Optimal Bayesian Network Design for Efficient Intrusion Detection

Igor Ruiz-Agundez, Yoseba K. Penya and Pablo Garcia Bringas
DeustoTech
University of Deusto
Bilbao, Basque Country
igor.ira, yoseba.penya, pablo.garcia.bringas@deusto.es

***Abstract*—Computer networks are nowadays subject to an increasing number of attacks. Intrusion Detection Systems (IDS) are designed to protect them by identifying malicious behaviours or improper uses. Since the scope is different in each case (register already-known menaces to later recognise them or model legitimate uses to trigger when a variation is detected), IDS have failed so far to respond against both kind of attacks. Lately, Bayesian networks (BN) have provided an innovative solution to fill this gap by integrating both domains within a common knowledge representation model. Still, the huge computational effort that has to be invested in designing and training the BN with such knowledge model makes them not feasible and not practical for real-world scenarios. Against this background, we propose the use of expert knowledge to enhance and optimise the design of the Bayesian network, shortening subsequently the training process. This expert knowledge is represented as a set of hypotheses that must be verified to justify their utility. In this way, we have tested our approach with several samples of data showing that all the hypotheses assumed were true and, therefore, that the proposed methodology to trim down the design and training processes yields an optimal Bayesian network for Intrusion Detection.**

I. Introduction

According to the estimations of the Internet System Consortium, nowadays more than 489 million computers are connected to the biggest network in the world (Internet System Consortium, 2009). Being part of such a vast community brings amazing possibilities but also worrying dangers. Overwhelmed by a record-braking growth of the 52% in the last decade, traditional passive measures for isolation and access control are simply inadequate to dam the increasing flood of digital attacks and intrusion attempts.

In this way, Intrusion Detection Systems (IDS) have become a substantial part of computer security since they help to protect from the wave of intrusion attempts. Based on their scope, IDS can be divided into misuse and anomaly-based detection. Misuse detection is chronologically the first approach and is based on a well-defined corpus of malicious behaviours in order to find such patterns in the supervised system. Anomaly Detection, on the contrary, focuses on not yet documented menaces since it models legitimate usage to obtain afterwards a certainty measure of potential deviations from that normal profile. Therefore, misuse detection has proved to be a very good at finding well-known attacks and anomaly detection at alerting against unknown ones. Unfortunately, they fail when applied to the other's natural role [1].

In this way, in previous works we have presented the first unified misuse and anomaly prevention system based on Bayesian networks to fully analyse network packets. It is able to simultaneously offer efficient response against both well-known and zero-day attacks. In order to ease the way to this goal, our system was conceived and deployed in a modular way that allowed decomposing of the problem into several smaller units. Still, the design and training process of the Bayesian network (BN) demanded huge computational efforts that prevented it from being applied in the real world. In order to face this constrain, the use of expert knowledge is proposed. Expert knowledge has been previously used in data mining classifiers [2], for normalizing and binning gene expression data in BN [3] or for generalized partition testing via Bayes linear methods [4].

Against this background, we advance the state of the art in two main ways. First, we present, for the first time, a methodology to enhance the design of the BN (and, thus, shorten the training process) by using expert knowledge. Second, we detail a knowledge representation model applicable to this problem domain based on independent cause-consequence hypotheses.

The remainder of the paper is structured as follows. Section II describes the general architecture of the system and presents the potentials and limitations of the use of Bayesian networks for Intrusion Detection Systems. Section III introduces the knowledge representation model, including the proposed hypotheses. Section IV describes the experiments carried out to verify the hypotheses and discusses their results. And, finally, section V concludes and outlines the avenues of future work.

II. Potential and limitations of the system architecture

Bayesian networks are probabilistic models for multivariate analysis. Formally, they are directed acyclic graphs associated to a probability distribution function [5]. Nodes in the graph represent variables (any kind, be it a premise or a conclusion), and the arcs, conditional dependencies between such variables. Further, the probability function illustrates

the strength of these relationships (i.e. arcs or edges) in the graph.

According to our needs, the most important ability of a BN is its capability to infer the probability with which a certain hypothesis becomes true out of the values that the rest of variables forming the BN take. In this way, we have divided the network traffic according to its type (TCP-IP, UDP-IP and ICMP-IP) and created three Bayesian networks (experts) in charge of analysing their respective packet headers (which is an strategy already proven successful in this area [6]). Moreover, in order to cover all possible kinds of menaces, we also have to take into account the payload (i.e. body) of the packet and the potential temporal dependencies between packets. Therefore, we have added two further experts, the protocol payload expert and the connection tracking expert, respectively. The division in temporal steps and in different traffic types, allow us to decrease the computational requirements of the BN. Section III-B describes this process in greater detail.

In each case, the Bayesian network is composed of several variables depending on the protocol and the expert; the value to be induced is always the probability that the analysed packet is part of an attack. See [1] for a more accurate description of the Bayesian experts.

Bayesian networks generally need two learning steps to be ready to infer results. The first one is the *structural learning* process (detailed in section III-D) that obtains the probability distribution table associated to each variable. The second one is the *parametric learning* process (detailed in section III-H) that refines the initial graph. Finally, the system uses a Naïve Bayesian network to unify the different experts providing an excellent balance between knowledge representation capacity and performance. It assumes the existence of conditional independence hypotheses within every possible cause and the standing of dependency edges between these causes and the effect or class applicable to this problem domain. These hypotheses are the representation of the experts knowledge that tunes the Bayesian network design and training, creating the optimal network. Section III-C presents the hypotheses for optimal BN design, reducing the computational needs of the system, for efficient intrusion detection.

The fact that the system is built upon several Bayesian networks implies a huge learning cost in terms of computational requirements, both in processing capacity and available memory. In this way, table I shows the estimated time that would take to perform the structural learning process for the data set used in the experiment. Each fragment of the data is an outfit consisting of 10.000 network packets requiring 7 structural learning processes, one for each significance level. The time of accomplishment is estimated from the initial experiments, performed in an Intel Pentium IV with 512 Megabytes of main memory. Each structural learning process for each fragment of data requires an average of three hours to complete. With this drawback in mind, we have followed a conducted-learning approach thanks to the adoption of expert knowledge. This expert knowledge is represented by the hypothesis of conditional dependency or independence between BN-s and is our proposed method to face the computational requirements of the learning processes.

Table I
PLANNING OF THE STRUCTURAL LEARNING PROCESS

Set of Data	Packets	Fragments	Learning Processes	Estimated Time
Complet evidence set	843.806	85	595	1.785 h
TCP-IP evidence set	837.058	84	588	1.764 h
UDP-IP evidence set	5.197	1	7	21 h
ICMP-IP evidence set	1.543	1	7	21 h
Total	1.687.604	171	1.197	3.591 h

The expert knowledge can give a certain value to the edges of a Bayesian network [7]. This is the least costly procedure since at the time of performing experiments an expert is present and no extra cost is needed to obtain further knowledge. We use expert knowledge to reduce the cost of obtaining knowledge for the representation model. This expert knowledge is represented as the hypotheses detailed in section III-C. We understand expert knowledge as those characteristics, skills and knowledge of a person, which distinguish experts from novices and less experienced people.

Taking all the above into account, these hypotheses have direct influence on the complexity of the Bayesian networks in the structural learning process and in the inference of the results. In this way, the reduction of the complexity in the Bayesian network implies a lower expressive capacity. Nevertheless, these reductions are curried out according to expert knowledge and hence, the impact in the knowledge representation expressiveness is worthless as we will show.

III. KNOWLEDGE REPRESENTATION MODEL

Keeping the system described in section II in mind, we have designed a knowledge representation model in order to face the limitations of the system architecture and develop its potential. This section is devoted to detail the whole process of the structural learning process and presents the expert knowledge applied to IDSs.

A. Obtaining the sample data

The step, consists on obtaining the evidential sample set of data. In order to get good results, the quality of the network traffic is crucial. It is represented in a set of data that is the input to the module of the structural learning and the result of this process is directly linked to the input. Hence, the data set is obtained from a real background as accomplished in other methodologies [8] [9]. More accurately, the data

set was fed with a simulation of network traffic comprising more than 700.000 network packets that were sniffed during a one-hour capture from a University network.

B. Progressive incorporation of temporal steps to the dynamic Bayesian network

Each time a new temporal step is added to the Bayesian network the computational requirements increase notably because the complexity of the network increases [10] [11] [12]. To avoid these computational requirements, we use an iterative methodology. This methodology allows to progressively add temporal steps into the Dynamic Bayesian networks.

Dynamic Bayesian networks allow us to represent the temporal magnitudes as well as to model conditional dependencies and independences of the events that took place in different times and infer conclusions by using this model [13] [10] [11] [12].

Each temporal magnitude is represented in a temporal step, increasing the capacity of the Bayesian network to remember the influence of each registered event on the past with events that took place later on. This features enables the projection of this influence into the future. The use of such temporal steps in our model allows us to follow sequential network events [14].

In the present study, the initial model approximation to the knowledge representation is based on only one temporal step. Later on, once the required experiments to get the initial approximation are done, more temporal steps will be incorporated.

C. Establishing the hypothesis of dependence and independence

At this point we can already add the expert knowledge. Due to the fact that the structural learning methods are able to infer by themselves the relations of dependence and independence of the structure, expert knowledge can refine the resulting model and, hence, optimise the exploitation of the Bayesian network [15] [16].

As mentioned in section II, we use hypothesis of dependence and independence to refine our knowledge representation model. In particular, the hypotheses are based on the specific issues of four network protocols (IP, ICMP, TCP and UDP). The expert knowledge is based on the following six hypotheses:

- **Hypothesis 1: Dependence between TCP and IP.** The set of the detection parameters of the TCP protocol is dependent of the set of the detection parameters of the IP, and vice versa.
- **Hypothesis 2: Dependence between UDP and IP.** The set of the detection parameters of the UDP protocol is dependent of the set of the detection parameters of the IP, and vice versa.
- **Hypothesis 3: Dependence between ICMP and IP.** The set of the detection parameters of the ICMP protocol is dependent of the set of the detection parameters of the IP, and vice versa.
- **Hypothesis 4: Independence between TCP and UDP.** The set of the detection parameters of the TCP protocol is dependent of the set of the detection parameters of the UDP, and vice versa.
- **Hypothesis 5: Independence between TCP and ICMP.** The set of the detection parameters of the TCP protocol is independent of the set of the detection parameters of the ICMP, and vice versa.
- **Hypothesis 6: Independence between UDP and ICMP.** The set of the detection parameters of the UDP protocol is independent of the set of the detection parameters of the ICMP, and vice versa.

These hypotheses are supported by the respective set of *Request For Comments (RFC)* of each protocol. An empirical demonstration of them is done demonstrating that the knowledge representation model generated from them can be successfully used in the reasoning engine.

Moreover, the heterogeneity of the detection parameters headers (information used by the protocols), and data (information used by the users) themselves implies a different formalization for each case. The analysis model is static (based on normal Bayesian networks) in the case of the head parameters and dynamic (based on Dynamic Bayesian networks) in the case of data parameters. The first group forms an information entity and it is, therefore, susceptible of being used directly in the process of analysis. On the other hand, the second group represents a variable flow of information entities in both lexical and a syntactic levels that requires a specific analysis. Considering all this, another hypothesis is pointed out:

- **Hypothesis 7: Independence between head and data fields.** The set of detection parameters corresponding to the head fields of IP, ICMP, TCP and UDP protocols is independent from the data fields of the corresponding protocols, and vice versa.

Finally, there is an additional aspect to consider. Since for this experiment only one time step is considered, it is not possible to include the second evidence required by the dynamic model. Please note that if more temporal steps are added this restriction disappears.

- **Hypothesis 8: Independence between static and dynamic parameters.** In the case of one temporal step Dynamic Bayesian networks, the set of detection parameters used in the static analysis methods are independent from those used in the dynamic analysis methods, and vice versa.

The specification of the previous hypotheses of dependence and independence defines separated working areas, in which different analysis methods will be applied depending

on the type of the detection parameter.

On one hand, we have the head parameters of all the protocols that can be treated in a homogeneous way. These cases can be introduced straightforward into the structural learning process. On the other hand, each protocol data has its own properties and therefore has to be resolved in an independent way. In the case of dynamic parameters, multiple evidences are required, and hence, they will have an independent treatment too.

D. The structural learning process

The previous step has defined the different fields of actuation based on the given hypotheses. Now, it is time to plan how the sample data set will be introduced in the structural learning process. This step will sort out the structural learning process for the different learning needs:

- Planning the process of structural learning for the protocol head parameters
- Planning the process of structural learning for the protocol data parameters
- Planning the process of structural learning for dynamic parameters

As it was outlined before, structural learning allows the modelling, in a completely automated way, of the set of dependence and independence relationships that can reside among the different detection parameters. Thus, it is also possible to proceed with further stages to be able to inference conclusions.

Nevertheless, for situations in which the volume of evidences is very big and the detection parameters also show large numbers of different possible states, the learning PC-Algorithm (and also other similar alternative methods) presents very high computational requirements [17]. Besides, depending on the inner complexity of the set of relationships, those requirements can grow even more; that complexity depends completely on the reality of data and therefore is so far unpredictable.

Thus, the high demands on memory and computer power of this method may restrict its application on limited-power computing platforms.

Against this problem, we have developed a specific methodology for horizontal splitting of the traffic sample and, consequently, of the subsequent structural learning process. This splitting not only allows the achievement of the desired results, but also enriches the research methodology, allowing the researchers to carry their work to the limit of the computational power available.

In the case of our experiment, only the head parameters of the different protocols (TCP, UDP, ICMP and IP) can be fractionated during the structural learning process as pointed in section III-C. Furthermore, in order to validate the hypothesis of dependence and independence, the whole sample data set is used in an additional structural learning process.

Please note that structural learning methods commonly use a significance parameter. That helps to define, in a flexible manner, the strength with which a relationship is definitely considered of dependence. In this way, the significance parameter can be used to make the concept of equality needed for the independence tests that are implemented inside the learning algorithms relative (in particular, inside the PC Algorithm). On one hand, a high significance value increments the number of connections in the Bayesian model, and the degree of representativeness, implies larger requirements in terms of main memory and computational power that may produce over-fitting. On the other hand, a low significance value generally results into a sparse Bayesian network, with lower requirements, but also much lower semantic power.

Keeping in mind the objective of finding a trade-off between representative capacity and system performance, we have expanded the structural learning process in multiple significance levels. More accurately, it consists of seven different levels and each of them will have the sample data set as an input.

Once the planning of the learning process has been done, the computational work itself can start. Considering the size of the sample data and the several significance levels, the structural learning process has to be applied several times generating a large set of partial Bayesian networks. These networks will be unified in a later step.

Besides, the presented hypotheses suggest the introduction of four sets of data from the sample data. On one hand, three sets of data corresponding to the head parameters of the protocols (TCP and IP, UDP and IP, ICMP and IP); and, on the other hand, a fourth set containing the whole evidence sample data. Table I summarizes the size of the different sets of data used in the experiment and the resulting fragments, the number of structural learning processes that took place, and the estimated time required for each analysis.

As it can be seen, the learning PC-Algorithm yields a set of 1197 partial BN, which have to be unified as described in the following sections. The resulting BN will represent the real set of relations of dependence and independence between the different detection parameters.

E. Unifying and adapting the partial results

In order to achieve the unification of the partial BNs, we have defined a statistical metrics from the partial Bayesian networks. This metric will consider the frequency of repetition in the partial structures in the corresponding significance levels of each relation between two different detection parameters. Table II shows a part of this statistics metrics.

The next step is to obtain one unique BN structure for each level in which each link is pondered depending on the number of times that appears in the partial structures. Once the unification of the different significance levels is done

Table II
STATISTICS OF LINKS FREQUENCY BETWEEN VARIABLES IN THE WHOLE EVIDENCE SET

child \parent	attack	icmp -h-chk	icmp -h-code	icmp -h-type	...
attack	-	4,16	-	-	...
icmp-h-chk	4,16	-	-	-	...
icmp-h-code	8,33	16,66	-	-	...
icmp-h-type	-	58,30	8,33	-	...
...	...	...	...	...	...

for each set of data, it is possible to proceed with the final unification.

This unification is achieved by finding the average value of the different percent's for each link between variables; hence, we obtain an unique and pondered structure of the Bayesian network. Furthermore, this unification provides a balance between representative capacity and performance.

On the other hand, this process introduces a negative effect into the knowledge representation model due to the fractionated learning process. During the unification process of the partial Bayesian networks, direct loops may appear, contravening the definition of Bayesian networks [18].

In order to avoid this phenomenon, once the unification process is performed, we require an additional model adapting process. The average value previously obtained will be used to eliminate the weakest links, and hence, eliminate the direct loops, if they exist.

The final result will be one Bayesian network for each set of data planned in the structural learning process. The Bayesian networks correspond to the evidences from UDP-IP, TCP-IP and ICMP-IP. At this point, the networks are ready for exploitation doing either parametric learning or inference of conclusions [19] [20] [21] and adaptation [22] [23].

F. Verifying the hypotheses of dependence and independence

Once the Bayesian network models that will be used by the reasoning engine are obtained, it is compulsory to verify the proposed hypotheses of dependence and independence. It is possible to prove the validity of the hypotheses by using the result sets of the structural learning process. The high complexity of this phase suggests a further description that is accomplished in section IV.

The final knowledge representation model can be built straight forward from the obtained Bayesian networks, since the results confirm the hypothesis of dependence and independence

G. Structural definition of the representation model of knowledge

The final knowledge representation model consists in:

- A Bayesian network representing the TCP-IP refined evidence
- A Bayesian network representing the UDP-IP evidence
- A Bayesian network representing the ICMP-IP evidence
- A Dynamic Bayesian network representing the analysis of the protocol parameters at a lexical level
- A Dynamic Bayesian network representing the analysis of the data parameters at a syntactical level
- A Bayesian network representing the ICMP-IP evidence
- A Dynamic Bayesian network representing the analysis of the dynamic parameters

Each knowledge model is isolated according to the problem hypotheses. This model of independence implies that the inference process will produce several conclusions. In order to fight over this issue, we need an additional component that joins all the conclusions.

The component used to gather up all the partial conclusions into a unique conclusion is a Naïve Bayesian network, which is an standard solution [1] for this kind of situations. This model points out an excellent relation between representation capacity and performance in evidence classification tasks [15] [24] [11].

H. The parametric learning process

The knowledge model fixed so far is qualitative. 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, we have implemented a maximum likelihood estimate [10] 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. Namely, structural learning finds the structure of probability distribution functions between detection parameters and parametric learning fills this structure with proper conditional probability values [1].

IV. EVALUATION AND RESULTS

In order to assess the validity of the work hypotheses described in section III-C, we have performed different sorts of experiments. We start our experiments from the knowledge representation model made up of different Bayesian networks that form the reasoning engine. From then on, and considering the hypotheses of dependence and independence, we analyse the obtained results of the structural learning process. As the results confirm the hypotheses of dependence and independence, the *RFC* specifications of each protocol are ratified. Finally, we build a knowledge representation model based on the different Bayesian networks.

Taking that modus operandi into account, and with the objective of minimising the possible appearance of noise in the results that could affect the final conclusions about

the hypotheses, we have set a threshold value. Above this threshold value a link between two variables will not be representative. According to our methodology, two protocols will be independent *if and only if* there are no representative relations between the set of parameters of either of them. The hypotheses of dependence and independence are proved to be true:

- **Hypothesis 1: Dependence between TCP and IP.** Table III show the relations between the parameters of TCP and IP, verifying that there are many significant links between the corresponding detection parameters of each protocol, in both ways. Therefore, there are variables of the TCP protocol Bayesian network that depend on variables of the IP protocol, and vice versa. Hence, the hypothesis of dependence between TCP and IP is confirmed.
- **Hypothesis 2: Dependence between UDP and IP.** Equally, as table IV show the data of the experiment points out the relation between the head parameters of the UDP and IP protocols. There are enough significant links between the detection parameters of both protocols, in both ways. Therefore, there are variables of the UDP protocol in the Bayesian network that depend on variables of the IP protocol, and vice versa. Hence, the hypothesis of dependence between UDP and IP is also confirmed.

Table IV
FREQUENCY OF LINKS APPEARANCE IN THE BAYESIAN NETWORK FOR RELATIONS OF DEPENDENCE OF UDP PARAMETERS OVER IP PARAMETERS

child\parent	udp-h-chk	udp-h-l	udp-h-dport	udp-h-sport
ip-h-dst	-	-	1,61	-
ip-h-src	9,65	1,02	3,88	1,79
ip-h-proto	-	-	2,04	3,65
ip-h-tll	-	-	-	-
ip-h-df	-	-	1,02	-
ip-h-id	-	-	-	-
ip-h-tl	-	-	-	-
ip-h-tos	-	-	-	-
ip-h-hl	-	-	-	-

- **Hypothesis 3: Dependence between ICMP and IP.** Table V show the case of ICMP and IP protocols, the data of the experiment points out the relation between the head parameters of both protocols. There are enough significant links between the detection parameters, in both ways. Therefore, there are variables of the ICMP protocol in the Bayesian network that depend on variables of the IP protocol, and vice versa. Hence, the hypothesis of dependence between ICMP and IP is confirmed similarly.
- **Hypothesis 4: Independence between TCP and UDP.** Table VI show the hypothesis of independence between TCP and UDP, the data of the experiment points out the independence between the detection parameters of both protocols, in none of both ways. There are not enough significant links between the detection parameters, in none of both ways. Therefore, there are not variables of the TCP protocol that depend on the variables of the UDP protocol, and vice versa. Hence, the hypothesis of independence between TCP and UDP is also verified.

Table V
FREQUENCY OF LINKS APPEARANCE IN THE BAYESIAN NETWORK FOR RELATIONS OF DEPENDENCE OF ICMP PARAMETERS OVER IP PARAMETERS

child\parent	icmp-hchk	icmp-hcode	icmp-htype
ip-h-dst	-	-	-
ip-h-src	-	-	-
ip-h-proto	-	2,90	-
ip-h-tll	-	-	-
ip-h-df	-	-	-
ip-h-id	-	-	-
ip-h-tl	-	-	-
ip-h-tos	-	8,94	-
ip-h-hl	-	-	-

Table VI
FREQUENCY OF LINKS APPEARANCE IN THE BAYESIAN NETWORK FOR RELATIONS OF DEPENDENCE OF UDP PARAMETERS OVER TCP PARAMETERS

child\parent	udp-hchk	udp-h-l	udp-h-dport	udp-h-sport
tcp-h-uptr	0,49	-	2,86	-
tcp-h-win	-	-	-	-
tcp-h-cwr	-	-	-	-
tcp-h-ece	0,59	-	2,63	1,79
tcp-h-psh	-	1,79	1,79	0,59
tcp-h-urg	1,02	-	1,57	4,11
tcp-h-ack	-	-	-	-
tcp-h-rst	-	-	-	1,79
tcp-h-syn	-	-	-	-
tcp-h-fin	-	-	-	-
tcp-h-off	-	-	-	-
tcp-h-ackn	-	-	-	-
tcp-h-seq	-	-	-	-
tcp-h-dport	-	-	-	-
tcp-h-sport	-	-	-	-

- **Hypothesis 5: Independence between TCP and ICMP.** Similarly, table VII and table **??** show that the data of the experiment points out the independence between the detection parameters of TCP and ICMP protocols, in any way. There are not enough significant links between the detection parameters, in any way. Therefore, there are not variables of the TCP protocol that depend on the variables of the ICMP protocol, and vice versa. Hence, the hypothesis of independence between TCP and ICMP is also proved.
- **Hypothesis 6: Independence between UDP and ICMP.** Finally, in table VIII and table IX, the data of the experiment points out the independence between the detection parameters of UDP and ICMP protocols, in anyway. There are not enough significant links

Table III
FREQUENCY OF LINKS APPEARANCE IN THE BAYESIAN NETWORK FOR RELATIONS OF DEPENDENCE OF IP PARAMETERS OVER TCP PARAMETERS

child\parent	ip-h-dst	ip-h-src	ip-h-proto	ip-h-tll	ip-h-df	ip-h-id	ip-h-tl	ip-h-tos	ip-h-hl
tcp-h-uptr	-	-	15,61	-	-	-	-	-	-
tcp-h-win	2,40	2,09	-	2,89	4,46	-	13,55	-	-
tcp-h-cwr	-	0,49	-	-	-	0,43	31,32	-	-
tcp-h-ece	-	-	11,08	-	-	-	-	2,33	-
tcp-h-psh	-	-	6,63	0,71	-	-	-	1,38	-
tcp-h-urg	-	-	12,43	-	-	-	-	-	-
tcp-h-ack	-	-	-	-	-	-	2,17	-	-
tcp-h-rst	-	-	1,31	1,08	1,92	-	-	-	-
tcp-h-syn	0,71	-	-	-	1,79	-	-	-	-
tcp-h-fin	-	-	-	1,02	-	-	-	-	-
tcp-h-off	-	-	-	1,98	-	-	28,74	-	-
tcp-h-ackn	-	-	-	-	-	-	-	-	-
tcp-h-seq	-	-	-	-	1,74	-	-	-	-
tcp-h-dport	3,84	1,08	-	-	-	-	9,04	-	-
tcp-h-sport	8,01	3,57	-	-	-	-	8,40	-	-

Table VII
FREQUENCY OF LINKS APPEARANCE IN THE BAYESIAN NETWORK FOR RELATIONS OF DEPENDENCE OF ICMP PARAMETERS OVER TCP PARAMETERS

child\parent	icmp-hchk	icmp-hcode	icmp-htype
tcp-h-uptr	-	2,86	-
tcp-h-win	-	-	-
tcp-h-cwr	-	-	-
tcp-h-ece	-	-	-
tcp-h-psh	-	0,59	-
tcp-h-urg	-	3,52	-
tcp-h-ack	-	-	-
tcp-h-rst	-	-	-
tcp-h-syn	-	-	-
tcp-h-fin	-	0,49	-
tcp-h-off	-	-	-
tcp-h-ackn	-	-	-
tcp-h-seq	-	-	-
tcp-h-dport	-	-	-
tcp-h-sport	-	-	-

between the detection parameters, in none of both ways. Therefore, there are not variables of the UDP protocol that depend on the variables of the ICMP protocol, and vice versa. Hence, the hypothesis of independence between UDP and ICMP is also verified.

Table VIII
FREQUENCY OF LINKS APPEARANCE IN THE BAYESIAN NETWORK FOR RELATIONS OF DEPENDENCE OF ICMP PARAMETERS OVER UDP PARAMETERS

child\parent	icmp-h-chk	icmp-h-code	icmp-h-type
udp-h-chk	-	-	-
udp-h-l	-	-	-
udp-h-dport	-	-	-
udp-h-sport	-	-	-

The validity of the hypotheses 7 and 8 is already proved by their set out. This is, the set of detection parameters corresponding to the head fields of IP, ICMP, TCP and UDP protocols is independent from the data fields of the corresponding protocols, and vice versa. In the case of one

Table IX
FREQUENCY OF LINKS APPEARANCE IN THE BAYESIAN NETWORK FOR RELATIONS OF DEPENDENCE OF UDP PARAMETERS OVER ICMP PARAMETERS

child\parent	udp-h-chk	udp-h-l	udp-h-dport	udp-h-sport
icmp-h-chk	-	-	-	-
icmp-h-code	-	-	-	-
icmp-h-type	-	-	-	-

temporal step dynamic BN, the set of detection parameters used in the static analysis methods are independent from those used in the dynamic analysis methods, and vice versa.

Since the results confirm the hypothesis of dependence and independence, the RFC specifications of each protocol are ratified. Therefore, a knowledge representation model based on the different Bayesian networks can be built, decreasing in this way the complexity of the design of the BN and minimising its training process.

V. CONCLUSION AND FUTURE LINES

As the use of Internet grows over all boundaries, the number of menaces rises to become subject of concern and increasing research. Within this scenario, Intrusion Detection Systems have proven themselves as a real candidate to separate real data from dangerous one by offering crucial information to provide safer networks. Still, current solutions concentrate only on well-known attacks (misuse) either on unknown ones (anomaly).

It has been already demonstrated [1] that the use of Bayesian networks to integrate anomaly and misuse detection in a Intrusion Detection System is a suitable architectural solution. Nevertheless, this Bayesian network-based approach faces big computational costs during the construction of the knowledge model. Within this paper, we proposed the use of expert knowledge to minimise these costs.

We have accurately studied how to create a model of knowledge representation. First of all, we obtained a rep-

resentative data sample. Second, we defined how many temporal steps we were going to use for our experiment. Third, we established the hypothesis according to the expert knowledge. Fourth, we planned the process of structural learning and performed it. After this step, we obtained statistical metrics from the partial Bayesian networks. These partial fragments were unified and adapted before verifying the hypotheses of dependence and independence. Finally, we obtained the optimal structural definition of the knowledge representation model on which we performed parametric learning. According to this experiment, we have proved the validity of the hypotheses and obtained the optimal BN for Intrusion Detection Systems (IDS). This knowledge model is currently being used as the expert system of our own IDS architecture.

Future work will focus on further research on the use of expert knowledge for Bayesian networks modelling over different domains beyond the Intrusion Detection and the creation of a formal metric. This metric will measure the impact of the use of expert knowledge in the model creation time and the final performance of a Bayesian network.

REFERENCES

[1] P. G. Bringas and Y. Penya, "Bayesian-networks-based misuse and anomalies detection system," in *Proceedings of the 10th International Conference on Enterprise Information Systems (ICEIS)*, 6 2008, pp. 12–16.

[2] A. R. Sinha and H. Zhao, "Incorporating domain knowledge into data mining classifiers: An application in indirect lending," *DECISION SUPPORT SYSTEMS*, vol. 46, no. 1, pp. 287–299, DEC 2008.

[3] P. Helman, R. Veroff, S. Atlas, and C. Willman, "A Bayesian network classification methodology for gene expression data," *JOURNAL OF COMPUTATIONAL BIOLOGY*, vol. 11, no. 4, pp. 581–615, 2004.

[4] F. Coolen, M. Goldstein, and M. Munro, "Generalized partition testing via Bayes linear methods," *INFORMATION AND SOFTWARE TECHNOLOGY*, vol. 43, no. 13, pp. 783–793, NOV 15 2001.

[5] E. Castillo, J. M. Gutierrez, and A. S. Hadi, *Expert Systems and Probabilistic Network Models*. Springer-Verlag, 1997.

[6] P. Alipio, P. Carvalho, and J. Neves, *Using CLIPS to Detect Network Intrusion*. Springer-Verlag, 2003, vol. 2902/2003.

[7] S. Meganck, P. Leray, and B. Manderick, "Learning causal bayesian networks from observations and experiments: A decision theoretic approach," *Lecture Notes in Computer Science*, vol. 3885/2006, pp. 58–69, 2 2006, iSBN 978-3-540-32780-6.

[8] P. Mell, V. Hu, R. Lippmann, J. Haines, and M. Zissman, "An overview of issues in testing intrusion detection systems," National Institute of Standards and Technology, Massachusetts Institute of Techonology Lincoln Laboratory, USA, Tech. Rep., 6 2003.

[9] N. Puketza, K. Zhang, M. Chung, B. Mukherjee, and R. Olsson, "A methodology for testing intrusion detection systems," *IEEE Transactions on Software Engineering*, vol. 22, no. 10, pp. 719–729, 10 1996.

[10] K. Murphy, "An introduction to graphical models," Intel Corporation, Tech. Rep., 5 2001.

[11] A. H. E. Castillo, J.M. Gutirrez, *Sistemas Expertos y Modelos de Redes Probabilsticas*. Monografas de la Academia de Ingeniera, 1998, iSBN 8460093956.

[12] F. J. Dez, *Introduccin al Razonamiento Aproximado*, D. de Inteligencia Artificial, Ed. Universidad de Educacin a Distancia, 1998.

[13] C. Intel, "Probabilistic network library home page," available: https://sourceforge.net/projects/openpnl/. [Online]. Available: https://sourceforge.net/projects/openpnl/

[14] Netfilter, "About the netfilter/iptables project," available: http://www.netfilter.org/. [Online]. Available: http://www.netfilter.org/

[15] J. Gutirrez, "Introduccin al data mining bayesiano," Seminar for DTF, 1 2005, director del Departamento de Matemtica Aplicada de la Universidad de Cantabria.

[16] I. Urzay, "Personal interview," 3 2005, chief Technology Officer de Panda Software.

[17] L. Fu, "A Comparison of State-of-the-Art Algorithms for Learning Bayesian Network Structure from Continuous Data," Ph.D. dissertation, Vanderbilt University, 2005.

[18] M. Kalisch and P. B
"uhlmann, "Estimating high-dimensional directed acyclic graphs with the PC-algorithm," *The Journal of Machine Learning Research*, vol. 8, p. 636, 2007.

[19] J. Pearl, "Reverend bayes on inference engines: A distributed hierarchical approach," in *Proceedings of the AAAI National Conference on AI*, 1982, pp. 133–136.

[20] ——, "Graphical models for probabilistic and causal reasoning," in *Handbook of Defeasible Reasoning and Uncertainty Management Systems, Volume 1: Quantified Representation of Uncertainty and Imprecision*, D. M. Gabbay and P. Smets, Eds. Dordrecht: Kluwer Academic Publishers, 1998, pp. 367–389.

[21] J. Kim and J. Pearl, "A computational model for combined causal and diagnostic reasoning in inference systems," in *Proceedings of the IJCAI-83*. Karlsruhe, Alemania, 1983, pp. 190–193.

[22] H. Olesen, S. Lauritzen, and F. Jensen, *Hugin: A System Creating Adaptive Causal Probabilistic Networks*. Dubois et al. Eds, 1992.

[23] A. Madsen, M. Lang, U. Kjrulff, and F. Jensen, "The hugin tool for learning bayesian networks," in *Proceedings of the 6th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*. Aalborg, Dinamarca: Springer, 7 2003, pp. 588–593, iSBN 3540404945.

[24] J. Cheng, R. Greiner, J. Kelly, D. Bell, and W. Liu, "Learning bayesian networks from data: an information-theory based approach," *Artif. Intell.*, vol. 137, no. 1-2, pp. 43–90, 2002.