Using Firefly and Genetic Metaheuristics for Anomaly Detection based on Network Flows

Fadir Salmen, Paulo R. Galego Hernandes Jr.

Security Information Department
São Paulo State Technological College (FATEC)
Ourinhos, Brazil
Email: {fadirsalmen, paulogalego}@gmail.com

Luiz F. Carvalho, Mario Lemes Proença Jr.

Computer Science Department
State University of Londrina (UEL)
Londrina, Brazil
Email: luizfcarvalhoo@gmail.com,proenca@uel.br

***Abstract*—Traffic monitoring is a challenging task which requires efficient ways to detect every deviation from the normal behavior on computer networks. In this paper, we present two models to detect network anomaly using flow data such as bits and packets per second based on: Firefly Algorithm and Genetic Algorithm. Both results were evaluated to measure their ability to detect network anomalies, and results were then compared. We experienced good results using data collected at the backbone of a university.**

***Keywords*-Anomaly Detection; Traffic Monitoring; Network Management; Genetic Algorithm, Firefly Algorithm.**

I. Introduction

Managing a network is a complex job and requires support from a number of tools and techniques, which help manage the resources efficiently. Administrators must have a smart use of bandwidth resources, identifying anomalous traffic without human supervision.

A Denial of Service (DoS) attack can be the reason for an unavailable network. The objective of a DoS is to crash a service by attempting to reach the machine's access limit. An attacker sends packets labeled to specific IP and port addresses, simulating a legitimate access, but it sends a huge quantity of packets, with the only intent of bring down a server or service, making it impossible for a real person to access this service. A Distributed Denial of Service (DDoS) attack uses multiple compromised systems to launch several DoS attacks, coordinated against one or more victims. In fact, a DDoS attack adds the many-to-one dimension to the DoS problem [1].

For many years, network administrators used to get their technical information using the Simple Network Management Protocol (SNMP). However, this protocol could not present many details about the real network usage due to its limited set of features. With the use of data flow, administrators could obtain more knowledge about their environments. A flow record is defined by a connection between two peers reporting fields in common, those could be the endpoint addresses, protocol, time, and volume of information transferred. This gives a more detailed view on the traffic and permits using it on large networks, due to the data reduction compared to SNMP [2].

In order to identify an anomaly, we have to know what is considered normal behaviour in the network. When the normal behavior is described, every deviation of this profile can be virtually described as an anomaly. A network anomaly detection system has to work without any supervision, and have to avoid security incidents, being useful and effective in order to keep the network available as frequently as possible.

There are some tools used by network managers to identify attacks in their environments. According to Teodoro *et al.* [3] there are signature-based systems, whose detection process is generally fast and reliable because of the usual pattern-matching procedure considered in the detection stage. Nevertheless, the signature database has to be updated every moment and a signature-based system is unable to detect attacks previously unobserved.

To overcome this lack of security, there are models based on traffic characterization, which are able to learn from the normal behavior of an environment, and based on its history, detect every change in the network routine. In this paper, we present a model to identify anomalous network traffic, based on traffic characterization, which uses the Firefly Algorithm (FA) to classify network flows, and compare this model with another method, based on Genetic Algorithm (GA). Our goal is to create a Digital Signature of Network Segment using Flow Analysis (DSNSF) utilizing both GA and FA, and use this DSNSF to identify anomalous traffic through the creation of a threshold. We use a real set of data to perform the process and evaluate the results to prove the accuracy of our DSNSF models. Also, we compared these two methods to identify the advantages and disadvantages of each one.

The metaheuristics FA and GA have powerful and distinct techniques in the optimization of an objective function, specially for a wide search space. Thus, a comparative study of these algorithms, measuring their efficiency and quality to detect anomalies in computer networks was necessary.

This paper is organized as follows: Section II presents the related work. Section III explains the DSNSF-GA method giving details of the DSNSF-FA generation. Section IV discusses the result of our evaluation tests, and finally Section V concludes this paper.

II. Related Work

FA is an algorithm based on the fireflies behavior and its emitted light characteristics. In the study presented by Gandomi *et al.* [4], they used Firefly Algorithm (FA) to efficiently solve several variable issues to structural engineering optimization. Despite its restrictions, FA was used in order to decrease the following production cost: physical characteristics of beams, cylindrical pressure vessel, helical compression spring design and a reinforced concrete beam design, besides helping the development of an automotive side impact protection.

In their study, Hassanzadeh *et al.* [5] used FA algorithms, due to its high convergence features with low processing time, to optimize Otsu's method on image segmentation. Research results showed the efficiency and accuracy of the method for segmentation.

The GA is an evolutionary algorithm developed by Holland [6], which is based on the natural evolution of species. Based on operators such as selection, crossover and mutation, GA is recognized as an ideal optimization technique to solve a large variety of problems, such as organizing data under some condition or optimizing search problems. In [7], a genetic algorithm was used to organize data in clusters, when the task of GA was to search for the appropriate cluster centers.

An anomaly detection system was proposed in [8], which utilizes the SNMP protocol and searches for a correlation on the behavior of some SNMP objects, avoiding the high rate of false alarms. Another work using correlation was found in [9], which utilizes the observation among the network nodes, measuring delays and drop rates between each connection. To characterize network traffic, certain techniques could be applied such as Holt-Winters for Digital Signature, a modification of the classic statistical method of forecasting Holt-Winters [10]. In [11], the Autoregressive Integrated Moving Average (ARIMA) was used to generate forecasts for data segments. The author introduces the use of a non-classical logic called Paraconsistent Logic to improve the DSNSF employment.

III. The Generation Of DSNSF

The target of our work is to permit network administrators to identify anomalous behavior in their environments based on traffic characterization. For this purpose, we created a DSNSF, which was introduced by Proença et al. [12] in which a Digital Signature of Network Segment (DSNS) was generated using historical traffic of workdays to describe the normal network usage for subsequent weeks. Research extended and improved by [13] and [14].

In this paper, we present two metaheuristic strategies to create a DSNSF using data as bits and packets. These data were collected from the networks assets using sFlow, a standard for monitoring high-speed switched and routed networks, which uses the sampling technique to collect flows [15]. Our purpose in this work is to demonstrate that these two flow attributes, bits and packets per second can be used to identify a normal, or expected, traffic pattern and consequently appoint every network anomaly in the traffic. The first model is based on fireflies behavior and its emitted light characteristics, and is used to optimize the K-means clustering algorithm. The second model is based on the natural evolution of species theory, implemented in computing as Genetic Algorithm, which simulates the natural process of evolution in a population. Both methods are appropriate to the DSNSF construction and they will be described ahead.

A. DSNSF-FA

DSNSF-FA is an algorithm developed to construct a normal network behavior profile, based on the network traffic patterns recognition and that will enable the creation of an anomaly detection system.

The DSNSF-FA structure is based on two other algorithms, k-means, used to clustering and FA, on the determination of centroids, which will be the points responsible for the construction of DSNSF. A centroid is a point which indicates the center of the cluster. This combination is required, due to a shortcoming presented by k-means, which is solved by FA. According to Gungor and Unler [16], k-means presents a big problem in its algorithm, which is related with the centers startup. If the centers are started very close, k-means will converge to a minimum local.

1) Firefly Algorithm: The optimization process is present in every system where you want to achieve certain goals, being on the professional range, searching a lower production cost or even in vacation planning, determining the shortest path to the desired place. Before several algorithms, the use and application of metaheuristic algorithms based on nature has grown, among them is the Firefly Algorithm (FA) [17].

The optimization performed by the algorithm FA is based on the attraction between fireflies. The lower brightness firefly will position even closer to a firefly with higher luminescence and when it does not find a brighter firefly, it will randomly move until it finds a brightness that attracts it. This behavior will repeat until every firefly gets together and then this place become the best solution, in other words, optimize an objective function [18].

2) K-means: K-means is an unsupervised clustering method, whose function is to group similar items in subgroups (clusters). Thus, this enables the partitioning R records into K groups, being $R > K$, where the distance between all the resulting data of a subgroup and its said center, summed by all subgroups, to be minimized.

An easy implementation and high-speed K-means was proposed by Macqueen [19], in which objective function is shown by Equation 1:

$$KM_{(x,c)} = \sum_{i=1}^{n} \sum_{j=1}^{k} |x_i - c_j|^2 \qquad (1)$$

Where x is the data vector and c is the vector of centers, n is the number of elements on x and k is the number of centers on c.

3) DSNSF-FA model: DSNSF-FA works with historical database, arranged in time frames of 5 minutes. We found in previous works [10] [20–22], that 5 minutes is an ideal interval, however using sFlow we are dealing with sampling of data. A 5 minutes interval, preserves the exportation pattern used by Nfdump [23]. For each workday in a week, we gathered data from their equivalent counterparts in the three previous weeks. That is, if a Monday is analyzed, the historical database to be used will be related to the previous three Mondays. This database will be divided into three clusters, according to similarities defined by K-means. For each one of the clusters, FA will determine its best representative, in other words, the centroid. This operation is performed with the optimization of the chosen objective function. The DSNSF-FA works as objective function such as the Euclidean Distance, presented by the Equation 2:

$$D_{ij} = \sum_{i=1}^{Q}\sum_{j=1}^{K}\sqrt{\sum_{n=1}^{d}(x_{in} - c_{jn})^2} \qquad (2)$$

in which Q is the amount of data to be clustered, K is the total of clusters, d the dimension, $x_i n$ indicates the data value i on n and $c_j n$ is clusters center value j on dimension n. At the end of the iterations, there will be three centroids, one to each cluster defined by K-means. For each one of these centroids, the DSNSF-FA will assign a weight to theirs luminosities, defined by Equation 3:

$$Lic_k = Lrc_k * (nc_k/N) \qquad (3)$$

according to the amount of data each one represents, in which Lrc_k corresponds to the resident brightness of the cluster centroid k, N to the total amount of fireflies by iteration and nc_k refers to the amount of fireflies of cluster k, and then FA is applied on these three centers, resulting in the representative centroid of the data initially selected. Therefore, the first point of DSNSF will be generated. This approach will be held until the entire historical database is processed and the points which will generate the DSNSF are known.

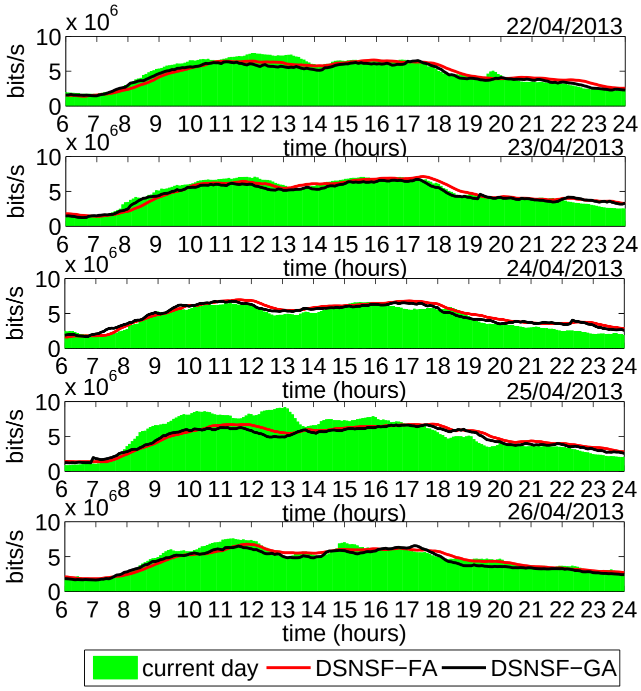


Figure 1. DSNSFs for bits/s - from 22nd to 26th April, 2013.

For the creation of DSNSF-FA, we used IP flows of historical data of State University of Londrina (UEL). These data were collected and stored in a historical basis for future reference and when requested, are delivered in files. The files were used containing bits and packets quantities, collected per second, using workdays from 22nd April to 3rd May of 2013, which served as the learning process and creation of DSNSF-FA. The DSNSF-FA, then, was superimposed on the real traffic, where it was possible to observe the traffic network anomalies.

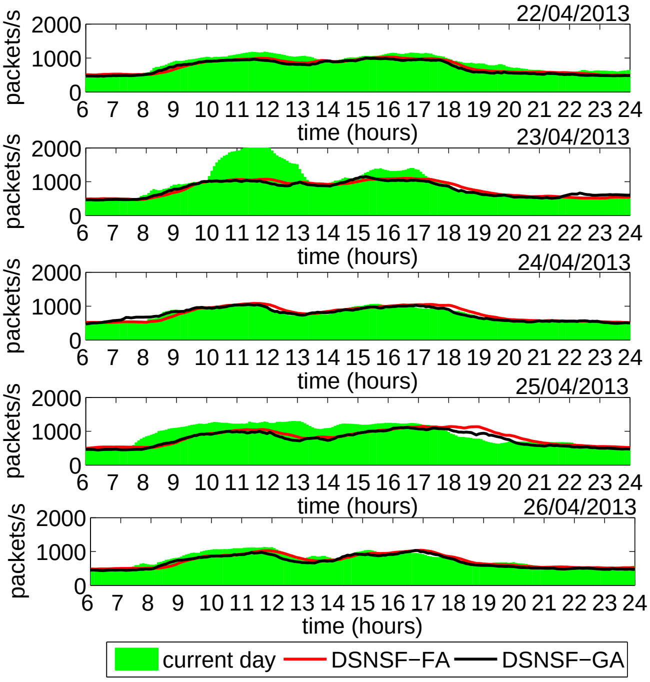


Figure 2. DSNSFs for packets/s - from 22nd to 26th April, 2013.

The DSNSF-FA algorithm operation is shown by DSNSF-FA Algorithm (1).

Algorithm 1 – DSNSF-FA

```
Require: set of bits and packets collected from historical
    database
Ensure: $X$: Vector representing the normal behavior for bits
    and packet sets of a day arranged in 288 intervals of 5
    minute, i.e. the DSNSF
for $i = 1$ to 288 do
    Applies K-means, K=3
    for $t = 1$ to number of iterations do
        Applies FA for each cluster
        Calculate the center of each cluster of the best solu-
        tion - objective function
    end for
    For each center, applies weight function
    for $t = 1$ to number of iterations do
        Applies FA to the three centers, K=1
        Calculate the center of cluster of the best solution -
        objective function
    end for
    $X_i \leftarrow$ Average among the clusters
end for
return $X$
```

Initially, the information contained in the files are prepared to provide data every 5 minutes, generating 288 samples. These data are initially processed by K-means algorithm, which distributes them in three clusters. The $K = 3$ choice was the result of the interpretation and validation of cluster, for the amount of data to be analyzed, performed by methods of Silhouette, Davies Bouldin, Calinski Harabasz, Dunn and Krzanwki Lai [24].

In each cluster, the FA algorithm is applied to find its respective centroid. This process optimizes the objective function used, where the luminosity of fireflies relates directly. After obtaining the three centroids, a weight is assigned to each one according to the amount of data they represent on their residual luminosity.

Then, the FA algorithm is used on the three centroids in order to find the result of the first 5 minutes sample analyzed. This centroid is responsible for the first data point of DNSNF-FA. In sequence, it will start the analysis of the other 287 samples, arriving at a total of 288 data points, which will then allow for the construction of the desired DSNSF-FA.

B. DSNSF-GA

The DSNSF-GA, presented in [22] uses a genetic algorithm based approach to organize data flow in clusters. Each cluster has its own centroid, and we measure the distance between the points to organize data and use the average among centroids to generate our DSNSF. The rule was the same for the DSNSF-FA, so for each workday in a week, we used data from the same day in the last three weeks, and compare them with the current day.

GA manipulates a population of potential problem solutions, trying to solve them using a coded representation of these solutions, which is the equivalent to genetic material (chromosomes) of individuals in nature. In GA, members of a population (the solutions) compete with each other to survive, reproduce and generate new solutions, using operators such as selection, crossover and mutation.

To start the process, we generate a random initial population in which we began applying the three operators. Our chromosomes have cluster centroids values. We appointed an initial population of forty parents. They create the new generation, which will replace the old one. It will repeat for a fixed number of iterations. At the end of this process, we have the best chromosomes based on their fitness function, which is the Euclidean Distance, the same as the FA algorithm. This value represents a single point in the DSNSF-GA. We have to apply the clusterization using GA for each point in the graphic, so it will be repeated for 288 times, one point every five minutes. Using the Silhouette method for interpretation and validation of clusters, best results were reached using $K = 3$.

To yield new generations, the crossover operator will combine chromosomes of two parents to create a new one. This process will continue until the old population be replaced by a new population of children. As in nature, the fittest individuals have a greater probability of generating a new offspring, who, in turn, will generate another a new one and so on. To determine the fittest individual, we calculate the sum of distance among all points and its cluster centroid in each one of the three clusters. If this distance is lower in an individual than in others, it means the data inside that cluster are well organized, i.e., there are more points closer to its central point in a cluster than in others. For our purpose, the exchange of chromosomes will improve the solution, where we are finding the shortest total distance in a chromosome.

Each chromosome also undergoes a mutation probability, which is a fixed number. Mutation allows the beginning and preservation of genetic variation in a population by introducing another genetic structure modifying some gene inside the chromosome. The new mutated chromosome will be used to generate a new offspring.

The best population will be acquired at the end of these processes, and from this we choose the best individual, which will then represent the shortest sum of distance between each point in the cluster and its respective centroid. So, we calculate the average among the three cluster centroids. This number represents a single point in the graphic, and this process will repeat for another 288 times, which represent all 5 minutes intervals during a day. By using data from three previous days to generate this single point, we now have a network signature of this day, or the DSNSF-GA.

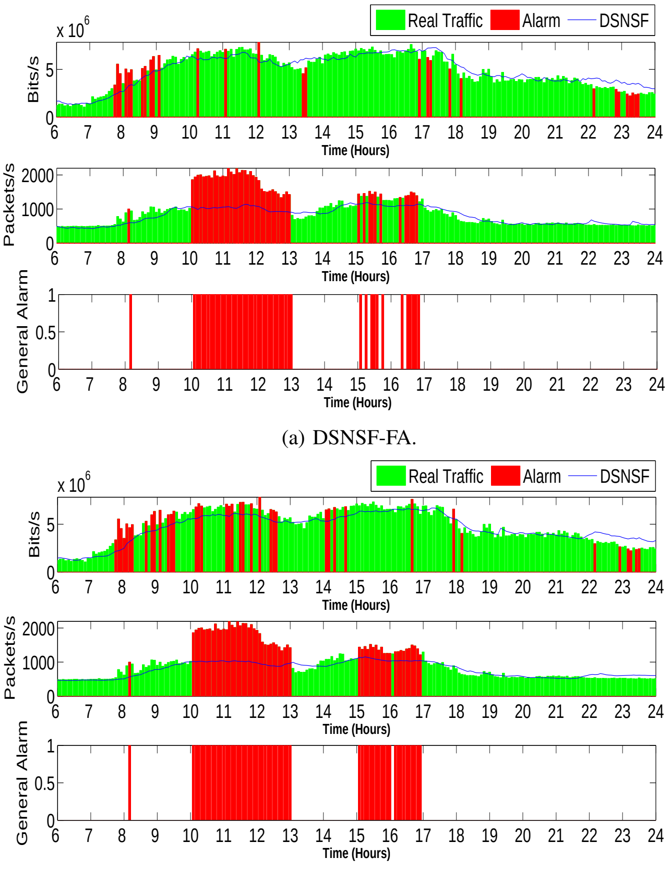
IV. Tests and Results

As described before, we used real information obtained from the historical database of the State University of Londrina (UEL). We generated the DSNSFs for the period of two weeks. Furthermore, we can see from Figure 3 the alarms generated by the change on traffic behavior. These alarms are clear during DDoS and DoS attacks artificially generated using the Scorpius software [25]. Basically, this tool injects abnormal flows directly into the exported real data flows according to the specific behavior of the desired anomaly. We have set an interval between 10:00 and 13:00 for the DDoS attack and between 15:00 and 17:00 for DoS attack for the 23^{rd} April. As the UEL working hours are from 07:00 to 23:00 hours, the historical database were analyzed for the period between 06:00 and 24:00 hours. The DSNSFs are presented in Figures 1 and 2 where the green color represents the real traffic, the red line represents the DSNSF-FA and the blue line the DSNSF-GA, both indicating the expected traffic according to their rules. The first week analyzed were from 22^{nd} to 26^{th} April 2013 and the second from 29^{th} April to 3rd May 2013.

The key process for an anomaly detection system is the traffic characterization. Both methods work characterizing traffic from sFlow data, each one using a different metaheuristic technique. Based on that traffic depiction, we can compare the prediction and the real traffic and identify the anomaly. Our intent is to compare both methods. To evaluate the accuracy of our models for these two weeks, three metrics were used: the Correlation Coefficient, the Normalized Mean Squared Error (NMSE) and the ROC curve [26].

The Correlation Coefficient (CC) function is to indicate the direction and strength of the relationship between two variables (for our propose, each DSNSF and the real data of the day). In other words, if the changes suffered by a variable are accompanied by the other, there is a correlation between them. CC has its value $\in [-1, 1]$, where 1 indicates strong positive correlation, -1 strong negative correlation and 0 corresponds to no correlation. Each week are shown in the Tables I and II.

In Tables I and II, according to the averages, both models showed good results with strong correlation in normal days, where CCs are very close to 1, and the differences found between the DSNSF-FA and DSNSF-GA were small. For the 23^{rd} April, we can see small values, both for FA and GA, specially when packets per second were analyzed. When bits per second were analyzed, there was no difference for CC.



(a) DSNSF-FA.

(b) DSNSF-GA.

Figure 3. DSNSFs Alarms for 23^{rd} April

Also, we found two other abnormal values. One from the 1st May 2013 caused by a national holiday, where we had few activities in the UEL and another for 25^{th} April. We have here a classical flash crowd traffic, caused by students applying for their enrollment in the Business Administration course, being this the last day for enrollment and only available via the Internet, where the web serves are located inside the UEL network.

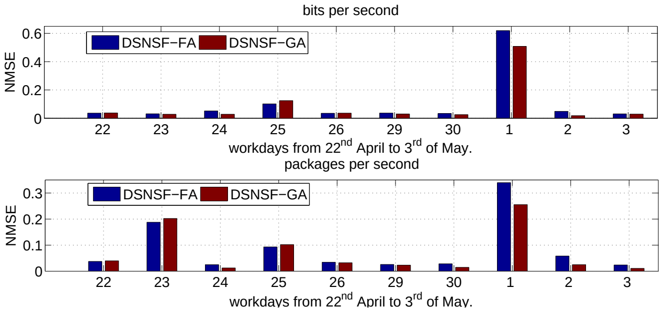


Figure 4. NMSE to DSNSF-FA and DSNSF-GA.

The NMSE is the mean square of the difference between analyzed values, checking the model's predictive ability. Their values are for 0≤NMSE≤1, and values closer to zero are the most faithful DSNSF. Figure 4 illustrates the NMSEs results for bits and packets per second, obtained by the models. Note that both DSNSF-FA and DSNSF-GA managed NMSE values below 0.02 in most days. For 23^{rd} April we found a high value for packets per second again, obviously caused by the injected attacks, which confirms that our models are able to identify deviations. Also, due to abnormal traffic on 1^{st} May 2013 caused by the national holiday, and for 25^{th} caused by the students enrollment, we found high values, both for packets and bits per second.

TABLE I. CC TABLES - DAYS BETWEEN 22^{nd} to 26^{th} OF APRIL 2013

CC\Days	22	23	24	25	26	Average
FA-bits	0.88	0.88	0.85	0.78	0.87	0.85
FA-Packets	0.87	0.74	0.86	0.64	0.82	0.81
GA-bits	0.91	0.91	0.92	0.77	0.89	0.88
GA-Packets	0.93	0.80	0.92	0.69	0.87	0.86

TABLE II. CC TABLES - DAYS BETWEEN 29^{th} OF APRIL TO 1^{st} OF MAY 2013

CC\Days	29	30	1	2	3	Average
FA-bits	0.88	0.87	0.36	0.85	0.88	0.77
FA-Packets	0.87	0.85	0.15	0.81	0.85	0.79
GA-bits	0.94	0.92	0.49	0.93	0.91	0.84
GA-Packets	0.93	0.91	0.16	0.88	0.88	0.84

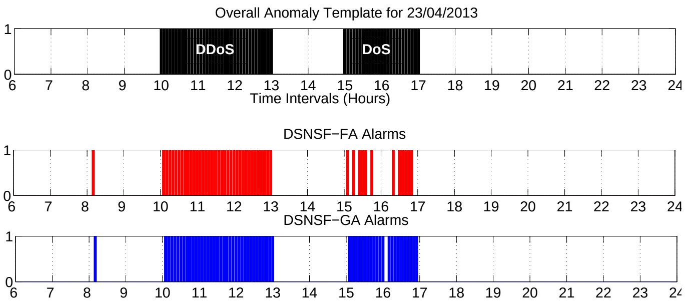


Figure 5. General alarm comparison.

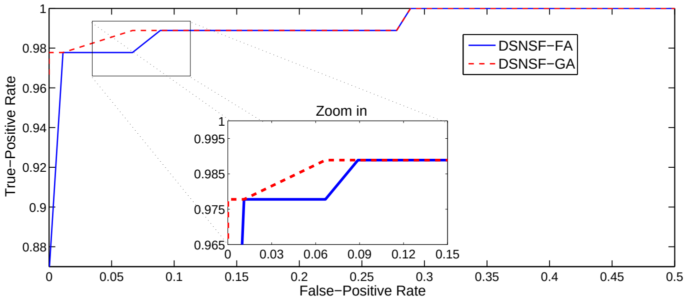


Figure 6. ROC curves comparing the trade-off between TPR and FPR rates of the proposed methods.

Figure 5 shows DDoS e DoS attacks artificially inserted and the alarms generated by the models. The data that triggered these true and false alarms, obtained by the technique of Adaptive Dynamic Time Warping (ADTW)[27], called true-positive rate (TPR) and false-positive rate (FPR) respectively, were used as the basis for the curve construction of the Receiver Operating Characteristic (ROC) and the extent accuracy of both models.

In DDoS attack's detection, both models obtained 97.3%. Moreover, for DoS attack the DSNSF-FA obtained 48% and the DSNSF-GA 88%.

The ROC curve, presented in Figure 6, describes the trade-off between TPR and FPR, which allowed to obtain the performance of DSNSF-FA and DSNSF-GA on the detection of generated artificial abnormalities. Analyzing the figure's zoom in, we notice that both models had a great performance with a minimum detection of false alarms. DSNSF-GA had a trade-off of 93.5% TPR with 0.4% FPR, as DSNSF-FA reaches 77.4% TPR with 0.4% FPR. Concerning the accuracy measure, DSNSF-GA had an accuracy of 98.3% and DSNSF-FA obtained 94.8%. The efficiency measure of the models were 96.5% to DSNSF-GA and 88.5% to DSNSF-FA.

V. Conclusion

In this work, we used two metaheuristics to create a Digital Signature of Network Segment using Flow Analysis (DSNSF). The first model uses FA to generate the DSNSF using data such as bits and packets per second, collected using sFlow pattern from the State University of Londrina (UEL). The second model uses GA to generate the DSNSF using the same set of data. Both models work characterizing traffic and comparing the predicted with the real traffic. In addition, we injected anomalous traffic in a specific day to analyze its behavior and evaluate the results to measure the efficiency of our models, finding good results.

We could see in the tables and graphs provided that both models are able to identify anomalous traffic using data such as bits and packets per second with a small advantage for the DSNSF-GA model, specially when we consider the number of true-positive alarms for DoS attacks, due to the efficiency measure and the accuracy. For future works, we intend to increase the number of dimensions in our search, since network flows can give us more data, such as IP and ports information for example.

Acknowledgment

This work was supported by SETI/Fundação Araucária and MCT/CNPq for Betelgeuse Project financial support. Also, the authors would like to thank the support from São Paulo State Technological College (Fatec Ourinhos).

References

[1] E. Petac, A. Alzoubaidi, and P. Duma, "Some experimental results about security solutions against ddos attacks," in Signals, Circuits and Systems (ISSCS), 2013 International Symposium on, July 2013, pp. 1–4.

[2] B. Trammell and E. Boschi, "An introduction to ip flow information export (ipfix)," IEEE Communications Magazine, vol. 49, no. 4, 2011, pp. 89–95.

[3] P. Teodoro, P. Feldstedt, and D. Zuiga, "Automatic signature generation for network services through selective extraction of anomalous contents," in Telecommunications (AICT), 2010 Sixth Advanced International Conference on, May 2010, pp. 370–375.

[4] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Mixed variable structural optimization using Firefly Algorithm," Computers & Structures, vol. 89, no. 23-24, Dec. 2011, pp. 2325–2336.

[5] T. Hassanzadeh, H. Vojodi, and A. M. E. Moghadam, "An image segmentation approach based on maximum variance Intra-cluster method and Firefly algorithm," in 2011 Seventh International Conference on Natural Computation. IEEE, Jul., pp. 1817–1821.

[6] J. H. Holland, Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence. U Michigan Press, 1975.

[7] U. Maulik and S. Bandyopadhyay, "Genetic algorithm-based clustering technique," Pattern Recognition, vol. 33, no. 9, 2000, pp. 1455–1465.

[8] M. L. Proença Jr., B. B. Zarpelão, and L. S. Mendes, "Anomaly detection for network servers using digital signature of network segment," in Proceedings - Advanced Industrial Conference on Telecommunications/Service Assurance with Partial and Intermittent Resources Conference/E-Learning on Telecommunications Workshop AICT/SAPIR/ELETE 2005, vol. 2005, 2005, pp. 290–295, doi:10.1109/AICT.2005.26.

[9] R. Steinert and D. Gillblad, "Towards distributed and adaptive detection and localisation of network faults," in Telecommunications (AICT), 2010 Sixth Advanced International Conference on, May 2010, pp. 384–389.

[10] M. V. O. Assis, J. J. P. C. Rodrigues, and M. L. Proença Jr., "A seven-dimensional flow analysis to help autonomous network management," Information Sciences, vol. 278, 2014, pp. 900 – 913, doi:10.1016/j.ins.2014.03.102.

[11] E. H. M. Pena, S. Barbon, J. J. P. C. Rodrigues, and M. L. Proença Jr., "Anomaly detection using digital signature of network segment with adaptive arima model and paraconsistent logic," in Computers and Communication (ISCC), 2014 IEEE Symposium on, June 2014, pp. 1–6, doi:10.1109/ISCC.2014.6 912 503.

[12] M. L. Proença Jr., C. Coppelmans, M. Bottoli, and L. Souza Mendes, "Baseline to help with network management," in e-Business and Telecommunication Networks. Springer Netherlands, 2006, pp. 158–166, doi: 10.1007/1–4020–4761–4_12.

[13] B. B. Zarpelão, L. S. Mendes, M. L. Proença Jr., and J. J. P. C. Rodrigues, "Parameterized anomaly detection system with automatic configuration," in Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE, Nov 2009, pp. 1–6, doi: 10.1109/GLOCOM.2009.5 426 189.

[14] A. A. Amaral, B. B. Zarpelão, L. de Souza Mendes, J. J. P. C. Rodrigues, and M. L. Proença Jr., "Inference of network anomaly propagation using spatio-temporal correlation," Journal of Network and Computer Applications, vol. 35, no. 6, 2012, pp. 1781 – 1792, doi: 10.1016/j.jnca.2012.07.003.

[15] P. Phaal, S. Panchen, and N. McKee, "InMon corporation s sFlow: A method for monitoring traffic in switched and routed networks," RFC 3176, Tech. Rep., 2001.

[16] Z. Güngör and A. Ünler, "K-harmonic means data clustering with simulated annealing heuristic," Applied Mathematics and Computation, vol. 184, no. 2, Jan. 2007, pp. 199–209.

[17] X.-S. Yang, "Firefly algorithms for multimodal optimization" , in: Stochastic Algorithms: Foundations and Applications, ser. Lecture Notes in Computer Science, O. Watanabe and T. Zeugmann, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, vol. 5792.

[18] I. Fister, X.-S. Yang, and J. Brest, "A comprehensive review of firefly algorithms," Swarm and Evolutionary Computation, Dec. 2013, pp. 34–46.

[19] J. Macqueen, "Some Methods for Classification and Analysis of Multivariate Observations," Proc. Fifth Berkeley Symp. on Math. Statist. and Prob., Vol. 1 (Univ. of Calif. Press, 1967), 281-297, vol. 1, 1967, pp. 281–297.

[20] I. Paschalidis and G. Smaragdakis, "Spatio-temporal network anomaly detection by assessing deviations of empirical measures," Networking, IEEE/ACM Transactions on, vol. 17, no. 3, June 2009, pp. 685–697.

[21] M. H. A. C. Adaniya, M. F. Lima, J. J. P. C. Rodrigues, T. Abrao, and M. L. Proença Jr., "Anomaly detection using dsns and firefly harmonic clustering algorithm," in Communications (ICC), 2012 IEEE International Conference on, June 2012, pp. 1183–1187, doi:10.1109/ICC.2012.6 364 088.

[22] P. R. G. Hernandes Jr, L. F. Carvalho, G. Fernandes Jr., and M. L. Proença Jr., "Digital signature of network segment using genetic algorithm and ant colony optimization metaheuristics," in The Eighth International Conference on Emerging Security Information, Systems and Technologies, Nov 2014, pp. 62–67.

[23] "nfdump - documentation," http://nfdump.sourceforge.net/, 2014, access date: May 13, 2015.

[24] K. Wang, B. Wang, and L. Peng, "CVAP: Validation for Cluster Analyses," Data Science Journal, vol. 8, 2009, pp. 88–93.

[25] "Scorpius - sflow anomaly simulator," http://redes.dc.uel.br/scorpius/, 2013, access date: Apr 28, 2015.

[26] B. Penney, M. King, and S. Glick, "Restoration of combined conjugate images in spect: comparison of a new wiener filter and the image-dependent metz filter," Nuclear Science, IEEE Transactions on, vol. 37, no. 2, Apr 1990, pp. 707–712.

[27] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," Acoustics, Speech and Signal Processing, IEEE Transactions on, vol. 26, no. 1, Feb 1978, pp. 43–49.