

Efficiency and Performance Issues in Distributed Intrusion Detection Systems

G. Koutepas, F. Stamatelopoulos, B. Maglaris
Network Management & Optimal Design Laboratory
Electrical & Computer Engineering Department
National Technical University of Athens, Zografou, GR 157 80, Athens, Greece
{gkoutep, fotis, maglaris}@netmode.ntua.gr

Keywords: Intrusion Detection, Architecture Efficiency, Trans-domain response.

Abstract. Distribution and hierarchy are the ideal means for providing load balancing and implementing systems with high scalability. In this paper, we discuss and identify performance bottlenecks and issues that reduce the efficiency of distributed Intrusion Detection Systems (IDSs) deployed within large enterprise networks. To minimize these problems and based on our recent experience in implementing distributed IDSs, we propose a hierarchical architecture that aims to decrease management traffic, support high levels of scalability and implement a distributed response ability spanning across different domains. The architecture consists of dual-role entities (manager/agent) that operate in multiple abstraction and hierarchy layers. We describe a top-level "domain IDS entity" structured to provide advanced functionality within the IDS organization and efficient communications with other such nodes outside it. We discuss how the proposed architecture can offer management and performance advantages during an attack. Further, we present work in progress on a pilot implementation of the architecture that is based on the Java Management Extensions API and the work of various groups (e.g. IDWG of the IETF [4]) on incident message exchange.

1. INTRODUCTION

One of the newest and most serious threats for computer network security has been the Denial of Service (DoS) Attack. Such attacks aim not to breach the computer systems themselves but at their very ability to provide services or to use the infrastructure that interconnects them. Although there may be no immediate destruction of equipment or stealing of information they usually result in the complete unavailability of the network's assets and services.

DoS attacks are floods of packets (usually with spoofed source addresses) targeted against one main machine in the network. Some alternatives may use a protocol specific operation (e.g. be TCP packets of a particular type like

SYN), trying to starve the resources of the target system. However, since the introduction of this type of attacks there have been improvements in the handling of these situations at the operating system level, patches that prevent complete resource starvation etc. The typical target of these attacks is the networking infrastructure; an overwhelming flood of packets that does not seek to achieve a connection is sent, eventually clogging the link or overloading the targeted service. Since this is difficult to implement and easy to detect if it is initiated from a single source machine, these attacks involve the use of a great number of trojan'ed or hacked systems. All these systems may be located anywhere around the world and are commanded to attack simultaneously or any other coordinated way. The result is a vast number of incoming packets from various links, practically impossible to trace to the source, constituting what is called a Distributed DoS (DDoS) attack. The ISPs, through which this traffic passes, may or may not want to initiate a response, unless security policies and/or contractual commitments impose such an action. Any reaction at this level is usually carried out manually.

It is mostly on this type of attacks that we focus our analysis here: we seek to identify the effectiveness of current IDS architectures in such cases, when they need to react with maximum speed and minimum extra load to the infrastructure.

Although not a new approach, the trend of interconnecting Intrusion Detection Systems in order to improve their data collection and analysis abilities is on-going and so far promising. The research efforts concentrate in developing intra-organization variable role systems that correlate and multiple single actions to detect a bigger pattern. Another focus is to effectively relay events and alerts between individual Intrusion Detection units. Still, the issue of interconnecting Intrusion Detection Systems is not entirely worrisome and may reveal disadvantages that make their operation less effective or even completely disable the detection capability. Inefficient IDS architectures can even hamper normal network activities. In this paper we try to

identify these problems within the context of extensive DoS attacks; we, then, propose ways to overcome them in an efficient and scalable way.

The rest of the paper is organized as follows: In section 2 we describe the typical distributed and/or hierarchical IDS architecture and then we identify and analyze the issues that affect its performance. Section 3 is an overview of a proposed architecture that aims to allow efficient IDS component cooperation within the enterprise network as well as across multiple domains. Section 4 presents work related to ours. Finally, section 5 summarizes conclusions, provides an overview of the current implementation work in progress and discusses evaluation methods for the proposed architecture.

2. IDENTIFICATION OF THE PROBLEM

Our work focuses on the operation of interconnected systems and not on the effectiveness of the intrusion detection process itself, e.g. the existence of false positives, efficiency of specific detection algorithms etc. Modern distributed Intrusion Detection Systems concentrate on discovering intrusion attempts that may come from numerous attackers and affect a large number of the enterprise's hosts. The types of attacks that mostly concern these architectures are scans or intrusion attempts from multiple sources and with multiple targets in the local network. Attackers will try to evade notice by making very slow scans, utilizing other, presumably innocent hosts and sending deceptive network traffic. Centralized, non-cooperating Intrusion Detection systems will fail to pick up any foul play in many such cases even if they monitor the total LAN traffic [8]. The solution is the combination of many such systems, on different hosts and different management hierarchy levels.

Recent trends in developing distributed Intrusion Detection Systems involve the installation of distinct, interconnected agents [10] rather than a monolithic approach, in the majority, or at least the strategically important hosts of a LAN. These undertake the tasks of data collection: measure all traffic and take clues from as many software components as possible. The next step is to transmit the data collected, with or without a first level analysis, to a central point for each sensor group. This gathering host analyses the data seeking patterns indicating malicious activity. This may be carried out slowly, hidden within normal traffic, coming from different sources or spreading to multiple hosts. The malicious activity sought after may range from simple vulnerability reconnaissance to a full-fledged attack. Usually, the next step is to perform a kind of data aggregation and transmit it to a higher-level analyzer that is in turn gathering information from many different LANs,

again trying to pick up those signs that will indicate that something is wrong. According to the size of the corporate network there may be many different such hierarchy levels achieving multiple layers of abstraction for the security analysis. However, after the second level there is no point in trying to further aggregate the data collected but rather it's more important to manage and make meaning out of all the messages and alerts that are exchanged and give the picture of the overall security condition to the administrator. Within any of these levels, host to top, an attack may indeed be detected and alerts have to propagate without delay either to the human operator or to an automatic system that will take the appropriate actions in response. Several approaches have been adopted for implementing variations of this basic architecture. However, they all share similar architectural problems, performance bottlenecks and drawbacks; the most serious are:

Communications Between Components

The first, obvious, concern in any architecture like the one we described is the amount of traffic generated between levels. The lower levels will attempt to combine information from many sources and perform first-level analysis and aggregation in their report. The nature of these notifications depends on the amount of intelligence of the IDS nodes. More intelligent nodes are able to make more in depth analysis and offer a comprehensive and concentrated report, but impose a bulkier computing footprint on the systems they are installed. Aggregation presents extra problems too, like the difficulty to define and gather at the low level components the data that will be more useful to the higher levels and for the network as a whole. The higher level analyzers may require the whole information set in order to clearly determine the characteristics of the attack. The result is extra traffic, generated both ways, upwards and downward, effecting network resources. Sensitive and "over zealous" components can seriously stress the infrastructure.

Information Exchange Within The Architecture

Attack signatures and optimal detection configurations, that may be well known and effectively used in one part of the network, have to be distributed to other partners too. This translates to a constant need for interaction and communication between the distributed system's components. It also, usually, involves the compulsory usage of the higher level with the corresponding delays and toll on performance. In a typical tree-like hierarchy the further we intend to send our reports the higher we have to scale the hierarchy. If this process is to be done automatically then there must be frequent scheduled and interrupt updates of this type, initiating bursts of traffic spanning large parts of the network.

Which Is The Highest Level

The responsibility of the modern IDS terminates at the domain border link (backbone link, Internet connection, etc.) They are not in the position to discover what is the situation outside their domain, if there are more networks with the same problems, or if the disruptions they are experiencing are results of major security events. More importantly they cannot communicate their findings to peer systems in uplink and adjacent domains that would find them useful in the prevention of new threats or the suppression of active attacks. Obviously any scaling capabilities of the architecture are limited within the domain and IDSs operate in domain isolation.

The Special Case Of ISP Networks

Examining trans-domain connectivity, we will notice that attacks passing through many links start to be detected only in the last stage, close to the target. Many times that is when it is already too late to respond quickly and effectively. Large transit networks, like that of ISPs, have trouble detecting attacks of small scale. Usually these have only minor effects to their backbone (with high capacity links) but are decisive to the connectivity of their client networks. An attack detected at the end network could effectively be limited or even blocked, much easier on the backbone links. That is something that today is done only manually, if available at all, resulting in waste of resources and effort. A DOS attack's effects on a client network link are usually immediate and cut off all downlink traffic for its duration.

IDS Response To DoS

Modern IDSs are focused in detection and combination of small scale events against individual hosts within a LAN. Their architecture scales parallel to that of the enterprise network. They concentrate on low level events and can be slow to react or even be overwhelmed during a DoS attack that is evident but massive. During such an attack we have a focus shift to the bigger picture which traditional IDSs are not able to do. The events that have to be examined involve the traffic patterns along links and the overall health and connectivity of the networks links.

Usually security problems target just one host or service and the consequences are limited to these systems. Contrary to this DoS attacks require urgent response, otherwise they prevent the ordinary operations of the whole network enterprise. Once a DoS attack has been detected and its characteristics established it is vital to be controlled through traffic shaping measures. Traditional IDSs offer no response capability of this type. Furthermore, even the administrator, once he is aware of the attack, has only limited abilities to control incoming traffic from within the network. It is very important that the response should primarily take place in

the backbone but the systems that detect the attack from its early stages cannot contact their findings upstream.

3. OUR PROPOSAL

So far we have considered the problems of modern hierarchical and distributed Intrusion Detection systems in a corporate network environment and especially in the event of a DoS attack. The solution we propose is based on our practical experience on the development and operation of such systems, such as the SiIDS prototype (a hierarchical distributed IDS developed by our team [6]), where we first encountered the problem of extending the detection and response capabilities outside the local domain.

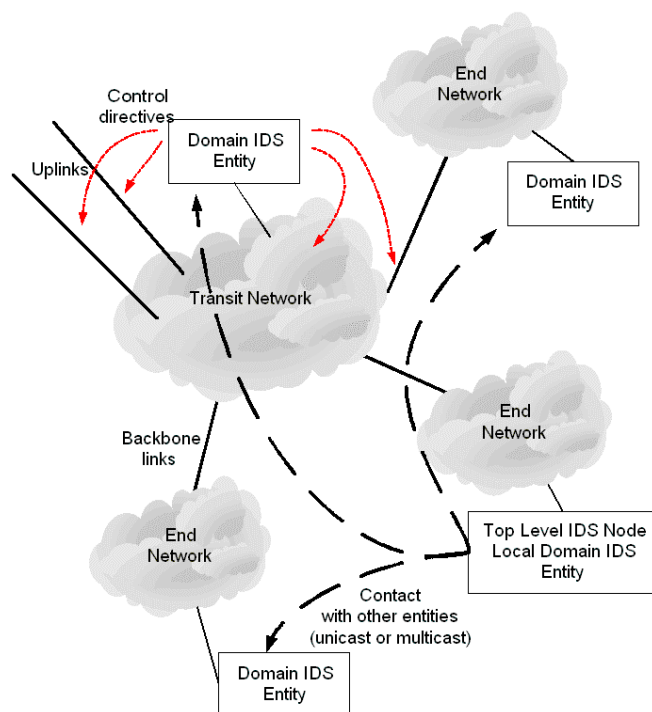


Figure 1: Proposed Architecture with cooperating domain IDS entities

More specifically we expand the top level IDS component of the hierarchy to become a "domain IDS entity" that participates in a hierarchy of cooperating, multi-domain IDSs. These entities have minor administrative and response capabilities and handle communications between cooperating domains. We intend to overcome the problem of "beyond the border response" by automating the cooperation within the framework of mutual agreement of the domains. We manage thus to build a web of automated response abilities much more effective than the manual approach. Figure 1 presents the proposed architecture along with some of the exchanged messages

However, even if we skip the part of having response abilities it is still beneficial if Intrusion Detection systems can exchange information on their findings and current security situation.

Parts Of The System

In each participating network there will be at least one special domain IDS entity that will (a) be trusted from the local network and have limited administrative abilities there, (b) cooperate with the local ID hierarchy, possibly being its top level, (c) have the ability to send and receive updates on the current security events or proliferate urgent security reports to the right partners and (d) be able to use ID information supplied from its own or other sources to enhance the detection capabilities of the group and even deter on-going attacks that have rendered other parts of the network unusable.

Local Network Access

Any security anxious administrator would be quick to dismiss the idea of having an automated component that will have administrative mandate in his network, even more so if its actions will be the results of automated decisions or initiatives of other networks. In this respect we determine the extend to which it is possible for the domain IDS entity to effect network operation by the usage of three safeguards: A policy file edited by the administrator that specifically states the active network components that can be affected by the domain IDS entity and the degree to which that can be done, e.g. regulate the rate limiting of a link to 10% for each different source and in addition not allow more than three administrative actions from the entity. An administrator that worries of a compromised entity getting out of hand can further restrict its connectivity with the network through access lists, firewalls and component access control. Second the changes will be for a limited time only (e.g. 10 minutes) enough to contain a full fledged DoS but with only a limited effect if it's a false positive. Third, all changes are sent as urgent messages to the management console of the network. This serves two purposes: to keep an accurate account of all the actions taken but also to give the administrator the prerogative to apply these restrictions manually for a more permanent period.

All the three restrictions would be better enforced if the domain IDS entity was implemented as a two module architecture. One module undertakes all communications, authentication, response initiative and rule compilation, the other one filters the actions of the first through the policy file, checks for integrity and informs the network manager of actions taken.

Local IDS hierarchy

The domain IDS entity can either be part of the local distributed IDS installed in the corporate network or just act as an intermediary between the results produced from it and the committed obligation to take action and communicate findings. Especially in cases of DoS attacks there is need to establish its characteristics and contact these to the exact places where reaction can be useful.

In this part we want to add a degree of pro-active reaction to enable quick response to security events that can severely impact the network, like DoS. The administrator compiles a list of important, and mission critical hosts, as well as LAN and connection links of the corporate network. The system then focuses its attention to these. Instead of searching through all the data generated by the low level IDS components (that data being aggregated or not) we focus on the critical elements that are more likely to be targets of malicious activity. We thus acquire a picture of the operational health of these components and the network in general. The domain IDS entity can contact its interpretation of these results to its partners. If network communications are then severed the sibling entities would have some indications of what could have happened and start their reactions.

Response Operations And Communications

Communications between entities are crucial to achieve effectiveness but also should be carefully planned to avoid overloading already overused network resources on the event of an attack.

In the area of message propagation we borrow from the concept of "interest driven communications" close to what is described in [9]. Each of the entities has a subscription list with all the other entities that it contacts, usually the adjacent ones. Actually this is a list created by the site's administrator according to his security, resource usage, detection policies and cooperation needs. More than that, the policy file mentioned earlier can also be used to restrict or ignore messages from particular or all other networks. One can view this as a strict "communicate only with those needed" approach, same like "watch the important hosts only" policy followed on the detection part. One promising approach to these communications is also multicast where we can combine the "need to know" basis with the concept of group membership.

We expect the local IDS components to determine the specifics of the attack, the source (rather improbable if source spoofing has been used), the final target, the protocol and ports used etc. Once these are established the domain IDS entity has to consider ways of reaction. The first action should be to apply filters to local links. This assumes that the

entity should have knowledge of local network mapping and be able to determine which links are more appropriate to act upon. As this will become apparent further on, this knowledge is an integral part of the system's effectiveness.

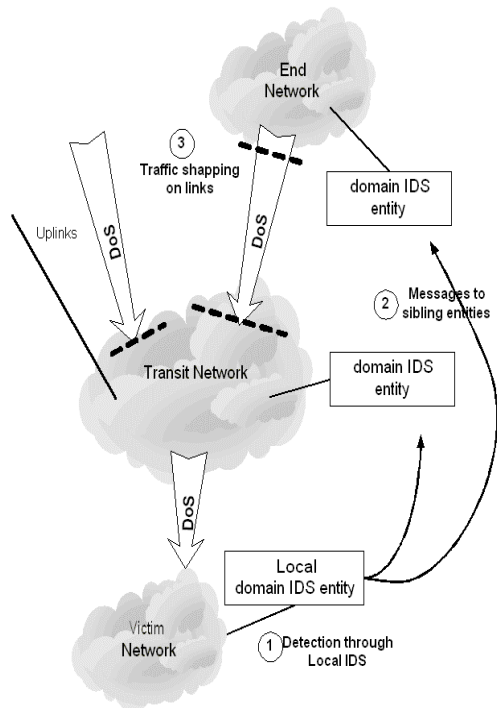


Figure 2: Coordinated reaction during a DoS. (1) The Local domain IDS entity detects the attack through its IDS hierarchy and attempts to inform the sibling entities (2). Either through these messages or through the notification cooperation of other domain IDS entities we have response to limit offending traffic on the appropriate links.

If the attack is coming from the uplink there are further actions to be taken. The entity has a general knowledge of the overall connectivity of the various cooperating domains, at least in AS level. The attack's characteristics, already determined earlier are contacted to the other entities. Of course here we encounter the problem of a possibly clogged link. The first solution is to attempt to pass the absolutely necessary information to the entity of the closest partner using the (free during a DoS) upward link. In such a case unacknowledged UDP packets can be used. Still, it is here that the distributed nature of the system comes into play. Even in the case that contact is completely severed the other cooperating entities are in position to determine, from their own IDS ability what is happening and initiate appropriate responses. The detection process takes place, to a big extent in parallel, using info available from both partner entities and local sources. The result is a more rapid propagation of attack acknowledgment and completely automatic responses.

Figure 2 gives the basic reaction process upon the detection of a DoS attack.

Extension of the Architectural Paradigm

The same advantages of focused, proactive and well scaling detection and response capability could be obtained in the local network if we implement the same paradigm there. The dual role entities described above can be positioned in strategic spots and monitor the health of important hosts and network assets. They can be the local gathering points of alert and event traffic, avoiding its propagation to the rest of the domain. And here we can utilize the same methods of limited, multicast type communications to avoid putting stress on the resources. In this a concept, we focus the distributed intelligence and communications control of our IDS hierarchy to a few well positioned, efficient (and well protected) entities.

4. RELATED WORK

The basic idea for the cooperating domain IDS entities comes from problems researched by our team for Managed Bandwidth Services [7] and solutions developed there. Although the Intrusion Detection Systems have been cooperating more and more, producing various architectures there is not much interest in the problems of detection and response of cooperating systems in different domains. The most notable efforts in the field are the IETF Intrusion Detection Message Exchange Format WG [4] and the European NRNs' WG to define a common data format and common exchange procedures for sharing information needed to handle incidents between different CSIRTs, called IODEF [5]. Still, both these efforts focus on the messages themselves and do not propose any specific architecture.

Research work closely related to ours is the Aggregate-Based Congestion Control [3] in the field of DoS response and the Cooperative Intrusion Traceback and Response Architecture (CITRA) [1-2] in the field of the cooperative components architecture. The former is a method to analyze dropped packets in the routers to discover offending traffic aggregates and to contact these upstream by a proprietary protocol (pushback), thus achieving rate limiting in "step-by-step" manner. In comparison our system operates by higher-level entities, using indirect links many times, thus overcoming non-cooperating domains. Further more we take a more general approach leaving the response decisions to the levels higher in the management hierarch. This makes our approach independent from specific implementations. The latter describes a system much like ours which achieves intrusion response by tracking an attack stream though the cooperating infrastructure, going back from the first node that detects the activity to as close to the attackers source as possible. Our approach is different in the fact that (a) there

is direct contact to the cooperating entity when we know the source thus achieving faster response, (b) when the source is not known or the attack comes from multiple sources we activate efficient multicast type methods of communication and (c) the proposed systems does not rely so much in all the domains between the source and victim to be in the cooperating infrastructure but we rather try to contact all that are reachable.

5. SYNOPSIS - FUTURE WORK

We have presented an architecture for achieving more efficient and timely cooperation between components of a distributed Intrusion Detection system, expanding the detection and response capability across domains and keeping the load on infrastructure resources to a minimum. The architecture proposed is independent of the underlying active network components, and scales well without requiring the full cooperation of every domain in order to achieve effective response. Moreover it allows significant autonomy to the local administrator. Through the usage of policy files he can define the degree of participation in the framework and the extend that he will accept configuration directives from a remote IDSs response initiative.

We are currently implementing a prototype of the multi-role domain IDS entity using the JMX Java API. We're also in the process of specifying the main messages exchanged between nodes in each scenario and the policy semantics that will go to the corresponding configuration files. The prototype will be installed on different systems and initially serve as proof of concept for configuration, message exchange and distributed identification of events.

Testing the proposed architecture in real conditions is not practical since it would involve the installation of probes in many different domains and initiating full scale DoS attacks on the infrastructure. In order to evaluate it we intend to follow two approaches:

A test-bed installation will be implemented using a number of systems interconnected through a small number of routers and switches. Each one of the nodes will be configured to play the role of the domain IDS entity on a different network. Additional machines on this model network will then initiate a number of typical DoS attacks. We will take measurements during the process of the IDS component reaction to test for any reduction on the effects of the attack, the extra load produced by their messages and the effectiveness of policies and configurations in controlling the distributed system.

In the second phase we plan to use the NS2 simulation toolkit to map a typical multi domain installation involving

networks of various sizes and connection speeds. We intend to test and fine tune the reaction of the cooperating IDS nodes to a number of different attacks. These tests will define the optimal configurations and reaction policies and will help us optimize the messages exchanged between nodes. Our hope is to achieve reaction capability for multiple attack scenarios (them being DoS or even single events) parallel to minimizing the load on network's resources.

6. REFERENCES

- [1]. D. Sterne, K. Djahandari, B. Wilson, B. Babson, D. Schnackenberg, H. Holliday and T. Reid, "Autonomic Response to Distributed Denial of Service Attacks," In *Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection, RAID 2001, Davis, CA, USA, October 2001*, pp.134-149, Springer-Verlag
- [2]. D. Schnackenberg, K. Djahandari, D. Sterne, "Infrastructure for Intrusion Detection and Response," In *Proceedings of the DARPA Information Survivability Conference and Exposition (DICEX II), Hilton Head, CS, January 2000*.
- [3]. S. Floyd, S. Bellovin, J. Ioannidis, R. Mahajan, V. Paxson and S. Shenker, "Aggregate-Based Congestion Control and Pushback," *ACIRI Annual Review*, December 2000
- [4]. Intrusion Detection Working Group (idwg) of the IETF, <http://www.ietf.org/html.charters/idwg-charter.html>
- [5]. Incident Object Description and Exchange Format Working Group, <http://www.terena.nl/task-forces/ff-csirt/iodef/index.html>
- [6]. P. Astithas, G. Koutepas, A. Moralis, B. Maglaris, "SIDS - A System for Enterprise-wide Intrusion Detection", In *Proceedings of the International System Security Engineering Association Conference '01*, Orlando, USA, February 2001
- [7]. F. Stamatelopoulos, G. Koutepas, P. Astithas, B. Maglaris, "End-to-End, Multiple-Domain Bandwidth Management," *Poster*, HP OpenView University Association (HPOVUA) Conference '01, Berlin, Germany, June 2001
- [8]. T. Ptacek and T. Newsham, "Insertion, Evasion, And Denial Of Service: Eluding Network Intrusion Detection," Technical Report, Secure Networks, Inc., January 1998
- [9]. R. Kopalakrishna, E. Spafford, "A Framework for Distributed Intrusion Detection using Interest Driven Cooperating Agents," *Fourth International Symposium on Recent Advances in Intrusion Detection, RAID 2001, October 2001*
- [10]. J. Balasubramaniyan, J. Garcia-Fernandez, E. Spafford, and D. Zamboni, "An Architecture for Intrusion Detection using Autonomous Agents," Technical Report TR 98-05. Department of Computer Sciences, Purdue University, 1998