Earlence Fernandes – Research Statement

I. Overview

I take a broad view of computer systems security. The classical interpretation focuses on the host operating system and network levels. However, computer systems *themselves* have evolved. Today, they are distributed, embedded and capable of learning. Although these evolved computer systems provide benefits in many areas including energy efficiency, productivity, convenience and safety, unless they have the correct security foundations, they amplify traditional computer security threats, giving those threats wider reach and impact and also introduce new classes of threats. My research goal is to ensure that human society will gain the benefits of these evolved computer systems without the security and privacy issues.

To achieve this vision, I use a systems security approach — (1) apply the security mindset to deeply understand the new ways in which these evolved computer systems are vulnerable; and (2) construct principled systems-level defenses that offer end-to-end guarantees. In general, I take particular inspiration from the secure design principles of computer systems, as articulated by Saltzer and Schroeder [26]. My current research agenda focuses on two themes that I believe are critical enablers of the evolved computer systems discussed above.

(1) Internet-scale user-centered automation systems. Due to the commoditization of cloud computing, all of our digital data and physical devices are now available on the Internet sitting behind vendor-specific cloud services and APIs (e.g., emails, personal files, messages, health data, smart home and building devices). Coupled with the rise of user-centered automation platforms, digital data and physical resources can be seamlessly composed. For example, it is currently trivial for a non-programmer to create automations that save emails of a certain subject in online storage folders, or control garage doors based on a motion sensor. Until a few years ago, this was very difficult for end-users to achieve, and required expert programmers. Clearly, this democratization of automation brings many benefits. Unfortunately, recent work, including my own, has established that user-centered automation platforms do not have the correct security foundations [1], [3], [20]. Left unchecked, these platforms pose a long-term and large-scale security threat — if they are compromised, our digital data and physical devices are left open to misuse by attackers. This research theme focuses on establishing the security foundations of Internet-scale user-centered automation. My core insight is that the well-defined structure of automation permits building strong security guarantees.

(2) AI-driven physical systems. Machine learning has supercharged many areas of computer systems. Simultaneously, the community has established that machine learning is fundamentally vulnerable to many kinds of attacks. This problem is particularly pernicious when we use ML in safety- and security-critical systems like autonomous vehicles or authentication systems in mixed reality. My key insight is that ultimately, machine learning is a piece in a larger computational pipeline and thus, understanding and building robustness *requires* an end-to-end *systems* perspective. Building on this insight, this theme characterizes the vulnerability of AI-powered physical systems with the final goal of creating robust computer systems that use ML in their pipelines.

Impact. I believe that if there are ways to benefit society *after* publishing papers, then it is important to facilitate such positive impact. Earlier in my career, I collaborated with Samsung to fix security issues in their smart home product line. Based on that experience, ideas from my subsequent work in secure smart home platforms (e.g., FlowFence [19]) was adapted into Samsung products. I am currently exploring how defense techniques from my work on secure Internet-scale automation can be applied to products from IFTTT (one of the largest user-centered automation providers). The Stop sign that we used in our physical adversarial example work is on display at the Science Museum in London (and has been added to their permanent collection of artifacts), and helps raise awareness in the general public about emergent security issues.

I often engage with the press when society at large would benefit from awareness about my results — my work on smart home security and physical adversarial perturbations work received widespread coverage, and raised awareness among key stakeholders. For example, I have advised a member of Congress on Internet of Things security, and several US government agencies including the FTC, the National Academies, and the JASONs on security issues related to the deployment of ML systems in critical infrastructure.

II. Research Themes

I will discuss recent research results that focus on two areas: (1) Secure Internet-scale user-centered automation and (2) Secure AI-driven autonomous vehicles. The interested reader may refer to the bibliography for pointers to my work in smart home security [10]–[12], [18], [19], [23], privacy-respecting data analysis systems [21], [22], and my early work in smartphone security [7], [15]–[17], [24], [25], [28].

A. Security and Privacy Foundations of Internet-Scale User-Centered Automation

There is a web-based API for everything — digital resources like email, health data, and online storage; physical resources like homes, buildings, cities, and industrial equipment. End-users, who are not trained in programming, create simple trigger-compute-action automations using these web APIs with the help of so-called Trigger-Action Platforms (TAPs). Common examples of TAP programs include routing emails with certain subject lines to specific folders, locking a garage door based on a user's location or an energy company monitoring the health of field devices and then triggering alerts for maintenance. TAPs have become a critical substrate for Internet-scale automation because of their widespread compatibility with diverse digital and physical resources and their ease-of-use that allows non-programmers to create automations (e.g., the popular IFTTT platform boasts 20 million users). Unfortunately, this makes them attractive targets for attackers who wish to wreak havoc at scale by compromising these systems and stealing user data and manipulating online services, including physical systems.

This research theme establishes the security and privacy foundations of trigger-action platforms. I have the singular goal of ensuring that the TAP only executes user-created rules without doing anything else, such as stealing user data or manipulating their physical and digital resources beyond what the user intends. I will achieve this guarantee under a threat model where the TAP is not trusted to carry out automation securely. I make this threat model assumption because it captures several subclasses of attackers who may compromise the system to various degrees. Securing computer systems in the face of compromise is a long-standing challenge that has eluded us. My thesis is that the unique TAP environment permits innovations in practically adapting powerful machinery from program analysis, trusted computing and techniques for computing on encrypted data.

Design Spectrum Exploration. The concrete research agenda is to explore the different trade-offs possible between functionality, security, usability and performance and ultimately contribute a framework that will help future TAP designers to make informed choices when building these systems. Below, I discuss concrete systems I have built that explore these trade-offs.

Cryptographic protections for TAP rules (IEEE S&P 2021) [6]. I investigated cryptographic techniques for protecting the confidentiality and integrity of data in a practical systems context. Our driving insight is that TAP rule computations are simpler and more well-defined than general-purpose software. Coupled with inherent latencies in TAP computing (because they operate across web services), we can tailor cryptographic primitives for computing on encrypted data and mask their costs.

Concretely, a trigger-action rule involves several operations on the trigger service data, and the computation follows a pre-defined path of: (1) sensitive data coming from the trigger service; (2) the TAP computing on that data to produce a result; (3) the action service performing tasks based on that output. Furthermore, the computation is unidirectional proceeding from the trigger service to the TAP to the action service without multiple rounds of communication. The trigger and action services do not have any dependency on each other — they are independent services. Thus, the TAP acts as a function evaluator that knows how to interact with the trigger and action services. My approach designates the TAP cloud service as a malicious function evaluator and uses garbled circuits (GCs) as a primitive to execute functions on encrypted trigger data. This will guarantee confidentiality of user data and integrity of the rule through the properties of garbled circuits as formalized by Bellare et al. [4].

To achieve this design, my work addresses the following open problems: (1) Although from a theoretical perspective, any computable function can be evaluated using GCs, performance matters. Therefore, what types of operations exist in trigger-compute-action rules and can we efficiently execute them using appropriate circuit representations? What innovative circuits do we need for the TAP domain? (2) The TAP is fully malicious implying that we must use malicious-secure protocols; however, they can become expensive and introduce multi-round communications between parties which drastically changes the TAP computing paradigm. Is there a design point where we can mitigate these issues while supporting real-world trigger-action rules?

At a high-level, this project found that indeed, it is possible to run TAP rules using garbled circuits in the presence of a malicious function evaluator using *only a semi-honest implementation of GCs*. This is possible because the TAP environment affords a trusted circuit generator and a trusted party that operates on the results of GC evaluation. In terms of circuit computations, the project found that string operations are widely used, and thus, we need an efficient technique to run string operations on GCs. Thus, the project also contributes a novel oblivious deterministic finite automata (DFA) technique that encodes DFAs as Boolean circuits (that also aggressively optimizes for XOR gates).

Practical data minimization (USENIX Security 2022) [5]. Although the above project provides strong confidentiality and integrity, it makes a trade-off: performance overhead is large, it can only support TAP rules that can be expressed efficiently as garbled circuits, and it requires changes to the TAP and the surrounding services (i.e., trigger and action services). This project seeks a different trade-off point. Specifically, can we support arbitrary TAP rule computations *without* changing the TAP cloud service itself? This is a compelling question because if such a system can be constructed, it would imply that users of TAPs can be protected immediately, without waiting for vendors to implement security updates.

This exploration culminated in the minTAP project, which will appear at USENIX Security 2022. The project builds on the theory of data minimization [2] and shows how basic program dependency analyses can help "minimize" TAP rule access to sensitive data, thus limiting the privacy impact if the TAP is malicious.

Consider the TAP rule from Fig. 1. Under the current TAP paradigm, this rule will receive all email data from a specific sender. Thus, the rule is not least privilege. To achieve its functionality, the rule only needs the email subject line, and only when the subject line contains the word "Confidential." Thus, TAPs are overprivileged and in the event they are compromised or otherwise malicious, the attacker is able to access a lot more than than the user intends. Within the encrypted TAP approach discussed earlier, this is not a problem because all data is encrypted when the TAP is computing on it. However, as I mentioned, the trade-off there is higher performance overhead and limited support for TAP rule computations. With minTAP, I am interested in supporting the full expressivity of TAP programming *without* changing how the TAP cloud itself operates.

The core observation here is that the trigger service can dynamically withhold information that a rule does not need for achieving its functionality, thus *minimizing* the rule's access. In the example, this means that only the email subject line is transmitted to the TAP cloud, and only when the subject line contains the user-specified keyword. minTAP achieves this using lightweight program dependency analyses (static and/or dynamic) that can compute the minimum set of information the rule needs to accomplish its stated function. Although minimizing programs in general is undecidable, the well-defined nature of TAP programs allows us to create practical minimization algorithms.

```
let str = NewEmail.Subject
if (str.indexOf('Confidential') ===
↪  -1) {
  Slack.postToChannel.skip()
} else {
  Slack.postToChannel.setMessage(
  NewEmail.Subject
  + ' just received!')
}
```

Fig. 1: Example TAP rule that computes on sensitive data, connecting an email service with Slack. It has access to ALL email data from a sender, but only needs the subject line, only when it contains the characters 'Confidential'. The 'skip' function call means that the entire rule is aborted. In that case, the rule does not need access to *any* sensitive email data.

Towards my vision of enforcing least privilege on trigger-action platforms, minTAP represents a different point in the design spectrum — one that supports the full range of expressivity of TAP rules, with lower performance overhead than the encrypted TAP approach outlined earlier, but at a lower security guarantee. The TAP still has access to sensitive data in plaintext. The crucial difference is that its access is now least-privilege. In exchange, we have an enforcement mechanism that can protect TAP users immediately because the system architecture does not require any changes to the TAP cloud service.

Current and future directions. The above two projects explore two different ends of the design spectrum. Encrypted TAP offers strongest security at the cost of limited expressiveness and considerable performance overhead. minTAP offers limited privacy guarantees (i.e., only least-privilege data is released in plain text to the TAP) and allows maximum expressiveness with limited performance overhead. Is there a point in the design space that can offer both, strong confidentiality and integrity, and expressiveness with low performance overhead? Based on current trends in *tailored* trusted computing (e.g., Keystone [13] or Komodo [9]), I am interested in understanding what minimal set of trusted hardware primitives would one need to run TAP rules with confidentiality and integrity. While

a technology like Intel SGX can achieve the security goals, it comes at the cost of having a large trusted computing and trusted hardware base because of its inflexible nature and lack of end-user customizability of the silicon. I am currently exploring how minimal trusted hardware extensions on architectures like RISC-V (e.g., physical memory protections) can help in enforcing security properties for TAPs with low performance overhead.

B. Securing AI-Driven Physical Systems

AI has super-charged many areas of computer systems, including those that interact with the physical world. The canonical example is autonomous vehicles. Prior work in the security community has examined the robustness of autonomous through the lens of traditional computer security issues. By contrast, there is a gap in our knowledge about the robustness of autonomous systems when examined through the lens of AI-specific vulnerabilities. It is crucial to understand end-to-end robustness issues (a classic principle in computer security) when AI is part of the pipeline. My research is making the first strides towards understanding these issues, with the overall goal of informing future defenses.

Realistic threat models (CVPR 2018 & 2021, AAAI 2021). The most realistic way to compromise an AI-driven autonomous vehicle is to manipulate its environment. When it senses this corrupted environment, its behavior will change from ideal and the attacker will achieve their goals. My early work showed the first example of how an attacker can throw a few stickers on a Stop sign to trick traffic sign classifiers [8]. However, this opens up several questions, such as: (1) What is the space of physical adversarial examples that are *effective* and *realistic*? (2) What are the end-to-end effects of attacks on machine learning models in a control pipeline?

For the first question on examining the attack space, my goal is to examine the differences between human and machine sensing and understand how attackers can exploit these differences. By establishing these new attack classes, we can begin to think about defenses. As preliminary work, we have recently demonstrated the first *invisible* physical attack that only manipulates the light shining on victim objects rather than the object's underlying texture [27]. This work exploits the fact that humans cannot perceive flickering light above 80 Hz. Fig. 2 shows an example. I contend that these types of attacks are more realistic and practical than current techniques (e.g., attacker flying a drone in front of a victim car or aiming lasers from the side of the road at moving objects with high precision) because the attacker only has to hack a smart LED bulb (e.g., in a tunnel) to achieve their goals.

For the second question on end-to-end effects, my goal is to determine how ML-based attacks affect the larger systems that use ML as part of their processing pipelines. This will help determine what parts of the pipelines are vulnerable and what kinds of defenses need to be in place to avoid the security issues. As preliminary work, we have recently contributed a sequential attack algorithm that can manipulate the behavior of Forward Collision Warning (FCW) systems — a common driver assistance feature available in most cars today. Using techniques from control theory, we demonstrate how an attacker can only compromise vision-related measurements to completely hijack FCW behavior [14]. For example, an attacker can force FCW to show no collision danger when in fact, there is an imminent collision.

With Attack Signal Without Attack Signal

Fig. 2: Example of an invisible physical attack: Images as seen by a human (without border) and as captured by a camera (in black border) with the attack signal (left two images) and without (right two images). The image without the attack signal is classified as coffee mug, while the image with the attack signal is classified as perfume. The attack is robust to camera orientation, distance, and ambient lighting.

Unfortunately, we discovered that attacking an end-to-end system such as FCW requires us to make threat model assumptions that would limit the attacker's ability to target multiple vehicles at scale, because they have to tailor an attack to a specific vehicle (concurrent work in the community has come to a similar conclusion through their own explorations). This drastically limits the applicability of the attack — for example, an attack that is only successful when a number of factors align is not a very widely effective attack and insufficiently informs future defenses. My hope is that the results on practical physical attacks will help us understand more realistic end-to-end attacks.

Current and future directions. Motivated by the above results, I am pushing our understanding of practical physical attacks along two directions: (1) Investigate other types of physical attacks that rely on differences between human

and machine sensing. For example, global shutter cameras are vulnerable to electromagnetic interference — could an attack direct specially-crafted RF signals and manipulate what the camera is seeing? (2) Apply practical physical attacks to end-to-end self-driving pipelines to understand global robustness properties. The second direction will help focus the defense efforts of the community.

REFERENCES

[1] M. M. Ahmadpanah, D. Hedin, M. Balliu, L. E. Olsson, and A. Sabelfeld, "SandTrap: Securing JavaScript-driven Trigger-Action Platforms," in *USENIX Security Symposium*, 2021.

[2] T. Antignac, D. Sands, and G. Schneider, "Data minimisation: A language-based approach," in *SEC*, ser. IFIP Advances in Information and Communication Technology, vol. 502. Springer, 2017, pp. 442–456.

[3] I. Bastys, M. Balliu, and A. Sabelfeld, "If this then what? controlling flows in iot apps," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1102–1119. [Online]. Available: https://doi.org/10.1145/3243734.3243841

[4] M. Bellare, V. T. Hoang, and P. Rogaway, "Foundations of garbled circuits," in *Proceedings of the 2012 ACM conference on Computer and communications security*, 2012, pp. 784–796.

[5] Y. Chen, M. Alhanahnah, A. Sabelfeld, R. Chatterjee, and **E. Fernandes**, "Practical Data Access Minimization in Trigger-Action Systems," in *Proceedings of the 31st USENIX Security Symposium*, 2022.

[6] Y. Chen, A. Chowdhury, R. Wang, A. Sabelfeld, R. Chatterjee, and **E. Fernandes**, "Data Privacy in Trigger-Action Systems," in *Proceedings of the 42nd IEEE Symposium on Security and Privacy (S&P)*, 2021.

[7] M. Conti, B. Crispo, **E. Fernandes**, and Y. Zhauniarovich, "CRePE: A System for Enforcing Fine-Grained Context-Related Policies on Android," *IEEE Transactions on Information Forensics and Security (TIFS)*, 2012.

[8] K. Eykholt, I. Evtimov, **E. Fernandes**, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust Physical-World Attacks on Deep Learning Visual Classification," in *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[9] A. Ferraiuolo, A. Baumann, C. Hawblitzel, and B. Parno, "Komodo: Using verification to disentangle secure-enclave hardware from software," in *Proceedings of the 26th Symposium on Operating Systems Principles*, ser. SOSP '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 287–305. [Online]. Available: https://doi.org/10.1145/3132747.3132782

[10] W. He, M. Golla, R. Padhi, J. Ofek, M. Dürmuth, **E. Fernandes**, and B. Ur, "Rethinking access control and authentication for the home internet of things (iot)," in *27th USENIX Security Symposium (USENIX Security 18)*. Baltimore, MD: USENIX Association, 2018, pp. 255–272. [Online]. Available: https://www.usenix.org/conference/usenixsecurity18/presentation/he

[11] W. He, V. Zhao, O. Morkved, S. Siddiqui, **E. Fernandes**, J. Hester, and B. Ur, "Sok: Context sensing for access control in the adversarial home iot," in *2021 IEEE European Symposium on Security and Privacy (EuroS P)*, 2021, pp. 37–53.

[12] Y. Jia, Q. A. Chen, S. Wang, A. Rahmati, **E. Fernandes**, Z. M. Mao, and A. Prakash, "ContexIoT: Towards Providing Contextual Integrity to Appified IoT Platforms," in *21st Network and Distributed Security Symposium*, 2017.

[13] D. Lee, D. Kohlbrenner, S. Shinde, K. Asanović, and D. Song, "Keystone: An open framework for architecting trusted execution environments," in *Proceedings of the Fifteenth European Conference on Computer Systems*, ser. EuroSys '20. New York, NY, USA: Association for Computing Machinery, 2020. [Online]. Available: https://doi.org/10.1145/3342195.3387532

[14] Y. Ma, J. Sharp, R. Wang, **E. Fernandes**, and X. Zhu, "Sequential Attacks on Kalman Filter-based Forward Collision Warning Systems," in *35th AAAI Conference on Artificial Intelligence (AAAI)*, Feb. 2021.

[15] **E. Fernandes**, A. Aluri, A. Crowell, and A. Prakash, "Decomposable Trust for Android Applications," in *2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2015.

[16] **E. Fernandes**, Q. A. Chen, J. Paupore, G. Essl, J. A. Halderman, Z. M. Mao, and A. Prakash, "Android UI Deception Revisited: Attacks and Defenses," in *Proceedings of the 20th International Conference on Financial Cryptography and Data Security (FC)*, 2016.

[17] **E. Fernandes**, B. Crispo, and M. Conti, "FM 99.9, Radio Virus: Exploiting FM Radio Broadcasts for Malware Deployment," *IEEE Transactions on Information Forensics and Security (TIFS)*, 2013.

[18] **E. Fernandes**, J. Jung, and A. Prakash, "Security Analysis of Emerging Smart Home Applications," in *Proceedings of the 37th IEEE Symposium on Security and Privacy (S&P)*, 2016.

[19] **E. Fernandes**, J. Paupore, A. Rahmati, D. Simionato, M. Conti, and A. Prakash, "FlowFence: Practical Data Protection for Emerging IoT Application Frameworks," in *Proceedings of the 25th USENIX Security Symposium*, 2016.

[20] **E. Fernandes**, A. Rahmati, J. Jung, and A. Prakash, "Decentralized Action Integrity for Trigger-Action IoT Platforms," in *22nd Network and Distributed Security Symposium (NDSS)*, 2018.

[21] **E. Fernandes**, O. Riva, and S. Nath, "My OS Ought to Know Me Better: In-app Behavioural Analytics as an OS Service," in *15th Workshop on Hot Topics in Operating Systems (HotOS XV)*, 2015.

[22] **E. Fernandes**, O. Riva, and S. Nath, "Appstract: On-The-Fly App Content Semantics with Better Privacy," in *Proceedings of the 22nd ACM Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2016.

[23] A. Rahmati, **E. Fernandes**, K. Eykholt, and A. Prakash, "Tyche: A risk-based permission model for smart homes," in *Proceedings of the 3rd IEEE CyberSecurity Development Conference (SecDev)*, 2018.

[24] G. Russello, M. Conti, B. Crispo, and **E. Fernandes**, "MOSES: Supporting Operation Modes on Smartphones," in *Proceedings of the 17th ACM Symposium on Access Control Models and Technologies (SACMAT)*, 2012.

[25] G. Russello, B. Crispo, **E. Fernandes**, and Y. Zhauniarovich, "YAASE: Yet Another Android Security Extension," in *3rd IEEE Conference on Privacy, Security, Risk and Trust (PASSAT)*, 2011.

[26] J. Saltzer and M. Schroeder, "The protection of information in computer systems," *Proceedings of the IEEE*, vol. 63, no. 9, pp. 1278–1308, 1975.

[27] A. Sayles, A. Hooda, M. Gupta, R. Chatterjee, and **E. Fernandes**, "Invisible perturbations: Physical adversarial examples exploiting the rolling shutter effect," in *In proceedings of CVPR 2021*.

[28] Y. Zhauniarovich, G. Russello, M. Conti, B. Crispo, and **E. Fernandes**, "MOSES: Supporting and Enforcing Security Profiles on Smartphones," *IEEE Transactions on Dependable and Secure Computing (TDSC)*, 2014.