2003

Authorization-enhanced security framework for OGSA support.

Haiyan. Cheng
University of Windsor

Authorization-Enhanced Security Framework For OGSA Support

By

Haiyan Cheng

A Thesis
Submitted to the Faculty of Graduate Studies and Research
Through the School of Computer Science
In Partial Fulfillment of the Requirement for
The Degree of Master of Science at the
University of Windsor

Windsor, Ontario, Canada
2003

Your file Votre référence
ISBN: 0-612-84553-2
Our file Notre référence
ISBN: 0-612-84553-2

ABSTRACT

Security plays an important role for a large distributed system in an open community. Without enough knowledge about the user, it is hard to make access decision to the local resources. Current Public Key Infrastructure (PKI) uses a trusted third party, called Certificate Authority (CA), to check the identity of the users. The assumption of PKI is that every entity trusts CA absolutely and equally. This is also a weakness of PKI. The security problem in a Single-Sign-On (SSO) environment is more difficult to manage. Most of the current SSO security approach relies heavily on the pre-established trust relationship. This prevents wider adoption of SSO and greatly affects the local autonomy of the security policy making. Such SSO approaches have been employed recently in the Security Assertion Mark-up Language (SAML).

Based on the Dempster-Shafer theory and derived subjective logic, we propose an authorization-enhanced framework for large-distributed systems using a Single-Sign-On security approach. We extended the SAML assertion set to include opinions of the assertion issuer about the user. Based on the assertion issuer's opinion about the user and the trust relationship between the asserting party and accepting party, new assertion is generated at each local site. The probability expectation about the user's trustworthiness is computed. This value provides a reference for the system to make the access control decision.

Two sub-frameworks will be discussed. The first is a Peer-to-Peer model involving two parties and a technique of discounting opinions. The second is a multi-party model. In the latter case, opinions about the user from many asserting parties are considered and computed using a consensus operator to combine opinions. A numerical study is performed to compare these two models. We also compare this approach with other work related with trust management.

To my husband, Cheng Hu,

my dear daughter Catherine,

and my parents,

for their endless love

ACKNOWLEDGEMENTS

My greatest thanks go to my advisor, Dr. Robert Kent, for his continuous support academically and financially. Without his enlightened discussions and encouragement, this thesis work would never have been finished.

I wish to extend my sincere thanks to Dr. Jianguo Lu and Dr. Frank Lemire for their interest and guidance, to Dr. Scott Goodwin as the chair.

Lastly, I would like to thank my friends and peer colleagues for their discussion and help during my thesis work.

Table of Contents

ABSTRACT iii

DEDICATION.....iv

ACKNOWLEDGEMENTS.....v

LIST OF TABLES.....x

LIST OF FIGURES.....xi

1. INTRODUCTION.....1

1.1 An Overview of Computer Security.....1

1.2 Thesis Problem and Contributions.....2

1.3 Outline of the Thesis.....3

2. LITERATURE REVIEW.....4

2.1 Security Basics.....4

2.1.1 Privacy.....4

2.1.2 Integrity.....4

2.1.3 Authentication.....5

2.1.4 Authorization.....5

2.1.5 Non-repudiation.....6

2.2 Subjective Logic.....6

2.2.1 The Dempster-Shafer Theory.....6

2.2.2 Subjective Logic.....9

2.3 Trust Model.....16

2.3.1 Trust Definition.....16

2.3.2 Trust Model 17

2.3.3 Discounting Operator.....18

2.3.4 Consensus Operator 20

2.3.5 Evidence and Opinion Space Mapping.....22

3. SECURITY TECHNOLOGIES.....24

3.1 Public Key Infrastructure (PKI) 24

3.2 Role-Based Access Control (RBAC) and Single-Sign-On (SSO) 27

3.2.1 Multiple-Sign-On (MSO).....28

3.2.2 Single-Sign-On (SSO).....29

3.3 Grid Security Infrastructure (GSI) 31

3.4 Open Grid Service Architecture (OGSA) Security.....33

3.5 Security Assertion Mark-up Language (SAML) 34

3.6 Ponder Language.....39

3.7 Extensible Access Control Mark-up Language (XACML) 40

4. RELATED TRUST MANAGEMENT APPROACHES.....42

4.1 Decentralized Trust Management System.....42

4.2 A Distributed Trust Model.....44

4.3 Opinion-Based Filtering Through Trust.....46

4.4 Authorization Based on Evidence and Trust.....48

5. PROPOSED TRUST MODEL FOR SINGLE-SIGN-ON.....50

5.1 Motivations.....50

5.1.1 Assumptions of Our Approach.....51

5.1.2 Methodology.....52

5.2 Peer-to-Peer model.....53

5.2.1 Definitions.....53

5.2.2 Algorithm to Evaluate User Trust Degree.....56

5.2.3 Numerical Study and Discussion.....57

5.3 Improved Multi-Party Model.....62

5.3.1 Definitions.....63

5.3.2 Algorithm to Evaluate User Trust Degree.....64

5.3.3 Numerical Study and Discussion.....66

5.4 Monitoring System.....69

5.5 Application in Cross-Domain Authentication.....71

6. FEATURES AND COMPARISON WITH OTHER WORK.....73

6.1 Features.....73

6.2 Comparison With Other Work.....73

7. CONCLUSIONS AND FUTURE WORK.....76

REFERENCES.....78

APPENDIX.....84

VITA AUCTORIS.....92

LIST OF TABLES

Table 5-1 Opinion in Peer-to-Peer Model.....59
Table 5-2 Opinion in Peer-to-Peer Model.....59
Table 5-3 Opinion in Peer-to-Peer Model.....60
Table 5-4 Opinion in Peer-to-Peer Model.....60
Table 5-5 Opinion in Multi-party Model.....67
Table 5-6 Opinion in Multi-party Model.....68
Table 5-7 Opinion in Multi-party Model.....68
Table 5-8 Opinion in Multi-party Model.....69

LIST OF FIGURES

Figure 2-1 A Frame of Discernment.....10

Figure 2-2 Belief function definition dependency on subsets y of x.....11

Figure 2-3 Disbelief function definition dependency on sets y exclusive of x.....11

Figure 2-4 Uncertainty function definition of dependency on overlap of sets y and x12

Figure 2-5 Frame of Discernment with four atomic states.....13

Figure 2-6 Opinion Triangle.....18

Figure 2-7 Suggested Trust Category.....21

Figure 3-1 Public Key Encryption and Private Key Decryption.....25

Figure 3-2 Private Key for Signature.....25

Figure 3-3 RBAC Users and Roles.....28

Figure 3-4 Single-Sign-On Use Scenario.....30

Figure 3-5 A Computational Grid Security Architecture.....32

Figure 3-6 Categories of Security Challenges in a Grid Environment.....34

Figure 3-7 SAML Domain Model.....35

Figure 3-8 Authorization Policy Syntax.....40

Figure 3-9 Negative Authorization Policy.....40

Figure 4-1 KeyNote Trust Management Architecture.....44

Figure 4-2 A Distributed Trust Model.....45

Figure 4-3 Trust-enhanced Role-mapping Server Architecture.....49

Figure 5-1 Use Scenario for Authorization-Enhanced Security System (Peer-to-Peer model).....53

Figure 5-2 Multi-party Model Use Scenario.....66

1. INTRODUCTION

In parallel with the rapid development of new computing and networking technologies, the amount of information flow and the number of the resource providers have increased dramatically. The nature of computing is getting more complex, and the amount of computing required has increased exponentially following Moore's law. Traditional computing resources are not adequate to fulfill the new tasks. On one side, people are seeking to improve the performance of computer; on the other side, the sharing of information, resources and computing capability has been greatly encouraged.

After years of using Internet technology extensively, the client-server model of information retrieval will soon be replaced by web service technology. Web service [WebService] technology provides more flexible ways of presenting information and offers more powerful support for users to customize their requests. At the research end, Computational Grids [Grids] encourage the collaborative usage of the heterogeneous computing power to process large amounts of data for scientific research purposes. With the concept of service in mind, grid research has evolved to Open Grid Service Architecture (OGSA) [OGSA], which combines service features into the grid infrastructure.

1.1 An Overview of Computer Security

Security plays an essential role in any computing system. It is, and will always be an important topic. Since there does not exist an absolute secure system, "How secure is the system?" will always be a difficult question to answer. This leaves many problems for security researchers. Work can always be done to improve the degree of security, or to explore different security architectures.

The security problem in large-scale, distributed computing environments is more complicated than the security problem in a single system. Heterogeneity and distributivity have caused a dramatic increase in the difficulty of dealing with security issues,

particularly authentication of user identity, and authorization to access resources. The following are two typical causes of difficulty:

- The resource is distributed and potentially heterogeneous, both as hardware and software systems, and it belongs to different administration domains with disparate local security policies.

- The user group is large and diverse. The trust degree about a specific user is hard to evaluate due to a lack of knowledge about the user.

Most of the current mature solutions emphasize the aspects of security authentication. Two typical solutions are the Public Key Infrastructure [PKI], which is considered stable and robust though still evolving, and the Secure Socket Layer [SSL]. The current solutions to authorization and access control mechanisms use Role-Based Access Control (RBAC) [RBAC] applied to each security policy domain. Installation and administration of each domain security policy is currently supported by human system administrator and policy committees. The process of decision-making for access control is different from one site to another. The access control decision is closely related with the trust to requester's identity and the stipulation in the local security policy.

With the increasingly wide adoption of Single-Sign-On [SSO] approach, the problem of measuring system-user trust relationship and system-system trust relationship is critical.

1.2 Thesis Problem and Contributions

This thesis work explores a new and novel approach to building an authorization-enhanced mechanism for Single-Sign-On support in large-scale distributed environments. To the best of the author's knowledge, this is the first work that applies subjective logic reasoning in trust-based management for large scale, distributed computing with the Single-Sign-On approach. The basis of our approach is an association between the security assertion and the assertion issuer's trust opinion about this assertion. A

mathematical algorithm derived from the subjective logic [Jøsa02] is used to calculate the degree of the trust. The computed trust values can then be used as references to make access decisions. The proposed framework improves the functionality of the authorization mechanism. It provides a more flexible approach to support and use local security policy making. In this thesis, we will discuss the effectiveness of this algorithm through numerical study. Also, we compare the subjective logic approach for computing the trust degree with other related works.

1.3 Outline of the Thesis

This thesis starts with a brief introduction to the problem. Chapter 2 provides a literature review about related security approaches and subjective logic. Chapter 3 introduces the prevailing technologies used in computer security, and their limitation and advantages are discussed. Chapter 4 concentrates on the work related to authorization and the limitations of these approaches. Chapter 5 and Chapter 6 present our proposed work directed at constructing a security-enhanced authorization framework. Two trust models are introduced and the benefits of using this framework are analyzed. The Conclusion summarizes the thesis results and includes a discussion of potential directions for further research related to proposed thesis.

2. LITERATURE REVIEW

2.1 Security Basics

Computer security usually involves two aspects:

- Hardware and data security
- Transportation security

Transportation security refers to the protection of data during transmission. This can be enforced using data encryption and secure the transmission protocols. The hardware and data security is related to the physical security of the storage media and legal (that is, authorized) access to data.

Security requirements include the following: privacy, integrity, authentication, authorization and non-repudiation. These factors are discussed briefly below:

2.1.1 Privacy

Privacy is insurance that the information transmitted has not been captured or transferred to the third party.

2.1.2 Integrity

Integrity guarantees that the data sent has not been changed or damaged.

The integrity check can be implemented by the network security protocol. For example, at the network layer, IPSec [IPSec] can be used to guarantee the data integrity. At the transportation layer, SSL [SSL] is a better choice for securely sending sensitive data.

2.1.3 Authentication

Authentication deals with the issue of verifying the identity of the users or resources. Several measures can be taken in this respect. In order to verify a user identity, the user has to provide something to verify. It can be things known only by the user, for example the password, or things that belong only to the user, for example a smart card or key. It can even be part of the user, such as fingerprint, voice, retinal pattern, etc. For a distribute system, users are not expected to be physically present at the computing facility. For this reason, the last two approaches are only suitable for local users. A remote user has to present something to show identity. This is often referred to as "credentials". The basic, most common credential is a user id and password known only to the user. There are several drawbacks to this approach:

1. Since text string inputs are not always encrypted, an inappropriately chosen password can easily be broken.
2. All users have to register their username and password beforehand in order to use this service. During the first registration, a user can provide any personal information, which is difficult to verify.

In a distributed environment, for security reasons, it is better to set different passwords for different sites. Thus, users have to memorize different user ids and passwords for different sites they visit; this practice is very inconvenient for users.

2.1.4 Authorization

Authorization is the step following authentication. It is the process of deciding whether the authenticated user has certain rights to access the desired resources or to perform specified operations on the target. After verifying the user ID, we need to check if the user belongs to some group, or qualifies according to certain rules or conditions. For this step the checking is performed against a standard rule. This is most often the security policy set by the target resource.

2.1.5 Non-repudiation

Non-repudiation is the process of legally proving the sending or receiving of information. It means that the user cannot reject the truth that he has sent the information. This situation requires a trusted third-party whose responsibility is to verify, similar to a notary service. Public Key Infrastructure (PKI) has a special timestamp service that is designed to guarantee non-repudiation.

Most security approaches developed so far concentrate on authentication. More recently, however, increasing numbers of researchers have directed their efforts towards authorization and access control. Trust management is an essential part that no authorization and access control systems can avoid.

2.2 Subjective Logic

2.2.1 The Dempster-Shafer Theory

The Dempster-Shafer Theory (DST) [Shaf76] [Shaf90], also known as the theory of belief function, is a mathematical theory of evidence. This work was originated by A. P. Dempster (1968), and was extended by Glenn Shafer (1976). It is the first theory that dealt with the degree of the uncertainty and the combination of different degrees of beliefs. Similar reasoning can be dated back to the seventeenth century [Shaf].

Compared with traditional probability theory, in which evidence is associated only with one possible event, the Dempster-Shafer theory associates evidence with multiple possible events. The building blocks for the Dempster-Theory are three important concepts: the basic probability assignment function (bpa or m), the belief function (Bel) and the plausibility function (Pl) [Kari02].

The original Dempster-Shafer theory defines Θ as the set of mutually exclusive and exhaustive propositions about a domain. This Θ is called the *frame of discernment*. The

power set 2^{Θ} is defined as all subsets of Θ. This theory mainly concerns the elements of 2^{Θ}.

Definition 2-1. *A* ***basic probability function (BPA)*** *is defined as:*

$$m: 2^{\Theta} \rightarrow [0, 1], \quad m \text{ satisfies } m(\varnothing) = 0 \text{ and } \sum_{A \subseteq \Theta} m(A) = 1$$

This definition can be interpreted as a mapping from a power set to a closed set between 0 and 1, where m applied to empty set is 0, and the sum of m applied over all subsets of the frame of discernment is 1. (Later in this thesis, we use the value m and refer to it as the belief mass assignment [Jøsa09]).

Definition 2-2. *The* ***belief function*** $Bel: 2^{\Theta} \rightarrow [0, 1]$ *is defined as:*

$$Bel(A) = \sum_{B \subseteq A} m(B), \quad for \ A \subseteq \Theta$$

This defines the belief of a subset A of the frame of discernment in terms of the summation of m applied to all subsets B of A.

Definition 2-3 *The* ***plausible function*** *is defined as:*

$$Pl: 2^{\Theta} \rightarrow [0, 1], \ Pl(A) = 1 - Bel(\neg A) \ for \ A \subseteq \Theta$$

In words, the plausibility of the subset A of the frame of discernment is 1 minus belief associated with the complement of A in Θ.

The Dempster-Shafer theory also specifies a way to combine different evidence. Assume m1 and m2 are two different evidences about the same frame of discernment, Θ. A new probability assignment is defined using the formula:

Definition 2-4 Combined evidence

$$m(\varnothing) = 0$$

$$m(A) = K \sum_{x \cap y = A} m1(X) \bullet m2(Y) \quad , \quad A \subseteq \Theta, A \neq \varnothing$$

$$K = \frac{1}{1 - \sum_{x \cap y = \varnothing} m1(X) m2(Y)}$$

This definition can be interpreted as: the combined basic probability assignment is the summation of $m_1(X)$ and $m_2(Y)$ for all X, Y with intersection A divided by 1 minus the summation of $m_1(X)$ and $m_2(Y)$ for all X, Y with empty intersection.

A special case is when $\sum_{x \cap y = \varnothing} m_1(X)\, m_2(Y)$ is 1, which means that when the belief that intersection of the two evidence is zero, which is equivalent to the two evidence being totally contradictory to each other, then the combined belief is not defined.

The logarithm log (K) is called the weight of conflict between the belief of evidence 1 and belief of evidence 2.

Dempster-Shafer theory provides a way to obtain a degree of belief for one question from subjective probability for another question. The following example quoted from Shafer illustrates this idea and the usage of the Dempster-Shafer theory:

> *"Suppose I have subjective probabilities for the reliability of my friend Betty. My probability that she is reliable is 0.9, and my probability that she is unreliable is 0.1. Suppose she tells me a limb fell on my car. This statement, which must be true if she is reliable, is not necessarily false if she is unreliable. So her testimony alone justifies a 0.9 degree of belief that a limb fell on my car, but only a zero degree of belief (not a 0.1 degree of belief) that no limb fell on my car. This zero does not mean that I am sure that no limb fell on my car, as a zero probability would; it merely*

means that Betty's testimony gives me no reason to believe that no limb fell on my car. The 0.9 and the zero together constitute a belief function."

*"To illustrate Dempster's rule for combining degrees of belief, suppose I also have a 0.9 subjective probability for the reliability of Sally, and suppose she too testifies, independently of Betty, that a limb fell on my car. The event that Betty is reliable is independent of the event that Sally is reliable, and we may multiply the probabilities of these events; the probability that both are reliable is 0.9*0.9 = 0.81, the probability that neither is reliable is 0.1*0.1 = 0.01, and the probability that at least one is reliable is 1 -0.01 = 0.99. Since they both said that a limb fell on my car, at least of them being reliable implies that a limb did fall on my car, and hence I may assign this event a degree of belief of 0.99. Suppose, on the other hand, that Betty and Sally contradict each other—Betty says that a limb fell on my car, and Sally says no limb fell on my car. In this case, they cannot both be right and hence cannot both be reliable—only one is reliable, or neither is reliable. The prior probabilities that only Betty is reliable, only Sally is reliable, and that neither is reliable are 0.09, 0.09, and 0.01, respectively, and the posterior probabilities (given that not both are reliable) are 9/19, 9/19, and 1/19, respectively. Hence we have a 9/19 degree of belief that a limb did fall on my car (because Betty is reliable) and a 9/19 degree of belief that no limb fell on my car (because Sally is reliable). In summary, we obtain degrees of belief for one question (Did a limb fall on my car?) from probabilities for another question (Is the witness reliable?)."* [Shaf]

2.2.2 Subjective Logic

A modeling theory of subjective logic reasoning for uncertain probability theory, derived from the Dempster-Shafer theory, was proposed in [Jøsa02]. This belief model starts by defining a set of possible situations, again called the Frame of Discernment. A simple illustration is given in the paper as a frame of discernment:

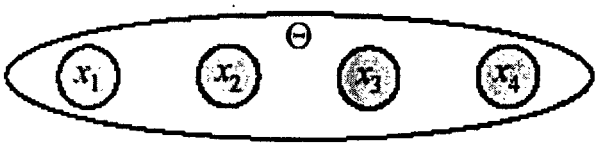

Figure 2-1 A Frame of Discernment [Jøsa02], page 2

Figure 2-1 represents all possible states of the given system in which only one state x_k can be true at any time. Because the elementary states do not contain sub-states, they are called atomic states. The notation 2^{Θ} is used to represent the power-set that contains the atomic states and all possible unions of the atomic states, including Θ itself and the empty set $\varnothing$. A state A that contains x_k is true if x_k is true, within Boolean logic.

A definition of Belief Mass Assignment is given as:

> ***Definition 2-5 (Belief Mass Assignment)*** *Let Θ be a frame of discernment. If, with each substate $x \in 2^{\Theta}$, a number $m_{\Theta}(x)$ is associated such that:*
>
> 1. $0 \le m_{\Theta}(x) \le 1$
> 2. $m_{\Theta}(\varnothing) = 0$
> 3. $\sum_{x \in 2^{\Theta}} m_{\Theta}(x) = 1$
>
> *Then, m_{Θ} is called a belief mass assignment on Θ, or BMA for short. For each sub-state $x \in 2^{\Theta}$, the number $m_{\Theta}(x)$ is called the belief mass of x.*

The belief mass expresses the relative belief assigned to the state x. Since the state x can be any union of the atomic states, or the atomic states themselves, or the empty set $\varnothing$, the definition cannot reflect our belief about the particular state x. So the definition of belief function has to rely on both the BMA of x and the BMA of substates of x.

Thus, based on the BMA definition, the belief, disbelief and uncertainty functions can be defined as follows:

Definition 2-6 (Belief Function) *Let Θ be a frame of discernment, and let m_Θ be a BMA on Θ. Then the belief function corresponding to m_Θ is the function b: $2^\Theta \rightarrow [0,1]$ defined by:*

$$b(x) = \sum_{y \subseteq x} m_\Theta(y), \quad x, y \in 2^\Theta$$

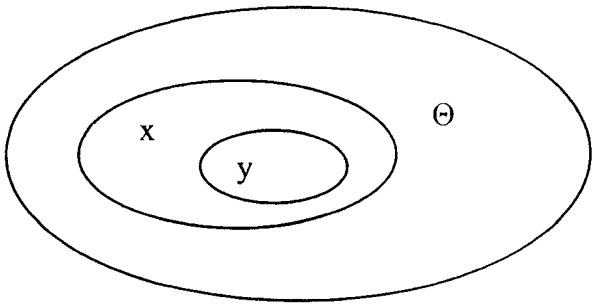

Figure 2-2 Belief function definition dependency on subsets y of x

Definition 2-7 (Disbelief Function) *Let Θ be a frame of discernment, and let m_Θ be a BMA on Θ. Then the disbelief function corresponding with m_Θ is the function d: $2^\Theta \rightarrow [0,1]$ defined by:*

$$d(x) = \sum_{y \cap x = \varnothing} m_\Theta(y), \quad x, y \in 2^\Theta$$

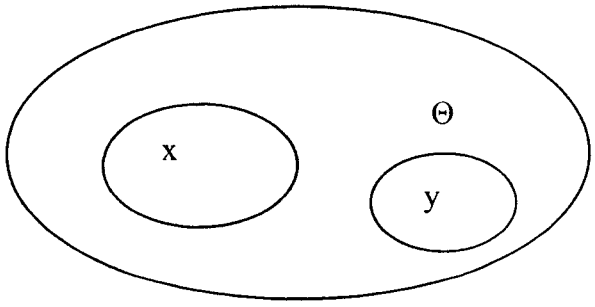

Figure 2-3 Disbelief function definition dependency on sets y exclusive of x

Definition 2-8 (Uncertainty Function) *Let Θ be a frame of discernment, and let m_Θ be a BMA on Θ. Then the uncertainty function corresponding with m_Θ is the function u: $2^\Theta \rightarrow [0,1]$ defined by:*

$$u(x) = \sum_{\substack{y \cap x \neq \varnothing \\ y \not\subseteq x}} m_\Theta(y), \quad x, y \in 2^\Theta$$

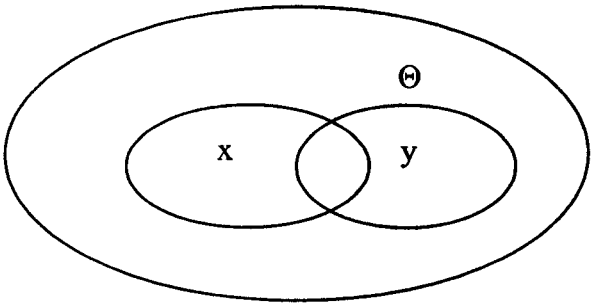

Figure 2-4 Uncertainty function definition of dependency on overlap of sets y and x

Theorem 2-1 (Belief Function Additivity)

$$b(x) + d(x) + u(x) = 1, x \in 2^\Theta, x \neq \varnothing$$

The above theorem can easily be proved according to the above definitions, particularly 2-1.

$$b(x) + d(x) + u(x) = \left(\sum_{y \subseteq x} + \sum_{y \cap x = \varnothing} + \sum_{\substack{y \cap x \neq \varnothing \\ y \not\subseteq x}}\right) m_\Theta(y) = \sum_{y \in 2^\Theta} m_\Theta(y) = 1$$

This theory is proposed by Josang for modeling subjective logic reasoning. The atomicity of a specific state x can be defined as the total number of the sub states it contains, denoted by |x|, with $|\varnothing| = 0$ and $|\Theta| = 2^n$ if Θ contains n atomic states.

Definition 2-9 (Relative Atomicity) *Let Θ be a frame of discernment, and let $x, y \in 2^{\Theta}$. Then for any given $y \neq \varnothing$, the relative atomicity of x to y is the function $a: 2^{\Theta} \rightarrow [0,1]$ defined by:*

$$a(x/y) = \frac{|x \cap y|}{|y|}, \qquad x, y \in 2^{\Theta}, \quad y \neq \varnothing.$$

By default, the relative atomicity a (x/ Θ) of a state relative to the frame of discernment is denoted by simply by a (x).

The probability expectation is associated with both uncertainty and the relative atomicity. It is defined as:

Definition 2-10 (Probability Expectation) *Let Θ be a frame of discernment with BMA m_{Θ}, then the probability expectation function corresponding with m_{Θ} is the function $E: 2^{\Theta} \rightarrow [0,1]$ defined by:*

$$E(x) = \sum_{y} m_{\Theta}(y)\, a(x/y); \quad x, y \in 2^{\Theta}$$

A simple example can be used to explain the relationship of these definitions. Figure2-5 represents the frame of discernment, which has four atomic states: x1, x2, x3 and x4.

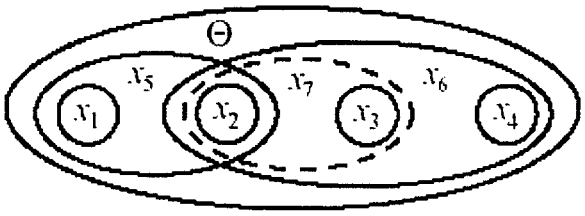

Figure 2-5 Frame of Discernment with four atomic states. [Jøsa02]

We assign each solid union of the atomic states as following (we omit those states y with m_Θ (y)=0).

$$m_\Theta: \begin{cases} m_\Theta(x_1) = 0.10 \\ m_\Theta(x_2) = 0.20 \\ m_\Theta(x_3) = 0.20 \\ m_\Theta(x_5) = 0.10 \\ m_\Theta(x_6) = 0.30 \\ m_\Theta(\Theta) = 0.10 \end{cases}$$

The detailed computation of E (x_7) is performed using the above assignments.

E (x_7) = m_Θ (x_1)*a (x_7 / x_1)+ m_Θ (x_2)* a (x_7 / x_2)+ m_Θ (x_3)*a (x_7 / x_3)+ m_Θ (x_4)*a (x_7 / x_4) + m_Θ (x_5)*a (x_7 / x_5)+ m_Θ (x_6)*a (x_7 / x_6)+ m_Θ (Θ)* a (x_7 / Θ).

According to the definition of relative atomicity, we have the following:

a (x_7 / x_1) = 0;

a (x_7 / x_2)= 1;

a (x_7 / x_3)=1;

a (x_7 / x_4)=0;

a (x_7 / x_5)=1/2;

a (x_7 / x_6)=2/3;

a (x_7 / Θ)= 2/4;

Hence, substituting values,

E (x_7) = m_Θ (x_1)*a (x_7 / x_1)+ m_Θ (x_2)*a (x_7 / x_2)+ m_Θ (x_3)* a (x_7 / x_3)+ m_Θ (x_4)*a (x_7 / x_4)+ m_Θ (x_5)* a (x_7 / x_5)+ m_Θ (x_6)*a (x_7 / x_6)+ m_Θ (Θ)*a (x_7 / Θ).

E (x_7) =0.1*0+0.2*1+0.2*1+0*0+0.1*1/2+0.3*2/3+0.1*2/4=0.7

So the value of E(x) is 0.7. This relative atomicity is then involved in the computation for trust degree.

The concept of subjective logic is appropriate for describing trust relationships, since trust is a fuzzy concept [Zimm91]. This means that there does not exist a clear boundary between trust and non-trust, as well as how much one party trusts another. The trust decision is made by an individual with their own judgement and understanding. Trust is an uncertain thing that cannot be described specifically or operationally. Generally speaking, there are two forms of uncertainty, aleatory uncertainty and epistemic uncertainty, as defined in [Helt97]:

- "Aleatory Uncertainty is the type of uncertainty which results from the fact that a system can behave in random ways." [Kari02] Other names for Aleatory uncertainty include: Stochastic uncertainty, type A uncertainty, Irreducible uncertainty, variability, objective uncertainty and, simply, uncertainty.

- Epistemic Uncertainty is the type of uncertainty "which results from the lack of knowledge about a system and is a property of the analysts performing the analysis."[Kari02] Other names of epistemic uncertainty include subjective uncertainty, type B uncertainty, reducible uncertainty, state of knowledge uncertainty and ignorance.

In the security problem for distributed systems, most of the uncertainties are of Epistemic (subjective) type. In our SSO case, the problem may be stated as the following: In an open environment, due to a lack of sufficient knowledge of the user who tries to access the resource and a lack of sufficient knowledge of the assertion issuer, for the same security assertion different servers might have different opinions about the trustworthiness of the assertion. This range of interpretation will affect decision making for access control.

As stated in [Jøsa96], "trust simply is a human belief, involving a subject (the trust party) and an object (the trusted party)." The security decision related to authorization can be measured by the subjective trust with measures m, b, d, u and E associated with the original requestor, the trust relationship between sites and their combination.

2.3 Trust Model

2.3.1 Trust Definition

Trust is an important aspect of decision-making. Although the word "trust" is used in many situations, to give trust a unified definition is difficult. There exist several standardized definitions for trust, but none of them can be used as the sole, or primary standard. Trust is defined in the Oxford English Dictionary as "the firm belief in the reliability or truth or strength of an entity" [Oxford]. Kini and Choobineh defined trust as "a belief that is influenced by the individual's opinion about certain critical system features." [Kini98] Jones defined trust as "the property of a business relationship, such that reliance can be placed on the business partners and the business transactions developed with them." [Jone99] In their survey of trust, Tyrone and Morris defined trust as "the firm belief in the competence of an entity to act dependably, securely, and reliably within a specified context" [Tyro00].

From these various versions of the trust definitions, we note that trust is a complex topic within the semantic of natural language. It is hard to measure trust, which means that there does not exist a clear definition for the set of trust elements.

Since trust is a subjective belief concept, it is natural for researchers to use subjective logic to model trust. Subjective logic is described as "a logic which operates on subjective belief about the world." [Jøsa02] By using this definition, subjective logic can be "seen as an extension of both probability calculus and binary logic" [Jøsa02].

Based on the description of subjective logic and the method to compute the probability expectation, the trust model can be simplified from the subjective logic into the following proposed model discussed in the following sector.

2.3.2 Trust Model

A trust model is a general model for expressing relative uncertain belief about the truth of a statement [Jøsa99]. In order to build such a model, the statements, or assertions, that are the foundation must be crisp [Zimm91], which means that each statement must be either true or false.

By making this assumption, the application of Dempster-Shafer theory has been simplified into a binary frame of discernment such that it contains two statements, or, elements, (x, and not x, or $\neg x$). In this case the default relative atomicity of x (or $\neg x$) is 0.5.

However, it is still impossible to measure the validity of the statement x because our knowledge of reality is generally not perfect. Thus, an alternative to Dempster-Shafer's theory was introduced, which the author has argued in [Jøsa99] that compared to the Dempster-Shafer theory, this subjective logic reasoning is "corresponds more closely with human intuitive reasoning".

The subjective logic starts with introducing the concept of "opinion" to represent the belief, disbelief and uncertainty about the validity of the statement [Jøsa99], with

$$b + d + u = 1,\ 0 \leq b, d, u \leq 1$$

and where b, d and u designate belief, disbelief and uncertainty respectively;

Here, w={b ,d , u} is used to represent an opinion. A graphic representation of this definition is a triangle shown in Figure 2-6 [Jøsa99]. The point shown in the graph represents an opinion w={0.7,0.1,0.2}.

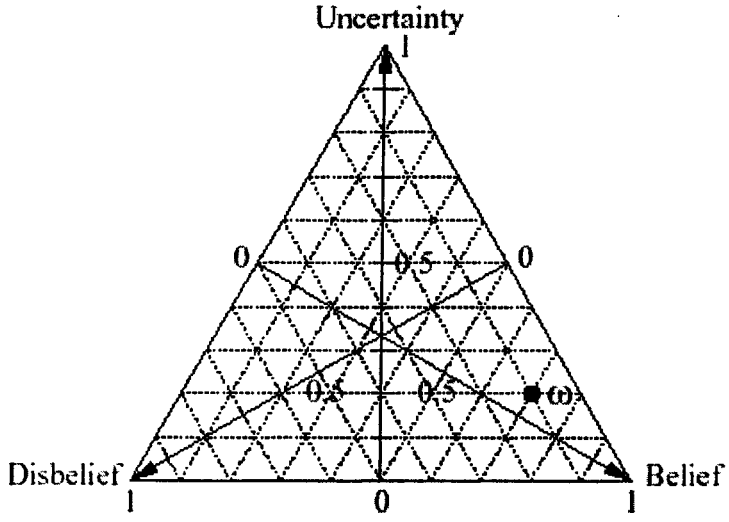

Figure 2-6 Opinion Triangle, From [Jøsa99], page 2

The graph in Figure 2-6 can be viewed as a two-dimension system. The horizontal bottom line is the probability dimension and the vertical line is the uncertainty dimension. The projection of the opinion on to a one-dimensional probability space produces the probability expectation value:

$$E(\{b,d,u\}) = b + u / 2$$

2.3.3 Discounting Operator

In this and the following section we introduce and discuss two operators used to combine opinions: discounting and consensus operator.

In the case that agent A has an opinion about an entity, and agent B has an opinion about A's trustworthiness, the discounting operator can be applied to obtain B's opinion about the entity through A. In this case the opinion of B towards A is associated with the

statement: "A is knowledgeable and will tell the truth." [Jøsa99] Clearly, assertion such as this contains semantic ambiguity reflected by the need to express and include uncertainty in any attempt to guarantee belief and disbelief.

The discounting operator, as defined in [Jøsa99], is the following:

Definition 3-7 (Discounting Operator)

Let $\omega^A_B = (b^A_B, d^A_B, u^A_B, a^A_B)$ *be agent A's opinion about agent B as a recommender, stated as a proposition about B to A, and let x be a proposition where* $\omega^B_x = (b^B_x, d^B_x, u^B_x, a^B_x)$ *is B's opinion about x expressed in a recommendation to . Let* $\omega^{A:B}_x = (b^{A:B}_x, d^{A:B}_x, u^{A:B}_x, a^{A:B}_x)$ *be the opinion such that*

1. $b^{A:B}_x = b^A_B\, b^B_x$
2. $d^{A:B}_x = b^A_B\, d^B_x$
3. $u^{A:B}_x = d^A_B + u^A_B + b^A_B\, u^B_x$
4. $a^{A:B}_x = a^B_x$

Then $\omega^{A:B}_x$ *is called the discounting of* ω^B_x *by* ω^A_B *expressing A's opinion about x as a result of B's advice to A. By using the symbol '⊗' to designate this operator, we define* $\omega^{A:B}_x = \omega^A_B \otimes \omega^B_x$.

One special interpretation applies in this definition, which is that when A disbelieve that B will give an accurate opinion. This is interpreted "as if A thinks that B is uncertain about the truth value of x so that A also is uncertain about the truth value of x no matter what B's actual advice is."

From this definition and the constraints on the belief, disbelief and uncertainty, some of the characteristics about this operator are obvious:

1. The discounting operator is associative.
2. The discounting operator is not commutative.

3. The discounted belief is non-increasing.
4. The discounted disbelief is non-increasing.

By using the discounting operator, two pieces of opinion about the same statement can be combined and justified into one opinion.

2.3.4 Consensus Operator

In contrast to the discounting operator situations where multiple independent opinions are available, the consensus operator can be used to compute the combined opinion. The result is equivalent to a comprehensive opinion achieved by an imaginary party, after considering all the opinions.

Definition 3-8 (Consensus Operator) [Jøsa00]

Let $\omega^A{}_x = (b^A{}_x, d^A{}_x, u^A{}_x, a^A{}_x)$ *and* $\omega^B{}_x = (b^B{}_x, d^B{}_x, u^B{}_x, a^B{}_x)$ *be opinions respectively held by agents A and B about the same proposition x. Let* $\omega^{A,B}{}_x = (b^{A,B}{}_x, d^{A,B}{}_x, u^{A,B}{}_x, a^{A,B}{}_x)$ *be the opinion such that*

1. $b^{A,B}{}_x = (b^A{}_x u^B{}_x + b^B{}_x u^A{}_x) / \kappa$
2. $d^{A,B}{}_x = (d^A{}_x u^B{}_x + d^B{}_x u^A{}_x) / \kappa$
3. $u^{A,B}{}_x = (u^A{}_x u^B{}_x) / \kappa$
4. $$a^{A,B}{}_x = \frac{a^B{}_x u^A{}_x + a^A{}_x u^B{}_x - (a^A{}_x + a^B{}_x) u^A{}_x u^B{}_x}{u^A{}_x + u^B{}_x - 2 u^A{}_x u^B{}_x}$$

Where $\kappa = u^A{}_x + u^B{}_x - u^A{}_x u^B{}_x$ *such that* $\kappa \neq 0$. *We define* $a^{A,B}{}_x = (a^A{}_x + a^B{}_x)/2$ *when* $u^A{}_x, u^B{}_x = 1$. *Then* $\omega^{A,B}{}_x$ *is called the consensus between* $\omega^A{}_x$ *and* $\omega^B{}_x$, *representing an imaginary agent [A, B]'s opinion about x, as if she represented both A and B. By using the symbol "⊕" to designate this operator, we define* $\omega^{A,B}{}_x = \omega^A{}_x \oplus \omega^B{}_x$.

The purpose of applying the consensus operator is to reduce the uncertainty. As we can see from the above definition, two opinions that both lacking uncertainties cannot be combined. The consensus of an infinite number of opinions with uncertainty will generate an opinion without uncertainty.

We can also observe some characteristics about the consensus operator as the following:

1. The consensus operator is commutative.
2. The consensus operator is associative.
3. The opinions must be independent.
4. One opinion cannot be applied repeatedly.

We note in general that the consensus operator permits b, d and u to either increase or decrease.

A suggested trust categorisation is given in the triangle graph in Figure 2-7[Jøsa99].

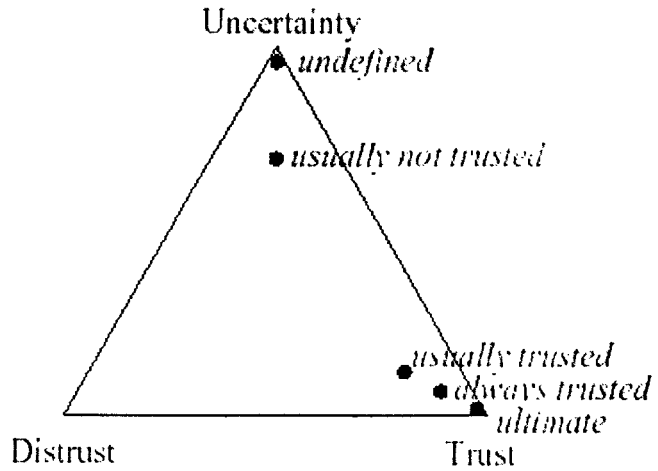

Figure 2-7 Suggested Trust Category, from [Jøsa99]

From the graph in Figure 2-7, we can see that, depending on the position of the points in the triangle grid, the trust degree can be roughly separated into several categories: ultimate trust, always trusted, usually trusted, usually not trusted, undefined, and distrust.

Different people may have different ways of dividing the trust zone and assigning interpretations.

Another interesting aspect about opinion is that they can be ordered according to the values of belief, disbelief and uncertainty. As suggested by [Jøsa99], the ordering of opinions should follow several rules:

1. *The opinion with the greatest probability expectation is the greatest opinion.*
2. *The opinion with the least uncertainty is the greatest opinion.*
3. *The opinion with the least relative atomicity is the greatest opinion.*

These rules can be reflected in the opinion triangle as:

1. The projection to the bottom line lies furthest right reflects the greatest opinion.
2. The opinion positioned furthest down in the triangle reflects the greatest opinion.
3. The opinion positioned furthest right reflects the greatest opinion.

2.3.5 Evidence and Opinion Space Mapping

One mapping between the evidence and opinion space is made by the following formula:

$$\begin{cases} b = \dfrac{r}{r+s+2} \\ d = \dfrac{s}{r+s+2} \\ u = \dfrac{2}{r+s+2} \end{cases} \quad \text{Where } u \neq 0$$

In the above formula, r represents the number of evidence that support the belief towards the statement, s represents the number of evidence that support the disbelief about the related statement.

By using this mapping, the traditional probability estimation of binary event can be represented using the opinion space representation. In our application, we use this mapping as our theoretical foundation to assign the initial values for opinions. We also use this in the proposed monitor system introduced in section 5.4.

3. Security Technologies

In this chapter, we present and discuss security technologies in order to establish a basis for justifying and analyzing the use of subjective logic and trust in authorization.

3.1 Public Key Infrastructure (PKI)

Public Key Infrastructure (PKI) [Adam02] is a secure architecture that uses one of the asymmetric encryption techniques-- public key cryptography [Delf02]. The concept of the Public-key cryptography was introduced in the 1970s. It uses a pair of mathematically related cryptographic keys and the hash function to encrypt messages. The two keys are referred to public key and private key. The public key, as its name suggests, is publicly distributed, and can be seen by everybody. The private key is stored by the user secretly, and it can be used to verify to the other party the identity of the user.

Depending on the situation, one of the keys is used to encrypt the message; the other is used to decrypt the message. These two keys are different from each other, and it has been proven mathematically that it is almost impossible to derive the other key from the known key.

The following discussion illustrates some scenarios using public key cryptography:

a. Using the public key for encryption and private key for decryption is shown in Figure 3-1.

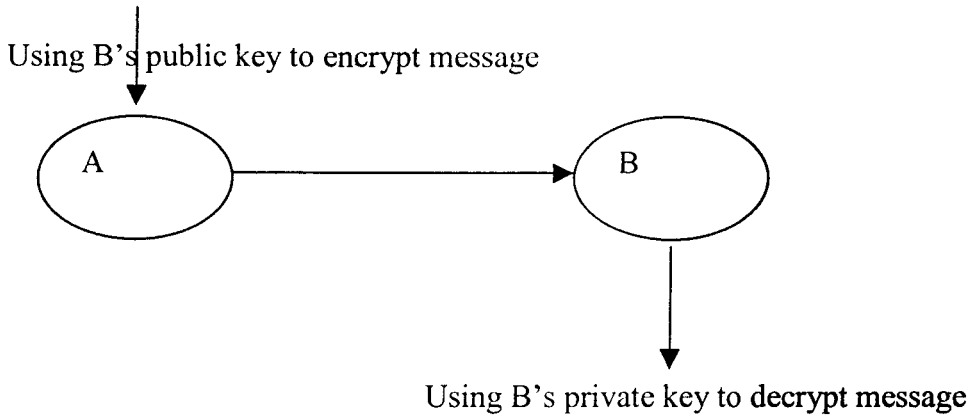

Figure 3-1 Public Key Encryption and Private Key Decryption

This works when another person wants to send you information that only can be read by you. He uses your public key to encrypt the message. When the message has been received, you can use your private key to decrypt the message.

This way the sender A is sure that the message can only be received by B and B is sure the message is designated to B alone. Note that B cannot verify the sender is really A.

b. Using private key for signature is shown in Figure 3-2.

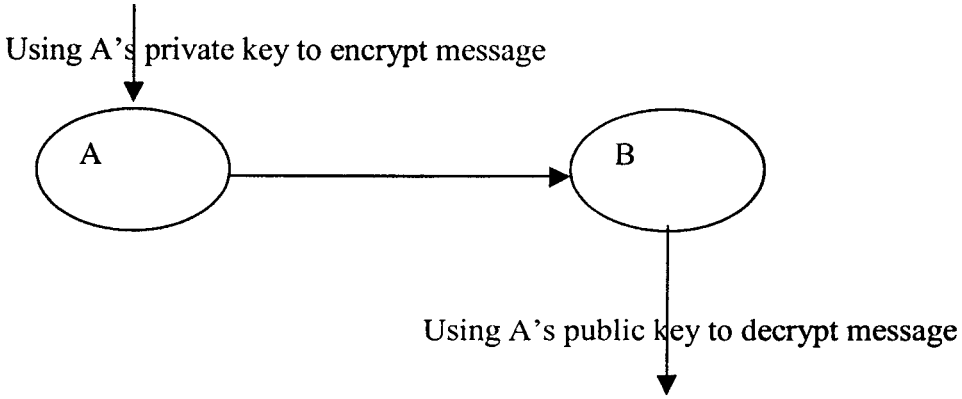

Figure 3-2 Private Key for Signature

In order for receiver B to guarantee the message is really coming from the claimed sender A, B can ask A to sign the information using A's private key. The signing process is the creation process of a digital signature, in which a unique mathematical value is calculated using either a "hashing" or "message authentication" algorithm, then this value is encrypted with the private key and sent along with the message.

c. Public key for signature

When receiver B receives a message from A, he uses A's public key to decrypt the hash value, then he calculates the hash of the information received using the same hashing algorithm; these two hashing values are compared. If they are the same, the receiver assumes that this information is intact during the transmission and, at the same time, the information source has also been verified.

From the discussion above, we can see the advantages of using Public key cryptography for secure authentication. It provides encryption/decryption and signing/verifying functionality to guarantee the integrity of the message and the identity of the end user.

PKI is a secure architecture that uses public key cryptography. It is designed to provide an increased level of confidence for information exchanging over the Internet.

PKI is comprised of four parts: registry authority, certificate authority, directory service and time stamping. The major responsibility of a Registry Authority (RA) is to verify the user's identity according the user provided information. The Certificate Authority (CA) is mainly responsible for issuing and managing the certificate and maintaining the Certificate Revocation List (CRL). The Directory Service is also an important part that is used to publish the certificate and the certificate revocation list. A special service, Time Stamping, is used to guarantee the non-repudiation feature, which works as a notary service that legally confirms the user has sent or received the digital documents.

Although PKI has been widely adopted as a security architecture by many organizations and companies, potential problems exist when implementing the PKI. Jøsang summarized those typical problems as the following [Jøsa00]:

1. The certificate policy should be designed in a way that the computer can easily and appropriately interpret.
2. If cross certification is to be implemented by the root CA, the relevant two PKIs must have equal policy strength.
3. Current PKI only supports binary trust mode [Jøsa00].
4. Multiple recommendations are not taken into account in PKI.
5. Decision making support is weak.

3.2 Role Based Access Control (RBAC) and Single-Sign-On (SSO)

Although PKI can be used to handle the authentication related issues. The authorization procedure that follows the authorization has to be dealt with using access control mechanism. "Who can access what resource" and "who can perform what operation" are two typical questions that the access control mechanism has to answer. Today, the security management for a large networked system is both costly and error-prone because administrators usually specify access control lists for each user on the system individually [RBAC]. A technology that attracts more and more attention came into being, which is the Role Based Access Control (RBAC).

Role Based Access Control (RBAC) [RBAC] utilizes a membership concept. The access permissions to certain resources are administrated by a group of users, users may belong to one or more group. When a user signs into a system, the system checks the user's credential and trust information provided to establish the user's role; then, based on the role mapping result, access is granted according to the role the user currently plays. An Access Control List (ACL) is maintained by the destination resource. One role can have multiple users and one user can have multiple roles, as illustrated in Figure 3-3.

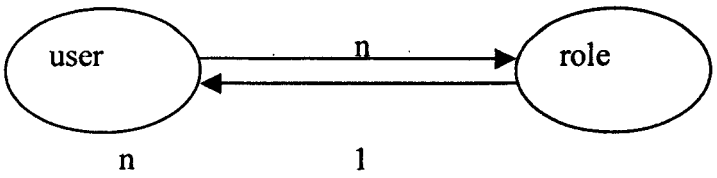

Figure 3-3 RBAC Users and Roles

RBAC not only provides a simplified approach for security management, but also greatly improved the flexibility of specifying and enforcing the local security policy [Ferr95]. As stated by Ferraiolo, "RBAC provides greater productivity on the part of security administrators, resulting in fewer errors and a greater degree of operational security" [Ferr95]. Currently, many enterprises are applying RBAC as their access control mechanism.

3.2.1 Multiple-Sign-On (MSO)

The most basic form of authentication is the user name and password. One of the prerequisites of using this approach is that users have to register their information in the administrator's database. After the user provides their information, the administrator has to verify its correctness, thereby assigning a role to the user. The process of verification is different for different security domains. That is where the trust plays an important role. If everybody can be trusted, then there's no need to spend time on verifying the information. A simple way to check the partial correctness of the information involves sending an e-mail to verify the e-mail address. To make it more trustworthy, the administrator usually requests the user provide some proof. These proofs can be their personal ID or working card with their institution name on it. Other than that, the administrator may have to rely on a third party authentication of the user.

The currently used digital certificate employs third party verification. The trusted CAs are the third parties, and when the certificate was issued, the CA had to rely on the third party to verify the certificate requestor. This third party is usually an authorized security officer

who has the responsibility and authority to guarantee the trustworthiness of the assertion about a certificate requestor.

Although we don't use the term "Multiple-Sign-On" very often, in actual practice, a user has to sign on to every site they visit. The term "Multiple-Sign-On" is used to distinguish from the recently emerging, popular usage of Single-Sign-On techniques. If the user names and passwords are used as credentials, for security reasons, it's better to use different pairs for different sites in Multiple-Sign-On. Thus, if a hacker knows one of the credentials, it won't jeopardize other credentials for accessing other sites.

Using MSO, a user can easily distinguish the transition from one site to another, since every time he visits a new site he is prompted for authentication. This is wasteful of time and not convenient for users.

3.2 2 Single-Sign-On (SSO)

In order to overcome the above-mentioned problems, Single-Sign-On (SSO) approach was introduced. The basic idea of SSO is that user only needs to sign on once, which means that the user can maintain the same identity in the Single-Sign-On domain. The most significant advantage of using SSO is that it provides the user with secure, reliable access to resources located at different sites without having to remember numerous passwords.

The Single-sign-on approach we discuss in this thesis is different from the single-sign-in, which is used by most of the portal-based service providers. Single-sign-in can only be used to sign in to a portal, but it does not support Single-Sign-On to the multiple sites and services [Jon02].

The SSO use-case scenario is shown in Figure 3-4.

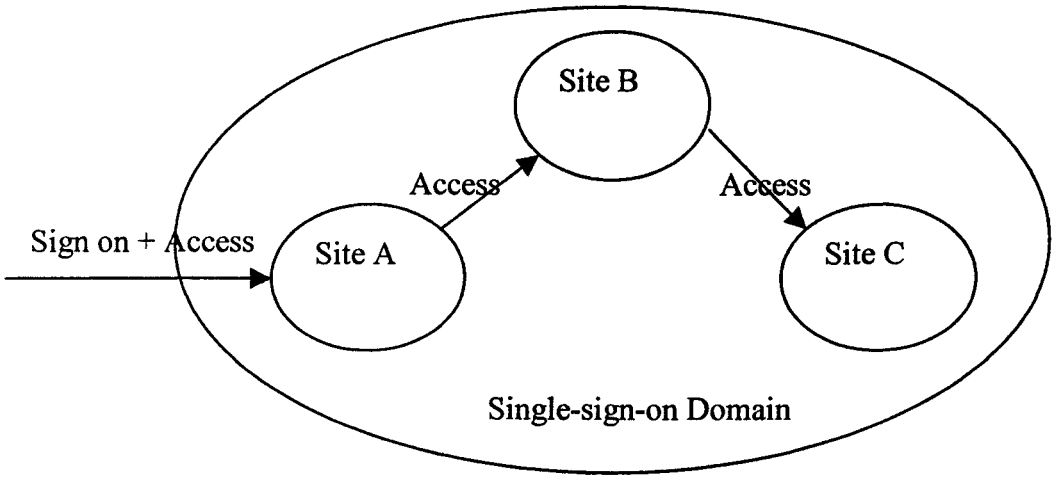

Figure 3-4 Single-Sign-On Use Scenario

Current problem with authorization system in SSO environment:

In a SSO domain, the user credential is not transmitted back and forth for authentication purpose. This way, the risk of losing the credential is greatly reduced. Instead, the security assertion is used solely for authentication purposes. How an appropriate access control decision can be made, based on the assertion, is heavily affected by the security approach of the whole system. There are several factors that affect the decision:

- The issuer of the assertion,
- The authentication method used by the user to log on to the first site.
- The relationship between each local autonomous domain and the issuer.

There are many uncertainties involved in each factor, especially with respect to trust. Thus, most of the current SSO solution relies on the pre-established trust relationship. This has greatly hindered wider adoption of the Single-Sign-On at the present time.

3.3 Grid Security Infrastructure (GSI)

Grid Security Infrastructure (GSI) [Fost98a] is a security architecture specifically designed for large-scale, distributed, heterogeneous high performance systems, i.e. a computational grid system. It integrates tools, libraries and protocols to allow user s to securely access the resources. At the present time, GSI uses public key infrastructure for authentication. It uses SSL to verify the certificate and perform message protection.

The single-sign-on functionality is an important feature for GSI, it allows users to sign in once and perform many operations without re-authentication. The single-sign-on is implemented by using the concept of "proxy". In GSI both user proxy and resource proxy are used. The user proxy is used in the process that acts on behalf of the user: the resource proxy is used as an interface to translate between grid security and local security architectures.

Figure 3-5 shows the GSI infrastructure. The whole process starts with a user signing into the system from a local host. The host generates a temporary credential for the user proxy. By signing this temporary credential using a private key, the user delegates some rights to the proxy, so that the proxy can act on behalf of the user. The proxy uses a temporary credential to request and negotiate with the resource proxy to obtain access control. This way, the user's long-lived credential is never actually transmitted over the Internet. The global-to-local mapping is performed by the resource proxy.

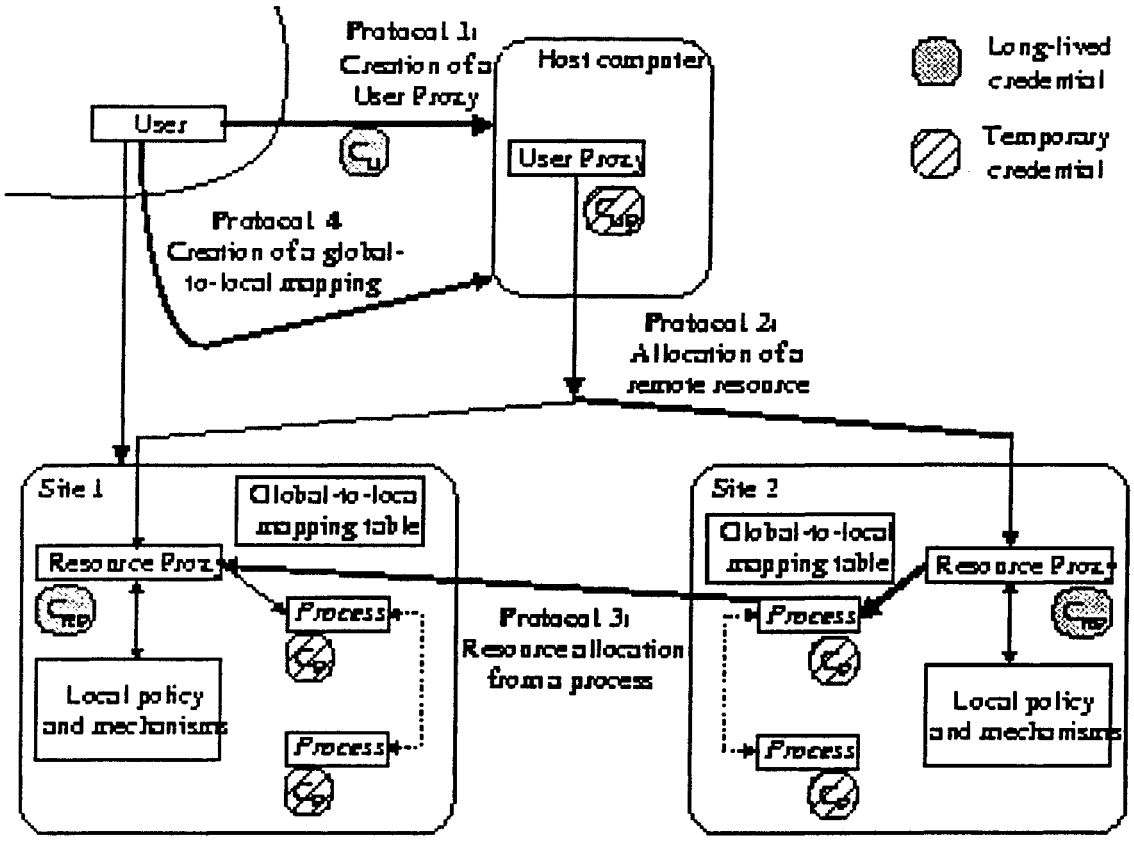

Figure 3-5 A Computational Grid Security Architecture [Fost98a], Page 4

There are several areas that GSI not cover: group secure communication; how to handle the continually increasing large size of the user group (the problem of scalability); and authorization.

As mentioned in [Fost98a], the security requirements for a large-scale distributed system (such as computational grids) must include several features. Among them the single-sign-on ability and the interoperability with local security policies are two crucial requirements.

GSI implemented single-sign-on by creating a user proxy; the user then delegates all or part of the user's rights to the proxy. The proxy generates a temporary credential and uses

this credential to negotiate with the local resource proxy for access control. This approach can be quite complex. Although it solves the mutual authentication and single-sign-on problem, the implementation and usage of this approach is very complicated. Several parties may be involved in the single request, and additional processes have to be added to handle the creation and termination of the temporary credentials (i.e, proxy management). There are several disadvantages to this approach:

1. One user cannot initialize two requests on the same resources at the same time. That is, one user cannot have two temporary credentials at the same resource.
2. If the user group is very large, the resource might not be able to handle the increased number of user credentials, as well as the proxy credentials.

The local server can only control the temporary credential at the time of its creation, by signing the temporary credential using user's private key; The user can restrict the permission by adding the active time interval and other conditions of usage through the local server. The flexibility of the system is greatly reduced.

The GSI approach is expected to evolve into the OGSA [Fost02] security infrastructure.

3.4 Open Grid Service Architecture (OGSA)

Open Grid Service Architecture (OGSA) [OGSA] is an architecture that integrates heterogeneous located resources to perform intensive computation. It is an extension and combination of both computational grids and web service technology. The formal proposal of the OGSA was proposed by the Global Grid Forum (GGF).

For large-scale distributed systems, such as web services or OGSA, the security architecture requires more features, but it should be easier to use. The following graph in [OGSA] summarizes the security requirements for open grid service architecture.

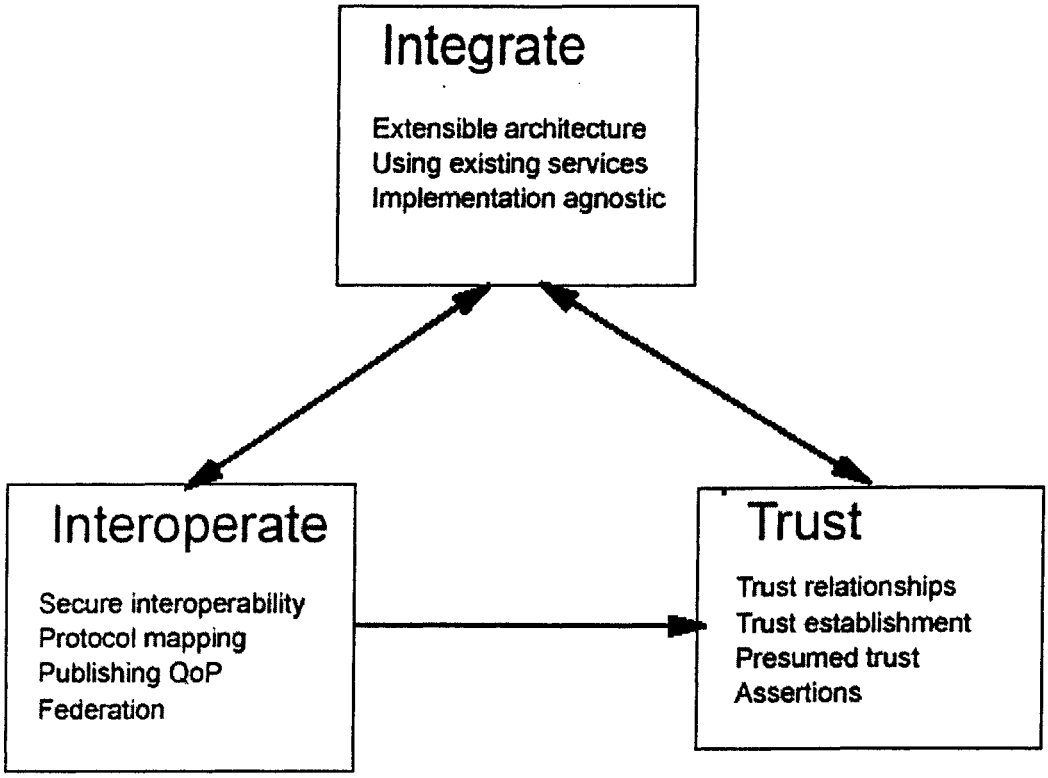

Figure 3-6 Categories of Security Challenges in a Grid Environment (from [OGSA])

The security working-group of the OGSA is still working on the drafts of the OGSA security requirements at the time of writing this thesis They proposed adoption of SAML as the authorization mechanism for OGSA [Von03]. The details will be discussed at the upcoming GGF-8 in June, 2003.

3.5 Security Assertion Mark-up Language (SAML)

The Security Assertion Mark-up Language (SAML) [SAML] is developed by Organization for the Advancement of Structured Information Standards (OASIS). It is "an XML-based framework for exchanging security information" [SAML]. The security information can be authentication information, authorization information or permissions. SAML is designed as a standard specification for B2B and B2C communications.

SAML is a combination of and improvement on two XML security standards: Securant Technologies' AuthXML and Netegrity's Security Services Mark-up Language (S2ML). SAML can be used to handle secure exchanging of the authentication and authorization information between domains. Some of the features of SAML are shown in Figure 3-7.

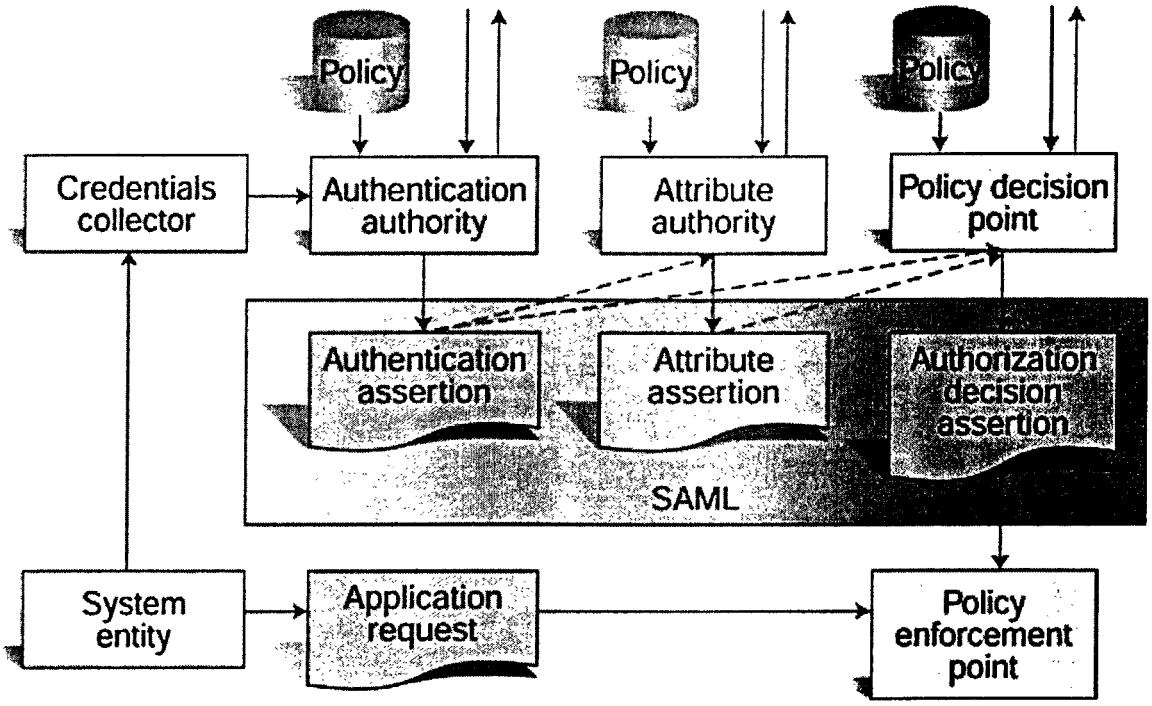

Figure 3-7 SAML Domain Model (from [SAML])

An important feature of SAML is its support of Single-Sign-On (SSO) ability. Compared to other single-sign-on solutions, such as Microsoft's Passport, SAML is not proprietary; indeed, OpenSAML [OpenSAML] reflects an open community development effort in support of open standards. It is also compatible with many existing XML security communication protocols, such as .NET, web service and SOAP.

Currently, SAML supports three types of assertions: authentication assertion, authorization assertion and attribute assertion. As their name suggests, the information contained in the assertion are for different purposes: the authentication assertion specifies that the user has authenticated in the assertion issuer's site by providing some form of

credentials. The authorization assertion includes the authorization related access decision that made by the assertion issuer. The attribute assertion includes information about the attributes of the user that has been associated by the assertion issuer.

SAML realizes Single-Sign-On by creating an authentication assertion at the time of first site authentication, instead of transporting the user credentials, whether original of proxy form, for repeated authentication at other sites. The authentication assertion is transported together with the user's request. This way the possibility of revealing user's credential has been greatly reduced. So, after the user passes authentication at the first site, the following sites will heavily rely on the assertion passed along to make their access control decision based on their local security policy. The best scenario for using SAML is within a business affiliate (sometimes called federation), in which a general security policy is agreed upon. Each site within the affiliation needs to have SAML installed, be able to interpret the assertion, and be able to map user roles.

When we talk about the heterogeneity of the different sites among a large scale, distributed, and high-performance computing environment, we often refer to the geographical remoteness and disparate nature of the sites, including different hardware platforms and systems, and, especially with respect security, disparate local security policies.

Users do not need to remember (i.e. store) many credentials in order to gain access to several sites. Instead, they may login once and gain access to any site within the federation. The use of SSO should have a domain restriction, among those sites where the user is recognized and accepted.

According to the current specification, a typical SAML assertion contains the following elements:

- Issuer ID and issuance timestamp
- Assertion ID

- Subject

  Name and security domain

  Subject's authentication data (optional)

- Advice (optional additional information provided by the issuing authority)

- Conditions under which the assertion is valid

  Assertion validity period (NotBefore and NotOnOrAfter)

  Audience restrictions

  Target restrictions (intended URLs for the assertion)

  Application specific conditions

A sample SAML message has a format similar to the following [Jon02]:

```
<samlp: Request ...>
  <samlp: AttributeQuery>
      <saml: Subject>
      <saml: NameIdentifier
        SecurityDomain="www.uwindsor. ca"
        Name="paul"/>
    </ saml: Subject>
        <saml: AttributeDesignator
      AttributeName="Employee_ ID"
      AttributeNamespace="www.uwindsor.ca">
    </ saml: AttributeDesignator>
  </ samlp: AttributeQuery>
</ samlp: Request>
```

In response, the issuing authority asserts that the subject (S) was authenticated by means (M) at time (T).

```
<saml: Response
  MajorVersion="1" MinorVersion="0"
  RequestID="128.14.234.20.90123456"
  InResponseTo="123.45.678.90.12345678"
  StatusCode="Success">

  <saml: Assertion
    MajorVersion="1" MinorVersion="0"
    AssertionID="123.45.678.90.12345678"
    Issuer="Sun Microsystems, Inc."
    IssueInstant="2002- 01- 14T10: 00: 23Z">

    <saml: Conditions
      NotBefore="2002- 01- 14T10: 00: 30Z"
      NotAfter="2002- 01- 14T10: 15: 00Z" />

    <saml: AuthenticationStatement
    AuthenticationMethod="Password"
    AuthenticationInstant="2001- 01- 14T10: 00: 20Z">

      <saml: Subject>
        <saml: NameIdentifier
          SecurityDomain="www.uwindsor.ca"
          Name="paul" />
      </ saml: Subject>
    </ saml: AuthenticationStatement>
  </ saml: Assertion>
</ samlp: Response>
```

All this information is used as a reference for making access control decisions at one specific local site.

SAML's SSO relies mainly on the trust relationship between sites. Since trust is not transitive, which means that if A trusts B and B trusts C, we cannot derive that A trusts C in general. One can only assert that the security of a downstream node can only be as good as the security of previous nodes. Now the problem becomes how to set up the local security policy, with respect to peer sites within the same SSO domain, so that the received security assertion can be properly interpreted and processed according to the local security need.

3.6 Ponder Language

The Ponder toolkit [Ponder] was developed by researchers in the Computer Science department at Imperial College, London. The development of Ponder was based on policy based security management. The toolkit itself has several components: Ponder compiler, Ponder policy editor and Ponder management toolkit. The Ponder language is designed to specify management and security policies for distributed systems. It is an object-oriented, declarative programming language that is used to write different kinds of security policies.

Those policies can be grouped under two major categories: access control policy, which limits the authenticated user activities; and, obligation policy, which specifies the action that must be performed by the system administrator [Dami01]. Within the division of the access control policy, it can be further categorized into authorization policy, information filtering policy, delegation policy, and refrain policy. Authorization policy defines "what activities a member of the subject domain can perform on the set of objects in the target domain" [Dami01]. The information filtering policy is used to transform the input and output parameters to appropriate format. The delegation policy is used to specify the "temporary transfer of access right". Finally, the refrain policy defines the actions that are forbidden to perform on the target objects.

Figure 3-8 illustrates the syntax of the authorization. In this syntax definition, the bold terms represent keywords in the language. The **auth+** and **auth-** represent the positive authorization and negative authorization respectively.

inst	***(auth+ | auth-)***	PolicyName "{"
	subject[< type >]	domain-Scope-Expression ;
	target [< type >]	domain-Scope-Expression ;
	action	action-list ;
	[***when***	constraint-Expression ;] "}"

Figure 3-8 Authorization Policy Syntax, from [Dami01], page 3.

The example in Figure 3-9, provided by [Dami01], specifies that trainee test engineers are not authorized to perform tests on routers.

```
inst    auth-  /negativeAuth/testRouter     {
        subject        /testEngineers/trainee ;
        action         performance_test() ;
        target  < routerT>    /routers ;
}
```

Figure 3-9 Negative Authorization Policy, from [Dami01], page 4.

3.7 Extensible Access Control Mark-up Language (XACML)

As of February 18, 2003, OASIS published its Standard for the eXtensible Access Control Mark-up Language (XACML) version 1.0. The original motivation was to design a language that can express the access control policy using a platform independent

language. XML [XML] language is a natural choice because of its wide adoption by many vendors and relative ease of extension with respect to semantics and syntax.

XACML is used together with SAML. It is standardized for access control decision-making. After the SAML assertion reaches the Policy Enforcement Point (PEP), XACML checks it against the policy defined by the resource administrator. Depending on the policy evaluation result, the access decision is made. After that, a SAML authorization decision assertion will be created to pass back for processing.

4. RELATED TRUST MANAGEMENT APPROACHES

This chapter presents a brief survey of related trust management approaches reported in the literature. This is intended to provide a context for the main thesis idea proposed in the next chapter and the remainder of this thesis.

4.1 Decentralized Trust Management System

The term "trust management" was first introduced by Blaze, Feigenbaum and Lacy in the *PolicyMaker* system [Blaz96] [Blaz99] [Chu97]. In [Blaz99], they first distinguished the trust management problem as an important component for computer security. The trust management framework they proposed includes the study of security policies, security credentials, and trust relationships. A comprehensive trust management tool they built is called *PolicyMaker*. By introducing a specific security layer that integrates several security services together, *PolicyMaker* provides a unified mechanism to ensure the privacy of a system.

The design components of *PolicyMaker* include a language that allows a system administrator to define authorization policy. It also provides an engine to enforce the authorization policies. The access control mechanism is defined by the system administrator. The *PolicyMaker* language functions like a query processing language. It accepts some credentials (for example, a public key or a sequence of public keys), a request and a policy, and returns either true or false. The following is an example of a typical format for a query [Blaz96]:

key 1, key 2, ...,key n REQUESTS *ActionString*

The core part for this trust management system using *PolicyMaker* is the compliance check [Blaz98]. The compliance checker applies specific algorithms to the input and

checks if the supplied credentials "constitute a proof that the request complies with the policy" [Blaz98].

Based on the general design prototype of *PolicyMaker*, a simplified version that directly supports public key infrastructure-like applications was developed, namely, the *KeyNote* trust management system [Blaz99]. Using *KeyNote*, users can easily write their own security policy in a standard language. This is a big advantage over traditional applications, where the security policy had to be hard coded into the application. A typical usage of *KeyNote* can be represented by the pseudo-code [Blaz01]:

```
/* At each point the application decides that someone's
   requesting an action, do the following: */

requester        = requesting principal's identifier;
action_description  = data structure describing action;
policy           = data structure describing local policy,
                   typically read from a local file;
credentials      = data structure with any relevant
                   credentials, typically sent along with
                   the request by the requesting principal;

PCV              = Call_KeyNote(requester,
                                action_description,
                                policy,
                                credentials);
if (PCV == "allowed")
   do the requested action
else
   tell principal that action isn't allowed
endif
```

Figure 4-1 [Blaz99] shows the structure of the *KeyNote* trust management system. The part that enclosed by the dash line is the integrated trust management system, where a Boolean value "yes" or "no" will be returned according to the compliance check result.

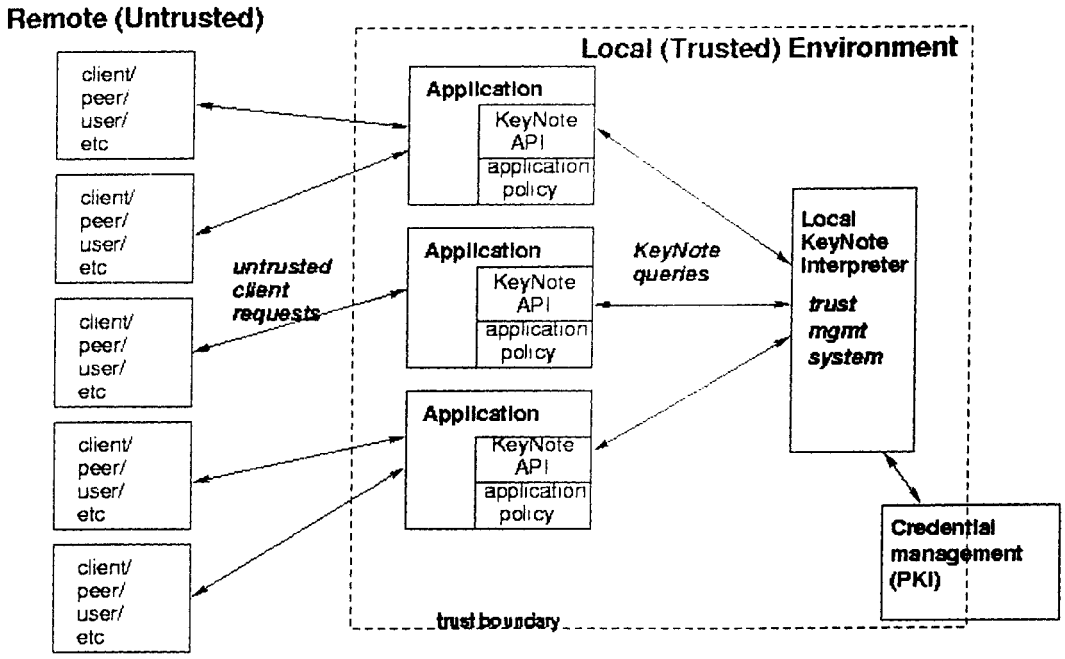

Figure 4-1 KeyNote Trust Management Architecture, From [Blaz99]

Compared with *PolicyMaker*, *KeyNote* has a narrower focus, but is easier to understand and implement.

4.2 A Distributed Trust Model

[Abdu97] proposed a trust model for distributed system that distinguishes the trust into two categories: direct trust, which is the direct trust relationship; and, recommender trust, which is the trust relationship of the recommender. For each category, one assigns a corresponding discrete value table. Direct trust values are –1, 0, 1, 2, 3 and 4, which represent distrust, ignorance, minimal, average, good and complete trust respectively. For the recommender trust values, -1 and 0 are used to represent in discrete enumerated

representation the distrust and ignorance, and 1,2,3,4 are used to represent the closeness of the recommender's own analog judgement.

The assumption of the [Abdu97] trust model is that the trust is transitive under some condition. The algorithm used to compute the trust value along the recommendation path is:

$$tv(T) = tv(R1) / 4 * tv(R2) / 4 *...* tv(Rn) / 4 * rtv(T),$$

In which the tv (Ri) is the recommender trust value of recommenders in the return path. rtv (T) is the recommended trust value of target T given in the recommendation. $tv_p(T)$ is the final trust value that derived from the recommendations. In case there are several trust values obtained by independent recommendation paths, these values are averaged out to obtain the combined trust value.

The above-defined algorithm is quite intuitive; but, it lacks any formal proof. The Figure 4-2 illustrates the usage [Abdu97].

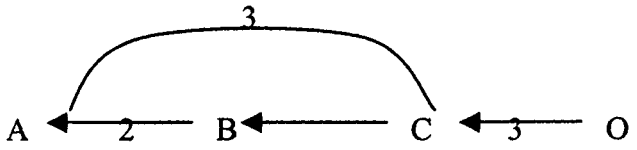

Figure 4-2. A Distributed Trust Model

Assume that A trusts B's recommendation with value 2 (average), which is a recommender trust value, and A trusts C's recommendation of value 3 (good), because B recommended C, and C had a direct trust value of 3 about O. Then applying the algorithm:

$$tv(O) = tv(B) /4*tv(C) /4*rtv(O) =2/4*3/4*3=1.125$$

If another value is obtained from another recommendation path, say, 2.5, then the combined trust value will be computed by taking the average of these two using the algorithm suggested in [Beth94], the resulting value is 2.375.

4.3 Opinion-Based Filtering Through Trust

A recommender system can be used as an information resource when evaluating a user's trust degree. The opinions about the user are collected and analyzed to make a recommendation. [Mont02] proposed an opinion-based filtering approach used to select appropriate opinions to make decisions that result in specific recommendations.

In order to get reliable opinion that accurately reflects the truth about an entity, reliable recommender has to be asked for an opinion. In this suggested framework, each agent has to set up a *friend* list first. Then they can ask their friends (with higher trust values) about the opinion of unknown parties. Then they filter the opinions through aggregation to reach consensus. The process of building the friend list is a proactive procedure. Each agent sends inquiries about known items to other agents and collects their opinions, the trust value about each agent is assigned based on the similarity of the obtained opinion and the agent's own opinion.

The formula used to compute the trust degree of other agent is:

$$t_{q,e} = \frac{\sum_{i=1}^{|P_t|} \delta_{pi} \left(1 - \left| v_{qi} - v_{ei} \right|\right)}{\sum_{i=1}^{|P_t|} \delta_{qi}}$$

in which: $t_{q,e}$ represents the trust of agent q to agent e; $P_1,\ldots P_t$ are known items; and δ is the relevance of the products according to the query agent's interest, the initial value of δ is 1.

According to the formula, if agent q sends out a query about five products, and the collected opinions from agent e are 0.9, 0.5, 0.8, 0.6, 0.4 respectively, and the agent q's opinion about these five products are: 0.9, 0.6, 0.7, 0.5, 0.2. Then the trust degree, which is also the similarity of the two opinions, can be computed as:

$$((1-0)+(1-0.1)+(1-0.1)+(1-0.1)+(1-0.2))/5=3.5/5=0.7$$

A fixed-length list of *friend* is maintained which only keeps the agents with higher trust values. After the friend list builds up, the agent can send out queries about an unknown item and ask their friend's opinion about the known item.

The formula used to compute the value of opinion about a new item is:

$$r_{new} = \frac{\sum_{i}^{|C_q|} t_{q,i} v_{ei,new}}{\sum_{i}^{n} t_{q,i}}$$

in which C_q is the cardinality of the *friend* list. Based on this formula, if agent q has five friends on its friend list, each of them has a trust value given by: 0.9, 0.9, 0.8, 0.8, 0.7, and their opinions about a new item are: 0.6, 0.8, 0.5, 0.6, and 0.9 respectively, then using the above formula, the computed value of opinion is:

$$r=(0.9*0.6+0.9*0.8+0.8*0.5+0.8*0.6+0.7*0.9)/(0.9+0.9+0.8+0.8+0.7) = 2.77/4.1=0.675$$

The agent can use this computed value to make a recommendation decision.
This approach is mainly an opinion average weighted against the trust degree that normalizes into [0,1].

Validation of the results and thorough numerical analysis is lacking in [Mont02].

4.4 Authorization Based on Evidence and Trust

Researchers in the Computer Science Department of Purdue University have suggested one of the authorization mechanisms studied during this thesis research [Yuhu02]. This research work is still under development at the time of this thesis. The proposed authorization framework uses a representation of the evidence and trust, together with a role-mapping server to realize secure authorization. Most of the previous work determines access control only according to the evidence or credential that users provide, but, this proposed authorization mechanism emphasizes the cooperation between the evidence and trust.

In [Yuhu02], they considered a particular representation of trust. An algorithm was developed to compute the reliability of the evidence/credential provided. Based on the evidence/credential, as well as the associated reliability of each evidence/credential, the user was assigned a role according to the local policy for a group of roles.

The architecture of the trust-enhanced role-mapping server has the structure shown in Figure 4-3:

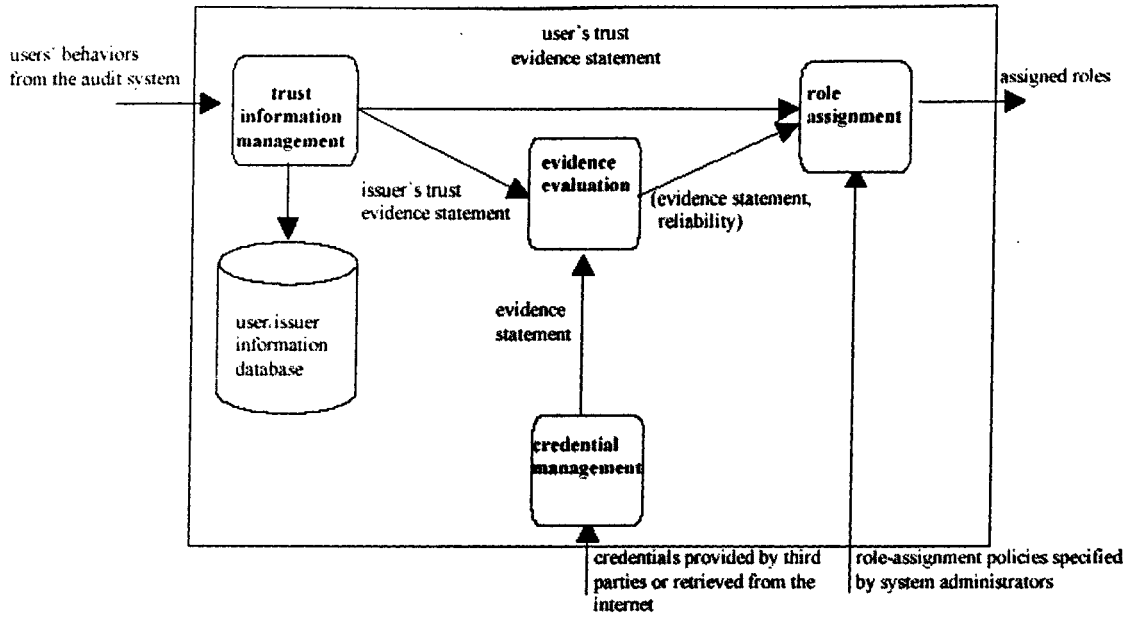

Figure 4-3. Trust-enhanced Role-mapping Server Architecture [Yuhu02]

This approach is an improvement to previous research work that evaluates user trust based only on the third party recommendation. [Yuhu02] has also considered the monitored and recorded previous direct-experience with the user to compute the trustworthiness of the user, as an evolving, dynamic measure.

This algorithm accepts an evidence statement as its input. The output of the algorithm is the reliability of the evidence. This value can then be used to facilitate the decision making to resource access. Obviously, the work in [Yuhu02] is an application using subjective logic of Jøsang. In their work, they did not have much discussion of the logic that supports the implementation. We found on studying their paper that the formula they used to compute the uncertainty is obviously incorrect. Some of their results are lacking proof and proper interpretation, which leads to incorrect decision-making results.

5. PROPOSED TRUST MODEL FOR SINGLE-SIGN-ON

In this chapter, we present the essential details of our proposed trust model to support Single-Sign-On authorization.

5.1 Motivations

Motivated by the related research work of [Yuhu02] and the subjective logic proposed by Jøsang [Jøsa00], as well as the known problems existing in the current Single-Sign-On approach, we propose in this thesis a trust based, authorization-enhanced framework for distributed systems that utilizes Single-Sign-On as an authentication and authorization mechanism. That is, the algorithm in [Yuhu02] is applied to different kinds of evidence. Thus, our work is intended to provide an extended application of the algorithm to a security assertion that is generated based on the user credential/evidence. Also, we explore the combination of different independent opinions. After analysing all relevant algorithms in the related work that are used to compute the trust degree, we find that many of them are applied without any verification of fundamental correctness. We realize that such verification can be difficult in principle, since there does not exist a standard way of measuring the trust degree due to the subjective nature of the trust. Generally speaking, however, we assert that the current subjective logic derived from Dempster-Shafer theory is a better choice for modeling trust than other models discussed in chapter 4. All those models represent, to varying degrees, combinations of classical statistics and logic reasoning applied to the artificial intelligence area, but all are fundamentally conjectures.

In this thesis, we report results of a numerical study and partial analysis of the tested data and some observed patterns, which can then be used to formulate conditions for applying the algorithms. We considered the data obtained from the algorithm in the framework of common human reasoning to distinguish between normal and abnormal situations.

We propose using our algorithm, utilizing both discounting and consensus operators, to obtain a measure of the opinion about an entity, in order to support decision-making. The algorithm itself has many advantages over similar work proposed in the literature [Yuhu02]. Evaluation of the trust degree of the assertion is expected to have supportive impact on the wider adoption of the Single-Sign-On approach.

5.1.1 Assumptions of Our Approach

A complete design and implementation of the authorization-enhanced framework requires a comprehensive monitoring and auditing system to provide necessary and fundamental system input information, since the trust environment is dynamically changing. Prior trust about an entity must evolve according to the information updates obtained from the monitoring system. A monitoring system has to be developed to monitor users' behaviour.

In general, user behaviour on one system can be grouped into categories. All user operations will be recorded. Increased amounts of normal operations will increase user trustworthiness. Harmful operations are recorded and used to decrease user trust. The original authentication of the user is also recorded. This information is used to build and update the databases of trust relationships with other sites. The authorization system is intended to provide information to a role assignment server and access control server. Local security policies can be designed and applied by a system administrator, thereby, guaranteeing the autonomy of each site with respect to security policy. Finally, we assume all the participating parties in a Single-Sign-On domain use Role Based Access Control (RBAC), which is the current prevailing access control mechanism.

In summary, we propose an authorization-enhanced mechanism for access control decision-making.

5.1.2 Methodology

There are three types of assertions defined in the SAML specification, namely, the authentication assertion, the attribute assertion and the authorization assertion. We already described that in section 3.5. In our proposed model, since the pre-established trust relationship either does not exist or is very loose, to pass along information such as the authorization result (access control decision) and attribute associated with the user at one site to another are meaningless. We are concerned only with the authentication assertion, which is used to verify that the user is who he claims he is. We extended the authentication assertion by proposing the addition of several new fields to the original SAML authentication assertion. At each site, we apply our algorithm to compute the trust degree about the user identity. The computed value is then used to make a role-assignment decision.

We analysed and compared our approach with other related work to determine its effectiveness.

We propose two trust models for Single-Sign-On in a large distributed system based on two scenarios of usage. The first model is the Peer-to-Peer model, in which only two sites are involved in computing the trust degree of a user. The second model is a multi-party model, in which opinions from many sites are considered. Two different operators are used to compute the combined opinion and generate the trust degree. In the Peer-to-Peer model we use the discounting operator; in the multi-party model the discounting and consensus operators are used together. The following sections present these two models in detail.

5.2 Peer-to-Peer model

5.2.1 Definitions

In our proposed Peer-to-Peer trust model only two parties are involved in computing the trust degree; this is an intuitive, simpler model for computing transitivity of opinion.

Figure 5-1 illustrates the Peer-to-Peer usage scenario.

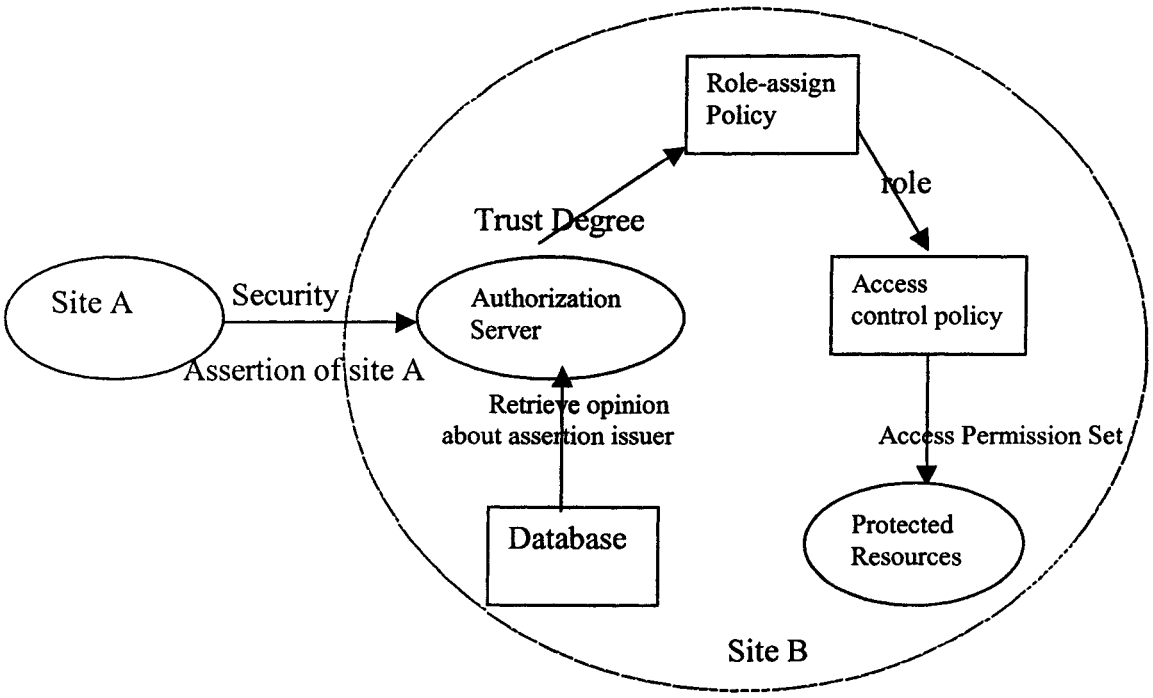

Figure 5-1 Use Scenario for Authorization-Enhanced Security System (Peer-to-Peer model)

The process starts with a user signing on to one site by providing some form of credentials, such as passwords or digital certificates. After the first site evaluates the credentials and makes the access control decision, a security assertion is generated that includes the information about the issuer and the subject, as well as the issuer's opinion

about the subject (represented as a [b, d, u] triple). This assertion is then transmitted to the next site along with the user's request. When the second site server receives this assertion, it extracts first the needed information from the assertion, especially the issuer identification and issuer's opinion of the user. Then, the server must retrieve information from the local database to get the local server opinion about the issuer. Using these two opinions about the site2-to-site1 and site1-to-user trust relationships, the combined trust degree of the assertion is computed using the discounting operator. The value obtained can then be used for role assignment and the access control decision.

Definition 5-1: Opinion is represented by a triple (b, d, u), in which b, d and u represent belief, disbelief and uncertainty, respectively; they satisfy: b+d+u=1, and b, d, u∈[0,1].

The opinion is usually associated with a statement. It represents the opinion about the trustworthiness of the statement.

Definition 5-2: The **Probability Expectation** is the degree of trust represented by an opinion.
Let O = (b, d, u) be an opinion and E (O) denote the probability expectation of the opinion O; then, E (O)= b+0.5*u.

Definition 5-3: Assertion is of the form *<assertion_id, valid_time, issuer, subject, authentication_type, opinion>*. The *assertion_id* is the unique id that refers to the assertion. The *valid_time* denotes the expiration time of the assertion. The issuer is the party who creates the assertion; *subject* is the user that this assertion refers to. The *authentication_type* is the type of the evidence provided by the user for authentication. The last argument *opinion*, is the subject of our current research interest, carries the information about the issuer's degree of trust towards the subject expressed in the form of a belief, disbelief and uncertainty triple.

Based on the above definitions, a typical authentication assertion looks like the following:

<"1234", "2003-2-14T14:58:34", "www.abc.com", "Bob", "password", (0.5, 0.2, 0.3) >

This assertion reveals the following information:

1. The assertion has a unique id, which is 1234.
2. The assertion was created by www.abc.com, and it will expire at 14:58:34, February 14, 2003.
3. The assertion is for the user Bob, who used password to authenticate himself at site www.abc.com.
4. When www.abc.com creates this assertion, it assigns belief about Bob's evidence of 0.5, disbelief about his evidence 0.2, and due to incomplete knowledge about Bob, the uncertainty is 0.3.

Definition 5-4: Statement_peer denotes the trustworthiness towards a peer site member. It is represented by a quadruple *<I, peer, opinion>*. *I* is the local server where the authorization decision is made. *Peer* is the assertion issuer. Testify denotes that the referred peer will provide accurate information about a user. The opinion has the same format as the opinion in the assertion. The opinion expresses how much the decision maker *I* believes the *peer* member.

Definition 5-5: Statement_assertion denotes the trustworthiness towards an assertion. It is represented by a quadruple *<I, subject, assertion, opinion>*. *I* is the server where the authorization decision is made. *Subject* is the assertion subject. *Assertion* is the information the local server used to make the authorization decision. The opinion has the same format as the opinion in the assertion, which denotes how much the decision maker believes the user credential.

In reality, the local server identity *I* is redundant. The purpose of introducing *I* here is for completeness of the representation for a trust relationship between a pair.

5.2.2 Algorithm to Evaluate User Trust Degree

Based on the above definitions 5-1 to 5-5, the following algorithm is used to compute the trust degree of the user.

Input: An assertion A1=<*assertion_id, valid_time, issuer, subject, authentication_type, opinion1*>.

Output: The reliability of the assertion. RE (A1).

Step 1: Extract opinion1=(b1, d1, u1) and issuer from A1.

Step 2: Extract opinion2 value of the issuer from the local database. Sp= *(I, peer, opinion).*

Step 3: Compute b3=b1*b2, d3=b1*d2, u3=d1+u1+b1*u2.

Step 4: Create a new assertion S = <*I, subject, assertion, opinion3*>, where opinion3= (b3, d3, u3). And *I* is the local server.

Step 5: Compute the probability expectation for opinion3; this gives us the reliability of the assertion, RE (A1) = b3 + 0.5 * u3.

The following scenario applies the above algorithm and demonstrates the effect of a unified model for quantifying trust of an assertion.

My server (www.123.com) has just received an assertion with the following information:

A1=<"1234", "2003-2-14T14:58:34", "www.abc.com","Bob","password", (0.5,0.2,0.3) >

In order to evaluate the trust of this assertion, first I get the issuer of this assertion by extracting the issuer field, www.abc.com, and also get the opinion (0.5, 0.2, 0.3) enclosed with this assertion.

Then, I check the local database for a www.abc.com entry and retrieve the information about the trust associated with it. Assuming that this information has the following content:

Sp= <"www.123.com", "www.abc.com", (0.8,0.1,0.1)>

This means that according to my experience or knowledge about www.abc.com, my opinion of trustworthiness of www.abc.com is that I believe this site 80%, disbelieve it 10%, and I have uncertainty 10%.

We compute the new, discounted opinion,

b3=0.5*0.8=0.4, d3=0.5*0.1=0.05, u3=0.2+0.3+0.5*0.1=0.55

The statement about the assertion is created with the computed information:

Sa=<"www.123.com", "Bob", "A1", (0.4,0.05,0.55)>

The last step is to compute the probability expectation for opinion3; we obtain:

RE (A1) = 0.4+0.5*0.55=0.675

This quantitative trust degree can be used to make the access control decision, depending on whether RE is greater than or equal to a critical value determined by the site policy at www.123.com.

5.2.3 Numerical Study and Discussion

We performed a numerical study based on the Peer-to-Peer model. We examined patterns where the computed trust degree is not appropriate to be used for decision-making in the context of common-sense reasoning.

Since we concentrated on the effect of the transitivity of the opinions, we did our test based on the following six extreme scenarios (typically defined by one of the b, d, u having value 0 or 1) and one general scenario, these test cases cover typical situations in practice:

1. b2=1. This is equivalent to the local server trusting the assertion issuer's opinion 100%.
2. d2=1, which is 100% disbelief about the assertion issuer's trustworthiness.
3. u2=1, this is the case that the local server does not have any knowledge about the assertion issuer.
4. b2=0, d2 and u2 are not 0 or 1. No evidence to support the belief about the assertion issuer.
5. d2=0, b2 and u2 are not 0 or 1. No evidence to support the disbelief about the assertion issuer.
6. u2=0. This scenario reflects the traditional probability way of evaluating things. In a real situation, this cannot happen, since human knowledge is not perfect. We cannot be absolutely certain about anything.
7. None of the six values: b1, b2, b3, d1, d2 and d3 have value 0 or 1.

In the following tables of results, the first three columns represent the first opinion (assertion issuer's trust about the user credential), with belief, disbelief and uncertainty, respectively. The second three columns represent the second opinion, the trust relationship between the asserting party and the accepting party. The last four columns are the computed opinion (belief, disbelief and uncertainty) and the probability expectation value.

Table 5-1 represents the local server's belief towards the asserting party is 100%, and asserting party's opinion about user credential has no uncertainty. The result shows that in the case that the assertion issuer's opinions are totally believable, the belief about the user b1 is mapped into the final opinion b3, but the summation of the disbelief and uncertainty about the user d1+u1 is mapped into uncertainty of the final opinion u3. This results in an increase of the probability expectation from opinion 1 to opinion 3 for some cases. In the real trust model, the final trustworthiness should not be better than the original trust relationships. Further, we observed that most of the unusual cases occurred at those entries with 0 or lower belief in opinion 1.

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	1.00	0.00	0.00	1.00	1.00	0.00	0.00	1.00
0.00	0.50	0.50	0.25	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	1.00	0.00	0.00	1.00	0.50	0.00	0.50	0.75
0.20	0.00	0.80	0.50	1.00	0.00	0.00	1.00	0.20	0.00	0.80	0.60
0.80	0.00	0.20	0.90	1.00	0.00	0.00	1.00	0.80	0.00	0.20	0.90
0.50	0.50	0.00	0.50	1.00	0.00	0.00	1.00	0.50	0.00	0.50	0.75
0.20	0.80	0.00	0.20	1.00	0.00	0.00	1.00	0.20	0.00	0.80	0.60
0.80	0.20	0.00	0.80	1.00	0.00	0.00	1.00	0.80	0.00	0.20	0.90

Table 5-1 Opinion in Peer-to-Peer Model

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.20	0.00	0.80	0.60	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.80	0.00	0.20	0.90	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.50	0.50	0.00	0.50	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.20	0.80	0.00	0.20	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.80	0.20	0.00	0.80	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50

Table 5-2 Opinion in Peer-to-Peer Model

Table 5-2 shows a similar pattern. In this case, the local server's opinion about the assertion issuer is totally uncertain, which means that the local server has no previous knowledge about this assertion issuer. Again, for entries with low belief in opinion 1, the computed probability expectation of opinion 3 is not suitable to be used for decision-making.

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.50	0.00	0.50	0.75	0.50	0.00	0.50	0.75
0.00	0.50	0.50	0.25	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.50	0.00	0.50	0.75	0.25	0.00	0.75	0.65
0.20	0.00	0.80	0.60	0.50	0.00	0.50	0.75	0.10	0.00	0.90	0.55
0.80	0.00	0.20	0.90	0.50	0.00	0.50	0.75	0.40	0.00	0.60	0.70
0.50	0.50	0.00	0.50	0.50	0.00	0.50	0.75	0.25	0.00	0.75	0.62
0.20	0.00	0.80	0.60	0.50	0.00	0.50	0.75	0.10	0.00	0.90	0.55
0.80	0.00	0.20	0.90	0.50	0.00	0.50	0.75	0.40	0.00	0.60	0.70

Table 5-3 Opinion in Peer-to-Peer Model

Table 5-3 shows that when the belief and uncertainty are both 0.5 in opinion 2, the enlarged probability expectation value again occurs at the entries with 0 belief in opinion 1.

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.20	0.00	0.80	0.60	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.20	0.00	0.80	0.60	0.00	1.00	0.00	0.00	0.00	0.20	0.80	0.40
0.20	0.00	0.80	0.60	1.00	0.00	0.00	1.00	0.20	0.00	0.80	0.60
0.20	0.00	0.80	0.60	0.00	0.50	0.50	0.25	0.00	0.10	0.90	0.45
0.20	0.00	0.80	0.60	0.00	0.80	0.20	0.10	0.00	0.16	0.84	0.42
0.20	0.00	0.80	0.60	0.00	0.20	0.80	0.40	0.00	0.04	0.96	0.48
0.20	0.00	0.80	0.60	0.50	0.00	0.50	0.75	0.10	0.00	0.90	0.55
0.20	0.00	0.80	0.60	0.20	0.00	0.80	0.50	0.04	0.00	0.90	0.52
0.20	0.00	0.80	0.60	0.80	0.00	0.20	0.90	0.16	0.00	0.84	0.58
0.20	0.00	0.80	0.60	0.50	0.50	0.00	0.50	0.10	0.10	0.80	0.50
0.20	0.00	0.80	0.60	0.20	0.80	0.00	0.20	0.04	0.16	0.80	0.44
0.20	0.00	0.80	0.60	0.80	0.20	0.00	0.80	0.16	0.04	0.80	0.56

Table 5-4 Opinion in Peer-to-Peer Model

Table 5-4 provides examples of using fixed value of opinion1 to test the computed opinion3. The probability expectation pattern is normal, but due to the low belief in opinion1, the computed uncertainty pe1 is still very high, which is very unreasonable.

In analysing the results of our tests, we found that the algorithm used by [Yuhu02], derived from Dempster-Shafer theory [Shaf76] [Shaf90] and subjective logic [Jøsa02], is not appropriate in some cases. Constraints must be added when applying the algorithm.

We summarize the features of this model as follows:

1. Two opinions are used to compute the final trust; namely, the assertion issuer's opinion about the user and the local server's opinion about the assertion issuer.
2. The assertion issuer's opinion is a subjective opinion made by assertion issuer. We do not have control over this value.
3. The local server's opinion about the assertion issuer is stored in the local database, and the opinion values about the assertion issuer are changing over time. A monitoring system could be used to record the number of transactions with the assertion issuer, the effectiveness of assertion issuer's past opinion, and so on. We explore this idea further in section 5.4 and also in the discussion of future work in chapter 7.
4. In the simulation process, there are a total of four free parameters, two for each opinion (the third one can be computed by 1 minus the sum of the first and the second).

We use the numerical simulation study of the algorithm to show the effectiveness of the trust transition. We will analyse and discuss typical cases.

Generally, we find that if none of the values of six parameters is zero or one, the computed result conforms to common sense human reasoning.

We carried out similar tests, which run through most of the typical cases. After analysing the generated results, we came to the following conclusions about applying the subjective logic algorithm to compute the discounting of two opinions.

1. The computed probability expectation value has to be processed and properly interpreted before usage. The basic process is taking the minimum of the computed PE value and the PE value for opinion 1.
2. We consider it inappropriate to use the computed value of probability expectation alone to make access control decision. We must consider other conditions as well.
3. Before applying the algorithm, the fields in opinion1 and opinion2 must be examined to eliminate 0 or 1 cases that represent the extreme cases.
4. By observation, we find that the lower the belief in opinion1, the more doubt the computed probability expectation expressed. We suggest that a threshold be set for the belief value in opinion1. If the value is lower then the threshold, the user is immediately rejected for access control, and hence no assertion will be generated for further transactions.

If the belief value is higher than the threshold, the basic access control is granted, which is equivalent to a "yes" to first step of the access control. Then the fine-grained authorization mechanism can be imposed to connect to the RBAC mechanism.

At this point, it is up to the local security policy to decide the role the user may hold according to the probability expectation value. This way the local security autonomy is guaranteed.

5.3 Improved Multi-Party Model

In the previous section we proposed a Peer-to-Peer model. Obviously that model suffers from the chain effect, in which a previous wrong opinion can be propagated without being noticed. Also in that model, except for the original authenticator, the only information about the user each site receives is the assertion from the previous site. This

kind of information can be very unreliable for decision-making. In this section, we propose an improved model, the multi-party model to compute the trust degree using opinion. This model will address the chain effect problem. The following model is an application of combination of both discounting operator and consensus operator proposed in [Jøsa99].

5.3.1 Definitions

In a multi-party model, the local server will use many previous sites' opinions (from the time the user first signs in) to calculate the trust degree.

In this model, the assertion is defined as:

Definition 5-6: Assertion is of the form *<assertion_id, valid_time, issuer, subject, authentication_type, origin, opinion_o, opinion>*. The *assertion_id* is the unique id that refers to the assertion. The *valid_time* denotes the expiration time of the assertion. The issuer is the party who creates the assertion; *subject* is the user that this assertion refers to. The *authentication_type* is the type of the evidence provided by the user for authentication. The *origin* is the site where the user gets authenticated. Originally, only the origin verified the user's credential and made the first subjective opinion about the trustworthiness of the user's identity. The *opinion_o* is the authenticator's opinion. And the last one, *opinion,* is the assertion issuer's opinion about the user, in the form of a belief, disbelief and uncertainty triple. An example of an assertion looks like the following:

<"1234", "2003-6-14T14:58:34", "www.abc.com", "Alice", "Password", "www.xyz.com", (0.8 0.1 0.1), (0.7, 0.2 0.1)>.

This assertion reveals the information about the user, the original authenticator and the assertion issuer. The assertion has an unique id "1234", the valid time for this assertion is "2003-6-14T14:58:34", the assertion is generated by www.abc.com. The user Alice was

authenticated at www.xyz.com. Where www.xyz.com has an opinion about Alice's trustworthiness as (0.8,0.1,0.1). The assertion issuer www.abc.com has an opinion of (0.7,0.2 0.1) about Alice's identity.

The definition for Statement_peer is the same with definition 5-4, and the definition for Statement_assertion is the same with the definition 5-5.

Using these definitions, we provide now a general format for an assertion in an assertion chain that includes all the necessary information about the previous sites' opinions and information about the user.

5.3.2 Algorithm to Evaluate User Trust Degree

Input: An assertion A= <*assertion_id, valid_time, issuer, subject, authentication_type, origin, opinion_o, opinion*>

Output: The reliability of the assertion towards the identity of the subject RE (A).

Step 1: Extract *assertion_ssuer*, *origin*, *opinion_o* and *opinion* from A.

Step 2: Find entry in local database, get local opinion about the origin (b^l_o, d^l_o, u^l_o), extract opinion_o (b^o_s, d^o_s, u^o_s).

Step3: if (Issuer==origin)

Apply discounting operator:

$(b^o_s, d^o_s, u^o_s) \otimes (b^l_o, d^l_o, u^l_o) = (b^l_s, d^l_s, u^l_s)$

else

Apply discounting operator:

$(b^o_s, d^o_s, u^o_s) \otimes (b^l_o, d^l_o, u^l_o) = (b^l_s, d^l_s, u^l_s)$

Then extract opinion (b^i_s, d^i_s, u^i_s) , apply consensus operator:

$(b^l_s, d^l_s, u^l_s) \oplus (b^i_s, d^i_s, u^i_s) = (b^l_s, d^l_s, u^l_s)$

Step 4: Compute the probability expectation for final opinion. This gives us the reliability of the assertion. RE (A) = $b^l_s+0.5*u^l_s$.

In the algorithm above, the discounting operation follows:

$(b1,d1,u1) \otimes (b2,d2,u2) = (b3,d3,u3)$, where

$$b3=b1*b2;$$

$$d3=b1*d2;$$

$$u3=d1+u1+b1*u2;$$

The consensus operation follows:

$(b1,d1,u1) \oplus (b2,d2,u2) = (b3,d3,u3)$, where

$$b3=(b1*u2+b2*u1)/k;$$

$$d3=(d1*u2+d2*u1)/k;$$

$$u3=u1*u2/k;$$

$$k=u1+u2-u1*u2;$$

Figure 5-2 shows an application of the multi-party model. User x submitted his credential and got authenticated at site A. After reviewing x's credential, A forms an opinion about the trustworthiness of x's identity, then A generates a security assertion that includes his opinion about x. When x requests to access B's resource, the security assertion generated by A will pass along together with x's request to B. At site B, B first extracts the assertion and assertion issuer, then B applies a discounting operator to A's opinion of x and B's opinion of A; the computed opinion will then be used for B's access control decision-making. At site C, C first applies discounting operator with A's opinion about x and C's opinion to A, then applies the consensus operator with the computed opinion and B's opinion to obtain the new opinion. From this scenario, we can see each site uses the trust relationship with A (the original authenticator, also the only site that can "see" the user's actual credential) to compute the discounted opinion. And the resulting opinion is then determined through consensus with other opinions that were obtained independently about the same user x.

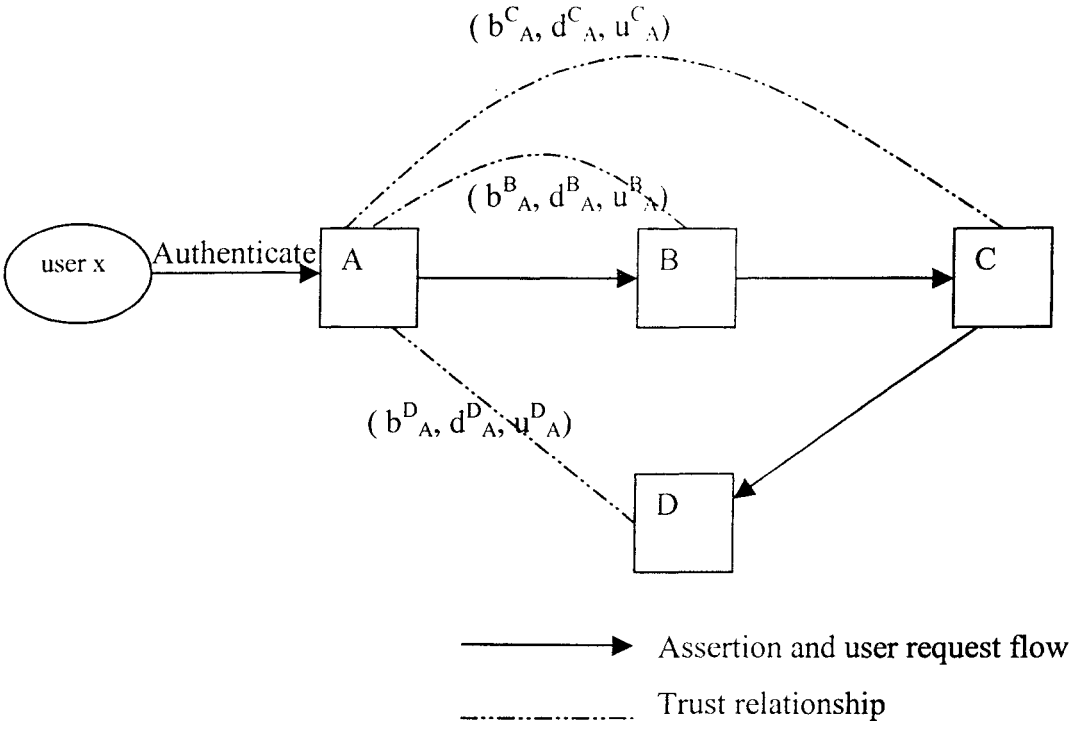

Figure 5-2 Multi-party Model Use Scenario

In Figure 5-2, assume that A's opinion towards x is (0.9, 0.05, 0.05), B's trust to A is (0.8, 0,1,0.1), C's trust to A is (0.7,0.1,0.2), D's trust to A is (0.5, 0,2,0.3). Then, after applying the algorithm, the computed B's opinion to x is (0.72, 0.09, 0.19), with probability expectation being 0.815; C's opinion to x is (0.771, 0.101, 0.128), probability expectation is 0.835; and D's opinion to x is (0.761, 0.134, 0.105), the probability expectation value is 0.813. So D's opinion is the result opinion that has taken B and C's opinion into consideration.

5.3.3 Numerical Study and Discussion

For the multi-party model, we performed a detailed numerical study to show some typical cases and the advantages of the model. The following numerical results represent several typical cases using this model. Our experiments simulate the use-case scenario described in the previous section 5.3.2 with four sites A, B, C, D involved. The user is represented by x. In the following tables, the first three columns represent the belief, disbelief, and

uncertainty in the corresponding opinions. The last column represents the probability expectation for each opinion. The rows represent different opinions. The opinion denoted using the qualifier "alone" represent the result after applying only the discounting operator, and the remaining opinion is the result after applying consensus operator with all previous site's opinion.

Table 5-5 shows the scenario in which B, C and D all have the same opinion about the trustworthiness of A. Then, after applying the discounting operator, the obtained opinions about x are all the same. Here we see the functionality of the consensus operator. In the highlighted rows for opinion C to x and opinion D to x, the trust towards x is strengthened by the other sites' independent positive opinions, which results a higher belief and probability expectation values compared with the result using only discounting operator. This is the ideal case we expect to see. There are no special values such as 0 or 1 appearing in the opinion fields.

	b	d	u	pe
Opinion B-A	0.800	0.100	0.100	0.850
Opinion C-A	0.800	0.100	0.100	0.850
Opinion D-A	0.800	0.100	0.100	0.850
Opinion A-x	0.800	0.100	0.100	0.850
Opinion B-x	0.640	0.080	0.280	0.780
Opinion C-x-alone	0.640	0.080	0.280	0.780
Opinion C-x	0.744	0.093	0.163	0.826
Opinion D-x-alone	0.640	0.080	0.280	0.780
Opinion D-x	0.787	0.098	0.115	0.844

Table 5-5 Opinion in Multi-party Model

Table 5-6 and 5-7 shows the adjustment of one low belief towards A. In Table 5-6, the low belief of C towards user x has been adjusted from 0.08 to 0.633 by applying the consensus operator with B's opinion, and the belief of D to x is further strengthened to 0.737. Table 5-7 varied the position of low belief in the trust chain. Interestingly but not surprisingly, we find out the resulting opinions of D towards x are the same. This conforms to the commutative and associative of the consensus operator.

	b	d	u	pe
Opinion B-A	0.800	0.100	0.100	0.850
Opinion C-A	0.100	0.100	0.800	0.500
Opinion D-A	0.800	0.100	0.100	0.850
Opinion A-x	0.800	0.100	0.100	0.850
Opinion B-x	0.640	0.080	0.280	0.780
Opinion C-x-alone	0.080	0.080	0.840	0.500
Opinion C-x	0.633	0.101	0.266	0.766
Opinion D-x-alone	0.640	0.080	0.280	0.780
Opinion D-x	0.737	0.105	0.158	0.816

Table 5-6 Opinion in Multi-party Model

	b	d	u	pe
Opinion B-A	0.800	0.100	0.100	0.850
Opinion C-A	0.800	0.100	0.100	0.850
Opinion D-A	0.100	0.100	0.800	0.500
Opinion A-x	0.800	0.100	0.100	0.850
Opinion B-x	0.640	0.080	0.280	0.780
Opinion C-x-alone	0.640	0.080	0.280	0.780
Opinion C-x	0.744	0.093	0.163	0.826
Opinion D-x-alone	0.080	0.080	0.840	0.500
Opinion D-x	0.737	0.105	0.158	0.816

Table 5-7 Opinion in Multi-party Model

The following table 5-8 shows one typical abnormal scenario, in which both C and D have 100% uncertainty about A, in this case, they rely on B's opinion about A to make their decision. This scenario reflects and suggested some restriction for using this trust model. We noticed that 0 or 1 appeared in the opinion filed, which might cause abnormal result.

	b	d	u	pe
Opinion B-A	0.800	0.100	0.100	0.850
Opinion C-A	0.000	0.000	1.000	0.500
Opinion D-A	0.000	0.000	1.000	0.500
Opinion A-x	0.900	0.050	0.050	0.925
Opinion B-x	0.720	0.090	0.190	0.815
Opinion C-x-alone	0.000	0.000	1.000	0.500
Opinion C-x	0.720	0.090	0.190	0.815
Opinion D-x-alone	0.000	0.000	1.000	0.500
Opinion D-x	0.720	0.090	0.190	0.815

Table 5-8 Opinion in Multi-party Model

Based on the observation, we propose several conditions for using this trust model:

1. The authenticator's belief towards the user should be greater than 0.5.
2. A newly formatted system should take some time to reach stability (to build up trust relationship between each other) before the computed value can be used to make decision.

Further verification of these conditions has to be done by the future simulation work.

5.4 Monitoring System

Monitoring systems play an important role in the whole trust model. This is because trust cannot be measured statically. Any trust relationship has to evolve with time and changing of related information. We suggest a basic approach for calculating and updating the belief, disbelief and uncertainty. This approach conforms to the mapping between evidence and opinion space proposed by [Jøsa99].

We use r to represent the number of transactions that support the belief towards a peer member site. And we use s to represent the number of transactions that support the disbelief towards a peer member. The belief, disbelief and uncertainty can be represented by the following formulas:

$$\begin{cases} b = \dfrac{r}{r+s+2} \\ d = \dfrac{s}{r+s+2} \\ u = \dfrac{2}{r+s+2} \end{cases} \qquad \text{Where } u \neq 0$$

A database is required to build up the user and peer site trust information. The positive and negative transactions are recorded.

Although our trust model for single-sign-on will increase the scalability of current approaches, there still exist implementation problems. In order to apply effectively the trust model in a working environment, each site has to build up a database for all known peers, since every site can be a potential authenticator who gives the subjective opinion about the user and generates the first authentication assertion.

Another drawback to this approach, also a drawback of subjective logic itself, is that the original assertion made by the authenticator site is still a subjective opinion, and this opinion will influence the other sites' decision making.

Compared with the previous Peer-to-Peer model, the multi-party model has several advantages. In the Peer-to-Peer model, since only discounting operator is used, the resulting belief is obtained by multiplying two beliefs. Since the maximum value of belief is 1 and in real situations 100% belief is almost impossible, the computed derived belief towards an entity is always decreasing. Also, in this model, for each site, other than the actual authenticator site, the only information obtained is through the security assertion passed along by the previous site. Thus, inaccurately computed or subjective errors will propagate rapidly, thereby, limiting the length of the assertion chain.

For the multi-party model the above-mentioned problems don't exist. The local opinion is computed by discounting the authenticator's opinion and the trust relationship with the authenticator, which is more direct then the Peer-to-Peer model. Another important aspect

is the utilization of the consensus operator. By using this operator, other sites' independent opinions have also been taken into consideration. This will lead to a more comprehensive evaluation about the user. The chain effect is minimized by applying the consensus operator. With increasing numbers of asserting parties involved in, the subjectivity of the opinion is greatly reduced.

Currently, we only consider in the trust towards the statement that the peer site will provide accurate information related to the user authentication assertion. In the future, more assertions should be considered, such as the authorization assertion, in which the authorization decision information made by the assertion issuer is stored. This information can be used by the local server to make access decision.

5.5 Application in Cross-Domain Authentication

Another problem with the current Single-Sign-On approach is the cross-domain authentication. In the current SSO approach, all sites are divided into groups, sometimes called a federation. Each federation has their own pre-established domain based trust relationship. Whenever a new party joins into the federation, the trust relationship has to be reset in order to accommodate the new party. This approach has not only greatly reduced the scalability of the system, but also prevents application of the cross-domain authentication.

During the process of completing this thesis, we discover that our proposed model might be useful in improving and at least simplifying the current cross-domain authentication approach.

The following scenario shows the current approach and our suggested approach:

In order to achieve the cross-domain authentication, current systems have to rely on a trusted third party, namely the CA (Certificate Authority), to get authentication information about a user. If an outsider wants to access the resources within a federation

that he does not belong to, the current SAML-like approaches will definitely fail, since the pre-established trust relationship does not exist. The safe approach is to have this outsider verified independently by each site with the CA.

In our proposed model, after the first site has verified the outsider identity with the CA, then he forms an opinion about this outsider. This opinion is then transmitted together with the assertion. By applying our proposed model, we envision that the first site that did the authentication with CA actually acts as a gatekeeper for the federation. Consequently, information about this user can then be obtained from this site's opinion about the outsider.

We recommend that further research in this respect should be conducted.

6. FEATURES AND COMPARISON WITH OTHER WORK

6.1 Features

Our proposed framework for a security-enhanced authorization system has several unique features that are lacking in current authorization mechanism:

- Flexibility: The fine-grained functionality of each component can be easily modified and rearranged to accommodate different user and system requirements.

- Improved scalability and security. In an open network, these two features usually contradict one another. Increased scalability leads to potential easier security breaches due to lack of sufficient knowledge of new users. Our system has not only revolutionized the expansion of the Single-Sign-On community, but also created a reasonable mechanism to ensure and guarantee the security for the whole system as well as individual site security.

- Autonomous local security policy. Our system allows autonomous customization of local security policy. This aspect provides resource providers autonomy in managing their own resources.

6.2 Comparison with Other Work

Our proposed framework for security-enhanced authorization system has many application advantages over the previous related work.

In the following, we compare our work with other related work.

1. Flexibility:

Both *PolicyMaker* and *KeyNote* are examples of trust management systems. They include a well-defined language is used by a system administrator to write security policy, an engine to enforce the authorization policies. The main characteristic about these approaches is that the access control mechanism is defined by the system. The inputs of the security management system are: user credential, user request and the related security policy. The output is either "yes" or "no", which represent the granting or denying of the request.

In order for these systems to work, users must have some knowledge about the specific resources to be accessed at the time of request. On the other hand, the security policy must cover all the combinations of the operations and resources in order to match users' requests. Changes in local resources or operations might require the re-writing of the whole security policy.

Our model provides a more flexible structure that can be easily integrated with existing Role Based Access Control (RBAC) mechanism. Our model assumes all the advantages of RBAC, and is easily adaptable to newer and better access control mechanisms yet to be defined or prescribed.

2. Expandability:

Compared with those applications for subjective logic in certificate and evidence [Yuhu02], our application in single-sign-on area for large distributed system is an extension and improvement for the usage of the theory. It has immediate practical significance as well as providing a firm basis for further investigation. In [Yuhu02], the author explored only the one-step discounting of the opinions without giving justification

of the results obtained. We applied the subjective logic in a more practical way, and did research on the multi-step application.

3. Theoretical Soundness:

Compared with [Abdu97], in which the author proposed dividing trust into six categories and simply multiplied the trust to show the transitivity, our approach provides a better and finer-grained approach to the trust division. By applying the subjective logic derived from the Dempster-Shafer theory in the artificial intelligence field, our proposed model follows from a firmer theoretical and conceptual basis.

4. Implementation Simplicity

Our Single-Sign-On approach is a much simpler approach than GSI SSO. GSI relies on the proxy to create the temporary credential. It uses an identity associated with the temporary credential to access different resources. The implementation of GSI is complex and has many limitations as described in chapter 3. Instead, our approach simplifies the SSO by taking the assertion approach for user identity verification.

5. Less Dependency on Pre-established Trust Relationship

Compared to current SAML approaches for SSO, which relies solely on the pre-established trust relationship among the members in a federation, our model attaches an opinion field with each assertion and dynamically builds up and updates the trust relationship with other peer sites. Thus, the dependency on the pre-established trust relationship is greatly reduced.

7. CONCLUSIONS AND FUTURE WORK

In this thesis we have presented an authorization-enhanced framework for single-sign-on based on subjective logic. Based on subjective logic theory, we first proposed a simple and intuitive Peer-to-Peer model to compute the trust degree by discounting two opinions. We did a numerical study of this model for different user scenarios and generated constraint conditions for applying this model. Further, we pointed out the drawbacks of this model in real applications. Based on that, we suggested an improved model that involves multiple parties. The multi-party model addressed some of the problems existing in the Peer-to-Peer model. It provides an improved approach to evaluate trust towards an entity. This trust relationship is reasonably quantified, and can be used as the reference point to develop and write local security policy, as well as make the access control decisions.

Our approach to applying subjective logic to compute the trust relationship is the first such attempt used in Single-Sign-On approach for large distributed systems. We envision the proposed framework to replace the current single-sign-on approach that relies heavily on the pre-established trust relationship and pre-formed trust federation.

The original and innovative contributions of this thesis can be summarized as follow:

1. Through Numerical study, we pointed out the inadequacy of the related work that directly applies the subjective logic, as discussed in section 5.2.3.
2. We generalized conditions of applying the subjective logic for trust modeling in section 5.2.3.
3. We proposed two trust models that can be applied in Single-Sign-On for large-scaled, distributed computing environment: Two-Party discounting and Multi-party consensus, which are discussed in chapter 5.
4. We performed detailed numerical study of the two models, and showed the reasonability of the proposed model in chapter 5. We noted that further

implementation and testing are required for the applicability of this approach, however.

In the future, we plan to design a simulation of this framework, which will include components such as the monitor system, audit system, authentication server, and so on. We will also explore the application of this approach in cross-domain authentication. Further independent verification of decisions is needed.

REFERENCES

[Abdu97] Abdul-Rahman, A., and Hailes, S. "*A Distributed Trust Model*". In Proceedings of the 1997 New Security Paradigms Workshop, pages 48--60. ACM, 1997.

[Adam02] Adams, C., and Lloys, Steve. "*Understanding PKI, Concepts, Standards, and Deployment Considerations*". Second Edition, Addison-Wesley, October, 2002.

[Beth94] Beth, T., Borchedring, M. and Klein, B. "*Valuation of Trust in Open Networks*". In proceedings, European Symposium on Research in Computer Security 1994, ESORICS94, pages 3--18.

[Blaz96] Blaze, M., Feigenbaum, J. and Lacy, J. "*Decentralized Trust Management*", In Proceedings 1996 IEEE Symposium on Security and Privacy, pages 164-173, May, 1996.

[Blaz98] Blaze, M., Feigenbaum, J. and Strauss, M. "*Compliance-checking in the PolicyMaker Trust Management System*". In Proceeding of Second International Conference on Financial Cryptography (FC'98), volume 1465 of Lecture Note in Computer Science, pages 254-274. Springer, 1998.

[Blaz99] Blaze, M., Feigenbaum, J., Ioannidis, J. and Keromytis, A. "*The Keynote Trust Management System*", Version 2., RFC-2704, IETF, September 1999.

[Blaz01] Blaze, M. "Using the KeyNote Trust Management System", March, 2001. http://www.crypto.com/trustmgt/kn.html

[Carb03] Carbo, J and Molina, J.M. "*Trust Management Through Fuzzy Reputation*". International Journal of Cooperative Information Systems, pg 135-155, Vol. 12, No. 1, March, 2003.

[Chu97] Chu, Y.H., Feigenbaum, J., LaMacchia, B. Resnick, P. and Strauss, M. "*REFEREE: Trust Management for Web Applications*". Word Wide Web Journal, pages 127-139, 1997.

[Dami01] Damianou N., Dulay, N., Lupu E. and Sloman M. "*The Ponder Policy Specification Language*". Lecture Notes in Computer Science, 1995:18—38, January 2001.

[Delf02] Delfs, H and Knebl H. "*Introduction to Cryptography, Principles and Applications*". Berlin Heidelberg New York: Springer-Verlag 2002.

[Ferr95] Ferraiolo, D., Cugini, J. and Kuhn, R. "*Role Based Access Control: Features and Motivations*". Proceedings, Annual Computer Security Applications Conference, IEEE Computer Society Press, 1995.

[Fost98a] Foster, I., Kesselman, C., Tsudik, G. and Tuecke, S. "*A Security Architecture for Computational Grids*". In ACM Conference on Computer and Communication Security, page 83-91, ACM Press, 1998.

[Fost02] Foster, I., Kesselman, C., Nick, M.J. and Turcke, S. "*The Physiology of the Grid. An Open Grid Service Architecture for Distributed System Integration*". http://www.globus.org/research/papers/orsa.pdf.

[Fran98] Frank, N.M. and Peters, L. "*Building Trust: the Importance of Both Task and Social Precursors*". Int'l. Conf. Engineering and Technology Management: Pioneering New Technologies - Management Issues and Challenges in the Third Millennium, 1998, http://ieeexplore.ieee.org/iel4/5884/15675/00727781.pdf

[Haye00] Hayes, J. "*Policy-based Authentication and Authorization: Secure Access to the Network Infrastructure*". Proceeding of the 16th Annual Computer Security Application Conference, 2000.

[Helt97] Helton, J. C. "*Uncertainty and Sensitivity Analysis in the Presence of Stochastic and Subjective Uncertainty*." Journal of Statistical Computation and Simulation 57: 3-76.

[IPSec] http://www.ietf.org/html.charters/ipsec-charter.html

[John99] John, L., Magnus, N. "*Attribute certification: an enabling technology for delegation and role-based controls in distributed environment*". Proc, 4th ACM Workshop on Role-based access control, ACM Press, pg 121-130, 1999.

[Jon02] Jon B. "*Single-sign-on simplicity with SAML: An Overview of Single Sign-on Capabilities Based on the Security Assertions Markup Language (SAML) Specification*". JAVA web site paper, May, 2002.

[Jone99] Jones, S. "*TRUST-EC: Requirements for Trust and Confidence in E-Commerce*". 1999, European Commission, Joint Research Centre.

[Jøsa99] Jøsang A. "*An Algebra for Assessing Trust in Certification Chains*". In Proc. Network and Distributed Systems Security Symposium. The Internet Society, 1999.

[Jøsa00] Jøsang, A., Pedersen, I.G. and Povey, D. "*PKI seeks a trusting relationship*". In Proceedings of the Fifth Australasian Conference on Information Security and Privacy (ACISP 2000), Brisbane, July 2000. Springer.

[Jøsa01] Jøsang A. "*A logic for Uncertain Probabilities*". International Journal of Uncertainty, Fuzziness and Knowledge-Based System, Vol. 9, No. 3, June, 2001.

[Jøsa02] Jøsang A. "*Subjective Evidential Reasoning*". The Proceeding of the 9th International Conference in Information Processing and Management of Uncertainty in Knowledge-Based System (IPMU 2002), Annecy, France, May, 2002.

[Kari02] Kari S. and Scott, F."*Combination of Evidence in Dempster-Shafer Theory*", SAND 2002-0835.

[Kini98] Kini, A. and Choobineh, J. "*Trust in Electronic Commerce: Definition and Theoretical Considerations*". 31st Annual Hawaii Int'l. Conf. System Sciences, 1998, Hawaii, http://ieeexplore.ieee.org/iel4/5217/14270/00655251.pdf

[Mont02] Montaner, M., Lopez, B. and Rosa, J. "*Opinion based filtering through trust*". Submitted to CIA'02.

[OGSA] The Security Architecture for Open Grid Services, www.cs.virginia.edu/~humphrey/ ogsa-sec-wg/OGSASec-ggf5.pdf.

[Oxford] Oxford Reference Dictionary.

[PKI] Internet X.509 Public Key Infrastructure Certificate and CRL Profile (RFC 2459).

[Perl03] Perlowitz, W. http://www.nwfusion.com/news/tech/2003/0421techupdate.html

[PONDER] http://www-dse.doc.ic.ac.uk/policies/

[Rama02] Ramakrishnan, L., Rehn, H., Alameda, J., Ananthakrishnan, R., Govindaraju, M., Slominski, A., Connelly, K., Welch, V., Gannon, D., Bramley, R. and Hampton, S. "*An authorization framework for a grid based component architecture*". Grid Computing - GRID 2002, Third International Workshop, Baltimore, MD, USA, November 18, 2002, Proceedings. Lecture Notes in Computer Science 2536 Springer 2002, ISBN 3-540-00133-6

[RBAC] http://csrc.nist.gov/rbac/

[SAML] http://xml.coverpages.org/saml.html

[Seok02] Seokwon, Y., Herman, L. and Stanley, Y.W.S. "*Trust-based Security Model and Enforcement Mechanism for Web Service Technology*". Technologies for E-Services, Third International Workshop, TES 2002, Hong Kong, China, August 23-24, 2002.

[Shaf76] Shafer, G. "*A mathematical Theory of Evidence*", Princeton University Press, 1976.

[Shaf90] Shafer, G. "*Perspective on the Theory and Practice of Belief Functions*". International Journal of approximate Reasoning 3 1-40, 1990.

[Shaf] http://www.glennshafer.com/assets/downloads/article48.pdf

[Simo97] Simon, R. and Zurko, M. "*Adage: An Architecture for Distributed Authorization*". OSF Research Institute, Cambridge 1997.

[SSL] http://www.openssl.org/

[Tyro00] Tyrone, G. and Morris, S. "*A Survey of Trust in Internet Applications*". IEEE Communications Surveys & Tutorials, 4th Quarter 2000.

[Vara98] Varadharajan, V., Kumar, N. and Mu, Y. "*Security agent based distributed authorizarion: an approach*". Proc. 21st NIST-NCSC National Information Systems Security Conference, pages 315-328, 1998.

[Von03] Von, W. Frank, S., Sam, M. and Laura P. "*Use of SAML for OGSA Authorization*". GWD-R, GGF OGSA Security Working Group document. http"//globus.org/ogsa/Security.

[WebService] Web Services, http://www.w3.org/2002/ws/

[XACML] eXtensible Access Control Markup Language (XACML) Version 1.0, OASIS Standard, 18 February 2003
http://www.oasis-open.org/committees/committees/xacml/repository/

[XML] http://www.w3.org/XML/

[Yuhu02] Yuhui, Z. and Bharat B. "*Authorization Based Evidence and Trust*". 4th International Conference, DaWaK 2002, Aix-en-Provence, France, September 4-6, 2002. Proceedings.

[Yun] Yun, T., Vir, V.P and Ben, C. "*Design of Trust Metrics Based on Dempster-Shafer Theory.*"

[Zimm91] Zimmermann, H.J. "*Fuzzy Set Theory-- and its Applications*". Kluwer: Dordrecht, 1991.

APPENDIX

This appendix contains the full set of experimental data collected for numerical study. Summary data has been presented within the main thesis text of chapter 5.

Peer-to-Peer model:

In the following table, the first four columns represent an opinion (belief, disbelief, uncertainty) of the assertion issuer towards the user and the corresponding probability expectation. The second four columns represent the opinion of the local site towards the assertion issuer and the probability expectation. The last four columns are the computed opinion of the local site towards the user. The highlighted fields identify the abnormal (deviation from human common-sense reasoning) entries we observed.

b2=1

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	1.00	0.00	0.00	1.00	1.00	0.00	0.00	1.00
0.00	0.50	0.50	0.25	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	1.00	0.00	0.00	1.00	0.50	0.00	0.50	0.75
0.20	0.00	0.80	0.50	1.00	0.00	0.00	1.00	0.20	0.00	0.80	0.60
0.80	0.00	0.20	0.90	1.00	0.00	0.00	1.00	0.80	0.00	0.20	0.90
0.50	0.50	0.00	0.50	1.00	0.00	0.00	1.00	0.50	0.00	0.50	0.75
0.20	0.80	0.00	0.20	1.00	0.00	0.00	1.00	0.20	0.00	0.80	0.60
0.80	0.20	0.00	0.80	1.00	0.00	0.00	1.00	0.80	0.00	0.20	0.90

d2=1

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.00	1.00	0.00	0.00	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.00	1.00	0.00	0.00	0.00	1.00	0.00	0.00
0.00	0.50	0.50	0.25	0.00	1.00	0.00	0.00	0.00	0.00	1.00	0.55
0.00	0.80	0.20	0.10	0.00	1.00	0.00	0.00	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.00	1.00	0.00	0.00	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.00	1.00	0.00	0.00	0.00	0.50	0.50	0.25
0.20	0.00	0.80	0.60	0.00	1.00	0.00	0.00	0.00	0.20	0.80	0.40
0.80	0.00	0.20	0.90	0.00	1.00	0.00	0.00	0.00	0.80	0.20	0.10
0.50	0.50	0.00	0.50	0.00	1.00	0.00	0.00	0.00	0.50	0.50	0.25
0.20	0.80	0.00	0.20	0.00	1.00	0.00	0.00	0.00	0.20	0.80	0.40
0.80	0.20	0.00	0.80	0.00	1.00	0.00	0.00	0.00	0.80	0.20	0.10

u2=1

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.20	0.00	0.80	0.60	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.80	0.00	0.20	0.90	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.50	0.50	0.00	0.50	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.20	0.80	0.00	0.20	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.80	0.20	0.00	0.80	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50

d2=0, b2=u2=0.5

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.50	0.00	0.50	0.75	0.50	0.00	0.50	0.75
0.00	0.50	0.50	0.25	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.50	0.00	0.50	0.75	0.25	0.00	0.75	0.65
0.20	0.00	0.80	0.60	0.50	0.00	0.50	0.75	0.10	0.00	0.90	0.55
0.80	0.00	0.20	0.90	0.50	0.00	0.50	0.75	0.40	0.00	0.60	0.70
0.50	0.50	0.00	0.50	0.50	0.00	0.50	0.75	0.25	0.00	0.75	0.62
0.20	0.00	0.80	0.60	0.50	0.00	0.50	0.75	0.10	0.00	0.90	0.55
0.80	0.00	0.20	0.90	0.50	0.00	0.50	0.75	0.40	0.00	0.60	0.70

d2=0, b2=0.8, u2=0.2

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.80	0.00	0.20	0.90	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.80	0.00	0.20	0.90	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.80	0.00	0.20	0.90	0.80	0.00	0.20	0.90
0.00	0.50	0.50	0.25	0.80	0.00	0.20	0.90	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	0.80	0.00	0.20	0.90	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.80	0.00	0.20	0.90	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.80	0.00	0.20	0.90	0.40	0.00	0.60	0.70
0.20	0.00	0.80	0.60	0.80	0.00	0.20	0.90	0.16	0.00	0.84	0.58
0.80	0.00	0.20	0.90	0.80	0.00	0.20	0.90	0.64	0.00	0.36	0.82
0.50	0.50	0.00	0.50	0.80	0.00	0.20	0.90	0.40	0.00	0.60	0.70
0.20	0.80	0.00	0.20	0.80	0.00	0.20	0.90	0.16	0.00	0.84	0.58
0.80	0.20	0.00	0.80	0.80	0.00	0.20	0.90	0.60	0.00	0.36	0.82

d2=0, b2=0.2, u2=0.8

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.20	0.00	0.80	0.60	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.20	0.00	0.80	0.60	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.20	0.00	0.80	0.60	0.20	0.00	0.80	0.60
0.00	0.50	0.50	0.25	0.20	0.00	0.80	0.60	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	0.20	0.00	0.80	0.60	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.20	0.00	0.80	0.60	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.20	0.00	0.80	0.60	0.10	0.00	0.90	0.55
0.20	0.00	0.80	0.60	0.20	0.00	0.80	0.60	0.04	0.00	0.96	0.52
0.80	0.00	0.20	0.90	0.20	0.00	0.80	0.60	0.16	0.00	0.84	0.58
0.50	0.50	0.00	0.50	0.20	0.00	0.80	0.60	0.10	0.00	0.90	0.55
0.20	0.80	0.00	0.20	0.20	0.00	0.80	0.60	0.04	0.00	0.96	0.52
0.80	0.20	0.00	0.80	0.20	0.00	0.80	0.60	0.16	0.00	0.84	0.58

u2=0, b2=d2=0.5

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.50	0.50	0.00	0.50	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.50	0.50	0.00	0.50	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.50	0.50	0.00	0.50	0.50	0.50	0.00	0.50
0.00	0.50	0.50	0.25	0.50	0.50	0.00	0.50	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	0.50	0.50	0.00	0.50	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.50	0.50	0.00	0.50	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.50	0.50	0.00	0.50	0.25	0.25	0.50	0.50
0.20	0.00	0.80	0.80	0.50	0.50	0.00	0.50	0.10	0.10	0.80	0.50
0.80	0.00	0.20	0.90	0.50	0.50	0.00	0.50	0.40	0.40	0.20	0.50
0.50	0.50	0.00	0.50	0.50	0.50	0.00	0.50	0.25	0.25	0.50	0.50
0.20	0.80	0.00	0.20	0.50	0.50	0.00	0.50	0.10	0.10	0.80	0.50
0.80	0.20	0.00	0.80	0.50	0.50	0.00	0.50	0.40	0.40	0.20	0.50

u2=0, b2=0.2, d2=0.8

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.20	0.80	0.00	0.20	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.20	0.80	0.00	0.20	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.20	0.80	0.00	0.20	0.20	0.80	0.00	0.20
0.00	0.50	0.50	0.25	0.20	0.80	0.00	0.20	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	0.20	0.80	0.00	0.20	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.20	0.80	0.00	0.20	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.20	0.80	0.00	0.20	0.10	0.40	0.50	0.35
0.20	0.00	0.80	0.60	0.20	0.80	0.00	0.20	0.04	0.16	0.80	0.44
0.80	0.00	0.20	0.90	0.20	0.80	0.00	0.20	0.16	0.64	0.20	0.26
0.50	0.50	0.00	0.50	0.20	0.80	0.00	0.20	0.10	0.40	0.50	0.35
0.20	0.80	0.00	0.20	0.20	0.80	0.00	0.20	0.04	0.16	0.80	0.44
0.80	0.20	0.00	0.80	0.20	0.80	0.00	0.20	0.16	0.64	0.20	0.26

u2=0, b2=0.8, d2=0.2

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.00	1.00	0.50	0.80	0.20	0.00	0.80	0.00	0.00	1.00	0.50
0.00	1.00	0.00	0.00	0.80	0.20	0.00	0.80	0.00	0.00	1.00	0.50
1.00	0.00	0.00	1.00	0.80	0.20	0.00	0.80	0.80	0.20	0.00	0.80
0.00	0.50	0.50	0.25	0.80	0.20	0.00	0.80	0.00	0.00	1.00	0.50
0.00	0.80	0.20	0.10	0.80	0.20	0.00	0.80	0.00	0.00	1.00	0.50
0.00	0.20	0.80	0.40	0.80	0.20	0.00	0.80	0.00	0.00	1.00	0.50
0.50	0.00	0.50	0.75	0.80	0.20	0.00	0.80	0.40	0.10	0.50	0.65
0.20	0.00	0.80	0.60	0.80	0.20	0.00	0.80	0.16	0.04	0.80	0.56
0.80	0.00	0.20	0.90	0.80	0.20	0.00	0.80	0.64	0.16	0.20	0.74
0.50	0.50	0.00	0.50	0.80	0.20	0.00	0.80	0.40	0.10	0.50	0.65
0.20	0.80	0.00	0.20	0.80	0.20	0.00	0.80	0.16	0.04	0.80	0.56
0.80	0.20	0.00	0.80	0.80	0.20	0.00	0.80	0.64	0.16	0.20	0.74

General cases, where no value is either 0 or 1.

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.30	0.30	0.40	0.50	0.30	0.30	0.40	0.50	0.09	0.09	0.82	0.50
0.10	0.10	0.80	0.50	0.30	0.30	0.40	0.50	0.03	0.03	0.94	0.50
0.10	0.80	0.10	0.105	0.30	0.30	0.40	0.50	0.03	0.03	0.94	0.50
0.80	0.10	0.10	0.80	0.30	0.30	0.40	0.50	0.24	0.24	0.52	0.50
0.30	0.30	0.40	0.50	0.10	0.10	0.80	0.50	0.03	0.00	0.94	0.50
0.30	0.30	0.40	0.50	0.10	0.80	0.10	0.105	0.03	0.24	0.73	0.395
0.30	0.30	0.40	0.50	0.80	0.10	0.10	0.805	0.24	0.03	0.73	0.605

Additional: Fix opinion 1

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.50	0.50	0.00	0.50	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.50	0.50	0.00	0.50	0.00	1.00	0.00	0.00	0.00	0.50	0.50	0.25
0.50	0.50	0.00	0.50	1.00	0.00	0.00	1.00	0.50	0.00	0.50	0.75
0.50	0.50	0.00	0.50	0.00	0.50	0.50	0.25	0.00	0.25	0.75	0.375
0.50	0.50	0.00	0.50	0.00	0.80	0.20	0.10	0.00	0.40	0.60	0.30
0.50	0.50	0.00	0.50	0.00	0.20	0.80	0.40	0.00	0.10	0.90	0.45
0.50	0.50	0.00	0.50	0.50	0.00	0.50	0.75	0.25	0.00	0.75	0.625
0.50	0.50	0.00	0.50	0.20	0.00	0.80	0.60	0.10	0.00	0.90	0.55
0.50	0.50	0.00	0.50	0.80	0.00	0.20	0.90	0.40	0.00	0.60	0.70
0.50	0.50	0.00	0.50	0.50	0.50	0.00	0.50	0.25	0.25	0.50	0.50
0.50	0.50	0.00	0.50	0.20	0.80	0.00	0.20	0.10	0.40	0.50	0.35
0.50	0.50	0.00	0.50	0.80	0.20	0.00	0.80	0.40	0.10	0.50	0.65

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.20	0.00	0.80	0.60	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.20	0.00	0.80	0.60	0.00	1.00	0.00	0.00	0.00	0.20	0.80	0.40
0.20	0.00	0.80	0.60	1.00	0.00	0.00	1.00	0.20	0.00	0.80	0.60
0.20	0.00	0.80	0.60	0.00	0.50	0.50	0.25	0.00	0.10	0.90	0.45
0.20	0.00	0.80	0.60	0.00	0.80	0.20	0.10	0.00	0.16	0.84	0.42
0.20	0.00	0.80	0.60	0.00	0.20	0.80	0.40	0.00	0.04	0.96	0.48
0.20	0.00	0.80	0.60	0.50	0.00	0.50	0.75	0.10	0.00	0.90	0.55
0.20	0.00	0.80	0.60	0.20	0.00	0.80	0.50	0.04	0.00	0.90	0.52
0.20	0.00	0.80	0.60	0.80	0.00	0.20	0.90	0.16	0.00	0.84	0.58
0.20	0.00	0.80	0.60	0.50	0.50	0.00	0.50	0.10	0.10	0.80	0.50
0.20	0.00	0.80	0.60	0.20	0.80	0.00	0.20	0.04	0.16	0.80	0.44
0.20	0.00	0.80	0.60	0.80	0.20	0.00	0.80	0.16	0.04	0.80	0.56

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.00	0.50	0.50	0.25	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.00	1.00	0.00	0.00	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.00	0.50	0.50	0.25	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.00	0.80	0.20	0.10	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.00	0.20	0.80	0.40	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.50	0.00	0.50	0.75	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.20	0.00	0.80	0.50	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.80	0.00	0.20	0.90	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.50	0.50	0.00	0.50	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.20	0.80	0.00	0.20	0.00	0.00	1.00	0.50
0.00	0.50	0.50	0.25	0.80	0.20	0.00	0.80	0.00	0.00	1.00	0.50

b1	d1	u1	pe1	b2	d2	u2	pe2	b3	d3	u3	pe3
0.80	0.00	0.20	0.90	0.00	0.00	1.00	0.50	0.00	0.00	1.00	0.50
0.80	0.00	0.20	0.90	0.00	1.00	0.00	0.00	0.00	0.80	0.20	0.10
0.80	0.00	0.20	0.90	1.00	0.00	0.00	1.00	0.80	0.00	0.20	0.90
0.80	0.00	0.20	0.90	0.00	0.50	0.50	0.25	0.00	0.40	0.60	0.30
0.80	0.00	0.20	0.90	0.00	0.80	0.20	0.10	0.00	0.64	0.36	0.18
0.80	0.00	0.20	0.90	0.00	0.20	0.80	0.40	0.00	0.16	0.84	0.42
0.80	0.00	0.20	0.90	0.50	0.00	0.50	0.75	0.40	0.00	0.60	0.70
0.80	0.00	0.20	0.90	0.20	0.00	0.80	0.60	0.16	0.00	0.84	0.58
0.80	0.00	0.20	0.90	0.80	0.00	0.20	0.90	0.64	0.00	0.36	0.82
0.80	0.00	0.20	0.90	0.50	0.50	0.00	0.50	0.40	0.40	0.20	0.50
0.80	0.00	0.20	0.90	0.20	0.80	0.00	0.20	0.16	0.64	0.20	0.26
0.80	0.00	0.20	0.90	0.80	0.00	0.20	0.90	0.64	0.00	0.36	0.82

Multi-party model:

In the following tables, the columns represent different opinions using belief, disbelief and uncertainty. The capital letters represent the independent sites. x represent user. The highlight fields are computed opinion of a local site towards the user. The highlighted fields represent the computed final opinion of each site.

	b	d	u	pe
Opinion B-A	0.800	0.100	0.100	0.850
Opinion C-A	0.100	0.800	0.100	0.150
Opinion D-A	0.200	0.100	0.100	0.250
Opinion A-x	0.200	0.700	0.100	0.250
Opinion B-x	0.160	0.020	0.820	0.570
Opinion C-x-alone	0.020	0.160	0.820	0.430
Opinion C-x	0.153	0.153	0.695	0.500
Opinion D-x-alone	0.040	0.020	0.820	0.450
Opinion D-x	0.162	0.147	0.603	0.463

	b	d	u	pe
Opinion B-A	0.800	0.100	0.100	0.850
Opinion C-A	0.600	0.200	0.200	0.700
Opinion D-A	0.900	0.050	0.050	0.925
Opinion A-x	0.800	0.100	0.100	0.850
Opinion B-x	0.640	0.080	0.280	0.780
Opinion C-x-alone	0.480	0.160	0.360	0.660
Opinion C-x	0.677	0.136	0.187	0.770
Opinion D-x-alone	0.720	0.040	0.240	0.840
Opinion D-x	0.777	0.105	0.117	0.836

	b	d	u	pe
Opinion B-A	0.800	0.100	0.100	0.850
Opinion C-A	0.700	0.100	0.200	0.800
Opinion D-A	0.500	0.200	0.300	0.650
Opinion A-x	0.900	0.050	0.050	0.925
Opinion B-x	0.720	0.090	0.190	0.815
Opinion C-x-alone	0.630	0.090	0.280	0.770
Opinion C-x	0.771	0.101	0.128	0.835
Opinion D-x-alone	0.450	0.180	0.370	0.635
Opinion D-x	0.761	0.134	0.105	0.813

	b	d	u	pe
Opinion B-A	0.800	0.100	0.100	0.850
Opinion C-A	0.700	0.100	0.200	0.800
Opinion D-A	0.500	0.100	0.400	0.700
Opinion A-x	0.200	0.700	0.100	0.250
Opinion B-x	0.160	0.020	0.820	0.570
Opinion C-x-alone	0.140	0.020	0.840	0.560
Opinion C-x	0.257	0.034	0.709	0.611
Opinion D-x-alone	0.100	0.020	0.880	0.540
Opinion D-x	0.307	0.046	0.647	0.631

	b	d	u	pe
Opinion B-A	0.300	0.500	0.200	0.400
Opinion C-A	0.400	0.200	0.400	0.600
Opinion D-A	0.500	0.100	0.400	0.700
Opinion A-x	0.200	0.600	0.200	0.300
Opinion B-x	0.060	0.100	0.840	0.480
Opinion C-x-alone	0.080	0.040	0.880	0.520
Opinion C-x	0.122	0.124	0.754	0.499
Opinion D-x-alone	0.100	0.020	0.880	0.540
Opinion D-x	0.189	0.128	0.683	0.530

	b	d	u	pe
Opinion B-A	0.000	0.000	1.000	0.500
Opinion C-A	0.000	0.000	1.000	0.500
Opinion D-A	0.000	0.000	1.000	0.500
Opinion A-x	0.900	0.050	0.050	0.925
Opinion B-x	0.000	0.000	1.000	0.500
Opinion C-x-alone	0.000	0.000	1.000	0.500
Opinion C-x	0.000	0.000	1.000	0.500
Opinion D-x-alone	0.000	0.000	1.000	0.500
Opinion D-x	0.000	0.000	1.000	0.500

	b	d	u	pe
Opinion B-A	0.800	0.100	0.100	0.850
Opinion C-A	0.000	0.000	1.000	0.500
Opinion D-A	0.500	0.000	0.500	0.750
Opinion A-x	0.900	0.050	0.050	0.925
Opinion B-x	0.720	0.090	0.190	0.815
Opinion C-x-alone	0.000	0.000	1.000	0.500
Opinion C-x	0.720	0.090	0.190	0.815
Opinion D-x-alone	0.450	0.000	0.550	0.725
Opinion D-x	0.758	0.078	0.164	0.840

VITA AUCTORIS

Haiyan Cheng was born in P. R. China, 1971. She obtained the Bachelor's degree in Computational Mathematics and Applied Software from Inner Mongolia University, China, in 1992. She obtained the Master of Science in Applied Mathematics from Michigan Technological University, USA, in 2000. She is an active academic researcher. She presented and published two papers in IMSE2000 (Integral Methods in Science and Engineering), Banff, 2000. As a student scholarship recipient, she attended the GGF-4 (Global Grid Forum) in Toronto, 2001. She was accepted into the volunteer program for SIGGRAPH conference, San Diego, 2003.

She plans to pursue a PhD degree in computer security research area. Her ultimate career goal is to become a researcher or professor in the computer science area.