PROBABILISTIC ESTIMATION OF TRUST MODEL AND THREAT RESISTANCE ANALYSIS IN SERVICE-ORIENTED SYSTEMS

Nataliia Kussul, Olga Kussul, Sergii Skakun

Abstract*: Trust and reputation models play an important role in enabling trusted computations over large-scale distributed Grids. Many models have been recently proposed and implemented within trust management systems. Nevertheless, the existing approaches usually assess performance of models in terms of resource management while less attention is paid to the analysis of security threat scenarios for such models. In this paper, we asses the most important and critical security threats for a utility-based reputation model in Grids. The existing model is extended to address these threat scenarios. Also we propose the probabilistic estimation of trust model. With simulations that were run using data collected from the EGEE Grid-Observatory project, we analyze efficiency of the utility-based reputation model against these threats.*

Keywords*: trust; reputation model; Grid computing; utility; security threats*

ACM Classification Keywords*: H.1.1 [Models and Principles] Systems and Information Theory; I.4.8 [Image Processing and Computer Vision] Scene Analysis - Sensor Fusion*

Introduction

Grid represents a distributed environment that integrates heterogeneous computing and storage resources administrated by multiple organizations. One of the main concepts in Grid is a virtual organization (VO) — a set of individuals and/or institutions defined by coordinated resource sharing rules for reaching common goals (Foster et al., 2001). VOs are formed dynamically, exist for some time and then resolve.

Trust and reputation models play an important role in enabling trusted computations over large-scale distributed Grids. Two types of trust management systems (TMSs) can be discriminated (Chakrabarti, 2007): policy-based and reputation-based. In policy-based systems, entities in a VO establish trust relationships based on certain predefined policies. In reputation-based systems, certain mechanisms exist in order evaluate the trust which is the function of reputation. Reputation can be viewed as an assumption about the expected quality or reliability of a resource based on existing information or observations about his behaviour in the past (Abdul-Rahman and Hailes, 2000).

Many trust and reputation models have been recently proposed for distributed systems and for Grids, in particular (Arenas et al., 2008; Azzedin and Maheswaran, 2002; Eymann et al., 2008; Gomez Marmol and Martınez Perez, 2008; Josang et al., 2007; Kamvar et al., 2003; Kerschbaum et al., 2006; Liang and Shi, 2010; Papaioannou and Stamoulis, 2008; Silaghi et al., 2007; Song et al., 2005; Srivatsa and Liu, 2006; von Laszewski et al., 2005; Wu and Sun, 2010). Nevertheless, the existing approaches usually assess performance of models in terms of resource management while very few of them focus on the analysis of security threat scenarios for such models.

Gomez Marmol and Martınez Perez (2009) described security threats scenarios in trust and reputation models for distributed systems and proposed possible solutions to tackle them. The study also shows how some of the most representative models (mostly for P2P systems) deal with those threats. von Laszewski et al. (2005) extended an EigenTrust model (Kamvar et al., 2003) to be used in Grids (GridEigenTrust). The obtained reputation value is integrated into a QoS management system providing a way to re-evaluate resource selection and service level agreement (SLA) mechanisms. Eymann et al. (2008) investigated economical issues in Grids along with information asymmetry. These issues are taken into consideration while proposing a reputation-based framework for enabling Grid markets and allowing grid service broker to deal effectively with hidden information. Srivatsa and Liu (2006) identified vulnerabilities that are crucial to decentralized reputation management and developed a safeguard framework for providing a highly dependable and efficient reputation system, called TrustGuard. The conducted experiments showed that the TrustGuard framework is effective in countering malicious nodes regarding oscillating behaviour, flooding malevolent feedbacks with fake transactions, and dishonest feedbacks.

In this paper, we asses the most important and critical security threats for a utility-based reputation model in Grids that was proposed by Silaghi et al. (2007) and Arenas et al. (2008). We will use security threat scenarios for trust and reputation models presented by Gomez Marmol and Martınez Perez (2009) as a reference in our study. These scenarios include: individual malicious peers, malicious collectives, malicious collectives with camouflage, malicious spies, Sybil attack, man in the middle attack, driving down the reputation of a reliable peer, partially malicious collectives, and malicious pre-trusted peers. The model is further extended to address these threat scenarios. With simulations that were run using data collected from the EGEE Grid-Observatory project (Germain-Renaud et al., 2011), we will analyze efficiency of the utility-based reputation model against these threats.

Utility-based reputation model for VOs in Grids

In this paper we extend the existing utility-based reputation model (Arenas et al., 2008) by incorporating a statistical model of user behaviour (SMUB) that was previously developed for computer networks and distributed systems (Kussul and Skakun, 2004; Shelestov et al. 2008, 2007; Skakun et al., 2005) and several new components to address security threat scenarios. The proposed extensions to the reputation model include:

- assigning initial reputation to a new entity in VO: when organization provides a new resource to be integrated in a VO there are no records from the monitoring system to infer reputation value for this specific resource. One possible way of assigning initial reputation to a new resource is to use a methodology of active experiment. There can be several benchmark tasks in the system to estimate the utility function and to provide initial reputation of the resource.

- alliance between consumer and resource: since reputation of resource is based on measure of satisfaction of a consumer in relation to this resource we should avoid cheating via collusions among a group of entities (Azzedin and Maheswaran, 2002). For this purpose, it is advisable to include into the model a factor that will reflect alliance between the consumer and resource.

- time decay function: reputation of resource is based on measuring average value of utility function over certain period of time (Azzedin and Maheswaran, 2002; Silaghi et al., 2007). But if a VO exists for a considerable period of time (e.g. for years) reputation of resource may vary considerably. That is, it is unlikely to use, for example, two years data to estimate current resource reputation if more recent records are available. So, we propose to

incorporate a time lag function into the model that will provide weights depending on the time of the transaction record between consumer and resource.

- score function: for different types of services offered by resource providers different reputation values will be used (Gomez Marmol and Martınez Perez, 2009). Namely, we will categorize services into categories, and a resource provider will get reputation value according to such a category. In Grid systems, tasks can be categorised by the computational complexity. Successful execution of tasks with a complex workflow and parallel programs, for example, environmental models like numerical weather prediction (Kussul et al., 2009; Hluchy et al., 2010), will provide to a resource provider higher reputation value.

Reputation model for resource providers

For the reputation model we will use the enhancement of the well-known model (Arenas et al., 2008), proposed in (Kussul, Novikov, 2009).

The reputation model is based on the utility function that measures the level of satisfaction of a user in relation to service provider. In order to define utility function an auxiliary function that indicates the SLA accorded between a VO user and a resource provider for a particular resource within a VO is implemented (Arenas et al., 2008):

$$\text{SLA} : \bigcup_l u \times \bigcup_k r_k \times \bigcup_m vo_m \rightarrow \text{R} \qquad (1)$$

where R denotes the set of real numbers.

The SLA value represents quality of resource provider as expected by user (Arenas et al., 2008). In order to define utility function based on SLA value we describe the notion of Event:

$$\text{Event} = \text{T} \times \bigcup_l u_l \times \bigcup_k r_k \times \bigcup_m vo_m \times \{\text{QoS name}\} \times \text{R} \qquad (2)$$

where T is a time domain.

Before defining utility function and reputation we will introduce three functions: the first one will characterise possible alliance between consumer and resource in order to avoid cheating (Azzedin and Maheswaran, 2002), the second one will account for a time when utility was estimated (Azzedin and Maheswaran, 2002; Silaghi et al., 2007), and the third one will provide different scores depending on the type of the provided service (Gomez Marmol and Martınez Perez, 2009). These functions provide extensions to the utility function and reputation originally proposed by Arenas et al. (2008).

Function h(u, r) will take a value between 0 and 1 and will show the level of alliance between user u and resource r. If there is no such an alliance between targets, h(u, r) will have a higher value. For example, one possible way of defining h(u, r) is as follows

$$h(u,r) = \begin{cases} 1, \text{ if } f_{vo}(r) \neq g_{vo}(u) \\ \theta, \text{if } f_{vo}(r) = g_{vo}(u) \end{cases}, \quad (3)$$

where θ is a parameter.

Function z(*t, tc*) will show what past records on user-resources interactions should be taken into consideration to estimate reputation of specific resource. Here t is the time, and tc is a parameter. In a simplest form z(t, tc) could be a stepwise function

$$z(t,t_c)=\begin{cases}1, t \geq t_c\\ 0, t < t_c\end{cases}. \qquad (4)$$

Function s(type(r)) will provide different values for different types of services provided by the resource r (function type(r) maps into category of service).

Now, we can define a utility function

utility : *Event* → R,

$$utility(\{t, u, r, vo, QoS, v\}) = \begin{cases}h(u,r)s(type(r)), & \text{if } SLA \text{ met}\\ penalty(v,SLA)h(u,r)s(type(r)), & \text{otherwise}\end{cases}, \qquad (5)$$

where *SLA* is the agreed *SLA* value between the user and resource provider, *penalty*(*v*, *SLA*) is a penalty function imposed on a resource provider if the agreed SLA is not met.

The form of penalty function depends on the QoS in place. For example, for time metrics which are usually to be minimised a penalty function can be represented by

$$penalty(v, SLA) = \begin{cases}1, & \text{if } v \leq SLA\\ \dfrac{SLA}{v}, & \text{if } v > SLA\end{cases}. \qquad (6)$$

Let us denote a set of traces that are used to estimate the reputation of resource r in a vo up to the current time t with

$$\text{Trace}|(vo, r, t) = \left\{\{t', u', r', vo\_id', QoS', v'\} \in Trace : r = r', vo\_id = vo', t' \leq t\right\}. \qquad (7)$$

Let us denote a set of utility() function values derived from traces Trace|(vo, r, t) with

$$O(vo, r, t) = \{ z(t, tc)\cdot utility(\{t, u, r, vo, QoS, v\}) \mid \{t, u, r, vo, QoS, v\} \in \text{Trace}|(vo, r, t)\}. \qquad (8)$$

A reputation is expectation of utility() function (in terms of probability theory)

$$rep(vo, r, t) = E[\, utility(O(vo, r, t))\,] = \int utility(O_{(vo,r,t)})\, p_{utility}(O_{(vo,r,t)})\, dO_{(vo,r,t)}. \qquad (9)$$

If we do not want to discriminate values from utility() function by time then we might use z(t, tc) = 1.

In order to approximate expectation we can use a sample mean

$$rep(vo, r, t) = \frac{1}{\left|O_{(vo,r,t)}\right|} \sum_{x \in O_{(vo,r,t)}} x, \qquad (10)$$

where | · | denotes the cardinality of the set.

The reputation of an organisation o in VO is the aggregation of the reputation of all resources it provides to VO:

$$rep(vo, t) = \frac{1}{\left|f_{vo}^{-1}(o)\right|} \sum_{r \in f_{vo}^{-1}(o)} rep(vo, r, t). \qquad (11)$$

The reputation of a resource in all VOs can be estimated as follows

$$rep(r, t) = \frac{1}{\left|VO|_r\right|} \sum_{vo \in VO|_r} rep(vo, r, t). \qquad (12)$$

Probabilistic reputation based trust model

Let's describe our model in terms of the theory of probability to enable the theoretical analysis of its properties and limitations, as well as assessing the security of the model against the threat scenarios. If *SLA* is a Service Level Agreement, *v* stands for the actual value of the provided services (obtained after the service has been provided to the user). We will denote as ξ the random value that shows the agreed SLA, also we will denote the meaning of *v* as η. After that we can define the penalty function *penalty*(*v*, *SLA*) and the corresponding random value θ as follows:

$$penalty(v, SLA) = \frac{SLA}{v},\ \theta = \frac{\xi}{\eta}, \qquad (13)$$

We will calculate distribution function of the random value θ through the corresponding functions of the variables ξ and η (provided that $\xi, \eta > 0$):

$$P\{\theta < z\} = P\left\{\frac{\xi}{\eta} < z\right\} = \iint\limits_{A} p_\xi(y)p_\eta(x)dxdy = \iint\limits_{0<y<k_x} p_\xi(y)p_\eta(x)dxdy = \\ = \int\limits_0^\infty \int\limits_0^{kx} p_\xi(y)p_\eta(x)dxdy = \int\limits_0^\infty p_\eta(x)\int\limits_0^{kx} p_\xi(y)dydx, \qquad (14)$$

where $A = \left\{(x,y): \frac{y}{x} < z, x > 0, y > 0\right\}$.

In case if *SLA* value for the specific service is constant, then we can present (14) as follows:

$$P\{\theta < z\} = P\left\{\frac{SLA}{\eta} < z\right\} = P\left\{\eta > \frac{SLA}{z}\right\} = \int\limits_{\frac{SLA}{z}}^{\infty} p_\eta(x)dx \qquad (15)$$

According to (10) the utility function:

$$u = \begin{cases} 1, \text{ if } \theta \geq 1 \\ \theta, \text{ if } \theta < 1 \end{cases}. \qquad (16)$$

In this case, the distribution function of the random value *u* will be defined as follows:

$$P\{u = 1\} = P\{\theta \geq 1\} \\ P\{u < x\} = P\{\theta < x\}, \quad where\ 0 \leq x < 1. \qquad (17)$$

Reputation is the mathematical expectation of utility function:

$$rep = M[u] = \int\limits_0^1 xp_u(x)dx, \qquad (18)$$

where $p_u(x)$ - density function of the random value *u*.

Lets calculate this expression.

$$\int\limits_0^1 xp_u(x)dx = 1 \cdot P\{\theta \geq 1\} + \int\limits_0^1 xp_\theta(x)dx = P\{\theta \geq 1\} + \int\limits_0^1 xp_\theta(x)dx = \\ = P\{\theta \geq 1\} + M[\theta \cdot \mathbf{1}_{\theta<1}],$$

where $\mathbf{1}_{\theta<1} = \begin{cases} 1, \text{ if } \theta < 1 \\ 0, \text{ otherwise} \end{cases}$.

That's why, the reputation of the resource can be estimated as follows:

$$rep = \mathrm{M}[u] = P\{\theta \geq 1\} + \mathrm{M}[\theta \cdot \mathbf{1}_{\theta<1}]. \quad (19)$$

The first summand $P\{\theta \geq 1\}$ shows the probability that the resource will fulfill the SLA, the second summand $\mathrm{M}[\theta \cdot \mathbf{1}_{\theta<1}]$ shows the mean value of the penalty function, if SLA will be violated.

Lets look at the following example. Let the *SLA* be a fixed value (in this case the parametric variable) and the random variable η is distributed according to Pareto distribution, so

$$P\{\eta < x\} = \begin{cases} 1 - \left(\frac{x_m}{x}\right)^{\alpha} & \text{if } x \geq x_m \\ 0 & \text{if } x < x_m \end{cases}. \quad (20)$$

According to (15) and (20):

$$P\{\theta < z\} = P\left\{\eta > \frac{SLA}{z}\right\} = \begin{cases} \left(\frac{z \cdot x_m}{SLA}\right)^{\alpha} & \text{if } z \leq \frac{SLA}{x_m} \\ 1 & \text{if } z > \frac{SLA}{x_m} \end{cases} \text{ and}$$

$$p_{\theta}(z) = \begin{cases} \alpha\left(\frac{x_m}{SLA}\right)^{\alpha} z^{\alpha-1} & \text{if } z \leq \frac{SLA}{x_m} \\ 0 & \text{if } z > \frac{SLA}{x_m} \end{cases}.$$

According to this expression:

$$P\{\theta \geq 1\} = P\{\eta \leq SLA\} = \begin{cases} 1 - \left(\frac{x_m}{SLA}\right)^{\alpha} & \text{for } SLA \geq x_m \\ 0 & \text{for } SLA < x_m \end{cases}.$$

The resource reputation assessment can be divided into the following cases:

1. If $\frac{SLA}{x_m} \leq 1$, then $P\{u = 1\} = P\{\theta \geq 1\} = 0$. This scenario describes the resource that is always providing a bad service. It means that this resource never meets SLA. Let's assess the reputation of such a resource.

$$rep = P\{\theta \geq 1\} + \mathrm{M}[\theta \cdot \mathbf{1}_{\theta<1}] = \mathrm{M}[\theta \cdot \mathbf{1}_{\theta<1}] = \int_0^1 x p_{\theta}(x) dx =$$

$$= \int_0^{\frac{SLA}{x_m}} x\alpha\left(\frac{x_m}{SLA}\right)^{\alpha} x^{\alpha-1} dx = \alpha\left(\frac{x_m}{SLA}\right)^{\alpha} \int_0^{\frac{SLA}{x_m}} x^{\alpha} dx = \alpha\left(\frac{x_m}{SLA}\right)^{\alpha} \left.\frac{x^{\alpha+1}}{\alpha+1}\right|_0^{\frac{SLA}{x_m}} =$$

$$= \alpha\left(\frac{x_m}{SLA}\right)^{\alpha} \frac{\left(\frac{SLA}{x_m}\right)^{\alpha+1}}{\alpha+1} = \frac{\alpha}{\alpha+1}\frac{SLA}{x_m}.$$

Therefore,

$$rep = \frac{\alpha}{\alpha+1}\frac{SLA}{x_m}. \qquad (21)$$

2. If x_m=0, then $P\{u=1\} = P\{\theta \geq 1\} = 1$. This scenario describes the resource that is always providing a bad service. It means that this resource always meets SLA. The reputation of such a resource is 1, because $rep = P\{\theta \geq 1\} + \mathrm{M}[\theta \cdot \mathbf{1}_{\theta<1}] = 1$.

3. If $\frac{SLA}{x_m}$>1, then $P\{u=1\} = P\{\theta \geq 1\} = 1 - \left(\frac{x_m}{SLA}\right)^{\alpha}$. This scenario describes the resource that is always providing a partially unreliable service. It means that in some situations the agreed SLA is met by the resource and in others violated. Let's assess the reputation of such a resource.

$$rep = P\{\theta \geq 1\} + \mathrm{M}[\theta \cdot \mathbf{1}_{\theta<1}] = 1 - \left(\frac{x_m}{SLA}\right)^{\alpha} + \int_0^1 x p_\theta(x)dx =$$

$$= 1 - \left(\frac{x_m}{SLA}\right)^{\alpha} + \int_0^1 x\alpha\left(\frac{x_m}{SLA}\right)^{\alpha} x^{\alpha-1}dx = 1 - \left(\frac{x_m}{SLA}\right)^{\alpha} + \alpha\left(\frac{x_m}{SLA}\right)^{\alpha}\int_0^1 x^{\alpha}dx =$$

$$= 1 - \left(\frac{x_m}{SLA}\right)^{\alpha} + \alpha\left(\frac{x_m}{SLA}\right)^{\alpha}\frac{x^{\alpha+1}}{\alpha+1}\Bigg|_0^1 = 1 - \left(\frac{x_m}{SLA}\right)^{\alpha} + \alpha\left(\frac{x_m}{SLA}\right)^{\alpha}\frac{1}{\alpha+1} =$$

$$= 1 - \left(\frac{x_m}{SLA}\right)^{\alpha}\frac{1}{\alpha+1}.$$

Therefore,

$$rep = 1 - \left(\frac{x_m}{SLA}\right)^{\alpha}\frac{1}{\alpha+1}. \qquad (22)$$

The obtained results are summarized in Table 1.

Table1 — Different service types and reputation, calculated for Pareto distribution of QoS metrics with x_m and α parameters and fixed SLA value.

Resource type	Parameters	$P\{\theta \geq 1\}$	$\mathrm{M}[\theta \cdot \mathbf{1}_{\theta<1}]$	Reputation
Always good service	$\frac{SLA}{x_m} \leq 1$	0	$\frac{\alpha}{\alpha+1}\frac{SLA}{x_m}$	$\frac{\alpha}{\alpha+1}\frac{SLA}{x_m}$
Always bad service	x_m=0	1	0	1
Partially unreliable	$\frac{SLA}{x_m}$>1	$1 - \left(\frac{x_m}{SLA}\right)^{\alpha}$	$\left(\frac{x_m}{SLA}\right)^{\alpha}\frac{\alpha}{\alpha+1}$	$1 - \left(\frac{x_m}{SLA}\right)^{\alpha}\frac{1}{\alpha+1}$

Let's analyze the obtained reputation values in terms of the parameters value. If $\frac{SLA}{x_m} \to 1+$ (tends to 1 on right), in this case partially unreliable resource always provides bad service: when $\frac{SLA}{x_m} = 1$ the reputation of this two resources equals to $\frac{\alpha}{\alpha+1}$. When $x_m \to 0$ and the SLA value is fixed, then in this case partially unreliable resource always provides good service, because $1-\left(\frac{x_m}{SLA}\right)^{\alpha}\frac{1}{\alpha+1} \to 1$. We can get the same result if we fix x_m and $SLA \to \infty$: $1-\left(\frac{x_m}{SLA}\right)^{\alpha}\frac{1}{\alpha+1} \to 1$.

For the resource that always provides bad service: if $x_m \to \infty$ or $SLA \to 0$ reputation $\frac{\alpha}{\alpha+1}\frac{SLA}{x_m} \to 0$.

Analysis of security threat scenarios for utility-based reputation model

Usually reputation models are analysed in terms of performance, for example resource management, while less attention is paid to the analysis of security threat scenarios. In this section we will study different security threats scenarios in the area of trust and reputation management that were proposed by (Gomez Marmol and Martınez Perez, 2009), and analyse how the proposed model responds to these threats. It should be noted that some of these attacks can be handled by existing mechanisms already implemented for Grids.

1 Individual malicious peers

Malicious peers always provide bad services (Gomez Marmol and Martınez Perez, 2009). From Grid perspective, there can be either a resource that always provides unreliable services, or a malicious user that always tries to harm a system. Such an unreliable resource will provide poor services to the users that will result that the agreed SLA would not be always met (for example, $v << SLA$ for time-related QoS metrics), and thus the reputation of this resource will be always low.

2 Malicious collectives

This is a situation when malicious peers that always provide bad service form a malicious collective (Gomez Marmol and Martınez Perez, 2009). In Grids, there could be a user that tries illegally to improve the reputation of a particular resource. If the user and resource belong to the same organization that kind of behaviour will be captured by the alliance function h(u, r). In order to improve the reputation value considerably the user will need to submit a lot of simple jobs. (Here, by simple jobs we mean jobs that would not require much CPU time and will be executed within seconds.) In such a case the reputation value of the resource will be bounded with the θ-parameter of the h(u, r) function.

3 Malicious collectives with camouflage

This is a threat which is not always easy to tackle, since its resilience will mostly depend on the behavioural pattern followed by malicious peers (Gomez Marmol and Martınez Perez, 2009). These correspond to the malicious collectives with the variable behaviour. In our user reputation model, such variability could be partially

detected with the SMUB model. Moreover, reputation value for such users will vary considerably over the time as well. Therefore, with such an approach it is possible to punish such behaviour with the reputation.

4 Malicious spies

This is a threat when malicious peers (spies) always provide good services when selected as service providers, but they also give the maximum rating values to those malicious peers who always provide bad services (Gomez Marmol and Martınez Perez, 2009). In Grids this corresponds to the situation when a user with high reputation provides the maximum rating to unreliable resources.

5 Sybil attack

An adversary initiates a large number of malicious peers in the system (Gomez Marmol and Martınez Perez, 2009). Each time one of the peers in the system is scheduled as a resource provider it provides malicious service and after that it is disconnected and replaced by another peer (Chakrabarti, 2007; Douceur, 2002; Gomez Marmol and Martınez Perez, 2009). In Grids, such an attack is hardly implemented in full form since appropriate certificate should be obtained from the certificate authority in order to integrate a resource into the Grid system. Other solutions to tackle this problem are to use the methodology of active experiment to monitor the availability of resources by sending, for example benchmark tasks, to incorporate Captcha mechanisms (von Ahn et al., 2008) or to require users to register with a valid telephone or credit card number (Kuhn et al., 2008). Up to this moment, there have not been many reports of Sybil attacks on Grid systems (Kuhn et al., 2008).

6 Man in the middle attack

A malicious peer can intercept the messages between other peers, rewrite the message and change reputation values (Gomez Marmol and Martınez Perez, 2009). Our model relies on the existing Grid security mechanisms to tackle this threat (Chakrabarti, 2007).

7 Driving down the reputation of a reliable peer

Malicious peers give the worst rating to those benevolent peers, who indeed provide good services (Gomez Marmol and Martınez Perez, 2009). Projecting onto the Grids, a malicious user will provide poor ratings to the resources, though user's jobs were completed successfully with appropriate QoS value. One possible way to tackle this problem is that jobs of users with low reputation value are never sent to resources with high reputation by the resource broker. Moreover, QoS metrics in Grids are measured by the monitoring system, and the malicious user should be able to illegally obtain necessary privileges to change these values and consequently the reputation value.

8 Partially malicious collectives

Malicious peers provide malicious actions for some kind of services, and, for others, they provide good services (Gomez Marmol and Martınez Perez, 2009). In Grids, this threat corresponds to users with variable behaviour (covered in malicious collectives with camouflage subsection), and to resources that for some types of services provide poor performance. By just considering a different score for every service offered by a resource, this threat is mitigated most of the times (Gomez Marmol and Martınez Perez, 2009). That is why, we included into our model a score function to provide different reputation values for different types of tasks executed by resources.

9 Malicious pre-trusted peers

Some models are based on the strategy that there is a set of peers that can be trusted before any transaction is carried out in the system, known as pre-trusted peers (Kamvar et al., 2003). This problem refers to assigning

initial reputation to resources when a VO is formed. One possible way is to have a set of benchmarks tasks with the desired QoS metrics, execute them on all VO resources and assign the received reputation values to the resources.

Results of experiments

In this section, results of experiments are presented to assess the performance of the described model. The performance is evaluated in terms of resistance to security threat scenarios discussed in the previous section. A dedicated software application has been developed to run simulations with different scenarios.

1 Data description

In order to generate workload within experiments, i.e. jobs inter-arrival time and jobs execution time, we used real data provided by the Grid Observatory project1. This project provides data on job cycle in the EGEE grid infrastructure. In particular, we used data collected by the Real Time Monitor (RTM) systems that summarizes various information on jobs executed in the Grid. In total, the trace registers 37 attributes which categorized into Information, Timestamps and Metrics (Germain-Renaud et al., 2011).

2 Schedulers used in simulations

In all our simulations jobs are scheduled immediately after arrival. Two on-line schedulers were used in the study: a heuristic on-line scheduler that maps a job to a resource which provides the job earliest completion time (ECT), and a scheduler that uses resource reputation to map jobs (ECT-reputation). In order to integrate reputation into the latter scheduler a non-linear trade-off scheme (Voronin, 2011) is used.

Let ECT(ri) be the estimated completion time of running a job on resource ri, and ECTn(ri) is the corresponding normalized value:

$$ECT_n = \frac{ECT(r_i)}{ECT_{\max}}, \quad (23)$$

where ECTmax is the upper bound value for the ECT value.

The ECT scheduler assigns job to a resource that minimizes the following expression

$$r^* = \arg\min_{r_i}\left(ECT(r_i)\right), \quad (24)$$

while ECT-reputation scheduler assigns job to a resource that minimizes

$$r^* = \arg\min_{r_i}\left[\frac{\alpha_1}{1-ECT_n(r_i)} + \frac{\alpha_2}{rep(r_i)}\right]. \quad (25)$$

where rep(ri) denotes reputation value of resource ri, and αk are parameters. In our experiments we used the following values for parameters: α1=α2=0.5.

3 Experimental parameters

[1] Grid Observatory: www.grid-observatory.org

All experiments were run for a Grid infrastructure of 20 resources with resource productivity (in unitless standard units) being uniformly selected from the range [1, 200]. Job complexity (also, in unitless standard units) was generated from traces provided by the Grid Observatory project lying in the range [1, 56000]. Distribution of job complexity is shown in Figure 1. Job execution time on a resource was estimated as jobComplexity/resourceProductivity. Jobs inter-arrival time and workload were also generated from EGEE traces. Figures 2 and 3 show cumulative number of submitted jobs over the time and job arrival rate (in jobs/min) respectively.

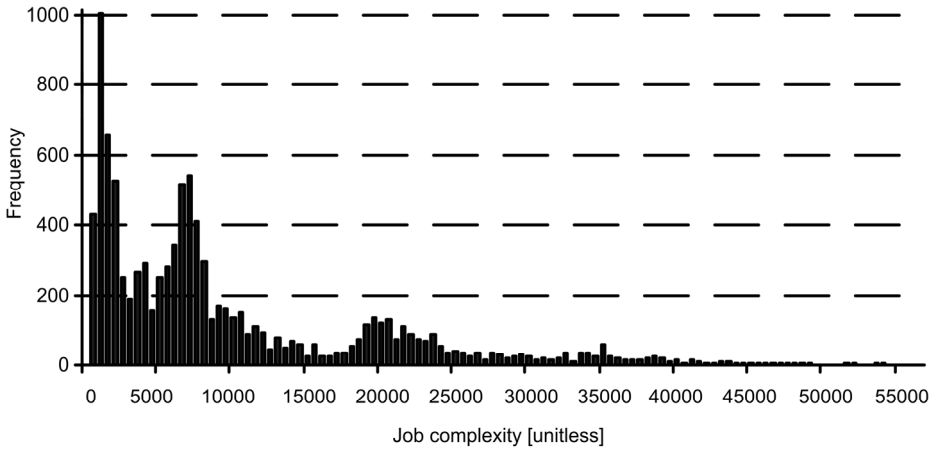

Figure 1. Distribution of job complexity within experiments (for 10000 jobs)

Within the experiments the following QoS metrics were considered: job waiting time, job execution time and job total completion time. The agreed SLA values were modelled as follows: the agreed waiting time was selected randomly from the range [1, 30000] sec, and the agreed execution time was selected as jobComplexity/minResourceProductivity. In order to simulate a scenario when a resource did not respect the agreed execution time the following approach was used: a random value from the interval [1, 2500] sec was added to the actual execution time value. The penalty function and reputation were estimated using Eq. (6) and (10) respectively. If not stated otherwise, the utility function (Eq. 5) was calculated for the job completion time QoS metric.

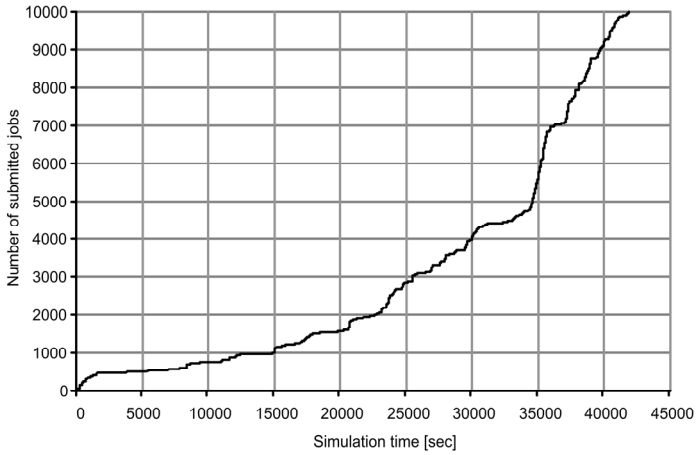

Figure 2. Cumulative number of submitted jobs

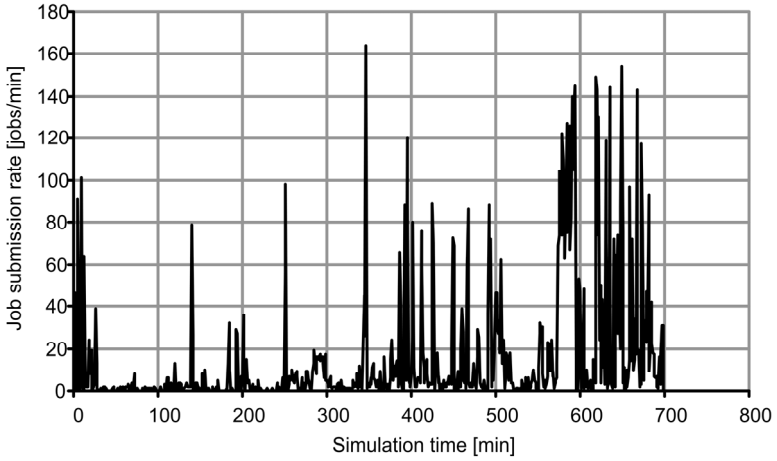

Figure 3. Job arrival rate

4 Analysis of security threat scenarios simulations

The following scenarios were run in order to evaluate the describe reputation model against security threat scenarios:

- Extreme cases. Some of the resources always provide bad services, and others always provide good services. Here, by “good” and “bad” services, we mean situations when a resource respects the agreed SLA, and when the agreed SLA is violated by a resource provider, respectively. This scenario also describes cases when a user tries to illegally improve reputation of the resource provider.

- Variable resources behaviour. Random and oscillating patterns of resource behaviour (Gomez Marmol and Martınez Perez, 2009) were considered within experiments. Within random pattern, at any time some of the resources provide either bad or good service. Within oscillating pattern, the resource is fully benevolent for a period of time and fully fraudulent for the next period, and so on (Gomez Marmol and Martınez Perez, 2009).

In both cases, ECT scheduler was applied as basic one.

Figure 4 shows how reputation for good and bad services changes over completion of the jobs.

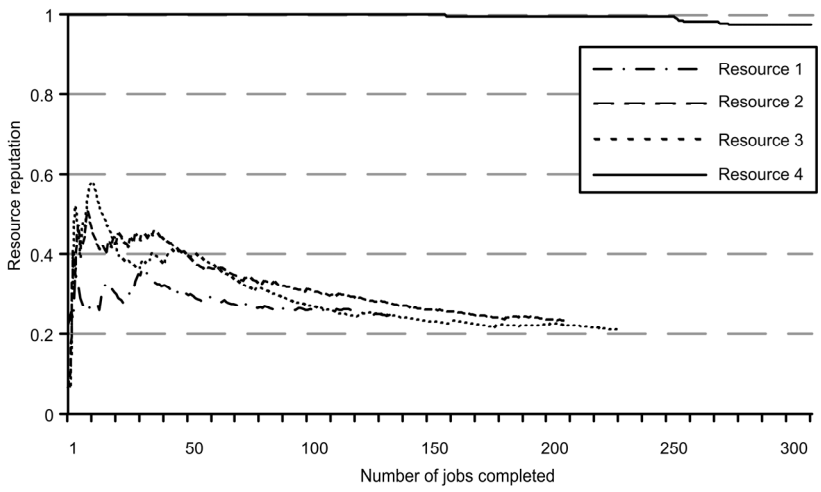

Figure 4. Reputation for resources that always provide good services (no. 1) and bad services (no. 2-4)

Since in our simulations random values are generated, it is important to analyse aggregated results for multiple runs. Figure 5 shows the average resulting reputation values at the end of simulations along with two standard deviations calculated for 10 runs.

The next scenario is when a user tries to illegally improve resource reputation. That behaviour will be captured by an alliance function (Eq. 15). Figure 6 shows an example when a user (from the same organisation as the resource) tries to illegally improve resource reputation by submitting a number of jobs for which the utility function is equal to alliance function h(u, r) (Eq. 15). Figure 6 shows how the resource reputation will change if no alliance function is applied, and if an alliance function is applied using different schemes for selecting θ-parameter (Eq. 15), in particular adaptive and fixed-value. From Figure 6, it is evident that adaptive scheme is preferable over the fixed-value scheme.

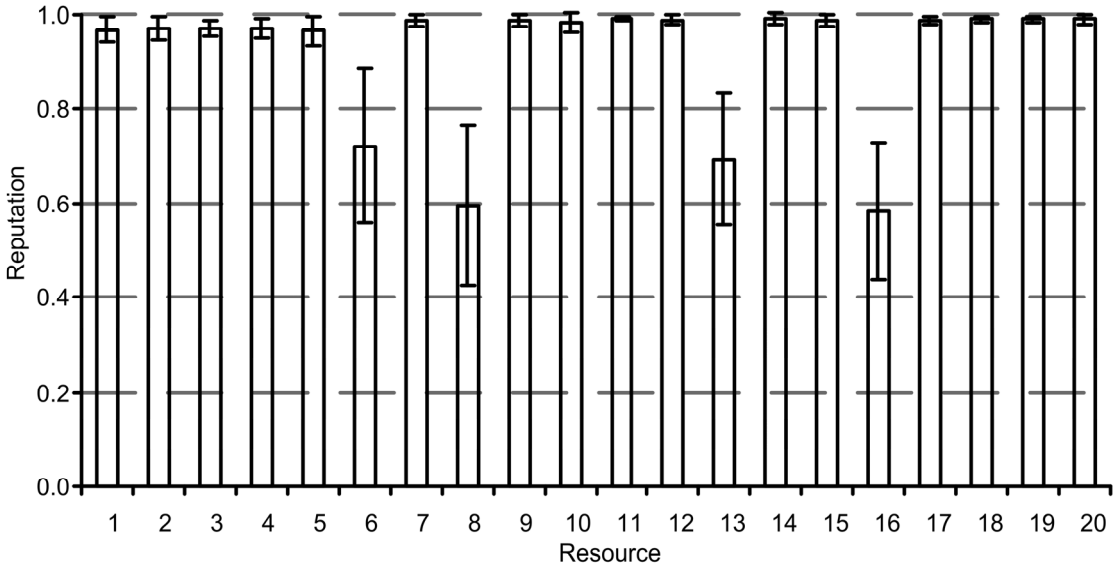

Figure 5. Resulting reputation averaged for multiple runs.
Resources with no. 6, 8, 13 and 16 always provide bad services

The following scenarios model the variable behaviour of resources. First, we consider random patterns that were modelled as follows: each “bad” resource was characterised by a trustworthiness rate. For example, if resource trustworthiness rate is equal to 0.6 then it meets the agreed SLA on average in 60% of cases. The following approach was used to simulate such scenarios: when untrustworthy resource was scheduled to execute a job, a random value uniformly distributed in the [0; 1] range was generated. If this random value was less than resource trustworthiness rate, then the resource met the agreed SLA. Otherwise, the agreed SLA is violated by the resource provider. Figure 7 shows the reputation of resources with random behavioural patterns with different trustworthiness rate.

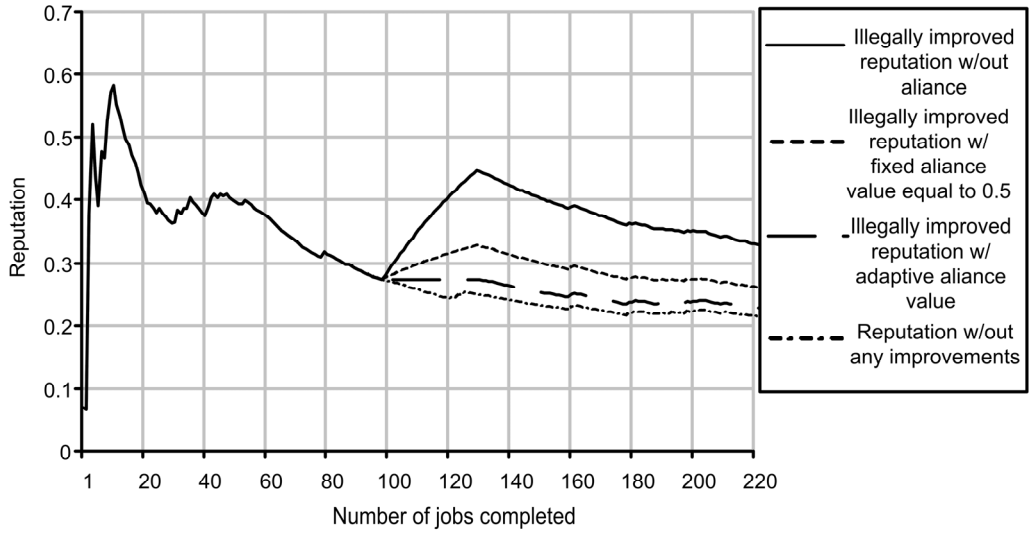

Figure 6. Situation when a user tries to illegally improve resource reputation. The user submits 30 jobs (no. 101-131) for which the estimated utility function (Eq. 17) is equal to the alliance function h(u, r) (Eq. 15). Without the alliance function, the utility would be equal to 1, and the reputation sharply increases.

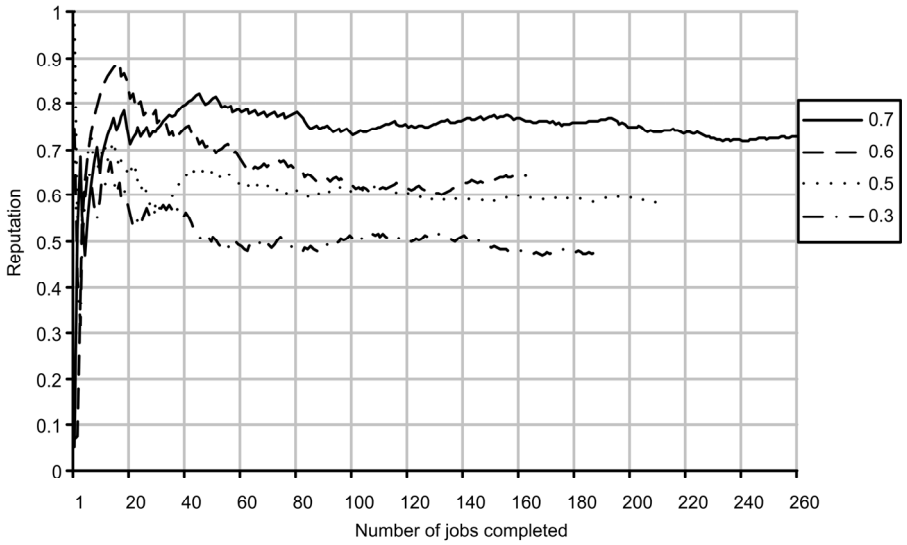

Figure 7. Scenario with resource random behavioural pattern at different trustworthiness rate

It is worth mentioning that the resulting resource reputation is close to trustworthiness rate (Table 2). It means that in such a case the proposed model was able to capture the variable pattern of resource behaviour.

Table 2. Comparison of trustworthiness rate and resulting resource reputation for random behavioural pattern scenario

Trustworthiness rate	Resulting resource reputation
0.7	0.728
0.6	0.644
0.5	0.582
0.3	0.475

The oscillating pattern suggests that the resource is fully benevolent for a period of time and fully fraudulent for the next period, and so on. Figure 8 shows how reputation changes for a resource with oscillating pattern.

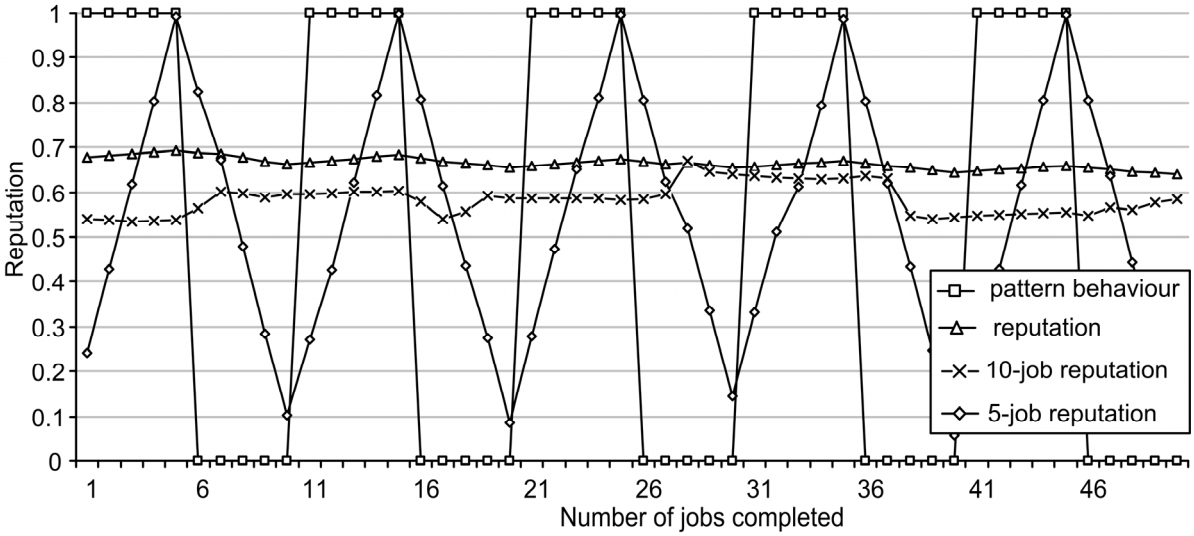

Figure 8. Scenario with resource oscillating behavioural pattern. In this example an oscillating period is equal to 5. Shown in the figure are: utility (Eq. 17), reputation estimated with no time decay function (Eq. 16), and reputation estimated with time decay function (for 5- and 10-jobs averaged utility)

From Figure 8, it is evident that just considering reputation without a time decay function does not allow us to detect the variable pattern of resource behaviour. It becomes partially possible when a parameter tc in Eq. (4) is appropriately chosen. In the shown example even a 10-jobs average utility does not exactly show the extent of variability.

The efficiency of the model was also tested against the error of the model (Gomez Marmol and Martınez Perez, 2009) which represents a malicious peers utilization (or selection percentage of malicious service providers). Here, experimental results are reported for both ECT and ECT-reputation schedulers. Within the first set of

experiments we varied number of untrustworthy service providers (that always provide bad services). Figure 9 shows the error of the model depending on the number of untrustworthy resources.

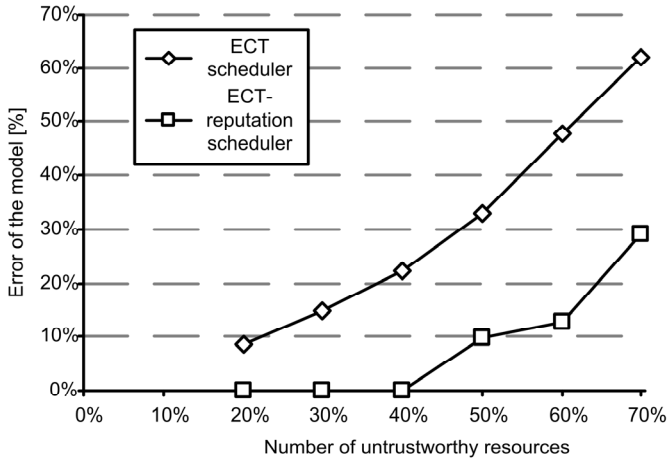

Figure 9. Error of the model depending on the number of untrustworthy resources

When using the ECT-reputation scheduler, the error of the model was below 15% when number of malicious resources was 60%, and below 30% in case of 70% of malicious resources. Within the second set of experiments we varied resource trustworthiness. We allowed 20% of the resources to be untrustworthy but with different degree of trustworthiness. Figure 10 shows the error of the model depending on resource trustworthiness rate.

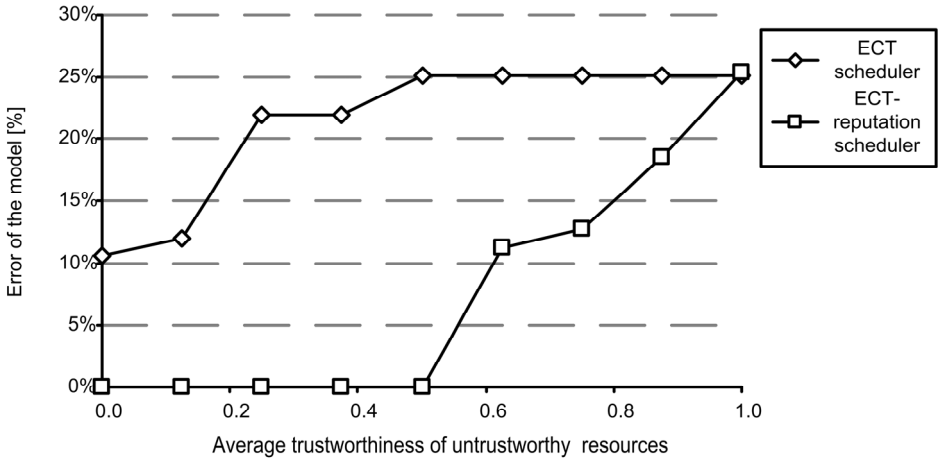

Figure 10. Error of the model depending on resource trustworthiness rate

The integration of reputation into the scheduler allowed us to reduce the error of the model. When using the ECT-reputation scheduler no jobs were scheduled until resource reputation became high (in our cases until average resource trustworthiness rate was more than 0.5, Figure 10).

Conclusions

In this paper we assessed most important and critical security threats for a utility-based reputation model in Grids. To tackle threats scenarios the model was further extended by incorporating a statistical model of user behaviour and several additional components, in particular: assigning initial reputation to a new entity in VO, capturing alliance between consumer and resource, determining time decay function, and score function.

The probabilistic estimation of trust model was proposed that allowed us to theoretically investigate properties of the model.

The experimental results showed that the model was effective in countering such threats as individual malicious peers, malicious collectives, driving down the reputation of a reliable peer. The error of the model was below 15% when number of malicious resources was 60%.

At the same time there were some limitations in countering malicious collectives with camouflage, in particular for oscillating behaviour pattern. Parameters in a time decay function have to be appropriately selected in order to detect the variable pattern of resource behaviour.

Future work should be directed on further investigation of oscillating patterns of resource behaviour and improving the model to counter these various patterns, and exploring application of the model for other large-scale service-oriented systems such as the Global Earth Observation System of Systems (GEOSS).

Acknowledgement

The paper is published with financial support by the project ITHEA XXI of the Institute of Information Theories and Applications FOI ITHEA (www.ithea.org) and the Association of Developers and Users of Intelligent Systems ADUIS Ukraine (www.aduis.com.ua).

Bibliography

[Abdul-Rahman and Hailes, 2000] Abdul-Rahman A, Hailes S. Supporting trust in virtual communities. In: Proc of the IEEE 33rd Hawaii Int Conf on Syst Sci, HICSS'00; 2000. p. 6007.

[Arenas et al, 2008] Arenas A, Aziz B, Silaghi GC. Reputation management in grid-based virtual organisations. In: Fernandez Medina E, Malek M, Hernando J, editors. Proc Int Conf on Secur and Cryptogr, SECRYPT 2008; 2008. p. 538–45.

[Azzedin and Maheswaran, 2002] Azzedin F, Maheswaran M. Integrating Trust into Grid Resource Management Systems. In: Int Conf on Parallel Process, ICPP 2002. Washington, DC, USA: IEEE Computer Society; 2002. p. 47–54.

[Chakrabarti, 2007] Chakrabarti A. Grid Computing Security. Berlin Heidelberg: Springer-Verlag; 2007.

[Douceur, 2002] Douceur JR. The Sybil attack. Lect Notes in Comput Sci 2002;2429:251–60.

[Eymann et al, 2008] Eymann T, König S, Matros R. A Framework for Trust and Reputation in Grid Environments. J of Grid Comput 2008;6(3);225-37.

[Foster et al, 2001] Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. Int J of Supercomput Appl 2001;15(3):200-22.

[Germain-Renaud et al, 2011] Germain-Renaud C, Cady A, Gauron P, Jouvin M, Loomis C, Martyniak J, et al. The Grid Observatory. In: 11th IEEE/ACM Int Symp on Cluster, Cloud and Grid Comput, CCGrid; 2011. p. 114–123.

[Marmol and Perez, 2009] Gomez Marmol F, Martınez Perez G. Security threats scenarios in trust and reputation models for distributed systems. Comput & Secur 2009;28:545–56.

[Marmol and Perez, 2008] Gomez Marmol F, Martınez Perez G. Providing trust in wireless sensor networks using a bio-inspired technique. In: Proc. of the Networking and Electronic Commerce Research Conf, NAEC'08, Lake Garda, Italy; Sep 2008.

[Haykin, 1999] Haykin S. Neural Networks: A Comprehensive Foundation. Upper Saddle River, New Jersey: Prentice Hall; 1999.

[Hluchy et al, 2010] Hluchy L, Kussul N, Shelestov A, Skakun S, Kravchenko O, Gripich Y, et al. The Data Fusion Grid Infrastructure: Project Objectives and Achievements. Comput and Inf 2010;29(2):319–34.

[Josang et al, 2007] Josang A, Ismail R, Boyd C. A Survey of Trust and Reputation Systems for Online Service Provision. Decis Support Syst 2007;43(2):618-44.

[Kamvar et al, 2003] Kamvar S, Schlosser M, Garcia-Molina H. The EigenTrust algorithm for reputation management in P2P networks. In: Proc of the 12th Int Conf on World Wide Web. New York, NY, USA: ACM Press; 2003. p. 640–51.

[Kerschbaum et al, 2006] Kerschbaum F, Haller J, Karabulut Y, Robinson P. PathTrust: A Trust-Based Reputation Service for Virtual Organization Formation. Lect. Notes in Comput Sci 2006;3986:193–205.

[Kuhn et al, 2008] Kuhn M, Schmid S, Wattenhofer R. Distributed asymmetric verification in computational grids. In: IEEE Int Symp on Parallel and Distrib Proces, IPDPS 2008; 2008. p. 1–10.

[Kussul et al, 2011] Kussul N, Shelestov A, Skakun S. Grid Technologies for Satellite Data Processing and Management within International Disaster Monitoring Projects. In: Fiore S, Aloisio G, editors. Grid and Cloud Database Management. Berlin Heidelberg: Springer-Verlag; 2011. p. 279–306.

[Kussul et al, 2009] Kussul N, Shelestov A, Skakun S. Grid and sensor web technologies for environmental monitoring. Earth Sci Inf 2009;2(1-2):37–51.

[Kussul and Skakun, 2004] Kussul N, Skakun S. Neural Network Approach for User Activity Monitoring in Computer Networks. In: Proc of Int Joint Conf on Neural Networks, Budapest, Hungary; 2004. p. 1557–62.

[Kussul and Novikov, 2009] Kussul O,Novikov O. Utility-based reputation model for VO in Grids. Visnuk NTUU "KPI": Informatics, control and computation 2009;50:137-145.

[Lecca et al, 2011] Lecca G, Petitdidier M, Hluchy L, Ivanovic M, Kussul N, Ray N, et al. Grid computing technology for hydrological applications. J of Hydrol 2011;403(1-2):186–99.

[Liang and Shi, 2010] Liang Z, Shi W. A reputation-driven scheduler for autonomic and sustainable resource sharing in Grid computing. J of Parallel and Distrib Comput 2010;70:111–25.

[Oh and Lee, 2003] Oh SH, Lee WS. An anomaly intrusion detection method by clustering normal user behavior. Comput & Secur 2003;22(7):596–612.

[Papaioannou and Stamoulis, 2008] Papaioannou TG, Stamoulis GD. Reputation-Based Estimation of Individual Performance in Grids. In: Eighth IEEE Int Symp on Cluster Comput and the Grid, CCGRID. Washington, DC, USA: IEEE Computer Society; 2008. p. 500–09.

[Schulter et al, 2008] Schulter A, Vieira K, Westphall C, Abderrahim S. Intrusion Detection for Computational Grids. In: Proc 2nd Int Conf New Technol, Mobil, and Secur. IEEE Press; 2008. p. 1–5.

[Shelestov et al, 2006] Shelestov A, Kussul N, Skakun S. Grid Technologies in Monitoring Systems Based on Satellite Data. J of Autom and Inf Sci 2006;38(3):69–80.

[Shelestov et al, 2008] Shelestov A, Skakun S, Kussul O. Intelligent Model of User Behavior in Distributed Systems. Int J on Inf Theory and Appl 2008;15(1):70–6.

[Shelestov et al, 2007] Shelestov A, Skakun S, Kussul O. Complex Neural Network Model of User Behavior in Distributed Systems. In: Proc of XIII-th Int Conf "Knowledge-Dialogue-Solutions", Varna, Bulgaria. Sofia, Bulgaria: FOI ITHEA; 2007. p. 42–9.

[Silaghi et al, 2007] Silaghi G, Arenas A, Silva L. A Utility-Based Reputation Model for Service-Oriented Computing. In: Priol T, Vanneschi M, editors. Toward Next Generation Grids. New York: Springer; 2007. p. 63–72.

[Skakun et al, 2005] Skakun S, Kussul N, Lobunets A. Implementation of the Neural Network Model of Users of Computer Systems on the Basis of Agent Technology. J of Autom and Inf Sci 2005;37(4):11–8.

[Song et al, 2005] Song S, Hwang K, Kwok YK. Trusted Grid Computing with Security Binding and Trust Integration. J of Grid Comput 2005;3(1-2):53-73.

[Srivatsa and Liu, 2006] Srivatsa M, Liu L. Securing decentralized reputation management using TrustGuard. J of Parallel and Distrib Comput 2006;66(9):1217–32.

[Sun et al, 2004] Sun HW, Lam KY, Chung SL, Gu M, Sun JG. Anomaly Detection in Grid Computing Based on Vector Quantization. Lect Notes in Comput Sci 2004;3251:883–6.

[Vieira et al, 2007] Vieira K, Schulter A, Westphall C, Westphall C. Intrusion Detection for Grid and Cloud Computing. IT Prof 2007;12(4):38–43.

[von Ahn et al, 2008] von Ahn L, Maurer B, McMillen C, Abraham D, Blum M. reCAPTCHA: Human-Based Character Recognition via Web Security Measures. Sci 2008;321(5895):1465–8.

[von Laszewski et al, 2005] von Laszewski G, Alunkal B, Veljkovic I. Towards Reputable Grids. Scalable Comput: Pract and Exp 2005;6(3):95–106.

[Voronin, 2011] Voronin AN. A multicriteria problem of distribution of bounded resources. Cybern and Syst Anal 2011;47(3):490-3.

[Wu and Sun, 2010] Wu CC, Sun RY. An integrated security-aware job scheduling strategy for large-scale computational grids. Future Generation Comput Syst 2010;26:198-206.

Authors' Information

Olga Kussul *– Phd Student, National Technical University of Ukraine "Kyiv Polytechnic Institute", Prospekt Peremogy 37, Kyiv 03056, Ukraine; e-mail: olgakussul@gmail.com*

Major Fields of Scientific Research: Grid computing, information security, design of distributed software systems.

Nataliia Kussul *– Deputy Director, Space Research Institute NASU-NSAU, Glushkov Prospekt 40, build. 4/1, Kyiv 03680, Ukraine; e-mail: inform@ikd.kiev.ua*

Major Fields of Scientific Research: Grid technologies, design of distributed software systems, parallel computations, intelligent data processing methods, neural networks, satellite data processing, risk management and space weather.

Sergii Skakun *– Senior Scientist, Space Research Institute NASU-NSAU, Glushkov Prospekt 40, build. 4/1, Kyiv 03680, Ukraine; e-mail: serhiy.skakun@ikd.kiev.ua*

Major Fields of Scientific Research: Grid computing, Sensor Web, Earth observation, satellite data processing, risk analysis.