JISSec 5(2) 2009

Journal of
Information System
Security

www.jissec.org

Supporting intrusion detection work practice

John R. Goodall

Secure Decisions division of Applied Visions,
6 Bayview Ave, Northport, USA

Wayne G. Lutters
Anita Komlodi

Department of Information Systems
University of Maryland, Baltimore County
Baltimore,USA

Abstract

In an increasingly networked world, information security is an increasingly important domain, but one that is not well understood. Yet, an understanding of how this work is accomplished is crucial to designing tools and management policies to better support it. The work practice of intrusion detection analysts is a complex fusion of individual and collaborative resource monitoring and problem solving. This paper details the practice of intrusion detection work, specifically highlighting the tasks that make up the work, and it concludes with a discussion of the implications that this work understanding has on future design of tools and organizational policies to make intrusion detection work more efficient

Keywords : Work practice, intrusion detection, computer network defense, task analysis, collaboration.

1. Introduction

In conjunction with a growing dependence on computing network infrastructure, the frequency and severity of network-based attacks have dramatically increased (Allen et al., 1999). Simultaneously, there has been an inverse relationship between the decreasing expertise required to execute attacks and the increasing sophistication of those attacks - less skill is required to do more damage (McHugh, 2001). Despite advances by researchers and developers on preventative security measures, vulnerabilities remain. These vulnerabilities are due to programming errors, design flaws in foundational protocols, and the problem of legitimate users misusing their privileges (Lee et al., 2000). In this increasingly hostile environment and in conjunction with proactive, preventative security technologies, organizations also typically employ reactive measures such as intrusion detection (ID), the monitoring of network activity for signs of malicious or abnormal activity. In the words of the participants in this study, intrusion detection systems (IDS) automatically detect "intrusions and behavioral misuse" (Participant 8; quotes from participants are labeled P# in the text) by matching patterns of known attacks, called signatures, against ongoing network activity to produce security alerts detailing those events. They provide security analysts with "awareness and control" (P1) over the operating environment and give "some indication if you are vulnerable" (P8). However, IDSs alone are not entirely effective. A comprehensive survey found that while 91% of organizations employed an IDS, only 50% of the respondents described their IDS as being effective (E-Crime Watch survey, 2005). The sheer number of IDS alerts can be overwhelming; some participants reported having thousands of alerts a day. Because of the potential for false positives or negatives and the potential for self-damaging responses to inaccurate alerts, fully automated IDSs, often called Intrusion Prevention Systems, are rarely a completely effective solution, instead requiring vigilant oversight by human security analysts. This paper is about those human experts - their work domain, their routine challenges, and their craft - and how understanding these elements can influence better designs and policies to support their work

Motivation

The global dependence on a robust and secure computing infrastructure prompts security analysts to attempt to detect threats against their cyber assets and data. As computer processing and storage costs rapidly decrease, more systems are brought online and more data is stored online in data warehouses. These increases make the defenders' jobs more difficult. The number of defenders is not increasing, but the number of cyber assets they protect is.

The tools with which these defenders accomplish this challenging job are quite limited. They are refined in some dimensions (e.g., scheduling cron jobs for monitoring purposes), impoverished in others (e.g., analyzing the large data sets that result), and ignored still others (e.g., collaboration and communication about analysis results). The tools used by defenders typically exhibit a poor task-tool fit for many tasks, as analysts are forced to use tools that were designed without a full awareness of the tasks the tools were meant to support. Designing effective tools to support the analysis of large, multi-dimensional data sets and to support collaborative systems in complex information environments is difficult and such tools are rarely successful in their initial versions. Socio-technical solutions are required: system designers must understand both sides, their interaction, and then allow them to co-evolve after implementation to reach sustainable equilibrium. A deep understanding of the people, the work, and the environment and their interactions will vastly improve the quality of initial system design and subsequent iterations.

One of the richest means of attaining this understanding involves using ethnographically informed field research methods that have emerged from various disciplines, including sociology, social anthropology, cognitive science, and computer science which have become central to the interdisciplinary fields of computer-supported cooperative work (CSCW) and human-computer interaction (HCI). This tradition of research seeks to develop technology "which takes the social and situated seriously, and which drives analytic attention towards the ways people use technologies to accomplish and coordinate their day-to-day practical activities" (Luff et al., 2000). Using ethnographically inspired methods to inform the design of technological tools has been successfully employed in high-reliability and information-rich domains; that is, workplaces where the consequences for work errors are severe and information needs are demanding. These have strong work practice similarities to ID work, including processes for monitoring of system activities and responding to abnormal events. Some examples include air traffic control (e.g., Bentley et al., 1992; Hughes et al., 1992), subway transportation control rooms (e.g., Heath & Luff, 1992), and aerospace service engineering (e.g., Lutters & Ackerman, 2002). This research approach has yielded a foundational understanding of the work practice in high-reliability organizations and has led to conceptual designs for new socio-technical systems to support tasks within those organizations more effectively. The need to understand how people use technology to accomplish their work formed the basis for the research described in this paper.

Related Research

While there are few research papers examining the work practice of security analysts, the following are noteworthy as a starting place for this research.

Yurcik and colleagues (2003) results were derived from the authors' own experiences performing security. This work framed the central challenge in information security - the asymmetry between attackers and defenders, which gives the former the advantage in this escalating battle. This asymmetry is categorized as:

- The Internet provides connectivity for users to access information from anywhere in the world, but also allows attackers this same worldwide access;
- Administrators must continuously identify and repair every vulnerability, while an attacker need only find a single vulnerability to exploit;
- Administrators are dependent on the security of all of the systems on their perimeter - one compromised system affects all of its peers;
- Administrators must protect all systems, while attackers can focus on only one;
- Attackers have the element of surprise since they can develop new exploits at any time.

These challenges highlight the advantage attackers have and the difficulties the defenders of computer networks face. Their work, as with the research presented in this paper, points out the irreplaceable importance of humans in providing network security.

Researchers at IBM investigated network operators' problem-solving tasks in the 24/7 monitoring of multiple customers' networks in a security operations center, or SOC (Stolze et al., 2003a; Stolze et al., 2003b). The goal in this type of managed security environment is to identify and report to the customer any anomalous network activity from data collected by sensors on the customer's network. Their operators related three core challenges: problem solving, learning, and cooperation. They focused on the problem-solving task in detail, and presented a descriptive model tailored specifically to the classification process of new security events, which occurs over multiple stages

- New event triage, in which operators determine whether the alert is obviously part of a sequence of events that the customer needs to be notified of, whether the alert requires further analysis, or whether the alert is obviously a false alarm;
- Strange event analysis, in which suspicious events are examined in more detail;
- Pattern assessment, in which operators keep track of open patterns that require further events before making a decision;

- Alert management, in which analysts must determine how to react, occurs if the operator needs to contact the customer;
- False positive management, in which analysts must determine if and how to modify the infrastructure, occurs if the event is determined to be a false positive.

A managed SOC is a unique environment different from the work done locally by security analysts defending their own network. While there are some overlaps between the description of the work in a SOC with our own results, the dedicated environment of a SOC is markedly different from the environments in which our analysts worked.

D'Amico and colleagues performed a cognitive task analysis of analysts from one commercial and six Department of Defense organizations responsible for network security (D'Amico et al., 2005). Through interviews and observations, they identified six roles describing the functions of the analysts: triage, escalation, correlation, threat, incidence response, and forensics. The roles described in that research overlap with the ones described here; their research differs primarily in that they were looking at the entire process of information assurance (outside of government, often referred to as information security), whereas in our research we detail primarily the reactive tasks associated with intrusion detection. So, for example, we did not examine threat analysis, which is primarily a proactive, rather than reactive task. Additionally, we focus on a more diverse sampling of analysts and examine not just the individual cognitive tasks, but the social and learning processes as well. This research confirmed an important finding from our prior work - the crucial nature of situated knowledge, the importance of understanding what is "normal" within a given environment (Goodall et al., 2004).

Methodology

Our study design involved the following data collection methods, with each method complementing the others by providing multiple perspectives on the same phenomena. Specifically, our data collection involved: individual contextual, semi-structured interviews, focus group interviews, mailing lists analysis, and a confirmatory survey. In addition to these data collection methods, we examined the tools used by security analysts to understand how those tools worked and put interviewee explanations into context. This also led us to inspect the various security-related resources, particularly on the web, common to the analysts we interviewed. These tools and resources were vital to analysts in accomplishing their work and, while not the focus of our analysis for this project, allowed our analysis of interview data to be put in the context of the actual work.

We interviewed information security experts individually to unpack the mundane and exceptional processes of their ID-related activities. Interview participants included a diverse cross-section of ID experts, some working as stand-alone analysts, some as members of teams. All participants possessed a working knowledge of at least one IDS, with a common reference point of Snort, an open-source, signature-based network IDS (Roesch, 1999). Interviews were semi-structured, following a prepared interview guide though allowing off-topic elaboration. Interviews were conducted in situ when possible, encouraging participants to demonstrate their interactions with their IDSs, support tools, and commonly used resources. We completed eleven interviews, which were recorded and transcribed.

We attended two ID user group meetings and interviewed the attendees as a group to gain an understanding of the face-to-face interactions among a community that is typically tied together by electronic connections. The focus group interviews were unstructured, not following a prepared guide. Six analysts participated in each of these, and each session lasted about two hours. These interviews took place early in the research and helped to orient the researchers to the concepts, terminology and topics of interest to the community. In addition, the focus groups allowed the researchers to witness the interpersonal communications between analysts.

Data analysis was informed by Grounded Theory (Strauss & Corbin, 1998). The transcriptions and interviewer notes were coded for similar themes to form chain of evidence for emergent relationships. No hypotheses were formed prior to coding the transcriptions, although the first author built up some domain knowledge in order to be able to speak the language of the interviewees. The first author did several iterations of open coding. These codes were then grouped into higher-level concepts, which were reviewed by the other authors for accuracy and consistency. Once these higher-level concepts had emerged, the first author made additional selective coding iterations through the data. Interviews and coding were done in parallel, and we stopped searching for new interviewees at the point of saturation - when the interviews yielded no new concepts that had not already emerged from the data. The concepts that emerged from the interviews were used to selectively code the results from the mailing list analysis and formed the basis for the survey questions.

In addition to these face-to-face interviews and focus groups, we monitored an intrusion detection email mailing list for a period of one year. During the one-year period, the list had 1,178 messages, averaging about 98 message posts per month. We also analyzed posts from various other security-related mailing lists for shorter periods to provide additional context. These electronic conversation threads were used to understand the flow of information within the community across organizational boundaries. Mailing list posts were examined and analyzed in bulk in order to gain a comprehensive picture of the threaded postings. While all posts were examined, the posts with the greatest number of replies were more closely analyzed. These were coded using the themes that were raised from the interviews. Posts were categorized into bins of each of the major concepts relating to the analysts' tasks and workflow, collaboration and communication, and expertise and knowledge development and transfer.

We also administered an online survey to confirm the results of the field-study with a broader audience; 54 security analysts responded. The survey provided corroboration for the understanding of work practice derived from the field-study. It was designed to be brief (15-20 minutes) to encourage response from a population with little spare time. The content of the survey aimed to: confirm the core tasks that emerged from the field-study; rank the respondents' value and trust in the tools, data sources, and other resources used in each of the task stages; identify the organizational roles of collaborators; and confirm mechanisms of external collaboration. While some new information was gleaned from the survey results, the primary motivation was to expand the scope of the results to a broader population of security analysts.

Participants

The participants in both the interviews and the survey had wide-ranging levels of ID experience, primary job duties, and organizational security needs. It was important to include a diversity of ID experts to ensure that our results were not limited to any single type of user or organization. While there were subtle differences and despite the diversity of the analysts interviewed, the work practices they engaged in were strikingly similar.

The interview participants' organizations and corresponding security practices ranged from the relatively open environments of university settings to highly secure defense contractors and financial service companies. The primary roles of the participants varied: most were network or systems administrators whose duties included ID, only a few were dedicated information security analysts, and two were IDS developers who provided support for other IT departments in their organization.

In order to confirm the results from the interviews, we designed a survey based on those results and administered it online. The participants in the survey followed a roughly similar demographic distribution as the interviews. The survey was designed after the interviews had been analyzed, and the close-ended questions were derived from this analysis. The following summarizes the demographic information collected from the survey, and compares those results to the demographics of the interviewees. Of those surveyed, 50% reported their primary job role as that of a security analyst (compared to 42% of interviewees), 24% as network administrators (17% of interviewees), 13% as site security officers (17% of interviewees), and 7% as systems administrators (25% of interviewees). One of the mailing lists the survey went out to was specific to a networking tool, which could indicate why the network administrator population was higher and system administrator population was lower in the survey.

A third of the survey respondents described their organization's primary industry as information and telecommunications, followed by government (14%), education/research (12%), banking and finance/insurance (12%), and military/defense industrial base (10%). This distribution is similar to that of the interview population, except for the large representation of the information and telecommunications industry; this under-representation in the interview stems from the researchers' inability to successfully solicit interviewees from this population. The majority of respondents (65%) reported the size of their entire organization to be greater than 500 people, which is similar to the size of organization among the interviewee population (75% reported organizations of similar size).

It is important to note that "most people aren't just analysts" (P3). Interacting with the IDS is just one part of a job that includes other systems, network, or security related tasks. This is particularly true in smaller companies where a dedicated security person is not likely to be cost effective because the organization does not believe their threat level to be high enough or the organizational security needs are limited. For example:

> That's how I describe myself now, more of a systems administrator who does security work, because my company isn't big enough to have a security person full time. Even with a hundred employees, I think I could spend my entire day, every day of the year doing security stuff, making things better than they are now, but from the company's point of view, they don't need that. (P9)

This analyst works primarily as the systems administrator for several dozen machines, but is also responsible for all of the organization's information security needs. This is typical of analysts being pulled in multiple directions by their organizational responsibilities, and security is often sacrificed for more visible organizational needs. Security in general, but ID in particular, is rarely noticed by management unless an attack is successful or a system is compromised - that is, when security fails and the notification is too late. As such, ID work is largely invisible (Star & Strauss, 1999). The analysts had the sense that their supervisors did not recognize the majority of their work - especially the mundane, routine work - until there was a crisis. It is difficult to measure the productivity of a security analyst, as much of the work involves the unseen ongoing processes of maintaining awareness. There is never enough time to keep updated of the latest security developments, improve existing security infrastructure, or comb through the various data sources looking for potential vulnerabilities or attacks. Despite the amount of work to be done, most analysts felt that security was not a priority for their organization. (The exception to this includes those organizations or analysts whose mission or role is security, such as that of a managed security service provider.)

A day in the life

As an introduction to the world of intrusion detection analysts, we present here a composite scenario of a "typical" day in the life of a security analyst from our study. This contextualizes the major work practice themes that are discussed in the remainder of this paper.

Marcus arrives at work early Tuesday morning and immediately begins sifting through the 38 IDS alerts that have piled up overnight. He quickly discards several that he recognizes as not pertinent to his environment - web server attacks against a host he knows does not have a web server and a Windows specific attack against a Linux server - and categorizes the remaining alerts into two groups. The first group includes two potentially severe alerts, which he will look into first. One alert in this first group is targeted against the company's email server, one of the most crucial systems to the day-to-day activity of the company, which also stores sensitive information. The other is labeled as 'severe' by the IDS and originated overseas. The rest of the alerts, targeted against client workstations or alerts that appear to be fairly innocuous based on his previous experience, he will look into later when he has more time.

He first deals with the more recent of the two alerts, from only a few minutes before he arrived. The alert describes an attack against his company's email server, which Marcus helped configure. He remembers that he installed firewall software to block certain ports, but realizes that the attack in this alert is directed at a port that is open to the world so that remote users on the road can connect through the firewall. He searches online security mailing list archives to learn about this type of attack. He finds information almost immediately and realizes that this is not a new attack and that there is already a patch available. Inexplicably, the patch had not been applied to this system.

Deciding that this alert could be potentially very dangerous, he sets up a spanning port on the network switch to begin collecting the network packets that are currently going to and coming from the email server. After looking at the packet traffic for twenty minutes, he comes to understand that the email server is sending out FTP packets to a machine he has never seen before. It is not the same machine as that described in the alert, but it could be another machine owned by the attacker. While incoming FTP traffic would have been flagged since FTP is an inherently insecure protocol, outgoing FTP traffic is generally allowed by the firewall rules, since employees often need to download files from clients' FTP servers. This could mean that the attacker is copying data from the email server to a remote system.

He calls up Dave, who runs the company firewall, on his cell phone. He explains the situation to the half-awake Dave, describing the alert and the network traffic that was continuing to fly between the attacker and the email server. He asks Dave to block the attacker at the border firewall. He knows this is not a permanent solution; the attacker is probably spoofing his source Internet address and can probably resume the attack from a different address in a few minutes, but it will buy him some time. Dave agrees and remotely logs into the firewall to reconfigure the rules. Within minutes the traffic ceases to show up in Marcus's packet capture logs.

Marcus logs into the email server and begins to quickly scan the log files, which show signs of tampering. The attacker was not very clever about covering his tracks; an entire two-hour period has been erased from the log. Under normal circumstances, there would be at least some activity logged. However, all activity is also logged to a remote log host, but the server was recently reconfigured to cease doing so. The attacker was a bit clever after all. At this point, Marcus realizes that the server was definitely compromised and will need to be examined to see what data may have been lost before reinstalling the system and restoring the data from backup tapes. He calls up Jane, the network operator on duty, and asks her to unplug the server's network cable. He quickly patches the backup server, wondering why no one had yet applied this patch. With the backup server now online, he sends an email to the entire organization saying that the backup email server will be in use, and to expect email to be slower than normal. He then heads to the computer room to assess the damage on the now offline email server, a tedious process that could take several days. However, with the backup server up and patched against this type of attack, he knows he can take some time.

Marcus doesn't get to the second important alert until that afternoon. It is a type of alert he has not seen before, related to an operating system vulnerability on an FTP server. The alert is also abnormal since the source machine is from overseas. None of the company's clients are overseas, the IP address of the server is not publicly known. The public should not be accessing this FTP server at all. He searches through his email archives looking for anything related to this kind of attack; he rarely has time to look at all of the emails that pour in from the mailing lists reporting on new bugs, security vulnerabilities, and updates of the various operating systems and applications within his environment. He doesn't find anything obvious in his search, so he starts searching the community forums on a web site for the IDS he uses. He eventually finds a description of the alert and a pointer to a patch for the vulnerability. He knows the administrator for this host had patched something recently, but is not sure if this is it. He sends an instant message to the system administrator of this host and asks. The administrator replies that this was indeed the patch that had recently been applied. Marcus decides to modify the IDS signature to reduce the severity should this particular signature be triggered again. He wants to know when this type of event happens, but does not want to have to waste time investigating it again.

The remaining dozen alerts from the night before and the five that came in while he was examining the hard drive on the email server, he quickly realizes, were port scans looking for vulnerabilities in his network. None of these would have been successful as they were all pretty naïve scans, not even randomized to avoid easy detection. These wannabe hackers are simply annoyances most of the time; not since he started has Marcus seen one actually get lucky. It is the sophisticated ones who took their time and covered their tracks, like the one this morning, that worry him. He makes a mental note of the source Internet addresses in case they show up again, knowing that he probably won't remember them, but that it isn't likely to matter. These kids rarely bother to learn enough about security to be more than a minor disturbance. With the previous night's alerts finally out of the way, he has time to continue analyze the damage on the email server before returning to set up the new intranet web server that he had started last week.

This scenario is a composite of actual events and activities described by analysts during our interviews. The actual events described did not all occur on the same day to the same analyst, but this presents a representative example of a typical - if any day in the information security world can be described as *typical* - day in the life of a security analyst. The themes raised in this overview will be explored in the remainder of the paper.

Work Practice

Providing security requires an integration of tasks that include detecting intrusions, choosing preventative technologies for "hardening" systems, implementing encryption and authentication schemes, and educating users in safety-smart work practice. The work of ID itself involves more than reviewing IDS alerts and occasionally responding to critical events; it cannot be accomplished effectively in isolation, but requires that systems tangential to the IDS be monitored and analyzed, while keeping abreast of the latest security-related information. All of our participants followed a similar ID workflow, which we analytically abstracted into four main phases: *monitoring, triage, analysis, and response.* There are not always clear demarcations between these tasks, and there is often overlap among them. Analysts move fluidly from one to the other with the results of one task phase feeding into others. In some cases, individual analysts perform one or more tasks and must then communicate the results to each other. Aligning the tasks into the four phases mentioned above demonstrates how ID is accomplished and how analysts describe their work, from an initial awareness through their reaction to an event.

The term event is used here to permit a common framework between seemingly disparate threads and to provide a starting point for a workflow that is inherently event-driven. An event could be, most obviously, an alert generated by an IDS. It is also used to describe other occurrences, such as the announcement of a new vulnerability, a phone call from an end-user, an email describing a new attack method, or a sluggish network speed. All of these situations trigger the ID processes described here. Usually, then, a discussion of the work of ID focuses on the trajectory from the discovery of an initial trigger event through its triage, analysis, and response. However, ID work cannot be accomplished without the awareness necessary to monitoring. The monitoring task must thus foster awareness and facilitate discovery. The subsequent tasks are initiated by the trigger event, but could not be accomplished without proper awareness during monitoring.

This ID trajectory is epitomized by a survey response to *Please describe how you detected and responded to a memorable attack*: "[I] saw the alert, reviewed the packets, captured more, audited the involved hosts, locked down the appropriate network resources until the situation was resolved." The process reported by this respondent aligns itself into the analytical construct used here as: monitoring ("saw the alert"), triage ("reviewed the packets"), analysis ("captured more, audited the involved hosts"), and response ("locked down the appropriate network resources until the situation was resolved"). Each of these tasks will be unpacked in turn, focusing on the two dimensions of resources that are leveraged in each. These two dimensions are organizationally *internal or external* resources.

Internal resources include systems within the analyst's network environment: the network infrastructure, one or more IDSs, firewalls and other network devices, and logging and monitoring systems. These internal resources collectively comprise an analyst's *environment*. External resources include mailing lists, web sites, and user groups. These external resources are within an analyst's *community*. While ID work focuses primarily, of course, on the resources within an organization (i.e., the goal of intrusion detection is just that - to detect intrusions in your environment), a holistic view of the work is necessary for two reasons. First, it is impossible to understand work practice in isolation without looking at the broader context in which the work takes place. Second, and more importantly, this is how the analysts themselves think of and talk about their practice.

Our analysts do not see security as something that takes place or can be understood in isolation. The work must be talked about and understood in context. Understanding how analysts think about protecting their internal resources is intrinsically linked to how they leverage external resources, which is why the primary focus of protecting the *environment* is discussed along with the work practice related to the *community* that feeds into and makes possible that protection.

The core tasks in the ID workflow, the inputs and outputs of these tasks, and the flow and feedback between them are shown in Figure 1. The left side of the diagram shows the internal environment, the right side shows the external community. The main tasks of the workflow diagram include:

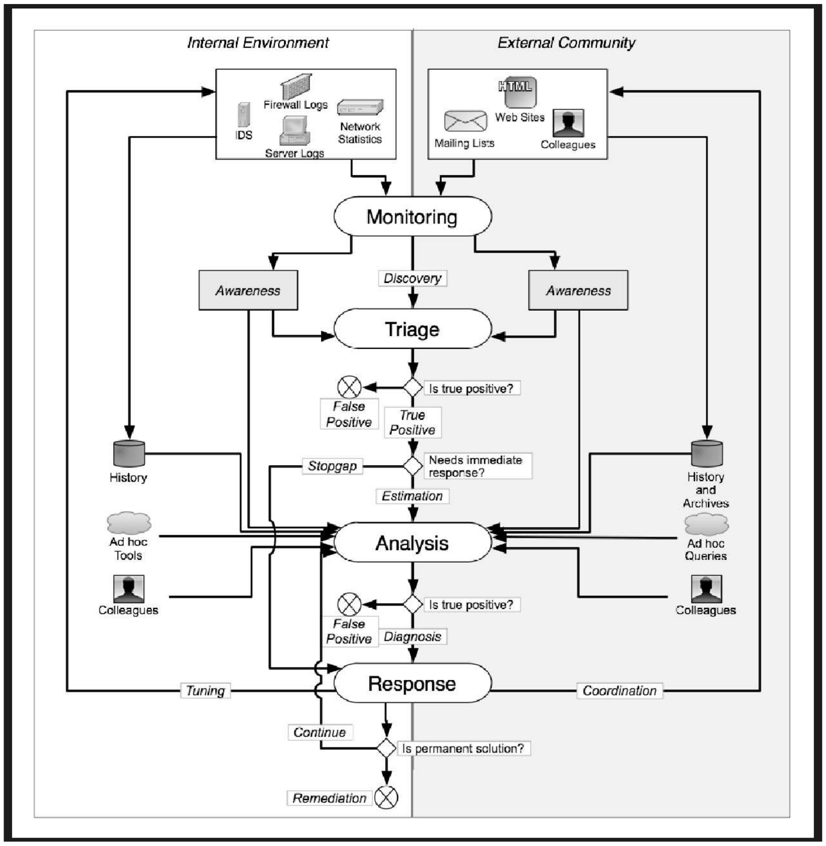

Figure 1. Intrusion detection workflow diagram.

- *Monitoring* leads both to the discovery of new events and to a general awareness of both the environment and the issues of the community, such as attack methods and vulnerabilities. A sample of the primary inputs to the monitoring task include: the IDS, firewall logs, system logs, and network statistics (for the internal environment), and mailing lists, web sites, and colleagues (for the external community).

- *Awareness* is leveraged extensively in both the triage and analysis activities. (Awareness is shown redundantly on both sides of the diagram to indicate that this refers to both internal environmental awareness and external community awareness.)
- *Triage* leads to either discarding the event as a false positive or conducting an initial estimation of the attack. Triage can also lead to an immediate, but temporary, stopgap response, such as blocking the attacker at the firewall.
- *Analysis* leverages multiple internal and external resources, including historical (embodied in, for example, a history of attacks against a particular machine and mailing list archives) and current (embodied in ad hoc data collection from packet capture or vulnerability analysis tools, or queries to mailing lists or colleagues) context. Analysis leads to dismissing the event as false, or a more concrete diagnosis, which leads to a response.
- *Response* can be a stopgap measure, such as unplugging the network cable, or a permanent solution, such as patching a system. Response also includes feedback into both the internal environment inputs to monitoring, such as tuning an IDS signature or adding a firewall rule, and the external community, such as posting a message to a mailing list about a new attack. Before examining each of these task phases in turn, we will first return to the daily work of analysts to concretize some of the themes raised in the scenario above.

Daily Rituals

P5: First thing every morning, Tripwire, Snort, and ISS. Everyday, we go through web logs and look at stuff, if [anything is] out of whack.

Interviewer: What do you look for in web logs?

P5: Something that looks malicious that we know is not on our system, some kind of file. Generally you will see it, like cmd.exe, or something like that.

Interviewer: Can you setup an intrusion detection system [to look for this]?

` P5: We still go through them in case, because the signatures aren't catch[ing] everything. Everyday, we look at Bugtraq, full disclosure, some of the other product lists that we subscribe to. I look for any vulnerabilities that come out. Any kind of attacks.

An examination of the rhythms of analysts' daily rituals exposes some of the characteristic themes that run through all tasks. Similar to Marcus' story and the interview excerpts above analysts describe their typical morning: first monitoring the environment, looking at the log files generated by the various IDSs and at server logs for anything strange or out of place that may not have been picked up by an IDS; and then checking the community mailing lists and web sites for any new vulnerabilities or attack methods. Each of the participants followed a similar daily ritual. Most analysts, like the example above, started their morning investigating the previous night's log files: "every morning for an hour to an hour and a half, I go look at it [the IDS log], look at all the alerts and see if there is anything that sticks out" (P9). This was accompanied by scanning the Internet for news of the latest attacks, vulnerabilities, and IDS signature updates. All participants acknowledged this was a necessary, though tedious, component of ID work and provided us with the numerous web sites , mailing lists, and colleagues that they regularly monitored for this kind of information. This activity was usually part of the analyst's daily ritual: "my first stop every morning is the security websites to see what the threat du jour is and if there's something that we can craft a signature for if it's not [already available]" (P6).

These daily rituals underscore the two key dimensions of ID work, the analyst's internal environmental concerns and external community involvement, embodied in the mailing lists and web postings that the analysts use to communicate and keep each other updated on recent developments. Also embedded in these rituals are the two primary motivations for intrusion detection: *discovery* and awareness. The discovery process - examining IDS logs and community resources for events and vulnerabilities that directly affect the analyst's environment - is intrusion detection per se. In addition to this discovery process is the equally important goal of keeping abreast of new developments and providing a context for interpretation of future events: *awareness*. More often than not, the results of this activity are internalized. The analyst remembers this information and when needed can recall the generalities and, if need be, knows where to go to find the specifics.

Monitoring

Monitoring is the first phase of ID work. It includes the ongoing surveillance of both the analyst's internal environment looking for indications of anomalous or malicious activity and external community resources looking for vulnerabilities and new attacks. Both awareness and discovery start here. These mundane tasks of ID constitute the majority of analysts' time.

Community Monitoring: "We're always monitoring all the lists"

> I always have it [the IDS console] up on my screen and the time of the last alert will always scroll through the bottom…. So, this alert thing is just a tiny little frame in a web window that

> alerts every few seconds and then tails the alert logs, the raw alert logs, before they're fed into the database. And if that starts changing more than a few times a minute, I've probably got something going on. Because, I mean…in some environments, it's normal to have two hundred alerts a minute, but for me it's not. (P3)

This analyst is describing a custom console screen, which he is always monitoring in real-time, though it is rarely the single focus of his attention. The console includes system log files and other internal sources, including IDS logs. The IDS logs scroll through the latest alerts, with the newest always at the bottom. This analyst always has the console in his peripheral vision, and looks more closely periodically, particularly when the frequency of alerts jumps up to more than a few times a minute. Another analyst describes this process as "observing and monitoring, just waiting for the next event." (P4) Continuous, real-time monitoring with a dedicated console or monitor that always displays new alerts as they occur is typical. Analysts do not always have time to look at each individual alert, but the alerts themselves are mentally noted and later used for context during the triage or analysis phases. Several of the analysts could not afford the time to do real-time monitoring, however, and only look at their IDS logs once or twice a day, typically first thing in the morning and before they leave for the night. Different analysts have different strategies for monitoring their environment, typically based on their available time dedicated to intrusion detection.

Community Monitoring: "We're always monitoring all the lists"

The most common external resource for monitoring was Internet mailing lists: "We're always monitoring all the [mailing] lists, everything" (P5). Analysts work in idiosyncratic environments; no two network environments are identical. This is reflected in the diversity of their frequently read mailing lists. For example, participants in academia monitored a variety of lists specific to security in higher education, while the analyst who administered a number of Apache web servers on Linux machines monitored the mailing lists relevant to his versions of Apache and Linux. These routine monitoring tasks are tailored for each individual's environment, but with a great deal of overlap. All of the participants monitored general incidence and vulnerability lists that attempt to quickly disseminate information about new bugs, vulnerabilities, and attacks. These high traffic lists are run by the community of information security experts; often the analysts themselves contribute to these lists. These mailing lists deal with a broad range of security issues for a wide variety of platforms and applications. They are general enough to pertain to nearly all environments, leaving it up to the analyst to determine which notifications pertain to their particular environment.

The participants took it for granted that we, the researchers, would be familiar with the importance of mailing lists and which ones were most well respected and referred to most often. This tone is apparent when one participant responded to a question about what information sources were used:

> Obviously, the mailing lists, Snort signatures, Snort users, ISS mailing lists, we're subscribed to all those. ISS forums, of course… [I] go to, of course, Bugtraq, and I go to Security Focus. (P5)

The first two mailing lists mentioned here refer to the two IDSs used in the organization; these lists foster general discussion about the particular IDS and topics specific to the announcement and development of new signatures. The signatures are the rules that the IDS uses to find intrusions in network data. Bugtraq, which another participant referred to as "the biggest resource" (P8), is a mailing list that "is a full disclosure moderated mailing list for the *detailed* discussion and announcement of computer security vulnerabilities: what they are, how to exploit them, and how to fix them" (http://securityfocus.com/). The community and the mailing lists that link the community together are discussed in more detail later in the paper.

Security analysts are tied together in a community of practice (Lave & Wenger, 1990; Wenger, 1998) by Internet mailing lists, the archives of which form a living knowledge base that embodies the collective knowledge of the members who contribute to it. Analysts speak in the same technical jargon, share their expertise, collaborate on problems, and help novices learn both by directly answering their questions and pointing them towards relevant resources - which are often previous mailing list posts. Analysts are joined in a fight against a common, unseen enemy and necessarily have joined together to form this community. While large and widely distributed and with many peripheral members, the shared domain, language, and work practice constitutes a community of practice that fosters learning and serves as a steward of knowledge. Communities of practice are "groups of people who share a concern, a set of problems, or a passion about a topic, and who deepen their knowledge and expertise in this area by interacting on an ongoing basis." (Wenger et al., 2002) Such a community shares information, helps members solve problems, explores ideas, and accumulates knowledge. Although the members of the IDS community rarely meet face-to-face and are geographically distributed with participation crossing organizational boundaries, it is a community of practice that goes beyond shared interests and a common domain. As described in this paper, the practice of intrusion detection - even in organizations of differing size, type, and mission - is remarkably similar.

Monitoring Strategies: Aggregation vs. Pruning

> When Snort [version] 2 came up, we wanted to see what it could do for us and we enabled every single signature that comes in the package by default and in the couple of hours we had it running, we clocked two and a half million alerts. (P6)

As this analyst explained, the number of alerts generated by an IDS - especially on large, heterogeneous networks - can quickly become overwhelming. Because the role of the analyst usually involves more responsibilities than ID, coping with data overload forces a difficult choice: analysts can choose to limit the IDS signature set and thus the number of alerts, or be inundated with alerts to the point where alerts can no longer be monitored on an individual basis, but only be looked at in aggregated summaries. Pruning out signatures begins with removing those that do not pertain to the analysts' environment:

> I've gone through and taken out countless signatures on ISS because most of them, nearly all of them don't apply to any of our software. They have stuff on PHP, we don't run PHP, they have Apache, we don't run Apache. (P5)

Reconfiguring and removing IDS signatures is usually the first step an analyst takes to reduce the number of false positives. In this case, the analyst removed signatures that pertained to a type of web application and server that they do not use in their environment - the signature will always be a false positive. However, severely limiting the signature set, although it can dramatically reduce the number of false positives, also has a downside - increasing the number of undetected attacks. One analyst rationalizes this balance as follows:

We also have a very limited signature set at the moment. Part of that is performance gear, the other part of that is just data inundation… If we were doing the alerts the way you should be doing alerts, which is, we don't have nearly as many things commented out [i.e., removed] as we do now. (P6)

Those participants that did restrict their signature set did so with the knowledge that they were probably missing many actual attacks, but had no effective means of monitoring the large numbers of alerts. Pursuing the opposite strategy, analysts were forced to look only at aggregated summaries of alerts, thus reducing alert fidelity. In this case, choosing which alerts to pursue can devolve to almost random selection: "Generally I only pick one or two [alerts] of interest [to investigate]…based on what problems we've been having lately" (P1). Picking the one or two alerts of interest out of the hundreds or thousands generated is one way of dealing with the problem, but leaves many alerts unresearched. Neither of these strategies, removing potentially valid signatures or leaving them in but only looking at summary data, represents the optimal approach to monitoring, but analysts often have no choice because of data overload and lack of time. Thus, with either approach, the primary strategy for monitoring could be described as *satisficing* (Simon, 1957). There is rarely time or resources to the best job or to do everything they would like to do, so analysts are satisfied with doing "good enough" to get by, even while they realize that this is not the best possible approach. More time would help, but better tools are also needed.

Monitoring the status of the environment involves interaction with an IDS and other internal monitoring tools as well as continually following external information sources looking for vulnerabilities that might apply to analysts' particular environment. Monitoring these internal and external data sources is done to increase awareness and to detect new events. All of these monitoring tasks are part of routine ID work: time-consuming, but not as cognitively challenging as the subsequent phases.

Triage

> A lot of that comes down to what's getting hit [attacked]. The limited staff, you know, we kind of triage it, in the sense that it's a critical campus service, it's one of the [domain] name servers, time servers, whatever servers, or if it's any of the systems in our applications support area, generally that involves dropping everything and running, kicking and screaming to deal with it. (P7)

The primary purposes of monitoring are to increase awareness and to discover new events, vulnerabilities, or attacks. If an analyst does discover a suspicious event within their environment or the community reports a new vulnerability or attack method, the immediate subsequent phase is triage. This is the quick dismissal (as a false positive) or prioritization of events, such as IDS alerts, that analysts perform after discovering a new event during monitoring. As in the example above, this can be based on the importance and function of the server that is attacked. This requires that the analyst know the importance and function of the servers in their environment, information that is rarely systematically collected into a repository. If this information actually is collected somewhere, it is usually static and gets out of date rather quickly as machines are moved, repurposed, upgraded, or retired and new machines are added to the network. The triage stage is necessarily very fast, almost over before it begins, with analysts either moving to the more in-depth analysis phase or dismissing the event as a false positive or not applicable to their environments.

It is important to note that most events do turn out to be false positives. One participant said that to the best of his knowledge, none of the alerts from either of his two IDSs has ever been a true positive: "We haven't had any break-ins, but we get alerts all the time. Our system has never been vulnerable to the alerts that have been through" (P5). Another analyst quantified the ratio of easily recognizable false positives at 90%:

> Very few of the incidents we ever see are active hacking activity where somebody is targeting a particular system on campus and they are trying really hard to break into it. Ninety percent of what we see is automated tools just canvassing the network looking for something. (P6)

Many events can be instantly dismissed, especially when they can be quickly recognized as automated scans or naïve attacks. Analysts often discussed monitoring as being essentially a waiting game, interrupted occasionally by the need to quickly determine if an event could be dismissed, or if it had to be looked at in-depth. Three examples of new event triage help demonstrate this:

> I know these criteria will always cause a false positive, even though there's different event types being triggered, you can always go and filter those out and you can just reassure yourself in two seconds that's another false positive. (P3)

> Certain IDS output…certain things you can always believe is a known attack, just because the experience you have and the rule signature may be so tuned to where it always detects that [attack]. (P8)

> If you look at the traffic and you know it is false alarming because a user name matches a signature and that occurrence, if you know that, but you still want the signature in, you just ignore that host, because it is hitting that signature. (P7)

In the first case, the analyst can easily determine that an alert is nothing to worry about because of experience with similar alerts in his environment with this signature's criteria tells him that this alert is always a false positive. No further investigation is necessary. However, as in the second example, there are times when it is immediately apparent that an alert is not a false positive, based on his personal experience and intimate knowledge of the particular signature that generated the alert. The third example is also concerned with knowledge of the signature; in this example, certain signatures are prone to generating false alarms because a username may match the pattern the signature is checking. It is difficult to tweak this kind of innocuous alert, also common with email traffic, because the free form text of the packet payload happens to match a signature. If a quick glance does not reveal the importance of the alert the process continues to the analysis phase.

Analysis

> So if I see this thing [IDS console window] going, I might click on this and then it will pop up my DeepSight window and

> I can go in and look for DeepSight data a year back and I can look for a certain IP a year back. Or I'll pop it up and I'll go into Ntop and I'll look to see what protocol traffic's going on. … I can tell if something's going on, I know the network. (P3)

This excerpt succinctly summarizes the core components of analysis to determine the accuracy and assess the severity of events: the use of historical data for context, the contemporary context gleaned from ad hoc tools (in this example, the Ntop network analysis tool), and the context provided by an intimate knowledge of the environment that the analyst works in. Each of these will be discussed below.

Historical Context

Analysts rely on a historical context to understand an event or recognize trends and patterns to accomplish the analysis task. This historical context can be fairly straightforward, such as: "This specific host on this specific port has been attacked x number of times and from these IPs." (P8). This analyst is talking about systematically collecting a list of hosts that have been attacked, including what kind of attack it was (as determined by the port number), the frequency of that attack, and where the attack originated. Having this historical context can vastly aid an analyst. However, not all of the participants had an efficient means of searching for these correlations.

A repository for historical context, when analysts did systematically collect it, came in various forms. This could be a history of previous IDS alerts, firewall log files, or a summary of network statistics such as network flow data. This type of historical contextual information, systematically collecting and storing statistical data, is often referred to as a baseline. Many of the interview participants noted their desire to have this information, but few were actually collecting it. While these data sources were typically used by analysts, most of the historical context - information related to previous successful or unsuccessful attacks or knowledge of historical network data - was stored, queried, and processed internally in the analyst's memory. Analysts reported that they often would just remember having seen a similar attack or an attack from the same address. This internal knowledge was critical to performing analysis. On the other hand, log files from firewalls were nearly ubiquitous sources of both historical and contemporary context, such as this survey respondent: "[I] retrieved logs from Firewalls and then started to look into the issue."

Contemporary Context

> In general if I get an alert…I will go in to the actual packets to do some overall summarization of that incident, what that box was doing at that time, are there any other related

> incidents, if it is someone externally attacking us, I would be interested in what other boxes did that external IP touch, so do a correlation. (P7)

The particular contemporary context of a situation is another important factor in ID event analysis. In the example above, the analyst describes capturing the raw network traffic, and then trying to correlate various sources of contemporary information together to build up a picture of what was happening on the network at the time the alert was generated. This contextual information is ephemeral; if the analyst does not have a mechanism for capturing the information quickly, it could be lost. Many aspects of a network are always in flux. In order to gain this context, analysts rely on myriad data sources and tools that provide historical and current state information. These information stores must be accessed through separate tools and procedures, collated, and correlated back to the original data. Ad hoc packet capture tools are commonly used to gain contemporary contextual data, as in this survey response:

> Use Cisco IDS to detect rogue IP addresses attempting access on the network; track that exact IP address in real time with Wireshark [packet capture tool] to see what is that other person attempting to do. (P8)

In this example, the IDS generated an alert of machines illicitly trying to gain access. The analyst used an ad hoc packet capture tool to try to determine exactly what the intruder was trying to do. Performing the analysis of an IDS alert or vulnerability is grounded in the experience and expertise of the analyst, and in the relevant contextual facts surrounding the activity. Successfully diagnosing an alert or vulnerability is a difficult, complex task that requires an ability to improvise and develop custom methods, tools, and scripts to facilitate data collection and correlation.

Environmental Context

In addition to the historical and contemporary context of an event, the most important resource analysts draw upon during analysis is their experience within their environment. Analysts' expertise includes general knowledge of network protocols and ID, but most importantly, knowledge of their unique network environment, because what is normal activity in one environment may be indicative of illicit activity in another. All of the participants echoed the importance of having an intimate knowledge of their particular environmental context: "I can tell if something is going on, I know the network" (P3). Analysts must not only learn the intricacies of network protocols and system operations, but how those are manifested in a particular environment that is constantly changing. One survey respondent reported that: "The most important thing is knowledge of the target [machine]." This knowledge is necessary to not only determine the accuracy of an event during analysis, but to also assess the impact and severity of true events.

Keeping up with changing configurations in the operating environment is difficult, but necessary to provide the context needed to analyze and diagnose an alert. This includes knowing the details of each machine on the network. This can be as simple as recognizing that a Windows IIS web server attack targeted at a Linux machine running Apache is clearly a false positive, though it becomes more challenging very quickly as the number and diversity of machines on a network increase. One survey respondent, for example, reported: "I also ignore all the Microsoft based-attacks since I don't allow any Microsoft-hosted applications in any of my external DMZs." (The DMZ is the semi-protected logical network that lies between the outside world and the internal, protected network.) In this example, the analyst can immediately discard any attacks that are related to Microsoft for all machines in that network, although will have to do further analysis for Microsoft-related alerts to internal machines.

Knowing the environment was also embodied in the less specific and difficult to describe phrases such as "knowing what's normal traffic" (P12). For the analysts we interviewed, tracking this knowledge is accomplished almost exclusively through personal memory, without any external support. (Several participants did note that creating a database of this information would be helpful.) For many of the participants, this detailed knowledge of the environment was manageable enough that they could do a reasonable job of using their memory to recognize certain target machines or services as being vulnerable or not to a particular attack. Obviously, as the size of the network increases and the systems and network administration tasks are more distributed, relying on personal memory for all of the details necessary for ID analysis becomes untenable.

Response

The most common forms of response in ID are intervention and feedback. Response also included reporting, such as generating incidence reports for legal action and reports for management. Intervention, actively intervening as a response to an event, can occur following triage or analysis. In response to triage, this is an immediate, temporary expedient intended to buy the analyst time to perform further analysis. In response to analysis, intervention involves remediation, more of a permanent solution. One of the interview participants describes both of these types of responses performed in conjunction:

> So, what we do is, once we spot it and are confident that it's a real attack and not just a false alarm, we block access from the outside. And if it looks like it tripped a signature

> that indicates a successful exploitation, we contact the owner and say, we've got a problem. (P6)

This example includes first a quick and dirty stopgap (blocking the attacker's access) to buy time for a more permanent remediation (contacting the victim machine's administrator).

Who intervenes depends on the role of the analyst in the organization and organizational policies. Analysts who are also administrators of targeted machines would likely intervene themselves. The response to an attack in progress could be as drastic as unplugging a network connection: "probably the first thing would be unplug it" (P9), echoed by a survey respondent who reported that her response to a memorable attack was that she "took [the] system off line." Responses can also occur after the fact, such as patching the vulnerability or reinstalling the compromised machine from backup. Especially in larger organizations, the analyst in charge of ID is not the administrator of most machines. In this case, the response involves coordination among other administrators in the organization, as in the following survey response reporting how an attack was handled: "Researched online the impact, and sent off an email with a summary of the impact to the administrators."

Feedback is usually directed at the IDS or other elements of the security infrastructure. It includes tweaking or removing IDS signatures that generate an excessive amount of false positives, even if the signature was not guaranteed to always generate a false positive. As noted earlier, this practice is dangerous because it can lead to false negatives yielding undetected intrusions. Configuring and tweaking the IDS for the particular environment is one of the most challenging ID tasks, but one that teaches the analyst the nuances of that environment and how the IDS operates in that context. Feedback often also involves submitting attack information to security mailing lists or vendors.

Collaboration

> [Snort] has an active development community… when I run into problems, chances are somebody else has already run into it and the mailing list has got something on it that I can find pretty quickly. If all else fails, I can call Arthur [another IDS expert at a different location] and he knows Mike [an IDS developer]. (P6).

In addition to the tasks that emerged from this research, collaboration was a major theme that is interwoven within each of these tasks. The mailing lists are both the means for connecting their participants together and the dynamic repository where knowledge generated by the community is automatically captured and ready to be searched and reused. This knowledge base is where analysts will go first, searching for similar issues that have already been resolved by the community. If the information cannot be culled from the archives, analysts will turn to the ID community, sometimes directly to personal contacts, but more often than not to the community as a whole. Novices and experts alike rely on community forums for finding answers to difficult or emergency problems, and in the ID world, these are Internet mailing lists.

Individual members contributing to the communal knowledge base, particularly during a widespread security crisis, is an essential activity of the community. One participant relays doing so during the spread of some well-known, pernicious Internet worms:

> We had some cases where there was a couple of weeks where there was a new variant coming out every week and people would just change, because… the first wave of this worm hits and everybody can fingerprint it right away, I mean, we had it fingerprinted. I think I was one of the first people on the incidences' list at Security Focus that supposedly fingerprinted it. (P3).

"Fingerprint" refers to configuring the IDS signature to detect the network data's attributes that identify the worm. This analyst was proud of quickly fingerprinting the attack and rapidly disseminating that knowledge to the community. Especially during outbreaks of new security threats, expert analysts collaborate across online channels to quickly develop signatures and techniques for identifying new attacks that can then be reused by the wider community. While this reuse is rarely a drop-in solution, signatures will require tweaking to fit the particular environment, this process of sharing ideas and testing technical solutions is the primary means of knowledge distribution.

Another example of collaboration across organizational boundaries was described by a participant working as a contractor within a government military agency. He took it upon himself to continually inform a colleague (and friend) at a different agency who was tasked with updating the commercial world of trends and vulnerabilities that the military had discovered. This communication was not through official channels; it was entirely due to the realization that there were mutual benefits to collaboration. There had previously been little communication between the two government agencies, and what information was passed was often delayed to the point where it was nearly useless. Through knowing someone in a different agency, the communication and dissemination of security information was vastly improved. This analyst had little oversight from his supervisor, a high-ranking military official temporarily assigned to oversee the security group, but forged his own ties with peers working in the same domain. Another analyst, working in academia, had an informal network that included many other analysts working within different academic institutions. Whenever a new attack method or vulnerability would arise, these analysts would quickly contact each other to help craft new IDS signatures and methods of detection. In both of these cases, the collaboration crossed organizational boundaries, bypassing management to improve each party's security.

Conclusion

This section illustrates some of the more important implications for tool design and organizational policies that resulted from this research. Primary among the design implications is the need to design tools to fit the specific tasks of ID work. Monitoring, triage, analysis, and response all call for different types of tools. Understanding the workflows of analysts can also help organizations institute policies and improve staffing to better support the work, and thus enhance organizational information security.

Monitoring, in particular, must facilitate awareness of the analyst's internal environment and developments in the external community. Monitoring should take advantage of analysts' pre-attentive processing capabilities, since those who are pulled in multiple directions cannot always give this task their full attention; a simple glance should be enough to quickly understand the state of the network environment and to notice important changes. However, analysts are unlikely to stare at a display for long periods to detect anomalous or malicious activity; it is simply beyond human capability for them to do the monitoring task without some kind of automation. Because of this, monitoring tools should not seek to supplant current automation techniques that aggregate and classify data, but rather to complement those automated methods. Information visualization displays can help, but are not a panacea and should attempt, particularly for the monitoring task, to help analysts in the perception of new events.

Triage tools need to support very fast event categorization and prioritization, in addition to supporting communication among multiple analysts. This class of tools does not need to support complex, exploratory data analysis. Rather, they should provide the analyst with enough information to make a fast, effective decision of whether or not something needs to be investigated further. Knowing the importance and functionality of the machines within the environment is very important to triage, but this information is rarely collected or kept up to date. Automatically identifying attributes of networked servers would vastly improve this, especially if this information could be automatically linked to the monitoring system. Such a system would likely combine a human supervision with a semi-automated approach, automatically suggesting the role of a machine and its relative importance based on a set of heuristics that can be tweaked and learn from human input. For example, a server with port 25 open and listening is probably an email server, since that is the SMTP port. If there is only one server with port 25 open, then that is likely to be the only email server, and would then be deemed very important. A machine that has a lot of outgoing traffic on port 80 on the other hand (indicating a client browsing the web), is likely to be a workstation and not critical.

Analysis tools should incorporate multiple data sources, allow for ad hoc and historical data correlation. These tools should also encourage analysts to learn what is normal and facilitate putting events into that context. In order to gain this context, analysts rely on myriad data sources and tools that provide historical and current state information. The analyst must locate or gather the data, determine if the data is relevant, and correlate the data with the event being analyzed. This time-consuming, difficult task is necessary to gain a full understanding of an event. This includes not just current state information, such as what services is a host running at immediately after the alert, but also historical information, such as if the host under attack has been targeted previously in a similar fashion. The difficulty lies in not just collecting and parsing all of this data, which is a nontrivial task, but in correlating all of the data together with the information surrounding the event. This complex task dictates that tools to support analysis incorporate both high- and low-level views of the data being analyzed. An overview will keep analysts from losing sight of the "big picture" (P3) when they are examining low-level details of an event, which are crucial in analysis. One participant described the importance of having all packet details available in the analysis task: "the most important [thing] would still have to come down to just having the raw data" (P7). Participants described the difficulties in moving back and forth between these macro- and micro-levels of details; support tools should ease this transition process. A visualization tool developed as part of this research, described in (Goodall et al., 2006), attempts to solve this problem through simultaneously presenting these different levels of details through information visualization to preserve context.

Analysts rely heavily on their internal memory for much of these tasks, but as their networked environment becomes more dynamic and heterogeneous, this will no longer be possible. Externalizing analysts' situated knowledge would help ease these growing pains. In addition, it would assist new analysts in coming up to speed. Simple static information like what operating system or services are running on a host can be difficult to keep track of mentally when networks are large or susceptible to change, so analysts could benefit even from this relatively simple information. Due to the dynamic nature of computer networks, care would need to be taken to ensure that this static information would be updated regularly to guarantee that analysts are working with the correct information. Support that is more complete would also include the ability to dynamically query historical data that analysts often keep in their memory or on scraps of paper.

Tools designed to support response should reflect the collaborative nature of this task and provide feedback into other tools, especially IDS signatures. Feedback loops are a particularly interesting area for future research. Visualization tools have the potential to help analysts discover patterns easily, but few systems allow the reuse of these patterns to inform the ongoing IDS configuration process. This tuning process is important as a means of keeping up with the dynamic nature of network environments and as a means of facilitating learning. Being able to use a visual display to create IDS signatures would greatly aid novices and experts in the necessary, ongoing tuning process and at the same time help them to learn normal network behavior to facilitate analysis.

Organizational implications of this work include shifts in policy and staffing to better support ID work. Organizations should implement policies to support external awareness, community collaboration, and the ongoing IDS tuning process. All of these activities are largely invisible to management, yet crucial to the ID work practice. While it may seem that surfing the web and following mailing lists for information related to new attack vectors, methods for detection, or vulnerabilities does not directly support the mission of keeping an environment safe, the awareness derived from these external monitoring practices are crucial to performing ID work. Organizations should recognize that these activities are intrinsically linked to successful ID and encourage time spent away from solely monitoring the environment to monitoring the information coming from the community.

Collaborating across organizational boundaries in areas related to information security is problematic due to the nature of the domain. Organizations do not want their secrets revealed. However, the ability to collaborate and share can lead to more effective ID. Data anonymization tools can help facilitate data sharing, communities of practice can help tie practitioners together, and organizational support can provide the time and resources for collaboration. In addition to supporting the intake of information from the community, organizations should support analysts' active participation in the larger community and try to encourage the growth of the communities.

The monitoring strategies that were typical of the participants in this study, having too few or too many signatures in their IDS, are problematic. Organizations could foster better security by recognizing and supporting the IDS tuning process. Configuring an IDS is an ongoing process, discussed in more detail in (Goodall et al., 2009). Organizations could provide additional manpower and resources to this important task. This lack of time and resources was a recurring theme for many of the participants. Many were responsible of the ID of their organization on their own. Additional personnel would help make security more robust. This would require an organizational shift towards dedicating more resources to security, even though much of the ID work is not directly seen by management.

In addition to policies to encourage the invisible aspects of ID work, organizations should also carefully consider staffing issues. Simply supplying additional human and Information Technology resources is unlikely to help enhance security. Instead, ID manpower can be pulled from the existing pool of systems and network administrators. There is often a trend towards specialization in IT. A systems administrator is typically responsible for a certain Operating System or class of machines. This helps personnel become deeply knowledgeable in a few areas. This specialization may be counterproductive when it comes to ID. Much of ID work requires holistic knowledge of the environment. This knowledge is essential to successful ID. It comes from understanding how technology is implemented within a particular environment. A systems administrator who set up and maintains a web server knows what services are running and what software patches have been applied. A network administrator who set up the access control lists on an Ethernet switch understands what subnets should not be able to access crucial servers. These kinds of knowledge come from doing administration work. By dedicating staff specifically to ID, this knowledge will not be sufficiently developed. Instead, additional staff could be rotated between systems and network administration work and security work. Then, all administrators could then contribute to the work of ID. More junior administrators could perform environmental monitoring and triage. Administrators that are more senior could perform analysis and response. This would help junior staff learn the environment and would ensure that IDS alerts are being analyzed by the people who best know the environment. Rather than having a small, dedicated security staff that is highly specialized doing ID work, a larger pool of administration staff would be sharing the work. With a larger pool of manpower, some of the important tasks, such as systematically collecting baseline statistics about a network and the ongoing IDS configuration, could be handled by a subset of the staff, since there would be more eyes available for monitoring, the most time consuming of the tasks.

Intrusion detection is a critical domain in today's world, but one that is not well understood. This paper has exposed the daily rituals, tasks, and collaboration that make up the practice of ID. The core tasks of ID - monitoring, triage, analysis, and response - each call for a different set of tools. However, the current set of tools analysts use are often designed without this understanding. The lack of effectively fitting tool to task is one of the primary drawbacks to security tool design; many tools that security analysts presently use consist of customized scripts or clever hacks. This paper has presented a detailed description of practice and outlined some directions for the better design or redesign of tools and policies to support ID. The enriched understanding of work in context can help system designers develop better tools and organizations implement better policies to help defenders keep up with their increasingly complex environments. As more computers are introduced into networks and more data is digitized and warehoused, network defenders will have more and more assets and information to protect. This situation demands that better tools are designed to help analysts cope with their ever-increasing purview, tools that fit their practice while making their work more efficient and organizational policies are implemented to better support the work practice of the security analyst.

References

(2005) 2005 E-Crime Watch Survey. CSO magazine / U.S. Secret Service / CERT Coordination Center.

Allen J, et al. (1999) State of the Practice of Intrusion Detection Technologies.

Bentley R, et al. (1992) Ethnographically-Informed Systems Design for Air Traffic Control. In Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW), pp 123-129.

D'Amico A, et al. (2005) Achieving Cyber Defense Situational Awareness: A Cognitive Task Analysis of Information Assurance Analysts. In Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting, pp 229-233.

Goodall JR, Lutters WG and Komlodi A (2004) I Know My Network: Collaboration and Expertise in Intrusion Detection. In Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW), pp 342-345, ACM Press.

Goodall JR, Lutters WG and Komlodi A (2009) Developing Expertise for Network Intrusion Detection. Information Technology & People 22(2), 92-108.

Goodall JR, et al. (2006) Focusing on Context in Network Traffic Analysis. IEEE Computer Graphics and Applications 26(2), 72-80.

Heath C and Luff P (1992) Collaboration and Control: Crisis Management and Multimedia Technology in London Underground Control Rooms. Journal of Computer Supported Cooperative Work 1(1), 24-48.

Hughes J, Randall D and Shapiro D (1992) Faltering from Ethnography to Design. In Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW), pp 115-122.

Lave J and Wenger E (1990) Situated Learning: Legitimate Peripheral Participation. Cambridge University Press, Cambridge, UK.

Lee W, Stolfo SJ and Mok KW (2000) Adaptive Intrusion Detection: A Data Mining Approach. Artificial Intelligence Review 14(6), 533-567.

Luff P, Hindmarsh J and Heath C (Eds.) (2000) Workplace Studies: Recovering Work Practice and Informing System Design. Cambridge University Press, Cambridge, UK.

Lutters WG and Ackerman MS (2002) Achieving Safety: A Field Study of Boundary Objects in Aircraft Technical Support. In Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW), pp 266-275.

McHugh J (2001) Intrusion and Intrusion Detection. International Journal of Information Security 1(1), 14-35.

Roesch M (1999) Snort - Lightweight Intrusion Detection for Networks. In Proceedings of Thirteenth Systems Administration Conference (LISA), pp 229-238.

Simon HA (1957) Models of Man. John Wiley and Sons, New York.

Star SL and Strauss A (1999) Layers of Silence, Arenas of Voice: The Ecology of Visible and Invisible Work. Journal of Computer Supported Cooperative Work 8(1/2), 9-30.

Stolze M, Pawlitzek R and Hild S (2003a) Task Support for Network Security Monitoring. In ACM CHI Workshop on System Administrators Are Users, Too: Designing Workspaces for Managing Internet-Scale Systems.

Stolze M, Pawlitzek R and Wespi A (2003b) Visual Problem-Solving Support for New Event Triage in Centralized Network Security Monitoring: Challenges, Tools and Benefits. In GI-SIDAR conference IT-Incident Management & IT-Forensics (IMF).

Strauss A and Corbin J (1998) Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory. SAGE Publications, Thousand Oaks, CA.

Wenger E (1998) Communities of Practice: Learning, Meaning, and Identity. Cambridge University Press, Cambridge, UK.

Wenger E, McDermott R and Snyder WM (2002) Cultivating Communities of Practice: A Guide to Managing Knowledge. Harvard Business School Press, Boston, MA.

Yurcik W, Barlow J and Rosendale J (2003) Maintaining Perspective on Who Is the Enemy in the Security Systems Administration of Computer Networks. In ACM CHI Workshop on System Administrators Are Users, Too: Designing Workspaces for Managing Internet-Scale Systems.

Acknowledgements

Thanks to Carolyn Seaman, Penny Rheingans and Anita D'Amico for the feedback on this work, and to Medha Umarji for her constant encouragement.

Autor Biography

John R. Goodall is a Research Scientist with the Secure Decisions division of Applied Visions, Inc. His research experience and interests include visual analytics, information visualization, human-computer interaction, computer network defense and computer-supported cooperative work, particularly the intersection between these areas. He chaired the annual International Workshop on Visualization for Cyber Security (VizSec) in 2007 and 2008. Dr. Goodall earned his MS and PhD in Information Systems from the University of Maryland, Baltimore County (UMBC) and his BA in History from Binghamton University.

Wayne G. Lutters is an Associate Professor of Information Systems at the University of Maryland, Baltimore County (UMBC). He has recently served as Program Director for Human-Centered Computing at the National Science Foundation. His research interests are at the nexus of computer-supported cooperative work, social computing, and knowledge management. He specializes in field studies of IT-mediated work, from a socio-technical perspective, to better inform the design and evaluation of collaborative systems. Dr. Lutters earned his MS and PhD in Information and Computer Science from the University of California, Irvine and his BA in Cognitive Science and History from Connecticut College.

Anita Komlodi is Associate Professor and Graduate Program Director for Human-Centered Computing at the Department of Information Systems, University of Maryland, Baltimore County (UMBC). Dr. Komlodi's research interests fall in the area of Human-Centered Computing. The first area of concentration is at the intersection of Human-Centered Computing and Information Retrieval/Information Behavior and focuses on the study of Human Information Behavior and the design of user interfaces for information systems. Dr. Komlodi is also interested in the needs of diverse user groups go technology: age, gender, and cultural differences in technology interactions.