Click here to view PowerPoint presentation; Press Esc to exit

Experiences with Network Intrusion Detection

R. Coolen, H.A.M. Luiijf
TNO Physics and Electronics Laboratory
P.O. Box 96864
2509 JG The Hague, The Netherlands

E-mail: *coolen@fel.tno.nl, luiijf@fel.tno.nl*

W.J.F. v. Geloven, E.A. Bakker
Defence Telematics Organisation
P.O. Box 104
3140 AC Maassluis, The Netherlands

E-mail: *wjf.v.geloven@mindef.nl, ea.bakker@mindef.nl*

Abstract

This paper describes our experience with several commercial Network Intrusion Detection Systems (NIDSs) deployed in a network connected to the Internet. Specific problems in the operation of NIDS are highlighted, and a number of solutions to identified problems will be presented. Finally, we shall present our view on the contribution of NIDS to the security posture of the network environment to be secured. Throughout the paper, the focus will be on the real-time aspects of incident detection in networks, and, to a lesser extent, to incident response.

Introduction

Numerous papers describe the proliferating threats to military communication and information systems (CIS), including those of NATO, non-governmental organisations, and coalitions. Other papers describe the increasing importance of networks and information systems in military operations, e.g. voice and video communication between and among soldiers, equipment and vehicles in the area of operations, and with the command and control centres.

The Internet Protocol (IP) protocol suite is likely to be the future standard for military communication networks. The vulnerabilities of this protocol suite have been the subject of many research projects. The open nature of the Internet protocols with its inherent insecurity, its lack of authentication, the vulnerability for denial of service, and the like, is well known [1].

This paper describes practical experiences with network intrusion detection systems (NIDS). NIDS are means to detect, and alarm on, incidents occurring in a CIS. Although the experiences described in this paper result from the deployment of NIDSs in a static infrastructure, the results and lessons learned are equally relevant and applicable for (future) tactical IP-based networks. Throughout this paper the focus will be on the real-time aspects of incident detection in networks and to a lesser extent to incident response.

Description of the test network

The Netherlands Defence organisation has a nation-wide network, with approximately 3000 kilometres of glass-fibre, connecting Army, Navy, and Air Force, the Dutch Ministry of Defence, and other parts of the defence organisation. This unclassified, well-protected defence network is called the Netherlands Armed Forces Integrated Network (NAFIN-network, or NAFNET). The NAFIN-network is managed by the Defence Telematics Organisation (DTO).

To compare various Network Intrusion Detection Systems (NIDS), the authors received permission to connect a test network to the front-end router of NAFNET. The routing was set up in such a way that all traffic coming into the access router which is situated in front of the boundary protection device (BPD) was duplicated (mirrored) to the test network (see figure 1). All NIDSs under test were connected to this test network. As this network was located in front of the boundary-protection services, all technically erroneous packets and packets of attack attempts from and to the Internet appeared on the 'test network'. In the perfect case, all NIDSs should see the same error and attack packets and should signal the same incidents.

Paper presented at the RTO IST Symposium on "Real Time Intrusion Detection", held in Estoril, Portugal from 27-28 May 2002, and published in RTO-MP-101.

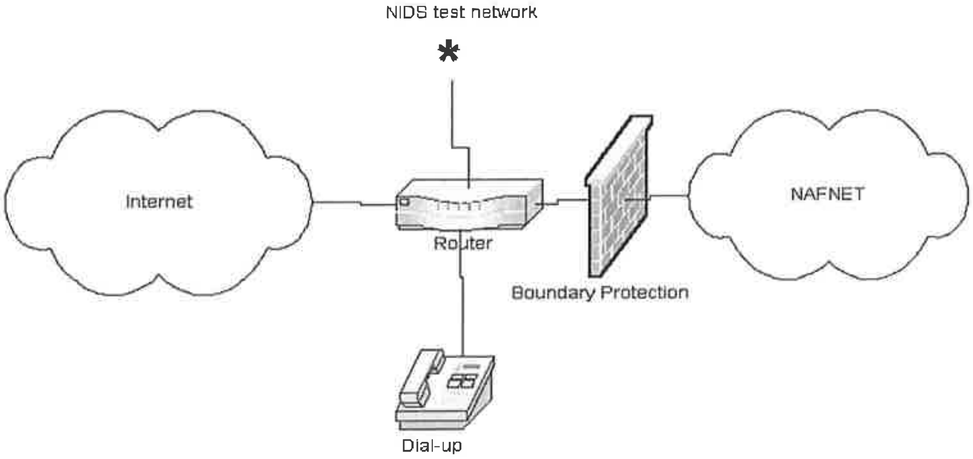

Figure 1: The NAFNET-Internet interconnection with the NIDS-test network

The experiences described below are from operating five Network Intrusion Detection Systems parallel and simultaneously. The experiments were executed in the spring of 2001. Below, we shall discuss our experiences in a non-product specific fashion.
Four of the NIDS products were commercial-off-the-shelf: Realsecure from ISS, BlackICE from Networkice, Netprowler from Symantec, Netranger from Cisco. The fifth product was an open-source NIDS called Snort [2].

The detailed layout of the test-network is depicted in figure 2 below.

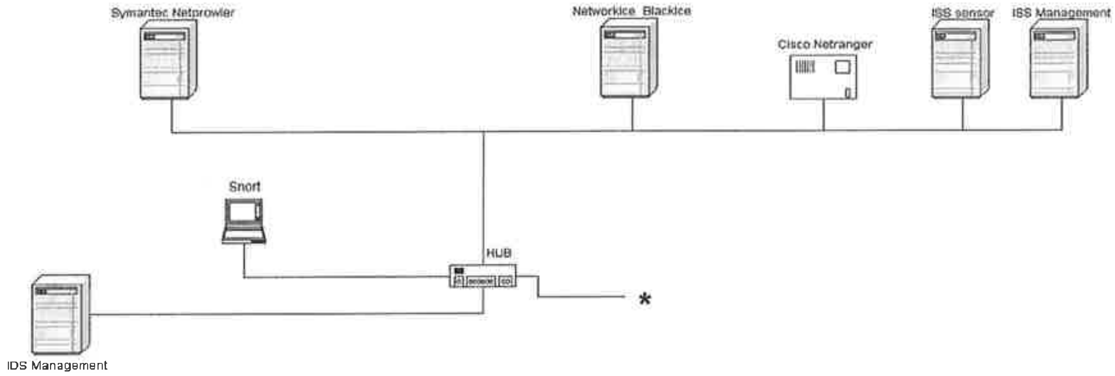

Figure 2: The test network

Testing network intrusion detection systems in a controlled laboratory environment involves the simulation of attacks and intrusion scenarios. We were however interested in real-life experience. Hence, the results of our experiments were dependent on the actual occurrence of erroneous and malicious packets which was, of necessity, beyond our control. In such a situation, it is difficult, if not impossible, to measure the accuracy (number of correct alarms divided by the total number of alarms) and completeness (of the number of detected incidents divided by the number of incidents that really occurred) of a product due to the lack of a controlled environment. In general, it is difficult to evaluate these aspects even in controlled laboratory environments as it is difficult to simulate sophisticated attacks. Furthermore, it is equally difficult to simulate realistic background traffic and a realistic background traffic load for the NIDS under test. MIT Lincoln Labs [3] evaluated NIDSs in a controlled test environment and in fact corroborated the above-mentioned difficulties with the simulation of attacks and background traffic.

The products used in our experiments are all signature-based. This means that they look for pre-registered patterns that could indicate a possible incident in the network traffic. Before a NIDS can detect an anomaly and signal an incident, it needs to know which byte pattern in (subsequent) packet(s) to look for in the traffic.

Such a pre-defined pattern is called a signature. Manufacturers of NIDSs usually deliver a few hundred known signatures together with the NIDS-products.

For an operational test of multiple NIDSs such as the one we envisaged, it was important to understand the overlap and differences between signature sets of these NIDSs, in order to understand differences in detected and undetected anomalies. To give an example, if only one of the NIDSs gave an alarm, this could mean that:

- the other NIDSs failed to detect the incident, or
- the alarm of the single NIDS that signalled an alarm turned out to be false,
- the other NIDSs did not have a signature of the incident.

The tested NIDS products typically consist of a management-console and one or more sensor-systems [5]. The sensor monitors the network-traffic for potential incidents. The management-console is used for the configuration of the sensors and for other NIDS-management tasks. The management display can be used for the real-time display of alarms as well, usually including an alert priority. Real-time in this case means: prompt detection of a possible incident by one of the sensors, communication of the alarm to the management-console, and then presentation of the alarm to the staff responsible for incident management.

We chose to let each product look at all signatures that were available from the vendor. Privacy sensitive signatures however were disabled. This approach has its advantages and disadvantages. The disadvantage was the increased complexity of comparing the incidents reported by the various NIDSs under test. The advantage of this approach was that we could learn more of what was actually happening on the network, and the way in which the various NIDSs behaved under identical conditions.

Problems with operating NIDSs

After the installation of the six NIDSs in the test network, we noticed that the NIDSs triggered a large number of alarms. Please note that this did not necessarily mean that we suffered from a range of incidents, or that damage was done. The alarms could be false; real incidents (including attacks) could have been stopped by preventive measures such as the boundary protection device (firewall), before reaching a target, and so on. These aspects will be discussed in more depth later in this section.

We observed that the large number of triggered alarms had a significant impact on the performance of the NIDSs. After a short period of operation, some of the products' databases suffered storage problems due to the huge amount of alarms. This required, amongst others, cleaning up the systems and making back-ups of the alarm data.

In earlier controlled laboratory experiments, we had already found that the NIDSs under test are very vulnerable to denial-of-service attacks. In our laboratory, we used the program Stick [4] to simulate thousands of attacks per minute. As a result, the performance of the NIDSs degraded significantly, in most cases resulting in a denial of service. The current experiments confirmed these problems in a operational test.

An important, initial observation during the experiments was that the NIDS products often disagree amongst themselves whether or not there was a possible incident and what type of incident that it could have been (even after taken into account the differences in signature sets).

Noticing this, we continued the experiments with a healthy suspicion about the accuracy and completeness of the NIDSs. A simple analysis showed us that, at best, one of the systems was accurate and complete. More likely, however, is that none of the NIDSs under test was accurate and complete. In order to solve the accuracy problem, i.e. to determine whether an alarm was false or not and whether the NIDS classified it correctly, we had to perform a thorough analysis of each alarm.

An alarm typically shows the following additional information on the management console:

- the name and/or type of incident
- source IP-address and protocol port
- destination IP-address and protocol port
- start time & date of the possible incident
- time & date of the last alarm of the incident sequence
- sensor that detected the incident
- priority

This basic alarm information is by far not enough to determine what really happened, whether security measures were breached, whether the target system suffered damage and who was responsible for the (potential) incident.

To accurately determine what caused the NIDS to trigger an alarm, additional information is needed about:

- The packet data that was analysed by the system and on which it based its decision to generate an alarm.
- The data that was *not* analysed by the system (missing some of the information, a NIDS could see that an intruder gathered a *PASSWORD* while in reality, someone just asked to *PASS the sWORD*).
- The signature and decision process of the NIDS: when and how does the NIDS consider something to be eligible for triggering an alarm? The signature is usually available to the user and often configurable, however the decision process and technical analysis details internal to the NIDS are unknown to the user. Although it is understandable from a business perspective that a manufacturer wants to keep these details company confidential, the use of, and the lack of access to, the NIDS can be limited by this absence. Note that since Snort is an open-source NIDS it does have the internal details available.

The following example illustrates the need for additional information with an alarm:

> *One of the NIDS showed a very large amount 'Stacheldraht' alarms. The NIDS could tell that 'Stacheldrah't is a distributed denial-of-service tool (DDoS). Further, desk research showed us that 'Stacheldraht' is a DDoS-tool that attacks systems with floods of ICMP traffic. Since we were quite certain that our test network was not suffering that many DDoS-attacks, we further analysed the traffic using a Sniffer application. This showed that a system in the network of the ISP was misconfigured, sending packets that contained the text-string: "This is not an attack". Still it was unclear to us what made the NIDS to decide that a DDoS-attack was occurring. In order to know this, a detailed study of the signature and the internal NIDS decision process would be necessary.*

Most NIDSs store some form of additional information with each alarm including the data bytes that the alarm decision was based upon. In general this is not enough to determine afterwards what had actually happened. For example, the data bytes that are not used in matching the signature, are often not displayed and logged. In some NIDS products the logged extra information is difficult to extract and to make visible, let alone to make it easily accessible through the user-interface. This implies that additional packet logging tools are required, such as sniffer applications. This also implies that, for proper incident management, a high level of expertise of both NIDS and protocol basics is required.

After having determined which packets were traversing the network and which ones triggered the NIDS, and whether the alarm was false or accurate, we still can not determine the potential impact of the possibly detected incident. To give an example, it is now difficult, if not impossible, to determine a priority for the response to an alarm. Even when an alarm is accurate, it is still unclear whether the 'dangerous' network traffic is able to pass or breach security measures between the location of the NIDS-sensor and the destination of the network-traffic.
This raises the question whether there is a relationship between the quality, capacity and potential interdependence of, and interaction between, the NIDS and the BPD. This important question needs further research as this combination is an implementation of the NATO security principle of defence-in-depth.

A question that needs to be posed, is: *can any damage occur?* To answer that question, detailed information is required about all of the following: the destination systems, their operating systems (versions and patch levels), the applications running on the various systems, all the configurations, and, last but not least, the security measures in place. Furthermore, the same information of the intermediate systems needs to be available to the NIDS-operators. To exemplify the foregoing, an attack on Windows NT will usually not harm a Linux-based system and an attack on an internal web-server will not reach its destination if a firewall filters the attack traffic.

After we analysed the accuracy of the alarm manually and assessed the potential damage of the incident, it is possible to determine (possibly some form of priority on) whether or not this specific alarm needs follow-up actions, i.e. incident response.
From this perspective, the real-time response aspect of NIDS has suffered a serious delay. Varying from minutes, if all necessary information is available to the NIDS operators, to hours if this information needs to

be gathered. Note that this is the case ***for every alarm!*** When a response to an incident needs to be initiated, this will typically include damage assessment, restoration and perpetrator/ attacker identification, digital forensics (also information forensics, or inforensics) and gathering of intelligence. These topics however are beyond the scope of this paper.

Determining the source of an attack can also be a problem. An attacker can use intermediate systems to hide its actual whereabouts, use falsified source addresses, and so forth. An incident response team can theoretically trace and identify internal perpetrators more easily, because the necessary information about users, network architecture, hardware addresses and the like is available to the organisation. The necessity of the latter is illustrated in the following example. We experienced that one system showed suspicious behaviour, in particular denial-of-service attacks, quite often. This turned out to be the proxy server, which by its nature, has to set up lots of connections for a large amount of users. This also means that if one of the users is involved in an incident, the proxy server accounting log needs to be analysed to find out the user's identity.

A small 'problem', after having analysed and responded to a single alarm, is that it would be useful to have some way of tracking alarms that have been already resolved. The current products do not have means to do this. This shows that most NIDS manufacturers do not have a full understanding for operational procedures, environments and settings of their products in a large infrastructure with heavy traffic.

The conclusion is that an alarm often needs to be analysed in detail. This process is often time, resource, and knowledge intensive. Furthermore, when dealing with 'too many' alarms, it will be necessary to have some form of prioritisation. These will be difficult to determine (and to define as well), because it will typically require information about the perpetrator(s), destination and intermediate systems, and the like. This information is often difficult to derive from the intruding log data that is supplied by the NIDSs.

Other problems with NIDS

A problem we address here is the scope or completeness of the NIDS-products. By design, the NIDS products search for patterns (or signatures) of (known) incidents in the network traffic. The use of heuristics, like anti-virus scanners do nowadays, is still in its infancy in NIDS-products. Hence, a prerequisite for detecting an attack is the availability of a proper signature for the type of attack. Consider a network with a wide variety of systems (different operating systems, hardware, and so on). Thousands of vulnerabilities have been published on the Internet (e.g. "bugtraq", "securityfocus"). A NIDS will generally only search for a few hundred of these attacks, often including signatures that are irrelevant for the specific network. As a consequence, the NIDS is far from complete in detecting possible incidents. In particular, novel or sophisticated attacks are not detected. Furthermore, the NIDS-products are lacking in detection of attacks for which they do have a signature. They are not 100% accurate!

When a new attack is published, a new attack signature is required to be able to detect the new attack with the NIDS. For obvious reasons, this signature has to be available to the system as soon as possible. Manual development of signatures is often possible. However, this would require a team of experts keeping track of new attacks and writing signatures. More convenient (although perhaps less reliable) would be that manufacturers update the signatures on a 24-hour basis. Currently, the update period varies from weeks to even some months! Can you imagine a virus scanner not being updated for weeks, let alone months?! Wouldn't the same hold true for NIDS-products?

It is disappointing to see that NIDS-manufacturers fail to update their products more frequently, devaluating the NIDS accordingly. Additionally, the manufacturers should allow users to customise their NIDSs to their own specific network environments, applications and operating systems, and hence bugs. Currently, this is not possible. The NIDS are far from complete in detecting attacks. In a way, this is extra worrisome as, by far, not all intruders and attackers are detected by the NIDS, despite the raised expectations.

A final problem we would like to mention here is the inability of NIDS to cope with encrypted traffic on the network with a NIDS-sensor. The use of encryption is increasing, with large manufacturers of software and hardware making the use of VPN (IPsec) connections easier to configure and use. In our specific situation this means that the NIDSs were unable to analyse tunnels that were set up from the internal network to destinations throughout the Internet.

The aforementioned problems are relevant for all of the signature-based NIDSs involved in the experiments. Finally, larger differences among the NIDS-products can be found in the usability and clarity of the user interface and the co-operation with other security measures.

Solutions to identified problems

Having identified such major problems, one starts to wonder if there are solutions to these problems. And, if so, what are these solutions? Some of the problems discussed are a direct consequence of aspects of the experiments, and/ or can be (partly) resolved, as we shall discuss in this section.

Firstly, there are two problems related to the large number of alarms the NIDS products had to deal with:

1. Performance of the systems,
2. Amount of required resources for analysis (and response) of an alarm.

The obvious solution to these problems is to decrease the number of alarms generated by the NIDS, in particular the number of false alarms. A good way to reduce the alarms is to install more preventive measures in front of the NIDS, or vice versa, locate the NIDS behind preventive security measures, such as the boundary protection services (e.g. firewalls). This in fact provides a trade-off between the number of generated alarms on the NIDS and the number of detected 'knocks on the outside of the network'. The priority and need to investigate incidents that only occur at the outside of the boundary of the corporate network can be low in case effective boundary protection measures block propagation of the incident to the corporate network. Locating the NIDS sensor behind the boundary protection to double-check the effect of the measure is nevertheless recommended in that case, as it forms an extra layer of defence. For example if a firewall blocks telnet traffic, the NIDS could have a signature that triggers an alarm if telnet traffic is passing the network (firewalls can fail as any other system), or it could be defeated and bypassed, in which case the NIDS should detect it.

Another way to reduce the number of alarms is to configure the priorities or signatures of the NIDS. It is essential for an accurate operation that a NIDS is fine-tuned to the specific environment in which it is deployed. Irrelevant signatures should be turned off, and tailor-made signatures may be required for specific mainframes or applications. Generally speaking, this requires a detailed threat assessment and vulnerability analysis of the ICT-infrastructure of the organisation deploying the NIDS. For example, the most relevant signatures can be chosen based on knowledge of the vulnerabilities that are present in the internal network. These signatures can then be configured with a high priority and alarms can be handled in real-time, whereas the low priority alarms could be aggregated in periodic reports. Determining priorities for attacks is, however, a non-trivial problem. Another way to fine-tune the NIDS is to configure thresholds for some signatures, which is possible for some types of signatures and NIDSs. A proxy-server for example has to set up a large number of connections and the threshold for a denial of service signature should be increased for these type of systems.

An obvious solution to the performance problem is to increase the performance of the systems. Of course, this solution is limited, but it may be cost effective. Equally, less human resources should be required to back-up or restore the system and so on. Note that whatever is done to reduce alarms, a significant amount of staff is still required to operate the NIDS (24hours x 7days a week). Furthermore, this staff needs a fair amount of technical expertise and knowledge of security. The problem of the NIDS vulnerability to denial-of-service attacks described earlier can be slightly limited by increasing the performance of the systems. However, the problem can not be resolved completely. It will always be one of the major disadvantages of signature-based NIDSs. It has also an important consequence: NIDS-systems shall be installed on systems separate of systems that offer application and other services. Otherwise, the NIDS can then be used as an instrument to attack those other applications and services. And, by the same token, the applications and services could be used in an attack on the NIDS. As a consequence, NIDS shall not be integrated in routers or installed on web servers.

A second area of problems, is the incompleteness and questionable accuracy of the products. The incompleteness stems from a more theoretical approach at signature detection. The accuracy problem was an inescapable conclusion from the disagreement and discrepancy between the NIDS products. The solution to the accuracy problem is that which was discussed in depth when the problems with operating a NIDS were addressed. To determine whether an alarm was accurately triggered or not, the raw data, signatures, decision process of the NIDS, and preventive measures on intermediate systems have to be analysed. This requires good tools, methods, and information at the hands of the expert NIDS operating staff.

The lack of completeness has to be taken for granted in part, so that at least a false sense of security is not present. Manufacturers can be stimulated to update the NIDS signature database more frequently and more customisable. Creating signatures manually would require an expert-team. However, the problem can not be solved completely. The products cannot detect new, unpublished, or sophisticated attacks, but frequent updates minimise this shortcoming.

It is a positive attribute for a NIDS-sensor to be a passive box, i.e. it can only receive data. The upstream link to the network is preferably unavailable, e.g. by means of traffic filtering. Or the sensor can operate on OSI-level 2, thereby making it invisible to Internet-users. In the latter case, the sensors should be connected to the management-consoles out-of-band (preferably using another network than the network being monitored). The communications between management-console and sensor should be encrypted to ensure authentication, integrity (including non-repudiation) and confidentiality of the data. All these measures minimise the possibility that the NIDS itself is attacked!

A view on the contribution of NIDS

Despite the numerous, partly unsolvable problems, we still see value in installing a NIDS. Our general philosophy is that either the user benefits from the NIDS, knowing its limitations, or (s)he sees nothing or little of what is happening on the own network!

The first contribution to the security of the network which we recognise is the basic operation of the NIDS, i.e. searching for malicious patterns in network traffic at any location in the ICT-infrastructure. The organisation that provided the test network experienced more sophisticated attacks that triggered the NIDS. Although, the system classified the attack incorrectly and only detected a small portion of it, the alarm was still extremely useful. It told us that something 'strange' was happening on the network and the systems involved. Manual analysis led to the discovery of a larger underlying serious problem (which is beyond the scope of this paper).

A second specific contribution of NIDS is that it can be an extra layer of defence or an addition to preventive measures, such as firewalls and security-gateways. A NIDS can be tuned to detect failures or circumvention of preventive measures, for example by implementing the firewall rule-set in NIDS signatures. When a NIDS-sensor is logically located in front of, and behind, a preventive measure, it is also possible to execute a differential analysis.

A third contribution is that periodic aggregated reports of the NIDS can be used to show statistical information about amount of, and trends in, anomalies on the infrastructure. Although possibly relevant for decision-makers and operators (depending on the level of aggregation), this is obviously not what was primarily in mind when acquiring, installing and operating a NIDS for real-time intrusion detection.

Recommendations for improvement

Based on the experiences with the different NIDSs, manufacturers and developers of Network Intrusion Detection Systems should:

- Improve the performance and effectiveness of the systems. Networks are becoming increasingly faster and the throughput will increase accordingly. It is expected that legitimate traffic will increase relative to malicious traffic (it goes without saying that both will grow rapidly). This begs for better mechanisms of detection and faster systems. This point needs to be emphasised given the current large numbers of false alarms that the systems already produce.
- Make the details of the signature, the systems' decision process, and the information that the alarm was based on, easily accessible through the user-interface. Furthermore, this information should be detailed enough to be able to quickly analyse whether an alarm is false. After all, time is of the essence, and is increasingly in shorter supply.

- Add a feature to keep track of the status of response to the system. As a minimum, it should be possible to provide an overview of the processed and unprocessed alarms.
- Perhaps most importantly, update the signature sets of the products more often and with extensive options for customisation of specific customer infrastructures.

Conclusions

Operating a NIDS, possibly with distributed sensors, in a large ICT infrastructure comes with major challenges. Some problems are situation specific, while others are relevant to all NIDS architectures and implementations in general.

Due to the size of the infrastructure, and perhaps its particular functionality, a 'large' amount of alarms needs to be dealt with. Partly, these alarms are false because the NIDS-products are inaccurate. Alarms may have a low priority, because they yield no damage or since preventive measures are located between the NIDS and target-systems. Additionally, not all incidents are detected, because the NIDS-products are incomplete. New and sophisticated attacks are not detected. The same occurs for incidents that are 'rare', or just have not been implemented in signatures by the manufacturers. Either way, a extensive amount of human resources, tools, methods, and knowledge is required to operate a NIDS. Hence, implementing and operating a NIDS is an expensive security measure.

Another set of problems is the performance issue. These issues are also related to the amount of alarms and traffic that needs to be monitored, as well as the vulnerability to denial-of-service attacks. Some of these problems can be partially resolved. The number of (false) alarms can be reduced by locating the NIDS behind preventive measures, and/ or by a good custom configuration and fine-tuning. Having done so, NIDS can have the following contributions, despite being expensive in both resources and knowledge:

1. *Basic intrusion detection*: either the user benefits from the NIDS knowing its limitations, or (s)he sees nothing or little of what is actually happening on the network!
2. *An extra layer of defence or an addition to preventive measures*: a NIDS can be tuned to detect failures or circumvention of preventive measures. When a NIDS-sensor is located in front of, and after, a preventive measure, it is also possible to see what attacks, that are detected by the NIDS before the measure, are stopped by this measure.
3. *Periodic aggregated reports by the NIDS*: these can be used to show statistical information about the amount of, and trends in, incidents on the infrastructure.

Acknowledgements

The authors would like to thank Peter Hupkens for his review of the paper and his valuable suggestions for improvement.

References

[1] Steven M. Bellovin (1989), *Security Problems in the TCP/IP Protocol Suite.* On-line: http://www.ja.net/CERT/Bellovin/TCP-IP_Security_Problems.html

[2] Snort web-site; www.snort.org

[3] Lippmann et al (2000), *The 1998 DARPA Off-line Intrusion Detection Evaluation.* RAID 2000 Proceedings, Lecture Notes in Computer Science, Springer, 2000.

[4] Stick; http://www.eurocompton.net/stick/

[5] Coolen, R., Luiijf, H.A.M. (2001), *Intrusion detection, Generics and State-of-the-Art.* NATO RTO TR-49, NATO RTA, Paris, France. On-line: www.rto.nato.int.