Data fusion algorithms for network anomaly detection: classification and evaluation

V. Chatzigiannakis, G. Androulidakis, K. Pelechrinis, S. Papavassiliou and V. Maglaris
Network Management & Optimal Design Laboratory (NETMODE),
School of Electrical & Computer Engineering
National Technical University of Athens (NTUA)
{vhatzi, gandr, kpele, papavass, maglaris}@netmode.ntua.gr

Abstract

In this paper, the problem of discovering anomalies in a large-scale network based on the data fusion of heterogeneous monitors is considered. We present a classification of anomaly detection algorithms based on data fusion, and motivated by this classification, the operational principles and characteristics of two different representative approaches, one based on the Demster-Shafer Theory of Evidence and one based on Principal Component Analysis, are described. The detection effectiveness of these strategies are evaluated and compared under different attack scenarios, based on both real data and simulations. Our study and corresponding numerical results revealed that in principle the conditions under which they operate efficiently are complementary, and therefore could be used effectively in an integrated way to detect a wider range of attacks..

1. Introduction

One of the main challenges in security management of large scale high speed networks is the detection of suspicious anomalies in network traffic patterns due to Distributed Denial of Service (DDoS) attacks or worm propagation [1] [2]. Network anomaly detection is one of the most frequently suggested methods for detecting network abuse. Anomaly detection can be uniformly applied in order to detect network attacks, even in cases where novel attacks are present and the nature of the intrusion is unknown [3]. Usually network anomaly detection methodologies rely on the analysis of network traffic and the characterization of the dynamic statistical properties of traffic normality, in order to accurately and timely detect network anomalies. Anomaly detection is based on the concept that perturbations of normal behavior suggest the presence of anomalies, faults, attacks, etc.

The goal of this paper is twofold: firstly, it provides a review and classification of data fusion algorithms inspired from the taxonomy presented in [4] but addressing specifically the problem of anomaly detection; and secondly, it focuses on the study and evaluation of two representative anomaly detection techniques, one based on the Dempster-Shafer theory of evidence and one based on Principal Component Analysis (PCA). Among the main objectives of this work is not only to evaluate the detection effectiveness of each one of these methodologies, but also to identify and study the conditions under which they operate efficiently.

The remaining of this paper is organized as follows. In section 2 we present a classification of some widely used anomaly detection approaches. Then, in section 3 we present the operational principles of two different representative data fusion algorithms, while in section 4 their performances under different attack scenarios are evaluated and compared based on real experiments and simulations. Finally section 5 concludes the paper.

2. Data Fusion Algorithm Classification

Multisensor data fusion, or distributed sensing, is a relatively new engineering discipline used to combine data from multiple and diverse sensors and sources in order to make inferences about events, activities, and situations [5]. These systems are often compared to the human cognitive process where the brain fuses sensory information from the various sensory organs, evaluates situations, makes decisions, and directs action. Among the most common examples where such systems have been developed and widely used, are military systems for threat assessment and weather forecast systems. Generally, data fusion is a process performed on multi-source data towards detection, association, correlation, estimation and combination of several data streams

Third International Conference on Networking and Services(ICNS'07)
0-7695-2858-9/07 $20.00 © 2007 IEEE

into one with a higher level of abstraction and greater meaningfulness.
In the following we present a classification and brief description of some widely used methods, motivated by the taxonomy that was originally proposed by Hall [4]. However our presentation and arguments are specifically targeted towards anomaly detection.

2.1. Physical Models

Physical models attempt to create an accurate model of the observed environment and make appropriate estimations, by matching predicted (modeled) data to actual observations. Included in this category are also methods that try to decompose the observed object (the network or a network element, such as a link) in descriptive components (or "primitives"). Such a method is M^3L [6] (described in more detail in section 3.1) that relies on PCA approach to decompose the network state in primitives (i.e. Principal Components) that capture the important interrelations and traffic patterns among network elements and therefore create a model of the monitored network

2.2. Parametric Classification

The algorithms that belong to this category make a direct mapping of parametric data to the classification space (e.g. the state of the system). These may be further divided into statistically based algorithms, such as Bayesian Inference and/or the Dempster-Shafer (D-S) methodologies, and information theoretic techniques such as neural networks and entropy based methods.
Bayesian Inference computes the probability of an observation given the assumption of an a priori hypothesis. Dempster-Shafer Theory of Evidence is a mathematical theory of evidence [7] based on belief functions and plausible reasoning, which is used to combine separate pieces of information (evidence) to calculate the probability of an event. In [8], D-S has been thoroughly tested for anomaly detection in an operational university campus network.

Adaptive Neural Networks provide an interesting and generic method that does not assume a model for the observed system, but bases its output on the successful training of its nodes (neurons) using training data. The different kinds of neural networks differ in the number of nodes and layers used, as well as the processing function that is performed in each node. These methods have been used in the context of Intrusion Detection Systems but require training data that are representative of the normal traffic data, which in general are quite hard to gather or generate [9].

Finally, entropy based methods use the concept of information entropy to describe the inherent randomness of a communication system. The entropy measure reflects and quantifies the information in a generalized "message" on the basis of its probability of occurrence. The basic idea is that frequent "messages" are of low entropy value and rare messages have greater value. In [10] the authors have developed an entropy-based approach that determines and reports entropy contents of traffic parameters such as IP addresses. Changes in the entropy content indicate a massive network event.

2.3. Cognitive Algorithms

Members of the third category, namely the cognitive based algorithms, try to mimic the human brain cognitive process for object identification. Two representative approaches that belong to this class are: expert systems and techniques based on fuzzy set theory. Expert systems consist of a knowledge base that represents the knowledge of some "field expert" usually in a production rule form. This knowledge can be facts, algorithms, heuristics etc. Expert systems have been widely used for Intrusion Detection purposes. For example, NIDES [11] has a rule database that employs expert rules to characterize known intrusive activity represented in activity logs, and raises alarms as matches are identified between the observed activity logs and the rule encodings. Fuzzy set theory is the fundamental theory that supports fuzzy logic, which is in turn used as an alternative to logical reasoning. In fuzzy logic, a statement is not just true or false but is rather a proposition with an associated value between 0, that represents a completely false proposition, and 1 - completely true (this is the membership value to the truthfulness set) [12].

3. Representative algorithms description

3.1. M^3L: a network-wide anomaly detection PCA-based approach

The objective of Multi-Metric-Multi-Link PCA-based method [6] is to provide a methodology of fusing and combining data of heterogeneous monitors spread throughout the network. This is achieved by applying a PCA-based approach simultaneously on several metrics of one or more links.

Principal Component Analysis aims at the reduction of the dimensionality of a data set in which there are a large number of interrelated variables, while retaining as much as possible of the variation present in the data set [13]. The extracted non-correlated components are called Principal Components (PCs) and are estimated from the eigenvectors of the covariance matrix or the correlation matrix of the original variables.

The overall procedure of this method may be divided into two different parts: the offline analysis, that creates a model of the normal traffic, and the real time analysis that detects anomalies by comparing the current (actual) with the modeled traffic patterns. The input of the offline analysis is a data set that contains only normal traffic. During the offline analysis, PCA is applied on this data set and then the first few most important derived Principal Components (PCs) are selected. Their number depends on the network and the number of metrics per link, and it represents the number of PCs required for capturing the percentage of variance that the system needs to model normal traffic. The output of the offline analysis is the PCs to be used in the Subspace Method.

The goal of the Subspace Method is to divide current traffic data in two different spaces: one containing traffic considered normal (y_{norm}) and resembles to the modeled traffic patterns and one containing the residual (y_{res}). In general, anomalies tend to result in great variations in the residual, since they present different characteristics from the modeled traffic. When an anomaly occurs, the residual vector presents great variation in some of its variables and the system detects the network path containing the anomaly by selecting these variables. The interested reader may refer to [6] for a more detailed description of PCA-based anomaly detection strategies.

3.2. D-S based anomaly detection

Dempster-Shafer's Theory of Evidence can be considered an extension of Bayesian inference. The goal of D-S is to infer the true system state without having an explicit model of the system, based only on some observations *that* can be considered as hints (with some uncertainty) towards some system states. Based on these observations D-S calculates two functions: Belief *Bel(H)* and Plausibility *Pl(H),* where *H* is the hypothesis for the current state. Generally we can characterize *Bel(H)* as a quantitative measure of all our supportive evidence and *Pl(H)* as a measure of how compatible our evidence is with *H* in terms of doubt. The true belief in the hypothesis for the current system state lies in the interval between. Our degree of ignorance is represented by the difference *Bel(H) - Pl(H)*. Theory of Evidence makes the distinction between uncertainty and ignorance, so it is a very useful way to reason with uncertainty based on incomplete and possibly contradictory information extracted from a stochastic environment. It does not need ”a priori” knowledge or probability distributions on the possible system states like the Bayesian approach and as such it is mostly useful when we do not have a model of our system. Theory of Evidence has a definite advantage in a vague and unknown environment especially when compared to other inference processes like first order logic that assumes complete and consistent knowledge exhibits monotonicity, or probability theory that requires knowledge in terms of probability distributions and exhibits non-monotonicity. The main disadvantage of Dempster-Shafer's theory is the assumption that the evidence is statistically independent from each other, since sources of information are often linked with some sort of dependence. The interested reader may refer to [8] for a detailed discussion about the application of D-S theory in network anomaly detection.

4. Performance Evaluation

4.1. Network Topology and experiments

In this section the performances of the two representative anomaly detection techniques - D-S and M^{3}L techniques - described in section 3, are evaluated and compared under various attack scenarios. The results and corresponding observations presented in this section are based on real data collected from an operational campus network. Specifically, we monitored the link between the National Technical University of Athens (NTUA) and the Greek Research and Technology Network (GRNET), which connects the university campus with the Internet. This link has an average traffic of 700-800Mbit/sec. It contains a rich network traffic mix, that carries standard network services like web, mail, ftp and p2p application traffic.

In our study, in order to evaluate the D-S algorithm we defined four possible states for the network: NORMAL, SYN-attack, ICMP-flood and UDP-flood. For the application of the D-S algorithm we used the following metrics: UDP packets in/out ratio, ICMP packets out/in ratio, TCP-SYN in/TCP-FIN out ratio. In order to transform the sensor measurements (metrics) to basic probability assignments (bpa) we used multiple thresholds per sensor measurement that were set manually after studying the “normal” data set.

In order to evaluate the PCA-based approach we implemented a single-link-multi-metric algorithm based on M^{3}L and used the following metrics: number of UDP packets in, UDP packets out, ICMP packets out, ICMP packets in, TCP-SYN packets, TCP-FIN packets, TCP packets out, TCP packets in, TCP flows out, TCP flows in. The sample dataset required to train the system and create the network model was a part of the recorded traffic that was relatively flat and considered to be normal.

SYN-attack was performed using a real DoS attack tool. The target of the attack was a host situated in the NTUA network at a 10 Mbps link and there were 3 attackers distributed at GRNET. Every one of the attackers was connected at a 100Mbps interface and was running the TFN2K tool that is used for DoS attacks. These attackers were sending TCP SYN packets towards the victim, using spoofed IP addresses from the C class network that they were part of. In that manner the three attackers managed an attack from 256 sources. The trace file of the attack lasts 8 minutes with the attack lasting for 60 seconds.

In the next section we provide some representative numerical results of our experiments, starting with the performance of each algorithm under various scenarios, and then conclude our experimental results by comparing their performance under common experiments. In the following experiments, ICMP-flood and UDP-flood attacks were injected manually in the network traces of the collected data

4.2. Performance Evaluation

4.2.1. D-S algorithm Detection Effectiveness

In Figure 1, an ICMP-flood attack, as detected by the D-S algorithm, is presented. In this scenario, the attack packets correspond to 5% of the background traffic. The four different diagrams correspond to each one of the four defined states (NORMAL, UDP-flood, ICMP-flood and SYN-ATTACK). As observed by this figure, during the attack, the belief and plausibility functions of ICMP-flood state have increased – together with the decrease of the respective functions for the NORMAL state - in a way that implies that the most likely state of our network is ICMP-flood.

In Figure 2 we present the corresponding results of a real SYN-attack scenario. In this case, the attack packets represent only 2% of the background traffic. As we can observe from the four diagrams given for the four possible states of the network, the belief and plausibility functions of SYN-attack state have not increased during this attack. Therefore based on the D-S algorithm we erroneously conclude that the network was always in NORMAL state.

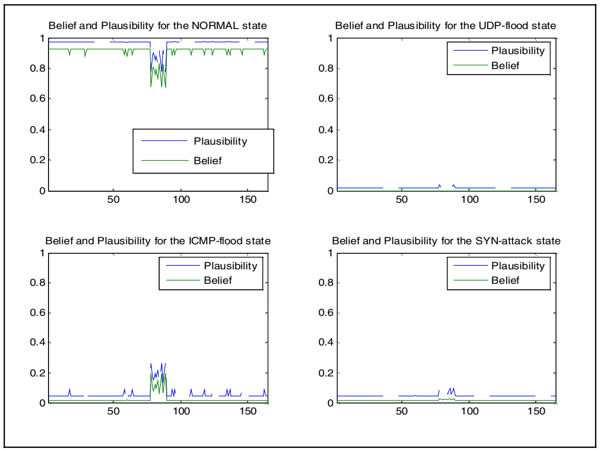

Figure 1. ICMP-flood of 5% rate detected by D-S algorithm

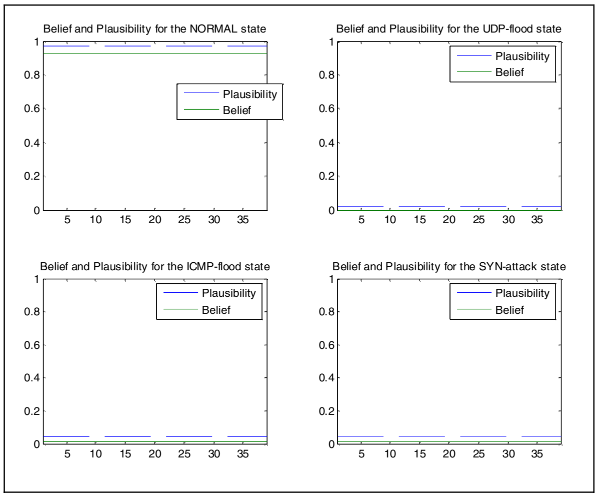

Figure 2. SYN-attack of 2% rate not-detected by D-S algorithm

In Figure 3 we depict the corresponding results for the D-S algorithm using a 20% SYN-attack rate. As we can observe there is a noticeable alteration of the belief and plausibility functions of the NORMAL and SYN-attack state, which increases our belief that the network is in SYN-attack state.

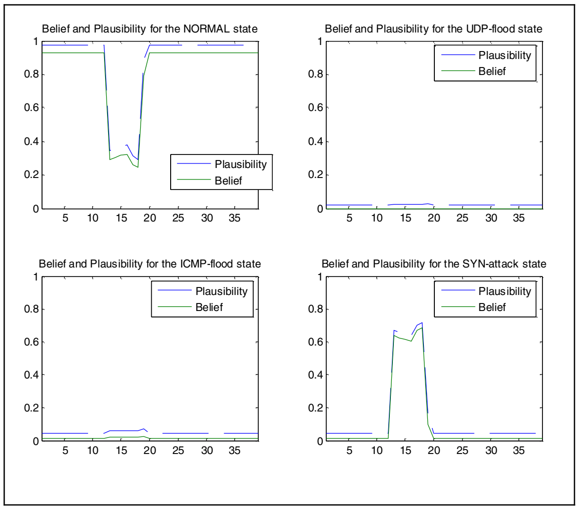

Figure 3. SYN-attack of 20% rate detected by D-S algorithm (real attack)

4.2.2. M^{3}L algorithm Detection Effectiveness

In the following, the detection effectiveness of the M^{3}L algorithm is evaluated. As observed by the results presented in the following figures, the behavior of the PCA algorithm differs significantly from the one of the D-S algorithm. In Figure 4 we present the corresponding Squared Prediction Error (SPE) for a simulated ICMP – flood attack. The attack packets correspond to 20% of the total background traffic.

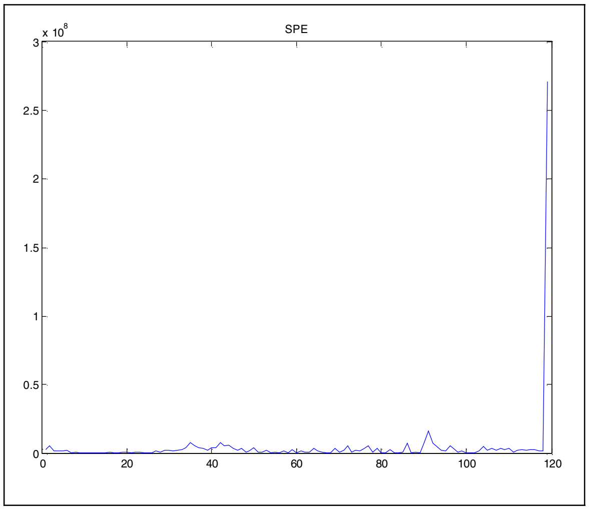

Figure 4. ICMP-flood attack of rate 20% not detected by PCA algorithm

As we can observe from figure 4, there is not any significant change at the SPE, and as a result one can imply that the network was always at the NORMAL state. In this case M^{3}L fails to detect the attack because the selection of metrics is inappropriate, namely the metrics utilized are uncorrelated and thus the algorithm cannot create a precise model of the network.

On the other hand, figure 5 presents the SPE for a number of different rates of SYN attack. In this figure we present various volumes of the attack, ranging from 1% attack rate up to 20%. We observe that even for a 2% attack rate the SPE changes significantly compared to the SPE for the NORMAL state of the network.

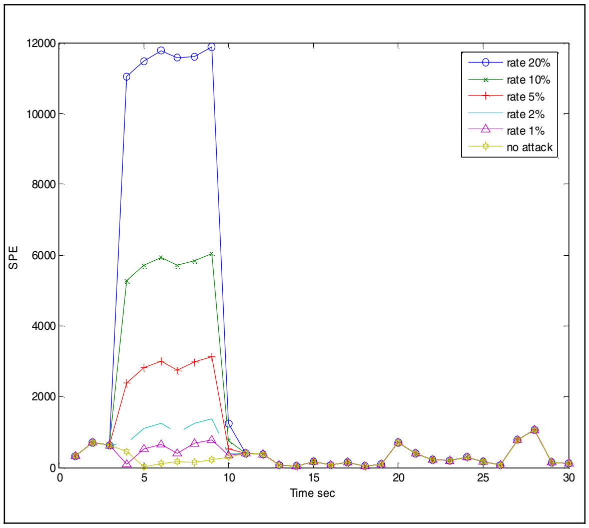

Figure 5. The SPE of the PCA algorithm for various rates of the SYN-attack

4.2.3. Comparative Results

In the following figures we present in common axes the discrete differential of the “alarm” function of each algorithm for the same attack. Along with D-S and M^{3}L we study the performance of another parametric classification algorithm: the Bayesian inference. The Bayesian inference simply utilizes a function for each one of the four possible states of the network that estimates the probability of the system being in each state. Referring to Figures 6 & 7 for the Bayesian Inference the “alarm” function is the probability function of the corresponding state, while for the D-S algorithm is either the belief function or the plausibility function, and for the PCA based algorithm (M^{3}L) is the SPE function.

More specifically, in Figure 6 we present the corresponding results for a simulated ICMP-flood, where the attack packets correspond to 10% of the background traffic. The attack was manually inserted in the corresponding traffic dump, starting at time bin 75 and ending at time bin 90 sec. In the differential diagram the large positive values indicate a large increase whereas the negative values indicate respective decrease. The change rate is significantly large for the parametric classification algorithms – Bayesian inference and DS theory of evidence. On the other hand M^3L fails to detect the attack and presents false positives between time bins 120 and 140.

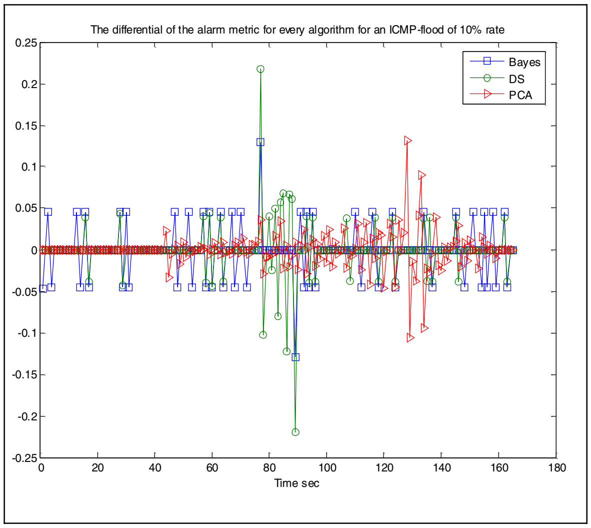

Figure 6. The deferential of the alarm metric of every algorithm for the same simulated ICMP-flood of 10% attack rate

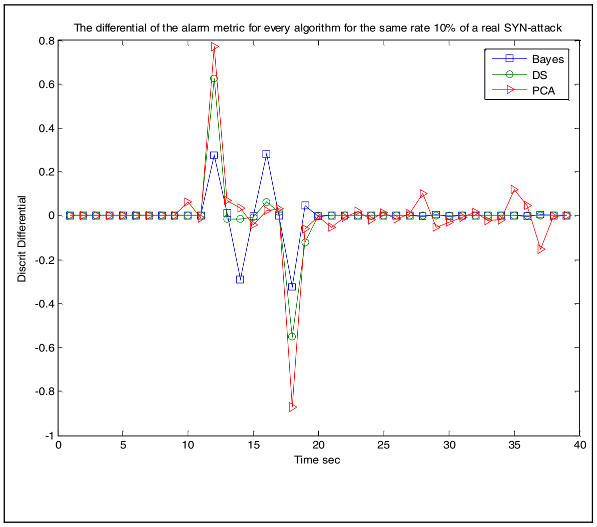

Figure 7. The deferential of the alarm metric of every algorithm for the real SYN-attack of 10% rate

The situation however is different in Figure 7, where we present the corresponding analysis for a 10% rate SYN-attack. The corresponding results verify that the PCA based algorithm is much more sensitive at the detection of such attacks. The attack was emulated according to the scenario described in section 4, starting at time bin 12 and ending at time bin 18 sec. Also in this differential diagram the large positive values indicate a large increase and the peaks at time bin 12 reveal the beginning of the attack whereas the negative peaks in time bin 18 denote its end.

4.2.4. Metric Correlation and Discussion

The explanation of the difference in the performance of the algorithms lies in the correlation of the metrics used. The D-S Theory of Evidence performs well on the detection of attacks that can be sensed by uncorrelated metrics because it requires that evidence originating from different sensors is independent.

On the other hand, M^3L requires that the metrics fed into the fusion algorithm present some degree of correlation. The method models traffic patterns and interrelations by extracting the eigenvectors from the correlation matrix of a sample data set. If there is no correlation among the utilized metrics then the model is not efficient. The test for determining whether or not two sets of series are correlated is to calculate their correlation coefficient $R_{X,Y}$. Variables with correlation coefficient close to 1 vary together in the same direction; whereas variables with correlation close to -1 vary together in opposite directions.

$$R_{X,Y} = \frac{Cov(X,Y)}{S_X \cdot S_Y}$$

In our experiments, based on data gathered by GRNET, we have confirmed that neighboring virtual links are highly correlated, as their correlation matrix comprises of elements that have value close to 1.

Metrics such as TCP SYN packets, TCP FIN packets, TCP in flows and TCP out flows are highly correlated and should be utilized in M^3L, whereas the combination of UDP in/out packets, ICMP in/out packets, TCP in/out packets are uncorrelated and should be used in D-S. This can be further analyzed and mapped to the detection capabilities of these methodologies with respect to different attack types. For instance, attacks that involve alteration in the percentage of UDP packets in traffic composition such as UDP flooding are better detected by D-S method. On the other hand, attacks such as SYN attacks, worms spreading, port scanning which affect the proportion of correlated metrics such as TCP in/out, SYN/FIN packets and TCP in/out flows are better detected with M^{3}L.

5. Conclusions

With the advent and explosive growth of the global Internet and the electronic commerce infrastructures, timely and proactive detection of network anomalies is a prerequisite for the operational and functional effectiveness of secure wide area networks. If the next generation of network technology is to operate beyond the levels of current networks, it will require a set of well-designed tools for its management that will provide the capability of dynamically and reliably identifying network anomalies.

In this paper, we studied the problem of discovering anomalies in a large-scale network based on the data fusion of heterogeneous monitors. We first presented and discussed taxonomy of anomaly detection algorithms based on the data fusion aspect. Moreover, we focused on the study of two different representative anomaly detection techniques, one based on the Demster-Shafer Theory of Evidence and one based on Principal Component Analysis. The two techniques that belong to different categories of data fusion algorithms were evaluated via emulation and simulation. Our study and corresponding numerical results revealed that in principle the conditions under which they operate efficiently are complementary, and therefore could be used effectively in an integrated way to detect a wide range of possible attacks.

6. Acknowledgements

This work was partially supported by the European Commission, GridCC Project (IST 511382).

7. References

[1] Christos Douligeris, Aikaterini Mitrokotsa, "DDoS attacks and defense mechanisms: classification and state-of-the-art", Computer Networks: The International Journal of Computer and Telecommunications Networking, Vol. 44, Issue 5 , pp: 643 - 666, 2004

[2] Z. Chen, L. Gao, K. Kwiat, "Modeling the spread of active worms", Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), Vol. 3, pp. 1890 – 1900, 2003.

[3] W. Lee and D. Xiang, "Information-Theoretic Measures for Anomaly Detection", In Proc. of the IEEE Symposium on Security and Privacy (S&P 2001), pp. 130 -143, 2001.

[4] D. Hall. Mathematical Techniques in Multisensor Data Fusion. Artech House, Norwood, Massachussets, 1992.

[5] Edward Waltz, James Llinas, "Multisensor Data Fusion", Artech House Boston, London, 1990.

[6] V. Chatzigiannakis, S. Papavassiliou, G. Androulidakis, B. Maglaris, "On the realization of a generalized data fusion and network anomaly detection framework", Fifth International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP'06), Patra, Greece, July 2006.

[7] Glenn Shafer. A Mathematical Theory of Evidence. Princeton University Press, Princeton, 1976.

[8] C.Siaterlis, B. Maglaris, "One step ahead to multisensor data fusion for DDoS detection", Journal of Computer Security, pp. 779 - 806 , Vol. 13, Issue 5, 2005.

[9] T. Brotherton, T. Johnson, "Anomaly detection for advanced military aircraft using neural networks", in Proc of the IEEE Aerospace Conference, Vol.6, pp.3113-3123, 2001

[10] A. Lakhina, M. Crovella, and C. Diot, "Mining Anomalies Using Traffic Feature Distributions", in Proc. of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM), pp. 217 - 228 , 2005.

[11] http://www.sdl.sri.com/projects/nides/

[12] J. Gomez, F. Gonzalez, D. Dasgupta, "An immuno-fuzzy approach to anomaly detection", In Proc. of The 12th IEEE International Conference on Fuzzy Systems, Vol. 2, pp. 1219- 1224, 2003.

[13] I.T. Jollife. "Principal Component Analysis", Second Edition, Springer, 2002

[14] J. E. Jackson and G. S. Mudholkar, "Control Procedures for Residuals Associated with Principal Component Analysis", Technometrics, pp. 341–349, 1979.

[15] A. Papoulis, "Probability, random variables and stochastic processes", McGraw-Hill, 1984, 2nd edition.