INT J COMPUT COMMUN, ISSN 1841-9836
8(1):70-78, February, 2013.

Detecting DDoS Attacks in Cloud Computing Environment

A.M. Lonea, D.E. Popescu, H. Tianfield

Alina Madalina Lonea
"Politehnica" University of Timisoara,
Faculty of Automation and Computers
B-dul Vasile Parvan, nr. 2, 300223, Timisoara, Romania
E-mail: madalina _ lonea@yahoo.com

Daniela Elena Popescu
University of Oradea, Faculty of Electrical Eng. and Information Tech.
Universitatii street, nr. 1, 410087, Oradea, Romania
E-mail: depopescu@uoradea.ro

Huaglory Tianfield
School of Engineering and Built Environment,
Glasgow Caledonian University
Cowcaddens Road, Glasgow G4 0BA, United Kingdom
E-mail: h.tianfield@gcu.ac.uk

Abstract:
This paper is focused on detecting and analyzing the Distributed Denial of Service (DDoS) attacks in cloud computing environments. This type of attacks is often the source of cloud services disruptions. Our solution is to combine the evidences obtained from Intrusion Detection Systems (IDSs) deployed in the virtual machines (VMs) of the cloud systems with a data fusion methodology in the front-end. Specifically, when the attacks appear, the VM-based IDS will yield alerts, which will be stored into the Mysql database placed within the Cloud Fusion Unit (CFU) of the front-end server. We propose a quantitative solution for analyzing alerts generated by the IDSs, using the Dempster-Shafer theory (DST) operations in 3-valued logic and the fault-tree analysis (FTA) for the mentioned flooding attacks. At the last step, our solution uses the Dempsters combination rule to fuse evidence from multiple independent sources.

Keywords: cloud computing, cloud security, Distributed Denial of Service (DDoS) attacks, Intrusion Detection Systems, data fusion, Dempster-Shafer theory.

1 Introduction

Cloud computing technology is in continuous development and with numerous challenges regarding security. In this context, one of the main concerns for cloud computing is represented by the trustworthiness of cloud services. This problem requires prompt resolution because otherwise organizations adopting cloud services would be exposed to increased expenditures while at a greater risk. A survey conducted by International Data Corporation (IDC) in August 2008 confirms that security is the major barrier for the cloud users.

There are two things that cloud service providers should guarantee all the time: connectivity and availability, and if there are not met, the entire organizations will suffer high costs [1].

This paper is focused on detecting and analyzing Distributed Denial of Service (DDoS) attacks in cloud computing environment. This type of attacks is often the source of cloud services disruptions. One of the efficient methods for detecting DDoS is to use the Intrusion Detection Systems (IDS), in order to assure usable cloud computing services [2]. However, IDS sensors have the limitations that they yield massive amount of alerts and produce high false positive rates and false negative rates [3].

With regards to these IDS issues, our proposed solution aims to detect and analyze Distributed Denial of Service (DDoS) attacks in cloud computing environments, using Dempster-Shafer Theory (DST) operations in 3-valued logic and Fault-Tree Analysis (FTA) for each VM-based Intrusion Detection System (IDS). The basic idea is to obtain information from multiple sensors, which are deployed and configured in each virtual machine (VM). The obtained information is integrated in a data fusion unit, which takes the alerts from multiple heterogeneous sources and combines them using the Dempster's combination rule. Our approach quantitatively represents the imprecision and efficiently utilizes it in IDS to reduce the false alarm rates.

Specifically, our solution combines the evidences obtained from Intrusion Detection Systems (IDSs) deployed in the virtual machines (VMs) of the cloud system with a data fusion methodology within the front-end.

Our proposed solution can also solve the problem of analysing the logs generated by sensors, which seems to be a big issue [4].

The remainder of this paper is organized as follows: section 2 introduces Dempster-Shafer Theory. Section 3 presents the related work of IDS in Cloud Computing and the related work of IDS using data fusion. Section 4 introduces the proposed solution of detecting DDoS attacks in Cloud Computing. Finally, in section 5 the paper presents the concluding remarks.

2 Dempster-Shafer Theory (DST)

Dempster-Shafer Theory is established by two persons: Arthur Dempster, who introduced it in the 1960's and Glenn Shafer, who developed it in the 1970's [5].

As an extension of Bayesian inference, Dempster-Shafer Theory (DST) of Evidence is a powerful method in statistical inference, diagnostics, risk analysis and decision analysis. While in the Bayesian method probabilities are assigned only for single elements of the state space (Ω),in DST probabilities are assigned on mutually exclusive elements of the power sets of possible states [6], [7].

According to DST method, for a given state space (Ω) the probability (called mass) is allocated for the set of all possible subsets of Ω, namely 2^{Ω} elements.

Consequently, the state space (Ω) is also called *frame of discernment*, whereas the assignment procedure of probabilities is called *basic probability assignment (bpa)* [6], [7], [8].

We will apply the particular case of DST, i.e., the DST operations in 3-valued logic using the fault-tree analysis (FTA), adopted by Guth (1991) and also used in Popescu, et al. (2010).

Thus, if a standard state space Ω is (True, False), then 2^{Ω} should have 4 elements: { ϕ, True, False, (True, False) }. The (True, False) element describes the imprecision component introduced by DST, which refers to the fact of being either true or false, but not both. DST is a useful method for fault-tree analysts in quantitatively representing the imprecision [8]. Another advantage of DST is it can efficiently be utilized in IDS to reduce the false alarm rates by the representation of ignorance [6], [7], [10].

For the reason that in DST the [sum of all masses] = 1 and m(ϕ) = 0,we have the following relation:

$$m(True) + m(False) + m(True, False) = 1 \tag{1}$$

In order to analyze the results of each sensor we'll use the fault tree analysis, which can be realized by boolean OR gate. Table 1 describes the Boolean truth table for the OR gate.

From Table 1 we have:

$$m(A) = (a1, a2, a3) = \{m(T), m(F), m(T, F)\} \tag{2}$$

Table 1: BOOLEAN TRUTH TABLE FOR THE OR GATE

		b1	b2	b3
	∨	T	F	(T,F)
a1	T	T	T	T
a2	F	T	F	(T,F)
a3	(T,F)	T	(T,F)	(T,F)

$$m(B) = (b1, b2, b3) = \{m(T), m(F), m(T, F)\} \tag{3}$$

$$\Rightarrow m(A \vee B) = (a1b1 + a1b2 + a1b3 + a2b1 + a3b1; a2b2; a2b3 + a3b2 + a3b3) \tag{4}$$

$$m(A \vee B) = (a1 + a2b1 + a3b1; a2b2; a2b3 + a3b2 + a3b3) \tag{5}$$

At the last step, our solution applies the Dempster's combination rule, which allows fusing evidences from multiple independent sources using a conjunctive operation (AND) between two bpa's m1 and m2 , called the joint m_{12} [11]:

$$m_{12}(A) = \frac{\sum_{B \bigcap C = A} m_1(B) m_2(C)}{1 - K}, \tag{6}$$

$when : A \neq \phi$
$m_{12}(\phi) = 0$
and $K = \sum_{B \bigcap C = \phi} m_1(B) m_2(C)$

The factor 1-K, called *normalization factor*, is constructive for entirely avoiding the conflict evidence.

Data fusion is also applied in real world examples: robotics, manufacturing, remote sensing and medical diagnosis, as well in military threat assessment and weather forecast systems [12].

Sentz and Ferson (2002) demonstrated in their study that Dempster's combination rule is suitable for the case that the sources of evidences are reliable and a minimal conflict or irrelevant conflict is generated.

3 Related Work

3.1 Intrusion Detection Systems (IDS) in Cloud Computing

One of the IDS strategies proved reliable in cloud computing environments is its applicability to each virtual machine. This is the method we'll choose for our proposed solution. Mazzariello, et al. (2010) presented and evaluated this method in comparison with another IDS deployment strategy, which uses single IDS near the cluster controller. IDS applied to each virtual machine in cloud computing platform eliminates the overloading problem, because in a way the network traffic is split to all IDSs. Thus, applying IDS to each virtual machine gets rid of the issue of the IDS strategy near the cluster controller, which tends to be overloaded because of its necessity to monitor all the supposed traffic from the cloud computing infrastructure. Another advantage of this strategy as described by Roschke, et al. (2009) is the benefit of reducing the impact of the possible attacks by the IDS Sensor VMs.

However, the limitation of IDS strategy applied to each virtual machine is the missing of the correlation phase, which is suggested in the future work by Mazzariello, et al. (2010).

The correlation phase will be included in our proposed solution, because beside the IDS for each virtual machine, our IDS cloud topology will include a Cloud Fusion Unit (CFU) on the front-end, with the purpose of obtaining and controlling the alerts received from the IDS sensor VMs as presented by Roschke, et al. (2009) in their theoretical IDS architecture for cloud, which utilizing an IDS Management Unit.

Compared to Roschke, et al. (2009) who suggested the utilization of IDMEF (Intrusion Detection Message Exchange) standard, a useful component for storage and exchange of the alerts from the management unit, the alerts in our proposed solution will be stored into the Mysql database of Cloud Fusion Unit. The Cloud Fusion Unit will add the capacity to analyze the results using the Dempster-Shafer theory (DST) of evidence in 3-valued logic and the Fault-Tree Analysis for the IDS of each virtual machine and at the end the results of the sensors will be fused using Dempster's combination rule.

A similar method of using a IDS Management Unit is proposed in Dhage, et al. (2011), who presented a theoretical model of an IDS model in cloud computing, by using a single IDS controller, which creates a single mini IDS instance for each user. This IDS instance can be used in multiple Node controllers and a node controller can contain IDS instances of multiple users. The analysis phase of the mini IDS instance for each user takes place in the IDS controller. Compared with Roschke, et al. (2009) where the emphasis is on how to realize the synchronization and integration of the IDS Sensor VMs, in Dhage, et al. (2011) the focus is to provide a clear understanding of the cardinality used in the basic architecture of IDS in cloud infrastructure.

Applying the IDS for each virtual machine is an idea suggested also by Lee, et al. (2011), who increases the effectiveness of IDS by assigning a multi-level intrusion detection system and the log management analysis in cloud computing. In this sense the users will receive appropriate level of security, which will be emphasized on the degree of the IDS applied to the virtual machine, and as well on the prioritization stage of the log analysis documents. This multi-level security model solves the issue of using effective resources.

Lo, et al. (2010) proposed a cooperative IDS system for detecting the DoS attacks in Cloud Computing networks, which has the advantage of preventing the system from single point of failure attack, even if it is a slower IDS solution than a pure Snort based IDS. Thus, the framework proposed by Lo, et al. (2010) is a distributed IDS system, where each IDS is composed of three additional modules: block, communication and cooperation, which are added into the Snort IDS system.

3.2 IDS using Dempster-Shafer theory

Dempster-Shafer Theory (DST) is an effective solution for assessing the likelihood of DDoS attacks, which was demonstrated by several research papers in the context of network intrusion detection systems. Dissanayake (2008) presented a survey upon intrusion detection using DST.

Our study is to detect DDoS attacks in cloud computing environments. Dempster-Shafer Theory (DST) is used to analyze the results received from each sensor (i.e. VM-based IDS).

Data used in experiments using DST vary: Yu and Frincke (2005) used DARPA DDoS intrusion detection evaluation datasets, Chou et al. (2008) used DARPA KDD99 intrusion detection evaluation dataset, Chen and Aickelin (2006) used the Wisconsin Breast cancer dataset and IRIS plant data, while others scientists generated their own data [7]. The data to be used in our proposed solution will be generated by ourselves, by performing DDoS attacks using specific tools against the VM-based IDS.

Siaterlis, et al. (2003) and Siaterlis and Maglaris (2005) performed a similar study of detecting DDoS using data fusion and their field was an operational university campus network, while in our solution the DDoS attacks are proposed to be detected and analyzed in our private cloud computing environment.

Additionally, we consider to analyze the attacks generated against the TCP, UDP, ICMP packets, like Siaterlis, et al. (2003) and Siaterlis and Maglaris (2005). However, instead of applying DST on the state space $\Omega = \{Normal, UDP - flood, SYN - flood, ICMP - flood\}$, our study uses DST operations in 3-valued logic as suggested by Guth (1991) for the same flooding attacks: TCP-flood, UDP-flood, ICMP-flood, for each VM-based IDS. Like Siaterlis and Maglaris (2005), Chatzigiannakis, et al., (2007) chosen the same frame of discernment, while Hu, et al. (2006) used a state space: {Normal, TCP, UDP and ICMP}.

Furthermore, compared with the study performed by Siaterlis, at al. (2003) and Siaterlis and Maglaris (2005), who use a minimal neural network at the sensor level, our proposed solution will assign the probabilities using: DST in 3-valued logic, the pseudocode and the fault tree analysis.

Whilst the computational complexity of DST is increasing exponentially with the number of elements in the frame of discernment [12], the DST 3-valued logic proposed to be used in our research will not encounter this issue, which will meet the efficiency requirements in terms of both detection rate and computation time [15].

Finally, the data fusion of the evidences obtained from sensors studied by Siaterlis and Maglaris (2005) will be used in our study. The data fusion will be realized using the Dempster-Shafer combination rule, which was demonstrated in Siaterlis and Maglaris (2005) for its advantages, i.e., maximization of DDoS true positive rates and minimization of the false positive alarm rate, by combining the evidence received from sensors. Therefore, the work of cloud administrators will be alleviated, whereas the number of alerts will decrease.

4 Proposed Solution

In order to detect and analyze Distributed Denial of Service (DDoS) attacks in cloud computing environments we propose a solution as presented in Figure 1. For illustration purpose, a private cloud with a front-end and three nodes is set up. Whilst the detection stage is executed within the nodes, more precisely inside the virtual machines (VMs), where the Intrusion Detection Systems (IDSs) are installed and configured; the attacks assessment phase is handled inside the front-end server, in the Cloud Fusion Unit (CFU).

The first step in our solution includes the deployment stage of a private cloud using Eucalyptus open-source version 2.0.3. The topology of the implemented private cloud is: a front-end (with Cloud Controller, Walrus, Cluster Controller, Storage Controller) and a back-end (i.e. three nodes). The Managed networking mode is chosen because of the advanced features that it provides and Xen hypervisor is used for virtualization.

Then, the VM-based IDS are created, by installing and configuring Snort into each VM. The reason of using this IDS location is because the overloading problems can be avoided and the impact of possible attacks can be reduced [2], [13].

These IDSs will yield alerts, which will be stored into the Mysql database placed within the Cloud Fusion Unit (CFU) of the front-end server. A single database is suggested to be used in order to reduce the risk of losing data, to maximize the resource usage inside the VMs and to simplify the work of cloud administrator, who will have all the alerts situated in the same place. A similar idea of obtaining and controlling the alerts received from the IDS Sensor VMs using an IDS Management Unit was presented by Roschke, et al. (2009) as a theoretical IDS architecture for cloud. A similar method of using an IDS Management Unit is proposed in Dhage, et al. (2011). However, our solution adds the capacity to analyse the results using the Dempster-Shafer theory of evidence in 3-valued logic.

As showed in Figure 1, the Cloud Fusion Unit (CFU) comprises 3 components: Mysql database, bpas calculation and attacks assessment.

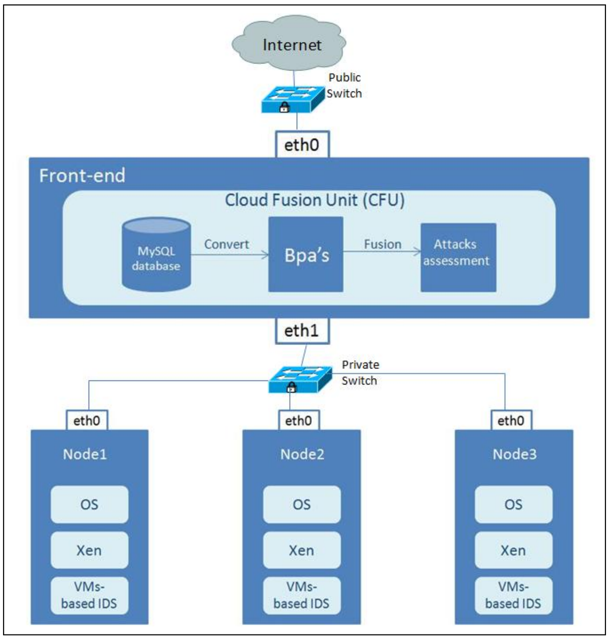

Figure 1: IDS Cloud Topology

I. Mysql database

The Mysql database is introduced with the purpose of storing the alerts received from the VM-based IDS. Furthermore, these alerts will be converted into Basic Probabilities Assignments (bpas), which will be calculated using the pseudocode below.

II. Basic probabilities assignment (bpa's) calculation

For calculating the basic probabilities assignment, first we decide on the state space Ω. In this paper we use DST operations in 3-valued logic {True, False, (True, False)} Guth (1991) for the following flooding attacks: TCP-flood, UDP-flood, ICMP-flood, for each VM-based IDS. Thus, the analyzed packets will be: TCP, UDP and ICMP. Further, a pseudocode for converting the alerts received from the VM-based IDS into bpas is provided. The purpose of this pseudocode is to obtain the following probabilities of the alerts received from each VM-based IDS:

$(m_{UDP}(T), m_{UDP}(F), m_{UDP}(T, F))$

$(m_{TCP}(T), m_{TCP}(F), m_{TCP}(T, F))$

$(m_{ICMP}(T), m_{ICMP}(F), m_{ICMP}(T, F))$

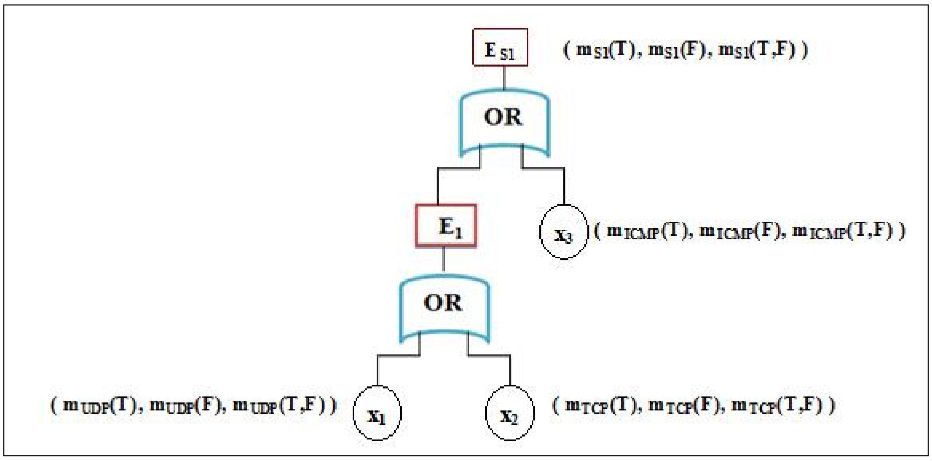
E S1
(mS1(T), mS1(F), mS1(T,F))
OR
E1
x3
(mICMP(T), mICMP(F), mICMP(T,F))
OR
(mUDP(T), mUDP(F), mUDP(T,F))
x1
x2
(mTCP(T), mTCP(F), mTCP(T,F))

Figure 2: BPA's calculation

Pseudocode for converting the alerts into bpa's:

```
For each node
Begin
    For each X ∈ {UDP; TCP; ICMP}:
    Begin
        1: Query the alerts from the database when a X attack occurs for the specified hostname
        2: Query the total number of possible X alerts for each hostname
        3: Query the alerts from the database when X attack is unknown
        4: Calculate the Belief (True) for X, by dividing the result obtained at step 1 with the result obtained at step 2
        5: Calculate the Belief (True, False) for X, by dividing the result obtained at step 3 with the result obtained at step 2
        6: Calculates Belief (False) for X: 1- Belief (True) - Belief (True, False)
    end
end
```

Furthermore, after obtaining the probabilities for each attack packet (i.e. UDP, TCP, ICMP) for each VM-based IDS, the probabilities for each VM-based IDS should be calculated following the fault-tree as shows in Figure 2. Figure 2 reveals only the calculation of the probabilities (i.e. $m_{S1}(T), m_{S1}(F), m_{S1}(T, F)$) for the first VM-based IDS.

Thus, using the DST with fault-tree analysis we can calculate the belief (Bel) and plausibility (Pl) values for each VM-based IDS:

$$Bel(S1) = m_{S1}(T) \tag{7}$$

$$Pl(S1) = m_{S1}(T) + m_{S1}(T, F) \tag{8}$$

III. Attacks assessment

The attacks assessment consists of data fusion of the evidences obtained from sensors by using the Dempster's combination rule, with the purpose of maximizing the DDoS true positive rates and minimizing the false positive alarm rate. $m_{S1,S2}(T)$ can be calculated using Table 2 and equation (6).

Table 2: BOOLEAN TRUTH TABLE FOR THE OR GATE

	m_{S1}(T)	m_{S1}(F)	m_{S1}(T,F)
m_{S2}(T)	m_{S1}(T) m_{S2}(T)	m_{S1}(F) m_{S2}(T)	m_{S1}(T,F) m_{S2}(T)
m_{S2}(F)	m_{S1}(T) m_{S2}(F)	m_{S1}(F) m_{S2}(F)	m_{S1}(T,F) m_{S2}(F)
m_{S2}(T,F)	m_{S1}(T) m_{S2}(T,F)	m_{S1}(F) m_{S2}(T,F)	m_{S1}(T,F) m_{S2}(T,F)

5 Conclusions

To detect and analyze Distributed Denial of Service (DDoS) attacks in cloud computing environments we have proposed a solution using Dempster-Shafer Theory (DST) operations in 3-valued logic and the Fault-Tree Analysis (FTA) for each VM-based Intrusion Detection System (IDS). Our solution quantitatively represents the imprecision and efficiently utilizes it in IDS to reduce the false alarm rates by the representation of the ignorance.

Whilst the computational complexity of DST is increasing exponentially with the number of elements in the frame of discernment [12], the DST 3-valued logic in our solution does not have this issue, which meets the efficiency requirements in terms of both detection rate and computation time. At the same time, the usability requirement has been accomplished, because the work of cloud administrators will be alleviated by using the Dempster rule of evidence combination whereas the number of alerts will decrease and the conflict generated by the combination of information provided by multiple sensors is entirely eliminated.

To sum up, by using DST our proposed solution has the following advantages: to accommodate the uncertain state, to reduce the false negative rates, to increase the detection rate, to resolve the conflicts generated by the combination of information provided by multiple sensors and to alleviate the work for cloud administrators.

Acknowledgment
This work was partially supported by the strategic grant POSDRU/88/1.5/S/50783, Project ID50783 (2009), co-financed by the European Social Fund - Investing in People, within the Sectoral Operational Programme Human Resources Development 2007-2013.

Bibliography

[1] Perry, G., *Minimizing public cloud disruptions*, TechTarget, [online]. Available at: http://searchdatacenter.techtarget.com/tip/Minimizing-public-cloud-disruptions, 2011.

[2] Roschke, S., Cheng, F. and Meinel, C.,*Intrusion Detection in the Cloud.* In Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, pp. 729-734, 2009.

[3] Yu, D. and Frincke, D.,*A Novel Framework for Alert Correlation and Understanding.* International Conference on Applied Cryptography and Network Security (ACNS) 2004, Springer's LNCS series, 3089, pp. 452-466, 2004.

[4] Lee, J-H., Park, M-W., Eom, J-H. And Chung, T-M., *Multi-level Intrusion Detection System and Log Management in Cloud Computing.* In 13th International Conference on Advanced Communication Technology (ICACT) ICACT 2011, Seoul, 13- 16 February, pp.552- 555, 2011.

[5] Chen, Q. and Aickelin, U., Dempster-Shafer for Anomaly Detection. *In Proceedings of the International Conference on Data Mining (DMIN 2006)*, Las Vegas, USA, pp. 232-238, 2006.

[6] Siaterlis, C., Maglaris, B. and Roris, P., *A novel approach for a Distributed Denial of Service Detection Engine*. National Technical University of Athens. Athens, Greece, 2003.

[7] Siaterlis, C. And Maglaris, B., One step ahead to Multisensor Data Fusion for DDoS Detection. *Journal of Computer Security*, 13(5):779-806, 2005.

[8] Guth, M.A.S., *A Probabilistic Foundation for Vagueness & Imprecision in Fault-Tree Analysis*. IEEE Transactions on Reliability, 40(5), pp.563-569, 1991.

[9] Popescu D.E., Lonea A.M., Zmaranda D.,Vancea C. and Tiurbe C. , Some Aspects about Vagueness & Imprecision in Computer Network Fault-Tree Analysis. *INT J COMPUT COMMUN*, ISSN: 1841-9836, 5(4):558-566, 2010.

[10] Esmaili, M., *Dempster-Shafer Theory and Network Intrusion Detection Systems*. Scientia Iranica, Vol. 3, No. 4, Sharif University of Technology, 1997.

[11] Sentz, K. and Ferson, S., *Combination of Evidence in Dempster-Shafer Theory*. Sandia National Laboratories, Sandia Report, 2002.

[12] Dissanayake, A., *Intrusion Detection Using the Dempster-Shafer Theory*. 60-510 Literature Review and Survey, School of Computer Science, University of Windsor, 2008.

[13] Mazzariello, C., Bifulco, R. and Canonico, R., *Integrating a Network IDS into an Open Source Cloud Computing Environment*. In Sixth International Conference on Information Assurance and Security, pp. 265-270, 2010.

[14] Dhage, S. N., et al., *Intrusion Detection System in Cloud Computing Environment*. In International Conference and Workshop on Emerging Trends in Technology (ICWET 2011) ' TCET, Mumbai, India, pp. 235-239, 2011.

[15] Lo, C-C. , Huang, C-C. And Ku, J., *A Cooperative Intrusion Detection System Framework for Cloud Computing Networks*. In 39th International Conference on Parallel Processing Workshops, pp.280-284, 2010.

[16] Yu, D. and Frincke, D., Alert Confidence Fusion in Intrusion Detection Systems with Extended Dempster-Shafer Theory. *ACM-SE 43: Proceedings of the 43rd ACM Southeast Conference*, pp. 142-147, 2005.

[17] Chou, T., Yen, K.K., Luo, J., Network intrusion detection design using feature selection of soft computing paradigms. *International Journal of Computational Intelligence*, 4(3):102-105, 2008.

[18] Chatzigiannakis, V., et al., *Data fusion algorithms for network anomaly detection: classification and evaluation*. Proceedings of the Third International Conference on Networking and Services (ICNS'07), 2007.

[19] Hu, W., Li, J. and Gao, Q., *Intrusion Detection Engine Based on Dempster-Shafer's Theory of Evidence*. Communications, Circuits and Systems Proceedings, 2006 International Conference, 3:1627-1631, 2006.