Agent-Based Network Intrusion Detection System

Vojtěch Krmíček, Pavel Čeleda
Institute of Computer Science
Masaryk University
{vojtec,celeda}@ics.muni.cz

Martin Rehák*, Michal Pěchouček†
Center for Applied Cybernetics*, Department of Cybernetics†
Czech Technical University in Prague
{mrehak,pechouc}@labe.felk.cvut.cz

Abstract

The paper presents security platform based on agents as an efficient and robust solution for high-performance intrusion detection system designed for deployment on high-speed network links. The proposed detection algorithm is based on extension of trust modeling techniques with representation of uncertain identities, context representation and implicit assumption that significant traffic anomalies are a result of potentially malicious action. The heterogeneous anomaly detection methods are used by cooperating agents and then correlated using a reputation mechanism. To satisfy the performance requirements, wire-speed data acquisition layer is based on hardware-accelerated NetFlow probes that provide overview of current network traffic. The output of multi-agent detection layer is presented to operator by a dedicated analyst interface agent, which retrieves additional information to facilitate incident analysis. Our performance results illustrate the potential of combination of high-speed hardware with agents-based detection and advanced analyst interface.[1]

1. Introduction

The goal of the presented work is to use an agent platform as a security layer in the Network Intrusion Detection System (NIDS), together with low-level high-speed traffic acquisition and preprocessing layer based on dedicated adaptive hardware and high-level operator interface. To face the problem of high false positive rates in today's NIDS, the proposed mechanism of the security agents platform is based on extension of trust modeling techniques with representation of uncertain identities, context representation and implicit assumption that significant traffic anomalies are a result of potentially malicious action.

[1]This material is based upon work supported by the European Research Office of the US Army under Contract No. N62558-07-C-0001. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the European Research Office of the US Army. Also supported by Czech Ministry of Education grants 1M0567 and 6840770038.

In our approach, we have decided not to develop a novel detection method, by rather to integrate existing methods with an extended trust models of a specialized agent. This combination allows us to correlate the results of the used methods and to combine them to improve their effectiveness. Our main research goal is to combine the efficient low-level methods for traffic observation, with multi-agent detection process to detect the attacks with comparatively lower error rate, and to provide the operator with efficient incident analysis layer. Analysis layer supports operator's decisions about detected anomalies by providing additional information from related data sources. It is also responsible for visualization of the anomalies and the detection layer status.

2. Related Work

The basic principle, how to detect and uncover the network anomalies, especially without any feedback from the affected hosts, is to analyze the patterns in the traffic data, compare them with normal behavior and conclude whether the irregularity corresponds to a known attack profile or not. This approach to Network Intrusion Detection, typically based on the flow information is currently an important field of research into *anomaly based intrusion detection*. Numerous existing systems, based on traffic volume analysis modeled by Principal Component Analysis methods [3], models of entropy of IP header fields for relevant subsets of traffic [13, 5], or just count of the flows corresponding to the selected criteria [2] offer each a particular valid perspective on the network traffic.

3. Anomaly Detection Approaches

Table 1 provides a very high-level assessment of estimated usefulness of the proposed methods for the autonomous detection of malicious traffic, more particularly in the context of agent-based detection mechanism presented in Section 5. The general pattern is clear – using the NetFlow data, we can detect mainly the attacks based on high number of near-simultaneous flows, regardless of

the fact whether these flows share the source IP, destination IP, ports or any combination of these features. Therefore, the number of bullets mainly represents our estimation of false positives/negatives rate measured when the method is used to identify the given attack.

The MINDS system [2] represents the flow by basic NetFlow aggregation features (srcIP, srcPrt, dstIP, dstPrt, protocol) and complements them by the number of the flows from the same srcIP, to the same dstIP and their combinations with dstPrt and srcPrt respectively. These properties are assessed both in time and number of connections defined windows, to account for slow scanning.

The system proposed by Xu *et al.* [13] for traffic analysis on backbone links also uses the NetFlow based identity 5-tuple. The context of the single connection is defined by the normalized entropy of srcPrt, dstPrt and dstIP dimensions of the set of all connections from the srcIP of the flow in the current time frame.

Another perspective anomaly detection mechanism can be based on the observation of traffic volumes in high-speed network. Lakhina *et al.* [4] uses statistical modeling to identify the anomalous origin-destination-aggregated flows. The method is based on Principal Component Analysis. In another work of the same authors [5], the PCA method is used to model the normal and residual entropy, to remove the systematic elements of the data before clustering. The clustering then emphasizes the anomalous characteristics of the traffic.

Technique	*MINDS* [2]	*Xu* [13]	*Volume* [4]	*Entropy* [5]	*Patterns*
IP address spoofing	●	● ● ●	●●	●	
Host scanning	●●	● ● ●	●●	● ● ●	
Host profiling				○	● ● ●
IRC coordinated attacks	●●			●●	●●
Buffer overflow					●●
Flooding	●●	●●	● ● ●	● ● ●	

Table 1. The relevance of the detection and attack techniques.

4. System Architecture Overview

The whole system architecture consist of these three layers: traffic acquisition and preprocessing layer, agent security platform layer and operator analyst layer. The demands for each layer at online processing, artificial intelligence and visualization process vary a lot. While the low-level layers need to be optimized to match the high speeds during the traffic acquisition and preprocessing, the higher layers, using preprocessed data are inferring the conclusions regarding the degree of anomaly.

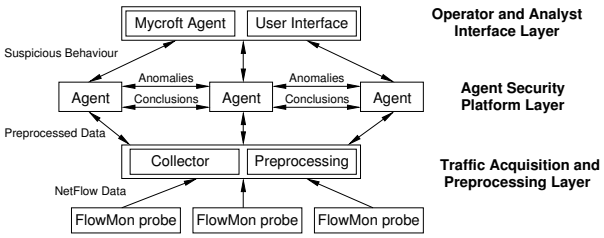

Figure 1. System overview.

• **Traffic Acquisition and Preprocessing Layer:** The components in this layer acquire the data from the network using the hardware accelerated NetFlow probes [1] and perform their preprocessing. This approach provides the real-time overview of all active unidirectional connections on the observed link. In order to speed-up the analysis of the data, the preprocessing layer aggregates meaningful global and per-flow (or group of) characteristics and statistics.

• **Agent Security Platform Layer:** This layer consists of specialized, heterogeneous agents that seek to identify the anomalies in the preprocessed traffic data by means of their extended trust models [10]. Their collective decision regarding the degree of maliciousness of a flow with certain characteristics use a reputation mechanism. The agents run inside the A-Globe agent platform [12] and use its features like agent migration and cloning to adapt the system to the traffic and relevant threats.

• **Operator and Analyst Interface Layer:** The agent security platform is coordinated by an intelligent agent called Mycroft. This agent works as an interface between the detection layer and the network operator. Every detected suspicious behavior on the network is automatically reported to Mycroft.

5. Agents Security Platform Principles and Methods

The added value of our platform is the cross-correlation of various anomaly detection (i.e. network behavior analysis) methods using the extended trust modeling [10]. Classic trust models developed in agent research [11] ignore several features that are essential in our domain: uncertain identity modeling, context modeling and non-existing (or severely delayed and limited) feedback.

Baseline trust models evaluate the behavior of individual agents, whose **identity** is guaranteed (to an extent) by the computational environment. In the network domain, we have to evaluate the trustfulness of network flows, and while they can be distinguished as unique identities, this distinction is unpractical due to their number and ephemeral nature. We represent the connections in a metric space, cluster them using an agent-specific metrics based on the NetFlow 5-tuple. However, merely representing the flow identities in

the metric space and evaluating their trustfulness gives unsatisfactory results, as it ignores the most important information from the NetFlow data – the information about the other, similar flows in the current traffic sample. This information constitutes the **context** of the trusting decision [8], and together with the identity defines the Identity-Context metric space, where the detection agents assess the trustfulness of flow representations. Each of the agents uses its own particular context space, dependent on its anomaly detection method. Definition of context information for current agent types can be found in the related paper [7].

The principal input of classic trust models is a result of past cooperations with the partner: quality of service, degree of success, on-time delivery and other domain specific parameters. In our case, it is very difficult to obtain the **feedback** that can be associated with the current traffic on the network. Therefore, we use the information regarding the flow anomaly *as assessed by the other agents* to replace the direct feedback, therefore connecting the anomaly detection between diverse agents.

While processing the information about the network flows, each trusting agent receives an identical copy of network flows list and associated pre-extracted statistics. Then, it determines the anomaly of each flow and shares it with other agents. The agent also receives the anomalies from the others, and starts the flow processing by its internal trust model. The trustfulness in the model is not associated to individual flows, but rather to selected objects in the Identity-Context space [7]. Individual flow is therefore represented by its identity and context in the metric space. Then, we retrieve the positions of nearby cluster's centroids (with attached trustfulness) from the current trust models and update their trustworthiness with an aggregated degree of flow anomaly as determined by the other agents. The details of the approach are presented in [10].

Performance of an isolated detection agent would be the same as a performance of the anomaly detection method it is based on. As we have suggested above, the agents base their evaluation of trustfulness not only on their local results, but also on the anomaly opinions of other agents. We argue that this cross-correlation will help to filter-out most false positives on the level of individual agents. In the second phase of evaluation, each agent selects the flows it considers as malicious and shares these flows with others. Agents then use a simple voting protocol to reach a collective conclusion regarding the estimated maliciousness of anomalous flows, further reducing the number of incidents reported. Collectively accepted flows are then sent to analyst interface layer.

The decision whether a given flow is trusted or untrusted depends on the typical degree of anomaly in the observed network traffic. This parameter varies widely with network type – the number of anomalies is typically low on managed, internal networks, but is significantly higher on public Internet backbone links. To avoid the problems with manual tuning of the system, we use a fuzzy-inference process integrated with the trust model to automatically adapt the model to the natural level of anomaly in the environment [9].

6. Agents Platform Evaluation

To evaluate vertical integration of all three system layers and to test capabilities of detecting malicious traffic, we have designed an use case intended to evaluate first capabilities of the whole system and the agent platform. Malicious traffic is presented by vertical port scan, performed against a personal computer inside the university network.

6.1 Malicious Traffic Description

We have based our use case attack on vertical port scanning using the Nmap [6] tool. Nmap is a free security scanner, used to evaluate the security of computers, and to discover services or servers on a computer network.

The port scan was launched as *TCP SYN scan* against the targeted machine. TCP SYN port scan is the most popular form of TCP scanning. Rather than use the operating system's network functions, the port scanner generates raw IP packets itself, and monitors responses. This scan type is also known as "half-open scanning", because it never actually opens a full TCP connection. The port scanner generates a SYN packet. If the target port is open, it will respond with a SYN-ACK packet. The scanner host responds with a RST packet, closing the connection before the handshake is completed.

6.2 Agent Security Platform Results

The graphs presented in this section show the distribution of trusted and untrusted traffic (or reference contexts used by the agents to assess the traffic) over the trustfulness scale. Completely trusted traffic would form a single peak at 1, while the untrusted traffic shall accumulate in the left part of the distribution.

In the Figure 2 we can observe the data before the attack which corresponds to normal traffic, albeit with slightly elevated rates of abnormal traffic reported by Lakhina Entropy method and MINDS.

The situation changes dramatically when we perform an attack, which is barely perceptible in traffic volume data. The Figure 3 clearly shows the peak of untrusted flows in the left part of the distributions, identified by all methods.

In this simple experiment, the detection layer demonstrated that it is ready to be used for the next stage of development and research. It has achieved its objective of selecting the suspicious traffic that requires inspection, and allowed the user interface layer agents to present the operator with a clear, well defined case to resolve.

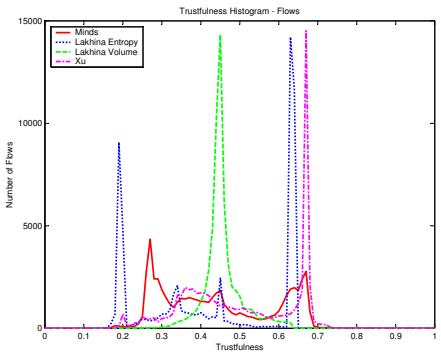

Figure 2. Normal traffic before the attack.

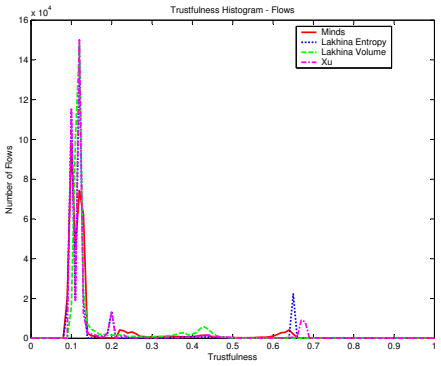

Figure 3. Malicious traffic during the attack.

7. Conclusions and Future Work

This paper presents a security agent platform as the core part of the network intrusion detection system designed to cope with a wide scale of network threats and anomalies. This system addresses two main limitations of existing intrusion detection systems – efficiency and effectiveness. Deployment on high-speed links implies the need to process the important quantity of data in near real-time, in order to prevent the spread of novel threats. Therefore, the individual agents do not acquire the data from the network directly, but receive the preprocessed data, with the level of detail that is appropriate for anomaly-based intrusion detection.

Each detection agent in the system is based on existing anomaly detection technique, which defines its perception of network flow identities in its trust model. Its private view of the data is complemented by the opinions of other agents regarding the anomaly of flows in the current traffic, therefore collaboratively improving the effectiveness of anomaly detection process. When the agent platform reaches a conclusion regarding the untrustfulness of a particular subset of flows, it submits this observation to user-interface agent that automatically retrieves context information (DNS records, history, etc.) to allow rapid analysis by human supervisors.

Our future work is focused on performing more detailed experiments to test the agent platform, improving effectiveness (in the mean of false positives/false negatives), efficiency and performance of the whole system and incorporating better adaptation properties to cope with the dynamics of network traffic.

References

[1] CESNET, z. s. p. o. Family of COMBO Cards. http://www.liberouter.org/hardware.php, 2007.

[2] L. Ertoz, E. Eilertson, A. Lazarevic, P.-N. Tan, V. Kumar, J. Srivastava, and P. Dokas. MINDS - Minnesota Intrusion Detection System. In *Next Generation Data Mining*. MIT Press, 2004.

[3] A. Lakhina, M. Crovella, and C. Diot. Characterization of Network-Wide Anomalies in Traffic Flows. In *ACM SIGCOMM conference on Internet measurement IMC '04*, pages 201–206, New York, NY, USA, 2004. ACM Press.

[4] A. Lakhina, M. Crovella, and C. Diot. Diagnosis Network-Wide Traffic Anomalies. In *ACM SIGCOMM '04*, pages 219–230, New York, NY, USA, 2004. ACM Press.

[5] A. Lakhina, M. Crovella, and C. Diot. Mining Anomalies using Traffic Feature Distributions. In *ACM SIGCOMM, Philadelphia, PA, August 2005*, pages 217–228, New York, NY, USA, 2005. ACM Press.

[6] G. Lyon. Nmap. `http://insecure.org/nmap/`.

[7] M. Rehak, , M. Pechoucek, K. Bartos, martin Grill, and P. Celeda. Network intrusion detection by means of community of trusting agents. In *IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2007 Main Conference Proceedings) (IAT'07)*, Los Alamitos, CA, USA, 2007. IEEE Computer Society.

[8] M. Rehak, M. Gregor, M. Pechoucek, and J. M. Bradshaw. Representing context for multiagent trust modeling. In *IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2006 Main Conference Proceedings) (IAT'06)*, pages 737–746, Los Alamitos, CA, USA, 2006. IEEE Computer Society.

[9] M. Rehák, Lukáš Foltýn, M. Pěchouček, and P. Benda. Trust Model for Open Ubiquitous Agent Systems. In *Intelligent Agent Technology, 2005 IEEE/WIC/ACM International Conference*, number PR2416 in IEEE, 2005.

[10] M. Rehak and M. Pechoucek. Trust modeling with context representation and generalized identities. In *Cooperative Information Agents XI*, number 4676 in LNAI/LNCS. Springer-Verlag, 2007.

[11] J. Sabater and C. Sierra. Review on computational trust and reputation models. *Artif. Intell. Rev.*, 24(1):33–60, 2005.

[12] D. Šišlák, M. Rehák, M. Pěchouček, M. Rollo, and D. Pavlíček. A-globe: Agent development platform with inaccessibility and mobility support. In *Software Agent-Based Applications, Platforms and Development Kits*, pages 21–46, Berlin, 2005. Birkhauser Verlag.

[13] K. Xu, Z.-L. Zhang, and S. Bhattacharyya. Reducing Unwanted Traffic in a Backbone Network. In *USENIX Workshop on Steps to Reduce Unwanted Traffic in the Internet (SRUTI)*, Boston, MA, July 2005.