RESEARCH ARTICLE OPEN ACCESS

A Survey On Enforcing End-To-End Application Security in Cloud Computing

T.Arunambika [1], P.Sudha [2]
Research Scholar [1], Assistant Professor [2]
Department of Computer Science
Sree Saraswathi Thyagaraja College, Pollachi.
Tamil Nadu – India

ABSTRACT

Cloud computing has recently emerged as a new paradigm for hosting and delivering services over the Internet. Cloud computing is attractive to business owners as it eliminates the requirement for users to plan ahead for provisioning, and allows enterprises to start from the small and increase resources only when there is a rise in service demand. However, despite the fact that cloud computing offers huge opportunities to the IT industry, the development of Cloud computing technology is currently at its infancy, with many issues still to be addressed. In this paper, we present a survey of cloud computing, highlighting its key concepts, architectural principles and research challenges.

Keywords:- Privacy Cloud, Private cloud, Public Cloud, Hybrid Cloud.

I. INTRODUCTION

Cloud computing is a type of computing that relies on sharing computing resources rather than having local servers or personal devices to handle applications. Cloud computing is comparable to grid computing, a type of computing where unused processing cycles of all computers in a network are harnesses to solve problems too intensive for any stand-alone machine. Cloud computing enables companies to consume compute resources as a utility just like electricity rather than having to build and maintain computing infrastructures in-house. Cloud computing promises several attractive benefits for businesses and end users. Four of the main benefits of cloud computing includes:

Self-service provisioning: End users can spin up computing resources for almost any type of workload on-demand.

Elasticity: Companies can scale up as computing needs increase and then scale down again as demands decrease.

Pay per use: Computing resources are measured at a granular level, allowing users to pay only for the resources and workloads they use.

Rule induction: The extraction of useful if-then rules from data based on statistical significance.

Cloud computing services can be private, public or hybrid. Private cloud services are delivered from a business' data center to internal users. This model offers versatility and convenience, while preserving management, control and security. Internal customers may or may not be billed for services through IT chargeback. In the public cloud model, a third-party provider delivers the cloud service over the Internet. Public cloud services are sold on-demand, typically by the minute or the hour. Customers only pay for the CPU cycles, storage or bandwidth they consume.

This paper surveys most of these issues (all but secure storage) from the perspective of trust decentralization, minimization, and management in clouds. Since users lack full control over resources in clouds, they must rely on trust mechanisms. A dictionary definition of trust is, “firm belief in the reliability, truth, ability, or strength of someone or something”. Thus it revolves around assurance and confidence that people, data, entities, information or processes will function or behave in expected ways. In a heterogeneous environment, this notion of trust is inevitably difficult to precisely quantify, so there is no universally accepted definition of trust in cloud computing. However, by reducing, eliminating, and/or distributing trust relationships between cloud infrastructure components and users, one can make relative, incremental improvements to the trustworthiness of clouds, improving their security. This relative notion of trust is well established in the general security literature.

The current system of clouds typically have centralized, universal trust of all the cloud nodes. This security paradigm suffers from a major drawback: though the nodes may be considered trustworthy by the clouds, if the attackers can compromise some, or even one, of the nodes in the cloud over time, the whole computation is compromised or data integrity and privacy can be breached. Therefore, it focuses on how to decentralize these trust relationships in clouds in order to improve security without impairing efficiency. To deal with this centralized trust in clouds; one can bring forth different possible solutions, which we can categorize into mainly two lines of defense:

First Line of Defense: Most cloud defense technologies seek to prevent attackers from compromising any cloud resources in the first place. One major category of such technologies is virtualization, which uses secure operating systems, hardware, and virtual machines to place layers of security between security-sensitive cloud resources and untrusted user activities. However, inevitably these defenses are not perfect. It is prudent to expect that some attackers will penetrate this first line of defense, motivating a second line of defense.

Second Line of Defense: Beneath the first line of defense, one can add a second line of defense to detect and mitigate successful intrusions. The classic approach adopts distributed fault tolerance|for instance, Byzantine fault tolerance. However, many fault tolerant approaches only target adversaries that act purely randomly (e.g., a hostile environment that randomly corrupts computations). In contrast, attackers are typically non-random. They strategically exploit attack vectors that subvert the defense with high probability. This has motivated research on trust management model

II. PROBLEM ISSUES

Revolutionary advances in hardware, networking, middleware, and virtual machine technologies have led to an emergence of new, globally distributed computing platforms, namely cloud computing, that provide computation facilities and storage as services accessible from anywhere via the Internet without significant investments in new infrastructure, training, or software licensing. Infograph reports that 63% of financial services, 62% of manufacturing, 59% of healthcare, and 51% of transportation industries are using cloud computing services. According to Rackspace, this pay-as-you-go service saves around 58% of cost.

As a result, more than 50% of global 1000 companies are projected to store sensitive data in public clouds by 2018. However, a significant barrier to the adoption of cloud services is customer fear of data integrity and privacy loss in the cloud.

There is numerous security issues involved in clouds, some of which include:

Privacy Preservation: preserving privacy of data and its owners,

Computation Integrity: ensuring computations are correct,

Secure Storage: storing data securely (e.g., via encryption),

Authentication and Authorization: cloud user access control, and

Secure Remote Platform Attestation: detecting and protecting against software tampering.

We show most of these issues (all but secure storage) from the perspective of trust decentralization, minimization, and management in clouds. Since users lack full control over resources in clouds, they must rely on trust mechanisms.

III. SCOPE OF THE RESEARCH

In a cloud computing environment, the traditional role of service provider is divided into two: the infrastructure providers who manage cloud platforms and lease resources according to a usage-based pricing model, and service providers, who rent resources from one or many infrastructure providers to serve the end users. The emergence of cloud computing has made a tremendous impact on the Information Technology (IT) industry over the past few years, where large companies such as Google, Amazon and Microsoft strive to provide more powerful, reliable and cost-efficient cloud platforms, and business enterprises seek to reshape their business models to gain benefit from this new paradigm. Indeed, cloud computing provides several compelling features that make it attractive to business owners, as shown below.

No up-front investment: Cloud computing uses a pay-as you- go pricing model. A service provider does not need to invest in the infrastructure to start gaining benefit from cloud computing. It simply rents resources from the cloud according to its own needs and pay for the usage.

Lowering operating cost: Resources in a cloud environment can be rapidly allocated and de-allocated on demand. Hence, a service provider no longer needs to provision capacities according to the peak load. This provides huge savings since resources can be released to save on operating costs when service demand is low. Highly scalable: Infrastructure providers pool large amount of resources from data centers and make them easily accessible. A service provider can easily expand its service to large scales in order to handle rapid increase in service demands (e.g., flash-crowd effect). This model is sometimes called surge computing [5].

Easy access: Services hosted in the cloud are generally web-based. Therefore, they are easily accessible through a variety of devices with Internet connections. These devices not only include desktop and laptop computers, but also cell phones and PDAs.

Reducing business risks and maintenance expenses: By outsourcing the service infrastructure to the clouds, a service provider shifts its business risks (such as hardware failures) to infrastructure providers, who often have better expertise and are better equipped for managing these risks. In addition, a service provider can cut down the hardware maintenance and the staff training costs.

IV. CLOUD ARCHITECTURE

The goal of cloud computing is to apply traditional supercomputing, or high-performance computing power, normally used by military and research facilities, to perform tens of trillions of computations per second, in consumer-oriented applications such as financial portfolios, to deliver personalized information, to provide data storage or to power large, immersive online computer games.

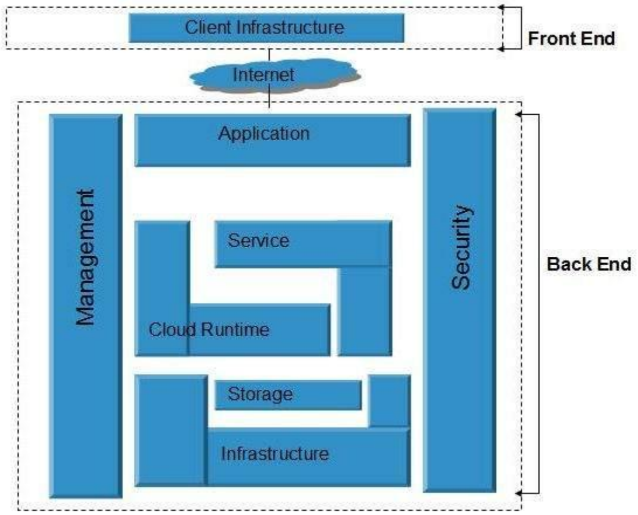

Figure -1: Architecture of Cloud Computing

To do this, cloud computing uses networks of large groups of servers typically running low-cost consumer PC technology with specialized connections to spread data-processing chores across them. This shared IT infrastructure contains large pools of systems that are linked together. Often, virtualization techniques are used to maximize the power of cloud computing. Cloud Computing architecture refers to the various components and sub-components of cloud that builds the structure of the system. Broadly, this architecture can be classified into two parts:

- Front-end
- Back-end

The front-end and back-end is connected to each other via virtual network or the internet. Besides, there are other components like Middleware, Cloud Resources etc., that is included in the Cloud Computing architecture.

(i) Front End

The front end refers to the client part of cloud computing system. It consists of interfaces and applications that are required to access the cloud computing platforms. Front-end is the side that is visible for the client, customer or the user. It includes the client's computer systemor network that is used for accessing the cloud system. Different Cloud Computing system has different user interfaces. For email programs, the support is driven from web browsers like Firefox, Chrome, and Internet Explorer etc. On the other hand, for other systems there are unique applications shared between the client and the service provider.

(ii) Back End

The back End refers to the cloud itself. It consists of all the resources required to provide cloud computing services. It comprises of huge data storage, virtual machines, security mechanism, services, deployment models, servers, etc. Back-end is the side used by the service provider. It includes various servers, computers, data storage systems, virtual machines etc., which builds together the cloud of computing services. This system can include different types of computer programs. Each application in this system is managed by its own dedicated server. The back-end side has some responsibilities to fulfill towards the client:

To provide security mechanisms, traffic control and protocols to employ protocols that connects networked computers for communication

(iii) Protocols

One central server is used to manage the entire Cloud Computing system. This server is responsible for monitoring the traffic and making each end run smoothly without any disruption. This process is followed with a fixed set of rules called Protocols. Also, a special software named as Middleware is used to perform the processes. Middleware connects networked computers to each other. Depending on the demand of client, the storage space is provided by the Cloud Computing service provider. While some companies require huge number of digital storage devices, few others require not as many. Cloud Computing service provider usually holds twice the number of storage space that required by the client. This is to keep a copy of client's data secured during the hours of system breakdown. Building copies of data for backup is called as Redundancy.

IV. POTENTIAL PRIVACY RISKS

When it comes to cloud computing, the security and privacy of personal information is extremely important. Given that personal information is being turned over to another organization, often in another country, it is vital to ensure that the information is safe and that only the people who need to access it are able to do so. There is the risk that personal information sent to a cloud provider might be kept indefinitely or used for other purposes. Such information could also be accessed by government agencies, domestic or foreign (if the cloud provider retains the information outside of Canada). For businesses that are considering using a cloud service, it is important to understand the security and privacy policies and practices of the provider. The terms of service that govern the relationship with the provider sometimes allow for rather liberal usage and retention practices.

While there are benefits, there are privacy and security concerns too. Data is travelling over the Internet and is stored in remote locations. In addition, cloud providers often serve multiple customers simultaneously. All of this may raise the scale of exposure to possible breaches, both accidental and deliberate.

Concerns have been raised by many that cloud computing may lead to "function creep" uses of data by cloud providers that were not anticipated when the information was originally collected and for which consent has typically not been obtained. Given how inexpensive it is to keep data, there is little incentive to remove the information from the cloud and more reasons to find other things to do with it.

Security issues, the need to segregate data when dealing with providers that serve multiple customers, potential secondary uses of the data—these are areas that organizations should keep in mind when considering a cloud provider and when negotiating contracts or reviewing terms of service with a cloud provider. Given that the organization transferring this information to the provider is ultimately accountable for its protection, it needs to ensure that the personal information is appropriate handled.

V. RELATED WORK

Trust plays an important role in cloud environments. It lends itself to estimate the trustworthiness of cloud service providers where trustworthiness could mean reliability, security, capability and availability. In this section, we will formulate the problem statement and then review related work.

Hatman proposed a decentralized trust through replication in the cloud, and ensures computation integrity. To evaluate reputation-based trust management in a realistic cloud environment, we augment a fullscale, production-level data processing cloud|Hadoop MapReduce [1] with a reputation-based trust management implementation based on EigenTrust [2]. The augmented system replicates Hadoop jobs and sub-jobs across the untrusted cloud nodes, comparing node responses for consistency. Consistencies and inconsistencies constitute feedback in the form of agreements and disagreements between nodes. These form a trust matrix whose eigenvector encodes the global reputations of all nodes in the cloud. The global trust vector is consulted when choosing between differing replica responses, with the most reliable response delivered to the user as the job outcome. To achieve high scalability and low overhead, we show that job replication, result consistency checking, and trust management can all be formulated as highly parallelized MapReduce computations. Thus, the security offered by the cloud scales with its computational power.

Different cloud computing platforms may vary in the details of their internal architectures, usually with one or few centralized master nodes and a large collection (e.g., hundreds of thousands) of slave nodes in Hadoop. The trust management system is centralized in the sense that master nodes maintain a small, trusted store of trust and reputation information; however, all computation is decentralized in that trust matrix computations and user-submitted job code is all dispatched to slave nodes.

AnonymousCloud [3] also proposed decentralized trust by decoupling billing information from submitted jobs. This work concerns the problem of privacy-preserving computation. AnonymousCloud conceals data provenance from cloud nodes that compute over the data, and conceals recipient identities in the form of IP addresses and ownership labels. Anonymization is achieved through the instantiation of a Tor anonymizing circuit [4] inside the cloud, through which private data and jobs are anonymously supplied by and returned to users. Circuit length is a tunable parameter k, according a exible trade-off between the degree of anonymity and the computational overhead of the circuit. To maintain a pay-per-use business model, clouds must inevitably track ownership information at some level for billing and auditing purposes. AnonymousCloud therefore implements a public-key cryptography-based anonymous authentication that disassociates data ownership metadata from the private data it labels.

Penny [3] proposed the above frameworks using centralized master nodes in clouds that are trusted for integrity, in order to reuse the existing cloud infrastructure. To eliminate that centralized trust, we can adopt a structured peer-to-peer (P2P) topology.. All of the master nodes can act as peers, and they distribute jobs and data between them. However, in order to obtain that level of decentralization, we must abandon the existing cloud structure and implement a whole new protocol. Penny uses a distributed reputation management system based on EigenTrust [2] to securely manage data labels without the introduction of a central authority. The data labels empower requester peers to avoid downloads of low-integrity data, and allow sender peers to deny low-privilege peers access to high-confidentiality data. In addition, sender peers may publish and serve their data anonymously, frustrating attacks that seeks to single out and target owners of security-relevant data.

Khan and Hamlen [3] proposed a cloud cover which decentralizes trust as well|this time on the user side. CloudCover allows untrusted Java computations in an untrusted environment to yield a proof of computation integrity as a side-effect of the computation. The proof can then be validated against the original code and the computation's result to formally verify that the result is correct. Neither the computation nor the proof (nor their origins) are trustedby CloudCover. A (possibly forged) proof either proves that a given computation results from a given code, or it does not. If the former, the result is correct regardless of where the proof came from; if the latter, the computation, the proof, or both are untrustworthy. Thus, CloudCover can be formalized as proof-carrying computation, similar to proof-carrying code[5]. CloudCover proofs have the advantageous quality that the task of verifying them can be parallelized almost arbitrarily even when the original computation is not parallelizable. Thus, they derive maximal benefit from massively parallel architectures,like clouds.

Khan et al.[3] proposed a SilverLine is a novel, more modularframework for enforcing mandatory information flow policies on commodity workflow clouds by leveraging Aspect-Oriented Programming (AOP) and In-lined Reference Monitors (IRMs). Unlike traditional system-level approaches, which typically require modifications to the cloud kernel software, OS/hypervisor, or cloud file system layout, SilverLine automatically in-lines secure information flow tracking code into untrusted job binaries as they arrive at the cloud. This facilitates efficient enforcement of a large, flexible class of information flow and mandatory access control policies without any customization of the cloud or its underlying infrastructure. The cloud and the enforcement framework can therefore be maintained completely separately and orthogonally.

Recent efforts have demonstrated interest in resource sharing across multi-site networks, such as Grids, in a coordinated manner. PlanetLab architecture is evolving to be deployed by other organisations and enable federations of PlanetLab’s [6]. Similarly, the Grid'5000 comprises nine sites geographically distributed in France [7]. Architecture and mechanisms based on the idea of peering arrangements between Grids to enable resource sharing across Grids is described in [8]. The focus of these prior works has been on the problem of resource exchange between different Grids. Trust is the firm belief in the competence of an entity to act as expected within a specific context at a given time [9]. Reputation is a measure that is derived from direct or indirect knowledge of earlier interactions of peers and is used to access the level of trust a peer puts into another [10, 9]. As an entity can trust another entity in the network based on a good reputation, we can use reputation to build trust [11]. This means reputation can serve, in the sense of reliability, as a measure of trustworthiness. Reputation management has an important role in establishing cooperative relationships between users and service providers by lowering some of the risks [10]. Trust can be used to measure our confidence that a secure system behaves as expected. A reliable trust management system provides capability to convert the unpredictable, highly dynamic pervasive environment into a trusted business platform. Thus, as the scope of hybrid cloud computing enlarges to ubiquitous and pervasive computing, there will be a need to assess and maintain the trustworthiness of the cloud computing entities.

The reputation scheme helps building trust between peers based on their past experiences and feedbacks from other peers. Research in the area of trust and reputation systems has put a lot of effort in developing various trust models and associated trust update algorithms that support users or their agents with different behavioral profiles. Trust and reputation systems have been recognized as playing an important role in decision making in the Internet world. The recent work on trust management for Grid computing shows that modeling trust is of great importance for the future developments of the Grid computing. As a result, integration of trust management system in standard grid computing has lately received attention [10,9]. For example, a trust brokering system that operates in a peer-to peer manner is proposed in [9]. An extension of Grid information service with reputation management service and its underlying algorithm for computing and managing reputation in service-oriented grid computing is discussed in [6]. An approach that enhanced the InetrCloud broker to evaluate the trustworthiness of the cloud service providers considering the user's feedback values and the performance criteria has been discussed in [6].

A method to filter out dishonest feedbacks for Bayesian reputation systems is presented in [13]. Bayesian reputation systems use the beta distribution in predicting a peer's reputation using the number of trustworthy and untrustworthy transactions as the distribution parameters. Honesty checking is performed by identifying feedbacks whose expected probability is less than the probability density function (PDF) at a certain quantile, q, and those exceeding the PDF at (1 - q) quantile in the beta distribution of the aggregated recommendations. The recommenders providing those outliers are considered to be dishonest.

Abawajy [14] introduced an honesty rating factor that represents opinion about how credible is as a provider of second-hand information. It is very difficult to distinguish between inactive or raters with fewer successful transaction from malicious raters. In the proposed approach, we develop a way to identify the credibility of the feedbacks and filter out those feedbacks that lie outside the norm. An approach for selecting computational Grid resources on the basis of trust and reputation to execute the jobs is discussed in [14]. A model based on controlled anonymity and cluster filtering methods to detect and exclude any highly unfair ratings is proposed in [15]. A collaborative filtering scheme is used to calculate an unbiased personalized reputation score. Using this method, groups of buyers who give similar ratings are classified into upper and lower classes. The final reputation score is calculated using the lower classes only. An approach based on the Dempster-Shafer theory of evidence to detect and protect users against spurious ratings is proposed in [26]. Although exiting works are complementary to the work proposed in this paper, to the best of our knowledge, this is the first work that attempts to build trustworthiness in intercloud computing. The focus on trust and reputation management systems is mainly to enhance the security of the web services. Moreover, exiting reputation systems for the standard Grids also suffer from a number of attacks that weaken trust management systems. The proposed trust model prevents many of such attacks and improves the reliability and the welfare of the system.

VI. A DATA OWNERSHIP PRIVACY TECHNIQUE IN CLOUD COMPUTING

Secure multiparty computation and differential privacy are both powerful approaches to privacy-preserving cloud computation on decrypted data, but are inapplicable to many real-world cloud computations. In particular, jobs submitted to the cloud as arbitrary binary code are difficult to automatically reformulate as secure multiparty computations, and high differential privacy sometimes comes at the expense of highly imprecise, noisy results. In these cases, the level of privacy can sometimes be improved by concealing data ownership, provenance, and/or semantics from the participants in a computation in addition to (or instead of) anonymizing the data itself. For example, a computation that mines medical data might be deemed insecure if cloud nodes receive sequences of numbers labeled "patient temperatures" with owner id "Appollo Hospital"; however, the same computation might be deemed suitably private if each node receives only unlabeled sequences of numbers amidst a context of millions of other similar anonymous jobs for thousands of diverse, anonymous users.

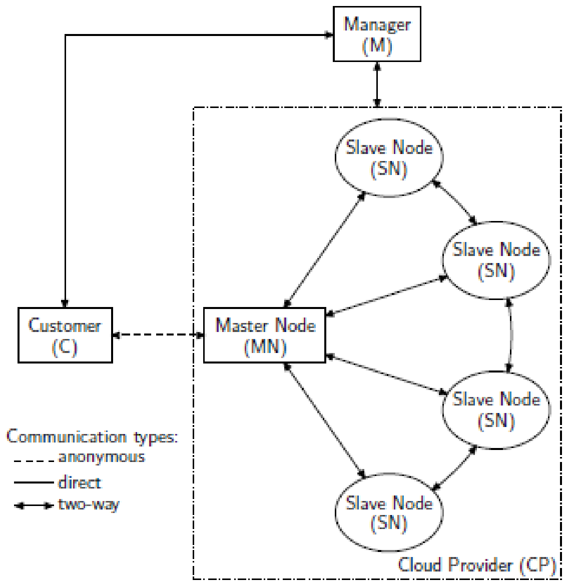

Figure-2: System architecture of AnonymousCloud

The cloud nodes that compute over the data, and conceals recipient identities in the form of IP addresses and ownership labels. Anonymization is achieved through the instantiation of a Tor anonymizing circuit inside the cloud, through which private data and jobs are anonymously supplied by and returned to users. Circuit length is a tunable parameter k, affording a flexible trade-off between the degree of anonymity and the computational overhead of the circuit. To maintain a pay-per-use business model, clouds must inevitably track ownership information at some level for billing and auditing purposes. AnonymousCloud therefore implements a public-key cryptography-based anonymous authentication that disassociates data ownership metadata from the private data it labels shown in Figure-2. Thus, a separate manager node that does not have access to the private data can bill customers appropriately using the ownership metadata, while computation nodes that have access to the private data but not the metadata can securely carry out the anonymous job. Managers are trusted not to collude with computation nodes to violate privacy, but all other nodes including the master node are potentially malicious.

VII. CONCLUSION

Cloud computing offers benefits for organizations and individuals. There are also privacy and security concerns. If we are considering a cloud service, we should think about how our personal information, and that of our customers, can best be protected. Carefully review the terms of service or contracts, and challenge the provider to meet our needs.

REFERENCES

[1] Apache (2013). Hadoop. http://hadoop.apache.org.

[2] Kamvar, S., M. Schlosser, and H. Garcia-Molina (2003). The EigenTrust algorithm for reputation management in P2P networks. In Proceedings of the 12th International World Wide Web Conference (WWW), pp. 640-651.

[3] Khan, S. M. and K. W. Hamlen (2012b). Hatman: Intra-cloud trust management for Hadoop. In Proceedings of the 5th IEEE International Conference on Cloud Computing (CLOUD), pp. 494-501.

[4] Dingledine, R., N. Mathewson, and P. Syverson (2004). Tor: The second-generation onion router. In Proceedings of the 13th USENIX Security Symposium, pp. 303-320.

[5] Necula, G. C. and P. Lee (1998). Safe, untrusted agents using proof-carrying code. In Proceedings of Mobile Agents and Security, pp. 61-91.

[6] Vijayakumar, V., WahidaBanu, R.S.D., and Abawajy, J. Novel Mechanism for Evaluating Feedback in the Grid Environment on Resource Allocation, The 2010 International Conference on Grid Computing and Applications(GCA 2010), pp:11-17, July 12 - 15, Las Vegas, Nevada, USA.

[8] F. Azzedin and M. Maheswaran, “A Trust Brokering System and Its Application to Resource Management in Public Resource Grids”, in Proceedings of IPDPS 2004.

[9] Marcos Dias de Assuncao: Provisioning Techniques and Policies to Enable Inter-Grid Resource Sharing, PhD Thesis, Melbourne University, 2009.

[10] L. Peterson and J. Wroclawski. Overview of the GENI architecture. GENI Design Document GDD- 06-11, GENI: Global Environment for Network Innovations, January 2007.

[11] Jemal H. Abawajy, Andrzej M. Goscinski: A Reputation-Based Grid Information Service. International Conference on Computational Science (4) 2006: 1015-1022.

[12] A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. Decision Support Systems, 43(2):618–644, March 2007.

[13] C. Catlett, P. Beckman, D. Skow, and I. Foster, “Creating and operating national-scale cyberinfrastructure services,” Cyberinfrastructure Technology Watch Quarterly, vol. 2, no. 2, pp. 2–10, May 2006.

[14] C. Dellarocas. Immunizing Online Reputation Reporting Systems Against Unfair Ratings and Discriminatory Behavior. In ACM Conf

[15] Jemal Abawajy, Determining Service Trustworthiness in Intercloud Computing Environments, Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN '09), pp.784~788, 2009.

[16] B. Yu, M. P. Singh, and K. Sycara, "Developing trust in large-scale peer-to-peer systems”, 2004, pp. 1-10.4-260, 1999.