Subject: Product Security and the Right to Repair February 11, 2019

To whom it may concern-

My name is Joe Grand and I'm the President of Grand Idea Studio, Inc. Throughout my career, I've straddled the fence between hacker and engineer. For nearly 15 years, I've been teaching individuals and organizations of all types about the hardware hacking process, mindset, and techniques to defeat security in order for them to understand how to better protect themselves from adversaries.[1] Formerly known as Kingpin, I was a member of the legendary hacker collective L0pht Heavy Industries[2], which testified in front of the United States Senate Committee on Governmental Affairs regarding the state of computer security in government and helped raise awareness of the benefits of security vulnerability research and disclosure. As a computer engineer with experience in designing and manufacturing hardware products, I'm able to understand the common mindset and constraints that current product manufacturers face.

It's a principle of cybersecurity that nothing is ever 100% secure. The best we can do is innovate and improve security of our systems while understanding that it's an on-going "cat and mouse" game between designer and adversary. When implementing security to modern day best practices, having physical access to a device should not weaken security in most situations, *particularly during the ordinary business of repair*. Devices with well-planned security initiatives will isolate components that are critical to security within a physically protected and access-controlled area. For example, Apple's Secure Enclave[3] and T2 Security Chip[4] provide security-related services for their devices (hardware root of trust, secure boot, and data encryption). ARM and Intel both provide trusted execution environments within their CPU architectures to isolate

[1] http://www.grandideastudio.com/portfolio/hardware-hacking-training

[2] https://en.wikipedia.org/wiki/L0pht

[3] https://www.computerhope.com/jargon/s/secure-enclave.htm

[4] https://www.apple.com/mac/docs/Apple_T2_Security_Chip_Overview.pdf

critical security functionality.[5] [6] By using such security elements, there's no reason to restrict access to non-critical and non-security-related components of the system (such as the battery, screen, or camera, which make up the majority of elements that need repair on mobile phones). The claims of security as a reason against Right to Repair are based more on *profit margins and maintaining control of the product's entire lifecycle* than real concerns of security.[7]

Depending on the design of the product, it does remain possible for hardware components to be modified or implanted in order to bypass security or steal user data – these actions have to be introduced deliberately by either the manufacturer or a determined intermediary with a premeditated plan and goal of attack.[8] The time and effort to create and achieve such an attack, particularly against a targeted business or individual, could take weeks, months, or years. *None of this is part of the ordinary business of repair.* This particular threat remains in place regardless of laws limiting end users from repairing their own devices. However, without the availability of and access to OEM original parts and documentation, the risk of such a compromise increases.[9] Those that repair devices may be innocent, unwitting parties in a malicious attack by being forced to obtain components from unverifiable sources and of questionable quality. This is why Right to Repair is so important - manufacturers who support Right to Repair will *help to improve*, not weaken, security, at the same time making it easier to extend the lifespan of damaged and/or obsolete devices.

In the past decade, the acceptance and acknowledgement of security vulnerabilities in software has led to bug bounty programs[10] [11], timely fixes and remediation by vendors, and arguably has led to more secure software. The hardware world lags far behind, with many vendors claiming that closing products and preventing physical access for end-user repair will solve security problems. This is simply not true. The majority of hardware-based security vulnerabilities have been due to poor overall product design, easy and unprotected access to critical test and debug interfaces, and the existence of default passwords, misconfigurations, or backdoors.

---

[5] https://www.arm.com/products/silicon-ip-security

[6] https://software.intel.com/en-us/sgx

[7] https://motherboard.vice.com/en_us/article/zmd9a5/tim-cook-to-investors-people-bought-fewer-new-iphones-because-they-repaired-their-old-ones

[8] https://en.wikipedia.org/wiki/NSA_ANT_catalog

[9] https://www.usenix.org/conference/woot17/workshop-program/presentation/shwartz

[10] https://hackerone.com/bug-bounty-programs

[11] https://www.bugcrowd.com/bug-bounty-list

Right to Repair legislation will be extremely helpful to security researchers like me. The US Copyright Office continues to grant exemptions to the heavy-handed Digital Millennium Copyright Act for security research, but that's not enough. We need more consistent, legal access to hardware and firmware in order to test for security vulnerabilities without fear of repercussion. We need to pay more attention to the extremely poor security of Internet of Things devices being shipped by the millions, placing responsibility on vendors to impart better security hygiene. We need more vendors to embrace and provide support to the cybersecurity community in order to enable more comprehensive security research. Furthermore, if end users are given the ability to acquire OEM original parts, diagnostic tools, firmware, and documentation, the need to use counterfeit parts for repair will dramatically decrease.

Repairing products or sharing information on how to do so *should not be a crime*. Rather, putting the public at risk by preventing authorized repair and controlling how end users are allowed to use a device they purchased should be.

Sincerely,

Joe Grand
Grand Idea Studio, Inc.
Portland, OR
http://www.grandideastudio.com