Intelligent Interaction Honeypots for Threat Hunting within the Internet of Things

James Gregory Surber
School of Technology & Computing (STC)
City University of Seattle
Seattle, WA USA
surberjames@cityuniversity.edu

Morgan Zantua
School of Technology & Computing (STC)
City University of Seattle
Seattle, WA USA
zantuamorgan@cityu.edu

Abstract—As the Internet of Things (IoT) grows exponentially, security is falling farther and farther behind. Several new initiatives show promise for expanding the privacy and security around these devices in the future. But what about the billions of devices already out there in the wild? Security researchers are responsible for developing the tools and procedures for discovering these devices quickly, understanding the risks they bring with them, and developing tools to mitigate those risks to more manageable levels. Honeypots and honeynets have traditionally supported this work in traditional IT. However, the challenges faced by the highly distributed, incredibly heterogeneous Internet of Things make deploying such tools difficult and costly. Recent research in honeypot architectures explicitly designed for the chaotic nature of the IoT ecosystem brings a new sense of hope that may lead to significant improvements in IoT security. There is still much work to do, but research continues. IoT cybersecurity experts and threat hunters are developing strategies for securing this new frontier of technology. This study will lay the foundations for an intelligent and highly interactive honeypot solution that can scale with the researchers' requirements, providing a much-needed framework for deploying targeted IoT honeypots.

Keywords—Internet of Things, IoT, threat hunting, cybersecurity, honeypot

I. Introduction

The Internet of Things (IoT) poses an ever-increasing risk to the security of the connected world. The number of devices that do not require human interaction is growing exponentially, already exceeding the human population many times over. Many experts expect the IoT to grow to as many as 80 billion unique devices as early as 2025 [1]. Finding a way to secure these devices must become a greater priority, or we should expect to see more incidents like in 2016's Mirai botnet attack [2]. Governments and standards bodies are developing new laws and frameworks to increase security for the future of the IoT [3] [4] [5]. However, they all share an inability to solve the problem for the billions of IoT devices already deployed. IoT security researchers must find solutions to secure those devices and the billions more currently making their way to market.

One of the primary hurdles with security in the IoT comes from the lack of proper understanding of its scope. Several tools already exist to help find new devices as they appear on the Internet: shodan.io being one famous example [6]. With IPv6's popularity with IoT device manufacturers increasing, the simple discovery process will become far more complex. With IPv4 addressing, even using network address translation (NAT) techniques, the entire public-facing Internet address space can be scanned using traditional IT network scanning tools in a matter of just a few hours or days. With the switch from a 32-bit address space to a 128-bit address space, the available pool of addressable space grows well beyond the ability to use the same techniques to manage [7].

Next, we must develop a better understanding of the actual security posture of the various devices found on the Internet. Since there has historically been no strong guidance or requirements for device creation or programming, there is now a vast gulf in understanding how these devices operate. By examining the inner workings of these devices, a security researcher may uncover vulnerabilities in their fundamental operation that bad actors could exploit. Furthermore, by understanding the risks imposed by these vulnerabilities, device manufacturers can develop mitigations to protect their devices.

Static analysis of IoT device firmware may uncover some vulnerabilities in their operation. Still, dynamic, interactive analysis of devices under attack is often the only way for security researchers to discover and understand their adversaries' tools, tactics, and techniques. In traditional IT environments, the deployment of honeypots, specialized systems that mimic production environments without exposing actual production resources, allows security researchers a testbed wherein they can capture activity from real-world bad actors. Only through the adoption of new IoT-centric honeypots can threat hunters tackle the unique problems posed by this unique environment. However, in the IoT ecosystem, deploying honeypots is not as simple as it is in more traditional areas. This paper will discuss two recent developments in the IoT honeypot arena that might provide the right kind of capability an IoT honeypot needs.

II. Threat Hunting with Honeypots

Cybersecurity researchers and threat hunters have long used honeypots to research current cybersecurity trends as part of the traditional IT environment. By simulating real-world systems, attracting malicious attackers to try their luck hacking them, and collecting all the details of that attacker's tools, tactics, and techniques, security researchers can get a unique view into how the bad guys are operating. The IoT landscape, however, is different. The sheer volume of devices, coupled with the vast diversity of device types (everything from coffeepots to industrial control systems), makes tuning a single honeypot to capture all the right kinds of data complicated. Unfortunately, IoT security can only ever play catch-up with the malicious actors without that data. Instead, we need a way to understand how the bad actors operate, where they look for vulnerabilities, what tools they use, and how they configure them if we are to play in the same league.

III. Honeypots in the IoT

The use of honeypots in the enterprise IT world has been widely compelling, but the realities of the IoT ecosystem do not bind themselves to the same model. The diversity of devices, new communication protocols, and machine-to-machine interactions versus human-to-machine interactions complicate the whole landscape make for a convoluted mess for IoT security researchers. Acien, Nieto, Fernandez, and Lopez [8] examined the current state of IoT honeypots. First, the authors have taken an exhaustive look into the eleven most deployed IoT-style honeypots, cataloged their use cases, and discussed how they had successfully deployed them in the past. Next, the authors detailed how they conducted their research, noting the difficulties with uncovering and deploying IoT honeypots that successfully fulfill the requirements security researchers need. Utilizing five search engines specialized in the techniques of IoT device detection, namely shodan.io, censys.io, reposify.com, thingful.net, and wiggle.net, the authors were able to ensure their research covered an actual cross-reference of the most popularly deployed IoT devices. They then conducted vulnerability research on those devices, utilizing popular vulnerability research tools like exploitee.rs and the Common Vulnerabilities and Exposure (CVE) database to help them uncover the exploitable IoT devices in the wild. Using that collated information, the authors proposed a methodology for honeypot creation and deployment to ensure the most significant return on investment for real-world IoT deployments. Utilizing virtualized IoT device simulations, including common IT honeypot frameworks Dionaea [9], Cowrie [10], and Honeytrap [11], the authors built a deployable honeypot framework inside their research lab. By injecting known malicious software and utilizing previously discovered attack techniques, they could simulate an IoT infection and study the resultant impacts on the rest of their environment. This low-interaction style of a honeypot, where the interaction between device and attacker is limited to simple command-response options, provides some insight into the attack vectors of simple IoT attacks but is easily detected by more competent attackers. Therefore, low-impact honeypots are much easier to deploy but offer less actionable information for the security researcher.

Some researchers have turned their attention to creating high-interaction honeypots to uncover more actionable intelligence regarding the nature of IoT attacks. Tabari & Ou [12] consider a multi-phased approach to developing a robust IoT honeypot, building from their successes with low-interaction honeypots to craft more realistic high-interaction honeypots. At their heart, the authors argue, IoT honeypots offer a challenge to the researcher. The volume of unique IoT devices within the ecosystem, both in services offered and in physical connective properties, makes creating believable honeypots complex. By developing and deploying simple, low-interaction honeypots, the authors gathered foundational data on the types of attacks perpetrated in the wild. Using that knowledge, they crafted their first low-interaction honeypot, emulating an IP-enabled camera called "Honeycam." As their study of the attacks and various interactions outsiders had with their honeypot, the authors increased the complexity and interaction level, even to the point of presenting real-time video. As this complexity increased, the researchers gathered more data about the tools and techniques employed by the attackers. Although the breadth of coverage was small, just one IP camera type, it demonstrated the need for increased complexity of honeypot interaction if they are to be used as security research tools.

Building on the concept of designing high-interaction honeypots, Wang, Dou, Sang, Zhang, and Huang [13] investigated two basic types of risks widely found within the IoT ecosystem, namely weak authentication protocols (SSH and telnet) and command injection. The authors proposed and built a hybrid IoT honeypot framework called IoTCMAL [13] for capturing increasingly sophisticated attack techniques. Utilizing a multi-pronged process, they began with honeypots running low-interactive services, like SSH and telnet, in virtual environments harvesting connection attempts. Meanwhile, they would run more high-interactive services utilizing physical devices. Finally, the traffic was forwarded from edge devices into an internal monitoring network for heightened interaction and observation. As a result, the authors got a deeper view of the true nature of attacks. In total, the authors deployed IoTCMAL on 36 virtual private cloud (VPC) instances in 13 cities around the world. Virtual private clouds are virtualized versions of physical networks, implemented in one of the many cloud network providers public cloud environments. As a result, they were able to uniquely identify eight malware families and at least 11 distinct groups of attackers.

Next, we will discuss two advanced high-interaction IoT honeypots. Both have their weaknesses, but both have the potential to be absolute game-changers in the field of IoT security research.

First, Hakim, Aksu, Uluagac, and Akkaya [14] examined the specific protocol Universal Plug and Play (UPnP) and its use in IoT in general and as a potential IoT honeypot framework fundamental mechanic. The authors discuss the near-ubiquity UPnP sees in the IoT arena, with everything from intelligent switches to intelligent hubs to surveillance cameras often interacting via the UPnP protocol. The authors' research discovered more than 1.65 million UPnP-enabled devices accessible from the public Internet in 2018. The honeypot framework the authors created, dubbed U-PoT [14], is said to emulate any known UPnP-enabled IoT device given sufficient description documentation. Designed to operate specifically with devices provided no other authentication mechanism if the UPnP protocol can establish connectivity between the server and the end device, pertinent information about that device can be harvested to create a fingerprint for use by the server in establishing additional services. By manipulating that fingerprint in the form of a description document, the researchers have shown they can automatically create several IoT honeypots emulating real-world IoT devices in a high interaction environment. These emulated IoT devices were even able to fool vendor-supplied management and control applications. The truest limitation to this framework is the UPnP protocol itself. With widely publicized, unmitigated vulnerabilities, vendors may be turning away from utilizing the UPnP protocol, giving this framework a deadline for usability.

Finally, we will look at the fascinating research of Luo, Xu, Jin, Jia, and Ouyang [15], who targeted their research on the areas of machine learning and how to focus it on the IoT honeypot problem. Utilizing a technique called "intelligent-interaction," the authors propose creating a machine-learning algorithm to teach a honeypot how to interact with an attacker. Their idea is to make the attacker believe they interact with an authentic and vulnerable IoT device while harvesting as much about the interaction as possible. As argued by the authors, the first step is to craft an active probing tool for cataloging expected IoT device responses to specific interactions. High-speed scanning tools like masscan [16] or unicorn scan [17] can create lists of Internet-accessible systems. Additional probes of suspected IoT devices can then determine running services, interaction capabilities, and response chains. Finally, this data is ported into a central database for collation and interpolation by the intelligent-interaction algorithm. The authors are careful to point out that an essential part of the scanner tool is to be "polite" about these interactions, minimizing the burden on both the Internet as a whole and the individual devices. Once the data was collected and categorized by IoT device type, the authors created catalogs of information, including potential ports, services, and interaction steps. Using the data gathered, the authors deployed test versions of their honeypot, named IoTCandyJar, [15] in their closed lab environment. They subjected it to simulated attacks and noted the learned responses the honeypot was able to define. Their research indicates a probability for machine learning-driven honeypots to emulate high-interaction IoT devices and aid in cybersecurity research in the field. Once the algorithm is finely tuned enough, the next problem to overcome is the diverse nature of the IoT ecosystem. While this technique might apply to groups of like devices, routers or IP cameras, for example, the type and nature of interactions between so many different devices will require additional fine-tuning to be truly applicable to multiple categories of IoT devices.

A. *Test Implementation of IoT Honeypots*

Two primary sources of direct data were configured for the final analysis. First, a series of publicly available honeypots were established, built on the Deutsche Telekom/T-Mobile "T-Pot" framework. The framework included 23 different individual honeypots, each tuned to collect data regarding a particular subset of Internet traffic. Among these were popular and well-established honeypots "cowrie," an ssh/telnet honeypot; "rdpy," a python-based implementation of the Microsoft remote desktop protocol; "Dionaea," a low-interaction honeypot explicitly designed to capture malware and attack information; and "conpot," a honeypot explicitly designed to monitor industrial control system protocols. Each of these honeypots collected data from connections and connection attempts made to their specific monitoring subsystems.

The second primary data source came from the network packet sniffing tool, Wireshark, implemented immediately in front of the host system's network interface [18]. Wireshark is a network monitoring tool commonly used by network and system engineers to capture and investigate all network traffic that passes through the monitored interface [19]. Data gathered from Wireshark was correlated with logs from the various honeypots. Analysis of discrepancies in traffic seen within each tool was used to instruct modifications to honeypot monitoring.

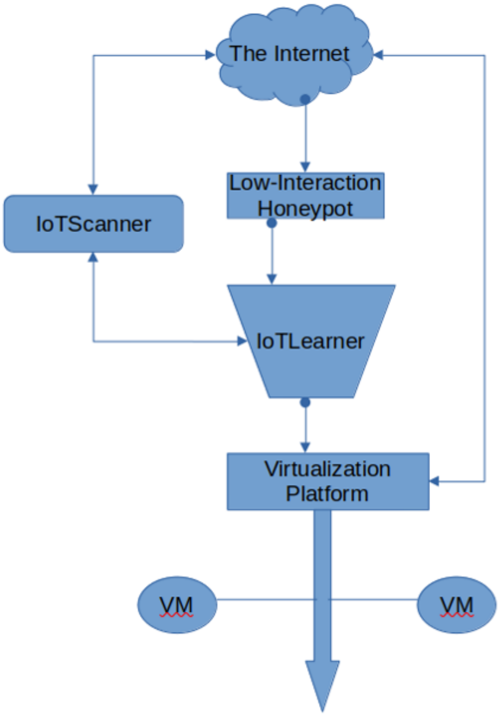
The Wireshark data was carved into individual files based on the originating IP address utilizing some simple bash scripts. This process limits the volume of data needed when comparing potential hits from both Wireshark and the honeypots. Next, network traffic from attackers with corresponding IP addresses found in both the individual Wireshark captures and honeypot logs was manually reviewed. Wireshark packet capture data provides information about attackers that do not show up within the honeypot logs, opening more exciting questions, like: Why are they not in both? Were they attempting to attack a port or service not covered by one of the honeypots? Or was the attacker able to avoid being caught by the honeynet?

Additionally, data aggregation has been automated and archiving steps created to minimize manual data handling. Having automated tools to do the gross manipulation of data saves considerable time. This data archive is stored on a separate volume to avoid resource restrictions on the primary honeypot system.

IV. Intelligent Interaction Framework Development

IoT security researchers and threat hunters need to develop a deeper understanding of bad actors' tools, tactics, and techniques to enable them to build effective countermeasures. This process starts by gathering as much data as possible about the attacks in the environment. The volume of already deployed IoT devices, with their varied firmware, operating systems, custom protocols, and cobbled together codebases, makes this a daunting task. By devising a framework to allow newly accumulated knowledge to build on previously gathered knowledge, we can make a model to utilize the power of machine learning to create a more intelligent honeypot solution. As found in traditional IT environments, honeypots offer some helpful stepping-off points for IoT honeypot deployments. IoT security researchers will have to overcome many limitations with the existing model to provide the support needed in this dynamic and quickly expanding ecosystem.

Here is a proposal for an intelligent-interaction honeypot (IIH). Built from the fundamental discussions of "IoTCandyJar" [15] for machine-learning enabled intelligent interaction, “Honware” [20] for a scalable virtualization platform, and “SIPHON” [21] for traffic redirection, the idea will still require significant work to realize its potential.



First, a simple, low-interaction honeypot, tuned to monitor the specific ports and protocols desired, is attached to the Internet. This honeypot could be an already existent one, like cowrie, conpot, or others, if the ports and protocols of interest are covered.

A second system, called IoTScanner[15], is used to perform IoT device discovery and enumeration, either within the researcher's lab or on the Internet. Besides being a simple port scanner used to discover IoT devices, researchers may use it interactively to connect specifically with the ports and services they wish to study. For example, IoT security researchers can harvest connection information, protocol-specific details, command options like request-response pairs, and other device-specific connection information by connecting to targeted IoT devices [13]. Once collected, they use that information to map possible interactions with each supported protocol and device.

A third system, IoTLearner [15], is connected to both the low-interaction honeypot and IoTScanner, receiving data from each for processing through its machine learning algorithm. This algorithm takes data from the low-interaction honeypot and IoTScanner, utilizes sequence prediction to build a neural network of potential responses to determine the following action an attacker might take and crafts the most appropriate response the attacker is expecting. The resultant processing of this neural network allows the system to build a tree of possible interactions, each focused on keeping the attacker connected and engaged.

When an attacker starts their attack by probing the ports and services of an IP address, the IIH responds to this probe with an intelligently selected appropriate response, posing as the requested device. Next, the attacker chooses their attack and sends their commands. Depending on the type of device the IIH is simulating, there are multiple possible attack pathways. The IIH system will respond with a selected response based on the attacker's command. Assuming the IIH response is what the attacker expected, they will continue sending their attacks, which will fall down the tree of potential reactions from the IIH.

As confidence grows, the IIH will spin up a virtual high-interaction honeypot, mimicking the device specifications in the ongoing communication. These new virtualized IoT devices can be deployed via Quick Emulator (QEMU) emulation techniques as outlined in "Honware: A Virtual Honeypot Framework for Capturing CPE and IoT Zero Days" [20]. This will allow them to mimic the appropriate firmware and system architecture easily. Once that virtual device is ready, the IIH will forward new traffic via wormhole switching [21] directly to the newly established high-interaction honeypot. Now provided with additional resources, such as new services/ports, discoverable data, additional system connections, and so on, the attacker is enticed to continue their interaction. In addition, each unique command they send is collected, dissected, and cataloged for further study and eventual implementation into the algorithm's command pathways.

Once the connection is severed, the virtual machine is saved and archived for forensic evaluation [22]. Newly discovered tools, tactics, and techniques are then coded and loaded into the IoTLearner's machine learning algorithm to increase its usefulness.

V. Conclusion and Future Work

The future state of IoT security is getting much-needed attention. Development of new security frameworks and models is underway, legislation is being written and enacted, and developers are learning that privacy, at least, is going to be demanded of their services and devices as we advance. However, billions of unsecured devices are out there already, interacting with each other and the world. We must develop ways to identify, diagnose, and treat these devices now. Honeypots can provide a much-needed glimpse into the tools, tactics, and techniques of bad actors in the IoT arena. Still, they are not so simple to develop or implement in the IoT as they are in more traditional enterprise IT environments. Therefore, it is incumbent upon the security researchers, the threat hunters of the IoT ecosystem, to craft their own tools, tactics, and techniques to counter those bad actors. Recent work in IoT honeypotting has shown some novel approaches to some of the problems facing us in that space. First, the Honware virtual honeypot framework [20] lays the foundation for dynamic, realistic honeypot creation based on actual IoT firmware utilizing complex architecture virtualization techniques. Next, IoTCandyJar's [15] machine learning methodology, where complexity within the honeypot can build relative to the external interactions it receives, provides a way to keep bad actors engaged longer. Finally, the novel use of network "wormhole" techniques taken from the SIPHON architecture [21] allow for the hand-off of connections from lower interaction honeypots to newly virtualized high-interaction honeypots. Together, these establish a new intelligent-interaction honeypot framework. However, the development of the intelligent-interactive framework is only the first step. Work needs to continue to develop the machine learning algorithms that will parse the data derived from the IoTScanner and the manual researcher efforts.

Additionally, to better understand the implications of this research, future efforts can help expand the research base, developing and deploying additional honeypots to more precisely target IoT protocols. The field of IoT is immense, so the scope for such an endeavor is also massive, but it is not insurmountable. There are already a few giants laying foundational works upon whose shoulders the next generation can stand.

REFERENCES

[1] C. Crane, “Re-Hashed: 27 Surprising IoT Statistics You Don’t Already Know,” Hashed Out by The SSL StoreTM, Feb. 17, 2021. https://www.thesslstore.com/blog/20-surprising-iot-statistics-you-dont-already-know/ (accessed Jun. 26, 2021).

[2] M. Antonakakis et al., “Understanding the Mirai Botnet,” in Proceedings of the 26th USENIX Security Symposium, 2017, pp. 1093–1110. [Online]. Available: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis

[3] “2018 California Code : Civil Code - CIV : DIVISION 3 - OBLIGATIONS : PART 4 - OBLIGATIONS ARISING FROM PARTICULAR TRANSACTIONS : TITLE 1.81.26.a - Security of Connected Devices : Section 1798.91.04.,” Justia Law. https://law.justia.com/codes/california/2018/code-civ/division-3/part-4/title-1.81.26.a/section-1798.91.04/ (accessed Jun. 26, 2021).

[4] R. L. Kelly, “H.R.1668 - 116th Congress (2019-2020): Internet of Things Cybersecurity Improvement Act of 2020,” Dec. 04, 2020. https://www.congress.gov/bill/116th-congress/house-bill/1668 (accessed Jun. 26, 2021).

[5] M. R. Warner, “Text - S.734 - 116th Congress (2019-2020): Internet of Things Cybersecurity Improvement Act of 2019,” 2019, [Online]. Available: https://www.congress.gov/bill/116th-congress/senate-bill/734

[6] S. Verma, “Searching Shodan For Fun And Profit,” 2014. [Online]. Available: https://www.exploit-db.com/docs/english/33859-searching-shodan-for-fun-and-profit.pdf

[7] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, “Demystifying IoT Security: An Exhaustive Survey on IoT Vulnerabilities and a First Empirical Look on Internet-Scale IoT Exploitations,” IEEE Commun. Surv. Tutor., vol. 21, no. 3, pp. 2702–2733, 2019, doi: 10.1109/COMST.2019.2910750.

[8] A. Acien, A. Nieto, G. Fernandez, and J. Lopez, “A comprehensive methodology for deploying iot honeypots,” 15th Int. Conf. Trust Priv. Secur. Digit. Bus. Trust. 2018, vol. 11033 LNCS, no. TrustBus, pp. 229–243, 2018, doi: 10.1007/978-3-319-98385-1_16.

[9] “GitHub - DinoTools/dionaea: Home of the dionaea honeypot,” GitHub. https://github.com/DinoTools/dionaea (accessed Jul. 21, 2021).

[10] “GitHub - cowrie/cowrie: Cowrie SSH/Telnet Honeypot https://cowrie.readthedocs.io,” GitHub. https://github.com/cowrie/cowrie (accessed Jul. 21, 2021).

[11] “GitHub - honeytrap/honeytrap: Advanced Honeypot framework.,” GitHub. https://github.com/honeytrap/honeytrap (accessed Jul. 21, 2021).

[12] A. Z. Tabari and X. Ou, “A first step towards understanding real-world attacks on IoT devices,” arXiv, 2020, [Online]. Available: https://arxiv.org/pdf/2003.01218.pdf

[13] B. Wang, Y. Dou, Y. Sang, Y. Zhang, and J. Huang, “IoTCMal: Towards A Hybrid IoT Honeypot for Capturing and Analyzing Malware,” IEEE Int. Conf. Commun., vol. 2020-June, no. November, 2020, doi: 10.1109/ICC40277.2020.9149314.

[14] M. A. Hakim, H. Aksu, A. S. Uluagac, and K. Akkaya, “U-PoT: A Honeypot Framework for UPnP-Based IoT Devices,” arXiv, 2018, [Online]. Available: https://arxiv.org/pdf/1812.05558.pdf

[15] T. Luo, Z. Xu, X. Jin, Y. Jia, and X. Ouyang, “IoTCandyJar: Towards an Intelligent-Interaction Honeypot for IoT Devices,” Black Hat 2017, pp. 1–11, 2017.

[16] R. D. Graham, “GitHub - robertdavidgraham/masscan: TCP port scanner, spews SYN packets asynchronously, scanning entire Internet in under 5 minutes.,” GitHub, 2021. https://github.com/robertdavidgraham/masscan (accessed Jul. 21, 2021).

[17] J. Louis, “unicornscan(1) - Linux man page,” unicornscan(1) - Linux man page, 2004. https://linux.die.net/man/1/unicornscan (accessed Jul. 21, 2021).

[18] F. Chantzis, I. Stais, P. Calderon, E. Deirmentzoglou, and B. Woods, Practical IoT Hacking. No Starch Press, Inc., 2021.

[19] C. Sanders, Practical Packet Analysis: Using Wireshark to Solve Real-World Network Problems. No Starch Press, Inc., 2017.

[20] A. Vetterl and R. Clayton, “Honware: a virtual honeypot framework for capturing CPE and IoT zero days,” in eCrime Researchers Summit, eCrime, 2019, vol. 2019-Novem. doi: 10.1109/eCrime47957.2019.9037501.

[21] J. Guarnizo et al., “SIPHON: Towards scalable high-interaction physical honeypots,” in CPSS 2017 - Proceedings of the 3rd ACM Workshop on Cyber-Physical System Security, co-located with ASIA CCS 2017, 2017, pp. 57–68. doi: 10.1145/3055186.3055192.

[22] F. Pouget and M. Dacier, “Honeypot-based Forensics,” AusCERT Asia Pac. Inf. Technol. Secur. Conf., pp. 1–15, 2004, doi: 10.1.1.137.4169.