Alarm Reduction and Correlation in Intrusion Detection Systems

Tobias Chyssler[1], Stefan Burschka[2], Michael Semling[2], Tomas Lingvall[2], Kalle Burbeck[1]

[1]Dept. of Computer and Information Science
Linköping University,

S-581 83 Linköping ,Sweden
[tobch,kalbu]@ida.liu.se

[2]Swisscom Innovations
Service & Security Management,

Ostermundigenstrasse 93

3050 Bern,Switzerland
[Stefan.Burschka,Michael.Semling,
Tomas.Lingvall]@Swisscom.com

Abstract: Large Critical Complex Infrastructures are increasingly dependent on IP networks. Reliability by redundancy and tolerance are an imperative for such dependable networks. In order to achieve the desired reliability, the detection of faults, misuse, and attacks is essential. This can be achieved by applying methods of intrusion detection. However, in large systems, these methods produce an uncontrollable vast amount of data which overwhelms human operators. This paper studies the role of alarm reduction and correlation in existing networks for building more intelligent safeguards that support and complement the decisions by the operator. We present an architecture that incorporates Intrusion Detection Systems as sensors, and provides quantitatively and qualitatively improved alarms to the human operator. Alarm reduction via static and adaptive filtering, aggregation, and correlation is demonstrated using realistic data from sensors such as Snort, Samhain, and Syslog.

1 Introduction

The economy and security of Europe is increasingly dependent on a range of Large Complex Critical Infrastructures (LCCI) such as electricity and telecommunication networks. Protecting these infrastructures requires an understanding of the vulnerabilities that exist in every layer of the network; from the physical layer up to the network and service layers as well as the organisational layer that supports the complex operation of these networks. This paper focuses on the support of operators analyzing messages created by various sources, in particular Network Intrusion Detection Systems (NIDS), Host Intrusion Detection Systems (HIDS) and general log data. The amount of data being produced by such Intrusion Detection Systems (IDS) exceeds by far human

capability of information processing. One study [YBU03] estimates that the number of intrusion attempts over the entire Internet is in the order of 25 billion each day and increasing. McHugh [Hug01] claims that attacks are getting more and more sophisticated while they get more automated. This requires detailed analysis in-depth leaving no time for adequate reactions. Most INFOrmation SECurity systems (INFOSEC) based on IDS technology attempt to respond to the operator's information overload.

INFOSEC systems either apply knowledge-based techniques (typically realised as signature-based misuse detection), or behaviour-based techniques (e.g. by applying machine learning for detection of anomalies). On the other hand, there are tools performing methods of data mining (e.g. on log results), or by simply collecting and grouping alerts for further examination by a human operator [MCZ+00]. However, these INFOSEC systems generate far too many alarms. While post mortem studies are possible on large data sets, the ability of reacting in real-time to intrusions is highly dependent on improved quality of the alarms, and in particular reduction of the false alarm rates.

When studying the range of problems in dealing with security issues in the management network of telecom service providers, the following needs are identified: (1) Reduction of the message counts such that the operator can cope with them. (2) A lower rate of false alarms. (3) Information collection and correlation from various sources to identify indices of attacks. (E.g. The combining of information about the network topology with IDS alarms). (4) Indication of general network "health" with predictive elements so that total service collapse is avoided. The work in this paper addresses the first two issues and to some extent the third issue which is a prerequisite for future sophisticated analysis of alarm data. The work is carried out in the context of the European Safeguard project [Saf03] aiming to demonstrate the use of distributed and coordinated software agents for enhancing existing defence mechanisms in telecom and electricity management networks. Safeguard agents at higher levels use data that is processed as presented in this paper. Thus, this paper describes methods how to improve the quality of IDS data, thus enabling a human operator to fulfil an in-depth analysis within acceptable time.

The structure of the paper is as follows. First, we present a safeguard architecture that has emerged during the work in the above project. Next, the improvement of the information quality of knowledge-based and behaviour-based approaches is discussed. The knowledge-based approach filters and aggregates alarms. Moreover, methods of automated text classifications utilizing naïve Bayesian networks are used to adapt the IDS filtering. Behaviour-based methods correlate the aggregated alarms. The three considered techniques are: An additive correlator, a Neuronal Network (NN) classifier, and a classifier based on K-nearest-neighbours.

We evaluated our methods on data generated in a test network of appropriate size; because the datasets used for a number of recent evaluations of alarm correlation approaches [HA03] were not available to the wide research community. Our test network consists of 50 machines, and has been set up at Swisscom as part of the Safeguard research effort, especially for evaluating various approaches to recognition,

reaction and recovery mechanisms. The data produced for these experiments can be shared and tested by other researchers. Finally we comment on the conclusions so far.

1.1. Related works

Several proposals for alarm correlation (e.g. [CM02], [DW01], [MD03], [NCR02] and [PFV02]) are limited to predefined rules for attack scenarios and countermeasures. Another proposal to find new scenarios is given by Qin et al. [QL03]. A different approach is the probabilistic alert correlation taken by Valdes et al. [VS01], where a mathematical framework is used to collate alerts based on their similarity. Our mechanism is based on text based distance metrics combined with Neural Networks and K-nearest neighbours' algorithms.

The idea to perform data mining in order to reduce for false alarm has been explored by Julisch et al. [JD02]; using conceptual clustering of old alarms to derive new filters. Our method for adaptive filtering has the same objective, but we use automatic text classification instead.

2 The Safeguard context

The Safeguard architecture is outlined in Figure 1.

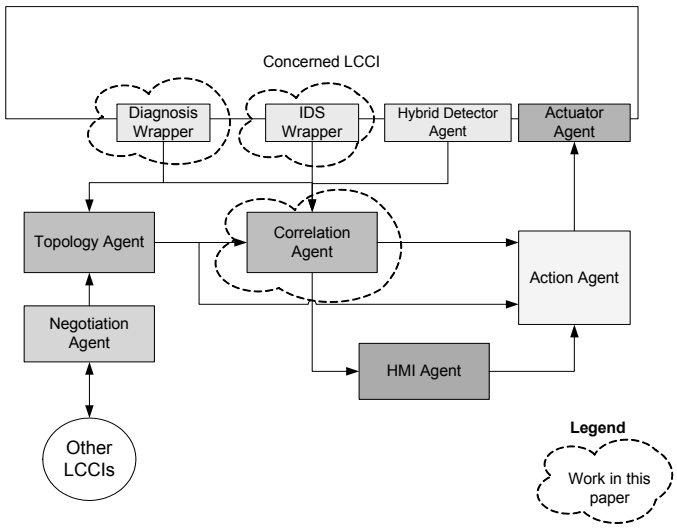


Figure 1: The Safeguard agent architecture

The roles of agents can be described as follows:

- **Wrapper agent**: Wraps standard INFOSEC devices and existing LCCI diagnosis mechanisms, and applies static filtering and normalisation on their outputs.

- **Topology agent:** Gather network topology information dynamically, e.g. host types, operating system types, services provided, known vulnerabilities.

- **Hybrid detector agent:** Uses data mining techniques such as clustering to detect anomalies.

- **Correlation agent:** Filters, aggregates, and correlates (section 3.4) normalized messages from different sources. Special filter lists are applied for network critical services [Bu03]. In practice the agent is split into a filter, an aggregation and a separate correlation and alarm generation part.

- **Action agent:** Initiates automatic and human controlled responses.

- **Negotiation agent:** Communicates with negotiation agents in neighboured LCCIs to pass on information.

- **HMI (Human Machine Interface) agent:** Provide an interface supporting the operator in his analysis. This agent also facilitates the incorporation of adjustable autonomy within the agent-based system [Sc01].

- **Actuator:** Interface for communication with lower layer software and hardware (e.g. changing firewall rules, renicing or killing processes, traffic shaping).

There may be several instances of each agent in each LCCI and only some variants are described in this paper.

2.1 Choice of sensors

As reported by Yin et al. [Yi03] the success of IDSs depends upon the data they process. In order to assess the relevance of an alarm, several sources of information have to be considered. For this reason we use three different INFOSEC devices. One the network level, the NIDS Snort, on host level the HIDS Samhain and various agents providing information about system relevant information as well as application logs via Syslog are applied. Multiple sources of additional information such as network topology information, connection statistics, policies, and vulnerabilities are further included in our selection of sensors. Anomaly detectors for network traffic as described in [BN04].

3 Alarm Reduction

In the following we describe one possible human way of analysing alarms. The expressiveness of alarms is evaluated by its:

- Severity
- Number
- Frequency
- Variety
- Uniqueness
- Payload

As outlined in Figure 2: Analysis procedure , the analysis starts by removing the alarms that are known to be of no interest. Syslog is checked for messages that look suspect. For each message found, the corresponding machine is checked in the HIDS to detect suspicious changes and the NIDS is checked for possible sources for attacks around the time that roughly coincides with the event.

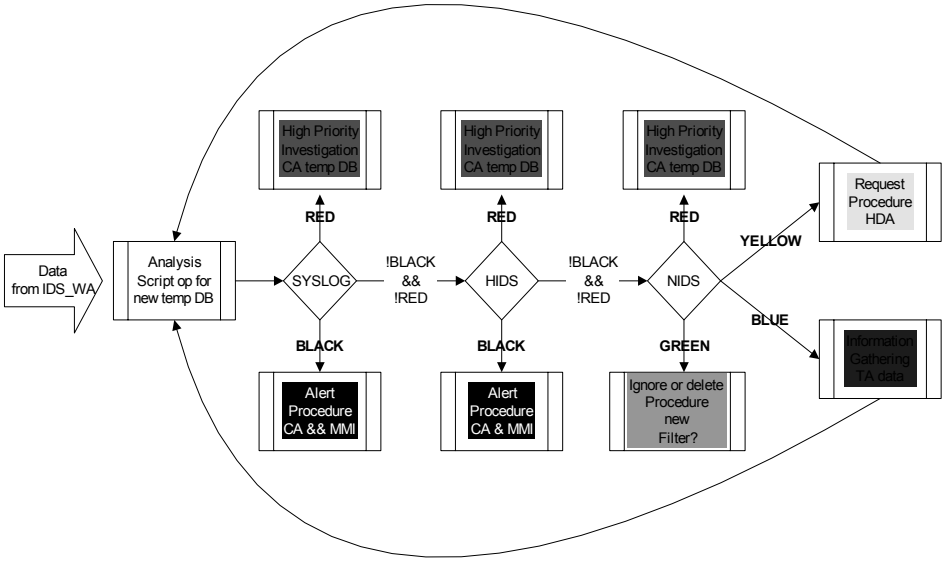


Figure 2: Analysis procedure

To enforce the relevance of alarms, knowledge about

- The attack signatures in the payload
- Perimeter defense settings (firewalls, reverse proxies, etc)
- Host properties and vulnerabilities (Operation System, patch level, etc)

- Abnormal messages around the time for the alarms. (e.g. reboots, service failures, etc)

The last step is to investigate the most frequent alarms. These may be an indication of misconfigurations or software bugs, denial of service attacks or virus respectively worms.

Overall, the alarm analysis activities can be broadly grouped into filtering, aggregation and correlation. Using the sum of the severities during a time period captures some of the characteristics for listed alarms (high severity, lots of alarms and high rate). This is one of the approaches taken in this paper.

Figure 3 presents the implementation in two of our agents based on the description above. Normalisation is the process to transform the data into a uniform format. Messages are aggregated using a text based distance metric. In the following the detail of static and adaptive filtering, aggregation and correlation is discussed.

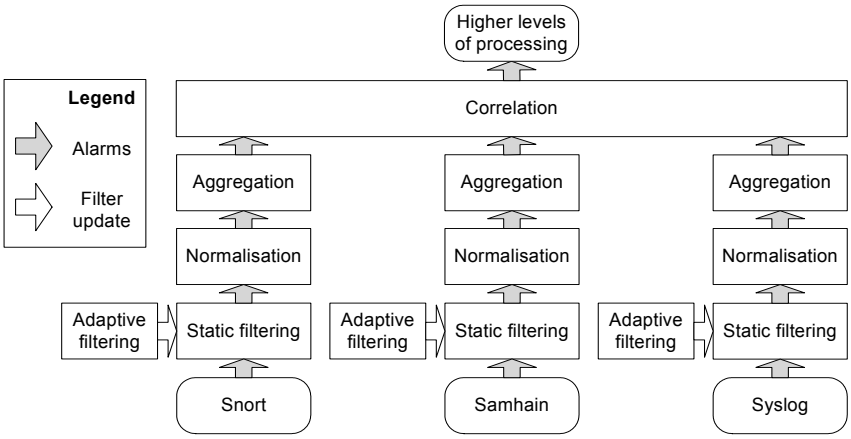


Figure 3: Overview of Methods

3.1 Static filtering

Under normal operations, IDSs produce a lot of messages, therefore it excessively time consuming to distinguish false alarms from real attacks. In particular there were a lot of uninteresting messages coming from untuned Syslog and Samhain. Practically in LCCIs, tuning requires a static network, were changes almost never happen or it requires a non-affordable effort for the administration. The resulting amounts of uninteresting messages have a low severity, though through their number of occurrence, they cannot be neglected. Therefore static filters exclude data from Samhain and Syslog that does not carry any valuable information, such as messages that Samhain checks a file or that Syslog is doing a garbage collection.

Filters are implemented either as an *ignore* filter or as a *delete* filter. The ignore filters keep the alarms in the database but they are not forwarded to the next agent. But they are saved for forensic investigation. The delete filters removes alarms permanently from the database. All static filters have a limited duty cycle defined by the operator and have to be reactivated afterwards.

3.2 Adaptive filtering

Static filter deals with known problems. However, since it is not possible to foresee all future misconfigurations, adaptive filtering algorithms were implemented to suggest new filter rules to the administrator. This is based on Naïve Bayesian (NB) learning [Hec96]. The idea is to train a NB text classifier to classify messages by looking at word statistics of messages. The NB-classifier has to be trained first with a subset of messages labelled as interesting or not. During training, a knowledge base is automatically acquired, enabling the NB-classifier to assess whether unknown messages are interesting.

The adaptive filters are used in the following workflow:

1. For performance reasons, the algorithm for adaptive filtering is launched on a periodic basis.

2. The algorithm suggests filter rules for top scoring events to a human expert via HMI.

3. The reply is also used to optimise the training corpus, achieving on-line retraining.

4. In order to avoid over learning effects, features, hardly ever occurring in time, are reduced in their significance and are finally forgotten.

3.3 Aggregation

Repeated, identical alarms do not provide any additional information. It would reduce the information overload if each all these alarms were represented in only one alarm including the number of its frequency. The relevant fields for aggregation have to be defined for each source individually. For Snort the source and destination IP, as well the server ports and the messages are relevant. Ephemeral ports can be ignored. For Syslog, the PID number is unimportant but the program and the message field is relevant. All data is aggregated within a given time window.

3.4 Correlation

Motivation to build an automatic correlation agent is the fact that human operators successfully correlate information from different INFOSEC mechanisms in search for known attacks. Moreover, humans are capable to find indications for attacks even though they are not known. However, the way a security expert analyses the information is

complicated therefore a complete model is impossible to find. The correlator uses data from the NIDS, HIDS and SYSLOG to find indications of unknown attacks.

All alarms regarding one host during a time window are considered, i.e. the selected correlation parameters are time, IP address, ports resp. application, etc. Given a time window, a time slot of a three dimensional vector is produced by taking the sum of severities from the three sensors. Based on the input pattern, the correlator decides whether this is an interesting alarm or not. For further processing just one single alarm is generated. The idea is illustrated in Figure 4.

This three-dimensional vector is then used as input for the following three correlation algorithms:

1. A simple additive correlator with the parameter 'added severity threshold'
2. A neural network [Sim98] with a variable set of neurons and hidden layers. Training is achieved by back propagation.
3. K-Nearest Neighbour [DGL96] with K as a variable parameter.

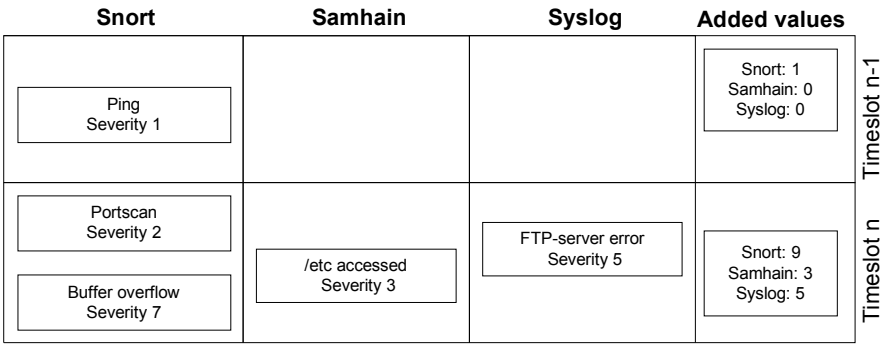


Figure 4: Alarms Appearing in Two Time Slots

The additive correlator simply builds a weighted sum of the number of occurrences of the severities within a given timeslot. The sum is then compared to a threshold generating an alarm if the chosen limit is exceeded. By varying the threshold different detection rates and false positive rates can be achieved. The other two correlation algorithms must be first trained on a subset of data before it is applied to the new real time dataflow.

4 Evaluation

4.1 Data generation

The topology of the test network used to perform the evaluation is given in Figure 5.

The test network is an image of a realistic IP environment and consists of:

- A server zone and a workstation zone. Both zones consists of Sparc 5, 10 and 20, Ultra 2,5,10 machines running Solaris 5.6 to 5.10 with various patch levels as well as several versions of Linux and PCs running on Windows 98, NT, 2000, XP installed on VMWARE.
- A jump-station running on OpenBSD can be used to login via SSH from the Internet.
- Switches, routers and a hub connecting the different nodes.
- An external zone simulating the Internet in case of trials with worms, virus, and (D)DoS.

This way, it is ensured, that no "bad" traffic contaminates the Internet.

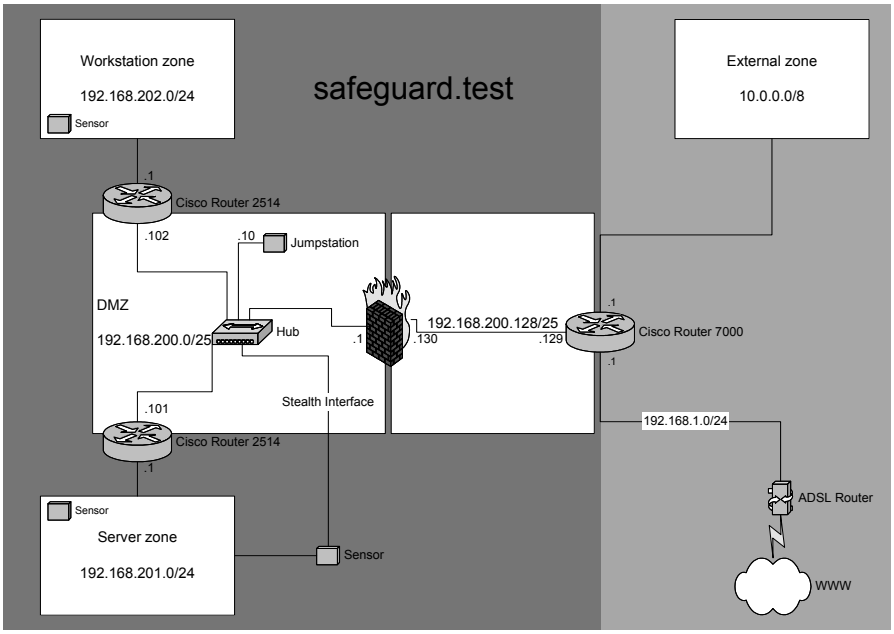


Figure 5: The Safeguard Telecom test network

From the test network data can be collected and labelled. This data may contain "normal" traffic and some attacks. In our experiments the attacks were performed using various tools and techniques, including:

- Scanning of the network using ping script once, Nessus (www.nessus.org) four times and Nmap (www.insecure.org/nmap) five times. Both Nessus and Nmap are security scanners.
- Brute-force password guessing for telnet with Brutus (www.hoobie.net/brutus), attack used twice.
- Sadmin buffer overflow attack, launched against two different hosts at separate times.
- Installing a rootkit for Solaris 2.6 once (www.honeynet.org/papers/motives).
- Various "bad" behaviour when logged in on a computer, such as allocating all space on the disk, killing as many processes as possible (once), imitating memory leaks.

The attacks were launched from the external zone simulating Internet attacks and from the server or workstation zone simulating insider attacks. During the attacks, normal usage of the computers by people working on them generated normal background noise.

Applying the resulting datasets led to the different experiments described in this paper.

4.2 Results

4.2.1 Static filtering

The data used to test the result of static filtering is gathered from all relevant hosts on the network during three weeks and includes alarms generated from attacks and from normal traffic. Figure 6 illustrates the result of static filtering on the Syslog, Snort alarms, and Samhain alarms. No alarms related to attacks were removed. Clearly Samhain alarms were reduced most drastically by this knowledge-based approach (order of magnitude). The method had the lowest impact on Snort alarms. This can be explained by the huge diversity of different network traffic patterns in contrast to the more precise knowledge of uninteresting Syslog and Samhain alarm types.

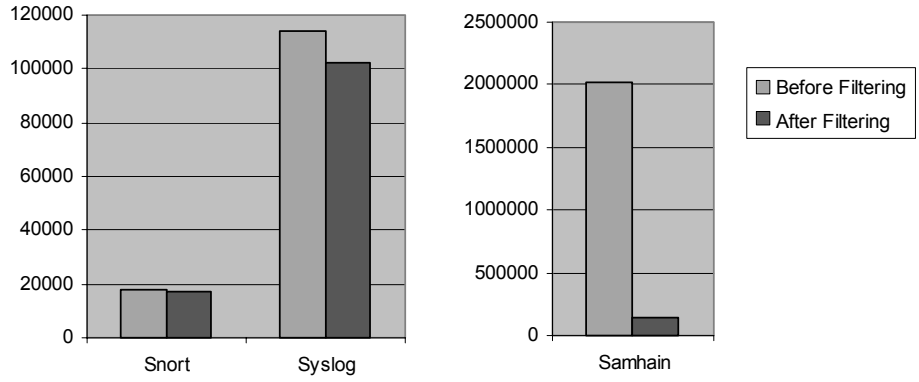


Figure 7: Static filtering for Snort Syslog and Samhain

4.2.2 Adaptive filtering

Text classification methods for adaptive filtering were applied to Syslog alarms. The data set used, when evaluating the adaptive filtering, consisted of messages collected over the period of two months. The data was labelled as "interesting" or "uninteresting" by a security expert. An interesting message contains valuable indications of an attack. For example, a message saying "telnetd[1]: Successful login as user root" or "File changed" is classified as interesting, but messages such as "Garbage collecting while idle" will be classified as uninteresting. When classifying the messages, the fields: Facility, Priority, Program and Message of each Syslog message were included. The field Level was excluded since it has the same value as the field priority.

The whole data set contained 156 212 alarms, where 18 941 of them are classified as interesting. These alarms where then divided into different sets for training and testing. The test data set contained 53 722 alarms, where 10 621 of them are classified as interesting.

Table 2 shows the results of the adaptive learning algorithm in terms of the precision of the results produced on the test data set. By precision here we mean the number of correct classification divided by the total number of alarms.

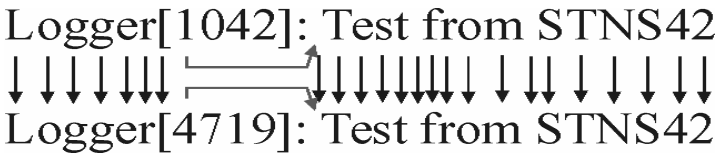
Data set	**Correct classifications**	**Incorrect classifications**	**Precision**
Test part	53682	40	0,99926

Table 1: Result of text classification for adaptive filtering

As the last column indicates, the method is likely to classify the alarms with a very high precision as long as the selected attributes for classifying the alarm messages as interesting/uninteresting and the training data remain valid in the eyes of the security expert.

4.2.3 Aggregation

Next we show the results of the aggregation algorithm, which is based on an edit distance measure with blank synchronisation. As long as the characters are the same, the next character is considered, if not, the next space is used for synchronisation. The number of equal characters in relation to the length is the percentage of similarity.



Empirical tests showed that with more than 65% similarity the same messages are recognised as being identical. A similarity of 70% means a reduction for Snort of 96.5% (from 3755 to 130) and for Syslog of 99.8% (from 13691 to 28).

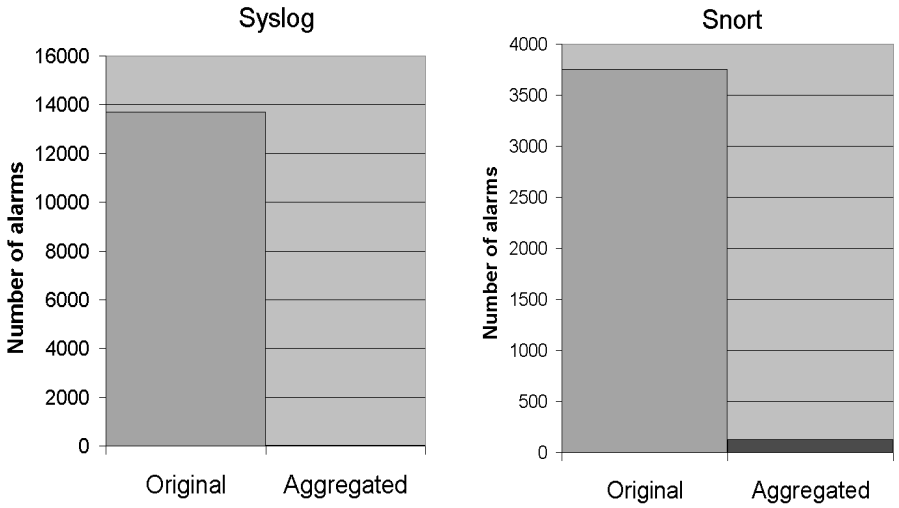


Figure 7: Original and Aggregated Messages

The influence of the time window size for the aggregation performance is shown in Figure 8. The data set being use here is smaller, but has the same characteristics as in above.

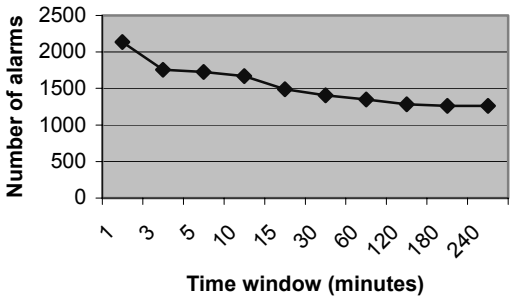


Figure 8: Number of Snort alarms after aggregation, varied size of time window

Obviously, the larger the time window the fewer alarms are left after aggregation. In order to limit the delay until alarms are sent to the higher layer processes, the first instance of a new alarm is instantly sent while at end of the time window the final aggregation result is delivered.

However, this is only the first stage in dealing with the information quality, and higher-level correlation agents are dependent on recognition of anomalous situations within an adequate short time period for vital reactions [Bu03].

4.2.4 Correlation

As mentioned in chapter 3.4 we compared the messages from the NIDS, HIDS and Syslog to produce estimation for unknown attacks. Three techniques for behaviour-based correlation were applied. Figure 9 shows the additive correlator, a neural network and a 1-nearest neighbour classifier algorithm. For the training we used a subset of eight hosts on the network during eight days.

Using a time window of 15 minutes, 6 048 three-dimensional time slots were created. 54 of the time slots were attacks. The dataset was divided into a training part, 3993 timeslots, 36 of these timeslots were attacks and the test part was the remaining part.

The additive correlator obviously gives less false positives (see ROC curve of Figure 9, left part) with a lower threshold but the detection rate decreases. For higher values of the threshold, the 1-NN algorithm and the neural network have lower false-positives rates.

The neural network correlator had an overall better performance with detection rate of 94,4% and false positives rate of 1.4%. Considering the vast amount of data being presented to a human per day this false positive rate is still rather high. Ongoing work using more correlation sources e.g. topology information, policy definition and health observations are expected to show significant improvement.

Figure 9 (right part) shows the results of the study of the K-NN algorithm for K=1...10 using the same data set, where K=1 performs best. Explanation: In real data, there are far

more time slots, that correspond to non-attack situations than to attack situations. Thus, the K-NN algorithm has more difficulties to distinguish our data sets.

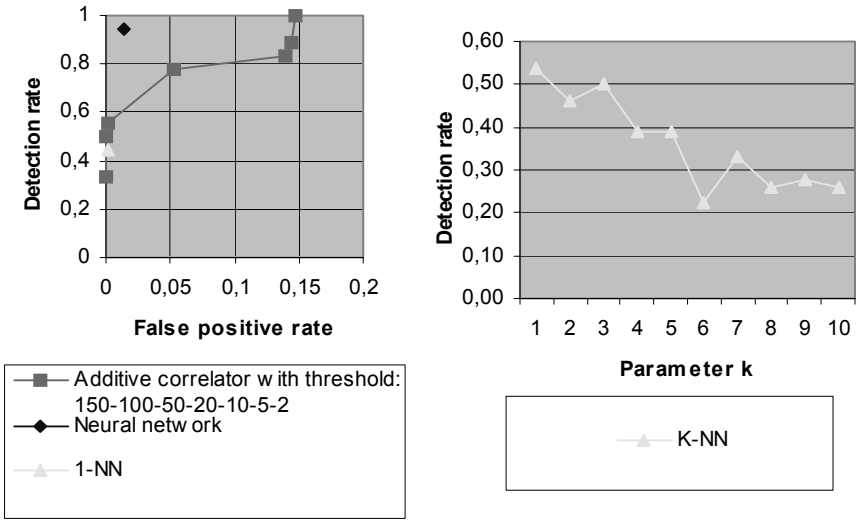


Figure 9: (Left) ROC curve for correlation methods.
(Right) Result of K-NN correlator for different values of parameter K

Table 2 summarizes the three studied methods utilizing the same data. TP, FP, FN, and TN stand for true positives, false positives, false negatives, and true negatives respectively. The overall accuracy is the sum of TP and TN divided by the sum of all four values.

Type	**TP**	**FP**	**FN**	**TN**	**Detection rate**	**FP rate**	**Overall accuracy**
Additive (Thrs.=20)	14	106	4	1931	0,778	0,052	0,946
Neural network	17	29	1	2008	0,944	0,014	0,985
1-NN	8	2	10	2035	0,444	0,001	0,994

Table 2: Correlation result summary

5 Conclusions

This paper shows that static and adaptive filtering as well as aggregation is required in order to provide a useable pre-processing of realistic IDS datasets. Practical experience revealed, that naïve Bayes classifiers reduced the amount of time to create filter rules and improved their quality.

In order to improve standard methods of correlation, e.g. rule based, three different classifiers were tested only operating on information of NIDS, HIDS, and SYSLOG. Used as an add-on, these classifiers also provide the benefit to detect unknown attacks. The neuronal network performed best. K-NN was found to be inappropriate for real world alarm message classification, where the system has to be adapted to its environment.

Nevertheless, human knowledge about the network, the vulnerabilities and the configuration is still inevitable in future. But an on-line retainable NB-classifier helps the expert to adapt the system to it dynamic environment, thus saves valuable time for appropriate countermeasures. Moreover the reduced amount and the quality of the alarms enable the expert to concentrate on the vital information.

The work in the Safeguard [Saf03] project is currently in progress, dealing with other aspects of recognition, e.g. anomaly detection [BN04], in combination with correlation of dynamic topology data, automatic actions, and information presentation to humans.

Acknowledgements

This present work was supported by the European project Safeguard IST-2001-32685. We would like to thank Thomas Dagonnier at Swisscom for valuable support. The Safeguard agent architecture has been developed with the input from all the research nodes of the project, the cooperation of whom is gratefully acknowledged.

References

[MCZ+00] S. Managanaris, M. Christensen, D. Zerkle, and K. Hermiz, A Data Mining Analysis of RTID Alarms, Computer Networks, 34(4), Elsevier pub., Oct 2000.

[BN04] K. Burbeck, S. Nadjm-Tehrani, "ADWICE - Anomaly Detection with Fast Incremental Clustering.", Technical Report, Dept. of Computer and Information Science, Linköping University. February 2004.

[Bu03] K. Burbeck, S. G. Andres, S. Nadjm-Tehrani, M. Semling, and T. Dagonnier. “Time as a Metric for Defence in Survivable Networks”. Proceedings of the Work in Progress session of 24th IEEE Real-Time Systems Symposium (RTSS 2003), Dec. 2003.

[Chy03] T. Chyssler. “Reducing False Alarm Rates in Intrusion Detection Systems”, Master thesis No. LITH-IDA-EX-03/067-SE, Linköping University (2003).

[CM02] F. Cuppens and A. Miége. “Alert Correlation in a Cooperative Intrusion Detection Framework”. Proceedings of the 2002 IEEE Symposium on Security and Privacy. 2002. Pages 187 – 200.

[DGL96] L.Devroy, L. Györfi and G. Lugosi. “A Probabilistic Theory of pattern Recognition”. Springer Verlag, New York Inc, 1996.

[DW01] H. Debar and A. Wespi. “Aggregation and Correlation of Intrusion-Detection Alerts”. Proceedings of the fourth International Symposium on Recent Advances in Intrusion Detection (RAID). Springer Verlag, October 2001. Pages 85-103.

[Hec96] D. Heckerman. “A Tutorial on Learning With Bayesian Networks”. Technical Report MSR-TR-95-06, Microsoft Research. March 1995 (Revised November 1996).

[HA03] J. Haines, D. K. Ryder, L. Tinnel and S. Taylor. “Validation of Sensor Alert Correlators”., Security & Privacy Magazine, IEEE, Vol. 1, No. 1. January/February 2003. Pages. 46 - 56.

[Hug01] J. McHugh. “Intrusion and Intrusion Detection”. International Journal of Information Security. Vol 1, No 1. Springer Verlag, August 2001. Pages 14 – 35.

[JD02] K. Julisch and M. Dacier. “Mining Intrusion Detection Alarms for Actionable Knowledge”. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM Press, July 2002. Pages 366-375.

[MD03] B. Morin and H. Debar. “Correlation of Intrusion Symptoms: an Application of Chronicles”. Proceedings of the sixth International Symposium on Recent Advances in Intrusion Detection (RAID). Springer Verlag, November 2003. Pages 97 – 112.

[NCR02] P. Ning. Y. Cui and D. S. Reeves. “Constructing Attack Scenarios through Correlation of Íntrusion Alerts”. Proceedings of the 9th ACM conference on Computer and communications security. ACM Press, 2002. Pages 245 – 254.

[PFV02] P. A. Porras, M. W. Fong and A. Valdes. ”A Mission-Impact-Based Approach to INFOSEC Alarm Correlation”. Proceedings of the fifth International Symposium on Recent Advances in Intrusion Detection (RAID). Springer Verlag, October 2002. Pages 95–114.

[QL03] X. Qin and W. Lee. “Statistical Causality Analysis of INFOSEC Alert Data”. Proceedings of the sixth International Symposium on Recent Advances in Intrusion Detection (RAID). Springer Verlag, November 2003. Pages 73 – 93.

[Saf03] Safeguard website: http://www.ist-safeguard.org.

[Sc01] P. Scerri. ”Designing Agents for Systems with Adjustable Autonomy”. PhD thesis No. 724, Linköping University (2001).

[Sim98] J. Sima. “Introductions to Neural Networks”. Technical report No V-755, ICS CAS, Prague, 1998.

[VS01] A. Valdes and K. Skinner. ”Probabilistic Alert Correlation”. Proceedings of the fourth International Symposium on Recent Advances in Intrusion Detection (RAID). Springer Verlag, October 2001. Pages 54-68.

[YBU03] V. Yegneswaran, P. Barford and J. Ullrich. “Internet Intrusions: Global Characteristics and Prevalence”. Proceedings of the 2003 ACM SIGMETRICS international conference on Measurement and Modelling of Computer Systems, June 2003. Pages 138-147.

[Yi03] X. Yin, K. Lakkaraju, Y. Li and W. Yurcik. ”Selecting Log Data Sources to Correlate Attack Traces for Computer Network Security: Preliminary Results”. Proceedings of the 11th International Conference on Telecommunication Systems, Modelling and Analysis, October 2003.