BAYESIAN-NETWORKS-BASED MISUSE AND ANOMALY PREVENTION SYSTEM

Pablo Garcia Bringas, Yoseba K. Penya
University of Deusto, Faculty of Engineering - ESIDE, 48007 Bilbao, Bizkaia, Spain

Stefano Paraboschi, Paolo Salvaneschi
University of Bergamo, Faculty of Engineering, 24044 Dalmine, Bergamo, Italy

Keywords: Intrusion Detection, Intrusion Prevention, Misuse Detection, Anomaly Detection, Data Mining, Machine Learning, Bayesian Networks.

Abstract: Network Intrusion Detection Systems (NIDS) aim at preventing network attacks and unauthorised remote use of computers. More accurately, depending on the kind of attack it targets, an NIDS can be oriented to detect misuses (by defining all possible attacks) or anomalies (by modelling legitimate behaviour and detecting those that do not fit on that model). Still, since their problem knowledge is restricted to possible attacks, misuse detection fails to notice anomalies and vice versa. Against this, we present here ESIDE-Depian, the first unified misuse and anomaly prevention system based on Bayesian Networks to analyse completely network packets, and the strategy to create a consistent knowledge model that integrates misuse and anomaly-based knowledge. Finally, we evaluate ESIDE-Depian against well-known and new attacks showing how it outperforms a well-established industrial NIDS.

1 INTRODUCTION

The Internet System Consortium estimates that, nowadays, more than 489 million computers are connected to the biggest network in the world (Internet System Consortium, 2007). Being part of such a vast community brings amazing possibilities but also worrying dangers. Against this record-breaking growth (the same survey in July 2000 yielded *only* 93 million computers) traditional passive measures for isolation and access control have been proved inadequate to dam the current flood of digital attacks and intrusion attempts.

In this way, the area of Computer Security and, more accurately, Network Intrusion Detection Systems (NIDS) have been lately subject of increasing interest and research as suited answer against the mentioned threat. Specifically, a NIDS is a software in charge of distinguishing among legitimate and malicious network users. Moreover, due to the rising complexity and volume of the attacks, NIDS are performed in an automated manner, so the NIDS software monitors system usage to identify behaviour breaking the security policy.

Based on their scope, NIDS can be divided into misuse or anomaly detectors. Initially, NIDS where conceived as *misuse detectors*. This is, they had a well-defined set of malicious behaviours and they just supervised the system to find those. Misuse Detection Systems are commonly characterized by a high accuracy in their decisions, as well as by an excellent throughput, since they are very good at detecting well-known attacks. Nevertheless, they also present an important flaw because they are not able to response against unknown attacks and, further, they require that an operator specifies the expert knowledge. In order to overcome this shortcoming, another strategy, known as *anomaly detection*, has been developed during the last decade. Anomaly Detection Systems model legitimate system usage in order to obtain afterwards a certainty measure of potential deviations from that normal profile. Each deviation that is found significant enough will be worth of being considered anomalous and notified to a human operator. This alarm can be analysed manually or processed

Garcia Bringas P., K. Penya Y., Paraboschi S. and Salvaneschi P. (2008).
BAYESIAN-NETWORKS-BASED MISUSE AND ANOMALY PREVENTION SYSTEM.
In *Proceedings of the Tenth International Conference on Enterprise Information Systems - AIDSS*, pages 62-69
DOI: 10.5220/0001702300620069
Copyright © SciTePress

automatically either to filter intruder actions (in line with Intrusion Prevention paradigm), reconfigure the environment or collect audit information. Anomaly Detection Systems, however, cannot compete with Misuse Detection ones when it comes to detect well-known attacks; therefore, each approach fails when it comes to the other's area of expertise.

Now, several paradigms have been used to develop diverse NIDS approaches (a detailed analysis of related work in this area can be found for instance in (Kabiri and Ghorbani, 2005)): Expert Systems (Alipio et al., 2003), Finite Automatons (Vigna et al., 2000), Rule Induction Systems (Kantzavelou and Katsikas, 1997), Neural Networks (Mukkamala et al., 2005), Intent Specification Languages (Doyle et al., 2001), Genetic Algorithms (Kim et al., 2005), Fuzzy Logic (Chavan et al., 2004) Support Vector Machines (Mukkamala et al., 2005), Intelligent Agent Systems (Helmer et al., 2003) or Data-Mining-based approaches (Lazarevic et al., 2003). Still, none of them tries to combine anomaly and misuse detection and, fail when applied to either well-known or zero-day attacks. There is one exception in (Valdes and Skinner, 2000), but the analysis of network packets is too superficial (only headers) to yield any good results in real life. Moreover, few proposed models such as (Singhal and Jajodia, 2006; Brugger, 2004) add historical data neither for analysis nor for sequential adaptation of the knowledge representations models used for detection, so this information about the essence and the potential trends of the target system is not commonly considered, so as to, e.g., obtain a baseline profile of normal behaviour.

Against this background, this paper advances the state of the art in two main ways. First, we present ESIDE-Depian (Intelligent Security Environment for Detection and Prevention of Network Intrusions), the first inherently unified Misuse and Anomaly Detector that analyses the whole network packet. Second, we detail a new methodology and knowledge representation model that allow the adaptive reasoning engine of ESIDE-Depian infer conclusions considering both Misuse and Anomaly Detection characteristic knowledge in an unified and homogeneous way.

The remainder of the paper is structured as follows. Section 2 describes the general architecture of ESIDE-Depian, including the creation process of the knowledge model for each kind of Bayesian Experts used for Misuse Detection and the integration of all verdicts in one Naive Bayesian Network to assure a coherent outcome. Section 3 presents the experiments carried out to evaluate ESIDE-Depian with real network traffic. Finally, Section 4 concludes and outlines the future work.

2 ARCHITECTURE AND APPROACH

The internal design of ESIDE-Depian is principally determined by its dual nature. Being both a misuse and anomaly detection system requires answering to sometimes clashing needs and demands. This is, it must be able to simultaneously offer efficient response against both well-known and zero-day attacks. In order to ease the way to this goal, ESIDE-Depian has been conceived and deployed in a modular way that allows decomposing of the problem into several smaller units. Thereby, Snort (a rule-based state of the art Misuse Detection System (Roesch, 1999)), has been integrated to improve the training procedure to increase the accuracy of ESIDE-Depian. Following a strategy proven successful in this area (Alipio et al., 2003), the reasoning engine we present here is composed of a number of Bayesian experts working over a common knowledge model.

The Bayesian experts must cover all possible areas where a menace may rise. In this way, there are 5 Bayesian experts in ESIDE-Depian, as follows: 3 of them deal with packet headers of TCP, UDP, ICMP and IP network protocols, the so-called TCP-IP, UDP-IP and ICMP-IP expert modules. A further one, the Connection Tracking Expert, analyses potential temporal dependencies between TCP network events and, finally, the Protocol Payload Expert in charge of the packet payload analysis. In order to obtain the knowledge model, each expert carries out separately a Snort-driven supervised learning process on its expertise area. Therefore, the final knowledge model is the sum of the individual ones obtained by each expert. Fig. 1 shows the general ESIDE-Depian architecture.

The rest of this section is devoted to detail the creation and up-dating of the knowledge model for each kind of Bayesian expert (including the exact role of Snort in this process) and the way ESIDE-Depian converges all experts' verdicts.

2.1 ESIDE-Depian Knowledge Model Generation Process

The obtaining of the knowledge model in an automated manner can be achieved in an unsupervised or supervised way.

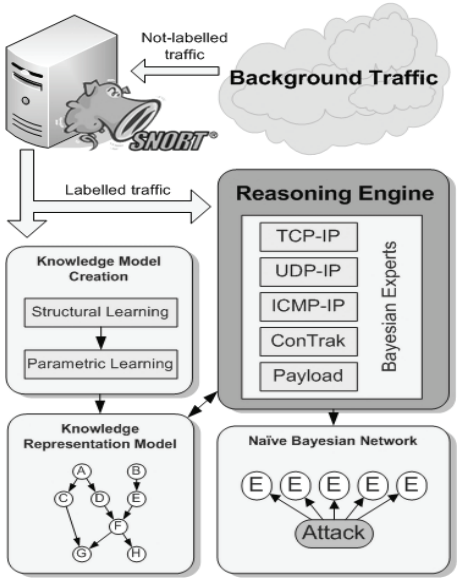

Figure 1: ESIDE-Depian General Architecture.

Typically, unsupervised learning approaches don't have into consideration expert knowledge about well-known attacks. They achieve their own decisions based on several mathematical representations of distance between observations from the target system, revealing themselves as ideal for performing Anomaly Detection. On the other hand, supervised learning models do use expert knowledge in their making of decisions, in the line of Misuse Detection paradigm, but usually present high-cost administrative requirements. Thus, both approaches present important advantages and several shortcomings. Being both ESIDE-Depian, it is necessary to set a balanced solution that enables to manage in an uniform way both kinds of knowledge. Therefore, ESIDE-Depian uses not only Snort information gathering capabilities, but also Snort's decision-based labelling of network traffic. Thereby, the learning processes inside ESIDE-Depian can be considered as automatically-supervised Bayesian learning, divided into the following phases. Please note that this sequence only applies for the standard generation process followed by the Packet Header Parameter Analysis experts, (i.e. the TCP-IP, UDP-IP and ICMP-IP expert modules):

- *Traffic Sample Obtaining*. First we need to establish the information source in order to gather the sample. This set usually includes normal traffic (typically gathered from the network by sniffing, arp poisoning or so), as well as malicious traffic generated by the well-known arsenal of hacking tools such as (Metasploit, 2006), etc. Subsequently, the Snort Intrusion Detection System embedded in ESIDE-Depian adds labelling information regarding the legitimacy or malice of the network packets. Specifically, Snort's main decision about a packet is added to the set of detection parameters, receiving the name of *attack variable*. In this way, it is possible to obtain a complete sample of evidences, including, in the formal aspect of the sample, both protocol fields and also Snort labelling information. Therefore, it combines knowledge about normal behaviour and also knowledge about well-known attacks, or, in other words, information necessary for Misuse Detection and for Anomaly Detection.
- *Structural Learning.* The next step is devoted to define the operational model ESIDE-Depian should work within. With this goal in mind, we have to provide logical support for knowledge extracted from network traffic information. Packet parameters need to be related into a Bayesian structure of nodes and edges, in order to ease the later conclusion inference over this mentioned structure. In particular, the PC-Algorithm (Spirtes et al., 2001) is used here to achieve the structure of causal and/or correlative relationships among given variables from data. In other words, the PC-Algorithm uses the traffic sample data to define the Bayesian model, representing the whole set of dependence and independence relationships among detection parameters. Due to its high requirements in terms of computational and temporal resources, this phase is usually performed in an off-line manner.
- *Parametric Learning*. The knowledge model fixed so far is a qualitative one. Therefore, the following step is to apply parametric learning in order to obtain the quantitative model representing the strength of the collection of previously learned relationships, before the exploitation phase began. Specifically, ESIDE-Depian implements maximum likelihood estimate (Murphy, 2001) to achieve this goal. This method completes the Bayesian model obtained in the previous step by defining the quantitative description of the set of edges between parameters. This is, structural learning finds the structure of probability distribution functions among detection parameters, and parametric learning fills this structure with proper conditional probability values.

- *Bayesian Inference*. Next, every packet capture from the target communication infrastructure needs one value for the posterior probability of a badness variable, (i.e. the Snort label), given the set of observable packet detection parameters. So, we need an inference engine based on Bayesian evidence propagation. More accurately, we use the Lauritzen and Spiegelhalter method for conclusion inference over junction trees, provided it is slightly more efficient than any other in terms of response time (Castillo et al., 1997). Thereby, already working in real time, incoming packets are analysed by this method (with the basis of observable detection parameters obtained from each network packet) to define the later probability of the attack variable. The continuous probability value produced here represents the certainty that an evidence is good or bad. Generally, a threshold based alarm mechanism can be added in order to get a balance between false positive and negative rates, depending on the context.
- *Adaptation*. Normally, the system operation does not keep a static on-going way, but usually presents more or less important deviations as a result of service installation or reconfiguration, deployment of new equipment, and so on. In order to keep the knowledge representation model updated with potential variations in the normal behaviour of the target system, ESIDE-Depian uses the general sequential/incremental maximum likelihood estimates (Murphy, 2001) (in a continuous or periodical way) in order to achieve continuous adaptation of the model to potential changes in the normal behaviour of traffic.

2.2 Connection Tracking and Payload Analysis Bayesian Experts Knowledge Model Generation

The Connection Tracking expert attends to potential temporal influence among network events within TCP-based protocols (Estevez-Tapiador et al., 2003), and, therefore, it requires an structure that allows to include the concept of time (predecessor, successor) in its model. Similarly, the Payload Analysis expert, devoted to packet payload analysis, needs to model state transitions among symbols and tokens in the payload (following the strategy proposed in (Kruegel and Vigna, 2003)). Usually, Markov models are used in such contexts due to their capability to represent problems based on stochastic state transitions. Nevertheless, the Bayesian concept is even more suited since it not only includes representation of time (in an inherent manner), but also provides generalization of the classical Markov models adding features for complex characterization of states. Specifically, the Dynamic Bayesian Network (DBN) concept is commonly recognized as a superset of Hidden Markov Models (Ghahramani, 1998), and, among other capabilities, it can represent dependence and independence relationships between parameters within one common state (i.e. in the traditional static Bayesian style), and also within different chronological states.

Thus, ESIDE-Depian implements a fixed two-node DBN structure to emulate the Markov-Chain Model (with at least the same representational power and also the possibility to be extended in the future with further features) because full-exploded use of Bayesian concepts can remove several restrictions of Markov-based designs. For instance, it is not necessary to establish the first-instance structural learning process used by the packet header analysis experts since the structure is clear in beforehand.

Moreover, according to (Estevez-Tapiador et al., 2003; Kruegel and Vigna, 2003), the introduction of an artificial parameter may ease this kind of analysis. Respectively, the Connection Tracking expert defines an artificial detection parameter, named TCP-h-flags (which is based on an arithmetical combination of TCP flags) and the Payload Analysis expert uses the symbol and token (in fact, there are two Payload Analysis experts: one for token analysis and another for symbol analysis).

Finally, traffic behaviour (and so TCP flags temporal transition patterns) as well as payload protocol lexical and syntactical patterns may differ substantially depending on the sort of service provided from each specific equipment (i.e. from each different IP address and from each specific TCP destination port). To this end, ESIDE-Depian uses a multi-instance schema, with several Dynamic Bayesian Networks, one for each combination of TCP destination address and port. Afterwards, in the exploitation phase, Bayesian inference can be performed from real-time incoming network packets. In this case, the a-priori fixed structure suggests the application of the expectation and maximization algorithm (Murphy, 2001), in order to calculate not the posterior probability of attack, but the probability which a single packet fits the learned model with.

2.3 Naive Bayesian Network of the Expert Modules

Having different Bayesian modules is a two-fold strategy. On the one hand, the more specific expertise of each module allows them to deliver more accurate verdicts but, on the other hand, there must be a way to solve possible conflicting decisions. In other words, a unique measure must emerge from the diverse judgements.

To this end, ESIDE-Depian presents a two-tiered schema where the first layer is composed of the results from the expert modules and the second layer includes only one class parameter: the most conservative response among those provided by Snort and the expert modules community (i.e. in order to prioritize the absence of false negatives in front of false positives). Thus, both layers form, in fact, a Naive Bayesian Network (as shown in Fig. 1 and Fig. 2).

Such a Naive classifier (Castillo et al., 1997) has been proposed sometimes in Network Intrusion Detection, mostly for Anomaly Detection. This approach provides a good balance between representative power and performance, and also affords interesting flexibility capabilities which allow, for instance, ESIDE-Depian's dynamical enabling and disabling of expert modules, in order to support heavy load conditions derived e.g. from denial of service attacks.

Now, Naive Bayesian Network parameters should have a discrete nature which, depending on the expert, could not be the case. To remove this problem, ESIDE-Depian allows the using of the aforementioned set of administratively-configured threshold conditioning functions.

Finally, the structure of the Naive Bayesian Network model is fixed in beforehand, assuming the existence of conditional independence hypothesis among every possible cause and the standing of dependency edges between these causes and the effect or class. Therefore, here is also not necessary to take into consideration any structural learning process for it; only sequential parametric learning must be performed, while the expert modules produce their packet classifying verdicts during their respective parametric learning stages.

Once this step is accomplished, the inference of unified conclusions and the sequential adaptation of knowledge can be provided in the same way mentioned before. Fig. 2 details the individual knowledge models and how do they fit to conform the general one.

3 EVALUATION

In order to measure the performance of ESIDE-Depian, we have designed two different kinds of experiments. In the first group, the network suffers well-known attacks (i.e. Misuse Detection) and in the second group, zero-day attacks (i.e. Anomaly Detection), putting each aspect of the double nature of ESIDE-Depian to the test. In both cases, the system was fed with a simulation of network traffic comprising more than 700.000 network packets that were sniffed during one-hour capture from a University network. The first experiment (corresponding to Misuse Detection) aimed to compare Snort and the Packet Header Parameters Analysis experts. To this end, Snort's rule-set-based knowledge was used as the main reference for the labelling process, instantiated through Sneeze Snort-stimulator (Snort, 2006). The sample analysed was a mixture of normal and poisoned traffic. Table 1 details the results of this experiment.

Table 1: Bayesian expert modules for TCP, UDP and ICMP header analysis results.

Indicator	TCP	UDP	ICMP
Analyzed network packets	699.568	5.130	1.432
Snort's hits	38	0	450
ESIDE-Depian's hits	38	0	450
Anomalous network packets	600	2	45
False negatives	0	0	0
Potential false positives rates	0,08%	0,03%	3,14%

As it can be seen, the three experts achieved a 100% rate of hitting success. Anyway, such results aren't surprising, since ESIDE-Depian integrates Snort's knowledge and if Snort is able to detect an attack, ESIDE-Depian should do so. Nevertheless, not only the number of hits is important; the number of anomalous packets detected reflects the level of integration between the anomaly and the misuse detection part of ESIDE-Depian. In fact, the latter can be highlighted as the most important achievement of ESIDE-Depian: detecting unusual packets preserving the misuse detection advantages at the same time. Concerning potential false rates, it is possible to observe that very good rates are reached for TCP and UDP protocols (according to the values defined in (Crothers, 2002) to be not human-operator-exhausting), but not so good for ICMP. Table 1 shows, however, a significant bias in the number of attacks introduced in the ICMP traffic sample (above 30%), and labelled as so by Snort; thus, it is not strange the slightly excessive rate of anomalous packets detected here by ESIDE-Depian.

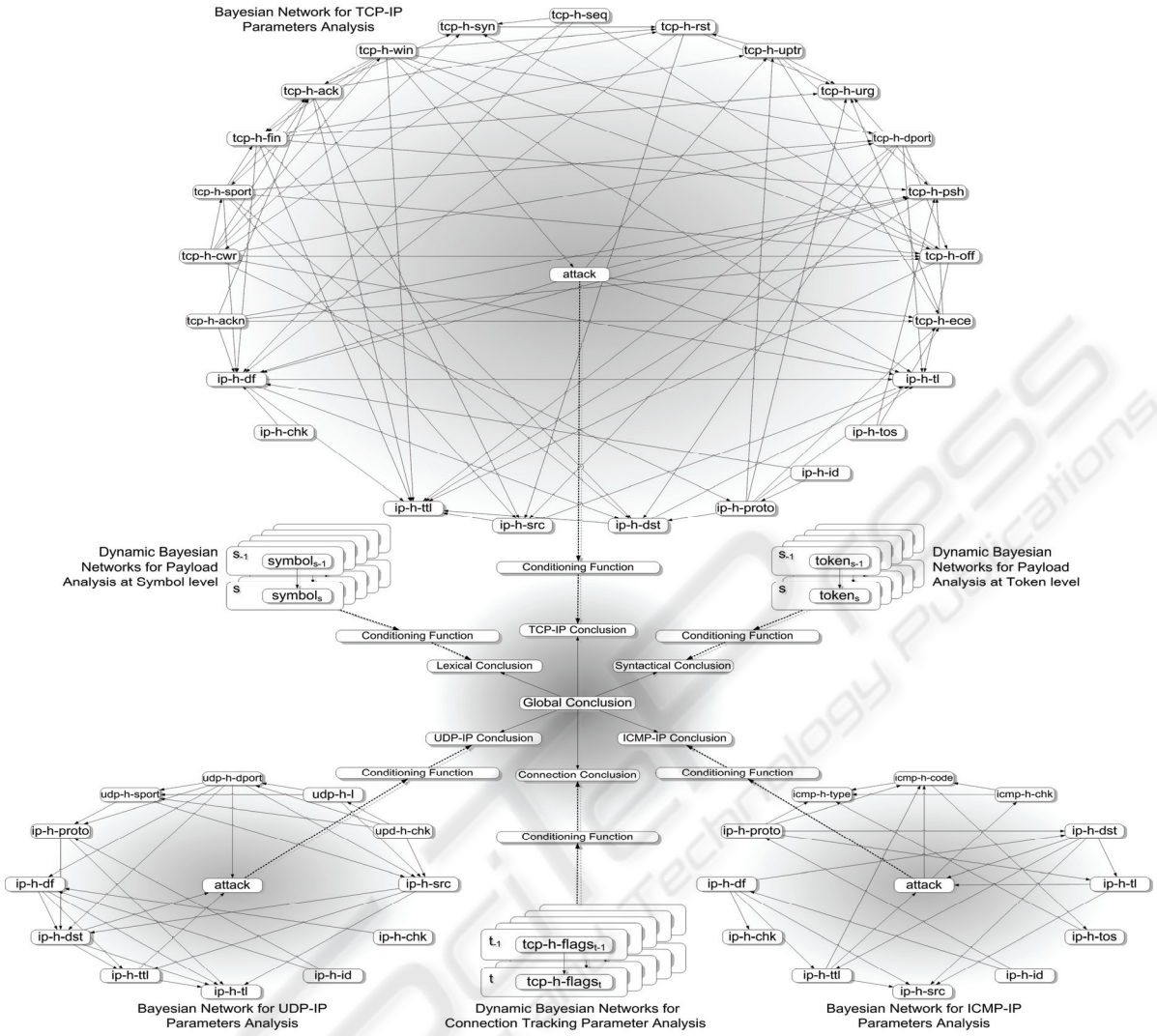

Figure 2: ESIDE-Depian Final Knowledge Representation Model.

In the second experiment (also corresponding to misuse detection), the goal was to test the other expert modules (Connection Tracking and Payload Analysis). With this objective in mind, a set of attacks against a representation of popular services were fired through several hacking tools such as (Metasploit, 2006). The outcome of this test is summarized in Table 2.

As we see, ESIDE-Depian prevailed in all cases with a 0% rate of false negatives and a 100% of hitting rate success. Still, not only Snort's knowledge and normal traffic behaviour absorption was tested; the third experiment intended to assess ESIDE-Depian's performance with zero-day attacks. With this idea in mind, a sample of artificial anomalies (Lee et al., 2001) was prepared with Snort's rule set as basis and crafted (by means of the tool Packit) with slight variations aiming to avoid Snort's detection (i.e. Zero-day attacks unnoticeable for misuse detection systems). Some of these attacks are detailed next. Table 3 shows the results of this experiment.

Table 2: Bayesian expert modules for connection tracking and payload analysis results.

Indicator	Connection Tracking	Symbol Analysis	Token Analysis
Analyzed network packets	226.428	2.676	2.676
Attacks in the sample	29	139	19
ESIDE-Depian's hits	29	139	19
Anomalous network packets	0	0	3
False negatives	0	0	0
Potential false positives rates	0,00%	0,00%	0,11%

Table 3: Example of Zero-Day attacks detected by ESIDE-Depian and not by Snort.

Protocol	Artificial Network Anomaly	Snort	ESIDE Depian
TCP	packit -nnn -s 10.12.206.2 -F SFP -d 10.10.10.100 -D 1023	✖	✓
TCP	packit -nnn -s 10.12.206.2 -F A -d 10.10.10.100 -q 1958810375	✖	✓
TCP	packit -nnn -s 10.12.206.2 -d 10.10.10.100 -F SAF	✖	✓
UDP	packit -t udp -s 127.0.0.1 -o 0x10 -n 1 -T ttl -S 13352 -D 21763 -d 10.10.10.2	✖	✓
UDP	packit -t udp -s 127.0.0.1 -o 0x10 -n 0 -T ttl -S 13353 -D 21763 -d 10.10.10.2	✖	✓
UDP	packit -t udp -s 127.0.0.1 -o 0x50 -n 0 -T ttl -S 13352 -D 21763 -d 10.10.10.2	✖	✓
ICMP	packit -i eth0 -t icmp -n 666 -K 0 -s 3.3.3.3 -d 10.10.10.2	✖	✓
ICMP	packit -i eth0 -t icmp -K 18 -C 0 -d 10.10.10.2	✖	✓
ICMP	packit -i eth0 -t icmp -K 17 -C 0 -d 10.10.10.2	✖	✓

Note that overcoming of Snort's expert knowledge only has sense in those expert modules using this knowledge. This is, in protocol header specialized modules, because the semantics of Snort's labelling doesn't fit the morphology of payload and dynamic nature parameters.

4 CONCLUSIONS AND FUTURE LINES

As the use of Internet grows beyond all boundaries, the number of menaces rises to become subject of concern and increasing research. Against this, Network Intrusion Detection Systems monitor local networks to separate legitimate from dangerous behaviours. According to their capabilities and goals, NIDS are divided into Misuse Detection Systems (which aim to detect well-known attacks) and Anomaly Detection Systems (which aim to detect zero-day attacks). So far, no system to our knowledge combines advantages of both without any of their disadvantages. Moreover, the use of historical data for analysis or sequential adaptation is usually ignored, missing in this way the possibility of anticipating the behaviour of the target system.

Our system addresses both needs. We present here ESIDE-Depian, a Bayesian-networks-based Misuse and Anomaly Detection system. Our approach integrates Snort as Misuse detector trainer so the Bayesian Network of five experts is able to react against both Misuse and Anomalies. The Bayesian Experts are devoted to the analysis of different network protocol aspects and obtain the common knowledge model by means of separated Snort-driven automated learning process. A naive Bayesian network integrates the results of the experts, all the partial verdicts achieved by them.

Since ESIDE-Depian has passed the experiments brilliantly, it is possible to conclude that ESIDE-Depian using of Bayesian Networking concepts allows to confirm an excellent basis for paradigm unifying Network Intrusion Detection, providing not only stable Misuse Detection but also effective Anomaly Detection capabilities, with one only flexible knowledge representation model and a well-proofed inference and adaptation bunch of methods.

On the other hand, the Bayesian approach also enables to implement powerful features over it, such as Dynamic-Bayesian-Network-based intrinsic full representation of time, in order to accomplish totally-characterised connection tracking and low-level chronological event correlation, or explanation tracking of the inferred cause-effect reasoning processes. Furthermore, contrary to other approaches such as Neural Networks, Bayesian networks allow administrative managing of inner information structures, so specific relationships among packet detection parameters and final conclusion can be explained, in a white-box manner. Moreover, it is not only possible to recover reasoning information, but also to act on both Bayesian network structures and conditional probability parameters, in order to adjust the whole behaviour of the Network Intrusion Detection System to special needs or configurations.

Further, dynamic regulation of knowledge representation model can be accomplished by using the sensibility analysis proposed in (Castillo et al., 1997), so as to avoid denial of service attacks, automatically enabling or disabling expert modules by means of one combined heuristic measure which considers specific throughputs and representative power. In addition, it is also possible to perform model optimization, to obtain the minimal set of representative parameters, and also the minimal set of edges among them, with the subsequent increase of the general performance.

Approximate evidence propagation methods can also be applied in order to improve inference and adaptation time of response. Current expert models only consider exact inference, but it is possible to find methods which provide fast responses, with only a small and affordable loss of accuracy.

Finally, Bayesian knowledge representation models present one further interesting capability in current Network Intrusion Detection state of art, the possibility to provide an ad-hoc method for NIDS evaluation. The Bayesian concept provides simulation of learned knowledge corresponding samples, so it is an ideal environment for artificial anomaly generation.

Future work will focus on further research on exploiting the aforementioned omni-directional inference capability of Bayesian networks to the prediction of the next event, as well as on comparing ESIDE-Depian to other cutting-edge Intrusion Detection Systems.

ACKNOWLEDGEMENTS

The authors would like to thank the Regional Government of Biscay and the Basque Goverment for their financial support.

REFERENCES

Alipio, P., Carvalho, P., Neves, J., 2003. Using CLIPS to Detect Network Intrusion. *Lecture Notes in Computer Science*, volume 2902/2003, pages 341-354, ISBN 0302-9743, Springer-Verlag.

Brugger, T., 2004. Data Mining Methods for Network Intrusion Detection. *PhD thesis*. University of California Davis.

Castillo, E., Gutierrez, J.M., Hadi, A. S., 1997. *Expert Systems and Probabilistic Network Models*. ISBN: 0-387-94858-9. Springer-Verlag.

Chavan, S., Shah, K., Dave, N., Mukherjee, S., Abraham, A., Sanyal, S., 2004. Adaptative neuro-fuzzy intrusion detection systems. *Proceedings of the 2004 International Conference on Information Technology: Coding and Computing*, volume 1, pages 70-74.

Crothers, T., 2002. *Implementing Intrusion Detection Systems: A Hands-On Guide for Securing the Network*. ISBN 0764549499, John Whiley & Sons Inc.

Doyle, J., Kohane, I., Long, W., Shrobe, H., Szolovits, P., 2001. Event recognition beyond signature and anomaly. *Proceedings of the 2001 IEEE Workshop on Information Assurance and Security*, pages 170-174.

Estevez-Tapiador, J., Garcia-Teodoro, P., Diaz-Verdejo, J., 2003. Stochastic protocol modelling for anomaly based network intrusion detection. *Proceedings of the first IEEE International Workshop on Information Assurance*, pages 3-12.

Ghahramani, Z., 1998. Learning Dynamic Bayesian Networks. *Lecture Notes in Computer Science*, volume 1387, page 168. Springer-Verlag.

Helmer, G., Wong, J., Honavar, V., Miller, L., Wang, Y., 2003. Lightweight agents for intrusion detection. *Journal of Systems and Software*, volume 67, pages 109-122.

Internet System Consortium, 2007. *Internet Domain Survey*. July 2007. Available at http://www.isc.org/.

Kabiri, P., Ghorbani, A. A., 2005. Research on intrusion detection and response: A survey. *International Journal on Information Security*, volume 1(2), pages 84-102.

Kantzavelou, I., Katsikas, S., 1997. An attack detection system for secure computer systems outline of the solution. *Proceedings of the IFIP TC11 13th International Conference on Information Security*, pages 123-135.

Kim, D., Nguyen, H., Park, J., 2005. Genetic algorithm to improve svm-based network intrusion detection system. *Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA)*, volume 2, pages 155-158.

Kruegel, C., Vigna, G., 2003. Anomaly detection of web-based attacks. *Proceedings of the 10th ACM Conference on Computer and Communications Security*, pages 251-261.

Lazarevic, A., Ertoz, L., Kumar, V., Ozgur, A., Srivastava, J., 2003. A comparative study of anomaly detection schemes in network intrusion detection. *Proceedings of the SIAM International Conference on Data Mining*.

Lee, W., Stolfo, S., Chan, P., Eskin, E., Fan, W., Miller, M., Hershkop, S., Zhang., J., 2001. Real time data mining-based intrusion detection. *Proceedings of the second DARPA Information Survivability Conference and Exposition*, volume 1, pages 89-100.

Metasploit, 2006. *Exploit research*. Available at http://www.metasploit.org/.

Mukkamala, S., Sung, A., Abraham, A., 2005. Intrusion detection using an ensemble of intelligent paradigms. *Journal of Network and Computer Applications*, volume 28, pages 167-182.

Murphy, K., 2001. An introduction to graphical models. *Technical report*. Intel Research, Intel Corporation.

Roesch, M. (1999). SNORT: Lightweight intrusion detection for networks. *Proceedings of LISA99: 13th Systems Administration Conference*, pages 229-238.

Singhal, A., Jajodia, S., 2006. Data warehousing and data mining techniques for intrusion detection systems. *International Journal on Information Security*, volume 1(2), pages 149-166.

Snort, 2006. *The facto standard for intrusion detection and prevention*. Available at http://www.snort.org/.

Spirtes, P., Glymour, C., Scheines, R., 2001. Causation, Prediction, and Search, Second Edition. *Adaptive Computation and Machine Learning*. The MIT Press.

Valdes, A., Skinner, K., 2000. Adaptive, model-based monitoring for cyber attack detection. *Proceedings of RAID 2000*, pages 80-92.

Vigna, G., Eckman, S., Kemmerer, R., 2000. The STAT tool suite. *Proceedings of the DARPA Information Survivability Conference and Exposition 2000*, volume 2, page 1046. IEEE Press.