Establishing Trust Beliefs based on a Uniform Disposition to Trust

Rachid Saadi
Email: rachid.saadi@liris.cnrs.fr
LIRIS, INSA de Lyon, France

Omar Hasan
Email: omar.hasan@insa-lyon.fr
LIRIS, INSA de Lyon, France

Jean Marc Pierson
Email: jean-marc.pierson@irit.fr
IRIT, University Paul Sabatier Toulouse, France

Lionel Brunie
Email: lionel.brunie@liris.cnrs.fr
LIRIS, INSA de Lyon, France

Abstract

Trust based access control models have recently attracted significant interest in the area of pervasive computing. In several trust based models, organizations are required to establish a set of trust beliefs regarding their neighboring nodes. In an organization there may be multiple administrators or the administrators may change with time. When multiple administrators contribute to the creation of the set of trust beliefs of an organization, inconsistencies may occur due to variations in their disposition to trust.

In this paper we introduce a novel method for establishing the set of trust beliefs, which is likely to be more consistent. The proposed solution attempts to tie the quantification of trust not to the multiple dispositions to trust of the administrators but to a uniform disposition to trust of the organization. Having administrators evaluate the trustworthiness of neighboring nodes in relation to other nodes and using a mapping function for assigning quantitative values are the highlights of the method.

1. Introduction

Trust based access control models are of particular interest in pervasive computing where perpetual reachability of central certificate authorities cannot be assumed. Moreover, in pervasive computing mobile entities are intended to roam security domains other than their own which are not always pre-configured to recognize their identities. Security solutions based on trust offer to alleviate these issues.

Almenarez et al's TrustAC [3] and Saadi et al's Chameleon [22] are some of the several access control models that have been proposed for pervasive environments. Most of these models augment established access control approaches with trust elements.

We observe that in several of the trust based models for pervasive environments (including, TrustAC [3] and Chameleon [22]), trust propagation is often employed to indirectly compute the trustworthiness of previously unknown nodes. However, the user is required to directly evaluate the trustworthiness of a set of known nodes. Either a numerical range such as 0 to 9 or a strata of labels, for example, high trust, medium trust, low trust, is used for the assignment of trust.

We note that the assignment of trust using either of these approaches is highly subjective to the personality or the "disposition to trust" of the user. Alice may perceive the trustworthiness of Cathy as 5, however based on similar experiences, Bob may evaluate Cathy's trustworthiness as 8. The difference in these trust beliefs occurs due to the difference in the disposition to trust of Alice and Bob. We use the term "trust belief" as a statement such as "Alice trusts Cathy as 5 on a scale of 0 to 9".

Let's consider the scenario where a node is not an individual user but an organization comprising of a number of users. The node represents a single trust domain with its members subscribing to a common set of trust beliefs for foreign nodes in the pervasive environment. This scenario occurs in the Chameleon architecture as well as others. For such an organization it cannot be assumed that the evaluation of trustworthiness will always be performed by the same person. The organization may simultaneously have several administrators or the administrators may change with time. The varying perceptions of human administrators regarding the trustworthiness of foreign nodes can result in a set of trust beliefs that lacks consistency. A case in point is Alice and Bob being two administrators in charge of creating trust beliefs for the same organization.

Assuming that Alice was the one who evaluated Cathy, and thus a trustworthiness rating of 5 was associated with her. If for a certain transaction the trust threshold is given

as 6, Cathy would fail the trustworthiness criteria. Had Bob evaluated Cathy and assigned her the value 8, the result would have been quite opposite. We can imagine the inconsistencies when some of the trust beliefs in the same set are created by Alice and others by Bob.

In this paper we propose a method for establishing trust beliefs based on a uniform disposition to trust. In summary the idea is to evaluate the foreign nodes not individually but in relation to other nodes. If Alice rates Cathy as more trustworthy than David, then based on similar experiences, Bob is also very likely to rate Cathy more trustworthy than David. Having ordered the nodes in terms of trustworthiness we algorithmically generate their trust values.

section II briefly introduces our access control architecture for pervasive environments called Chameleon [22]. In section III we present the method for establishing trust beliefs based on a uniform disposition to trust. section IV is a general review of literature pertaining to trust in pervasive computing. section V comprises of a discussion on the method and the conclusion.

In this paper we use the terms organization, site and node interchangeably.

2. Chameleon: An Access Control Model for Pervasive Environments

One of the primary characteristics of a pervasive environment is to allow users to roam ubiquitously between disparate administrative domains. The issue is how a local site can authenticate and allow access to previously unknown foreign users.

Our Chameleon architecture works as a front-end for each site and controls access to it by foreign users. When a foreign user approaches a site, the Chameleon system upon authenticating the user, transforms them into a local user and grants them access. The architecture is named after the animal chameleon which has the ability to transform itself to fit into its environment.

To set up the Chameleon architecture we define three modules:

Credential Manager Module (CMM): used to authenticate a user. ***Trust Manager Module (TMM):*** used for interaction between trusted sites. ***Mapping Access Module (MAM):*** maps the profile of a foreign user to a local profile.

Our architecture allows a user to authenticate on a remote site and to grant them access to the site without them being locally recognized in advance. The architecture is a trust-based access control model that uses the dynamic certification mechanism called "X316: Morph Access Pass Certificate" [23].

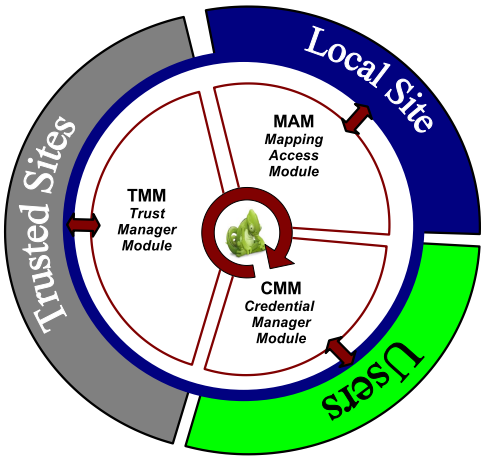

Figure 1. Chameleon Architecture

2.1. Selecting a certificate

The X316 works as a pass, allowing its owner to roam among different sites. Each site issues all its members a Home certificate: H316 that contains the member's local profile and rights. A target site can authenticate the user and attribute them a Trust certificate: T316, if they are approved as trusted.

When a user arrives at a target site, the user's device selects and transmits a valid credential depending on the identity of the target site (hospital, university, airport etc.). The target receives the credential and its Credential Manager Module (CMM) identifies the certificate owner by selecting an authentication process such as challenge response, biometric etc.

2.2. Evaluating the trustworthiness of a user

Once the user is authenticated, the target site attempts to assign them a profile based on the identity of the certificate issuer. We define a trust model to enable organizations to communicate and share certain information about their members.

- ***Trust relation:*** a trust relation is defined as the means for a site to evaluate the trustworthiness of neighbouring sites.

  Let S denote a set of sites. Let A and B be two sites, $A \in S, B \in S$. If A trusts B then we say that the relation Trust exists between A and B and we note *"A Trust B"*.

  Properties:

- ***Reflexivity:*** $\forall A \in S, A\ Trust\ A$
  Trivially, a site trusts itself.
- ***Non-Symmetry:*** The Trust relation is not symmetric. A site is fully responsible for its trust policy and there is no obligation of reciprocity, so we can have
  $A\ Trust\ B \wedge \neg B\ Trust\ A$
- ***Transitivity:*** The Trust relation is transitive: $\forall A, B, C \in S, A\ Trust\ B \wedge B\ Trust\ C \Rightarrow A\ Trust\ C$. This property is fundamental for the effectiveness of our proposition. It allows defining "trust chains" between sites that do not know each other. We however note that a trust chain is not allowed to grow to arbitrary lengths but is limited by the result of the trust function which is discussed below.

- ***Trust evaluation:*** Based on the *Trust* relation, we introduce the trust function t^0, to estimate the level of trust between two sites.

  Properties

  - ***Self trust:*** $\forall A \in S, t^0(A, A) = 0$
  - ***Non-commutativity:*** $\exists A, B \in S / t^0(A, B) = d_1 \wedge t^0(B, A) = d_2 \wedge d_1 \neq d_2$
  - ***Composition*** Let A, B, C be three sites. The *composition* of the trust degrees $t^0(A, B)$ and $t^0(B, C)$, noted $t^0(A, C) = t^0(A, B) \oplus t^0(B, C)$. According to the result of the trust function, the target site is able to accept or reject the trust chain.

Each site administrator builds its local **trust set** which contains all the trusted sites and assigns each one a numerical trust value between 0 and T^0, where T^0 represents the local trust threshold. In section 3 we present our new method to replace this original manual approach. The environment can be seen as a Trust graph noted as $T_g(S, E)$, a valued and directed graph such that:

- The nodes of the graph represent the sites of S.
- Each Trust relation between two sites is represented by a directed edge e. The set of edges is consequently identified with the set of relations, E.
- Each edge is valued by the trust degree between the sites represented by the source and destination nodes of this edge (use of the t^0 function).

The trust function is discussed in detail in [21].

When a user arrives at a target site, the Trust Management Module (TMM) searches the graph (by asking its trusted sites) to locate the user's home site "H". Once H is located, a trust chain is created between the target site and the home site.

The evaluation of this path allows the target site to decide if the foreign user can be allowed to access the target site resources (for example, to decide if a user having no account on the home site can get access). We consider two kinds of access: direct access and transitive access.

- ***A direct access*** is provided by a target site to all users registered by its trusted sites (the sites in its trust set). A direct access is given if the foreign user is a member of the target site's trust set.

- ***A transitive access*** If direct access cannot be granted, the target site tries to locate the home site of the user through the sites in its trust set. Transitive access can be provided by a target site to a user who does not belong to sites in its trusted set on the condition that there exists a trust chain between the user's home site and the target site. The trust function value for the trust chain must be positive. In case of existence of several possible chains, the target site is responsible for selecting one of the chains.

This model, using community collaboration, enables the target site to evaluate the nomadic user in relation to their home site.

2.3. Attributing an access profile

Attributing an access profile to a foreign user requires us to first define two constructs: Analogous profile and Mapping policy.

Once a user is allowed to access a target site, the MAM attributes with them an analogous profile using a mapping policy. Each site defines some analogous profiles (local profiles), which can be attributed to trusted foreign users. The mapping policy is implemented to correspond the home profile of a foreign user to an analogous one. Each site creates a mapping table that enables matching between the different profiles of trusted sites and its own analogous profiles. For example, a user Bob, having an access profile with level 5 in his home site C, wants to access a site B, which provides Bob a new access level, for instance, level 3.

3. Establishing Trust Beliefs based on a Uniform Disposition to Trust

A local site's trust set is composed of the sites that it can evaluate directly for their trustworthiness. In other words

the members of the set are those sites with whom the local site has a direct trust relationship.

We introduce a novel method for the evaluation of trustworthiness of sites in the trust set. The method comprises of the following three steps:

1. Specify the disposition to trust of the local site.
2. Trust sort.
3. Generate the quantitative evaluations of each of the trust set members.

The objective of this method is to allow creation of trust beliefs that are based not on the disposition to trust of individual administrators but on a uniform disposition to trust of the home organization. The result is a set of trust beliefs that are neutral to the disposition to trust of multiple administrators that contribute to its creation. The trust beliefs are however consistent with a uniform disposition to trust defined for the home site.

3.1. Specification of disposition to trust

Disposition to trust is the inherent propensity of an individual to trust or distrust others. An individual's disposition to trust does not vary for specific entities but is a stable characteristic of their personality that governs how they view the trustworthiness of every other entity that they encounter. McKnight et al [18] define disposition to trust as the "extent to which a person displays a tendency to be willing to depend on others across a broad spectrum of situations and persons".

Although disposition to trust has been discussed in the literature as the characteristic of an individual, for our purpose we propose its definition as the characteristic of an organization. We define a variable "d" that represents the disposition to trust of an organization. "d" may be a variable on a range such as 0 to 9 with 0 representing high disposition to trust and 9 representing low disposition to trust. Low disposition to trust indicates that an individual or in our case an organization is less willing to trust a foreign entity and vice versa. The value of "d" may be selected after consensus between all the administrators in the organization.

3.2. Trust sort

Instead of assigning trust values to individual nodes, we propose that an administrator perform trust evaluations in relation to other nodes. We reiterate the example discussed in the introduction to demonstrate the advantage of this approach.

We noted that on a range of 0 to 9, Alice may perceive the trustworthiness of Cathy as 5. Whereas based on similar experiences, Bob may evaluate Cathy's trustworthiness as 8. This difference occurs due to the difference in the disposition to trust of Alice and Bob.

However, if the administrators are required to evaluate the trustworthiness of nodes in relation to other nodes we may have the following scenario. Let's say that Alice rates Cathy as more trustworthy than David. Based on similar experiences with Cathy and David, Bob is also very likely to rate Cathy more trustworthy than David. We make the hypothesis that with this alternate approach we are more likely to have more consistent trustworthiness evaluations.

We call the notion of evaluating nodes in relation to other nodes as "Trust Sort". An administrator is in effect sorting the foreign nodes in terms of their trustworthiness. The product is a sorted list of nodes.

3.3. Generation of quantitative trust values

3.3.1 A classification of sites

We can broadly classify sites into two categories based on their disposition to trust. The first category represents sites that generally exhibit high levels of trust in the members of their trust set. In contrast, the second category represents the sites that are inclined towards low levels of trust in the members of their trust set.

We define a mathematical function $y = f_d(x)$ that we call the BV (**B**eha**V**ior) function. The function represents a curve in the Cartesian coordinate system.

- The input 'x' is a positive integer that represents the order number of a node in the sorted list. The list is numbered from 1 to n where n is the total number of nodes in the list. The node in position 1 is the most trusted node.
- The output 'y' represents the corresponding quantitative trust value for the node based on the disposition to trust of the local site.

We note that in our model we consider zero as the maximum trust value.

We now present the contrast between sites that exhibit trustful and distrustful disposition to trust or behavior in terms of the BV function.

1. **Class 1 "Sites that exhibit Trustful Behavior":** This class represents the behavior of sites which are more trusting. We define that this characteristic is represented by the BV function when it takes a hyperbola form. As illustrated in the figure 2, the projections of the x values are gathered closer to the maximum trust value (zero).
2. **Class 2 "Sites that exhibit Distrustful Behavior":** This class represents the behavior of sites which are

less trusting. We define that this characteristic is represented by the BV function when it takes a parabola form. As illustrated in the figure 2 the projections of the x values are gathered closer to the minimum trust value.

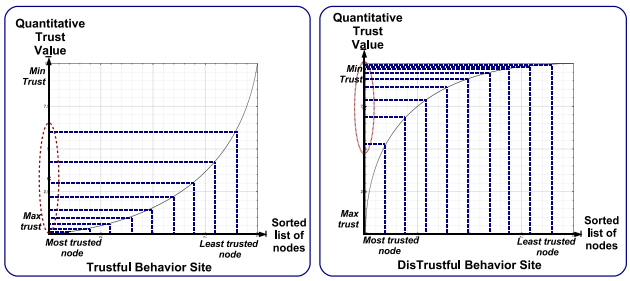

Figure 2. The Trust Behavior

3.3.2 The behavior function

We use a Bezier curve to implement the BV function due to the flexibility it allows in plotting geometric curves.

The Bezier Curve is a parametric form to draw a smooth curve. It is fulfilled through some points $P_0, P_1 ... P_n$, starting at P_0 going towards $P_1 ... P_{n-1}$ and terminating at P_n(see figure 3).

In our model we will use a Bezier curve with three points, which is called a Quadratic Bezier curve. It is defined as follows:

A quadratic Bezier curve is the path traced by the function B(t), given points $P_0, P_1, and P_2$.

$B(t) = (1-t)^2 * P_0 + 2t(1-t) * P_1 + t^2 * P_2.$

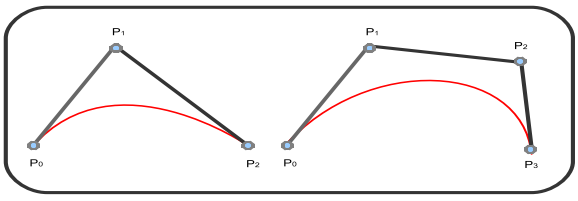

Figure 3. The Bezier curve

The BV function is expressed by a Bezier curve that passes through three points:

- The origin point ($P_0(0,0)$).
- The behavior point ($P_1(b_x, b_y)$)
- The threshold point ($P_2(h_x, h_y)$) where h_x represents the number of sorted site and h_y represents the trust threshold.

As illustrated in figure 4, by moving the behavior point P_1 inside the rectangle that is defined by P_0 and P_2, we are able to adjust the curvature.

Based on the Bezier curve, let us now define the "BV function".

The BV function describes the trust behavior of a site. It takes the order number of a node in the sorted list: **"x"** and returns the corresponding "Quantitative trust value": **"y"**. To apply the BV function with the Bezier curve, we modify the Bezier curve to obtain the output 'y' as a function of 'x', instead of taking a temporal variable 't' as input to compute 'x' and 'y',

The BV function curve can be drawn through the three points $P_0(0,0)$, $P_1(b_x, b_y)$ and $P_2(h_x, h_y)$ using the Bezier curve as follows:

$$
\begin{array}{rcl}
BV : [0, h_x] & \longrightarrow & [0, h_y] \\
X & \longrightarrow & Y
\end{array}
$$

$$
BV_{P_1,P_2}(X) =
$$

$$
\ldots\ldots \begin{cases} Y = (h_y - 2b_y)(\propto (X))^2 + 2b_y \propto (X), & if\ (h_x \neq 2b_x); \\ Y = \frac{h_y}{h_x} x, & otherwise. \end{cases}
$$

$$
Where\ \propto (X) = \frac{-b_x + \sqrt{{b_x}^2 - 2b_x * X + h_x * X}}{h_x - 2b_x} \wedge \begin{cases} 0 < X < h_x \\ 0 < b_x < h_x \\ 0 < b_y < h_y \end{cases}
$$

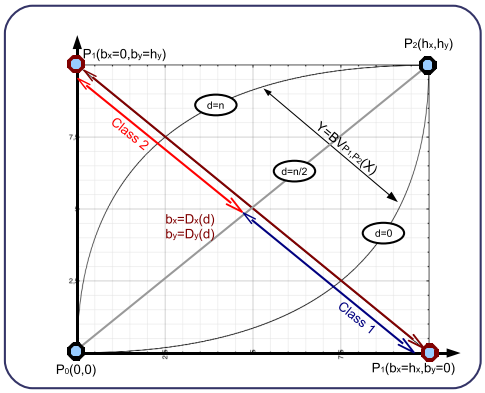

Figure 4. The Behavior curve functions

3.3.3 The Disposition to Trust function D

As discussed earlier, the disposition to trust d of a site is given on a range. We now define a function called the D function which operates on the behavior point P_1 to control the curvature of the BV function.

The D function operates on the point P_1. According to the position of the point P_1 the Bezier curve will transition between parabola and hyperbola forms. As illustrated in figure 4 the first and the last points delimit the curve frame. This frame is a rectangle and it is defined by source point $P_0(0,0)$ and the threshold point $P_2(h_x, h_y)$. The middle point $P_1(b_x, b_y)$ controls site behavior. We assume that this point can move through the second diagonal of the defined

rectangle $b_x = \frac{-h_y}{h_x} * b_y + h_y$. We define the Disposition to Trust function "D" as follows, such that scrolling d between 0 and n gives updated positions for P_1:

$$\begin{array}{rcl} D : [0, n] & \longrightarrow & [0, h_x] * [0, h_y] \\ d & \longrightarrow & (b_x, b_y) \end{array}$$

$$D(d) = \begin{cases} b_x = \frac{-h_x}{n} d + h_x \\ b_y = \frac{h_y}{n} d \end{cases}$$

The variable d represents the disposition to trust of a site. The value 0 indicates maximum trustful behavior and n represents maximum distrustful behavior.

3.4. Generating quantitative trust values

Given d and the threshold points (P_1, P_2), the BV function is able to assign each site in the sorted list a corresponding quantitative trust value as follows:

1. Specifying the P_1 is fulfilled by selecting the corresponding disposition to trust d between 0 and n.

2. The P_2 point is specified by assigning h_x and h_y the following values:

   - h_x=(Number of trusted sites) + 1
   - h_y=T^0(the trust threshold).

3. Putting the trusted sites as classified along the abscissa of the BV function.

Example: Let's consider two sites, where the disposition to trust of each one (the point P_1) is bounded between 0 (very trustful) and 9 (very distrustful);
- S_1: Trustful site, d=1;
- S_2: Distrustful site, d=8.

These sites evaluate five trusted sites (A,B,C,D,E). The threshold point P_2 has the coordinates:
$h_x = 5 + 1 = 6 \; and \; h_y = T^0 = 50$.

- The sorted list of both S_1 and S_2 is:
  (high trust)(+). **SiteD. SiteC. SiteE. Site A. SiteB.** (-)(low trust)

- As illustrated in the figure 5, by performing the BV function the values assigned to the trusted sites would be as follows:

	Site A	Site B	Site C	Site D	Site E
Site S1	13,9	24,2	3,9	1,5	7,8
Site S2	46,1	48,6	36,1	25,8	42,2

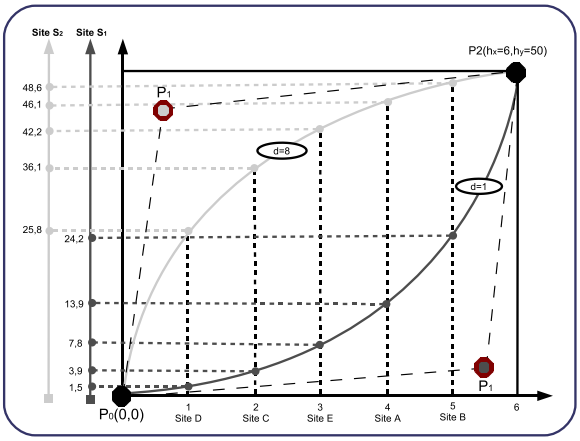

Figure 5. Site Classification

4. Literature Review

In this section we present a general overview of literature on trust in pervasive computing.

In the domain of social sciences there has been substantial research into the concept of trust. Some distinguished authors include Luhmann [16], Barber [5], Coleman [10], and Fukuyama [11]. The findings have been applied in areas including economics, finance, management, government, and psychology. In recent years trust has garnered considerable interest in the computer science community as the basis of solutions to various network security issues such as authentication, authorization and access control.

We now delineate some systems discussed in the literature that pertain particularly to trust-based security and access control in pervasive environments.

In [7] Capra describes hTrust, a trust management system targeted towards mobile / pervasive computing. The architecture is decentralized and makes each entity in the network responsible for its own security. The backbone of the system is a trust formation function which forms an opinion about the trustworthiness of an entity based on aggregated trust information that comprises both locally maintained history of direct experiences as well as recommendations received from other entities. In computing trustworthiness, the function allows an entity to assign more weight to its own past experiences, thus preferring trust reflexivity, or assign more weight to recommendations, thus preferring trust transitivity.

The Pervasive Trust Management (PTM) model [2, 4] by Almenarez et al aims to enable pervasive devices to establish spontaneous relationships in infrastructure-less ad hoc networks. In PTM trust is established between entities either directly or indirectly (through recommendations). The two approaches are considered largely independent of each other. In contrast, Capra's hTrust treats trust forma-

tion as a function of both direct experiences and recommendations. When computing trustworthiness using the indirect approach, PTM assigns weights to the trust values reported by the recommenders which reflect the reliability of the sources. Almenarez et al also present TrustAC [3], which defines access control policies based on trust values obtained from PTM.

Capra and Musolesi [8] present an autonomic trust prediction model for pervasive environments. The model requires that each service provider in the environment advertises a service specification which is a promise of a certain quality of service. Given the service specification and the actual quality of service observed in previous interactions with the service provider, the model uses a Kalman filter [14] to assess its trustworthiness for future interactions.

There are a number of other studies that, although do not address pervasive computing in particular, have been influential in the area of trust management.

[1] by Abdul-Rahman and Hailes is one of the earlier works to describe a practical model to support trust in electronic communities. Trust Builder [24] by Winslett et al and Trust-X [6] by Bertino et al are significant Automated Trust Negotiation systems. Guha et al [12] describe a set of trust propagation schemes and evaluate them on a large trust network consisting of 800K trust scores expressed among 130K people.

To give it precise meaning and to make the concept mathematically manipulable, several authors have proposed formal models of trust.

Marsh [17] was one of the earliest researchers to give a formalism of trust. The model is based on simple linear mathematics. The utility of the model is demonstrated by its application to agents in cooperative situations. Carbone et al [9] propose a formal model of trust in the context of pervasive computing, which focuses on the aspects of trust formation, evolution and propagation. The model is based on domain theory [20]. Jonker and Treur [13] stress that trust is a function of experiences between two entities over time. Based on this notion they develop formal trust evolution and update functions.

Several authors have employed a graph theoretic approach towards the formalization of trust, particularly its evolution and propagation.

Sant and Maple's [19] graph theoretic framework for trust is grounded in the belief that trust is not a local but rather a global phenomenon. The authors suggest that it is important to take a global view to ensure an accurate level of trust in networks. Levien's Advogato system [15] shares similarities with that of Sant and Maple. Saadi et al [21] present a model based on graph theory geared towards pervasive environments. The model addresses distrust as well as trust as an important factor in relationships between entities.

5. Discussion and Conclusion

We have made the argument that in organizations where there may be multiple administrators or administrators may change with time, inconsistencies may occur in the set of trust beliefs of the organization due to variations in the disposition to trust of the administrators.

We have presented our Access Control Model for Pervasive Environments called Chameleon and in the context of this architecture we have introduced a new method for administrators to establish the set of trust beliefs which is more likely to be free from inconsistencies. The key to this solution is tying the quantification of trust not to the multiple dispositions to trust of the administrators but to a single disposition to trust of the organization.

Having administrators evaluate the trustworthiness of neighboring nodes in relation to other nodes (trust sort) and using a mapping function (BV function) for assigning quantitative values are the highlights of the method.

One of the shortcomings that we recognize in this solution is that the BV function assigns trust values evenly to all the nodes. It doesn't take into consideration that the trust values may not be evenly distributed. Elimination of this shortcoming can be a target for future work.

We suggest that the contribution of this paper is to highlight the issue of inconsistencies in a set of trust beliefs and to present a workable solution. More optimal solutions may build upon the ideas that have been presented in this paper.

We have presented our solution in the context of pervasive environments. However, we believe that our proposed method can be adapted to other distributed computing models as well, such as, peer-to-peer, ad-hoc networks, grid computing etc.

References

[1] Abdul-Rahman A., Hailes S. Supporting Trust in Virtual Communities. Hawaii Int. Conference on System Sciences, January 2000.

[2] F. Almenarez, A. Marin, C. Campo, C. Garcia-Rubio. PTM: A Pervasive Trust Management Model for Dynamic Open Environments. In Proc. of the First Workshop on Pervasive Security, Privacy and Trust, PSPT 04 in conjuntion with Mobiquitous 2004. Boston, MA, USA, August, 2004.

[3] F. Almenarez, A. Marin, C. Campo, C. Garc?a-Rubio. TrustAC: Trust-Based Access Control for Pervasive Devices. Security in Pervasive Computing: Second International Conference, SPC 2005, Boppard, Germany, April 6-8, 2005.

[4] Almenarez, F., Marin, A., Diaz, D., and Sanchez, J. Developing a Model for Trust Management in Pervasive Devices. In Proc. of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications, 13-17 March, 2006.

[5] Bernard Barber. The Logic and Limits of Trust. Rutgers University Press, NJ, USA, 1983.

[6] Elisa Bertino, Elena Ferrari, Anna Squicciarini. Trust-X: A Peer-to-Peer Framework for Trust Establishment. IEEE Transactions on Knowledge and Data Engineering, 2004.

[7] Licia Capra. Engineering Human Trust in Mobile System Collaborations. In Proc. of the 12th International Symposium on the Foundations of Software Engineering (SIGSOFT 2004), pp. 107-116, November 2004.

[8] L. Capra and M. Musolesi. Autonomic Trust Prediction for Pervasive Systems. In Proc. of IEEE International Workshop on Trusted and Autonomic Computing Systems (TACS-06), in conjunction with 20th IEEE International Conference on Advanced Information Networking and Applications (AINA 2006), April 2006.

[9] Marco Carbone, Mogens Nielsen, and Vladimiro Sassone. A Formal Model for Trust in Dynamic Networks. BRICS Report RS-03-4, 2003.

[10] James Coleman. Foundations of Social Theory. Harvard University Press, 1990.

[11] Francis Fukuyama. Trust: The Social Virtues and the Creation of Prosperity. Free Press, 1995.

[12] R. Guha, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Propagation of Trust and Distrust. In Proceedings of the International World Wide Web Conference, 2004 (WWW2004).

[13] Catholijn M. Jonker and Jan Treur. Formal Analysis of Models for the Dynamics of Trust Based on Experiences. In Proc. of the 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, 1999.

[14] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. Transactions of the ASME - Journal of Basic Engineering, 82(Series D):35-45, 1960.

[15] R. Levien and A. Aiken. Attack Resistant Trust Metrics for Public Key Certification. In Proc. of the 7th USENIX Security Symposium, pp. 265 -298, January 1998.

[16] Niklas Luhmann. Trust and Power. Wiley, Chichester, England, 1979.

[17] Marsh, S. Formalising Trust as a Computational Concept. Ph.D. Thesis. Department of Mathematics and Computer Science, University of Stirling, Scotland, UK. 1994.

[18] D. Harrison McKnight, Vivek Choudhury and Charles Kacmar. Developing and Validating Trust Measures for e-Commerce: An Integrative Typology. Information Systems Research, September 2002.

[19] Sant, P. and Maple, C. A Graph Theoretic Framework for Trust - From Local to Global. Information Visualization, July 2006.

[20] Dana S. Scott. Domains for Denotational Semantics. ICALP '82 - LNCS, 140, 1982.

[21] Rachid Saadi, Jean-Marc Pierson, Lionel Brunie. (Dis)trust Certification Model for Large Access in Pervasive Environment. J. Pervasive Comput. & Comm, Dec 2005.

[22] Rachid Saadi, Jean-Marc Pierson and Lionel Brunie. The Chameleon: A Pervasive Grid Security Architecture. Third International Conference on Networking and Services (ICNS 2007), June 2007.

[23] Rachid Saadi, Jean-Marc Pierson and Lionel Brunie. Context Adapted Certificate Using Morph Template Signature for Pervasive Environments. Accepted at the The International Symposium on Ubiquitous Computing Systems (UCS 2007), Nov 2007.

[24] Winslett, M., Yu, T., Seamons, K.E., Hess, A., Jacobson, J., Jarvis, R., Smith, B., Yu, L. Negotiating trust in the Web. IEEE Internet Computing, Nov/Dec 2002.