Smart Cyber-Physical Systems: Beyond Usable Security to Security Ergonomics by Design

Barnaby Craggs
Security Lancaster
Lancaster University, UK
Email: b.craggs@lancaster.ac.uk

Awais Rashid
Security Lancaster
Lancaster University, UK
Email: a.rashid@lancaster.ac.uk

***Abstract*—Securing cyber-physical systems is hard. They are complex infrastructures comprising multiple technological artefacts, designers, operators and users. Existing research has established the security challenges in such systems as well as the role of usable security to support humans in effective security decisions and actions. In this paper we focus on *smart* cyber-physical systems, such as those based on the Internet of Things (IoT). Such smart systems aim to intelligently automate a variety of functions, with the goal of hiding that complexity from the user. Furthermore, the interactions of the user with such systems are more often implicit than explicit, for instance, a pedestrian with wearables walking through a smart city environment will most likely interact with the smart environment implicitly through a variety of inferred preferences based on previously provided or automatically collected data. The key question that we explore is that of empowering software engineers to pragmatically take into account how users make informed security choices about their data and information in such a pervasive environment. We discuss a range of existing frameworks considering the impact of automation on user behaviours and argue for the need of a shift—*from usability to security ergonomics* as a key requirement when designing and implementing security features in smart cyber-physical environments. Of course, the considerations apply more broadly than security but, in this paper, we focus only on security as a key concern.**

***Index Terms*—Security Ergonomics, Human Factors, Internet of Things, IoT, HFACS, Cyber-Physical Systems, CPS**

I. Introduction

For some, cyber-physical systems (CPS) are purely "*engineered systems that are built from, and depend upon, the seamless integration of computational and physical components*" [1]. The reality is that all cyber-physical systems also contain a human component – from their design and implementation through to deployment, usage, maintenance, evolution and decommissioning.

Within this complex socio-technical system all three components are not only potential points of weakness but may also attack (maliciously or inadvertently) any other component. However, each component also plays a vital role in the protection of the others. As an emergent CPS, the Internet of Things (IoT) is no different. However the sheer scale, velocity of adoption and pervasiveness of the IoT presents, combined with on-system resource limitations, fundamental challenges to software engineering and how best to ensure the safety and security of the IoT. Understanding which role each component (including the human) should fulfil and which responsibility it should take in security is critical.

Security work in CPS has often focussed upon technical advancement, the role of humans being viewed as little more than a hazard - one who through error or mistake (intentional or otherwise) is at fault or to blame for security problems.

Whilst this may be true to some extent – the human is an intrinsic component of the CPS – the culture of blame has led to a misconception that better security comes from better systems, with better smarter technology ideally removing the need for or use by humans. In other words, by eliminating those erroneous or mistaken behaviours for which the human condition is *responsible*, security risk can be more readily managed. If anything this is a self-fulfilling prophecy — the more effort placed into better smarter technology the more likely it is that, in the event of failure, the human is seen as in error and therefore further effort needing to be put into further technological improvement. Research in the safety domain has shown that, whilst technological advances can remove cogntive load from humans, and to a degree mitigate error, there comes a point where "*automation... simply shifts the error*" [2] – normally to the designer / developer. An example of smart-automation failing is the case of Air France Flight 447 where advances in auto-pilot software resulted in critical manual flight training and practice being deemed unnecessary, with fatal consequences [3].

Recent work [4] has analysed the role of latent design conditions in impacting security perceptions of operators in industrial control systems and highlighted the challenges posed by smart CPS, notably their emergent design arising from dynamic aggregation of a range of devices and services and the focus on automation that aims to "hide" complexity from the users. Whilst usability is considered a key non-functional requirement during software engineering and there is a body of research on usable security, in this paper we argue that the very properties of emergent design and automation pose key challenges with regards to security behaviours in smart CPS. It is part of the *human condition* to err [5], sometimes deliberately, but more often inadvertently. But being able to recognise and learn from those errors is fundamental in moving socio-technical systems, such as IoT, forward. However, as smartness takes the human increasingly out of the loop, it is difficult to understand '*how did a security breach happen*' in relation to human behaviour and how that behaviour might have been fostered or indeed made inevitable by the design and / or environment of the system.

We, therefore, argue for a new type of non-functional requirement for smart CPS: *security ergonomics* that considers humans as integral to a smart CPS environment and, following the definition by the International Ergonomics Association definition, is concerned with "*the understanding of interactions among humans and other elements of a system, and applying theory, principles, data and methods to design in order to optimize human well-being and overall system performance*." Such a notion of security ergonomics moves far beyond understanding user requirements or simple human-machine interactions pertaining to security. Instead it brings insight into how humans behave (physically and psychologically) in relation to particular environments, products, or services.

The novel contributions of our work are a set of foundational design principles for security ergonomics which can be utilised to mitigate the impact of human error and security vulnerabilities in smart cyber-physical systems.

The rest of the paper is structured as follows. Firstly background and related work introducing our conceptual model and usable security works. Secondly we briefly detail two extensions to this conceptual model. Thirdly we present rationale for our initial design principles and open these up for discussion.

II. Background and Related Work

A. Conceptual Model

Our conceptual model of a smart CPS environment is the SHEL model developed by Edwards in the early 1970s, which Hawkins [6] modified into the building block structure (see Figure 1a) widely used in human factors research today. SHEL is simply an acronym for the core components at play:

Software - the procedures, checklists, symbology and computer software.

Hardware - the physical thing(s) being interacted with.

Environment - the situation within which the L-H-S system must operate.

Liveware - the human(s) with the one at the centre of the model being the most critical.

The model also has a SCHELL variant which separates out organisational **C**ulture from the environment but essentially remains synonymous with Hawkins' blocks. The SHEL(L) model is widely used to explore the interfaces between humans (the liveware) and the other blocks in aviation where some mismatch in those interfaces may lead to a source of human error or system vulnerability.

B. Usable Security

Existing research in non-CPS settings, for instance, Adams' and Sasse's seminal paper [7], has highlighted the multiple roles that humans play in a secure system and how issues in the design of security controls lead to poor security practises on the part of the user. Building upon this work, West [8] looked to explain, from a psychological perspective, why users made poor security decisions with user interfaces. He found, principally, that users were unmotivated in making good security choices, often not feeling at risk and that security was a secondary task compared to real work. West concluded that in order for users to make better security choices they needed to be better motivated by rewarding pro-secure behaviour (the antithesis of blame which seeks to punish poor choices).

In 2011 the National Science and Technology Council (NTSC), when addressing the current state of security called for a more scientific approach to security research including "*sound methods for integrating humans in the system*" [9]. Three years later Manusco [10] acknowledged that whilst this more scientific human factors approach to addressing cyber-security issues had begun it had "*yet to scratch the surface.*"

Though research has looked at using smart CPS devices as a means to counter usability limitations of existing security feature designs, e.g., [11], the issues of security ergonomics in smart CPS environments have not been considered to date. Recent work by Frey *et al* [4] has discussed the role that system design plays in influencing operator perceptions during security incidents and highlighted potential challenges in the context of smart CPS. In this paper, we respond to the challenges highlighted by Frey *et al* and focus on security ergonomics as a key requirement for smart CPS.

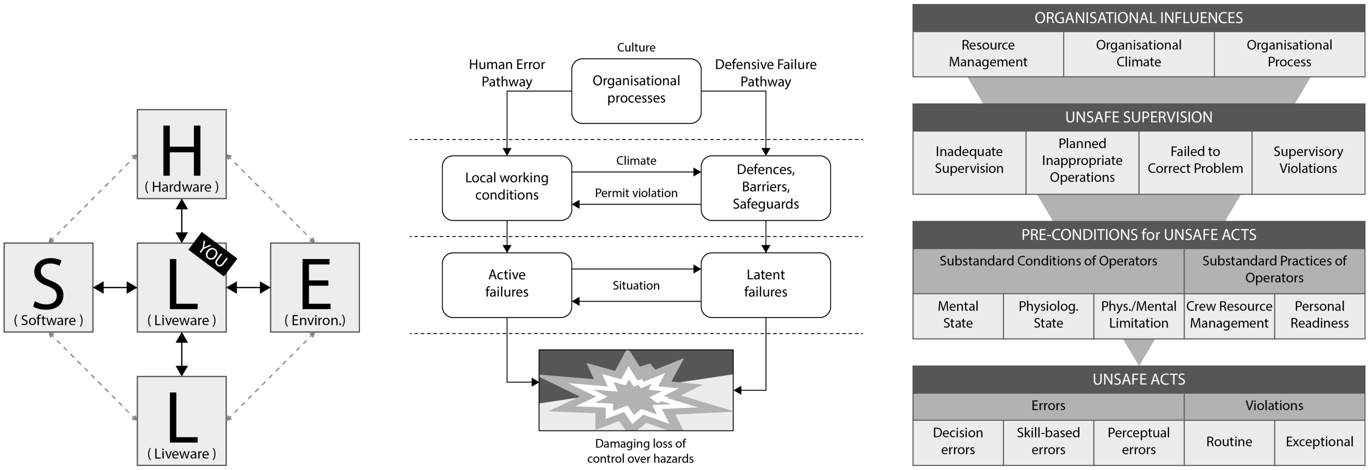
III. Existing Ergonomics Frameworks

To develop our security ergonomics concepts for smart CPS, we draw upon two existing frameworks in the safety and aviation domain, namely the Swiss Cheese model [12] and the Human Factors Analysis and Classification System (HFACS) [13].

A. Swiss Cheese

The human factors psychologist James Reason's works on industrial, aviation and clinical accident investigation [12] have developed what has become known as the *Swiss Cheese model (SC)*. The SC looks to human error (an unsafe act) as an active failure, and these may be both intentional or unintentional. System vulnerability is termed a latent failure. SC takes the view that truly bad events occur when both an active and a latent failure coincide. A mid-nineties variation of the SC (see Figure 1b) nicely illustrates the relationship between such active and latent failures (later versions exist but this illustration is more self-explanatory).

By way of an example, we have seen a number of malevolent secondments of consumer grade IoT devices into the Mirai botnets [14], [15] during 2016. A great many of the bots were gained through a confluence of user error (active failures) in not changing default passwords and security issues with the devices themselves (latent failures), such as providing no method for retrospectively applying security patches or updating firmware [16], hard-coding administration user-names and passwords into the firmware of devices and using the same administration credentials for both web and SSH/Telnet



(a) SHEL(L) based on Hawkins' model

(b) Reason's Swiss Cheese model (1990's variation)

(c) HFACS (Aviation) Classification System

Fig. 1: Models for Human Factors

access [17]. From the user's perspective they were utterly uninformed about these latent failures in the devices - a situational level alignment in the SC model. Moreover, it seems likely that the promise of a webcam which enabled the owner to watch, for example, a pet munching on biscuits probably over-rode any awareness that the device was effectively broadcasting their home to anyone with those default credentials and that had they been aware their purchase and erroneous installation might have been more considered.

B. Human Factors Analysis and Classification System

Borrowing heavily from Reason's SC, the Human Factors Analysis and Classification System (HFACS) [13] developed by Shappel & Weigmann for use in aviation is a root-cause analysis that can identify latent (and active) factors that can contribute to incidents. Critically, and congruent with Reason, HFACS specifies "*that in order for an incident to occur, failures in defences at all levels of the system must line up,*" [18]. HFACS describes four levels of failure: i) Unsafe acts (on the part of the human or liveware), ii) Preconditions for those unsafe acts (the direct factors impacting upon the human), iii) Unsafe supervision, and iv) Organisational influence (see Figure 1c). Whilst HFACS is not without its critics — these primarily being around the validity of statistical techniques and inter-coder reliability — it has been successfully modified and implemented beyond aviation for the design of robust and resilient healthcare [19], [20], rail transportation [18] and mining [21] practices.

IV. Design Principles for Security Ergonomics in Smart Cyber-Physical Systems

Human factors has evolved from conceptual models (SHEL) into actionable methodology (HFACS) for systems design to the point where, certainly in aviation, it is no longer a '*thing*' that is done but an embedded part of the system lifecycle [2]. A position we believe software engineering should aspire to.

When looking at the relationship between '*Privacy by Design*' (PbD) and the concept of '*privacy enhancing technologies*' (PETs) incepted some ten years prior, European Data Protection Supervisor Peter Hustinx links an increasing number of data breaches to being a structural problem not resolved by those PETs and argues that this presents an opportunity for further PbD focus [22]. Therefore, whilst we agree with Proctor and Chen [23] that human factors (as ergonomics) should be contributing to the science of cyber-security we firmly believe that this contribution would be of greater impact were those ergonomics - as a set of clear fundamental ideas - codified into simple design principles that could be adopted and utilised by anyone developing smart CPS.

As with '*security by design*' and '*privacy by design*' before, such principles look to embed those fundamentals within the very fabric of the development life-cycle rather than bolting human factor aware security on as an afterthought as is so often the case with CPS. We would also recommend that '*security ergonomics by design*' should form one of the basic principles of '*security by design*' efforts, such as the UK National Cyber Security Centre (NCSC) Security Design Principle for Digital Services [24].

To this we propose that a set of design principles for security ergonomics within smart CPS be developed collaboratively by the software engineering, human factors and security communities. We offer an initial five principles, based on Cavourkian's PbD work [25], to stimulate discussion:

1 - Proactive security ergonomic design, not reactive remedy. The approach anticipates / identifies and prevents active user error and latent system failures before they happen. It does not provide remedy for error / failure - it aims to prevent them before-the-fact. E.g., with reference to the Mirai botnet example already expressed, security ergonomics should identify the likelihood that users would not change default settings and make this visible to developers from the outset.

2 - Security ergonomics embedded into the design. The approach is embedded into the design and architecture of smart cyber-physical systems and not bolted on as a feature or add-on. Security ergonomics therefore becomes an essential integral property of the system without diminishing functionality.

3 - Design should encourage secure behaviours. By default secure non-erroneous user behaviour is encouraged and where possible enforced. E.g., users should be prompted to set secure passwords at setup, and where possible and appropriate this should be a modal activity, not allowing the user to proceed until an action is taken.

4 - Non-alignment by default. As human error is inevitable, security ergonomic design should prevent alignment of active error and latent failures. E.g., the alignment of users being able to not change default passwords at setup and those defaults being hard-coded into firmware, by default should not be allowed.

5 - External design validation. As any smart CPS is developed by human(s) — who by definition are subject to bias, assumption and indeed mistake — we must apply human factors to the development itself. Standard software development practices such as automated and unit testing can help with this validation although care must be taken that these themselves are not biased.

V. Conclusion

In this paper we reiterate the need for human factors to be utilised within cyber-security. We present argument as to how smart CPS, as complex socio-technical systems, pose an additional set of challenges to humans from design to usage, and set out how in other safety-concerned domains methodology such as HFACS has been used to good effect to both understand and inform system design.

Firstly, we propose that to untangle the complex socio-technical relationships which are inherent and vastly complicated in smart CPS, and that give rise to safety issues and security failings in IoT, HFACS offers a methodological route forward. Diller *et al*'s work in moving HFACS to health care [19] provides a potential method which would enable the software engineering community to re-purpose HFACS.

Secondly, to stimulate security ergonomics discussion, we present five initial design principles. We call upon the combined efforts of the human factors, software engineering and security communities to undertake the refinements needed to derive a clear set of simple, usable design principles for security ergonomics to help address the increasing tide of safety and security concerns within smart CPS and especially the Internet of Things.

Acknowledgment

This work has been funded by the UK EPSRC as part of the PETRAS IoT Research Hub - Cybersecurity of the Internet of Things grant no EP/N023234/1.

References

[1] SEsCPS'17. (2017) Context and goals. [Online]. Available: http://d3s.mff.cuni.cz/conferences/sescps2017/

[2] M. Bromiley. (2015) Human factors in clinical practice. [Online]. Available: https://vimeo.com/177542101

[3] T. Harford. (2016) Crash: how computers are setting us up for disaster. [Online]. Available: https://www.theguardian.com/technology/2016/oct/11/crash-how-computers-are-setting-us-up-disaster

[4] S. Frey, A. Rashid, A. Zanutto, J. Busby, and K. Follis, "On the role of latent design conditions in cyber-physical systems security," in *Proceedings of the 2nd International Workshop on Software Engineering for Smart Cyber-Physical Systems*. ACM, 2016, pp. 43–46.

[5] L. T. Kohn, J. M. Corrigan, M. S. Donaldson *et al.*, *To err is human: building a safer health system*. National Academies Press, 2000, vol. 6.

[6] F. H. Hawkins, *Human Factors in Flight*. Gower Technical Press, 1987.

[7] A. Adams and M. A. Sasse, "Users are not the enemy," *Commun. ACM*, vol. 42, no. 12, pp. 40–46, 1999.

[8] R. West, "The psychology of security," *Commun. ACM*, vol. 51, no. 4, pp. 34–40, 2008.

[9] National Science and Technology Council. (2011) Trustworthy cyberspace: Strategic plan for the federal cybersecurity research and development program. [Online]. Available: https://www.nitrd.gov/Publications/PublicationDetail.aspx?pubid=39

[10] V. F. Mancuso, "Human factors in cyber warfare ii," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 58, no. 1, pp. 415–418, 2014.

[11] F. Stajano, "Pico: No more passwords!" in *Security Protocols XIX - 19th International Workshop, Cambridge, UK, March 28-30, 2011, Revised Selected Papers*, 2011, pp. 49–81.

[12] J. Reason, *The Human Contribution: Unsafe Acts, Accidents and Heroic Recoveries*. Ashgate Publishing, 2008.

[13] S. A. Shappel and D. A. Wiegmann, "The human factors analysis and classification system–hfacs," US Federal Aviation Administration, Office of Aviation Medicine, Tech. Rep., 2000.

[14] B. Krebs. (2016) Ddos on dyn impacts twitter, spotify, reddit. [Online]. Available: https://krebsonsecurity.com/2016/10/ddos-on-dyn-impacts-twitter-spotify-reddit/

[15] B. Schneier. (2016) Security economics of the internet of things. [Online]. Available: https://www.schneier.com/blog/archives/2016/10/security_econom_1.html

[16] B. Schneier. (2014) The internet of things is wildly insecure—and often unpatchable. [Online]. Available: https://www.schneier.com/essays/archives/2014/01/the_internet_of_thin.html

[17] B. Krebs. (2016) Who makes the iot things under attack? [Online]. Available: https://krebsonsecurity.com/2016/10/who-makes-the-iot-things-under-attack/

[18] R. Madigan, D. Golightly, and R. Madders, "Application of human factors analysis and classification system (hfacs) to uk rail safety of the line incidents," *Accident Analysis & Prevention*, vol. 97, pp. 122–131, 2016.

[19] T. Diller, G. Helmrich, S. Dunning, S. Cox, A. Buchanan, and S. Shappell, "The human factors analysis classification system (hfacs) applied to health care," *American Journal of Medical Quality*, 2013.

[20] A. W. ElBardissi, D. A. Wiegmann, J. A. Dearani, R. C. Daly, and T. M. Sundt, "Application of the human factors analysis and classification system methodology to the cardiovascular surgery operating room," *The Annals of Thoracic Surgery*, vol. 83, no. 4, pp. 1412–1419, 2007.

[21] J. M. Patterson and S. A. Shappell, "Operator error and system deficiencies: analysis of 508 mining incidents and accidents from queensland, australia using hfacs," *Accident Analysis & Prevention*, vol. 42, no. 4, pp. 1379–1385, 2010.

[22] P. Hustinx, "Privacy by design: delivering the promises," *Identity in the Information Society*, vol. 3, no. 2, pp. 253–255, 2010.

[23] R. W. Proctor and J. Chen, "The role of human factors/ergonomics in the science of security: decision making and action selection in cyberspace," *Human factors*, vol. 57, no. 5, pp. 721–727, 2015.

[24] National Cyber Security Centre. (2016) Security design principles for digital services: Making services hard to compromise. [Online]. Available: https://www.ncsc.gov.uk/guidance/design-principles-making-services-hard-compromise

[25] A. Cavourkian. (2009) Privacy by design, the 7 foundational principles. [Online]. Available: https://www.ipc.on.ca/wp-content/uploads/Resources/7foundationalprinciples.pdf