Turning Contradictions into Innovations or: How We Learned to Stop Whining and Improve Security Operations

Sathya Chandran Sundaramurthy, *University of South Florida;* **John McHugh,** *RedJack, LLC;* **Xinming Ou,** *University of South Florida;* **Michael Wesch and Alexandru G. Bardas,** *Kansas State University;* **S. Raj Rajagopalan,** *Honeywell Labs*

https://www.usenix.org/conference/soups2016/technical-sessions/presentation/sundaramurthy

This paper is included in the Proceedings of the Twelfth Symposium on Usable Privacy and Security (SOUPS 2016).

June 22–24, 2016 • Denver, CO, USA

ISBN 978-1-931971-31-7

Open access to the Proceedings of the Twelfth Symposium on Usable Privacy and Security (SOUPS 2016) is sponsored by USENIX.

Turning Contradictions into Innovations or: How We Learned to Stop Whining and Improve Security Operations

Sathya Chandran Sundaramurthy
University of South Florida
sathyachandr@mail.usf.edu

John McHugh
RedJack, LLC.
john.mchugh@redjack.com

Xinming Ou
University of South Florida
xou@usf.edu

Michael Wesch
Kansas State University
mwesch@ksu.edu

Alexandru G. Bardas
Kansas State University
bardasag@ksu.edu

S. Raj Rajagopalan
Honeywell Labs
siva.rajagopalan@honeywell.com

ABSTRACT

Efforts to improve the efficiency of security operation centers (SOCs) have emphasized building tools for analysts or understanding the human and organizational factors involved. The importance of viewing the viability of a solution from multiple perspectives has been largely ignored. Multiple perspectives arise because of inherent conflicts among the objectives a SOC has to meet and differences between the goals of the parties involved. During the 3.5 years that we have used anthropological fieldwork methods to study SOCs, we discovered that successful SOC innovations must resolve these conflicts to be effective in improving operational efficiency. This discovery was guided by Activity Theory (AT), which provided a framework for analyzing our fieldwork data. We use the version of AT proposed by Engeström to model SOC operations. Template analysis, a qualitative data analysis technique, guided by AT validated the existence of contradictions in SOCs. The same technique was used to elicit from the data concrete contradictions and how they were resolved. Our analysis provide evidence of the importance of conflict resolution as a prerequisite for operations improvement. AT enabled us to understand why some of our innovations worked in the SOCs we studied (and why others failed). AT helps us see a potentially successful and repeatable mechanism for introducing new technologies to future SOCs. Understanding and supporting all of the spoken and unspoken requirements of SOC analysts and managers appears to be the only way to get new technologies accepted and used in SOCs.

1. INTRODUCTION

Over the years, there have been a number of research efforts focused on understanding the problems of security operation centers (SOCs). The goal of most of these efforts has been to develop useful operational tools [5, 15, 27]. Researchers have conducted interviews and, in some cases, shadowed security analysts to understand human and organizational challenges [4, 30, 31, 32] in security operations. Most of these efforts resulted in recommendations to developers building tools for SOCs. Despite the correct orientation of these efforts, a common feature of these contributions is that they suggest technical solutions to problems without considering contextual factors that may support or hinder the deployment of the solution. A consequence of the lack of a clear understanding of the operational environment is that the proposed solutions are partially successful at best.

We have been conducting an anthropological study of SOCs at two universities and two commercial corporations for 3.5 years. Our aim has been to understand real operational environments. As computer security researchers and tool builders, one of our major goals was to study the effectiveness of tools currently used in SOCs. With the help of an anthropologist, we trained five computer science students with computer security backgrounds in participant observation methods. The students then took jobs as security analysts in academic and corporate SOCs. They took detailed field notes of SOC events throughout their fieldwork. While documenting events, *e.g.*, usage of a specific tool, they also recorded related activities to establish the context for the event. Without the contextual information the intent behind the recorded actions could not be uncovered during the analysis process leaving gaps in our understanding of the event and its handling.

The motivation for any anthropological study is to obtain insights into various activities humans perform within their cultural context. Each SOC has a culture of its own and it is within that culture that the meaning of tools and processes have to be interpreted. *Activity Theory (AT)* as proposed by Leont'ev [20] and further refined by Engeström [9] is used to facilitate our understanding. At the core of AT based modeling is the notion that humans are collective beings and their activities are goal- or objective-directed. Without an objective there is no meaning to any deliberate human activity. AT also models how we use tools to achieve an objective while emphasizing the distributed nature of accomplishment. Thus, the framework proposed by AT is well suited for analyzing work in operational environments.

Copyright is held by the author/owner. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee.
Symposium on Usable Privacy and Security (SOUPS) 2016, June 22–24, 2016, Denver, Colorado.

Our most interesting discovery was the existence of tensions and contradictions within the SOC environments. In the SOC context, we found tensions between the analysts and the tools they used as well as conflicts between analysts and various operating rules. We first model SOC operations as an activity (in AT sense) and then list the multiple levels of contradictions that existed in the SOCs we studied. To the best of our knowledge we are the first to systematically identify and study conflicts within SOCs.

Based on our understanding of the systemic tensions in SOCs, our research reveals that the action-operation dynamics from AT indicate a way to resolve certain tensions, *e.g.*, building tools that automate analysts tasks that have become "operations," *i.e.*, repetitive and boring. This frees analysts to perform more creative analytical actions while also generating new tensions and contradictions in the organization and workflow. This process is on-going and tools need to be constantly adapted in a SOC environment as threats change and events evolve. Analysts move constantly between the *acting* and *operating* stages. This is the reason why "static" or inflexible tools fail in SOCs. Our success stories occur when the tools we co-create with analysts keep evolving to resolve new conflicts. It will become clear in the later sections of the paper that the tensions do not always revolve around operational tools. A tool is one component of a set of forces that interact together creating friction due to certain inherent contradictions.

We form a novel "Pentagon Model," an extension of the hierarchical structure of human activity originally proposed in Activity Theory, to capture the knowledge generation and transformation in SOCs and the proper roles of tools in SOC operations. It provides a novel framework within which developers for SOCs can elicit requirements for their tools. We show that identifying and resolving contradictions is a prerequisite not just to building a useful tool but to implementing any novel idea in a SOC. A tool is part of the larger context of SOC workflow and becomes involved in complex interactions that impact multiple dimensions and domains within the SOC. In this way, a tool is not "just a tool" and must be understood within this broader context.

A 3.5 year journey and a substantial amount of data analysis was required to reach these conclusions. In the rest of the paper, we use one story about building an incident response portal for a SOC to illustrate this journey, and explain rationales behind any methods we used in the research and models/results formulated from the analysis.

2. THE STORY OF THE INCIDENT RESPONSE PORTAL

The incident response portal was built for the first SOC we studied, one managing security for a public university in the United States. It consists of a team of 3 to 4 analysts headed by a manager. Each analyst specializes in tasks such as firewall management, incident response, PCI compliance, *etc.* Due to the small team size, the analysts often have to perform non-routine tasks usually done by other analysts. During our fieldwork the students worked as analysts performing these operational tasks. Before continuing, we need to explain our core anthropological research method, *participant observation.*

2.1 Participant Observation

Understanding security operations requires access to operational SOCs and the cooperation of the analysts who work in them. This access is not easy to obtain for reasons that include:

- *The sensitivity of the data handled.* Analysts deal with exploits that can result in loss of valuable information, compromise the privacy of users, or physical damage to infrastructures. A degree of paranoia seems to come with the job. With the academic research literature's current focus on discovery and public disclosure of vulnerabilities, researchers are seen as untrustworthy outsiders. Gaining the subjects' trust is a first step towards performing useful research. Management support is necessary, but not sufficient.

- *The problem of tacit knowledge.* The job of a security analyst is highly complex and decisions are made based on intuitions and hunches that are not documented [26]. In many cases, analysts are unable to articulate what they know or describe clearly the basis for a conclusion or action.

- *The workload.* SOC analysts are always confronted with more incidents than they can resolve. Any process that requires additional efforts but does not directly help the analysts' job is resented.

These factors limit the utility of traditional research methods such as interviews, questionnaires, and passive observation.

Cultural anthropology is a branch of anthropology aimed at studying human beings in their natural settings. The research method employed by cultural anthropologists is *long-term participant observation* in which researchers traditionally spend a year or more within an indigenous population as a member of the community. They take part in the day to day activities and follow the practices of the population. This allows them to obtain an increased understanding of local practices beyond common assumptions about such practices. As they pull themselves deeper into local practices they come to feel and experience the world and may eventually be able to approximate the *native* point of view, in other words, understand how an insider perceives their own culture. This leads to the researcher understanding the symbols, artifacts, and activities as they are perceived by the members of the subject community. Without this understanding, an observer tends to process every event performed by the subjects using the observer's own cultural bias. Such a bias does not lead the researcher to the true reason behind the observed activities [14]. Viewing or attempting to view the activities from the native's point of view is the best one could do in understanding another culture.

The idea of attaining the native point of view resonated very well with our goal of studying security operations because of the well defined closed culture of the SOC. We sought and obtained the cooperation of the SOC management. Our team anthropologist trained five computer science students having a computer security background in participant observation methods which included the observation and note taking that would occur during the fieldwork process.

Over a period of 3.5 years our students occupied positions as security analysts in *four* different SOCs, two in universities

and two in major corporations, a deployment that continues part of the ongoing research effort. The student researchers have worked as level-1 & 2 analysts, incident responders, software-developers, and forensic analysts. They have helped in training security staff and designing security policies, becoming something like "natives" in the SOC cultures, while also keeping detailed notes about their experiences and ongoing SOC activities.

2.1.1 Ethics and Participant Safety

In our research the security analysts and the managers were considered as human subjects. The research was reviewed and approved by the Institutional Review Board (IRB) and analysts completed informed consent forms that explained the research objectives and the voluntary nature of participation. We addressed any concerns expressed, with a detailed description of the nature and expected outcomes of the research. We used aliases when referring to analysts and their managers during discussion and data analysis to preserve their anonymity.

2.2 Why Build the Tool

Early on in our research we observed that the bulk of the analysts' time is spent responding to security incidents reported by external third party entities. The most common of those incidents is malware trying to connect to its command and control (C&C) server. The third party provides the university with information containing the type of malware, the IP address on which the malware activity was observed, usually that of the external interface of the NAT firewall, and the time at which the activity was detected. All this information is sent as an alert via email messages. The responding analyst has to follow the following steps in sequence.

- Identify the internal IP of the infected client from the firewall NAT logs.
- Use the internal IP to identify the MAC address of the infected host from DHCP and/or ARP logs.
- Look up the identity of the user of the infected device using the MAC address from the authentication logs.
- Determine the point of contact (POC) for the incident based on the location of the user (*e.g.*, a department).

Once the analyst obtains all or most of the information, he recommends a potential remediation measure (*e.g.*, format the host disk and re-install the OS), and then puts all the information into a ticket and sends it to the POC. The owner of the infected device also gets a notification about the infection and the recommended remediation steps.

This seemingly simple task is laborious and time consuming. No single tool available at the SOC can provide the direct answer to the question "who is the owner of the infected device," even though the correlations from the multiple logs are straightforward. The deployed security information and event management (SIEM) solution was very slow even for searches on a single week's data. Discovering correlations in the data within the SIEM was almost impossible due to its unacceptably slow performance. The analyst had to manually inspect multiple logs for each of the alerts and it took 10 minutes (on average) to correlate the logs and file a single ticket. The SOC received approximately 15 such alerts per day. It was obvious to our student researchers that the analyst got *burned out* by this repetitive task as did the student researcher tasked to do the same job. He felt that his time was spent on meaningless activity and that he was doing nothing interesting. Further aggravating the situation was the manager's insistence on detailed documentation of the manual method (by the student) so that anyone could perform it.

2.2.1 Reflection on the Process

At this point the student became frustrated by the repetitiveness of his SOC job. This is the moment at which he started to gain the *native point of view* as an analyst. Just as our student researcher was feeling that he had lost the direction of his research, he and the whole research team engaged in a *reflection process*, where the field worker discussed his problems with the rest of the research team. Through this process, we realized that these specific problems can be addressed by building a custom tool for responding to this type of incident. It was clear that this insight arose because the student had reached an essential native point of view unattainable through other means such as interviews. At the same time, it was clear that the student brought uncommon skills, *i.e.*, tool building, to the analyst position.

2.3 How the Tool Worked

In the reflection process, we identified steps in this repetitive process that could be automated. For the malware incident described above the task of a security analyst could be decomposed to answering the following set of questions.

- *What* - Type of threat reported.
- *Who* - Users, IP address, security personnel, *etc.*
- *When* - Time the threat was reported and other temporal information.
- *Where* - Location of the infected device in the network/organization.
- *How and Why* - Context that could have raised the alert, perhaps the most important and interesting.

The analyst was stuck in this process because he was spending more time gathering the basic information such as *who and where* rather than on establishing the context – *how and why*. Our realization was that tools must gather and deduce information along the four basic dimensions of information (what, who, when and where) so that the analysts could spend their cognitive effort along the analytical dimensions (how and why). This insight helped us build the incident response tool.

2.3.1 Automated Incident Response

Together with the analysts we built an incident response portal based on this insight. We used a database to store log information and collected and parsed logs using periodically executed scripts, making the process more efficient. The database also contained a relationship between net blocks and the POC that allowed the notification of the responsible incident response personnel.

The tool has a web interface through which the analyst enters: (1) the external facing IP address and port number where malicious activities were reported; (2) the remote IP address and port number involved in the activities; (3) the timestamp and time zone when the activities were observed. The tool correlates this information and presents the analyst with a filled-in incident ticket with all the required information such as the user of the infected device and the POC.

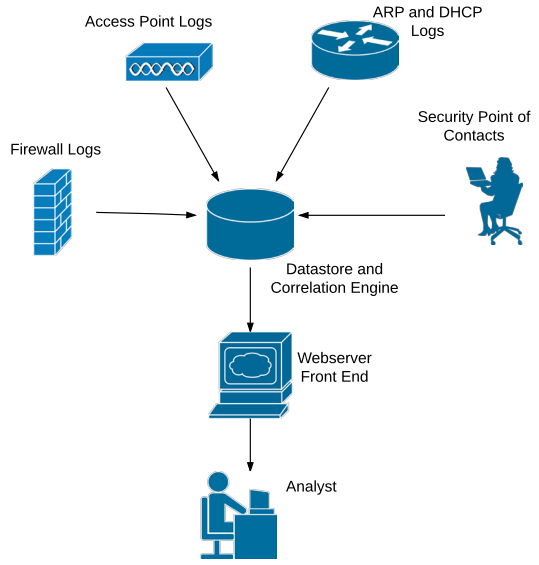

Figure 1: The Incident Response Portal

The analyst then performs the analytical steps answering *how* and *why* the incident might have occurred in the first place. He then suggests possible remediation measures and submits them to the ticketing system. **Using our tool the whole incident response process was reduced from 10 minutes to 10 seconds.** The time saving is due to the automation of the old tasks of manually searching the various logs to establish the *who* and *where* aspects of the incident, now done through automated database queries using the information entered into the web interface. Figure 1 shows the basic workflow of the tool. **This appears similar to a SIEM workflow yet none of the SIEM products that we found in the SOCs provides the automation provided in our incident response portal.**

This shows a major problem in the design methods used for security products. Without understanding the workflow of a SOC and where the friction points are, a tool is useless. Our tool was quickly adopted by our SOC analysts. It not only resolved a major bottleneck in the SOC's workflow, but also broke a major trust barrier for our student fieldworkers. After this tool was successfully built and used by the SOC analysts, the analysts immediately became more open to discussing other challenges in their work to our fieldworkers, and sought our help in building other tools that ease their job. This tool co-creation process was our first major finding in our 3.5 years' anthropological study [26].

2.4 What Happened Afterwards

After this initial success we identified a number of other problems in the SOC that can benefit from automation. The research team developed a number of tools to automate those recurring analyses. The tools were well received and the SOC process was more efficient than before.

Our research went on and we conducted fieldwork at three additional SOCs – another university SOC and two corporate SOCs. Unlike the university SOCs, the corporate SOCs were highly hierarchical. Analysts in one corporate SOC are classified as level-1 (L1, lowest level), level-2 (L2), and incident response (IR, highest level). In this SOC, one of the students worked as L1 and IR analyst while at the same time developing some forensic analysis tools. The other corporate SOC outsourced its L1 tasks to a third party and our student fieldworker took the role of L2 analyst. The corporate SOCs had more analysts (around 22 L1s, 2 L2s, and 5 IRs in one SOC) compared to the university SOCs. Analysts in the corporate SOCs had well-defined roles while in the university SOCs they always had to engage in cross-training and wear multiple hats due to small team sizes.

Through this additional field work we identified the cause of burnout in SOCs using Grounded Theory [25]. We identified the vicious cycles among a number of human, technical, and managerial factors that lead to burnout. We also found a few cases where the vicious cycles were turned into virtuous ones thus mitigating the burnout. In some of those cases the automation of repetitive tasks resulting from tool co-creation was the key enabler.

When the student researchers returned to the first university SOC after a few months, they found that the incident response portal had been rarely used in their absence. We realized that lack of support for the tool was the cause for concerns. New requirements kept emerging and the analysts in the SOC analysts had neither time nor the skills required to customize the tools as the requirements evolved. We then realized that there was more to the success or failure of the tools beyond their technical features.

3. FURTHER ANALYSIS OF THE FIELD WORK DATA

Our experience with the incident response portal encouraged us to return to our field notes and dig deeper to further understand the role of tool building in SOCs and whether there is a guiding principle that could allow us to replicate the success we had in terms of building successful tools to help SOC operations. After six months of analysis, we discovered that an adapted version of a well known model called Activity Theory can form the cornerstone of this guiding principle.

3.1 Activity Theory

The origin of Activity Theory (AT) is found in the works of the Russian psychologists Leont'ev [20] and Vygotsky [28] during the 1970s and 1980s. AT has a proven record of helping researchers comprehend various challenges in work environments. For example, it has been used to study the use of technology in educational environments, to understand the changes brought on by introducing new technology (laptops) into teaching practices [7], and to study the differences between the teachers' beliefs and actual practice when a new tool is introduced in learning [6, 22, 24]. Researchers used AT to understand the effect of new tools on learners, especially their resistance to newly introduced technology for learning, and on highlighting how old habits impede the adoption of new tools [2].

The AT model in Figure 2 is adapted from Engeström [9]. Elements of the original model are shown in parentheses and in red font. Un-parenthesized elements result from our application of the model to SOC operations. Engeström defines an activity system as *object oriented, collective, and culturally mediated human activity* [12]. The fundamental

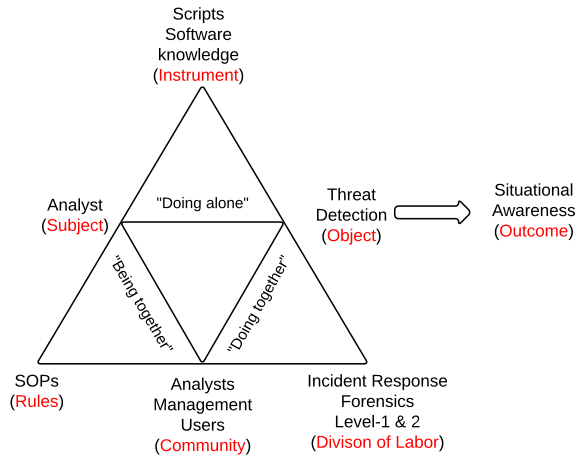

Figure 2: Activity Theory Model of Security Operations

idea of AT is that humans perform tasks to achieve an objective. Without that objective the task has no meaning. The inner downward-pointing triangle symbolizes the interactions of individuals and the collective in achieving an objective. Each edge in the downward triangle represents the relationship between the three nodes [9]: (a) an individual does certain tasks to achieve an objective, (b) an individual is part of a social structure represented by the community node, and (c) the community of which the individual is a part of acts together to achieve an objective. Furthermore, the three relationships are mediated by three different aspects – instrument, rules and division of labor, forming the encompassing upward-pointing triangle. In trying to accomplish their objective humans use certain tools or in AT terms *instrument*. The tools can be physical, such as a hammer when breaking rocks, or symbolic such as language for communication. AT further states that human beings do not act in isolation but within a community. There are certain rules that govern interactions among the members of the community. In order to achieve their objective, people take up different roles (division of labor) based on their expertise.

According to AT, tool mediation – design, use, and preservation of physical and symbolic instruments – is seen as a major distinguishing factor between human and animal activities [9, 17]. The two triangles in the AT model together represent three different types of mediated interactions [17]: (a) subject-object interaction is mediated by Instrument, (b) interaction of subject with their community is governed by Rules, and (c) a community achieves their objective by taking up specific roles corresponding to Division of Labor. The three different mediations arise due to social, cultural, and cognitive aspects of human life.

A SOC can be modeled as an activity system where the subjects are the analysts and their objective is to monitor/mitigate threats and provide situational awareness. To achieve this objective they use tools such as SIEM, homebrewed software and scripts, and their knowledge in computer security. The community they interact with includes other analysts, management, and end users. The traditional rules governing the communication between analysts and other stakeholders are the so-called standard operating procedures (SOP). SOPs recommend course of action for every incident type guiding the analyst in drafting a communication and mitigation plan in response to a security incident. Analysts also assume roles on the operations floor, *e.g.*, level-1 (junior) analyst, level-2 (journeyman) analyst, incident responder, forensic analyst, *etc.* Under this interpretation, it is easy to see that a SOC work environment fits nicely within the AT framework.

AT has been successful in understanding distributed human activity ranging from primeval hunting to modern day work environments. So it is natural that SOC operations can be successfully captured by the AT model. AT also sheds light on the use of tools by humans in achieving their goals in collaborative activities. Since one of our goals were to obtain insights on the role of tools in SOC operations, it further convinced us to use AT to drive further analysis.

3.2 Analysis Methods and the First Result

Throughout the 3.5 years of fieldwork spanning 4 SOCs we observed many recurring patterns and similarities in their problems. Due to the large amount of field note data, a systematic approach is needed to ensure the objectivity and comprehensiveness of the analysis.

Our analysis of field note data is both inductive and deductive. It is inductive in the sense that we look for patterns in data *without* any preconceived hypothesis. As we formulate theories to explain the patterns we found in one part of the data, we also test those theories on the other parts of the data. In this sense our analysis is also deductive. To facilitate this type of analysis, we leveraged a qualitative data analysis technique called *template analysis*.

3.2.1 Template Analysis of Data

Template analysis is a qualitative data analysis technique developed by Nigel King [18]. It is useful when the researcher has a partial understanding of the concepts to be identified in the data. This technique starts with an *a priori* set of codes or themes that the researcher is interested in and the codes evolve as the analysis is performed. The technique is flexible in that the researcher starts with some preconceived concepts but can also identify and add new concepts as they are discovered. Below are the steps in the template analysis process.

Define a priori themes A set of themes are developed based on the concepts the researcher is interested in identifying in the data.

Transcribe and familiarize The researcher reads through the field notes and familiarizes herself with the data she is going to analyze.

Initial coding Parts of the field notes that are relevant to the research questions are identified. Then the *a priori* codes are attached to those parts of the data wherever they are applicable. When a section of fieldnote data matches the research question but no existing code could be applied, a new code is devised or an existing one is broadened to cover it.

Produce initial template Once a subset of the data is coded a set of themes is generated. These form the initial template. The template might have a hierarchy of codes within each of the themes.

Develop the template The initial template is applied to the entire data set repeatedly. Modifications to the template are performed whenever a text does not fit

into the template. This iterative process refines the code set and a final template is produced.

Interpretation At this point, the researcher has coded the entire data using the developed template and writes up her findings based on the final template.

Quality checks and reflexivity The researcher periodically consults with an expert team that includes fellow researchers on the project to ensure quality of the analysis she performs. The coding researcher must also perform frequent reflections to make sure her own personal beliefs and biases do not affect the interpretation of the collected data.

A study by Frambach *et al.* exploring the effect of globalization on medical education provided the inspiration for combining AT with template analysis [13]. Following this work, we began by looking for the basic elements of the AT model in our fieldwork data and found that the model provided substantial explanatory power for understanding work carried out in SOCs. We then applied more concepts from the AT theory to further understand the data. Thus we first developed a list of codes based on the AT model and performed data coding. New codes were added as new themes emerged. This continuous application of template analysis eventually resulted in **one of our main discoveries in this paper: the existence of contradictions in SOC operations and its key role in preventing SOCs from doing an effective job.**

4. CONTRADICTIONS

A key feature that arises when using AT to study work environments is the notion of *contradictions*. From AT perspectives, contradictions are defined as "a misfit within elements, between them, between different activities, or between different developmental phases of a single activity" [19]. Some researchers have referred to contradictions as *systemic tensions* [1]. Other definitions include "unintentional deviations from the script [which] cause disco-ordinations in interaction" [11] and "problems, ruptures, breakdowns, clashes" in activities [19]. Engeström [10] even recognized contradictions as "the motive force of change and development" [12]. In a typical scenario when contradictions arise, individual(s) begin to question the established norms and start to deviate from the rules. A positive outcome is that individuals get together and develop a new course of action that resolves the original contradiction leading to a better workflow [10].

4.1 Primary Contradictions

A tension that exists within a single node in the AT model (Figure 2) is called a primary contradiction [9]. In a work environment, these tensions arise due to the dichotomy between the "professional logic" of the employees and the "commercial logic" imposed by their organization [3]. The professional logic of security analysts (subject) dictates that they constantly improve their skills and be efficient in detecting and mitigating security threats. On the other hand, SOCs are under constant pressure to demonstrate their value to the parent organization. This results in a number of metrics being defined to measure the performance of SOC analysts. Ultimately, the job of the analysts is skewed very much towards generating those defined metrics. This creates a conflict within them. **They are confounded with two non-identical objectives – doing the *right thing* versus the *required thing*.**

Returning now to the incident response portal story, the analysts' frustration was caused by a conflict between their desire to continuously improve their skills and thus wanting to handle more intellectually challenging incidents, and the fact that SOC management emphasizes metrics such as number of resolved incidents instead of the complexity or subtlety of the incidents. As an analyst one has to tend to both these objectives which are often in conflict with each other. The analyst can choose to close a high quantity of easy tickets (thereby scoring high marks on managerial metrics) or attend to more complex incidents that may be more fulfilling. This leads to frustration and eventually burnout. This contradiction is faced by the analysts within themselves; that is, it is a contradiction that exists inherently in the "Subject" node of the AT triangle of Figure 2.

We went back to our field notes to find more examples of such primary contradictions. Following the template analysis methodology, we coded our data with the initial goal of identifying contradictions in the SOC's operations. The initial template generated as a result of coding a subset of the data is shown in Table 1 in Appendix A. The initial template was then used to code the entire field notes, resulting in the final template which was used to interpret the results. Below we illustrate some findings from the analysis.

4.1.1 Primary Contradiction within Subject (Analyst)

In addition to the frustration we witnessed in the first SOC, this primary contradiction within analysts is observed across the SOCs we studied. One analyst in a corporate SOC noted:

> "I wanted to work in an environment where there will be continuous learning and I have started to feel that I am not learning anything new in my current job. In fact, I took the current job hoping to analyze malware every day and learn more in that process. I feel that the SOC currently is not doing any real threat detection which in turn is limiting my opportunities for learning. I have decided in my life, to spend a significant amount of time for the improvement of my career. Now I feel bad that my commitment is not paying off."

In another instance a SOC manager asked his analysts to work towards generating metrics:

> "There will be metrics collected for all analysts from the case management tool (CMT) so that a report can be generated and shown to the upper management. If the team has to scale, handling a number of cases, we need to produce numbers to show to upper management. So far this is being done through success stories and this does not scale as it looks very general. Some part of the management is also interested in knowing the impact our team has on the infrastructure. Go over the metrics and say which ones make sense and do not. You have to live with it and get involved. If you do not get involved now then when the change is made into CMT you will have to provide the data. I do not want to push it out

there without questioning and for the sake of doing it. I also want to measure the fidelity of the incident. Features in CMT that do not lead to any metric must be removed."

4.1.2 Primary Contradiction within Instrument (Tools)

Security analysts use a number of tools to perform their job. Some of them are physical such as software and scripts, while others are cognitive, such as knowledge and training. In an ideal case tools will help analysts become efficient in their job. From the professional-logic perspective this is the true purpose of a tool. Interestingly, the tools in operation floors are purchased instead due to reasons not aligned with efficiency. Typically the most expensive product in a SOC, SIEMs are purchased because they are considered information security "best practice." Ironically, most of the SIEM solutions we saw deployed at the SOCs were not up to the task of basic event correlations necessary for incident analysis, as illustrated in our incident response portal story. **Here the commercial logic for having the tool is compliance not operational efficiency, resulting in this primary contradiction.**

> In one of the corporate SOCs, the management decided to use a particular case management system (CMS) due to the support it provided with the existing SIEM solution. While the integration seemed helpful at the beginning, the CMS turned out not to fit the workflow of the SOC. The CMS was never replaced, which subsequently lead to secondary contradictions with the analysts (Section 4.2.2).

4.1.3 Primary Contradiction within Rules (SOPs)

As we noted earlier, the rules in SOCs are the standard operating procedures, or SOPs. The purpose of SOPs is to make sure for a given incident every analyst will respond in a similar way. In other words, they ensure predictability in operations. However, there is a fundamental conflict that SOPs face which is between expected behavior and creativity of analysts. Security operation is a distributed activity involving a number of analysts. If they are encouraged to act their own way *all the time* there will be chaos. On the other hand, one does not know when to deviate from the norm and try out new techniques. This inflexibility hinders detecting and mitigating threats which are constantly adapting. This dualism is at the core of the conflict that exists within the SOPs used in operations.

For example, an analyst encountered an operational scenario where he had to email a member of a business units to validate an alert but was very hesitant to proceed. After waiting for a while he contacted a senior analyst and asked him for advice on how to proceed. The junior analyst specifically said that he did not know how to proceed as this scenario was not covered by any of the procedures. This example demonstrates a familiar problem we encountered throughout our fieldwork. While SOPs can empower an analyst within limits, the same SOPs can dis-empower the analyst from acting beyond them.

4.1.4 Primary Contradiction within Division of Labor

In work environments, the division of labor is achieved by assignment of roles to employees. In a SOC typical roles include level-1&2, forensics, incident response, and content development engineer. The role assignment ensures that people have the right skills and expertise for the assigned task. There exists a dualism within division of work that leads to efficiency problems. The very specific role assignments to analysts leads to analysts working in silos; thus they often lack empathy for other analysts. On the other hand, analysts have to constantly work with their colleagues in other roles; the lack of empathy creates barriers in this collaboration, thus fundamentally defeating the purpose of division of labor.

For example, a level-1 analyst was frustrated about the high volume of events generated by a rule written by a level-2 engineer:

> "The engineering team is very stubborn. Jack (name changed) thinks that he knows everything and does not understand the frustration of analysts."

Likewise, upper-level analysts become frustrated by those in lower levels. Level-1&2 analysts escalate incidents to incident response teams whenever they require assistance. One day the incident response team members complained that they were getting too many escalations. Having worked at both teams the fieldworker found the two teams to be completely unaware of the priorities, problems, and concerns of each other.

4.1.5 Primary Contradiction Within Objective

Finally, there is also a primary contradiction within the objective of the SOC itself. The primary objective of the SOC as commonly understood is to detect and mitigate security threats for their parent organization. Perversely, the better a SOC gets at detecting/preventing threats the harder it becomes to show their value to the organization.

In one of the corporate SOCs alerts that were insignificant were deliberately left unoptimized as optimization would reduce the number of alerts in the stream. Fewer alerts would then mean that management would perceive that the SOC team could do their job with less number of analysts and the parent organization would then put pressure on the SOC management to reduce the team size by laying off some of their analysts. As a result analysts have to deal with a large number of useless events and eventually get worn out.

4.2 Secondary Contradictions

The existence of primary contradictions will also create conflicts *between* elements of the AT model. In AT these are called *secondary contradictions* – tensions that exist between any pair of nodes in the AT triangle of Figure 2. They are a manifestation of the inherent primary contradiction within the single nodes [3]. Our template analysis revealed a number of pair-wise contradictions in SOCs.

4.2.1 Subject - Rules

Throughout our study we observed a constant tension between the analysts and the standard operating procedures (SOPs) they are required to follow. A security analyst wants to solve intellectually challenging security incidents. This requires using novel analysis methods that are not in the SOPs. The SOP rules do not provide enough freedom for an analyst as there is a written down procedure for every

type of incident. The mundane nature of executing procedures time and again hinders creativity. The rules define the tasks of the analyst based on the opinion from the management. The SOC management wanted the SOPs on the argument that SOPs help ensure predictable performance of the SOCs (commercial logic). But at the same time this prevents the analysts from being creative in their jobs (professional logic), and thus prevents them from being more efficient in operations. The secondary contradiction, *i.e.*, the conflicts between the subject (analysts) and the rule (SOPs), is a manifestation of the primary contradictions inherent in the analysts and in the SOP which we discussed before. This secondary contradiction is also a main cause of the frustration at the first SOC we worked at that eventually led us to develop the incident response portal to help address.

As one analyst in another SOC complained:

> "The procedures were turning us into robots. The procedures were so detailed at some point that all the analysts were doing was to click and fill in data."

If not tended to, this contradiction has been found to cause adverse effects such as analyst burnout leading to frequent turnovers, as pointed out in our prior work [25]. Periodic review of rules to identify patterns that could be automated is one way to mitigate the effects of this contradiction, but this is not done often enough (or at all) in most SOCs.

4.2.2 Subject - Instrument

From the perspective of technology transfer, this contradiction is the most interesting to explore as it involves interaction of analysts with technology. The SOCs we studied did not have the right tools to help their analysts as most of the tools were developed without proper understanding of the analysts' workflow. A top-down decision was made by the management on the type of tools to be procured for the SOC. This is essentially a manifestation of the primary contradiction within the tools (Section 4.1.2). As a result SOC tools have suffered from a number of shortcomings.

One of the major concerns about the tools in SOCs pertains to *poor attribution*. To make the best decision a security analyst must be provided with all the temporal and spatial information related to an alert. The purchased tools were designed with no knowledge of operational workflows and thus completely missed this aspect. Analysts were provided with partial information making it hard to attribute the alert to an owner or a device. In another case we observed, analysts were not able to query the wireless domain controller to extract the authenticated user IDs along with the device host name because the vendor had not anticipated this need and decided not to provide that feature. Such shortcomings result in analysts spending most of their time performing low-level data processing tasks to gather the missing information, rather than creative investigation.

4.2.3 Division of Labor - Instrument

A SOC is comprised of analysts with specific role assignments. In order to achieve the goal of division of labor, where analysts perform the tasks they are good at, it is imperative that they have the right tools to assist them. The preference for features in a tool depends upon the role and technical expertise of the analyst. A forensic analyst might like to use a Linux desktop and might be comfortable using a command-line interface. A compliance analyst whose primary task is to check for conformation of systems to rules might be comfortable only with a graphical user interface (GUI). Tools are oftentimes purchased based on the managerial logic that interferes with the preferences and requirements of the analysts (Section 4.1.2). As a result, analysts in different roles could not accomplish their tasks and the purpose of dividing work based on expertise is defeated.

This contradiction is well illustrated by our story with the incident response portal. After the tool was built, the process of responding to the malware incidents was simplified to the point that it could be handed off to the Network Operations Center (NOC) of the university. The NOC analysts were less skilled compared to the SOC staff and their job was to handle cognitively less intensive tasks. Our tool, however efficient in handling malware alerts, was not ready to be used by the NOC staff simply because it used a command line interface. The conflict our tool ran into was between Division of Labor (skill set of analysts) and Instrument (tools they had to use). The incident response portal exposed an interface that required more cognitive work than the NOC analysts are comfortable with. As a result, the SOC's effort to transfer this task to NOC did not happen for a long time, and the more skilled SOC analysts were still stuck performing the mundane ticketing task for malware incidents (though more efficiently than before).

We resolved this contradiction by providing an alternate web interface to the portal in addition to the command line access for SOC staff. The web interface abstracted away a number of technical tasks and pushed them into the background. The NOC staff were then able to file malware tickets at the push of a button. Clearly, the same tool needs to have multiple interfaces depending on the type of analysts who will be using it. Otherwise one cannot get the expected benefit of distributing work among analysts.

It is important to note that this is also an example of how an attempt to resolve one contradiction may create a new contradiction. The tool was originally designed to improve the work of SOC analysts, but it ultimately had an impact on the division of labor, being accepted as a tool for the NOC. But here the tool failed because it had been designed with a command-line interface for the SOC. This highlights the fact that **conflicts will keep emerging in a SOC no matter how much you can do to improve its process. Such conflicts must be resolved on a continuous basis.**

5. FROM CONTRADICTIONS TO INNOVATIONS

The previous section discussed the contradictions we identified in SOCs during our anthropological study. Each contradiction requires a different course of action to be resolved. Some measures are technical while others are managerial. The rest are influenced by economic considerations. This leads to a question of particular interest to the audience of this conference:

Can technologists do something to turn some of the contradictions into innovations? If so, how?

Contradictions are at the heart of Activity Theory and they are the potential triggers for workplace innovations [9, 12]. When we looked back at our fieldwork data we realized that

it was by identifying and resolving certain contradictions that we succeeded in bringing an innovation to security operations.

Let us return once again to the incident response portal story. The analysts were stuck performing a high volume repetitive task. Neither the analysts nor the field workers could invest time in any creative security projects because the repetitive malware incidents had to be taken care of as high priority. The analysts would get penalized if they did not close the malware tickets in a timely fashion as required by their manager. They have to balance between two conflicting motives of their job: engage in creative security analysis, and resolve the constant stream of incoming security alerts. The presence or lack of the right tool will either reconcile or aggravate the two contradictory motives. The incident response portal we built resolved/mitigated a number of contradictions manifested in this story.

Our tool was built in the context of the SOC environment and hence fits the operational workflow. Our tool development process is *analyst-developer co-creation*. In this model the fieldworkers are also analysts themselves, and they engage in developing tools that aid in analysts' work. As fieldworkers, we switched hats between developer and analyst to enable co-creation within ourselves. This addressed the secondary contradiction of tools falling short of analysts' expectations (Subject - Instrument in Section 4.2.2). The incident response portal reduced the ticketing time from 10min to 10sec, allowing the analysts to close the immediate incidents more quickly. As a result they will have more time for creative analysis. Therefore the incident response portal mitigated the primary contradiction within the analysts (Section 4.1.1), since they can now more easily balance the two conflicting objectives of their job. The tool also mitigated the primary contradiction within the tools (Section 4.1.2). While the SIEM used by the SOC (considered a must-have due to "best practice") was not up to the task, the incident response portal bridged this gap by introducing some real value (helping analysts in their job) into the SOC's tool box.

We continued to conduct template analysis on the field notes to revisit all the cases when we built tools for SOCs. Every one of them confirmed to us that **the reason the tools we built were adopted by a SOC and became useful was because they all helped resolve some contradictions in the SOC. They will keep being useful and used by the SOC as long as we continue updating the tool to resolve new contradictions as they emerge (including contradictions that emerge in part due to the tool itself). If we stop the process of identifying/resolving contradictions, the tool will stop being used in the SOC.**

After combing through all the success and failure stories of our tools in the SOCs we studied, we further realized that the process of resolving contradictions in a SOC can be placed in the proper perspective by looking at another important aspect of activity theory – the dynamic nature of activity.

5.1 Human Activity Dynamics

Humans performing an activity operate at multiple cognitive levels in achieving their objective. We use an example by

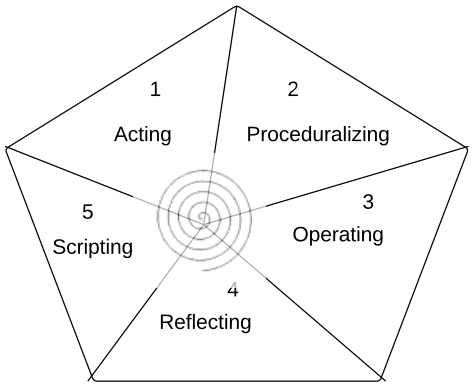

Figure 3: Pentagon Model for Knowledge Transformation

Kaptelinin *et al.* to give further insight into this hierarchy of activity [17]. The example sheds light on the non-stationary nature of the hierarchy, *i.e.*, the hierarchy evolves over time and the importance of specific actions shifts. Consider the activity of learning to drive a car. For the first few days, the learner consciously performs tasks such as changing lanes, looking in the mirrors, and shifting gears. Each of these in AT terms is called an *action*. Broadly, human tasks that require explicit attention are categorized as actions. The high level of cognitive effort required by each activity prevents the learner from multitasking during the learning period. With practice and continued instruction, the actions become second nature and can be performed subconsciously. At this point, they become internalized and are now called *operations*. The cognitive effort needed to perform operations is almost negligible, thereby enabling multitasking. The ability to perform operations persists even after years of non use. One never forgets how to ride a bicycle[1]. We refer interested reader to a more detailed discussion in Appendix B.

We now look at the process that we carried out when turning some of the contradictions in security operations into innovations, through the lens of the hierarchical model of human activities.

5.2 Activity in Security Operations

We found the action-operation dynamics to be applicable to tasks performed by security analysts. Steps such as log analysis, filing incident tickets, and communicating with stakeholders, when performed consciously by an analyst can be categorized as *actions*. After repeated applications these steps can be internalized within an analyst and be performed with very minimal cognitive effort, at which time they become *operations*. Our template analysis revealed that the action-operation transition in SOCs involves some interesting aspects of knowledge transformation. Specifically, **our analysis identified three additional stages in this transition that are *not* present in the traditional AT literature**. Figure 3 shows what we call the *pentagon model* for knowledge transformation in SOCs. The five stages of activity repeat as a cycle; each stage is described below.

Acting Analysts in the acting stage are handling a new se-

[1]There is a neurological explanation for this. See `http://www.abdn.ac.uk/news/3275/`.

curity incident, *e.g.*, a zero day or previously unidentified incident. A new incident does not have an SOP or other written procedures for its handling. As a result, the analysts have to consciously perform each step of the investigation. This stage requires a creative mindset and demands a high cognitive effort from analysts.

Proceduralizing Once analysts understand the incident, they develop a procedure for handling similar incidents. Documentation needs to be written describing the procedure. This ensures that other team members are aware of the new incident handling process and preserves the knowledge. This is one of the newly identified stages of the activity hierarchy. Because documenting the procedure usually requires multiple iterations and is a cognitive activity distinct from handling the original incident, it deserves its own place in the hierarchy.

Operating The operating stage occurs when the procedure for handling the new incident is mature and predictable enough for the analyst to perform it subconsciously. There is a self-contradictory nature to this stage. On the one hand, the cognitive effort needed to perform the procedure has become minimal or nonexistent. On the other hand, when the analysts are in the repetitive operating mode (for periods of days) they do nothing creative. This can lead to severe problems such as burnout [25] and partially explains the high turnover rate among SOC analysts, unless a separate set of people with suitable personalities are tasked with this job.

Reflecting This is the second of the three new stages of the SOC activity we identified. Reflection is a process whereby analysts identify aspects of the operational tasks that have become repetitive and require little or no cognitive effort. These are candidates for automation or for delegating to a lesser skilled organization. In a highly efficient SOC, this is performed as often as once a month. We have observed operational environments where no reflection takes place. Analyst burnout and a high turnover are more common in these environments.

Scripting In the scripting stage analysts, either themselves or by working with a development team, automate aspects of incident handling that have been identified as candidates for automation in the reflection process. Usually these are scripts written in rapid development languages such as Python or Ruby. However, implementation can also be done via long-term developmental efforts using web frameworks or coding in a lower level language. This is the third new stage we identified in the SOC activity.

5.3 Automation and Conflict Resolution Revisited

Every new analytical task starts being performed consciously by an analyst (*acting*). The task then, after some stabilization, is documented as an SOP (*proceduralizing*). The stabilized task is eventually internalized by analysts (*operating*). Most SOC managers and analysts stop at this stage. As explained in the previous sections, this will result in primary and secondary contradictions within and between analysts, their tools, and SOPs, leading to frustration and burnout. Let's look back at the contradictions we saw in the incident response portal story. The analysts got frustrated and burned out because they were stuck in the operation stage and did not have any time to think about new threats and problems. Automation of the repetitive operations resolved this contradiction and allowed the analysts to move from the operation stage to the acting stage. This also allowed for the analysts to be more prepared to deal with new threats.

Unfortunately, our fieldwork finds that the process of incremental automation in SOCs is predominantly reactive. Scripts are written only in response to high workload, such as when the volume of an alert stream is too high. We propose that senior analysts and managers should conduct periodic reviews of analytical tasks and identify those that have been *operationalized* within the analysts. In other words, the review should focus on identifying aspects of SOPs that have become cognitively repetitive for the analysts. Those tasks could then be automated *proactively* by either the analysts or software developers with the requirements provided by the analysts. Our incident response portal is an outcome of such a process. Tools created this way will fit well within the cognitive analytical process of analysts and free them to perform more creative tasks.

The pentagon model is also well aligned with the nature of detecting and responding to cyber threats. The variety of security threats evolve rapidly these days demanding creative analysis. Analysts must remain in the conscious *acting* stage as much as possible to be effective. Tools developed following the pentagon model are not static. The constraints that determined the requirements of the tool might change creating new conflicts. This will first push the tasks back to the acting stage demanding manual intervention by analysts and developers. Using the co-creation process, the tool can be adapted to resolve the new conflicts by going through the reflecting and scripting stage again.

Implications of Pentagon Model for Analyst Burnout

The net effect of the cycle in the pentagon model is to recognize that a new incident serves as a potential harbinger for a flood of similar incidents to come. Converting its mitigation from a challenging cognitive task to something that can be offloaded or automated, frees the more capable analysts to meet the next challenge. Thus the cycle repeats. There is another potential problem that we identified in the model – *the rate of transition from the scripting to the acting stages.* If the arrival rate of new incidents exceeds the rate of the cycle time in the model, burnout may occur despite the cognitive challenges, due to the lack of time to automate the operation. If the arrival rate of new incidents is much lower than the rate of the cycle time, burnout may be supplanted by boredom which also leads to a high turnover.

6. TOOLS AND BEYOND

The incident response portal was part of a broader workflow innovation process. The tool would have no meaning if one removed the objective the SOC wanted to attain using that tool. The SOC wanted to implement a hierarchy in the operational workflow. Its staff is composed of highly skilled analysts but a small team. They wanted their job to be-

come investigating *novel* incidents and devising mitigation plans to deal with similar events in the future. They could then write down an SOP document listing out the steps that should be taken to respond to each of the novel incidents. Once the response steps have become stable enough and highly repetitive, they can then transfer it to teams composed of less skilled analysts such as the Network Operations Center (NOC). This ideal did not happen until our fieldworkers helped the SOC identify and resolve a number of contradictions in their workflows by building the incident response portal.

It is within this background that the development and deployment of operational tools must be viewed. Hence it is appropriate to say that resolving contradictions is a prerequisite for not just developing successful operational tools, but to implement any novel idea in SOCs. And due to the complex activity system in which tools and new ideas are deployed, they must be continually updated and re-adapted to address new and emergent contradictions, some of which are created by the innovation itself.

6.1 Conflict Resolution is a Sensitive Process

Identifying and turning contradictions into useful innovations is a challenging task. The chance of a contradiction becoming a useful workflow improvement depends largely on first acknowledging the contradiction [23]. Many contradictions go unnoticed due to a variety of factors including lack of management support or denial by those affected. During the fieldwork we observed many contradictions that were never spoken of by L-1 analysts fearing repercussions. It has been observed that turning a contradiction into an innovation does not happen only at an individual level. A collective effort by the community is needed and tools used by the community may need to be transformed together to enable the innovation [29]. The incident response portal required collaborative effort from the analysts and fieldworkers who acted as analyst/developer. The tool's development required the approval of the SOC manager who allowed the analysts to spend their work time in the co-creation process. Due to different roles and objectives within the activity system, it may be difficult to achieve sufficient consensus around an innovation. Sometimes contradictions are not openly discussed because they are just embarrassing [8]. SOC analysts frequently encounter security breaches; discussing the problems in handling security incidents with other people will put them in a bad light.

In our work, the use of anthropological methods helped us earn the trust of analysts in discussing embarrassing or otherwise undiscussable contradictions. We worked as analysts ourselves and hence were able to experience the contradictions first hand. **It becomes clear that building *trust* among analysts and between various SOC teams is a key enabler for acknowledging and discussing contradictions, and is thus a pre-requisite for bringing about useful innovations to SOCs.** SOC managers must view friction in operations as opportunities for making things better rather than simply reprimanding the analysts. Above all, managers should earn the trust of their analysts and be a participant in the conflict resolution process as they are the authoritative persons to bring actual changes to operations.

6.2 Conflict Resolution is a Continuous Process

As mentioned in Section 2.4, we returned to the SOC where the incident response portal was deployed after a brief hiatus of a few months. To our surprise we found that our tool was shelved and not used by SOC or the NOC staff. As we renewed our fieldwork, which involved continued co-creation, our tool once again was adopted into daily operations by the analysts. Reflecting back on this experience, our incident response portal was temporarily out of operations due to the hiatus in conflict resolution when the fieldworkers were absent in the SOC. This led us to the realization that **successful tools must address contradictions on a continuous basis for their continued usefulness**. This explains why the SIEM solution at this SOC (and at other SOCs we studied), which was essentially a static tool, was barely functional. In short, human activity is a dynamic system. If a tool is to be and remain effective, it must also be dynamic.

7. DISCUSSION

Our conclusion that useful tools for SOCs must help resolve the various contradictions in the work environment on a continuous basis seems to be at odds with how security product vendors produce technologies these days. Many vendors still view this as a "build-once-sell-to-everyone" market, without much understanding of the variations in the workflows and contradictions that may arise within the various SOCs they tend to sell the products to. Our research results imply that tools built this way will not work effectively to help SOC analysts. It seems to follow that useful security tools for SOCs may best be built within SOCs, by people who can identify and understand the contradictions within the work environments. Our experience in the anthropological study shows that to achieve this understanding, it takes a person becoming an analyst and doing the job in the SOC.

Our pentagon model highlights the importance of the "reflecting" and "scripting" stages in SOCs. Unfortunately oftentimes SOC management does not understand the importance of automation and does not allocate enough work force to ensure analysts have time to perform reflection and automation. As a result the analysts are stuck in operation mode, leading to burnout. On the other hand, when the event rate is low, simulation-based approaches could be used to generate events that turn analysts to the acting mode when there are not enough real interesting events.

The ability of analysts to transition to acting stage in the pentagon model depends on their skill set to do rapid software prototyping. In our work the student fieldworkers were skilled programmers, and at the same time security analysts. This allowed them to develop tools that automate the operations. We found that a typical analyst has two problems when it comes to developing quality tools. The first issue arises from a lack of time to write code. In operations, priority is given to handling incidents and responding to tickets. A large number of events per analyst means that analysts do not get the right amount of time to write software, and are not even encouraged to do so. The second issue is that some analysts just do not have the skills to program. As discussed above, good tools can be written only when you actually do the job. This implies that the analysts may be the right people to develop the required tools, which begs

the question of whether programming ability should be a desired qualification for SOC analysts.

8. LIMITATIONS

The main limitation of our work is the subjectivity of the researchers in collecting and analyzing fieldwork data. To address this limitation the collected data was anonymized and shared with the entire research team and extensively discussed. The results presented here are based on our collective study of four different SOCs by five student fieldworkers and a number of senior researchers. We acknowledge that in order to further generalize the findings we need to expand our study to even more SOCs. However it is also important to point out that in our experience thus far SOCs can be very different and overall generalization may be unwarranted and misguided. It might instead be fruitful to pursue a more particularist approach, in which each SOC is studied within its own terms and in an effort to understand its own tensions and contradictions. In this regard, the generalizable aspect of our work is the approach in the work. After further study of more SOCs it may become apparent that the primary and secondary contradictions identified here are evident in all SOCs. Or further study of several SOCs might eventually result in the creation of a typology that can identify different types of SOCs with different sets of tensions and contradictions. Furthermore, we hope to expand our analysis to explore tertiary and quaternary contradictions – contradictions between different activity systems and business units within broader organizations. This is an ongoing effort and we hope to conduct similar studies to gather more insights. Notwithstanding these limitations, we would like to emphasize that conducting long-term anthropological study for SOCs is a process that yields perspectives that are otherwise unobtainable.

9. RELATED WORK

The use of anthropological methods to study SOCs and the idea of co-creation as a means to develop usable operational tools was first reported by us in our prior work [26]. Continuing our anthropological study, we then studied the problem of burnout among security analysts [25]. The work identified multiple vicious cycles between a number of human, organizational, and technological factors to be the primary reasons for burnout and high turnover of analysts in SOCs. The work presented in this paper uncovered a more fundamental principle when it comes to understanding SOC work efficiency and tool building. We present an activity theory model to explain the burnout and tool building in SOCs, yielding insights that were not obtained in our prior work.

There have been prior efforts in studying security operations mainly focused on tool development. Jaferian *et al.* [16] used activity theory to model challenges in reviewing access control policies in organizations. They design a tool that enables easy decision making for access control. Others used interviews and focused on providing guidelines for developing operational tools [5, 15, 27]. There have also been research efforts focused on understanding human, organizational, social, and other factors such as communication in the context of security operations [4, 30, 31, 32]. The main limiting factors of these prior works is the limited time spent in SOCs. From our own experience, it takes time to gain the trust of analysts and their management which is key to understanding the real problems causing inefficiency in security operations. We earned the trust of analysts by working alongside with them. We spent between 6 months and a few years in each of the SOCs, enabling us to understand problems as they evolved over a longer period of time. We believe that the insights we obtained are much deeper than if we had used short term methods such as interviews and questionnaires.

10. CONCLUSION

In this paper we present an activity theory (AT) model to explain the inefficiency in security operation centers (SOCs) and how tool building can help bring useful innovations. We analyzed field notes from our 3.5 year long anthropological study of four academic and corporate SOCs. The analysis revealed a number of primary and secondary contradictions in operational environments that manifest as conflicts. A concrete list of contradictions is presented by modeling SOC operations within the AT framework. Success or failure of technology solutions to improve SOC efficiency depends on acknowledging and mitigating these contradictions. By studying the resolved conflicts we understand why our tools were adopted into operations and became successful. With the reason in hand it becomes possible to reproduce it to solve similar other problems for SOCs. We further found that for a tool to be useful and usable in an operations floor it has to constantly resolve new conflicts that emerge. We leverage the hierarchical structure of human activities proposed in AT and extend it to a Pentagon Model for knowledge generation and transformation in SOCs. This model can be used by SOC managers and developers to identify tasks that could be automated periodically, resolving contradictions and improving SOC efficiency. Finally, the framework presented in this paper can be used to not just build tools, but for other positive changes that improve analysts' efficiency in general.

11. ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for the constructive comments that helped us in revising the paper, and Allison Woodruff for shepherding our paper. We thank the four SOCs who opened their doors for this research, and the analysts who worked together with us. This research is supported by the U.S. National Science Foundation under Grant No. 1314925. It is also partially supported by the Department of Homeland Security Science and Technology Directorate through DHS S&T/HSARPA/CSD BAA 11-02 and the Air Force Research Laboratory, Information Directorate under agreement number FA8750-12-2-0258. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

12. REFERENCES

[1] S. Barab, M. Barnett, L. Yamagata-Lynch, K. Squire, and T. Keating. Using activity theory to understand the contradictions characterizing a technology-rich introductory astronomy course. *Mind, Culture, and Activity*, 9(2):76–107, 2002.

[2] F. Blin. Call and the development of learner autonomy: Towards an activity-theoretical perspective. *ReCALL*, 16(02):377–395, 2004.

[3] C. Bonneau. Contradictions and their concrete manifestations: an activity-theoretical analysis of the intra-organizational co-configuration of open source software. In *Proceedings from EGOS Colloquium, Sub-theme*, volume 50, 2013.

[4] D. Botta, K. Muldner, K. Hawkey, and K. Beznosov. Toward understanding distributed cognition in IT security management: The role of cues and norms. *Cognition, Technology & Work*, 13(2):121–134, 2011.

[5] D. Botta, R. Werlinger, A. Gagné, K. Beznosov, L. Iverson, S. Fels, and B. Fisher. Towards understanding IT security professionals and their tools. In *Proceedings of the 3rd symposium on Usable privacy and security*, pages 100–111. ACM, 2007.

[6] J. Buell. COWs in the classroom: Technology introduction and teacher change through the lens of activity theory. *Unpublished manuscript. University of Illinois at Urbana-Champaign*, 2003.

[7] J. Buell. Learning to teach with laptops: A case study of teacher change. In *Society for Information Technology & Teacher Education International Conference*, pages 1984–1985, 2004.

[8] P. Capper and B. Williams. Cultural historical activity theory (CHAT). In *American Evaluation Association Conference*, pages CHAT–1 – CHAT–23. American Evaluation Association, November 2004. From a workbook entitled *Enhancing evaluation using systems concepts*. Available as of March 1st 2016 at `http://www.bobwilliams.co.nz/Systems_Resources_files/activity.pdf`.

[9] Y. Engeström. *Learning by Expanding: An Activity-Theoretical Approach to Developmental Research*. Orienta-Konsultit Oy, 1987. A second edition was published by Cambridge University Press in 2014.

[10] Y. Engeström. Expansive learning at work: Toward an activity theoretical reconceptualization. *Journal of Education and Work*, 14(1):133–156, 2001.

[11] Y. Engeström, K. Brown, L. C. Christopher, and J. Gregory. Coordination, cooperation, and communication in the courts: Expansive transitions in legal work. In M. Cole, Y. Engeström, and O. A. Vasquez, editors, *Mind, Culture, and Activity. Seminal Papers from the Laboratory of Comparative Human Cognition*, chapter 28, pages 369–388. Cambridge University Press, Oct. 1997.

[12] Y. Engeström, R. Miettinen, and R.-L. Punamäki. *Perspectives on activity theory*. Cambridge University Press, 1999.

[13] J. M. Frambach, E. W. Driessen, and C. P. M. van der Vleuten. Using activity theory to study cultural complexity in medical education. *Perspectives on Medical Education*, 3(3):190–203, June 2014. On line at `http://doi.org/10.1007/s40037-014-0114-3`.

[14] C. Geertz. From the native's point of view: On the nature of anthropological understanding. *Bulletin of the American Academy of Arts and Sciences*, 28(1):26–45, 1974.

[15] P. Jaferian, D. Botta, F. Raja, K. Hawkey, and K. Beznosov. Guidelines for designing IT security management tools. In *Proceedings of the 2nd ACM Symposium on Computer Human Interaction for Management of Information Technology*, page 7. ACM, 2008.

[16] P. Jaferian, H. Rashtian, and K. Beznosov. To authorize or not authorize: Helping users review access policies in organizations. In *Symposium On Usable Privacy and Security (SOUPS 2014)*, pages 301–320, Menlo Park, CA, July 2014. USENIX Association.

[17] V. Kaptelinin and B. Nardi. Activity theory in HCI: Fundamentals and reflections. *Synthesis Lectures Human-Centered Informatics*, 5(1):1–105, 2012.

[18] N. King. Template analysis. In G. Symon and C. Cassell, editors, *Qualitative Methods and Analysis in Organisational Research: A Practical Guide*. Sage Publications Ltd, London, 1998.

[19] K. Kuutti. Activity theory as a potential framework for human-computer interaction research. *Context and consciousness: Activity theory and human-computer interaction*, pages 17–44, 1996.

[20] A. N. Leont'ev. The problem of activity in psychology. *Soviet psychology*, 13(2):4–33, 1974.

[21] A. N. Leontjev. *Problems of the development of the mind*. Progress, Moscow, 1981.

[22] C. P. Lim and D. Hang. An activity theory approach to research of ICT integration in Singapore schools. *Computers & Education*, 41(1):49–63, 2003.

[23] C. P. Nelson. *Contradictions in learning to write in a second language classroom: Insights from radical constructivism, activity theory, and complexity theory*. PhD thesis, The University of Texas at Austin, Austin, TX, 2002.

[24] D. L. Russell and A. Schneiderheinze. Understanding innovation in education using activity theory. *Educational Technology & Society*, 8(1):38–53, 2005.

[25] S. C. Sundaramurthy, A. G. Bardas, J. Case, X. Ou, M. Wesch, J. McHugh, and S. R. Rajagopalan. A human capital model for mitigating security analyst burnout. In *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*, pages 347–359, 2015.

[26] S. C. Sundaramurthy, J. McHugh, X. Ou, S. R. Rajagopalan, and M. Wesch. An anthropological approach to studying CSIRTs. *IEEE Security and Privacy Magazine*, Sept/Oct 2014.

[27] N. F. Velasquez and S. P. Weisband. Work practices of system administrators: Implications for tool design. In *Proceedings of the 2nd ACM Symposium on Computer Human Interaction for Management of Information Technology*, page 1. ACM, 2008.

[28] L. S. Vygotsky. *Mind in society: The development of higher psychological processes*. Harvard university press, 1980.

[29] E. Wardle. Can cross-disciplinary links help us teach 'academic discourse' in FYC? *Across the Disciplines*,

1, July 2004. Found as of March 1st 2016 at `http://wac.colostate.edu/atd/articles/wardle2004/index.cfm`.

[30] R. Werlinger, K. Hawkey, and K. Beznosov. Security practitioners in context: Their activities and interactions. In *CHI'08 Extended Abstracts on Human Factors in Computing Systems*, pages 3789–3794. ACM, 2008.

[31] R. Werlinger, K. Hawkey, and K. Beznosov. An integrated view of human, organizational, and technological challenges of IT security management. *Information Management & Computer Security*, 17(1):4–19, 2009.

[32] R. Werlinger, K. Muldner, K. Hawkey, and K. Beznosov. Preparation, detection, and analysis: the diagnostic work of IT security incident response. *Information Management & Computer Security*, 18(1):26–42, 2010.

APPENDIX

A. SNAPSHOT OF TEMPLATE ANALYSIS

Table 1: Snapshot of Initial Template after Coding a Subset of Data

Theme	Sub-themes	Examples
Primary contradiction	Subject	Metrics define the job.
Secondary contradiction	Subject - Rules	Hinders creativity. Unreasonable.
	Subject - Instrument	Poor attribution. Lack of customization. Lack of analyst perspective. Wrong assumptions. Long tuning process. Lack of visibility into tool functionality. High learning curve. Poor documentation.
	Subject - Community	Misaligned priorities. Pushback.
	Division of labor - Object	Inflexible role assignments. Lack of peer visibility.

B. HIERARCHICAL NATURE OF ACTIVITY

According to AT, human activity can be organized into a hierarchy of levels. This idea is often illustrated using a classical example from Leont'ev [21]. He differentiates between two different types of objects that come into play when people are engaged in socially distributed activities. Usually there is a *motivating object* that inspires the people to perform a particular activity and there is a *directing object* that is more immediate and guides them towards the motivating object. He explains this distinction using the example of hunting. When hunting together, people are divided into two groups: one that scares the animals by beating the bushes. These are called *the beaters*. The other group, called *the ambushers* (or *shooters* in current terminology) waits for the scared animals to come towards them so they can kill them. The original motivating object for the collective activity was food. An outsider positioned to examine only the activities of one group would find them difficult to fathom. The game is often well in advance of the beaters and might not be visible to an observer following them. The ambushers appear to be waiting idly, as they must be in position before the beaters start their drive. It is only when the observer discerns the relationship between the two groups that the hunt becomes apparent.

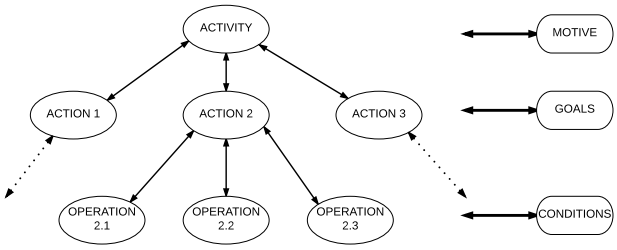

Figure 4: Activity Hierarchy

Figure 4 shows three levels in the hierarchy of human activity [17]. This abstraction can be adapted to fit any context. At the top level is the *activity* itself which is guided by the *motive*. The activity is broken down into sub-units called *actions*. The actions are motivated by *goals* that, seen in isolation, may appear to have nothing to do with the overall motive of the activity *e.g.*, the action of beaters may appear to have nothing to do with the overall motive of hunting. Each action is then decomposed into further smaller units called *operations*. Operations are in fact actions that have been customized to the environment under which they are carried out. The distinction between an action and an operation is that one may be aware of the fact that they are performing an action while an operation is a subconscious routinized task.