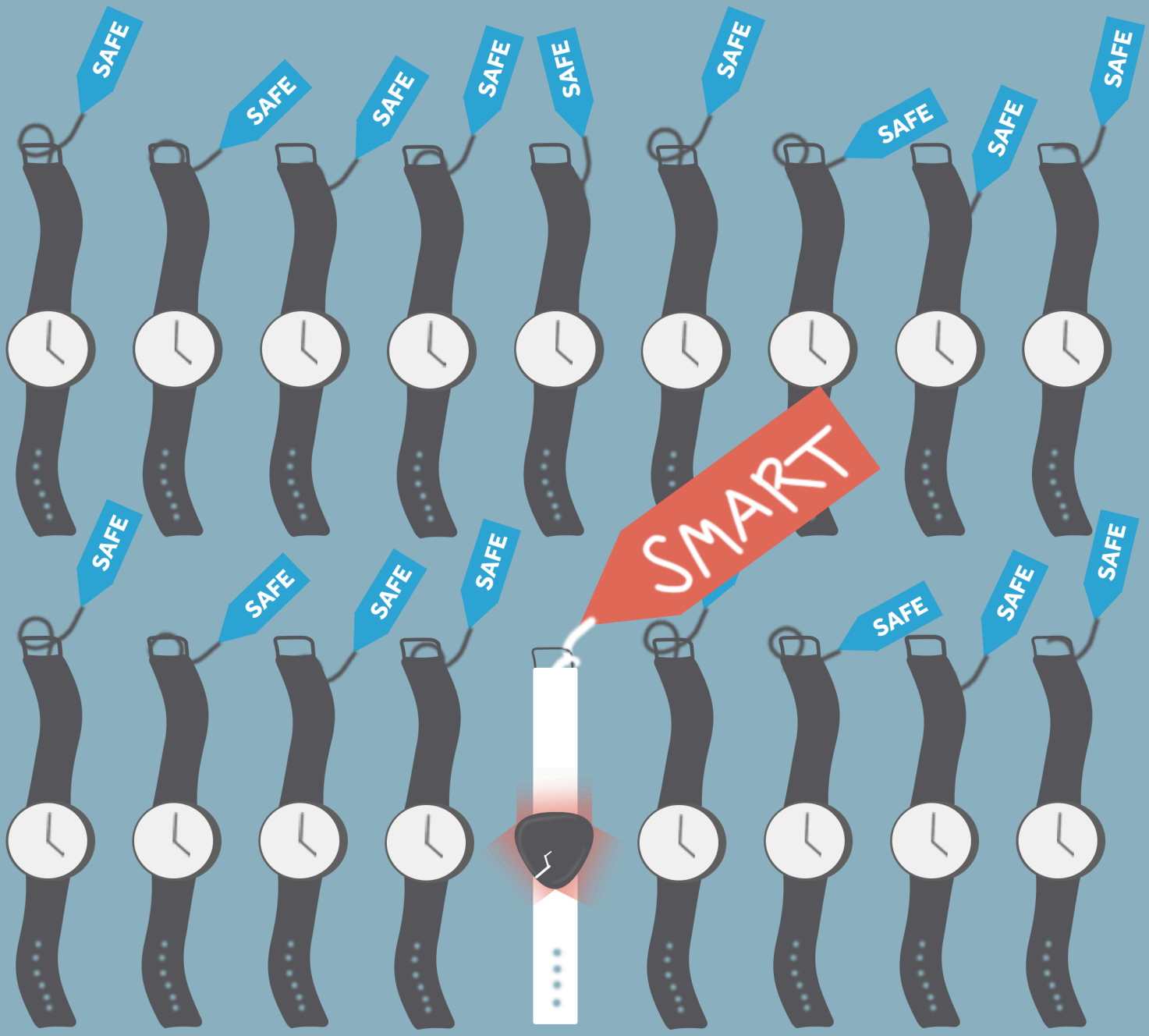
STRICT PRODUCTS LIABILITY
AND THE INTERNET OF THINGS



APRIL 2018

CENTER FOR DEMOCRACY & TECHNOLOGY

An Exploration of Strict Products Liability and the Internet of Things

Benjamin C. Dean
Center for Democracy & Technology

INTRODUCTION

Internet connectivity, software, and autonomous capabilities are increasingly integrated into all manner of devices and objects, creating the so-called Internet of Things (IoT). These devices range from fitness trackers to household appliances, automobiles to critical infrastructure. The goal is to create 'smart' objects - with greater convenience and efficiency - through the integration of digital components and capabilities.

Historically, products with these digital components and capabilities have not had sufficient security measures incorporated into their design. Several high-profile events of the past year, such as the WannaCry ransomware outbreak, which made use of flaws in the Windows operating system, and the discovery of the Spectre/Meltdown flaws in Intel chips, show that decades of defects are piling up. The well-being and safety of the millions of individuals who use these 'smart' products, as well as the commercial viability of enterprises that also use these technologies, are at stake.

In the past, strict products liability has not tended to apply to the designers, manufacturers, and/or retailers of digital products. This is because the impacts from the failures of these products have been limited to mostly economic damages, such as the inconvenient need to reboot a device or restore data from a backup. In the future, failures of increasingly ubiquitous IoT devices are likely to have more serious consequences such as damage to property, personal harm, or even death. This significant change in harms calls for policy makers to consider the allocation of responsibility for the harms. This different assignment of liability will affect the business models of companies that design, manufacture, and sell these technologies; the insurance market; and, ultimately, the trajectory of technological change.

The intent of this paper is to identify new risks driven by continued waves of technological change and the accompanying policy issues they raise. It aims to help the policy community understand a complex, multi-disciplinary domain without requiring deep technical expertise in

---

The author would like to thank Wendy Knox Everette, Philippa Scarlett, Vince Vitkowski, and David Vladeck for reviewing and providing comments on earlier versions of this paper. Thanks also go to CDT staff and members of CDT's Digital Privacy and Security Working Group and Internet Privacy Working Group for their comments during finalization of the paper. All errors and omissions remain the responsibility of the author.

each field explored (e.g. economics, law, and technology). It does this by providing concise explanations of relevant concepts then links them together in a way that identifies relevant questions. Such an approach is necessarily incomplete – perhaps even incorrect in parts. It is however a first step towards developing policy to effectively manage the potential risks identified and understand the potential allocations of associated liability. This will be crucial to ensure that those best placed to implement measures to make these technologies more secure at the lowest cost, do so, while, at the same time, ensuring that the pace of innovation is not unnecessarily stymied.

This paper starts by taking stock of the causes of insecure digital technologies. It then considers the potential harms that their failures have and may impose upon society, with a view to proposing ways in which to potentially allocate liability for the harms in the future. The conclusions of this report should be thought of as a research roadmap. Answering the questions raised in the conclusions – and using those questions as a springboard to identify and answer others – will provide a foundation for better future policy decisions in this space.

Why are digital technologies often insecure?

The insecurity of some digital technologies is taken as a given by some consumers. Some software requires annual updates, which in turn requires periodic patch downloads and installation to fix bugs in software. Anti-virus software often has to be purchased with a new computer then kept up-to-date in response to new threats.

Insufficient security measures, dangerous design or adding-on of security features post-design are not unlike the past practices of the automotive industry. Prior to the 1970s, some automobiles were designed without certain safety features e.g. the glare from dashboards that came with chrome finishing would periodically blind drivers. Some safety features were optional e.g. seat belts had to be separately purchased and installed by the customer. Some cars were even found to have been designed in a way that was inherently unsafe (e.g., the Ford Pinto).[1] A wave of civil suits in the 1960s and 70s led to new automotive safety laws and regulations, which in turn incentivized the safer design of cars that are now enjoyed today.

Why do unsafe design practices occur and persist? Cyber security is commonly considered to be a technical domain. If that were the case, solutions would lie simply in installing more technical security measures. However, this narrow approach ignores the root causes that lead some

[1] Nader R. (1965), “Unsafe at any speed”, Grossman Publishers: New York.

producers to not install security measures – potentially at a lower net societal cost – earlier in the design process.

A broader view partly attributes the absence of certain cyber security measures to weak economic incentives. These include network effects and a number of market failures, such as information asymmetry, negative externalities, and moral hazard.

> **Network effects**: Internet- and software-related industries (information technology industries) are characterized by demand-side network effects. When one person joins the network, it increases the value of using the network to others, which then encourages more people to join the network. Strong network effects are possible in markets where products or services can be provided at a low marginal cost, as is the case in many digital technology sectors. The end result is a winner-take-all dynamic where a premium is placed on moving first in a market so as to generate the network effects that ensure market dominance. Delays in delivering a product to consumers can be the difference between dominance of a multi-billion dollar market or bankruptcy. Installing additional security measures and following more rigorous software and hardware development processes can slow down development time and thus time to market. As a result, for a subset of companies, security measures are not designed into products.[2,3] Over time, this problem becomes more pernicious because design defects can be layered on top of one another.[4]
>
> **Information asymmetry**: It can be difficult for consumers to evaluate the security features of highly-technical products. It can also be difficult for consumers to evaluate the relative quality of software code because many producers use technical protection measures to protect software from inspection. These measures are illegal to bypass. When considering which products to purchase, if given a choice, consumers are not always able to assess which is truly the more secure option. Compounding matters, the inability to assess the relative security of a product means that those producers who have invested in more secure products cannot easily differentiate themselves in the

[2] Pfleeger S. L., Libicki M. and Webber M. (2007), "I'll buy that! Cybersecurity in the internet marketplace", IEEE Security & Privacy, Issue No. 03, May/June, Vol. 5.
[3] The industry practice has been to 'ship-now, patch-later'. This involves shipping software known to have bugs (though not known where and which bugs precisely) then, when bugs are found later, providing a patch for download to software users (at those users' bandwidth expense).
[4] Digital technologies are not invented anew with successive generations. If not rectified, flaws in old digital technologies persist while new flaws are introduced due to the addition of new, under-tested components. Often components interact, which introduces additional points of failure beyond the original and newly added flaws.

market. This prevents them from passing the additional cost onto consumers in the form of higher prices. This can result in a similar outcome to a 'market for lemons' for some products, i.e., where 'good' products are crowded out by the 'bad'.[5]

Externalities and moral hazard: Negative externalities are a cost imposed on society as a result of another party's actions. With digital technologies, the costs from cyber security incidents aren't always borne by the producer of the technology in question. As a result, a socially suboptimal (i.e., excessive) amount of the product in question is produced relative to the quantity that would be produced if the costs were borne by the responsible party ('priced-in'). Moreover, some producers do not implement sufficient security measures to reduce the probability of some classes of incidents given that the costs of the incidents are borne by others. This points to another market failure, moral hazard, which involves one party bearing the costs and losses due to the risky actions of others.

The consequences of market failures and insecure technology

Over many decades, weak incentives and market failures have led to an accumulation of insecure hardware and software. Events of the past year, such as the WannaCry ransomware outbreak (see box below), suggest that the potential liabilities due to this technical deficit are becoming increasingly real.

As these incidents occur, an essential question arises: Who is or should be liable for the costs of the incidents? In some cases, where complex and international supply chains are involved, the answers are not immediately obvious. However, finding such answers will become paramount over the coming decade if the full benefits from continued adoption and use of these digital technologies are to be realized.

[5] Akerlof G. A. (1970), "The Market for Lemons: Quality Uncertainty and the Market Mechanism", Quarterly Journal of Economics, The MIT Press. 84 (3): 488–500. doi:10.2307/1879431

CENTER FOR DEMOCRACY & TECHNOLOGY

Box. Who is responsible for damage due to the WannaCry ransomware?

In May 2017, a strain of ransomware malware dubbed WannaCry spread to hundreds of thousands of computers in at least 150 nations.[6] Ransomware encrypts the hard drive of the computer in question and demands that the computer's owner pay a ransom (usually in Bitcoin). If paid, keys are provided to decrypt the computer. If not paid, and no other solution is found, the data on the computer are rendered useless.

Thousands of individuals and organizations were harmed worldwide. Hospitals in the National Health Service in the United Kingdom, train stations, a Toyota car factory, and other organizations – as well as their clients and employees – were subsequently affected. The ransomware spread automatically to some systems while others required a person to infect the machine (e.g., by opening an infected email attachment).[7] Those who paid the $300-600 ransom could decrypt their hard drives. Some managed to decrypt without paying the ransom.[8] Others did not pay – or had back-ups. At the time of writing, the equivalent of over $150,000 had been paid in ransom to Bitcoin wallets associated with the malware.[9]

WannaCry made use of a vulnerability in various iterations of Microsoft Windows operating systems.[10] Microsoft had detected the vulnerability and issued a patch in March 2017, two months before WannaCry emerged.[11] However, not all users of the more than one decade old operating system had installed the patch.

In a twist to the story, the National Security Agency (NSA) of the United States of America had, for an unknown amount of time, possessed knowledge of the vulnerability and had been using it for intelligence purposes. Knowledge of the vulnerability became public when a cache of exploits, which

---

[6] Goodin D. (2017), "Massive cryptocurrency botnet used leaked NSA exploits weeks before WCry", Ars Technica, available from: https://arstechnica.com/security/2017/05/massive-cryptocurrency-botnet-used-leaked-nsa-exploits-weeks-before-wcry/ (accessed 18 September 2017).

[7] Goodin D. (2017), "Windows XP PCc infected by WannaCry can be decrypted without paying ransom", Ars Technica, available from: https://arstechnica.co.uk/security/2017/05/windows-xp-wannacry-decryption/ (accessed 18 September 2018).

[8] Ibid.

[9] To view the activity of the wallets associated with the malware see:
a. https://blockchain.info/address/115p7UMMngoj1pMvkpHijcRdfJNXj6LrLn,
b. https://blockchain.info/address/12t9YDPgwueZ9NyMgw519p7AA8isjr6SMw,
c. https://blockchain.info/address/13AM4VW2dhxYgXeQepoHkHSQuy6NgaEb94

[10] Twitter feed: VessOnSecurity; post at 9:51am on 18 May 2017, avaiable from: https://mobile.twitter.com/VessOnSecurity/status/865203180677812225 (accessed 18 September 2017).

[11] Goodin D. (2017), "An NSA-derived ransomware worm is shutting down computers worldwide", Ars Technica, available from: https://arstechnica.com/security/2017/05/an-nsa-derived-ransomware-worm-is-shutting-down-computers-worldwide/ (accessed 18 September 2017).

had been developed by and then stolen from the NSA, was dumped on the internet in April 2017. It is not yet clear if the NSA should have disclosed that it possessed knowledge of the vulnerability, as the Vulnerabilities Equities Process[12] would dictate, or whether the espionage value that the vulnerability possessed trumped the security interests of the individuals and companies later affected by the ransomware.

A group of unknown membership named the Shadow Brokers claimed responsibility for the theft of the NSA's cache of exploits and related vulnerabilities. American intelligence agencies have concluded that the North Korean government was responsible for the development of WannaCry.[13,14] This introduces another layer of complexity as an incident caused intentionally by a malicious state actor carries with it different legal treatment and consequences to those caused accidentally and/or by a non-state actor.

Whose fault is WannaCry and who should bear its costs given this complex situation? Is it the fault of the hundreds of thousands of individuals or organizations that did not implement adequate cyber security measures to protect themselves? Is it Microsoft, which created and sold over the course of a decade, the buggy operating system on which so many thousands of organizations rely? Is it the U.S. NSA, and the U.S. government by proxy, which procured and then lost the knowledge of the vulnerability used for WannaCry (and many other strains of malware)?[15] Is it the thieves who stole the NSA arsenal then publicly released the exploits? Or is it the developer of the WannaCry malware itself?

[12] The Vulnerabilities Equities Process (VEP) was put in place by the Obama Administration to help determine when knowledge of 'zero-day' vulnerabilities, purchased and/or held by intelligence agencies, should be kept secret or made public. It was an attempt to reconcile, on the one side, the needs of intelligence agencies to exploit vulnerabilities as a part of their signals intelligence mission, against the risks that continued existence of the vulnerabilities in question pose to all other users of the related technologies. For more information, see: https://jia.sipa.columbia.edu/online-articles/healey_vulnerability_equities_process#_edn8. The Trump Administration has built on the VEP by releasing specifics on the criteria used to determine whether vulnerabilities should be disclosed or not. For more information see: https://www.whitehouse.gov/sites/whitehouse.gov/files/images/External%20-%20White%20House%20Fact%20Sheet%20on%20VEP%20-%20FINAL%2011152017.PDF

[13] Bossert T. (2017), "It's official: North Korea is behind WannaCry", Wall St Journal, available from: https://www.wsj.com/articles/its-official-north-korea-is-behind-wannacry-1513642537 (accessed 3 January 2018).

[14] Ibid.

[15] An attack on IDT Corporation used the same stolen EternalBlue malware as that used for WannaCry. From: Perlroth N. (2017), "A cyberattack 'the world isn't ready for'", New York Times, available from: https://www.nytimes.com/2017/06/22/technology/ransomware-attack-nsa-cyberweapons.html (accessed 18 September 2017); Attacks on the mining software for crypto-currency Monero have been attrbuted to the stolen DoublePulsar backdoor. From: Zhao W. (2017), NSA 'DoubleStar' backdoor blamed for cryptocurrency mining malware, Coin Desk, available from: http://www.coindesk.com/nsa-doublestar-backdoor-blamed-cryptocurrency-mining-malware/ (accessed 18 September 2017).

Bugs in software are defects that lead to incidents such as WannaCry. Buggy software is not exceptional. Steve McConnell estimates that programmers make between 10 and 50 errors for every 1,000 lines of code. Careful checking at big software companies, he says, can push that down to 0.5 per 1,000 or so.[16] Consider that hundreds of thousands of lines of code are found in a typical device. For instance, Alfred Katzenbach, the director of information technology management at Daimler, reportedly said that the radio and navigation system alone in the S-class Mercedes-Benz requires over 200 million lines of code.[17] That equates to tens of thousands of bugs in just that component of an automobile – let alone the software already embedded in countless other devices today. Consider that the number of devices, and average size of source code, will continue to increase over the coming decade. This software is and will continue to be integrated into vehicles, medical devices, critical infrastructure, and all manner of devices that, in the event of failure, can cause bodily injury and/or property damage. In the absence of a profound paradigm shift in existing development practices, the scale of this already severe problem will only increase further.

Software bugs are only part of the larger security picture. A number of initiatives have emerged around determining a set of basic security measures or features that should be present in devices or objects that are connected to the internet and contain software. That there are so many initiatives is indicative of a widespread consensus that basic security measures are not being implemented in these products. The emergence of these efforts has been driven partly by decades of learning from real world incidents and partly driven by the requirements of the European Union's General Data Protection Regulation (GDPR) and Directive and the Directive on Security of Network and Information Systems (NIS Directive). A number of initiatives are currently underway to enumerate these security measures – for instance, *Consumer Reports*' development of Security and Privacy Standards for IoT Products[18] and the Open Internet of Things Certification Mark project.[19] The Cyber Independent Test Lab is also developing measures of software risk.[20] Some proposed measures relate to ensuring appropriate patching capabilities, use of end-to-end encryption (where appropriate), requirement for a user-defined or unique password in order to function (where appropriate), among many others. These

[16] The Economist (2017), "Why computer security is broken from top-to-bottom", http://www.economist.com/news/science-and-technology/21720268-consequences-pile-up-things-are-starting-improve-computer-security
[17] Robert Charette (2009), "This car runs on code", IEEE Spectrum, http://spectrum.ieee.org/transportation/systems/this-car-runs-on-code
[18] Consumer Reports (2017), "The Digital Standard", available from: https://www.thedigitalstandard.org
[19] IoT Mark, available from: https://iotmark.wordpress.com/principles/
[20] Cyber Independent Testing Lab website: http://cyber-itl.org

relatively basic features are not always included in new IoT products.[21] Ensuring that these security features are designed into new technologies in the future will be essential to a reduction in the occurrence of device failure as well as the reduction in the potential and eventual cost or injury to users of these technologies.

Public policy options to address market failures

The presence of market failures typically provides the justification for government intervention to correct those failures.[22] There are many possible interventions to address market failures including fines, labeling and standards, regulation, and product liability.

Regulatory measures seek to incentivize companies to internalize the damages caused by their practices. Fines are a common way in which to do so. However, fines are sometimes small in relation to the revenue that could be generated from the product in question, which may weaken the effectiveness of the fines. For instance, in the United States, the Federal Trade Commission (FTC) lacks the general authority to issue civil penalties. Instead, a company must become subject to a final order by the FTC, either via settlement or adjudication of an enforcement action, after which the FTC can bring an action in federal court to obtain civil penalties or other consumer redress.[23] The FTC can bring enforcement actions against companies alleged to have engaged in "unfair practices," which are defined as those that "cause[] or [are] likely to cause substantial injury to consumers which [are] not reasonably avoidable by consumers themselves and not outweighed by countervailing benefits to consumers or to competition."[24] Missing security measures on digital technologies might be considered as unfair practices in the event that they fulfill this criteria. The fines for continuing unfair or deceptive practices following such a decision are capped at a relatively low amount (depending on the nature of the violation – up to $40,000).[25] These fines are not linked to the

[21] Rose A. and Ramsey B. (2016). "Picking Bluetooth Low Energy Locks from a Quarter Mile Away". Presentation at DEF CON 2016, available from: https:// www.defcon.org/html/defcon-24/dc-24-village-talks.html; Antonakakis et al. (2017), "Understanding the Mirai Botnet", available from: https://www.usenix.org/system/les/conference/usenixsecurity17/sec17-antonakakis.pdf. For more information see: Kleinhans J. (2017), "Internet of Insecure Things: Can security assessment cure market failures?", Stiftung Neue Verantwortung, available from: https://www.stiftung-nv.de/en/node/2119

[22] OECD (1993), "Glossary of Industrial Organisation Economics and Competition Law", compiled by R. S. Khemani and D. M. Shapiro, commissioned by the Directorate for Financial, Fiscal and Enterprise Affairs, OECD: Paris.

[23] Federal Trade Commission (2008), "A Brief Overview of the Federal Trade Commission's Investigative and Law Enforcement Authority", available from: https://www.ftc.gov/about-ftc/what-we-do/enforcement-authority.

[24] 15 U.S. Code § 45(n).

[25] Federal Trade Commission (2016), "Inflation increases for maximum civil penalty amounts", available from: https://www.ftc.gov/news-events/blogs/competition-matters/2016/06/inflation-increases-maximum-civil-penalty-amounts

gross cost to consumers and society as a result of the unfair practices that have occasioned substantial injury to consumers. This limits their effectiveness as a means by which to incentivize companies to internalize the costs to consumers – and society more generally – of their products with missing security features.

To address information asymmetries, there is a need to provide clearer and more visible indications of product characteristics so as to allow buyers and sellers to interact on equal footing. This has typically been addressed through development of standards and associated labelling (e.g., EnergyStar ratings for the energy efficiency of some household appliances). No such label yet exists for security in the IoT space. Mandating minimum security standards is one other option, e.g., the proposed Internet of Things Cybersecurity Improvement Act of 2017, which would apply only to public procurement.[26] The difficulty with this option is that it can be hard to ensure that new problems are not created due to government failure, including regulatory capture. Ensuring that the minimum standard remains adequate in the face of continued technological development can also pose issues.

Finally, tort law is an avenue that allows people to receive compensation due to harm caused by the deceptive, negligent, and/or harmful practices of individuals or corporations. In the context of new technology, a strength of tort law is that it does not prescribe what specific measures should be put in place to improve the security of the technologies in question. Of course, tort litigation must necessarily occur *ex post* (i.e., after the harm has been incurred). At the same time, the threat of future liability due to application of tort can act as a deterrent (e.g. an incentive to take measures ahead of time to reduce the probability of subsequent harm/damage).

Strict Products liability: internalizing damage from products

Products liability establishes the liability of manufacturers, processors, distributors, and sellers when their products cause personal harm or property damage to others.[27] Three possible legal paths may be pursued: strict liability, negligence, and defective or inadequate warnings.[28] This

[26] IoT Cybersecurity Improvement Act 2017, available from: https://fr.scribd.com/document/355269230/Internet-of-Things-Cybersecurity-Improvement-Act-of-2017 (accessed 18 September 2017).
[27] Rusted M. and Koenig T. (2005), "The tort of negligent enablement of cybercrime", Berkeley Technology Law Journal, Volume 20 (4), 1553, available at: http://scholarship.law.berkeley.edu/btlj/vol20/iss4/4 (accessed 18 September 2017).
[28] Butler A. (2017), "Products Liability and the Internet of (Insecure) Things: Should Manufacturers Be Liable for Damage Caused by Hacked Devices?", University of Michigan Journal of Law Reform, Vol 50 (4),

paper focuses on just one of these paths, strict liability, and will not open an examination of the other paths.

Strict products liability ensues when harm is caused by or threatened by unreasonably dangerous products. One of its purposes is to ensure that the costs of harm to a person – or property of that person or a third party – due to a product are borne by the producer of the product.[29] Negligence and strict products liability have some similarities and differences, which emerge from varying past decisions across courts. Negligence usually involves a breach of a duty of care which leads to injuries by a first party. A breach may occur when measures are not taken to mitigate known risks. Strict products liability, by contrast, does not require the plaintiff to prove the defendant's prior knowledge of a risk (i.e. lack of knowledge of a risk is not a defense).[30] It can be invoked by third parties who have suffered harm and/or damage to their property due to a defect in a product regardless of whether a risk was previously known. Therefore, strict products liability cannot be transferred from product to user via contract, which renders infeasible the common practice in the software industry of absolving liability for the vendor through End User Licensing Agreements.[31]

The primary goal of strict liability is to assign risk to parties most proximate to it and able to prevent the harm. Two positive policy outcomes of this goal are to:

- Ensure that those who rush a product to market, at the expense of implementing security measures, are subsequently penalized for doing so. Products liability incentivizes producers to weigh the potentially small cost of mitigating the defective design or manufacturing element in their product against releasing the product with defects and having to cover potentially large eventual damages that these defects cause.
- Place all producers on the same legal footing. There would be no more shirking responsibility for poorly designed or manufactured products. This means that those who incur the costs of better security can pass it onto consumers in higher prices thus decreasing the "market for lemons" phenomenon.

For strict products liability to apply, the product has to cause harm, death, or property damage due to a defect in the product (proximate cause). A product may be defective in three possible

[29] Cohen H. and Brooks N. (2005), "Products Liability: A legal overview", CRS Issue Brief, Congressional Research Service, available from: https://digital.library.unt.edu/ark:/67531/metacrs6659/ (accessed 18 September 2017).
[30] Galligan T. C. (1991), "Strict liability in action: The truncated Learned Hand formula", Louisiana Law Review, Vol 52 (2).
[31] Alheit K. (2001), "The applicability of the EU Product Liability Directive to software", Comparative and International Law Journal of Southern Africa, Volume 34 (2), 188-209.

ways. There may exist a defect in the **design** of the product itself, a defect may be introduced during the **manufacturing** process, or defective or inadequate **warnings** may be provided to the consumer about dangers posed by the product. Wendy Knox Everette has already made the case for application of inadequate warnings about dangers posed by software linked to vulnerability disclosure.[32] This paper will instead focus on design defects.

For a product's design to be deemed defective, the plaintiff must prove that, "there is hypothetical alternative design that would be safer that the original design, as economically feasible as the original design, and as practical as the original design, retaining the primary purpose behind the original design despite the changes made".[33] In the U.S., one of two tests are commonly used to determine if this threshold has been crossed: the risk-utility test or the consumer expectations test. The application of each test varies across states.

The risk-utility test involves running a kind of cost-benefit analysis, referred to as Hand's Test, to weigh the cost to the producer of redesigning the product in a way that removes the dangerous defect against the benefits to society from removal of the defect. A product is considered defective if "the cost of an alternative design incorporating safety checks was less than the social risk resulting from their omission."[34] The consumer expectations test asks whether the consumer would expect a product to behave in a certain way or not. In the event that the product imposes damage or harm due to some expected behavior – such as a knife – then strict products liability is unlikely to apply. However, were the product to cause damage or harm due to unexpected behavior, then the consumer expectations test might be satisfied.

Institutional differences between the U.S. and the EU results in products liability being structured and applied differently. In short, products liability in the U.S. is found mainly in common law and in Article 2 of the Uniform Commercial Code, which is adopted and enforced at a state-level and deals with the sales of goods. In the EU, the Product Liability Directive was one of the first Directives to be put in place in 1985. Member States are then responsible for developing, implementing, and enforcing their own national legislation.

---

[32] Knox Everette W. (2016), "Security Vulnerabilities, the Current State of Consumer Protection Law, & How IOT Might Change It", BSidesLV 2016, available from: https://www.youtube.com/watch?v=EFGcZwjw9Q4&index=4 (accessed 18 September 2017).

[33] 'Design Defect', Legal Information Institute, Cornell Law School, available from: https://www.law.cornell.edu/wex/design_defect (accessed 17 September 2017).

[34] Hecht M. (2005), "Products liability issues for embedded software in consumer applications", Conference paper, IEEE Symposium on Product Safety Engineering, DOI: 10.1109/PSES.2005.1529521, available from: https://www.researchgate.net/publication/4186013_Products_liability_issues_for_embedded_software_in_consumer_applications (accessed 17 September 2017).

Given these differences, policymakers on either side of the Atlantic are beginning to embrace the idea of products liability for IoT devices in different ways. For instance, the U.S. President's Commission on Enhancing National Cybersecurity released a report in December 2016 entitled 'Securing and Growing the Digital Economy'. In this report, the commission recommended, "A law with regard to liability for harm caused by faulty IoT devices and…recommendations within 180 days."[35] Additionally, starting in January 2017, the European Commission commenced a process of reviewing the Products Liability Directive to examine whether it is fit-for-purpose with regard to the new technological developments, such as IoT and autonomous systems.[36] The outcome of these efforts will require a number of questions to be answered.

Turning to history for answers

Questions related to strict products liability have to be resolved:

A. *When is a digital product deemed defective?*
B. *Who is responsible for the defect?*
C. *Who is responsible for the damage caused by the failures of the digital technologies?*

There are still no conclusive answers – though history may provide some clues.

In the past, after waves of technological change, it has sometimes taken many decades for liability to be (re)apportioned in industries where technologies and associated products impose harms and costs on society (e.g., automotives, asbestos, nuclear energy, radon). This is partly because it can take time for the harms and costs of new technologies to become evident. It also takes time to develop an evidence base with which to conclusively prove the link between certain technologies and their associated harms and costs.

In the past, once damages have been identified, liability has commonly been allocated to the seller of the product in question. However, complicating the picture for digital (IoT) products are complex supply chains for the design, manufacture, assemblage, shipping, and sale of these

[35] Commission on Enhancing Cybersecurity (2016), "Report on securing and growing the digital economy", National Institute of Standards and Technology, available from: https://www.nist.gov/sites/default/files/documents/2016/12/02/cybersecurity-commission-report-final-post.pdf (accessed 25 January 2017).

[36] European Commission (2017), "Evaluation of Directive 85/374/EEC", presentation made at the Workshop on liability in the area of autonomous systems and advanced robots / IoT-systems, 13 July 2017, Brussels, available from: http://ec.europa.eu/information_society/newsroom/image/document/2017-30/hans_ingels_-_the_evaluation_of_the_product_liability_directive_62081123-0251-9FA1-AD58076EF15FAB7D_46142.pdf (accessed 18 September 2017).

technologies. It may be relatively easier to identify fault and allocate liability for IoT products in sectors where the original equipment manufacturer/value-added manufacturer (OEM/VAM) model is more established than others.[37] For instance, the OEM model is well established in industries such as the automotive industry, which is an industry where a great deal of digital innovation is occurring around automobiles with autonomous capabilities. Drawing lessons from this industry, and applying them to other industries and sectors, may be one promising way in which to resolve some elementary liability questions raised by technological change associated with the IoT.

The issue of liability allocation and digital technologies has been wrestled with several times over the past twenty years as the various concerned industries have evolved: first, as the niche software industry emerged and grew in the 1980-90s; then, as the internet was commercialized and globalized in the 1990s; and most recently, as digital technologies found their way into the hands of people worldwide in the 2000s.[38] Each time, different reasons for and against the imposition of liability have been provided and decisions made by either governmental bodies or the judiciary.

When has product liability applied to digital products in the past?

There have been a limited number of past instances when class action lawsuits, some of which related specifically to strict products liability, as well as regulatory action, were successfully brought against companies linked to defects in their digital products. These suits have principally been brought in the U.S. rather than Europe. This is perhaps more reflective of the differing class action system in the U.S. than it is absence of similar issues having arisen in Europe. In some instances in the U.S., the lawsuits ended in a settlement, thereby preventing a conclusive ruling as to whether or not software or security practices could be considered as defects leading to the imposition of strict product liability.

One of the earliest cases, dating back to the late 1980s, was in the medical devices field. It involved cancer patients receiving overdoses of radiation due to a bug in the software that operated the radiation-dispensing device. Two patients died and several others suffered

[37] The OEM model involves one company manufacturing products (the original equipment manufacturer [OEM]) that are then marketed and/or branded by a different company (the value-added reseller [VAR]). Depending on the way in which the contractual relationships are set-up, certain liabilities may reside with either party.
[38] Zollers F. E., McMullin A., Hurd S. N. and Shears P. (2005), "No more soft landings for software: Liability for defects in an industry that has come of age, Santa Clara High Technology Law Journal, Volume 21 (4), 767.

injuries as a result of the defect.[39] Alleging that the product was defective, unreasonably dangerous, and not fit for its intended use, the claim was eventually settled for an undisclosed amount of money.[40] Another case involved a malfunctioning device that allowed patients to self-administer pain medication. Again, the case was settled, so no formal outcome emerged, though the case was presumed to have been brought under strict liability linked to defective software.[41]

In automotives, a New Jersey court applied product liability to a case where defective software on the on-board computer of a tractor-trailer failed, leading to a collision.[42] General Motors paid $15 million in punitive damages after the Alabama Supreme Court agreed that a defective computer chip in a model of pick-up truck had led the the death of a driver.[43] Another example occurred in 2012 when the Toyota Motor Corporation was subject to design defect litigation related to the 2005 Camry L4 model and sudden vehicle acceleration without any act by the driver. The suit claimed a software defect prevented drivers from braking or disengaging the accelerator pedal once the vehicle had suddenly accelerated. A first round of analysis by NASA scientists, "found no evidence that a malfunction in electronics caused large unintended accelerations".[44] Subsequent examination by two expert witnesses found that there were defects in the software throughout the automobile.[45] After inspection of the codebase one expert witness characterized it as 'spaghetti code'.[46] He was echoing the use of the same term by Toyota employees in prior discussion of the software with National Highway Traffic Safety Administration representatives.[47] A class action lawsuit was settled for economic loss due to the defect. The settlement was for $1.3 billion and other relief.[48] More recently, Tesla faced a class

[39] Armour J. and Humphrey W. S., 1993, "Software Product Liability", Technical Report CMU/SEI-93-TR-13, August 1993.
[40] Zollers et al. (2005).
[41] Tyde J. (1990), "Medical computer software: Rx for deadly errors", Software Law Journal, Volume 4(1), 117.
[42] Roberts v Rich Foods, Inc.; in Rustad and Koenig.
[43] General Motors Corp. v Johnston, 592 So.2d 1054 (Ala. 1992); in Rustad and Koenig.
[44] NASA (2011), "NASA's Toyota Study Released by Dept. of Transportation", available from: https://www.nasa.gov/topics/nasalife/features/nesc-toyota-study.html (accessed 5 January 2018).
[45] Both experts' full testimony can be read at: http://www.safetyresearch.net/Library/Bookout_v_Toyota_Barr_REDACTED.pdf. The slides of one expert, Michael Barr, can be found at: http://www.safetyresearch.net/Library/BarrSlides_FINAL_SCRUBBED.pdf
[46] Incomprehensible and badly structured source code (typically including apparently meaningless jumps of gotos or a high degree of unnecessary coupling between modules.
[47] See: http://www.safetyresearch.net/Library/Bookout_v_Toyota_Barr_REDACTED.pdf
[48] Vladeck D. C. (2014), "Machines without principals: Liability rules and artificial intelligence, Washington Law Review, Volume 89(1), 142.

action in California linked to the sudden and unstoppable acceleration of a Model X.[49] Amongst the many complaints is a strict product liability claim. Court filings make many references to software in the context of cars being "computers on wheels," which carries with it cybersecurity issues.[50] At the time of writing the case is in mediation.[51]

Other instances are unrelated to products liability though they involved the failure of digital technologies. FDA action, in the form of product recalls, have occurred due to defects in software. For instance, the software application cards of Medtronic devices have several FDA-enforced recalls (in at least 2004, 2012, and 2016) due to software issues leading to overdoses to patients,[52] which led to patient harm and death.[53]

Finally, the aviation industry has a long history of software-related incidents and crashes, although strict product liability suits have not usually been the outcome. For instance, it is believed that software failure was the probable cause of the 1996 crash of a Boeing 757 in Peru, which killed seventy people.[54] An Airbus A330 nose-dived suddenly, twice, while flying off the coast of Western Australia during a Qantas flight in 2008. A number of passengers who were not wearing seat belts, and crew members, were seriously injured. According to a government investigation following the incident, the incidents were due to the malfunction of the software on one of the air data inertial reference units, which are used in the on-board autopilot system.[55] Both the aircraft manufacturer, Airbus, and the company responsible for producing the faulty ADIRU, Northrop Grumman, were subsequently sued by passengers and the Qantas pilots with millions of dollars in damages awarded.[56]

---

[49] Reuters (2017), "UPDATE 1-Tesla owner files lawsuit in California claiming sudden acceleration", available from: https://in.reuters.com/article/tesla-lawsuit/update-1-tesla-owner-files-lawsuit-in-california-claiming-sudden-acceleration-idINL1N1EQ01O (accessed 22 September 2017).

[50] Ji Chang Son, et al. v. Tesla Motors, Inc., No. 16-2282, C.D. Calif, available from: https://fr.scribd.com/document/335459806/Telsa-Lawsuit (accessed 22 September 2017).

[51] See: https://www.pacermonitor.com/public/case/20251580/Ji_Chang_Son_et_al_v_Tesla_Motors,_Inc

[52] US FDA recall notices for MedTronic devices linked to software issues.
For 2004: https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfRES/res.cfm?id=34649;
For 2012: https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfRes/res.cfm?id=107986;
For 2016: https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfRes/res.cfm?id=150480

[53] Zollers et al. (2005), 767.

[54] Rustad and Koenig (2005), 1578

[55] Australian Transport Safety Bureau (2008), "In-flight upset - Airbus A330-303, VH-QPA, 154 km west of Learmonth, WA, 7 October 2008", Investigation number: AO-2008-070, available from: http://www.atsb.gov.au/publications/investigation_reports/2008/AAIR/pdf/AO2008070_interim.pdf (accessed 18 September 2017).

[56] WAToday (2011), "Qantas plunge injured win payout", available from: http://www.stuff.co.nz/travel/travel-troubles/6173360/Qantas-plunge-injured-win-payout (accessed 22 September 2017).

What has prevented application of strict products liability in the past and what may change?

Insecure and defective digital products are not new. Why then has strict products liability only been applied in a few instances? There are a number of answers to this question.

Strict products liability requires that the plaintiff show physical harm, death, or property damage due to the defective product.[57] Called the 'economic loss doctrine', there are limits placed on claims based on financial losses linked to productivity loss, business interruption, and loss of data.[58] These are the kinds of impacts that have resulted from insecure digital products in the past. In at least two cases, the economic loss rule precluded recovery in tort.[59] As digital technologies are built into more and more devices the potential for physical harm may grow.

Moreover, it has been difficult to demonstrate that missing security features or digital defects alone led to harm or damage and provide an empirically-based cost/benefit calculation with supporting probabilities for claims.[60] This situation might be changing though as historical incidents provide a basis with which to calculate the likelihood of certain classes of device intrusion and/or failure. Commonly understood development practices might also arise that can reduce these risks.

At a technical level, satisfying the risk-utility test might be possible given that software can be copied at close to zero marginal cost and the devices are purchased and used by millions of (potentially) harmed people.[61] This analysis is not clear cut given the additional cost that the rearrangement of development processes to incorporate more secure or reliable practices would entail. However quality programming frameworks and standards are now better developed and established than they were in the past. Their application is thus easier and potentially cheaper than at points in the past where they either did not exist or were not sufficiently developed for application.

It may also become easier to satisfy the risk-utility test (following the Hand formula) because of improvements in the methods with which to estimate the economic costs and losses of incidents. To reiterate, the risk-utility test requires weighing the cost to the producer of

[57] Butler (2017).
[58] Vitkowski V. (2015), "The Internet of Things: A new era of cyber liability and insurance", Declarations, International Association of Claim Professionals, Spring 2015.
[59] See: Benning v Wit Capital Group Inc. and NMP Corp. v Parametric Tech Corp; in Rustad and Koenig (2005), 1580.
[60] Ibdi, 1579.
[61] Hecht (2005).

redesigning the product in a way that removes the dangerous defect against the benefits to society from removal of the defect. In the past, it was difficult to quantify the economic costs and benefits of each side given that the technologies themselves were so new and both the short- and long-term consequences of their failure hadn't been witnessed or measured. With decades of historical data with which to conduct analysis, the consequences of incidents can now be more reliably estimated than in the past.

While mapping the risk-utility test to digital security measures might not be easy, and the convenience benefits of devices with less secure features would have to be balanced against the costs associated with their absence, the point is that the estimation of these values is becoming relatively easier. In addition, insurers are increasingly asked to underwrite the costs and losses from cyber security incidents. As their models improve, based on historical claims data, insurers will be increasingly able to estimate potential losses in the course of underwriting new policies. This evidence base might also be used to satisfy requirements under products liability law such as the risk-utility test and the least-cost avoider principle.

An additional wrinkle is that there is ambiguity as to whether software is considered a product or a service.[62] This is important because products liability only applies to products – not services. In some U.S. states, software has been treated as tangible or intangible property.[63] In some instances, software or firmware were claimed to be a list of instructions – not a tangible item.[64,65] Yet over time, products liability has stretched the definition of tangible property to also include intangibles (gas), naturals (pets), real estate (house), and writings (navigational charts).[66] If put to the test, it is possible, at least in some U.S. states, that software integrated into objects would be considered both as a part of tangible property and as a product, particularly in the context of incidents linked to IoT devices. In this event, liability is likely to accrue to the seller of the product and/or the designated OEM depending on the context.

[62] Neuburger J. D. and Garde M. E. (2004), "Information security vulnerabilities: Should we litigate or mitigate?", Working Paper Series 121, Washington Legal Foundation, available from: http://www.wlf.org/upload/NeuburgerWP.pdf (accessed 18 September 2017).
[63] Alheit K. (2001).
[64] Hecht (2005).
[65] Scott M. D. (2007), "Tort liability for vendors of insecure software: Has the time finally come?, Maryland Law Review, Volume 67 (2), available from: http://digitalcommons.law.umaryland.edu/cgi/viewcontent.cgi?article=3320&context=mlr (accessed 18 September 2017).
[66] 'Products liability', Legal Information Institute, Cornell Law School, available from: https://www.law.cornell.edu/wex/Products_liability (accessed 18 September 2017).

Finally, three defenses might be mounted by producers - with varying levels of success depending on the court and/or individual facts of a case in question - so as to evade application of strict liability due to design defects. First, producers might claim that, although the cost of the safety change is known after the product is in use, it could not have been reasonably foreseen at design time.[67] Second, producers might claim that the 'state of the art' of the field was such that it was not technically feasible to identify and remedy the defects at the time of design.[68] Finally, producers might claim that the incident occasioning harm or property damage was caused by a third party ('cyber criminals' or government-sponsored hackers) and, therefore, the third party, not the producer, should bear the liability.[69] Each of these defenses might be weakening.

The foreseeability of certain incidents occurring due to poorly written software code or poorly configured (or not implemented) cybersecurity features is now much higher than one or two decades ago. Previously, a producer might claim that an empirically-based risk/benefit calculation for certain security features would be impossible given the impossibility to foresee computer intrusions.[70] After twenty years of such intrusions occurring, this defense is less and less satisfying, especially in instances where security measures aren't taken for relatively common threat actors and/or known vulnerabilities (e.g., using memory-unsafe programming languages can result in buffer overflows).

With regard to the state of the art of the field defense, rapid advances in the field of artificial intelligence are making it more and more technically feasible to identify and remedy bugs in code bases at scale.[71]

Finally, with regard to the proximate cause defense, while incidents related to devices with software coupled with internet connectivity can often be initiated by a third party, with a better established history of incidents linked to internet connectivity or poorly written software, the concept of 'foreseeable bystanders' may now apply under certain conditions.[72] For instance, it would be difficult to say that it was unforeseeable that routers with hard-coded, factory-set passwords could be hijacked as part of a botnet and used for denial of service attacks against innocent parties. This and many other high-profile incidents have raised awareness as to the

[67] Hecht (2005).
[68] Alheit (2001).
[69] Butler (2017).
[70] Rusted and Koenig (2005).
[71] Indeed, many of the companies that have been responsible for buggy code bases in the past are the very companies that are at the forefront of the development of artificial intelligence.
[72] See Butler (2017) pp 109-110.

known risks that certain software development practice entail – and the risks that internet connectivity brings – that mitigates previous defenses related to proximate cause and unforeseeability.

Areas of further study

The advent of the IoT might result in greater application of strict product liability law in ways that have not traditionally applied to prior products with software, internet connectivity and/or autonomous capabilities. Some key elements driving this shift are the increased use of these technologies in what are considered more 'traditional' industries (i.e., automotives, consumer electronics, medical devices, etc.), which heightens the risk of property damage and/or personal injury from incidents linked to defects in the software (possibly satisfying the economic loss doctrine), and the coupling with the now well-established historical record of incidents linked to known defects in software (possibly satisfying foreseeability).

Businesses that wish to take advantage of the commercial opportunities afforded by these technological advances would be wise to begin considering ways to manage the risk of future litigation by improving the security of relevant products. A strict products liability system does not necessarily have to result in lower innovation. Rather, encouraging responsible innovation can foster consumer trust, which in turn can result in faster or greater adoption of the technology. Such a liability system may also put producers on an even playing field in that competitors are not able to undercut the market with less secure devices then avoid incurring the costs linked to the damage caused by the failure of these devices.

However, ensuring such an outcome requires a well thought out and carefully considered response from stakeholders in the private and public sector. Below are a series of pending questions which, if answered, might lead to decisions concerning a strict liability system that balances the needs of consumers to safely benefit from these products against the needs of producers to be able to innovate and sell their products profitably. This is by no means a complete list – particularly given the as of yet unknown trajectory of technological change.

What are 'digital defects'?

Clearly defining what can be considered defects, and under what conditions or levels, will be important to ensure that strict products liability is applied in appropriate instances.[73]

[73] Vitkowski (2015).

For instance, when narrowly considering software, not all bugs are vulnerabilities and not all vulnerabilities can be exploited.[74] Clearly delineating between these concepts, across different types of software and/or firmware, and applying the designation of design defects, when appropriate, is likely to be necessary.

It may also be important to establish the absence of which security features, in certain contexts, should be considered as a defect. Establishing standards around IoT security, which is already occurring, is a major step forward in resolving when the absence of certain security features, in certain contexts, should be considered as a defect.

Moreover, it may be necessary to determine when and under what conditions 'faulty' data could be considered a design defect. Increasingly, critical decisions will be based upon data delivered in real, or near real, time from the IoT and other sensors. Failures may occur if the adequacy of the data used to make these decisions is not sufficient (in terms of accuracy, completeness, timeliness, precision, etc.). There is a body of case law that could be applied to help develop answers to these questions.[75]

All of these various definitions will have to be able to cope with technological change over time so as to remain relevant and effective. For instance, the definition of 'adequate' security measures, or what specific bug or vulnerability is considered a defect, is likely to change over time as common or best practices evolve in line with technological change (for instance, bugs that allow for SQL injections, or lack of measures to mitigate SQL injections, might have been considered as non-defective in 1994 but is unlikely to be widely considered as such in the present day).

Should data be considered as property?

Data has not traditionally been considered property. As a result, when malfunctioning software or hardware has occasioned the corruption of data, strict products liability has not been applied. However, as people's lives continue to be 'virtualized' and value continues to be created by and be contingent upon data, a conception of 'digital assets' or 'digital property' may emerge. This is not the first time that such a designation has occurred. For instance, in the EU Product Liability Directive, specific mention is given to 'electricity' as a product. While a

[74] ICANN (2015), "Threats, vulnerabilities and exploits - oh my!", available from: https://www.icann.org/news/blog/threats-vulnerabilities-and-exploits-oh-my (accessed 22 September 2017).
[75] For instance, in the 'Jeppesen & Co cases' regarding erroneous information in mass produced aviation charts, courts held that certain types of information could in certain contexts be deemed a product and that product liability law would apply to errors in such information. For more information see: Scott (2007), p49-52.

similar designation for 'data' or 'software' may not be necessary, there is a need for further definition around the property rights and ownership with relation to certain classes of data (e.g., personally identifiable data).

What is the allocation of liability along the supply chain?

Value/supply chains in many industries are set to change as software, internet connectivity, and autonomous capabilities are integrated into more-and-more objects. The liability issues raised by adoption of these technologies may be different to those that have traditionally applied to these industries (e.g., utilities, some consumer goods, etc.). Clearly demarcating where liability or joint-liability lie as these chains evolve will be important if unnecessary and costly lawsuits are to be avoided in the near future. While contractual arrangements might allow for the allocation of liability between parties, in the event that strict products liability applies (which cannot be transferred by contract), companies will need to be able to show who was responsible for which component in different products and the quality assurance or safety standard used to assess that component. This will in turn require development of digital technology failure standards and thorough incident investigation.

Will software failure standards develop?

Most components on modern aircraft that make use of software are considered highly reliable. This is because aircraft manufacturer must test and ensure that software meets a minimum standard in terms of the time to failure.[76] No such standards exist for software in automotives or other consumer devices. Developing and implementing such standards would go a long way towards improving software reliability, under the threat of application of strict product liability in the event of failure.

Will mandatory incident investigation arise for device failure?

Part of being able to establish who is liable following an incident is a thorough, publicly available and mandatory investigation of the incident in question. This has been a standard procedure in the aviation industry for many decades. These investigations, often undertaken by a public authority, have proven to be helpful when litigation following an accident has occurred and liability has had to be allocated amongst many different parties. Investigations allow one to identify the source(s) of failure that led to the incidents and, in this way, identify whether defects (such as bugs in software and/or missing security measures) contributed to the incident.

[76] Australian Transport Safety Bureau (2008).

So far, there is no equivalent for digital security incidents although proposals for such investigations – or a public authority to undertake such investigations – have been made for many decades.[77] The automotive sector saw the creation of the Cornell Automotive Crash Injury Research Center at first (formerly known as the Cornell Aeronautical Laboratory), then the National Highway Traffic Safety Administration and National Transportation Safety Board, then major automotive companies themselves opened their own labs. Funding for an academic center for digital security incident investigation might be a first step in this area.

What is the role of open versus closed source software?

There are a number of complicated issues surrounding the potential application of strict products liability to open source or closed source software.

It may be necessary to treat open source software differently than closed source software. Often one cannot examine the codebase for closed source software due to technical protection measures and/or clauses forbidding such practices in End User Licensing Agreements. This means that bugs in that closed source software can persist, and place the safety of users of that software in jeopardy. Open source software, on the other hand, can be audited by users (or professional auditors) so as to identify and patch software bugs. If the concept of strict products liability is applied to incidents involving software failure, it may be necessary to establish whether closed source software should be treated differently.

However, open source software cannot be considered facially superior because it can be written in a way that is complex and thus difficult to understand. This would preclude users or experts from being able to adequately audit the software. To absolve those who write open source software entirely from liability may end up encouraging the opening up of existing closed source codebases but, at the same time, could encourage the unnecessary development of overly complex codebases. This would have the consequence of increasing the risk of software failure, a counterproductive outcome for user safety. Finally, open source software is commonly developed by communities of people. It may not be clear if and to whom liability could or should be allocated in the event that bugs in open source software contribute to incidents, which in turn cause physical harm or property damage.

[77] Bellovin B. and Shostack A. (2016), "Input to the Commission on Enhancing Cyber Security", available from: https://www.cs.columbia.edu/~smb/papers/Current_and_Future_States_of_Cybersecurity-Bellovin-Shostack.pdf (18 September 2017).

How to manage liability issues associated with autonomous machines?

The integration of autonomous capabilities into objects raises unresolved questions pertaining to the allocation of liability in the event that the objects cause property damage or personal injury. These will be machines "that can define [their] own path, make [their] own decisions, and set [their] own priorities."[78] There will also be failures of these machines due to the inducement of third parties, such as has already been proven through research on adversarial perturbations.[79] Reaching answers to questions of liability for such machines in these contexts will require deep examination. Suggestions to date have included application of a system of strict liability, which would require a court-compelled insurance regime given the shortcomings of tort law in resolving questions of liability; the application of a common enterprise liability system where each entity within a set of interrelated companies is held jointly and liable for the actions of other entities in the group; or granting legal personhood to the autonomous machines, which would allow one to essentially sue the machine but would, in turn, require that all stakeholders including the 'owner' of the machine take part in a self-insurance pool.[80]

Conclusion

These questions offer a first step towards developing effective policy to manage expanding risks to consumers, business, and society as a whole from the proliferation of IoT products. Strict products liability is not an area that stakeholders in the digital technology space have had to contend with a great deal in the past. There is no denying that it involves complex relationships, balancing of equities, and difficult to allocate responsibility. Yet, ignoring the issue is not an option as the mounting costs and potential harms from the failure of IoT products become apparent. CDT is planning future research to explore the most pertinent of these questions in-depth to provide relevant, rigorous guidance to policymakers in this increasingly important and constantly evolving area.

[78] Vladeck (2014).
[79] Adversarial perturbations are small modifications to images or objects that lead to misclassification by deep neural networks. These modifications can be imperceptible to the human eye. For more information see: Evtimov I., Eykholt K., Fernandes E., Kohno T., Li B., Prakash A., Rahmati A. and Song D. (2017), "Robust Physical-World Attacks on Machine Learning Models", available from: https://arxiv.org/pdf/1707.08945.pdf (accessed 11 October 2017).
[80] Ibid, 149 (note 94).