. Haenni. Credential networks: a general model for distributed trust and authenticity management. In A. Ghorbani and S. Marsh, editors, *PST'05: 3rd Annual Conference on Privacy, Security and Trust*, pages 101–112, St. Andrews, Canada, 2005.
13. J. Jonczy, M. Wüthrich, and R. Haenni. A probabilistic trust model for GnuPG. In *23C3, 23rd Chaos Communication Congress*, pages 61–66, Berlin, Germany, 2006.
14. A. Jøsang. An algebra for assessing trust in certification chains. In *NDSS'99: 6th Annual Symposium on Network and Distributed System Security*, San Diego, USA, 1999.
15. A. Jøsang. A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(3):279–311, 2001.
16. J. Kohlas and P. A. Monney. *A Mathematical Theory of Hints. An Approach to the Dempster-Shafer Theory of Evidence*, volume 425 of *Lecture Notes in Economics and Mathematical Systems*. Springer, 1995.
17. R. Kohlas, R. Haenni, and J. Jonczy. A new model for public-key authentication. In T. Braun, G. Carle, and B. Stiller, editors, *KiVS'07, Kommunikation in Verteilten Systemen*, pages 213–224, Berne, Switzerland, 2007.
18. R. Kohlas and U. Maurer. Confidence valuation in a public-key infrastructure based on uncertain evidence. In H. Imai and Y. Zheng, editors, *PKC'2000, Third International Workshop on Practice and Theory in Public Key Cryptography*, LNCS 1751, pages 93–112, Melbourne, Australia, 2000. Springer.
19. R. Levien and A. Aiken. Attack-resistant trust metrics for public key certification. In *7th on USENIX Security Symposium*, pages 229–242, 1998.
20. G. Mahoney, W. Myrvold, and G. C. Shoja. Generic reliability trust model. In A. Ghorbani and S. Marsh, editors, *PST'05: 3rd Annual Conference on Privacy, Security and Trust*, pages 113–120, St. Andrews, Canada, 2005.
21. U. Maurer. Modelling a public-key infrastructure. In E. Bertino, H. Kurth, G. Martella, and E. Montolivo, editors, *ESORICS, European Symposium on Research in Computer Security*, LNCS 1146, pages 324–350. Springer, 1996.
22. M. K. Reiter and S. G. Stubblebine. Path independence for authentication in large-scale systems. In *CCS'97, 4th ACM Conference on Computer and Communications Security*, pages 57–66, Zürich, Switzerland, 1997. Academic Press.
23. M. K. Reiter and S. G. Stubblebine. Toward acceptable metrics of authentication. In *SP'97: 18th IEEE Symposium on Security and Privacy*, pages 10–20, Oakland, USA, 1997.
24. R. L. Rivest, A. Shamir, and L. M. Adelman. A method for obtaining digital signatures and public-key cryptosystems. Technical Report TM-82, MIT, Cambridge, USA, 1977.
25. C. G. Zarba. Many-sorted logic. http://theory.stanford.edu/˜zarba/snow/ch01.pdf.
26. P. R. Zimmermann. *PGP User's Guide Volume I: Essential Topics*, 1994.