A famous aphorism of David Wheeler goes - "Any problem in computer science can be solved with another level of indirection"[1]. Indeed, such abstractions have proven essential for us to design and manage large complex computing systems. However, as core design primitives are obscured, these indirections can also engender new security challenges. My research looks past these abstractions to study the core functionalities of modern computing systems. Through identifying the security and privacy challenges within these systems, I build defense mechanisms that prevent security violations while meeting the design and operational requirements of system users.

Prior Work Overview. My work has applied this research philosophy to the areas of Emerging Cloud Architectures, Internet of Things (IoT), and Enterprise Security. I have designed information flow control (IFC), access control and auditing mechanisms in improving security in these domains. I have found that the "stateless, ephemeral, and isolated" abstractions of serverless functions mask a reality in which functions are cached in memory for long periods of time, enabling attackers to gain quasi-persistence. I addressed this by building systems for serverless authorization and auditing that mitigate these vulnerabilities [WWW20; ACSAC20; SACMAT21; USENIXSec22]. In IoT automation platforms, device interactions are obscured by abstract natural language automation rules. My work has enabled us to reason about the attack surface of these platforms through applying natural language processing to rule descriptions [CCS19] and designing user studies [SP23]. In large enterprises the task of security auditing is often simplified by inspecting individual endpoint system logs without regard for application or network context. This lost context can be derived using data provenance - metadata that describes the history of digital artifacts. I designed provenance-based auditing framework to integrate these disparate event streams and create a unified view of the enterprise [NDSS20a; NDSS20b].

Impact and Future Roadmap. My work is highly relevant in the modern computing world because I identify real-world problems through partnerships with industry (e.g., Samsung Research [WWW20], VMware [InSub23a; InPrep23a]) and I ground my work by surveying the real-world threat landscape [CCS19; SP23; InSub23b]. I believe that to build practical and widely deployable secure systems, it is necessary to both understand the requirements of systems as well as identify the root cause of violations of these requirements. While developing security solutions, I am eager to tackle problems at scale and to ensure compatibility with legacy systems. I have designed and open-sourced security systems (e.g., Valve, Will.IAM, Alastor) that are in active use by other researchers in the community. In future, I wish to continue my serverless security investigation through redefining access control using control flow integrity mechanisms. I will explore the move towards a hybrid "Extended Detection and Response" (XDR) model of Enterprise Security through the design of provenance-based analysis and user studies with security analysts. In longer-term, I will investigate the security requirements of marginalized populations in designing better smart-systems that prioritize their comfort and dignity.

Securing Emerging Cloud Architectures **[WWW20; ACSAC20; SACMAT21; USENIXSec22]**

Serverless computing empowers web developers by liberating them from hassles of cloud infrastructure management, and presents an abstract model of application development where monolithic cloud applications are decomposed into short-lived stateless isolated functions. Unsurprisingly, serverless has become popular with industry, including Samsung Research of America where I started exploring this space while investigating the potential of serverless for IoT. There, I found out that the abstracted view of serverless applications is incongruous with the underlying systems. For example, serverless platforms typically reuse virtualized function containers, thus breaking the "stateless" and "isolated" abstractions. Further, individual function-centric Role-Based Access Control (RBAC) policies in cloud are unable to encode the security requirements of function workflows and create the potential of data exfiltration through downstream authorized functions and legitimate platform APIs.

[1] `http://bwlampson.site/Slides/TuringLecture.htm`

Serverless Authorization. To address these problems, I designed Valve [WWW20] - the first language-agnostic serverless information flow control system. Rather than applying function-by-function access controls, Valve monitors end-to-end function workflows within an application. To prioritize transparency and interoperability, Valve does not modify function code and instead instruments the function-instance (container) to monitor the function's file and network behaviors. Individual function behaviors are then coalesced to provide insight into the information flows across the application. These flows can then be translated into a security policy that restricts each function according to the principle of least privilege. To evaluate Valve, I profiled and specified a security policy for Nordstrom's open-source retail application 'Hello,Retail!', demonstrating that Valve imposed just 2.8% runtime overhead and 6.25% deployment overhead on functions.

Valve demonstrated that a workflow-centric perspective on access control can more effectively mediate serverless functions, especially when compared to the commonly-used RBAC policies. Taking the idea of workflow-centric authorization a step further, I designed Will.IAM [ACSAC20] that identifies the function workflow that will be followed by a web request at the point of ingress. Will.IAM then proactively mediates the end-to-end workflow by examining the permissions required at each step. This makes it possible to reject requests with insufficient permission earlier, preventing the costs of partial execution and reducing the attack surface of the application. Will.IAM thus provides additional evidence of the power of workflow-centric authorization in the serverless paradigm.

Through Valve and Will.IAM, I demonstrated that the innate event-driven nature of serverless can enable precise and lightweight tracking of data flows. In subsequent work, I leveraged this unique feature in the design of SCIFFS [SACMAT21], a serverless security analytics platform. Third party analytics services aggregate telemetry data from many clients for proactive threat identification. Unfortunately, this data aggregation simultaneously risks exposing sensitive client data. SCIFFS leverages serverless cloud computing in designing a Decentralized Information Flow Control (DIFC) model to prevent the risk of exposure through precise flow-tracking, and also eliminates the problem of label explosion through deploying DIFC in a serverless setting for security analytics platforms.

Serverless Auditing. When existing serverless access control mechanisms fail it is necessary to detect and investigate attacks. Traditional approaches to system auditing lack the understanding of serverless semantics and are unable to accurately track serverless attacks. I built Alastor [USENIXSec22], the first provenance-based auditing framework for serverless. Alastor captures provenance information at both container and platform layers and then meaningfully reconstructs attack paths by bridging the semantic gap between lower layer system events and application layer flows. Alastor is both function- and language-agnostic, and can easily be integrated into existing platforms with minimal modification. Alastor imposes 13.74% overhead in response latency which is consistent with traditional provenance-based auditing techniques. In exchange Alastor provides significantly improved forensic capabilities as compared to commercially-available monitoring tools.

Misconfigurations in cloud applications' access management policies lead to many of the attacks discussed above. We studied the extent of this problem in real-world through creating a dataset of 1600 open-source AWS lambda serverless applications. We designed a formal access policy model encoding serverless access primitives and operations. With an application's access policies and configuration files as inputs, this model can generate reachability graphs explaining access paths to sensitive resources within the application (e.g., sensitive resources reachable from public internet). We discovered that 227 applications in our dataset expose at least one publicly accessible read path to a resource [InSub23b].

Enterprise Security [NDSS20a; NDSS20b]

Another cost paid for the layered abstractions in designing computing systems is the resulting difficulty in investigating security attacks because the clues are scattered around the stack. I worked on combining multiple sources of telemetry information for effective attack investigation at large scale enterprises during my VMware internship. Threat detection systems deployed at such enterprises fire threat alerts based on attack signatures. However, these alerts only consider local attack behaviors, and lack context to investigate the root cause and to correlate subsequent malicious activities during persistent multi-stage multi-node attacks.

My early experience in combatting this problem includes designing the OmegaLog [NDSS20b] system. OmegaLog demonstrated that application-layer semantics are usually found in applications' event logs, and through integrating such application logs into system logs, the attack context lost between different layers of software could be reconstructed. OmegaLog models application logging behaviors using static binary analysis and these models reconcile application-layer events with system-layer events for better understanding of attack behavior. Omegalog enabled accurate attack reconstructions with 4% runtime overhead without requiring instrumentation to system or application code.

From my serverless intrusion investigation work Alastor I realized that efficient threat investigation in large distributed systems requires correlation of telemetry information from different layers of the software stack and the endpoint instances. Armed with these understandings, I proposed a distributed persistent attack investigation system on top of CarbonBlack (i.e., Endpoint Detection & Response (EDR) solution at VMware) [InPrep23a]. I defined a provenance data model based on the events captured by CarbonBlack that encodes both network and host events from endpoints in a whole-provenance graph. Then an anomaly score is assigned to each edge in the provenance graph based on the frequency with which related events have happened before in the organization. These scores are propagated and aggregated over paths in the provenance graph to reconstruct the most suspicious attack paths. The graph events can be matched against a knowledge base of adversarial Tactics, Techniques, and Procedures to prune irrelevant paths and guide the attack path reconstruction algorithm in the correct direction navigating through multiple nodes.

In another thread of threat investigation research at VMware, I am involved in exploring machine learning based process behavior modeling [InSub23a]. We developed a concise yet expressive vocabulary to map process-level events (and their properties) to sequences of tokens that serve as input in building a self-supervised language model trained with events captured by CarbonBlack. This foundational model then can be easily fine-tuned for further downstream tasks (e.g., malware detection, alert triaging) instead of developing ad-hoc, end-to-end models for each task. Not only does this avoid costly manual feature development, but it also requires less labeled training data while maintaining accuracy.

Auditing systems work on the assumption that the captured events are true. However, industry-surveys report that up to 72% of EDR analysts had encountered tampered logs, suggesting attackers regularly erase forensic evidence present in system logs to hide their footprints. To solve this problem, I have built Custos [NDSS20b] to provide secure and expressive audit logs for attack investigation. Custos is a system consisting of two inter-related mechanisms: a tamper-evident logging protocol to generate integrity proofs for system audit logs and a decentralized auditing framework to verify the proofs and detect log tampering in real time. The decentralized auditing algorithm is executed by networked host nodes acting as auditors. The auditors randomly initiate audit challenges to a subset of its peers to verify the integrity proofs of peers' logs. Custos logging protocol is three orders of magnitude (1000×) faster than prior solutions and incurs less than 7% runtime overhead over insecure logging. Custos' auditing protocol can detect violations in near realtime even in the presence of a powerful distributed adversary with minimal (3%) network overhead. Additionally, Custos provides strong probabilistic security guarantees. In a network of 100 hosts, even with 50 compromised nodes, only 4 auditor nodes can lower the probability of attack success to less than 5.88%.

Internet of Things Security [CCS19; SP23]

To make smart-home management easy for the end-users, automation platforms (e.g., IFTTT, Zapier) offer simple dashboards to configure a home using abstract natural language rules. These trigger-action rules can easily connect IoT devices and online services, but users may not anticipate the potential of security risks because these simple rules hosted on opaque closed-source third party servers obscure the complex information flows among different IoT components during rule execution. For example, 'If humidity goes over 55% then open the window' and 'If humidity is less than 45% on weather channel, then turn the humidifier on' - these rules could be leveraged by an attacker to gain physical entry to a house if executed simultaneously. I identified six types of such vulnerable rule interactions and developed an automatic information flow analysis technique called iRuler [CCS19]. iRuler infers inter-rule dependencies in a smart home based on the inspection of rules' text descriptions available on the trigger-action platform websites. iRuler uses NLP techniques to extract meaningful rule components which are then semantically connected to form an information flow (IF) graph to visualize smart home configurations. This technique reduces false flows in the IF graph by 72% as compared to an exhaustive approach and increases the accuracy of formal property verification using this IF graph in reasoning about security vulnerabilities in a smart home.

Violations studied in iRuler raised a question - do these violations frequent in the day-to-day security and privacy problems faced by the consumers? To answer this question, I became part of a research collaboration with University of Maryland in designing a user-study to understand the most likely risks of harm posed by smart devices [SP23]. This project explored how smart devices are misused by device owners' everyday associates. In a preliminary characterization survey with 100 participants we captured the kinds of unauthorized use and misuse incidents participants have experienced or engaged in. Then, in another survey with 483 participants the prevalence of these incidents was assessed within a demographically-representative population. The findings from this study show that unauthorized use of smart devices is widespread (experienced by 43% of participants), and that misuse is also common (experienced by at least 19% of participants). Through a focus on everyday abuses rather than severe-but-unlikely attacks, this work sheds light on the most prevalent security and privacy threats faced by smart homeowners.

Future Directions

Data Security for Distributed Cloud Applications. My work thus far has developed methods for measuring and enforcing the security of serverless cloud applications. Moving forward, I am interested in developing new conceptual frameworks for reasoning about security in cloud platforms. The move to serverless has spurred developers to decompose monolithic applications into small reentrant functions in which control flow transfers occur over network APIs. It seems only natural, then, that new access control paradigms should look to recent advancements in software security for mechanisms through which to provide finer-grained authorization. For example, I plan to explore how Control Flow Integrity (CFI) mechanisms can be applied to serverless applications and, in turn, whether known limitations of CFI such as control flow graph overapproximation are mitigated by the serverless setting. Concurrently, I am also interested in developing methods that allow us to better understand the current state of cloud (in)security. I intend to combine my policy-layer analysis [InSub23b] with application-layer control flow analysis to measure serverless application overprivilege. By identifying unnecessary permissions in serverless security policies, it will then be possible for me to develop methods of automatic policy rewriting that bring cloud applications closer to satisfying the principle of least privilege. I am also interested in exploring other real-world cloud security challenges, such as the management of API keys in the cloud, and analyzing security-efficiency tradeoffs in cloud-based federated learning applications.

Data Provenance as the Foundation of Extended Detection and Response (XDR). Industry is moving towards a hybrid XDR model in which endpoint and network event streams are fused to provide integrated monitoring of suspicious activity. However, without a reliable method of merging these event streams, XDR will continue to suffer from many of the past limitations of endpoint and network monitoring. I plan to employ provenance based data models in combining different sources of telemetry information in a causality preserving way to aid XDR-based threat intelligence. The application of data provenance opens new opportunities in this space including subgraph analysis to identify event-patterns to detect network wide attacks (e.g., DDoS), summarize benign events to reduce false alerts, and application of machine learning in graph anomaly detection. Moreover, I want to explore the human aspect of threat analytics research. While threat analytics engines automatically raise alerts, the conclusive investigation and response in case of a breach are conducted by security analysts. They often apply their own set of "thumb-rules" (look for suspicious events involving '/tmp' directories, specific usernames or processes) along with (or instead of) sophisticated EDR dashboard features in their investigation - these are quick and easy wide nets to catch organization wide anomalies. I am interested in designing a large-scale user study to systemize and encode the collective knowledge of security analysts that can help in identifying more effective ways of threat investigation.

IoT Security and Privacy for Marginalized Populations. IoT devices have become ubiquitous not only in daily life but also in security sensitive settings, such as border control and assisted senior living facilities. My IoT user study revealed that people are concerned about widespread unauthorized use and privacy violations. I am interested in studying how such problems evolve as we move from a privileged population to a more marginalized population (e.g., refugees, immigrants and senior population). For example, the usage of new IoT based smart infrastructure for border protection (including robotic lie detectors, eye-scanners and voice-imprinting softwares) has been scrutinized by human rights researchers asking whether these tools enable even harsher surveillance of migrant people. While physical constructions like a border wall are heavily criticized by media, the same purpose achieved in a digital way is sometimes passed off as bias-free. I am interested in exploring in what ways data collection using such technology impacts the privacy of refugees and immigrants and puts them at risk. Is there potential for security violations in usage and storage of such sensitive data that reaches beyond the intended context and purpose of the data collection? Through understanding the security and privacy violations in such scenarios I will further be able to address core design limitations of smart IoT systems that hinders ethical deployment of such systems in the context of marginalized population.

References

[CCS19] **Pubali Datta**, Qi Wang, Wei Yang, Si Liu, Adam Bates, and Carl A. Gunter. "Charting the Attack Surface of Trigger-Action IoT Platforms". In: *ACM SIGSAC Conference on Computer and Communications Security*. 2019.

[NDSS20a] Riccardo Paccagnella, **Pubali Datta**, Wajih Ul Hassan, Adam Bates, Christopher Fletcher, Andrew Miller, and Dave Tian. "Custos: Practical tamper-evident auditing of operating systems using trusted execution". In: *Network and Distributed System Security Symposium*. 2020.

[NDSS20b] Wajih Ul Hassan, Mohammad Ali Noureddine, **Pubali Datta**, and Adam Bates. "OmegaLog: High-fidelity attack investigation via transparent multi-layer log analysis". In: *Network and Distributed System Security Symposium*. 2020.

[WWW20] **Pubali Datta**, Prabuddha Kumar, Tristan Morris, Michael Grace, Amir Rahmati, and Adam Bates. "Valve: Securing Function Workflows on Serverless Computing Platforms". In: *The Web Conference*. 2020.

[ACSAC20] Arnav Sankaran, **Pubali Datta**, and Adam Bates. "Workflow Integration Alleviates Identity and Access Management in Serverless Computing". In: *Annual Computer Security Applications Conference*. 2020.

[SACMAT21] Isaac Polinsky, **Pubali Datta**, Adam Bates, and William Enck. "SCIFFS: Enabling Secure Third-Party Security Analytics using Serverless Computing". In: *The ACM Symposium on Access Control Models and Technologies (SACMAT)*. 2021.

[USENIXSec22] **Pubali Datta**, Isaac Polinsky, Adam Bates, and William Enck. "Alastor: Reconstructing the Provenance of Serverless Intrusions". In: *USENIX Security Symposium*. 2022.

[SP23] Phoebe Moh, **Pubali Datta**, Noel Warford, Adam Bates, Nathan Malkin, and Michelle L. Mazurek. "Characterizing Everyday Misuse of Smart Home Devices". In: *Accepted to appear in IEEE Symposium on Security and Privacy (S&P)*. 2023.

[InSub23a] Mahmood Sharif, **Pubali Datta**, Andy Riddle, Kim Westfall, Matthew Lentz, Adam Bates, and David Ott. "Flexible Distributed Representations for Efficient Endpoint Security". In submission to IEEE Symposium on Security and Privacy (S&P). 2023.

[InSub23b] Isaac Polinsky, **Pubali Datta**, Adam Bates, and William Enck. "Graph Reachability Analysis of Serverless Security Policies". In submission to IEEE Symposium on Security and Privacy (S&P). 2023.

[InPrep23a] **Pubali Datta**, Adam Bates, Matthew Lentz, and David Ott. "Distributed Provenance Analysis for Endpoint Detection and Response Systems". In preparation for submission to USENIX Security Symposium. 2023.