Adaptation Techniques for Intrusion Detection and Intrusion Response Systems

Daniel J. Ragsdale
Information Technology and Operations Center, United States Military Academy

Curtis A. Carver, Jr., Jeffrey W. Humphries, and Udo W. Pooch
Department of Computer Science, Texas A&M University

Abstract

This paper examines techniques for providing adaptation in intrusion detection and intrusion response systems. As attacks on computer systems are becoming increasingly numerous and sophisticated, there is a growing need for intrusion detection and response systems to dynamically adapt to better detect and respond to attacks. The Adaptive Hierarchical Agent-based Intrusion Detection System (AHA! IDS) provides detection adaptation by adjusting the amount of system resources devoted to the task of detecting intrusive activities. This is accomplished by dynamically invoking new combinations of lower level detection agents in response to changing circumstances and by adjusting the confidence associated with these lower-level agents. The Adaptive Agent-based Intrusion Response System (AAIRS) provides response adaptation by weighting those responses that have been successful in the past over those techniques that have not been as successful. As a result, the more successful responses are used more often than the less successful techniques. It also adapts responses based on the system's belief that intrusion detection reports are valid. Intuitively, adaptive detection and response systems will provide more robust protection than static, non-adaptive systems.

1 Introduction

The number of information warfare attacks is increasing, and they are becoming increasingly sophisticated. Annual reports from the Computer Emergency Response Team (CERT) indicate a significant increase in the number of computer security incidents each year. Figure 1 depicts the rise in computer security incidents: six incidents were reported in the 1988 and over 8,200 were reported in 1999 [1]. Not only are these attacks becoming more numerous, they are also becoming more sophisticated. The 1998 CERT Annual Report reports the growing use of "widespread attacks using scripted tools to control a collection of information-gathering and exploitation tools" [2]. The 1999 CERT Distributed Denial of Service Workshop likewise reports the growing use of automated scripts that launch and control tens of thousands of attacks against one or more targets. Each attacked computer has limited information on who is initiating the attack and from where these attacks originate [3]. The threat of sophisticated computer attacks is growing. Unfortunately, intrusion detection and response systems have not kept up with the increasing threat.

2 Previous Work

2.1 Intrusion Detection

Over the past twenty years, with particular emphasis during the last five, a great deal of research has been devoted to support the design and construction of effective Intrusion Detection Systems (IDS) and Intrusion Response Systems (IRS). This research has done much to advance the state-of-the-art in this increasingly important area. Research prototypes and commercial IDS and IRS built during this period now number nearly 100 [4]. Notwithstanding all of this effort and, despite the obvious benefits of adaptation, only limited research has been devoted to support the adaptation of detection and response capabilities in these systems.

The research systems that do support some form of adaptation focus primarily on the issue of *learning or discovering* patterns or states that are indicative of intrusive behavior [4-10]. Some system, for example the Security Adaptation Manager (SAM), do provide a higher degree of adaptation. SAM adapts its *protection posture* based upon the current state of the system [11, 12]. Specifically, if SAM determines that the system(s) it protected are in a *calm* state it uses a configuration that minimizes its resource usage. When the system is under attack (the *panic* state), a more aggressive detection posture is maintained. The research conducted to support the design of SAM, as well as the autonomous agent research performed at the Purdue Center for Research and Education in Information Assurance and Security (CERIAS) [13, 14] were the primary sources of inspiration for the research presented in this paper.

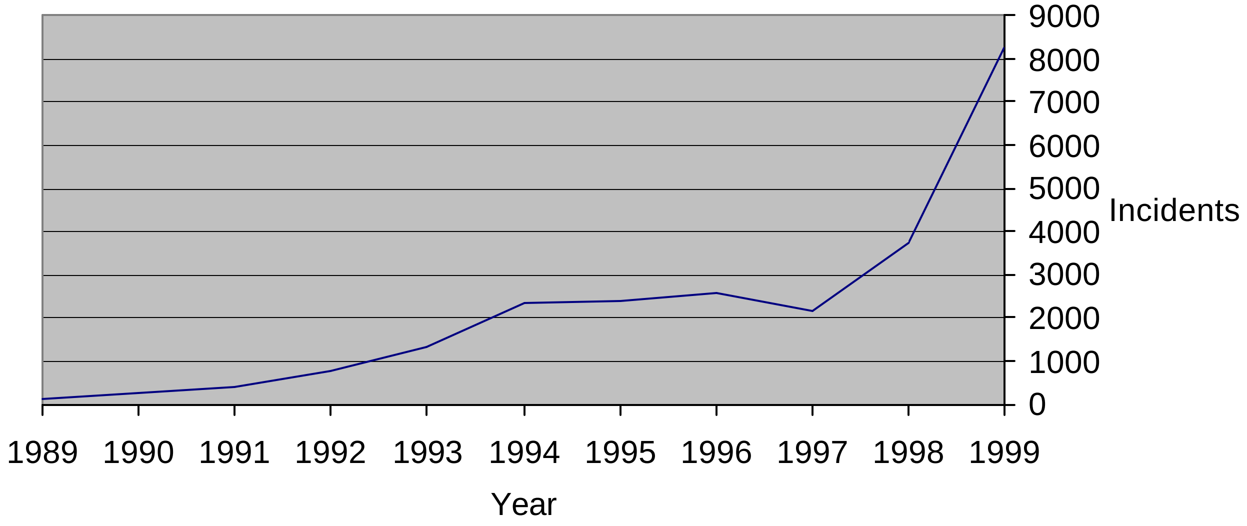

Figure 1: CERT Reported Incidents per Year

Another system, the Intrusion Detection Inter-component Adaptive Negotiation (IDIAN) project supports adaptation by providing a protocol to allow ID components to dynamically negotiate and modify agreements with other ID components [15]. Yet another set of researchers have implemented a prototype system, the Distributed Assessment and REsponse System (DARE), that supports dynamic changes, based upon the currently perceived threat to the "defense posture of individual systems and of an overall network [16]."

Finally, at least one commercial security system vendor, Internet Security Systems Inc. (ISS), claims to be "the leading provider of adaptive network security solutions [17]." The ISS SAFESuite™ product does provides a wide array of security services and it can adapt these services based on frequently conducted risk assessments. However, the ISS intrusion detection product, RealSecure™, while capable of detecting a wide array of intrusive activities, appears to support only limited adaptation of its detection capabilities during operation.

2.1 Intrusion Response

As with intrusion detection systems, over the past twenty years, a number of systems that provide for a response to intrusive actions have been developed. These response systems can be categorized as notification systems, manual response systems, or automatic response systems (see Table 1). The majority of intrusion detection and response systems are notification systems - systems that generate reports and alarms only. Some systems provide the additional capability for the system administrator to initiate a manual response from a limited preprogrammed set of responses. While this capability is more useful than notification only, there is still a time gap between when the intrusion is detected and when a response is initiated. Automatic response systems immediately respond to an intrusion through pre-programmed responses. With two exceptions, all of these automatic response systems use a simple decision table where a particular response is associated with a particular attack. If an attack occurs, a preprogrammed response executes. These preprogrammed responses consist predominantly of the execution of a single command or action instead of the invocation of a series of actions in order to limit the effectiveness of the attacker.

The two exceptions are the Cooperating Security Managers (CSM) and Event Monitoring Enabling Responses to Anomalous Live Disturbances (EMERALD). These systems provide some degree of real-time response adaptation. Both systems use an expert system to determine an appropriate response, which changes as the degree of

Intrusion Response Classification	# of Systems
Notification	31
Manual Response	8
Automatic Response	17
Total	56

Table 1: Classification of Intrusion Response Systems (as of March 2000)

suspicion about an ongoing attack changes. Different responses are associated with different suspicion levels and the system adapts its responses based on the degree of suspicion. The EMERALD system also uses a severity metric to determine the appropriate response [18-21].

Existing intrusion response systems do not adapt responses to computer attacks based on the degree of belief in the validity of intrusion reports generated by intrusion detection systems. In addition, these systems do not adapt based on the actual success rate of the various responses that have been previously used. This paper addresses both of these open research issues.

3 Adaptation in Intrusion Detection

3.1 Methodology Overview

The Adaptive Hierarchical Agent-based Intrusion Detection System (AHA! IDS) employs a fully distributed, multi-agent framework. The major components in this framework are *Director* agents, *Manager* agents, *Tool* agents, and *Surrogate* agents. In this framework, which is based loosely on the agent architecture developed at CERIAS [13], Director agents are responsible for detecting intrusive action on a collection of systems and network segments. In a large-scale network, multiple levels of Directors can exist. At each level in this hierarchical arrangement, Directors are responsible for a subset of the systems for which its immediate supervisor is responsible. Associated with each Director agent is a Surrogate agent. This surrogate resides on another part of the network, and its role is to assume the Director's responsibilities in the event that the Director fails. The use of surrogates partially mitigates the inherent shortcomings of hierarchical arrangements (i.e. the existence of single points of failure), while retaining much of the inherent advantages of hierarchical arrangements [22]. The lowest–level Director agents supervise a set of Manager agents, each of which is responsible for detecting intrusive activities on possibly overlapping subsets of the systems for which the Director is responsible.

Manager agents detect intrusive activity on the system for which they are responsible by employing a dynamically changeable set of Tool agents. Tool agents are low-level, lightweight, but fully functional, intrusion detection systems. The determination, by the Manager agents, of the most suitable number and type of Tool agents to employ, at any given time, is one of the most important responsibilities of Manager agents.

The low-level Tool agents encapsulate the functionality of many varying IDSs. Consequently they employ a wide variety of methods to analyze data from a wide array of sensors. Each individual Tool agent focuses on either specific systems (including hosts, routers, switches, etc.) or on one or more network segments. Tool agents, like the other agents in this framework, have a high degree of autonomy and can function independently of any of the other agents in the system. All agents in this framework communicate and collaborate with peer agents, using a subset of the agent communication language and protocol, Knowledge Query and Manipulation Language (KQML) [23, 24]. Agents also communicate with their supervisor and their supervisor's surrogate. The communication between agents is enabled through the use of a specialized intrusion detection ontology, which is based, in part, on the Common Intrusion Specification Language (CISL) [25].

In the event that a Tool agent detects (or suspects) that intrusive activities are occurring, it prepares and sends a KQML *Tell performative* message to its peers, its supervisor, and its supervisor's surrogate. These messages include *degree of confidence,* which allows Tool agents to render reports even in the presence of uncertainty.

Supervisor agents, at all levels, maintain and continuously update the *threat level* that exists on the systems for which they are responsible. To determine this level, agents take into account the information from their subordinate agents, and their peers, as well as, from their supervisors. If necessary, Supervisor agents prepare and send KQML *Ask performative* messages to solicit the information necessary to make this determination.

4.2 Adaptation Support

The proposed AHA! IDS framework provides detection adaptation in three specific areas. First, it supports adaptation by adjusting the amount of system resources devoted to the task of detecting intrusive activities. There will always be a tradeoff between the amounts of system resources devoted to perform useful work (functionality) vs. that which is devoted to securing the system. For example, there are periods when, due to a perceived low degree of threat, that only a small proportion of system resources should be devoted to detecting intrusive activities. On the other hand, during period of perceived high threat, significantly more resources should be devoted to this task. There is direct support within the AHA! IDS framework for the reasoning process necessary to determine the appropriate level of system resources for accomplishing the intrusion detection task.

Second, the AHA! IDS adapts by dynamically invoking new combinations of low-level detection agents in response to changing circumstances. As the conditions in a given network change, resulting in increased or decreased resources levels for intrusion detection, so does the need for varying types of low-level intrusion tools. For instance, if

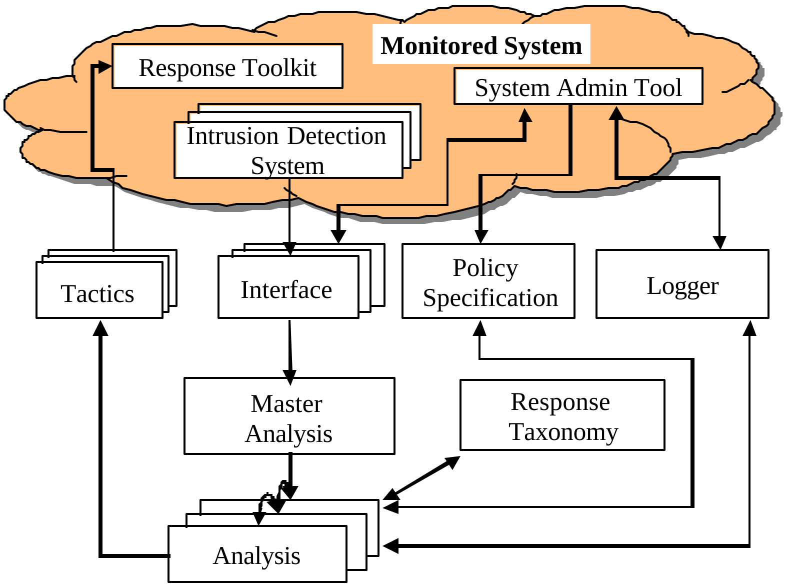

Figure 2: AAIRS Methodology

the rates of a specific type of attack were expected to increase, then it would clearly be sensible for IDS to use tools that are designed to detect that type of attack. The AHA! IDS framework provides explicit support for this form of adaptation.

Finally, AHA! IDS adapts by adjusting the *confidence* metric that it associates with the low-level detection agents. All IDS can generate both false positives and false negatives. Unfortunately, these rates are not always known a priori. One way to improve the overall IDS which uses these tools would be to keep track of the performance of the low level tools and maintain a confidence metric. The AHA! IDS framework incorporates this form of adaptation.

4 Adaptation in Intrusion Response

4.1 Methodology Overview

The Adaptive Agent-based Intrusion Response System (AAIRS) methodology is summarized in Figure 2. Multiple IDSs monitor a computer system and generate intrusion alarms. *Interface agents* maintain a model of each IDS based on number of false positives/negatives previously generated. It uses this model to generate an attack confidence metric and passes this metric along with the intrusion report to the *Master Analysis agent*. The Master Analysis agent classifies whether the incident is a continuation of an existing incident or is a new attack. If it is a new attack, the Master Analysis agent creates a new *Analysis agent* to develop a response plan to the new attack. If the incident is a continuation of an existing attack, the Master Analysis agent passes the attack confidence metric and intrusion report to the existing Analysis agent handing the attack. The Analysis agent analyzes an incident until it is resolved and generates an abstract course of action to resolve the incident [26]. To generate this course of action, the Analysis agent invokes the *Response Taxonomy agent* to classify the attack and *Policy Specification agent* to determine a response goal and to limit the response based on legal, ethical, institutional, or resource constraints [27]. The Analysis agent passes the selected course of action to the *Tactics agent*. The Tactics agent decomposes the abstract course of action into very specific actions and then invokes the appropriate components of the *Response Toolkit*. Both the Analysis and Tactics agents employ adaptive decision-making based on the success of previous responses. The *Logger* records Analysis and Tactics agents' decisions for system administrator review [26].

4.2 Adaptation Support

The proposed methodology provides response adaptation through three components: the Interface, Analysis, and Tactics agents. The Interface agent adapts by modifying the confidence metric associated with each IDS. IDSs are not perfect and will generate false positive and false negative alarms. The response must be tailored by the degree to which the response system believes that the reported incident is a real attack and not a false alarm. The confidence metric is the ratio of false positive reports to actual reports. The number of false positives is generated through a feedback loop between the interface agent and the system administration tool. After each incident, the system administrator can indicate whether the incident was a real attack or a false alarm. This results in an update to the confidence metric for the reporting IDS and over time, response adaptation. Responses to incidents from IDSs that generate a high number of false positives will be less severe than reports from IDSs that seldom generate false alarms.

The Analysis and Tactics components also provide response adaptation. As the Analysis components receive additional incident reports, these reports may lead to reclassification of the type of attacker and/or type of attack. This reclassification may lead to the formulation of a new plan or a change in how the response goal was being accomplished. The Analysis component may change the plan steps being used to accomplish the goal if alternative steps are available and can be substituted into the plan. Alternatively, the Analysis component can request the Tactics component to perform the adaptation. The Tactics components may have multiple techniques for implementing the plan step. By changing techniques with the same plan, the system can adapt its approach in an attempt to stop the intruder. Finally, both the Analysis and Tactics components maintain success metrics on their plans and actions respectively. Those plans and actions that are successful in resolving intrusions are weighted so that they are used more frequently while those plan steps and techniques that are not as successful are used less often.

5 Summary

Current intrusion detection systems (IDS) and intrusion response systems (IRS) have a limited ability to adapt their detection and response capabilities to these increasingly sophisticated attacks. In this paper we have described two systems, the AHA! IDS and AAIRS. These two prototype systems adapt their capabilities during use to improve their overall detection and response effectiveness. In addition, by having the ability to adapt, these systems are more robust and more resistant to subversion by an attacker.

6 References

[1] CERT Coordination Center, “CERT/CC Statistics for 1988 through 1998,” Available at http://www.cert.org/stats/cert_stats.html, January, 2000.

[2] CERT Coordination Center, “CERT Coordination Center 1998,” Available at http://www.cert.org/annual_rpts/cert_rpt_98.html, January, 2000.

[3] CERT Coordination Center, “Results of the Distributed-Systems Intruder Tools Workshop,” Available at http://www.cert.org/reports/dsit_workshop-final.html, June, 2000.

[4] M. Sobirey, “The Intrusion Detection System AID,” Available at http://www-rnks.informatik.tu-cottbus.de/~sobirey/aid.e.html.

[5] D. Anderson, T. Frivold, and A. Valdes, “Next Generation Intrusion Detection Expert System (NIDES): A Summary,” Tech Report SRI-CSL-95-07, SRI International, Menlo Park, CA, May, 1995.

[6] W. Lee and S. J. Stolfo, “Data Mining Approaches for Intrusion Detection,” in *Proceedings of the Proceedings of the 7th USENIX Security Symposium*, San Antonio, TX, January 26-29, 1998, pp. 79-94.

[7] S. Cheung, R. Crawford, M. Dilger, J. Frank, J. Hoagland, K. N. Levitt, J. Rowe, S. Staniford-Chen, R. Yip, and D. Zerkle, “The Design of GrIDS: A Graph-Based Intrusion Detection System,” CSE-99-2, Department of Computer Science, University of California, Davis, CA, September, 1999.

[8] J. Cannady, “Artificial Neural Networks for Misuse Detection,” in *Proceedings of the 21st National Information Systems Security Conference*, Crystal City, Virginia, October 6-9, 1998, pp. 441-454.

[9] M. Meneganti, F. S. Saviello, and R. Tagliaferri, “Fuzzy Neural Networks for Classification and Detection of Anomalies,” *IEEE Transactions on Neural Networks*, vol. 9, no. 5, 1998, pp. 848-861.

[10] L. Mé, “GASSATA, a Genetic Algorithm as an Alternative Tool for Security Audit Trails Analysis,” in *Proceedings of the First International Workshop on the Recent Advances in Intrusion Detection*, Louvain-la-Neuve, Belgium, September 14-16, 1998, http://www.zurich.ibm.com/~dac/Prog_RAID98/Full_Papers/gassata_paper.ps.

[11] Crispin Cowan, Calton Pu, Dave Maier, Heather Hinton, Peat Bakke, Steve Beattie, Aaron Grier, Perry Wagle, and Q. Zhang, “StackGuard: Automatic Adaptive Detection and Prevention of Buffer-Overflow Attacks,” in *Proceedings of the 7th USENIX Security Conference*, San Antonio, TX, January 1998, http://www.acsac.org/1999/papers/fri-b-1030-hinton.pdf.

[12] Heather Hinton, Crispin Cowan, Lois Delcambre, and S. Bowers, “SAM: Security

Adaptation Manager," in *Proceedings of the Annual Computer Security Applications Conference*, Phoenix, AZ, December, 1999, http://www.cse.ogi.edu/DISC/projects/immunix/sam.ps.gz.

[13] J. Balasubramaniyan, J. O. Garcia-Fernandez, D. Isacoff, E. H. Spafford, and D. M. Zamboni, "An Architecture for Intrusion Detection using Autonomous Agents," Tech Report 98-05, COAST Laboratory, Department of Computer Science, Purdue University, West Lafayette, IN, May 1998.

[14] M. Crosbie and E. H. Spafford, "Active Defense of a Computer System using Autonomous Agents," 95-008, COAST Laboratory, Department of Computer Science, Purdue University, West Lafayette, IN, February, 1995.

[15] R. Feiertag, L. Benzinger, S. Rho, S. Wu, K. Levitt, D. Peticolas, M. Heckman, S. Staniford-Chen, and C. Zhang, "Intrusion Detection Inter-component Adaptive Negotiation," in *Proceedings of the RAID 99: Recent Advances in Intrusion Detection*, West Lafayette, Indiana, USA, September 7-9, 1999, http://seclab.cs.ucdavis.edu/papers/tis99.pdf.

[16] E. D. Fred Cohen, Tim Berg, Cindy Phillips, Vitus Leung, and Stefan Chakerian, "An Automated, Dynamic, Flexible, Distributed, Scalable Network Defense," Available at http://www.all.net/journal/ntb/flex.html, June 29, 2000.

[17] Staff, Internet Security Systems, Inc., "Market Engineering Marketing Strategy Award," Available at http://www.iss.net/cgi-bin/dbt-display.exe/db_data/press_rel/release/062800264.plt, June, 2000.

[18] E. A. Fisch, "Intrusion Damage Control and Assessment: A Taxonomy and Implementation of Automated Responses to Intrusive Behavior," Ph.D. Dissertation, Texas A&M University, College Station, TX, 1996.

[19] G. B. White, E. A. Fisch, and U. W. Pooch, "Cooperating Security Managers: A Peer-based Intrusion Detection System," *IEEE Network*, vol. 10, no. 1, January/February, 1996, pp. 20-23.

[20] P. A. Porras and P. G. Neumann, "EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances," in *Proceedings of the Proceedings of the 20th National Information Systems Security Conference*, Baltimore, MD, October 7-10, 1997, pp. 353-365.

[21] P. G. Neumann and P. A. Porras, "Experience with EMERALD to Date," in *Proceedings of the 1st USENIX Workshop on Intrusion Detection and Network Monitoring*, Santa Clara, CA, April 11-12, 1999, http://www.sage.usenix.org/publications/library/proceedings/detection99/neumann/neumann.pdf.

[22] M. Travers, "Programming with Agents: New Metaphors for Thinking about Computation," Ph.D. Dissertation, Massachusetts Institute of Technology, 1996.

[23] T. Finin, J. Weber, G. Wiederhold, M. Genesereth, R. Fritzson, D. McKay, J. McGuire, R. Pelavin, S. Shapiro, and C. Beck, "Draft Specification of the KQML Agent-Communication Language," Available at http://www.cs.umbc.edu/kqml/kqmlspec.ps, June, 2000.

[24] T. Finin, R. Fritzson, D. McKay, and R. McEntire, "KQML as an Agent Communication Language," in *Proceedings of the Third International Conference on Information and Knowledge Management*, Gaithersburg, Maryland, November 29 - December 2, 1994, pp. 456-463.

[25] R. Feiertag, C. Kahn, P. Porras, D. Schnackenberg, S. Staniford-Chen, and B. Tung, "A Common Intrusion Specification Language (CISL)," Available at http://www.gidos.org/drafts/language.txt, June, 2000.

[26] C. A. Carver, J. M. D. Hill, J. R. Surdu, and U. W. Pooch, "A Methodology for using Intelligent Agents to provide Automated Intrusion Response," in *Proceedings of the IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*, West Point, NY, June 6-7, 2000, 2000, pp. CD-ROM.

[27] C. A. Carver and U. W. Pooch, "An Intrusion Response Taxonomy and its Role in Automatic Intrusion Response," in *Proceedings of the IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*, West Point, NY, June 6-7, 2000, 2000, pp. CD-ROM.