An Evolutionary Multi-Agent Approach to Anomaly Detection and Cyber Defense

[Extended Abstract]

Marco Carvalho
Institute for Human and Machine Cognition
15 SE Osceola Ave
Ocala, FL
mcarvalho@ihmc.us

Carlos Perez
Institute for Human and Machine Cognition
15 SE Osceola Ave
Ocala, FL
cperez@ihmc.us

ABSTRACT

In this paper we present an evolutionary multi-agent approach for anomaly detection based on adaptive clustering and classification. An evolutionary algorithm is proposed to allow agents to self-organize and cluster the data using different subsets of attributes, and dynamically created meta-attributes. A performance metric is defined to allow the best agents to be reinforced and evolve, and to progressively eliminate ineffective agents. Our preliminary results show how the proposed approach can be used in isolation for intrusion detection, or in combination with other mechanisms to improve the performance and capabilities of intrusion, and anomaly detection systems

Categories and Subject Descriptors

D.4.6 [**Security and Protection**]: Unauthorized access

General Terms

Cybersecurity

Keywords

Cybersecurity, feature selection, evolutionary approaches

1. INTRODUCTION

There are two general approaches to intrusion detection: anomaly detection and signature detection [2]. Anomaly detection systems work under the assumption that abnormal events are suspicious and have higher chances of being part of an intrusion. Signature detection systems are based on knowledge of what constitutes an intrusion, matching each event against a set of rules do determine if they constitute legal and illegal events.

Both approaches have their advantages and disadvantages. Signature-based systems are limited to detecting known attacks, well pre-defined signatures. Conversely, anomaly detection systems can potentially detect novel intrusions, with never seen before patterns or signatures. But at the same time, and for the same reason, anomaly detection systems have higher false positive rates, often requiring user validation for any practical use. However, when high false positive rate events are given to a human analyst for consideration they can easily be overwhelming, leading the user to disable, or ignore the intrusion detection system (IDS) . This phenomenon is described by [3] as the base rate fallacy and it has been described as a possible vulnerability for attacking IDS systems [12]. The base rate fallacy states that when the base intrusion rate is low, the overall effectiveness of the system is highly affected by the false alarm rate. So, from that perspective, the main limitation of IDS is their ability to suppress false alarms rather than accurately detecting true intrusions.

In this paper we will present an evolutionary approach for cluster analysis that in itself constitutes an anomaly detection system. If used in combination with a signature detection system, it can help improved the overall performance of the system, leveraging the strengths of both approaches. The following section presents a short overview of related work. Section 3 describes in detail the proposed approach for evolutionary anomaly detection. Section 4 describes how the proposed anomaly detection system can be combined with a signature detection system. Section 5 presents some preliminary experimental results. And finally, section 6 discusses some conclusions and some opportunities for future work.

2. RELATED WORK

There have been several proposals for intrusion detection systems based on anomaly detection. The authors of [13] use clustering to group abnormal traffic. In [11], neural networks and support vector machines are used for the same purpose. Decision trees [4], and Hidden Markov models have also been used for anomaly detection [14]. For a brief background and review, Lazarevic at al. [8] provide a comparative study of several anomaly detection systems. Similar to our approach, Kim et al. [6] propose the use of genetic algorithms to improve an intrusion detection system based on support vector machines. However, they do not explore the online construction of abstract features, of the correlation of such features with a alarms, or a secondary classification systems, which is a key capability of our proposed approach.

The authors of [7] use Bayesian event classification to improve the false alarm rate of an IDS. They identified as the two main reasons for the large number of false alarms the simplistic aggregation of information from multiple sources, and the lack of integration of additional information into the decision process. They propose using Bayesian networks to improve the aggregation and to seamlessly incorporate additional information. In this paper we are also proposing a Bayesian approach to easily combine information provided by existing signature detection systems with the anomaly detection approach that we are proposing.

3. AN EVOLUTIONARY APPROACH FOR ANOMALY DETECTION

The proposed approach consists of a multi-agent infrastructure where individual software agents are responsible for clustering network flows (or events, in general) based on a self-selected set of attributes, and a chosen number of clusters. The attributes selected by each agent can be some of basic event attributes, or derived attributes composed of the output from other agents, possibly combined with other basic attributes. Each agent will maintain a subset of attributes and a number of clusters, which loosely represents its genetic code, or ”DNA”. At a first level, the evolutionary process of system consists on the selection of improved populations of clustering agents.

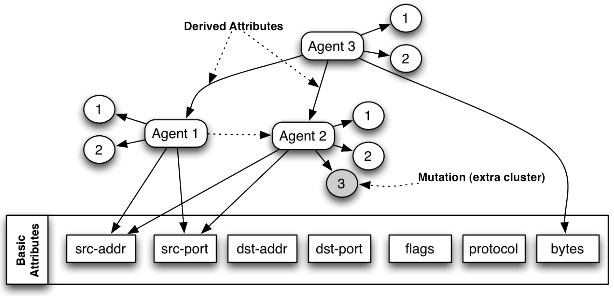

Figure 1: Evolutionary Anomaly Detection Procedure

Through the evolutionary process, the performance of the clustering agents can be measured to explore and identify the best set of clustering agents for the events. Using a genetic algorithm, agents with good performance will be replicated with a small mutation in DNA. Possible mutations include, the example, the removal or addition of a clustering attribute, or the increase/decrease of the number of clusters. At each generation, agents with a poor clustering performance are removed. The agent population is maintained to satisfy a minimum and maximum level that is based on available system resources. Figure 1 illustrates the proposed approach. In that example, Agent 1 is clustering data in 2 clusters using the source address and source port. Agent 2 is a descendant of agent 1, with a single mutation. Agent 2 is clustering the data in 3 clusters. Agent 3 is using derived attributes that consist of the output from agents 1 and 2, and one basic attribute. Agent 3 is also clustering the data in two clusters.

For the evolution of the system, a fitness function must be defined for the different agents. Several approaches can be used for measuring the performance of the agents. For anomaly detection, a reasonable metric is their ability of the agent to cluster (i.e. separate) the normal data, allowing the detection of anomalous events . Intuitively, a clustering agent that produces clusters with high dispersion is less effective than a clustering agent that produces clusters with a low dispersion of elements. From that perspective, one may choose a performance measure based on the minimization of the squared distances of elements to the centroids of the clusters.

Unfortunately, this type of measure will favor complex models, that is, models with high number of clusters. The higher the number of clusters, the better the data will fit the model, leading to an over-fitting of the data. This problem has already been dealt with in statistical model selection. For our system, we decided to use a variation of a well known scoring function known as the Bayesian Information Criterion (BIC) [15]. BIC scoring is used in statistical model selection to find a good balance between the fit and the complexity of a given model. The complexity of the model, in our case, is represented by the number of clusters in the model, and the fitness is represented by the error variance. The BIC scoring function for our system is:

$$BIC = N \cdot \ln(s^2) + k \cdot \ln(N), \tag{1}$$

where N is size of the training sample, k number of clusters, and s^2 is the error variance of the model, which in our case is the average squared distances to the centroids for the training data.

The first component of the BIC score controls the fitness of the model, while the second component controls the complexity. The best model is the model with the lowest BIC score. Whether this function, or more exactly, this variation of the BIC scoring function would have good asymptotic properties is still unclear, but it servers well as an approximate performance metric for our purposes.

The purpose of the clustering, however, is to enable the anomaly detection, so given a set of clustering agents we can define the notion of anomaly for a given event in the following way: Each event will be clustered by each agent in the system. For a given event and a selected cluster, we can use the squared error of the event in the assigned cluster to determine how “odd” the event looks in the cluster, even given the fact that the current cluster is the best possible one for the current event. For the purpose, a *oddity* score will then be defined as the cumulative distribution function of the squared error:

$$Oddity(x) = \frac{\sum_{i \in C} I(|i-c|^2 \leq |x-c|^2)}{|C|}, \tag{2}$$

where C is the set of events added to the cluster from the training data, c is the cluster’s centroid, and I is the function:

$$I(x \leq y) = \begin{cases} 1, & \text{if } x \leq y, \\ 0, & \text{otherwise.} \end{cases} \qquad (3)$$

If the squared error is higher than normal, then the cumulative distribution function will be high also, indicating a highly *odd* event, from the perspective of that agent. With that information we can then combine the *oddity* measure for all agents for the current event, by computing the simple average.

4. EVENT CLUSTERING AND IDS ALARM CORRELATIONS

The proposed algorithm was initially design to support and improve conventional intrusion detection systems, and to correlate potentially malicious events with related events that may have not matched a signature. For the purpose, the anomaly detection described in the previous section can be easily combined with a signature detection system, by tracking the correlation between the alarms generated by the signature detection system and the clusters assigned to the events.

Taking the output from the clustering agents and the generated alarms from a signature detection system, we can estimate the probability of the event being part of an intrusion using a naïve Bayesian classifier (Figure 2). A naïve Bayesian classifier allows to combine evidence from multiple classifiers, in this case represented by the response of the clustering agents, assuming that their classifications are conditionally independent from each other given the presence of an anomaly. When combined with the IDS alarm, the naïve Bayesian classifier estimates the probability of the event being part of an intrusion using the following formula:

$$P(I|A_1 \ldots A_n) = \frac{P(I)\prod_{i=1}^{n} P(A_i|I)}{P(I)\prod_{i=1}^{n} P(A_i|I) + P(\neg I)\prod_{i=1}^{n} P(A_i|\neg I)}, \qquad (4)$$

where, A_i represents the cluster that agent i selected for the event, $P(A_i|I)$ is the probability that an intrusion event will be assigned to the cluster selected by agent i, and $P(A_i|\neg I)$ is the probability that a clean event will be assigned to the cluster selected by agent i. These probabilities can easily be estimated by keeping counts of the number of intrusion and non-intrusion events for each cluster of every agent. The probability $P(I)$ is the base intrusion rate of the system, which can be estimated from the historical data.

In addition to the information provided from the clusters, we can also include in the Bayesian classification, the output from the signature detection system, just as an additional piece of evidence. In this case, the $P(S|I)$ (probability of a signature firing given an intrusion), and the $P(S|\neg I)$ (probability of a signature firing given a clean event), correspond to the true positive rate and false positive rate of the signature detection system, which can be provided by the signature detection system itself, or estimated from historical data.

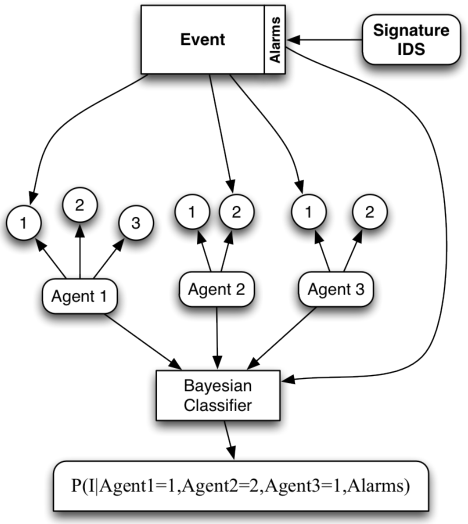

Figure 2: Cluster-Alarm Correlation Procedure

5. EXPERIMENTAL RESULTS

One of the claims of this work, is that the proposed hierarchical clustering algorithms can be combined with an IDS system to improve classification and reduce error rates. Ideally, the proposed hypothesis should be tested under realistic traffic conditions, with regular attacks and natural cluster structures. However, given the difficulties associated with the acquisition of annotated datasets, and since our focus was primarily on the improvement of the classification, we choose to relax the requirements for realism and use a publicly available and well know dataset.

Table 1: True positive and false positive rates for the Snort and the proposed approached

IDS	TPR		FPR	
	Average	Deviation	Average	Deviation
Snort	0.6542	–	0.0411	–
Oddity	0.4267	0.1052	0.1453	0.06732
Cluster-Alarm	0.8078	0.0403	0.1413	0.0442

The experimental evaluation was performed using a subset of synthetic data from the DARPA 1999 dataset for off-line intrusion detection evaluation [9]. The DARPA 1999 dataset consists of 5 weeks of data. The first 3 weeks are intended to be training data. Weeks 1 and 3 do not contain attacks, week 2 contains some attacks. Weeks 4 and 5 are intended to be the testing data. They contain some network based attacks mixed with normal background data. The data that was used for testing was only the data from week 5. The DARPA 1999 is known to have various issues [10] and we are fully aware of these problems, however, we choose to use it in this preliminary work as a starting point for intrusion detection because we were focused on a strict comparison of the combined cluster-alarm approach with a simple IDS notification systems.

The signature detection system that was used as reference was Snort [1] with the default rule set. Again, the focus of the work was on the improvement of the classification, so we chose not to optimize or extend the default ruleset. The clustering algorithm that was used for testing the system is known as k-means [5], but the proposed approach is agnostic of the specific clustering algorithm, as long as it relies on a pre-defined set of clusters.

Since the evolutionary process is randomized, different runs of the algorithm over the same data may yield different clustering structures. The results, however, of an evolved population is likely to be similar, regardless of the structure. In order to verify that claim, we measured not only the accuracy of the composed classified but also the variance of its error rates. The oddity based detection and the cluster-alarm correlation detection approaches were executed on the same data multiples times and their averages and variances computed (Table 1). An example of the learning structure is show in figure 3.

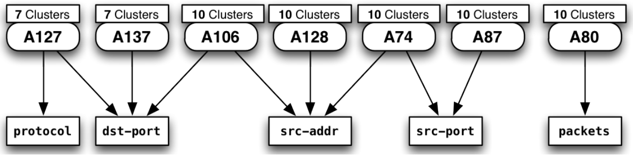

Figure 3: Learning structure example

The oddity based detection system shows a bad performance in terms of true positive rate and false positive rate with respect to Snort and shows high variability on the results. This variability can be improved by calibrating the system to increase the convergence rate of the evolutionary algorithm. On the positive side, it is noticeable that, as it was pointed before, the oddity based was able to catch attacks without requiring any external or specific knowledge about the attacks.

The Cluster-Alarm correlation shows a better performance in terms of the true positive rate, indicating that it was successful at combining the detection capabilities of both, the signature based system and the oddity based system. However, the false positive rate is still worst than the false positive rate of Snort, showing a small but not significant improvement with respect to the oddity based detection.

6. CONCLUSIONS

In this paper we presented an approach for evolutionary anomaly detection. We showed how this approach could be used for detecting intrusions, and also showed how this approach can be easily combined with existing intrusion detection systems to improve the intrusion detection capabilities. The results, while encouraging, still leave a lot of room for improvement, in particular in terms of the false positive rate. Because, as it was noted in the introduction, when the intrusion rate is low, the main limitation of an intrusion detection system is their ability to suppress false alarms rather than accurately detecting intrusions. As part of our future work we will compare our results with other anomaly detection algorithms for the same dataset, and test our approach on more realistic datasets.

7. ACKNOWLEDGEMENTS

This material is partially based upon work supported by the Department of Energy National Energy Technology Laboratory under Award Number(s) DE-OE0000511.

8. REFERENCES

[1] Snort. `http://www.snort.org/`.

[2] S. Axelsson. Intrusion detection systems: A survey and taxonomy. 2000.

[3] S. Axelsson. The base-rate fallacy and the difficulty of intrusion detection. *ACM Transactions on Information and System Security (TISSEC)*, 3(3):186–205, 2000.

[4] M. Carvalho and C. M. Teng. Automatic discovery of attack messages and pre- and post-conditions automatic discovery of attack messages and pre- and post-conditions for attack graph generation. In E. L. Armistead, editor, *5th International Conference on Information Warfare and Security (ICIW)*, pages 378–388, Wright-Patterson AFB, Ohio, USA, April 8-9 2010. AFRL, Academic Publishing Limited.

[5] J. Hartigan and M. Wong. A k-means clustering algorithm. *JR Stat. Soc., Ser. C*, 28:100–108, 1979.

[6] D. Kim, H. Nguyen, and J. Park. Genetic algorithm to improve SVM based network intrusion detection system. In *Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on*, volume 2, pages 155–158. IEEE, 2005.

[7] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur. Bayesian event classification for intrusion detection. 2003.

[8] A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur, and J. Srivastava. A comparative study of anomaly detection schemes in network intrusion detection. In *Proceedings of the Third SIAM International Conference on Data Mining*, volume 3, 2003.

[9] R. Lippmann, J. Haines, D. Fried, J. Korba, and K. Das. The 1999 DARPA off-line intrusion detection evaluation. *Computer Networks*, 34(4):579–595, 2000.

[10] J. McHugh. Testing intrusion detection systems: A critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by lincoln laboratory. *ACM Transactions on Information and System Security*, 3(4):262–294, 2000.

[11] S. Mukkamala, G. Janoski, and A. Sung. Intrusion detection using neural networks and support vector machines. In *Proceedings of IEEE international joint conference on neural networks*, volume 1702, 2002.

[12] S. Patton, W. Yurcik, and D. Doss. An Achilles' heel in signature-based IDS: Squealing false positives in SNORT. In *Proceedings of RAID 2001 fourth International Symposium on Recent Advances in Intrusion Detection October*, volume 10, page 12. Citeseer, 2001.

[13] L. Portnoy, E. Eskin, and S. Stolfo. Intrusion detection with unlabeled data using clustering. In *Proceedings of ACM CSS Workshop on Data Mining Applied to Security, Philadelphia, PA*, 2001.

[14] Y. Qiao, X. Xin, Y. Bin, and S. Ge. Anomaly intrusion detection method based on HMM. *Electronics Letters*, 38(13):663–664, 2002.

[15] G. Schwarz. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.