Providing Data Security in Cloud Computing using Novel and Mixed Agent based Approach

Siva Tarigonda
Department of CSE,
SVCE College of Engg,
Tirupati, 517507, India

Ganesh A
Department of CSE,
SVCE College of Engg,
Tirupati, 517507, India

Srinivasulu Asadi
Department of CSE,
SVCE College of Engg,
Tirupati, 517507, India

ABSTRACT
Determining the user's trust is a growing concern for ensuring privacy and security in a cloud computing environment. In a cloud, user's data is stored in one or more remote server(s) which poses more security challenges for the system. Most important concern is to protect user's sensitive information from other users and hackers who may cause data leakage in cloud storage. This paper is to aims towards proposing a new trusted and collaborative agent-based two-tier framework to protect cloud resources. Uniqueness of the proposed security solution is to ensure security and privacy both at the service provider level as well as at the user level in a cloud environment. Existing System is mainly designed under traditional cryptography techniques which will be frequently affected with attacks such as SQL injections, Cross Site Scripting, Domain name service (DNS) attack, Denial of service (DOS) attack and Distributed Denial of service (DDOS) attack. Disadvantages of Existing System is less secured, It is frequently affected with attacks such as SQL injection, Cross Site Scripting, Domain Name Service (DNS) attack, Denial of Service (DOS) attack, Distributed Denial of Service (DDOS) attack, there is frequent data leakage and it has poor performance. Proposed Cloud Security Framework is Two-tier Architecture includes Broker Domain and Cloud Service Provider Domain. Another one is Broker Domain includes Cloud Service User (CSU), Proxy Server and Cloud Service User Agent (CSU_A). Another one is Cloud Service Provider Domain includes Cloud Service Provider (CSP) and Cloud Service Provider Agent (CSP_A). Features of Proposed Model are domain-based and set a special trust agent in each domain to manage trust. It distinguishes two different roles in cloud: customer and provider and designs different trust strategies for them. Advantages are Domain remains unaffected (with only decreased amount of trust degree than that of non-trusted users) when a said non-trusted CSU does malicious activities in the system. The trust degree of the domain will decrease accordingly with the malicious activities and updating policies. The CSP_A and CSU_A maintain their own databases, user activities information and updated trust degrees for calculating updated trust degree. It provides more security when compared to earlier models. The performance of the system is high to that of the previous models.

Keywords
Cloud security, Denial of service (DOS) attack, ensuring privacy and security, Distributed Denial of service (DDOS), Domain Name Service (DNS), Cloud Service User (CSU), Proxy Server and Cloud Service User Agent (CSU_A),cloud computing environment,
VM (Virtual Machine) monitoring system.

1. INTRODUCTION
In recent times, cloud computing is evolving as a revolutionary technique for the way we compute. In its way of evolution, starting from cluster computing through grid computing, it considered two important parameters of distributed computing paradigm namely flexibility and utilization. While cluster computing provides high flexibility of managing the resources at the cost of lower resource utilization and grid computing provides better utilization of resources at the cost of lesser flexibility of managing those resources, cloud computing provides both high flexibility as well as high resources utilization. However we are gaining those advantages at the cost of high security threats and privacy challenges since cloud computing deals with the computation and data at third party's infrastructure. Cloud computing deals with providing storage and computation resources as a service to the Cloud Service User (CSU) in the form of Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS), Storage as a Service (SaaS) etc. Software as a Service ensures to provide services as pay- as- you- go pricing scheme where customer does not need to install configure or run the application on their local computers. Platform as a Service offers a software execution environment to deploy Web- based applications. Users do not need to think about the cost and complexity of buying servers or setting the infrastructure. Therefore PaaS refers to provide a development platform to deploy, host or maintains their applications. Infrastructure as a Service shares hardware resources for executing services using virtualization. To date from a small investor to a big IT company everyone is now relying on this system. Cloud computing has several advantages such as ease to use and maintenance, need low power consumption for operation and reductions in the overhead for storing and servicing the data. In spite of several advantages cloud also suffers from different security threats and risks to protect its resources from unauthorized users and hackers. These security threats and attacks are the biggest concern towards the improvement of a more secure cloud infrastructure. Traditional mythologies are not enough to adopt for protecting cloud resources as they become obsolete with respect to the ever evolving security threats as well as to avoid data losses in the cloud environment. Moreover data stored in cloud is not just merely stored, but rather this data gets accessed by large number of times and changes in the form of insertion, deletion or updation that take place from time to time. Security and privacy even with traditional information security systems and networks has been difficult to satisfy and this is also a challenging job for cloud environment. The primary focus of this project is to introduce a novel and trusted security framework for securing cloud resources. The problem of determining users' trust for ensuring privacy and security in a cloud computing environment, though hitherto has been studied in literature, reported no proper and correct

techniques to prevent data leakage in cloud storage. Furthermore, the existing techniques are outmoded. One of the goals of this project is to highlight the importance of the problem, introduce a novel and trusted security framework for securing cloud resources and analyse the performance of the proposed scheme in a simulated test bed.

2. RELATED WORK

Trust is referred to the recognition of entity's identity and the confidence on its behaviours. Trust model is trust management methods or protocols including trust establishment, trust renewal and trust withdrawal. Many kinds of trust models have been designed for distributed systems, such as PKI-based trust model, network topology based trust model, behaviour-based trust model, subjective trust model, domain-based trust model and so on. Paper [14] compared the above trust models. Based on former technical frameworks, some researchers proposed the hierarchical cloud architecture. The hierarchical model contains five main layers which are classified based on different level of service abstraction. Paper [15] demonstrated the detail of the layered architecture and illustrated inter-relations of each layer. In contrast, some experts believed that cloud computing is the business model of earlier technologies in which cloud providers should first of all meet different customers' QoS requirements. So they proposed market-oriented cloud architecture [16-17]. Unfortunately so far no cloud architecture can illuminate in detail how to design and deploy security module in their models. While in fact security risks can never be ignored and there are still a lot of other challenges in the real cloud commercial applications for example pricing and reliability. So this project proposed a novel security model. The proposed security model has the following unique features are It adds an independent trust management module on the top of traditional security modules and Compared to former models, it can achieve high transaction success rate. The Exiting system mostly used the security models which are based on traditional cryptographic approaches. Some referred to dynamic security measures for a cloud environment while other domain based applications discussed the growing security concerns of cloud infrastructure. The traditional security measures are now not enough for that purpose.

DISADVANTAGES: The following are the disadvantages of the existing system:

- ✓ It is less secured.
- ✓ It is frequently affected with attacks such as SQL injection, Cross Site Scripting, Domain Name Service (DNS) attack, Denial of Service (DOS) attack, Distributed Denial of Service (DDOS) attack, etc...
- ✓ It has poor performance.
- ✓ There is frequent data leakage.

3. PROPOSED SYSTEM

This project introduces a novel domain-based trust model to ensure the security and interoperability of cloud and cross-clouds environment. It also introduces a novel security framework with an independent trust management module on top of traditional security modules. Using the new security model, it put forward some trust-based security strategies for the safety of both cloud customers and providers. The term 'WAY' denotes a way of secure data communication between the Cloud Service Provider (CSP) and the Cloud Service User (CSU) in a heterogeneous cloud computing environment. It is done by asking 'Who Are You?' to determine and satisfy the basic trust requirements of the service requesting CSUs. Two- tier architecture has been proposed in this paper. One is Broker Domain and another is Cloud Service Provider Domain where Broker Domain is denoted by Level_1 and Cloud Service Provide Domain is denoted by Level_2. This two level communication authenticates the trusted CSUs for accessing private information from cloud data storage.

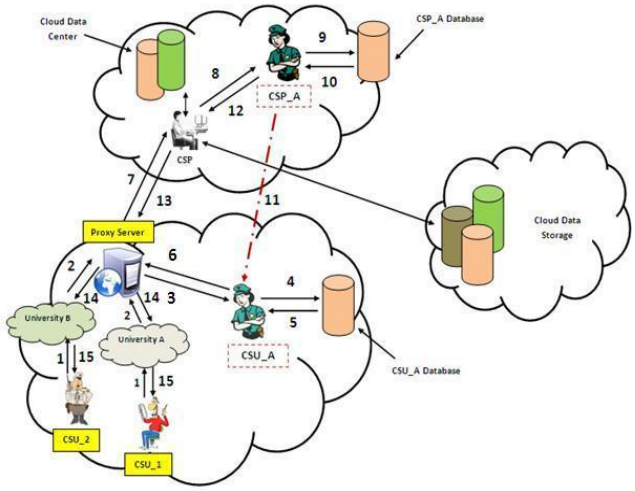

Figure 3.1: Proposed Cloud Security Framework

Proposed novel VM monitoring techniques assures the trustworthiness of the system by calculating current or updated trust degree for each service requesting CSU and the domain from where the request is coming. In this model CSU (such as CSU_1 in Figure 1) requests information from the Cloud Service Provider (CSP). As the first step, when any CSU who wants to send any request to the CSP, they have to pass the correct authentication data (such as user id and password set for them by the system) through a proxy server situated in its domain. In this approach proxy server is used as a communication channel between two domains. As an example, University A and University B is denoted for a specific group of users who requires their University specific authentication data for sending their requests to CSP through the proxy server. When the request passes through the proxy server it reaches to the trusted- agent situated at broker. When this request for information reaches to the CSP, it's immediately passes the request to the trusted- agent (denoted as Cloud Service Provider Agent, simply CSP_A) situated in the same domain to check the trust degree of the domain from where this request came. Agent CSP_A then checks the current or updated (this updation is done by the previous successful of unsuccessful iteration information) trust degree for this particular domain and then sends this result back to the CSP, only if trust degree is greater than the current threshold value set for this domain. Then the CSP will allow passing the requested information back to that particular CSU through the proxy server. CSP_A will update the trust degree after successfully executing the task. If the domain trust is less than that of the current or updated trust degree, or the CSU does any malicious activities, CSP_A immediately inform immediately inform this report to the CSU_A situated at broker domain for taking necessary actions. CSU_A will in turn decrease the trust value for this particular user. After few (depending on the types of communications) non- trusted activities or reports, CSU_A will remove this particular CSU from its domain.

3.1 Features of Proposed Model

The newly proposed model has the following features:

- ✓ It is domain-based and sets a special trust agent in each domain to manage trust. Cloud resources belong to one cloud provider will be managed in one trust domain.
- ✓ It distinguishes two different roles in cloud: customer and provider and designs different trust strategies for them.
- ✓ It treats trust recommendation as one kind of service which will accelerate the establishment of entities trust relationship.
- ✓ The trust decision and refresh mechanisms takes into account both the time factor and transaction factor.

3.2 Advantages

The Proposed system has the following advantages:

a) Domain remains unaffected (with only decreased amount of trust degree than that of non-trusted users) when a said non-trusted CSU does malicious activities in the system.

b) The trust degree of the domain will decrease accordingly with the malicious activities and updating policies.

c) The CSP_A and CSU_A maintain their own databases, user activities information and updated trust degrees for calculating updated trust degree.

d) It provides more security when compared to earlier models.

e) The performance of the system is high to that of the previous models.

4. IMPLEMENTATION

In a cloud computing environment different services are carried out on behalf of customers on hardware to which the customers have no access. The input data for cloud services is uploaded by the user to the cloud storage that means they typically result in user's data being present in unencrypted form on a machine that the user does not own or control. This poses some inherent challenges in terms of security and privacy for the system where one of the top risks is the delivery of private data to an unauthorized user. The basic motivation of developing this proposed architecture is to stop the services of a non- trusted CSU in a heterogeneous cloud environment after unsuccessfully executing any request. For simulation three types of CSU have been considered, they are as Trusted, Innocent and Non- Trusted. There are two types of tasks they may be carried out automatically during the communication and they are denoted as trusted task and non- trusted task. There are several requests that can be processed in a certain interval of time to perform these tasks. User id, Task id and Domain id for corresponding CSU, task and domains are given in the system. Simulation code is written in Java programming language, the performance analysis is done on a computer having following configuration: 2 GB RAM, 500 GB Hard disk an Intel core i3 processor @2.4 GHz. In a cloud computing environment different services are carried out on behalf of customers on hardware to which the customers have no access. The input data for cloud services is uploaded by the user to the cloud storage that means they typically result in user's data being present in unencrypted form on a machine that the user does not own or control. This poses some inherent challenges in terms of security and privacy for the system where one of the top risks is the delivery of private data to an unauthorized user. The basic motivation of developing this proposed architecture is to stop the services of a non- trusted CSU in a heterogeneous cloud environment after unsuccessfully executing any request.

For simulation three types of CSU have been considered, they are as Trusted, Innocent and Non- Trusted. There are two types of tasks they may be carried out automatically during the communication and they are denoted as trusted task and non- trusted task. There are several requests that can be processed in a certain interval of time to perform these tasks. User id, Task id and Domain id for corresponding CSU, task and domains are given in the system. Simulation code is written in Java programming language, the performance analysis is done on a computer having following configuration: 2 GB RAM, 500 GB Hard disk an Intel core i3 processor @2.4 GHz.

4.1 Assumptions for this Simulation

CSU's trust is the user's identity trust. User's identity trust in cloud is not enough; the issues of user's behaviour trust should also be evaluated and managed. So it needs a mutual mechanism to establish trust between the CSUs and the CSPs as:

(1) User's trust to the provider and

(2) Provider's trust to the users.

A trusted monitoring function should be integrated into the system to supervise the participant CSU's behaviour and depending on user's behaviour a trust management mechanism must be incorporated to update CSU's trust value. After evaluating user's behaviour, it is required to manage this trust value efficiently. Proposed novel trusted and collaborative agent- based security framework takes users behaviour evidence from CSP and manage user trust value from it.

The proposed security framework is based on a trust model. The trust degree of any CSU increases after performing any trusted communication. Similarly for any non-trusted communication the trust degree decreases and the corresponding trust table updated by the trusted- agents for next task. The probability of executing a task successfully for any non- trusted user is very low than that of any trusted and innocent users in the same domain. A novel trust- based algorithm is used to determine the trust of any service requesting user to deliver the requested information from cloud data storage. Suppose there are U numbers of users present in a domain D where U = {u1, u2, u3, … , un}. These users may be trusted, innocent or non- trusted. U Є {T1, T2, T3}. Where T1 represent trusted user, T2 represent innocent user and T3 represent non- trusted user. Similarly there are two kinds of tasks that may be performed during any communication; they are trusted task, denoted by Tt and non- trusted task denoted by Tn. There are N numbers of tasks (where N Є {n1, n2, n3, … , nn}) that can be performed in a simulation. For any instance a user u1 belongs to T1 can perform the task of type Tt for n1 times. As for example, a trusted user after successfully completion of a given task T the trust degree will be increased and after an unsuccessful communication the trust degree will be decreased accordingly to the performance. Probability function is used to determine the trust degree of any service requesting CSU and then marked them as trusted, innocent or non- trusted. It should be noted that the probability of getting higher trust

value by performing a trusted task by a trusted user, is always better than a non- trusted or an innocent user. Actions for any task can be positive or negative. However, it is not assumed that all negative actions are not the same that is the reason because we distinguish between wrong actions and malicious actions: Positive, i.e. right actions done by the trusted user; Wrong, i.e. bad actions that do not cause any damage or may cause damages done by the innocent user; and Malicious, i.e. harmful actions such as attacks done by the non- trusted user. Accesses to authorized resources and suitable use of them are considered right actions. An entity can make wrong actions by mistake or intentionally, but it is difficult to know. To calculate the action value Va, we take into account the performed action weight, but this value is penalized or rewarded by the past behaviour. This function increases or decreases according to the performed positive and negative actions respectively. The equation is denoted as follows:

$$V_a = (1 - \frac{A_N}{Total_a}).\ W_a^{(m)}$$

Where $0 \leq Va \leq 1$.

In the above equation, represents the past behaviour of any CSU. This value tends to 0 when the behaviour is negative, and it tends to 1 when the behaviour is positive. AN is the number of negative actions and Totala is the total number of performed actions. Wa is the action's weight according to its nature (positive, wrong, and malicious) depending upon the requesting cloud service user and performance of task ($0 \leq wa \leq 1$).

$$(1 - \frac{A_N}{Total_a})$$

Parameter m is the security level, where m>= 1. This security level affects the action weight, for this reason we raise the action weight to the power of (m). The exponential really influences when the actions are wrong. We will show later in following diagrams how the security level affects the action values. When a new action is performed, Va is recalculated, reflecting the present behaviour of the entity. The new trust value will take it into account and modify the current trust value of the service requesting CSU. If it is assumed that,

1. Initially trust value $V_a = 1$
2. w_a for positive action = 1 and for malicious action = 0.8, and
3. Security level m = 1.

Table 4.1: Representative computation of trust value for any CSU

Iteration	Action Behaviour	A_N	$Total_a$	V_a
1	Positive	0	1	1
2	Malicious	1	2	0.4
3	Positive	1	3	0.7

A CSU can perform positive or negative activity in its VM. Depending on its activity trust value in Domain Trust Table (DTT) as well as in User Trust Table (UTT) are updated. In this simulation we have chosen three different types of CSUs: Trusted, Non-Trusted and Innocent. Each type of user has different probability to perform positive activity as,

i. Trusted User has the probability 0.8

ii. Non Trusted User has the probability 0.2

iii. Innocent User has the probability 0.5

In this simulation the following parameters are also assumed for the two layers of the framework. For Domain Layer (Level 1 or Broker Domain)

1. For positive activity W_a =1 & negative activity W_a =0.9
2. Security level m=1
3. Threshold value is 0.1

For User Layer (Level 2, Cloud Service Provider Domain)

1. For positive activity Wa =0.9 & negative activity Wa =0.8
2. Security level m=1
3. Threshold value is 0.2

The above experimental result shows that the probability of doing malicious communication is much less in case of a trusted user than that of innocent and non- trusted users. We simulated several experiments in this test- simulated environment to get the results. It is clear from each simulation that the probability of reaching the threshold value in case of a non- trusted user is much higher than that of any other users. Hence the results proved that trust degree for non- trusted users is increased after performing some trusted tasks and decreases after performing malicious activities. It has also been proved that the trust degree of an innocent cloud service user is much greater after successfully performing some trusted communication with CSP. The trust degree of trusted user decreased after some time and gradually reaches high after performing several trusted communication with the cloud service provider. The non- trusted user reaches to the threshold rapidly. As for example, a trusted user after successfully completion of a given task T the trust degree will be increased and after an unsuccessful communication the trust degree will be decreased accordingly to the performance. Probability function is used to determine the trust degree of any service requesting CSU and then marked them as trusted, innocent or non- trusted. It should be noted that the probability of getting higher trust value by performing a trusted task by a trusted user, is always better than a non- trusted or an innocent user.

5. CONCLUSION

The proposed framework tries to maintain the domain reputation as long as possible by discarding malicious users from the domain reducing the CSP's workload. It also increases some workload of domains and this framework fails to prevent malicious activity without CSP's information. The proposed framework is based on two stage interoperability to secure the cloud data. The strength of proposed algorithm is quite simple than any other security algorithm used in cloud computing for secure and trusted storage. This model guide, how to allow only authorized access to cloud data. It works on the information provided by

user agent and Cloud service provider agent, by this mutual approach it's provide a trusted and secure storage for cloud data storage. Although the framework is dependent on information provided by the agents but this model is the best approach to provide user friendly secure and trusted framework. For more secure and trusted model it is required that the framework should work independently. It would be a new site or direction we have to work to enhance the proposed framework. In future malicious activity identifying approach can be imposed into the proposed framework which in turn makes the system to work independent of CSP's information about malicious activity. This would help to prevent unauthorized accesses to cloud data. Research is currently going on to evaluate the performance of this framework in a real- time environment. The framework may also be extended to eradicate data leakages in a heterogeneous cloud computing platform. Although the framework is dependent on information provided by the agents but this model is the best approach to provide user friendly secure and trusted framework. For more secure and trusted model it is required that the framework should work independently. It would be a new site or direction we have to work to enhance the proposed framework. In future malicious activity identifying approach can be imposed into the proposed framework which in turn makes the system to work independent of CSP's information about malicious activity. This would help to prevent unauthorized accesses to cloud data. Research is currently going on to evaluate the performance of this framework in a real- time environment. The framework may also be extended to eradicate data leakages in a heterogeneous cloud computing platform.

6. REFERENCES

[1] Shantanu Pal, SunirmalKhatua, Nabendu Chaki, SugataSanyal, "A New Trusted and Collaborative Agent Based Approach for Ensuring Cloud Security", Annals of faculty engineering hunedoara – International Journal Of Engineering,2012.

[2] Mell, Peter, and Tim Grance. "The NIST definition of cloud computing." National Institute of Standards and Technology 53.6 (2009): 50.

[3] G. Reese, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, in: Theory in Practice, O'Reilly Media, 2009.

[4] B. Rajkumar, C. Yeo, S. Venugopal, S. Malpani, Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5thutility, Future Generation Computer Systems (2009).

[5] Ramgovind, S., Mariki M. Eloff, and E. Smith. "The management of security in cloud computing." Information Security for South Africa (ISSA), 2010, IEEE, 2010.

[6] Armbrust, Michael, et al. "A view of cloud computing." Communications of the ACM 53.4 (2010): 50-58.

[7] Blaze M, loannidis J, Keromytis A D.: Experience with the KeyNote Trust Management System, Applications and Future Directions[C].iTrust 2003, pp.284-300(2003).

[8] XiangyiMeng, Guangwei Zhang, Jianchu Kang, Hesong Li, Deyi Li.: A New Subjective Trust Model Based on Cloud Model. ICNSC 2008, 5th IEEE International Conference on Networking, Sensing and Control Sanya China April 6-8, 2008.

[9] LI Xiao-Yong, GUI Xiao-Lin.: Research on Dynamic Trust Model for Large Scale Distributed Environment. Journal of Software, vol.18 (6), pp.1510-1521.Beijing (2007).

[10] Li, W., Wang, X., Fu, Y., Fu, Z.: "Study on Several Trust Models in Grid Environment". Journal of Fuzhou University Natural Science Edition 34(2), pp.189-193, 2006.

[11] Youseff, L. Butrico, M. Da Silva, D." Toward a Unified Ontology of Cloud Computing". In: Grid Computing Environments Workshop, 2008. GCE '08, pp. 1-10, November 2008.

[12] RajkumarBuyya, Chee Shin Yeo1and SrikumarVenugopal "Market- Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities".In: HPCC 2008, 10th IEEE International Conference on High Performance Computing and Communications, pp.5-13, 2008.

[13] RajkumarBuyya, Rajiv Ranjan and Rodrigo N. Calheiros "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities". In: Proc. of the 7th High Performance Computing and Simulation Conference (HPCS09), IEEE Computer Society, June 2009.

[14] Cyril Onwubiko, "Security Issues to Cloud Computing", in Cloud Computing: Principles, Systems and Applications, Computer Communications and Networks, N. Antonopoulos and L. Gillam (Eds.), Springer-Verlag London Limited 2010, DOI 10.1007/978-1-84996-241-4_16, pp. 271-288, 2010.

[15] Wenjuan Li and Lingdi Ping, "Trust Model to Enhance Security and Interoperability of Cloud Environment", Proc. of CloudCom 2009, Springer-Verlag Berlin Heidelberg 2009, LNCS 5931, pp. 69–79, 2009.

[16] Jeff Sedayao, Steven Su, Xiaohao Ma, Minghao Jiang, and Kai Miao, "A Simple Technique for Securing Data at Rest Stored in a Computing Cloud", Proc. of CloudCom 2009, Springer-Verlag Berlin Heidelberg 2009, LNCS 5931, pp. 553–558,

[17] Jin-Song Xu, Ru-Cheng Huang, Wan-Ming Huang, and Geng Yang, "Secure Document Service for Cloud Computing", Proc. of CloudCom 2009, Springer-Verlag Berlin Heidelberg 2009, LNCS 5931, pp. 541–546, 2009.

[18] Marios D. Dikaiakos, DimitriosKatsaros, PankajMehra, George Pallis and Athena Vakali, "Cloud Computing: Distributed Internet Computing for IT andScientific Research" in IEEE Internet Computing, IEEE Computer Society, pp. 10-13, September/October 2009, Vol. 13 No. 5, DOI: doi:10.1109/MIC.2009.103.

[19] Char Sample, Senior Scientist, BBN Technologies, Diana Kelley, Partner, Security Curve, "Cloud computing security: Routing and DNS security threats", http://searchsecurity.techtarget.com/.

[20] Siani Pearson, Yun Shen and Miranda Mowbray, "A Privacy Manager for Cloud Computing", in CloudCom 2009, LNCS 5931, M.G. Jaatun, G. Zhao, and C. Rong (Eds.), Springer-Verlag Berlin Heidelberg 2009, pp. 90–106, 2009.

[21] FlorinaAlmenarez, Andres Marın, Celeste Campo and Carlos Garcıa R., "PTM: A Pervasive Trust Management Model for Dynamic Open Environments", Proceedings of First Workshop on Pervasive Security, Privacy and Trust PSPT'04, Boston, MA, USA, 2004.