An Anthropological Approach to Studying CSIRTs

Sathya Chandran Sundaramurthy | Kansas State University
John McHugh | RedJack
Xinming "Simon" Ou | Kansas State University
S. Raj Rajagopalan | Honeywell
Michael Wesch | Kansas State University

The ethnographic method of participant observation can help researchers better understand the challenges computer security incident response teams face by illuminating underlying assumptions and tacit practices that shape how tools are actually used in different contexts.

Sathya Chandran Sundaramurthy eyed his screen with a paradoxically detached intensity as he scanned a small bit of the more than 70 Gbytes of log data accumulated that day. Working as an analyst for a university's security operations center (SOC), his job was to find a simple data match that could move the latest investigation forward. This was no glamorous or exciting game of catching the hacker. Instead, it was rather simple, yet intense, tedium. He had to be alert and move fast to close tickets quickly, but this particular part of the job—scanning data and numbers—didn't exactly engage his capacities for complex thought and analysis.

He calculated that he was in the midst of a series of five-minute cycles: receive an alert, scan the logs (three minutes), look up an address (one minute), find the user information (another minute), repeat. This would be the 47th cycle this week. Such is the life of an analyst.

However, Sundaramurthy is no analyst. He's part of our cybersecurity research team, which has spent years trying to understand the workings of SOCs to build tools for analysts. We never seemed to get the data, knowledge, and insight we needed, and convincing analysts to try our tools was difficult. So, we added anthropologist Michael Wesch to the team to advise us on ethnographic methods and participant observation.

In his first month at the university SOC, Sundaramurthy was starting to understand why traditional research methods, such as interviews and surveys, hadn't worked. A lot of what he learned in that first month wasn't easy to talk about; it was tacit, embodied knowledge. Much of it wouldn't seem interesting or relevant enough to mention in an interview, and the environment—consisting of analysts battling unseen adversaries in a basement—didn't exactly foster trust. He thought, "Why should I believe that the interviewer isn't trying to obtain information that would allow him to hack the system? Who has time for an interview anyway? More tickets are flowing in as we speak. I have to close tickets."

By this point, Sundaramurthy understood that analysts have little time for interviews with researchers; are unlikely to trust them; and, even if they do make the time and the leap of faith, might not know what to

 Copublished by the IEEE Computer and Reliability Societies 1540-7993/14/$31.00 © 2014 IEEE

say. Much of what they know isn't easily put into words because it's unconscious, inappropriate to discuss, apparently unimportant, or irrelevant.

Fast-forward one year. Sundaramurthy has spent nearly 500 hours working as an analyst at the SOC. The challenges of previous research methodologies didn't simply go away. It took Sundaramurthy many hours and a great deal of effort to move past the barriers—to earn the trust of the other analysts and upper management, reflect on the knowledge embodied through the fieldwork process, and make this knowledge more explicit. This allowed him to analyze the broader policies, social relations, and bureaucratic requirements that continually change and reshape the context in which the work takes place.

In this article, we describe the process Sundaramurthy and our team underwent to address two questions: How can we make SOCs—and in particular, computer security incident response teams (CSIRTs)—more effective (by any reasonable metric), and how can cybersecurity researchers play a significant role in improving SOCs and CSIRTs? We made progress on both fronts.

Anthropology Today

We often imagine cultural anthropologists working at the far edges of mainstream civilization, studying people isolated from technology to understand human behavioral characteristics. However, over the past several decades, more cultural anthropologists have been using their research methods to explore familiar territories, often with surprisingly practical results.

In the late 1990s, Leinbach and Sears used anthropology methods to design recreation vehicles. Spending six months on the road living in a giant RV and staying at campsites, the researchers learned that current RV design didn't match everyday RV users' needs and desires—for example, most RV users don't use the in-vehicle shower (they prefer the high-pressure campground showers and use the space as a closet). Earlier designers had imagined what RV users would need but never lived with them. Leinbach and Sears built a prototype embodying their findings that was so successful the manufacturer dropped all other models to meet the demand.[1]

There are many other success stories. Genevieve Bell, a cultural anthropologist at Intel, explored the social and cultural aspects of ubiquitous computing with Paul Dourish, significantly shaping ubiquitous computing research methodologies. They looked beyond the technical, illuminating the ways in which ubiquitous computing was lived and thought about in people's everyday lives.[2] Go-Gurt, the handheld yogurt that children can enjoy like a popsicle while appeasing parents' interest in nutrition, was based on an anthropological study by Susan Squires.[3] People widely—and wrongly—believe that an anthropologist invented Xerox machines' famous big green "start" button. In 1979, anthropologist Lucy Suchman filmed user interactions with a new Xerox printer, widely disparaged by users as too complicated. Suchman's footage became famous for the meme-like image "When User Hits Machine" that shows PARC researchers Allen Newell and Ron Kaplan trying to use it. The printer already had the big green button, but when users pushed it, the machine wouldn't do what they wanted it to. From this, Suchman made the point that no machine or tool is self-explanatory: "We need time to make unfamiliar devices our familiars."[4]

As we make a tool our familiar, we often change its use and function from those originally intended. In this way, tool users become, in part, tool authors as well. Although traditional research methods might elicit somebody's explicit understandings, needs, and desires, only participant observation can help us understand the embedding context, underlying assumptions, and tacit practices that shape how tools are actually used in different contexts.

As the ethnographic methods anthropologists developed and honed in the most remote areas of the world were adapted and used in more practical applications, "ethnography" came to designate a wide variety of research practices. By the mid-1990s, cultural anthropologist Michael Agar expressed a growing concern among his fellow anthropologists that there was now a "dangerous delusion" that a few hours with a focus group could pose as ethnography.[5]

Nonetheless, even mild forms of ethnography were having profound effects, allowing people to see and understand new perspectives and gain new insights. Instead of drawing a line in the sand and denying short-term fieldwork and focus groups the status of ethnography, Agar drew a broad spectrum of ethnographic possibilities from focus groups and workplace observations on one end to long-term intensive participant observation in which researchers join a community for months or years on the other.

Bringing Anthropology to the Study of CSIRTs

The traditional academic cybersecurity research approach is to identify problems and areas for improvement by studying the research literature, then develop tools and methodologies to address those problems. This process often results in "solutions" that aren't usable in the real world. We believe the problem results from a discrepancy between what security practitioners actually need and researchers' perception of that need. As a result, research results rarely find their way into practical use.

Researchers in similar domains have observed

comparable problems and adopted ethnography as a key component of their research. The ongoing challenge is applying ethnographic methods to close the gaps between researcher and participant, designer and user, and researcher and designer. This process continues to enrich the methodological toolkits available to human-computer interaction, participatory design (PD), and computer-supported cooperative work communities, among others. Conceptual and theoretical toolkits continue to grow as well. As Andy Crabtree and Tom Rodden pointed out, new concepts emerging from ethnographic work alert and sensitize us to issues and insights that otherwise might remain unnoticed.[6] Although many researchers using ethnography in computer science have moved toward short-term techniques—tightening their research scope and using multiple methods to match increasingly short production cycles—we find unique challenges in studying CSIRTs that require us to return to a more traditional form of anthropological fieldwork—long-term participant observation.

> **SOC jobs such as incident response and forensic analysis have become so sophisticated and expertise driven that understanding the process is nearly impossible without doing the job.**

Gaining Trust

SOC culture can't be easily understood from the outside. The main challenge facing researchers undertaking ethnographic fieldwork in a SOC is earning trust. SOC employees have many reasons not to trust researchers. Due to the sensitive environment, sharing information with outsiders is discouraged or prohibited. SOCs are frenzied and stressful workplaces with intensive workloads. Employees are evaluated by how many tickets or incidents they close per day and don't have the time or desire to share information among themselves, much less with outsiders. SOCs' "tribal culture" dictates that newcomers learn the routine on their own.

Our fieldworkers experienced these difficulties when they started work. We faced these problems when designing cybersecurity tools and techniques for SOC analysts, and over the years, we've heard similar anecdotal stories from many sources: there's a significant gap between researcher and practitioner cultures.

Bridging the Chasm

We've been involved in academic and operational security communities for several decades. Our observations of and participation in both communities have led us to many conclusions that shed light on the problems that we address in this article. With few exceptions (for instance, Lawrence Berkeley Laboratory staff), cybersecurity practitioners and academic researchers have little contact. They work under different conditions, attend different conferences, and have developed a mutual feeling of distance from one another. Academics are accustomed to creating, contesting, and sharing knowledge and expertise, whereas practitioners have long worked independently without sharing detailed expositions of their knowledge.

Owing to the tacit nature of practitioners' knowledge, fostering conversations between the two camps doesn't necessarily transfer knowledge. As Michael Polanyi noted, "We can know more than we can tell."[7] Cybersecurity practitioners often work from hunches or intuitions. They know what to do, where to look, and how to investigate a case but often can't state this knowledge explicitly. SOC jobs such as incident response and forensic analysis have become so sophisticated and expertise driven that understanding the process is nearly impossible without doing the job. Few researchers do fieldwork to understand the problem firsthand before attempting to design models and algorithms that purportedly solve the problem.

Researchers perceive SOC operations as shrouded in secrecy and find it hard to get data or deploy and evaluate research prototypes. Analysts are guarded when approached by researchers, who are often perceived to be more interested in publications than solving real problems. Most academic research gets little credibility with and has little impact on the practitioner community. Researchers' studies are often plagued by lack of scientific rigor (for example, see the 2010 New Security Paradigm Workshop panel and the 2011 Learning from Authoritative Security Experiment Results workshop panel and conferences) and are largely irrelevant or divorced from reality (see Felix Lindner's keynote, "On Hackers and Academia," from the 2010 European Conference on Computer Network Defense[8]). The result is infrequent technology transfer from academic research to security practitioners.

Long-term participant observation is the key. The research dynamics change completely when researchers are willing to join a community in a long-term effort. This is not unlike the "old-day" work in which anthropologists join a remote indigenous community to understand its culture (as Wesch did for 18 months, studying the effects of writing on an indigenous culture in Papua New Guinea). This type of study often takes

years. It sometimes takes months for an anthropologist to earn a local community's trust. Gaining trust is incremental and usually takes a breakthrough after a painful "being pushed to the side" period. We observed this in our SOC fieldwork. Short-term ethnographic work (days or weeks) used in other computer science domains simply doesn't work in a SOC.

Our work shares many foundational assumptions with participatory design. At the dawn of the PD movement, Paul Czyzewski and his colleagues noted that their research turned "the traditional designer-user relationship on its head, viewing the users as the experts—the ones with the most knowledge about what they do and what they need." They viewed the tools they developed "not in isolation, but rather in the context of a workplace; as processes rather than products." Their work rejected "the assumption that the goal of computerization is to automate the skills of human workers, instead seeing it as an attempt to give workers better tools for doing their jobs."[9]

We generally agree with these principles; however, we also believe that long-term anthropological fieldwork, focused on earning trust, building rapport, and ultimately facilitating long-term collaboration with a relatively small group of analysts, is better suited to working with SOC analysts.

Even Experts Can't Tell How

SOC analysts handle various cyberattack-related events. They use numerous tools and follow set incident-handling procedures. We talked with SOC analysts at many organizations; all were uniformly unhappy with current forensics and incident response solutions. Commercial solutions like security information and event management (SIEM) systems don't address operational needs, probably because vendors and researchers don't understand how analysts think and work.

SOC analysts often perform sophisticated investigations, and the process required to connect the dots is unclear even to analysts. Incident response isn't just a technical problem; it involves people with various skills interacting in a closed culture, using specific workflows for each incident type. Current solutions aren't informed by these workflows and are only partially helpful.

An analyst's job is highly dynamic and requires dealing with constantly evolving threats. Doing the job is more art than science. Ad hoc, on-the-job training for new analysts is the norm. Rookies are intentionally left to find answers by themselves, even though the more experienced analysts could provide hints. Common SOC wisdom is that one must learn the trade through pain. This private initiation process contributes to the persistence of hidden or tacit knowledge.

This phenomenon has long been studied in the social sciences, especially anthropology. Tacit knowledge can't easily be put into words.[7] CSIRT tasks are sophisticated, but there's no manual or textbook to explain them. Even experienced analysts might find it hard to explain exactly how they discover connections in an investigation. The fact that new analysts get little help in training isn't surprising. The profession is so nascent that the how-tos haven't been fully realized, even by the people who have the knowledge.

Again, long-term participant observation is the key. Cultural and social anthropology could be described as the study of tacit knowledge.[10] Anthropologists do intensive long-term fieldwork in an attempt to reveal and make explicit the "native point of view," which isn't just what natives say (explicit knowledge) but, more important, the underlying concepts, presuppositions, and know-how that make up tacit knowledge. Although the native point of view is never fully attainable,[11] the foundational participant observation method lets anthropologists explore subjects' perspectives and practices by actually taking part in their daily lives and activities (participation) while also standing back from them to gain new perspectives (observation).[12] Reflecting on the observations and making what is tacit explicit takes a substantial amount of time.

Ethnographic Fieldwork at a University SOC

Four PhD students in computer science—Alexandru Bardas, Yuping Li, Sathya Chandran Sundaramurthy, and Loai Zomlot—conducted ethnographic fieldwork at a university SOC for 15 months. As members of the operations team, they performed many of the analysts' tasks, experiencing their pains, frustrations, and occasional triumphs. They worked with Kansas State University faculty members Xinming "Simon" Ou (computer science) and Michael Wesch (anthropology), and with John McHugh (RedJack, LLC) and S. Raj Rajagopalan (Honeywell). The team met regularly via video conference to discuss the research.

Fieldwork Setup

The SOC where the students conducted their fieldwork consists of a chief information security officer (CISO) and four analysts. Each analyst has specific responsibilities, such as incident response, firewall and network management, payment card industry compliance, and antivirus maintenance. The fieldworkers were initially introduced as student helpers, paid by their adviser (Ou) to learn operational security by doing the work themselves. The CISO was very supportive of the effort.

It took six months for the research team and the SOC to find the best model to manage this relationship. For the first few months, the SOC analysts didn't understand

the students' role. Sundaramurthy was asked to perform mundane tasks such as maintaining out-of-date antivirus servers and developing hardware requirements for upgrade. These were day-to-day tasks that the analysts didn't have time for but weren't tasks that would allow Sundaramurthy to gain insight into incident investigation processes. The fieldworkers' schedules also caused problems. Graduate students don't usually work fixed schedules, but the SOC requires dependability and accountability, especially when incidents need to be handled. Ou and the CISO worked to find a mutually agreeable solution. The CISO released Sundaramurthy from tasks unrelated to the study goals, and the student fieldworkers agreed to work fixed schedules in the SOC and notify the SOC when they would not be present.

For the first few months, Sundaramurthy spent 15 hours per week in the SOC. In that time, he had worked more than 460 hours. Bardas and Zomlot worked 160 hours each over four months, and Li (still there) worked more than 200 hours in nine months.

We realized the best way to do the study was to perform SOC tasks, be they ticket handling or documentation, in parallel with the research. Later, we learned that building tools to help analysts improve their job efficiency was the best way to access and use the SOC's tacit knowledge. This arrangement benefited both the SOC and the researchers.

Earning Trust Is the Breakthrough

Sundaramurthy was immediately introduced to the tedium of the job's time-consuming, low-level tasks. The SOC receives alerts on malicious network traffic from several trusted sources as well as from its own intrusion detection system. The alerts contain the infected host's IP address (usually that of the NATing [network address translator] firewall) and the external IP address with which it was communicating. The real internal IP address must be extracted from the firewall logs, with the media access control (MAC) address identifying the infected host from DHCP (Dynamic Host Control Protocol) logs. Finding the log entry for a given event and looking up the associated information to resolve the ticket takes approximately five minutes. This repeats and repeats. Before long, Sundaramurthy was fully absorbed. He wasn't creating tools. He was a tool.

We were surprised that no off-the-shelf SIEM product could address such a simple event correlation problem. We now know from the literature and anecdotal stories that this type of problem is common in other SOCs (commercial, government, and educational).

Much of Sundaramurthy's time was spent on carrying out repetitive operational tasks. He felt frustrated because he didn't feel he was gaining any insight. Many times he tried to talk to the chief incident response/forensics analyst, hoping to learn more advanced investigation skills but was told just to handle the tickets—the highest priority. The chief analyst spent most of his time handling incidents, some too sensitive to give to Sundaramurthy. Sundaramurthy was tied up with low-level, repetitive tasks and felt that he wasn't doing anything useful for his research.

Because the typical SOC mentality focuses on getting incidents processed quickly, there's no time to contemplate a long-term vision of improved efficiency. Sundaramurthy was consumed by this mentality. He lost his perspective by becoming part of the SOC! He thought that this was the way things were done, and there was nothing he could do to change it. It wasn't until Sundaramurthy discussed his work with his adviser and the rest of the research team that he realized things didn't have to be this way.

After suggestions from S. Raj Rajagopalan and Dan Moor, a senior industrial security analyst collaborating on the project, Sundaramurthy decided to find ways to speed up ticket handling by building a database of connections and an IP-to-MAC address mapping. Noting that most active alerts are less than one week old, he built a caching database retaining seven days of mapping information. He first tried using MySQL, which wasn't able to index the inputs in real time. He then chose MongoDB, which stores data as JSON type (schemaless) objects and has a sufficiently high ingest capability.

After the database was operational, Sundaramurthy asked the chief incident response analyst to use it. The analyst was extremely happy with the performance improvement, which reduced ticket-handling time from five minutes to two seconds. The most important result was that the analyst became enthusiastic and willing to talk to Sundaramurthy about possible tool extensions and providing data to expand the database. The two had a long brainstorming session and arrived at a "threat intelligence framework" that added information sources and relationships among them to the database to handle various incidents.

This is the kind of thing that Sundaramurthy had wanted to do from the beginning but wasn't able to until he understood the workflow and demonstrated his worth. Doing mundane tasks didn't appear to support his real objective—designing better tools for SOC analysts—but it was key in extracting the tacit knowledge necessary to build the tool. Once the analyst saw the simple tool Sundaramurthy had built, he completely changed his attitude. Sundaramurthy found his first "informant." The key to this event is having a subject's trust, which was gained by providing something useful.

This first success is a great example of a researcher in the field moving from peripheral to full participation. Now that Sundaramurthy had trust and acceptance, he

was closer to seeing the tasks through the analysts' eyes and gaining a much deeper understanding of the SOC's operations. In other words, he opened the door to access tacit knowledge and made it explicit by embodying it in a tool.

Enhanced Participant Observation Facilitated by Tool Building

Over the past 20 years, researchers in organizational studies have expanded on and refined a model for how tacit knowledge can be accessed and ultimately transformed into explicit knowledge. Most prominent and relevant for our discussion is Ikujiro Nonaka's SECI model of knowledge conversion (see Figure 1), which defines four modes of knowledge creation:

- socialization—sharing tacit knowledge through apprenticeship and mentoring;
- externalization—converting from tacit to explicit through questioning, reflection, and reconstruction;
- combination—accumulating explicit knowledge; and
- internalization—converting new forms of explicit knowledge into tacit knowledge.

The model has changed over the years. Today, it's understood that tacit and explicit knowledge exist on a continuum[13] and that movement along this continuum is best achieved through reflection; ongoing practice; and, above all, undergoing a diverse range of alternative experiences that can help disorient oneself just enough to see the tacit dimension—that is, to stop taking for granted the taken-for-granted. In this regard, practice is essential. Joining the community isn't enough, as the elements of knowledge aren't necessarily shared across all members. One must be fully engaged in the day-to-day practice.

Based on this strategy and the lessons we learned in earning the analysts' trust, we adopted a framework to study the SOC (see Figure 2). Researchers

1. become apprentices of CSIRTs analysts (socialization);
2. reflect, question, and reconstruct what they do (externalization);
3. design models and algorithms and build tools to help the analysts' job (combination); and
4. take the tool into the workplace and use it to begin more discussions with the analysts and identify more tacit knowledge (internalization).

For example, after Sundaramurthy released the tool, the chief incident response analyst wanted to enhance it to handle other incident types. These adaptations were unexpected. One enhancement helps find stolen laptops. Here, the perpetrator is usually a student. If the CSIRT knows the stolen laptop's MAC address, it can apprehend the perpetrator using access point information if he or she uses the university's authenticated wireless service. If the perpetrator uses any campus service that requires authentication, even through the unauthenticated guest wireless service, the CSIRT can use its logs and other information collected from the authenticated services in the threat intelligence framework for identification.

Analysts also applied the framework to phishing scam detection. Whenever the CSIRT identifies a phishing email, it responds from a honeypot university email address using a fake user ID. The CSIRT then watches

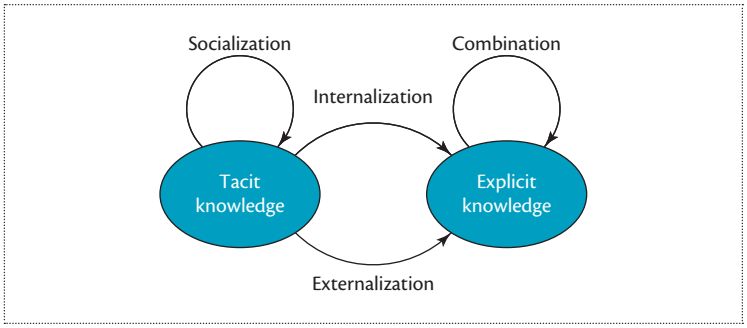

Figure 1. Knowledge conversion model: SECI (socialization, externalization, combination, and internalization) diagram. The socialization process creates the body of practice and its associated tacit knowledge that remains locked in a community. Externalization makes this knowledge explicit so that it can be described and taught. This allows others to work with the knowledge, enhancing it or combining it in ways to create new knowledge, which can then be internalized as practice, creating additional tacit knowledge.

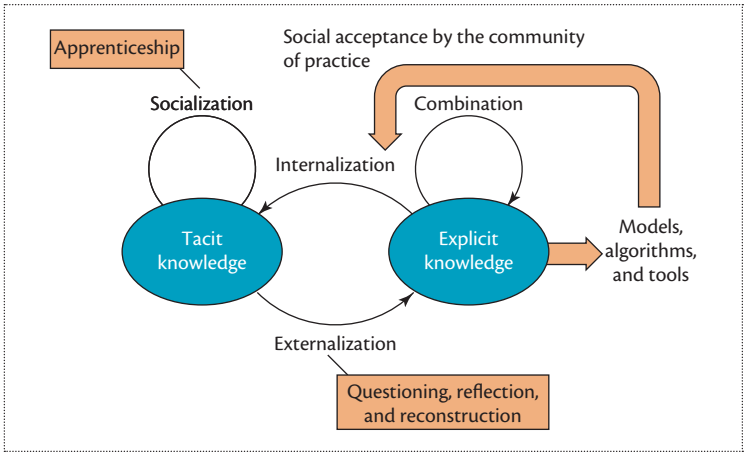

Figure 2. Tool-oriented SECI. This version explicitly acknowledges the role of apprenticeship in socialization and of questioning, reflection, and reconstruction in externalization. However, its main contribution is the use of the resulting explicit knowledge to construct models and algorithms embodied in tools to gain acceptance by the community of practice. Once accepted, these tools quickly find their way back into the body of practice and tacit knowledge.

for a login from that user ID in the future. It notes the IP address associated with the activity and checks this against other logins made using the same address but different user IDs, as attackers usually harvest numerous university accounts and try them in quick succession. Using the framework, the CSIRT can place a watch for logins from the honeypot account and automatically identify other accounts that are possibly compromised, reducing the time the analysts spend responding to phishing scams.

The success of these efforts created a demand for similar automation. Two successful adaptations illustrate the approach's strength. The manual process of intercepting and extracting executable email attachments, evaluating them, and submitting malicious samples to the antivirus vendor was completely automated. Another process detects machines running an unsupported OS (Windows XP), combining information from browser `UserAgent` strings obtained from deep packet inspection and address and platform information, allowing automatic tracking of such machines and their removal from the network. The framework automated the handling of multiple incident types and let the SOC turn these over to the university's lower-level network operations center.

In this ongoing collaboration, the tool's true author becomes blurred. The researcher develops a tool that analysts take up and use in ways the researcher might never have imagined. This cycle produces findings and tools at the same time, a research methodology that differs from both traditional cybersecurity and anthropological research. Instead of building algorithms and tools first, researchers base their model on concrete ethnographic fieldwork data, which yields algorithms and tools that demonstrably help analysts. CSIRT community members no longer resist adopting the research prototype because the tool builder is seen as one of their own. Most important, the tool provides an opportunity for analysts to brainstorm with researchers on additional problems that the tool could address, opening up more venues for sharing tacit knowledge.

Observations from Fieldwork

Our fieldwork produced two key observations. The first relates to the fieldwork itself. We believe that we have a new paradigm in which participant observers who are also researchers in the operational area can produce significant findings in both academic and operational areas. The second addresses the longstanding problem of developing tools that are both useful and likely to get used.

On research methodology. Past researchers have realized the tacit nature of IT security operations knowledge and adopted short-term participant observation.[14] We believe long-term participant observation, on the order of years, is necessary to gain insights that reveal deep problems in SOC operations. Our four student researchers conducted nearly 1,000 hours of fieldwork in one SOC over a 15-month period. It took them three months to just earn trust from the analysts and start discussion through tool co-creation.

Sundaramurthy's realization that he could build a tool to speed up incident response illustrates a paradox of fieldwork. On one hand, the further researchers are from the community they want to study, the more daring their ideas can be. However, without being part of the community, their ideas might lack relevance or might not be implementable. Fieldworkers must be members of the community they're studying and remind themselves often that they're observers as well. Subjective findings are inevitable. It's important for researchers to practice reflexivity—stepping out of the subject role to reflect on and question what they do and how things are perceived. Anthropologists exist "betwixt and between" the world of researcher and subject. Again, in environments like CSIRTs, this approach is necessary because the subjects often can't identify and articulate the critical relevant information. Long-term observation and participation in the target environment are critical to understanding the problem.

Our approach differs markedly from the classical design ethnography process, wherein there's a distinct difference among researchers (anthropologists), designers (tool builders), and users (participants). In our work, the three roles combine, and our fieldworkers do ethnographic fieldwork while designing new tools both they and other SOC analysts will use. This unique mode of ethnography is determined by the SOC's environment—we wouldn't be able to simply observe analysts using third-party tools and draw the same deep insights. There's a tight collaboration between researcher and subjects, and the fieldworker's role is the perpetual trinity of researcher, designer, and user.

On tools' and technologies' role in the SOC. Our own experience—like that of other anthropologists engaging in similar practical applications of participant observation—has been one of a continuous flow of subtle and sometimes not-so-subtle insights that continually reshape our understanding. Workplaces are complex social environments, made even more complex by the use of systems and tools. We don't just use tools; when we use tools, our routines and habits change. They change the way we think about and address problems. They might change who we collaborate with and how. They might even be the catalyst for a complete restructuring of an organizational chart or workflow. As John Culkin (invoking the insight of his colleague, Marshall

McLuhan) noted, "We shape our tools and thereafter our tools shape us."[15] We recognize that the relationship between humans and their tools will always be complex.

Computer science started as a discipline devoted to developing tools to solve computational problems from other areas, with many first-generation researchers having roots in mathematics, physics, engineering, business, or other areas with hard computational problems. As the field came into its own, many academic computer scientists drifted away from real-world problems and concentrated on more tractable abstractions or simplified cases. In security, noise-free test data is one example.

By joining forces with anthropology, we learned the importance of understanding the SOC analyst's world and have been able to gain their trust through participant observation. By reflecting on our observations, we were able to bridge the gap between building an arbitrary "computer science-y" tool and a tool that supports analysts' needs. This is part of a learning process for both researchers/tool builders and analysts. There are no definite endpoints to such a learning process. We must be continuously aware of how our presence—and the presence of the tools we build—might shape the research environment.

The entire process is inherently reflexive and demands ongoing commitment to a careful and critical analysis of our own biases and assumptions. As we learn about a CSIRT, the external world keeps changing and the CSIRT has to continuously adapt. New knowledge that gets incorporated into the CSIRT is very likely to be tacit because of the problem's experiential nature. It must be converted to explicit form as we go forward.

We're extending and expanding our effort to include additional SOCs. We would like to find more partners to work with so our study can be more representative. We need our collaborators to dedicate some human resources to fieldwork. Collaborating organizations will benefit from a third-party perspective of operational effectiveness, intrateam interactions, and so forth, in the context of cybersecurity operations. They might also benefit from tools that fieldworkers build or help build for the organization. At the end of the project, we expect to write a training manual for organizations employing cybersecurity operations personnel. The manual should be useful to commercial, academic, and government SOCs. ■

Acknowledgments

This research is supported by the US National Science Foundation under grant 1314925. 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 National Science Foundation.

References

1. C. Leinbach, "Managing for Breakthroughs: A View from Industrial Design," *Creating Breakthrough Ideas: The Collaboration of Anthropologists and Designers in the Product Development Process*, S. Squires and B. Byrne, eds., Greenwood, 2002, pp. 3–16.
2. P. Dourish and G. Bell, *Divining a Digital Future: Mess and Mythology in Ubiquitous Computing*, MIT Press, 2011.
3. S. Squires and B. Byrne, *Creating Breakthrough Ideas: The Collaboration of Anthropologists and Designers in the Product Development Industry*, Greenwood, 2002.
4. L. Suchman, "Where the Big Green Copier Button Came From," *A Thinking Person*, comment on blog post; 31 Oct. 2011; http://athinkingperson.com/2010/06/02/where-the-big-green-copier-button-came-from/#comments.
5. M.H. Agar, *The Professional Stranger: An Informal Introduction to Ethnography*, Emerald Group, 1996.
6. A. Crabtree and T. Rodden, "Ethnography and Design?," *Int'l Workshop Interpretive Approaches to Information Systems and Computing Research*, 2002.
7. M. Polanyi, *The Tacit Dimension*, DoubleDay, 1966.
8. F. Lindner, "On Hackers and Academia," keynote, *European Conf. Computer Network Defense*, 2010; http://2010.ec2nd.org/program/keynote2.
9. P. Czyzewski, J. Johnson, and E. Roberts, "Introduction: Purpose of PDC 90," *Proc. PDC Conf. Participatory Design*, 1990, pp. ii–iii.
10. J. Elyachar, "Before (and after) Neoliberalism: Tacit Knowledge, Secrets of the Trade, and the Public Sector in Egypt," *Cultural Anthropology*, vol. 27, no. 1, 2012, pp. 76–96.
11. C. Geertz, "From the Native's Point of View: On the Nature of Anthropological Understanding," *Bulletin Am. Academy of Arts and Sciences*, vol. 28, no. 1, 1974, pp. 26–45.
12. H.R. Bernard, *Research Methods in Anthropology: Qualitative and Quantitative Approaches*, 5th ed., *AltaMira*, 2011.
13. I. Nonaka and G. von Krogh, "Tacit Knowledge and Knowledge Conversion: Controversy and Advancement in Organizational Knowledge Creation Theory," *Organization Science*, vol. 20, no. 3, 2009, pp. 635–652.
14. R. Werlinger, K. Hawkey, and K. Beznosov, "Security Practitioners in Context: Their Activities and Interactions," *CHI 08 Extended Abstracts on Human Factors in Computing Systems*, 2008.
15. J. Culkin, "A Schoolman's Guide to Marshall McLuhan," *Saturday Rev.*, 18 Mar. 1967, pp. 51–53, 70–72.

Sathya Chandran Sundaramurthy is a PhD candidate in computer science at Kansas State University. His research interests include studying security operation centers using anthropological methods. Sundaramurthy received a BS in computer science and engineering

from Anna University, India. Contact him at sathya@ksu.edu.

John McHugh is the senior principal and chief analyst at RedJack and an adjunct professor of computer science at the University of North Carolina. His research interests include network data analysis and operational security. McHugh received a PhD in computer science from the University of Texas. He's a senior life member of IEEE. Contact him at mchugh@cs.unc.edu.

Xinming "Simon" Ou is an associate professor of computer science and the Gary and Peggy Edwards Chair in Engineering at Kansas State University. His research interests include designing better technologies to facilitate cyberdefense. Ou received a PhD in computer science from Princeton University. Contact him at xou@ksu.edu.

S. Raj Rajagopalan is a senior principal scientist with Honeywell Automation and Control Systems Research Labs. His research interests include safety and security for operational control systems. Rajagopalan received a PhD in computer science from Boston University. Contact him at siva.rajagopalan@honeywell.com.

Michael Wesch is an associate professor of cultural anthropology at Kansas State University. His research interests include the effects of media and technology on global society. Wesch received a PhD from the University of Virginia. Contact him at mwesch@ksu.edu.

Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.

Call for Articles

IEEE Software seeks practical, readable articles that will appeal to experts and nonexperts alike. The magazine aims to deliver reliable information to software developers and managers to help them stay on top of rapid technology change. Submissions must be original and no more than 4,700 words, including 200 words for each table and figure.

IEEE Software

Author guidelines: www.computer.org/software/author.htm
Further details: software@computer.org
www.computer.org/software