An Autonomic Intrusion Detection System based on Behavioral Network Engineering

Claudio Mazzariello & Francesco Oliviero
{cmazzari, folivier}@unina.it
Advisors: Prof. Simon Pietro Romano & Prof. Carlo Sansone
Dipartimento di Informatica e Sistemistica, via Claudio 21, 80125
University of Napoli "Federico II", Napoli (Italy)

***Abstract*— As network infrastructures become more and more complex, network protection, accordingly, becomes a harder and harder task to fulfill. In order to cope with the frequent changes in the monitored scenarios, we propose a system which, by observing users' behaviors and auditing the security status of the network, exploits grid services in order to correctly deploy the required security modules, aimed at assuring a prescribed security level.**

I. Introduction

The wide deployment of new services and applications over the Internet imposes the careful project of network infrastructures, in order to face the new potential threats coming from either accidental events or malicious users. Indeed, the complexity of the offered services, as well as the everchanging new threat methodologies, require the definition of a new approach to network security. This new approach must cope with the problem of both defining a new user behavior (in order to point out the potential malicious usage of the network), and developing new systems capable to dynamically detect and react to threats. According to the new paradigm named *Autonomic Communication* [1], the issues above should be faced by assuming at the outset that network entities (either network users or the components of a defensive system) can be grouped into *communities*, each community representing a set of elements sharing some features and interests and cooperating in order to reach a well-defined objective.

Based on the above considerations, we have defined a novel approach to network analysis, design and implementation, named *Behavioral Network Engineering*. Such an approach concerns all services and applications that require knowledge of the network users' behavior in order to effectively operate. Intrusion Detection represents an interesting example of an application which can highly benefit of the adoption of such an approach.

In the following we will first introduce the general idea of Behavioral Network Engineering, by proposing a model for the definition of user's behavior. Then, we will describe a framework for intrusion detection based on such a model, and embracing the "autonomic" approach.

II. Behavioral Network Engineering

Network services and applications mandatorily impose to adapt the network to new users' needs through the development of innovative techniques for traffic and resource engineering. Given the strict correlation between users' behavior and network resources management, it becomes of paramount importance to obtain users' information in order to optimize network operation. To assure this requirement, we propose a comprehensive approach, named *Behavioral Network Engineering*, which aims at exploiting data about users' behavior in order to effectively manage, secure and dimension the network.

Behavioral Network Engineering exploits information about both the user's behavior and the current network status in order to define the proper actions to be performed onto the network. *Context* information and users' profiles together contribute to the extraction of the behavioral knowledge needed to successfully build an *autonomic* system.

As stated before, network security systems, and in particular Intrusion Detection Systems (IDS), represent the main networking application exploiting users' behavioral information in order to detect malicious activities ongoing in the network. An IDS needs to know what all users are doing, in order to effectively face the threats carried out by the malicious ones. Moreover, this information must also take into account the relationships among the sets of users sharing common features, resources and purposes. Unfortunately, the current real-time intrusion detection systems do not adopt any user characterization including such "inter-users" information. A Distributed Denial of Service (DDoS), for example, is generally realized through coordinating activities involving actors which are widely distributed throughout the network. According to this property, we claim that the detection of such attacks also requires the knowledge of information regarding the "cooperation" among different network entities. Such entities share the same "purpose" (i.e. the compromission of either a service or a single host) and make use of the same resources (e.g. those belonging to the network infrastructure) in order to fulfill their task. For this reason, the set of hosts involved in a DDoS attack can be considered as a "community", whose members all aim to achieve the same result. Thus, the information about this community as a whole can also contribute to realize a deeper knowledge, i.e. the behavioral information, needed to improve the detection process.

According to these considerations, for the proposed IDS we have adopted a user behavior model which includes both single

user and community information. An example of such a model is the one proposed in [2]. Lee and Stolfo identify every single network connection as a network user. User behaviors are in turn specified by means of a set of parameters collectively known as *connection features*. Among such features, those containing information about a single connection are called *intrinsic features*. On the other hand, features summarizing information about all the connections sharing common properties are called *traffic features*. In our model, these last features represent the information on the user community needed in order to effectively identify malicious behaviors.

III. A FRAMEWORK FOR INTRUSION DETECTION

In order to develop a complete system able to detect intrusions, it is well known that several components are needed, each in charge of implementing one key functionality of the system. Of course, we need:

- a *sniffer*, committed to capture packets while they flow through the network;
- a so called *preprocessor*, which extracts high level information about the behavior of a particular entity of the network. This is done by elaborating sniffed packets;
- a *classifier* which is in charge of deciding, based on some set of classification criteria, whether or not the examined behavior is normal.

We propose to look at the network as a self-aware and a self-healing environment. In such an environment, a distributed framework for network protection, still complying with the inherently monolithic IDS structure described above, and capable of providing a previously agreed upon protection level, is well suited. A distributed system allows the separation of concerns among a well-defined set of entities, each suited to deal with a particular aspect of the problem. This on one hand simplifies the task of each involved entity, and on the other hand allows a deeper specialization of each module (which can thus be modified without necessarily affecting performance of the overall system). If we assume that the network be aware of itself, security assurance might be regarded as a *service* inherently provided by network infrastructures. In such a scenario, a framework capable to deploy, both proactively and reactively, on-demand security services is well suited. The IDS structure described above relies on single entities, each performing just one of the prescribed tasks. A distributed IDS, instead, would naturally adapt itself to a much more context-aware deployment of the available resources, thus allowing a more effective placement of the modules.

We propose a system which, by means of a grid-based infrastructure, dynamically deploys the entities in charge of providing network security, based on the knowledge of the *security status* of the network and its components. In order to accomplish its task, the IDS might need several probes sniffing traffic in crucial points of the network, and several classification nodes, each exploiting the best fitting detection technique according to the system status and node location. A *broker* entity is also needed, capable to instruct all the network nodes to execute a specific application server together with the appropriate grid services.
Such a dynamically distributed system might, for instance, adapt itself at the occurrence of a DDoS attack, by appropriately placing intrusion detection engines in the most critical nodes (i.e. the nodes along the attackers' path) and by coordinating such nodes through a flexible signalling protocol (e.g. a protocol for tracing back the attack).

A. A multiclassifier approach

As far as intrusion detection is concerned, there exist four possible outcomes for a system attempting to classify users' behaviors. Indeed, besides the correct detection of normal behavior or of an attack pattern, a classifier might mistake an attack for normal behavior, thus resulting in a missed detection, or some normal operations for an attack, thus generating a false alarm. The correctness of such outcomes of a classifier is the main criterion for evaluating the effectiveness of such instruments. Though mistaking the occurrence of an event for any other event might seem equally harmful, regardless which class of events is mistaken for which other class, not all errors are equally severe. In the context of intrusion detection, missed detections have different importance than false alarms, and tougher endeavors are needed for coping with the minimization of the one out of such two parameters which is deemed to be the most important. Unfortunately, there exists a well known trade off between false alarms and missed detections, which can't thus be minimized together.

By exploiting the natural capability of the aforesaid distributed framework, we can try to decrease the number of classification errors by exploiting the detection capabilities of several classifiers of different types, also using artificial intelligence algorithms. While a single classifier can't be too specialized in solving each of the problems it faces, several smaller classifiers may be chosen in order for each one of them to cope with a subset of the problem. This allows us to reach a higher specialization degree, and to perform a better detection on the few attack classes that a single classifier can discover within the network traffic. Multiclassifier systems are widely used in intrusion detection problems, and they usually perform better than systems based on a single classifier, thus encouraging in the exploitation of such a technique.

IV. CONCLUSION

A self-aware and self-organizing IDS is proposed, which exploits several artificial intelligence techniques, by suitably combining their outcomes, aimed at correctly detecting intrusions, by providing the security administrator with as few false alarms and as many correctly detected attacks as possible.

REFERENCES

[1] M. Smirnov, *Autonomic Communication - Research Agenda for a New Communication Paradigm*, White Paper, November 2004.

[2] W. Lee and S. J. Stolfo, *A framework for constructing features and models for intrusion detection systems*, ACM Transactions on Information and Systems Security (TISSEC), 3(4):227–261, November 2000.