A Highly Reliable Access Control Model for Ad Hoc Networks

Insoo Cho[1], Man Ju Lee[1], Tea Sang Yun[1], Jung Ah Kim[1],
Pho Duc Giang[2], and Sungyoung Lee[2]
[1]*WIZ Information Technology Company, Seoul, Korea*
[2]*Computer Engineering Department, Kyung Hee University, Korea*
{bility, niceman, yun3x3, rosakim}@wizit.com, {pdgiang, sylee}@oslab.khu.ac.kr

Abstract

Unlike the conventional networks, the unique characteristics of mobile ad hoc networks (MANETs) pose a number of nontrivial challenges for pervasive service provision. Particularly, mobility of users/devices causes un-predefined and unpredictable changes in physical location and in available resources and services, event at runtime and during the same service session, thus forcing us to consider very dynamic aspects of evaluation when designing a security access control model. Alternatively, there is generally no a priori trust relationship among entities interacting in ad hoc networks which makes it essential to establish trust from scratch. This task becomes extremely challenging when it is simultaneously necessary to protect the privacy of the users involved. In this study, we show how trust evaluation process of a system can be based on previous interactions and peer recommendations. Regarding the combination of these two factors, our trust-aware access control model can establish appropriate trust values for different situations, providing a confident supervision mechanism for ad hoc users.

1. Introduction

The widespread availability of ad hoc networks in the environments where users live and work together with the increasing diffusion of portable devices, such as PDAs, laptops, and mobile phones, creates novel chances for users to access services anywhere, at any time and from various access devices [1]. However, the flexibility of the ad hoc networks comes at a cost – higher risks and privacy disclosures. The environment itself lacks a priori trust among parties and the interactions are ad hoc naturally. In other words, trust relationships have to be started from scratch. The traditional association with a network provider may not exist, replaced by a far more vague connection with a number of unknown entities, network nodes and service providers. Therefore, designing a sufficient and suitable access control system for security and privacy in ad hoc networks becomes very topical.

Additionally, in ad hoc networking community, users' access rights change dynamically in terms of their relationship with the medium by which data are generated and sometimes the clients cannot be anticipated. Traditional authentication and access control are effective only in situations where the system knows in advance which users are going to access and what their access rights are. Hence, we need a robust solution capable of control the security and privacy issues on the runtime so as to provide essential amount of services to requesters who are either unfamiliar with the system or do not have enough access rights to certain services. We believe that decision to allow or deny certain request towards the user's resources/services in ad hoc context should rely on a flexible and dynamic access control model.

In this paper, we introduce the idea of using trust to provide finer-grained access control over the sensitive resources, thus helping to manage the security and privacy issues efficiently. In order to determine whether someone is trusted or not to allow her access different parts of our services, we first depend upon two different evaluation factors: peer recommendation, and time-based past access history to calculate the trust value. After that, based on the outcome of trust estimation process, we assign one of the two possible access permissions: *allow* or *block* to the requester. By applying pre-defined trust-based security access control policies, we are able to administer and disseminate appropriate services/resources to the partner.

The remaining paper is organized as follows. We briefly overview related work in Section 2. Next, in Section 3, we formalize fundamental concepts of trust

0-7695-2943-7/07 $25.00 © 2007 IEEE
DOI 10.1109/NPC.2007.96

IEEE computer society

to elaborate the functional aspects of the scheme proposed in Section 4. We are then describing the methodology in detail in this section. Finally, in Section 5, conclusions and future work are drawn.

2. Related Work

So as to provide network security, support for confidentiality, integrity, non-repudiation, and access control should be prepared for in advance [2]. We believe that access control is the cornerstone service, since other services depend on the control access of communication entities.

Traditional access control methods such as mandatory access control (MAC) and discretionary access control (DAC), delegate or revoke users' access privilege directly. However, due to the problem that MAC and DAC mechanisms assign a security clearance to each user to restrain access capability, these systems will become inconvenient and complicated when the number of users and the relationship among them increase rapidly.

Role Based Access Control (RBAC) [3,4] is probably one of the best known methods for access control, where entities are assigned roles in which permissions associated with each role, instead of users. Unfortunately, this is difficult for systems where it is not possible to assign roles to all users and in the situation that foreign users are common.

Pirzada et al. [5] extends Kerberos protocol for mobile ad hoc network security authentication by deploying multiple Kerberos servers for distributed authentication and load distribution. All servers share a secret key, and copy the other users' hashed password periodically or on demand. Their solution overcomes the single point failure created by central key server of the traditional Kerberos model. Nevertheless, authentication is relied upon users' identity. Thus, it more or less affects the privacy aspects of the entities joining the medium.

Kagal et al. [6] applied trust factor which is based on time-lived signed delegations and XML signatures (www.w3.org/signature) to examine unfamiliar requests before making a decision whether those requests should be allowed to access certain service or not. They also mentioned about the issue that a stranger who wishes to access some resource should find privileged users for asking delegation. However, they have not shown how and which evaluation method the stranger should be trusted properly.

In the field of ad hoc networks, a key property is that it contains mobile computers or devices [7]. The devices within the environment are not physically secured and can move freely in and out of various mobile ad-hoc networks. Each mobile device has the potential to encounter thousands of other mobile devices within a short period of time. Therefore attempting to identify every device to enforce static security policies becomes impossible. For example, a handful of people can form an ad hoc group and record their meeting using a camera that is administrated by the environment. They should only have accesses to the video produced during the meeting period but not others. The system must be able to associate a piece of information with the correct set of users while it is being produced.

3. Basic Concepts

In this section, we first discuss possible definitions of trust to give an idea of the different meanings associated with it, and we point out the notion that we refer to in the remainder of the discussion. We then discuss two different aspects related to trust: how trust is established and how it is utilized. Generally, the notion of trust is exploited in a large number of different contexts and with various meanings. It is a fuzzy notion about which no agreement exists in the computer science literature, although its importance has been widely recognized. Different people with different background have tried to base their own views on their circumstances. For instance, Grandison and Sloman [8] defined trust as: *"the firm belief in the competence of an entity to act dependably, securely, and reliably within a specified context."* They argue that trust is a composition of many characteristics – reliability, dependability, honesty, trustfulness, security, competence, and timeliness – which may have to be considered when deploying trust.

Out of several definitions of trust, one definition closely suitable to our approach is by Blaze and Feigenbaum [9]. According to whom, trust issues include formulating security policies and security credentials, determining whether particular sets of credentials satisfy the relevant policies. Such a definition of trust basically refers to security policies regulating accesses to resources and credentials that are required to satisfy such policies. Regarding the notion of a requester, which we refer to as a principal, it can be formally defined as follows:

Definition 3.1 A user, a service, an application, or a system which requests or can send requests to other users, services, applications, or systems is called a principal.

We denote a principal by P or Q for the rest of this paper. In our proposed approach, every principal has its own trust-based access control policy which indicates different types of resources to be disclosed.

Definition 3.2 The trust of principal P on principal Q is a real number between 0 and 1.

We denote the trust of P on Q as $T_{P,Q}$. According to the definition, $T_{P,Q} \in [0,1]$. Hence, P completely trusts Q if $T_{P,Q} = 1$ and completely distrusts Q if $T_{P,Q} = 0$.

Definition 3.3 The access control policy $P_{P,k}$ of a principal P, having k types of resources to be shared, is defined as a mapping from its policy to the set of actions {A - Allow, B - Block}.

Assume that a principal P provides two different types of resources (k = 2). Hence, $P_{P,2}$ = A implies full access to the second resource and $P_{P,1}$ = B implies no access to the first resource at all.

Definition 3.4 For a principal P, a **trust-access mapping** denoted by M_P is a mapping from [0,1] to its access control policy $P_{P,k}$ defined as:

$$M_P(x) = \begin{cases} A & , c_k \le x \le 1 \\ A & , c_{k-1} \le x \le 1 \\ \vdots & \quad \vdots \\ A & , c_2 \le x \le 1 \\ A & , c_1 \le x \le 1 \end{cases} \quad \Leftrightarrow$$

$$M_P(x) = \begin{cases} B & , 0 \le x < c_k \\ B & , 0 \le x < c_{k-1} \\ \vdots & \quad \vdots \\ B & , 0 \le x < c_2 \\ B & , 0 \le x < c_1 \end{cases}$$

Where x, $c_1, c_2, \dots, c_k \in [0,1]$.

In the previous example, the principle P might define a mapping function as:

$$M_P(x) = \begin{cases} A & , 0.55 \le x \le 1 \\ & \\ A & , 0.70 \le x \le 1 \end{cases} \quad \Leftrightarrow$$

$$M_P(x) = \begin{cases} B & , 0 \le x < 0.55 \quad (2) \\ & \\ B & , 0 \le x < 0.70 \quad (1) \end{cases}$$

If the trust evaluation of P on another principal Q which wishes to access P's 1st resource is just 0.2, then respecting (1) and even (2) $M_P(T_{P,Q}) = M_P(0.2)$ = B (Block), implies that P has no resource exposure for the request Q. In other words, if Q requests for access 1st resource of P, Q will not receive any related information since the Q's trust value is unacceptably low. In the next section we will technically present a procedure to evaluate the trust value and develop different aspects of a trust evaluation method to calculate the trust of any principal.

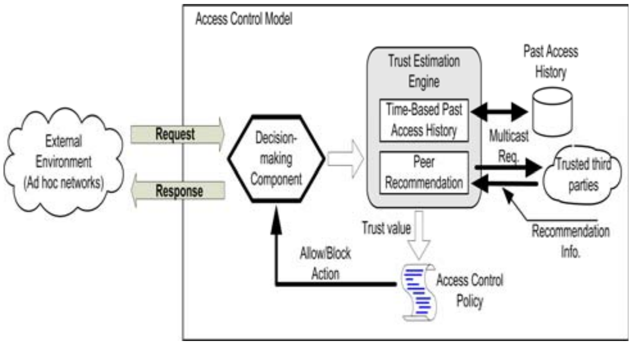
4. Our Proposed Solution

In this section, we propose an access control scheme based on the concept of trust with peer recommendation and past access history, and the trust-based security policy to guarantee that users' resources will not be delivered in a wrong way to a wrongdoer. There are two different stages in our solution: i) we estimate the trust value for each request coming from an entity; ii) we exploit the trust-based policy to make decision whether to accept the request or not. All these two phases can be performed automatically. We aimed to develop a system that required minimal ongoing user involvement. In particular, we did not want users to have to repeatedly evaluate the acceptability of a request for private resources. Instead, we wanted to push a query's acceptance or rejection to the system itself and only bring a query to users' consideration if they had not established a policy to handle it. Moreover, we believe users' resources/services should be protected by default; as a consequence, the system architecture lets a user elect to share certain resource rather than protect specific one.

4.1. Trust Evaluation Module

This module will initially base on the past access history stored in log-files of an entity in an ad hoc group during specific time interval to produce proper trust value for the request. If there is no any previous interaction correspondent to this query, now this module will ask other trusted entities who are currently active in a certain range of this ad hoc environment to give recommendations for Q. The general model of trust evaluation is shown in Fig. 1.

Figure 1. Our Trust-Based Access Control Model



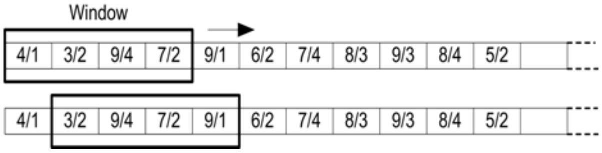
4.1.1 Time-based Past Access History. Past Access History is an entity's previous interaction knowledge to certain principal. As a matter of fact, past access history is usually recorded in log files on the subjects' systems that keep track of all actions relational participants took with the system. Since the log file is configured to keep monitoring events for a specified amount of time, it is reasonable for us to apply trust evaluation based on the temporal factor.

We can generally define successful and unsuccessful access between principal *Q* and system *P* established on the past behaviors in which an unsuccessful access means that the principal did not get the outcome as it expected. Let us define SA_t and UA_t as the number of successful past access times and unsuccessful access times of the system at time t respectively. Now, the trust value of *Q* as calculated by a system *P* is defined as follows:

$$T_{P,Q} = \left[\frac{SA_t}{SA_t + UA_t}\right]\left[1 - \frac{1}{Ae^{(\alpha SA_t - \beta UA_t)}}\right]$$

Where α, β, and A are adjustable positive constants in the system and can be tuned if necessary.

The expression $\left[1 - \frac{1}{Ae^{(\alpha SA_t - \beta UA_t)}}\right]$ approaches '1' quickly with an increase in the number of Successful Accesses and/or a decrease in the number of Unsuccessful Accesses within certain period of time. Notice that our choice of the above expression is for the smooth property of the exponential function and ease of calculation. It turns out that $T_{P,Q} = 0$ if $(\alpha SA_t - \beta UA_t) < 0$. In other words, the trust value of principal *Q* is equal to 0 if its number of Unsuccessful Accesses is greater than the number of Successful Access times of the system *P*. The factor $\left[\frac{SA_t}{SA_t + UA_t}\right]$ indicates the percentage of successful interactions in the whole communication session. We actually exploit the time-based sliding window mechanism [10] to estimate the percentage of successful communications.



4/1	3/2	9/4	7/2	9/1	6/2	7/4	8/3	9/3	8/4	5/2
4/1	3/2	9/4	7/2	9/1	6/2	7/4	8/3	9/3	8/4	5/2

Figure 2. Time-Based Sliding Window Mechanism

A sliding window is a variable-duration window that allows the system to compute different principals' trust value relied on the number of successful access times in a specified number of timing units. Note that the window size could be changed depending on the user's configuration. In Fig. 2, the current window length is presumably configured as a 4-unit sliding window. During the first timing interaction unit, the number of successful and unsuccessful accesses was 4 and 1 respectively. Once a unit of time passes, the window slides one time unit from left to right, eliminating the previous interactions in the first unit from the trust calculation. Hence, very old past history information will not be involved in working out a trust evaluation as time goes by. Under the simple example shown in Fig.2 with α = 1, β = 2, and A = 1, $T_{P,Q} = \left[\frac{23}{(23+9)}\right]\left[1 - \frac{1}{e^{(1.23-2.9)}}\right] = \frac{23}{32}\left[1 - \frac{1}{e^5}\right] \approx 0.70$ for the first interval. However, $T_{P,Q}$ will be changed in the next interaction interval since the number of successful and unsuccessful access times are 9 and 1 which are slightly different from the previous ones: $T_{P,Q} = \left[\frac{28}{(28+9)}\right]\left[1 - \frac{1}{e^{(1.28-2.9)}}\right] = \frac{28}{37}\left[1 - \frac{1}{e^{10}}\right] \approx 0.76.$

4.1.2. Peer Recommendation. Peer Recommendation factor is required when the system has no or not enough information about a principal. Obviously, if there exists certain peer having more interactions with this principal, his suggestion should be likely logical and important for assessing the trust value.

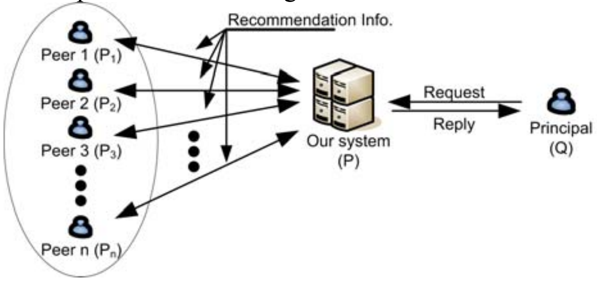


Figure 3. A Peer Recommendation Scenario

Assume that the system was not familiar with this kind of request before so our system *P* has to ask other peers in the environment for their suggestions. In this situation, the system will send multicast a request for comments about the new principal *Q* to its confident community. We denote the time stamp between a principal *Q* and the system *P* as $\tau_{P,Q}$ and τ is the time at which *Q* decides to interact with P. Suppose n is the number of principals currently active in the environment. Let $P_1, P_2, \ldots, P_n$ represent the principals in the space. We also say that principals with high trust values will not send false recommendations. Moreover, let Δτ denote the threshold time interval. Under those assumptions, definition 3.2, and Fig. 3, the trust value for the requesting principal *Q* is defined as follows:

$$T_{P,Q}=\frac{\eta_1 T_{P,P_1}T_{P_1,Q}+\eta_2 T_{P,P_2}T_{P_2,Q}+\eta_3 T_{P,P_3}T_{P_3,Q}+\ldots+\eta_n T_{P,P_n}T_{P_n,Q}}{n} \quad (\mathrm{n} \neq 0)$$

$$\Leftrightarrow T_{P,Q}=\frac{\sum_{i=1}^{n}\eta_i T_{P,P_i}T_{P_i,Q}}{n} \quad (\mathrm{n} \neq 0)$$

Where $\eta_i = Be^{\theta\frac{\Delta\tau_{P_i,Q}}{\Delta\tau}} \in (0,1]$, with $\Delta\tau_{P_i,Q} = \tau_{P_i,Q} - \tau$. B and θ are adaptable positive constants which can be chosen apart to guarantee that $\eta_i \le 1$. For example, we select θ = 1. To establish $\eta_i \le 1$, B must be picked out such that B ∈ $(0, \frac{1}{e^{\frac{\Delta\tau_{P_i,Q}}{\Delta\tau}}}]$. Since $\Delta\tau_{P_i,Q} \le \Delta\tau$, we have $B_{max} \approx 0.46$. Obviously, $T_{P,Q} = 0$ if $n = 0$. In other words, peer recommendation will not be involved in trust evaluation process if there is no peer in the space. Besides, notice that η_i swiftly approaches '1' with increase in the argument $\Delta\tau_{P_i,Q}$. This means that very old and short experiences of peers with the principal in a period of time Δτ should have less weight in trust estimation than the new and long ones. After finishing the trust evaluation phase, we move towards the second phase in order to decide whether to deliver protected resources to the principal.

4.2. Trust-Based Access Control Policy

We design a Trust-based Access Control Policy (TACP) module to describe the constraints such that the end-user's resources are shared in the manner that she would expect. Requesters cannot directly access available resources/services, but get a reference to the TACP. Whenever an ad hoc principal asks to access a resource, the trust evaluation module intercepts its request and estimates its trust value. Once a principal's trust level was quantized by our system, it will be considered as one of two pre-defined states: *Allow* or *Block* with the support of a trust-privacy mapping (definition 3.4) according to specific resource.

We consider the case of Alice's ad hoc supported smart office in which different resources/services, such as printers, fax machines, storage servers, etc are available for sharing. When Alice hosts a teleconference to present the company proposals to her colleagues, she takes her own Pocket PC to access her office's resources, retrieving the necessary files and programs. Once the conference is established, the ad hoc group can also share applications and use a common resource like some ftp server located at Alice's place to upload/download material but not others. This scenario raises access control policy need. On the one hand, Alice's resources have to be protected from illicit accesses from unauthorized members; on the other hand, local resources/services have to be secured from attendees' unauthorized actions. So as to accomplish access control successfully, we show an example of particular access control policy as in Table1.

Table 1. The content of an entry in an access control policy

Order	Resources/ Services	Trust Value Threshold	Action	Comment
01	Printer01	0.3	Allow	Alice's Printer01
02	Fax_MachineA	0.5	Allow	Alice's Fax Machine
03	FTP_ServerB	0.75	Allow	Alice's FTP Server
04	Storage_ServerC	0.90	Allow	Alice's Storage Server
0.5	Any	Any	Block	Any

We demonstrate such a policy through a simple scenario. Assume that the access control model of Alice's system has to process 100 requests with different trust values quantized by the trust estimation engine. Fig. 4 shows how unauthorized and/or malicious queries are blocked with regard to different thresholds chosen by Alice. Two observations are drawn. First, the red line indicates that if Alice uses a weak threshold which is equal to 0.30, the system is only able to filter a very small quantity of malicious requests coming to her system. Second, after adjusting the threshold to 0.75, we see that much more unauthorized accesses have been denied (pointed out by the blue line). As a summary, the important note for ad hoc users in general cases is that spiteful accesses can be easily defeated as long as the threshold in the access control policy is properly configured.

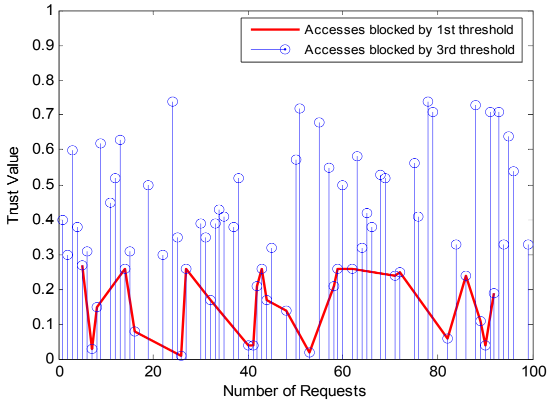


Figure 4. The effect of access control policy with different trust value thresholds

5. Conclusions and Future Work

Security for ad hoc networks is really a fascinating and challenging research topic. The inherent features of the ad hoc environment such as invisibility and mobility raise new difficult and stimulating tasks. Access control, like other security technologies, requires appropriate management and operation in order to safely protect resources or services.

This paper discusses our plan to design a trust-based access control model for MANETs. In this study, we have introduced a trust-based access control model by taking uncertainty of trust into account with a precise computation model. Additionally, we apply customizable access control policy to efficiently handle malicious principals. The calculation of trust depends on the time of last accesses and peer reputation common to the entities. Besides, several tuning parameters and options are suggested which can be technically adapted to meet the requirements of a pervasive computing space. A highly secure and private system can fit these variables such that only a small number of principals with appropriate reputation and recommendation are allowed to gain sensitive resources.

At last, we believe that there is lots of work to do in the implementation area. Our future research plan includes building up major modules, such as trust evaluation module and access control policy module, putting our findings into practice, allowing ad hoc users to differentiate sharing their resources confidently.

Acknowledgement

This work was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2006-C1090-0602-0002). Dr Sungyoung Lee is the corresponding author.

References

[1] P. Bellavista, A. Corradi, and C. Stefanelli, "The Ubiquitous Provisioning of Internet Services to Portable Devices," IEEE Pervasive Computing, Vol. 1, No. 3, July-September 2002.

[2] N. Aboudagga, M. T. Refaei, M. Eltoweissy, L. A. DaSilva, and J. J. Quisquater, "Authentication Protocols for Ad Hoc Networks: Taxonomy and Research Issues," in Proc. 1st ACM international workshop on Quality of service & security in wireless and mobile networks, Quebec, Canada, pp.96-104, 2005.

[3] E.C. Lupu, D.A. Marriott, M.S. Sloman and N. Yialelis, "A policy based role framework for access control," First ACM/NIST Role Based Access Control Workshop, Gaithersburg, USA, Dec. 1995.

[4] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role based access control models," IEEE Computer, 29(2):38-47, Feb. 1996.

[5] A. A. Pirzada and C. McDonald, "Kerberos Assisted Authentication in Mobile Ad-hoc Networks," in Proc. 27th Australasian Computer Science Conference (ACSC'04), Dunedin, New Zealand, 26(1):41-46, January 2004.

[6] Lalana Kagal, Tim Finin, and Anupam Joshi, "Trust-based security in pervasive computing environments," IEEE Computer, 34(12):154–157, December 2001.

[7] Maria Moloney and Stefan Weber, "A Context-aware Trust-Based Security System for Ad Hoc Networks," appears in Workshop of the 1st International Conference on Security and Privacy for Emerging Area in Communication Networks, pp. 153-160, 2005.

[8] T. Grandison and M. Sloman, "A survey of trust in Internet applications," IEEE Communications Surveys and Tutorials, vol. 3, no. 4, pp. 2–16, 4th Quarter 2000.

[9] M. Blaze, J. Feigenbaum, and J. Lacy, "Decentralized trust management," in Proc. IEEE Symposium on Security Privacy, pp. 164–173, 1996.

[10] Riaz Ahmed Shaikh et al., "Intrusion Tolerant Group-based Trust Management Scheme for Wireless Sensor Networks", submitted for publication.