S4: Simple, Secure, Survivable Systems Human-first crisis technology design principles

Kelsie Nabben and Paul Gardner-Stephen
PhD candidate, RMIT Blockchain Innovation Hub, Melbourne, Australia, kelsie.nabben@rmit.edu.au
College of Science & Engineering, Flinders University, Australia
Email: paul.gardner-stephen@flinders.edu.au

2020 IEEE Global Humanitarian Technology Conference (GHTC) | 978-1-7281-7388-7/20/$31.00 ©2020 IEEE | DOI: 10.1109/GHTC46280.2020.9342856

***Abstract*—Information technology has become embedded in almost every area of modern life. The many complex digital systems that support modern societies are now highly dependent on the correct function of complex and highly interdependent technological systems. Digital tools are increasingly becoming part of traditional crisis response efforts by government and non-government organisations. While digital tools have substantial capabilities to enhance crisis response efforts, they also pose significant risks to user communities when deployed in time-sensitive, vulnerable and fragile crisis contexts – as part of an already complex system. These risks and inefficiencies have been demonstrated in the contact-tracing application debate in the response to the COVID-19 pandemic.**

Technology must be intentionally designed and implemented, both to help solve the problem at hand and support end user communities. The principles of simple, secure and survivable systems (S4) offer a framework for technology that serves the interests of end-users and maintains human dignity, especially in crisis situations. The S4 principles are already evident in a number of technology projects, across research, design, build and deployment phases. Instead of high-risk, ad hoc, reactive digital solutions, crisis responders can pre-emptively share information, invest and work with existing technology design and development experts that reflect the S4 principles for efficient, effective solutions that enhance response capabilities both now and in future scenarios.

I. Introduction

This paper seeks to partially explore the interdisciplinary design and engineering research questions, of:

- What design principles can be applied in order to effectively design, build and deploy technology that serves people; both before, during and after a crisis?
- How can we actively support existing crisis technologies, instead of reactively rushing to design, build and deploy digital solutions during a crisis?

A. *Contributions*

The contributions of this paper are:

1) This paper proposes four fundamental socio-technical design principles of simple, secure survivable systems (S4) for the conception and creation of crisis response digital solutions. These core principles offer a human-first technology design framework.
2) Identifies the rather pessimistic prospects of adopting S4 principles broadly, and that rather a targeted approach is required, that focuses efforts on the niche areas where S4 systems can already be competitive or attractive enough to succeed in the market place.

B. *Structure of this paper*

Information technology has been leveraged throughout almost every part of society, especially in the developed world, to amplify the effectiveness and efficiency of our lives. Infrastructure, from traffic signals and public transport operation, through to bill payment and retail sales, all depends on these interconnected systems. These inter-dependencies create significant fragilities and vulnerabilities that become acutely apparent during crises.

The inaccessibility or failure of even one of these systems can cause significant damage, from loss of productivity to loss of lives. The concurrent failure of multiple systems has the potential to create widespread and long-lasting disruption, as each additional degraded or non-functional system increases the probability of complications in restoring the failed services [1]. The problem of how to bring the system to operation in the event of loss of many or all systems, when each depends on the other, can be referred to as a 'bootstrap problem' [2]–[5]. Because these systems have each evolved in the presence of one another, it may no longer be possible to bootstrap from a cold start situation where some critical set of systems have failed. Alternatively, it may be possible, but require such a long time that other systems will fail or degrade, resulting in a cascading failure of critical systems.

Information systems have become ever more complex as they have evolved. It is now not uncommon for critical systems to include many millions or even billions of lines of source code, such as mobile telephones and cellular network infrastructure. The result is that it is impossible to ensure the correct operation or resilience in the face of crisis or failure; even relatively simplistic cyber-attacks or incorrect data input. For example, new software flaws that require constant patching and software updates are routinely discovered in even old software, which may themselves create new problems.

II. System Errors

Real-world examples of information technology demonstrate the problem with complexity.

A. *General Computing*

SPECTRE is a family of hardware-level security vulnerabilities discovered to exist in practically all modern computer processors [6]–[9]. That the problems exist in the hardware design of the processors means that a complete solution would require the replacement of billions of devices. Instead, the approach taken has been to create software patches that mitigate but don't completely eliminate the vulnerability, at the cost of reducing the effective speed of the affected computers by up to thirty percent [8], [10]. However, the initial patches made the computers unreliable, and it was several months before they could be considered of a high enough quality to be safely used [10].

B. *'Smart' Phones*

Mobile phone insecurity has been a significant problem for at least a decade, with many vulnerabilities being discovered each year, exposing hundreds of millions of people's primary communications devices subject to subversion or denial by hostile parties. Making this problem worse is even widely adopted mobile operating systems such as Android are infamous for the lack of ongoing updates for almost all models of telephones [11]. The Android operating system is so complex that it requires considerable effort to maintain support for old handsets. In any case, a vendor's older models are the main competition for their future sales, so there is a double economic disincentive for them to provide such updates. This is an example of the inter-dependencies crossing over from being purely technology issues into the commercial and social domain.

C. *Telecommunication networks*

Core network equipment is also problematic. For example, the current attention on Huawei's 5G cellular network offerings stems from the simple reality that these modern systems are so complex that it is impossible to determine whether or not there are any backdoors installed, whether in software or hardware [12]–[14]. The same issues apply to Cisco, Ericsson and many major vendors in this space, because it is the complexity of these systems that makes them untrustable, rather than their nation of origin.

D. *Capability maintenance in crisis*

It is this issue of ever-increasing complexity, driven by the desire for increasing functionality and performance at the cost of simplicity, that is at the heart of the problem: Long-term stable future operation is traded-off for reduced time-to-market, increased performance or other economic or social benefits. Jonathan Blow captures the result of this succinctly, when he notes that we are no longer surprised when software malfunctions, having accepted it as normality [15].

The implicit justification is that the businesses producing these systems will continue to be able to provide updates and new products, sustaining capabilities enough or providing customer support where needed, so that system failure can be perpetually avoided. That is, we incur a substantial collective debt in future maintenance of capability. However, this only works in predictable circumstances – until there is a sufficient shock that overwhelms the ability of one or more of these capability maintainers.

From a macro perspective, the collective loss of capability and productivity reduces the ability of society to sufficiently rapidly deploy or reallocate resources to restore the failed capability maintenance functions or integrate new systems that work harmoniously with existing infrastructure.

It is also possible for capability maintenance to "fail live", meaning, for the capability maintenance function to be lost while the capability remains available to society for some time before it fails. In the best case, this can lead to expensive re-creation of the maintenance capability, such as the US Department of Defence having to spend $300 million dollars to re-discover how to make a key component of their nuclear weapon after the capability had been lost [16]. In the worst case, it can lead to long-term or permanent loss of capability, which could lead to societal collapse if there are no effective substitutes, or if the loss of capability comes at a critical time.

The ability to design, audit, modify and deploy critical systems that work with existing infrastructure or independently is acutely important in crisis situations.

III. Design and deploy during crisis

One way that complexity can be circumvented to maintain critical capabilities during crisis is by designing and deploying entirely new, replacement systems. An example of this is the creation of a new class of mass-scale tracing technology for contact tracing, a process of identifying exposed individuals who have come into contact with diagnosed individuals, during the COVID-19 crisis [17]. The hot pursuit of governments for a virus tracing solution led to compromises in privacy and data rights policies that were reflected in the smartphone application solutions. The result, so far, has been widespread concerns for civil liberties and resistance to comply with app downloading in some jurisdictions. This ad hoc, reactive response to the potential for digital enhancement to the process of contact tracing demonstrates the need for pre-determined design principles, and investigation of existing, complimentary approaches, that could save time, money and lives.

Therefore, there is a need to examine how information technology enabled systems are built, upon which critical systems and social functions they are constructed, and to explore how they can be designed or redesigned to mitigate fragility.

It is in this context that the S4 principles of crisis technology design framework exists; to explore design paradigms, algorithms and related problem spaces to create systems that can avoid the bootstrap problem, and reduce the long-term capability maintenance burden of society, or short-term deployment needs, with the overall purpose of supporting people during and post a crisis.

IV. Case study example: COVID-19 and contact-tracing apps

A prime example of a reactive, fractured response to technology development mid-crisis is seen in the contact-tracing application debate of COVID-19. More efficient contact-tracing solutions through digital means has tremendous potential to reduce the pace and spread of the disease and save lives.

Smartphone applications emerged early as an approach, based on contact-tracing in China and Singapore [18], [19]. Governments, organisations, research groups and informal coalitions are all working together across various social, legal, political and cultural contexts. However, the concern with contact-tracing through smartphones is location data, privacy, security and social network mapping [20]–[22]. If such a system is poorly designed or overseen, then they risk establishing a highly intrusive surveillance system, which has no place in a healthy democracy. This has sparked widespread debate, suspicion, mistrust and resistance of this approach to the public health crisis, and hampered crisis response efforts.

The question is not whether we need a contact tracing system (we clearly do), but whether a digital one can be created without the inherent risks of centralised digital infrastructure. It would be truly ironic if the attempt to improve a humanitarian disaster in the form of COVID-19 resulted in a new humanitarian disaster through the leak of highly sensitive population data which undermines the freedoms necessary for a democracy and human rights to exist.

Mid-crisis is the worst time to develop and deploy new complex systems. The COVID-19 contact-tracing case demonstrates the shortcomings of trying to develop a reactive solution, without a solid crisis technology design framework.

Groups of researchers have been working hard on this question throughout recent weeks and have discovered that it is quite possible to create such de-centralised systems, e.g., [23].

Their proposed approach is just as effective as the centralised approach for the intended purpose; detecting people at risk of having contracted COVID-19. The only difference is that they are deliberately not able to be abused for other purposes. At the core principles of the problem being solved, an effective system will record every time that every person comes near another person.

The simplest approach is a small physical token that does the rolling data collection. Both systems have phones listening for "beacon" transmitted by other phones. The difference is that the privacy respecting systems then leave the data sit on your phone, never uploading it to a centralised server. Instead, if you get a positive test for COVID-19, together with your doctor, you would upload the list of random beacons that your phone has transmitted during the time when you are believed to have been contagious. This is the only data that ends up in a central database. Other phones download this list of "probably infected beacons" and search through the beacons that they have captured, looking for any matches. An algorithm built into the app then alerts the user if they are at risk of having contracted COVID-19.

It has some privacy issues, but it has the advantage that as a discrete object you can choose to have it with you or not (or to turn it off), and once you have handed it in, it doesn't track you anymore. This can be achieved simply, securely and systemically through non-smartphone devices, whose data and functionality do not need to survive post-crisis.

The keys point is that if you are not infected, your data never leaves your phone, dramatically reducing the amount of data in the central database, and that the data that does end up there is contains no information about which phones were near one another. In short, it cannot be used to construct the social graph of a proportion of an entire nation, unlike the government and multilateral agencies centralised, proposed approach.

This S4 style approach is under active research in relation to COVID-19 [24]–[26].

We understand that governments are keen to get tracking apps running quickly, that will be trusted by the public. In that case they should concentrate their effort on setting information requirements and design specs to join, support and accelerate these expert, industry led international privacy and human rights preserving efforts. Otherwise, the very real risk is faced that people will refuse to run the application and the disease will continue to spread.

V. What should be done about it?

A. Technology System Design Principles

Resilience is a key concept, underpinning the design of digital tool and systems. Resilience provides a concept and solid base when preparing for and responding to disasters and emergency events. Seven characteristics of resilience are positive outcomes, threats and events, being prepared, commitment to survive, adaptability, gaining experience and coordinated response [27]. The concept of resilience emphasises planning for inevitable emergencies, disasters and catastrophes, of which digital preparedness is a crucial element. Resilience is an important guiding principle for numerous aid and development organisations in disaster preparedness [28], [29].

Taleb takes the concept of resilience a step further by proposing the notion of anti-fragility; that systems should not just withstand or learn from crisis, but they should improve under stress [30]. This is the goal of the S4 framework.

Robust approaches to technical design principles are available. For example, the Internet Engineering Task Force publishes internet architectural principles, to help guide engineering processes in a complex and changing environment that may be useful to those designing new protocols or evaluating such designs. These offer some solid technical design considerations, including heterogeneity, scalability, performance, cost, functionality, modularity, simplicity and avoiding circular dependencies, where two or more modules rely on each other in order to function [31].

Within these guidelines, the success of a protocol can be defined according to scale and purpose. According to these metrics, a "successful" protocol is one that is used for its original purpose and at the originally intended scale, or exceeds its original intended purpose or scale, such as HTTP [32].

B. Human - Technology Design Principles

A prominent technology design approach is 'human centered design'. Human centered design involves a process of interviewing and observing people as 'users' of technology, to produce solutions. It has been widely popularised in technology consulting for agile, iterative solution prototyping. This approach, however, is fraught with limitations. The danger, according to Norman, is that listening to users in relation to tasks does guarantee good products but does not lead to great design [33]. Therefore, human centered design as a standalone approach does not serve to create systems for communities, and is especially ill-suited for critical, crisis response digital tools.

Norman proposes an evolution to the more rigorous research-based approach of mapping the activities of users in an action-centered design model [33].

C. Socio-technical Design Principles

A viable approach to technology design principles which considered both technical factors and holistic, human user communities is socio-technical design.

Socio-technical design recognises the interaction between people and technology, or complex systems and human behaviour [34]. Computer system designers alone, (or government, or organisations, or UX designers), are not trained to count human cognitive and behavioural traits as factors in digital system requirements, with few exceptions leading to security and system vulnerabilities [35].

This understanding is at the core of the S4 principles; to explore design paradigms, algorithms and related problem spaces to create systems that reduce the long-term capability maintenance burden of society, or short-term deployment needs and support communities in crisis.

VI. Design principles for crisis technology

The S4 program is named for the four properties that are vital for the long-term sustainability of critical capabilities, and therefore the strengthening of societies against existing and emergent shocks and threats. These four properties do not stand in isolation from one another, but rather support and complement one another.

It is expected that this set of properties and justifications will continue to evolve as the program is carried out as more is learnt about creating such systems, as well as respective strengths and weaknesses. Thus, the following provide a putative starting point which will be used to seed initial activity and will be informed by the results of the ongoing, collaborative research and industry engagement.

A. S1: Simple

Complexity makes every stage of capability delivery harder: design, construction, maintenance, troubleshooting and integration with other systems. Complexity also greatly increases the volume of knowledge that must be passed from generation to generation for long-term capability maintenance and has the effect of reducing the available number of people who are competent to install, maintain, repair or adapt a capability.

Growing complexity effectively makes security impossible to achieve. Once a system becomes more complex than a single highly-trained expert or small group of experts can comprehend, it ceases to be possible to efficiently reason about the security of a system. Once the complexity grows such that even a well-resourced team cannot fully comprehend a system, then security becomes impossible to achieve.

The best that is achievable in these circumstances is a heuristic approach to security, similar to that of the human immune system, where a combination of inherent and adaptive defensive mechanisms acts to increase the resistance to infection but are unable to prevent it. This is an untenable situation for critical digital infrastructure, because it is highly likely for a system and the capability it provides to be subverted, disabled or destroyed before any reaction can be mounted to stop the compromise.

This again has analogy to the human immune system and pandemic-capable pathogens such as Ebola or COVID-19: The problem is not that the body cannot eventually identify and react to an Ebola infection, but rather that Ebola is able to damage the body beyond repair before the body can eliminate the Ebola virus. Or alternatively in the case of COVID-19, that the complexity of the body's immune system does not result in an erroneous reaction that damages the body itself.

Systems must be simple enough to avoid these problems. Alternatively, they must have a simple mode of operation that can be activated in the face of attack, so that the core capability can be sustained, even while the attack is identified and eventually defeated, at which time full capability can be restored.

In practice, this means that systems should have no more functionality (i.e., complexity) than is required to deliver the required utility (i.e, benefit to society or user). It also requires the simplification of external interfaces, so that the attack surface can be reduced as much as possible. This requires a radical re-think of how software-embodying systems are conceived, designed, implemented and maintained.

There are also related sub-themes of transparency (ability to inspect failure to understand how to correct it), documentation (ability for maintenance capability to be sustained between generations) and user expectation management (so that simpler systems are acceptable).

An environment where the value of simplicity is taken seriously is one where users enjoy and value the reliability of systems, even where for some uncommon special cases the end-result may be acceptably sub-optimal.

An example is to compare the Apollo Moon landing program's navigation and fuel control system that was purposely designed to be very simple and reliable, with only 78KB of memory [36], at an acceptable fuel efficiency [37], [38], and was thus able to be rapidly developed and deployed. This stands in contrast to the Ariane 5 rocket that suffered failure due to undetected problems in its control and navigation systems [39], [40]. This is not to say that simple systems cannot also suffer catastrophic failures, but the chance of being able to detect the defects that lead to such failures is much higher, as is post-mortem analysis and improvement.

Simple systems also have the advantage of being potentially much cheaper to create, precisely because of their lack of complexity.

B. S2: Secure

As the geopolitical and criminal activity environment of cyber and cyber-physical systems in crisis continues to worsen, such as email hacking, phishing or geo-political attack, security continues to grow in importance [41], [42]. Therefore, security must become a first-order objective of system design. For critical infrastructure and systems, security should take priority over functionality.

Security requires simplicity to be achieved. As described under the theme of simplicity; if a system contains millions to billions of lines of code or transistors, it not rational to expect that even a large well-resourced team will be able to provide any guarantees of security. In contrast, a sufficiently simple system can be audited, verified and proven to be functioning correctly.

A key requirement for security in interconnected systems is that the interfaces between the systems much be simplified and hardened as much as possible. Simple protocols and data formats should be used unless truly unavoidable, as many security problems are directly related to faults in these systems. Innovative approaches to addressing such common security problems are required, such as automatic translation of protocols into hardware implementations that are much more strongly resistant to cyber-attack due to their lack of flexibility.

Consideration must also be given to supply chain security, as recently highlighted by Google's detection of malware being installed in the supply-chain stage of the development of smartphones [43].

C. S3: Survivable

Survivability is the ability of a system to survive a shock or attack. Weak survivability implies only that a capability can be rapidly restored after a shock or attack, while strong survivability requires that a capability continues to be available during a shock or attack, although perhaps at reduced functionality, service level or capacity. This also requires that systems be secure.

Survivable systems must be able to operate when surrounding systems fail. That is, they must be able to fall-back to a stand-alone mode of operation, even if the result is sub-optimal performance. They must also be able to fall-forward to full-service when conditions again allow it.

Survivable systems must also be able to survive generational informational transfer, so that the capability remains creatable, maintainable and adaptable after its creators are no longer available. Making systems simple and well-documented is the best way to achieve this goal, so that the volume of information required is as small as possible, and as accessible as possible.

D. S4: Systems of S4 Systems

Society consists of systems of systems. For this complex web of interdependent systems and the capabilities they provide to continue to support society, they must be able to interact in a manner that avoids the bootstrap problem.

This is directly supported by creating survivable systems that are able to provide a base level of the capabilities that are required by other systems. Similarly, systems should be designed so that they can bootstrap using only the base-level of fall-back service that the capabilities that they depend on are able to provide in a standalone mode. Further, systems should avoid all bootstrap dependencies where possible, so as to completely eliminate the bootstrap problem.

Systems of systems create meta-systems with emergent properties of their own, including new fragilities and weaknesses that are not inherent in their component systems. Measures should be taken to avoid this where possible, including through measures such as simplifying the dependency and communications graph among systems. At the micro-scale of creation of individual systems and capabilities, this should take the form of minimising the number of dependent systems.

VII. How S4 solutions can be supported before the crisis hits

Crises take many forms, and vary considerably in magnitude, and trigger differing types of societal responses. The COVID-19 crisis has, despite some missteps, elicited a much greater and more rapid societal response than many disasters before it. It has triggered nation-states to contemplate the international supply-chain interdependencies that support their critical infrastructure, as they witnessed their assumptions of continuity of supply being disrupted by the particular nature of the crisis.

The challenge is to harness the temporary increased awareness and willingness by society to respond, to cause the necessary changes in practice and approach to problem solving, technology generation and systems design and implementation.

In the words of Winston Churchill, we should "never waste a good crisis". However, society does frequently waste such opportunities, and there are many examples of the failure of industries or societies to do so. For example, the UK building industry in the coincidentally named "Never Waste a Good Crisis: A Review of Progress since Rethinking Construction and Thoughts for Our Future" demonstrated an inability and/or unwillingness to address the majority its systemic problems [44]. We see similar effects surrounding many crisis issues, and a general marginalisation of science-informed policy in the public sphere.

The net result is considerable and sustained indifference to the adoption of anti-fragile design principles on a societal scale. Thus, we see a continuing need to identify ways to find targeted opportunities where S4 design principles provide competitive advantage, or intrinsically result in products that are distinctive and attractive in their respective market places, before broad-scale adoption of such principles is likely.

It is the S4 design principles that have motivated a number of our, and others', initiatives in various ways, several of which we mention below.

A. The MEGAphone S4 Mobile Phone

One area where we see potential, is for the creation of a privacy-respecting and highly-resilient mobile phone like device [45]. Here the security comes from the simplicity of the system, and the simplicity also helps to greatly reduce the effort required to develop the system. Survivability is aided by including a multitude of communications modes, as well as planning to include the Serval Mesh peer-to-peer communications protocols, which are themselves also designed around the S4 principles to create a highly survivable communications network [46]–[50]. It also includes a high-performance solar panel and large battery to provide energy self-sufficiency, to avoid the problem of sustained power outages during crises and maintaining mobile phone charge in such circumstances [51], [52]. In short, the MEGAphone creates a mobile communications capability that is independent of communications or energy infrastructure, and that is simple enough for a determined user to maintain these capabilities long-term.

B. ETC Lali

A second example is the Emergency Telecommunications Cluster (ETC) Lali system, that reduces the cost and complexity of deploying tsunami and other hazard early warning systems, through reducing the system to small low-cost hardware that can be more easily manufactured locally, and deployed without requiring heavy machinery [53], [54].

C. Simmel COVID-19 tracker

An example of a third-party project that reflects the S4 principles is the Simmel COVID-19 tracking hardware system [24], that replaces complex mobile applications and untrustrable support infrastructure with a fully distributed and verifable-through-simplicity hardware device for logging proximity to other such devices.

D. Summary

As mentioned above, these products tend to be rather niche, and none of them are (yet) sufficiently complete for commercial deployment, thus it is difficult to appraise their likelihood of market success, to test this concept of targeting of niche S4-embodying products in the crisis communications sector. This is why response organisations and groups must work alongside existing industry experts.

VIII. Conclusion and Future Directions

This paper has outlined a number of the problems and opportunities related to the creation of technologies that can survive significant societal shocks, and remain useful to their owners over the long-term. There are both social and technological challenges that remain to be tested and overcome before the S4 approach can be adequately appraised as to its feasibility. The S4 principles themselves are also acknowledged to be somewhat fluid and subject to refinement until this can occur.

Our efforts continue to focus on realising at-least one S4-conforming system, which at the time of writing is the MEGAphone secure mobile communications device. We invite other practitioners to critique and consider the S4 design principles that we have elucidated above. We also invite crisis response organisations to engage, support and provide feedback and insights to achieve better digital responses that solve real-world problems.

References

[1] S. Hasan and G. Foliente, “Modeling infrastructure system interdependencies and socioeconomic impacts of failure in extreme events: emerging r&d challenges,” *Natural Hazards*, vol. 78, no. 3, pp. 2143–2168, 2015.

[2] D. I. Wolinsky, P. S. Juste, P. O. Boykin, and R. Figueiredo, “Addressing the p2p bootstrap problem for small overlay networks,” in *2010 IEEE Tenth International Conference on Peer-to-Peer Computing (P2P)*. IEEE, 2010, pp. 1–10.

[3] A. Ozment and S. E. Schechter, “Bootstrapping the adoption of internet security protocols.” in *WEIS*, 2006.

[4] O. Hanseth and K. Lyytinen, “Design theory for dynamic complexity in information infrastructures: the case of building internet,” *Journal of information technology*, vol. 25, no. 1, pp. 1–19, 2010.

[5] S. P. Ryan and C. Tucker, “Heterogeneity and the dynamics of technology adoption,” *Quantitative Marketing and Economics*, vol. 10, no. 1, pp. 63–109, 2012.

[6] P. Kocher, J. Horn, A. Fogh, D. Genkin, D. Gruss, W. Haas, M. Hamburg, M. Lipp, S. Mangard, T. Prescher *et al.*, “Spectre attacks: Exploiting speculative execution,” in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 1–19.

[7] J. Van Bulck, F. Piessens, and R. Strackx, “Nemesis: Studying microarchitectural timing leaks in rudimentary cpu interrupt logic,” in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 178–195.

[8] M. Löw, “Overview of meltdown and spectre patches and their impacts,” *Advanced Microkernel Operating Systems*, p. 53, 2018.

[9] O. Alhubaiti and E.-S. M. El-Alfy, “Impact of spectre/meltdown kernel patches on crypto-algorithms on windows platforms,” in *2019 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*. IEEE, 2019, pp. 1–6.

[10] P. S. Deb, “An analysis on effects after mitigating meltdown and spectre vulnerabilities,” 2018.

[11] L. Xing, X. Pan, R. Wang, K. Yuan, and X. Wang, “Upgrading your android, elevating my malware: Privilege escalation through mobile os updating,” in *2014 IEEE Symposium on Security and Privacy*. IEEE, 2014, pp. 393–408.

[12] N. Al-Falahy and O. Y. Alani, “Technologies for 5g networks: Challenges and opportunities,” *IT Professional*, vol. 19, no. 1, pp. 12–20, 2017.

[13] J. Salo and M. Liyanage, “Regulatory impact on 5g security and privacy,” *A Comprehensive Guide to 5G Security*, p. 399, 2018.

[14] J. Suomalainen, K. Ahola, M. Majanen, O. Mämmelä, and P. Ruuska, “Security awareness in software-defined multi-domain 5g networks,” *Future Internet*, vol. 10, no. 3, p. 27, 2018.

[15] “Preventing the collapse of civilisation,” 2016. [Online]. Available: https://www.youtube.com/watch?v=pW-SOdj4Kkk

[16] J. Lillard, “Fogbank: Lost knowledge regained,” pp. 20–21, 2009.

[17] V. Hart, D. Siddarth, B. Cantrell, L. Tretikov, P. Eckersley, J. Langford, S. Leibrand, S. Kakade, S. Latta, D. Lewis, S. Tessaro, and G. Weyl, “Outpacing the virus: Digital response to containing the spread of covid-19 while mitigating privacy risks,” 2020. [Online]. Available: https://ethics.harvard.edu/outpacing-virus

[18] J. Bell, D. Butler, C. Hicks, and J. Crowcroft, “Tracesecure: Towards privacy preserving contact tracing,” *arXiv preprint arXiv:2004.04059*, 2020.

[19] H. Cho, D. Ippolito, and Y. W. Yu, “Contact tracing mobile apps for covid-19: Privacy considerations and related trade-offs,” *arXiv preprint arXiv:2003.11511*, 2020.

[20] C. Kuhn, M. Beck, and T. Strufe, “Covid notions: Towards formal definitions–and documented understanding–of privacy goals and claimed protection in proximity-tracing services,” *arXiv preprint arXiv:2004.07723*, 2020.

[21] Q. Tang, “Privacy-preserving contact tracing: current solutions and open questions,” *arXiv preprint arXiv:2004.06818*, 2020.

[22] R. Canetti, A. Trachtenberg, and M. Varia, “Private colocation discovery: Taming the coronavirus while preserving privacy,” *arXiv preprint arXiv:2003.13670*, 2020.

[23] E. Bugnion, S. Capkun, J. Larus, K. Paterson, M. Payer, B. Preneel, and N. Smart, “Decentralized privacy-preserving proximity tracing,” 2020. [Online]. Available: https://github.com/DP-3T/documents

[24] B. Huang, "Simmel wearable covid19 tracker," 2020. [Online]. Available: https://github.com/simmel-project/frontpage

[25] M. Daalder, "Coronavirus: New zealand considering $100m contact tracing covidcard," 2020. [Online]. Available: https://www.stuff.co.nz/national/health/coronavirus/121083996/coronavirus-new-zealand-considering-100m-contact-tracing-covidcard

[26] H. Cho, D. Ippolito, and Y. W. Yu, "Contact tracing mobile apps for covid-19: Privacy considerations and related trade-offs," 2020. [Online]. Available: https://arxiv.org/abs/2003.11511

[27] A. McAslan, "The concept of resilience: Understanding its origins, meaning and utility." [Online]. Available: https://www.flinders.edu.au/content/dam/documents/research/torrens-resilience-institute/resilience-origins-and-utility.pdf

[28] I. F. of Red Cross and R. C. Societies, "Regional and international disaster response tools and systems," 2020. [Online]. Available: https://www.ifrc.org/what-we-do/disaster-management/responding/disaster-response-system/dr-tools-and-systems/

[29] U. N. O. for Disaster Risk Reduction, "Disaster resilience scorecard for cities," 2017. [Online]. Available: https://www.unisdr.org/campaign/resilientcities/assets/toolkit/Scorecard/UNDRR_Disaster\%20resilience\%20\%20scorecard\%20for\%20cities_Preliminary_English.pdf

[30]

[31] B. Carpenter, "Architectural Principles of the Internet," IETF, RFC 1958, Jun. 1996. [Online]. Available: http://tools.ietf.org/rfc/rfc1958.txt

[32] D. Thaler and B. Aboba, "What Makes for a Successful Protocol?" IETF, RFC 5218, Jul. 2008. [Online]. Available: http://tools.ietf.org/rfc/rfc5218.txt

[33] D. A. Norman, "Human-centered design considered harmful," *interactions*, vol. 12, no. 4, pp. 14–19, 2005.

[34] S. Patel, "Socioanalytic methods: Discovering the hidden in organizations and social systems," 2014.

[35] A. Ferreira, J.-L. Huynen, V. Koenig, and G. Lenzini, "A conceptual framework to study socio-technical security," in *International Conference on Human Aspects of Information Security, Privacy, and Trust*. Springer, 2014, pp. 318–329.

[36] E. C. Hall, *Journey to the moon: the history of the Apollo guidance computer*. Aiaa, 1996.

[37] F. H. Martin and R. H. Battin, "Computer-controlled steering of the apollo spacecraft." *Journal of Spacecraft and Rockets*, vol. 5, no. 4, pp. 400–407, 1968.

[38] D. C. Cheatham, F. V. Bennett, and T. M. Branch, "Apollo lunar module landing strategy," in *Apollo Lunar Landing Symposium*, 1966, pp. 175–240.

[39] M. Dowson, "The ariane 5 software failure," *ACM SIGSOFT Software Engineering Notes*, vol. 22, no. 2, p. 84, 1997.

[40] E. J. Weyuker, "Testing component-based software: A cautionary tale," *IEEE software*, vol. 15, no. 5, pp. 54–59, 1998.

[41] L. Matthews, "500,000 hacked zoom accounts given away for free on the dark web," 2020. [Online]. Available: https://www.forbes.com/sites/leemathews/2020/04/13/500000-hacked-zoom-accounts-given-away-for-free-on-the-dark-web/#6907aabe58c5

[46] P. Gardner-Stephen, R. Challans, J. Lakeman, A. Bettison, D. Gardner-Stephen, and M. Lloyd, "The serval mesh: A platform for resilient communications in disaster & crisis," in *Global Humanitarian Technology Conference (GHTC), 2013 IEEE*. IEEE, 2013, pp. 162–166.

[42] C. Timberg and S. Mekhennet, "Email addresses and passwords allegedly from nih, who and gates foundation, are dumped online," 2020. [Online]. Available: https://www.seattletimes.com/nation-world/email-addresses-and-passwords-allegedly-from-nih-who-and-gates-foundation-are-dumped-online/

[43] A.-D. Schmidt, H.-G. Schmidt, L. Batyuk, J. H. Clausen, S. A. Camtepe, S. Albayrak, and C. Yildizli, "Smartphone malware evolution revisited: Android next target?" in *2009 4th International conference on malicious and unwanted software (MALWARE)*. IEEE, 2009, pp. 1–7.

[44] A. Wolstenholme, "Never waste a good crisis: A review of progress since rethinking constructionand thoughts for our future," 2014. [Online]. Available: https://constructingexcellence.org.uk/wp-content/uploads/2014/12/Wolstenholme_Report_Oct_2009.pdf

[45] P. Gardner-Stephen, A. Wallace, L. Moss, L. Lagadec, and M. Lloyd, "Designing a combined personal communicator and data entry terminal for disaster relief & remote operations," in *2019 IEEE Global Humanitarian Technology Conference (GHTC)*. IEEE, 2019, pp. 1–8.

[47] P. Gardner-Stephen, S. Farouque, M. Lloyd, A. Bate, and A. Cullen, "Piloting the serval mesh and serval mesh extender 2.0 in vanuatu: Preliminary results," in *Global Humanitarian Technology Conference (GHTC), 2017 IEEE*. IEEE, 2017, pp. 1–10.

[48] P. Gardner-Stephen, R. Challans, J. Lakeman, A. Bettison, P. Lieser, R. Steinmetz, F. Alvarez, and M. Lloyd, "Productizing humanitarian telecommunications research: A case study of the serval mesh extender," in *2017 IEEE Global Humanitarian Technology Conference (GHTC)*. IEEE, 2017, pp. 1–10.

[49] L. Baumgärtner, P. Gardner-Stephen, P. Graubner, J. Lakeman, J. Höchst, P. Lampe, N. Schmidt, S. Schulz, A. Sterz, and B. Freisleben, "An experimental evaluation of delay-tolerant networking with serval," in *Global Humanitarian Technology Conference (GHTC), 2016*. IEEE, 2016, pp. 70–79.

[50] J. Lakeman, M. Lloyd, R. Challans, A. Wallace, P. Gardner-Stephen, M. Stute, and M. Hollick, "A practical and secure social media facility for internet-deprived populations," in *2017 IEEE Global Humanitarian Technology Conference (GHTC)*. IEEE, 2017, pp. 1–9.

[51] W. Kongsiriwattana and P. Gardner-Stephen, "Smart-phone battery-life short-fall in disaster response: Quantifying the gap," in *Global Humanitarian Technology Conference (GHTC), 2016*. IEEE, 2016, pp. 220–225.

[52] ——, "The exploration of alternative phone charging strategies for disaster or emergency situations," in *Global Humanitarian Technology Conference (GHTC), 2016*. IEEE, 2016, pp. 233–240.

[53] G. Al-Nuaimi, J. Lakeman, P. Gardner-Stephen, and M. Lloyd, "Demonstrating a low-cost and zero-recurrent-cost hybrid mesh & satellite based early warning system," in *2018 IEEE Global Humanitarian Technology Conference (GHTC)*. IEEE, 2018, pp. 1–8.

[54] P. Gardner-Stephen, A. Wallace, K. Hawtin, G. Al-Nuaimi, A. Tran, T. Le Mozo, and M. Lloyd, "Reducing cost while increasing the resilience & effectiveness of tsunami early warning systems," in *2019 IEEE Global Humanitarian Technology Conference (GHTC)*. IEEE, 2019, pp. 1–8.