CHAPTER 2

# A Historical Perspective

The purpose of this chapter is to increase your understanding of intrusion detection through historical insight. Understanding where the industry was 14 years ago will help you understand where we are today. Many of the early systems contained brilliant ideas and capabilities that are hard to find in today's commercial systems. Many lessons learned long ago are still applicable today.

In this chapter you will learn

- A brief timeline of the history of intrusion detection
- Relevant and interesting facts about the early systems
- A features comparison of early systems
- Historical lessons from the roots of intrusion detection

## A Timeline

The field of intrusion detection has exploded in recent years, but the roots of intrusion detection are considerably more humble. In the beginning intrusion detection research was focused on host-based event log analysis. Here is a brief timeline that highlights the host-based and research origins of intrusion detection through the emergence of network-based technologies and commercial companies.

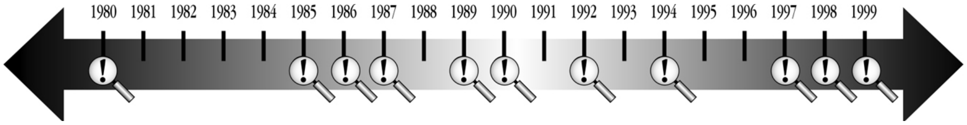


**Figure 2-1.** Intrusion Detection Timeline

1980 James P. Anderson writes in a technical report[1] for a classified customer that audit records could be used to identify computer misuse. He builds taxonomy of threat classification and offers suggestions to improve upon audit subsystems so that they can be used to detect misuse.

1985 SRI is funded by the U.S. Navy (SPAWAR) to build the initial prototype of Intrusion Detection Expert System (IDES). IDES[2] was one of the most influential systems in intrusion detection research. It was the first system to use both statistical and rule-based techniques in one application.

1986 Dorothy Denning publishes *An Intrusion-Detection Model.*[3] This paper is widely considered the seminal work on intrusion detection. It explores the basic mechanics of behavioral analysis and presents several possible methods to implement a system.

1987 First Annual Intrusion Detection Workshop at SRI. This workshop is formed by a handful of intrusion detection researchers who want to share information.

1989 Todd Heberlien, a student at University of California, Davis, writes the Network Security Monitor[4] (NSM) designed to capture TCP/IP packets and detect anomalous activity in a heterogeneous network. Network intrusion detection is born.

1990 The U.S. Navy completes a study of intrusion detection research projects for the purpose of choosing one to implement in a selected Navy enterprise. The report included 29 systems, of which five were examined in detail. The five systems were placed in a graph that showed 30 criteria (see Figure 2-1).

[1] Anderson, James P. *Computer Security Threat Monitoring and Surveillance.* James P. Anderson Co., 1980.

[2] Lunt, Teresa, et. al. "A Real-Time Intrusion Detection Expert System (IDES)." Computer Science Lab, SRI International, Menlo Park, CA, May 1990.

[3] Denning, Dorothy. "An Intrusion Detection Model." Proceedings of the 1986 IEEE Symposium on Security and Privacy (pp. 119–131), May 1986.

[4] Heberlein, L. Todd. "A Network Security Monitor." Proceedings of the 1990 IEEE Symposium on Research in Security and Privacy, Oakland, CA, May 1990.

1992 Computer Misuse Detection System (CMDS) is developed by Screen Application International Corporation (SAIC) and Stalker is developed by Haystack Labs. CMDS is based on work completed during the investigation of the Navy report. Stalker is based on the original Haystack work completed for the Air Force. These were the first commercially available host-based intrusion detection systems and are targeted at UNIX.

1994 A group of researchers at the Air Force Cryptological Support Center create a robust network intrusion detection system, ASIM, for wide deployment in the Air Force. The developers form a commercial company, Wheelgroup, to commercialize network intrusion detection technology.

1997 Cisco acquires Wheelgroup and begins a program to build network intrusion detection into Cisco routers. Internet Security Systems releases Realsecure, a widely distributed network intrusion detection system built for Windows NT. This is the start of the network intrusion detection revolution.

1998 Centrax Corporation releases eNTrax, a widely distributed host-based intrusion detection system built for Windows NT. Centrax was formed by the developers of CMDS and later joined by the technical team that built Stalker.

1999 Presidential Decision Directive 63 establishes a program of industry and government cooperation with a goal of increasing the use of intrusion detection to protect the national infrastructure. The Federal Intrusion Detection Network (FIDNet) is created to detect network infrastructure attacks against government sites.

## THE EARLY SYSTEMS

Technology has progressed and regressed to some degree in the last 15 years. Much of the information in this section is derived from the 1991 Navy study[5] of the state of the art discussed previously. Even as early as 1990 there were as many as 30 research systems. A selected list is as follows:

- Multics Intrusion Detection and Alerting System (MIDAS)—The original intrusion detection system used on the National Computer Security Center's public message system, Dockmaster. MIDAS was replaced on Dockmaster II by CMDS.

[5] Proctor, Paul E. Requirements Definition and Computer Misuse Detection Systems Analysis, U.S. Navy Technical Report SAI100505-133-1000, February 28, 1991.

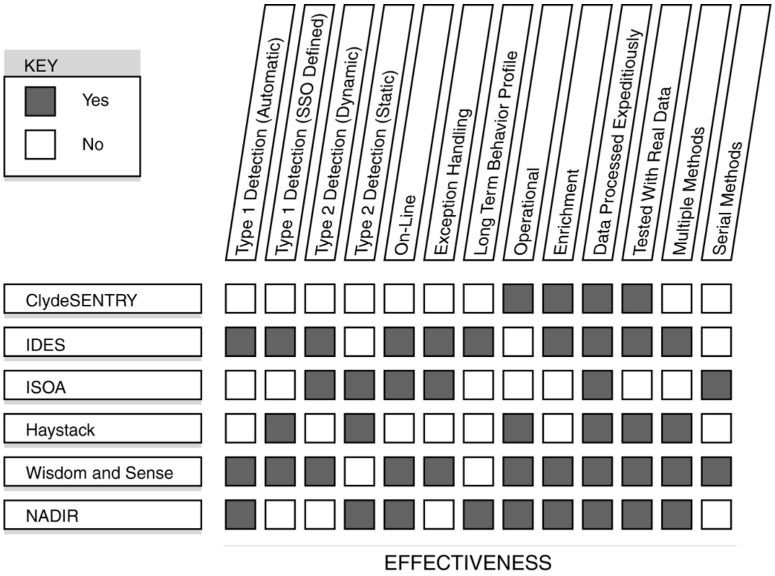
- Discovery—Used successfully for several years in the late 1980s to detect anomalous activity in a database maintained by TRW Credit Data Services.
- DRISC—Detect and Recover Intrusion Using System Criticality (DRISC) proved that early researchers were much better at developing technology than selecting names.
- Protocol Data Analysis Tool (PDAT)—PDAT was developed by Siemens AG in Germany to provide heterogeneous intrusion detection.
- Essence—A real-time, Lisp-based, forward chaining rule system prototyped within Digital Equipment Corporation for detecting suspicious activity in VMS.
- Harris Neural Network Prototype—The prototype targeted at VMS employed a Kohonen Self-Organizing Feature Map to measure deviations from normalcy using 11 statistical performance measures.
- Intrusion Detection Expert System (IDES)—IDES was a host-based intrusion detection system and research project. It was the first system to combine statistical behavioral analysis with rule-based signature analysis. There were three versions of IDES originally. The basic IDES system was targeted toward the TOPS-20 operating system. Sun IDES processed data from SunOS. A special version of IDES was built for the FBI to process data from the MVS operating system. The IDES work later became NIDES (Next Intrusion Detection Expert System). Today, the NIDES work is the basis for Emerald.
- Information Security Officers Assistant (ISOA)—ISOA was a host-based intrusion detection system developed at Planning Research Corporation (PRC) in McLean, Virginia with government-supported internal research and development (IR&D) funding. It combined a set of statistical tools, an expert system, and a hierarchical set of “concern levels.” The technology was based on an Indications and Warnings (I&W) model derived from surveillance applications intended to provide advance warning of imminent attack. Incoming audit data were compared to a set of expected indications and arranged hierarchically to reflect growing levels of concern. Abnormalities were detected using profiles in three categories: users, nodes, and the full system. The ISOA work later was used in the PRC intrusion detection system PReCis.
- Wisdom & Sense (W&S)—W&S was a host-based anomaly detection system that was first developed in 1984 by Hank Vaccaro at Los Alamos National Labs (LANL) for the National Computer Security Center (NCSC) and the Department of Energy (DOE). W&S processed a dataset (training data) and generated metarules that described the characteristics of the data. Then, when presented with a new dataset, it applied these metarules to detect anomalies. W&S technology was originally designed to detect anomalies in storage records for nuclear material. It

was modified to work with VMS operating system audit records and applied to detecting deviations in human behavior. The results were mixed and W&S was never used operationally in a production environment. The difficulty was in understanding how the metarule trees were constructed and pruned, making it difficult to interpret the results.

- Haystack—Haystack was developed at Los Alamos National Laboratory by Tracor Applied Sciences and Haystack Laboratories with funding from the Air Force Cryptological Support Center. Haystack was designed to assist security officers in detecting and investigating misuse. It performed two types of statistical analysis. The first yielded a set of suspicious quotients (a measure of anomalousness with respect to a weighting of selected measures for a session). The measures reflected the degree to which a session resembled a predefined intrusion profile. The second statistical analysis measured significant changes or trends in recent sessions compared to previous sessions. Haystack generated a summary report of system usage statistics, new users, security events, and user sessions that resembled intrusion profiles. Haystack ran on a 286 PC and processed event log data from the OS/1100 operating system. Haystack Labs, located in Austin, Texas, went on to develop one of the first commercial intrusion detection systems, Stalker, before being purchased by Trusted Information Systems (TIS). After Network Associates acquired TIS, the doors were closed on the Austin office. The Haystack technical team joined Centrax Corporation to help develop the eNTrax product in 1998.
- Network Security Monitor (NSM). An NSM prototype developed by the University of California Davis (UCD) and currently running on a Sun 3/50. NSM was designed to analyze data from an Ethernet local area network (LAN) and the connected to it. NSM was a research system, and UCD had hoped to expand its scope to include real environments, real attacks, and perhaps wide area networks.

## EARLY CAPABILITIES COMPARISON

The graphs presented in Figures 2-2 and 2-3 measure six of the early systems against criteria derived from the three requirement groupings: effectiveness, interface, and adaptability. The purpose of the graph was to describe functionality; therefore, a shaded box reflects the presence of the feature. This was originally published in 1990, and you may recognize the format from the Trusted Computer Security Evaluation Criteria (TCSEC), also known as the Orange Book. This was done on purpose to give the reader something familiar at the time. Some of the terminology has been modified from the original version to reflect a richer present-day vocabulary.

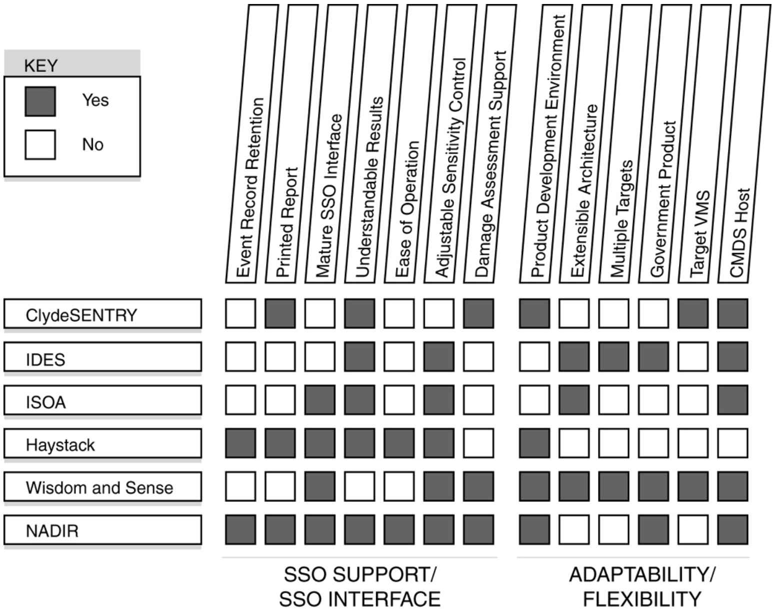


KEY
■ Yes
□ No

| | Type 1 Detection (Automatic) | Type 1 Detection (SSO Defined) | Type 2 Detection (Dynamic) | Type 2 Detection (Static) | On-Line | Exception Handling | Long Term Behavior Profile | Operational | Enrichment | Data Processed Expeditiously | Tested With Real Data | Multiple Methods | Serial Methods |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ClydeSENTRY | □ | □ | □ | □ | □ | □ | □ | ■ | ■ | ■ | ■ | □ | □ |
| IDES | ■ | ■ | ■ | □ | ■ | ■ | ■ | □ | ■ | ■ | ■ | ■ | □ |
| ISOA | □ | □ | ■ | ■ | ■ | ■ | □ | □ | □ | ■ | □ | □ | ■ |
| Haystack | □ | ■ | □ | ■ | □ | □ | □ | ■ | □ | ■ | ■ | ■ | □ |
| Wisdom and Sense | ■ | ■ | ■ | □ | ■ | ■ | □ | ■ | ■ | ■ | ■ | ■ | ■ |
| NADIR | ■ | □ | □ | ■ | ■ | □ | ■ | ■ | ■ | ■ | ■ | ■ | □ |

EFFECTIVENESS

**Figure 2-2.** Early Systems Capability Comparison



KEY
■ Yes
□ No

| | SSO SUPPORT/ SSO INTERFACE | | | | | | | ADAPTABILITY/ FLEXIBILITY | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Event Record Retention | Printed Report | Mature SSO Interface | Understandable Results | Ease of Operation | Adjustable Sensitivity Control | Damage Assessment Support | Product Development Environment | Extensible Architecture | Multiple Targets | Government Product | Target VMS | CMDS Host |
| ClydeSENTRY | □ | ■ | □ | ■ | □ | □ | ■ | ■ | □ | □ | □ | ■ | ■ |
| IDES | □ | □ | □ | ■ | □ | ■ | □ | □ | ■ | ■ | ■ | □ | ■ |
| ISOA | □ | □ | ■ | ■ | □ | ■ | □ | □ | ■ | □ | □ | □ | ■ |
| Haystack | ■ | ■ | ■ | ■ | ■ | ■ | □ | ■ | □ | □ | □ | □ | □ |
| Wisdom and Sense | □ | □ | ■ | □ | □ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| NADIR | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | □ | □ | ■ | □ | ■ |

**Figure 2-3.** Early Systems Capability Comparison (Continued)

### Effectiveness

The effectiveness group includes evaluation factors that focus on the technical ability of the intrusion detection system to identify computer misuse. The factors are not of equal importance but instead try to make meaningful, technical discriminations among the systems.

**Self-configuring anomaly detection—**Acceptable behavior is defined by historical data. New data are then compared to the profile of acceptable behavior and anomalies are reported.

**Statically defined anomaly detection—**The security officer can define the profile of acceptable behavior.

**User-definable rule-based detection—**The security officer can specify patterns of misuse detect.

**Vendor-provided rule-based—**The system is delivered with a predefined database of signatures. This was not a common practice in the beginning. Now it is standard practice.

**Real-time—**The system processes data as they are created on the target system. Remember that all of these systems were host based.

**Parameter tuning—**Selected parameters may be tuned to reduce false-positives.

**Long-term behavior profile—**Acceptable behavior profiles can reflect arbitrarily long user activity periods. This measure reflects the system's ability to minimize the effect of short-term aberrations in normal user behavior.

**Operational use—**The system has been used operationally to catch misuse. This is critical to determining the viability of a system.

**Performance—**Can the system process 24 hours worth of data in 24 hours? The numbers were much smaller back in the late 1980s but the concept remains the same. This measure is closely related to scalability in today's commercial systems.

**Hybrid without correlation—**The system has multiple detection methods with no data correlation in the detection mechanism. Examples in the late 1980s would include anomaly detection and host-based detection. A good example today would be a host-based and network-based system that has both capabilities but provides no correlation in the detection mechanisms.

**Hybrid with correlation**—The system has multiple detection methods that feed a single detection mechanism. An example in the late 1980s would be a rule mechanism that feeds a statistical model. Today's example would be compound signatures that include both host and network data.

### SSO Support/SSO Interface

This collection of seven factors addresses the ability of an intrusion detection system to support the system security officer (SSO) (i.e., the operator of the system). These features reflect the system's ability to detect misuse effectively without adding significantly to the SSO's workload.

**Raw data retention—**The system stores the raw data for later review and evidentiary purposes.

**Printed reports—**Many of the early systems could not print reports from the user interface.

**Mature interface—**The definition of a mature interface has certainly changed in the last 10 years, but the concept remains constant.

**Understandable results—**This category was mostly subjective but was intended to address poor presentation. Surprisingly, this is one of the areas where the early systems excelled in comparison to today's commercial systems.

**Ease of operation—**The early systems suffered badly from lack of interface design—if they had an interface at all. This measure is still critical in today's systems, as the bar has risen considerably with the advances in interface and the lowering of the operator's expertise with broader deployment.

**Adjustable sensitivity control—**This reflects a system's ability to filter noise. This is not the same as false-positive reduction, which reflects the ability to adjust detection parameters. This is just the removing of alarms from the interface.

**Damage assessment—**The ability to assess the degree of compromise and investigate the methods used and event leading to an attack.

### Adaptability/Flexibility

These four measures reflect an intrusion detection system's ability to be targeted at new environments and new data sources.

**Product development environment—**The system is developed in a structured environment for maintainability and support. This includes features that are taken for granted today, such as a revision control system, design documentation, and end-user documentation.

**Extensible architecture—**The system is designed in such a way that new target sources and detection mechanisms may be added without a complete rewrite of the underlying code.

**Multiple targets—**The system has a heterogeneous architecture and supports multiple data sources.

**Government product—**The system is owned and funded by the government. This is usually indicative of research systems, a lack of maintainability, and support. However, if you are affiliated with the government you can get the system for free.

**Target VMS—**The system can process VMS operating system data. This was a very project-specific requirement.

**CMDS host—**The command console runs on a Sun Solaris system. This was the desired intrusion detection console for the sponsor of this work.

## HISTORICAL LESSONS

The early systems showed three fundamental flaws. First, they were not able to process data from systems other than the original targets to which they had been designed. Second, they were not able to analyze data from different target environments than they were designed toward. Third, the user interfaces were terrible. None of these facts should be a surprise because early researchers were funded with specific goals and commercialization was not high on the list. Unfortunately, all three characteristics needed to be addressed satisfactorily if intrusion detection was ever going to transition out of the lab and into operational environments.

Several systems were designed to address these flaws. The first two to be commercialized were the Computer Misuse Detection System (CMDS), built by Science Applications International Corporation (SAIC), and Stalker, built by Haystack Labs. CMDS used the CLIPS expert system, which included a full forward chaining inference engine for rule-based detection and a very simple statistical detection model for anomaly detection. Stalker provided an interface for processing and interpreting raw operating system logs.

In the early years (1984–1992), the most significant challenge to the adoption of intrusion detection was the mind-set in the end-user community that monitoring was worthless. Unfortunately, this was particularly true among the end users with money to spend on security. The prevailing wisdom was that if you had a dollar to spend, it was spent on protection. The Information Technology Security Evaluation Criteria (ITSEC), the Orange Book, was the standard for computer security and had little mention of traditional auditing and no mention of intrusion detection. These were the early days of computer security, and intrusion detection was just an interesting research area.

Intrusion detection is not an exact science. Human behavior is not stable, and automated mechanisms for analyzing it are severely challenged. Even the deterministic nature of rule-based mechanisms does not remove the need for a human element (the person in the middle) for an effective intrusion detection operation. In that regard, intrusion detection systems can best be thought of as decision support systems rather than the cybercops who identify misuse while you sleep, as the popular press would have you believe. In other words, an intrusion detection system is going to help you think rather than do the thinking for you. This revelation does not remotely decrease their value. I have watched many security officers increase their value to an organization overnight by helping to identify back-door accounts, malicious users, and network attacks.

### SCIENCE OR ART?

At one of the early intrusion detection (ID) workshops at SRI (which I attended), there was a particularly vehement discussion concerning the reduction of false-positives in multivariant statistical analysis of heterogeneous audit data. This is a very technical way of saying that we were exploring ways to reduce the amount of noise generated by intrusion detection systems. It was at that moment that I realized how effective intrusion detection could be achieved so I spoke up. "Interpreting behavior with the quality of available data is never going to be a precise science. If that's true then intrusion detection is more of an art than a science and intrusion detection tools are really best used as decision support systems as opposed to definitive measuring devices." I'm pretty sure I heard a cricket in the long silence that followed. It was also at this moment that I realized researchers don't like their projects compared to art.

## SUMMARY

Intrusion detection has existed as a research area since about 1986. In the beginning almost all intrusion detection systems were host based. Early systems had lousy user interfaces; they were unable to be used in environments outside those for which they were designed and could monitor a very small number of targets. The end-user community focused on prevention to the exclusion of detection and response up until about 1996.

One of the most significant conclusions that can be drawn from the history of intrusion detection systems is that they may operate better in a decision support context as opposed to being a misuse cop. This is more true for host-based systems than network systems. In Chapter 3 we'll examine the technology behind network-based systems. In Chapter 4 we'll examine host-based systems.