A Vector Model of Trust for Developing Trustworthy Systems

Indrajit Ray and Sudip Chakraborty

Colorado State University
Fort Collins, CO 80523, USA
{indrajit, sudip}@cs.colostate.edu

Abstract. All security services rely to a great extent on some notion of trust. However, even today, there is no accepted formalism or technique for the specification of trust and for reasoning about trust. Secure systems have been developed under the premise that concepts like "trusted" or "trustworthy" are well understood, unfortunately without even agreeing to what "trust" means, how to measure it, how to compare two trust values and how to combine two trust values. In this work we propose a new vector model of trust. Our model proposes the notion of different degrees of trust, differentiates between trust and distrust and formalizes the dependence of trust on time. We believe that our model will help answer some of the questions posed earlier.

1 Introduction

Confidentiality, integrity and availability of systems and information resources are increasingly becoming critical in our everyday life. To protect such resources it is important that we are able to determine the appropriate security policies. The notion of *trust* plays a critical role for the proper formulation of security policies. However, even today, there are no accepted formalisms or techniques for the specification of trust and for reasoning about trust. Secure systems have been built under the premise that concepts like "trustworthiness" or "trusted" are well understood, unfortunately without even agreeing on what "trust" means, how to measure it, how to compare two trust values and how to compose the same. This creates a number of problems in building secure systems, particularly those that are composed from several different components.

Consider, for example, the operational information base in a large corporation. Typically, this is generated with the accumulation of information from several sources. Some of these sources are under the direct administrative control of the corporation and thus are considered trustworthy. Other sources are "friendly" sources and information originating directly from them are also considered trustworthy. However, these "friendly" sources may have derived information from their own sources which the corporation does not have any first hand knowledge about; if such third-hand information is made available to the corporation, then the corporation has no real basis for determining the quality of that information. It will be rather naive for the corporation to trust this information to the same extent that it trusts information from sources under its direct control. Similarly not trusting this information at all is also too simplistic. Existing binary models of trust (where trust has only two values, "no trust" and "complete trust" and which

are the ones most widely used in computer systems) will, nonetheless, categorize the trust value to one of these two levels. Existing trust models (even those that associate multiple levels to trust) do not provide satisfactory answers to questions such as: (i) What expectations can the corporation reasonably have about the usefulness of such information? (ii) What are the activities that the corporation can expect such information to fulfill without much problem? (iii) What are the activities that the corporation does not want to fulfill using this information?

The above observations prompt us to propose a new model of trust in which trust is defined as a vector of numeric values. Each element of the vector is a parameter in determining the value of trust. We identify three such parameters in our model. We propose methods to determine the values corresponding to these parameters. Substituting values for each of these parameters in the trust vector provides a value for trust of a certain degree. To make the concept of different degrees of trust more intuitive, we associate a numeric value in the range $[-1, 1]$ with the trust vector. The value in the positive region of this range is used to express trust and that in the negative region is used to express distrust. Uncertainty about trust and distrust is expressed using the value zero. We define operators to map a trust vector to a trust value within this range and also from a trust value to a trust vector. We investigate the dynamic nature of trust – how trust (or distrust) changes over time. Finally we observe that trust depends on trust itself – that is a trust relationship established at some point of time in the past will influence the computation of trust at the current time. We formalize this notion in our model.

The rest of the paper is organized as follows: In section 2 we briefly describe some of the more important works in the area of trust models. In section 3 we present our model of trust. We begin this section with the definition of trust that we use in the rest of the work. We define the parameters that contribute towards a value for trust. In sections 3.3, 3.4 and 3.5 we derive expressions to estimate each of these parameters. Then in section 3.6 we introduce the concept of normalized trust followed by, in section 3.7, the definition of the concept of value of trust. Section 3.8 deals with trust dynamics – the dependence of trust on time. In section 4 we define the dominance relation between two trust relationships that allow us to identify how two trust relationships compare. Finally section 5 concludes with a discussion of our future work.

2 Related Work

A number of logic-based formalisms of trust have been proposed by researchers. Almost all of these view trust as a binary relation. Forms of first order logic [1, 2, 3] and modal logic or its modification [4] have been variously used to model trust in these cases. Simple relational formulae like *A* trusts *B* are used to model trust between two entities. Each formalism extends this primitive construct to include features such as temporal constraints and predicate arguments. Given these primitives and the traditional conjunction, disjunction, negation and implication operators, these logical frameworks express trust rules in some language and reason about these properties. Abdul-Rahman and Hailes [3] propose a trust model, based on "reputation" that allows artificial agents to reason about trustworthiness and allows real people to automate that process. Jones and Firozabadi [5] models trust as the issue of reliability of an agent's transmission.

They use a variant of modal logic to model various trust scenarios. They also use their language to model the concepts of deception and an entity's trust in another entity.

Yahalom et al. [6, 7] propose a formal model for deriving new trust relationships from existing ones. In [6] the authors propose a model for expressing trust relations in authentication protocols, together with an algorithm for deriving trust relations from recommendations. In [7] rules and algorithms for obtaining public keys based on trust relationships are developed. Neither of these works define what is meant by trust. Beth et al. [8] extend the ideas presented by Yahalom et al. to include relative trust. This work proposes a method for extracting trust values based on experiences and recommendations and also a method for deriving new trust values from existing ones within a network of trust relationships. Jøsang [9, 10, 11] proposes a model for trust based on a general model for expressing relatively uncertain beliefs about the truth of statements. Trust is an opinion, which is expressed as a triplet $< b,d,u > \in \{b,d,u\}$. Here b, d, and u are respectively measures of one's belief, disbelief, and uncertainty in a proposition. A major shortcoming of this model is that it has no mechanism for monitoring trust relationships to re-evaluate their constraints. Cohen et al. [12] propose an alternative, more differentiated conception of trust, called Argument-based Probabilistic Trust model (APT). The most important use of APT is to chart how trust varies, from one user to another, from one decision aid to another, from one situation to another, and across phases of decision aid use.

Xiong and Liu [13] present a coherent adaptive trust model for quantifying and comparing the trustworthiness of peers based on a transaction-based feedback system. They propose three basic trust parameters – peer feedback through transactions, total number of transactions a peer performs, and credibility of the feedback sources. The authors address factors that influence peer-to-peer trust, like reputation systems and misbehavior of peers by giving false feedback. The authors also provide a trust metric for predicting a given peer's likelihood of a successful transaction in the future. Purser [14] presents a simple, graphical approach to model trust. He points out the relationship between trust and risk and argues that for every trust relationship, there exists a risk associated with a breach of the trust extended. Trust relationships are modeled as directed graphs where trust is an unidirectional directed edge from the trusting entity to the trusted entity. The author includes context (to define scope of trust), associated confidence level, associated risk and transitivity value. Bacharach and Gambetta [15] embark on a re-orientation of the theory of trust. They define trust as a particular belief, which arises in games with a certain payoff structure. They also identify the source of the primary trust problem in the uncertainty about the payoffs of the trustee. According to the authors, the trustor must judge whether apparent signs of trustworthiness are themselves to be trusted.

3 Our Model

We adopt the definition of trust as provided by Grandison and Sloman [16].

Definition 1. *Trust is defined to be the firm belief in the competence of an entity to act dependably, reliably and securely within a specific context.*

In the same work, Grandison and Sloman define *distrust* as the "lack of firm belief in the competence of an entity to act dependably, securely and reliably". However, we

believe distrust is somewhat stronger than just "lacking a belief". Grandison and Sloman's definition suggests the possibility of ambivalence in making a decision regarding distrust. We choose to be more precise and thus define distrust as follows.

Definition 2. *Distrust is defined as the firm belief in the incompetence of an entity to act dependably, securely and reliably within a specific context.*

Trust is specified as a trust relationship between a truster – an entity that trusts the target entity – and a trustee – the entity that is trusted. The truster is always an active entity (for example, a human being or a subject). The trustee can either be an active entity or a passive entity (for example, a piece of information or a software). We use the following notation to specify a trust relationship – $(A \xrightarrow{c} B)_t^N$. We call this the *normalized* trust relationship. It specifies A's *normalized* trust on B at a given time t for a particular context c. This relationship is obtained from the simple trust relationship – $(A \xrightarrow{c} B)_t$ – by combining the latter with a normalizing factor. We also introduce a concept called the *value* of a trust relationship. This is denoted by the expression $\mathbf{v}(A \xrightarrow{c} B)_t^N$ and is a number in $[-1,1]$ that is associated with the normalized trust relationship.

3.1 Trust Context

A trust relationship between a truster, A, and a trustee, B, is never absolute [16]. Always, the truster trusts the trustee with respect to its ability to perform a specific action or provide a specific service. For example, an entity A may trust another entity B about the latter's ability to keep a secret. However, this does not mean if A wants a job done efficiently, A will trust B to it. Similarly, if we want to compare two trust values, we just cannot compare two arbitrary trust values. We need to compare the values for trust which serves similar purposes. This leads us to associate a notion of *context* with a trust relationship. We begin by defining the notion of *atomic purpose* of a trust relationship.

Definition 3. *The* atomic purpose *of a trust relationship* $(A \xrightarrow{c} B)_t$ *is one of*

1. **TS-1** *The truster trusts a trustee to* access resources *that the truster controls.*
2. **TS-2** *The truster trusts the trustee to* provide a service *that does not involve access to the truster's resources.*
3. **TS-3** *The truster trusts the trustee to* make decisions *on its behalf.*

The truster may also trust the trustee for some combination of these atomic purposes. For example the truster may trust the trustee to provide a service and make decisions.

Definition 4. *The* purpose *of a trust relationship is defined as follows.*

1. *An atomic purpose is a purpose of a trust relationship.*
2. *The negation of a purpose denoted by "not" purpose, is a purpose.*
3. *Two purposes connected by the operator "and" form a purpose.*
4. *Two purposes connected by the operator "or" form a purpose.*
5. *Nothing else is a purpose.*

We are interested in three *aspects* – dependability, security and reliability – of the trustee. Combining the concepts of trust purposes and trustee aspects, we define the notion of trust *context* as the interrelated conditions in which trust exists or occurs. For example, let a truster, *A*, trust a trustee, *B*'s dependability to provide a service and make a decision. The "dependability to provide a service and make a decision" is considered to be the trust context. Let $\mathcal{S}$ denote the set of trust purposes and $\mathcal{A}$, the set of trustee aspects identified above. Then a trust context is defined as follows.

Definition 5. *The* context, *c(T), of a trust relationship T is defined as a function that takes a trust relationship as an input and returns a sequence of tuples of the form* $< s_1, a_1 > | < s_2,\ a_2 > | \ldots$ *where*

1. $s_i : \mathcal{S} \times \mathcal{S} \rightarrow \mathcal{S}$ *and*
2. $a_i : \mathcal{A} \times \mathcal{A} \rightarrow \mathcal{A}$

3.2 Trust Evaluation

We define a trust value in terms of a vector of numbers. Each element in the trust vector represents a parameter that contributes towards the trust value. Before we formally define these trust parameters, we would like to point to two characteristics of trust (or distrust). The first is the dynamic nature of trust. Trust changes over time. Even if there is no change in the underlying factors that influence trust over a time period, the value of trust at the end of the period is not the same as that at the beginning of the period. Irrespective of our initial trust or distrust decision, over a period of time we gradually become non-decisive or uncertain about the trust decision. This leads us to claim that trust (and alternately distrust) decays over time - both tends towards a non-decisive value over time.

The second characteristic is, what is often called the *propensity* to trust [16]. Given the same set of values for the factors that influence trust, two trusters may come up with two different trust values for the same trustee. We believe that there are two main reasons for this. First, during evaluation of a trust value, a truster may assign different weights to the different factors that influence trust. The weights will depend on the trust evaluation policy of the truster. So if two different trusters assign two different sets of weights, then the resulting trust value will be different. The second reason is applicable only when the truster is a human being and is completely subjective in nature – one person may be more trusting than another. We believe that this latter concept is extremely difficult to model. We choose to disregard this feature in our model and assume that all trusters are trusting to the same extent. We capture the first factor using the concept of a *trust evaluation policy vector*, which is simply a vector of weight values.

We begin by identifying three different parameters that influence trust values.

Definition 6. *The* experience *of a truster about a trustee is defined as the measure of the cumulative effect of a number of events that were encountered by the truster with respect to the trustee in a particular context and over a specified period of time.*

The trust value of a truster on a trustee can change because of the truster's *experiences* with the trustee in the particular context. Each experience that can influence the degree of trust is interpreted by the truster as either a *trust-positive experience* or a *trust-negative experience*. A trust-positive experience contributes towards a gain in trust degree whereas a trust-negative experience contributes towards a loss in trust degree.

Definition 7. *The* knowledge *of the truster regarding a trustee for a particular context is defined as a measure of the condition of awareness of the truster through acquaintance with, familiarty of or understanding of a science, art or technique.*

The trust value of a truster on a trustee can change because of some *knowledge* that the truster comes to posses regarding the trustee for the particular context. Knowledge can be of two types – *direct knowledge* and *indirect knowledge*. Direct knowledge is one which the truster acquires by itself. It may be obtained by the truster in some earlier time for some purpose or, it may be a piece of information about the trustee for which the truster has a concrete proof to be true. Indirect knowledge, on the other hand, is something that the truster does not acquire by itself. The source of indirect knowledge is the *reputation* of the trustee in the context. The truster may get the idea about the reputation of trustee from various sources like reviews, journals, news bulletin, people's opinion etc. As with experience, we can have *trust-positive knowledge* and *trust-negative knowledge*.

Definition 8. *A* recommendation *about a trustee is defined as a measure of the subjective or objective judgment of a recommender about the trustee to the truster.*

The trust value of a truster on a trustee can change because of a *recommendation* for the trustee. We can have a *trust-positive recommendation* and a *trust-negative recommendation*. Moreover, recommendation can be obtained by the truster from more than one source.

To compute a trust relationship we assume that each of these three factors is expressed in terms of a numeric value in the range $[-1,1]$. A -ve value for the component is used to indicate the *trust-negative* type for the component, whereas a +ve value for the component is used to indicate the *trust-positive* type of the component. A 0 (zero) value for the component indicates neither positive effect nore negative effect on the trust value.

3.3 Evaluating Experience

We model experience in terms of the number of events encountered by a truster, A, regarding a trustee, B in the context c within a specified period of time $[t_0,t_n]$. We assume that A has a record of the events since time t_0. An event can be either trust-positive or trust-negative depending whether it contributes towards a trust-positive experience or a trust-negative experience.

Let N denote the set of natural numbers. The set of time instances $\{t_0, t_1, \ldots, t_n\}$ is a totally ordered set ordered by the temporal relation $\prec$ (called the *precedes-in-time* relation) as follows: $\forall i, j \in N,\ t_i \prec t_j \Leftrightarrow i < j$. We use the symbol $t_i \preceq t_j$ to signify either $t_i \prec t_j$ or $t_i = t_j$. Let also e_k denote the kth event. Events happen at time instances. We define the concept *event-occurrence-time* as follows:

Definition 9. Event-occurrence-time ET *is a function that takes an event* e_k *as input and returns the time instance,* t_i *at which the event occurred. Formally,* $ET : e_k \rightarrow t_i$.

We divide the time period [t_0,t_n] over which the events have occurred into a set $\mathcal{T}$ of n intervals, [t_0,t_1], [t_1,t_2], ..., [t_{n-1},t_n] such that for any interval [t_i,t_j], $t_i \prec t_j$. A particular interval, [t_{k-1},t_k], is referred to the kth interval. We extend the $\prec$ relation on $\mathcal{T}$ and the time intervals are also totally ordered by the $\prec$ relation as follows – $\forall i,j,k,l \in N,\ [t_i,t_j] \prec [t_k,t_l] \Leftrightarrow t_j \prec t_k$. Finally, the intervals are non-overlapping, that is, $\forall i,j,k,l \in N,\ [t_i,t_j] \cap [t_k,t_l] = \phi$.

Definition 10. *Let* $\mathcal{E}$ *denote the set of all events. A* sequence of events, C_E, *is the set of events* e_1, e_2, ..., e_n, $e_i \in \mathcal{E}$, *such that* $\forall i,j,\ ET(e_i) \in [t_k,t_l] \Leftrightarrow ET(e_j) \in [t_k,t_l]$ *and such that* $\forall i,j \in N,\ e_i \prec e_j \Leftrightarrow i < j$.

Let P denote the set of all trust-positive events and Q denote the set of all trust-negative events (that is, $\mathcal{E} = \{\mathrm{P} \cup \mathrm{Q}\}$). We assign equal numeric weights to all events, trust-positive or trust-negative, within a given interval. Let v_{k_i} be the weight of the k^{th} event in the i^{th} interval. We assign a weight of +1 if an event is in the set P and -1 if the event is in the set Q. Thus,

$$v_{k_i} = \begin{cases} +1 & \text{, if } e_{k_i} \in \mathrm{P} \\ -1 & \text{, if } e_{k_i} \in \mathrm{Q} \end{cases}$$

Definition 11. *The* incidents I_i, *corresponding to the* i^{th} *time interval is the sum of the values of all the events, trust-positive or trust-negative for the time interval. It is given by* $I_i = \sum_{k=1}^{n_i} v_{k_i}$ *where* n_i *is the number of events occured in the* i^{th} *time interval.*

Typically, events far back in time does not count just as strongly as very recent events. To accomodate this we assign a *non-negative* weight w_i to the i^{th} interval such that $w_i > w_j$ whenever $j < i,\ i,j \in N$. We then define *experience* as follows:

Definition 12. *The* experience *of an entity A about another entity B for a particular context c, is the accumulation of all trust-positive and trust-negative events that A has with regards to B over a given period of time [*t_0,t_n*], scaled to be in the range [-1,1].*

To ensure that the value of experience is within this range [-1,1] we define the weight w_i for the i^{th} interval as

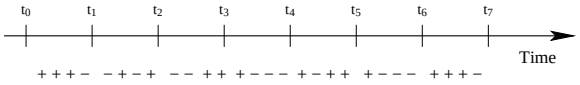
$$w_i = \frac{i}{S} \quad \forall i = 1,2,\ldots,n \text{ where } S = \frac{n(n+1)}{2} \tag{1}$$

Then the experience of A with regards to B for a particular context c is given by

$${}_A\mathrm{E}_B^c = \frac{\sum_{i=1}^{n} w_i I_i}{\sum_{i=1}^{n} n_i} \tag{2}$$

To illustrate our concept of experience we use the following example. We use the symbol "+" to denote positive events and the symbol "-" to denote negative events.

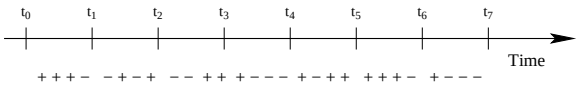
Example 1. Consider the following happening of events over time period $\mathrm{t}_0 - \mathrm{t}_7$.



We divide the time period into the intervals – [t_0,t_1], ... [t_6,t_7]. Applying our theory, we have the following incidents: I_0 for interval [t_0,t_1] = +2, I_1 = 0, I_2 = 0, I_3 = -2, I_4 = +2, I_5 = -2 and I_6 = +2.

The weights assigned to each time interval are as follows – w_0 (for interval [t_0,t_1]) = 0.04, w_1 = 0.07, w_2 = 0.11, w_3 = 0.14, w_4 = 0.18, w_5 = 0.21 and w_6 = 0.25 (for interval [t_6,t_7]. Thus the value for experience over the period [t_0,t_7] is 0.00857.

Example 2. Consider the second set of events over the same time period $t_0 - t_7$.



The difference between this set of events and the one in example 1 is that we have more negative events that have happened recently. The total number of trust-positive and trust-negative events are the same in both. We get a value of 0.00286 for experience with this set of events.

3.4 Evaluating Knowledge

The parameter, knowledge, is more difficult to compute and is, to some extent, subjective. To begin with, each truster must define its own criteria for gradation of knowledge regarding a particular entity. To assign a value to the *knowledge* component, the truster must come up with two values between -1 and +1 for direct knowledge as well as indirect knowledge or reputation. How the values are assigned, depends on the scheme and policy of the truster. Also the truster is solely responsible to assign the relative weights for these two types of knowledge. We represent this as, ${}_AK_B^c = w_1 d + w_2 r$, where $d, r \in [-1, 1]$ and $w_1 + w_2 = 1$. d and r are the values corresponding to direct knowledge and reputation respectively. The weights w_1 and w_2 are determined by the underlying policy where $w_i \in [0, 1] \quad \forall i = 1, 2$.

The truster needs not have values for both the components. That is, there may be situation where either $d = 0$ or $r = 0$. A may not have any knowledge at all about B in the context. For these types of cases, where the truster doesnot have any information regarding the trustee in the context c, we assign ${}_A\mathbf{K}_B^c = 0$ where '0' represents 'no knowledge' about B in context c. This is the situation when both d and r are zero.

3.5 Evaluating Recommendation

An initial recommendation, V_R, is a value in the range [-1,1] that is provided to the truster by the recommender. To assist the recommender in generating this value, the truster provides a questionnaire to the recommender. The recommender uses the positive range to express his faith in the trustee while uses the negative range to express his discontent. If the recommender has no conclusive decision, he uses a zero value for recommendation.

Now, a truster A will often have a trust relationship with the recommender R. The context of this trust relationship will be to act "reliably to provide a service (recommendation, in this case)". This trust relationship will have an effect on the value of the recommendation provided by the recommender. For example, let us say that A trusts R to quite a great extent to provide an appropriate recommendation for B but does not trust C as much as R. R provides a recommendation value of -0.5 to A and C also provides the same recommendation value. To A, R's -0.5 value will have more weightage for computing the trust value on B than C's, although A will consider both the values. To model this scenario we use the trust of the truster on the recommender as a weight factor to the initial recommendation value returned by the recommender. We had introduced the expression $\mathbf{v}(A \xrightarrow{c} B)_t^N$ earlier in section 3 to denote the *value* of a normalized trust relationship. This is a value in the range $[-1,1]$. We use the absolute value of this value as the weight factor. At this stage we do not specify how we generate this value. We leave that to a later section. At this stage we express the *recommendation* ${}_CR_B$ of a recommeder C for an entity B to the truster A as ${}_CR_B = |\mathbf{v}(A \xrightarrow{rec} C)_t^N| V_R$.

Finally, the truster A may get recommendations about the trustee B from many different recommenders not just one. Thus the recommendation value that the truster uses to compute the trust in the trustee is specified as the sum of all recommendations scaled to the range $[-1,1]$. This is given by the equation

$$ {}_\Psi R_B^c = \frac{\sum_{j=1}^{n} |\mathbf{v}(A \xrightarrow{rec} j)_t^N| \cdot V_j}{\sum_{j=1}^{n} |\mathbf{v}(A \xrightarrow{rec} j)_t^N|} \tag{3} $$

where, Ψ is a group of n recommenders.

3.6 Normalization of Trust Vector

Having determined the values for each component of the trust vector we specify the simple trust relationship between the truster A and the trustee B in a context c as $(A \xrightarrow{c} B)_t = [{}_AE_B^c, {}_AK_B^c, {}_\Psi R_B^c]$

As mentioned earlier in section 3.2, a truster may give more weight to one of the parameters than other in computing a trust relationship. For example, a truster A may choose to lay more emphasis on experience than recommendation in computing trust. Or for example, a truster may be quite skeptical regarding recommendations about the trustee. In that case the truster may want to consider the recommendation factor to a lesser extent in computing trust than experience and knowledge about the trustee. Which particular component needs to be emphsized more than the others, is a matter of trust evaluation policy of the truster. The policy is represented by the truster as a trust policy vector.

Definition 13. *The* trust policy vector, **W** *is a vector that has the same dimension as the simple-trust vector. The elements are real numbers in the range* $[0,1]$ *and the sum of all elements is equal to 1.*

The normalized trust relationship between a truster A and a trustee B at a time t and for a particular context c is given by

$$(A \xrightarrow{c} B)_t^N = \mathbf{W} \odot (A \xrightarrow{c} B)_t \tag{4}$$

The $\odot$ operator represents the normalization operator. Let $(A \xrightarrow{c} B)_t = [{}_AE_B^c, {}_AK_B^c, {}_\psi R_B^c]$ be a trust vector such that ${}_AE_B^c,\ {}_AK_B^c,\ {}_\psi R_B^c \in [-1,1]$. Let also $\mathbf{W} = [W_e, W_k, W_r]$ be the corresponding trust policy vector such that $W_e + W_k + W_r = 1$ and $W_e, W_k, W_r \in [0,1]$. The $\odot$ operator generates the normalized trust relationship as

$$\begin{aligned}(A \xrightarrow{c} B)_t^N &= \mathbf{W} \odot (A \xrightarrow{c} B)_t \\ &= [W_e,\ W_k,\ W_r] \odot [{}_AE_B^c,\ {}_AK_B^c,\ {}_\psi R_B^c] \\ &= [W_e \cdot {}_AE_B^c,\ W_k \cdot {}_AE_B^c,\ W_r \cdot {}_\psi R_B^c] \\ &= [{}_A\hat{E}_B^c,\ {}_A\hat{K}_B^c,\ {}_\psi \hat{R}_B^c]\end{aligned}$$

It follows from above that each element ${}_A\hat{E}_B^c,\ {}_A\hat{K}_B^c,\ {}_\psi \hat{R}_B^c$ of the normalized trust vector also lies within $[-1,1]$.

3.7 Value of the Normalized Trust Vector

So far we have defined a trust relationship in terms of a vector which is *normalized* by a trust policy. Recall, however, from section 3.5 that there is at least one scenario in which we need to use a trust value as a weight for a real number (namely recommendation). Thus it seems appropriate to define the concept of a *value* corresponding to the normalized trust vector. Moreover, although we had previously argued against using a single value for trust, there is a big advantage of using a single value. A single value is more intuitive than a vector. In the next section we also show how such a single value helps us in assessing the dynamics of trust.

Definition 14. *The* value *of a normalized trust relationship* $(A \xrightarrow{c} B)_t^N = [{}_A\hat{E}_B^c, {}_A\hat{K}_B^c, {}_\psi \hat{R}_B^c]$ *is a number in the range* $[-1,1]$ *and is defined as*

$$\boldsymbol{v}(A \xrightarrow{c} B)_t^N = {}_A\hat{E}_B^c + {}_A\hat{K}_B^c + {}_\psi \hat{R}_B^c \tag{5}$$

Having defined the value for a trust relationship we revise the terms "trust" and "distrust" as follows:

1. If the value, T, of a normalized trust relationship is such that $0 < T \leq 1$ then it is trust.
2. If the value, T, of a normalized trust relationship is such that $-1 \leq T < 0$ then it is distrust.
3. If the value, T, is 0 then it is neither trust nor distrust.

3.8 Trust Dynamics

Trust (and distrust) changes over time. Let us suppose that we have initially computed a trust relationship T_{t_i} at time t_i, based on the values of the underlying parameters at that time. Suppose now that we try to recompute the trust relationship T_{t_n} at time t_n.

We claim that even if the underlying parameters do not change between times t_i and t_n, the trust relationship will change. This change of trust over time is often called *trust dynamics*.

To model trust dynamics we refer to the old adage – Time the great healer. The general tendency is to forget about past happenings. This leads us to claim that trust (and distrust) tends towards neutrality as time increases. Initially, the value does not change much; after a certain period the change is more rapid; finally the change becomes more stable as the value approaches the neutral (value = 0) level. Also we assert the following:

$$\lim_{t\to\infty} \mathbf{v}(T_t) = 0$$

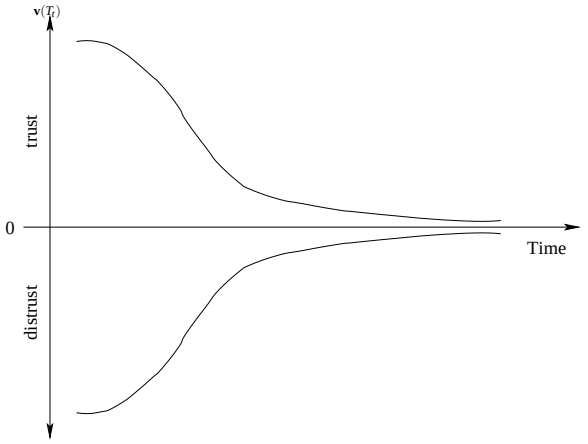


Fig. 1. Graph Showing the Nature of Trust Dynamics

How fast trust (or distrust) will decay over time, is, we believe, dependent on the truster's policy. The truster may choose to forget about trust relationships which are 3 years old or 5 years old. The model cannot dictate this. Our goal is to provide a basis by which the truster can at least estimate, based on the truster's individual perception about this, the trust at time t_n. We further believe that trust relationship at present time is not only dependent on the values of the underlying parameters, but also on the "decayed" value of the previous trust. We discuss this in more details in the next section.

Let $\mathbf{v}(T_{t_i})$, be the value of a trust relationship, T_{t_i}, at time t_i and $\mathbf{v}(T_{t_n})$ be the decayed value of the same at time t_n. Then the *time-dependent value* of T_{t_i} is defined as follows.

Definition 15. *The* time-dependent value *of a trust relationship T_{t_i} from time t_i, computed at present time t_n, is given by*

$$\boldsymbol{v}(T_{t_n}) = \mathbf{v}(T_{t_i}) e^{-(\boldsymbol{v}(T_{t_i})\Delta t)^{2k}} \tag{6}$$

where $\Delta t = t_n - t_i$ and k is any small integer ≥ 1.

The value of k determines the rate of change of trust with time and is assigned by the truster based on its perception about the change. If $\Delta t = 0$ that is at $t_n = t_i$, $e^{-(\mathbf{v}(T_{t_i})\Delta t)^{2k}} = 1$ and hence $\mathbf{v}(T_{t_n}) = \mathbf{v}(T_{t_i})$. When $\Delta t \to \infty$, then $e^{-(T_i \Delta t)^{2k}} \to 0$ and hence $\mathbf{v}(T_{t_n}) \to 0$. This corroborates the fact the time-dependent value of the last known trust value is asymptotic to zero at infinite time.

To obtain the trust vector T_{t_n} at time t_n, we distribute the value $\mathbf{v}(T_{t_n})$ obtained in equation (6) evenly over the components. The rational behind this is that between t_i and t_n we do not have sufficient information to assign different weights to the different components. Thus we have the time-dependent vector as

$$T_{t_n} = [\frac{\mathbf{v}(T_{t_n})}{3}, \frac{\mathbf{v}(T_{t_n})}{3}, \frac{\mathbf{v}(T_{t_n})}{3}]$$

3.9 Trust Vector at Present Time

As indicated earlier, the trust of a truster A on a trustee B in a context c at time t_n depends not only on the underlying components of the trust vector but also on the trust established earlier at time t_i. Consider for example that at time t_i Alice trusts Bob to the fullest extent (value = 1). At time t_n Alice re-evaluates the trust relationship and determines the value to be -0.5 (distrust). However, we believe that Alice will lay some importance to the previous trust value and will not distrust Bob as much as a -0.5 value. So, the normalized trust vector at t_n is a linear combination of time-dependent trust vector and the normalized trust vector calculated at present time. The weight Alice will give to old trust vector and present normalized trust vector is, again, a matter of policy. However, this leads us to refine the expression for normalized trust vector at time t_n as follows. Let $\hat{T}$ be the time-dependent trust vector derived from $\mathbf{v}(T_{t_i})$ at time t_n. Also, let α and β are the weights corresponding to present normalized vector and time-dependent vector, respectively.

Definition 16. *The normalized trust relationship between a truster A and a trustee B at time t_n in a particular context c is given by*

$$(A \xrightarrow{c} B)^N_{t_n} = \begin{cases} [{}_A\hat{E}^c_B,\ {}_A\hat{K}^c_B,\ {}_\psi\hat{R}^c_B] \\ \qquad\qquad \textit{if } t_n = 0 \\ [\frac{\mathbf{v}(\hat{T})}{3},\ \frac{\mathbf{v}(\hat{T})}{3},\ \frac{\mathbf{v}(\hat{T})}{3}] \\ \qquad\qquad \textit{if } t_n \neq 0 \textit{ and } {}_A\hat{E}^c_B = {}_A\hat{K}^c_B = {}_\psi\hat{R}^c_B = 0 \\ [{}_A\hat{E}^c_B,\ {}_A\hat{K}^c_B,\ {}_\psi\hat{R}^c_B] \oplus [\frac{\mathbf{v}(\hat{T})}{3},\ \frac{\mathbf{v}(\hat{T})}{3},\ \frac{\mathbf{v}(\hat{T})}{3}] \\ \qquad\qquad \textit{if } t_n \neq 0 \textit{ and at least one of} \\ \qquad\qquad {}_A\hat{E}^c_B, {}_A\hat{K}^c_B, {}_\psi\hat{R}^c_B \neq 0 \end{cases} \tag{7}$$

The $\oplus$ operator is defined as follows.

$$\begin{aligned} [{}_A\hat{E}^c_B, {}_A\hat{K}^c_B, {}_\psi\hat{R}^c_B] \oplus [\frac{\mathbf{v}(\hat{T})}{3}, \frac{\mathbf{v}(\hat{T})}{3}, \frac{\mathbf{v}(\hat{T})}{3}] &= \alpha \cdot [{}_A\hat{E}^c_B, {}_A\hat{K}^c_B, {}_\psi\hat{R}^c_B] + \beta \cdot [\frac{\mathbf{v}(\hat{T})}{3}, \frac{\mathbf{v}(\hat{T})}{3}, \frac{\mathbf{v}(\hat{T})}{3}] \\ &= [\alpha \cdot {}_A\hat{E}^c_B + \beta \cdot \frac{\mathbf{v}(\hat{T})}{3},\ \alpha \cdot {}_A\hat{K}^c_B + \beta \cdot \frac{\mathbf{v}(\hat{T})}{3},\ \alpha \cdot {}_\psi\hat{R}^c_B + \beta \cdot \frac{\mathbf{v}(\hat{T})}{3}] \end{aligned}$$

where $\alpha, \beta \in [0,1]$ and $\alpha + \beta = 1$.

4 Comparison Operation on Trust Vectors

In many real life scenarios we need to determine the relative trustworthiness of two trustees. Consider the following example. Suppose entity A gets two conflicting pieces of information from two different sources B and C. In this case A will probably want to compare its trust relationships with entities B and C and accept the information that originated from the "more" trustworthy entity. This lead us to define a comparison operator on trust relationships.

Let $T = (A \xrightarrow{\dot{c}} B)_t^N$ and $T' = (A \xrightarrow{\ddot{c}} C)_t^N$ be two normalized trust relationships – between A and B, and between A and C respectively – at a particular time t. We have the following definition.

Definition 17. *Two trust relationships, T and T' are said to be* compatible *if the trust relationships have been defined under the same policy vector and the context $c(T)$ for the trust relationship T is the same as the context $c(T')$ for T', that is $c(T) = c(T')$. Otherwise the two trust relationships are called* incompatible.

Note that to determine if two trust relationships are compatible or not we do not make any assumptions about the truster and the trustee involved in the relationships nor about the time instances of the relationships. In order to be able to compare the two trust relationships T and T' from above it has to be the case that the two contexts $\dot{c}$ and $\ddot{c}$ are the same.

The most intuitive way to compare two trust relationships T and T' is to compare the values of the trust relationships in a numerical manner. Thus for A to determine the relative levels of trustworthiness of B and C, A evaluates $\mathbf{v}(A \xrightarrow{c} B)_t^N$ and $\mathbf{v}(A \xrightarrow{c} C)_t^N$. If $\mathbf{v}(A \xrightarrow{c} B)_t^N > \mathbf{v}(A \xrightarrow{c} C)_t^N$, then A trust B more than C in the context c. We say that T *dominates* T', given by $T \succ T'$.

However, if $\mathbf{v}(A \xrightarrow{c} B)_t^N = \mathbf{v}(A \xrightarrow{c} C)_t^N$, A cannot judge the relative trustworthiness of B and C. This is because there can be two vectors whose individual component values are different but their scalar values are the same. For such cases we need to compare the individual elements of the two trust relationships to determine the relative degree of trustworthiness.

Let $(A \xrightarrow{c} B)_t^N = [{}_A\hat{E}_B^c,\ {}_A\hat{K}_B^c,\ {}_\psi\hat{R}_B^c]$ and $(A \xrightarrow{c} C)_t^N = [{}_A\hat{E}_C^c,\ {}_A\hat{K}_C^c,\ {}_\psi\hat{R}_C^c]$ such that $\mathbf{v}(A \xrightarrow{c} B)_t^N = \mathbf{v}(A \xrightarrow{c} C)_t^N$. Let also the underlying trust policy vector be given by $W = (w_1,\ w_2,\ w_3)$ where $w_1 + w_2 + w_3 = 1$ and $w_i \geq 0\ \forall i = 1,2,3$. To determine the dominance relation between T and T' we first determine the *ordered* trust relationships $\bar{T}$ corresponding to T.

Definition 18. *The* ordered *trust relationship $\bar{T}$ is generated from a trust relationship T as follows:*

1. *Order the w_i's in the trust policy vector corresponding to T in descending order of magnitude.*

2. *Sort the components of the trust vector T according to the corresponding weight components.*

We compare the two ordered trust relationships $\bar{T}$ and $\bar{T'}$, corresponding to T and T', componentwise to determine the dominance relation between the two. Note that we assume that the same underlying trust policy vector has been used to determine the trust relationships. If the first component of $\bar{T}$ is numerically greater than the first component of $\bar{T'}$ then $T \succ T'$. Else if the first components are equal then compare the second components. If the second component of $\bar{T}$ is greater than the second component of $\bar{T'}$ then $T \succ T'$, and so on. If we cannot conclude a dominance relation between the two trust relationship, then we say that the two trust relationships are *incomparable*. This is formalized by the following definition.

Definition 19. *Let T and T' be two trust relationships and $\bar{T}$ and $\bar{T'}$ be the corresponding ordered trust relationships. Let also $\bar{T_i}$ and $\bar{T_i'}$ represent the i^{th} component of each ordered trust relationships and w_i represent the i^{th} weight component in the corresponding trust policy vector. T is said to dominate T' if any one of the following holds.*

1. $v(T) > v(T')$*; or*
2. *if* $\forall\, i, j,\ i \neq j,\ (w_i = w_j)$ *then* $\forall\, i,\ \bar{T_i} > \bar{T_i'}$*; or*
3. *if* $\exists\, i, \bar{T_i} > \bar{T_i'}$ *and for* $k = 0 \ldots (i-1),\ \bar{T_{i-k}} \not\prec \bar{T'_{i-k}}$

Otherwise T is said to be incomparable *with T'.*

5 Conclusions and Future Work

In this paper we introduce a new model of trust which we term the vector model. Trust is specified as a trust relationship between a truster and a trustee at a particular time instance and for a particular context. We identify three parameters namely, experience, knowledge and recommendation that contribute towards defining this trust relationship. We propose expression for evaluating these factors. Next we introduce the concept of normalized trust. We show how to factor in a notion of trust policy in computing the trust vector. We also model the notion of trust dynamics, that is the change of trust with time. Incorporating all these different notions we finally provide an expression to compute a trust vector that also includes the effect of a previous trust relationship between the same truster, trustee in the same context. We also define ways by which we can compare two trust vectors.

To our knowledge our model is the first to (1) formally differentiate between trust and distrust, (2) address explicitly the contributions of different factors towards formation of a trust relationship, (3) explore and formalize the dynamic nature of trust and (4) address the influence of a previous trust relationship in computing the current trust relationship. A novel feature of our model is that it is easily adaptable if the underlying parameters are changed to include more than the current three parameters (the parameters all need to be orthogonal to each other.).

A lot of work remains to be done. We are currently extending this model to define trust combination operators so that we can formulate the trust relationships between

many trusters and many trustees beginning with simple trust relationships between one truster and one trustee as in this work. We also plan to formalize the notion of trust chains in the context of our model. In the current work we have not addressed the issue of determining trust policy. We have assumed that there is an underlying trust policy that helps us assign weights to the various components of the model. How to assign these weights? What will be an appropriate guideline for that? These are some of the issues we will address in future. This will be followed by a formal language to manage and manipulate trust relationships. We are looking towards an SQL like language for this purpose. Finally we plan to develop a complete trust management framework based on our model.

Acknowledgment

This work was partially supported by the U.S. Air Force Research Laboratory (AFRL) and the Federal Aviation Administration (FAA) under contract F30602-03-1-0101. The views presented here are solely those of the authors and do not necessarily represent those of the AFRL or the FAA.

References

[1] Burrows, M., Abadi, M., Needham, R.: A logic of authentication. ACM Transactions on Computer Systems **8** (1990) 18–36

[2] Jajodia, S., Samarati, P., Subrahmanian, V.: A logical language for expressing authorizations. In: Proceedings of the 1997 IEEE Symposium on Security and Privacy, Oakland, California, USA, IEEE Computer Society (1997) 31–42

[3] Abdul-Rahman, A., Hailes, S.: Supporting trust in virtual communities. In: Proceedings of the 33rd Annual Hawaii International Conference on System Sciences (HICSS-33), Maui, Hawaii, USA, IEEE Computer Society (2000) 1769–1777

[4] Rangan, P.: An axiomatic basis of trust in distributed systems. In: Proceedings of the 1988 IEEE Computer Society Symposium on Security and Privacy, Oakland, California, USA, IEEE Computer Society (1988) 204–211

[5] Jones, A., Firozabadi, B.: On the characterization of a trusting agent – aspects of a formal approach. In C.Castelfranchi, Y.Tan, eds.: Trust and Deception in Virtual Societies. Kluwer Academic Publishers (2000) 163–174

[6] Yahalom, R., Klein, B., Beth, T.: Trust relationship in secure systems: A distributed authentication perspective. In: Proceedings of the IEEE Computer Society Symposium on Security and Privacy, Oakland, California, USA, IEEE Computer Society (1993) 150–164

[7] Yahalom, R., Klein, B.: Trust-based navigation in distributed systems. Computing Systems **7** (1994) 45–73

[8] Beth, T., Borcherding, M., Klein, B.: Valuation of trust in open networks. In Gollmann, D., ed.: Proceedings of the 3rd European Symposium on Research in Computer Security - ESORICS '94. Volume 875 of Lecture Notes in Computer Science., Brighton, UK, Springer-Verlag (1994) 3–18

[9] Jøsang, A.: Artificial reasoning with subjective logic. In: Proceedings of the Second Australian Workshop on Commonsense Reasoning, Perth, Australia (1997)

[10] Jøsang, A.: A subjective metric of authentication. In Quisquater, J.J., et al., eds.: Proceedings of the 5th European Symposium on Research in Computer Security (ESORICS'98). Volume 1485 of Lecture Notes in Computer Science., Louvain-la-Neuve, Belgium, Springer-Verlag (1998) 329–344
[11] Jøsang, A.: An algebra for assessing trust in certification chains. In: Proceedings of Network and Distributed Systems Security Symposium(NDSS'99), San Diego, California, USA, Internet Society (1999)
[12] Cohen, M., Parasuraman, R., Serfaty, R., Andes, R.: Trust in decision aids:a model and a training strategy. Technical Report USAATCOM TR 97-D-4, Cognitive Technologies Inc., Fort Eustis, Virginia, USA (1997)
[13] Li, L.X., Liu, L.: A reputation-based trust model for peer-to-peer ecommerce communities. In: Proceedings of IEEE Conference on E-Commerce (CEC'03), Newport Beach, California, USA, IEEE Computer Society (2003) 275–284
[14] Purser, S.: A simple graphical tool for modelling trust. Computers & Security **20** (2001) 479–484
[15] Bacharach, M., Gambetta, D.: Trust as type identification. In Castelfranchi, C., Tan, Y., eds.: Trust and Deception in Virtual Societies. Kluwer Academic Publishers (2000) 1–26
[16] Grandison, T., Sloman, M.: A survey of trust in internet applications. IEEE Communications Surveys and Tutorials **3** (2000) 2–16