I aspire to design a trustworthy Internet of Things (IoT). In contrast with traditional ubiquitous computing, IoT devices use new user-interaction modalities, are more complex, and are interconnected. Thus they introduce new attack surfaces which can result in financial, emotional and physical harm to individuals: the Mirai botnet exploited myriads of insecure IoT devices to bring down a swathe of popular online services; adversaries took advantage of vulnerable smart baby monitors to scream at babies; intelligent vehicles were remotely attacked allowing an adversary to take control of steering, brake and transmission functions.

My work has focused on smartphone security, which is the de facto user interface to consumer-facing smart environments and devices. I have unearthed design flaws in real-world systems, which affect millions of users. In particular, I discovered new side channels on Android, found connectivity issues with wireless devices and exposed remote code execution threats. In response I designed new security mechanisms that can be directly integrated into popular smartphone operating systems, application markets and network routers.

During my Ph.D., I fostered collaborations between 32 researchers across 9 institutions in both industry and academia. My work resulted in publications in major systems and security conferences but also had industry impact. Google introduced security enhancements to Android after we unearthed system flaws; Samsung and Hewlett-Packard Enterprise recognized my work with prizes while some of the technology I invented resulted in pending patents. I plan to build on my collaborations with industry and academia to help solve more pressing real-world problems in emerging consumer-facing IoT devices and environments.

Unearthing Real-World Threats

Detecting important real-world threats requires a systematic analysis of systems. I have extensively scrutinized the Android operating system and downloaded and analyzed thousands of mobile applications. I have used manual, static and dynamic code analysis, network traffic fingerprinting and finite state machine models to unearth significant threats in smartphones which affect millions of users [11, 6, 4, 5, 9, 8].

Side-channel Attacks: The Android operating system uses a combination of mandatory and discretionary access control schemes to guarantee process isolation. However, we showed how a zero-privileged mobile application could leverage its unfettered access to network traffic time series and a smartphone's speaker on/off status to infer a user's location, identity, disease and financial information, and the user's driving route [11]. We demonstrated these through targetted attacks on popular applications such as Twitter, WebMD, Yahoo Finance and Google Maps, which are downloaded millions of times from Google Play (video demonstrations can be found here). Since then, Google has responded with restricting third-party application access to other processes' network traffic information. This work was published at *CCS '13*.

Mis-Bonding Problem: Imagine connecting your smartphone to a Bluetooth Pulse Oximeter. This will allow you to use the device's companion app on a smartphone to track your heart rate and your blood oxygen level. We observed that any application with the Bluetooth permission could surreptitiously connect and read information from medical and fitness Bluetooth devices [6]. We call this the device mis-bonding problem to highlight the failure of the mobile system and the networking protocol to create application-level bonds. We found that this does not only affect Bluetooth devices, but also NFC devices, audio devices, incoming sensitive SMSs from banking services and social media, and connections to the Internet [4]. Our security analysis on Google Play apps revealed leakage of medical and financial information, fitness data, and password and authentication tokens, affecting millions of users. Video demonstrations are available here and here.

Confused Origin Attacks: We found that the Android platform is confused about the origin of an app update [5], the origin of dynamically loaded web code [9] or a permission owner [8]. We showed that a malicious application can exploit such problems to hijack the installation process on the most popular application installer apps such as Android Amazon AppStore, Google Play, Baidu etc. [5]. Furthermore, the majority of Google Play apps, utilize in-app embedded browsers to display web content. As we demonstrated on popular apps from a postal company, a pharmacy and a job search company, untrusted web code can gain

access to platform and application functionality, resulting in severe leakage of sensitive user information [9]. In some cases, Android relies on a permission model to govern access to resources. However, it cannot distinguish whether a permission has been declared by the system or an untrusted third-party app. This allows untrusted apps to silently access a device's microphone, camera, contacts, SMSs and much more [8]. We have reported these findings to Google, which has acknowledged them as serious security vulnerabilities.

Strengthening the Smartphone Ecosystem

Understanding of real-world threats should drive secure system design. During my Ph.D. I developed practical solutions buttressing the Android operating system and tools utilizing access control theory, natural language processing, machine learning and binary code intrumentation to detect data leakage [6, 4, 8, 9, 7, 2] .

Discretionary and Mandatory Access Control on Android: Android uses a strict mandatory access control (MAC) scheme based on a Linux kernel security module (Security-Enhanced Linux or SELinux for short) to isolate system from untrusted processes. However, it fails to address mis-bonding problems because of three main reasons: it is coarse-grained; requires prior knowledge of resources; requires security experts to write SELinux-compatible policies. In [6] we built *Dabinder*, which can distinguish between untrusted apps and leverages user input to govern connections with unanticipated personal Bluetooth devices. In [4] I built a new system, *SEACAT*, which can uniformly protect access to multiple external resources (Bluetooth, NFC, Audio, SMS, Internet) while offering stronger security guarantees. SEACAT extends SELinux on Android and combines discretionary and mandatory access control to enforce application-level access policies for users and enterprises respectively. Dabinder and SEACAT were implemented in the Android open source project and their effectiveness and efficiency were verified on real smartphones. Dabinder was published at *NDSS '14*. SEACAT was published at *NDSS '15* and a patent has been filed.

Android Permission Model and Security of In-app Embedded Browsers: Android uses system permissions to protect platform resources. At the same time it allows untrusted third-party applications to regulate access to their exported functionality and data, by declaring their own custom permissions. However, the platform fails to distinguish between system and third-party permission declarations leading to confused origin attacks. In *CUSPER* I introduce a new backward-compatible runtime permission model which ensures that untrusted apps cannot exploit custom permissions to silently access sensitive APIs and other applications' data [8]. Moreover, Android does not provide mechanisms for developers to protect their apps from untrusted web code. In [9] I propose a new declarative policy language for application developers to enforce origin-based access rules. Rules are enforced by a runtime system, *Draco*, which can be seamlessly deployed with a mere application update. Both systems were implemented and evaluated on real smartphones. CUSPER will be presented at *NDSS '18* while *Draco* was published at *CCS '16*.

Tools to Detect Data Leakage: Increasingly, mobile apps can utilize camera data and off-the-shelf vision libraries to perform common recognition tasks. In [7] we performed user studies and application code analysis to identify that most of them extract information from visual data in unexpected ways. To improve user awareness, we built a record and replay tool (*CamForensics*), which uses code instrumentation and machine learning to identify the information Android apps extract from camera data. Android apps also increasingly use advertising libraries. Libraries run within the same process as their host app and hence can inherit all its privileges. Given the fact that advertising networks rely on detailed user profiles, my research put forth a very important question: *what if advertising networks took full advantage of their information access capabilities on Android* [2]*?* To answer this question I built a tool called *Pluto*. Pluto uses dynamic code analysis, novel NLP techniques and out of the box machine learning and data mining algorithms to determine an app's sensitive information leakage to its advertising libraries, and assign it a risk score. Both tools can be used in application markets to provide privacy signals to users when downloading an app. CamForensics was published at *SenSys '17*; Pluto is open-sourced, its code is available here, and it was published at *NDSS '16*.

Future Work

I plan to build tools, protocols, systems and frameworks to enable **trustworthy Internet of Things**. I envision a secure, privacy-aware open-sensing environment where data from IoT endpoints facilitate personalized services. Towards this, I will work on the design of IoT systems and relevant machine learning models that are robust to adversarial inputs and produce trusted outputs. I want to see operating systems and networking protocols that facilitate seamless secure interconnectivity across endpoints. Moreover, I want to develop secure authentication mechanisms and adaptive user-driven access control and management schemes for consumer-facing IoT (cIoT) systems. My current research agenda has the following main thrusts:

Trust: Establishing trust in an open-sensing environment, entails understanding the security of devices and designing guidelines and frameworks for their development. In the short term, I will complement my work on IoT security [10] using network and information flow analysis to better understand different endpoints' encryption, authentication and authorization practices, and how information is collected, processed and propagated. Moreover, I will build on the techniques I introduced at [4], to regulate resource access across endpoints and services; I will further explore strengthening endpoints of different complexities with trusted platform modules to enable hardware and software-assisted root of trust and remote attestation interfaces and protocols. Another challenge is provisioning and managing cryptographic keys. In consumer-facing IoT, a traditional centralized certificate authority is not always practical. I plan to experiment with the use of blockchain technology for decentralized key management. In the long term, I want to design domain-specific guidelines and frameworks for the development of trustworthy IoT endpoints, services and applications.

Interconnectivity: IoT devices are heterogeneous. cIoT platforms such as SmartThings and IFTT, or presence advertising protocols such as UPNP and BLE facilitate interoperability. Unfortunately all of them suffer from security issues [10]. Moreover, intelligent vehicles are equipped with advanced sensors, communicate with either the infrastructure (V2I) or other vehicles (V2V), and run machine learning algorithms for positioning and object detection. In [1] I utilize such technology to improve the positioning of every car encountered on the road by a sensor-rich car. Nonetheless, this requires utilizing and sharing massive sensing data volumes with a service provider which stimulates grave privacy concerns. To address these issues, I plan to better understand the threats in real-world scenarios by studying the devices' susceptibility to identity, confidentiality and integrity attacks; how machine learning models in sensor-rich devices such as drones and vehicles can be manipulated by adversarial inputs; and what kind of inferences a curious service provider can perform in crowdsourced cIoT data. In the long term I want to harden existing and develop new secure IoT networking protocols, and explore the use of privacy preserving technologies to ensure the anonymity of the users and endpoints, and the confidentiality of the shared data in an open-sensing environment.

Awareness, Authentication & Management: A throng of consumer-facing IoT devices are developed from a variety of manufacturers. However, there is a lack of solutions which can help users perceive the implications of selecting one device over another. I plan to continue developing tools (like Pluto [2] and CamForensics [7]) to facilitate disambiguation of complex textual descriptions of products, applications and services and provide hints regarding the security practices of the devices. Moreover, users interface with IoT devices through new interaction modalities such as voice and gestures. Such signals can be spoofed or recorded and replayed. To mitigate such threats, I plan to leverage physical and network signals to prove liveness and user intent to the authenticator device. In some environments, devices either lack proper authentication or erroneously trust all other devices on the same network. To address this I built *HanGuard* [3] a distributed fine-grained role-based access control system for smart homes. In [3, 6, 4] I observed that resources in IoT environments might be unknown at development time and can be ephemeral at runtime. Moreover, in cIoT we expect non-experts to set-up access control rules. In the short term I plan to continue working on usable and practical reference monitors for emerging IoT environments. In the long term I want to develop frameworks for building user-driven, context-aware access control IoT systems with secure user and device authentication.

References

[1] S. Demetriou, P. Jain, and K.-H. Kim. Codrive: Precise automobile positioning in urban environments. In *IEEE International Conference on Computer Communications (IEEE INFOCOM)*, 2018.

[2] S. Demetriou, W. Merrill, W. Yang, A. Zhang, and C. A. Gunter. Free for all! assessing user data exposure to advertising libraries on android. In *ISOC Network and Distributed Systems Security Symposium (NDSS)*, 2016.

[3] S. Demetriou, N. Zhang, Y. Lee, X. Wang, C. Gunter, X. Zhou, and M. Grace. Hanguard: Sdn-driven protection of wifi smart-home devices from malicious mobile apps. In *ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, 2017.

[4] S. Demetriou, X. Zhou, M. Naveed, Y. Lee, K. Yuan, X. Wang, and C. A. Gunter. What's in your dongle and bank account? mandatory and discretionary protection of android external resources. In *ISOC Network and Distributed Systems Security Symposium (NDSS)*, 2015.

[5] Y. Lee, T. Li, N. Zhang, S. Demetriou, M. Zha, X. Wang, K. Chen, X. Zhou, X. Han, and M. Grace. Ghost installer in the shadow: Security analysis of app installation on android. In *IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2017.

[6] M. Naveed, X. Zhou, S. Demetriou, X. Wang, and C. A. Gunter. Inside job: Understanding and mitigating the threat of external device mis-bonding on android. In *ISOC Network and Distributed System Security Symposium (NDSS)*, 2014.

[7] A. Shrivastava, P. Jain, S. Demetriou, P. L. Cox, and K.-H. Kim. Camforensics: Understanding visual privacy leaks in the wild. In *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2017.

[8] G. S. Tuncay, S. Demetriou, K. Ganju, and C. A. Gunter. Resolving the predicament of android custom permissions. In *ISOC Network and Distributed Systems Security Symposium (NDSS)*, 2018.

[9] G. S. Tuncay, S. Demetriou, and C. A. Gunter. Draco: Uniform and fine-grained control of web code access on android. In *ACM SIGSAC Conference on Computer & Communications Security (CCS)*, 2016.

[10] N. Zhang, S. Demetriou, Y. Tian, X. Mi, X. Wang, and Q. Feng. Sok: Understanding iot security through the data crystal ball: Where we are now and where we are going to be. In *Preprint arXiv:1703.09809*, 2017.

[11] X. Zhou, S. Demetriou, D. He, M. Naveed, X. Pan, X. Wang, C. A. Gunter, and K. Nahrstedt. Identity, location, disease and more: Inferring your secrets from android public resources. In *ACM SIGSAC Conference on Computer & Communications Security (CCS)*, 2013.