SCALABLE, AUTONOMOUS MONITORING AND RESPONSE FOR NETWORK COMMUNICATIONS RELIABILITY

Lynn J. Gasch
Stottler Henke Associates, Inc., Seattle, Washington
lynng@stottlerhenke.com

ABSTRACT

Ensuring availability of network resources is a challenging task, requiring considerable amounts of human time and expertise. An automated system is needed to monitor the behavior of the network and take preventive and corrective measures to maximize network health. Such a system must meet objectives in scalability, survivability, interoperability, robustness, trustworthiness, and adaptability. Signature-based attack, intrusion and fault detection is impossible, not only due to the fact that an effective system must recognize, respond to and recover from network degradations whose causes have not previously been observed, but also because of the diversity and complexity of computer networks. Stottler Henke Associates has made significant progress in enhancing network availability and security with our Multi-Agent System for network Resource Reliability (MASRR), a decentralized architecture of autonomous, collaborative agents that can model normal network operations, detect departures from expected behaviors, and take remedial actions. A unique adaptive anomaly-detection algorithm supports automatic construction of highly customized baseline models while avoiding brittleness. This, teamed with intelligent agent reasoning capabilities, bring our system's "alarm" accuracy in line with that required of automated response systems. To satisfy military communications needs in tactical environments, agents may additionally be enhanced with goal-oriented planning, enabling the system to protect or find alternatives to the resources most critical for achieving mission objectives.

INTRODUCTION

As computing power has grown, so has our reliance on computers and networks. Communications, transportation, just-in-time inventory and production, energy and water supply – the very foundations of our economic, personal, and political safety all depend on reliable network availability. Our military and security institutions are now operating in a domain described by new terminology such as *information warfare* and *cyber-security.* But with this dependence comes new dangers. Even a static or fixed network, such as a small corporate local area network (LAN), is fraught with vulnerabilities and plagued with outages. Military communications networks introduce additional complexities by joining radio frequency (RF), satellite, and microwave media with the Internet Protocol (IP) transport. Current research is underway to develop new protocols centering on security and dynamic routing. Future communications networks may comprise fixed and *ad hoc* routing mechanisms with both stationary and mobile components. Affecting all computer networks are threats including "crackers", intruders, and masqueraders who continue to find new exploits to deny service, spy and corrupt information, as well as any number of hardware and software faults, failures, misconfigurations and incompatibilities. Couple these hazards with the risks of radio, electromagnetic, and weather interference; line of sight obstruction; and network node destruction or capture, and ensuring successful mission-support communications in tactical military environments becomes an incredibly challenging task.

There are, of course, many existing efforts to monitor network traffic and events and manage networks. A number of tools are aimed at particular security aspects, such as policy management consoles ([1], [2]), virus scanners ([3], [4]), and intrusion detection systems (IDSs) ([5], [6]). Network hardware and software vendors typically provide means for monitoring and managing component health and performance; however, many of these are device- or vendor-specific. Newer management applications are designed to integrate the output of various management and security tools into a unified interface ([7], [8], [9]).

Despite improvements in these areas, relying on a human operator for maintaining network availability is a less than optimal solution. Problems are hard to diagnose: there is no exact correspondence between observable fault, misconfiguration, or attack symptoms and their underlying causes [10]. Problems may be intermittent and difficult to consistently reproduce. Relatively minor faults can persist undetected, exacerbating and masking the causes of larger events that might occur. System operators with an effective level of experience and expertise are uncommon and expensive, requiring ongoing training to keep up with new devices, applications, protocols and threats. Particularly in the domain of military communications, the network expert represents a single point of failure risk, whereas in tactical operations, warfighters must be able to change

0-7803-8847-X/04/$20.00 ©2004 IEEE

roles and complete the mission objectives. The sheer volume of information that must be absorbed and the speed with which response is required indicate the need for an automated network monitoring and response system.

An automated response system must merit trust of its level of accuracy in detecting events and raising alarms. Responses to network incidents like attacks or faults typically involve reconfiguring routes or filtering traffic. These can be costly and come with the possible side effect of denying valid traffic as well as shutting out the attacker. Thus it is critical that an automated response system be trusted to detect and respond to real events but not be falsely triggered into committing "fratricide" [11] against legitimate users. Many of today's IDSs do not have trustworthy accuracy, particularly those that are signature-based, matching suspicious files or activities against previously identified definitions. Signature-based systems frequently have unsatisfactory levels of both false positives, in which an alarm is raised but attributed to valid behavior, and false negatives, in which harmful or unpermitted activity goes undetected. False positives arise, for example, from the similarity of portions of an attack sequence to routine, allowable usage. False negatives can be caused by "brittle" definitions that allow detection to be circumvented by making small changes in the attack sequence or by varying the time window in which the attack is mounted.

Other weaknesses trouble signature-based systems. While they may be fairly successful at averting known attacks with relatively sequences of events, signature-based IDSs are always a step behind the enemy, knowing only the profiles of attacks that have already occurred. They cannot protect the first victims of a new exploit, and other networks remain vulnerable during the time required to update definitions and patches. New exploits are constantly being discovered, and growing signature libraries consume processing and disk resources while overlapping or similar definitions make identification uncertain.

To combat these problems, IDSs are moving away from signature-based systems in favor of anomaly detection [12]. Anomaly detection in general involves developing a model of normal behavior and then determining whether the current observed behavior deviates from that model. Baseline model generation can be supervised (inductively learned from examples labeled as representing either normal or abnormal behavior) or unsupervised (learned from unlabeled examples). Because data example collection and preparation is such a costly endeavor, and because models generated from one set of network data likely cannot be generalized to other networks possessing different characteristics, the systems that will prove to be more accurate and easier to deploy are those that can develop their baseline models in place on the "live" network. There are some such systems available today that are aimed at assisting the network administrator [13], [14] and others that focus on automated response [15], [16]. Most of these systems are geared to security monitoring, looking at traffic patterns and usage, and while they might new attacks and even spot misconfigurations, it is unlikely that they could detect precursors to other outages like those described in [10], [17], and [18].

Military communications networks may be highly dynamic. There will be instances in which a network must be rapidly deployed or reconfigured by personnel who may or may not carry a high degree of network management expertise. Even static or fixed networks may provide critical resources whose immediate availability must be ensured with high reliability. Ideally, we would endow these networks with the ability to effectively manage themselves against outages of all types: routine faults, failures and misconfigurations, in addition to disruption from intrusion or attack. The control system for such a network would need to meet objectives in:

Scalability. The system must perform adequately on networks of any size, consuming acceptable portions of processor cycles, communication bandwidth, data storage, electrical power, and the like.

Survivability. A management goal for any network is to retain the best performance possible when confronted with an attack or failure. In order to achieve this, the control system itself must also remain functional under duress.

Interoperability. Networks comprise diverse components. The control system must be able to monitor and affect the behavior of a variety of network elements.

Robustness. In addition to providing interoperability among today's hardware, protocols, and software, the system must robustly accommodate future elements as well.

Trustworthiness. Automated response without trust is a vulnerability unto itself. An acceptable system can be trusted to make accurate diagnoses – both positive and negative – and to react accordingly.

Adaptability. The system must stay "in tune" with the network as it changes over time. Otherwise, the system either becomes brittle and untrustworthy or it consumes an inordinate amount of administrative tweaking.

In addition, for the military communications domain, a desirable enhancement would be:

Planning and mission support. Network elements provide services of varying importance to the completion of a mission objective. The control system should take this into account in formulating its response, even going so far as to identify alternate services or functions.

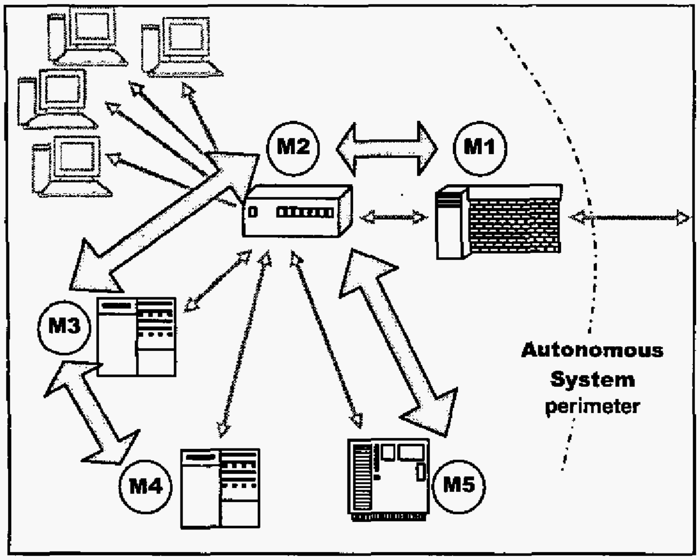

Figure 1. Inter-agent communication.

Stottler Henke Associates has made significant progress in the development of such an automated monitoring and response system. The Multi-Agent System for network Resource Reliability (MASRR) was developed under a multi-year, DARPA-funded contract. The following sections describe MASRR and how it achieves these objectives through its architecture and anomaly detection and agent reasoning components.

DESCRIPTION OF MASRR

To achieve scalability, MASRR is designed around the notion of semi-autonomous intelligent agents deployed throughout a computer network. The tasks of each agent are to monitor network behavior, to steer the network in order to avoid problems, and to take corrective action when performance degrades Agents converse with each other in decentralized fashion, avoiding the communication and processing bottlenecks of centralized systems. **Error! Reference source not found.** illustrates how agents might be assigned in a simple Internet Protocol (IP) network. Agent M1 monitors the firewalling gateway and M2 a switch. Agents M3 and M4 monitor two servers and agent M5 a mainframe. The thick arrows indicate regular and expected message passing. The hierarchical nature of Ethernet IP networks does tend to focus communication, as seen at M2, but agent M2 does not alone bear the responsibility for processing information from all other points in the network. Communication between agents is not limited to these paths; if, for example, agent M4 found need to restart its server, it could inform M2 of the action so that M2 could expect a change in traffic.

MASRR's approach to anomaly detection is for each agent to create a "thumbprint" of network behavior under normal circumstances. Agents compare these thumbprints to observed performance to detect departures from normal or expected activities. Thumbprinting allows the MASRR agent to recognize and respond to anomalous behavior whether arising from a known cause or from an event never before observed. An initial response to anomalous, undesired network behavior might be taken very quickly to prevent further degradation of service. Meanwhile, the agent can continue its diagnostics, following multiple lines of reasoning and de-conflicting or resolving information it collects or receives from its peers. A first response can be modified or rolled back as the agent analyzes the root-cause with increasing certainty.

An open question remains for anomaly detection: what features should be monitored to introduce as little data collection and processing overhead as possible and still accurately detect problems in the network? When developing identification signatures or classification models (typically learned from positive and negative examples of the attacks or other event to be recognized), one simply needs to monitor the features that appear in the signatures or model descriptions. In contrast, knowing the features whose values may indicate a yet-to-be-seen problem is a difficult task [19], [20].

For preliminary development of MASRR, we also applied our criteria for interoperability and small collection and processing overhead and chose to focus on using a subset of the Simple Network Management Protocol (SNMP) defined variables. SNMP has a standard interface for collecting information such as counts of packets delivered or dropped, counts of packets received for each protocol, the length of time that the system has been operating, and measures of the environment such as temperature and voltage. SNMP is widely implemented and, though some of the variable definitions are unique to specific elements, use of the protocol should also accommodate monitoring of network elements yet to be developed. Some MASRR agents would be installed on network hosts that are not the actual elements that they monitor – for example, an agent monitoring a network switch would likely be running on a nearby PC. The agent can collect SNMP data using efficient transport for low bandwidth and connection consumption.

The wide variety of variables that can be monitored with this standard protocol make SNMP data informative for predicting and detecting many sorts of usage, fault, or failure based anomalies, but more information may be needed for root cause analysis. The MASRR agent has the option of maintaining highly efficient ongoing monitoring and turning on more detailed analysis as needed, such as construction and examination of network flows. In addition, the agents can be configured to take as input into their reasoning engines the output of tools more specifically tailored to, for example, intrusion or misuse detection.

MASRR ARCHITECTURE

The MASRR architecture shown in Figure 2 is designed with interoperability in mind. The agent's logic and reasoning module is held separate from its interactions with the network elements that it monitors and on which it executes. This is in keeping with a typical model – view – controller layered architecture and allows the agent's knowledge base to be written and maintained for all platforms. Prototype development was coded in the platform-independent Java language with an additional library in C++ which can be compiled to different target machine environments.

The component we have called the Mailroom encapsulates the agent program's input-output (I/O). It allows the agent to generically obtain and reason about information from specific platforms, such as requesting its host's IP address or default routing gateway and relying on the Mailroom to handle details such as whether to use the command `ipconfig` (on Windows®) or `route` or `ifconfig` (on Linux™), and how to parse the desired information from the command output. The Mailroom also abstracts away the management of network communication with peer agents and other devices, so that the agent proper can send a message to "Peer M3" without concerning itself about addressing and connection management.

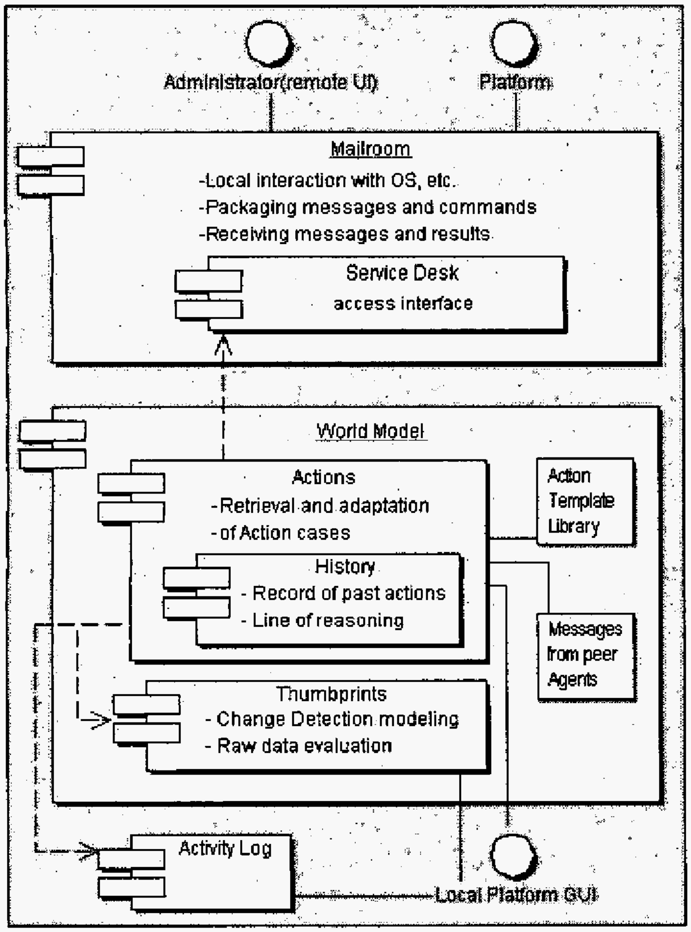

Figure 2. MASRR architecture

The World Model comprises what we think of as the agent's mind. It includes the ongoing Thumbprint assessments of network behavior. An important part of the Thumbprints component is Stottler Henke's innovative Change and Anomaly Detection (ChAD) data mining system [21]. ChAD's unique, adaptive approach allows it to report on changes in behavior, to transition between learned normal periods of behavior, and to adapt itself to the changes as needed. Thumbprinting includes heuristic evaluation of raw data as well as the output of the ChAD system, using encoded expertise from network security and management areas.

The Actions component further integrates network management and security. Using domain knowledge, it links symptoms to responses. The prototype MASRR system includes a library of action cases, which are composed of one or more steps and may contain their own logic for handling ordering of steps, time-outs, and contingencies. Action cases are indexed by evaluations of network behavior, including the Thumbprints and messages from peer agents. One or more action cases are selected in response to a given set of evaluations. The agent can pursue multiple lines of reasoning, using a recorded history, and can take intermediate action while gathering more information to decrease uncertainty about possible causes of problems (concurrent diagnosis). Certainly the library cannot contain actions for every conceivable event, but there are sufficient representative cases that can be adapted to respond to and improve many situations, even those not previously observed.

The Thumbprinting and the Actions modules combine to provide effective response to network-degrading events, whether caused by fault or attack. False alarms are reduced by a thumbprinting method that adapts to changes even as it reports them, and by encoding actions and their selection indices that promote remediative response as the agent obtains diagnostic information and reassesses the effects of its actions. More detail is given in the next sections on ChAD and on enhancing agent reasoning for mission-critical resource availability.

ANOMALY DETECTION DETAILS

The inability of many IDSs to accurately identify alarm conditions has been an obstacle to trusting automated response systems. Stottler Henke developed ChAD, a unique anomaly detection component that can form a picture of what constitutes normal behavior in-place with "live" data.

This innovation allows MASRR agents to model their localized network views at a very close fit, an important contribution to improved accuracy, as generic models or those built for one particular system may not prove accurate when installed on other systems. This is particularly true of the network management domain, in which every network has unique topology and usage characteristics. Other research has been done on anomaly detection using models constructed from normal data for intrusion detection and insider misuse [22], [23]. Recent work by [19] applies a signal processing approach to fault as well as intrusion detection by looking for anomalous abrupt changes in network measures.

In addition to initial accuracy, ChAD provides adaptive modeling to maintain its fit as network characteristics change over time. ChAD extends a recent development in data mining, the Concept-adaptive Very Fast Decision Tree learner (CVFDT) [24], which has been shown to accurately detect and adapt to statistical changes in the data distribution and to operate robustly on data with noise. Decision tree models themselves are not typically used to develop a signature of "normal" behavior of a system. They require labeled examples of the various possible behavior categories in order to build useful and accurate models. However, Stottler Henke has developed a method of using the CVFDT algorithm to build signature models in order to detect when the system deviates from normal or expected operation. ChAD can indicate not only when the modeled system is changing, but also can be quite informative as to the kind and severity of the change. When ChAD models are stable, we know that the system is operating normally.

During a training period, changes in behavior detected by ChAD can be used to segment its models into intervals of expected behaviors. For example, it is likely that typical network usage on a Monday morning will look very different from that on a Saturday night. Usage-specific modeling provides closer-fitting models at any given time and informs the agent reasoning component as to whether observed changes are expected and thus normal.

ChAD's adaptive modeling imparts robustness, revising the baseline models along with changing normal characteristics and avoiding the brittleness described in [10]. ChAD is self-calibrating after routine maintenance or other changes impacting system performance, and permit the addition or removal of monitored features or custom properties as network elements are added, removed and upgraded. While the statistical approaches outlined by [19] will adapt over time as well, ChAD adaptive modeling coupled with agent reasoning can be used to detect anomalous or detrimental behaviors that may be slowly introduced by accruing faults or stealthy attacks.

AGENT REASONING, AUTOMATED RESPONSE

MASRR's reasoning component acts as the knowledge base of the agent. This is the module that will interpret output from ChAD and choose what, if any, adjustments should be made to network elements. In its most straightforward form, the MASRR agent uses the current Thumbprint to retrieve an action template, fills in the template and submits it to the Mailroom for execution. This "quick and dirty" approach has the appeal of low overhead and rapid response, treating observed symptoms without necessarily having all diagnostic information available. It also serves as a reasonable default when confronted with previously unknown problems.

Building the current Thumbprint entails interpreting output from the ChAD module and combining it with other information sources, including reports received from peer agents. These possibly conflicting details are reconciled and the results are sent to peers. De-conflicting and sharing information reduces duplication of messages and effort by agents and increases their levels of certainty in diagnosis and action selection. This information fusion by cooperating agents gives a "big-picture" view of the network to an individual agent monitoring only a part of the network.

MASRR's flexible architecture supports a much richer agent knowledge base as well. Multi-part actions describe incremental mitigation along with possible root causes to be confirmed or refuted. As the agent gathers more information, its corrective measures become more specific to the actual problem and will be applied closer to the source, promoting the better survivability of the rest of the network. With ongoing monitoring, the agent can learn which actions are most effective against certain symptoms and can order possible causes by likelihood to speed diagnosis. Agents can also learn to be proactive, spotting problem precursors in the data stream and attaching corrective actions to those Thumbprints. Agents should also be outfitted with knowledge about administrative policies and guidelines on network performance so that any pre-existing violations in the network are corrected rather than learned as part of normal behavior.

The MASRR action library can also support several enhancements. Dynamic cost-benefit analysis can be added to the action selection module. Agents might find and deliver resources when observing certain traffic types, in a way similar to that proposed in Active Networks research [25]; this could be particularly useful in compact, mobile, or remote environments. For military communications and tactical networks, implementing agent course-of-action planning in support of mission objectives might be an attractive extension. An interface could be designed for specifying critical resources or communication endpoints so that agents can focus on keeping those resources and channels available or locating alternatives or substitutes.

CONCLUSION AND FUTURE WORK

Stottler Henke has developed an innovative approach for monitoring network performance and responding to both security and fault events to maintain reliability. MASRR achieves scalability and survivability using a decentralized collection of independent but cooperating agents. Interoperability is provided by a platform-independent development language and by a design that separates platform-dependent features from abstract agent reasoning. The system is robust under changing network elements and topologies, relying on common interfaces and modeling network behavior in-place. Improved accuracy and a configurable action library lends sufficient trust for automated response. Adaptive modeling ensures that accuracy does not degrade into brittleness over time.

However, there are modifications needed to deploy MASRR in a military communications environment. Feature selection will need to be revisited, as military communications networks comprise such variety as low and high radio frequencies, fixed and mobile position transmitters and receivers, fixed and ad hoc networks, line-of-site requirements, etc. ChAD model segmentation and change sensitivity levels will need to be tuned to support highly dynamic environments. In support of mobile devices, we will need to pay particular attention to a compact representation of action and symptom libraries for a small installation footprint. Processing and inter-agent communication will need to be optimized to conserve power and bandwidth constraints, and "silent" mode reasoning may also need to be included. Agent code and authentication measures must be "hardened" against tampering, masquerading, and reverse engineering. Stottler Henke, specializing in artificial intelligence research and development, is in the process of partnering with military and network domain experts to extend the capabilities of the MASRR system.

REFERENCES

[1] Tripwire Policy Manager, http://www.tripwire.com.

[2] Microsoft Windows® Group Policy and Active Directory, http://www.microsoft.com.

[3] Symantec AntiVirus. http://www.symantec.com.

[4] VirusScan. http://www.mcafee.com.

[5] RealSecure®, http://www.isi.net.

[6] Cisco IDS (formerly NetRanger). http://www.cisco.com.

[7] SolSoft NP, http://www.solsoft.com.

[8] SPECTRUM suite, http://www.aprisma.com.

[9] StormWatch, http://www.okena.com.

[10] T. Oates, *Fault Identification in Computer Networks: A Review and a New Approach*, Technical Report 95-113, University of Massachusetts at Amherst, Computer Science Department. 1995.

[11] "Reducing Fratricide in CyberWar", talk given by Ed Sherman, Aprisma Management Technologies, at the Lockheed Martin DDoS Conference, 12/19/01.

[12] Mike Fratto, *Anomaly-Detection Services: Know they Enemy*, Information Week Security Pipeline, February 19, 2004. http://informationweek.securitypipeline.com/howto/17602432.

[13] Peakflow by Arbor Networks, http://www.arbornetworks.com/.

[14] StealthWatch by Lancope, http://www.lancope.com.

[15] Captus IPS. http://www.captusnetworks.com/.

[16] Top Layer Attack Mitigator IPS. http://www.toplayer.com/.

[17] Stephanie Forrest, Steven A. Hofmeyr, Anil Somayaji and Thomas A. Longstaff, *A Sense of Self for Unix Processes.* In Proceedings of the 1996 IEEE Symposium on Security and Privacy, IEEE Computer Society Press, Los Alamitos, CA, pp. 120–128 (1996).

[18] Cynthia S. Hood and Chuanyi Ji, *Intelligent Agents for Proactive Fault Detection*, IEEE Internet Computing, vol.2, no.2, pp.65-72, Mar/Apr 1998.

[19] Marina Thottan & Chuanyi Ji, *Anomaly Detection in IP Networks,* IEEE Transactions on Signal Processing Vol. 51, No. 8, AUGUST 2003

[20] João B. D. Cabrera, Lundy Lewis, Xinzhou Qin, Wenke Lee, Ravi K. Prasanth, B. Ravichandran and Raman K. Mehra. *Proactive Detection of Distributed Denial of Service Attacks using MIB Traffic Variables –A Feasibility Study,* Proceedings of the 7th IFIP/IEEE International Symposium on Integrated Network Management, Seattle, WA - May 14-18, 2001.

[21] L. W. Jones, ChAD: *Change and Anomaly Detection - Discovering when a system is not behaving normally,* http://www.stottlerhenke.com/solutions/computer_security/MASRR/ChAD.htm

[22] Eleazar Eskin, Matthew Miller, Zhi-Da Zhong, George Yi, Wei-Ang Lee, Sal Stolfo. *Adaptive Model Generation for Intrusion Detection Systems,* Workshop on Intrusion Detection and Prevention, 7th ACM Conference on Computer Security, Athens, GR: November, 2000.

[23] Christina Warrender, Stephanie Forrest, and Barak Pearlmutter. *Detecting Intrusions using system calls: alternative data models.* In Proceedings of the 1999 IEEE Symposium on Security and Privacy, pages 133–145. IEEE Computer Society, 1999.

[24] Geoff Hulten, Laurie Spencer, and Pedro Domingos. *Mining Time-changing Data Streams*, Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 26-29, 2001, San Francisco.

[25] David Wetherall, Ulana Legedza, and John Guttag, *Introducing new internet services: Why and how,* IEEE Network, May/June 1998.