Trust in Large-Scale Computational Grids: An SPKI/SDSI Extension for Representing Opinion *

José de R. Braga P. Jr
Alexandre C. T. Vidal
Fabio Kon
Marcelo Finger
Dept. of Computer Science
University of São Paulo, Brazil
{jrbraga, vidal, kon,
mfinger}@ime.usp.br

ABSTRACT

SPKI/SDSI is a flexible and extensible decentralized security model that provides authentication, confidentiality, and access control. However, SPKI/SDSI certification chains are not suitable for large scale, highly dynamic environments such as computational grids. This work extends the SPKI/SDSI model by including an opinion model based on subjective logic. A simulation is performed to evaluate the effectiveness of the proposal.

Categories and Subject Descriptors

C.2.4 [**Computer-communication Networks**]: Distributed Systems; D.4.6 [**Operating Systems**]: Security and Protection

1. INTRODUCTION

Securing information access is a hard task. Security technologies were created to make it difficult for non authorized users to access information. On the other hand, people and software agents that have legitimate rights could use their prerogatives to execute forbidden actions. Consider, for example, a secure system that is responsible for information protection in a company. In this case, if an employee, possibly new in the company, is able to access a corporate software, then he can misuse this system. However, the employee could need privileged rights to execute his functions correctly.

The problem behind the facts presented is that part of the security systems is not prepared to consider the relationship between subjects. In these systems, a subject receives resource rights in boolean form (true or false) and their interaction history is not considered. We must be aware that users, even if legitimate, can at anytime execute forbidden actions and thus be unreliable.

In centralized network environments, this problem is solved manually by administrators. They are responsible for deciding who is, or is not, reliable. Generally, in these cases, there is one or more databases responsible for user identification and resource access control. The resource owners trust administrators fully and delegate to them resource rights. They are responsible for knowing all subjects and they must decide about the reliability of users.

In environments, like computational grids [6, 4], these tasks are more difficult to manage. In these cases, it is very common, and likely, that the grids will be formed by different administrative domains. For this reason, if using the same solution presented before, the administrators are responsible for deciding about the use of resources. This strategy facilitates administration, but it makes the management in large environments more difficult, and even impractical since the resources can be usually added or removed at any time. The resource owners cannot give opinions about the use of their resources, they must accept the administators' domain policies.

Opportunistic grids are a particular kind of computational grid [15]. In these grids, users donate their idle resources to the grid and the resources can be used by applications in the grid according to their availability. To be useful Opportunistic grids must be scalable and need additional attention in relation to security, because the administration costs of including and removing on demand would be very high.

The solution we investigate for decentralized resource access control is trust chains . Trust chains are based on mutual trust relations between subjects. Through these relations, the subjects can transmit their resource access rights directly or indirectly. In the first case, the resource owner gives his resource to a trusty subject. This same subject can then delegate again the resource, building a trust chain.

SPKI/SDSI (*Simple Public Key Infrastructure/Simple Distributed Security Infrastructure*) [5] is an option for implementation of trust chains. In SPKI/SDSI, each subject manages his name space locally. The subject is represented by a public key and he decides about resource use following controlled policies. SPKI/SDSI, however, represents trust in a boolean form. When the subject decides that a user

*This work is supported by a grant from CNPq, Brazil, process #55.0094/2005-9.

is reliable, he has to have total certainty about this. This implies that chains are just valid or invalid. This binary trust model is not adequate in a computational grid environment. Grids are formed by many institutions which may not have total certainty about each others nature. For this reason, the SPKI/SDSI model alone is not practical in this environment.

To solve this problem we present in this paper an extension of the SPKI/SDSI to allow intermediate levels of trust. This solution is based on a trust model that uses subjective logic concepts to represent trust relations between subjects allowing a fine-grained resource access control. The presented work has scalability and is well suited to the opportunistic grid environment.

This work is organized as follows. Section 2 presents an overview of the SPKI/SDSI model. Section 3 presents the related work. In Section 4, we describe an existing model and propose its use for stating opinions in grid environments. Section 5 presents a simulation and, based on the results, an evaluation of the proposed solution. Finally, Section 6 presents the conclusions obtained from the work and describes our next steps.

2. THE SPKI/SDSI MODEL

SDSI [16] was designed by Ronald Rivets and Butler Lampson. Its development was motivated by the complexity of conventional public key infrastructures, specially their dependence on global name space. SDSI is a public key infrastructure with local name space, which makes it a decentralized security system. SPKI was developed by Carl Ellison and others [5] and it is a simple authorization and authentication system. The union of both projects resulted in SPKI/SDSI, an authorization and authentication system that combines SDSI local name spaces with SPKI authorization system.

SPKI/SDSI is a fully distributed solution. Each user is a certification authority and is responsible for managing certificates himself. In SPKI/SDSI, the subjects are identified by a public key and it associates a public key with a local user name space. This association is valid locally, i.e., the associated name is not globally unique. SPKI/SDSI allows the definition of groups of subjects.

In SPKI/SDSI, there are two kinds of certificate: Name Certificate and Authorization Certificate. The Name Certificate certifies that a name in issuer name space is valid. In its turn, the Authorization Certificate grants resource access rights to subjects in a certificate.

Name Certificate is composed of four fields: *issuer*, *identifier*, *subject*, and *validity specification* [3]. The *issuer* is the public key that signs the certificate. The *identifier* is a byte string that represents a name. The *subject* can be a name or a public key. If the *subject* is a name, it is in local name space and the related public key can be recovered. Finally, *validity specification* is a validity condition of the certificate, it could be a validity date or an access control list (ACL).

Authorization Certificate is composed of five fields: *issuer*, *subject*, *delegation*, *tag*, and *validity specification*. *Issuer* and *subject* have the same function described for Name Certificate as seen before. However, the subject can be a group of users. The field *delegation* indicates that the certificate could be delegated to other subjects. *Tag* specifies what authorization was received. As in the case of Name Certificate, *validity specification* is a validity condition of the certificate.

Figure 1 shows a simple example of SPKI/SDSI delegation. The file system resource owner generates two authorization certificates. A certificate is given to D_1 with the authorization: read, write and non delegation (RW:ND). Another certificate is given to D_2 with authorization: read and delegation (R:D). In the same figure, D_2 generates a new certificate to D_3 with reading rights but delegation is not permitted (R:ND). When D_3 needs to access the file system, he presents the delegation chain ($D_2\ R:D \longrightarrow D_3\ R:ND$) to the system.

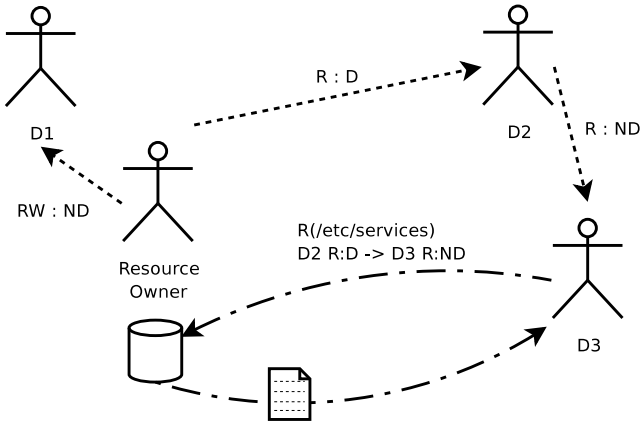

Figure 1: Certificate Delegation

3. RELATED WORK

SPKI/SDSI has an advantage when used in grid systems: it has no single point of failure. However, the management of distributed secure information is difficult; SPKI/SDSI needs chain validation for each access to the resource. In spite of this characteristic, it does not define a certificate repository. If the chain is not resolved, the resource cannot be accessed. Santin *et al*, [17] proposed a federation-based SPKI/SDSI extension. In the proposed model, members of federations can share name and authorization certificates. However, the federation proposed by Santin is not formed by mutual trustworthy relationship of subjects, it is defined either geographically or administratively. We propose a system that consider opinions of subjects about each others for deciding the resource use and forming virtual clusters of mutually trusted subjects.

Resolution and reduction of certificate chains, and name space formalism in SPKI/SDSI have received considerable attention from the network security community [1, 10, 11, 14, 3]. However, these works does not consider that the length of authorization chains could be a problem for SPKI/SDSI-based security systems. The question presented here does not concern only resolving the chain , but limiting the number of delegations. For instance, Bob delegated resource access permission to Clarie. In SPKI/SDSI, Bob cannot limit the redelegations that Clarie might do during the validity of the certificate. In our work, we propose a solution where the chain length problem is minimized by the trust between the participants. A chain can be long, but be no trust.

Trust relations in SPKI/SDSI are based on certificate chain signatures. In this model, the level of trust between subjects is boolean, a chain is considered valid or not. On the other

hand, human relations are more subtle, which means that people can be classified with fine granularity. In real life, we can say that a person is more trustworthy than others, according to our certainty. In the next section, we present a proposal to represent opinions in SPKI/SDSI.

4. OPINIONS IN GRID ENVIRONMENTS

In Grid Computing, there are dynamic trust relations, an new subject can join the grid at anytime. The SPKI/SDSI model is not adjusted to this situation. For example, consider the trust chain shown in Figure 2. This figure represents the relations between subjects named A to D. If subject C is untrusted, because he is a new subject or he has not acted correctly since he joined the grid, then all then chain could be untrusted.

Figure 2: Certificate chain. Subject C is non-trusted.

The subjective logic is defined as a logic that operates on our beliefs about the world [12]. In this work, we propose that subjective logic concepts should be used for representing trust relations between SPKI/SDSI subjects in grid computing. With this in mind, subjects, represented in SPKI/SDSI as public keys, could have non binary opinions about other subjects. The resource owner decides, using security policies, who can access his resources based on a formal trust model. Next, we present a model used to represent opinions.

4.1 Jøsang's Model

We use Jøsang's Model [12, 13] to represent opinion in SPKI/SDSI. In this model, an opinion expresses the user's belief in the truth of a statement. For example, this model can be used for representing the expression: "user's key is authentic". The opinion ω is mathematically represented as: $\omega = \{b, d, u\}$ where $b + d + u = 1$, $\{b, d, u\} \in [0, 1]^3$ and b, d, and u represent belief, disbelief, and uncertainty, respectively.

Jøsang defines various operators for opinions [13]. Some are equivalent to traditional logic operators such as *AND*, *OR* and *NOT*, while others are non-traditional like *CONJUNCTION*, *RECOMMENDATION*, and *CONSENSUS*. The *CONJUNCTION* is used when a subject needs opinions about two independent binary statements. Likewise, in the *RECOMMENDATION* operator, subject B recommends his opinion about a statement to subject A. The resulting opinion can be interpreted as an opinion about the statement as a result of the recommendation from B and not as B's opinion. Finally, the opinion of subjects A and B about a same statement is combined by the *CONSENSUS* operator.

Jøsang's subject logic concepts can be used for minimizing the problem seen in Figure 2. Opinions between consecutive elements of a trust chain could help subjects to decide about using idle resource. The operations defined by Jøsang presented above can be used for composing opinions about trust chains.

Definition 1: Opinion between subjects.

Let A be a subject and b be the statement "subject B is trusted", then

$$\omega_b^A = \{b_b^A, d_b^A, u_b^A\} \quad (1)$$

is A's opinion about whether B is trusted.

Figure 3 shows a trust chain that uses the concepts presented here. In this example, subject A wants to verify a trust chain, from A to D. He has well formed opinions about each subject, except for C. A uses his opinions about E's recommendation for composing an opinion about C. The final opinion is formed by consensus of all opinions of A, including E's recommendation.

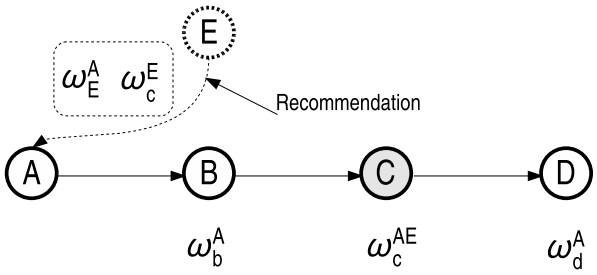

Figure 3: Subject E recommends C to A

Mechanisms for generating opinions are necessary to decide whether a subject is trusted or not. Figure 4 shows two kinds of data input that can be used for making a judgment. The relationship history between pairs of subjects could be used for generating positive opinions about other subjects, in the same way that relevant security logs could be used for generating negative opinions. A user could manually intervene in the opinion generator system to change opinions in according to his own beliefs. User's usage pattern could indicate improper behavior that could result in the reduction of user's confidence. Finally, other informations could help compose opinions about subjects in a trust chain.

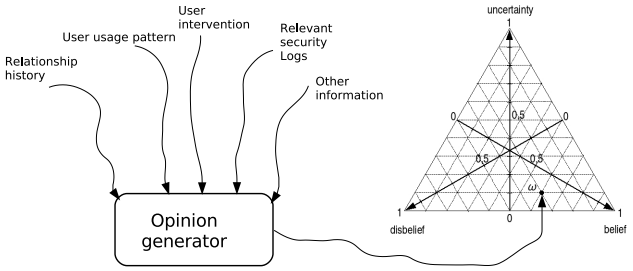

Figure 4: Opinion generator

Table 1 shows a simple credit system used for incrementing opinions. In this credit system, belief, disbelief, and uncertainty credits occur according to weight w defined by the user for each operation. As seen in Table 1, belief credit operations correspond to a debit in the disbelief values. In its turn, a credit in uncertainty correspond to both belief and disbelief debit.

Table 1: Credit system for increment opinions

Credit operation	b	d	u
Belief	$+w$	$-w$	0
Unbelief	$-w$	$+w$	0
Uncertainty	$-\frac{w}{2}$	$-\frac{w}{2}$	$+w$

4.2 SPKI/SDSI Extension

In this work we propose an extension of the SPKI/SDSI model that defines a new certificate: opinion certificate. An Opinion Certificate contains four fields: *issuer*, *subject*, *opinion* and *validity specification*. *Issuer* is the subject that gave the opinion, he is the certificate owner. The *subject* is the entity that received the opinion. The *Opinion* field is the opinion represented using the model presented above. The opinion is formed by three sub-fields: belief, disbelief and uncertainty in accordance to (1). This opinion is made about the statement "*issuer* believes in *subject*". Finally, *validity specification* represents the validity of the certificate.

The proposed extension has clear advantages over the original model. First, the opinion composition allows chains to be verified considering non-binary opinions of all related subjects. Second, a long chain results in an unfavorable opinion if we use the Jøsang model to compose it. Next, if the federation concepts are used in a grid, then the strong belief between its participants is strengthened. Finally, non-binary opinions may be used to have an access control list with fine granularity. For example, a subject about whom we have good opinion could be allowed to read and write in a file system. However, another that is untrusted could only be allowed to read the same file system.

5. SIMULATION

This section presents a simulation of the extension to the SDSI/SPKI trust model we propose. The simulation was based on the InteGrade architecture[1] [8, 7]. An InteGrade grid is formed by computer clusters organized hierarchically. In this grid system, two modules cooperate to manage cluster resources: the LRM (*Local Resource Manager*) and the GRM (*Global Resource Manager*). The LRM is responsible for collecting information, controlling local resource use and running applications in a grid node. The GRM schedules processes and allows communication with other clusters. The LUPA (*Local Usage Pattern Analyzer*) gathers information about resource usage pattern in a single machine and tries to make predictions about the future utilization of resources.

The simulation was performed using the Java language and the Bamboo[2] simulator. The simulated grid environment was composed of subjects that represent LRMS in an InteGrade grid. To simplify the model, each subject controls only one resource. In the simulated model, the GRM[3] provides a chain resolution service, therefore it searches for the opinion about unknown subjects in a trust chain. Finally, LUPA gathers information about user usage patterns and this information is used to decide whether a subject acts improperly or not.

The experiment was simulated as follows. Initially, each subject has an uncertain opinion about others, in Jøsang's model this is represented as $\omega(0, 0, 1)$. Then, some subjects delegate their resources randomly. Later, they try to access the delegated resource and, according to their actions, the opinions are formed. Table 2 shows how credit operations were used for updating opinions about subjects.

Table 2: Credit operation used in simulation

Executed actions	Credit operations	w
Legitimate Access	Belief	0.1
Legitimate Access but different usage pattern	Disbelief	0.1
Illegitimate access	Disbelief	0.1

To access a resource, a subject presents a trust chain. Three levels of opinions are considered to allow access to resources. In the first case, if the **conjunction** of opinions in the chain reaches high belief levels ($\omega_r(b, d, u)$ where $b \geq 0.6, d \leq 0.2, u \leq 0.2$), then the resources can be accessed without restrictions. In an intermediate level, where the value is in a intermediary range ($\omega_b(b, d, u)$, where $b > 0.2, d \leq 0.2, u < 0.6$)), the system limits the access. Finally, the access is denied if the resulting opinion reaches other values. In all cases, if the resource owner does not have an opinion about a subject in the presented chain, he must search the grid and compose an opinion with the **recommendation** operator.

The simulations were divided in two parts. In both parts there are hostile and normal subjects. Hostile subjects execute improper actions and receive unsatisfactory opinions. Their presence in a trust chain can result in denied access. On the other hand, normal subjects only execute correct actions and receive positive opinion. In the first part of the simulation, trust chains forms a random graph. In the other part, they form a scale-free network topology.

The scale-free network is a specific kind of graph in which some vertexes (named hubs) have a high degree of connectivity, although the others have low degrees. In 1999, Albert-László Barabási and Réka Albert mapped the connectivity of Web pages and discovered that their vertexes connectivity follows a scale-free power-law distribution [2]. These features occur in other networks such as power-grids, social networks, and article references in scientific literature. If we consider the potential growth of the computational grid then we can assume that this model is adequate for representing trust relations in a long-scale grid.

In a scale-free network the probability of the number of edges connected in a vertex follows the distribution $P(k) \backsim K^{-\gamma}$. A scale-free network can be built through addition of nodes to an existing network. Links are introduced to existing nodes with the probability $\prod(k_i) = k_i / \sum_j k_j$ where k_i is the connectivity of the vertex. We use the Albert and Barabási algorithm for generating a scale-free network:

1. Start with a complete graph with a small number of nodes (m_0)

[1] `http://www.integrade.org.br`

[2] `http://bamboo-dht.org`

[3] In the original implementation, the GRM has a trader service. This service was represented in the simulation.

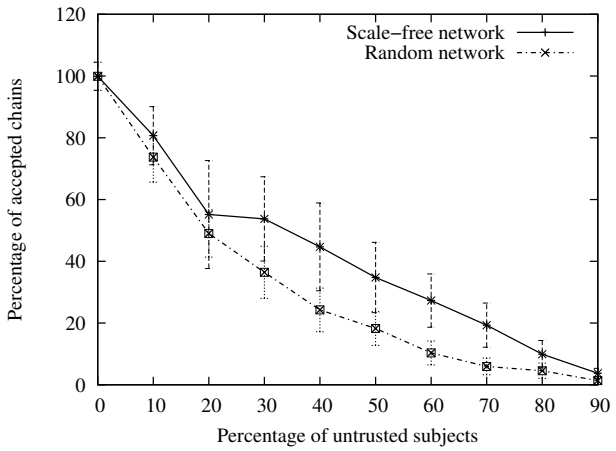

Figure 5: Accepted chains versus percentage of untrusted subjects

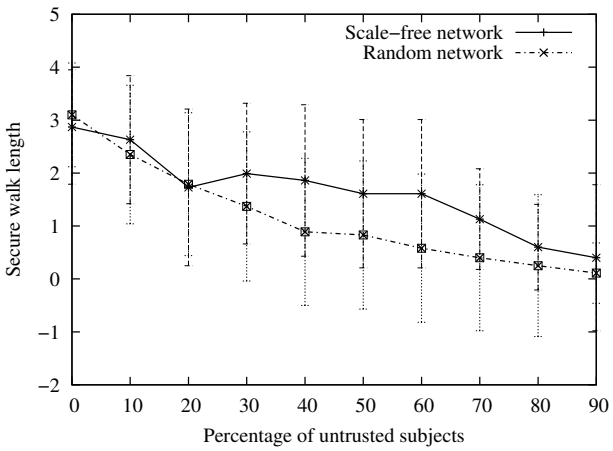

Figure 6: Secure chain length versus percentage of untrusted subjects

2. At each timestep add a new node with degree $m < m_0$. The probability of linking to node i is $\prod(k_i) = k_i/\sum_j k_j$.

The simulation was repeated 30 times. As each simulation is independent and takes almost an hour, we performed then in parallel on an InteGrade grid. We used as network parameters the king data set [9] that is widely used in simulations and represents a real network environment. This data set is formed by a matrix that represents the latencies between nodes in a network, which was extracted from real measurements of 2,048 DNS servers. Our simulation randomly sampled network latencies from the King data set.

We measured the relation between the number of accepted and rejected chains, and the percentage of hostile elements in the grid. The objective of this measure was to verify that the proposed extension of SPKI/SDSI has the desired behavior in a grid environment. Moreover, we analyze the behavior of the proposed extension both in the random and scale-free network topologies.

Figure 5 shows the curve that represents the percentage of accepted chains in relation to percentage of untrusted subjects in the grid. As shown in the figure, as the expected there is a decrease in the accepted chains as the percentage of untrusted subjects increases. However, the scale-free network decreases with a lower rate. We verify that the presence of trusted hubs in the scale-free network topology improves the opinions of all delegated chains that contain hubs.

Figure 6 shows the curve that represents the valid length of the chains. An valid length of a chain is defined as the length of the largest chain fragment that has valid opinion. For example, the chain $A \rightarrow B \rightarrow C \rightarrow D$ has length equal 2 if the conjunction $\omega^A_{b\wedge c}(b_{b\wedge c}, d_{b\wedge c}, u_{b\wedge c})$ has $b_{b\wedge c} = 0.6, d_{b\wedge c} = 0.2, u_{b\wedge c} = 0.2$ and $\omega^A_d(b_d, d_d, u_d)$ has $b_d < 0.6$. As shown in the same figure, the length decrease along with an increase of untrusted subjects. The short valid chain length made the grid nodes form virtual cluster of mutually trusted subjects.

Another analyzed information was the secure walk on the delegations chain. We consider a trust chain like a graph whose nodes are subjects, directed edges are delegations (trust statements), and the opinion, as presented in this paper, are the edge's weight. A secure walk can be defined as a graph walk with the conjunction of its edge weight is higher than a defined opinion.

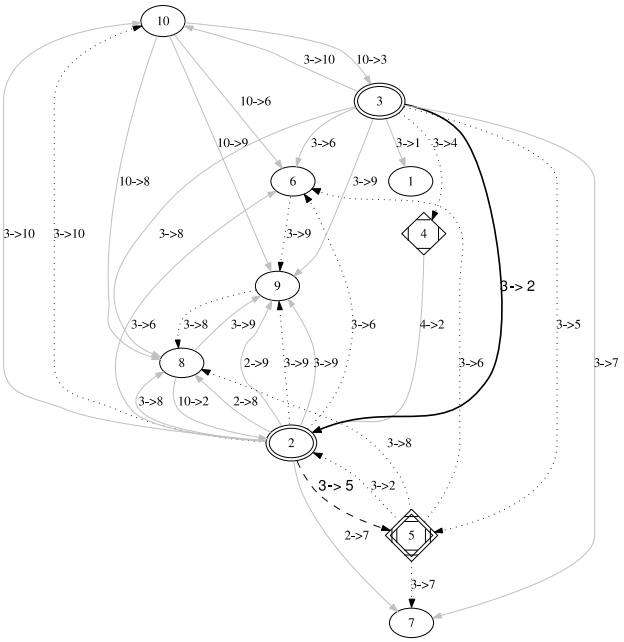

Figure 7: Secure walk in network

Figure 7 shows secure walks in trust network of a grid. This simulation has only ten nodes and represents a grid in a scale-free network topology. The edge label symbolizes the delegations between subjects. For example, $10 \rightarrow 3$ means that the delegated resource 10 was passed to subject 3. An edge printed with a dotted line indicates an insecure walk. In other words, the resulting opinion has reached low values. For instance, the highlighted trust chain in the figure ($3 \rightarrow 2 \rightarrow 5$) indicates that the resource delegated for subject 3 can be used by subject 2, but the conjunction of opinions

that includes subject 5 denies access for him.

Untrusted subjects in Figure 7 were represented as diamond shapes. We can note that these subjects were isolated in the grid; there is no valid delegation arriving in these nodes. This occurs because their reputations were propagated in the network. In grid environments this situation is interesting because it connects subject of various administrative domains. If there is a malicious subject in the grid, then the system reacts and isolates it.

6. CONCLUSIONS

The SPKI/SDSI extension proposed in this paper introduces a new concept to the initial SPKI/SDSI model: the subjectivity. Subjectivity allows SPKI/SDSI to be used in highly dynamic environments such as computational grids. In these environments, it becomes possible to consider the relationship history between subjects for deciding about the use of their resources.

Jøsang's model was considered adequate for represeting opinions. The formalism used in its definitions was the pillar of our work. However, we need to verify whether the mathematical operations are sufficient. Experiments in real environments would indicate an extension in Jøsang model.

The implementation of a grid security middleware using the SPKI/SDSI extension will be the next step in our work. SPKI/SDSI provides flexibility and decentralization for heterogeneous and geographically dispersed environments such as computational grids. Utilization of trusted chains, especially with the extension presented here, will enable fine granularity in security policy definitions.

7. REFERENCES

[1] M. Abadi. On SDSI's linked local name spaces. *Journal of Computer Security*, 6(1-2):3–21, 1998.

[2] A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, October 1999.

[3] D. Clarke, J.-E. Elien, C. Ellison, M. Fredette, A. Morcos, and R. L. Rivest. Certificate chain discovery in spki/sdsi. *J. Comput. Secur.*, 9(4):285–322, 2001.

[4] R. Y. de Camargo, A. Goldchleger, M. Carneiro, and F. Kon. *Pattern Languages of Program Design 5*, chapter The Grid Architectural Pattern: Leveraging Distributed Processing Capabilities, pages 337–356. Software Pattern Series. Addison-Wesley, 2006.

[5] C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, S. Bell, and T. Ylonen. SPKI Certificate Theory. Internet RFC #2693, 1999.

[6] I. Foster and C. Kesselman. *The Grid 2: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers Inc., 2003.

[7] A. Goldchleger, F. Kon, A. Goldman, M. Finger, and G. C. Bezerra. InteGrade: object-oriented Grid middleware leveraging the idle computing power of desktop machines. *Concurrency and Computation: Practice and Experience*, 16(5):449–459, March 2004.

[8] A. Goldchleger, F. Kon, A. G. vel Lejbman, and M. Finger. InteGrade: Object-Oriented Grid Middleware Leveraging Idle Computing Power of Desktop Machines. In *Proceedings of the ACM/IFIP/USENIX Middleware'2003 1st International Workshop on Middleware for Grid Computing*, pages 232–234, Rio de Janeiro, Brazil, June 2003.

[9] K. P. Gummadi, S. Saroiu, and S. D. Gribble. King: Estimating Latency Between Arbitrary Internet End Hosts. In *Proc. of the Second ACM SIGCOMM Workshop on Internet measurment*, pages 5–18, New York, NY, USA, 2002.

[10] J. Y. Halpern and R. V. der Meyden. A logical reconstruction of SPKI. In *CSFW '01: Proceedings of the 14th IEEE Workshop on Computer Security Foundations*, pages 59–70, Washington, DC, USA, 2001. IEEE Computer Society.

[11] J. Howell and D. Kotz. A formal semantics for SPKI. In *ESORICS '00: Proceedings of the 6th European Symposium on Research in Computer Security*, pages 140–158, London, UK, 2000. Springer-Verlag.

[12] A. Jøsang. Artificial reasoning with subjective logic. In *Second Australian Workshop on Commonsense Reasoning*, Perth, Australia, 1997.

[13] A. Jøsang. An algebra for assessing trust in certification chains. In *Network and Distributed Systems Security Symposium (NDSS 99)*, San Diego, USA, 1999. The Internet Society.

[14] N. Li. Local names in SPKI/SDSI. In *CSFW '00: Proceedings of the 13th IEEE Computer Security Foundations Workshop (CSFW'00)*, pages 2–15, Washington, DC, USA, 2000. IEEE Computer Society.

[15] M. Litzkow, M. Livny, and M. Mutka. Condor - A Hunter of Idle Workstations. In *Proceedings of the 8th International Conference of Distributed Computing Systems*, pages 104–111, Palo Alto, CA, June 1988.

[16] R. L. Rivest and B. Lampson. SDSI – A simple distributed security infrastructure. Presented at CRYPTO'96 Rumpsession, 1996.

[17] A. Santin, J. da Silva Fraga, E. R. de Mello, and F. Siqueira. Extending the SDSI / SPKI Model through Federation Webs. In *Communications and Multimedia Security (CMS2003)*, pages 132–145, Turim, Italy, January 2003.