Proceedings of the 4th WSEAS Int. Conf. on Information Security, Communications and Computers, Tenerife, Spain, December 16-18, 2005 (pp63-68)

Analysis and Presentation of Distributed Sensor Data for Network Intrusion Detection

JOHN C. McEACHEN, CHENG KAH WAI, and VONDA B. OLSAVSKY
Department of Electrical and Computer Engineering
Naval Postgraduate School
Monterey, California
USA

Abstract: - Distributed network intrusion detection systems which incorporate tens, hundreds, even thousands, of sensors are becoming increasing popular. Managing and presenting the information from these sensors is becoming an increasingly difficult task. The paper explores the use of Conversation Exchange Dynamics (CED) to integrate and display sensor information from multiple nodes. We present an experimental setup consisting of multiple sensors reporting individual findings to a central server for aggregated analysis. Different scenarios of network attacks and intrusions were planned to investigate the effectiveness of the distributed system. The network attacks were taken from the M.I.T Lincoln Lab 1999 Data Sets. The distributed system was subjected to different combinations of network attacks in various parts of the network. The results were then analyzed to understand the behavior of the distributed system in response to the different attacks. In general, the distributed system detected all attacks under each scenario. Some surprising observations also indicated attack responses occurring in unanticipated scenarios.

Key-Words: - Distributed Network Intrusion Detection, Conversation Exchange Dynamics

1 Introduction

Intrusion Detection Systems (IDS) have gained increasing importance in ensuring the overall security of organizations. They act as an additional layer of security to the organization's perimeter defense, which usually, is implemented using firewalls. Firewalls are effective in preventing unauthorized entry into the organization's network. However, firewalls cannot detect unauthorized behavior that is present in network traffic they allow to go through.

The role of detecting anomalous behavior is performed by IDS, which try to identify and report attacks and security incidents [1]. There are two categories of IDS: network-based IDS and host-based IDS [2]. Network-based IDS monitor and analyze network traffic in the network segments where they are installed. Host-based IDS monitor and analyze network traffic that goes in and out of specific hosts.

Further, network-based IDS can be further specified based on their structure. Centralized IDS operate standalone, with centralized applications physically integrated within a box, while distributed IDS consist of multiple IDS over a large network, all of which communicate with each other.

The biggest shortcoming in centralized, standalone IDS is that they are built on a single physical entity, which is responsible for both collecting and analyzing data. This can impose severe limitations on efficiency and the system resources, especially when a high volume of data needs to be processed. Distributed IDS (DIDS) can overcome this shortcoming by performing distributed data collection and possibly preprocessing, depending on the design of the system.

DIDS consist of multiple sensors deployed in different areas of a large network, all of which report to a central server that aggregates the information and processes it. The sensors should ideally be deployed on separate network segments and geographical locations [3].

Several established efforts in DIDS are on-going. Oft cited efforts include [4], [5] and [6]. These efforts tend to focus on the collection and distribution architecture of the DIDS. Unfortunately, how this information is collated, managed and presented is not addressed.

The area of interest in this research is implementing, analyzing and presenting sensor information from a distributed IDS using conversation exchange dynamics (CED) [7]. This is achieved by distributing the sensors on different network segments to monitor the traffic in different

parts of the network. The sensors relay the information to a core component, which converts it into a pre-defined format before analyzing it and displaying it on the GUI.

This research is accomplished by analyzing the response of the distributed IDS to network attacks under a variety of conditions. In particular, the Smurf, Mailbomb and Apache2 attacks extracted from the MIT Lincoln Lab IDS datasets will be used to generate a distinctive response [8].

2 Conversation Exchange Dynamics

The underlying concept in [7] is based on the conversation exchange model, which is used to model network traffic. It defines a conversation exchange as an exchange of information between two conversation groups. These conversation groups may represent network nodes, protocols or the tasks which network nodes perform (e.g. client or server). This model uses buckets to represent conversation groups and balls to represent the information that is exchanged between the conversation groups.

Network traffic analysis is based on decision trees that have buckets as leaf nodes. At the beginning of the analysis, each bucket starts off with an initial number of balls. These balls are dynamically moved around in accordance with conversation exchanges that are modeled on information extracted from the network traffic. For instance, the buckets in Figure 1 represent four network nodes. A conversation exchange of n network packets between nodes A and B will result in the movement of n balls from bucket BA to bucket BB. However, the number of balls in each bucket cannot decreased below a minimum level or increased beyond a maximum level, as pre-defined in the decision tree.

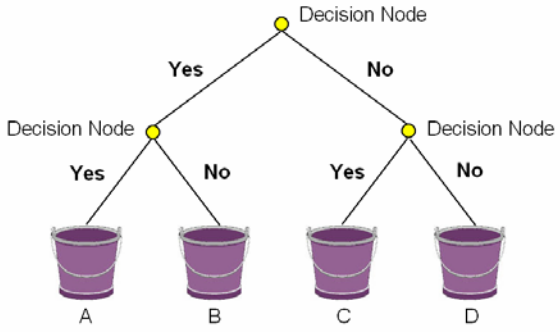

Figure 1. A decision tree with four defined buckets.

During the network traffic analysis, the number of balls in each bucket is constantly varying. A network state is the combination of the number of balls in each bucket at any given time. The state space covers the entire range of possible number of states, *N*, which is determined by the following binomial:

$$N = \begin{pmatrix} M + K - 1 \\ M - 1 \end{pmatrix} \quad (1)$$

where *M* is the number of buckets, and *K* is the total number of balls in the system. A state space walk traces all the states that were visited during a given period of time.

The average number of balls in the buckets can be represented in real time on a 3-D graphical display, known as a thermal tower. Information about the network states visited and the number of occurrences can also be accumulated and plotted on a 3-D graphical display, known as a thermal canyon. When there are unusually high counts of certain states, or when there are a large number of states that are usually not visited, it can be an indication of anomalous network activity. Thermodynamic principles of energy, entropy and temperature can be applied to the thermal canyon, which reveal more information about the network health.

Actual implementations using CED often run multiple instantiations of decision trees simultaneous on the same collected data. A decision tree instantiation most often is projection of a certain network policy such as allowing traffic with a certain service to be exchanged with a certain host. Consequently, a typical implementation might include decision trees specifying e-mail, WWW, and FTP instances. Figure 2 is an example of a decision tree instance for e-mail. The aggregating buckets are labelled numerically from left to right, 0 to 7. A detailed description of the different bucket representations is provided in table 1.

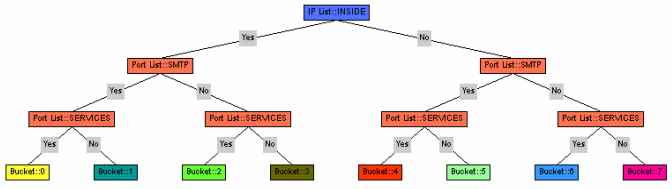

Figure 2. E-mail decision tree instantiation.

Decision tree instances are typically multi-tiered with eight buckets each. In practice, each bucket is initialized with 5 balls, and can have a minimum of 0 balls and a maximum of 10 balls.

Bucket No.	Classification
0	N.A.
1	Insider IP address with TCP ports no. 25, 110, 113 or 161
2	Insider IP address with TCP port no. lower than 1024, excluding 25, 110, 113 and 161.
3	Insider IP address that does not have TCP port no. lower than 1024.
4	N.A.
5	Outsider IP address with TCP ports no. 25, 110, 113 or 161.
6	Outsider IP address with TCP port no. lower than 1024, excluding 25, 110, 113 and 161.
7	Outsider IP address that does not have TCP port no. lower than 1024.

Table 1. Denotation of buckets for the e-mail decision tree instance. Buckets 0 and 4 are not applicable (N.A.) because they represent classifications that cannot occur (i.e. – a port number 25, 110, 113 or 161 that is at the same time not 25, 110, 113, or 161).

3 Experimental Configuration

Figure 3 shows the network topology of the experiment setup. The network attack replay workstation has two network interface cards, which separately send out pre-recorded network traffic containing both normal user traffic and simulated network attacks. The hubs receive and broadcast the network packets, which are then picked up by sensors A and B.

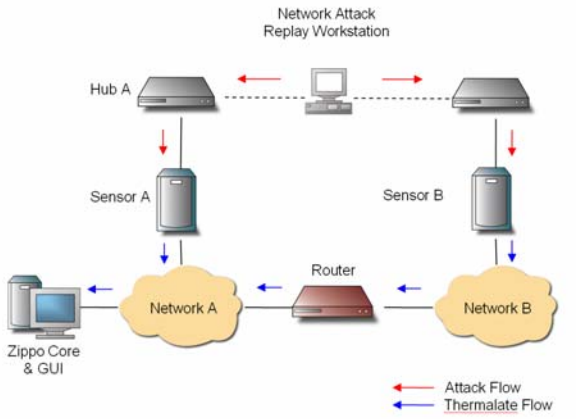

Figure 3. Network topology of experimental setup.

Sensor A and the CED core component reside on Network A, while sensor B resides on Network B. The two sensors sniff every network packet on their respective network segments and produce sensor-related data – thermalate – that is sent to the CED core. Upon processing and analysis, the core generates thermal canyons that reflect the state of the networks.

In this paper, two general scenarios are tested:

i. Sensors A and B detect the same network attack concurrently.
ii. Sensors A and B detect different network attacks concurrently.

Further, for each scenario we will examine the results for attacks against a service from the perspective of the service's decision tree instantiation and from an orthogonal service's decision tree. These experiments are explained in more detail in the next section.

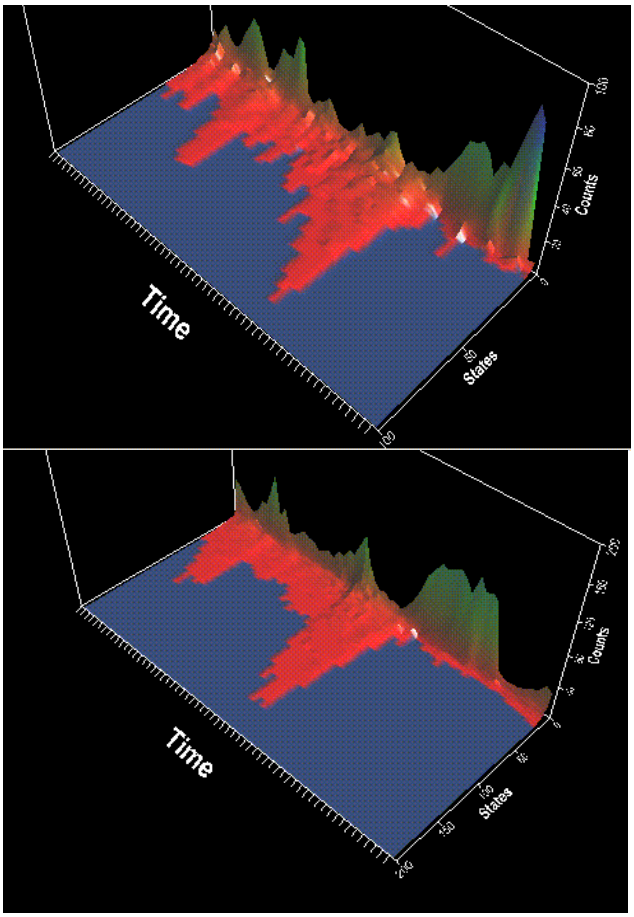

Figure 4. Thermal canyon displays for a single sensor (top) and dual sensors (bottom) during a Mailbomb attack.

4 Detecting a Global Attack

The lower portion of figure 4 shows the results of the aggregated analysis when both sensors A and B detect the Mailbomb attack at the same time. This is achieved by having the network attack reply workstation replay two instances of 42155148.tcpdump at the same time. The CED core

receives two sets of thermalate with approximately the same information.

The top of figure 4 illustrates the thermal canyon of the attack against only a single sensor. Comparing the top of figure 4 with the bottom, it can be observed the shapes of both canyons are very similar. As might be expected, the difference between the two canyons is that the number of bucket states and the counts for each bucket state have doubled in the bottom of figure 4. For instance, the highest peak on the canyon floor in the top of figure 4 indicates 80 visited states. The corresponding peak in the bottom of figure 4 is double that of the top, at approximately 160 states. This behavior was confirmed for all attacks tested.

5 Detecting Simultaneous Local Attacks

In this section, the network attack replay workstation replays the Smurf, Mailbomb and Apache2 attacks on different network segments. The e-mail and WWW decision tree instances are activated on the CED core to perform an aggregated analysis on the thermalate sent from sensors A and B.

5.1 The E-mail Instance

Figure 5 shows the different thermal canyon displays for the e-mail decision tree instance when the Mailbomb attack, the Smurf attack and the combined Mailbomb and Smurf attacks are launched respectively and viewed using the e-mail decision tree instantiation..

We specifically observe that the Smurf attack in the middle of figure 5 shows an orthogonal response to the Mailbomb response in the top of figure 5. Thus in the case of the Smurf attack, even though we do not have an attack targeting the specific service of the decision tree instance, some attack if still observable. Looking closely at the middle of figure 5 during the Smurf attack, the large number of ICMP reply packets from the attackers to the victim result in ball transfers from Bucket 7 to Bucket 3. There are few ball exchanges between the other buckets, which is why there are few bucket states on the thermal canyon.

5.2 The WWW Instance

Figures 6 shows the different thermal canyons of the WWW decision tree instance, corresponding to the detection of Apache2 attack, Smurf attack and a combination of the two attacks. The bucket definitions for the WWW decision tree instance are listed in table 2.

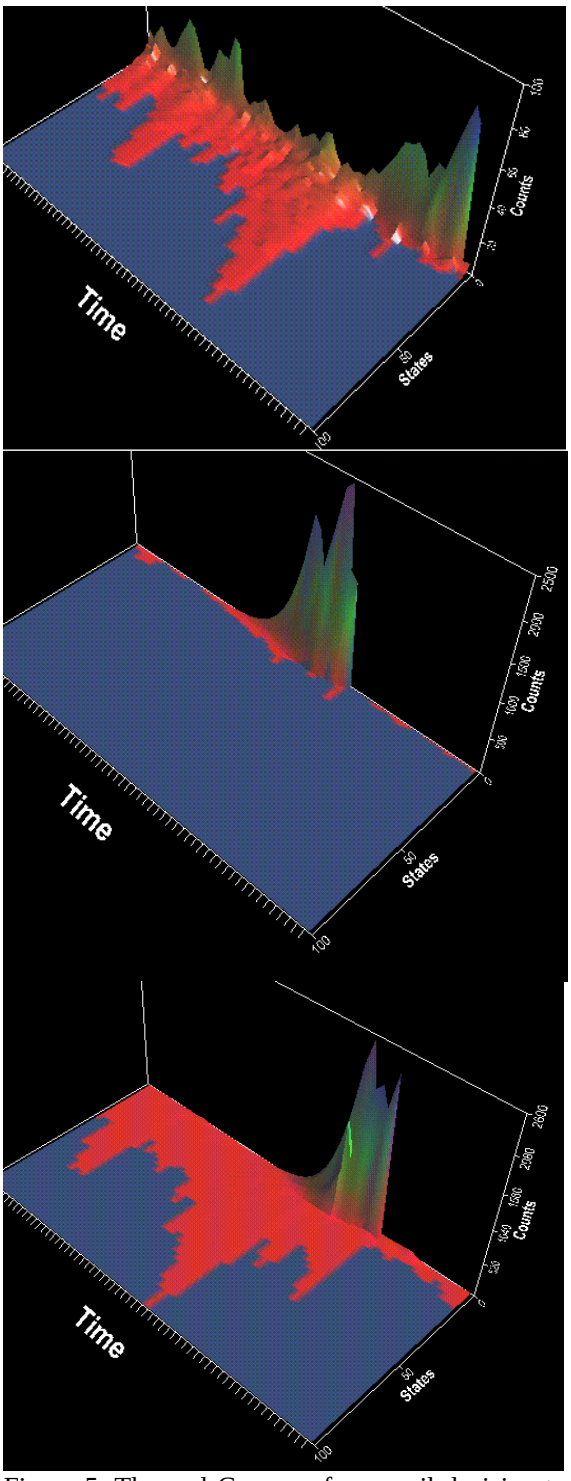

Figure 5. Thermal Canyons for e-mail decision tree instance during a Mailbomb attack (top); Smurf attack (middle) and combined Mailbomb and Smurf attacks (bottom).

Again we see an orthogonal response for an attack that is not related to the specific decision tree service instance. In this case, the Smurf attack results in high peaks on the thermal canyon, as shown in the middle of figure 6. The high counts of the states visited are due to massive ball transfers from bucket 7 to 3.

Bucket No.	Classification
0	N.A
1	Insider IP address with TCP port no. lower than 1024, excluding 80 and 443.
2	Insider IP address with TCP port no. 80 or 443.
3	Insider IP address that does not TCP port no. lower than 1024.
4	N.A
5	Outsider IP address with TCP port no. lower than 1024, excluding 80 and 443.
6	Outsider IP address with TCP port no. 80 or 443.
7	Outsider IP address that does not TCP port no. lower than 1024.

Table 2. Denotation of Buckets for HTTP PID Instance. Similar to table 1, buckets 0 and 4 represent classifications that cannot occur.

5.3 The ICMP Instance

Figure 7 compares the thermal canyon displays of the ICMP decision tree instance to a Smurf attack, an Apache2 attack and combined Smurf and Apache2 attacks respectively. The ICMP decision tree bucket definitions are provide in table 3.

Bucket No.	Classification
0	Insider IP address with ICMP type 3, 4, 5, 11 or 12.
1	Insider IP address with ICMP type 8 or 17.
2	Insider IP address with ICMP type 0 or 18.
3	Outsider IP address with ICMP type 3, 4, 5, 11 or 12.
4	Outsider IP address with ICMP type 8 or 17.
5	Outsider IP address with ICMP type 0 or 18.
6	Insider IP address that does not have ICMP type 0, 3, 4, 5, 8, 11, 12, 17 or 18.
7	Outsider IP address that does not have ICMP type 0, 3, 4, 5, 8, 11, 12, 17 or 18.

Table 3. Denotation of Buckets for ICMP decision tree instance.

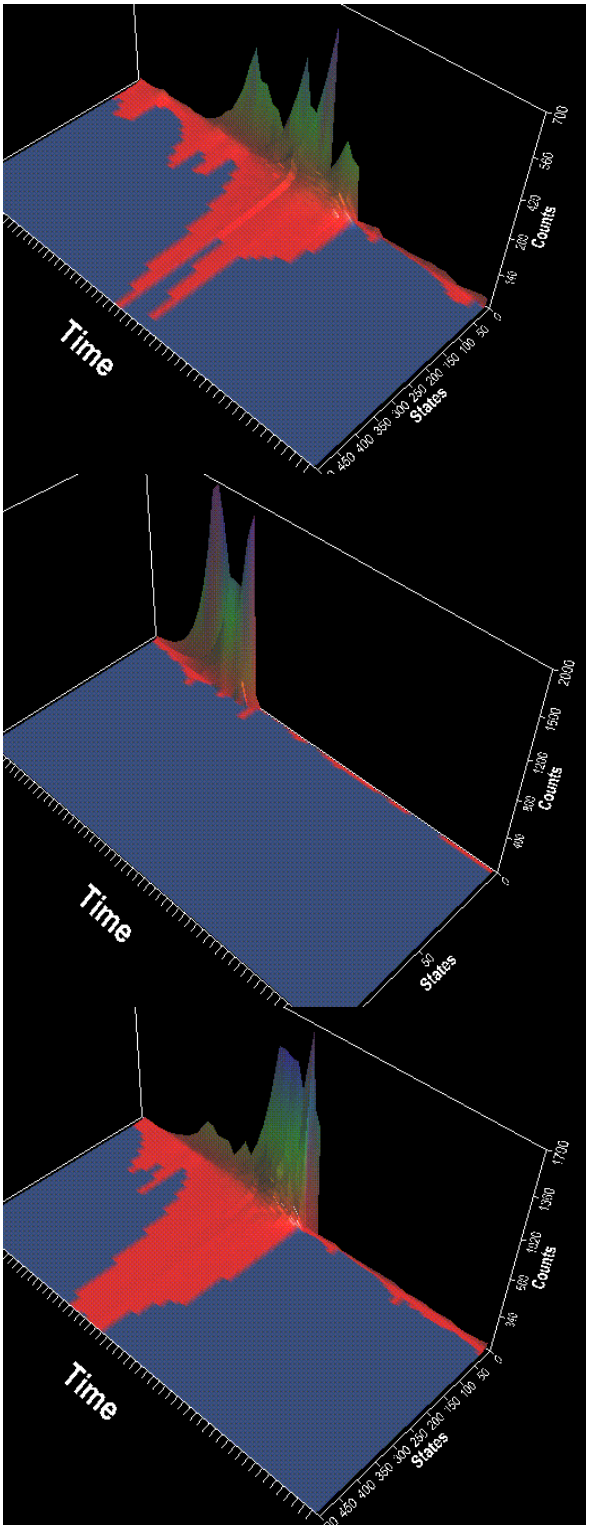

Figure 6. Thermal Canyons for WWW decision tree instance during a Apache2 attack (top); Smurf attack (middle) and combined Apache2 and Smurf attacks (bottom).

The Apache2 attack results in ball transfers between buckets 6 and 7, as shown in middle of

figure 7. The peaks on the thermal canyon clearly indicate an anomalous situation.

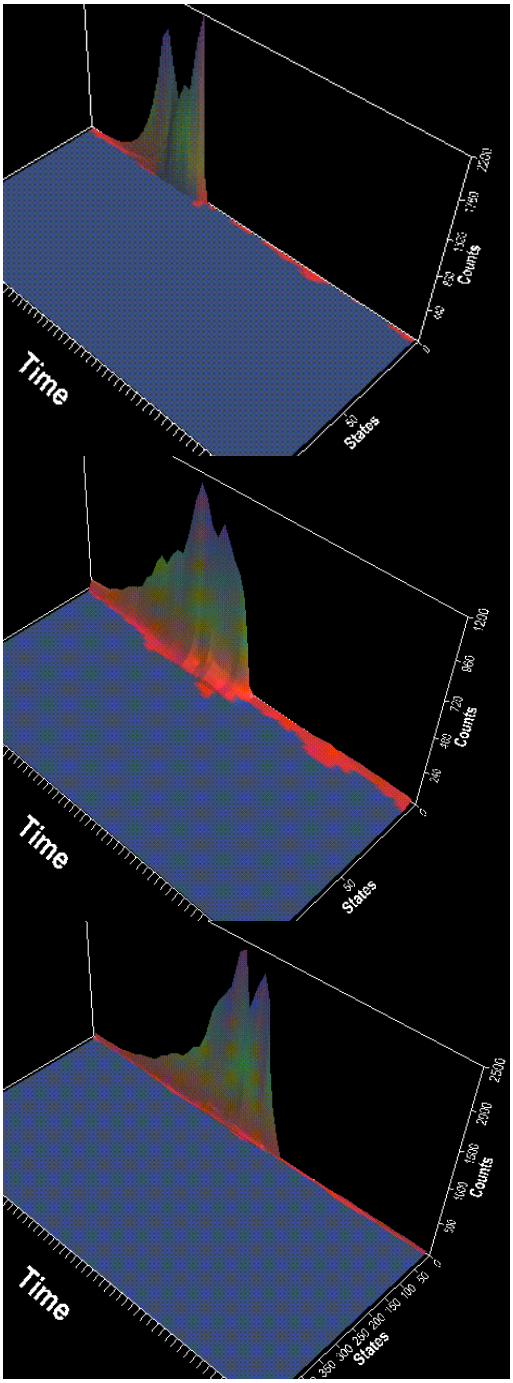

Figure 7. Thermal Canyons for ICMP decision tree instance during a Smurf attack (top); Apache2 attack (middle) and combined Apache2 and Smurf attacks (bottom).

During the combined Smurf and Apache2 attacks, the peaks on the thermal canyon are much higher, due to the large combined volume of traffic. Balls are transferred mainly between Buckets 5 and 2, and Buckets 6 and 7. It is difficult to tell from the thermal canyon alone that there are two attacks going on, unless the details about the network packets are obtained from the thermalate.

4 Conclusion

We have presented a novel approach to integrating and presenting aggregated IDS sensor data. We presented several scenarios of attacks directly globally against the network or against particular segments, even simultaneously occurring with attacks on different segments. In each scenario, so form of response was observed, even when the attack was not directed at the particular service of interest. This last point is significant because in intrusion detection, it is more important to have some response that may not exactly match the service under question than to have no response at all. Further, this approach allows us to identify attacks for which no decision tree has been specifically instantiated.

References:

[1] D. E. Denning, "An Intrusion-Detection Model," *Proc. Of IEEE Symposium on Security and Privacy*, pp. 118 - 131, 1986.

[2] S. Northcutt and J. Novak, *Network Intrusion Detection*, 3rd Edition, SANS, August 2002.

[3] S. Axelsson , "Research in Intrusion Detection Systems: A Survey," Technical Report, 1999.

[4] P. Porras and P. Neumann, "EMERALD: Event monitoring enabling responses to anomalous live disturbances," *Proceedings of the 20th National Information Systems Security Conference*, Baltimore, MD, October, 1997, pp. 353 – 365.

[5] C. Kruegel, T. Toth, and E. Kirda, "Sparta – A Mobile Agent based Intrusion Detection System." Technical Report, TUV-1841-2002-24, Technical University of Vienna, April 2002.

[6] C. Manikopoulos and S. Papavassiliou, "Network Intrusion and Fault Detection: A Statistical Anomaly Approach," *IEEE Network*, October 2002, pp. 76 – 82.

[7] J. McEachen and J. Zachary, "Accentuating Anomalies in Computer Network Conversations for Enhanced Security," *WSEAS Trans. on Info. Science and Apps.*, 2:10, October 2005, pp. 1551 – 1561.

[8] Massachusetts Institute of Technology, Lincoln Laboratory, DARPA Intrusion Detection Evaluation, http://www.ll.mit.edu/IST/ideval/, last accessed October 1, 2005.