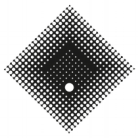


Multi-level delegations with trust management in access control systems

Li, Min, Sun, Xiaoxun, Wang, Hua and Zhang, Yanchun (2012) Multi-level delegations with trust management in access control systems. Journal of Intelligent Information Systems, 39 (3). pp. 611-626. ISSN 0925-9902 (print), 1573-7675 (online)

Multi-level Delegations with Trust Management in Access Control Systems

Min Li[1], Xiaoxun Sun[2], Hua Wang[1], Yanchun Zhang[3]

[1]Department of Mathematics & Computing

University of Southern Queensland, Australia

[2]Australian Council for Educational Research

[3]School of Engineering and Science

Victoria University, Australia

Abstract

Delegation is a mechanism that allows one agent to act on another's privilege. It is important that the privileges should be delegated to a person who is trustworthy. In this paper, we propose a multi-level delegation model with trust management in access control systems. We organize the delegation tasks into three levels, Low, $Medium$, and $High$, according to the sensitivity of the information contained in the delegation tasks. It motivates us that the more sensitive the delegated task is, the more trustworthy the delegatee should be. In order to assess how trustworthy a delegatee is, we devise trust evaluation techniques to describe a delegatee's trust history and also predict the future trend of trust. In our proposed delegation model, a delegatee with a higher trust level could be assigned with a higher level delegation task. Extensive experiments show that our proposed multi-level delegation model is effective in accurately predicting trust and avoiding sensitive information disclosure.

1 Introduction

In a multi-agent system, delegation is the primary mechanism of inter-agent collaboration and cooperation [8, 9, 12, 16]. The basic idea behind the delegation is that some active entity in a system delegates authority to another active entity to carry out some functions on behalf of the former. For example, when an agent is unable to perform a task due to sickness, s/he may delegate the privileges to another agent so that the latter agent can use the privileges to complete the task on time. It is through the delegation that the agent is able to function effectively. Normally, a *delegator* in a delegation is an agent that delegates a certain task to another agent or a group of agents. The delegator has the permission to perform a certain action and also the ability to further delegate this right. A *delegatee* is the one who has been delegated to execute a delegated task.

Role-based delegation based on role-based access control (RBAC) has been proven to be a flexible and useful access control for information sharing in distributed collaborative environment [3, 28, 15]. In contrast to normal access right administration operations, which are performed centrally, delegation operations are usually performed in a distributed manner. Security of delegation becomes one big issue that has received attention during the past few years in distributed systems. In this paper, we are interested in the delegation of tasks (task-delegation) as compared with the delegation of rights only (right-delegation) described in [1]. Both task-delegation and rights-delegation involve the release of rights from one principal to another. However, in the case of task-delegation we consider the situation in which entity issues an imperative command to another entity to perform the delegated task within the broad area of security.

Our line of reasoning is motivated by the real-world situations in which one entity delegates some rights to a second entity with the explicit command to complete a given task validly and securely. Loosely defined, a task consists of a number of computational operations to be performed based on some data which may be sensitive and insecure to be misused or disclosed to public. Here, we organize delegated tasks into three different levels according to their sensitivity as shown in Table 1. For the simplicity of discussion, in this paper we consider three-level partitions of delegated tasks, which are Low, $Medium$, and $High$. The classification standard is flexible, which can be determined by a delegator with his or her

Task level	Information	Properties
Low	Public	The information is not sensitive and can be delegated to anyone.
Medium	Not public partially sensitive	The information is partially sensitive and should be delegated to reliable delegatees.
High	Not public totally sensitive	The information is totally sensitive and should be delegated to someone with higher reliability.

Table 1: The classification of delegation tasks

subjective preference. A *Low* task level indicates the delegation task does not include sensitive information or resources that can cause a breach. The information in a *Low* level of the delegation task is the public information that can be delegated to anyone for information sharing. The tasks in the $Medium$ level contain the information that is partially public and partially sensitive. When referring to the delegation, there should be a higher requirement on the reliability of the delegatee, since the more reliable the delegatee is, the less chance the sensitive information would be misused, and the more likely the delegation task could be accomplished successfully. The $High$ task level indicates the delegation task is very important and contains highly sensitive information and requires that the delegatee should be totally trustworthy.

Essentially, a delegation operation could temporarily change the access control state so as to allow an agent to use another agent's access privileges. Due to its effect on the access control state, delegation may lead to violation of security policies. More precisely, information breaching may happen even during the delegation phase. Any information, the loss, misuse, or unauthorized access to or modification of which could adversely affect the privacy to which individuals are entitled. Thus, risk during the delegation must not be overlooked, and more sophisticated methods are needed to create a secure delegation system. More specifically, delegation policies may depend on private aspects concerning both the delegatee's reliability and the sensitivity of the delegated tasks.

1.1 Motivation

In an open environment, the entities are customarily alien to each other. When entering into a delegation, the delegator is entering into an uncertain interaction in which there is a risk of failure due to the delegation decisions. In other words, a given delegatee may not be reliable for the delegated task, especially, when sensitive information is included in the delegation tasks, the delegator's privacy may be breached because of the unreliability of the delegatee. For example, if the task being delegated is a goal comprising of multiple tasks and requiring access to multiple resources and sensitive information, the delegation in this case should be very cautious, since the failure of the delegation has a considerable influence on privacy disclosure. Therefore, when delegating a task, the choice of the cooperative partner plays an important role in determining whether the task would be fulfilled successful or not. In order to operate effectively, delegators need some mechanisms for finding reliable partners, and this requirement could be satisfied with the help of trust. Trust is well recognized as a means of assessing the risk of cooperating with others [6, 10, 14, 25]. There are two main categories of trust: experience-based and recommendation-based [18, 26]. In the former category, agents assess the trust solely based on their own experience; in the latter, trust is evaluated based on information provided by others (typically in addition to individual experience). Within this trust evaluation mechanism, a final trust value is computed to reflect the general trust status of every service provider. However, such a single trust value cannot reflect the real trust status very well. For example, assume the trust values are in the range of [0,1]. A person with a higher trust value 0.9 may behave worse in future than the one with the trust value 0.6. This simple example demonstrates that the single-value trust evaluation approach can not reflect the changes of the trust any more. Trust trend evaluation becomes important in order to indicate whether the trust will become better or worse in the forthcoming cooperation. Therefore, new effective trust evaluation approaches are required to provide more precise trust information that could indicate to what extent and during which period a delegatee is reliable and trustworthy.

Even though delegation is well recognized as a very useful component of access control systems [3, 5, 28], to our best knowledge, no current work has performed in-depth study on how to manage a delegation in a secure manner. Typically, we are facing the following

challenges in developing a secure multi-level delegation model by taking trust into account:

Challenge One: Since the sharing of the sensitive information must only be restricted to trustworthy parties, how to develop effective trust evaluation approaches to provide more precise trust information?

Challenge Two: Facing with the fact that the delegation tasks in different levels require different reliability of delegatees, how to build the projection between the reliability of the delegatee's and the sensitivity of the delegated tasks, and further construct a secure multi-level delegation model?

1.2 Contributions

Confronting with these challenges, we provide the following solutions.

• Decomposing delegation tasks into three different levels according to the sensitivity of each delegation task. Each level has different requirement of reliability of cooperation partners.

• Proposing a new effective trust evaluation technique which considers both trust values and trust trend. The trust value provides an indication for the final trust level while, the trust trend value is used to predict the future trend of trust.

• Building a projection between the reliability of the delegatee and the sensitivity of delegated tasks, which leads to a secure multi-level delegation model.

• Investigating the effectiveness of our proposed multi-level delegation model and the experimental studies confirm the advantages of our model in terms of accurate prediction and sensitive information protection.

The remainder of the paper is structured as follows. In Section 2, the new trust evaluation approach is proposed by combining trust values and trust trend together to predict a delegatee's trustworthiness. In Section 3, we propose a multi-level delegation model with trust management and discuss several different delegation types. We show our experimental results in Section 4 and provide a brief survey of related work in Section 5. Finally, we conclude the paper in Section 6.

2 The trust evaluation

The notion of trust is well recognized as a means of assessing the risk of cooperating with others [6, 25, 20]. In a delegation, it is important to tell delegators to what extent a delegatee is trustworthy for the delegated task. Corresponding to the different levels of delegated tasks, in this section we organize the trust into three trust levels, in which delegators could evaluate the trustworthiness of delegatees.

Trust represents an agent's estimate of how likely another is to fulfil its commitments. Trust influences the delegators attitudes and actions, but can also have effects on the delegatee and other elements in the environment. As discussed before, a trust value can be calculated to provide more precise indication of the trust history to a delegator. However, it is not enough to indicate the real trust status of a delegatee very well, i.e., the single-value approach cannot reflect changes of the trust trend. In this paper, we adopt two interpretations of trust. One is to view trust as the perceived reliable history of somebody, called "reliability trust", while the other is to view trust as a trend of trust changes in a given period, called "future trust".

Definition 1 (Reliability trust). *Reliability trust is the trust status of individuals dependent on his/her history behavior.*

As the name suggested, reliability trust can be interpreted as the subjective probability of someone by performing a given action on which its success lies. In our previous work [13], we evaluate the reliability trust in three steps: (1) Calculate the trust value based on histories; (2) Calculate the trust value from recommendations; (3) Combine the observed trust values from histories and recommendations. With this approach, we can obtain a delegatee's reliability trust value. However, trust can be more complex. Future trust aims to capture the changes of trust trend in the forthcoming future. Namely, given a set of delegatees with the same trust value, the one which is becoming better is more desirable to delegators and more reliable to fulfill the delegated work well.

Definition 2 (Future trust). *Future trust is a general trend of trust changes which could be useful to predict the future trust level of service quality.*

In order to evaluate the future trust, we refer to the idea of exponential regression [22]. In this paper, we introduce a weighted exponential regression method to evaluate the trust trend

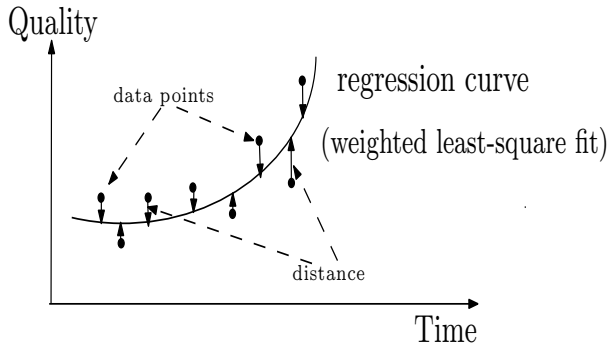


Figure 1: Weighted least-squares exponential regression

(shown as Figure 1). This method is used to obtain the best exponential fit from a set of given data points. This best exponential fit is characterized by the sum of weighted squared residuals with its least value, where a residual is the difference between a data point and the regression curve. Once obtaining the exponential regression, the gradient at each data point can be taken as our future trust value. Now we introduce the trust trend evaluation method.

Let $(t_1, q_1), (t_2, q_2), \ldots, (t_n, q_n)$ denote the given data points in a certain period, where $q_i (q_i \in [0, 1])$ is the service quality value at time $t_i (t_i < t_{i+1}, 1 \leq i \leq n)$. Then the exponential regression can be represented as

$$q = a_0 e^{a_1 t} + a_2 \tag{1}$$

where a_0, a_1 and a_2 are constants to be determined, specially, the product of a_0 and a_1 indicates the trust trend value. As the distance from point (t_i, q_i) to the regression curve is

$$d_i = |q_i - (a_0 e^{a_1 t_i} + a_2)| \tag{2}$$

Based on the method of weighted least squares, we let $w(i)$ be the weight function for the service quality q_i at the i^{th} service $(i = 1 \cdots n)$. The choice of $w(i)$ could be flexible. Any monotonic increasing function could be a candidate of $w(i)$. For simplicity, in this paper, we adopt $w(i) = i^\beta$, $(1 \leq i \leq n, \beta \geq 1)$ as our weight function. Thus, the sum of squares of the

distance can be calculated as follows:

$$S = \sum_{i=1}^{n} w(i)^2 d_i^2 = \sum_{i=1}^{n} w(i)^2 (q_i - (a_0 e^{a_1 t_i} + a_2))^2 \tag{3}$$

Now our task is to minimize the sum of the distance S with respect to the parameters a_0, a_1 and a_2, with the method of undetermined coefficients.

Since function S is continuous and differentiable, based on Lagrange Multiplier method [19], the minimization point of S makes the first derivative of function S be zero. Thus, we differentiate S with respect to a_0, a_1 and a_2, and set the results to zero, which gives

$$\frac{\partial S}{\partial a_1} = -2 \sum_{i=1}^{n} w(i)^2 (q_i - (a_0 e^{a_1 t_i} + a_2))(a_0 t_i e^{a_1 t_i}) = 0 \tag{4}$$

$$\frac{\partial S}{\partial a_2} = -2 \sum_{i=1}^{n} w(i)^2 (q_i - (a_0 e^{a_1 t_i} + a_2)) = 0 \tag{5}$$

and

$$\frac{\partial S}{\partial a_0} = -2 \sum_{i=1}^{n} w(i)^2 (q_i - (a_0 e^{a_1 t_i} + a_2)) e^{a_1 t_i} = 0 \tag{6}$$

Equations (4), (5) and (6) can be solved for the unknown a_0, a_1 and a_2. Thus, based on the method of weighted least squares exponential regression, we can obtain the trust trend value $a_0 a_1 (a_0, a_1 \in R)$. The trust trend value shows a general trend of changes of trust in the near future, which is important when we choose a delegatee with serious caution. If $a_0 a_1 > 0$, it indicates that the future trust is up-going, whereas, $a_0 a_1 < 0$ indicates that the future trust is dropping; and $a_0 a_1 = 0$ indicates the future trust remains unchanged.

Both reliability trust and future trust reflect different trust status about the individuals on whom the delegator depend for the delegation task. Reliability trust is most naturally measured as a degree of reliability, which is expressed as a continuous function mapped into [0,1], whereas future trust indicates the trend of trust changes, which ranges from $-\infty$ to $+\infty$. To work efficiently, we combine reliability trust and future trust into different rust levels to

illustrate the trustworthiness of a delegatee. To be consistent with delegated task levels, three trust levels are organized through the following projection:

Definition 3 (Trust level). *Let T be the set of reliability trust values and TT be the set of future trust values. The F function projects reliability trust and future trust into three different trust levels.*

$$F : T \times TT \rightarrow \{L, M, H\}$$

where L, M, H refers to Low, Medium and High trust levels.

High trust level denotes the person at this level is highly trusted, which means not only his final trust value is high but also the trust trend is up-going. *Low* level denotes the person is less trusted, where his final trust value is low, also the trust trend is dropping. *Medium* level is the intermediate state. So the trust level assignments can be further explained as follows:

$\forall t \in T, a_0a_1 \in TT$

- $F(t, a_0a_1) = L$, if $t \in (0, 0.5)$ and $a_0a_1 \in (-\infty, 0)$
- $F(t, a_0a_1) = M$, if $t \in [0.5, 1)$ and $a_0a_1 \in (-\infty, 0]$; or $t \in (0, 0.5]$ and $a_0a_1 \in [0, +\infty)$
- $F(t, a_0a_1) = H$, if $t \in (0.5, 1)$ and $a_0a_1 \in (0, +\infty)$

Until now, each delegatee is companied with a trust level, which could indicate to what extent the delegatee is reliable. So far, the problem left is to build the delegation model based on the evaluation of trust. Our idea is that the delegatee who is trusted at a greater degree would have a higher probability to complete the delegated task than a delegatee with a lower trust level. The formalized delegation model is described in the next section.

3 The multi-level delegation model

Delegation has received significant attention from the research community in recent years. A number of delegation models have been proposed [5, 9, 21, 12] and most of them are for Role-Based Access Control (RBAC). A few of research works related to introducing subjective

trust into delegation model have been reported [7, 24]. In this section, we build a multi-level delegation model with trust management.

3.1 The delegation model

Delegation is a mechanism that allows an agent A to act on another agent B's behalf by making B's access rights available to A. Suppose a task is delegated from one to another, the latter actually gets the access right to work on this task. It needs to be organized as an important mechanism to provide resiliency and flexibility in access control systems.

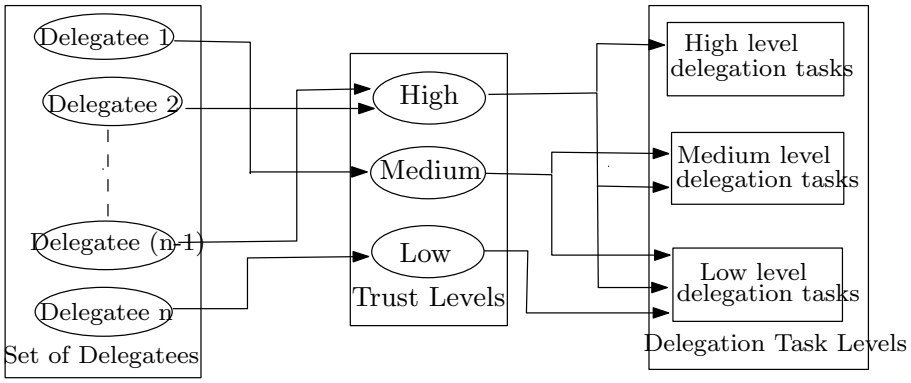


Figure 2: Distribution of delegations based on trust levels

Since delegation tasks are divided into three different levels, it is important to address how to distribute these tasks to delegatees based on their trust levels? The idea is that a delegatee in the high trust level can be assigned with the delegation task of all levels, which are Low, $Medium$, and $High$. The delegatee in the medium trust level can be assigned with Low and $Medium$ level tasks, while the delegatee in the low trust level can only be assigned with Low level tasks. In this case, all delegated tasks are assigned in a hierarchal style, since the delegatee in a higher trust level is more trustworthy and is more likely to finish a higher level delegation task than the one in a lower trust level. The distribution of delegation is shown in Figure 2. In order to describe the delegation in a precise manner, we focus on a specific model about how delegatees gain the access right.

Definition 4. *Let D_r, D_e, D_t be the set of delegators, delegatees, and delegated tasks respectively. $Level = \{L, M, H\}$ is the set of trust levels (or delegated task levels). A delegation relationship is defined as $DR \subseteq D_r \times D_eL \times D_tL \times \{g, t\}$, where $D_eL \subseteq D_e \times Level$ is the membership between delegatees and trust levels, $D_tL \subseteq D_t \times Level$ is the membership between delegated tasks and task levels, and g, t refers to grant or $transfer$ operation.*

The delegatee-trust level membership D_eL denotes that each delegatee is assigned with different trust levels and D_tL denotes that each delegated task is assigned with different task levels. For example, the delegation relationship $(d_r, (d_e, L), (d_t, M), g) \in DR$ indicates that delegator d_r has delegated the L level task d_t to delegatee d_e in the M trust level via a $grant$ operation, while $(d_r, (d_e, L), (d_t, M), t)$ indicates that delegator d_r has delegated L level task d_t to delegatee d_e in M trust level via a $transfer$ operation. The difference between $grant$ and $transfer$ is shown as follows. A delegation operation is essentially an access control state transition operation, which takes one of the following three forms:

- $grant(d_r, (d_e, l), (d_t, l))$: delegator d_r $grants$ the access of l level delegation task to delegatee d_e who is in l trust level. After the delegation operation, d_e gains the access right to d_t and d_r still keeps d_t, where $l \in \{L, M, H\}$.
- $trans(d_r, (d_e, l), (d_t, l))$: delegator d_r $transfers$ the access of l level delegation task to delegatee d_e who is in l trust level. After the delegation operation, d_e gains the access right to d_t and d_r temporarily loses d_t, where $l \in \{L, M, H\}$.
- $revoke(d_r, (d_e, l), (d_t, l))$: delegator d_r $revokes$ the delegated task d_t from delegatee d_e.

Note that a delegator can grant or transfer different level tasks to delegatees, and only the corresponding delegator can revoke the delegated task from the delegatee. For example, $grant(Alice, (Bob, H),$ (*read all emails*, $M), g)$ means Alice delegated the $Medium$ level task "read all emails" to Bob with $High$ trust level via a $grant$ operation, while after the delegation Bob gains the access right to all emails and *Alice* still keeps the access right on all emails. However, $transfer(Alice, (Bob, H),$ (*read all emails*, $M), t)$ means *Alice* delegated

the $Medium$ level task "read all emails" to Bob with $High$ trust level via a $transfer$ operation, and after the delegation *Alice* temporarily loses the access right to all emails. Definitely, only *Alice* could revoke the delegated task "read all emails" from Bob.

Since delegation is performed in a distributed manner, in the sense that everyone may perform delegation operations, it is undesirable to allow a delegator to delegate the tasks in a completely unrestricted way. Delegation operations are thus subject to the control of authorization rules, which takes one of the following three forms:

- $can\_grant(cond, (d_t, l))$: a delegator who satisfies condition *cond* can grant the l level task d_t to other delegatees, where $l \in \{L, M, H\}$, *cond* is an expression formed through using the binary operators $\vee$ and $\wedge$, the unary operator $\neg$, and parentheses.

- $ca\_transfer(cond, (d_t, l))$: a delegator who satisfies condition *cond* can transfer the l level task d_t to other delegatees, where $l \in \{L, M, H\}$.

- $can\_receive(cond, (d_t, l))$: a delegatee who satisfies condition *cond* can receive the l level task d_t from other delegators, where $l \in \{L, M, H\}$.

For example, the rule $can\_receive$(Clerk $\wedge$ M, ("read the documents", M)) states that anyone who is at the $Medium$ trust level and a member of Clerk can receive the $Medium$ level task "read the document".

3.2 Types of delegations

Delegation models could be complicated. To create a delegation model, one needs to decide on a number of features, such as whether the delegation is dated and valid only for a certain period of time, whether delegatees can further delegate the tasks to others and so on. Retention period refers to during which time the delegation is valid. We denote TI as the set of time intervals. Different types of delegations contended in our delegation model are discussed as follows.

- ***Time Bound Delegation*** $TBD \subseteq TI \times D_r \times D_eL \times D_tL$: It is a delegation that is valid only for a certain time period, where T is the set of time intervals.

For example, delegation *([12/06/2008, 10/08/2008], Alice, (Bob, H), (read all emails, M))* denotes that this delegation is only valid between 12/06/2008 and 10/08/2008 and only during this period, Bob has the access right to all emails.

- ***Group Delegation*** $GD \subseteq D_r \times D_eL \times D_tL$: It can be used to delegate access rights to a group of delegatees who satisfy certain conditions.

  For example, delegation*(Alice, (Employee, M), (read all emails, M))*denotes that Alice delegates the $Medium$ level task "read all emails" to a group of employees who are in $Medium$ trust level.

- ***Action Restricted Delegation*** $ARD \subseteq D_r \times D_eL \times D_tL \times CD$: This forces the delegatee to satisfy certain conditions before the delegated task can be carried out, where CD is the set of conditions.

  For example, delegation*(Alice, (Employee, M), (read all emails, M), (age(24), name(Bob)))* states that only employees who is in $Medium$ trust level, aged 24 and named Bob can gain the access right to "read all emails".

- ***Re-delegable Delegation*** $RD \subseteq D_r \times D_eL \times D_tL \times \{Ture, False\}$: In this delegation, $Ture$ means the delegated task could be re-delegated to others, while $False$ means not.

  For example, delegation*(Alice, (Employee, M), (read all emails, M), true)* denotes that the delegatee is allowed to further delegate the task.

Delegation policy: Delegation policies describe rules for delegation of the rights. A rule for delegation would be checking that an agent has the ability to delegate before allowing the delegation to be approved. A policy can be viewed as a set of rules for a particular domain that defines what permissions a user has and what permissions she/he can obtain. A policy also contains basic or axiomatic rights that all individuals possess.

4 Experimental evaluations

The main goals of the experiments are two-fold. First, we study the precision of our trust model in predicting the trend of the trust. Second, we investigate the effectiveness of our

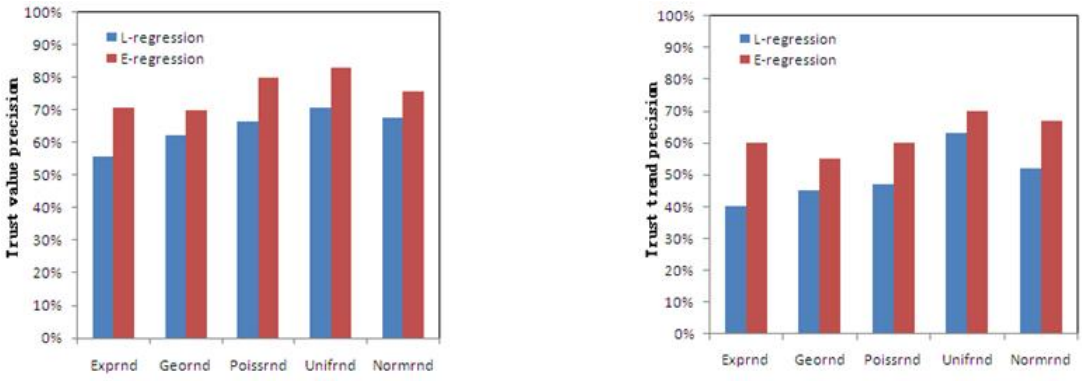
proposed multi-level trust-based delegation model in terms of *disclosure rate*.

No. of data set	Probability distribution function
1	exponential distribution (Exprnd)
2	geometric distribution (Geornd)
3	Poisson distribution (Poissrnd)
4	Uniform distribution (Unifrnd)
5	Normal distribution (Normrnd)

Table 2: Distributions of the data sets

Trust value and its trend evaluation: In this set of experiments, we compared the precision of both the trust value and trust trend prediction with the existing method proposed in [11]. We denote *E-regression* as the exponential regression model proposed in this paper and *L-regression* as the regression model of [11]. In order to evaluate the precision of two approaches, we generate five data sets with five different probability distribution functions as our test data, and each data set contains 5000 records, and each record is in the form of (x, y), where $1 \leq x \leq 5000$ and $0 \leq y \leq 1$. Table 2 shows the probability distributions of each data set. Different metrics are adopted in evaluating the precision of the trust value and trust trend. For evaluating the precision of trust value, each data set is first divided into training and testing sets, and both regression models are trained by the training sets and tested by testing sets. If the predicted trust value is t_{pre} and the actual trust value is t_{act}, then the precision is calculated as $1 - \frac{|t_{pre} - t_{act}|}{t_{act}}$. The higher the value is, the more precise the predicted trust value is. To evaluate the precision of the trust trend, we use the metric named *vector angle*, which is to compute the angle between two vectors(trends). The vector angle is defined to be the angle ϕ between 0 and 180 degrees that satisfies the relationship: $\cos\phi = \frac{t_1 \cdot t_2}{|t_1||t_2|}$, where $|\cdot|$ refers to the vector length and the numerator denotes the inner product of the trends t_1 and t_2. The more close the cosine value is to 1, the more similar two trends are. To reduce the randomness, we run the evaluation for 1000 times for each data set to obtain the average.

The evaluation results are shown in Figure 3. Figure 3(a) displays the precision of the trust value of both regression models under five different distributed data sets. We can easily see that the average precision of our proposed exponential regression model is around 70%, which

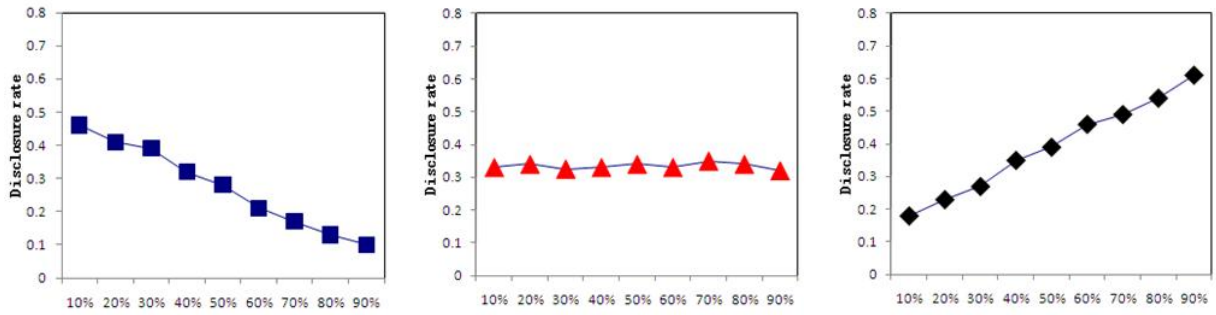


(a) (b)

Figure 3: (a) The precision of the trust value; (b) The precision of the trust trend.

is superior to the linear regression model over all the five different distributed data sets. Figure 3(b) reports the precision of the trust trend for both regression models. From the graph, the exponential regression model brings with us more accurate trust trend compared with the linear regression model over all the five different distributed data sets. The precision of the trust trend for the linear regression model sometimes is pretty low, for example, only 40% for the exponential distributed data set. This is because sometimes the linear regression model predicts the opposite trust trend, which makes the cosine value negative and hence dragging down the average precision. Overall, the exponential regress model proposed in this paper has more accurate precision in predicting both the trust value and trust trend compared with the linear regression model.

Effectiveness: Having verifying the precision of our technique, we proceed to test its effectiveness. In this set of experiments, we use the *disclosure rate* to measure the effectiveness of our proposed multi-level delegation model. We are going to use H, M and L to denote the High, Medium and Low level in the classification of delegation tasks or the trust level of the delegatees, separately. Recall our trust-based delegation model, if a data requester is in High trust level, then s/he can be assigned with H, M or L level tasks; if the data requester is in Medium trust level, then s/he can be assigned with M or L level tasks; Otherwise, the data requester can only be assigned with L level tasks. Suppose there are n data requesters, among which there are n_H data requesters are with High level of trust, n_M requesters are with



(a) (b) (c)

Figure 4: Disclosure rate comparison when varying (a) the number of H levels; (b) the number of M levels; (c) the number of L levels.

Medium level of trust, and n_L are with Low level of trust, where $n_H + n_M + n_L = n$. In this case, the requesters could totally access $3n_H + 2n_M + n_L$ delegation tasks, which indicates the number of secure delegations. Consider the situation where there is no specification of trust levels, the data requester, whatever the trust value and trend s/he holds, could receive three possible task assignments. Then it would be $3(n_H + n_M + n_L)$ delegations, and among those, there will be $3(n_H + n_M + n_L) - (3n_H + 2n_M + n_L)$ insecure delegations. Thus, we define the *disclosure rate* as $1 - \frac{3n_H+2n_M+n_L}{3(n_H+n_M+n_L)}$. The lower the rate is, the more secure the delegation is. We randomly generate n data requesters, and evaluate how the number of data requesters in H, M or L levels affect the disclosure rate. In order to reduce the randomness, we run the each test for 500 times for each data and use the average to mark the graph.

The results are shown in Figure 4. Figure 4(a) displays the disclosure rate by varying the portion of H from 10% to 90%. From the graph, we can see that the disclosure rate is decreasing as the amount of H increases. This is expected, since the more the H level requesters are, the less the insecure delegations are and the lower the disclosure rate is. Figure 4(b) describes the disclosure rate by varying M from 10% to 90%. The graph shows that the disclosure rate almost remains unchanged with the increased portion of M. Figure 4(c) reports the effect of L on the disclosure rate. When varying the portion of L from 10% to 90%, the disclosure rate is ascending. It indicates that the more L level requesters are assigned to delegation tasks, the higher chances for the sensitive information to be disclosed. However,

our proposed delegation model could better avoid the sensitive information disclosure by specifying requesters' trust levels. Therefore, in this case, our proposed multi-level delegation model is superior to the traditional delegation model.

5 Related work

Delegation has received considerable attention from the research community. In [3], Barka and Sandhu proposed a framework for role-based delegation models (RBDM), which identifies a number of characteristics related to delegation. Example characteristics are monotonicity, totality, and levels of delegation.

There exists a wealth of delegation models in literature [27, 28, 5]. Zhang et al. [27] presented a role-based delegation model called RDM2000. Their model supports the specification of delegation authorization rules to impose restrictions on which roles can be delegated to whom. Zhang et al. [28] proposed a role-based delegation model called PBDM, which supports both role and permission level delegation. Their model controls delegation operations through the notion of delegatable roles such that only permissions assigned to these roles can be delegated to others. In [5], Crampton and Khambhammettu proposed a delegation model that supports both grant and transfer. Atluri and Warner [2] studied how to support delegation in workflow systems. They extended the notion of delegation to allow conditional delegation, where conditions can be determined on time, workload and task attributes. One may specify rules to determine under what condition a delegation operation should be performed.

All of the above work focus on the modeling and management of delegation, while our paper focuses on developing a secure delegation model in access control systems. More importantly, none of the above work discusses the trust relationship between delegators and delegatees, but our delegation model is founded on trust. We also investigate the effectiveness of our proposed multi-level delegation model and the experimental results confirm the advantages of our model in privacy protection.

Trust evaluation is a recent approach for access control systems that enables resource requesters and providers in open systems to establish trust. Bonatti and Samarati [4] proposed

a framework based on a policy language and an interaction model for regulating access to network services. Their trust establishment framework uses logical rules for accessing services and avoiding the unnecessary disclosure of sensitive information. Winsborough and Li [23] introduced the Trust Target Graph (TTG) protocol for conducting trust negotiation. A particular emphasis of their work was protection against leaking sensitive information during a trust negotiation. PeerTrust [17] is a trust management system that uses a simple and expressive policy language based on distributed logic programs. PeerTrust agents perform automated trust negotiation to obtain access to sensitive resources. However, these studies are more focus on trust negotiation policies rather than build trust evaluation approaches. In our paper, we organize trust into different trust levels based on trust values and trust trend. The trust value depicts the history trust, while trust trend depicts the future change of trust. Moreover, we apply trust levels to delegation and develop a multi-level delegation model.

6 Conclusions and future work

In this paper, we propose a multi-level delegation model with trust management, where both delegation tasks and trust are organized into three levels. The delegation task levels are classified according to the information sensitivity, while, the trust levels combine trust values and trust trend together to indicate to what extent a delegatee is reliable or trustworthy. Our multi-level delegation model allows that a delegatee in a higher trust level can be assigned with a higher level of task. In the experimental evaluations, we study the precision of our trust model in predicting the trend of the trust and investigate the effectiveness of our proposed multi-level delegation model in terms of information disclosures.

This work motivates several directions for future research. First, since delegation operations could temporarily change the access control state so as to allow an agent to use another agent's access privileges, colluding users may abuse the delegation support of access control systems to circumvent security policies, such as separation of duty. We are intend to consider an enhanced form of delegation in order to avoid collusion in our future work. Second, we did not discuss much on the revocation of delegation. It is interesting to develop a revocation model to protect security under our multi-level delegation model.

References

[1] M. Abadi, M. Burrows, B. Lampson, and G. Plotkin, A calculus for access control in distributed systems, Technical Report 70, Digital Systems Research Center, February 1991.

[2] V. Atluri, and J. Warner, Supporting conditional delegation in secure workflow management systems. In: SACMAT 2005: Proceedings of the tenth ACM symposium on Access control models and technologies, pp. 49-58. ACM Press, New York (2005)

[3] E. Barka, and R. Sandhu, Framework for role-based delegation models. In: ACSAC 2000: Proceedings of the 16th Annual Computer Security Applications Conference, Washington, DC, USA, p.168. IEEE Computer Society Press, Los Alamitos (2000)

[4] P. Bonatti, and P. Samarati, A Unified Framework for Regulating Access and Information Release on the Web. In Journal of Computer Security, 10, 3, (2002), 241-271.

[5] J. Crampton, and H. Khambhammettu, Delegation in role-based access control. In: Proceedings of 11th European Symposium on Research in Computer Security (2006)

[6] E. Damiani, S. D. C. di Vimercati, S. Paraboschi, P. Samarati, and F. Violante. A reputation based approach for choosing reliable resources in peertopeer networks. In Proceedings of ACM CCS02, pages 207-216, Washington DC, USA, November 2002.

[7] N. Griffiths, Task Delegation using Experience-Based Multi-Dimensional Trust, In the Proceedings of the Fourth International Conference on Autonomous Agents and Multi-agent Systems (AAMAS-05), Utrecht, The Netherlands, 2005, pp. 489-496.

[8] T. Hardjono, T. Chikaraishi, and T. Ohta, Secure Delegation of Tasks in Distributed Systems. In Proceedings of the 10th International Symposium on the TRON Project, Los Alamitos, California, USA, 1993.

[9] J.B.D. Joshi, and E. Bertino, Fine-grained role-based delegation in presence of the hybrid role hierarchy. In: SACMAT 2006: Proceedings of the eleventh ACM symposium on Access control models and technologies, pp. 81-90. ACM Press, New York (2006)

[10] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in P2P networks. In Proceedings of the 12th International WWW Conference, Budapest, Hungary, May 2003.

[11] L. Li, Y. Wang and V. Varadharajan, Fuzzy Regression Based Trust Prediction in Service-Oriented Applications, the Sixth International Conference on Autonomic and Trusted Computing (ATC-09), Brisbane, Australia, 7-9 July, 2009.

[12] M. Li, and H. Wang , ABDM: An Extended Flexible Delegation Model in RBAC, Accepted by the IEEE 8th International Conference on Computer and Information Technology (CIT'2008), July 8-11, 2008, Sydney, Australia.

[13] M. Li, H. Wang and D. Ross. Trust-based Access Control for Privacy Protection in Collaborative Environment, to appear in the 2009 IEEE International Conference on e-Business Engineering (ICEBE 2009), Macau, China, October, 2009.

[14] S. Marti and H. Garcia-Molina. Limited reputation sharing in P2P systems. In Proceedings of ACM EC04, pages 91-101, New York, USA, May 2004.

[15] S. Na, and S. Cheon, Role delegation in role-based access control. In: RBAC 2000: Proceedings of the fifth ACM workshop on Role-based access control, pp. 39-44. ACM Press, New York (2000)

[16] T. J. Norman, and C. A. Reed, A Model of Delegation for Multi Agent Systems. In M. d'Inverno, M. M. Luck, M. Fisher and C. Preist (editors), Foundations and Applications of Multi Agent Systems, volume 2403 of Lecture Notes in Artificial Intelligence, Springer-Verlag, pages 185-204, 2002.

[17] W. Nejdl, D. Olmedilla, and M. Winslett, PeerTrust: Automated Trust Negotiation for Peers on the Semantic Web. In Proceedings of the Workshop on Secure Data Management in a Connected World (SDM 04) in conjunction with 30th International Conference on Very Large Databases, Aug./Sept. 2004.

[18] S. D. Ramchurn, C. Sierra, L. Godo, and N. R. Jennings. A computational trust model for multi-agent interactions based on confidence and reputation. In Proc. of the 6th Int. Workshop of Deception, Fraud and Trust in Agent Societies, pages 69-75, 2003.

[19] I. B. Vapnyarskii, "Lagrange multipliers", in Hazewinkel, Michiel, Encyclopaedia of Mathematics, Kluwer Academic Publishers, ISBN 978-1556080104.

[20] Y. Wang and V. Varadharajan. Interaction trust evaluation in decentralized environments. In K. Bauknecht, M. Bichler, and B. Proll, editors, Proceedings of 5th International Conference on Electronic Commerce and Web Technologies (EC-Web04), volume LNCS 3182, Springer-Verlag, pages 144-153, Zaragoza, Spain, August-September 2004.

[21] J. Wainer, and A. Kumar, A fine-grained, controllable, user-to-user delegation method in rbac. In: SACMAT 2005: Proceedings of the tenth ACM symposium on Access control models and technologies, pp. 59-66. ACM Press, New York (2005)

[22] S. Waner and S. R. Costenoble. Applied Calculus. 4th edition. Apr 2007. Publisher: Brooks/Cole Pub Co.

[23] W. Winsborough, and N. Li, Towards Practical Automated Trust Negotiation. In Third International Workshop on Policies for Distributed Systems and Networks (POLICY 2002), Monterey, CA, June 2002.

[24] Z. Xie and C.H. Chi: Quantifying Trust through Delegation in Service Oriented Architecture. IEEE SCW 2007, pp. 308-315.

[25] L. Xiong and L. Liu. PeerTrust: Supporting reputation-based trust for peer-to-peer electronic communities. IEEE Trans. on Knowlege and Data Engineering, 16(7):843-857, 2004.

[26] G. Zacharia and P. Maes. Trust management through reputation mechanisms. Applied Artificial Intelligence Journal, 9:881-908, 2000.

[27] L. Zhang, G.J. Ahn, and B.T. Chu, A rule-based framework for role-based delegation and revocation. ACM Trans. Inf. Syst. Secur. 6(3), 404-441 (2003)

[28] X. Zhang, S. Oh, and R. Sandhu, Pbdm: a flexible delegation model in rbac. In: SACMAT 2003: Proceedings of the eighth ACM symposium on Access control models and technologies, pp. 149-157. ACM Press, New York (2003)