

A Comparative Analysis of Current Intrusion Detection Technologies

James Cannady

Jay Harrell

Georgia Tech Research Institute
Georgia Institute of Technology
Atlanta, Georgia 30332-0800

Georgia Tech Research Institute
Georgia Institute of Technology
Atlanta, Georgia 30332-0800

ABSTRACT

Intrusion detection is a significant focus of research in the security of computer systems and networks. This paper presents an analysis of the progress being made in the development of effective intrusion detection systems for computer systems and distributed computer networks. The technologies which are discussed are designed to detect instances of the access of computer systems by unauthorized individuals and the misuse of system resources by authorized system users. A review of the foundations of intrusion detection systems and the methodologies which are the focus of current development efforts are discussed. The results of an informal survey of security and network professionals is discussed to offer a real-world view of intrusion detection. Finally, a discussion of the future technologies and methodologies which promise to enhance the ability of computer systems to detect intrusions is provided.

Keywords: Intrusion detection, anomaly detection, misuse detection, computer security.

1. INTRODUCTION

This paper discusses the current research and development efforts to detect internal and external penetrations of computer systems and networks. The area of intrusion detection is central to the concept of computer security. While a number of methods can be employed to protect the data stored within a computer system, the ability to identify instances of an attack on the computer is paramount if an effective security mechanism is to be developed.

Halme and Bauer ([6]) have identified intrusion detection as one of six components in their taxonomy of anti-intrusion techniques. The first three components which they identified; prevention, preemption, and deterrence, are primarily based on passive measures which decrease the likelihood of a successful attack on a system. These components address the policy related issues of information security and those elements which can be incorporated into a system with minimal effort. Examples of these include the establishment of organizational security guidelines, security education and training, and the posting of warning notices on the initial screens of a system.

The last three components, deflection, detection, and countermeasures, are more active measures designed to protect the critical elements of a system. Of the six components, the accurate detection of a system intrusion is the most critical. While additional measures may be very effective at preventing an eventual

penetration of the system, all security measures rely on the accurate identification of an attacker prior to the employment of defensive measures.

The need for effective intrusion detection mechanisms as part of a security mechanism for computer systems was recommended by Denning and Neumann ([16]). They identified four reasons for utilizing intrusion detection within a secure computing framework:

1. Many existing systems have security flaws which make them vulnerable, but which are very difficult to identify and eliminate because of technical and economic reasons.
2. Existing system with security flaws cannot be easily replaced by more secure systems because of application and economic considerations.
3. The development of completely secure systems is probably impossible.
4. Even highly secure systems are vulnerable to misuse by legitimate users.

This report attempts to build upon previous surveys of intrusion detection research and technology, ([14] and [13]). While a satisfactory review of the topic of intrusion detection requires a review of the theoretical basis for the area of intrusion detection, techniques and methodologies discussed in previous surveys will receive a minimum of attention in this document. The purpose of this paper is to present the current state-of-the-art in intrusion detection techniques, methodologies, and tools. An attempt is made to establish a baseline of the current state of research so that subsequent research in the field can benefit from the efforts conducted to date. However, this report does not attempt to address every research effort which is currently underway. The field of intrusion detection is in the process of expanding beyond the methodologies which have been utilized for the past fifteen years. While many of these approaches have yet to demonstrate a completely effective and efficient intrusion detection mechanism, these new efforts are addressing areas of computer vulnerabilities which had long been outside of the reach of traditional intrusion detection research.

This paper is divided into three primary areas. The first section provides an overview of intrusion detection fundamentals. These include the metrics which are commonly used for quantitative analysis of available data, the models, and approaches which are utilized most often in the development of intrusion detection systems. The second section describes some of the current technologies and methodologies which are being developed in the area of intrusion detection research. Finally, the results of a recent informal survey are presented to offer a glimpse of those security officials, system administrators and network professionals who actually develop, utilize and evaluate intrusion detection systems on a day to day basis.

2. INTRUSION DETECTION FUNDAMENTALS

2.1 Development/Evolution of Intrusion Detection Mechanisms

The first major work in the area of intrusion detection was discussed by J.P Anderson in [2]. Anderson introduced the concept that certain types of threats to the security of computer systems could be identified through a review of information contained in the system's audit trail. Many types of operating systems, particularly the various "flavors" of UNIX, automatically create a report which details the activity occurring on the system. Anderson identified three threats which could be identified from a concentrated review of the audit data:

1. External Penetrations - Unauthorized users of the system.

2. Internal Penetrations - Authorized system users who utilize the system in an unauthorized manner.
3. Misfeasors - Authorized user who mislead their access privileges.

Anderson indicated that a particular class of external attackers, known as clandestine users, were particularly dangerous to the system resources. Clandestine users are those who evade both system access controls and auditing mechanisms through the manipulation of system privileges or by operating at a level that is lower than what is regularly monitored by the audit trail. Anderson suggested that clandestine users could be detected by lowering the level which is monitored by the audit trail, monitoring the functions that turn off the audit systems, or through a comparison of defined "normal" usage patterns of system resource usage with those levels which are currently observed.

While the concept of manually reviewing operating system audit records for indications of intrusions was recognized as an extremely inefficient method of securing a computer system, Anderson's article served to initiate research into the area of intrusion detection. Subsequent research involved the development of automated techniques for the review of audit record data. Until recently, most intrusion detection mechanisms were based on an automated approach to Anderson's concepts. However, the recent development of new intrusion detection approaches and, more significantly, the necessary application of intrusion detection technologies to networked environments, is changing the focus of intrusion detection research. The most significant of these new technologies and approaches will be discussed in this report.

Dr. Dorothy Denning proposed an intrusion detection model in 1987 which became a landmark in the research in this area. [4] The model which she proposed forms the fundamental core of most intrusion detection methodologies in use today. Because of the applicability of these concepts to most accepted intrusion detection systems, an overview of the primary concepts of the model are presented here to provide a basis of understanding the core technology.

2.2 Foundations of an Intrusion Detection System

Metrics

Any statistical intrusion detection methodology requires the use of a set of definable metrics. These indices are the elements upon which all of the tool's statistical analysis is based. These metrics characterize the utilization of a variety of system resources. The resources which would be used in the definition of the metrics are required to be system characteristics which can be statistically based, (i.e., CPU usage, number of files accessed, number of login attempts).

These metrics are usually one of three different types. Event counters identify the occurrences of a specific action over a period of time. These metrics may include the number of login attempts, the number of times that a file has been accessed, or a measure of the number of incorrect passwords that are entered.

The second metric, time intervals, identify the time interval between two related events. Each time interval compares the delay in occurrence of the same or similar event. An example of a time interval metric is the periods of time between a user's logins.

Finally, resource measurement is the concept of quantifying the amount of resources used by the system over a given period of time. Resource measurement incorporates individual event counters and time interval metrics to quantify the system. Examples of resource measurements include the expenditure of CPU time, number of records written to a database, or the number of files transmitted over the network.

While not normally considered with the "traditional" intrusion detection metrics, keystroke dynamics is another method of quantifying a user's activities which offers an effective measure of user identification. The concept involves the development of an electronic signature of a user based on their individual typing characteristics. These characteristics usually include typing speed, intervals in typing, number of errors, and the user's typing rhythm. These characteristics may be verified on login and/or monitored throughout a session. Complete intrusion detection mechanisms have been developed exclusively around the use of keystroke dynamics techniques. [13]

Models

The selected metrics are then used in statistical models which attempt to identify deviations from an established norm. The models which have been most frequently used include the Operational Model, Average and Standard Deviation Model, the Multivaried model, the Markovian model, and the Time Series Model. [3]

The Operational Model makes the assumption that an anomaly can be identified through a comparison of an observation with a predefined limit. This model is frequently used in the situations where a specific number of events, (i.e., failed logins), is a direct indication of a probable attack.

The Average and Standard Deviation Model is based on the traditional statistical determination of the normalcy of an observation based on its position relative to a specified confidence range. This model offers the advantage that it "learns" a user's behavior over time instead of requiring prior knowledge of the user's activities. As a result, the model can establish a foundation for the identification of potential anomalies for each user and identify potential problems from users who consistently behave in a manner which would normally indicate the misuse of system resources. This is particularly useful in identifying what is normal for an individual user without relying on a comparison with other users.

The Multivaried Model is built upon the Average and Standard Deviation Model. The difference between these two approaches is that the Multivaried Model is based on a correlation of two or more metrics. This model permits the identification of potential anomalies where the complexity of the situation requires the comparison of multiple parameters.

The Markovian Model is used with the event counter metric to determine the normalcy of a particular event, based on the events which preceded it. The model characterizes each observation as a specific state and utilizes a state transition matrix to determine if the probability of the event is high (normal) based on the preceding events. This model is particularly useful when the sequence of activities is particularly important.

The final model, the Time Series Model, attempts to identify anomalies by reviewing the order and time interval of activities on the network. If the probability of the occurrence of an observation is low, then the event is labeled as abnormal. This model provides the ability to evolve over time based on the activities of the users.

Profiles

These models are then used in the development of a variety of profiles which attempt to map the non-intrusive activities of the system. The profiles serve to establish a baseline of a user's behavior which can then be used for comparisons with the current observations.

Profiles usually consist of specific characteristics which would adequately identify a user. These characteristics often include measures such as login information, (i.e., frequency, origin, duration), program execution information, (i.e., frequency, CPU utilization), database access information, (i.e., tables accessed, data manipulation functions), and file access information (i.e., types of files accessed, created, or destroyed).

Behavior profiles can be devised for authorized system users, groups of users, or “strawman” profiles of typical network attacks.

Analysis Techniques

The final element in the basic structure of an intrusion detection system is determining how the collected information will be reviewed by the mechanism. Statistical Analysis and Rule-Based Analysis are the two general categories which are most commonly employed in intrusion detection systems. Statistical Analysis involves statistical comparison of specific events based on a predetermined set of criteria. The technique is typically employed in the detection of deviations from typical behavior and/or the similarity of events to those which are indicative of an attack.

The primary advantage of statistical analysis is in its ability to detect unanticipated intrusion patterns because the specific patterns do not have to be predefined. The technique looks for individual elements which may be part of an intrusion without relying on the completion of an entire sequence of specific activity. The disadvantage of statistical analysis is that it is prone to report an unacceptable number of false alarms because of its inability to quickly adapt to legitimate changes in a user's behavior.

Rule-Based Systems rely on sets of predefined rules which are provided by an administrator, automatically created by the system, or both. Each rule is mapped to a specific operation in the system. The rules serve as operational preconditions which are continuously checked in the audit record by the intrusion detection mechanism. If the required conditions of a rule are satisfied by user activity the specified operation is executed. [17]

Both statistical analysis and rule-based systems suffer from an inability to detect attack scenarios which may occur over an extended period of time. While the individual instances of suspicious activity may be detected by the system, it may not be reported if it appears to occur in a vacuum. In addition, intrusion scenarios in which multiple attackers operate in concert are also difficult for these methods to detect. Because they do not focus on the sequence of events in an attack, instead concentrating on the occurrence of individual elements, the division of an attack among several seemingly unrelated attackers is difficult for these methods to detect.

Rule-based systems also suffer from a lack of flexibility in the rule-to-audit record representation. Slight variations in an attack sequence can effect the comparison of the activity in the audit record to the existing rule to a degree that the intrusion is not detected by the intrusion detection mechanism. While increasing the level of abstraction of the rule-base does provide a partial solution to this weakness, it also reduces the granularity of the intrusion detection device.

Expert Systems

The early research efforts realized the inefficiency of any approach which attempted to require a manual review of a system's audit data. While the information necessary to identify attacks was believed to be present within the often voluminous audit data, an effective review of the material required the use of an automated system. The use of expert system techniques in intrusion detection mechanisms was a significant milestone in the development of effective detection-based information security systems.

An expert system consists of a set of rules which encode the knowledge of a human "expert". These rules are used by the system to make conclusions about the data which is returned by the intrusion detection system. Expert systems permit the incorporation of an extensive amount of human experience into a computer application which utilizes that information to identify activities which matched the defined characteristics of misuse and attack.

Unfortunately, expert systems require frequent updates by a system administrator to remain current. While expert systems offer an enhanced ability to review audit data, the frequently required updates may be ignored or performed infrequently by the administrator. At a minimum, this leads to an expert system with reduced capabilities. At worst, this lack of maintenance will degrade the security of the entire system by causing the system's users to be misled into believing that the system is secure, even as one of the key components becomes increasingly ineffective over time.

2.3 Approaches to Intrusion Detection

All current intrusion detection systems make four assumptions about the systems that they are designed to protect:

1. Activities taken by system users, either authorized or unauthorized, can be monitored.
2. It is possible to identify those actions which are indications of an attack on a system
3. Information obtained from the intrusion detection system can be utilized to enhance the overall security of the network.
4. A fourth element which is desirable from any intrusion detection mechanism is the ability of the system to make an analysis of an attack in real-time. This would allow the intrusion detection mechanism to limit the adverse effects which are perpetrated on the system. An effective use of this element is probably the most difficult component of an intrusion detection system to achieve. While metrics can be developed which monitor all aspects of a user's behavior, the resulting degradation on the overall performance of the system may require that a thorough analysis be conducted off-line, thus eliminating a real-time detection capability.

There are currently a variety of approaches being utilized to accomplish the desirable elements of an intrusion detection system. Two of these, anomaly detection and misuse detection, form the core of several intrusion detection techniques which currently exist. Other approaches, such as pattern recognition, are attempting to identify new methods of identifying information system attacks.

Anomaly Detection

Anomaly detection is the general category of intrusion detection which works by identifying activities which vary from established patterns for users, or groups of users. Since masquerading as a legitimate user is a very powerful method for an attacker to gain access to system resources, this type of approach looks for the variations in behavior which might indicate a masquerade. Anomaly detection typically involves the creation of knowledge bases which contain the profiles of the monitored activities.

Several types of profiles are generally used in anomaly detection. User profiles contain the parameters of a user's typical session. While these profiles are potentially the most useful in identifying indications of anomalous behavior, they are also the most difficult to create and to maintain. A balance must be struck between establishing short-term profiles, which establish patterns of recent activity and long-term profiles, which establish a historical overview of a user's activities. Unless they are updated frequently, user profiles can lead to a large number of false alarms as the user's activities change over time.

To avoid, or at least modify, the adverse effects of the system's legitimate users, some anomaly detection systems include the use of user group profiles. In this method the user is placed in a work group which may or may not represent the actual assigned duties of the user. More frequently the group characterizes individuals with similar computer usage patterns. While group profiling assists in the maintenance of the detection mechanism, these profiles are often defined so broadly that unauthorized users can slip through the screen by behaving roughly similar to the typical user in the group.

Other profiles which are frequently used in anomaly detection include resource profiling, (monitoring the system-wide use of accounts, applications, communication ports, etc.), and executable profiling, (monitoring the use of printers, files, and other resources which cannot easily be attributed to a single user). This user-independent form of profiling is useful in detecting the presence of viruses and Trojan horses.

Anomaly detection mechanisms are usually dependent on input from an operating system's audit record. This analysis of the audit trail imposes potentially significant overhead requirements on the system because of the increased amount of processing power which must be utilized by the anomaly detector. Depending on the size of the audit trail and the processing ability of the system, the review of audit data could result in the loss of a real-time analysis capability.

Misuse Detection

The second general approach to intrusion detection is misuse detection. This technique involves the comparison of a user's activities with the known behaviors of attackers attempting to penetrate a system. Misuse detection also utilizes a knowledge base of information. The misuse knowledge bases include specific metrics on the various techniques employed by attackers when the knowledge base was created.

While anomaly detection typically utilizes threshold monitoring to indicate when a certain established metric has been reached, misuse detection techniques frequently utilize a rule-based approach. When applied to misuse detection, the rules become scenarios for network attacks. The intrusion detection mechanism identifies a potential attack if a user's activities are found to be consistent with the established rules. The use of comprehensive rules is critical in the application of expert systems for intrusion detection.

Like anomaly detection techniques, misuse detection systems suffer from the potential performance degradation which results from a dependency on audit trails for input. This disadvantage can be mitigated by improved system performance and reduced audit record sets.

Combined Anomaly/Misuse detection

Research has also been conducted into intrusion detection methodologies which combine the anomaly detection approach and the misuse detection approach ([12]). These techniques seek to incorporate the benefits of both of the standard approaches to intrusion detection. The combined approach permits a single intrusion detection system to monitor for indications of external and internal attacks.

While a significant advantage over the singular use of either method separately, the use of a combined anomaly/misuse detection mechanism does possess some disadvantages. The use of two knowledge bases for the intrusion detection system will increase the amount of system resources which must be dedicated to the system. Additional disk space will be required for the storage of the profiles, and increased memory requirements will be encountered as the mechanism compares user activities with information in the dual knowledge bases. In addition, the technique will share the disadvantage of either method individually in its inability to detect collaborative or extended attack scenarios.

Pattern Recognition

One of the few intrusion detection methodologies which has departed from the established use of anomaly and misuse detection profiles is pattern recognition. In this approach, a series of penetration scenarios are coded into the system.

Pattern recognition possesses a distinct advantage over anomaly and misuse detection methods in that it is capable of identifying attacks which may occur over an extended period of time, a series of user sessions, or by multiple attackers working in concert. This approach is effective in reducing the need to review a potentially large amount of audit data.

The key disadvantage of pattern-recognition techniques is the reliance of the system on predefined intrusion scenarios. If an attack characteristics do not match one which has been coded into the system, the intrusion may not be detected. As a result, pattern-recognition mechanisms are still dependent on a statistical-type of intrusion detection approach to be a truly effective security mechanism.

Network Monitoring

A final method of detecting system intrusions which is currently in use is the use of various network monitoring techniques. [15] These methodologies passively monitor network activity for indications of attacks.

Network monitoring offers several advantages over traditional audit-based intrusion detection systems. Because many intrusions occur over network at some point, and because networks are increasingly becoming the targets of attack, these techniques are an excellent method of detecting many attacks which may be missed by audit-based intrusion detection mechanisms.

The greatest advantage of network monitoring mechanisms is their independence from reliance on audit data. Because these methods do not require input from any operating system's audit trail they can use standard network protocols to monitor heterogeneous sets of operating systems and hosts.

Independence from audit trails also frees network monitoring systems from possessing an inherent weakness caused by the vulnerability of the audit trail to attack. Intruder actions which interfere with audit functions or which modify audit data can lead to the prevention of intrusion detection or the inability to identify the nature of an attack. Network monitors are able to avoid attracting the attention of intruders by passively observing network activity and reporting unusual occurrences.

Another significant advantage of detecting intrusions without relying on audit data is the improvement of system performance which results from the removal of the overhead imposed by audit trails. The process of analyzing audit trails increases the performance degradation of the system. In addition, techniques which move the audit data across network connections reduce the bandwidth available to other functions. Network monitoring techniques can increase performance of networks by 5 to 20 percent compared to audit-based systems. [15]

3. CURRENT INTRUSION DETECTION TECHNIQUES

Several intrusion detection techniques were described in detail in [13] and [14]. This paper will not attempt to readdress those techniques and tools. The following is a review of the significant developments in intrusion detection research which have been made in the past several years.

3.1 NIDES

SRI International began research into an intrusion detection expert system in 1985. [12] The result of the research, the Intrusion Detection Expert System (IDES) has become a standard in intrusion detection systems. Several current systems are based in part on IDES prototype technology, ([19, 20, 24]).

The Next-Generation Intrusion Detection Expert System (NIDES) is the comprehensive enhancement to IDES. [1] NIDES is a real-time intrusion detection application which integrates a statistical analysis-based anomaly detector and a rule-based misuse detection system. This combination gives NIDES the ability to detect penetrations from internal and external attacks.

SRI incorporated a number of significant improvements into NIDES. In addition to modularizing the application, NIDES includes an enhanced statistical analysis component and additional support for a strict client-server model. NIDES also includes a comprehensive user interface that permits access to all of the applications capabilities, as well as a context-sensitive help system.

While NIDES is regarded as the current state-of-the-art in a combined anomaly and misuse detection system, the application retains the difficulty possessed by all similar models in detecting collaborative attacks, long-term penetration scenarios and virus propagation. Another potential disadvantage is that NIDES retains a reliance on the system's audit record for input. Future expansions of the rulebase and the development of profiles of entities other than users should reduce the potential vulnerabilities which are not adequately addressed by the current system.

3.2 DIDS

The Distributed Intrusion Detection System (DIDS) is an intrusion detection mechanism which was developed jointly by the University of California at Davis, Lawrence Livermore Laboratory, Haystack Laboratory and the U.S. Air Force. DIDS combines attributes of a network monitoring system with the system-level capabilities of an audit record-based combined anomaly/misuse detector. DIDS incorporates a monitor on each host, a monitor on the local area network (LAN), and a DIDS director.

Each host monitor consists of a host event generator and a host agent. The host event generator reviews the audit data from the host for indications of events which may be part of an attack. The DIDS host event generators also utilize user and group profiles to identify anomalous behaviors in the audit record. The information identified by the host event generator is reported to the DIDS director by the host agent.

The LAN monitor is the network equivalent of the host monitor. It includes the LAN event generator and the LAN agent. However, unlike the host event generator, the LAN event generator does not review audit data. The LAN event generator utilizes the network monitoring approach to review all network traffic, including host-to-host connections and resources used. The information obtained by the LAN event generator is reported to the DIDS director by the LAN agent.

The DIDS director forms the heart of the intrusion detection mechanism. It is composed of three components, the communications manager, an expert system and a user interface. The communications manager receives input from each of the host monitors and from the LAN monitor and forwards the information to the expert system for analysis. The communications manager is also capable of forwarding requests for additional information from the expert system to the host monitors and the LAN monitor. The DIDS expert system is a rule-based system which is responsible to analyzing the information received from the monitors and reporting it to the security official. The final component of the DIDS system, the user interface, allows a security official to interactively review the status of the system, receive reports from the expert system, and request additional security-related information from the system.

One of the essential elements of the DIDS system is the use of a Network-user Identification (NID). This is a process of establishing an identifier for each individual when they are initially logged into the network. This is especially important because many attackers use multiple accounts to attack a network or use the

interconnectivity of computer networks to attempt to disguise their identity. Once a user has logged into the network and been assigned a NID all subsequent activity conducted by that user is attributable through the NID.

While the NID offers the potential to track an intruder through a variety of hosts and possible identities, there are ways to defeat the mechanism. By logging out of the monitored domain and then reentering under a different userid, an attacker can prevent DIDS from relating the two sessions. In addition, the DIDS system probably cannot attribute two related sessions to the same user if the user passes through an unmonitored domain. These difficulties aside, an initial DIDS prototype has successfully demonstrated the ability to track users through a monitored domain.

Because of the complexity of the system and its use of audit data, DIDS retains the negative effect on the performance of the system which plagues most traditional intrusion detection systems. While this could be a significant disadvantage of the DIDS system, the innovative design of the system effectively addresses the difficulty in identifying intrusions in a networked environment.

3.3 STAT/USTAT

The State Transition Analysis Tool (STAT) and USTAT, the variation of STAT which was designed specifically for the UNIX operating system environment, are rule-based penetration detection approaches which characterize the process of an attack on a computer system as a series of transitions from an initial state to a compromised state. The technique defines specific events, called signature actions, which occur between each of the intermediate transitions. The omission of any of the signature actions results in a failed attack on the system.

Once the relevant system states have been defined and the required signature actions have been identified, the approach utilizes state transition diagrams to describe the attack's progress through a penetration scenario. State transition diagrams are useful because they provide a graphical representation of the requirements and compromise of the penetration while describing the events which must occur for the attack to be successful.

If the current pattern of activity matches an established intrusion scenario, STAT/USTAT has the ability to predict the future activities of an attacker. The ability to predict behavior offers the advantage of allowing the Security Administrator to be more confident that an actual attack is occurring prior to utilizing any countermeasures. As more of the established scenario's activities are matched, the confidence level that an attack is occurring increases. In addition, because this technique does not rely on possibly unrelated events to indicate a potential attack, the incidence of false alarms reported by the system should be significantly reduced.

Another advantage of this approach is that because STAT/USTAT selects specific audit data for confirmation of potential intrusion patterns, only a portion of the audit data is actually reviewed. This reduces the reliance of the system on the entire set of audit data, thereby reducing the required storage space and memory requirements necessary for processing an entire audit trail.

While STAT/USTAT offers significant advantages in its approach to intrusion detection, the technique is unable to detect other attack-type behavior such as denial of service attacks and masquerading. Because these indications of an attack cannot be ignored by an effective intrusion detection system, a mechanism which employs STAT/USTAT would also require a complementary rule-based anomaly/misuse detection system. [8]

3.4 Tripwire

In November 1992, the COAST laboratory at Purdue University introduced Tripwire. [10] Tripwire is an integrity checking program which permits a system administrator to monitor system files for addition, deletion, or modification. The program is estimated to have been installed on several thousand systems worldwide.

While it is not an intrusion detection mechanism, Tripwire does provide valuable information for the process of detecting attacks on a system. Tripwire is designed for the UNIX operating system environment. The program has proven to be scalable, portable, and manageable.

Tripwire utilizes input from a configuration file and a database to identify areas of interest. The configuration file consists of a description of the file systems which are to be monitored. The database contains the signatures of files which match the configuration. The signatures of the files are calculated based on the contents of the system files. The signature computation is easy to derive but impossible to reverse.

Tripwire operates in one of four modes. In the database initialization mode, the program generates a database which contains all of the relevant information on the system files, including signatures. Because the baseline database is being generated based on the files which currently exist in the system, it is critical that the existing database is free of logic bombs, viruses, Trojan horses, or other attack programs.

The integrity checking mode results in the creation of a new database from information contained in the configuration. The information in the new database is compared with the results contained in the original database. Any discrepancies are processed through a filter which determines which file attributes can be changed without adversely affecting the system. The remaining identified changes are then reported to the system administrator.

The final two operating modes are used to ensure that the information in the database is consistent. The database update mode calculates new signatures for those files which have been legitimately changed. In the interactive database update mode the program generates a list of those files which have been modified and updates those which are identified by the system administrator as legitimate.

Tripwire is a good tool for monitoring the status of system files. However, it is limited in its capabilities. Tripwire makes no pretense of insuring the complete security of the computer system. It functions to notify system administrators of a very important indication of an intrusion. This information, combined with other security-related tools, should provide a more secure operating environment.

3.5 GrIDS

Researchers in the COAST laboratory have recently proposed a novel approach to intrusion detection based on the analysis of activity graphs. The Graph-Based Intrusion Detection System (GrIDS) is designed to analyze network activity in large networks for the presence of attacks. [11]

GrIDS aggregates the actions of a network's users into the activity graphs. Based on a review of the structure of these graphs the system can identify patterns which indicate intrusive behavior. In addition to diagramming the basic network activity, GrIDS incorporates supplementary information in the form of attributes to the tree-like structure of the diagram. Information received from other intrusion detection devices and network monitors can be included in the attributes of the activity graphs.

Individual types of graphs will be maintained in graph spaces with the GrIDS system. Because there are a number of possible attacks on the network, multiple graph spaces must be maintained. Each graph space is dependent on a specific rule set which modifies the graphs within its graph space based on inputs to the system.

GrIDS is able to analyze activity on large networks because of its ability to model networks as a series of hierarchies. Each area within the hierarchy has a GrIDS module which is responsible for that area. Any activity which crosses area boundaries will be passed up to the GrIDS in the next higher level for resolution. The GrIDS in that level builds reduced graphs which model the underlying structure on a smaller scale. This ability to model subhierarchies allows GrIDS to monitor networks of increasing complexity.

The true promise in the GrIDS system is in its ability to assist users in creating rule sets for the system. GrIDS includes a policy language which enables administrators to translate organizational policies and guidelines into rule sets which are used to analyze the network activity. This technique allows the GrIDS to expand from merely identifying indications of external attacks to detecting any activity which violates established network usage policies.

The designers of GrIDS have not attempted to develop a complete intrusion detection device. Instead, they have proposed an innovative technique which addresses elements of intrusion detection which have been largely ignored in the past.

3.6 Thumbprinting

Thumbprinting is a method of tracking intruders through a sequence of logins, referred to by the authors as a connection chain. [23] While it is not intended to be an independent intrusion detection system, it could prove to be a valuable addition to other technologies.

Thumbprinting was developed by researchers at the University of California at Davis in response to a weakness in DIDS. Because DIDS is unable to correlate to parts of a connection chain when a user has exited and then reentered outside of the DIDS domain, thumbprinting was devised to compare the content of the connections in the chain. Since commands issued by a user should remain the same as they pass through the various hosts in the connection chain, summaries of the content of connection at two points could be compared to determine if they were links in the same chain. The summaries would be generated by passively monitoring the network traffic at each host.

A current weakness in this approach is that it assumes that the content of the connections along the chain are the same. As a result, the use of different encryption techniques by two points would render the method useless.

3.7 Cooperating Security Managers

While DIDS takes a centralized security approach to network intrusion detection, Cooperating Security Managers (CSM) decentralizes the process. A separate CSM is run on each computer which is connected to the network. [25]

Each CSM consists of six elements. The heart of the CSM is the Security Manager (SECMGR). The SECMGR receives input from the various CSM components and coordinates with CSM's on other hosts as users pass through the network. The command monitor (CMNDMON) intercepts the commands from the user and forwards them to the host intrusion detection system (IDS). While CSM requires the presence of an intrusion detection system on each host, the actual mechanism is separate from the CSM and can therefore be any intrusion detection tool. Any intrusions detected by the IDS are reported to the SECMGR. The CSM Intrusion Handler (IH) is one of the distinguishing characteristics of CSM. Instead of simply reporting intrusive activity to a security administrator, the IH can also be configured to take more active measures against an intruder. These include terminating the user's current session, disabling the account being utilized by the alleged intruder, or backing up files which may be modified or deleted by an attacker. The SECMGR uses TCP to communicate with other CSM's through the communication handler (TCP COM).

CSM only communicates with the CSM immediately before it in the connection chain, not all hosts on the network. Each CSM is responsible for relaying the message through the network.

In addition to addressing the need for detecting intrusive activity in a networked environment, CSM is also scalable and portable because it is not specifically designed for any particular network-wide operating system. Each CSM is unaware of the operating environment on the other CSM's hosts. As long as a CSM has been developed for the operating system which is used on a host, it can be attached to a CSM-monitored network.

CSM's ability to utilize a variety of intrusion detection systems also prevents the system from being limited by any of the specific approaches to intrusion detection. As new approaches are developed which more efficiently process user information, they can be incorporated into the CSM, effectively upgrading the CSM as a whole.

4. RESULTS OF SURVEY

Over the period of one month, (April 1-May 1, 1996), we conducted an informal survey of computer users, administrators, managers, and others involved in the protection of computer resources. The survey was not intended to be a statistically-sound poll of information security professionals. Instead we attempted to receive inputs on the real-world application of intrusion detection and prevention systems from a limited range of individuals. The survey was disseminated over the Internet to subscribers of several computer security-related newsgroups. Because of this, we recognized that the results of the survey would be skewed toward those which are security-conscious and those who have experienced attacks on their networks.

There was no "typical" respondent to the survey. While most of the individuals who returned completed questionnaires were somehow directly responsible for administering a computer network, there were also a sizable number of users, managers, consultants, and data security specialists who responded to our request for information. Similarly, there was a good representation of types of organizations which employed the respondents, types of operating systems used on the networks, and numbers of workstations and users on the networks.

When questioned about the levels of concern for security which are expressed in their organizations, the results were generally as we expected. The respondents were asked to rate the perceived significance of security to themselves, their network administrators (if different), their senior management, and the users of their systems, on a scale of 1 (minimal) to 10 (very significant). The respondents considered security to be a major concern with an average score of 8.9. The reported levels of perceived significance of the other categories diminished to the typical users level of concern for security rating of 2.4.

We also asked the survey respondents to rate six types of threats in order of concern from 1 (lowest perceived threat) to 6 (most significant threat to their network). Because of the wide range of organizations represented by the respondents there was a noticeable dichotomy in the responses which were received. Individuals who are currently in academia viewed hackers/crackers as their greatest threat, while disgruntled employees and economic competitors were perceived as the most significant threat to networks in business. Two categories of threats, phreakers and individuals acting on behalf of foreign governments, were consistently evaluated as representing a very low threat to their networks.

The survey results indicated that most of the respondents utilize a combination of security devices on their networks. Ninety percent of the respondents utilize the existing security features which are present in their host operating system. Some type of intrusion detection system and firewall mechanism are used in seventy-two percent of the networks. The respondents reported that they were utilizing their security mechanisms to defend their networks from external penetrations, masquerades, internal attacks, viruses and

denial-of-service attacks. There were very few respondents who reported that they were attempting to protect their network from a single threat.

The area of this survey which provided the most insight for the purposes of this paper were the questions which attempted to obtain information on intrusions which have occurred to the respondent's networks. The questions asked were intentionally vague in an attempt to reduce the disclosure of specific network vulnerabilities, thereby increasing the number of survey responses.

Seventy-seven percent of the respondents reported that their networks had been attacked in the past. A further breakdown of that group indicated that sixty-nine percent reported the attack to a superior or other authority. The same group responded that seventy-two percent were utilizing a firewall mechanism at the time that the attack occurred. Among those respondents who were using a security mechanism when attacked, sixty-three percent reported that the mechanism had reduced the severity of the attack. This often consisted of a timely notification of the security administrator which allowed active defensive measures to be conducted before significant damage could occur to the system. Additional security measures were implemented after the attack by sixty-three percent of the group. Most of these measures consisted of improved security education and training of the users and the correction of well-known system flaws.

Overall the survey achieved the desired effect of providing information on the actual employment of security mechanisms in the field today. We believe that the results of the survey demonstrate that intrusion detection systems should be capable of identifying various types of threats, or be capable of being seamlessly incorporated with other security mechanisms which can defend against those threats not addressed by the intrusion detection system. In addition, while specific tools were not identified in the survey, there appears to exist a significant level of vulnerability remaining in those networks which utilize security mechanisms which are currently available.

5. FUTURE TRENDS IN INTRUSION DETECTION RESEARCH

5.1 Artificial Intelligence, Neural Networks and Machine Learning

The practical application of artificial intelligence techniques to the area of intrusion detection has been anticipated for several years. However, while expert systems have been widely incorporated into many intrusion detection systems, the effective application of AI has been elusive. Some of the difficulties in applying AI to intrusion detection were presented in [5].

However, there are tangible areas where AI techniques could be applied to intrusion detection methodologies in the future. In general, AI could provide significant benefits to intrusion detection through data reduction, the ability to analyze a collection of data to identify the most important components, and classification, the process of identifying intruders.

In particular, there are four areas where AI and machine learning could be applied to intrusion detection systems:

1. By using concept learning, the ability to train a system to classify elements into categories, the intrusion detection system would have enhanced capabilities to differentiate normal activities from intrusive.
2. Clustering, the partitioning of elements into groups based on a specified criteria, could be applied to the effective classification of users, groups, sessions, etc.
3. Predictive learning techniques applied to intrusion detection would allow the system to develop a temporal model of data and permit the system to learn of intrusive behavior from temporal data and sequences of individual events.

4. The ability to extract relevant features from irrelevant data and the possibility of combining relevant features into functions that identify intrusive events.

In addition to AI and machine learning, neural networks could provide a valuable addition to intrusion detection systems because of the flexible pattern recognition capabilities of the technology. The ability to adaptively model users and system behaviors, and the capability to effectively handle intrusive events are some of the potential advantages of neural networks. Most importantly, neural networks are particularly useful in identifying gradual changes to a system or in the behavior of a user. While expert systems are currently capable of recognizing rapid changes in a system, the identification of slower changes in behavior requires the employment of improved techniques.

AI, machine learning techniques, and neural networks, if properly refined and implemented, could result in the development of a comprehensive intrusion detection system. However, AI research has obstacles unrelated to intrusion detection, including system complexity and effective training techniques, which must be resolved before an effective intrusion detection application can be created.

5.2 Improved Software Development Techniques

While enhanced automated techniques offer the promise of improved detection abilities, a major source of vulnerabilities in computer systems and networks is the structure of the operating systems and applications on the system. Many of these vulnerabilities facilitate attacks on computer systems by reducing the amount of effort required by an intruder to gain access or further extend existing access privileges. The employment of structured software engineering techniques could eliminate numerous potential sources of insecurity. Three types of flaws could be addressed with improved development techniques:

1. Design flaws which result from inaccurate interpretation of software requirements and the improper analysis of the intended design of the system can lead to the inclusion of numerous unintended errors in the final application. The use of structured software validation and verification methods would reduce the number of these errors.
2. Faults within the application occur from the development of computer code which does not follow the defined specifications of the intended application. The complete eradication of all coding errors is extremely difficult, if not impossible, to achieve. However, the investment in the reduction of software faults offers the return of increasingly secure systems.
3. Operational and administrative flaws are those which result from the improper configuration of applications, operating systems, and security systems. These flaws are also difficult to eliminate completely, but the removal of configuration errors which are commonly known greatly enhances the security of the system.

Regardless of the intended effectiveness of future intrusion detection systems, the existence of unseen flaws can render the mechanism useless. As software development techniques improve, and as the configuration of finished systems become more straightforward, these types of vulnerabilities should be reduced in the future.

6. CONCLUSION

We have presented an overview of the technologies which are being utilized for the detection of attacks against computer systems, and a survey of the experiences of those most effected by intrusion detection

technology. We have also reviewed of some of the significant techniques which hold the promise of effectively protecting computer systems.

The security of information in computer-based systems and networks continues to be a major concern to researchers. The work in intrusion detection techniques and methodologies which has been a major focus of information security-related research in the past two decades is certain to continue. The area of intrusion detection is continuing to evolve. While a number of methodologies and tools have been designed to assist in the identification of intruders, no definable standard has been developed which could serve as the basis for a deployable intrusion detection tool. However, as the processing capabilities of computer systems improve and the innovative approaches to intrusion detection continue to be developed, the creation of an effective intrusion detection standard is inevitable.

BIBLIOGRAPHY

- [1] Anderson, D., Frivold, T. & Valdes, A. (May, 1995). Next-generation Intrusion Detection Expert System (NIDES): A Summary. SRI International Technical Report SRI-CSL-95-07.
- [2] Anderson, J.P. (April, 1980). Computer Security Threat Monitoring and Surveillance. Technical Report, J.P. Anderson Company, Fort Washington, Pennsylvania.
- [3] Castano, S., Fugini, M., Martella, G. & Samarati, P. (1995). Database Security. Addison-Wesley Publishing Company, New York.
- [4] Denning, Dorothy. (February, 1987). An Intrusion-Detection Model. IEEE Transactions on Software Engineering, Vol. SE-13, No. 2.
- [5] Frank, Jeremy. (1994). Artificial Intelligence and Intrusion Detection: Current and Future Directions. Proceedings of the 17th National Computer Security Conference.
- [6] Halme, L.R. & Bauer, R.K. (1995). AINT Misbehaving: A Taxonomy of Anti-Intrusion Techniques. Proceedings of the 18th National Information Systems Security Conference. Baltimore, MD.
- [7] Hubbard, B., Haley, T., McAuliffe, N., Schaefer, L., Kelem, N., Wolcott, D., Feiertag, R., & Schaefer, M. (September 11, 1990). Computer System Intrusion Detection: Final Technical Report. TIS Report #348, Trusted Information Systems, Glenwood, MD.
- [8] Ilgun, Koral. (1993). USTAT: A Real-time Intrusion Detection System for UNIX. Technical Report TRCS93-26, Computer Science Department, University of California, Santa Barbara.
- [9] Kim, G.H. & Spafford, E.H. (November 19, 1993). The Design and Implementation of Tripwire: A File System Integrity Checker. Purdue Technical Report CSD-TR-93-071.
- [10] Kim, G.H. & Spafford, E.H. (February 21, 1994). Experiences with Tripwire: Using Integrity Checkers for Intrusion Detection. Purdue Technical Report CSD-TR-94-012.
- [11] Levitt, K. (March 4, 1996). GrIDS-A Graph-Based Intrusion Detection System for Large Networks. Technical Report. University of California Davis.
- [12] Lunt, T.F. (1989). Real-Time Intrusion Detection. Proceedings from IEEE COMPCON.

- [13] Lunt, T.F. (1990). Automated Audit Trail Analysis and Intrusion Detection: A Survey. Proceedings of the 11th National Computer Security Conference.
- [14] McAuliffe, N., Wolcott, D., Schaefer, L., Kelem, N., Hubbard, B., & Haley, Theresa. (1990). Is Your Computer Being Misused?: A Survey of Current Intrusion Detection System Technology. Proceedings of the 14th National Computer Security Conference. Washington, D.C.
- [15] Mukherjee, B., Heberlein, L.T. & Levitt, K.N. (May/June, 1994). Network Intrusion Detection. IEEE Network. pp. 26-41.
- [16] Neumann, P.G. (1985). Audit Trail Analysis and Usage Collection and Processing. Technical Report Project 5910, SRI International.
- [17] Page, J., Heaney, J., Adkins, M. & Dolsen, G. (1989). Evaluation of Security Model Rule Bases. Technical Report. Planning Research Corporation.
- [18] Porras, P.A. & Kemmerer, R.A. (1992). Penetration State Transition Analysis: A Rule-Based Intrusion Detection Approach. Proceedings of the Eighth Annual Computer Security Applications Conference. San Antonio, TX.
- [19] Smaha, S.E. (1988). Haystack: An Intrusion Detection System. Proceedings of the 4th Aerospace Computer Security Applications Conference.
- [20] Sebring, M., Shellhouse, E., Hanna, M. & Whitehurst, R. (1988). Expert Systems in Intrusion Detection: A Case Study. Proceedings of the 11th National Computer Security Conference.
- [21] Shieh, S.W. & Gligor, V.D. (1991). A Pattern-Oriented Intrusion-Detection Model and Its Applications. IEEE Symposium on Security and Privacy.
- [22] Spafford, E.H. (1992). Common System Vulnerabilities. Proceedings of the Workshop on Future Directions in Computer Misuse and Anomaly Detection. University of California at Davis.
- [23] Staniford-Chen, Stuart. (May 7, 1995). Using Thumbprints to Trace Intruders. University of California, Davis.
- [24] Vaccaro, H.S. & Liepins, G.E. (1989). Detection of Anomalous Computer Session Activity. Proceedings of the IEEE Symposium on Security and Privacy.
- [25] White, G.B., Fisch, E.A. & Pooch, U.W. (January/February, 1996). Cooperating Security Managers: A Peer-Based Intrusion Detection System. IEEE Network.