Proceedings of the 2005 IEEE
Workshop on Information Assurance
United States Military Academy, West Point, NY June 2005

Toward a Decentralized Trust-based Access Control System for Dynamic Collaboration

William J. Adams, *Member, IEEE*, and Nathaniel J. Davis, IV, *Senior Member, IEEE*

Abstract – As ad-hoc collaborative environments become more common, the need for access control becomes more imperative. Centralized access control determination fails to work in mobile ad-hoc networking environments, as the information necessary for pre-configuration is not available. This situation is exacerbated by the dynamic nature of the environment's membership, so that the time and resources expended in off-line management are largely wasted. This paper presents a decentralized access control system that implements sociological trust constructs in a quantitative system to evaluate interaction partners. A distributed, node-centric approach to reputation management processes nodal behavior feedback and provides a reputation index that nodes use to determine trustworthiness their peers before establishing associations. The availability of a reputation index gives a measure of expectation of a peer's behavior, based on past performance, and makes a MANET a more distributed operational environment.

Index terms –access control, trust management

I. Introduction

The goal of this research is to create a decentralized trust-based access control system for a mobile ad-hoc collaborative environment. User permissions can be determined and assigned by using behavior grading without the need for pre-configured, centrally managed role hierarchies or permission sets. The system will provide trust-based access control suitable for deployment in a rapidly assembled, highly fluid, collaborative environment.

Ad-hoc collaborative environments assemble and change membership as required to achieve the goals of the group. A feature of these environments is that there is no way of knowing who will join the group, no way of refusing anyone entry into group, and no way of determining how long members will remain in the group. Ad-hoc collaborative environments are formed quickly to enable participants to share information while, at the same time, allowing them to retain control over the resources that they brought with them to the coalition. The trust management system (TMS) developed through this research effectively implements a decentralized access and permission management scheme. Each resource owner uses the linked characteristics of identity, reputation, and risk to make access decisions. Because the TMS tracks a user's behavior, using past behavior as an indication of future performance, no *a priori* user or resource configuration is required.

The TMS also offers a unique ability to enforce multiple access levels without the burden of implementing and managing multiple cryptographic keys or hierarchies of roles. A node provides its peers customized views of its contents and services based on its individual trust profile and the peer's trustworthiness. As peers' reputations change, their access changes to safeguard the node's resources for those peers that have shown themselves to contribute to the node's and the coalition's goals.

The contribution of this work lies in its node-centric approach to trust-based access control that combines reputation and risk (representing two distinct types of trust) and a system memory to sustain them both. Creating a trust-based access control mechanism will enable coalitions to organize quickly so that the group can cooperate to work toward shared goals through the fusion of credential and reputation-based access control system research and applying their concepts to wireless ad-hoc networks.

The rest of this paper is organized as follows. Section II defines the sociological concept of trust and differentiates the types of trust. We present some related work in Section III. In Section IV, we describe the overall security architecture for the system, emphasizing the role of the trust management system. Section V focuses on the design of the TMS that resides on each node. Subsections provide further information on the three critical modules of the TMS. Section VI concludes the paper and describes the future work that is planned for the TMS.

II. Defining Trust

Trust, and more importantly decisions on trustworthiness, are omnipresent in life [1]. Luhmann's sociological approach [2] considered trust as "a means for reducing the complexity in society." This complexity was created as

Bradley Dept. of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061

ISBN 555555555/$10.00 © 2003 IEEE

individuals interacted using their own perceptions, motivations, and goals. Solomon and Flores [3] contended that "trust forms the foundation, or the dynamic precondition, for any free enterprise society." They pointed out that what constituted freedom was the right to make promises and, more importantly, the responsibility for fulfilling them. Trust, therefore, was the basic underpinning of a cooperative environment. Trust was not an inherited trait but was learned as a member of the environment interacted with others. Another applicable definition of trust was provided by Gambetta [4]:

> "…trust (or, symmetrically, distrust) is a particular level of the subjective probability with which an agent will perform a particular action…"

Humans usually based the decision to trust on historical evidence that led them to predict another person or entities' future behavior [5]. When this prediction was shown to be incorrect, the other person was trusted less, if at all. Rather than accept a philosophical betrayal, because "trust can only concern that which one person can rightly demand of another" [6], humans acknowledged the presence of selfishness in their environment [7] and took steps to avoid being victimized by self-centered peers.

Any declaration of another's selfishness was dependent on establishing the context of the trust evaluation. Time and context were two characteristics of the multi-dimensional nature of trust. The time aspect showed that trust was dynamic; a disreputable person could redeem themselves through honest actions and a trusted person could become less reputable if they demonstrated deceit. An example of context was that Alice may trust Bob to order wine at dinner but wouldn't trust him to fix her car. Trust could also be conditionally transitive. In other words, Alice trusted Bob to pick wine and Bob trusted Charles to pick wine, so Alice might reasonably trust Charles in wine selection. In this example, the transitivity of trust was dependent on the context in which the decision to extend trust was being made. In addition to selecting good wine, Bob might trust Charles to split the bill fairly. Alice, who was willing to risk Charles' wine choice, might not be expected to trust the way he divided the check, however.

The presence of selfishness was an example of a trust boundary. On the other side of that boundary was distrust [3], but saying that distrust was either an absence of trusts or the opposite of trust was overly simplistic. A more correct expression would be that just over the trust boundary was "guarded trust," a cynical view of the trust relationship that implied a lack of sufficient information to judge the other individual's trustworthiness. By introducing cynicism, trust was expressed in a continuous, rather than discrete, manner.

In sociological settings, trust was expressed in continuous terms, either through qualifying trust in terms of context (e.g., Alice trusts Bob's taste in wine) or acceptance of risk (e.g., since the bill is only $5, Alice was willing to see how Charles splits the check.) Trust was built through risking betrayal. Peers earned trust as they performed to positive expectations. Individuals collected and evaluated evidence of the behavior of their peers, forming a perception of behavior or reputation for each individual they interacted with.

Given the many, sometimes confusingly contradictory definitions of trust, McKnight and Chevrany [8] attempted to describe a framework that provided a taxonomy of three types of trust (see Figure 1). From this taxonomy, they were able to quantify and generalize the process an individual used to influence their behavior. Starting from the bottom of the figure, this model shows how the trust types combine with trust beliefs and are adjusted by trust intentions before becoming behavior; the expression of the trust decision. The way these are implemented in the TMS is discussed in Section V.

System Trust was the extent to which an individual placed trust in the environment around them. In a personal sense, this type of trust reflected a person's feeling of safety in their current location or present situation (e.g., Alice always locked her car doors when she drove through the downtown area.) System trust was built from both structural assurances and situational norms. The individual's belief that the system's rules and regulations would protect them was an example of structural assurance. Similarly, if a user's past experience was that a certain area was risky, the situational norm prompted his System Trust to provide appropriate protection.

Interpersonal Trust was user centric and described an individual's general willingness to extend trust across a broad range of situations to any number of people. This

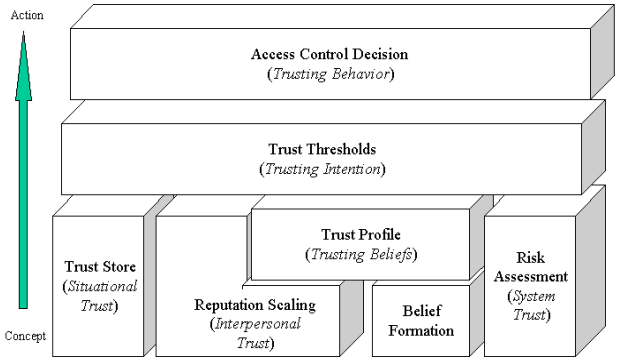

Figure 1 Implementation of Trust Types and Constructs [8]

ISBN 555555555/$10.00 © 2003 IEEE

type of trust formed the basis of an individual's approach to interaction. It demonstrated an expectation of other people once their trustworthiness had been evaluated. Interpersonal trust was built from experience, either referred from other trusted individuals or from direct contact with the person in question. This was modified by a trait that McKnight and Cheverny call the Trusting Beliefs, reflecting the general tendencies people had toward extending trust. Some people were trusting, believing in the goodwill of their fellow man. Other individuals were more cynical, requiring others to demonstrate their trustworthiness before risking interaction.

Literature often confused trust with reputation and it is necessary to differentiate between the two concepts [9] clearly. Trust is active; it is a node's belief in the trust qualities of a peer. Trust is extended from a node to its peer. Reputation is passive; it is the perception that peers form about a node. Reputations are individual in the sense that peers can form different reputations about the same node, based on the fact that they can have different experiences or observe different behavior.

Humans developed a concept of reputation as an aggregation of trust information. They used this concept to predict the actions of others based on historical behavior information gained through personal interaction or the shared observations of peers [10]. Researchers pointed out that reputation could be utilized in a virtual society, such as a MANET, to make up for the lack of the physical, interpersonal clues that humans use to determine trustworthiness.

Situational Trust described the degree of trust that an individual was prepared to trust any other person in a given situation. This trust was formed upon the intention to extend trust in a particular situation, regardless of what the person knew or did not know about the other party in the situation. It was suggested that this type of trust occurred when the trusting party stood to gain with very little attendant risk. Situational trust was different than System trust because there were no implied structural or system safeguards. It was, in short, an individually conceived situational strategy and did not involve an evaluation of the trustworthiness of the other party.

III. Related Work

Resnick [11] discussed the risks in exchanging valuable information with parties who are identified only through pseudonyms and self-descriptions in centralized systems. He elaborated on the concept of reputation systems, a means by which the behavior of the participants in information exchanges was tracked and each contributor gained an expectation of the others' actions. The psychological concept that individuals would assist others in order to build and maintain a positive reputation helped instigate a line of research into the application of behavior grading and cooperation incentives in networked systems.

Without a persistent and verifiable identity construct, peers could assume multiple personas in the network for malicious purposes. Douceur [12] pointed out that the benefit of having a reputation was limited by this ability. He dubbed this phenomenon the Sybil attack. Concern over the Sybil attack led to creation of an atomic identity for each node in the network. Once a system had an atomic identity for each node in the network, it could link each node with its reputation. Global identity also enabled the system to track the source of reports and defeat possible attacks by misbehaving nodes through non-repudiation of the reports. Distributed Public Key Infrastructure (PKI) schemes offered a means of affixing a persistent identity to each node in the network. Example PKI schemes include PGP systems like the one presented in [13] and X.509-based certificate schemes [14].

Three types of reputation systems have been studied: positive reputation, negative reputation, and a combination of both. Positive reputation systems only recorded positive behavior observations or feedback. An example of such a system was CORE [15]. CORE used positive feedback to enforce node cooperation in a network. In the CORE system, nodes built positive reputations by collaborating (e.g., routing traffic) with their peers. The drawback of this system was its reliance on positive reports, without the facility to submit negative feedback.

Negative reputation systems only shared complaints or negative behavior observations. Peers were assumed to be trustworthy, so behavior feedback was used to negatively modify a node's reputation. Systems that utilized only negative feedback [16] lacked an ingrained incentive for a node to participate in the network. Since a node's reputation could only decrease, the system promoted selfishness (i.e., a node only participated when it wanted something) or malicious behavior (i.e., a node actively pursued a poor reputation so that it would not have to expend resources to help its peers, also called freeloading.)

The drawbacks in negative reputation systems were addressed in the CONFIDANT system [17, 18], which extended the notion of reputation to include both positive and negative reports. This extension fostered the elimination of misbehaving nodes through isolation. CONFIDANT policed the network through rating lists and a scheme of weighting observations. It also enabled the rehabilitation of a node's reputation, provided the node demonstrated sustained positive performance.

ISBN 555555555/$10.00 © 2003 IEEE

A better example of trust-based access control in a wireless environment was offered by the SECURE project [19]. This project investigated the application of trust-based methods to enable short term, ad-hoc collaboration between handheld devices. Nodes joined the network by requesting access from a self-appointed group leader. The leader then queried the other group members and they voted on whether or not to admit the new node. SECURE did not attempt to implement a behavior history because nodes did not possess a verifiable, non-reputable identity and, therefore, any behavior history would be rendered suspect by the possibility of a Sybil attack [12]. Only after a node was admitted to the network could its peers track its behavior and then for only that session. The authors' admit that their method was susceptible to the Sybil attack and did not implement trust thresholds due to the lack of a persistent behavior history. These deficiencies made the project's results unsuitable for implementation in a situation where the TMS was expected to protect live data.

IV. System Security Architecture

The TMS was implemented as a central, data processing layer of the overall system security architecture, as shown in Figure 2. The Key Management System (KMS) managed user identity certificates and established the rules for issuing, reissuing, and revoking certificates [20]. In a centralized network, this KMS relied on directory replication and certificate revocation lists (CRLs). In a decentralized environment, the goal was to provide the KMS with access control decisions based on the trustworthiness of the perspective peer node. The TMS provided the KMS with a layer of abstraction of the overall trustworthiness of nodes, based on the activity of the nodes in the network. As the central layer, the TMS resided on each node and helped determine whether to trust or distrust its peers based on its individual trust thresholds. The TMS then reported its trust decisions to the KMS for its consideration.

At the lowest layer, an intrusion detection system (IDS) or monitoring scheme [15, 18] provided periodic performance observations to the network. These observations were distributed throughout the system in a modified epidemic routing algorithm, similar to the selective dissemination scheme proposed in [21]. The observations compared a node's expectations against the observed performance of its neighbors. Observations were made on trusted peers (TPs) as well as on neighboring nodes that were within "listening range" but were not necessarily directly trusted. Nodes observed performance in areas such as routing or file access and periodically generated positive or negative feedback. These Feedback Items (FIs) were passed only to other TPs.

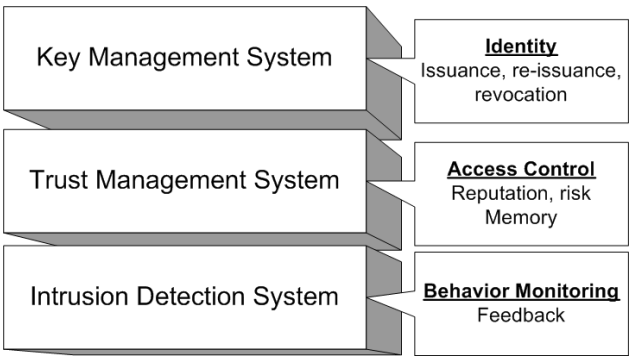

Figure 2 System Security Architecture

V. Trust Management System

This research has developed a system where user nodes cooperated to exchange behavior reports and establish a record of each node's behavior history. This history, based on reports and observations, was expressed as a reputation index (RI). The RI, with evidence in the form of signed FIs, provided an expectation of their partner's behavior before entering into or dissolving an SA. By providing an indication of each other's trustworthiness, nodes avoided misbehaving nodes. Figure 3 shows the TMS that is installed on each node. In the following sections, this paper discusses how the TMS implements each of McKnight and Chervany's [8] constructs (see Figure 1) to produce an access control decision.

A. Implementing the Reputation Scaling module

Each node gathered and processed feedback to calculate a usable RI for its peers. The TMS implemented an

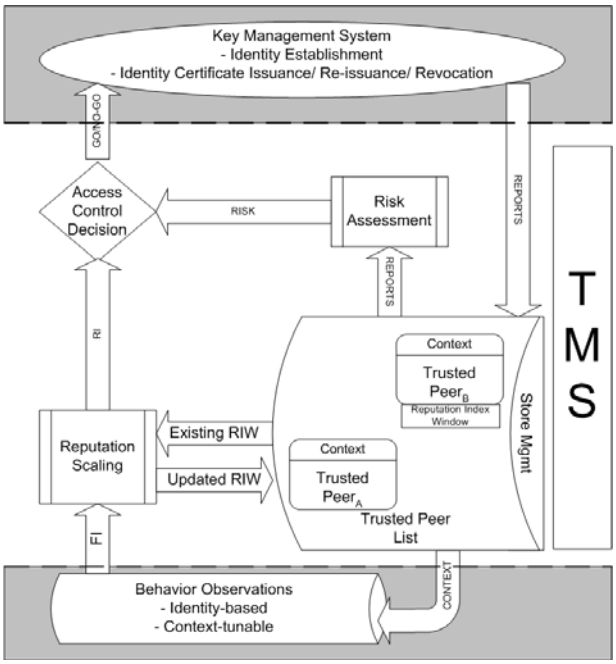

Figure 3 Trust Management System Architecture

ISBN 555555555/$10.00 © 2003 IEEE

Interpersonal Trust model [10] to represent the reputations that were compiled by a node on each of its peers. This trust type was node specific, so that the trust of one node to another was direct and not transitive. The following summarizes the trust model:

1. Trust was context dependent.
2. Trust had positive and negative degrees of trustworthiness.
3. Trust was expressed in continuous values, as described by Marsh [1].
4. Trust was based on experiences and observations between individuals.
5. Trust information was exchanged between nodes.
6. Trust was subjective. Nodes calculated different reputation values for the same observed node.
7. Trust was dynamic and was modified, in a positive or negative direction, based on new observations and reports.

Once the reports and observations had been gathered, they were processed to provide a meaningful value that a node used for its trustworthiness evaluation. The reputation value needed to give a conservative approximation of the feedback input. We also wanted to emphasize current behavior while aging older input to diminish its impact on the reputation calculation. As in CORE [15] and CONFIDANT [17], a node maintained a reputation value for each TP. Nodes entered the network with a reputation value of 0, a basic level of trust. Our expectation was that a node would desire a positive reputation. A node with a negative reputation would be isolated as nodes refused to interact with it [11].

Our Reputation Scaling module applied different levels of trust to reports and observations. Nodes placed full trust in KMS reports. On the other hand, periodic observations (obs_x) from other peers and friends were weighted using the reporting node's reputation (RI_x) before being integrated into the reputation calculation, as shown in Equation 1. These weighted observations were called Feedback Items (FIs).

$$FI = RI_x * obs_x \quad (1)$$

We developed the 3Win method to calculate the RI for each peer node [22]. Modeled after a method of removing transients using sub-samples [23], this method divided a node's history into three weighted performance windows that revealed tendencies in a node's behavior, shown in Equation 2 [22]. These windows were named Reputation Indexing Windows (RIWs) and numbered one through three. RIW_1 contained the newest FIs and RIW_3 held the oldest. FIs were "pushed" through the windows (i.e., from RIW_1 to RIW_2 to RIW_3) as new FI arrived. When an FI was pushed out of RIW_3 it was discarded.

$$\lambda = 0.66;\ \mu = 0.22;\ \nu = 0.11$$
$$RI = (\lambda * RIW_1) + (\mu * RIW_2) + (\nu * RIW_3) \quad (2)$$

Pushing FIs through the RIW diminished the impact of older items on the reputation calculation. The resulting RI emphasized current performance and provided a conservative approximation of the behavior trend. The 3Win approach allowed a node to recover from a poor RI by demonstrating sustained "good" behavior. Allowing rehabilitation offered a chance for nodes to show that they could be trusted members of the network. Finally, 3Win provided the non-repudiation not available in other RM systems by maintaining the FIs that had been used to derive the reputation of the prospective partner. These digitally signed proofs substantiated the given reputation value and meant that nodes could note refute their RI.

B. Implementing the Trust Store module

Where Interpersonal trust was dependent upon peer behavior trends and System trust was determined through an evaluation of system behavior tendencies, Situational trust was independent of the behavior of other users altogether. This type of trust used the trust store, representing the user's memory of previous peers and situations, to determine what action it would take. A situational trust decision was predicated on remembering a previous decision that had yielded a positive outcome, regardless of the behavior of peers that may or may not have been involved.

The trust store contained the certificates and the RIWs for each of a node's TPs in structures called Atomic Behavior Records (ABRs), as depicted in Figure 4. ABRs consisted of a permanent identity portion and the the reputation scaling mechanism's RIWs. ABRs of active partners were kept in the Active Cache, migrating to the Passive Cache

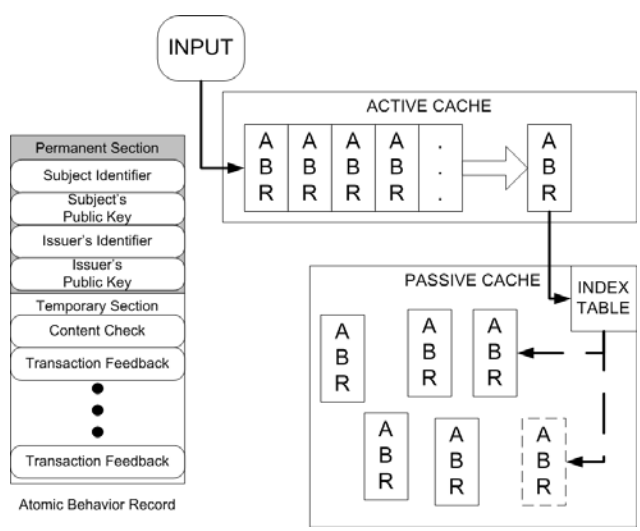

Figure 4 Trust Store Contents and Organization

ISBN 555555555/$10.00 © 2003 IEEE

as new partnerships were formed and forced the older and less active ABRs to be removed. When the Passive Cache was full, the node selectively eliminated or "forgot" associates to make space for new objects [24].

Efficiency, in this case, was defined as limiting the number of "re-introductions" that a node required. Re-introductions were cases where a node forgot a peer that it once had security associations with and had to go through the entire introduction process, as if the two nodes had never dealt with each other. While modern mobile networking devices continue to expand in memory size and processing capacity, we have found that there is a practical limit to how much information needs to be stored. By limiting the amount of stored information, we can increase the memory and processing resources that can be made available to applications and data.

C. Implementing the Risk Assessment module

The Risk Assessment module (RAM) implemented System Trust. The RAM monitored globally available information, in the form of reports and complaints from the KMS, to determine the general "riskiness" of the network as a situational norm. This was an expression of how risky an action was likely to be, given the current state of trust events in the network. Simplistically, this trust state could be phrased as "if other people are having success then I'm more likely to give it a try." The risk assessment was derived from the information provided by the network control plane.

Risk evaluation was different from reputation scaling because the risk assessment considered all of the trust reports in the system, not just those of a node's TPs. Once determined, the risk assessment of the network was applied to a node's trust thresholds, adjusting the threshold to protect the node. The adjustment was temporary; as the node made a risk assessment on a regular basis and never altered the node's trust profile or its trust thresholds.

The risk assessment was performed using a Bayesian technique similar to the one described by Buchegger [25]. Our previous work [22] demonstrated the technique's consistent performance and, because it did not require the report messages to be stored, the averaging technique produced an approximation of the ratio of trust transactions being established and dissolved. Using this ratio, the system made a risk assessment about the general "state of trust" in the system. This ratio, therefore, was representative of the risk that a node assumed when it extended its trust. The concept was that nodes should be more cautious in environments where there is a lack of trust.

D. Implementing Trust Profiles

Users established how trusting they could afford to be based on their individual goals and objectives. In a general sense, users are grouped into four trust profiles after Prietula and Carley [26]. **Altruistic** users trusted any peer, regardless of that peer's reputation. **Forgiving** users established trust thresholds and evaluate peer reputations against these thresholds to determine trustworthiness. **Cynical** users were similar to Forgiving users but did not allow rehabilitation. Once a cynical user determined that a misbehaving peer was untrustworthy, no attempt was made to collect further behavior information or readjust the peer's reputation. **Distrusting** users allowed neither negative reputations ($RI_x < 0$) nor rehabilitation.

E. Implementing Trust Thresholds

A user established a trust threshold based on the selected trust profile. Every user had two trust thresholds. One was a *trust* threshold, above which a user extended trust. The other was a *distrust* threshold, below which a user withdrew trust. RIs were expressed in continuous values between [-1,1], allowing a user to evaluate the RIs of prospective peers against these thresholds. If the peer's reputation exceeded the trust threshold, trust was allowed and a security association might be established. If, at a later time, the peer's reputation fell below the distrust threshold, a user would withdraw its trust and might, based on its individual security policy, report the access attempt to the network's control plane.

The system denoted the reputation Alice maintained of Bob as $RI_A(B)$. The RI is represented by number values in the range [-1, 1]. This value represented the trust Alice placed in Bob; the higher the number, the more trust was imparted. Peers viewed a user with a reputation of –1 as completely untrustworthy. Peers view a user with a reputation of +1 as completely trustworthy.

Peers viewed users that they have no information about with a reputation value of 0. Liu and Issarny [9] point out that this assignment made no attempt to differentiate between new comers, strangers, users that had not participated (free-loaders), or users whose reputations had been calculated to be 0. Zero was considered a neutral value as it gave a new user a basic reputation to start with while limiting the impact strangers, free-loaders, and active users with low reputations had to the network. Referring back to Section II, an RI_x of 0 implied that the user needed more information before making a decision, rather than implying trust or distrust.

Figure 5 illustrates a practical example of the use of trust thresholds in reputation-based systems. Alice and Bob are members of a dynamic collaborative environment. They

ISBN 555555555/$10.00 © 2003 IEEE

meet at t_1 and Alice calculates that Bob's reputation meets her Trust Threshold. They establish an association and share resources. At t_2, Bob's reputation dips below Alice's trust threshold but, since his RI is above her distrust threshold, she continues to trust him. The lowered reputation may be because of transmission or resource difficulties, rather than misbehavior, and Alice accepts the risk rather than sever the association and go through re-introduction in the near future. At t_3, however, Bob's reputation goes below Alice's distrust threshold and the association is severed. At this point, Bob's reputation will have to exceed the trust threshold before Alice will consider trusting him again. Even after Bob's RI exceeds Alice's trust threshold, Fernandes recommended [27] that Bob be forced to wait through a probationary period before Alice considers trusting him again. While this approach may be viable in large systems with large numbers of transactions, it is yet to be determined if the added cost of timers and processing is worthwhile in environments with fewer, less active members.

VI. Future Work and Conclusion

Trust management offers the ability to make access control decisions in mobile ad-hoc collaborative environments without the need for pre-configuration or centralized management. By linking a node's identity to observations on its performance, its peers can calculate its reputation and evaluate its trustworthiness. Through a process of introduction, nodes share performance observations and are able to calculate reputations of newly encountered nodes in a peer to peer manner.

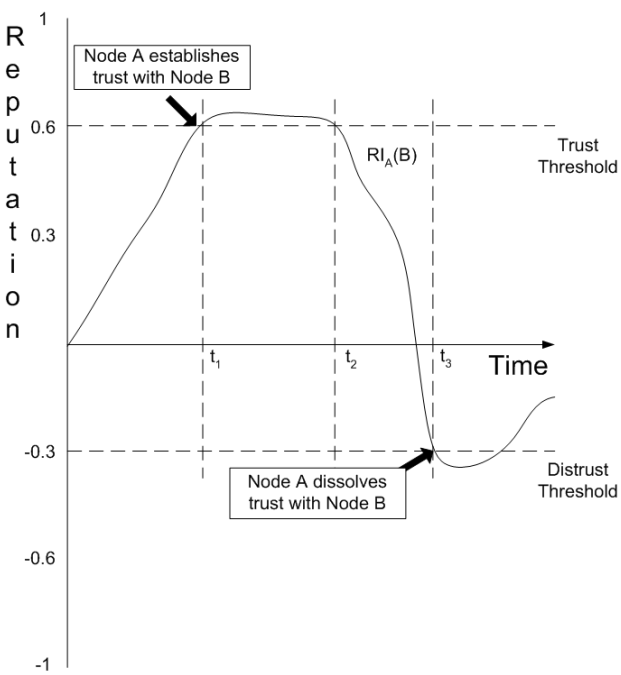

Figure 5 Example of Trust Thresholds

In this paper we have shown that a node-based TMS can implement the three types of trust, apply risk and reputation calculations, and present the application layer security system with an access control decision. The TMS provide a unique ability to enforce multiple access levels dynamically, based on behavior information. A node provides its peers customized views of its resources based on its individual trust profile and the peer's trustworthiness. As reputations change, the amount of access changes to preserve a node's resources for those peers that have demonstrated positive contributions to achieving the coalition's goals.

This work has contributed an alternative access control systems that is more suitable to ad-hoc environments than traditional centralized security methods. Using the trust constructs of McKnight and Chervany, we have begun work on realizing a complete trust management-based access control system. This system will provide security based on an analytical evaluation of a peer node's past performance. To date, this system has proven its ability to calculate nodal reputations and react appropriately to attacks on the TMS. The reputation scaling mechanism and trust store have been modeled and tested against a set of behavior scenarios. The results of these tests have been documented in our previous research [22, 28]. Currently, a baseline system is being constructed using the parameters of the CONFIDANT research [25]. This system will be used to provide a reference to measure the success of the TMS against. Once verified, the TMS will be validated as the access control mechanism in a peer-to-peer collaborative setting. Our goal is to provide these dynamic environments an access control system that is more resilient and robust than can be provided through current, centralized means.

References

[1] S. Marsh, "Formalising Trust as a Computational Concept," in *Department of Mathematics and Computer Science*: University of Stirling, 1994.
[2] N. Luhmann, *Trust and Power*: Wiley, 1979.
[3] R. Solomon and F. Flores, *Building Trust*. New York, NY: Oxford University Press, 2001.
[4] D. Gambetta, "Can we trust trust?," in *Trust, Making and breaking cooperative relations, electronic edition*, D. Gambetta, Ed.: Univ. of Oxford, 1988, pp. Ch 13, pp 213 - 237.
[5] K. Aberer and Z. Despotovic, "Managing trust in a peer-2-peer information system," presented at the tenth international conference on Information and knowledge management, Atlanta, GA, 2001.
[6] L. Hertzberg, "On the Attitude of Trust," *Inquiry*, vol. 31, pp. 319, 1988.
[7] P. Michiardi and R. Molva, "Simulation-based Analysis of Security Exposures in Mobile Ad

ISBN 555555555/$10.00 © 2003 IEEE

Hoc Networks," presented at the European Wireless Conference, 2002.

[8] D. McKnight and N. Chervany, "The Meanings of Trust," Carlson School of Management, University of Minnesota, Technical Report TR 94-04, 1996.

[9] J. Liu and V. Issarny, "Enhanced Reputation Mechanism for Mobile Ad Hoc Networks," presented at the 2nd International Conference of Trust Management (iTrust 2004), Oxford, UK, 2004.

[10] A. Abdul-Rahman and S. Hailes, "Supporting trust in virtual communities," presented at the 33rd Annual Hawaii International Conference on System Sciences, 2000.

[11] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman, "Reputation Systems," *Communications of the ACM*, vol. 43, pp. 45 - 48, 2000.

[12] J. Douceur, "The sybil attack," presented at IPTPS02 Workshop, Cambridge, MA, 2002.

[13] S. Capkun, L. Buttyan, and J.-P. Hubaux, "Self-Organized Public-Key Management for Mobile Ad Hoc Networks," presented at the ACM International Workshop on Wireless Security, WiSe 2002, 2002.

[14] L. A. DaSilva, S. F. Midkiff, J. S. Park, G. C. Hadjichristofi, N. J. Davis, K. S. Phanse, and T. Lin, "Network mobility and protocol interoperability in ad hoc networks," *Communications Magazine, IEEE*, vol. 42, pp. 88-96, 2004.

[15] P. Michiardi and R. Molva, "CORE: A Collaborative Reputation Mechanism to enforce node cooperation in Mobile Ad-hoc Networks.," presented at IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security: Advanced Communications and Multimedia Security, 2002.

[16] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," presented at 6th annual international conference on Mobile computing and networking, Boston MA, 2000.

[17] S. Buchegger and J.-Y. Le Boudec, "Performance analysis of the CONFIDANT protocol," presented at the 3rd ACM international symposium on Mobile ad hoc networking & computing, Lausanne, CH 2002.

[18] S. Buchegger and J.-Y. Le Boudec, "Nodes bearing grudges: towards routing security, fairness, and robustness in mobile ad hoc networks," presented at the 10th Euromicro Workshop on Parallel, Distributed and Network-based Processing, 2002.

[19] E. Gray, P. O'Connell, C. Jensen, S. Weber, J.-M. Seigneur, and C. Yong, "Towards a Framework for Assessing Trust-Based Admission Control in Collaborative Ad Hoc Applications," Department of Computer Science, Trinity College Dublin, Dublin, IE, Technical Report TD-2002-66, 2002.

[20] G. C. Hadjichristofi, W. J. Adams, and N. J. Davis, "A Framework for Key Management in a Mobile Ad-Hoc Network," presented at the International Conference on Information Technology Coding and Computing (ITCC 05), Las Vegas, NV, 2005.

[21] A. Datta, S. Quarteroni, and K. Aberer, "Autonomous Gossiping: A self-organizing epidemic algorithm for selective information dissemination in mobile ad-hoc networks.," Ecole Polytechnique Federale de Lausanne 2004.

[22] W. J. Adams, G. C. Hadjichristofi, and N. J. Davis, "Calculating a Node's Reputation in a Mobile Ad-Hoc Network," presented at the 24th IEEE International Performance Computing and Communications Conference (IPCCC 2005), Phoenix, AZ, 2005.

[23] R. Jain, *The Art of Computer Systems Performance Analysis*. New York, NY: John Wiley & Sons, 1991.

[24] E. Gray, J.-M. Seigneur, Y. Chen, and C. Jensen, "Trust propagation in small worlds," presented at the First International Conference on Trust Management (iTrust2003), 2003.

[25] S. Buchegger and J.-Y. Le Boudec, "A Robust Reputation System for Mobile Ad-Hoc Networks," EPFL 2003.

[26] M. Prietula and K. Carley, "Boundedly rational and emotional agents - cooperation trust and rumor," in *Trust and Deception in Virtual Societies*, C. Castelfranchi and Y.-H. Tan, Eds. Norwood, MA: Kluwer Academic Publisher, 2001, pp. 169 - 193.

[27] A. Fernandes, E. Kotsovinos, S. Ostring, and B. Dragovic, "Pinocchio: Incentives for honest participation In Global-Scale Distributed trust management," University of Cambridge, Cambridge, UK 2003 2003.

[28] W. Adams, R. Thomas, and N. Davis, "Sizing the Credential Cache in a Trust-based Access Control System," submitted to IEEE Global Telecommunications Conference (GLOBECOM 2005), St. Louis, MO, 2005.

ISBN 555555555/$10.00 © 2003 IEEE