Designing Forensic Analysis Techniques through Anthropology

Sathya Chandran Sundaramurthy
sathya@ksu.edu
(Advisor: Dr. Xinming Ou)

Computing and Information Sciences
Kansas State University
Manhattan, KS, USA

1 Objectives of my Dissertation

Current tools and solutions to handle incident response and forensics focus only on one piece of evidence, doing very little towards presenting the big picture. My PhD dissertation will focus on developing analytical tools that can automate repeated tasks whenever possible and also be able to connect the dots among multiple data sources. The tools of my research will focus more on reducing the time incident responders spend on mundane tasks through automation also by providing data in a more abstract and context specific manner. Such presentation will be more useful in constructing the intrusion scenario than when it is presented raw. Another challenge security researchers face today is in validating their research ideas on real-world data. I will describe the methodology I have adopted in conducting my research trying to address the above problems, which will be the focus of this document.

2 Problems in Current Research Methodology

The traditional approach taken by the security research community for innovation is to read the current literature on a problem, identify areas for improvement, and then develop tools and methodologies that address those problems. While this process may result in theoretically sound solutions there has always been a issue of how usable these solutions are in the real-world. The main reason, I believe, for this problem is the discrepancy between what the security practitioners actually want and what the researchers perceive as what they want. As a result, the research solutions hardly find their way into practical use.

Few years back I worked on validating our previous work SnIPS [7] a correlation engine that works on top of Snort alerts and host logs to identify high-confidence attacks in an enterprise network. In addition to manual analysis we also worked with the Kansas State University (K-State) Computer Science Department System Administrator to identify the value he sees in such a tool as ours. We did this by spending some time asking questions for an hour or so every semester. Since he was already overwhelmed with other departmental duties we were not able to continue our interview process further on.

When I started my PhD work towards automating possible phases of incident response and forensics, we decided that the best way to develop security tools that can be highly useful is to work alongside security professionals on a daily basis. This method of working inside a community on a daily basis has been well studied in Anthropology.

3 Relevance of Anthropological Methods to Cybersecurity Research

Anthropology is defined as the “science of humanity” and in the United States is divided into four fields: cultural anthropology, archaeology, linguistic anthropology, and biological anthropology. Social anthropology is a branch of anthropology where researchers conduct intensive field studies using participant observation methods to understand how members of the community behave in a group. This method has also led to innovation in product development in the past.

In the late 1990s, Charles Leinbach and Ron Sears brought the methods of anthropology to the design of Recreation Vehicle (RV) campers. Their method was, of course, to become RV campers themselves (Participant Observation), spending 6 months on the road with a giant RV and living in RV campsites. They learned that there was a whole culture in the RV world, and that much of the design of the RV was completely inadequate for the actual needs and desires of this culture. They found, for example, that most RV campers never use the shower on board (they prefer the high pressure showers offered at the campgrounds that do not waste their limited water supply), and instead use the shower as an extra closet. Office-bound designers had designed what they imagined RVers would need - but had never actually lived with them to find out more about their culture. When Charles and Ron finished their study they had hundreds of ideas and built a prototype that was so successful that the company had to cancel the manufacture of all their other models to meet the demand of this new prototype. The prototype has now become one of the most copied in the history of the industry.

Genevieve Bell, a cultural anthropologist at Intel studied how people of different cultures around the world used the technology. Her work [2] along with Paul Dourish explored the social and cultural aspects of ubiquitous computing. Their work has significantly shaped the ubiquitous computing research methodologies.

We believe adopting the methods of cultural anthropology to cybersecurity research will lead to corresponding innovations.

3.1 Tacit knowledge in Anthropology

Anthropologists try to study the tacit knowledge of a community by spending significant amount of time with the people of that community [3]. The reason being that one cannot understand the internal thinking or “tacit knowledge” of the people by just observing from outside as pointed out by Michael Polanyi [6]. Polanyi also found out that "We can know more than we can tell." Cybersecurity practitioners work based on "intuition" or "hunch feeling" which is primarily due to their years of experience in looking at data. Through the study of Jeane Lave and Etienne Wenger it is found that knowledge in a community is (1) not always explicit, (2) often embodied in practice, and (3) the knowledge may not even be "in" an individual but emobodied in the community of practice [5]. Also this tacit knowledge can only be acquired not just by being part of the community but also doing what they do on a daily basis [1]. Clifford Geertz [4] in his article "Deep play" talks about how he and his wife, both anthropologists, trying to study the Balinese people were able to gain acceptance of the villagers. They remained invisible to the villagers until after an incident where they decided to do what the villagers were doing.

My idea is to apply anthropological methods to cybersecurity research, specifically developing tools and solutions for incident response and forensics to be used by Security Operations Center (SOC) professionals. In the following sections I will describe my current progress in this direction and how I envision this work in the future. I am also advised by Dr. Michael Wesch who is a Professor of Anthropology at K-State and an accomplished researcher in his field.

4 My Current Research Progress

Currently I am embedded as a member of the Security Incident Response Team (SIRT) at K-State. I started off by learning the procedure for blocking hosts compromised by malware, handling Digital Rights Management violations etc. By becoming part of the SIRT team and doing the activities they do on a daily basis I was slowly gaining their trust but still that was not enough to do research and development in that environment. But I was able to break that barrier by showing them that I can improve the efficiency of their work through a very small tool I built for them.

4.1 Caching database for faster incident response

We receive alerts on malicious network flows from a number of trusted sources as well as from our own Snort instance. We extract the IP address belonging to us and block that host from the network until that host is cleaned up. The IP address most of the times is of the border firewall hence the internal host is actually NATed and one has to extract its internal IP address from firewall logs. From the internal IP address the corresponding MAC address is obtained from ARP logs. K-State's network generates approximately 70 GB of firewall log data on an average day during weekdays. Finding a "connection built" entry for a given timestamp, firewall IP address and port number sometimes may take upto 3 minutes and the ARP lookup takes another minute or so. Then looking up user information for that MAC address takes up another minute, so the whole process may take up to 5 minutes. I decided to speed up firewall and ARP log lookup by building a database of “connection builds” along with IP address to MAC address mapping with timestamp. I worked on parsing out the authentication logs using which we can identify the user whose device is compromised using MAC address to user ID mapping. The challenge however was that I cannot keep adding data to this database as it will exhaust our storage space very quickly but then most of the alerts we get are usually not more than 3 days old. So I decided to build a database that caches this mapping information for 3 days including the current day. The window keeps moving purging the data for the earliest day.

I first built the database using MySQL but then I was falling behind on log collection from the firewall by 25 minute or so. The reason was that I was indexing on a few attributes for faster lookup and since the inserts were in real-time the indexes have to be adjusted for each insert and also have to be committed to the disk. After reading a bit on database optimization I decided to adopt NoSQL solutions that are efficient in handling applications that have high real-time write throughput, such as log collection. I finally settled on MongoDB which stores data as JSON type (schema-less) objects. Now the inserts into the database are keeping up with the logs. To speed up lookups, I used a combination of timestamp, firewall IP address and port number as the key (MongoDB creates index on keys by default).

Once I built this cache database and found it to be stable, I asked the analyst to use it. First of all, he was extremely happy to see the speedup in incident response the tool has brought him from 5 minutes down to 2 seconds. Secondly he was interested in sharing more data to expand the database infrastructure. Through this work I was able to gain his trust and hence convinced him to shared more useful data for research.

4.2 Enhanced research and development facilitated by the trust obtained from the analyst

The analyst became more open to new ideas after the speed up in incident response brought in by my caching database. Figure 1 shows the overall idea we arrived at after a brainstorming session. We would like to build an infrastructure for incident response and forensics at K-State. The following data will be collected into our “threat intelligence database.” We would leverage the Collective Intelligence Framework (CIF) [8] client to extract data wherever applicable.

- TCP and UDP connection information
- ARP data from core routers throughout the campus
- Remote IP addresses from the alerts raised by our Snort instance
- People to MAC address mapping
- Reputation information on IP addresses from REN-ISAC, Shadowserver, Robtex, Emerging Threats etc.

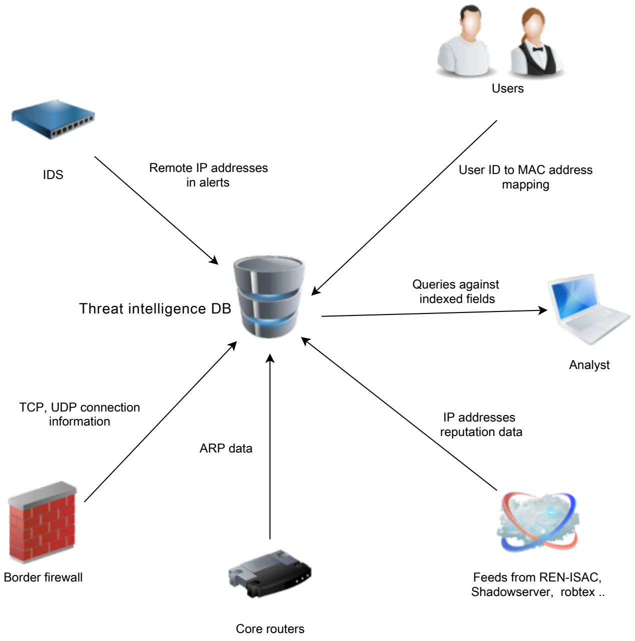
Users
IDS
Remote IP addresses in alerts
User ID to MAC address mapping
Threat intelligence DB
Queries against indexed fields
Analyst
TCP, UDP connection information
ARP data
IP addresses reputation data
Border firewall
Core routers
Feeds from REN-ISAC, Shadowserver, robtex ..

Fig. 1. Threat Intelligence Framework

The following sections will throw light on two use cases for the above infrastructure. It is important to note that the following are also incident response techniques that were once “tacit knowledge” in the minds of the analyst made explicit using the tool chain I developed.

UDP connection tracking on the firewall: We get frequently alerted for malware infection, such as Zeus, that use UDP as transport protocol for communication. When we see an alert for this event, often times the timestamp on the alert does not exactly match the one in the caching database. The reason is due to the fact that a firewall’s notion of connection when it comes to UDP starts when the first packet is sent between the socket end points and ends when there is no flow between them for a timeout period, approximately 2 minutes in the case of Cisco devices. We have observed some cases where the malware sends a packet to its P2P Command and Control server but then keeps the connection open for days together (sending at least one packet just within the timeout window). Hence, when we observe an alert the timestamp might be any time in between these days and logging just the connection built time might not be sufficient. I am now modifying database such that it records the start and end times for the UDP flows. This is a very subtle fact in firewall connection logging which came up only after the follow up discussion with the analyst on my caching database.

Identifying malicious flows in real-time: The next idea I am currently working on is to identify internal hosts talking to malicious remote end points in real time. Our caching database, as mentioned before, will contain the NAT translation information along with the remote end point for each flow. Now, if we have reputation information for the remote IP address involved in a flow we can classify the flow as malicious or non-malicious with certain confidence. There are a number of sources where one can extract this reputation information as mentioned in the figure and this has to be combined to extract a overall confidence on the maliciousness of an IP address. I am planning to leverage my previous work [9] in which we investigated the use of Dempster-Shafer theory for belief combination of evidence. This will again help the SIRT analyst to speed up the incident response process.

Web interface for the framework: Since I submitted the paper for NSPHD I started working on a web interface for the framework. The analyst's initial intention of using this framework was as a automated ticket generation system. Whenever an alert is received all the necessary information required to block the compromised host from the network are sent to the network team. The network team blocks the host and then it now becomes the responsibility of the SIRT member in whose network the compromised host was found to make sure the host is cleaned up and then requests a unblock request. Currently the task of finding out the responsible SIRT representative is a manual process performed by the analyst in the security office. Using a mapping of IP address blocks to the responsible SIRT representative all the required information can be sent directly to the responsible SIRT representative through email. This function has been added to the framework. There was an important lesson learned from the anthropology perspective which I describe in the lessons learned section.

Tracking stolen laptops: Once the analyst started using the web interface he wanted to enhance it with other features that might be useful in some of the incidents he handles very often. Among all the enhancements one of them is used in a very interesting scenario. Students have their laptops stolen quite frequently during the Fall and Spring semester. The perpetrator is usually a student. Assuming the victim knows the MAC address of his/her laptop and reports to us. We are lucky if the perpetrator uses the University's authenticated wireless service then we can associate his University ID against the MAC address of the stolen device and even catch him/her red-handed using the Access Point (AP) information. It gets interesting when the perpetrator uses the Guest Wireless system that does not require any authentication. There is an appliance used to monitor user experience of web-servers by creating profiles of users by intercepting sessions between the user and the servers. The University uses the appliance for the campus web-servers, the most common among the students being the online course manager. Any student who needs to access it must authenticate to the web-server and that means we can retrieve the unique University ID for that user along with the IP address and timestamp used in that access. Now if our perpetrator, being a student, even though is using only Guest Wireless is accessing his/her course materials using the stolen laptop, we can still identify him/her using the appliance' logs. This feature also helps us to identify owners of compromised machines even after the Wireless Access Points flush their authentication information.

Application identification for connection data: One of the ways I have found security analysts identify anomalies in the network is by looking at unusual protocol-port pairs. Though HTTP on port 9090 is not necessarily malicious it is definitely an anomaly worth looking into, more so when analysts at other networks report the same, independently. Our University uses a tool that identifies the application from packets through Deep Packet Inspection (DPI). But the tools suffers from issues common to other solutions such as inability to integrate with other frameworks due to proprietary data format, poor search feature in the interface, and limited caching due to licensing limitations. After a session with the analyst it looked like annotating the firewall log entries with the application information will be useful during incident analysis and response. Following that discussion I have started looking into open source DPI solutions and will be adding the feature to the framework very soon.

Automating phishing scam detection: Whenever a phishing email is received the security team responds with a honeypot university identifier and when the team sees a login using that identifier in the future the IP address associated with that activity is noted. The noted IP address is than matched against connections made using the same address but for different identifiers within a time window. Usually it is the case that the attacker has harvested many University accounts through phishing along with the honyepot account and tries one by one in quick succession. This is being done manually but using my framework this process can be automated. A simple back end script can check one of the the logs that collects user logins to our web servers and single sign-on page. If it sees a login from the honeypot account it is trivial to extract the other user accounts possibly compromised by the attacker.

5 Lessons Learned from my Experience and Future Work

I had to develop some useful tools and convince the SIRT analyst that its worth his time to talk to me in brainstorming for more ideas. To get to his tacit knowledge on improving the incident response infrastructure I had to be part of his environment and become his apprentice before he shares anything with me. It is significant to note this entire process took me almost two months.

During my field work I discovered a caveat that is often described in anthropology where the observer transitions into the observed. The SIRT at K-State is organized such that each department in the University has a designated person responsible for handling infections within their department. The main purpose of the framework is to make the data necessary to handle incidents available to the SIRT so that the primary security analyst can focus on more sophisticated analysis of infections. After the paper submission I developed a web interface so that whenever an alert is raised the SIRT member responsible for the incident can query for firewall and ARP data to submit a network block ticket for the infected host. I gave a demonstration of the framework with the web interface to the SIRT a few weeks back in which my advisor was also present. Once the demonstration was over my advisor noticed an inefficiency that I completely missed. When an alert is received for an infection there is no way currently to identify automatically the responsible SIRT member. That means each member has to process the alert, extract the subnet or user information to check if they are responsible for that incident, which is highly inefficient. Instead my advisor suggested to add to the database the association between IP blocks and SIRT members so that whenever an alert is received all the relevant information to block the host can be emailed directly to the appropriate SIRT member. We are now working on this optimization.

The lesson here is that when you are doing an anthropological field work there is a caveat that you might be absorbed by the community that you are part of that you start thinking more like them. As in this case I was part of the security team at K-State and was doing my field work by observing the security analyst and developing tools to automate the repetitive tasks. Typical thought process in the operations environment is to get things done quickly rather than thinking about the long term vision and I got consumed by it too. On the contrary, my advisor was visiting the office only once a week to get updates on my work at the security office and was not part of the field work. But he was able to see the inefficiency in the framework which me and the analyst completely missed. In fact this entire idea of threat intelligence framework arose from a discussion with one of our collaborators who works remotely, that neither me nor my advisor believed was worth trying. The bottom line is the farther you are from the community you want to study more daring you are to try new ideas. Field workers in anthropological study need to step into the shoes of the members of the community they are studying but also remind themselves often that they are just observers not one among them.

Another observation I made through the filed work is that it takes considerable time to enable the SIRT professionals to use any new tool or procedure. For example, except for the analyst with whom I was personally working with in the security office other analysts were not very interested in what I was doing. But then there was a compromise incident in the campus and neither the main analyst nor me was available at that time hence it had to be taken care by the other analysts in the office. I then got requested for access to the tool and then I scheduled a demonstration of the tool to the entire team in the office. There is a cultural difference between how things are done in a research environment and an operations environment. In research we always look for areas for improvement and optimization but in operations people do not try new things unless they are needed. In my case I got requested for access mainly because the analysts who wanted to take the response for the incident were not equipped enough to do the job in the traditional way, as it was before by accessing the various log collection servers for collecting relevant information. I realized as a researcher working alongside operations staff I need to be very patient before the research results actually get utilized in production.

I will continue my current efforts in taking an anthropological approach to research in incident response and forensics hoping to build usable solutions for the SIRT analysts. Building small tools such as those I mentioned in the previous section are focused toward understanding the "tacit knowledge" embedded into the mind of the security analyst. This tacit knowledge, once obtained will help me to identify more profound research problems leading to mathematical modeling of incident response and forensics.

I received great feedback from the participants of the workshop that will help me in carrying out the research further. Especially information on hierarchy in other SOC environments that might pose a challenge for effective communication of research ideas, evaluation, and smooth hand off of the tools to SOC staff were very helpful. I would like to thank the organizers of NSPW for accepting my work in this inaugural NSPHD track.

References

1. J. S. Brown and P. Duguid. Knowledge and organization: A social-practice perspective. *Organization science*, 12(2):198–213, 2001.
2. P. Dourish and G. Bell. *Divining a digital future: mess and mythology in ubiquitous computing.* MIT Press, 2011.
3. J. Elyachar. Before (and after) neoliberalism: Tacit knowledge, secrets of the trade, and the public sector in egypt. *Cultural Anthropology*, 27(1):76–96, 2012.
4. C. Geertz. Deep play: Notes on the balinese cockfight. *Daedalus*, 101(1):1–37, 1972.
5. J. Lave and E. Wenger. *Situated learning: Legitimate peripheral participation.* Cambridge university press, 1991.
6. M. Polanyi and A. Sen. *The tacit dimension.* Peter Smith Gloucester, MA, 1983.
7. S. C. Sundaramurthy, L. Zomlot, and X. Ou. Practical ids alert correlation in the face of dynamic threats. In *the 2011 International Conference on Security and Management (SAM11), Las Vegas, USA*, 2011.
8. W. Young. Collective Intelligence Framework. `https://code.google.com/p/collective-intelligence-framework/`.
9. L. Zomlot, S. C. Sundaramurthy, K. Luo, X. Ou, and S. R. Rajagopalan. Prioritizing intrusion analysis using dempster-shafer theory. In *Proceedings of the 4th ACM workshop on Security and artificial intelligence*, pages 59–70. ACM, 2011.