Encouraging Responsible Disclosure of Software Vulnerabilities

Robert S. Prostko

Critical Challenge Project

Final

Executive Master in Cybersecurity

Brown University

February 2019

ENCOURAGING RESPONSIBLE DISCLOSURE OF SOFTWARE VULNERABILITIES

ROBERT S. PROSTKO

I. INTRODUCTION

Vulnerabilities in software are commonplace. They are routinely exploited by bad actors to cause harm and for financial gain. As software is increasingly embedded in daily life, from refrigerators to implanted medical devices to transportation to power management systems, it becomes ever more important to address the problem of rampant software vulnerabilities.

One way to improve the situation is wider use of programs for responsible disclosure of software vulnerabilities. Responsible disclosure programs are established by organizations[1] to provide a mechanism for security researchers to disclose vulnerabilities to the entity that developed the product or service. When disclosing responsibly, the researcher typically reveals the vulnerability to the company, but otherwise keeps the vulnerability confidential for a period of time, so that the organization can develop and deploy a software fix for the vulnerability before it becomes publicly known.

If the security researcher were to disclose the vulnerability publicly without disclosing it first to the organization, it would be a so-called "zero-day" vulnerability because it was a previously unknown vulnerability with no published fix for the software. Malicious actors could use the vulnerability to attack the software because the company has not had a chance to develop and release a fix or patch.

A wide range of groups and organizations support the proper disclosure of software vulnerabilities. For example, the Department of Justice ("DOJ") issued *A Framework for a*

[1] This could be for-profit and non-profit companies, government(s), among others.

Vulnerability Disclosure Program for Online Systems.[2] The DOJ encouraged organizations to adopt a vulnerability disclosure program by providing this comprehensive framework for organizations to consider when instituting a responsible disclosure or bug bounty program.[3] Furthermore, non-profit groups, such as the Electronic Frontier Foundation ("EFF")[4], also advocate for the greater the disclosure of software vulnerabilities. But even with broad-based support, more and better responsible disclosure of software vulnerabilities is chilled by uncertainties about existing laws and other impediments.

It is desirable, therefore, to reduce barriers to responsible disclosure and to increase incentives. The following sections will elucidate the problem of software vulnerabilities and their important ramifications, describe how disclosure currently functions, and examine impediments to disclosure. Building upon that base, a series of recommendations will demonstrate how to incentivize more and better responsible disclosure of software vulnerabilities.

II. PROBLEM: VULNERABLE SOFTWARE

The development and use of software has had tremendous impact throughout the economy and society. Its power lies in being able to use a computer to perform repetitive tasks that were previously done by a person or to compute complex calculations that were not capable of being calculated at all.

---

[2] Department of Justice, *A Framework for a Vulnerability Disclosure Program for Online Systems*, July 2017, https://www.justice.gov/criminal-ccips/page/file/983996/download (last accessed January 17, 2019).

[3] "A proactive extension to a Vulnerability Disclosure Program (VDP), where a cash incentive is added to reward the first white hat hacker to find and report each unique vulnerability within the scope of the program."
BugCrowd, *Bug Bounty Program*, https://www.bugcrowd.com/resources/glossary/bug-bounty-program-bbp/ (last accessed January 25, 2019).

[4] Electronic Frontier Foundation, https://www.eff.org (last accessed January 25, 2019).

A. Software is Buggy

Software, like any technology, has its shortcomings. One of them is the presence of vulnerabilities, or "bugs," in the software. Traditionally, software was written by programmers in different programming languages, which have varying levels of security. Programmers designed the software based on the requirements that they were given. Typically, product managers did not enumerate security requirements along with the product feature requirements. Thus, any security in the software had to be designed and implemented solely by the programmers on the project. Secure programming, historically, has not been a prominent part of computer engineering and computer science programs until more recently. Therefore, much of the software that exists was not written or designed with security in mind.

Even when programmers implemented security into software, such as by using cryptography, those cryptographic protocols tended to become obsolete over time, as processing power increased and new discoveries in cryptography were made. Thus, sometimes software that was once secure became no longer secure because, for example, the encryption protocol became obsolete.

Vulnerabilities are often found after the software has been released and in use. The key issue is not whether software has bugs or vulnerabilities because it often does. Indeed, people are accustomed to seeing software updates and patches for software on many devices. The key issue is how those vulnerabilities are addressed and fixed.

B. Proliferation of Vulnerable Software Exacerbates the Problem

The scalability of software and its ability to automate nearly every type of task has led to the proliferation of software in products and services. The number of purely mechanical products

is dwindling. Thus, the vulnerabilities attendant to software are now being introduced in the devices that historically were purely mechanical or in categories that did exist in the past. Consumers, companies, and government officials are all having to figure out how to best tackle this problem of proliferating software vulnerabilities in more and more products and services.

Today, during the development of products and services, programmers rarely write whole programs from scratch and more frequently stitch together pieces of open source software, writing only the unique pieces for a given application. Accordingly, software vulnerabilities in open source software are cascaded to all products incorporating it, which in some cases is incredibly widespread across the Internet.[5]

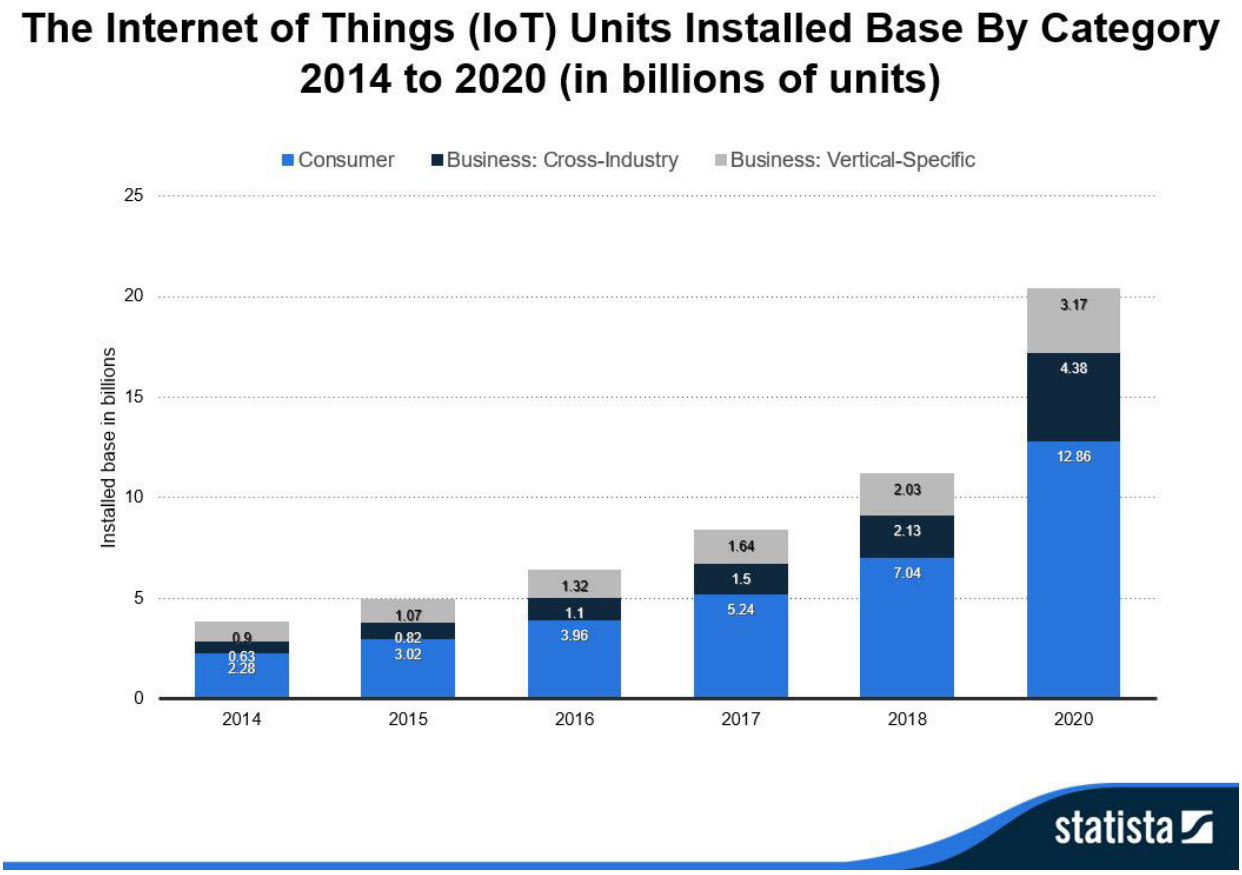
C. The Internet of Things Amplifies the Problem

The Internet of Things ("IoT") "encompasses everything connected to the internet, including objects that 'talk' to each other."[6] IoT devices are generally devices that include software, which enables them to connect to the Internet, to communicate with other devices, and to share data. They are also often referred to as "smart" devices. Lots of different smart devices

[5] The Heartbleed vulnerability affected the OpenSSL library that is open source software. OpenSSL is widely used by devices to encrypt communications on the Internet, which is one of, if not the most, common encryption protocol on the Internet. Only one line of code caused the vulnerability and existed for two years before it was publicly disclosed. Up to 17% of all devices on the Internet were affected by the bug, which is an incredibly large number of devices given the size of the Internet. *See*, Josh Fruhlinger, *What is the Heartbleed bug, how does it work and how was it fixed?*, https://www.csoonline.com/article/3223203/vulnerabilities/what-is-the-heartbleed-bug-how-does-it-work-and-how-was-it-fixed.html (last accessed January 16, 2019).

[6] Matt Burgess, *What is the Internet of Things? WIRED Explains*, February 16, 2018, https://www.wired.co.uk/article/internet-of-things-what-is-explained-iot (last accessed January 16, 2019); Jacob Morgan, *A Simple Explanation of 'The Internet of Things,'* May 13, 2014, https://www.forbes.com/sites/jacobmorgan/2014/05/13/simple-explanation-internet-things-that-anyone-can-understand/#5c37e85a1d09 (last accessed January 16, 2019) (The Internet of Things "is the concept of basically connecting any device with an on and off switch to the Internet (and/or to each other).").

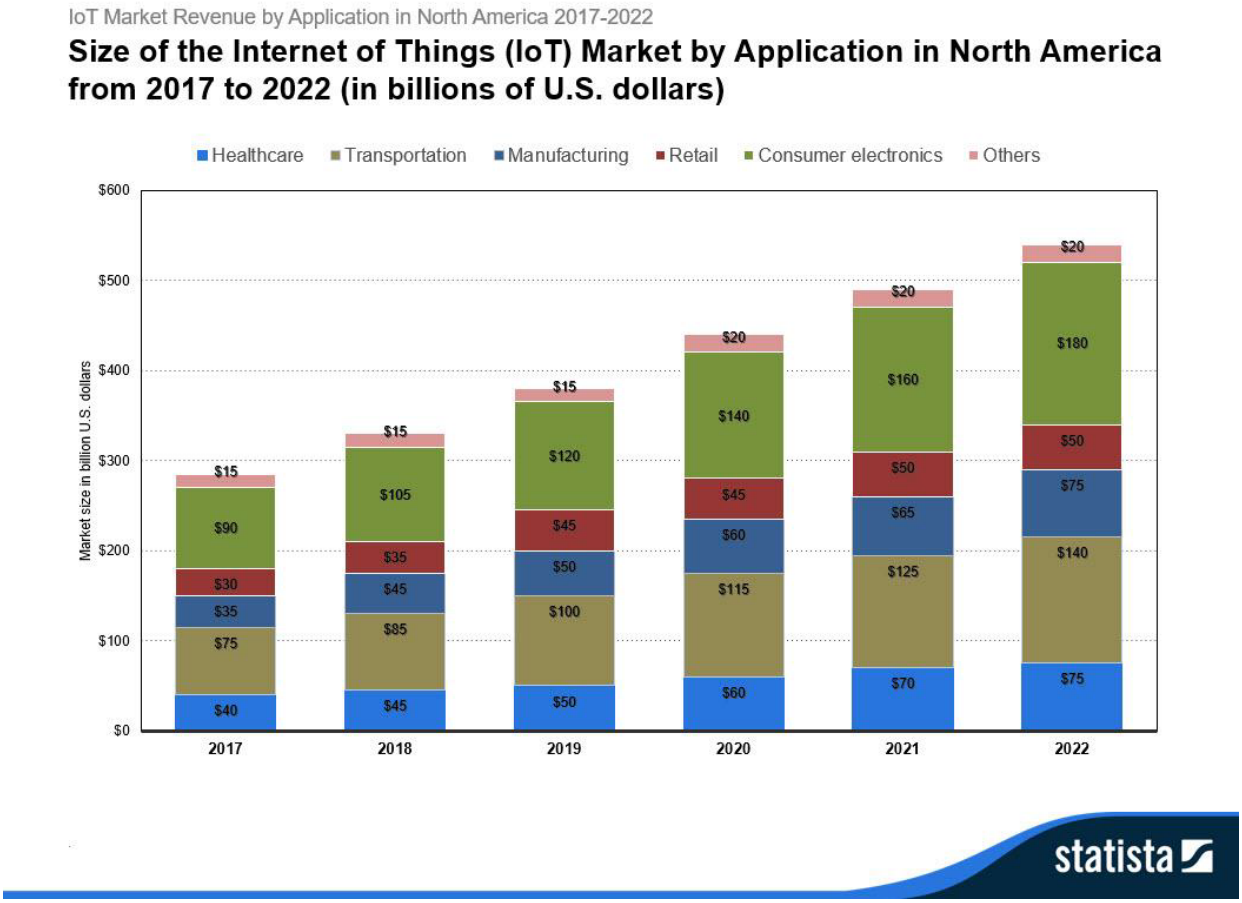
fall into this broad category, including lights, refrigerators, door locks, baby monitors, routers, mobile phones, laptops, tablet computers, televisions, etc. There are already billions of IoT devices connected to the Internet, with projected explosive growth, such that the number of installed IoT devices will more than quadruple between 2014 and 2020:[7]



Source: Forbes, *10 Charts That Will Challenge Your Perspective of IoT's Growth*.[8]

[7] Louis Columbus, *10 Charts That Will Challenge Your Perspective of IoT's Growth*, June 6, 2018, https://www.forbes.com/sites/louiscolumbus/2018/06/06/10-charts-that-will-challenge-your-perspective-of-iots-growth/#36353c2a3ecc (last accessed January 16, 2019).
[8] *Ibid*.

The pervasiveness of IoT adoption is furthered underscored by the chart below, which illustrates projected IoT adoption by industry sector.[9]



Source: Forbes, *10 Charts That Will Challenge Your Perspective of IoT's Growth*.[10]

IoT devices will become ubiquitous throughout the economy, society, and daily life. IoT devices are often battery-powered and resource-constrained, with the result that their ability to be updated to newer versions of software that add security features will likely be limited. As a consequence, many IoT devices will retain vulnerabilities that will never be addressed.

[9] *Ibid.*
[10] *Ibid.*

Government officials recognize the security problem created by IoT. In the absence of federal laws or regulations, California enacted a law in 2018 focused on the security of IoT devices.[11] This is the first law of its kind to be enacted in the United States at the state or federal level. The California IoT law requires manufacturers of connected devices to implement reasonable security features that are "(1) Appropriate to the nature and function of the device. (2) Appropriate to the information it may collect, contain, or transmit. (3) Designed to protect the device and any information contained therein from unauthorized access, destruction, use, modification, or disclosure."[12] Examples of reasonable security features include "(1) The preprogrammed password is unique to each device manufactured. (2) The device contains a security feature that requires a user to generate a new means of authentication before access is granted to the device for the first time."[13]

Although it is encouraging that California's legislature was concerned about the security of IoT devices, the better outcome would be a federal law, which would apply uniformly across the United States, rather than a patchwork of state laws being adopted, with possibly different and conflicting requirements. The California law is a good development, however, even though it is narrow, because it provided an example, set a baseline of security requirements for IoT devices, and will stimulate additional legislative activity in the United States.

With the widespread use of IoT devices, vulnerabilities in these products need to be identified early and often. Moreover, barriers need to be removed and researchers need to be

[11] Cal Civ. Code § 1798.91.04 ("California IoT Law").
[12] *Ibid.*
[13] *Ibid.*

incentivized to identify and report vulnerabilities, so that there is not a web of connected devices with vulnerabilities that are never fixed.

D. IoT is in Critical Infrastructure

One of the most troubling aspects of the current proliferation of insecure IoT devices is their increasing use in critical infrastructure in the United States. Critical infrastructure consists of the sectors so important to the United States that, if attacked, it would threaten not only lives, but the viability of the country.[14] The United States government designated the following 16 sectors and agencies as critical infrastructure: "chemical; commercial facilities; communications; critical manufacturing; dams; defense industrial base; emergency services; energy; financial services; food and agriculture; government facilities; healthcare and public health; information technology; nuclear reactors, materials, and waste; transportation systems; and water and wastewater systems."[15]

Federal laws provide special requirements and protections for critical infrastructure. For example, the Department of Homeland Security ("DHS") is tasked with identifying critical infrastructure and prioritizing resources for improving the security and resiliency of the critical infrastructure.[16] DHS also models and simulates attacks on critical infrastructure, so that risks may

[14] The government of the United States defines critical infrastructure as the "systems and assets, whether physical or virtual, so vital to the United States that the incapacity or destruction of such systems and assets would have a debilitating impact on security, national economic security, national public health or safety, or any combination of those matters." 42 U.S. Code § 5195c(e).
[15] The White House, *Presidential Policy Directive—Critical Infrastructure Security and Resilience*, February 12, 2013, https://obamawhitehouse.archives.gov/the-press-office/2013/02/12/presidential-policy-directive-critical-infrastructure-security-and-resil (last accessed January 14, 2019).
[16] *Ibid.*; *see also,* Department of Homeland Security, Infrastructure Security, https://www.dhs.gov/topic/critical-infrastructure-security (last accessed January 14, 2019).

be mitigated.[17] Another responsibility assigned to DHS is to "[c]oordinate Federal Government responses to significant cyber or physical incidents affecting critical infrastructure."[18]

Although there are already IoT devices in US critical infrastructure, Executive Order 13636 excludes "commercial information technology products or consumer information technology services under this section."[19] Some speculated that this exception applies to companies like Google, Facebook, Microsoft, and Twitter.[20] While the impact is unknown because the list of critical infrastructure devices and systems is confidential, there is some risk that, due to this order, the government and private sector will not identify and mitigate vulnerabilities in commercial and consumer IoT devices that are in critical infrastructure.

III. DISSEMINATING INFORMATION ABOUT VULNERABILITIES IS CURRENTLY BOTH A PROBLEM AND A SOLUTION

Security researchers are individuals who study the security of information and communication systems. Some researchers test software for vulnerabilities. They may be independent or work for organizations. Over time, several categories of security researchers have emerged: black-hat hackers, grey-hat hackers, and white-hat hackers.

[17] 42 U.S.C. § 5195c(e).
[18] The White House, *Presidential Policy Directive—Critical Infrastructure Security and Resilience*, February 12, 2013, https://obamawhitehouse.archives.gov/the-press-office/2013/02/12/presidential-policy-directive-critical-infrastructure-security-and-resil (last accessed January 14, 2019).
[19] Executive Order 13636 – Improving Critical Infrastructure Cybersecurity, Section 9, February 12, 2013.
[20] Rebecca S. Eisner, Howard W. Waltzman, and Lei Shen, *The 2013 Cybersecurity Executive Order: Potential Impacts on the Private Sector*, Summer 2013, https://www.mayerbrown.com/the-2013-cybersecurity-executive-order-potential-impacts-on-the-private-sector-08-13-2013/ (last accessed January 16, 2019).

A. The Role of Security Researchers

Security researchers play an important role in the software development lifecycle. Even though many companies have internal programs to find vulnerabilities before releasing software, and maybe even have third parties perform the vulnerability assessments, vulnerabilities will inevitably still find their way into software products and services. This can happen for a number of reasons. For example, internal security assessment teams are familiar with the product, so they may be conditioned to attack the product in certain ways, while an independent researcher may approach it differently. Also, from time to time, vulnerabilities are identified in open standards and protocols that are widely used or industry standard, such as the BlueBorne[21] and Krack[22] vulnerabilities identified in Bluetooth and Wi-Fi, respectively, in 2017.

B. Harmful Dissemination

Black-hat hackers are individuals who find software vulnerabilities and exploit them to cause harm or to sell them for financial gain to others in illegal black markets on the Dark Web. [23] Black-hat hackers may be able to sell the software vulnerabilities for up to six or seven figures, depending on the type of software and platform vulnerability. For example, so-called "zero-day" vulnerabilities to jailbreak an Apple iPhone or Android phone, which then enable unapproved software to be installed, are often in that price range. They do not disclose or sell the vulnerability

---

[21] *The Attack Vector "BlueBorne" Exposes Almost Every Connected Device*, https://armis.com/blueborne/ (last accessed January 16, 2019).

[22] Mathy Vanhoef, *Key Reinstallation Attacks: Breaking WPA2 by Forcing Nonce Reuse*, 2017, https://www.krackattacks.com/ (last accessed January 16, 2019).

[23] "The dark web is part of the internet that isn't visible to search engines and requires the use of an anonymizing browser called Tor to be accessed." Darren Guccione, *What is the dark web? How to access it and what you'll find*, CSO, January 11, 2019, https://www.csoonline.com/article/3249765/data-breach/what-is-the-dark-web-how-to-access-it-and-what-youll-find.html (last accessed January 25, 2019).

to the company or entity that developed the software. The activities of black-hat hackers are typically illegal.

B. USEFUL DISSEMINATION

White-hat hackers are individuals who find software vulnerabilities and then disclose the vulnerabilities to the company or entity that developed the software, so that a fix or patch can created and deployed. White-hat hackers are genuinely looking for vulnerabilities in existing software so that they can be remediated, not exploited by others. Yet, white-hat security researchers may be violating criminal laws by engaging in vulnerability discovery. Many, if not all, of the applicable laws, such as the federal Computer Fraud and Abuse Act ("CFAA"), 18 U.S.C. § 1030, and California's Unauthorized Computer Access Law, contain no express exceptions to protect white-hat security researchers.[24] As a result, white-hat security researchers struggle with how ethically to disclose vulnerabilities to companies or the government, without subjecting themselves to criminal prosecution or civil liability.

Grey-hat hackers are somewhere in between black-hat hackers and white-hat hackers. For example, a grey-hat hacker may find a software vulnerability but may demand a ransom payment from the developer before showing the developer the vulnerability.

IV. IMPEDIMENTS TO INCENTIVIZING RESPONSIBLE DISCLOSURE OF VULNERABILITIES

There are many complex and diverse reasons why software vulnerabilities are not disclosed better. Most security researchers want to do the right thing by disclosing the vulnerability to the company or the government before publishing the results publicly, but ambiguity in laws, coupled with prosecutorial discretion, places security researchers in a difficult position. This uncertainty

[24] 18 U.S.C. § 1030; California Penal Code 502(c).

discourages security researchers from disclosing the vulnerabilities to companies or the government for fear of criminal prosecution. Furthermore, complexities in implementing a vulnerability disclosure program have slowed the adoption of such programs by some companies.

A. Current Laws Create Uncertainty About the Legality of Security Research

The main law that generates concern in the security researcher world is the Computer Fraud and Abuse Act ("CFAA").[25] The CFAA was enacted in the 1980s in response to the new threat that connected computers could have on critical infrastructure and other sensitive operations in the United States.[26]

The CFAA prohibits, among other things, accessing a protected computer without authorization. In particular, "The CFAA was originally designed as a criminal statute aimed at deterring and punishing hackers, particularly those who attack computers used for compelling federal interests, but it also includes a trespass-like civil remedy under federal law."[27] Furthermore, the CFAA provides for the possibility of long prison sentences for violations of the statute.

The Electronic Frontier Foundation succinctly explained the security research issues created by the CFAA:

> The CFAA makes it illegal to engage in 'unauthorized access' to a computer connected to the Internet, but the statute doesn't tells [sic] us what 'authorization' or 'without authorization' means. This vague language might have seemed innocuous to some back in 1986 when the statute was passed, but in today's networked world, where we all regularly connect to and use computers owned by others,

[25] 18 U.S.C. § 1030.

[26] Sara Sun Beale and Peter Berris, *Hacking the Internet of Things: Vulnerabilities, Dangers, And Legal Responses*, DIGITALIZATION AND THE LAW (2018). Moreover, "there is some evidence that Congress passed the CFAA in response to the movie War Games, where the protagonist accidentally hacks into the computer controlling America's nuclear weaponry and nearly starts a third world war." *Ibid* at 169 (internal quotations omitted).

[27] *Ibid* at 169 (internal quotations omitted).

> courts cannot even agree on what the law covers. And as a result, this pre-Web law is causing serious problems.[28]

Importantly, there is no express provision in the CFAA that protects from criminal liability a security researcher who discloses the vulnerability to the company or the government before disclosing it publicly. Thus, even a white-hat hacker could be prosecuted under the CFAA.

The catastrophic 2017 Equifax breach provides a troubling example. Incredibly, "a security researcher warned Equifax that it was vulnerable to the kind of attack that later compromised the personal data of more than 145 million Americans."[29] However, Equifax did not patch the vulnerability immediately and only patched it six months later, after the massive data breach was publicly known.[30] The EFF argues that "[t]he security researcher who discovered the vulnerability in Equifax's system back in 2016 should have been empowered to bring their findings to someone else's attention after Equifax ignored them. If they had, the breach may have been avoided."[31] But the security researcher "faced the risk of a CFAA lawsuit and potentially decades in federal prison," and therefore, did not publicly disclose the vulnerability to anyone outside of Equifax.[32]

The CFAA is not the only law engendering legal uncertainty. States, such as California, have laws like the California Unauthorized Computer Access Law, which prohibits "[k]nowingly

---

[28] Jamie Williams, Electronic Frontier Foundation, *The Worst Law in Technology Strikes Again: 2017 in Review*, December 29, 2017, https://www.eff.org/deeplinks/2017/12/worst-law-technology-strikes-again-2017-review (last accessed January 17, 2019).
[29] Lorenzo Franceschi-Bicchierai, *Equifax was Warned*, October 26, 2017, https://motherboard.vice.com/en_us/article/ne3bv7/equifax-breach-social-security-numbers-researcher-warning (last accessed January 16, 2019).
[30] *Ibid.*
[31] Jamie Williams, Electronic Frontier Foundation, *The Worst Law in Technology Strikes Again: 2017 in Review*, December 29, 2017, https://www.eff.org/deeplinks/2017/12/worst-law-technology-strikes-again-2017-review (last accessed January 17, 2019).
[32] *Ibid.*

and without permission accesses or causes to be accessed any computer, computer system, or computer network."[33] This law is similar to the CFAA in some respects and may cause a chilling effect because the California law does not include an exception for legitimate security research.

B. Provide More Guidance For Prosecutors and Security Researchers

As discussed earlier, the DOJ developed recommendations to encourage organizations to institute vulnerability disclosure programs. The DOJ, however, has not provided guidance for prosecutors, so that there will be more notice and information to them, as well as to security researchers, about the acceptable parameters for vulnerability research, in order to encourage such research and disclosure and to offer reassurance and greater clarity about the scope of permissible activities and possible criminal exposure.

The DOJ document states, "the framework outlines a process for designing a vulnerability disclosure program that will clearly describe authorized vulnerability disclosure and discovery conduct, thereby substantially reducing the likelihood that such described activities will result in a civil or criminal violation of law under the Computer Fraud and Abuse Act (18 U.S.C. § 1030)."[34] The DOJ is signaling to security researchers that, if the researchers follow the discovery and disclosure process in an organization's program, the DOJ will very likely not prosecute the researchers. However, what if a researcher discovers a vulnerability in software developed by an organization that does not have a disclosure program? The researcher is left in the unenviable position of choosing to disclose and risk prosecution or choosing not to disclose, thereby allowing

[33] California Penal Code 502(c).
[34] Department of Justice, *A Framework for a Vulnerability Disclosure Program for Online Systems*, July 2017, pgs. 1-2, https://www.justice.gov/criminal-ccips/page/file/983996/download (last accessed January 17, 2019).

the vulnerability to persist. Should researchers only analyze and report vulnerabilities to organizations with disclosure programs?

Earlier, in 2014, the DOJ issued prosecution guidance on the CFAA, but this guidance did not provide any express exception from prosecution if the conduct was related to security research.[35] The earlier guidance listed various factors for prosecutors to consider, but none provided any reassurances to security researchers.[36]

Indeed, the scope of potential criminal liability under the CFAA is still unclear and depends in part on whether an organization implemented a disclosure program in accordance with the DOJ's framework. Thus, it is unclear what actions by a security researcher are acceptable to avoid liability and prosecution under the CFAA. This is because federal prosecutors retain prosecutorial discretion on who to prosecute. Greater clarity is needed to remove liability concerns for researchers, so software vulnerability discovery and disclosure may occur.

C. Organizations Do Not Always Want to Learn About Software Vulnerabilities

Some organizations prefer not to learn about software vulnerabilities in their products, which creates yet another impediment. If an organization does not want vulnerabilities reported, security researchers are put in a difficult position because there is a higher likelihood that the organization will have a negative reaction to a disclosure and contact law enforcement authorities to press for prosecution.

---

[35] *DOJ CFAA Charging Memo (2014)*, https://www.eff.org/document/doj-cfaa-charging-memo-2014; Jamie Williams, *New Federal Guidelines For Computer Crime Law Do Nothing To Reign In Prosecutorial Overreach Under Notoriously Vague Statute*, October 31, 2016, https://www.eff.org/deeplinks/2016/10/what-were-scared-about-halloween-prosecutorial-discretion-under-notoriously-vague (last accessed January 17, 2019).

[36] *Ibid.*

Some of the reasons for organizations not welcoming vulnerability disclosures are legitimate. Patching software vulnerabilities in IoT devices can be difficult to impossible for several reasons.[37] First, many of these IoT devices are designed to be in service for a decade or more.[38] An example may be traffic lights or other smart city IoT devices.[39] No can anticipate all the types of attacks or vulnerabilities that will be present in three years, let alone a decade from now.[40]

Another reason that IoT devices are difficult to patch is that IoT devices are typically special purpose electronic devices with limited resources.[41] These IoT devices may not be able to be patched because the devices do not have enough processing power or memory to execute or store the new code. Some IoT devices are not connected to the Internet, further complicating patching because a person typically must visit each IoT device to update them individually, a costly endeavor.

Other legitimate concerns about patching IoT devices with software vulnerabilities include unnecessarily drawing the attention of hackers (whether benign or malicious) where no one was looking before. Another concern is whether the company will have enough engineering resources to handle an increase in software vulnerability reports. A related issue is the difficulty in

---

[37] Sandip Ray, Abhishek Basak and Swarup Bhunia, IEEE, *To Secure the Internet of Things, We Must Build It Out of "Patchable" Hardware*, https://spectrum.ieee.org/telecom/security/to-secure-the-internet-of-things-we-must-build-it-out-of-patchable-hardware (last accessed January 16, 2019).

[38] *Ibid.*

[39] *Ibid.*

[40] *Ibid.;* The Heartbleed vulnerability in the widely-used encryption protocol used in Internet communications for two years before being disclosed. Another example are the Spectre and Meltdown vulnerabilities in microprocessors. https://meltdownattack.com/ (last accessed January 17, 2019).

[41] *Ibid.*

prioritizing fixing software vulnerabilities without delaying new product development. Not all vulnerabilities are critical, drop-everything issues. Notably, the Department of Justice's Vulnerability Disclosure Framework is silent on these issues of resources and prioritization.[42]

Unfortunately, these factors lead some companies to conclude that the company would be better off not knowing about vulnerabilities in their products, rather than encouraging researchers to report vulnerabilities.

V. SOLUTIONS TO CREATE AN ENVIRONMENT OF RESPONSIBLE DISCLOSURE

Currently, there is a perfect storm of unclear laws and a viable, even if illegal, market to sell software vulnerabilities. The following proposals identify solutions to provide greater legal certainty to security researchers, in order to encourage more and higher quality software vulnerabilities disclosures.

A. GOVERNMENT INTERVENTION TO ENCOURAGE RESPONSIBLE DISCLOSURE

The government can use several different mechanisms to encourage greater responsible disclosure of software vulnerabilities.

1. Modify the CFAA

The most significant obstacle impeding greater vulnerability research and disclosure is the CFAA. Accordingly, the CFAA should be modified to include an express exception for white-hat security researchers who responsibly disclose. This amendment would clarify that conducting security vulnerability research, including discovery and disclosure, would not result in legal liability, civil or criminal, under the CFAA. The amendment should include some minimum

[42] Department of Justice, *A Framework for a Vulnerability Disclosure Program for Online Systems*, https://www.justice.gov/criminal-ccips/page/file/983996/download (last accessed January 17, 2019).

actions that the researcher should perform in properly disclosing the vulnerability to the organization, thereby being free from liability, such as clearly showing the organization the vulnerability and providing the organization an opportunity to address the vulnerability (e.g., 90 days) before publishing it publicly.

Calls for an amendment to the CFAA are not new but are more urgent than ever because of the proliferation of software and IoT devices. Security researchers need a safe, reliable way of disclosing software vulnerabilities that they discover. The EFF has been calling for reforms to the CFAA for years.[43] Various bills have been introduced in Congress that propose amending the CFAA with respect to security research, but none have been enacted.

One interesting example is the bill introduced by Senator Mark Warner (D-VA) in 2017, the "Internet of Things (IoT) Cybersecurity Improvement Act of 2017."[44] The proposed law would have established certain security requirements for electronic devices purchased and used by the federal government.[45] Moreover, the bill is a good example of how the federal government could use its purse strings to establish a baseline of security requirements. The bill also included an amendment to the CFAA, so that white-hat security researchers who followed certain procedures to disclose software vulnerabilities responsibly would not face criminal prosecution. A law like Senator Warner's, if enacted, would likely act as the *de facto* security standard for IoT products, whether purchased by the United States government or not.

---

[43] *See, e.g.*, Trevor Timm, *Prominent Security Researchers, Academics, and Lawyers Demand Congress Reform the CFAA and Support Aaron's Law*, August 2, 2013, https://www.eff.org/deeplinks/2013/08/letter (last accessed January 17, 2019).
[44] S.1691, Internet of Things (IoT) Cybersecurity Improvement Act of 2017.
[45] *Ibid.*

2. Prosecutorial Guidance

The DOJ has provided guidance to federal prosecutors on when to prosecute someone under the CFAA. However, the guidance does not provide any clear exception from liability under the CFAA for security research.[46] Instead, the EFF notes that the "federal guidelines for prosecutions under the Computer Fraud and Abuse Act (CFAA), [which are] intended to assist prosecutors in deciding when to bring charges under the notoriously vague federal statute targeting computer break-ins, demonstrate that prosecutors have far too much discretion in applying the CFAA."[47] Thus, further guidance is needed from the DOJ regarding the CFAA to encourage security researchers to discover and report vulnerabilities they find.

3. Bully Pulpit

The government can use its unique position to persuade and encourage organizations to adopt responsible disclosure programs. For example, the Federal Trade Commission ("FTC") is in a position to have a significant influence on private enterprises. Indeed, commentators have noted that the FTC and DOJ "are signaling that in the future organizations must have some form of vulnerability disclosure program (VDP) that lets good-faith security researchers report bugs."[48] For example, the FTC stated in public comments to the Consumer Product Safety Commission that "in many cases, the FTC has alleged, among other things, that the failure to maintain an

[46] *DOJ CFAA Charging Memo (2014)*, https://www.eff.org/document/doj-cfaa-charging-memo-2014 (last accessed January 17, 2019).
[47] Jamie Williams, *New Federal Guidelines For Computer Crime Law Do Nothing To Reign In Prosecutorial Overreach Under Notoriously Vague Statute*, October 31, 2016, https://www.eff.org/deeplinks/2016/10/what-were-scared-about-halloween-prosecutorial-discretion-under-notoriously-vague (last accessed January 17, 2019).
[48] J.M. Porup, *Do You Need A Vulnerability Disclosure Program? The Feds Say Yes*, August 7, 2018, https://www.csoonline.com/article/3294418/vulnerabilities/do-you-need-a-vulnerability-disclosure-program-the-feds-say-yes.html (last accessed January 17, 2019).

adequate process for receiving and addressing security vulnerability reports from security researchers and academics is an unreasonable practice, in violation of Section 5 of the FTC Act."[49] It would be powerful if the FTC and DOJ coordinated on the issue of responsible disclosure to publish clear guidance for security researchers to follow to properly disclose and not be subject to liability under the CFAA.

B. COMPANIES' ROLE

Companies can encourage and incentivize responsible disclosure of software vulnerabilities in several different ways.

1. Responsible Disclosure and Bug Bounty Programs

Companies have begun implementing responsible disclosure programs, which typically include some type of web form or email address for the security researcher to disclose the vulnerability. Some companies have developed bug bounty programs, which are essentially responsible disclosure programs, with the addition that the company pays a fee to the security researcher in exchange for responsibly disclosing the vulnerability. The amount of money paid for the vulnerability is something that can be difficult to quantify.

[49] Comments of the Staff of the Federal Trade Commission's Bureau of Consumer Protection, *In the Matter of The Internet of Things and Consumer Product Hazards*, Consumer Product Safety Commission, June 15, 2018, https://www.ftc.gov/system/files/documents/advocacy_documents/comment-staff-federal-trade-commissions-bureau-consumer-protection-consumer-product-safety/p185404_ftc_staff_comment_to_the_consumer_product_safety_commission.pdf (last accessed January 17, 2019).

Some companies like Google,[50] Apple,[51] and Microsoft[52] have bug bounty programs, while others like Amazon[53] simply have a responsible disclosure program. Bug bounty programs entice security researchers to disclose the vulnerabilities to the company, instead of monetizing the vulnerabilities on the Dark Web.

Google paid $112,500 to a security team that found a significant flaw in its Pixel phone software.[54] This is one of the largest known bug bounty payments. Even if companies do not advertise that they pay for vulnerabilities, they may choose to do so in certain circumstances. In addition, there are other forms of rewards like recognizing the security researcher that found a vulnerability on a "hall of fame" web page.

Other benefits of implementing a vulnerability disclosure program include mitigating issues before they become full-blown incidents with significant data breach, which will be much costlier.

---

[50] Google, *Google Vulnerability Reward Program (VRP) Rules*, https://www.google.com/about/appsecurity/reward-program/ (last accessed January 25, 2019).
[51] Apple, *Contact Apple About Security Issues*, https://support.apple.com/en-us/HT201220 (last accessed January 25, 2019); Apple, *Apple Web Server Notifications*, https://support.apple.com/en-us/HT201536 (last accessed January 25, 2019); Apple, *Submitting Bugs and Feedback*, https://developer.apple.com/bug-reporting/ (last accessed January 25, 2019).
[52] Microsoft, *Microsoft Bug Bounty Program*, https://www.microsoft.com/en-us/msrc/bounty (last accessed January 25, 2019).
[53] Amazon, *Vulnerability Reporting*, https://aws.amazon.com/security/vulnerability-reporting/ (last accessed January 25, 2019); Amazon, *Security & Privacy*, https://amazon.com/security (last accessed January 25, 2019).
[54] TechCrunch, *Google's bug bounty programs paid out almost $3m in 2017*, https://techcrunch.com/2018/02/07/googles-bug-bounty-programs-paid-out-almost-3m-in-2017/ (last accessed January 25, 2019).

2. Outsourcing Security Analysis

If a company cannot implement their own responsible disclosure program with its internal staff of engineers to respond to vulnerability reports, bug bounty as a service and crowdsourced bug bounty programs may be important options. An organization can hire a company, such as BugCrowd[55] or HackerOne,[56] to perform security assessments against their products or services to identify vulnerabilities that need to be fixed. This procedure is likely a less expensive option, since researchers are only paid if they find a vulnerability, rather than having to hire full-time employees to do this work. In addition, the company can set the scope for the assessment, so only portions of a product or service are assessed.

V. CONCLUSION

Software, while tremendously useful and deeply embedded in daily life, also includes vulnerabilities that threaten economic and social functioning and need to be fixed. Even if organizations that develop software do their best to find and fix vulnerabilities, security researchers still play important role in finding and reporting software vulnerabilities. But, myriad issues cloud the legitimacy of the work that security researchers do and inhibit the identification and rectification of vulnerabilities.

A multi-faceted approach to vulnerability disclosure is required. First, vulnerability discovery and disclosure can be encouraged by amending laws like the CFAA to provide exceptions for security research. Next, guidelines can be provided to prosecutors that clearly exclude security research from prosecution under computer crime laws. Third, the government can

[55] BugCrowd, https://www.bugcrowd.com/ (last accessed January 25, 2019).
[56] HackerOne, https://www.hackerone.com/ (last accessed January 25, 2019).

encourage and incentivize organizations to adopt responsible disclosure programs to remove legal uncertainty, thereby facilitating disclosure of vulnerabilities by researchers to these organizations. Finally, organizations can implement responsible disclosure or bug bounty programs, which clearly define a process for working with security researchers.