Grid Security and Trust Management Overview

Slavomír Kavecký[1]

[1] **Department of Informatics, Faculty of Management Science and Informatics, University of Žilina**
Žilina, Slovakia

Abstract

Security is one of the most important aspects in all grid environments. Researchers and engineers developed many technologies and frameworks used to establish an environment, in which users can use grid capabilities in a secure manner. In traditional grid environments security is based on user authentication and authorization of user's actions on shared resources. However, this approach demands a pre-established trust relationship between the grid users and the resource providers. Security based on trust management enables dynamic creation of trust relationships between unknown parties. This article reviews various trust models designed for grid environments and lists their main characteristics and purpose in traditional and emerging grids.

Keywords: *Grid, Security, Trust, Trust management, Trust model.*

1. Introduction

Grid [1] emerged as a technology for high-performance computing, large data processing and data storage. Researchers worldwide benefit from the technology when sharing of resources is less difficult than obtaining expensive hardware and software solutions. Also, when the involved participants are geographically dispersed, sharing of resources may be the only possibility to cooperate on the same experiment or be part of one research group.

Grid as a technology is not fully standardized, however a set of recommendations for grid development was designed and is known as OGSA (Open Grid Services Architecture) [2]. OGSA introduces the following grid capabilities: user tasks execution management, data manipulation management, allocation and management of shared resources, secure job execution and resource sharing, information provision of executed tasks and shared resources, and finally support for grid configuration.

Grid is a technology intended to support development of service and business inter-domain oriented applications. If two or more grid users are to cooperate through applications build on grid middleware there must exist means to protect their shared resources and data from malicious users and resource providers. According to OGSA must grid security provide user authentication, since only authenticated user can try to access shared resources. Before a user is allowed to access the resource, he must first be authorized to execute his actions. The communication between the user and the resource provider also must be secured to protect the transmitted information.

Security was not always part of the grid middleware. In the early 90s researchers and developers added the ability to share resources to supercomputers and subsequently they added the ability to share data. Supercomputers capable of resource and data sharing were referred to as first-generation grids. In the late 90s a framework enabling the usage and combination of different grid middleware systems was outlined and developed. Grids built on that framework are referred to as second-generation grids. Finally in the early years of the new millennium the third-generation grids were born by merging grid infrastructure with Web technologies, which enabled to hide grid complexity through resource and data virtualization. However, the evolution of the grids had not stopped at that moment and we are encountering new emerging grids referred as NGG (Next Generation Grids) [3].

The rest of the paper is organized as follows: Section 2 reviews security techniques and frameworks used in traditional grid environments; Section 3 introduces soft security approach to the protection of grid users and resource providers based on trust management; Review of recent trust models is in section 4; Usage of trust models in traditional and emerging grids is discussed in section 5; And finally, the last section concludes with the information about reviewed models and their usage in grid environments.

2. Hard Security

The purpose of a grid security infrastructure is to protect resource providers from malicious users and to protect user data from unauthorized access.

The fundamental features of every security infrastructure are authentication and authorization (these are referred to as hard security [4]). Authentication is a process of checking the authenticity of an entity (i.e. whether the entity is really the entity it claims to be). Authorization is a process of determining who is allowed

to access which shared resources and under what conditions (i.e. whether the entity is trusted to execute jobs on the shared resources). It is important to understand how these features are implemented, therefore the rest of the section reviews existing authentication and authorization infrastructures.

2.1 Authentication Infrastructures

In the early stages of grid infrastructure development the few grid users had unnamed trust relationships between them and the absence of grid security was not a big issue. However, with growing number of grid users security became more important and the infrastructures for secure grid usage were developed.

Probably the most known authentication infrastructure is the **Public Key Infrastructure** (PKI) [5], which is based on the concept of public key cryptography. The trust in a user identity is established through a trusted third party. The trusted mediator is called Certificate Authority and it is responsible for allocating the user's home domain identity into the grid identity and for issuing certificates with the allocated identity. User with the obtained PKI certificate can authenticate to a resource shared in the grid community. For this infrastructure to work a trust relationship between the certificate authority, grid users and resource providers needs to be pre-established.

Another security infrastructure enabling the authentication of a user identity is **Kerberos** [6]. The trust in the user's identity is mediated with session keys issued by the Authentication Server acting as the trusted third party. The idea of identity mediating is basically the same as in PKI. However, in Kerberos special tokens are issued instead of the certificates in PKI. This infrastructure is also based on pre-established trust relationships and the role of the trusted mediator is played by the authentication server.

Athens [7] is another authentication infrastructure developed to control access to a wide range of shared resources. Users have an account for each resource they wish to access and these accounts are managed centrally by the Account Server. An agent enforcing access control is installed in every site which is sharing resources. The user must provide his username and password before he can access the requested resources. This step must be repeated every time the user wants to access one of the available resources.

All of the reviewed authentication infrastructures have common features: they are able to confirm user identity and manage user information centrally. A more detailed overview of authentication infrastructures developed for grid environments can be found in [8, 9].

2.2 Authorization Infrastructures

As the grid infrastructure was growing more popular, access control based only on user identity soon became insufficient and new means for fine-grained access control were needed.

The first infrastructure for access control based not on user identity is called **Grid-Map Files** (GMFs) [8]. The main idea behind GMFs is the usage of access control lists. A list pairing distinguished names of authenticated grid users and local user accounts to which these names are mapped is stored on each shared resource. It is then left to the resource operating system and application access control mechanism to enforce the access to the resource.

A more complex authorization infrastructure is the **Community Authorization Server** (CAS) [10]. CAS defines access control on two levels – resource and Virtual Organization (VO) level. Resource administrators can delegate part of their authorization rights to the CAS administrator. In order to access a resource a user needs to obtain his capabilities assigned to the user by the CAS according to access policies. These policies define what type of access the user can request from CAS. User presents his capabilities to a resource, whereby the resource can apply local access policies and the access is granted or refused.

Virtual Organization Membership Service (VOMS) [11] mediates trust between users and resource providers through a trusted third party – VOMS server. All information about users is managed centrally on VO level by the VO administrator. VOMS server provides users with attributes needed to access a shared resource in the form of attribute certificates. Users present their attribute certificates issued and signed by VOMS server to resources in order to access them. Resources check the validity of the attribute certificate and the attributes contained in it. Subsequently, local resource access policies are applied and the user is granted or refused the access to the resource.

Another example of an authorization infrastructure is **PrivilEdge and Role Management Infrastructure Standard** (PERMIS) [12]. In order to access a resource protected by the PERMIS infrastructure the user needs to present a role based attribute certificate. The attribute certificates are issued by sources of authority and contain the user's role and attributes. PERMIS enables distributed role management, whereby certificates can be stored locally on the sites that allocated them. Before a decision whether to permit or refuse the access to a resource is made, the resource checks the user's certificate, role assigned to the user and whether the certificate was issued by the trusted source of authority.

In **Akenti** [13] authorization infrastructure the users are issued certificates in order to access a shared resource.

Akenti defines a special type of trusted entity called stakeholder. Stakeholders are trusted to issue use-condition certificates, which place conditions on certificates that the user must have to authorize and gain access to the resource. Every stakeholder can define use-condition certificates independently from other stakeholders, so that one resource can be protected by more access control requirements.

The reviewed authorization infrastructures enable access control based on more information than only user identity. Additional information about users is expressed in the form of certificates that contain attributes, roles or other user's data. A more detailed overview of existing authorization infrastructures can be found in [8, 9].

3. Soft security

The security mechanism is responsible for protection against malicious parties. In traditional security mechanisms the protection is provided by securing the resources against malicious users and their activities that could harm the data stored on the resources or the resources themselves. However, in many situations the user has to be protected from the resource providers, so the problem is in fact reversed. While the traditional mechanisms are unable to provide this type of protection, the trust and reputation systems are [4]. The traditional mechanisms like authentication and authorization are referred to as hard security, while the term soft security is used for trust management based systems.

3.1 Definition of Trust

We rely on trust every day and it is easy to understand what the meaning of trust is. However, the term trust is vague in its nature and hard to define generally. Fortunately, the scope of trust can be reduced to a level where it concerns only online environments, such as the Internet or distributed online systems.

In the literature two common trust definitions are used. The **reliability trust** [4, 14] is defined as follows: *Trust is a subjective probability by which an individual, A, expects that another individual, B, performs a given action on which its welfare depends.* The **decision trust** [4, 14] is defined as follows: *Trust is an extent to which one party is willing to depend on something or somebody in a given situation with a feeling of relative security, even though negative consequences are possible.*

The definition of reliability trust does not take context into account and enables the trustor to make decision on whether or not to collaborate with the trustee based only on an estimation of the trustee's reliability. However, the decision trust definition takes not only the context into account, but it also binds the estimation of the trustee's reliability with the risk that arises from uncertain outcome of the collaboration. Hence the usage of decision trust for the purpose of trust modeling seems to be a better choice.

3.2 Trust Management

Trust between two entities is a bidirectional relationship and can be seen from two sides. The success and survival of an entity is dependent on the willingness of other entities to collaborate. Hence the ability to gain trust of other entities is an important criterion, because we tend to collaborate only with trusted entities. Humans have many strategies (whether genetically determined or culturally acquired) for appearing reliable and trustworthy. However, the attempt to give false impression of trustworthiness is not uncommon for humans. Therefore, we can see the importance of the ability to correctly determine the trustworthiness of target entities.

According to the two sides of a trust relationship the **trust management** [14] is defined as follows: *The activity of creating systems and methods that allow relying parties to make assessments and decisions regarding the dependability of potential transactions involving risk, and that also allow the players and system administrators to increase and correctly represent the reliability of themselves and their systems.*

There is a need for methodologies that enable the relying parties to assess trustworthiness of remote parties through computer mediated communication and collaboration, and that enable at the same time the entities to be recognized as trustworthy. This need arises due to the fact that computer networks move us away from direct style of interaction. We can collaborate with people we have never met and that we might never meet in person. The traditional methods for representing and assessing trustworthiness used in physical world can therefore no longer be used. Simply expressed, the application of methodologies that enable such trusted collaboration in online environments can be called trust management.

3.2 Trust Classes

Trust is a directional relationship between two parties – a trustor and a trustee. It is assumed that the trustor is a thinking entity, because he makes decisions whether or not to start collaboration with the trustee based on the trustee's trustworthiness. In online environments like grids there is a need for mutual trust, because both parties (grid user and resource provider) are thinking entities and they must trust each other for the same purpose, otherwise any collaboration is not possible [14].

The mutual trust relationship is described by **trust classes** [4] as follows:

- **Provision trust** describes the user's trust in a service or resource provider. User trusts the provider to provide a service that implements the advertised functionality and does not harm his resources. Service provision trust is meant to ensure the reliability of the provider and is related to the integrity of the user's data obtained from and/or stored in the provided resource.
- **Access trust** describes the resource provider's trust in the user accessing the provided resource, i.e. the provider trusts the user to use the resource in an agreed manner. This relates to the access control paradigm which is a central element in computer security.
- **Delegation trust** describes trust in an agent (the delegate) which acts and makes decisions on behalf of the relying party. Delegation trust can be seen as a special case of provision trust, because relying party trusts the delegate not to misuse the delegated rights.
- **Identity trust** describes the belief that an entity identity is as claimed.
- **Context trust** describes the extent to which the trusting party believes that the distributed system contains mechanisms necessary to support the transaction in case that something goes wrong.

Traditional security mechanisms do not implement all of the above mentioned trust classes. Identity trust is implemented through authentication, access trust through authorization and delegation trust through rights delegation among grid sites. Implementations of context trust and particularly provision trust are missing [15]. However, soft security mechanisms enable users and resource providers to dynamically establish mutual trust relationships and to make decisions supported by all trust classes. To what extent the classes are part of the decision depends on the trust model on which the security mechanism is built.

4. Trust Models

In the face of increasing uncertainty and risk, users and program agents must be allowed to effectively reason about the trustworthiness of other entities. Hence, the goal of trust models is to support decision–making in online interactions. The models integrating trust management into grid environments have many similarities and differences. The typical and novel features of recent trust models [16, 17, 18, 19, 20, 21, 22, 23, 24] enable us to propose a trust model classification based on the following categories: trust value modeling approach, trust relationship types, trust value structure, support of initial trust, levels of trust modeling, and purpose of modeled trust.

4.1 Trust Value Modeling Approach

As already mentioned, trust is a term that can be expressed in many different ways. Trust value can be modeled based on its vague nature typical for trust. Trust value can also be modeled as a prediction of possible future collaboration or as an exactly calculated value. Depending on the approach used to calculate the trust value, trust models can be divided into the following three groups:

- Fuzzy logic based models,
- Models based on probability theory,
- Models based on other mathematical methods.

Trust value in fuzzy logic based models [16, 17, 18, 19] is modeled to express to what extent a relying party is willing to depend on another entity. Trust is not an objective property of the trusted entity, instead it is a subjective belief of the relying party about that entity.

Fuzzy logic models use linguistic terms rather than exactly calculated trust values to state how much an entity believes in the collaborating entity. The relying party can describe the trusted entity as "Very trustworthy", "Trustworthy", "Untrustworthy" or "Very untrustworthy". The granularity of the used expressions can in fact vary and it can be either defined directly in the model or the decision about values of the trust variable is left to the grid node access policy.

The modeling approach in fuzzy logic based models is built on fuzzy inference system. Grid node attributes and other relevant properties (e.g. direct trust and recommended trust, which are discussed later) are first transformed from crisp values into membership grades for linguistic terms of fuzzy sets. The membership functions are a subject of the designer's choice. The transformed values are processed by applying fuzzy rules provided by experts or extracted from numerical data. The output fuzzy set is processed through the process of defuzzification, i.e. output fuzzy values are transformed into crisp values.

The output values obtained from fuzzy inference system enable to make decision on whether or not the relying party should start the transaction with another entity, e.g. in [18, 19] the calculated crisp value is called trust index (TI) and represents the trustworthiness of the trusted entity. On the other hand, the relying party demands from the other entity to provide security assurance by issuing a security demand (SD). These two parameters must satisfy the security-assurance condition: $TI \geq SD$. The condition must be satisfied already before the transaction start.

In a grid environment the collaborating parties execute actions on which their welfare depends. However, the outcome of the executed actions is not known in advance. In models based on probability theory [20, 21] the trust is related to some form of prediction of what that outcome will probably be.

In probabilistic models trust is built by experience. The outcome of previous actions determines the outcome of future actions, but the outcome cannot be predicted exactly. The predicted outcome is only an estimation based on previous observations. The probability that the next execution of actions will be a point within a space of possible outcomes is in [21] described by a probability distribution called outcome distribution and the estimated trust value can be applied to several utility models for the purpose of decision making.

In models based not on fuzzy logic or probability theory trust reflects the belief one entity has about another. The trusting entity expects the trusted entity to act in a certain way. This expectation is based on information about the trusted entity's attributes (e.g. technical capabilities, skills), previous experience with that entity and recommendations from other trusted entities. Interesting is also the concept of suspicion level defined in [23], which indicates how likely an entity will act improperly. The suspicion level changes trust established between the entities and imposes requirements on access control.

4.2 Trust Relationship Types

Trust between two entities is formed and updated over time through direct interactions or through information provided by other entities in the community about their experience. Each event that can influence the degree of trust is interpreted by the entity as either a negative or a positive experience. If the event is interpreted as a negative experience, the trust of the entity is lowered and if the event is interpreted as positive, the trust of the entity is increased by some degree [21]. The state of the system itself has influence on the entity's degree of trust as well. Therefore, the direct experience, information about other entities' experience and also the context of an interaction between entities are considered as factors determining the overall degree of trust.

The trust relationship types are divided as follows based on the factors influencing the degree of trust:

- Direct trust,
- Recommended trust,
- Situational / context trust.

The direct trust a trusting entity has in another entity is mainly formed as a result of previous interactions between the two entities. The concept of direct trust is used in every trust model and it is the basic trust relationship that two entities can have. The models however differ in the method how the direct trust is calculated and which attributes of a trusted entity are taken into account. The model described in [18, 19] is the most specific about the attributes considered when calculating direct trust of an entity. The model uses such attributes as prior job success rate, firewall capabilities, anti-virus capabilities and capabilities of intrusion detection system. The model in [24] evaluates direct trust one entity has in another based on the behavior of the evaluated entity. Behavior of an entity is expressed as the willingness to abide requirements that the trusting entity has declared and violation of these requirements leads to a penalty in direct trust.

Recommended trust can be characterized as a reputation the trusted entity has. Reputation of an entity can be seen as everything that is generally said or believed about the entity's character or standing. If the trusting entity is aware of the trusted entity's reputation it can base its trust on that reputation, i.e. the trusted entity is trusted because of its good reputation. On the other hand, if the trusting entity has a private knowledge about the trusted entity (e.g. through direct experience) and the private information overrules any reputation the trusted entity might have, then the trusted entity can for example be trusted despite its bad reputation.

Entities reveal and obtain reputation for the purpose of decision making in several ways. In [20] the model builds reputation relationships among VOs instead of grid entities. The reason for such an approach is the fact, that the number of VOs is much smaller than the number of the entities. The model in [23] monitors behavior of entities and if some action on one grid entity is regarded as insecure, the same behavior is likely to be insecure to other similar entities as well. Therefore, when an entity detects a threat, it distributes warnings among entities in the community.

Situational trust is not fully recognized as a trust relationship between two grid entities, but it is a factor influencing the overall trust the relying entity has in another entity. It can be described with the following example [14]: *Consider a person who distrusts a rope for climbing from the third floor of a house during a fire exercise. Imagine now that the same person is trapped in a real fire in the same house, and that the only escape is to climb from the third floor window with the same old rope. In a real fire, most people would trust the rope.*

In the example the reliability trust is in both situations the same: the rope is old and hence distrusted. However, the decision whether or not to use that rope is influenced by the context of the current situation. In the case of a fire the decision trust is high enough to use the rope to escape from the building.

The definition of decision trust recognizes context of a particular situation as a part of the trust value, however only few models explicitly address situational trust (e.g. model in [21] directly defines situational trust as a part of the whole trust model).

4.3 Trust Value Structure

Trust relationships between two entities are not the only factors influencing the decision whether or not to start collaboration in a particular situation. Even though the information about other entity that is obtained through direct interactions or is provided by experience of other entities represents the principle of trust evaluation, as the definition of decision trust suggests, other factors have to be considered as well. Factors participating in the decision making are:

- Trust,
- Risk,
- Uncertainty.

Trust can be in many cases expressed as a result of previous direct and mediated interactions, but each of these interactions was performed under certain circumstances. Every action in an open environment is coupled with the danger of failure and damage in the case of failure. Therefore, reasoning about the possibility of failure and its severity for the relying party has to be done during the process of decision making.

Dangers are part of any global computing system and these dangers require explicit reasoning about risk. Risk is a combination of the likelihood of an outcome occurring and the cost it incurs. Trust and risk are related in the sense that there is no need for a trusting decision unless there is risk involved. Two alternative views of the relationship between trust and risk exist: risk determining level of trust and trust determining level of risk.

The former can be described as follows: in a particular situation or a particular action with a certain level of risk a principal should be enough trustworthy in order to be allowed to enter the situation or carry out the action, i.e. the level of risk determines the minimal level of required trustworthiness. The latter case is described as follows: in a particular situation or a particular action involving a principal with a certain level of trustworthiness the risk should be low enough in order to allow the principal to enter the situation or carry out the action, i.e. the level or trustworthiness determines the maximal level of acceptable risk [25]. If the costs and benefits of the entered situation or executed action are quantifiable, the second view seems more appropriate for risk evaluating.

Decision making about collaboration may need to be done in the absence of complete information, which requires that trust, risk and also uncertainty are considered. E.g. imagine a situation where two completely unknown entities have to collaborate with each other and they have neither a direct experience nor information about experience of other entities. A similar situation can occur also if some information about the other entity is present, but knowledge about other relevant decision factors is still missing. The lack of information must not necessarily result in a change of trust in the other entity, but it changes the certainty about the decision being made. And if the certainty is changed significantly, then the level of trust is changed as well [25]. An interested reader can find a more detailed discussion about trust, risk and uncertainty in the process of decision making in [25, 26].

There are not many models that consider trust, risk and uncertainty as a part of the decision making process and/or these factors are taken into account only indirectly. In the fuzzy models the uncertainty is modeled implicitly, because trust value is considered to be vague and is not calculated exactly. Risk is modeled indirectly as well. In model [23] the concept of suspicion level is used to declare how likely an entity will behave in an unpredicted manner and how risky it would be to start a transaction.

4.4 Initial Trust

Collaboration with unrecognized or completely unknown entities is another possibility in an open environment. In the case that the relying entity has no direct experience with another entity and other entities in the community also have no experience with that particular entity, then there is the need for an approach to evaluate the trustworthiness of the unknown entity.

The decision about the entity's trustworthiness is made in a situation of complete uncertainty. It could be reasoned that the lack of any information about the entity makes any possible collaboration too risky and it should be decided not to collaborate. However, in this way every new entity trying to join the open community would be automatically rejected from any kind of cooperation with entities already in the community. Some existing models therefore integrated means to evaluate trustworthiness in a situation of complete or partial uncertainty, but these models differ in the way the evaluation is made. Independent from the differences in the initial trust evaluation process, models can be divided as follows:

- Models integrating initial trust,
- Models without initial trust evaluation.

Initial trust in model [21] is called basic trust and is derived from past experiences in all situations through the entity's entire lifetime. However, it is not the amount of trust that one entity has in another; it is the representation of a general trusting disposition of the entity. This enables entities that are part of the community for some time to establish trust relationships with previously not encountered entities that are also part of the community.

Model in [23] uses trust negotiation for calculation of initial trust based on attributes other than identity. Trust negotiation enables to establish trust in highly dynamic environments and can be described as a process of sequential exchange of private information. The exchange is governed by access policies. Each policy assigns one or

several credentials, which are accessible only if the policy is satisfied. If the entity requesting information about other entity does not satisfy access policy that discloses the protected information, the requested entity asks the requestor for its protected information as well. The requestor can disclose the requested information or it asks for additional information if disclosure is not possible with already known information about the negotiated entity. The cycle of requests for protected information continues until all requested information can be disclosed and the negotiation is successful; or one of the two entities cannot disclose protected information and the negotiation fails [27].

Fuzzy model in [18, 19] evaluates trust partially based on past experiences and partially on current entity's attributes. If an entity is new in the community and past experiences cannot be evaluated, trust value is calculated only according those attributes. Trust value in this case can be considered as initial trust value of previously unknown entity.

4.5 Levels of Trust Modeling

Trust value in grid environment is typically calculated as the level of trustworthiness of a grid node. This approach enables to directly express relationships among grid entities. However, thanks to its open nature the grid community can become large in the number of integrated nodes. This is often the case of service grids which group many different domains providing services such as provision of services for scientific computing, data services, communication services, information services, etc.

Some trust models consider the growth in number of grid domains as a factor influencing the performance of trust evaluation, even though scalable trust value calculation is one of the requirements imposed on trust models. Depending on the number of grid domains the trust modeling level is divided to:

- Grid node level,
- Grid domain level.

As previously stated, trust value modeling has to be independent from the number of domains or nodes integrated into a grid community, therefore evaluation of trust value on a grid node level is the basic used approach. However, the method for calculating the trust value can be modified to consider the state of the grid community. The purpose of trust value modeling on domain level is to optimize the process of trust value calculation. E.g. in the model [22] recommendations are managed on domain level and trustworthiness of entities is managed within their domains. The overall trust of an entity is determined as trustworthiness of the entity assessed within its domain and the recommendation trust of its domain. A similar principle is used in model [20], where entity trust consists of the reputation of the VO of which the entity is part of and the direct trust among entities.

4.6 Purpose of Modeled Trust

The purpose of trust management in grid environments is to ensure collaboration of grid entities in a secure manner. Most of the existing trust models interpret secure collaboration as insurance that jobs will be successfully executed; stored data will be accessible by trusted and authorized entities; data will not be altered; jobs will not harm shared resources, etc. However, users and resource providers can require that their QoS requirements are satisfied as well. The consideration of the QoS requirements is closely coupled with job scheduling. The scheduler chooses those resources for job execution that are not only trustworthy, but also satisfy the declared QoS requirements. Naturally, the resource will be willing to start the collaboration only if the job owner is trustworthy and the resource's QoS requirements are satisfied.

Consideration of QoS requirements can be coupled with trust modeling as well, but in a different way. The QoS requirements are considered during job scheduling before the decision about collaboration is made. In trust modeling the QoS requirements are considered after the collaboration has ended. The trust model evaluates QoS delivered with QoS agreed during job scheduling after the collaboration end and reflects the result of the evaluation in the overall entity trustworthiness.

There are not many models that take the QoS requirements into account. The existing models can be divided as follows:

- Models with QoS consideration,
- Models ensuring secure collaboration.

Secure collaboration is a goal of all trust models. Only few models see QoS requirements as a factor influencing trust among entities. Model in [20] explicitly includes QoS requirements into trust evaluation process more or less the same way as already explained in this section (after collaboration end the comparison of delivered and expected QoS is reflected in overall entity trustworthiness).

5. Ad Hoc Grids

Traditional grids are typically based on centralized architecture, where activities like maintenance of resources, monitoring and access control enforcement are performed by a dedicated administrative authority. All participants of the grid community share a non-conflicting objective and collaborations are executed under the control of agreed policies on usage, privileges and application deployment, while these policies are rarely changed during

the lifetime of the collaboration. However, there is a need for the support of sporadic and ad hoc communities and collaborations with dynamically changing members and access policies [28].

The motivation for ad hoc grid development is its ability to handle short-term collaborations and resource sharing in a secure environment. If a group of individuals needs to pool resources and execute computation tasks in a one-time collaboration, then administrative overhead resulting from establishment of traditional grid environment is impractical for such a transient community. In this scenario no individual can be entrusted with administrative privileges, but still all shared resources and provided services must be protected. A few infrastructures implementing these principles already exist and more information about their architecture, structure and features can be found in [28, 29]

Ad hoc grids contain geographically dispersed resources with various management policies. However, unlike traditional grids there is no centralized control. An ad hoc grid can be defined as [28]: *distributed computing architecture offering structure-, technology-, and control-independent grid solutions that support sporadic and ad hoc use modalities*.

Structural independence provides several benefits lacking in traditional grid frameworks. It avoids a single point of failure (in a decentralized architecture failure of one peer does not lead to a failure of the whole system) and it enables the participating peers to establish collaborations on the fly without depending on any external infrastructure for assistance. Technology independence in an ad hoc grid reflects its ability to support diverse grid technologies and protocols. Control independence enables to manage security in the absence of a central controller. Therefore, every entity is responsible for maintaining and securing its resources.

Security infrastructure in traditional grids is implemented differently than in ad hoc grids. The most significant difference is the absence of central administrative authority, which in traditional grids assigns unique grid identity and a set of privileges within the scope of the pre-established trust. The administrative services responsible for membership access and usage control on resources are in ad hoc grids hosted on participating peers [30]. Therefore, the usage of trust management to provide secure collaboration is reasonable in this decentralized architecture. Ad hoc grid entities have no pre-established trust relationships which could guide the decision whether or not to start the collaboration. However, security infrastructure based on trust management enables to establish trust among entities without the need for any support from centralized services. Trust management can also be integrated into traditional grids, but the usage scenario is different. Trust modeled with trust management can be part of QoS that the users and resource providers require.

6. Conclusion

Infrastructure of traditional grids enables the grid community participants to collaborate in a secure environment. However, it may be impractical having to establish such infrastructure for short-time and/or one-time collaborations. Ad hoc grids overcome this drawback by making the infrastructure more dynamic where the participants can join and leave the ad hoc grid on the fly as needed. Security in traditional grids is based on pre-established trust relationships. In ad hoc grids the trust among entities must be built without any external infrastructure. Trust management seems to be a reasonable approach to establish and maintain the dynamic trust relationships among ad hoc grid entities. However, trust models used to define these relationships are still not including many factors and use cases. Classification of trust models proposed in this paper shows that risk and uncertainty are a major part of any trust model. Integration of initial trust value modeling is also important. It is our belief that a model fully based on classified properties is able to make the ad hoc grid infrastructure more secure and therefore more attractive for wider.

References

[1] Foster, I, Kesselman, C, “The Anatomy of the Grid – Enabling Scalable Virtual Organizations”, International Journal of Supercomputer Aplications, 2002.

[2] Foster, I., et al., “The Open Grid Services Architecture, Version 1.5”. Available: http://www.ogf.org/documents/GFD.80.pdf, July 2006.

[3] Kurdi, H., Li, M., Al-Raweshidy, H. “A Classification of Emerging and Traditional Grid Systems”, *IEEE Distributed Systems Online*, 9 (3), March 2008.

[4] Jøsang, A., Ismail, R., Boyd, C., “A Survey of Trust and Reputation Systems for Online Service Provision”, *Decision Support Systems*, 43 (2), 618 – 644, March 2007.

[5] Weise, J, “Public Key Infrastructure Overview“, Available: http://highsecu.free.fr/db/outils_de_securite/cryptographie/pki/publickey.pdf, August 2001.

[6] Neuman, B. C., Ts’o, T, “Kerberos: An Authentication Service for Computer Networks”, *IEEE Communications Magazine*, 32 (9), 33 – 38, September 1994.

[7] OpenAthens. Available: http://www.openathens.net/.

[8] Jie, W., Arshad, J., Sinnott, R., Townend, P., Lei, Z. “A review of grid authentication and authorization technologies and support for federated access control”, *ACM Computing Surveys*, 43 (2), January 2011.

[9] Jie, W., Arshad, J., Ekin, P., “Authentication and Authorization Infrastructure for Grids - Issues, Technologies, Trends and Experiences”, *The Journal of Supercomputing*, 52 (1), 82-96, April 2010.

[10] Pearlman, L., et al., "A Community Authorization Service for Group Collaboration", *Third International Workshop on Policies for Distributed Systems and Networks*, 50-59, June 2002.
[11] Alfieri, R., et al., "Managing Dynamic User Communities in a Grid of Autonomous Resources", *Computing in High Energy and Nuclear Physics*, 24-28, March 2003.
[12] Chadwick, D.W., Otenko, A., Ball, E., "Role-Based Access Control with X.509 Attribute Certificates", *IEEE Internet Computing*, 7 (2), 62-69, March 2003.
[13] Akenti. Available: http://acs.lbl.gov/software/Akenti/.
[14] Jøsang, A., Keser, C., Dimitrakos, T., "Can We Manage Trust?", *Third International Conference on Trust Management*, 93-107, 2005.
[15] Kavecký, S., "Trust Based Grid Security and Security Models", *International Journal on Information Technologies & Security*, 4 (3), 81-92, September 2012.
[16] Liao, H., Wang, Q., Li, G., "A Reliable Fuzzy Theory Based reputation System in Grid", *Journal of Computers*, 5 (5), 782-790, May 2010.
[17] Liao, H., Wang, Q., Li, G., "A Fuzzy Logic-based Trust Model in Grid", *Network Security, Wireless Communications and Trust Computing*, 1, 608-614, April 2009.
[18] Song, S., Hwang, K., Kwok, Y., "Trusted Grid Computing with Security Binding and Trust Integration", *Journal of Grid Computing*, 3 (1-2), 53-73, June 2005.
[19] Song, S., Hwang, K., Macwan, M., "Fuzzy Trust Integration for Security Enforcement in Grid Computing", *Network and Parallel Computing*, 9-21, October 2004.
[20] Ding, Ch., et al., "A Novel Trust Model Based on Bayesian Network for Service-Oriented Grid", *IEEE/ACIS International Conference on Computer and Information Science*, 494-499, June 2009.
[21] Shi, J., v. Bochmann, G., Adams, C., "A Trust Model with Statistical Foundation", *Formal Aspects in Security and Trust*, 145-158, August 2004.
[22] Ying, G., Jiang, Z., "A Layered Trust Model Based on Behavior in Service Grid", *International Conference on Advanced Computer Control*, 5, 511-515, March 2010.
[23] Ryutov, T., et al., "Adaptive Trust Negotiation and Access Control for Grids", *IEEE/ACM International Workshop on Grid Computing*, 55-62, November 2005.
[24] Azzedin, F., Maheswaran, M., "Evolving and Managing Trust in Grid Computing Systems", IEEE CCECE Canadian Conference on Electrical and Computer Engineering, 3, 1424-1429, 2002.
[25] English, C., Terzis, S., Wagealla, W., "Engineering Trust Based Collaborations in a Global Computing Environment", *Trust Management*, 120-134, April 2004.
[26] Jøsang, A., Lo Presti, S., "Analysing the Relationship between Risk and Trust", *Trust Management*, 135-145, April 2004.
[27] Winsborough, W. H., Seamons,. K. E., Jones, V. E., "Automated Trust Negotiation", *In DARPA Information Survivability Conference and Exposition, volume I*, 88-102, 2000.
[28] Amin, K., von Laszewski, G., Mikler, A. R., "Toward an Architecture for Ad Hoc Grids", *In 12th International Conference on Advanced Computing and Communications*, 15-18, 2004.
[29] Tiburcio, P. G. S., Spohn, M. A., "Ad Hoc Grid: An Adaptive and Self-Organizing Peer-to-Peer Computing Grid", IEEE 10th International Conference on Computer and Information Technology (CIT), 2010
[30] Amin, K., von Laszewski, G., Sosonkin, M., Mikler, A. R., Hatage, M., "Ad Hoc Grid Security Infrastructure", The 6th *IEEE/ACM International Workshop on Grid Computing*, 2005.

Slavomír Kavecký PhD. student at the Department of Informatics, Faculty of Science Management and Informatics, University of Žilina. The main subject of his work is to design new trust model enhancing current methods of trust management implemented in grid security infrastructures.